# Document structure

Richard Power and Donia Scott*
University of Brighton

Nadjet Bouayad-Agha†
University Pompeu Fabra

*We argue the case for* abstract document structure *as a separate descriptive level in the analysis and generation of written texts. The purpose of this representation is to mediate between the message of a text (i.e., its discourse structure) and its physical presentation (i.e., its organisation into graphical constituents like sections, paragraphs, sentences, bulleted lists, figures, footnotes and so forth). Abstract document structure can be seen as an extension of Nunberg's 'text-grammar'; it is also closely related to 'logical' mark-up in languages like HTML and LaTeX. We show that by using this intermediate representation, several subtasks in language generation and language understanding can be defined more cleanly.*

## 1 Introduction

When language is written, it appears as a collection of words set out on one or more (actual or virtual) pages. In fact, much of what we tend to call 'text' has a strong graphical component (Schriver, 1997; Scott and Power, 2001). Not only are the words often accompanied by conventional graphics such as pictures or diagrams, but they themselves form graphical elements such as titles, headings, chapters, sections, captions, paragraphs, bulleted lists and the like.

The overlay of graphics on text is in many ways equivalent to the overlay of prosody on speech. Just as all speech has prosody (even if it is a monotone), so too do all texts have layout (even it is simple wrapped format, in a single face and font, and makes rudimentary use of white space). And just as prosody undoubtedly contributes to the meaning of utterances, so too does a text's graphical presentation contribute to its meaning. However, while there is a long tradition and rich linguistic framework for describing and representing speech prosody (e.g., Halliday, 1967; Chomsky and Halle, 1968; Crystal, 1969; Bolinger, 1972; Pierrehumbert, 1980; 't Hart, Collier, and Cohen, 1990; Ladd, 1996), the same is not true for text layout. Perhaps not surprisingly, therefore, few natural language understanding (NLU) systems use graphical presentational features to aid interpretation, and few natural language generation (NLG) systems attempt to render the output texts in a principled way.

Of course, since all texts have a graphical dimension, all NLG systems will, by definition, produce laid-out texts. In all but a few recent cases (the ICONOCLAST system (Power, 2000; Bouayad-Agha, Power, and Scott, 2000; Bouayad-Agha, Scott, and Power, 2001; Bouayad-Agha, 2001) and the DArt$_{bio}$ system (Bateman et al., 2001)), this is achieved by mapping directly from the underlying discourse structure (Arens and Hovy, 1990; DiMarco et al., 1995; Paris et al., 1995; Power and Cavallotto, 1996; Lavoie and Rambow, 1997; Mittal et al., 1998). In other cases, the text is mapped onto predetermined genre-specific layout patterns — for example, for verbalising mathematical

* Information Technology Research Institute, University of Brighton, Lewes Road, Brighton BN2 4GJ, UK. Email: {firstname.lastname@itri.bton.ac.uk}
† Department de Tecnologia, University Pompeu Fabra, Barcelona, Spain. Email: tt {Nadjet.Bouayad@tecm.upf.es}

proofs (Huang and Fiedler, 1997) or producing letters for customers (Coch, 1996). If we take, as most do, the level of discourse structure as representative of the underlying *message* of a text, such systems are subject to a fundamental limitation. Simply put, for each message there will be but one possible form of presentation.

As an illustration let us briefly consider the well-known consensus architecture for NLG systems proposed by Reiter (1994). This architecture, based on a survey of NLG systems from the 1980s and early 1990s, takes the form a 'pipeline' in which five modules are applied in sequence: content determination, sentence planning, surface generation, morphology, and formatting. Sentence planning maps 'conceptual structures into linguistic ones . . . grouping information into clauses and sentences' (Reiter, 1994, pg. 164), but formatting (specified, for example, by LaTeXmark-up) occurs only in the final formatting stage. In consequence, the organisation of material into paragraphs, bulleted lists, etc., is considered only after the wording has been fixed.

Graphical presentation, however, clearly interacts with wording. For example, the section of a message that, at the level of discourse, is composed of a LIST relation, will be expressed differently if, at the presentational level, it is mapped onto a vertical or horizontal list. Consider a simple example like the following, taken from a patient information leaflet (PIL):

(1) Are you taking any of the following:

- Anticoagulants?
- Lithium?
- Methotrexate?
- Any other medicines which your doctor does not know about?

*(Voltarol leaflet, Geigy; from APBI, 1997)*

If the very same content were presented instead as a horizontal list, we would expect to get something like:

(2) Are you taking anticoagulants, lithium, methotrexate, or any other medicines which your doctor does not know about?

Now all the information is packed into one sentence, with some missing and additional words, wildly different punctuation, and less generous use of upper-case letters.[1] Mapping directly from discourse structure to graphical presentation during generation therefore limits not only the choice of possible layout, but also the choice of possible wording.

There have been some recent attempts to develop NLG systems that generate *documents* rather than just *texts*. Instead of producing *text plans*, they produce *document plans*. Typically these are the text plans of old (i.e., structures of ordered content elements represented in terms of Rhetorical Structure Theory (Mann and Thompson, 1986; Mann and Thompson, 1987)), but extended to include pictures or diagrams as content elements, and with additional annotations for meta-level elements such as paragraph or sentence boundaries. Figure 1 shows the type of document plan proposed by Reiter and Dale (2000). Although this approach allows for a more reasoned presentational format, by conflating discourse and presentational features into one structure, the possible generated expressions of any given message are once again strongly limited relative to the set of all possible *valid* expressions.

We wish to argue the case for a separate descriptive level in the analysis and gener-

1 Indeed, it appears to be the case that the more graphical the presentation is, the greater the diffence in wording is likely to be over the unmarked case of plain text (Bouayad-Agha, Scott, and Power, 2000).

**Weather Summary for July 1996**

The month was slightly warmer than average with almost exactly the average rainfall, but rainfall for the year is still very depleted. Heavy rain fell on the 27th and 28th.

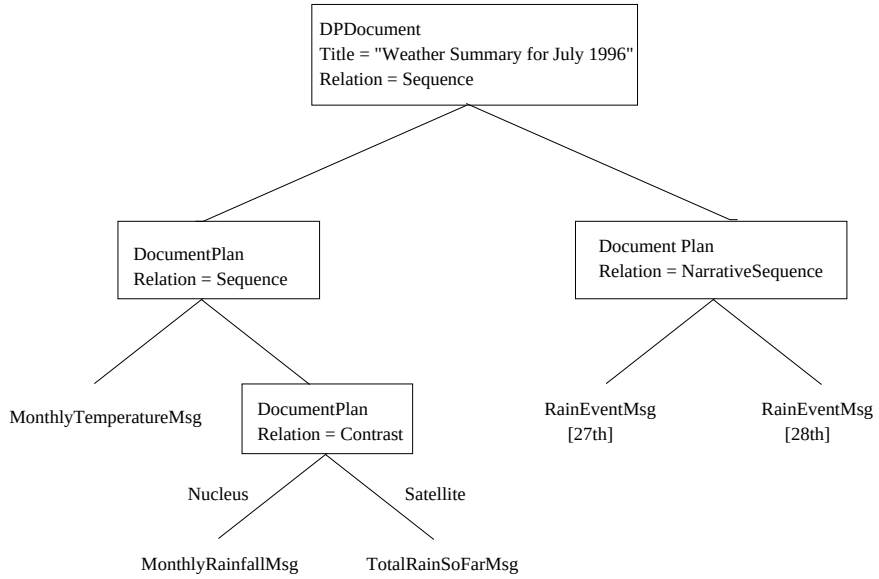



**Figure 1**
A Document Plan and associated text (Reiter and Dale, 2000)

ation of written texts, which we will call *document structure*. Informally, document structure describes the organisation of a document into graphical constituents like sections, paragraphs, sentences, bulleted lists, and figures; it also covers some features within sentences, including quotation and emphasis. Although document structure applies equally to NLU and NLG, we will focus our attention here on its role in the generation of appropriate presentations (both wording and layout) of texts. As we will try to show, texts (unless they are fairly simple) cannot be produced to a satisfactory standard unless document structure is specified earlier in the process than suggested in previous works (e.g., Reiter, 1994; Reiter and Dale, 2000), so that interactions with meaning and wording are taken into account.

The plan of the paper is as follows. In section 2 we explain more fully what we mean by 'document structure', acknowledging its origins in Nunberg's work (Nunberg, 1990) and text mark-up languages such as LaTeXand HTML. Section 3 discusses the relationship of document structure with rhetorical organisation and syntax. Section 4 presents the formal theory of document structure, and Section 5 shows how it is applied in the ICONOCLAST system. Finally, section 6 summarises the argument, and discusses some other approaches to the representation and generation of layout.

## 2 Defining Document Structure

### 2.1 Nunberg's text-grammar

Our point of departure has been the theory of text structure proposed by Nunberg (1990) in his book *The Linguistics of Punctuation*. This book introduces two crucial clarifications.

First, it distinguishes *text structure*,[2] which is realised by punctuation and layout, from *syntactic structure*. Secondly, it distinguishes *abstract* features of text structure from the *concrete* (or graphical) features by which they are expressed.

The distinction between text structure and syntax can best be explained by considering two interpretations of the word 'sentence'. In linguistics, 'sentence' is used mainly as a syntactic category, defined by phrase-structure rules such as $S \rightarrow NP + VP$. However, a sentence can also be viewed orthographically as portion of text starting with a capital letter and ending in a full stop; to distinguish this from the syntactic category, Nunberg calls it a 'text-sentence'. Sometimes the two categories of sentence coincide, but often they do not. Thus in the following passage:

(3) He entered the office. Disaster. The safe was open and the money had gone.

the first text-sentence is also a syntactic sentence, but the second is merely a noun, while the third comprises two syntactic sentences (or three if we count the whole as well as its parts). Nunberg argues that if we have two kinds of category, then we need *two kinds of grammar*: he calls them the 'lexical' grammar (we prefer 'syntactic') and the text-grammar. In addition to text-sentence, the text-categories include 'text-clause', 'paragraph', and 'section', and the text-grammar allows us to formulate constituent structure rules such as

$$S_t \rightarrow {C_t}^+$$

meaning that a text-sentence comprises one or more text-clauses.

In introducing the concepts 'text-sentence', 'text-clause', etc., it is convenient to explain them in terms of their realisation in punctuation and layout: thus a text-sentence starts with a capital letter and ends in a full stop; a text-clause ends in a semicolon; a paragraph begins on a new line with a tab. However, this is not strictly correct. In Nunberg's theory, these concepts represent *abstract* structural properties of the text which may be realised differently according to context or convention. In the case of 'paragraph' this distinction is obvious, since we are all familiar with several devices for expressing paragraph boundaries: instead of a new line with a tab, for example, an editor might prefer two new lines (or some other vertical space) with no tab. However, the abstract/concrete distinction also applies to the other text-categories. For example, the passage:

(4) The safe was open; the money had gone.

contains two text-clauses, but the second has no semicolon because its ending coincides with the closure of a larger unit, a text-sentence, which is marked by a full-stop. Similarly, the stop at the end of a text-sentence is often dropped when the sentence is an item in a vertical list, for instance in a sequence of instructions:

(5) To save the file:
1. Open the Save dialogue-box
2. Enter the filename
3. Click on the Save button

Thus text structure is *realised* by punctuation (and layout), but the two are not equivalent.

2 This should not be confused with the use of this term within the NLG literature to refer to the discourse structure of a text.

**2.2 Mark-up languages**

Nunberg's notion of text structure, or our wider notion of document structure, have an obvious connection to the mark-up languages (e.g., LaTeX, SGML) now in common use as a method for specifying layout in an ASCII source file. The common philosophy of these languages is that mark-up should abstract from the visual appearance of the document, using concepts like 'paragraph' which might be realised graphically in different ways, depending on a separate style definition.

This approach has several advantages, of which the most obvious is flexibility. An exact specification of the desired spatial layout can yield only one printed form of the document; by employing abstract categories, definitions using LaTeX or SGML can produce a range of printed forms, depending on which style file is used. Less obviously, the mark-up language can be tailored to the genre of the document, so that for example a poem may have a constituent marked 'stanza', while a letter may have one marked 'address'.

In practice, this separation of abstract structure from visual realisation is not carried through consistently; for reasons of convenience, authors sometimes prefer to have direct control over appearance. Thus LaTeX, for example, allows both the abstract tag `em`, meaning 'emphasis', and the visual tag `it`, meaning 'italic face'. Vertical separation is usually achieved through abstract tags like `section` and `itemize`, but may also be imposed directly using `vspace`. All these devices have counterparts in HTML: thus a typical reference guide to HTML (Ford and Dixon, 1996) explicitly distinguishes 'logical' tags such as `<EM>` from 'visual' tags such as `<I>`.

More subtly, the mark-up languages in common use do not attempt to cover structural units that are realised by *punctuation* rather than layout. Paragraphs may be marked (albeit implicitly in LaTeX), but lower units such as text-sentence and text-clause are not. No doubt there are good practical reasons for this policy, but some opportunities for stylistic variation are thereby lost. Consider, for example, a simple case of reported speech in example 6a. If this were marked up as a sentence containing a quotation, it could be punctuated differently — with the full-stop outside the closing quote (6b), perhaps, or using double-quotes (6c), or even using a dash with no quotation marks at all (6d):[3]

(6)

(a) She said 'Come up and see me sometime.'

(b) She said 'Come up and see me sometime'.

(c) She said "Come up and see me sometime."

(d) She said — Come up and see me sometime.

Thus although the mark-up languages provide some guidance towards a formal treatment of document structure, they often deviate, for practical reasons, from the philosophical ideal of separating abstract structure from visual presentation. On the one hand, some tags (e.g., italic face) are clearly visual, and should not be included in abstract document structure at all. On the other hand, some abstract categories (e.g., text-sentence, quoted speech) are omitted from the tag set, and thus cannot be realised in a range of different graphical styles.

Despite these compromises, mark-up languages are based on a key insight that is highly relevant to natural language analysis and generation: *layout can be matched to wording through the mediation of abstract document structure*. Consider the following three

3 As in James Joyce's *Ulysses*.

versions of a passage adapted from a patient information leaflet:

(7)

(a) **Elixir** is a white cream.
It is used in the treatment of cold sores.
It contains aliprosan. This is effective against a range of viral skin disorders.
It should be used **only** on your lips and face.

(b) **Elixir** is a white cream.

It is used in the treatment of cold sores.

It contains aliprosan. This is effective against a range of viral skin disorders.

It should be used ONLY on your lips and face.

(c) **Elixir** is a white cream. It is used in the treatment of cold sores. It contains aliprosan. This is effective against a range of viral skin disorders. N.B. **Elixir** should be used only on your lips and face.

Suppose that example 7(a) has been produced by author $A$, a novice in document design and passed to a more experienced designer $B$ for revision. The passage looks odd because it has four very short paragraphs, but short paragraphs are common in this genre and $B$ decides that the ugly appearance can be corrected simply by realising paragraphs by a vertical space with no tab. In addition, $B$ notices that bold face has been used for two different purposes: highlighting the product name, and emphasizing 'only'; this ambiguity is removed by changing emphatic bold face to small caps. Both these revisions — shown in version 7(b) — concern realisation rather than abstract structure, and consequently they do not affect the validity of the wording.

For final checking the passage is then passed to a senior expert $C$, whose preferences are more traditional: in particular, $C$ dislikes short paragraphs and variations in typeface. Glaring at the waste of space in version 7(b), $C$ takes out the paragraph boundaries and removes the emphasis on 'only'. These are not merely changes in graphical realisation: they also affect the abstract document structure. In general, *such changes endanger the validity of the wording*. Reading through the new version, $C$ notices that the pronoun 'it' in the final sentence now seems to refer to aliprosan, not Elixir, so it has to be replaced by the product name. To reinstate the emphasis in this sentence, $C$ also inserts the expression 'N.B.'. These changes lead to version 7(c).

In summary, abstract document structure interacts with wording, while visual realisation does not. This principle explains why abstract mark-up is useful; it also shows where the boundary should be drawn. By applying this principle, we might discover that a category previously treated as visual should be reclassified as abstract. If, for example, the change in the realisation of paragraph boundaries in examples like version 7(b) required a rewording of the text, we would have to extend our set of abstract document categories so that there were two types of paragraph instead of one.

## 3 Document Structure and Rhetorical Structure

### 3.1 Form and meaning

Logically there are four possible relationships between document structure (DS) and rhetorical structure (RS): either DS is part of RS, or RS is part of DS, or they partially overlap, or they are distinct. Our view is that they should be distinct, just as syntax is distinct from semantics. Document structure, like syntax, describes the form of a (mainly) linguistic product. Rhetorical structure, like semantics, describes meaning, in-

terpreting 'meaning' in a broad sense that includes pragmatic features.

As an example, suppose that $approve(fda, elixir)$ is a semantic formula meaning that the Food and Drug Administration (FDA) approves the medicine Elixir. This semantic formula can be *realised* (in English) by a range of syntactic forms, including:

[S [NP [DET the] [N FDA]] [VP [V approves] [NP Elixir]]]

which would yield 'the FDA approves Elixir'; alternative syntactic forms could be obtained by replacing descriptions by pronouns, or by putting the whole sentence into the passive (e.g., 'it is approved by the FDA'). Now, suppose that we add a second semantic formula $contain(elixir, gestodene)$, and suppose that the author knows that gestodene is a controversial ingredient. On this basis, a rhetorical relation of CONCESSION might be applied to the two formulae:

$$concession(contain(elixir, gestodene), approve(fda, elixir))$$

where the second argument of CONCESSION is the central one, and is supported by the first argument[4]. To realise this more complex message we may need a linguistic form that cannot be described only by a syntactic phrase-marker. In other words, we need to consider document structure as well as syntax. Ignoring possibilities for variations in the wording of the constituent propositions, and assuming that CONCESSION may be marked by the discourse connectives 'although' and 'however', we can choose among the following realisations of the whole message:

(8)

(a) Although Elixir contains gestodene, it is approved by the FDA.

(b) The FDA approves Elixir although it contains gestodene.

(c) Elixir contains gestodene; however, it is approved by the FDA.

(d) Elixir contains gestodene. However, it is approved by the FDA.

(e) Elixir contains gestodene.
However, it is approved by the FDA.

In versions (a) and (b), the rhetorical relationship is realised within a single syntactic sentence — although before adding punctuation we need to know that this syntactic sentence is also a text-sentence. In versions (c)–(e), the arguments of the CONCESSION relation are expressed in separate syntactic sentences, so that the relationship is realised by document structure as well as syntax. In each case, the units realising the arguments are coordinated in document structure, satellite precedes nucleus, and the discourse connective 'however' is placed within the nucleus.

Our claim, then, is that document structure combines with syntax in the realisation of the meaning of a document, and that rhetorical structure should be regarded as part of the meaning, not part of the document structure. However, as we will now show, this clear separation of meaning and form has not always been followed.

### 3.2 Rhetorical Structure Theory

Rhetorical Structure Theory (Mann and Thompson, 1986; Mann and Thompson, 1987) was developed as a method of analysing the rhetorical organisation of texts. Formally,

4 These are termed 'nucleus' and 'satellite' in RST (Mann and Thompson, 1987).

the theory is remarkably simple. It proposes that a text can be analysed, by rhetorical function, into a set of nested spans, each span being represented by a node on an ordered tree. Each non-terminal node on this tree is labelled by a single term describing the relationship that holds among its constituents. These constituents may have equal importance, in which case the relation is said to be *multinuclear*, or one may be rhetorically subordinated to the other, in which case they are said to fulfill the roles of *satellite* and *nucleus*. Although this scheme is obviously intended as a first approximation, it has been widely adopted, not only by literary analysts but also by computational linguists, especially in the NLG community.

One aspect of RST — perhaps more presentational than substantial — has led, in our view, to confusion: both in the text of their article and the accompanying diagrams, the authors seem to assert that rhetorical relations hold between spans of text, rather than between the meanings of these texts. In other words, they treat rhetorical structure as part of document structure. Mann and Thompson assert this explicitly (1987, pg. 4):

> Relations are defined to hold between two non-overlapping text spans, here called the *nucleus* and the *satellite*, denoted by N and S.

where

> A *text span* is an uninterrupted linear sequence of text.

However, it is unclear whether their claim is intentional, or simply the result of an informal style of presentation.

Would it make any sense to treat rhetorical relations as holding literally between text spans? For clearly pragmatic relations, such as RESTATEMENT, this seems a possible position. A text span, after all, is an instrument of the writer, so it makes sense to make a statement about the function that the instrument is supposed to serve. For clearly semantic relations, such as NONVOLITIONAL-CAUSE, the position is harder to maintain. In a text like 'Mary was sad because she lost her doll', the causal relation plainly holds between two events, not between two spans of text.

Leaving aside intuitive plausibility, the crucial question, we think, is whether the argument in a document can be formalised without reference to the particular text spans by which it is realised. Could the same argument be realised by two English texts with a different structure, or by an English text and a French text? Could someone forget a text but remember its argument? If so, it must be possible to treat relations like CONCESSION and NONVOLITIONAL-CAUSE as holding between ideas rather than units of text. Rather than creating two kinds of rhetorical relation, it seems more parsimonious to treat the relation between ideas as primary, so that the argument of a document can be planned before the writer (or NLG system) has considered issues of wording or linear order.[5]

To illustrate this point, let us go back to example 8. Suppose that we want to show that versions (b) and (c) express the same argument.

(8 b) The FDA approves Elixir although it contains gestodene.
(8 c) Elixir contains gestodene; however, it is approved by the FDA.

Figure 2 shows RST annotations for these texts, on the assumption that rhetorical relations hold between text spans; note, incidentally, that since related spans must be consecutive, the discourse connectives 'although' and 'however' have to be included somewhat arbitrarily in one span or the other. These annotations fail to bring out that the two texts express the same argument, since at a textual level the spans in 8(b) and 8(c) are simply different, both in wording and position.

5 This proposal was first made by Scott and Souza (1990).

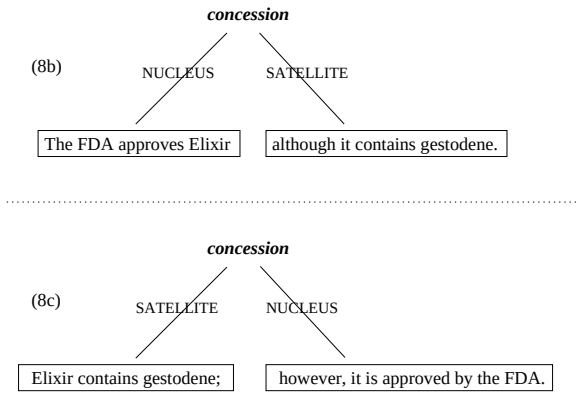



**Figure 2**
RST analysis

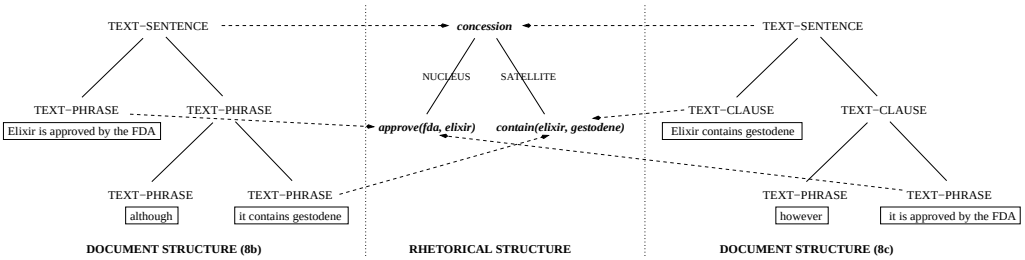



**Figure 3**
Document structure realises rhetorical structure

In figure 3, instead, different document structures are specified for versions (b) and (c), and linked to a common rhetorical structure; the dotted arrows express the relation 'realises', so that for example each text-sentence realises the whole rhetorical structure governed by the CONCESSION relation. Note that rhetorical structures are always unordered, while document structures are ordered trees. Thus the two unordered propositions in the rhetorical structure of figure 3 are realised in different orders in the two document structures.

This distinction between rhetorical structure and document structure accounts for some of the difficulties encountered when RST is applied as an analytic tool. The core of the problem is that when faced with a text to analyse, the analyst applies the principles of RST analysis to the text itself rather than to the message underlying the text. What the analyst really needs to do is, in some way, to 'get behind the text' to its constituent propositions and the rhetorical relations that hold between them (Scott and Paris, 1995). But instead, by applying relations to text-spans, he or she is heavily constrained by the evident document structure of the text, and the result is a rhetorical structure that is isomorphic to the document structure. However, as we have seen from the above example, the underlying rhetorical structure is not necessarily isomorphic to the document structure;[6] this means that the analysis obtained in this way is not necessarily an accurate representation of the *actual* discourse structure of the text. Consider for example, the

6 This is discussed in more detail in Bouayad-Agha, Power, and Scott (2000), Bouayad-Agha (2001) and in section 5.3.

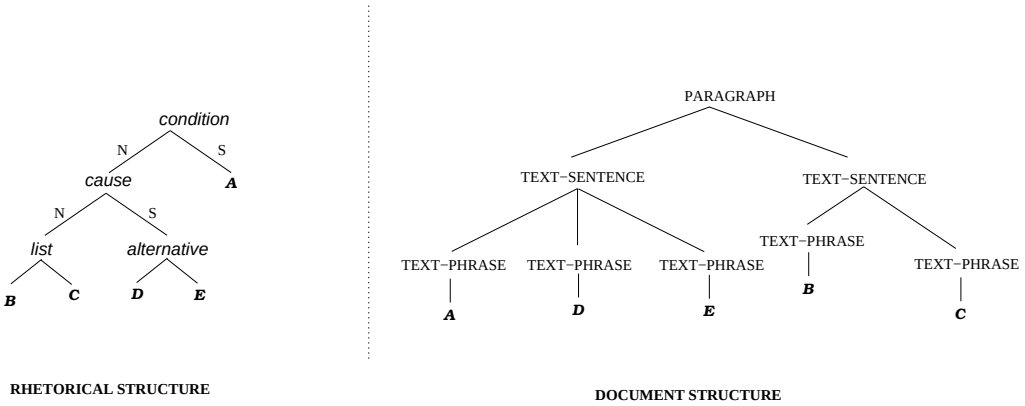



**Figure 4**
Document structure is not always isomorphic with rhetorical structure

following excerpt from a patient information leaflet:

(9) IF you find your condition gets worse during treatment$_A$ you may be allergic to the cream$_D$ or have a skin infection$_E$.
STOP USING THE CREAM$_B$ AND TELL YOUR DOCTOR AS SOON AS POSSIBLE$_C$
*(Betnovate leaflet, Glaxo; from APBI, 1997)*

Careful reading of the text, combined with world knowledge, suggests that the following logical condition holds between the propositional content of $A$ and the pair $B$ and $C$:

```
IF <condition of patient worsens during treatment>
   THEN <patient must stop taking cream>
    AND <patient must tell patient's doctor>
ENDIF
```

In other words, the rhetorical relation of CONDITION holds between $A$ as satellite and the complex of $B$ and $C$ (joined by a LIST relation) as nucleus. We learn additionally that the reason why the patient must carry out the imperative actions of $B$ and $C$ is that he or she may be either allergic to the cream ($D$) or have a skin infection ($E$). In other words, there is a causal relation between the complex of $B$ and $C$ (the effect) and complex of $D$ and $E$ (the cause). Representing all this in RST would yield the structure in figure 4.

Most people would probably agree that what is depicted in the RST structure shown in figure 4 captures the intended meaning of the text in the example, and that the text itself is of reasonable quality. However, a traditional RST analyis (i.e., of the text itself, as opposed to its underlying meaning) would not be able to produce the structure shown here. To explicate, let us go through what the typical RST analyst would do with this text.

The analyst would probably start by segmenting the text into elementary 'text spans' (i.e., clauses); this would lead to the same assigment of $A - E$ given above. The next step would focus on the first sentence: the discourse marker 'if' clearly suggests the CONDITION relation; similarly, the marker 'or' suggests the ALTERNATIVE relation. So far so good. However, the next step would be to find a relation that holds between the text spans in the sentence; if this cannot be done, then according to the tenets of RST, the text is not coherent. Following this rule, the analyst is likely to make $D$ and $E$ the components of the identified (multi-nuclear) ALTERNATIVE relation. Next they would attempt to assign the satellite and nucleus of the identified CONDITION relation; the

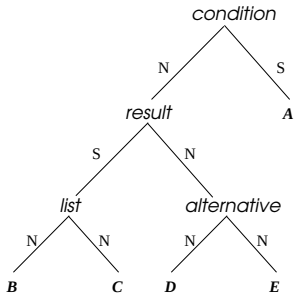



**Figure 5**
The analysis when RST is applied directly to the text

choice would be between $A$ and the complex of $D$ and $E$. Since the marker 'if' must attach to the satellite of CONDITION, the answer seems clear:

$$condition(alternative(\text{D,E}),\text{A})$$

(corresponding to the RST structure shown in figure 5). But it is also clearly wrong, since we know from our semantic analysis that what really holds is that shown in figure 4 :

$$condition(list(\text{B,C}),\text{A})$$

Indeed, even the layout of the text reinforces, through the use of capital letters, the strong relationship between $A$ and the pair $B$ and $C$.

In principle, an RST structure that is derived from the analysis of a given text should, when used as the input to an NLG system, produce the very same text and other *semantically equivalent* versions of it. By separating rhetorical structure from document structure, we can now provide a coherent framework for achieving this result. For instance, we can now produce not only the original text of example 9 (shown here with neutral layout):

(9 b) If you find your condition gets worse during treatment, you may be allergic to the cream or have a skin infection. Stop using the cream and tell your doctor as soon as possible.

but also the (in some contexts perhaps better) variant:

(9 c) If you find your condition gets worse during treatment, stop using the cream and tell your doctor as soon as possible; you may be allergic to the cream or have a skin infection.

Of course, NLG systems that ignore the level of document structure would still be able to produce the text in version 9 (b) from the RST structure in figure 5, but they would not be able to produce version 9 (c). Moreover, they could also end up producing the following, incorrect text:[7]

(9 c) # Stop using the cream and tell your doctor as soon as possible because you may be allergic to the cream or have a skin infection if your condition gets worse during treatment.

7 This text would result from systems that treat the leaves of a rhetorical/text plan as ordered.

A number of other researchers have identified cases where 'orthodox' RST analysis of a text is problematic (e.g., Moore and Pollack, 1992; Moser and Moore, 1996; Knott et al., 2001). For example, Knott *et. al.* (2001) report on texts from a corpus of museum labels that violate the RST principle of *continuous constituency* (i.e., adjacent units must be linked by a relation) but which are nonetheless coherent. These are cases where the satellite of a relation is not adjacent to its own nucleus in the text. In all the texts that they discuss, the 'dislocated' relation is ELABORATION, and they attribute the source of the problem to the relation itself: ELABORATION is not, they claim, a proper relation; it is a very weak relation which commands a different treatment from the other stronger ones.

While there may well be a strong case to be made for the 'demotion' of ELABORATION and for a special treatment of it in NLG systems, the phenomenon of dislocated satellites that Knott *et. al.* (2001) describe is not, in fact, confined to ELABORATION. Indeed, it corresponds precisely to the problem we have just seen with example 9, which involves not ELABORATION, but CONDITION. We have also reported elsewhere of other similar examples of non-isomorphic rhetorical and document structures (Bouayad-Agha, Power, and Scott, 2000). In all cases that we have seen, the principles of RST (e.g., compositionality, nuclearity, continuous constituency) appear to be violated only because they are being applied (in the orthodox manner) to the surface text (i.e., at the level of document structure) rather than more properly to the underlying propositional structure of the text (i.e., at the level of rhetorical structure).

But why would one want to produce a text whose document structure is not isomorphic with its rhetorical structure? One rather practical reason is that as a rhetorical structure becomes more complex, it becomes increasingly difficult for the writer to produce a text that satisfies both the demands of coherence (as defined by theories such as RST) and those of syntax. This happens quite frequently, for example, with conditionals: syntax dictates that expressions using the subordinating discourse marker 'if' must have the antecedent and consequent in the same sentence. For example:

(10) If you eat too many sweets, you will become ill.

but not

(11) # If you eat too many sweets. You will become ill.

As the consequent or its antecedent becomes more complex, the chances of satisfying the syntactic constraints are reduced. Here is a typical example from a PIL:

(12) **If you get any of the following:**

> Stomach pain, indigestion, heartburn or feeling sick for the first time.
> Any sign of bleeding in the stomach or intestine, for example, passing black stools.
> ...
> An unexpected change in the amount of urine produced and/or its appearance.

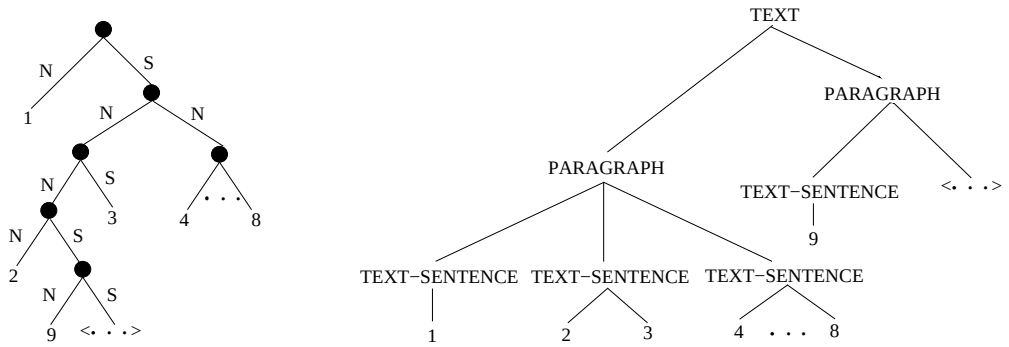
**STOP taking the tablets and tell your doctor.**

*(Voltarol leaflet, Geigy; from APBI, 1997)*

Here the conditional is expressed within one big sentence which itself contains several other sentences — indeed, paragraphs — each describing a set of symptoms organised around areas of the body. In writing this text, the author has clearly chosen to remain faithful to the rhetorical structure at the expense of syntax.

(1) In the women's quarters the business of running the household took place.
(2) Much of the furniture was made up of chests (3) arranged vertically in matching pairs. ... (4) Female guests were entertained in these rooms, (5) which often had beautifully carved wooden toilet boxes (6) with fold−away mirrors and sewing boxes, (7) and folding screens, (8) painted with birds and flowers.

(9) Chests were used for storage of clothes. ...




RHETORICAL STRUCTURE DOCUMENT STRUCTURE

**Figure 6**
Another example of non-isomorphic rhetorical and document structures

At other times, such as the example in figure 6 taken from Knott *et al.* (2001), it is the rhetorical structure that loses out. In this case, the author has decided that the content of the satellite associated with (9) warrants its own paragraph (perhaps for reasons to do with its size).

## 4 Formal theory of document structure

We will describe here the formal theory of document structure that we have developed as part of the ICONOCLAST system (Power, 2000; Bouayad-Agha, Power, and Scott, 2000; Bouayad-Agha, 2000; Bouayad-Agha, Scott, and Power, 2001), which generates multiple versions of the same message in different styles (i.e., with different wording and layout). In describing the theory, we will concentrate on units above the level of text-sentence; our treatment of the lower levels varies only slightly from Nunberg's theory, which is described in great detail in his book (Nunberg, 1990).

### 4.1 Basic hierarchy of document units

Informally it seems clear that units of document structure are ranked: sentences are grouped into paragraphs, paragraphs into sections, sections into chapters, and so forth. The hierarchy of categories differs from one document-type to another, but there is always *some* hierarchy;[8] there might for instance be subsubsections and subsections between paragraphs and sections. As a basis for discussion, let us assume a hierarchy of six levels, which we will number from 0 (assumed to be the lowest possible level of

8 Exceptionally, some elements — in particular, footnotes and pictures — will be 'floating'.

document structure) to 5[9]:

| | |
|---|---|
| 0 | text-phrase |
| 1 | text-clause |
| 2 | text-sentence |
| 3 | paragraph |
| 4 | section |
| 5 | chapter |

The fundamental organising principle of document structure is that a unit of a given level is composed of one or more units of the next level down. This observation could be expressed by a set of constituent structure rules, one for each level:

$chapter \rightarrow section^{+}$
$section \rightarrow paragraph^{+}$ (etc.)

Alternatively, we could generalise over this set of rules by introducing the symbol $L_N$ to denote a unit of level $N$ (so that $L_0$ means text-phrase, $L_1$ means text-clause, etc.). A single rule will now cover units of all levels:

$$L_N \rightarrow {L_{N-1}}^{+} \quad (N > 0)$$

Two consequences of this rule should be noted. First, it disallows document structures in which a unit contains a sub-unit of higher level — for instance, a text-sentence may not contain a paragraph. Secondly, it disallows document structures in which coordinated units have different levels. A section, for example, may not be formed by coordinating two paragraphs and a text-sentence, as in figure 7(a); the sentence should be the only constituent of a further paragraph unit, as in 7(b), so that when the abstract document structure is realised graphically, the sentence will be formatted with a paragraph break as well as with a capital letter and a full stop.

This deals with the basic organisation of text into hierarchical document units; however, a full description should take account of many other patterns including headings, bulleted lists, footnotes, and figures. We cannot deal with all these patterns here, so we focus on what is probably the most complex problem: the treatment of indented structures such as quotations, bulleted lists, and enumerated lists.

### 4.2 Indented document units

At first sight it might seem that indentation is a feature of graphical realisation rather than underlying structure — in other words, that it belongs to concrete rather than abstract document structure. We have several reasons for rejecting this view. First, it is at least suggestive that both LaTeX and HTML include tags for indented structures:

| **Pattern** | **HTML tag** | **LaTeX tag** |
|---|---|---|
| Quotation | `<QUOTE>` | `\begin{quote}` |
| Bulleted list | `<UL>` | `\begin{itemize}` |
| Enumerated list | `<OL>` | `\begin{enumerate}` |
| Description list | `<DL>` | `\begin{description}` |

Secondly, one can find examples of vertical lists that seem structurally equivalent to (say) a bulleted list, but are presented without item markers and without the use of *horizontal indentation*. Here is a case in point, taken from a patient information leaflet with formatting exactly preserved:

9 We use 'text-sentence' and 'text-clause' in Nunberg's sense, as units typically marked by a full-stop and a semi-colon. However, unlike Nunberg, we use 'text-phrase' for any constituent of a text-clause, whether it is marked by a comma or not.

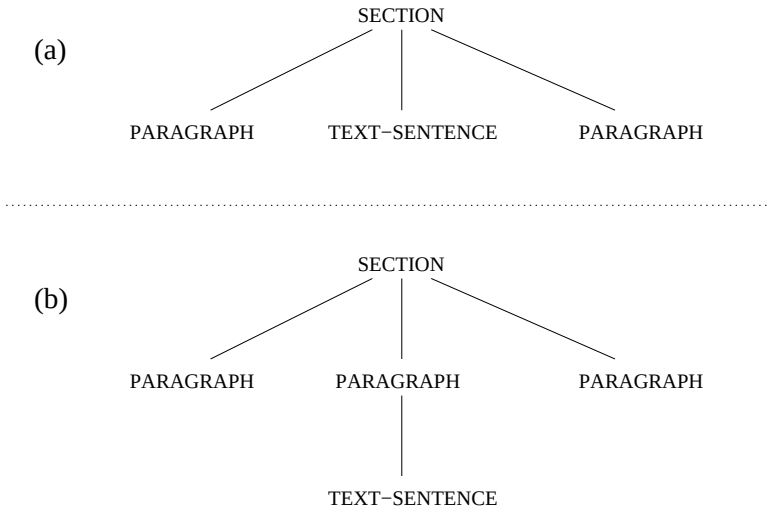



**Figure 7**
Ill-formed (a) and well-formed (b) document structures

(13) If you experience any of the following or any other unusual effects, tell your doctor:

Poor appetite or a slight sick feeling
Mild abdominal pains or fullness
Alterations in your sense of taste
Diarrhoea
Itching or rash
Pain in your muscles or joints

If you notice yellowing of the skin or eyes, tell your doctor straight away.
*(Lamisil, Sandoz; from APBI, 1997)*

Thirdly, and most important of all, indented structures may introduce apparent violations of the hierarchical ranking described in the last section: for instance, a sentence may contain a paragraph *provided that the paragraph is indented*. We have already seen this in the case of complex conditionals in example 12. However, the phenomenon is much more widespread:

(14) In rare cases the treatment can be prolonged for another week; however, this is risky since

- The side-effects are likely to get worse. Some patients have reported severe headache and nausea.
- Permanent damage to the liver might result.

(15) The opening of *Pride and Prejudice*

> It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife. However little known the feelings or views of such a man may be on his first entering a neighbourhood, this truth is so well fixed in the minds of the surrounding families, that he is considered as the rightful property of some one or other of their daughters.

is one of the most famous paragraphs in English literature.

In these examples too, note that the important issue is not graphical indentation but what we might call 'logical indentation'. Thus in the case of the quotation, the logical indentation of the quoted paragraph might be shown without graphical indentation — e.g., by using a distinctive font or character size:

(16) The opening of 'Pride and Prejudice'

> *It is a truth universally acknowledged, that a single man in possession of a good fortune, must be in want of a wife. However little known the feelings or views of such a man may be on his first entering a neighbourhood, this truth is so well fixed in the minds of the surrounding families, that he is considered as the rightful property of some one or other of their daughters.*

is one of the most famous paragraphs in English literature.

How can the formation rules for document structure be extended so as to accommodate these patterns? Our main proposal, implemented in the ICONOCLAST document planner, is that a document unit should be defined by (at least) two features: its level, and its indentation. The level is the usual ranking from $L_0$ (text-phrase) to $L_5$ (chapter) — or whatever hierarchy of units the author decides to employ; the indentation is a value in the range $I_0 \ldots I_{Max}$, where $I_0$ means that the unit is not indented at all, $I_1$ means it is indented one place, and so forth, with $I_{Max}$ representing the deepest embedding that the author is prepared to contemplate. The passages in examples 14, 15 and 16 can then be described formally as unindented text-sentences (i.e., units with the category $[L_2, I_0]$), which have among their constituents indented paragraphs (units with the category $[L_3, I_1]$). Such structures can be permitted if we change the basic rule of the last section so that a unit of indentation $I_N$ always outranks a unit of indentation $I_{N+1}$, no matter what their respective levels may be. Instead of one general rule we now need two: the first for unindented constituents, the second for indented ones.

Unindented constituents $\quad [L_N, I_M] \rightarrow [L_{N-1}, I_M]^+$

Indented constituents $\quad [L_A, I_M] \rightarrow [L_B, I_{M+1}]^+$

In the second rule, $L_A$ and $L_B$ are unrelated, so that $L_B$ could represent a higher level than $L_A$ (e.g., a paragraph inside a text-sentence). For most document-types one would presumably prefer to set an upper limit on the level of an indented constituent (e.g., an indented chapter seems ridiculous); this could be done for instance by adding the constraint $B < 4$.

Figures 8 and 9 give document structures for examples 14 and 15; the latter also describes example 16, which differs from 15 only in the graphical realisation of the indented paragraph.

One feature of these analyses might at first sight appear strange: they assign the minimal text level $L_0$ to the node representing the quotation or the bulleted list. How, one might ask, can a paragraph, or indeed a list of paragraphs, be regarded as a text-phrase? Should it not be a unit *higher* than the paragraph — perhaps some kind of section, or better, a new text category *vertical list*?

On first tackling indented structures, we followed precisely this reasoning, introducing *vertical list* as a text category. However, despite its initial plausibility, this decision has several irritating consequences. First, the concept of *vertical list* has no relationship to the text level hierarchy. Does it belong between text-sentence and paragraph, or perhaps between paragraph and section? Any placement seems arbitrary. Secondly, since the vertical list in a document structure like figure 8 is clearly coordinated with a text-phrase, such an analysis would violate the rule that coordinated constituents have the same level. Thirdly — and perhaps most persuasively of all — it turns out that in

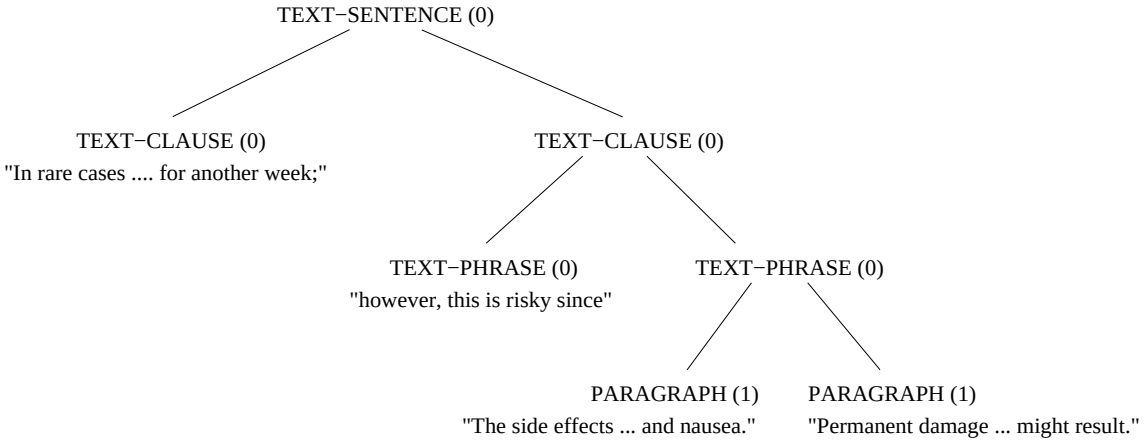



**Figure 8**
Indented document structure (Example 14)

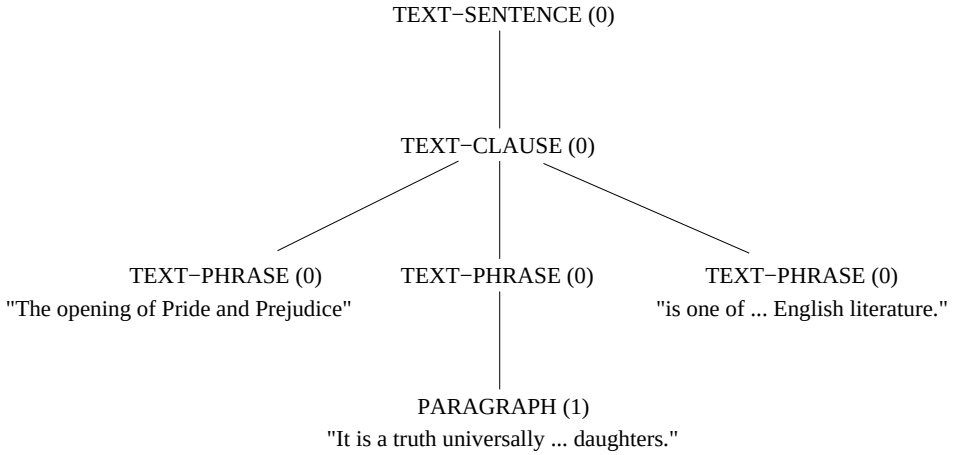



**Figure 9**
Indented document structure (Examples 15 and 16)

their interactions with discourse connectives, vertical lists *behave like text-phrases*, not like sentences, paragraphs, or higher units. Thus the vertical list in figure 8 is coordinated with the discourse connective 'since', which would not normally link constituents higher than text-phrases (we will discuss this constraint more fully in the next section)[10].

All these difficulties are removed by denying appearances and assigning the vertical-list node a text-level of $L_0$. With this treatment, no new arbitrary level in the hierarchy of units is needed, the basic rule of document structure holds (i.e., coordinated constituents have the same level), and the interaction of the unit with discourse connectives can be described by the same rules that hold for non-indented structures. Note also that the difference in indentation between the node and its daughters is sufficient to identify it

10 Note that a vertical list can also be coordinated with units larger than text-phrases, such as text-sentences or paragraphs. The formal rules allow this because a text-phrase can be the only constituent of a text-clause, which can in turn be the only constituent of a text-sentence, and so forth.

as a vertical list; any extra feature (such as a new value in the text-level hierarchy) would be redundant.

## 5 Methods of document structuring

### 5.1 Defining the task

The aim of *document structuring* within NLG is to create a document structure that satisfactorily realises a rhetorical structure. This can be achieved through a variety of architectures, as reflected in the RAGS framework.[11] For example, systems aimed at generating technical abstracts or captions for graphics would probably need to specify a 'one paragraph only' limit quite early on in the process; in such cases, document structuring would start before any rhetorical or syntactic decisions were made (e.g., the RAGS reimplementation of the Caption Generation System, Mellish et al. (2000)). Alternatively, systems that need to generate texts with rich layout might prefer to interleave document structuring with rhetorical structuring (e.g., Cahill et al. (2001)).

For simplicity of presentation, we will assume here an NLG architecture in which a preliminary module selects simple propositions from a knowledge base and organises them into some kind of argument; the document structurer distributes the various parts of this argument among units like paragraphs and text-sentences; a syntactic realiser takes over in order to determine the wording of the propositions; finally, a formatter decides how the abstract categories and features of document structure will be realised graphically (e.g., whether paragraphs will be marked by a tab on a new line, or by a vertical space). In the context of such an architecture, two issues are crucial: what input does the document structurer receive from the earlier planning module; and how does the output of the document structurer guide syntactic realisation?

In ICONOCLAST, our initial assumption — following Scott and Souza (1990) — was that the rhetorical structure would take the form of a graph in which terminal nodes represent simple propositions and non-terminal nodes represent rhetorical relationships. By organising all propositions into a single hierarchy, such an input simplifies the task of the document structuring module — indeed, it could be argued that part of the work has been done already. An alternative assumption is that the rhetorical structure takes the form of a set of assertions, each describing a rhetorical relationship between two propositions (Marcu, 1997). With this flat representation, the document structurer must take more responsibility for grouping the propositions. In this paper, we will focus on the first of these methods (i.e, starting from a hierarchical rhetorical input). The second method (starting from a flat rhetorical input) is described in Bouayad-Agha (2001).

Regarding the interface between document structuring and syntax, the difficult issue is where the boundary should be drawn. Above the level of text-clause, it seems clear that syntax plays little part; however, at the level in which text-phrases combine to form text-clauses, there is an interplay between the 'syntactic' grammar and the 'text' grammar, as Nunberg (1990) has shown. As an example, consider the simple rhetorical structure discussed earlier (example 8 and figure 2):

$$concession(contain(elixir, gestodene), approve(fda, elixir))$$

As mentioned earlier, this rhetorical-semantic input could be realised by a text-sentence comprising two text-clauses:

(8c) Elixir contains gestodene; however, it is approved by the FDA.

11 See *The* RAGS *Reference Manual*, http://www.itri.bton.ac.uk/projects/rags/RefMan/refman.ps.

In this case it seems clear where the boundary should lie: the document structurer decides that $approve(fda, elixir)$ should be expressed in the first text-clause, and that $contain(elixir, gestodene)$ should be expressed in the second, along with the connective 'however'. It might also impose constraints on the clauses that will realise these propositions, for instance by requiring declarative clauses rather than imperative or interrogative ones. The rest is left to the syntactic realiser, which must decide how the two propositions should be worded, and how the second clause should be combined with the connective. Several alternative versions might result, each having exactly the same document structure:

(8′) Elixir contains gestodene; the FDA, however, approves it.
(8″) Elixir contains gestodene; however, the FDA approves it.
(8‴) Gestodene is an ingredient of Elixir; Elixir is approved, however, by the FDA.

Suppose, however, that the document structurer decides to realise the two propositions within the same text-clause, as in the following version:

(8⁗) Elixir is approved by the FDA although it contains gestodene.

Where now does the boundary lie? Should the document structurer create a text-sentence containing a single text-clause, leave instructions that this text-clause should express the whole concession relationship, and then wash its hands of the affair, leaving everything else to the syntactic realiser? Or should it trespass into the domain of syntactic realisation by constructing a syntactic clause of the form $S_1\ although\ S_2$, with instructions that a sub-clause realising $approve(fda, elixir)$ should be inserted in position $S_1$, and a second sub-clause realising $contain(elixir, gestodene)$ in position $S_2$?

In ICONOCLAST we have found it more convenient to adopt the latter policy. The linguistic structure of a document is represented by a single tree in which nodes may be labelled both with document-structure features and with syntactic features. The document structurer's task is to build the upper part of this tree, extending from the root all the way down to the nodes that express simple propositions. In the upper reaches of the tree, nodes will be labelled with units like 'section' and 'paragraph', and syntactic features will be irrelevant; lower down, the document structure may have to assign some syntactic features when constructing compound clauses. To simplify, we assume that conjunctive adverbs like 'however' will always be placed at the beginning of a clause; this means that discourse connectives can be placed into the tree by the document structurer, so limiting the task of the syntactic realiser to the wording of the simple clauses that realise propositions.

Having clarified the document structurer's task, we posed the following question: given a well-formed rhetorical structure, together with a set of formation rules for document structure and a set of discourse connectives for realising rhetorical relations, can we enumerate all document structures that correctly realise the rhetorical structure — and further, can we evaluate these document structures by some metric so that we can choose the best? The generation of many alternative solutions was essential to the project, which focussed on the problem of controlling style in an NLG system: by varying a set of stylistic parameters, the user of the system can influence the evaluation metric that is applied to the set of potential solutions, and so influence the type of solution that will be preferred.

As mentioned above, we have explored two methods of enumerating and evaluating document structures, one (the focus of this paper) assuming a hierarchical input, the other (described in Bouayad-Agha (2001)) assuming a non-hierarchical input. The two methods have much in common: they share the same formation rules for document structure, the same discourse connectives, and (mainly) the same constraints on the cor-

rect realisation of rhetorical relationships. Before describing the hierarchical method in detail, it will be useful to review the constraints that they share; these constraints will be needed, in some form or other, in any system that performs document structuring.

**5.2 Constraints on realising rhetorical structure**

In order to realise a rhetorical structure $R(A_1, A_2)$ as a document structure, several decisions must be made:

- What should be the level (e.g., section, paragraph, text-sentence) of the document unit that realises the whole relationship $R(A_1, A_2)$?
- What should be the levels of the units realising the arguments $A_1$ and $A_2$?
- Should the units realising $A_1$ and $A_2$ be indented items, or should they have the same indentation as the unit realising $R(A_1, A_2)$?
- In what linear order should the units realising $A_1$ and $A_2$ occur?
- Should the rhetorical relation $R$ be expressed by a discourse connective, or left implicit?
- If a discourse connective is used, should it be linked to the span realising $A_1$ or the span realising $A_2$?

These decisions are closely related, as the following examples show:

- If $R(A_1, A_2)$ is realised by a text-sentence, then the arguments $A_1$ and $A_2$ cannot be realised by a higher unit such as a paragraph unless they are indented one place further. (This follows from the formation rules for document structure.)
- If $R$ is a nucleus-satellite relation rather than a multinuclear one, the arguments $A_1$ and $A_2$ should not be realised by indented items[12].
- If a subordinating conjunction like 'although' is used to express the relation $R$, the arguments $A_1$ and $A_2$ should be realised within the same syntactic clause; hence they should be text-phrases, rather than text-clauses or some higher unit. Moreover, 'although' should be attached to the clause expressing $A_1$ (assuming this is the satellite).

Constraints arising from the formation rules for document structure have already been covered; we will therefore focus here on constraints that concern the realisation of rhetorical relations.

Leaving aside quotations for the time being, our first suggestion is that indentation should be employed only for the arguments of a *multinuclear* relation. Vertical lists are typically used when the items play the same role in the discourse — for instance, they might be symptoms of a disease, or potential side-effects of a medicine. By definition, the arguments of a nucleus-satellite relation have different purposes; the nucleus makes the main point, while the satellite's role is supportive. Consider for example the following rhetorical structure:

12 This constraint is based on the intuition that the items in a vertical list should have parallel roles in the argument – a condition that clearly fails to hold for a nucleus-satellite relation.

$$justify(list(tested(elixir), approve(fda, elixir)), safe(elixir))$$

which gives two reasons why Elixir is safe to use. The LIST relation is multinuclear, while JUSTIFY is nucleus-satellite; according to our rule, then, the document structures in 17 and 18 should be avoided, while 19 and 20 should be acceptable:

(17)
- Elixir is safe to use
- because it has been carefully tested and is approved by the FDA.

(18)
- Elixir has been carefully tested and is approved by the FDA,
- therefore, it is safe to use.

(19) Elixir is safe to use because
- it has been carefully tested
- it is approved by the FDA

(20)
- Elixir has been carefully tested
- Elixir is approved by the FDA

Therefore, it is safe to use.

Note that when the arguments of a multinuclear relation are presented within a text-clause, syntax requires a connective like 'and' or 'or'. Instead, when they are presented as indented items, the connective is often omitted, leaving the relation implicit; the reader has to use common sense to divine whether the list represents a conjunction rather or a disjunction.

**Discourse connectives**

A comprehensive treatment of discourse connectives will not be attempted here, but we will cover the three main categories identified by Knott (1996): subordinating conjunctions, coordinating conjunctions, and conjunctive adverbs.

The properties of a discourse connective can be fully specified by four features: MEANING, SYNTAX, LOCUS, and SPELLING. As examples, here are three definitions for the CONCESSION relation:

| | |
|---|---|
| MEANING | `concession` |
| SYNTAX | `subordinating conjunction` |
| LOCUS | `satellite` |
| SPELLING | 'although' |

| | |
|---|---|
| MEANING | `concession` |
| SYNTAX | `coordinating conjunction` |
| LOCUS | `nucleus` |
| SPELLING | 'but' |

| | |
|---|---|
| MEANING | `concession` |
| SYNTAX | `conjunctive adverb` |
| LOCUS | `nucleus` |
| SPELLING | 'however' |

The LOCUS specifies which argument of the relation carries the discourse connective. For a nucleus-satellite relation, the argument is identified either by `nucleus` or `satellite`; for a multinuclear relation it is identified by an ordinal specification such as `initial` or `final`.

For each type of discourse connective, it is possible to state specific constraints on the order of arguments, and on the document units that express them[13].

13 Some of these are also mentioned by Scott and de Souza (1990) and Rosner and Stede (1992).

*Subordinating conjunction*
The spans linked by a subordinating conjunction can be arranged in either order (nucleus first or satellite first), but must be expressed within the same text-clause. For example:

(21) Although it has no significant side effects, never give Elixir to other patients.
(22) Never give Elixir to other patients, although it has no significant side effects.
(23) # Although it has no significant side effects; never give Elixir to other patients.
(24) # Although it has no significant side effects. Never give Elixir to other patients.
(25) # Never give Elixir to other patients; although it has no significant side effects.
(26) # Never give Elixir to other patients. Although it has no significant side effects.

*Coordinating conjunction*
Spans linked by a coordinating conjunction can occur in the same text-clause or in different text-clauses (or higher units), but must be ordered so that the discourse connective is located in the final span (i.e., the second span in the case of a nucleus-satellite relation).

(27) Elixir has no significant side effects, but never give it to other patients.
(28) # But never give Elixir to other patients, it has no significant side effects.
(29) Elixir has no significant side effects; but never give it to other patients.
(30) # But never give Elixir to other patients; it has no significant side effects.
(31) Elixir has no significant side effects. But never give it to other patients.
(32) # But never give Elixir to other patients. It has no significant side effects.

Of course the three examples marked '#' here are prohibited only as a means of realising a CONCESSION relation between the two propositions. They might be acceptable in a text realising a different rhetorical structure in which the satellite had already been expressed.

*Conjuctive adverb*
Spans linked by a conjunctive adverb can occur in different text-clauses (or higher units), but not in the same text-clause. For a nucleus-satellite relation they must be ordered so that the discourse connective is located in the second span.

(33) # Elixir has no significant side effects, however, never give it to other patients.
(34) # However, never give Elixir to other patients, it has no significant side effects.
(35) Elixir has no significant side effects; however, never give it to other patients.
(36) # However, never give Elixir to other patients; it has no significant side effects.
(37) Elixir has no significant side effects. However, never give it to other patients.
(38) # However, never give Elixir to other patients. It has no significant side effects.

Again, some examples marked '#' here would be acceptable in realisations of different rhetorical structures in which the satellite had already been presented.

### 5.3 Planning document structure using hierarchical input

Using the hierarchical method, the input rhetorical structure is a tree in which the terminal nodes are formulas representing elementary propositions (i.e., propositions having no internal rhetorical complexity), while the non-terminal nodes are labelled with rhetorical relations (see for example figure 10). This tree is unordered: the roles of daughter nodes are shown by labels on the arcs — NUCLEUS or SATELLITE in the case of

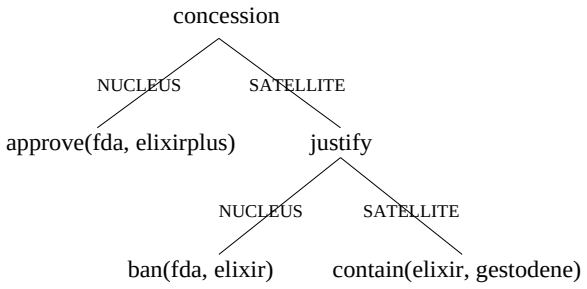



**Figure 10**
Example of Rhetorical Structure

a nucleus-satellite relation, or an integer greater than zero in the case of a multinuclear relation.

The output is a linguistic structure, represented formally by an ordered tree in which each node corresponds to a span of the document. Nodes are labelled by document-structure features (e.g., the level and indentation of the unit), and also by syntactic features, which usually become relevant only at the level of text-clause or below. Discourse connectives are already selected and positioned correctly in the tree; the only task left to the syntactic realiser is to elaborate the tree further by generating clauses that express the elementary propositions.

Our aim in ICONOCLAST has been to find a document-structuring method that will generate *all* document structures that correctly realise an input rhetorical structure, given certain simplifying assumptions about the composition of the document structure and the discourse connectives available in the lexicon. During this initial enumeration of potential solutions, we are not concerned with good style: the procedure should generate clumsy realisations as well as elegant ones. But we do require a minimal standard of correctness. There is no point considering solutions that leave propositions in the rhetorical structure unexpressed in the document structure, or which group them wrongly. The strategy is first to define a procedure that generates all solutions that are worth considering at all; using this minimally correct set as a basis, we can then posit further constraints that impose particular stylistic preferences.

We assume that a minimally correct solution must satisfy three conditions:

1. The terminal nodes of the document structure should express all the propositions in the rhetorical structure.

2. As well as satisfying the document-structure formation rules, the solution must conform to correct syntax within text-clauses. For example, two elementary propositions within a text clause must be linked by a conjunction placed in the right position.

3. The document structure must be structurally compatible with the rhetorical structure.

The first two of these conditions are obvious, but the third needs clarification. What exactly is meant by 'structurally compatible'?

We have explored two definitions of structural compatibility, which are closely related to the mathematical notions of 'isomorphism' and 'homomorphism' (Landman, 1991; Bouayad-Agha, Power, and Scott, 2000). Each notion can be conveniently expressed

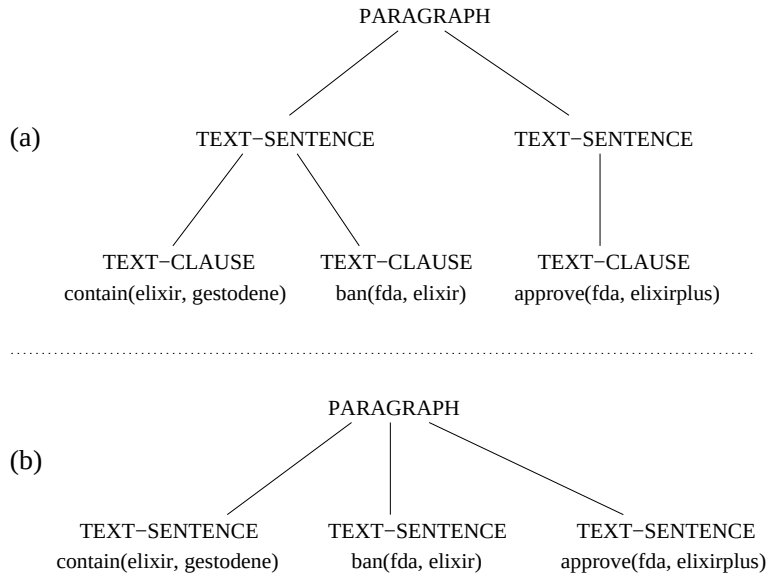



**Figure 11**
Isomorphic and homomorphic document structures

in terms of groupings of propositions in the rhetorical structure and the discourse structure. For an *isomorphism* (the strongest definition of compatibility), two conditions must hold:

1. If a node in the document structure represents a span in which the set of propositions expressed is $P$, there must be a corresponding node in the rhetorical structure that dominates exactly the same set $P$ of propositions.

2. If a node in the rhetorical structure dominates a set $P$ of propositions, there must be a node in the document structure representing a span in which exactly this set $P$ is expressed.

Informally, for an isomorphism, all groupings of propositions in the rhetorical structure must be transmitted faithfully to the document structure, and no new groupings should be introduced For a *homomorphism* from rhetorical structure to document structure, only the first condition is required. Any grouping found in the document structure must correspond to a grouping in the rhetorical structure, but the document structure may 'flatten out' the rhetorical structure by leaving some groupings unexpressed.

These two kinds of compatibility are illustrated by the document structures (a) and (b) in figure 11, which are alternative realisations of the rhetorical structure in figure 10; to simplify, discourse connectives have been left out. Solution (a) is isomorphic with the rhetorical structure, since the propositions dominated by the JUSTIFY relation are grouped together in a separate text-sentence. Solution (b) is only homomorphic with the rhetorical structure, since the propositions are expressed in three separate text-sentences, the internal grouping remaining implicit. Here are two texts that might result:

(39)

(a) Elixir contains gestodene; therefore, it is banned by the FDA. However, the FDA approves ElixirPlus.

(b) Elixir contains gestodene. Therefore, it is banned by the FDA. However, the FDA approves ElixirPlus.

Note that by losing a grouping from the rhetorical structure, text (b) introduces an ambiguity: its form would also be consistent with an alternative rhetorical structure in which the relation JUSTIFY dominated CONCESSION. From the point of view of expressing the rhetorical input precisely, the isomorphic solution is always better. However, when the resulting document structure is complex, or when the correct interpretation can be inferred easily from the content, there might be stylistic reasons for preferring a looser homomorphic solution. As a criterion for generating all minimally correct solutions, therefore, we believe the homomorphic definition of compatibility is more appropriate[14].

Having decided what counts as a correct solution, our next task is to find an efficient algorithm that will generate all and only these solutions. In ICONOCLAST, this is done through a constraint-solving method which formalises the options for realising each part of the rhetorical structure, then eliminates those combined choices that violate constraints. The technique has been explained fully elsewhere (Power, 2000), so we will confine ourselves here to a sketch of the main points.

For each node $N_R$ on the rhetorical structure, we can lay out some options on how the rhetorical fragment dominated by this node will be realised in the document structure. The crucial choices are as follows:

- What should be the level and indentation of the document structure node that realises the proposition or relationship $N_R$?
- If $N_R$ is a non-terminal node, labelled with a rhetorical relation, what discourse connective (if any) should express this relation in the document-structure?
- If $N_R$ is non-terminal, in what linear order should its daughter nodes be realised in the document structure?

These decisions can be formalised by associating with each node four variables which we will call LEVEL, INDENTATION, CONNECTIVE and POSITION. The potential values of these variables represent options for realising the relevant part of the rhetorical structure, and these options will be reduced by considering constraints on which *combinations* of values are allowed. Initially, the possibilities are as follows:

LEVEL

The level of the document structure unit realising $N_R$ will be a document unit along the hierarchy from text-phrase to chapter. In the program, it is convenient to represent this by an integer, so that constraints on higher or lower level can be implemented using the operations '>' and '<'; we will here continue the earlier notation in which $L_0$ means text-phrase, $L_1$ means text-clause, and so forth.

14 We actually believe that even the less strict requirement of a homomorphism from RS to DS is too strong, because one occasionally finds natural texts which violate it. For example, in figure 4, RS and DS are not only non-isomorphic, but non-homomorphic, because there is a grouping in the DS (propositions A, D, E) that is not found in the RS (no node in the RS dominates just these three propositions). This raises the question of how one can distinguish non-homomorphic solutions that are acceptable (like figure 4) from ones that are unacceptable (probably the great majority). To defer this difficult issue, we have preferred in this paper to confine our attention to homomorphic solutions. For further discussion see Bouayad-Agha, Power, and Scott (2000).

INDENTATION
The indentation of the document structure unit will be a value in the range $I_0$ to $I_{Max}$, as explained earlier. Again, in the implementation, it is convenient to use an integer.

CONNECTIVE
The value of this variable is either $\emptyset$, meaning that no connective should be used, or a word from the lexicon. Thus if the node $N_R$ has the label CONCESSION, the potential values might be $\emptyset$, $although$, $but$ and $however$.

POSITION
This represents the order in which $N_R$ will be realised in the document structure, in comparison with its sisters. The value must lie in the range $1..S$, where $S$ is the total number of sisters (including $N_R$). Thus if $N_R$ is an argument in a nucleus-satellite relation, the range will be 1..2; if it is an argument in a multinuclear relation, the range may be larger.

The solution process consists in determining the set of options for each variable (these are known in Constraint Logic Programming as the *domain* of the variable), then imposing constraints on combinations of values, then enumerating all combinations that satisfy the constraints. Each admissible combination of values can then be used as a blueprint for building one of the document structures that may realise the input rhetorical structure. The constraints imposed during the second phase of this algorithm are essentially those described in the last section.

## 6 Examples of document structuring

We now give two examples of the document-structuring method outlined in the last section. First, we look in detail at a very simple task, to make it clear how the method works. Then we show some output for a more complex task, for which the program will generate dozens of solutions even if the wording of individual propositions is held constant.

### 6.1 Simple example

To view the document-structuring method from close up, we will use a simplified version of the task discussed in the last section. The rhetorical structure will comprise just two propositions linked by a CONCESSION relation:

$$concession(ban(fda, elixir), approve(fda, elixirplus))$$

The method works by computing the options for realising each constituent of the rhetorical structure, where a 'constituent' is any node in the rhetorical structure tree along with its descendents. Thus in the present example there are three constituents — the two propositions, and the whole relationship — and for convenience we will label them as follows:

$$A: \ approve(fda, elixirplus)$$
$$B: \ ban(fda, elixir)$$
$$C: \ concession(ban(fda, elixir), approve(fda, elixirplus))$$

We can now begin to characterise the units in the document structure that will realise the rhetorical constituents $A, B, C$. For each unit, four choices must be made: its $level$, its $indentation$, its $position$ in relation to its sisters, and its $connective$. We therefore have a total of twelve variables. Any combination of values that satisfies the constraints (as discussed in the last section) will serve as the blueprint for a solution.

Following the usual technique for solving constraint satisfaction problems (Hentenryck, 1989), we assign to each variable a domain of potential values:

| **Constituent** | **Level** | **Indentation** | **Position** | **Connective** |
|---|---|---|---|---|
| $A$ $(approve)$ | $\{L_0..L_3\}$ | $\{I_0, I_1\}$ | $\{P_1, P_2\}$ | $C_0$ |
| $B$ $(ban)$ | $\{L_0..L_3\}$ | $\{I_0, I_1\}$ | $\{P_1, P_2\}$ | $C_0$ |
| $C$ $(concession)$ | $\{L_0..L_3\}$ | $\{I_0, I_1\}$ | $P_1$ | $\{C_0, C_{alt}, C_{but}, C_{how}\}$ |

Some of these assignments require some explanation:

- We make the simplifying assumption that the highest unit required will be the paragraph (formalised as level $L_3$), and that the maximum indentation will be one place (formalised as $I_1$).
- Since constituents $A$ and $B$ are sisters in the rhetorical structure, their relative order is formalised by a choice between two positions, $P_1$ and $P_2$. Instead, constituent $C$ has no sister in the rhetorical structure, so as an 'only child' it can have only one position value.
- Constituents $A$ and $B$ are not associated with a discourse connective, because they are elementary propositions, so their value for the $connective$ variable is $C_\emptyset$ (meaning 'no connective'). Instead, constituent $C$ is asociated with the CONCESSION relation, for which we assume that the lexicon offers three connectives: $although$ ($C_{alt}$), $but$ ($C_{but}$), and $however$ ($C_{how}$). The initial domain for the variable $connective(C)$ therefore comprises these three possibilities along with $C_\emptyset$, the option of leaving the relation implicit.

Having assigned the initial domains, we next proceed to apply some constraints; these will have the effect of reducing some of the domains. First of all, there are some constraints applicable to the unit that will serve as the root of the whole document structure:

**Root Level**
In a document generation task, we usually have some preconception about the size of the whole document — e.g., it might be a chapter, or a section. Since this is a small rhetorical structure, we might decree that the whole text should consist of a paragraph, so that $level(C) = L_3$.

**Root Indentation**
It makes no sense for a whole document to be an indented item, so to realise the root constituent $C$ we may stipulate that $indentation(C) = I_0$.

Having applied these constraints to the variables realising $C$, we can impose some further constraints that arise from the relationship between $C$ and its direct constituents $A$ and $B$:

**Argument Indentation**
As pointed out in the last section, it is permissible to indent the arguments of a multinuclear relation, but not a nucleus-satellite relation. Therefore, given that the indentation of $C$ has been fixed as $I_0$, the indentations of $A$ and $B$ must also be $I_0$.

**Parental Domination**
Since $A$ and $B$ are constituents of $C$ in the rhetorical structure, the units realising $A$ and $B$ in the document structure will occur within the constituent realising $C$. Given that all indentations have been set to $I_0$, this means that $level(C)$ should outrank both $level(A)$ and $level(B)$, and that consequently the option $L_3$ should be removed from the domains of the last two variables.

**Sister Equality**
Because $A$ and $B$ are sisters in the rhetorical structure (i.e., arguments of the same relation), it is appropriate that they should be realised in the document structure by units of the

same level. We can therefore impose the constraint that $level(A) = level(B)$. This does not immediately affect the domains of these variables, but it does mean that as soon as one is fixed, so is the other.

**Sister Position**

The units realising the sisters $A$ and $B$ must occur in one of two linear orders in the document structure — they cannot both come first, or both second. We may therefore set the constraint $position(A) \neq position(B)$. Again this has no immediate effect on the domains, but as soon as one value is fixed, so is the other.

**Obligatory Connective**

As Scott and Souza (1990) point out, an NLG system is ill-equipped to judge when a rhetorical relation may be left implicit, so it makes sense to play safe by *always* realising the relation by a discourse connective. Following this policy, we can remove $C_0$ from the domain of $connective(C)$.

Through applying these constraints, the domains of the variables have been reduced as follows:

| Constituent | Level | Indentation | Position | Connective |
|---|---|---|---|---|
| $A$ $(approve)$ | $\{L_0..L_2\}$ | $I_0$ | $\{P_1, P_2\}$ | $C_0$ |
| $B$ $(ban)$ | $\{L_0..L_2\}$ | $I_0$ | $\{P_1, P_2\}$ | $C_0$ |
| $C$ $(concession)$ | $L_3$ | $I_0$ | $P_1$ | $\{C_{alt}, C_{but}, C_{how}\}$ |

Before enumerating solutions, we need to impose a final set of constraints that are conditional on the choice of discourse connective for $C$. These constraints have been explained fully above, so here we only point out how they are stated in terms of our variables.

**Subordinating Conjunction**

The units connected by $although$ may occur in any order, but they must be text-phrases. Therefore, if $connective(C) = C_{alt}$, then $level(A) = level(B) = L_0$.

**Coordinating Conjunction**

We will assume that a coordinating conjunction like *but* may link spans within a text-clause, or across text-clauses and text-sentences, so that no constraint on the levels of $A$ and $B$ results. However, the satellite must precede the nucleus, so we have $position(B) = P_1$ (and hence, by Sister Position, $position(A) = P_2$).

**Conjunctive Adverb**

A conjunctive adverb like $however$ should link spans in units of text-clause or higher,[15] so we can impose the constraints $level(A) > L_0$ and $level(B) > L_0$. (Note that in the implementation, only one of these need be applied since the other results from Sister Equality.) For $however$, the satellite must precede the nucleus, so again we have $position(B) = P_1$ and $position(A) = P_2$.

With all potential interactions among decisions taken care of, enumeration can proceed. In an implementation in Constraint Logic Programming, this is done by a method called 'labelling' in which the remaining possible values for each variable are tried one by one, with backtracking, until all permitted combinations have been produced. For this example, this process can be understood most easily if we explore in turn the possible values for $connective(C)$.

Suppose first that $connective(C) = C_{alt}$ — i.e., that we try $although$. Through the constraint Subordinating Conjunction this choice immediately fixes all values of $level$, yielding the following domains:

| Constituent | Level | Indentation | Position | Connective |
|---|---|---|---|---|
| $A$ $(approve)$ | $L_0$ | $I_0$ | $\{P_1, P_2\}$ | $C_0$ |
| $B$ $(ban)$ | $L_0$ | $I_0$ | $\{P_1, P_2\}$ | $C_0$ |
| $C$ $(concession)$ | $L_3$ | $I_0$ | $P_1$ | $C_{alt}$ |

15 As Oates (2001) has shown, this is not true if *multiple* discourse connectives are allowed — for instance, one might say 'The FDA bans Elixir but, however, it approves ElixirPlus'. But in the present example we assume, for simplicity, that only single discourse connectives are used.

The only remaining issue is the relative orders of the spans realising $A$ and $B$. Enumerating arbitarily on $position(A)$, we first try the value $P_1$; by Sister Position this immediately yields $position(B) = P_2$, whereupon all values in the table are fixed. We therefore have our first complete solution:

| **Constituent** | **Level** | **Indentation** | **Position** | **Connective** |
|---|---|---|---|---|
| $A\ (approve)$ | $L_0$ | $I_0$ | $P_1$ | $C_0$ |
| $B\ (ban)$ | $L_0$ | $I_0$ | $P_2$ | $C_0$ |
| $C\ (concession)$ | $L_3$ | $I_0$ | $P_1$ | $C_{alt}$ |

This is still only a blueprint for a document structure, but with these specifications the rest can be inferred. First, since the units realising $A$ and $B$ are text-phrases, while the root unit realising $C$ is a paragraph, the program can interpolate the units needed to make a well-formed document structure: the paragraph has a single text-sentence, which has a single text-clause, which comprises the two text-phrases. Next, since 'although' attaches to the satellite, the program coordinates it with the clause that will syntactically realise $B$. The document structure is now complete, and after syntactic realisation of the two propositions we might obtain the following paragraph:

**Solution 1:** The FDA approves ElixirPlus, although it bans Elixir.

Backing up, the program can now try the alternative order in which $position(A) = P_2$. After inferring the complete document structure in the same way, we thereby obtain a second solution.

**Solution 2:** Although the FDA bans Elixir, it approves ElixirPlus.

Having fully explored the possibilities with $connective(C) = C_{alt}$, we back up and try the next value, namely $C_{but}$. This time the constraint Coordinating Conjunction applies, fixing the $position$ values but leaving several possiblities for $level$:

| **Constituent** | **Level** | **Indentation** | **Position** | **Connective** |
|---|---|---|---|---|
| $A\ (approve)$ | $\{L_0..L_2\}$ | $I_0$ | $P_2$ | $C_0$ |
| $B\ (ban)$ | $\{L_0..L_2\}$ | $I_0$ | $P_1$ | $C_0$ |
| $C\ (concession)$ | $L_3$ | $I_0$ | $P_1$ | $C_{but}$ |

Enumerating on one of the $level$ variables, for instance $level(A)$, we can now try in turn the values $L_0$, $L_1$ and $L_2$ (text-phrase, text-clause and text-sentence); by Sister Equality, any choice will be copied across to $level(B)$, so that we obtain only three further solutions rather than nine:

**Solution 3:** The FDA bans Elixir, but it approves ElixirPlus.

**Solution 4:** The FDA bans Elixir; but it approves ElixirPlus.

**Solution 5:** The FDA bans Elixir. But it approves ElixirPlus.

Finally, we try the remaining option for which $connective(C) = C_{how}$. The constraint Conjunctive Adverb now applies, fixing the order of $A$ and $B$ and ruling out solutions for which these propositions are realised by text-phrases:

| **Constituent** | **Level** | **Indentation** | **Position** | **Connective** |
|---|---|---|---|---|
| $A\ (approve)$ | $\{L_1, L_2\}$ | $I_0$ | $P_2$ | $C_0$ |
| $B\ (ban)$ | $\{L_1, L_2\}$ | $I_0$ | $P_1$ | $C_0$ |
| $C\ (concession)$ | $L_3$ | $I_0$ | $P_1$ | $C_{how}$ |

It remains to enumerate (as before) on $level(A)$, trying in turn the values $L_1$ and $L_2$ (text-clause and text-sentence). Sister Equality again copies any selected value across to $level(B)$, so that we obtain just two more solutions:

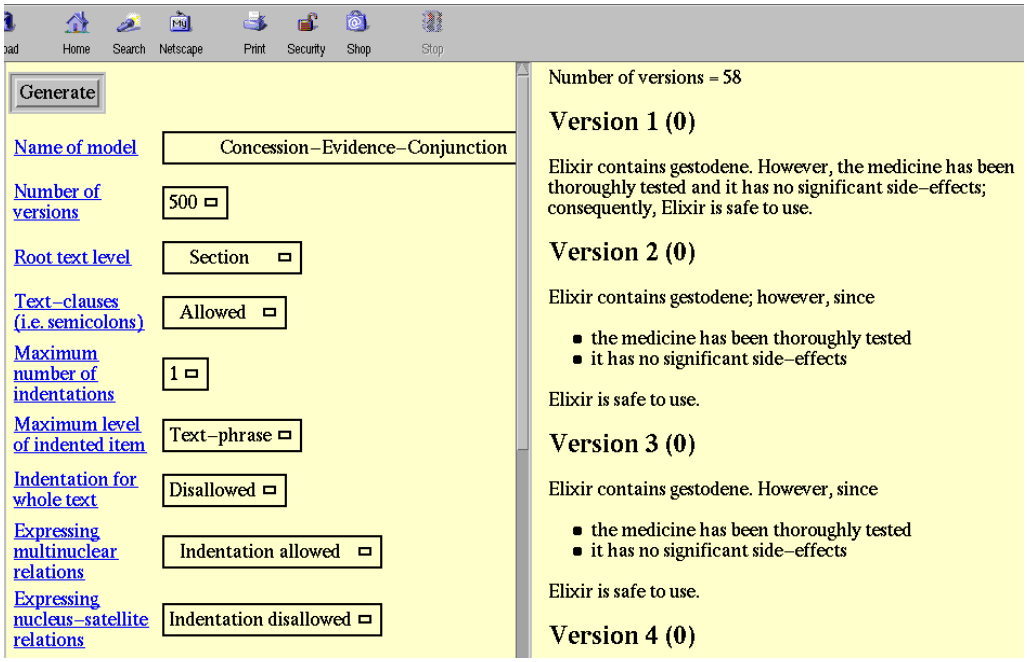



**Figure 12**
Running the document planner

**Solution 6:** The FDA bans Elixir; however, it approves ElixirPlus.

**Solution 7:** The FDA bans Elixir. However, it approves ElixirPlus.

Obviously further texts could be obtained by different wordings of the propositions, or different placements of 'however' (e.g., 'The FDA bans Elixir; the FDA approves ElixirPlus, however'). But these variations do not concern us here since we assume they are introduced during syntactic realisation, not during document structuring.

### 6.2 Complex example

To examine a more complex example it is convenient to use a version of ICONOCLAST in which the wording of individual propositions is prespecified, so that the program's only task is to explore the set of possible document structures. The input to this program is provided in the form of an XML file using the tags `RhetRep` (for rhetorical relationships) and `SemRep` (for propositions) (Cahill et al., 1999). The roles of satellite and nucleus are distinguished implicitly by the order of the elements (satellite precedes nucleus). Thus the example discussed in the last section

$$concession(ban(fda, elixir), approve(fda, elixirplus))$$

could be encoded as follows:

```
<RhetRep relation=concession>
  <SemRep prop="the FDA bans Elixir"/>
  <SemRep prop="the FDA approves ElixirPlus"/>
</RhetRep>
```

Obviously this will sometimes lead to some clumsy wording (e.g., through inappropriate decisions on whether to use pronouns), but we can try to ignore this when evaluating the generated document structures.

As a more complicated example we will consider a rhetorical structure with two nucleus-satellite relations (EVIDENCE, CONCESSION) and one multinuclear relation (LIST):

```
<RhetRep relation=concession>
  <SemRep prop="Elixir contains gestodene"/>
  <RhetRep relation=evidence>
    <RhetRep relation=list>
      <SemRep prop="the medicine has been thoroughly tested"/>
      <SemRep prop="it has no significant side effects"/>
    </RhetRep>
    <SemRep prop="Elixir is safe to use"/>
  </RhetRep>
</RhetRep>
```

The main idea here is that Elixir is safe to use although it contains gestodene; the claim that Elixir is safe is supported by evidence comprising two conjoined facts: it has been thoroughly tested, and it has no significant side effects.

Figure 12 shows a snapshot of the program running on this example. Through the control panel on the left, the user can decide which XML model to use as input; at present this is set to 'Concession-Evidence-Conjunction', the name given to this input model. Using the button labelled 'Number of versions', the user can determine the maximum number of versions that will be generated; since this has been set to 500, the program will return the first 500 solutions that it finds.[16] The other buttons control hard constraints, and have been set deliberately to rather restrictive values (e.g., indented items have been allowed only to one level of indentation, and may consist only of text-phrases, not a larger unit like text-sentence or paragraph). Even with these restrictive settings, 58 solutions have been generated; they are presented in the pane on the right, ordered (partially) from best to worst. The scores in brackets, after the version number, report the number of defects that the program detected: thus for all the solutions that appear in the snapshot, no defects were found.[17] As a comparison, here is version 58, which came bottom of the class with six defects:

**Version 58 (6)**

Elixir contains gestodene.

However, the medicine has been thoroughly tested.

It has no significant side effects.

Consequently, Elixir is safe to use.

The specific defects here were 'Single-sentence paragraph' (occurring four times) and 'Lost rhetorical grouping' (occurring twice). The full set of solutions is given in the Appendix.

At present the program looks for six stylistic defects, which were formulated mainly by looking at generated solutions and making an intuitive judgement why they were bad. We have not tried to give the stylistic assessment a sound theoretical or empirical basis; the aim at this stage is to confirm that by applying some simple intuitive principles we can separate reasonably good solutions from obviously horrible ones. The six defects are as follows — and we stress again that they are only provisional:

16 Obviously, if there are fewer than 500 solutions, it will return all the solutions it finds.
17 The system also reports on the type of defect that it finds.

**Nucleus before satellite**

It is generally the case for English that the more important information is placed at the end of the sentence (i.e., *end focus* (Quirk et al., 1985)). For example, the rheme of a sentence comes after its theme, and new information is typically placed after given information (Halliday, 1985; Givón, 1988). There is also psycholinguistic evidence to suggest that sentences that conform to this general pattern are processed more easily (Yekovich, Walker, and Blackman, 1979). Since by definition the nucleus is more important than the satellite (Mann and Thompson, 1986, pg. 6), it thus makes sense to place the nucleus second. Obviously this principle is debatable, and the best order might differ from one relation to another, but we have noticed that for the very common relations such as CONCESSION and EVIDENCE the order satellite-nucleus seems to work better. The program therefore scores a defect every time that a nucleus is placed before its satellite.

**Left-branching structure**

Fodor, Bever, and Garrett (1974) report that left-branching structures take more time to process and remember than right-branching ones. We have also noticed that when a document structure coordinates two units of different sizes, it reads best when the smaller unit is placed first. We believe that this may be related to the more general organizational principle of *end weight* (Quirk et al., 1985). Thus in the programme, when an elementary proposition is coordinated with a unit containing several propositions, a defect is scored whenever the elementary proposition is placed second.

**Lost rhetorical grouping**

As discussed, our method of document structuring does not demand an isomorphism between rhetorical structure and document structure; consequently, a grouping that is present in rhetorical structure may be left implicit in the document structure. Although we allow a grouping to be lost in this way, a defect is scored every time it happens.

**Single-sentence paragraph**

Paragraphs containing a single text-sentence usually look strange, so they are scored as a defect.

**Oversimple text-clauses**

This is a subtler defect that we noticed only as a result of experience with the program. In most cases it looks odd to compose a sentence from two text-clauses, each expressing a single proposition:

> Elixir contains gestodene; therefore, it is banned by the FDA.
> The FDA bans Elixir; however, it approves ElixirPlus.

Assuming it is agreed that these sentences are a little strange, why should this be so? We would suggest the following reason. The semicolon is a somewhat unusual device, more sophisticated than the comma, and one therefore expects it to be used only when ordinary methods are unsatisfactory. In these examples, containing only two propositions, a single text-clause using conjunctions (e.g., since, although) instead of adverbs would express the meaning equally clearly, so the semicolon seems unnecessary, and therefore distracting. A defect is therefore scored every time this happens.

**Repeated discourse connective**

If a rhetorical structure contains two relations of the same type, one dominating the other, a defect is scored if they are expressed by the same discourse connective. Here is a simple example of this defect (repetition of 'although'), followed by an alternative solution that avoids the defect:

> The FDA approves ElixirPlus although it bans Elixir although Elixir has been thoroughly tested.
>
> The FDA approves ElixirPlus but it bans Elixir although Elixir has been thoroughly tested.

## 7 Discussion and Conclusion

Our main aim has been to introduce, and motivate, *abstract document structure* as an important representational level in natural language processing. This proposal rests on two separate claims: first, that it is useful to distinguish abstract document structure from concrete graphical realisation; second, that abstract document structure should also be distinguished from rhetorical structure.

*Abstract vs. Concrete:* We argue that the transition from a rhetorical-semantic message to a fully specified document can usefully be divided into two stages. During the first stage, the author puts the message into words, and organises the words into higher linguistic units like sentences, paragraphs, and bulleted lists. All decisions pertaining to the realisation of literal content take place during this stage. During the second stage, detailed formatting takes place: quotations are realised either by single or double quotes (or some other method); emphasis is realised through italics, bold face, small capitals, or underlining; paragraphs are realised by an introductory tab or by double-line spacing; and so forth. These decisions do not affect the factual or logical content of the document, although they might convey 'meanings' of a subtler sort, communicated through the typographical preferences of the authors (e.g., traditional *vs.* trendy, ornamental *vs.* puritanical, compact *vs.* expansive).

It is worth noting that similar distinctions occur in other branches of linguistics. Thus in phonology, a distinction is made between a *phonemic* level and a *phonetic* level. The word 'grass' has a single phonemic representation, but will be pronounced differently by people from different regions or different social classes; these distinctions are phonetic, not phonemic. To refer to an area in the garden by 'grass' rather than 'lawn' is one kind of decision; to pronounce 'grass' with a short or long vowel is another. A theory that mapped directly from the semantic concept to the phonetic form would miss a generalisation that is not only obvious theoretically, but useful practically. The invention of writing provides additional support for an intermediate phonemic level, because the different pronunciations of 'grass' are all written down in the same way; similarly, we would argue, the representational level of abstract document structure has received more recent support from the invention of mark-up languages like HTML and LaTeX.

The concept of abstract document structure is not linked to any particular architecture for natural language generation or understanding. In the RAGS "reference architecture" for NLG (Mellish et al., 2000; Cahill et al., 2001), document structure is distinguished from rhetorical structure as a data type, with no commitment as to when these two data structures are created during the generation process. The ICONOCLAST system, described in this paper, assumes that rhetorical planning fully precedes document structuring: in other words, the RST tree has to be complete before the process of creating a

document structure can begin. Such an architecture could be thought of as a refinement of the standard pipeline (Reiter, 1994), with the document-planning phase divided into two parts (rhetorical planning and document structuring). However, the ICONOCLAST method would work equally well if a partial assignment of document structure was part of its input – this would be treated merely as a more specific set of constraints on possible solutions; this is precisely the arrangment that is used in the RAGS reimplementation of the Caption Generation system (Mellish et al., 2000).

*Document Structure vs. Rhetorical Structure:* In the terminology of HTML and other mark-up languages, tags like 'section' and 'description list' are sometimes called *logical*, suggesting that they are rhetorical rather than linguistic categories. We have argued that this is a mistake, comparable to a confusion of syntax with semantics. In our view, the term 'rhetorical structure' should properly be applied to the higher-level pragmatic and semantic organisation of the message, with no commitment to the means by which this message will be expressed — whether by speech, or gesture, or diagram, or written document. By contrast, the categories 'section' and 'description list' are specific to a particular medium, the written document; hence the term 'document structure'. The two levels are easily confused because we often refer to spans of a document by a noun that describes their rhetorical role (e.g., 'summary' in an academic paper).

Some of the distinctions made in this paper have parallels in work on document analysis. Various representations for document structure have been proposed in this community, of which the most developed is the Document Attribute Format Specification (DAFS) (Dori et al., 1997). In DAFS, the physical structure of a document is distinguished from its logical structure; typical physical units are block, frame, page, and page set, while typical logical units are sentence, paragraph, section, and header. More formally, the units of logical structure, called "textons", are organised into levels from character (level 0), word (level 1), sentence or phrase (level 2), up to sections and whole documents; the lower levels (up to paragraph) are called "simple textons", and the higher levels are called "compound textons". Simple textons are realised through blocks of text, while compound textons additionally have a heading and (optionally) a trailer. There is considerable overlap here with our distinction between concrete and abstract document structure; differences arise because the analysis community is concerned mainly with the relationship between logical structure and physical structure (to use their terminology), whilst the generation community, coming from the opposite direction, is concerned mainly with the relationship between logical structure (i.e., abstract document structure) and rhetorical structure.

Having drawn these distinctions, we have sketched a formal theory of abstract document structure, and shown through an implemented NLG system that this theory allows us to enumerate systematically the high-level linguistic structures that can realise a given rhetorical-semantic input. The formal description of document structure is based on Nunberg's (1990) text-grammar, which has been extended in two ways. First, we introduce larger units such as sections and chapters; Nunberg instead focusses on the levels relevant for punctuation (i.e., text-sentence and below). Secondly, we introduce a second feature, that of *abstract indentation*. While Nunberg categorises units only by the feature we have called LEVEL (the hierarchy from text-phrase, text-clause, etc., up to section and chapter), we categorise units by two features, LEVEL and INDENTATION. This allows the generation of such patterns as bulleted lists — including more complex cases in which one list is embedded within another. Using this descriptive scheme, it has proved relatively easy to state constraints on the interaction between content, layout and wording, especially as regards the use of discourse connectives.

By introducing a formal scheme for representing document structure, we have been

able to define the task of *document structuring* in a simple and clear way. Following Scott and Souza (1990), and many other researchers on NLG, it is assumed that the rhetorical input takes the form of an RST tree; the output is a tree representing high-level linguistic structure, each node being labelled with a document structure category defined by the features LEVEL and INDENTATION. Our scheme is by no means complete (e.g., it has no treatment of tables, or figures, or text presented in boxes); however, it is sufficient to generate hundreds of alternative solutions even for a rhetorical structure containing only four or five elementary propositions. By clarifying the task of document structuring in this way, we have been able to define it as a constraint satisfaction problem, and thus to implement a system in which the relevant constraints are defined declaratively; this means that constraints can be added or removed without changing the rest of the program. Such a system is useful not only as a module in an NLG architecture, but also as a tool for theoretical investigation — the results of any proposed combination of constraints can be quickly tested.

In pursuing this investigation, our methodology has been essentially the same as Nunberg's, relying largely on intuition as a means of separating the wheat from the chaff. Moreover, by implementing the theory in a system that can enumerate solutions systematically, we are able to test more thoroughly any proposed rule or constraint — at least for simple examples. Such a method assumes, firstly, that intuition is a reliable guide, and secondly, that constraints derived from small examples will apply also to full-scale examples. In the initial stages of an investigation, these assumptions seem reasonably safe. Many of the solutions generated by the program are so obviously good or bad that there is no point in submitting them to the judgement of literary experts, or some other kind of empirical test. No doubt large-scale examples will require additional constraints, but much can still be learned from simple ones: an intrinsically bad paragraph will usually remain bad when placed into a larger context.

As a contrast, it is interesting to consider the investigation into layout by Bateman *et al* (2001). Their approach could not be more different. Instead of simple examples, they analyse (and regenerate) an exceedingly complicated page from a magazine. This page, which describes the game of hockey, includes several drawings, a photograph, diagrams of the pitch, boxes of text, two headers, and a glossary, all laid out in five different grids, each having a different division into columns. Their RST analysis of this page is correspondingly complex, with 45 elementary propositions and the same number of rhetorical relationships (the whole RST tree therefore has nearly 100 nodes). To analyse such an example informally may be a useful source of insights, but to attempt a complete formal analysis (and generation) of the page seems bold in the extreme. However, despite this difference in approach, the framework that emerges from Bateman *et al.*'s work is broadly similar to ours. First, a distinction is made between 'layout structure' and 'physical layout' (section 3.1); although the discussion here concerns boxes in a grid rather than more conventional linguistic units like section and paragraph, this distinction reflects the need for an abstract level of representation which can be related more easily to the rhetorical structure of the message. Secondly, in sections 4 and 5, they distinguish clearly between layout structure and rhetorical structure, pointing out that the two are not necessarily isomorphic, and that constraints on the mapping must therefore be considered:

> Mapping is generally achieved by placing parts of the RST-structure in correspondence with particular nodes in layout structure [...] As we have now seen, however, this correspondence is complicated by the fact that the layout structure and the RST tree need not remain congruent. *(Bateman et al. 2001, page 430)*

However, Bateman and his colleagues do not provide a detailed account of the formation rules for layout structure, or the constraints on the mapping between the RST tree and the layout structure. We are unsure, for example, whether 'layout structure' would include such patterns as sections, paragraphs, and bulleted lists. Nevertheless, the role played by these two abstract representations seems similar — they mediate between rhetorical structure and physical layout — so there is some reason to think that the two approaches are yielding results that are compatible.

Both in the Bateman *et al.* study and in our own work, there is a problem in how the proposed representations and constraints should be validated. Our own approach, at least provisionally, has been that the theory should be embodied in a program which can generate many alternative solutions and rank them by some kind of cost function; at this point, we rely on intuition to judge whether the system has generated a plausible set of solutions and ranked them in an appropriate order. For the very complex examples considered by Bateman *et al.* the set of solutions could only be sampled: even keeping the wording of individual propositions constant, they would number billions. Evaluation in this field has to take account of style as well as quality; in other words, it has a subjective side as well as an objective one. The problem is not just to generate a good solution, but to generate one that satisfies a set of subjective preferences, so that for example different documents produced for the same company will exhibit the desired consistency of style. Eventually, some kind of empirical investigation will be needed (e.g., an expert evaluation, or a study of the impression made on the intended readers). At the present state of knowledge, however, such refinements seem exaggerated: if we can separate the satisfactory from the barbaric we will be more than content.

**Acknowledgments**

This research was carried out as part of the ICONOCLAST project (*http://www.itri.bton.ac.uk/projects/iconoclast*), with funding from the Engineering and Physical Sciences Research Council (EPSRC) grant number L77102. We are extremely grateful to Kees van Deemter and three anonymous reviewers for their critical feedback on an earlier draft of this paper.

**References**

ABPI. 1997. *Compendium of Patient Information Leaflets*. Association of British Pharmaceutical Industry, London.

Arens, Yigal and Eduard Hovy. 1990. Text layout as a problem of modality selection. In *Proceedings of the 5th Conference on Knowledge-Based Specification. RADC Workshop*, pages 87–94, Syracuse, NY.

Bateman, John, Thomas Kamps, Jorge Kleinz, and Klaus Reichenberger. 2001. Towards constructive text, diagram, and layout generation for information presentation. *Computational Linguistics*, 27(3):409–449.

Bolinger, Dwight, editor. 1972. *Intonation*. Penguin, Harmonsworth, England.

Bouayad-Agha, Nadjet. 2000. Using an abstract rhetorical representation to generate a variety of pragmatically congruent texts. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL), Student Workshop*, pages 16–22, Hong Kong, October.

Bouayad-Agha, Nadjet. 2001. *The Role of Document Structure in Text Generation*. Ph.D. thesis, University of Brighton. Also available as technical report ITRI-01-24.

Bouayad-Agha, Nadjet, Richard Power, and Donia Scott. 2000. Can text structure be incompatible with rhetorical structure? In *Proceedings of the International Natural Language Generation Conference*, pages 194–200, Mitzpe Ramon, Israel, 12–16 June.

Bouayad-Agha, Nadjet, Donia Scott, and Richard Power. 2000. Integrating content and style in documents: a case study of patient information leaflets. *Information design journal*, 9(2):161–176.

Bouayad-Agha, Nadjet, Donia Scott, and Richard Power. 2001. The influence of layout on the interpretation of referring expressions. In L. Degand, Y. Bestgen, W. Spooren, and L. van Waes, editors, *Multidisciplinary Approaches to Discourse*. Amsterdam & Nodus Publications, pages 133–141. Presented at the Multidisciplinary approaches to discourse (MAD) workshop, August 2001, Ittre, Belgium. A slightly modified version is also available as a technical report ITRI-01-23.

Cahill, Lynne, John Carroll, Roger Evans, Daniel Paiva, Richard Power, Donia Scott, and Kees van Deemter. 2001. From rags to riches: exploiting the potential of a flexible gen eration architecture. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL'01)*, pages 98–105, Toulouse, France. Also available as ITRI Technical Report, ITRI-01-07.

Cahill, Lynne, Christine Doran, Roger Evans, Chris Mellish, Daniel Paiva, Mike Reape, Donia Scott, and Neil Tipper. 1999. Towards a reference architecture for natural language generation systems. Technical Report ITRI-99-14, Information Technology Research Institute, March. First Year Report.

Chomsky, Noam and Morris Halle. 1968. *The Sound Pattern of English*. Harper and Row, New York.

Coch, Jose. 1996. Overview of AlethGen. In *Demonstrations and Posters of the Eighth International Natural Language Generation Workshop (INLG'96)*, pages 25–28, Herstmonceux Castle, Sussex, UK.

Crystal, David. 1969. *Prosodic Systems and Intonation in English*. Cambridge University Press, Cambridge.

DiMarco, Chrysanne, Graeme Hirst, Leo Wanner, and John Wilkinson. 1995. Healthdoc: Customizing patient information and health education by medical condition and personal characteristics. In *First International Workshop on Artificial Intelligence in Patient Education*, Glasgow, August.

Dori, Dov, David Doermann, Christian Shin, Robert Haralick, Ihsin Phillips, Mitchell Buckman, and David Ross. 1997. The representation of document structure: a generic object-process analysis. In P. Wang and H. Bunke, editors, *Handbook on Optical Character Recognition and Document Image Analysis*. World Scientific Publishing Company, Singapore, pages 421–456.

Fodor, Janet, Thomas Bever, and Merril Garrett. 1974. *The Psychology of Language*. McGraw-Hill, New York.

Ford, Andrew and Tim Dixon. 1996. *Spinning the Web*. International Thompson Computer Press, London.

Givón, Talmy. 1988. The pragmatics of word order: predictability, importance and attention. In M. Hammond, E. Moravcsik, and J. Werth, editors, *Studies in Syntactic Typology*. John Benjamins, Amsterdam, pages 243–284.

Halliday, Michael. 1967. *Intonation and Grammar in British English*. Mouton, The Hague.

Halliday, Michael. 1985. *An Introduction to Functional Grammar*. Edward Arnold, Baltimore.

Hentenryck, Pascal Van. 1989. *Constraint Satisfaction in Logic Programming*. MIT Press, Cambridge, Mass.

Huang, Xiaorong and Armin Fiedler. 1997. Proof verbalization as an application of NLG. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'97)*, pages 965–970, Nagoya, Japan.

Knott, Alistair. 1996. *A Data-Driven Methodology for Motivating a Set of Coherence Relations*. Ph.D. thesis, University of Edinburgh.

Knott, Alistair, John Oberlander, Mick O'Donnell, and Chris Mellish. 2001. Beyond elaboration: the interaction of relations and focus in coherent text. In T.Sanders, J.Schilperoord, and W.Spooren, editors, *Text Representation: Linguistic and Psycholinguistics Aspects*. Benjamins, Amsterdam, pages 181–196.

Ladd, Robert. 1996. *Intonational Phonology*. Cambridge University Press, Cambridge.

Landman, Fred. 1991. *Structures for Semantics*. Studies in Linguistics and Philosophy. Kluwer Academic Publishers.

Lavoie, Benoit and Owen Rambow. 1997. A fast and portable realizer for text generation systems. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP'97)*, pages 265–68, Washington, DC.

Mann, William and Sandra Thompson. 1986. Rhetorical structure theory: Description and construction of text structures. In *Proceedings of the 3rd International Workshop on Text Generation*, Nijmegen, The Netherlands, August.

Mann, William and Sandra Thompson. 1987. Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, Information Sciences Institute, University of Southern California.

Marcu, Daniel. 1997. From local to global coherence: A bottom-up approach to text planning. In *The Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 629–635, Providence, Rhode Island, July. AAAI.

Mellish, Chris, Roger Evans, Lynne Cahill, Christine Doran, Daniel Paiva, Mike Reape, Donia Scott, and Neil Tipper. 2000. A representation for complex and evolving data dependencies in generation. In *Proceedings of the 6th Applied Natural Language Processing Conference (ANLP'00)*, Seattle.

Mittal, Vibhu, Johanna Moore, Guiseppe Carenini, and Steven Roth. 1998. Describing complex charts in natural language: A caption generation system. *Computational Linguistics*, 24(3):431–468.

Moore, Johanna and Martha Pollack. 1992. A problem for rst: The need for multi-level discourse analysis. *Computational linguistics*, 18(4):537–544.

Moser, Megan and Johanna Moore. 1996. Towards a synthesis of two accounts of discourse structure. *Computational linguistics*, 22(3):409–419.

Nunberg, Geoff. 1990. *The Linguistics of Punctuation*. Number 18 in CSLI Lecture Notes. CSLI Publications, Stanford, CA.

Oates, Sarah. 2001. Generating multiple discourse markers in text. Master's thesis, University of Brighton. Also available as ITRI Technical Report ITRI-01-25.

Paris, Cecile, Keith Vander Linden, Marcus Fischer, Anthony Hartley, Lynne Pemberton, Richard Power, and Donia Scott. 1995. A support tool for writing multilingual instructions. In *Proceedings of the Fourteenth International Joint Conference in Artificial Intelligence (IJCAI-95)*, pages 1398–1404. Also available as a technical report ITRI-95-11 at http://www.itri.bton.ac.uk/techreports/.

Pierrehumbert, Janet. 1980. *The Phonology and Phonetics of English Intonation*. Ph.D. thesis, MIT.

Power, Richard. 2000. Planning texts by constraint satisfaction. In *Proceedings of the 18th International Conference in Computational Linguistics (COLING)*, pages 642–648, Saarbrücken.

Power, Richard and Nico Cavallotto. 1996. Multilingual generation of administrative forms. In *Proceedings of the 8th International Workshop on Natural Language Generation*, pages 17–19, Herstmonceux Castle, UK.

Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, London.

Reiter, Ehud. 1994. Has a consensus NL generation architecture appeared, and is it psycholinguistically plausible? In *The Proceedings of the Seventh International Workshop on Natural Language Generation (INLGW-1994)*, pages 163–170, Kennebunkport, Maine, USA.

Reiter, Ehud and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge.

Rosner, Dietmar and Manfred Stede. 1992. Customizing RST for the automatic production of technical manuals. In E.H. Hovy R. Dale, D. Rosner and O. Stock, editors, *Aspects of Automated Natural Language Generation*, pages 199–214, Heidelberg, Germany. Springer-Verlag.

Schriver, Karen. 1997. *Dynamics in Document Design: Creating Text for Readers*. Wiley Computer Publishing, New York.

Scott, Donia and Clarisse de Souza. 1990. Getting the message across in RST-based text generation. In R. Dale, C. Mellish, and M. Zock, editors, *Current Research in Natural Language Generation*, volume 4 of *Cognitive Science Series*. Academic Press, New York, pages 47–73.

Scott, Donia and Cecile Paris. 1995. Identifying the mapping of semantics onto language: Going beyond the text. In *Working Papers of the AAAI Symposium on Empirical Methods in Discourse Interpretation and Generation*, Stanford University, March.

Scott, Donia and Richard Power. 2001. Generating textual diagrams and diagrammatic texts. In H. Bunt and R-J. Beun, editors, *Cooperative Multimodal Communication*, number 2155 in Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, pages 13–29. Also available as ITRI Technical Report ITRI-01-02.

't Hart, Johan, René Collier, and Antonie Cohen. 1990. *A Perceptual Study of Intonation*. Cambridge Studies in Speech Science and Communication. Cambridge University Press, Cambridge, U.K.

Yekovich, Frank, Carol Walker, and Harold Blackman. 1979. The role of presupposed and focal information in integrating sentences. *Journal of Verbal Learning and Verbal Behavior*, 18:535–548.

## A The solutions found for the example in Section 6.1

Number of versions = 58

**Version 1 (0)**
Elixir contains gestodene. However, the medicine has been thoroughly tested and it has no significant side effects; consequently, Elixir is safe to use.

**Version 2 (0)**
Elixir contains gestodene; however, since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use.

**Version 3 (0)**
Elixir contains gestodene. However, since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use.

**Version 4 (0)**
Elixir contains gestodene; however,

- the medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use.

**Version 5 (0)**
Elixir contains gestodene. However,

- the medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use.

**Version 6 (0)**
Elixir contains gestodene. However,

- the medicine has been thoroughly tested
- it has no significant side effects;

consequently, Elixir is safe to use.

**Version 7 (1)**
Elixir contains gestodene. However, the medicine has been thoroughly tested and it has no significant side effects. Consequently, Elixir is safe to use.

*Lost rhetorical grouping*

**Version 8 (1)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested and it has no significant side effects. Consequently, Elixir is safe to use.

*Single-sentence paragraph*

**Version 9 (1)**
Although Elixir contains gestodene since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use.

*Lost rhetorical grouping*

**Version 10 (1)**
Although Elixir contains gestodene

- the medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use.

*Lost rhetorical grouping*

**Version 11 (1)**
Elixir contains gestodene but since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use.

*Lost rhetorical grouping*

**Version 12 (1)**
Elixir contains gestodene but

- the medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use.

*Lost rhetorical grouping*

**Version 13 (1)**
Elixir contains gestodene; however, Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects.

*Nucleus precedes satellite*

**Version 14 (1)**
Elixir contains gestodene. However, since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use.

*Lost rhetorical grouping*

**Version 15 (1)**
Elixir contains gestodene. However, Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects.

*Nucleus precedes satellite*

**Version 16 (1)**
Elixir contains gestodene; however, the medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use.

*Lost rhetorical grouping*

**Version 17 (1)**
Elixir contains gestodene. However, the medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use.

*Lost rhetorical grouping*

**Version 18 (1)**
Elixir contains gestodene; however, the medicine has been thoroughly tested and it has no significant side effects; consequently, Elixir is safe to use.

*Lost rhetorical grouping*

**Version 19 (1)**
Elixir contains gestodene; however,

- the medicine has been thoroughly tested
- it has no significant side effects;

consequently, Elixir is safe to use.

*Lost rhetorical grouping*

**Version 20 (1)**
Elixir contains gestodene; however, since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use.

*Lost rhetorical grouping*

**Version 21 (1)**
Elixir contains gestodene. However,

- the medicine has been thoroughly tested
- it has no significant side effects.

Consequently, Elixir is safe to use.

*Lost rhetorical grouping*

**Version 22 (1)**
Elixir contains gestodene.
However,

- the medicine has been thoroughly tested
- it has no significant side effects.

Consequently, Elixir is safe to use.

*Single-sentence paragraph*

**Version 23 (2)**
Elixir contains gestodene but since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 24 (2)**

- The medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use although Elixir contains gestodene.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 25 (2)**
Elixir contains gestodene but the medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 26 (2)**
Elixir contains gestodene but Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 27 (2)**
Elixir contains gestodene; however, Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 28 (2)**
Although Elixir contains gestodene since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 29 (2)**
Elixir contains gestodene. However, Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 30 (2)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested and it has no significant side effects; consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*

**Version 31 (2)**
Although Elixir contains gestodene the medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 32 (2)**
Elixir contains gestodene. However, the medicine has been thoroughly tested; it has no significant side effects; consequently, Elixir is safe to use.

*Lost rhetorical grouping*
*Oversimple text-clauses*

**Version 33 (2)**
Elixir contains gestodene.
However, since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*

**Version 34 (2)**
Elixir contains gestodene. However, the medicine has been thoroughly tested; it has no significant side effects. Consequently, Elixir is safe to use.

*Lost rhetorical grouping*
*Oversimple text-clauses*

**Version 35 (2)**
Elixir contains gestodene. However, the medicine has been thoroughly tested. It has no significant side effects. Consequently, Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 36 (2)**
Elixir contains gestodene.
However,

- the medicine has been thoroughly tested
- it has no significant side effects

so Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*

**Version 37 (2)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested; it has no significant side effects. Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Oversimple text-clauses*

**Version 38 (2)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested. It has no significant side effects. Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*

**Version 39 (2)**
Since

- the medicine has been thoroughly tested
- it has no significant side effects

Elixir is safe to use although Elixir contains gestodene.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 40 (2)**
Elixir contains gestodene.
However,

- the medicine has been thoroughly tested
- it has no significant side effects;

consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*

**Version 41 (2)**
Although Elixir contains gestodene Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 42 (3)**
Elixir contains gestodene.
However, Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects.

*Single-sentence paragraph*
*Nucleus precedes satellite*
*Single-sentence paragraph*

**Version 43 (3)**
Since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use although Elixir contains gestodene.

*Nucleus precedes satellite*
*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 44 (3)**
The medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use although Elixir contains gestodene.

*Nucleus precedes satellite*
*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 45 (3)**
Elixir contains gestodene.
However, since the medicine has been thoroughly tested and it has no significant side effects Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*
*Lost rhetorical grouping*

**Version 46 (3)**
Although Elixir contains gestodene Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 47 (3)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested. It has no significant side effects.
Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*
*Single-sentence paragraph*

**Version 48 (3)**
Elixir is safe to use since

- the medicine has been thoroughly tested
- it has no significant side effects

although Elixir contains gestodene.

*Nucleus precedes satellite*
*Nucleus precedes satellite*
*Lost rhetorical grouping*

**Version 49 (3)**
Elixir contains gestodene but Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects.

*Nucleus precedes satellite*
*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 50 (3)**
Elixir contains gestodene; however, the medicine has been thoroughly tested; it has no significant side effects; consequently, Elixir is safe to use.

*Lost rhetorical grouping*
*Lost rhetorical grouping*
*Oversimple text-clauses*

**Version 51 (3)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested and it has no significant side effects so Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*
*Lost rhetorical grouping*

**Version 52 (3)**
Elixir contains gestodene.
However,

- the medicine has been thoroughly tested
- it has no significant side effects.

Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*
*Single-sentence paragraph*

**Version 53 (4)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested; it has no significant side effects; consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Single-sentence paragraph*
*Lost rhetorical grouping*
*Oversimple text-clauses*

**Version 54 (4)**
Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects although Elixir contains gestodene.

*Nucleus precedes satellite*
*Nucleus precedes satellite*
*Lost rhetorical grouping*
*Lost rhetorical grouping*

**Version 55 (4)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested and it has no significant side effects.
Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*
*Single-sentence paragraph*
*Single-sentence paragraph*

**Version 56 (4)**
Elixir contains gestodene.
However, Elixir is safe to use since the medicine has been thoroughly tested and it has no significant side effects.

*Single-sentence paragraph*
*Nucleus precedes satellite*
*Single-sentence paragraph*
*Lost rhetorical grouping*

**Version 57 (5)**
Elixir contains gestodene.
However, the medicine has been thoroughly tested; it has no significant side effects.
Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*
*Single-sentence paragraph*
*Oversimple text-clauses*
*Single-sentence paragraph*

**Version 58 (6)**
Elixir contains gestodene.

However, the medicine has been thoroughly tested.

It has no significant side effects.

Consequently, Elixir is safe to use.

*Single-sentence paragraph*
*Lost rhetorical grouping*
*Lost rhetorical grouping*
*Single-sentence paragraph*
*Single-sentence paragraph*
*Single-sentence paragraph*