Towards Constructive Text, Diagram, and Layout Generation for Information Presentation

John Bateman*
Bremen University

Thomas Kamps†
Intelligent Views, Darmstadt

Jörg Kleinz†
GMD-IPSI, Darmstadt

Klaus Reichenberger†
Intelligent Views, Darmstadt

Combining elements appropriately within a coherent page layout is a well-recognized and crucial aspect of sophisticated information presentation. The precise function and nature of layout has not, however, been sufficiently addressed within computational approaches; attention is often restricted to relatively local issues of typography and text-formatting, leaving broader issues of layout unaddressed. In this paper we focus on the selection and function of layout in pages that appropriately combine textual and graphical representation styles to yield coherent presentation designs. We demonstrate that layout offers a rich resource for achieving presentational coherence, alongside more traditional resources such as text-formatting and the text-internal marking of discourse connections. We also introduce an integrated approach to layout, text, and diagram generation. Our approach is developed on the basis of a preliminary empirical investigation of professionally produced layouts, followed by implementation within a prototype information system in the area of art history.

1. Introduction

The desirability of combining text, layout, graphics, diagrams, punctuation, and typesetting in order to present information most effectively is uncontroversial—indeed, in traditional graphic design and publishing, they could scarcely be conceived of as separate. It is therefore natural that *computational* attempts to synthesize texts, diagrams, and layout automatically should also now converge. In this paper, we argue that effective and coherent information presentation is best supported by adopting a common framework for physical layout and language/diagram generation. Whereas previous research has made this point convincingly for graphical and textual representations—particularly, for example, in the WIP (André et al. 1993), COMET (Feiner and McKeown 1993), and SAGE (Kerpedjiev et al. 1997; Green, Carenini, and Moore 1998) systems—we take this further and demonstrate that the same commonalities extend to include overall **page layout**, an area that has not previously received sufficient attention.

* FBIO, Sprach-und Literaturwissenschaften, University of Bremen, 28334 Breman, Germany, E-mail: bateman@uni-bremen.de.
† Intelligent Views GmbH, Julius-Reiber-Str. 17 G423 Darmstadt, Germany, E-mail: {t.kamps, j.kleinz, k.reichenberger}@i-views.de.

The paper focuses on two aspects of automatic information presentation new in our work:

- a general mechanism for organizing presentations around informational regularities in the data to be expressed—the regularities then inform the presentational strategies used for natural language, diagram, and layout generation;
- the construction of an indirect relationship between structured communicative intentions (typically represented in both mono- and multimodal work by some kind of rhetorical structure) and their expression in page layout.

The former allows us to ensure broad consistency of perspective and informational organization across elements presented using different media—e.g., across diagram and text; the latter allows us to draw closer to the kind of sophisticated layout that is observable in human-produced presentations.

We organize the paper as follows. We first introduce the mechanism for data-driven aggregation that we have developed, since this underlies our approach to both natural language generation and diagram design (Section 2). We then sketch the place of layout as an organizing framework within our approach as a whole (Section 3), setting out by means of examples some of the issues focused upon in the empirical investigation (Section 4). We then summarize the results of the empirical study in terms of an abstract specification for performing page layout (Section 5) and provide a first illustration of its application within the prototype multimodal information-presentation system DArt$_{bio}$ (Dictionary of Art: biographies) (Section 6). We conclude by summarizing the main contributions of our work and some of the follow-up research and development to which it is now leading (Section 7).

2. Data-driven Aggregation for Visualization and Natural Language Generation

It is commonly recognized in work on multimodal information presentation that much of the true value of such presentations lies in appropriate juxtapositions of non-identical but overlapping information. Textual presentations and graphical presentations have differing strengths and weaknesses and so their combination can achieve powerful synergies. Conversely, simply placing textual and graphical information together is no guarantee that one view is supportive of another: if the perspective on the data taken in a graphic and that taken in a text have no relation (or, worse, even clash), then the result is incoherence rather than synergy—cf. the discussions by authors such as Arens, Hovy, and Vossers (1993), Fasciano and Lapalme (1996), Green et al. (1998), and Fasciano and Lapalme (2000).

One means of ensuring mutually compatible presentations across modes is to drive both the language and the graphic generation from the same communicative intentions. If an automatic natural language generator and an automatic graphic generator both receive the task of expressing broadly similar, or compatible, intentions then there is a good chance that the resulting presentations will also be perceived to be compatible and mutually supportive. This has been used to good affect in systems such as CGS (Caption Generation System) of Mittal et al. (1998), where it is clearly crucial that the text and the graphic be in close correspondence. Another, in some ways related, approach is to derive both the graphic and textual elements from different components of a single presentation plan: thus, for example, one part of the presentation plan might express textually an instruction that must be carried out (*turn the dial*), while another part of the plan elaborates on that instruction by showing a diagram in which the location of the action to be performed is identified graphically. This has been explored extensively in systems such as WIP (André et al. 1993), PPP (André, Rist, and Müller 1998), and COMET (Feiner and McKeown 1993).

While both of these approaches are essentially *top-down*, or goal driven, effective presentations can also be produced by responding to regularities found in the data to be presented. Such regularities are difficult to predict as they are strongly contingent on what set of data happens to have been selected. "Data-driven" methods of this kind are commonly found in automatic visualization, where the goal is to present users with some comprehensible view of large collections of data. Utilizing regularities in the data is essential for effective visualization. In previous work (Reichenberger, Kamps, and Golovchinsky 1995), a set of techniques for generative diagram design were developed for precisely this task, i.e., for presenting *overviews* of datasets. We subsequently recognized that this approach also has applications to the task of **aggregation** in natural language generation, and we thus adapted it for use across both textual and graphical presentation modes. This provides a further technique for ensuring consistency between graphical and textual presentations—if both the graphical and textual presentations express the same regularities, or redundancies, that have been found in a dataset, then they are necessarily compatible in this respect. This allows us to use contingent data-driven organizations for generating information while nevertheless preserving coherent and mutually supportive views across presentation modalities.

2.1 Data-driven Aggregation: the Mechanism

The original generative diagram-design algorithms developed by Reichenberger, Kamps, and Golovchinsky (1995) built on the landmark work of Mackinlay (1986). Here, a data-classification algorithm flexibly links relational data with elements of a graphical language. These elements are allocated particular degrees of **expressiveness** so that appropriate graphical resources can be selected as required to capture the data being described. Reichenberger et al. extended this approach by employing a general type hierarchy of data properties to determine algorithmically the *most specific property subtype* (e.g., transitive, acyclic directed graph, inclusion, etc.) that accurately describes a dataset to be visualized. This subtype allows in turn selection of the particular forms of diagrammatic representation (e.g., trees, nested boxes, directed arrows, etc.) that are expressively adequate, but not over-expressive, for that dataset.

The theoretical basis of these methods is given in detail in Kamps (1997; 1998). They rest on a new application of Formal Concept Analysis (FCA) (Wille 1982). FCA is an applied mathematical discipline based on a formal notion of concepts and concept hierarchies and allowing the exploitation of mathematical reasoning for conceptual data analysis and processing. In particular, FCA permits the efficient construction of dependency lattices that effectively represent the functional and set-valued dependencies established among the domains of some data relation. Such dependency lattices can then motivate the differential selection of appropriate graphical presentations.

FCA starts from the notion of a formal context (G, M, I) representing a dataset in which G is a set of objects, M is a set of attributes, and I establishes a binary relation between the two sets. $I(g, m)$ is read *object g has property m* where $g \in G$ and $m \in M$. Such a context is called a **one-valued context**. For illustration, we draw on the domain of the DArt$_{bio}$ system that we discuss below: an example of a one-valued context corresponding to the attribute *Profession* for a set of artists is shown in the table to the left of Figure 1. **Concepts** in FCA are defined in accordance with the

	Architect	Designer	Urban Planner
Gropius	X		X
Breuer	X		X
A. Albers		X	
J. Albers			X
Moholy-Nagy			X
Hilberseimer	X		

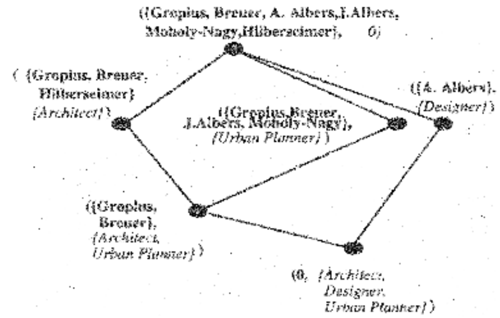


Figure 1
Example of a one-valued context and its corresponding lattice.

traditional theory of concepts, and consist of an **extension** and an **intension**. The extension is a subset A of the set of objects G and the intension is a subset B of the set of attributes M. We call the pair (A, B) a **formal concept** if each object of the extension has all the properties of the intension. Thus, for the data shown in Figure 1, the pair ({Gropius, Breuer}, {Urban Planner, Architect}) represents a formal concept: each of the members of the extension possesses all the attributes mentioned in the intension. The set of all concepts for some formal context can be computed effectively using the Next Closure algorithm developed by Ganter and Wille (1996).

The main theorem of concept analysis then shows that the set of concepts for a formal context can be organized into a complete lattice structure under the following definition of the **subconcept** relation: a concept (A, B) is a subconcept of (A^*, B^*) if and only if $A \subseteq A^* \Leftrightarrow B^* \subseteq B$ (Wille 1982). The concept lattice may be constructed by starting from the top concept (the one that has no superconcepts) and proceeding top-down recursively. In each step we compute the set of direct subconcepts and link them to the respective superconcept until we reach the greatest lower bound of the lattice itself (the existence of which is always guaranteed for finite-input data structures). An efficient implementation of this algorithm is given in Kamps (1997). The lattice corresponding to our example one-valued context is given to the right of Figure 1. This lattice shows the full labeling of formal concepts in order to ease comparison with the originating table. Much of this information is redundant, however, and so we generally use variations on the abbreviated, more concise, form shown in Figure 2. Such lattices naturally capture similarities and differences between the values of the specified attributes of objects: each concept of the lattice indicates objects with some set of values in common. Moreover, the generalizations are organized by subsumption, which supports the selection of most-specific subtypes.

When considering datasets in general, we typically need to express more information than that of single attributes and for this we require **multi-valued contexts**. An example of a multi-valued context is shown in Table 1, which includes our previous one-valued context as one of its columns; for ease of discussion, however, we will for the present restrict the *Profession* attribute so that each artist has only one profession. The table shows the subject areas/professions, institutions, and time periods in which the indicated artists were active. Formally, a multivalued context is a generalisation of a one-valued context and may be represented as a quadruple (G, M, W, I) where G, M, and I are as before, and W represents the set of values of the attributes—

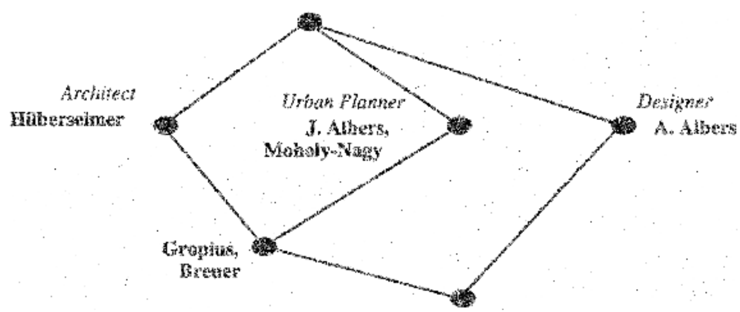


Figure 2
Concept lattice example, more succinctly labeled. Here, the extension label for each node consists of just those elements which are *added* at that node moving *up* the lattice; conversely the members of the intensions are shown moving *down* the lattice, again adding just those elements that are new for that node. For example, the node labeled simply *Gropius, Breuer* corresponds to the full form ({Gropius, Breuer}, {Architect, Urban Planner}) since both Gropius and Breuer are newly added to the extension at that node, while no new elements are added to the intension—'Architect' and 'Urban Planner' are both inherited from above.

Table 1
A collection of facts concerning artists and their professions drawn from the frame-based domain model used for the Dictionary of Art: biographies and re-expressed as a table of facts and attributes. (The facts are for illustrative purposes only and should not be taken as reliable statements of art history!)

	Person	Profession	School	Workperiod
g1	Gropius	Architect	Harvard	1937–1951
g2	Breuer	Architect	Harvard	1937–1946
g3	A. Albers	Designer	Black Mountain College	1933–1949
g4	J. Albers	Urban Planner	Black Mountain College	1933–1949
g5	Moholy-Nagy	Urban Planner	New Bauhaus	1937–1938
g6	Hilberseimer	Architect	Illinois Institute of Technology	1938–1967

which are, in contrast to the one-valued-context case, not trivially either true or false, applicable or not. To identify the value $w \in W$ of attribute $m \in M$ for an object $g \in G$, we adopt the notation $\mathsf{m(g)} = \mathsf{w}$ and read this as *attribute m of object g has value w*.

Kamps (1997) renders multivalued contexts amenable to the techniques for dependency-lattice construction by deriving a one-valued context that captures the **functional dependencies** of the original multivalued context. To see how this works, we first note that a functional dependency in a relation table is established when the following implication is always true: for two arbitrary objects $g, h \in G$ and two domain sets $D, D^* \in M$, then $D(g) = D(h) \Rightarrow D^*(g) = D^*(h)$. This implication suggests the following construction for an appropriate one-valued dependency context: for the set of objects take the set of subsets of two elements of the given multi-valued context $P_2(G)$; for the set of attributes take the set of domains M; and for the connecting incidence relation take $I_N(\{g,h\}, m) :\Leftrightarrow m(g) = m(h)$. The required dependency context is then represented by the triple $(P_2(G), M, I_N)$. This is illustrated in the table to the left of Figure 3, which shows the one-valued context corresponding to the multivalued context of Table 1. An entry here indicates that the identified attribute has the same

	Person	Profession	School	Workperiod
g1g2		X	X	
g1g6		X		
g2g6		X		
g3g4			X	X
g4g5		X		

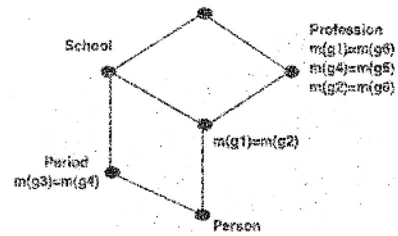


Figure 3
Example dependency context and corresponding lattice.

value for both the facts identified in the object labels of the leftmost column: for example, g1 and g2 share the values of their Profession and School attributes. This provides a wholistic view of the dependency structure of the original data and is, moreover, computationally simple to achieve.

It is then straightforward to construct a dependency lattice as described above; this is shown to the right of Figure 3. The arcs in this lattice now represent the functional dependencies between the involved domains, and the equalities (e.g., m(g1)=m(g2)) represent the redundancies present in the data. For example, the lower left node labeled *Period* indicates not only that the third- and fourth-row entries under *Period* (g3 and g4) are identical but also, following the upward arc, that these entries are equal with respect to *School*; similarly, following upward arcs, the middle node (m(g1)=m(g2)) indicates that the first- and second-row entries (e.g., g1 and g2) are equal with respect to both *School* and *Profession*. The lattice as a whole indicates that there are functional relationships from the set of persons into the set of professions, the set of periods, and the set of schools. A further functional relationship exists from the set of periods into the set of schools.

Once such a lattice has been constructed, we also have as a consequence a set of classifications of the original relational input, or dataset. This can directly drive visualization as follows. For graphics generation, it is important that all domains of the relation become graphically encoded: this means the encoding is complete. Kamps (1997) proposes a corresponding graphical encoding algorithm that starts encoding the bottom domain and walks up the lattice employing a bottom-up/left-to-right strategy for encoding the upper domains. The idea of this model, much abbreviated, is that the cardinality of the bottom domain is the largest, whereas the domains further up in the lattice contain fewer elements. Thus, the bottom domain is graphically encoded using so-called **graphical elements** (rectangle, circle, line, etc.), whereas the upper domains are encoded using **graphical attributes** (color, width, radius) and **set-valued attributes** that must be attached to graphical elements. In general, it is preferable to maximize graphical attributes over set-valued attributes as this keeps graphical complexity moderate.

Figure 4 shows two example diagrams that are produced from the dataset of Table 1 via the dependency lattice shown to the right of Figure 3. Informally, from the lattice we can see directly that artists (*Person*) can be classified on the one hand according to work period (following the lefthand arc upwards) and, on the other hand, *jointly* according to school and profession (following the vertical arc). The algorithm first allocates the attribute *Person*, indicated in the lowest node of the lattice, to the basic graphical element *rectangle*; the individual identities of the set members are given by a graphical attachment: a string giving the artist's name. The functional relationship between the set of persons and the set of time periods is then represented by the further

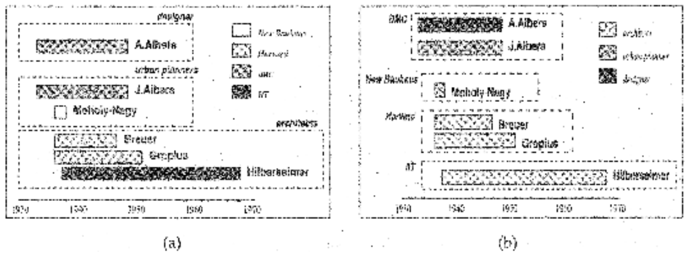


Figure 4
Example diagrams generated for the example data. Alternatives are produced by two distinct traversals of the aggregation lattice.

graphical attribute of the *length* of the rectangle. This is motivated by the equivalence of the properties of temporal intervals in the data and the properties of the graphical relationship of spatial intervals on the page. Two paths are then open: following the functional relationship first to either a set of schools or to a set of professions. Diagram (a) in Figure 4 adopts the first path and encodes the school relationship by means of the further graphical attribute of the *color* of the rectangle, followed by a nesting rectangle for the relationship to Professions; diagram (b) illustrates the second path, in which the selection of graphical encodings is reversed. Both the selection of color and of nesting rectangles are again motivated by the correspondence between the formal properties of the graphical relations and those of the dependencies observed in the data. Reinstating the multiple professions of Gropius and Breuer mentioned in Figure 1 gives rise to a rather different dependency lattice in which the second solution is no longer possible.

All of these mechanisms were implemented and used extensively for visualization in the context of an Editor's Workbench for supporting editorial decisions during the design of large-scale publications such as encyclopedias (Rostek, Möhr, and Fischer 1994; Kamps et al. 1996).[1]

2.2 The Partial Equivalence of Diagram Design and Text Design

A selection of particular graphical elements entails the expression of particular functional dependencies. This is similar to decisions that need to be made when generating text. For instance, the equality $\mathsf{m}(\mathsf{g1}) = \mathsf{m}(\mathsf{g2})$ in the lattice of Figure 3 above can also motivate a particular grouping of information in a corresponding linguistic presentation. That is, whereas graphically the equality motivates an association of both Gropius and Breuer with the graphical attributes allocated to Professions and Schools, textually we may connect both artists in a single sentence: i.e., g1 (concerning Gropius) and g2 (concerning Breuer) can be compactly expressed by collapsing their (identical) school and profession attributes:

> Both Gropius and Breuer were architects and taught at Harvard.

A similar phenomenon holds for the grouping $\mathsf{m}(\mathsf{g3}) = \mathsf{m}(\mathsf{g4})$; here, g3 (concerning A. Albers) and g4 (concerning J. Albers) may be succinctly expressed by collapsing their identical period and school attributes.

1 Initially developed within the European Union Research and Development in Advanced Communications Technology in Europe (RACE) project 2042 EUROPUBLISHING (Hüser et al. 1995).

Combining these considerations motivates the following approximate textual re-rendering of diagram (b):

> Anni Albers (who was a designer) and J. Albers (who was an urban planner) both taught at the BMC from 1933 until 1949. Moholy-Nagy (who was also an urban planner) taught from 1937 until 1938 at the New Bauhaus. Gropius and Breuer (both architects) were, at partially overlapping times (1937–1951 and 1937–1946 respectively), at Harvard. Hilberseimer (who was an architect too) taught at the IIT from 1938 until 1967.

A textual re-rendering of diagram (a) would reflect the contrasting groupings entailed there: i.e., Breuer, Gropius, and Hilberseimer would be grouped at top level whereas the two Albers would not.

A dependency lattice extracts partial commonalities that remain constant over subsets of the data to be presented and this is closely related to the problem of aggregation in NLG (cf., Dalianis [1999]). The functional redundancies captured by the lattice construction are precisely those redundancies that indicate opportunities for structurally induced aggregation. Selecting a particular graphical element or attribute to realize some aspect of the data is in fact an aggregation step. In Bateman et al. (1998), we have shown this in terms more familiar to NLG by re-interpreting in dependency lattice terms some of the standard examples of aggregation discussed in the literature. Below, we show that mutual consistency between textual fragments produced by our NLG component and graphical elements produced by the automatic visualization component can be enforced by driving both from a common dependency lattice.

3. Preliminaries for Layout: Inputs and Outputs for the Layout Determination Task

Page layout, more properly termed **typographic design**, is usually divided into three levels: **microtypography**, **macrotypography** (layout proper), and **style**. Here we are most concerned with macrotypography—the segmentation of a page of information into more or less closely related "visual blocks." Macrotypography is a central component of professional document design; indeed,

> Every designer knows that how elements are put together on a page communicates a powerful message (Adobe Inc., InDesign product information sheet).

Unfortunately, with some valuable exceptions (cf., for example Schriver [1996], Waller [1988] and Bernhardt [1985]), the professionals do not then go on to tell us just what that message might be.

Our starting point for investigating layout and its message rests on the fact that layout is not a fixed property of information presentation; i.e., similar information can be subjected to diverse layouts. We then assume, following Schriver (1996) and others, that layout decisions should be functionally motivated in terms of a presentation's communicative purposes. We illustrate this further, while at the same time setting the scene for our empirical investigation, by briefly considering the kinds of layout variation that are commonly found. We do this in two steps. First, we characterize more finely the notion of layout as such; then, we consider how selections among possible layouts may be motivated.

3.1 Layout structure

Issues of layout were already present within the visualization framework discussed above. For example, the relationship of the graphical blocks representing particular artists, or the positioning of the diagrams' legends with respect to the diagrams themselves, all involve decisions of layout. The solution developed as part of the automatic visualization component used in the Editor's Workbench was to consider layout itself as a particular class of diagrams, with their own particular properties and concerns. An automatic page-layout component (APALO) was accordingly implemented as a specialization of the general visualization task.

Fully specified layout diagrams specify the physical placement and appearance of elements on a page. In order to generalize across such layouts we define an abstract level of representation called **Layout Structure**. Layout structure abstracts across the precise details of physical layouts to focus on classes of layouts that are visually "equivalent." Visually equivalent layouts suggest the same page blocks, with similar inter-block relationships of perceived prominence and similarity.

Our view of layout structure draws heavily on Southall (1992), who defines a restricted set of typographical relation types. These include: **containment**, i.e. recursive block structure; **reading order**, i.e., generally left-to-right, top-to-bottom reading paths in Western cultures; **similarity**, describing blocks that share some visual properties such as size, typeface selection, structure, etc.; and **reference**, where a connection between visual blocks is suggested by physical proximity. We represent layout structures in terms of a tree structure (representing containment) augmented by a restricted set of possible additional annotations corresponding to the remaining typographical relation types. The annotations thus serve either to further constrain the possible physical layouts that may render the layout structure, or to place mutual constraints on the rendering possibilities—for example, a type-equivalence annotation requires consistency in rendering decisions across the units declared to be type-equivalent.

A simple example of layout structure and its correspondence to a physical layout is shown in Figure 5. Here we see that annotations also provide a numerical summary of the information to be displayed in any layout element (which may be either descriptive or denote a target): for example, node 2.3.2 in the figure is annotated 403w+3p:50, indicating that it consists of a block of text with 403 words and 3 pictures, and is allocated an importance score of 50%.[2] These scores impose target visual weights for corresponding page elements (i.e., more important nodes should be more prominent, which can be achieved by larger surface area combined with less but more heavy type, by use of prominent colors, etc.). More information concerning layout structure and its motivation is given in Reichenberger et al. (1996).

Given a fully specified layout structure as input, APALO renders it as a physical page by mapping constituency to nested boxes (i.e., inclusion diagrams), and strength of connection and sequence to spatial displacement: the boxes included within an enclosing box are arrayed two-dimensionally to influence reading order. Typographic attributes, such as type size, specific type face within the family (bold, italic etc.), arrangement of the type (ragged right, flush matter, etc.), leading, coloring and orientation, are all assigned at this stage, respecting any constraints on presentation given in the abstract layout structure. Since it is rarely the case that a layout structure is so tightly specified that only one physical layout is possible, the implementation uses progressive refinement and allows a user either to stop the process at any point or

2 The numbering of the nodes starts with 2.3 to maintain ready comparability with the discussion in Reichenberger et al. (1996).

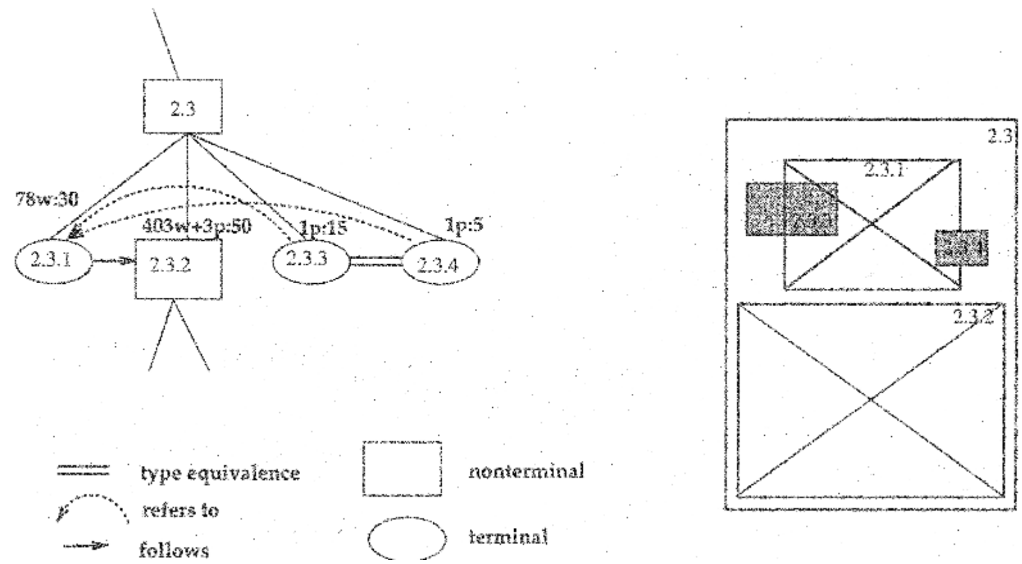


Figure 5
Example layout structure and its correspondence to a segment of page layout (adapted from Reichenberger et al. 1996). The "follows" annotation requires a page rendition where the reader will encounter 2.3.1 before 2.3.2, and the "refers to" links attract the type equivalent units 2.3.3 and 2.3.4 towards 2.3.1 without establishing a further visual block contrasting with 2.3.2.

to continue searching for better layouts until a stable state is achieved. The rendering aspect of APALO corresponds broadly to other components that have been designed for page layout (e.g., Feiner [1988] or Graf [1995]), but differs from these in that it is not restricted to any particular page model (e.g., a grid system).

Our subsequent considerations of layout motivation adopt fully specified layout structures as their target. Such structures will include presentation content as produced by the natural-language-generation and automatic-diagram-generation components in their terminal elements, and are responsible for enforcing communicatively effective layouts.

3.2 Communicative-Functional Structure and its Relationship to Layout Structure

For automatic information presentation we also require a representation of the communicative intentions to be fulfilled by a presentation. For this, we adopt "standard" rhetorical structure theory (RST) as set out in Mann and Thompson (1986). RST is selected for two reasons: first, it is one of the most elaborated and widely used forms of analysis of communicative intentions and has been applied to a wide variety of texts; and second, it is well established in NLG and has already been applied to multimodal information presentation (cf., André and Rist [1993]).

Originally, RST sought to describe the recursive structure of any text in terms of **rhetorical relations** which hold between the segments (called **spans**) of the text. Rhetorical relations are either symmetric (**multinuclear**), in which case the text spans related are considered of equal importance for the text, or assymetric, in which case one text segment among those related by a relation is singled out as being more essential to the writer's purposes (the **nucleus**) than the others (the **satellites**). RST defines itself as a functional theory, in that the segments related are functional rather than textual—i.e., a rhetorical relation need not have any specific grammatical or lexical

[1] There are many reasons that the Bauhaus movement spread to the U.S. [2] For example, more people became aware of their work in magazines. [3] Also, Bauhaus-designed objects came onto the market in increasing numbers. [4] But there were two main reasons. [5] First, many ex-Bauhaus members emigrated to the U.S. [6] And second, they started teaching Bauhaus methods in U.S. colleges.

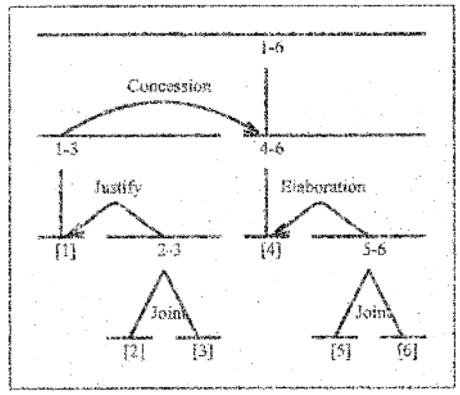


Figure 6
An example text representing both itself and its underlying propositions, and the corresponding communicative-intentional text plan represented in terms of RST.

realization. RST's functional orientation supported a further re-interpretation of RST in which rhetorical relations become plan operators for text construction (cf., Moore and Paris [1993]; Hovy [1993]). From this perspective, rhetorical structures represent a plan for achieving some goal via linguistic means. The information maintained at the leaves of such RST **text plan** structures is thus not pre-existing text, but rather chunks of information that are to receive textual realizations.

If, as is common in NLG, we consider the information-presentation task to involve the expression, or realization, of a text plan expressed according to RST and, further, we see the layout used in that information presentation as one of the decisions that needs to be made, then we face the question as to whether the text plan can also motivate the necessary layout decisions. In Figure 6, we show a simple invented text fragment and a corresponding RST analysis. We will use this text in two ways: First, it stands as a shorthand for a set of propositions from the knowledge base or domain model—during NLG, and following text planning, the rhetorical structure is as shown but the leaves identify the propositions indicated rather than textual elements. Second, the text stands as one possible *result* of the subsequent NLG process, where the propositions grouped in the text plan have already received their linguistic realization. This double use is harmless because we do not use the textual version to motivate any of the decisions taken during the NLG process whose task it is to produce it.

The RST structure states that the text divides into two main segments, span (1–3) and span (4–6), and that these are related by the rhetorical relation **concession**. This indicates that the main allegiance and point of the writer lies with the nucleus (span 4–6), but that, at the same time, the writer does not claim that the seemingly contradictory information in the satellite (span 1–3) does not hold. The principal purpose of using an RST-concession relationship is to strengthen one's own argument by showing that one is aware of a possible objection but are already able to dismiss it. Common linguistic markers of concession are *however*, *although* and, as adopted here, *but*. In the embedded trees, the satellite tree involves an RST **justify** relationship between a (relative) nucleus (1) and the span (2–3), the nucleus tree involves an **elaboration** relationship between a (relative) nucleus (4) and the span (5–6). Both spans (2–3) and (5–6) are examples of the RST **joint** schema: this simply combines the connected propositions into a set without assigning any difference in nuclearity among them. A justify relationship holds when the satellite gives information intended to justify the writer's right to present the information in the nucleus (i.e., in this case, the claim that there are many reasons for the Bauhaus spreading is backed up by the examples offered in (2) and (3)); and the elaboration relationship holds when further specifying information concerning some aspect of the nucleus is given. There are approximately 25 RST relations in the standard set and space precludes giving more details of them here: their definitions and examples of use are given, however, in numerous places in both the computational and (text-)linguistic literature.

The realization of an RST structure as a natural language text involves many important issues which we will also not address here. Instead, we focus on the relationship between an RST-style presentation plan, and layout decisions; in particular, given our goal of determining a layout structure that can then be rendered as a physical page layout, we ask how the information in a rhetorical structure may be placed in correspondence with appropriate layout structures. In the text of the previous figure, for example, one layout decision that could be made is that there is no particular layout to be done—this is then equivalent to a straightforward, monomodal NLG system producing text only. The text in the figure is already an example. Another possible decision, lying at the opposite extreme, is to say that *every node* of the RST structure should find some direct correspondence in a node of the layout structure. This would be to require that the entire constituency structure of the RST tree be signaled visually in the layout. While both options are possible, it should be clear that there are many others. For example, span (1–3) could be represented as a single layout element (i.e., a single visual block consisting of a text paragraph) with span (4–6) breaking down into, perhaps, a single layout element for the nucleus (6) and two further layout elements, one for each of the members of the satellite.

The situation is made yet more complex by the presence of a steady trade-off between information expressed via layout and information expressed via text. For example, if the spans (1–3) and (4–6) are allocated to distinct layout units, allowing them to vary independently of one another in terms of microtypography and physical placement, then it is considerably less likely that the RST concession relationship will be preserved linguistically. Similar considerations arise within the two spans: the more the RST structure is decomposed, resulting in typographical distinctions, the less use is made of explicit linguistic discourse marking. A compelling illustration of this can be seen in the complex graphic-text combination discussed in Kerpedjiev et al. (1998) in which a text fragment is re-expressed graphically.

This variability needs to be brought under motivated control. Information presented together, or in similar styles, is perceived as related regardless of whether this was intended by the page designer or not; conversely, information presented in different styles, or separated widely on the page, is interpreted as unrelated. Critical work in professional document design demonstrates that failure to respect such entailments of layout makes a page or diagram difficult to interpret and possibly misleading: see, for example, Schriver (1996) and the numerous references cited there. But it is also well known that not all of the possible details of document structure are normally presented in layout: the relationship is substantially more flexible. This presents a problem for any approach to multimodal presentation that adopts too close a relationship between presentation-plan elements and layout—as most previous systems have in fact done.

4. A Methodology for the Empirical Investigation of Communicative-Functional Page Layout

Given our goal of understanding more precisely what can happen between a specification of communicative intentions—expressed in terms of a presentation plan using a representation such as RST—and a fully specified layout structure that passes on intention-appropriate constraints to a page-rendering engine, we embarked on an empirical investigation of the kinds of variability that occur in real documents. For this, we developed a methodology for exploring the functional basis of page layout in general. Two caveats are required here: first, our experimental method was exploratory—as one of the first studies of its kind, we needed to respond with flexibility to the results of the analysis; and second, since our aim was to move quickly from first analysis results to prototype system in order to evaluate the feasibility and value of the entire scenario in practice, the study was deliberately restricted in scope.

4.1 Method of Analysis

The provision of appropriate multimedia corpora for supporting the kind of empirical analysis we required is still, several years later, very much in its infancy.[3] The main criteria for the selection of pages for our investigation were (a) that the entire page be concerned with a single topic, while nevertheless presenting various aspects of that topic by means of varied text structuring and typographical and layout decisions, and (b) that the page demonstrate "interesting" layout. This led us to consider pages drawn from popular magazines, since these exhibit highly varied typographical and layout decisions in the hope of being eye-catching and maintaining interest.

The detailed structure of our study was as follows. We took a set of pages selected according to criteria (a) and (b), and asked for each page why it was set out as it was. We answered this question by:

1. providing a layout structure representation for each page;
2. constructing a single "text" out of the entire "content" of the page (including headings, picture captions, pictures, etc.) that captured as far as possible the perceived purpose of the page;
3. performing a "functional text structure analysis" of the constructed text respecting the perceived purpose of the page (and therefore of the constructed text);
4. considering whether the page-layout structure could be derived straightforwardly from the text structure;
5. deriving and informally evaluating alternative "possible" layouts on the basis of a progressively refined set of layout principles.

We illustrate our approach by setting out in detail one round of analysis concerning a single illustrated page. The page used is shown in Figure 7 and is drawn from the German illustrated sport and health magazine *Fit for Fun* (1995 (5), 92). The article describes various aspects of the game *Unihoc*, presenting background information concerning how the game is played, where it is popular, and why it is popular, as well as some pointers to further information and the equipment needed to play it. The page was typical of a series of feature articles being run in the magazine at that time. Particular elements expressed using layout in the page are indicated in the figure: significant here is solely the physical layout of the page, not the content.

3 Two projects currently involved with gathering and annotating such data are ICONOCLAST (Bouayad-Agha, Scott, and Power 1996) and GEM (Delin, Bateman, and Allen, forthcoming). Corio and Lapalme (1998) have also presented results of a corpus study.

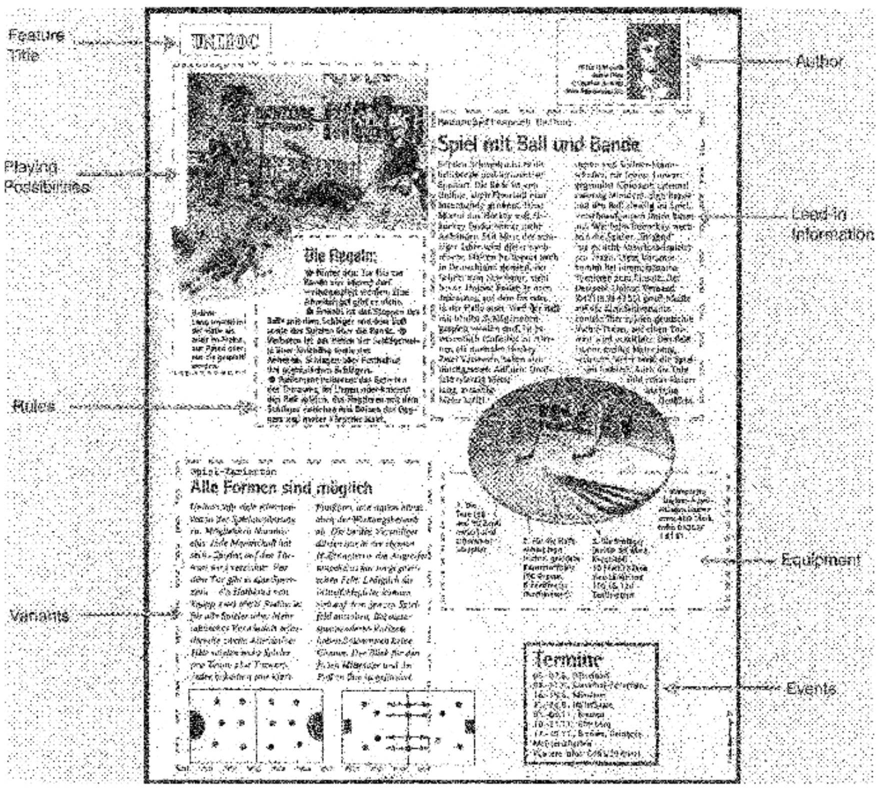


Figure 7
An original page for analysis. Source: 'Fit for Fun' **5**: p92 (1995). The page is annotated here to show the major layout units.

4.2 An Illustrative Round of Analysis: Preparation

The starting point for our analysis was provided by our initial page-selection criteria, i.e., that there is some body of material that the author(s) of the page wish to present. It is then relevant (and necessary) to ask how this information is to be broken up for effective presentation. To do this, we set out the content of each page as a single constructed text: The constructed text for the Unihoc page is given in Figure 8. Note that this text already reflects our understanding of the function of the page in its context of use: we assume that the page has the main functions of informing readers about a game that they might not be familiar with and telling them that the game is in fact finding increasing support both internationally and nationally.

This approximate, "pre-analytic" understanding of the page's communicative function is then made more explicit by providing a detailed functional text-structure analysis using RST. The RST structure offers a plausible presentation plan that can serve as a starting point for considering possible and contrasting realizations of this information as a two-dimensional page. The constructed text has to then be viewed in a similar way to the example fragment concerning the Bauhaus that we used in Section 3.2. In many respects, it is not a text at all, in that it overcommits to particular forms of linguistic expression and linear ordering. NLG uses of RST do not (in general) allow the RST structure to specify particular relative orderings between nuclei and their respective satellites. The ordering needs to be decided during generation so that issues of textual fluency can be addressed appropriately. The same then holds for our con-

[0] Astrid Frula, captain of the German National Unihoc team, writes: [1] Among the Swedes it is the most popular and best-known branch of sport. [2] We are talking about Unihoc, also called Floorball or Indoor Bandy. [3] This mixture of hockey and ice hockey is attracting ever more supporters. [4a] Since the middle of the eighties, this dynamic team sport has also been played in Germany and [4b] the step to becoming a school sport is imminent. [5] Unihoc can be played in the gym as well as outside, on grass or ice. [6a] Because the ball can be played with both sides of the stick, [6b] it is much easier to master than normal hockey. [7] One can continue playing behind the goal (four metres up to the board) and [8] there is no offside rule. [9] Stopping the ball with the stick and the foot is allowed, as well as playing via the board. [10] Raising the blade of the stick above knee height, or lifting, hitting, or holding the opponent's stick is not allowed. [11] Entering the goal area, playing the ball while lying or kneeling, moving the stick between the legs of the opponent, or engaging in hard body contact is also prohibited. [12] Unihoc allows many alternatives in how it is played. [13] One possibility: [14] each team has six players, and no goalie. [15a] In front of the goal there is a no-go area - [15b] no players are allowed within a semicircle of almost 2 meters in radius. [16] The second alternative requires more tactical insight: [17] here there are 6 players per team, plus a goalie. [18] Each receives a clear function, which determines their effective playing area. [19a] The two defenders may only act within their own half; [19b] in contrast, the two attackers may only play within their opponents' half. [20] Only the midfield players can run as they wish over the entire playing field. [21] In this exciting variant solo artists have no chance. [22] Spotting a free team member and passing the ball on are essential. [23] Two variants have become dominant. [24] On the large field (forty metres long, twenty metres wide) two six-person teams with fixed goalie oppose each other (Playing time: two times twenty minutes). [25a] A board keeps the ball continuously in play; [25b] rest periods hardly ever occur. [26] As in ice hockey, a player substituting for another does not lead to an interruption of play (up to eight substitute players per team). [27] This is the variant that is used in international matches. [28] The German Unihoc Union (0421/4984255) frequently goes back to the small field variant: [29] where mixed 4-person teams play without a goalie. [30] The playing field is only thirty metres long and sixteen metres wide, while the playing time is halved. [31] The goals are also smaller than on the large field. [32] The goals (60×90 centimeters) are collapsible. [33] In the gym, light holed plastic balls (20 grams, 8 centimeter diameter) are used. [34] The sticks (Kevlar 95 Mark, plastic 10 Mark) are 100 to 120 centimeters in length. [35] Complete sets of Unihoc equipment cost around 450 Marks. [36] Info: 05357/18181. [37] 05.-07.5., Düsseldorf, [38] 09.-11.6., Clausthal-Zellerfeld, [39] 16.-18.6., München, [40] 23.-25.6., Halle/Saale, [41] 03.-05.11., Bremen, [42] 10.-21.11., Göteborg, [43] 17.-19.11., Bremen, Deutsche Meisterschaften. [44] Further Info: 0421/23 94 01.

Additional graphical material:

- A: Astrid Frula (photograph): 'authorial voice'
- B: Player positions (diagram): two variations, B1 and B2
- C: Unihoc equipment (photograph)
- D: Unihoc being played on ice (photograph)
- E: Unihoc being played in the gym (photograph)

Figure 8
Constructed Unihoc text and graphical material used. Independent clauses (or major information units) are numbered for ease of reference and the graphical content of the page is also summarized.

structed text. Text spans corresponding to sibling nuclei and satellites are considered unordered; any ordering in the text is only partial, as given by the RST presentation plan.

This is doubly appropriate here because we cannot generally provide a unique linear text that re-expresses precisely the information that is given in an original two-dimensional page, and so this is not the constructed text's function here. The text simply provides a shorthand for the content that must necessarily be presumed to underlie the page's production. Moreover, it is necessary in any case to reconstruct the intended content of the page because the page itself is unsuited to stand as the basis for an RST analysis. A page's most salient features are visual—i.e., typographical—and part of our claim is that this is not directly indicative of rhetorical organization: relationships between visual blocks on the page are at a different level of abstraction than rhetorical relations. Assuming an equivalence of levels would probably lead to rhetorical analyses that primarily involve joint schemas conjoining the (many) top-level visual blocks of any page such as the Unihoc page; and at the same time it would leave the alternations across differing but communicatively similar layouts unexplained.

The RST analysis for the constructed Unihoc text, and hence of the page, is shown in Figure 9. The RST analyses that we use were arrived at following the standard techniques of cross-coder checking, consultation and consensus. The RST diagram follows the conventions illustrated above in Figure 6 but uses considerably more rhetorical relations. Again, space precludes listing the definitions of the RST relations found in the analysis, but the definitions employed are exactly as given in the literature. The analysis in the figure also includes the information presented in the original page as photographs or diagrams. These have been labeled alphabetically (A, B, etc.) as identified in Figure 8 and have been anchored into the RST-tree at appropriate places with plausible relations.[4]

The RST analysis makes our interpretation of the function of the text/page fully explicit. The central nucleus for the page as a whole is unit [3], i.e., *This mixture of hockey and ice hockey is attracting ever more supporters.* We are therefore considering the primary purpose of the page to be a statement that Unihoc is becoming very popular (and so the reader should be well up on it). The segments immediately following the nuclear span, [5] and [6], give some of the reasons why the sport is becoming so popular (with rhetorical relation **volitional cause**), and then segment [7–11] gives an overview of the do's and don'ts of the game. The main bulk of the constructed text consists of a concession that, although *Unihoc allows many alternatives in how it is played* [12], *two variants have become dominant* [23]. The existence of alternatives is supported by the two possibilities presented in segments [13–15] and [16–22], both related by the relation *evidence* to the nuclear [12]; the two main alternatives are *elaborate*d in the explicit *contrast* drawn in segments [24–27] and [28–31]. Finally, the segments [32–36] and [37–44] provide additional elaborating material concerning where and when the game can be seen and what equipment is necessary to play it.

A page such as this presents a considerable challenge for models of automatic layout. A closer indication of this complexity is provided by Table 2, which shows the correspondence between **informational** segments identified in the RST presentation

4 The analysis as shown is not a correct RST structure because it admits differing relations as satellites of a single nucleus: a strictly correct structure would need to show more intermediate segments in order to have one type of relation apply within each segment. Since this would have complicated the diagram even further, we have simplified for purposes of exposition, although the existence of further structure simplifies the task of the layout-structure construction specification we give below.

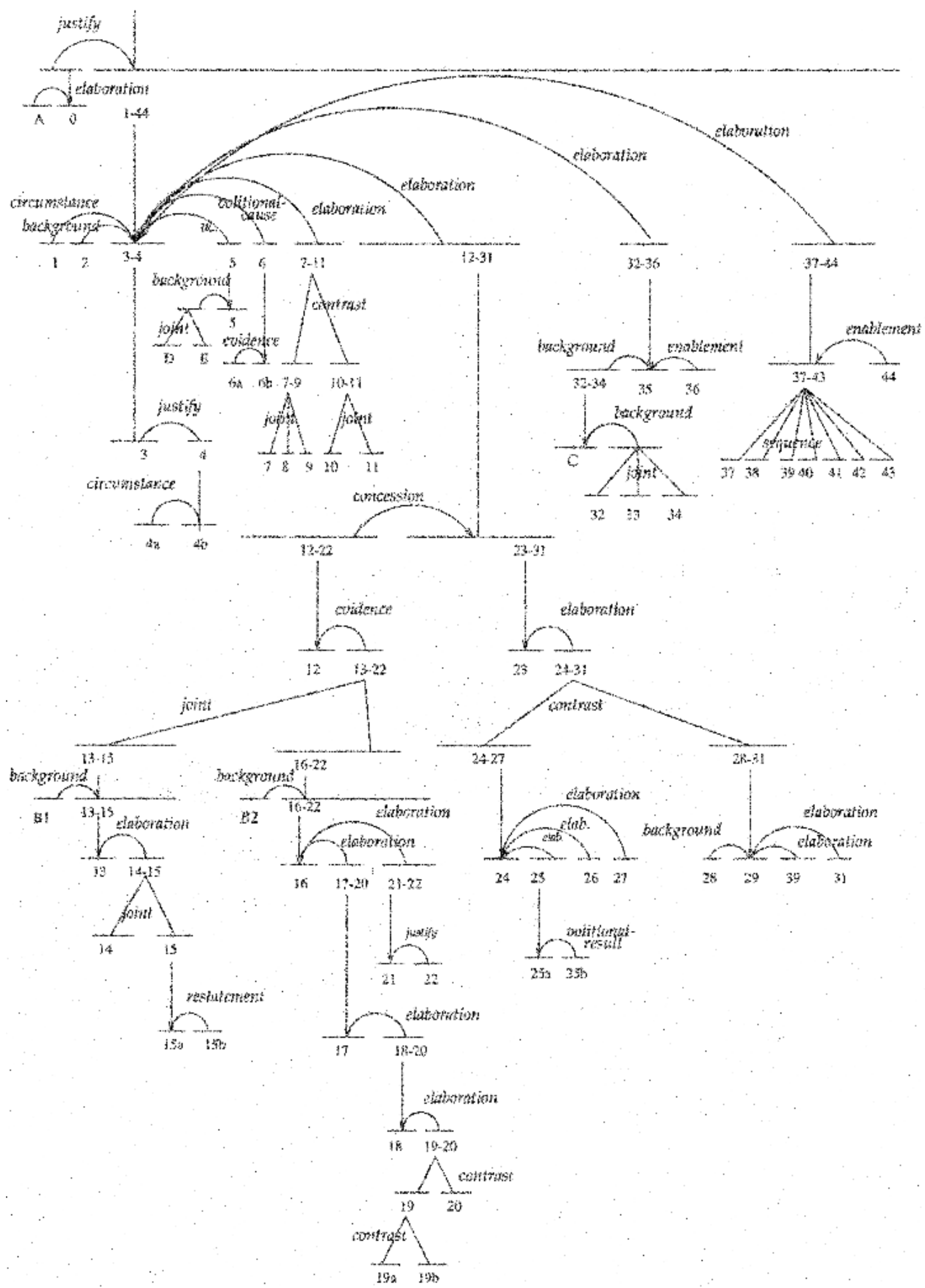


Figure 9
Rhetorical structure theory analysis of the constructed Unihoc text.

plan and the layout elements identifiable in the page.[5] We do not believe that it will be possible in general to predict precisely which aspects of a presentation plan will be rendered as distinctions in layout and which will be passed on for linguistic or graphi-

5 For the purposes of the present paper, we will continue to use informal descriptions of the typefaces and formatting options taken up; standard typographical terminology is more appropriate however.

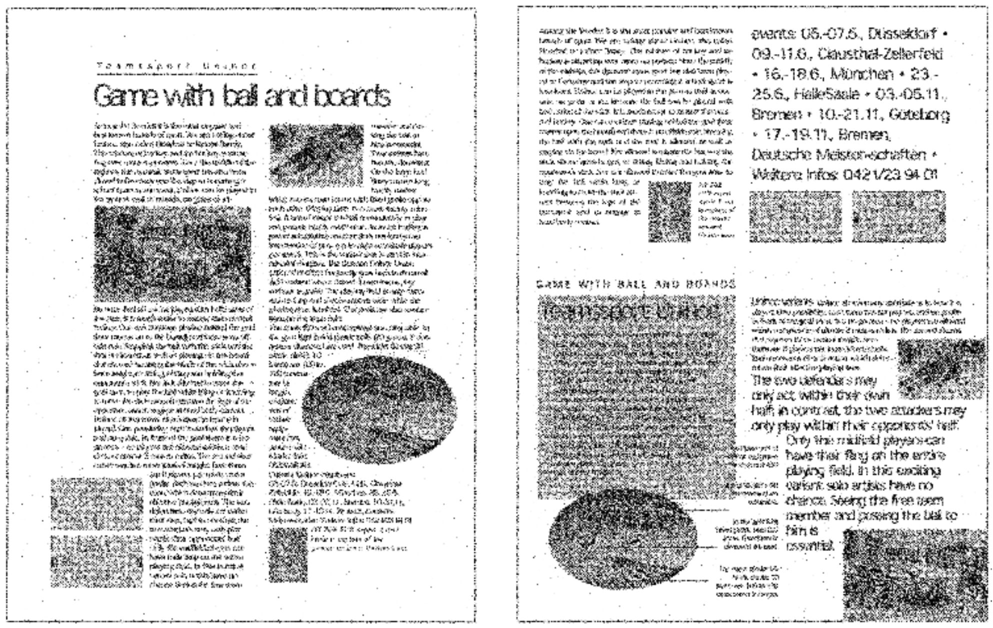
Table 2
Distribution of information and layout forms for the original layout. Here: "flowing" with respect to a bullet or other list item indicates that items are run on within single lines and do not form separate paragraphs; relative font sizes are indicated by +. All text units are left-justified and right-ragged.

		discriminations		
page-layout unit	*text segments*	*typeface*	*formatting*	*size*
Intro	[1]–[6] + [23]–[31]	neutral	2-column	neutral
Rules	[7]–[11]	bold	bullet-list, flowing, wrapping picture	neutral
Variants	[12]–[22]	italics	2-column	neutral
Equipment	[32]–[36]	bold	enumeration+ summary, wrapping picture, arrow links	small
Events	[37]–[44]	sans serif	enumeration-by-date + trailer, separate items, boxed	neutral
Author	[0]	italic	neutral	smaller++
Caption for Intro	[1]–[6] + [23]–[31]	bold	typewriter	larger
Caption for Rules	[7]–[11]	narrow	typewriter	larger
Caption for Events	[37]–[44]	bold	further distinct face	larger+
Page Title	[0–44]	hollow		larger++
Caption for D and E	[5]	bold	caption	small

cal presentation. The purpose of our empirical investigation was therefore a somewhat different one: we sought to constrain this process of producing layout structures as far as possible, and to examine whether certain allocations of informational units to layout units could be ruled out on general principles.

4.3 Alternative Renderings of the Constructed Text and their Evaluation

In order to probe the limits of the flexibility suggested above, we next considered alternative page-layout realizations of the communicative-functional intent represented in the rhetorically organized presentation plan. Our initial hypothesis was that, since the RST analysis represents a statement of the varying degrees of centrality attributed to the text segments present on the page, then these nuclearity assignments should also be reflected in the organization of any page layouts selected. Concretely, this hypothesis means relating text spans to nodes in the layout structure, with the nucleus of any span assigning higher weight to its corresponding layout structure node. This requires that the units found in the layout structure correspond only to proper subtrees within the RST structure and that elements grouped in the layout also be grouped within the RST tree. We refer to such a layout structure as respecting the "natural divisions" of the RST structure. Our investigation then evaluated this hypothesis by considering successively more complex "possible" layouts.



(a) (b)

Figure 10
Contrasting page layouts decomposing the source text: (a) Page rendition with minimal layout decisions, (b) Page rendition with random layout decisions.

First, for example, we can note that if the aim of the author/editor/multimodal presentation system were *not* to present an interesting page design, then a layout such as that shown in Figure 10(a) might suffice. This presentation contains almost no layout decisions that subdivide the text into segments or establish relations of similarity and difference among those segments. The only subdivisions are the heading, text-body, and author divisions; the pictures of the original have been inset into the main text block approximately where their content is touched upon in the text. Nevertheless, this would seem a perfectly possible (if, by current tastes, dull) rendering of the material to be presented. It could be appropriate, for example, in an extremely densely presented lexicon or encyclopedia where space constraints and tradition suppress layout variation. This layout therefore serves to represent one endpoint in a continuum of possible layouts that need to be accommodated in any general account. It is also a further illustration of the trade-off between information that is expressed linguistically through explicit textual realization and information that is carried by the layout: our constructed Unihoc text needs to do more explicit linguistic signalling of discourse relations and communicative function than does the version employing layout. A reader should be able to recover this information from reading the text but it is not supported by an explicit layout encoding; we return to this issue below.

We consider this case to emphasize that layout is concerned with *choice*. This is very similar to the state of affairs in NLG: the main principle is that a speaker/writer has to choose *how to present information* and, whenever there is choice, there is meaning: that is, the choice is not free and some choices will be more appropriate than others in particular contexts. Moreover, the layout decomposition that is selected should be in some way **coherent** with respect to the communicative functions of the page. The next example layout, shown in Figure 10(b), illustrates this by presenting a layout

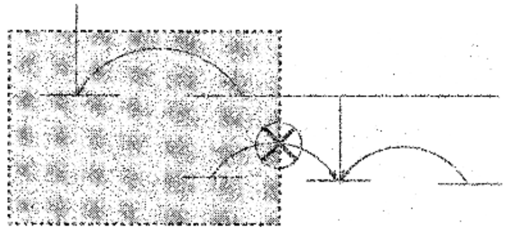


Figure 11
Example of a "non-natural" division of an RST structure.

in which choices, we would argue, have *not* been made coherently. The unmotivated decisions that make this layout incomprehensible include: the *Unihoc variants* section breaks down into two parts of differing visual appearance; additional information (events) becomes excessively important due to its prominent position and font size; and the information about the author is related only to the first textblock and not to the entire article. Such problems are identical to those arising in the NLG task: when more flexibility of expression/presentation is made available—for example, by considering generation with respect to a grammar with broader coverage—it is essential to control this flexibility appropriately in order to avoid wrong decisions.

The problematic nature of the layout of Figure 10(b) can be succinctly stated: it violates our initial hypothesis concerning the desired correspondence between RST structure and layout structure: the **constituency structure** of the layout and that of the RST tree are in direct opposition at several points. For example, there is no distinction drawn in type face between page elements introducing the game and describing the rules (which are relatively more nuclear), and elements describing some of the variants, whereas other variants are presented as a separate element with a larger, more prominent type face; these decisions are not, therefore, motivatable on the basis of the RST structure. Most serious, however, is the way in which the division into two elements describing the game variants has been done: these elements involve segments which do not correspond to RST-subtrees at all. Indeed, the second segment (corresponding to [19a]–[22] in the RST tree) goes further and breaks the RST structure at two points: segment [19a]–[20] is related by elaboration to unit [18] while segment [21]–[22] is related, also by elaboration, to unit [16]. The combined segment, [19a]–[22], is therefore composed of two completely disjoint and unrelated parts of the RST tree; we show this graphically in Figure 11. In general, such layout structures are indeed found to be difficult to interpret coherently (cf., Schriver [1996]).

The simplest strategy for producing an appropriate layout is therefore to restrict layout decomposition and constituency to the "natural divisions" established in the RST analysis. Any subtree is then, at least in principle, a candidate for selection as a layout unit. Accordingly, we suggest that the constructed layout shown in Figure 12 does fulfill the tasks of rendering the communicative intentions of the original page quite well. A relatively large number of layout decisions have been taken—for instance, the most important statements form a block of their own at the top of the page, additional peripheral information is placed in a vertical gray margin bar, and the main text is divided in two sections: *The rules* and *Unihoc variants*. Despite this diversity, the page remains coherent by virtue of the congruence of its layout structure with the RST structure; this is summarized in Table 3.

While this rendering of the RST structure is perhaps acceptable as a simple layout, we must observe that it still does not approach the complexity and diversity of the


Appendix A2

Among the Swedes it is the most popular and best-known branch of sport. We are talking about Unihoc, also called Floorball or Indoor Bandy. This mixture of hockey and ice hockey is attracting ever more supporters. Since the middle of the eighties, this dynamic team sport has also been played in Germany and the step to becoming a school sport is imminent. Unihoc can be played in the gym as well as outside, on grass or ice.

Teamsport Unihoc

Game with ball and boards

The rules

Unihoc variants

Figure 12
An example constructed page layout respecting natural divisions in the rhetorical structure.

Table 3
Distribution of information for a simple, coherent layout.

page layout unit	*text segments*	*discriminations*	*nuclearity*
lead-in	[1]–[5]	larger type face	√
rules	[6a]–[11]	neutral	–
variants	[12]–[31]	neutral	–
equipment info	[32]–[36]	small, sans serif, on margin bar	×
events	[37]–[44]	small, sans serif, on margin bar	×
authorship	[A+0]	small, italic, on margin bar	×

natural layouts produced for the kind of magazine from which the original page was taken. Therefore, while conformance to the RST structure may prove itself a necessary (or at least, very desirable) condition, it is by no means sufficient for the construction of appropriate layouts. The layout decisions taken in the original Unihoc page included both significantly more variety in the layout and typeface resources allocated, and a greater degree of decomposition. Moreover, although most of the design decisions appear coherent with respect to the notion of natural division developed so far, it is not the case that all of the layout decompositions are covered—in fact, rather *more* decompositions and discriminations are being made. This is shown graph-

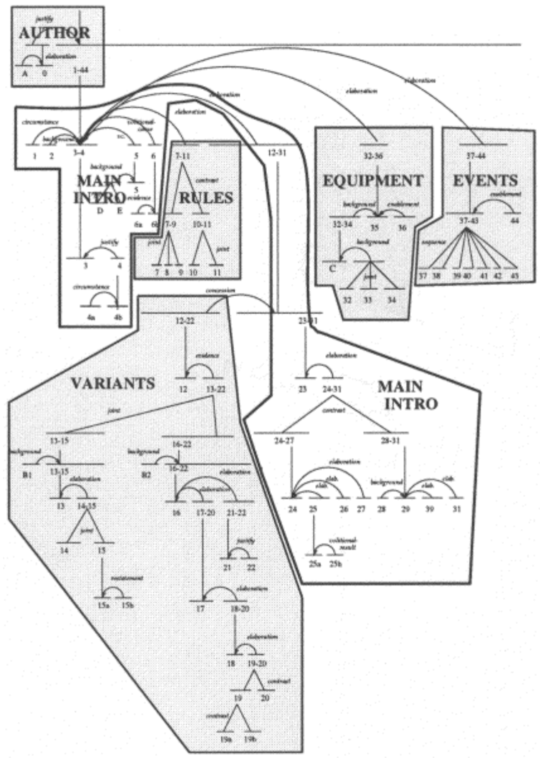


Figure 13
Correspondence between layout units and rhetorical structure in the original page. All of the blocks shown end up as siblings in the top-level layout structure because they are all visually quite distinct and variably moveable: they adopt differing typefaces, have no common left or right margins, are more or less evenly distributed over the page, etc.

ically in Figure 13 where the layout decomposition is explicitly contrasted with the RST structure; here we can see that, although the decomposition generally respects RST subtrees, it is not the case that we can transparently map this to a corresponding recursive layout structure. This rather common situation led us to accept that rendering the information in an RST-structure is not a simple decomposition, but is itself a planned activity. Information is taken from the RST-structure for various purposes and, as a consequence, both differing degrees of detail and varying decompositions must be supported.

5. Towards Automatic Page Layout: An Algorithm for Communicatively Motivated Layout

We have shown that an RST-analysis of the desired content for a page can be used to argue that a layout structure is more, or less, appropriate, and can indicate possible points of decomposition into layout units. We now go one step further and set out a procedure for the mapping of RST-style presentation plans into layout structures. Mapping is generally achieved by placing parts of the RST-structure in correspondence with particular nodes in a layout structure. This proceeds recursively down through the RST tree. As we have now seen, however, the correspondence is complicated by the fact the layout structure and the RST tree need not remain congruent.

We divide the mapping procedure into a core component, which supports the full range of layout options that appear possible given our corpus study, and a set of further heuristics that guide the translation process in particular cases. Most of the heuristics are concerned with the question of whether decomposition of the RST structure should stop at some point or continue to produce a more specific layout structure. In contrast, the core component is straightforward—mostly because the layouts observed exhibit such flexibility that few hard constraints seem to apply.

5.1 Core Translation Procedure: Recursive Descent

At any point in an RST structure, we can consider the correspondence between the RST subtree descending from that point and an appropriate layout structure. The translation procedure assumes strong locality in that, when decomposing a subtree, it does not look outside of that tree to make its decisions. As the layout structure is constructed, layout units can be either **open**, in which case they are still accepting additional material from the RST structure as recursive descent continues, or **closed**, in which case the extent of their content has been fixed. Translation begins by positing a single layout unit that will hold all of the content of the RST-subtree, and a single, at that point empty, open descendant layout unit. If the mapping process decides that there is no need for any further layout decomposition—as in a monomodal, traditional NLG environment with neither graphics nor headings—then the open layout unit is closed off containing the entire RST tree being considered. This content (and its rhetorical organization) would then be passed off to the natural language generator in the usual way.

In most documents, however, layout decomposition occurs, and so the layout structure has to be grown further. This is achieved by partitioning the set of RST-structure segments available at the current level of RST structure (segments are indicated in RST diagrams by the horizontal lines: cf., Figures 6 and 9) into those that are to be accepted into the current layout unit and those that are not. When any segment is not accepted, the RST relational link is cut at that point, the accepted segments are added to the current layout unit (which remains open), and the cut segment is added to a new, closed layout unit inserted *parallel to* the original layout unit. The accepted partition must include at least the nucleus at that level of RST structure; this means that it is not possible to include satellites in the current layout unit and then exclude the nucleus—this would correspond to the unnatural division shown in Figure 11 above. Crucially, it is quite possible for all of the segments to be accepted and for recursive descent to continue *without having closed the current layout unit*. Both closed and open layout units are then further developed according to the recursive structure of their respective RST segments.

When a decision is made to cut an RST link, and the segments to be partitioned belong to a multinuclear relation or joint schema, then *all* the related segments are extracted and associated with sibling layout units. The layout units thus introduced are also annotated with type-equivalence links, while **reference annotations** are added between pictures and text within single RST-structure segments. The weight annotations are allocated in direct proportion to the nucleus and satellite distinctions: the nucleus receives the highest weight, with the remaining distributed over the satellites (see Section 3). We have not yet been able to explore appropriate weightings in any sophisticated manner, and so their effect on alternative layouts remains a subject for further investigation.

In summary, the process of recursive descent of the RST tree can be seen broadly as one of deciding where to break RST links. Breaking a link (either a span-to-nucleus link or a nucleus-to-satellite link) causes the material thus separated from the tree

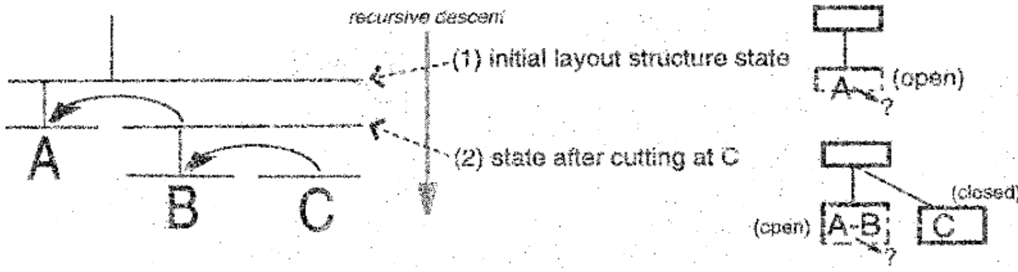


Figure 14
Construction of correspondence between rhetorical structure and layout structure assuming a decision is made to cut the RST structure between segment B and C. Uncut material is added into the currently open layout unit and so can contribute to the current top-level layout; cut material is added into closed units and so cannot thus contribute.

to be allocated its own layout unit, standing in a sibling relationship to the current layout unit, i.e., that unit corresponding to the RST material from which the span was broken. Material that is not separated in this way remains part of the information content to be expressed by the current layout unit. This means that RST subtrees related hierarchically in the RST tree may be assigned to sibling layout units. The process is shown graphically in Figure 14.

This can therefore lead to a significant loss of structure resulting from the fact that the dominance relations holding in the RST structure are *not* maintained in the layout structure. The core translation procedure thus results in a steady reduction in the complexity of the structure being constructed compared to the RST structure from which it derives. In the Unihoc original page, this is the situation with the segment [12–22] (Unihoc variants): this satellite is broken out of its potential embedding within the dominating segment [1–6]+[12–31] and appears instead as a layout sibling of the remaining Intro unit that realizes [1–6]+[23–31]. This action has two common consequences: first, the layout weight of the broken-out unit is reduced and, second, the fact that a layout unit has been severed from its functionally motivated parent (i.e., the nucleus) appears in many text types sufficient grounds for introducing a further layout unit that has the task of explicitly signalling what the relation was. This additional layout unit appears as a header or title (e.g., the *Alle Formen sind möglich*—(all forms are possible—of the original page). A header is then the layout equivalent of a discourse marker. However, since headers cannot typically function as relationally as discourse markers (because they cannot readily point back to their point of origin), they tend instead to give sufficient information for the reader to make a connection—for example, by summarizing the nucleus of their segments. While we currently attribute the reduction in layout weight to the core translation, the provision of headers again appears to be a genre-specific concern and so we keep this as a heuristic.

Removing structure can, if unconstrained, lead to some very undesirable layouts: for example, we could take the entire RST structure of the Unihoc constructed text *apart from* the last satellite of the Equipment subtree to produce two parallel layout units: the rest of the text and the fact that one can get information about the Equipment by telephoning the given number. Our translation process partially avoids this situation by employing media-allocation (i.e., whether layout units are to be realized textually, graphically, pictorially, etc.) to force partitions whenever there is a difference in medium. For example, if one segment of a partition is to be realized as a photograph and another as a textblock, then it is not possible to accept both of these into the open layout unit. This is motivated by the fact that we do not have a means of combining the content of photographs and text other than by allocating them to distinct layout units and positioning them in appropriate proximity; we return to the issue of media allocation below. The rule serves to avoid insignificant satellites being giving a prominent layout status by increasing the likelihood that prior decomposition will have already taken place.

Moreover, we have observed two further forces that restrict objectionable layouts of this kind: first, whenever a collection of layout units are constructed as siblings, their relative weights need to be ascertained—the further down (vertically and in terms of nucleus-satellite status) in the RST structure, the lower relative weight an associated layout unit receives.[6] Second, we add a heuristic that disprefers isolating relatively insignificant satellites that lack substantial substructure of their own. The strength of this heuristic appears to vary significantly according to the genre of the information being presented: for example, newspapers now appear to be considerably more willing to promote lower subtrees to the status of layout siblings than are the pages that were the subject of our empirical investigation. We will return to this aspect of layout variation in the conclusion.

Finally, although we cannot pursue this theme further here, the gradual reduction in structural complexity proposed has several suggestive similarities with what happens when monomodal, linear text is produced from an underlying RST-like structure. There too, structural complexity is replaced by explicit linguistic coding operating linearly from one clause or utterance to the next. The linguistic encoding must give sufficient clues for a reader/hearer to make an attempt at recovering the intended structure, but does not itself appear to exhibit that structure. In this respect, one-dimensional text and two-dimensional layout may not be as different as might have been thought.

5.2 Layout Units and Layout Forms: Bottoming Out

When the decision has been made not to continue decomposition of an RST subtree into further layout units, the entire content associated with the subtree, as well as its rhetorical organization, is placed in correspondence with a terminal layout unit. That layout unit is then allocated a **layout form**. Layout forms can be either graphical (e.g., diagram, photograph, etc.), or textual. Examples of textual layout forms are textblocks consisting of paragraphs, enumerated lists, itemized lists, and the like. The broad choice between graphical and textual layout forms is made by the media-allocation decision described below.

Significantly, some textual layout forms have substructure of their own. We consider them complex forms within a single layout unit (rather than conjoined layout units), because they appear to have rather limited scope for independently flexible layout within their enclosing layout units—it is unusual to format distinct elements in an enumerated list with different type families, type sizes, or colors, etc. When a layout form requires further structure, we talk of layout **elements** extending the orig-

6 This may even occur in the Unihoc page. On the righthand edge towards the bottom there is a very small piece of text citing the source of the photographs. Since this information can be seen as elaborations of the photographs, we have an example of a situation where a very insignificant segment (for the content) is broken out and given its own layout unit. The consequence is, predictably, that it receives very little visual weight.

Table 4
Distribution of information to layout elements within a layout unit, depending on the selection of layout form.

RST configuration	*rendered in*	*layout unit/element*
span[32–36]	↘	layout form: itemized-list
joint([32],[33],[34])	↘	itemization: by number
		[32] ↘ numbered item
		[33] ↘ numbered item
		[34] ↘ numbered item
enablement([35],[36])	↘	list-trailer
		[36] ↘ trailer

inal layout structure downwards: for example, the textual layout form **itemized list** introduces layout elements for each of its items. We have considered modeling this in two ways: first, by allowing the textual layout-form textblock to include formatting similar to that provided in HTML or LaTeX;[7] and, second, by simply allowing the RST-to-layout translation process to recurse further with a restricted range of allowable layout forms.

While the first approach is straightforward to implement and is closer to existing approaches to punctuation and formatting in NLG (Sefton 1990; Hovy and Arens 1991; White 1995; Pascual 1996), the latter appears to offer more scope for investigating the trade-offs between layout and linguistic realization mentioned above. It may also prove necessary for more aggressively graphical documents that include traditionally graphical elements (e.g., pictures, icons, etc.) within the confines of traditionally textual elements (e.g., items in itemized lists). A textual layout form is still most often, however, a site of relative textual stability and will generally enforce linguistic and layout (e.g., style, punctuation) uniformity on the content that is expressed: for example, rendering all items of a list as nominal phrases, verb phrases, sentences, etc., as considered appropriate. This occurs in several places within the Unihoc page: in the case of the *Equipment* segment, for example, we have the correspondences indicated in Table 4. The *Events* segment is similar, with enumeration by date followed by a textblock as the enumeration trailer. In both cases, typographical constraints are selected for the overall layout unit and these are then enforced for each constituent layout element.

At present, we consider the motivations for distinct textual forms only very straightforwardly: if there are diverse RST relations present in the content corresponding to a layout unit, then we favor generating running text for that content; if there is a strong multinuclear RST organization (as, in the present example, the sequence seen in [37–44] or the joints in [7–11]), then we favor an itemized list of some kind. Exhaustive presentation combined with strict sequencing leads us to favor an enumeration; less exhaustivity or lack of strict sequencing moves us through bulleted lists to simple sequences of offset paragraphs. These heuristics are currently enforced by simple calculations concerning depth of RST relation, ratio of relation types to number of segments, etc.

7 This would appear to correspond closely to the level of representation called a DocRep (Document Representation) in the reference model for generation architectures that have been proposed within the RAGS project (RAGS Project 1999).

5.3 Media Allocation

During the construction of layout structure, the translation process attempts to determine appropriate media allocations for the layout units introduced. Although, in general, this needs to be a motivated selection of presentation mode using criteria such as those discussed by Arens and Hovy (1990), Arens, Hovy, and Vossers (1993), Feiner and McKeown (1993), and others, for our current purposes we have restricted this decision to information that can be gained from a dependency-lattice analysis of the information content associated with segments from the RST tree. Thus, when considering the partition of segments at some point in the recursive descent through the RST tree, the translation process requests a dependency lattice for each segment and, depending on the result, assigns a likely medium choice. The currently possible allocations are **diagram**, **photo**, **text**, and **complex**. The main choice, however, is between diagram and text; the photo option is self-selecting—i.e., if information is only available as a picture then that is what will be slotted into the layout structure, and the complex option is selected whenever there are self-selecting segments deeper in the RST subtree being considered.

The heuristic that we employ for allocation assumes that the more simultaneous dimensions of regularity are present, the more likely it is that a diagram will be the more perspicuous representation. The rationale for this can be readily seen from the suggested text version of the diagrams in Figure 4. The text version does not read fluently precisely because it struggles to express simultaneously the co-varying dimensions of school, profession and time period. The fact that there are so many co-varying dimensions is, however, readily available from the dependency lattice: when a dataset has extensive and simultaneous regularities then these are present as nodes in the lattice. Thus, when there are co-varying dimensions of potential aggregation, a medium allocation of diagram is made, when not, text is selected. These allocations are then employed in the decision of whether to cut an RST relational link as described above.

The dependency lattices used for media selection are maintained as additional information that is drawn on during both natural language and diagram generation. Moreover, and depending on the rhetorical relation, there can be co-ordination such that a satellite must re-use the dependency lattice already calculated for its nucleus. This is shown concretely in our DArt$_{bio}$ discussion below.

5.4 The Unihoc Pages

With the translation process described so far, we can come close to producing a realistic layout structure for the Unihoc RST structure. Beginning at the top of the RST structure in Figure 9, our first set of segments to partition is {[A+0], [1–44]}: both have a complex medium allocation and so we cut the RST structure at this point. The satellite [A+0] is associated with a closed layout unit and the nucleus [1–44] with a sibling open unit. Whereas both segments have further recursive structure, their treatment is now different: because the satellite unit is closed, its rhetorical substructure now contributes to *embedded* layout units (i.e., the process recurses here); in contrast, the nucleus unit is open, which means that its rhetorical structure may still contribute top-level sibling layout units. And this is what happens. The next set of segments to be partitioned is {[1], [2], [3–4], [5+D,E], [6], [7–11], [12–31], [32–36], [37–44]}; the nucleus is [3–4] and so the decision concerns which satellites, if any, are to be grouped with the nucleus and which are not. Our heuristic concerning regular substructure (ratio of relation types to segments) picks out [5], [7–11], [32–36] and [37–44] as good candidates for their own layout units; moreover, the units [5] and [32–36] are also complex by virtue of the fact that they include photographs. This leaves the segments {[1], [2],

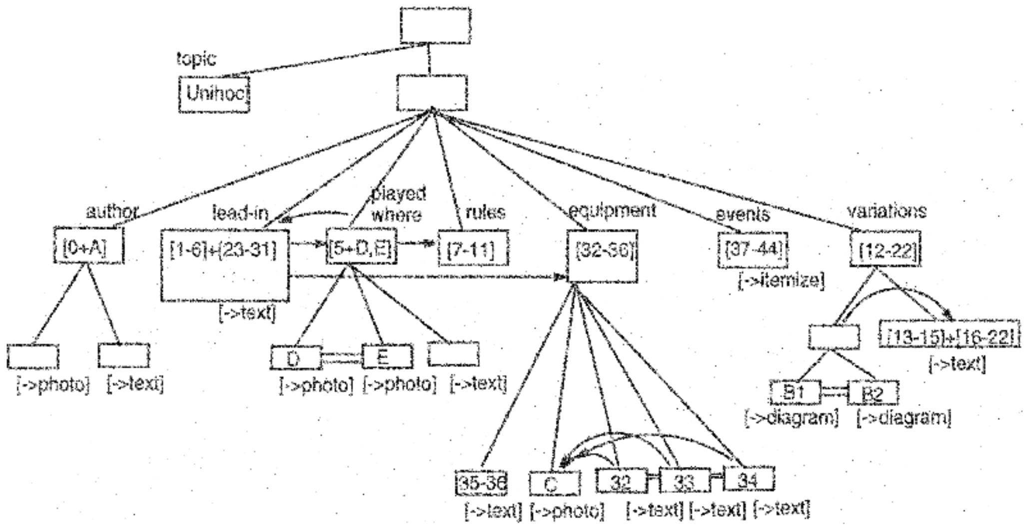


Figure 15
Layout structure for the original Unihoc page.

[3–4], [6], [12–31]} for inclusion within the current open layout unit, while the others are each added as sibling closed units. The process continues in this fashion, eventually adding all of the material under [3–4] to the open unit, as well as the nucleus of [12–31]. The satellite of [12–31] is cut, however, and is therefore added as a closed layout unit. Note that since we are still adding material to the original open layout unit, this latter satellite also ends up as a top-level sibling. An indication of the entire layout structure of the original page is shown in Figure 15. In addition to the constituency relationships we have discussed so far, the layout structure includes reference links introduced between pictures and text related within single RST structure segments, as well as equivalence links introduced between units associated with members of RST joint schemas.

When rendered as a page, the layout structure does not indicate any type-equivalence links between the top level units; so, apart from the prominence that should be accorded to the main-line, nuclear textblock, there is no clear relationship of similarity between any of the remaining decomposed units. We can therefore expect either the same form to be used for all of them (for a "quieter" page layout) or differing forms to be used for each (for a "louder" page layout).[8] Probably unsurprisingly, given its intended audience and function, the page as published adopts the latter option, yielding the diversity that we saw displayed in Table 2.

The same RST structure also stands as a starting point for the production of layout structures corresponding to the alternative layouts shown in Figures 10(a) and 12 above—although some questions are raised concerning the adequacy of our proposed "simplest" layout in Figure 10(a). To produce the simplest layout, we simply refuse to cut any RST relational links until the occurrence of a photograph or a diagram forces us to. This yields one layout unit for the photograph of the author and associated text, one layout unit for the rest of the text (with a 2-column textblock layout form), and

8 Thanks to Judy Delin for suggesting the **volume control** metaphor.

layout units for the remaining photographs and diagrams. However, given our RST analysis, it is then impossible to motivate the disappearance of the equipment **call outs** because the photograph is nuclear. This may either indicate that the RST analysis is incorrect at this point or, more likely, that the layout is not a faithful representation. In fact, just distributing pictures liberally around the page does not appear to yield an effective layout, and thus, this "simplest" rendering is in fact *too* simple: it no longer communicates the intended meaning; either the picture of the equipment has to go, or the individual equipment descriptions have to be reinstated. This is done in Figure 12, where the layout divisions are similar to the original page, with two important exceptions: first, the RST segments [1–6] are maintained as a single layout unit with a substructure consisting of one textblock and the two photographs as referring sibling layout units;[9] second, the RST segment [12–31] is maintained as a single layout unit without further decomposition. Finally, it should also be clear why the unacceptable layout shown in Figure 10(b) cannot be produced according to the translation procedure we have described since; for example, it is not possible to accumulate such wildly displaced segments of the RST tree into single layout units.

5.5 Interim Conclusion

This section has set out a procedure for producing layout structures from rhetorical structures. However, the mapping remains highly nondeterministic. Further sources of constraint will be required before it will be possible to let the procedure simply run and produce layouts. Part of this indeterminacy lies in the fact that many fine-grained decisions in the microtypography of real publications (e.g., what typeface to select, whether text in a layout unit is typeset ragged or justified, etc.) are *not* motivated by the communicative-functional intentions as captured by the RST-analysis of single texts, but are instead fixed by higher level decisions concerning a magazine's intended style and feel (cf., Reichenberger et al. [1996] and the conclusion below for further discussion). Another part of the indeterminacy arises out of the fact that layout is genuinely very flexible. Considerably more investigation will be required before the parameters and limits of that flexibility are charted. Nevertheless, armed with this first mapping specification, we now turn to a particular application to see how it functions with rather simpler tasks than the world of Unihoc pages.

6. Page Generation within the Prototype Information System

The application scenario described in this section was a natural outgrowth of the Editor's Workbench (see Section 2.1), moving from its original conception as a tool for editors, to take on more of a role of an information system for general use. Prior interface-design studies investigating beneficial applications of the Editor's Workbench suggested a usage scenario in which the information system provides "overinformative" responses to information retrieval requests. In this scenario, the user is assumed to be browsing rather than posing focused queries. The overinformative response avoids burdening the user with unwanted data by presenting the information as a coherently organized multimodal page: the user can quickly scan the information on offer in the same way as he or she would scan a newspaper or magazine page. For this to be effective, the layout must correctly communicate which information is central and which information is more peripheral—i.e., it must signal the communicative

9 This is in fact the layout structure shown in Figure 5 above; nodes 2.3.1, 2.3.2, 2.3.3, and 2.3.4 in the figure correspond to the RST spans [1–6], [12–31]—D and E, respectively.

intentions of the page as a whole. The resulting DArt$_{bio}$ system aimed specifically at providing pages that present artists' biographical information: the Editor's Workbench domain model had already accumulated information of this kind for several thousand artists and so offered a solid basis for prototype construction. Finally, for something of a local connection, we concentrated on the Bauhaus.

There are many aspects of the design and implementation of the complete DArt$_{bio}$ system that space precludes us giving here. Moreover, the system remains an initial testbed for our approach—many considerations crucial for a practical system have not been addressed. Our focus centers on just those aspects relevant for our claims concerning layout and the mechanisms by which it is realized—and even here the presentation will sometimes need to be schematic. The central components that are relevant are the automatic visualization engine (AVE: Section 2.1 and Kamps [1998]), the automatic page-layout component (APALO: Section 3), the natural language generator (KOMET: Bateman and Teich [1995] and Bateman, Teich, and Alexa [1998]), and the very large domain model (ca. 500,000 objects) of the Editor's Workbench (Rostek, Möhr, and Fischer 1994). The slice through the system that we describe lies between the goal-driven establishment of a rhetorically structured presentation plan on the one hand, and the construction of a fully specified layout structure on the other. To achieve this transformation we apply the general mapping procedure described in the previous section augmented by application-specific decision heuristics. As with most multimodal information systems, the terminal elements of our layout structure receive particular media allocations that determine the means employed for their production; but, in contrast to previous systems, we detail the new role played by dependency lattices in both media-allocation and media-coordination and then some of the consequences of adopting a flexible rhetorical structure to layout-structure mapping.

6.1 Abstract Page Specification

We begin by setting up some closely related presentation plans. Our examples revolve around possible answers to questions art historians typically ask in discussing the spread of art movements: for example, *How did the Bauhaus spread to the United States?* The interaction style supported by the Editor's Workbench—and hence by its further incarnation as DArt$_{bio}$—is described in Kamps et al. (1996); users explore the database progressively by following proposed links. Asking for information about the Bauhaus retrieves, amongst other objects, the set of all artists classified as Bauhaus artists. The Workbench also includes preset question types for distinct types of object; these question types were established through consultation with editors expert in the domain. One such question for the object type *art movement* is spread of influence. The domain knowledge representation for art history includes a range of conditions that are known to contribute to a spread of influence: for example, artists moving to the country being influenced. Following the link *spread of influence* and cross-classifying this against a country, such as *U.S.A.*, then restricts the former set of Bauhaus artists down to those that are known to the system to have moved to the U.S. This information then needs to be presented to the user.

At this point, DArt$_{bio}$ must construct presentation plans. Presentation plans are composed by appealing to knowledge of how particular kinds of texts are structured. For example, biographies as a text type, or genre, have certain regularly reoccuring features of organization and content. The NLG component of the system, KOMET, is strongly genre-based: generation proceeds first and foremost by receiving a request to generate a text belonging to a specific genre, e.g., a biography. The linguistic details of the texts required are established by prior corpus studies. This use of genre as a prestructuring device for texts resembles schemas as introduced into NLG by McKeown

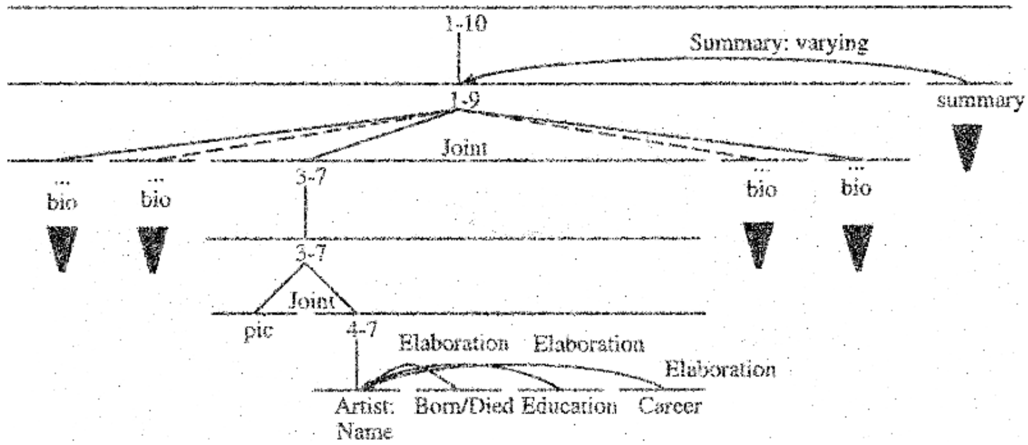


Figure 16
Page presentation plan for a set of biographies with summary. Downward arrows indicate points of further refinement. The biographies are filled in the same way as the biography that is shown, following the linguistic genre constraints for biographies. The summary is filled in using a dependency lattice for information extraction.

(1985), but is grounded linguistically more in notions of generic text structure (Hasan 1996); there is also a clear relationship to domain communication knowledge (Kittredge, Korelsky, and Rambow 1991). We extended this genre-based view to include all of the information offerings of the DArt$_{bio}$ system. Presentation plans therefore begin their life as partially specified generic structures, defined as incomplete rhetorical structures.

We consider two styles of presentation arising from this scenario: in the first, the user is simply presented with information about the list of selected artists; in the second, the information focuses more on the movement of those artists to the U.S. Both presentations use a common generic presentation style and were motivated by user/editor consultations within the earlier phases of the Dictionary of Art project—the main information content of the communicative move is allocated to the nucleus of a **summary** rhetorical structure. The initial presentation plan is then grown further by recursive descent. The contents of the satellite are filled in according to the definition of summary as given below. The contents of the nucleus were provided by the information-retrieval request—in our first presentation plan, this is a set of artists combined by a joint rhetorical schema. Each of these is then expanded further: since the default presentational style adopted for an artist within DArt$_{bio}$ is the biography genre, the NLG component constructs a rhetorical fragment conforming to the structure of this text type, collecting information from the domain model as required. The particular model of biography adopted here also includes the use of a photograph if the Editor's Workbench domain model includes one for the artist in question.

This results in a presentation plan of the form shown in Figure 16. While all of the biographies are filled in in the same way, employing standard NLG-techniques, the summary is more interesting. DArt$_{bio}$ distinguishes between two subtypes of summary: a summary giving an overview of shared properties with varying attributes (*summary-varying*), and a summary giving an overview of shared properties only (*summary-shared*). In the current presentation plan, the former is used. Our approach to summaries relies heavily on the dependency lattice-mechanism; regularities are assumed to provide useful summaries of the data being presented. The information content of the summary is therefore produced by constructing a dependency-lattice of the information to be summarized—i.e., the list of propositions contained in the nucleus—and picking out regularities (i.e., nodes higher in the lattice).[10] Performing this for the information in the biographies picks out and partially orders the information that recurs: i.e., name, date and place of birth and death, and profession (which always occur), and then certain contingent aspects of education and career depending on the particular artists selected. The highest scoring facts are selected for inclusion in the summary (which may either occur immediately by some preset cut-off point or, more interestingly, be left for later when space constraints may determine how much information to give). This set of information is not organized further rhetorically.

6.2 Determining Page Layout

If, for the sake of concreteness, we now assume that the user has restricted his or her attention to a selection of just five artists: Gropius, Hilberseimer, Anni Albers, Josef Albers, and Breuer—again we emphasize that the names and information presented here are for *illustrative purposes only*—we can apply the layout mechanism of the previous section to the presentation plan in order to produce a corresponding page. The presentation plan is much simpler than the Unihoc case and so the options are somewhat more restricted. Beginning at the top of the plan, we may either decompose the plan or not. We consider the media allocations for the possible partitions, which are initially comprised of the nucleus to the summary as well as the satellite. The nucleus can only be allocated a complex medium-status (since it is structurally complex and includes subtrees with pictorial content), while the satellite can be allocated either text or diagram status. Given this choice, DArt_{bio} selects diagram as preferable on the grounds discussed above. The members of the partition therefore receive different media-allocations and so a layout decomposition is introduced. The summary-satellite is a leaf of the presentation plan and so there is no further layout decomposition to be considered. The layout element is allocated a diagram-layout form and the content is passed on directly to the visualization component for diagram generation.

Layout decomposition then proceeds with the nucleus, which consists of the joint schema. The decision is whether to stop decomposition at this point or whether to continue. Each member of the joint schema still has a complex medium-status and so our heuristics favor decomposition. The core layout mechanism does not allow us to break just one member of a multinuclear configuration off from the presentation plan and thus all the sibling biographies must be taken. This introduces a further five layout elements into the layout structure, each corresponding to a biography. Moreover, these five are linked by equivalence annotations and, because they are nuclear in the presentation plan, their combined weight must remain greater than that of the diagram. Since subtrees have now been cut from the presentation plan, the layout mechanism also has the option of introducing additional layout elements as headers. In the DArt_{bio} case, we set up our heuristics, rather arbitrarily, to give headers to textblocks only; a more realistic decision-making process here can only be constructed after considerably more empirical investigation. Thus, since the individual biography layout units are still complex, we do not immediately construct a header element but pass this option down so that it may be picked up by the first nuclear textblock that

10 Other, more sophisticated approaches to summarization can, of course, be considered; moreover, it may not be necessary to consider *all* of the propositions in the nucleus. RST suggests that only nuclei need be considered, while Cristea, Ide, and Romary (1998) refine this further and suggest that some nuclei plus a specified set of their satellites (those that are **vein heads**) is preferable.

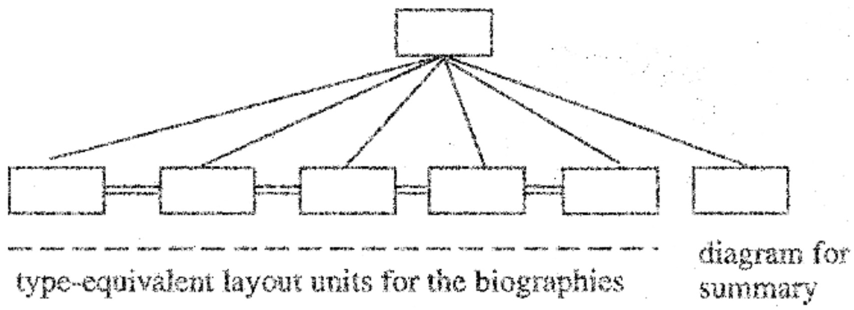


Figure 17
Partial layout structure following decomposition of the presentation plan according to summary and biographies.

is encountered. The state of the layout structure following these decisions is shown graphically in Figure 17.

The constraints on layout and presentation that are being accumulated in the layout structure now begin to bite, restricting possibilities for further layout-structure development. First, as specified by the layout mechanism, when the visualization engine attempts to design a diagram, it must construct a dependency lattice for the information that is to be expressed. It is now constrained, however, to taking the dependency lattice that has already been constructed for the nucleus to which the summary is connected. This guarantees that the grouping and graphical-attribute allocations deployed in the diagram will be consistent with the regularities observable in the information presented in the set of biographies. Second, when the layout structure is decomposed further, the equivalence relation will require that the same layout decisions be made for all the biographies. This then guarantees that regardless of precise physical placement, the typographical message will be that these biographies have the same status.

Decomposition of the biography layout units is straightforward. The only partition is in terms of the photograph and the biographical information: the medium-allocation of the photograph is self-selecting, and that of the biographical information is text because of the few co-varying dimensions of information revealed by the dependency lattice. Therefore, the layout unit is decomposed, and the picture receives a *refers to* link. The remaining biographical information can either be grouped within another single layout unit, or decomposed further. Our heuristics do not bother to decompose when there is little content to be expressed; and as each of the satellites in the biography contains only a few propositions (rarely more than four with most artists in the Editor's Workbench knowledge base), we do not decompose further here, selecting a single layout unit with a textblock layout form. However, since we have the option of inserting a header layout element from above, we discharge this here, giving a layout structure as shown to the left of Figure 18. A page-rendering solution for this structure is shown in the upper right of the figure: layout elements have been mapped to rectangles with particular dimensions and locations on the page. The contents of the layout elements are produced according to the layout forms allocated: the picture is simply inserted (with an appropriate scaling for the layout unit as a whole); a text is generated by the NLG component (conforming to the generic text constraints of biographies); and a header is selected (a domain-specific process). When these contents are filled in, the results is page fragments such as that shown in the lower right of the figure. Similar solutions are provided for all of the biographies because of the specified type-equivalence.

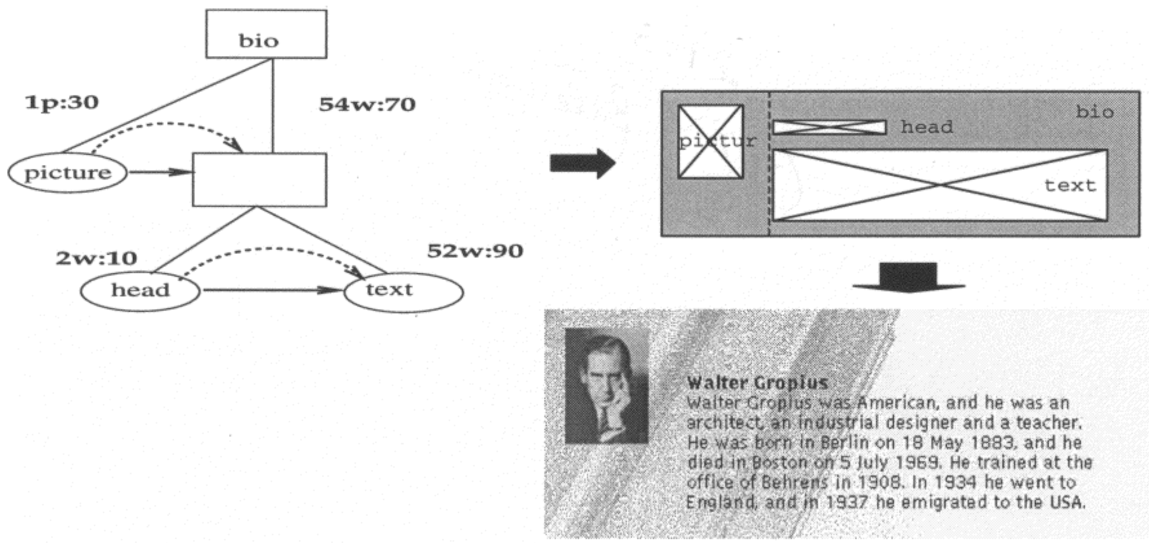


Figure 18
Segment of the layout structure for the layout unit corresponding to a single biography, together with the resulting segment layout and the filled-in page extract.

6.3 Page Rendering: Examples

The layout structure for the current presentation plan is now complete. An example of a complete generated page that renders the layout structure is shown in Figure 19. The individual biographies are each as described above; we can see that the generic constraint that a photograph of the artist be presented has not been satisfied in all cases, but this does not affect the general layout forms applied to the set as a whole. More significantly, the regularities expressed in the diagram are precisely the regularities that are expressed in the text parts of the biographies. The diagram picks out artists as the basic graphical elements and allocates to these elements attributes of color (for profession) and extent (for lifespan). In addition, time and place of birth and death are added as icon graphical attributes, again color coded. Conformity and consistency in design has therefore been achieved for the page as a whole even when there are minor deviations in available information in some of the page elements.

The layout process is very flexible in terms of the solutions that it pursues. The apparent resemblance of the page to a grid-based layout is in fact simply a by-product of the constraint-resolution process and its attempt to fit 7 boxes (the 5 biography layout units, the diagram, and the diagram's key) into a limited space. For comparison, some other solutions offered by the layout process are shown in the alternative **thumbnail** layouts of Figure 20. These give a graphical sense of the notion of **layout unit**: we can see that layout units are maintained as persistent visual groupings regardless of their precise placement. Since all of the top-level layout units are siblings in the layout structure, the layout process must also work to avoid suggesting any grouping among them. This is achieved by spreading the units evenly over the page as a whole.

The co-ordination of text, diagram, and layout demonstrated in the Figure 19 can be further illustrated very briefly by considering our second presentation plan:

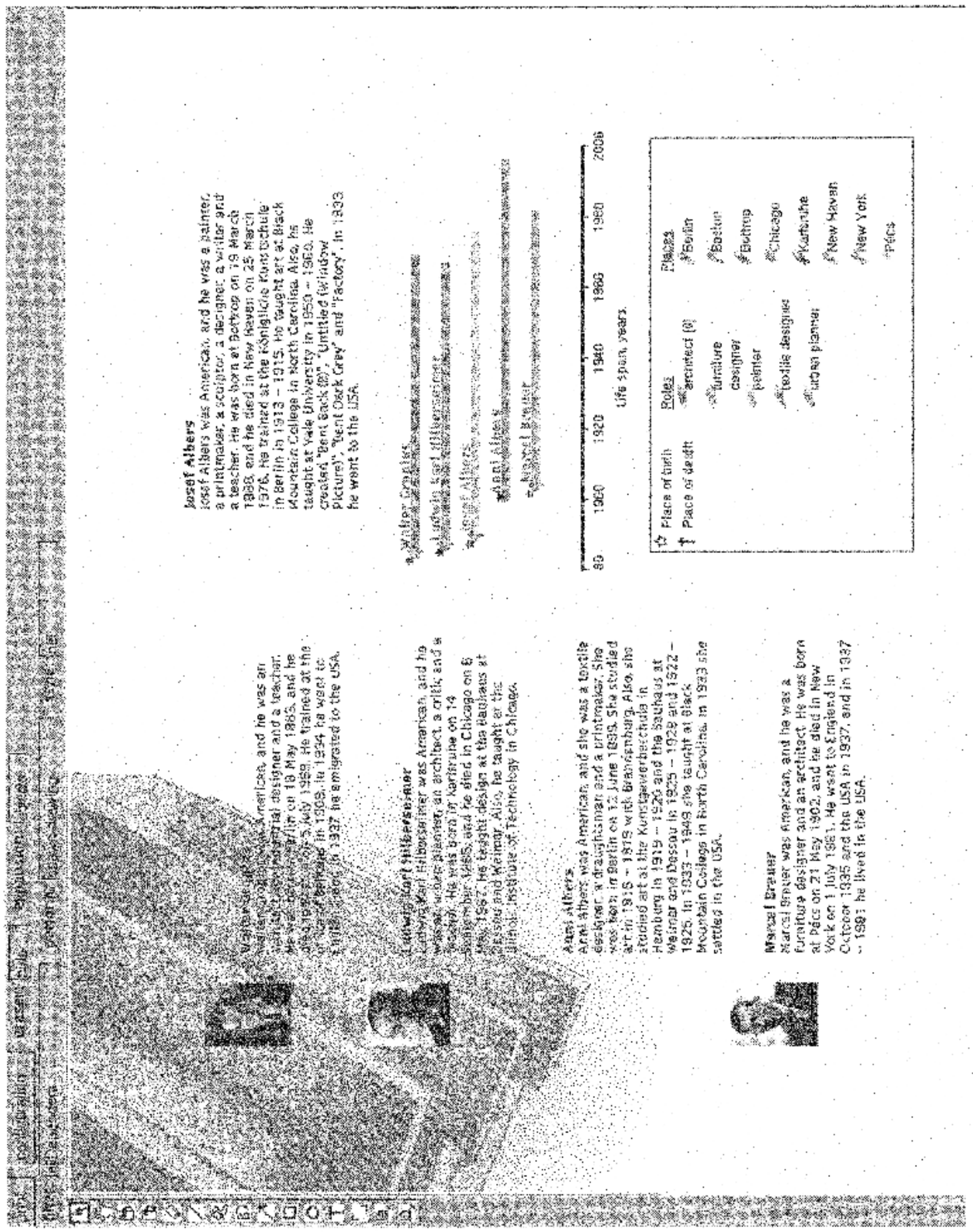


Figure 19
Example screenshot of a generated page from the art history system.

here, the main nucleus of the generic rhetorical structure is not a set of artists but instead a set of propositions, each describing the movement to the U.S. and subsequent professional activity of an artist; such facts are readily retrieved from the Editor's Workbench knowledge base and result in relations of the form (cf., Kamps et al. [1996]):

```
migrate(A.Albers,fromTo(Germany,USA),1933)
facultyAt(A.Albers,Black Mountain College,betweenAnd([1933,1949]))
```

Figure 20
"Thumbnail" views showing screenshots of a range of different solutions to the layout-constraint resolution process.

Page generation with this information can proceed in exactly the same way as for the previous example. There is, however, less information under the main nucleus, as there are no longer full biographies to generate but rather text elements of the form:

> Albers settled in the USA in 1933. In 1933–1949 she taught at Black Mountain College in North Carolina.

These may be allocated either to distinct layout units as in the previous example—which would distribute them about the page as shown—or to a single layout unit as a textblock, in which case the aggregations illustrated in Section 2.2 apply. When, however, the summary is produced, the visualization component receives a very different dependency lattice than that used for the diagram in the previous example, one that is almost identical to that used as an example in Section 2.1 above. The diagram produced in this case is therefore also quite different and focuses instead on the regularities of teaching (i.e., profession, workperiod, and school), as shown in Figure 4. The text and diagram are therefore again appropriately coordinated.

6.4 Summary

The DArt$_{bio}$ prototype can produce similar pages to the one shown in Figure 19 for any set of artists selected by the user in the course of interaction with the system. The presentation environment is implemented in Smalltalk, the visualization and layout engines in C, and the text generation component in Common Lisp; page generation is in real-time. The **overanswering** present in any page is balanced by the use of a page to set out the information in a way that does not commit the user to reading it all. The layout process is flexible within the limits that are compatible with the goal of presenting a fixed amount of information on a page, but would require substantially more work to be sufficiently robust for a real-world system. The question of evaluation has not, therefore, been relevant: sufficiently fine-grained heuristics are in place to prevent layout disasters, but considerably more empirical work is now required in order to generalize these heuristics across differing domains and differing target document types.

7. Conclusions and Outlook

In this paper we have argued that many of the decisions that need to be made for an effective diagram are also found in the construction of an effective text. Making both sets of decisions dependent on a single, shared representation of regularities in a dataset provides a straightforward way of achieving consistency in the perspectives presented in the differing media: diagrams and texts then mutually reinforce one another by applying common information groupings. We have also argued that a message is provided by the physical layout of information on a page: relations of similarity, difference, and connection are commonly expressed by layout. In designing a page, therefore, layout relations must be made consistent with the overall communicative intent of the page. We have shown how consistency between layout and communicative intent can be achieved by deriving the former (layout structure) from a representation of the latter (rhetorically organized presentation plan). We have discussed both how such a derivation can be motivated and how it can be used for automatic page generation.

Our study of the relationship between the communicative structure of a page of information and the coherent layout of that information demonstrates that layout needs to be treated as an integral and complex part of the overall generation process; it can in no way be treated as a final piece of postprocessing. Many of the decisions required for segmenting a text effectively (e.g., into thematic paragraphs, into rhetorically related segments, etc.) also have correlates in the decisions that produce a coherent layout structure. This entails several areas of trade-off between layout and text: segmentation and grouping information may be expressed in language, in layout, or in some mixture of the two. By treating language expressions and layout as arising ultimately from a common source, we expect that potentially costly constraint-resolution at a local level (i.e., between arbitrary segments) can be avoided, or reduced, by enforcing consistent layout-language decisions for the layout structure of a page as a whole.

The last five years have seen rapid growth in the awareness of the importance of consistent and functional style selections: notions of style sheets from professional publishing have made their way into the mainstream of web-based document design—including consistent presentation formatting using Cascaded Style Sheets and flexible document rendering using, for example, the Document Style Semantics and Specification Language (DSSSL, ISO/IEC 10179) and the Extensible Stylesheet Language's Formatting Objects (XSL:FO; W3C). Professional bodies concerned with the closely related theme of Information Design—the application of *processes of design (that is, planning) to the communication of information (its content and language as well as its form)* (Waller 1996)—have also grown considerably both in number and membership. Moreover, the importance of some notion of rhetoric for professional design and layout is increasingly accepted (cf., Schriver [1996]). Less clear in such work is precisely how to characterize and teach useful notions of rhetoric. Our work on the rhetorical basis of layout interfaces directly with these developments, and establishes a new bridge between options for consistent microtypography at one end, right through to high-level communicative goal-based text design at the other.

Concretely, our work has highlighted the importance of further rounds of detailed empirical investigation, interlocked with critical evaluation. Our general mechanism for assigning the material of a rhetorical structure to a layout is highly nondeterministic—at virtually all points we need to appeal to heuristics to make a final decision. Our investigations so far indicate, however, that this is not a weakness in our formulation: the nondeterminism arises instead from the fact that the layout process is just as flexible as we describe it. The further restriction of the process so as to produce "appropriate" layouts can only proceed by establishing more motivated heuristics—and these heuristics will depend crucially on the particular applications, document types, target audiences, and informational content concerned. The rather heterogeneous set of heuristics currently adopted within the DArt$_{bio}$ system, for example, will need to be replaced by a framework of empirically motivated constraints; some of this work has now been started (Bateman, Delin, and Allen 2000; Delin, Bateman, and

Allen, forthcoming). The layouts produced will then need detailed evaluation—both from design experts and from users—as Schriver et al. (1996) should be the case for human-designed documents. Only in this way will it be possible to start refining the model of layout we have developed so as to even begin to cover the diversity and flexibility of layouts observable in professionally produced documents.

Acknowledgments

The work in this paper was carried out in the context of the KOMET-PAVE multimedia page-generation experiment (GMD-IPSI, 1994–1996). KOMET (Knowledge-oriented production of multimodal documents) and PAVE (Publication and advanced visualization environments) were two departments of the German National Research Center for Information Technology's (GMD) institute for Integrated Publication and Information Systems (IPSI) in Darmstadt that cooperated closely on this work. The authors of the present paper would therefore like to thank all the members of those departments who contributed, and particularly Lothar Rostek, Melina Alexa, Elke Teich, Wiebke Möhr and Christoph Hüser. A preliminary discussion presenting some background and explanatory material for Section 4 appeared as Reichenberger et al. (1996); we additionally thank Klaas Jan Rondhuis for his contribution to that work. The final form of this paper has been improved significantly by the detailed comments of three anonymous reviewers, whom we thank for their close reading and criticisms.

References

André, Elizabeth, Wolfgang Finkler, Winfried Graf, Thomas Rist, Anne Schauder, and Wolfgang Wahlster. 1993. WIP: the automatic synthesis of multimodal presentations. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press/The MIT Press, Menlo Park, CA, Cambridge, MA, London, pages 75–93.

André, Elisabeth and Thomas Rist. 1993. The design of illustrated documents as a planning task. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press/The MIT Press, Menlo Park, CA, Cambridge, MA, London, pages 94–116.

André, Elisabeth, Thomas Rist, and Jochen Müller. 1998. Guiding the user through dynamically generated hypermedia presentations with a life-like character. In *Proceedings of the 1998 International Conference on Intelligent User Interfaces (IUI'98)*, pages 21–28. Association for Computing Machinery.

Arens, Yigal and Eduard Hovy. 1990. How to describe what? towards a theory of modality utilization. In *The Twelfth Annual Conference of the Cognitive Science Society*. Lawrence Erlbaum Associates, Hillsdale, NJ, pages 487–494.

Arens, Yigal, Eduard Hovy, and Mira Vossers. 1993. On the knowledge underlying multimedia presentations. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press/The MIT Press, Menlo Park, CA, Cambridge, MA, London, pages 280–306.

Bateman, John, Judy Delin, and Patrick Allen. 2000. Constraints on layout in multimodal document generation. In *Proceedings of the First International Natural Language Generation Conference, Workshop on Coherence in Generated Multimedia*, Mitzpe Ramon, Israel. Association for Computational Linguistics.

Bateman, John, Thomas Kamps, Jörg Kleinz, and Klaus Reichenberger. 1998. Communicative goal-driven NL generation and data-driven graphics generation: an architectural synthesis for multimedia page generation. In *Proceedings of the 1998 International Workshop on Natural Language Generation*, pages 8–17. Niagara-on-the-Lake, Canada.

Bateman, John and Elke Teich. 1995. Selective information presentation in an integrated publication system: an application of genre-driven text generation. *Information Processing and Management*, (Special Issue on Summarizing Text), 31(5):753–768, September.

Bateman, John, Elke Teich, and Melina Alexa. 1998. Generic technologies for selective information presentation: an application of computational linguistic methods. In Peter Fankhauser and Marlies Ockenfeld, editors, *Integrated Publication and Information Systems: 10 years of research and development*. GMD, Forschungszentrum Informationstechnik, Sankt Augustin, Germany, pages 237–258.

Bernhardt, Stephen. 1985. Text structure and graphic design: the visible design. In James D. Benson and William S. Greaves, editors, *Systemic Perspectives on Discourse,*

Volume 1. Ablex, Norwood, NJ, pages 18–38.
Bouayad-Agha, Nadjet, Donia Scott, and Richard Power. 1996. Integrating content and style in documents: a case study of patient information leaflets. *Information Design Journal*, 9(2):161–176.
Corio, Marc and Guy Lapalme. 1998. Integrated generation of graphics and text: a corpus study. In M. T. Maybury and J. Pustejovsky, editors, *Proceedings of the COLING-ACL Workshop on Content Visualization and Intermedia Representations (CVIR'98)*, pages 63–68, Montréal, August.
Cristea, Dan, Nancy Ide, and Laurent Romary. 1998. Veins theory: a model of global discourse cohesion and coherence. In *Coling-ACL '98*, pages 281–285, Montréal.
Dalianis, Hercules. 1999. Aggregation in natural language generation. *Journal of Computational Intelligence*, 15(4):384–414, November.
Delin, Judy, John Bateman, and Patrick Allen. forthcoming. A model of genre in document layout. *Information Design Journal*.
Fasciano, Massimo and Guy Lapalme. 1996. PostGraphe: a system for the generation of statistical graphics and text. In *Proceedings of the Eighth International Workshop on Natural Language Generation (INLG '96)*, pages 51–60, Herstmonceux, England, June.
Fasciano, Massimo and Guy Lapalme. 2000. Intentions in the coordinated generation of graphics and text from tabular data. *Knowledge and Information Systems*, 2(3):310–339, August. URL: http://www.iro.umontreal.ca/ scriptum/IntentionsKAIS.ps.gz.
Feiner, Steven K. 1988. A grid-based approach to automating display layout. In *Proceedings of the Graphics Interface*, pages 192–197, Morgan Kaufman, Los Angeles, CA.
Feiner, Steven K. and Kathleen R. McKeown. 1993. Automating the generation of coordinated multimedia explanations. In Mark T. Maybury, editor, *Intelligent Multimedia Interfaces*. AAAI Press/The MIT Press, Menlo Park, CA, Cambridge, MA, London, pages 117–138.
Ganter, Bernhard and Rudolf Wille. 1996. *Formale Begriffsanalyse—Mathematische Grundlagen*. Springer-Verlag, Berlin/Heidelberg.
Graf, Winfried H. 1995. The constraint-based layout framework laylab and its applications. In *Proceedings of ACM Workshop on Effective Abstractions in Multimedia, Layout and Interaction*, San Francisco, California. ACM. URL: http://www.cs.tufts.edu/~isabel/mmwsproc.html
Green, Nancy, Guiseppe Carenini, Stephen Kerpedjiev, Stephen F. Roth, and Johanna D. Moore. 1998. A media-independent content language for integrated text and graphics generation. In *Proceedings of the Workshop on Content Visualization and Intermedia Representations (CVIR'98) of the 17th International Conference on Computational Linguistics (COLING'98) and the 36th Annual Meeting of the Association for Computational Linguistics (ACL'98)*, Montreal. Association for Computational Linguistics.
Green, Nancy, Guiseppe Carenini, and Johanna D. Moore. 1998. A principled representation of attributive descriptions for generating integrated text and information graphics presentations. In *Ninth INLG*, pages 18–27, Niagara-on-the-Lake, Canada.
Hasan, Ruqaiya. 1996. The nursery tale as a genre. In C. Cloran, D. Butt, and G. Williams, editors, *Ways of saying, ways of meaning: selected papers of Ruqaiya Hasan*. Cassell, London, pages 51–72.
Hovy, Eduard H. 1993. Automated discourse generation using discourse relations. *Artificial Intelligence*, 63(1-2):341–385.
Hovy, Eduard H. and Yigal Arens. 1991. Automatic generation of formatted text. In *Proceedings of the Eighth Conference of the American Association for Artificial Intelligence*, pages 92–96, Anaheim, CA.
Hüser, Christoph, Klaus Reichenberger, Lothar Rostek, and Norbert Streitz. 1995. Knowledge-based editing and visualization for hypermedia encyclopedias. *Communications of the ACM*, 38(4):49–51, April.
Kamps, Thomas. 1997. *A constructive theory for diagram design and its algorithmic implementation*. Ph.D. thesis, Darmstadt University of Technology, Darmstadt, Germany.
Kamps, Thomas. 1998. A constructive approach to automatic diagram design. In Peter Fankhauser and Marlies Ockenfeld, editors, *Integrated Publication and Information Systems: 10 years of research and development*, GMD, Forschungszentrum Informationstechnik, Sankt Augustin, Germany, pages 223–236.
Kamps, Thomas, Christoph Hüser, Wiebke Möhr, and Ingrid Schmidt. 1996. Knowledge-based information access for hypermedia reference works: exploring the spread of the bauhaus movement. In

Maristella Agosti and Alan F. Smeaton, editors, *Information retrieval and hypertext*. Kluwer Academic Publishers, Boston/London/Dordrecht, pages 225–255.

Kerpedjiev, Stephan, Guiseppe Carenini, Nancy Green, Johanna D. Moore, and Steven Roth. 1998. Saying it in graphics: from intentions to visualizations. In *IEEE Symposium on Information Visualization (InfoVis'98)*, pages 97–101, Research Triangle Park, NC, October. IEEE.

Kerpedjiev, Stephan, Guiseppe Carenini, Steven Roth, and Johanna D. Moore. 1997. Integrating planning and task-based design for multimedia presentation. In *Proceedings of IUI-97*.

Kittredge, Richard, Tanya Korelsky, and Owen Rambow. 1991. On the need for domain communication knowledge. *Computational Intelligence*, 7(4):305–314.

Mackinlay, Jock D. 1986. *Automatic design of graphical presentations*. Ph.D. thesis, Computer Science Department, University of Stanford, Stanford, CA.

Mann, William C. and Sandra A. Thompson. 1986. Relational propositions in discourse. *Discourse Processes*, 9(1):57–90, January-March. Also available as ISI/RR-83-115.

McKeown, Kathleen R. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press, Cambridge, England.

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

Moore, Johanna D. and Cécile L. Paris. 1993. Planning texts for advisory dialogs: capturing intentional and rhetorical information. *Computational Linguistics*, 19(4):651–694, December.

Pascual, Elsa. 1996. Integrating text formatting and text generation. In G. Adorni and M. Zock, editors, *Trends in Natural Language Generation: an artificial intelligence perspective*, number 1036 in Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, New York, pages 205–221. (Selected Papers from the fourth European Workshop on Natural Language Generation, Pisa, Italy, 28–30 April 1993).

RAGS Project. 1999. Towards a reference architecture for natural language generation systems. Technical Report ITRI-99-14 and HCRC/TR-102, Information Technology Research Institute (U. Brighton) and Division of *Informatics/Human Communication Research Centre (U. Edinburgh)*, Brighton and Edinburgh, March. Contributors: Lynne Cahill, Christy Doran, Roger Evans, Chris Mellish, Daniel Paiva, Mike Reape, Donia Scott and Neil Tipper.

Reichenberger, Klaus, Thomas Kamps, and Gene Golovchinsky. 1995. Towards a generative theory of diagram design. In *Proceedings of 1995 IEEE Symposium on Information Visualization*, pages 217–223, Los Alamitos, USA. IEEE Computer Society Press.

Reichenberger, Klaus, Klaas Jan Rondhuis, Jörg Kleinz, and John A. Bateman. 1996. Effective presentation of information through page layout: a linguistically-based approach. In *Proceedings of ACM Workshop on Effective Abstractions in Multimedia, Layout and Interaction*, Association for Computing Machinery. November 1995, San Francisco, CA. URL: http://www.cs.tufts.edu/\∼isabel/mmwsproc.html.

Rostek, Lothar, Wiebke Möhr, and Dietrich H. Fischer. 1994. Weaving a web: The structure and creation of an object network representing an electronic reference network. In C. Hüser and W. Möhr and V. Quint, editor, *Proceedings of Electronic Publishing (EP) '94*. Wiley, Chichester, pages 495–506. (Special issue of the International Journal of *Electronic publishing–origination, dissemination and design*, Volume 6(4).)

Schriver, Karen A. 1996. *Dynamics in document design: creating texts for readers*. John Wiley and Sons, New York.

Sefton, Petie M. 1990. Making plans for Nigel (or defining interfaces between computational representations of linguistic structure and output systems: Adding intonation, punctuation and typography systems to the PENMAN system). B.A. Honours Thesis. Technical report, Linguistic Department, University of Sydney, Sydney, Australia, November.

Southall, Richard. 1992. Presentation rules and rules of composition in the formatting of complex text. In C. Vanoirbeek and G. Coray, editors, *Electronic Publishing '92*, pages 275–290. Cambridge University Press, Cambridge, England.

Waller, Robert. 1988. *The typographical contribution to language: towards a model of typographic genres and their underlying structures*. Ph.D. thesis, Department of Typography and Graphic Communicatoin, University of Reading, Reading, U.K.

Waller, Robert. 1996. The origins of the Information Design Association. In *The*

1996 Annual Report of the IDA. Information Design Association.

White, Michael. 1995. Presenting punctuation. In *Proceedings of the Fifth European Workshop on Natural Language Generation*, pages 107–125, Faculty of Social and Behavioural Sciences, University of Leiden, Leiden, the Netherlands, 20–22 May 1995.

Wille, Rudolf. 1982. Restructuring lattice theory: an approach based on hierarchies of concept. In I. Rival, editor, *Ordered Sets*. Reidel, Dordecht/Boston, pages 445–470.