

Intentions in the Coordinated Generation of Graphics and Text from Tabular Data

Massimo Fasciano and Guy Lapalme

Département d'informatique et de recherche opérationnelle, Université de Montréal,
Montréal, Québec, Canada

Abstract. To use graphics efficiently in an automatic report generation system, one has to model messages and how they go from the writer (intention) to the reader (interpretation). This paper describes PostGraphe, a system which generates a report integrating graphics and text from a set of writer's intentions. The system is given the data in tabular form as might be found in a spreadsheet; also input is a declaration of the types of values in the columns of the table. The user then indicates the intentions to be conveyed in the graphics (e.g., compare two variables or show the evolution of a set of variables) and the system generates a report in L^AT_EX with the appropriate PostScript graphic files. PostGraphe uses the same information to generate the accompanying text that helps the reader to focus on the important points of the graphics. We also describe how these ideas have been embedded to create a new Chart Wizard for Microsoft Excel.

Keywords: Automatic graphics generation; Text generation; User intentions

1. Introduction: Important Factors in the Generation Process

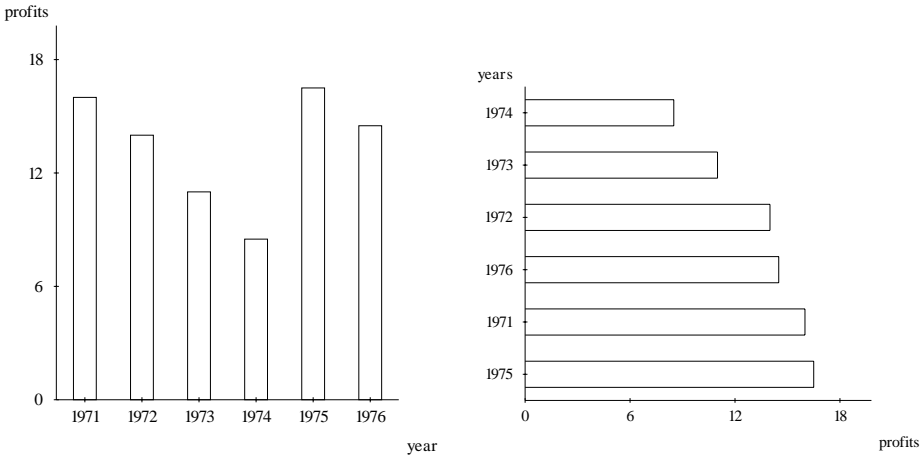
Graphics and text are very different media. Fortunately, when their integration is successful, they complement each other very well: a picture shows whereas a text describes. In this paper, we are studying the interaction between the text of a report and its graphical elements. Reports are an organized synthesis of data that span a whole array of forms going from tables of numbers to a text summarizing the findings. A reader might easily be overwhelmed by the raw data without an efficient organization and presentation.

Many sophisticated tools can be used to build a presentation using statistical graphs. However, most of them focus on producing professional-looking graphics without trying to help the user decide how to organize the presentation of the data.

Received 20 August 1998

Revised 2 September 1999

Accepted 27 September 1999



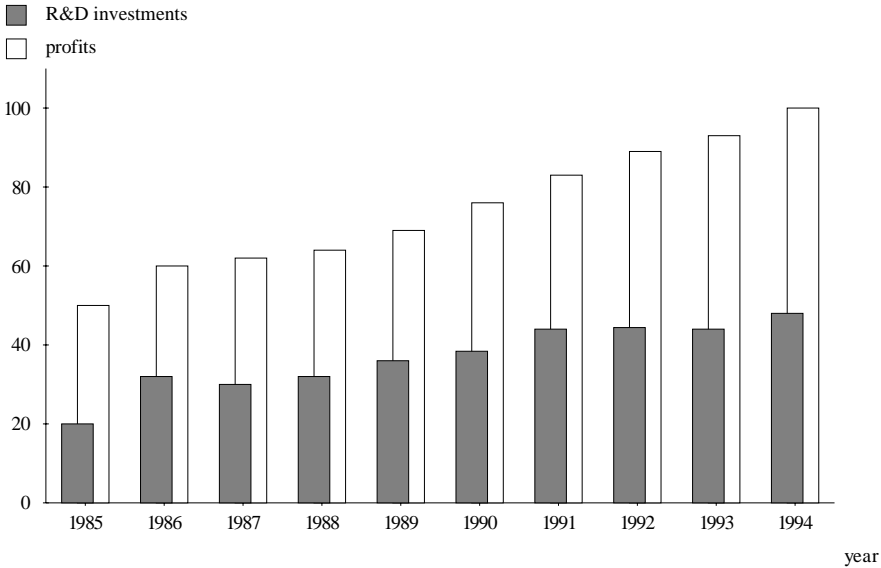
Globally, the profits have gone down despite a strong rise from 1974 to 1975.

The profits were at their highest in 1975 and 1971. They were at their lowest in 1974, with about half their 1975 value.

Fig. 1. Two communicative goals: evolution and comparison.

In particular, choosing the right graph or table for a task is not an easy process and subtle differences in internal structure can drastically change the perceived message. Our research is based on the assumption that to automatically generate a good statistical report the writer’s intentions or goals must be understood and used as the driving force behind the planning stages of the generator. The influence of these goals has to be weighed against data-related factors in order to produce a good presentation.

Writers’ goals have a major role in the generation process. In Fig. 1, the same set of data – profits of a company during the years 1971–1976 – is expressed in two ways depending on the intention of the writer. In the left part of the figure, the goal is to present the evolution of the profits, while the right part of the figure compares profits for the 6 years of the data set. The difference can be seen in the organization of the graphs and in the wording of the text. In the left graph, the evolution is emphasized by using the horizontal axis for the years. This is the accepted way of presenting temporal data. The years are sorted in ascending order to reinforce the impression of evolution. The associated text describes the overall evolution and points out an interesting irregularity. On the other hand, the writer’s intention for the right part of the figure is different. In order to show a comparison, some structural changes have been made. The years are presented on the vertical axis, thus eliminating the impression of evolution. This change is very important for the perception of the graph because it makes the message clearer by eliminating a false inference. Sorting the years according to the profit values has two positive effects: it further destroys the impression of evolution by making the years nonsequential and it allows a better comparison of the profits. The text, instead of describing how the profits evolved, merely points out the best and the worst years for profits. This presentation is not a standard one and we will see later that some authors disagree with it, but this difference in perspective is very important for a writer, especially when trying to convey more subjective



The graph above shows the evolution of the profits of a company and its investments in research and development between 1985 and 1994. Both have risen gradually during that period (50 and 28 millions respectively). It appears that the rise of profits is linked to the rise of investments in r&d. This is confirmed by the graph below.

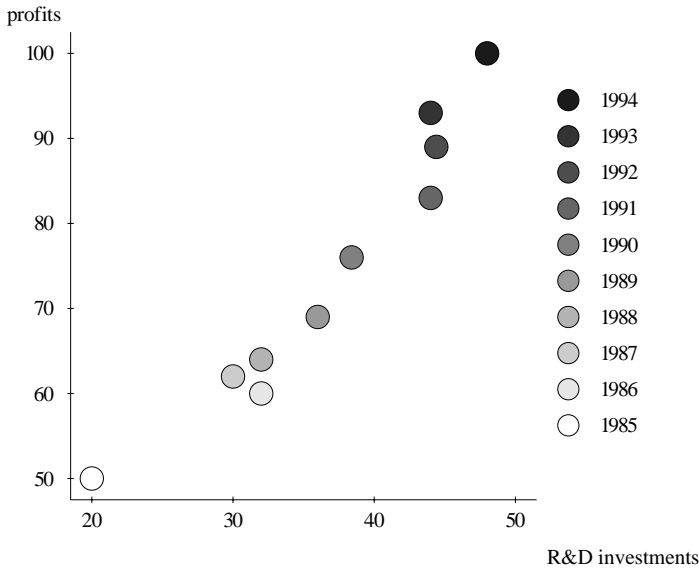


Fig. 2. Combined communicative goals: evolution and correlation.

messages. If the communicative goals aren't well identified, there is a risk of conveying the wrong impression to the reader.

This problem is further complicated by the fact that a single graph or text can convey many messages at once, some more direct than others. For example, Fig. 2 shows a situation where two graphs share a subset of intentions. The main

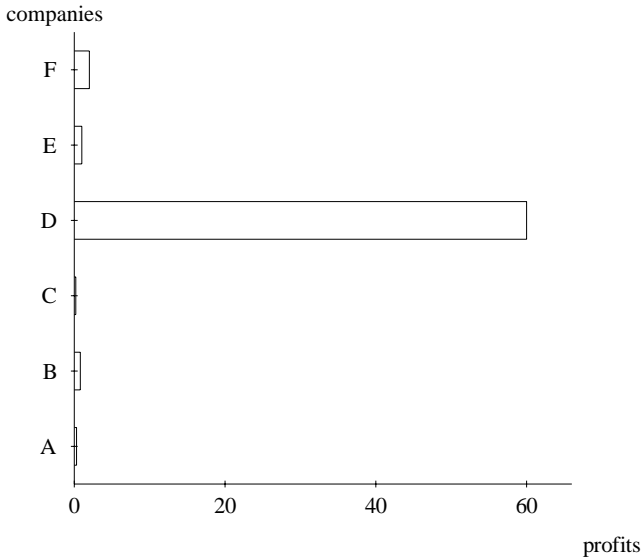


Fig. 3. Extreme ranges cause low readability.

message is one of evolution in the top graph and correlation in the bottom one, but both graphs also transmit, with lower efficiency, the main message of the other graph. Correlation is perceptible in the column graph because the two sets of data can be followed together and evolution can be perceived in the point graph by following the gray scale of the dots. Thus, determining which types of graphs or text best satisfy single goals is not sufficient; one also has to take into account the cumulative influence of the secondary messages conveyed by all parts of the report.

As might be expected, the types of variables give a lot of information about the structure of the elements of the report. For example, although a continuous variable is better represented by a line graph, the nature of a discrete variable will become more apparent using a column graph.

To further refine the selection process, we have to take into account not only the types, but also the specific values of the data samples. The number of values sometimes has a lot of influence on the choice of an expression schema. For example, a discrete variable with 200 values will often be treated as continuous, thus overriding the influence of its natural type. In other cases, the range of values has a strong influence. Indeed, as can be seen in Fig. 3, a seemingly good choice for comparison can be invalidated when the spread of values is large.

These factors influence the structure and contents of a statistical report and have to be looked at simultaneously to be effective. Some graphic generation systems use types to determine structure, but specific values are often overlooked and the combination of types and goals is extremely rare. To further illustrate the importance of simultaneous application of these factors, let's look at Fig. 3 again, where small values are not readable. In general, this is considered a problem and can be corrected by splitting the scale or using a logarithmic one. However, if the intention of the writer is to illustrate the enormous difference between company *D* and the others, the graph is very efficient as it is.

1.1. Graphical Elements of a Report

To create an efficient report from tabular data as might be found in a spreadsheet, choices must be made between modes of expression (text vs. graphics) by taking into account their strong points but also their effect on the reader.

Graphics are usually floating elements that do not appear just beside the accompanying text; they are often moved to satisfy other graphical constraints such as avoiding blank space at the end of a page. Graphics make important elements of the data stand out and catch the eye of the reader. The text of the report does not only describe or analyse data but it also links with the graphics by means of references to reinforce the intentions of the writer.

Graphics such as bar and column graphs, pie charts, line, or points plots are well known and often used because modern computer tools like spreadsheets (Microsoft, 1997) or statistical graphic packages make them very easy to produce from tabular data. There are many guidelines and tips (Cleveland, 1994; Tufte, 1983, 1990, 1997) for using graphics but statistical packages do not implement them; they merely reference them in their manual.

Text and graphic coordination poses important problems, identified in (Roth et al, 1991), such as

- *Structural incompatibility*: Text and graphics do not compose in the same way: for example, in a graphic representation of a tree, dependents are near the root, but in a pre-order textual description the links might be harder to make.
- *Lack of cohesion*: A text should make an explicit link with the structural elements of the graphics but this is rarely done.
- *Redundancy*: The text should not repeat information that is better conveyed by the graphics; although, in some cases, important information should be re-emphasized in the text.

In a report, the separation between text and graphics may not be as simple as one might think. For example, a table presents text or data in two dimensions for the reading of individual values; see Ehrenberg (1977) for guidelines for creating readable tables. Even within the text, graphical elements are used: font changing, emphasis by means of bold or italics, global organization in paragraphs or enumerated lists or page layout rules; Hovy (1993) discusses these aspects in the context of a text generator.

In the next sections, we will first present aspects of graphic design that influenced our system PostGraphe which emphasizes the importance of the writer's intentions. We will then show the adaptation of these ideas to build a new 'Chart Wizard' within the Microsoft Excel spreadsheet. We then compare PostGraphe with previous work on automatic report generation.

2. Graphic Design

The classic reference on the structure of graphics is Bertin (1983), who defines some basic elements of graphics and their properties. He presents graphics as a mapping between a number of variables and an invariant which links them. For example, in a relation between the price of a car and the year of its construction, the variables are *price* and *year* and the invariant is *car*.

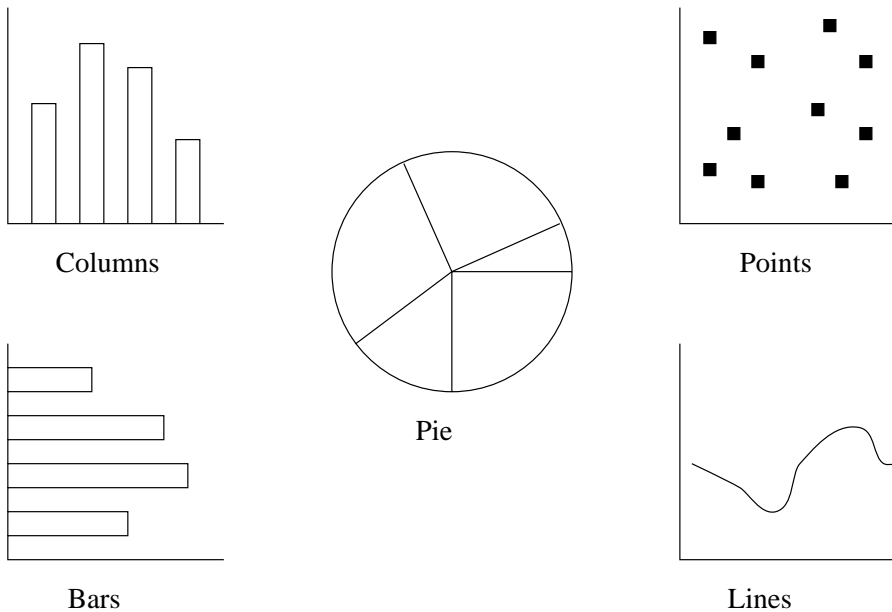


Fig. 4. Five basic graphics of Zelazny.

Bertin characterizes a graphic by its components each having a *length*, the number of possible values, and a *level of organization* (continuous or not, ordered or not). A graphic is also defined by the marks it contains having up to eight properties such as position in the plane, size, texture, and orientation.

Bertin defines the efficiency of a graphic in terms of the length of time it takes to decode the pertinent information and gives a set of rules for choosing the appropriate representation to get an efficient graphic. This work has had a deep influence on almost all major projects (Casner, 1991; Mackinlay, 1986b; Roth et al, 1991) in the field of automatic presentation.

A good reference on the effects of graphics upon a reader is Zelazny (1989, 1996) who defines a selection method for graphics depending on the information the writer wants to convey. Zelazny takes into account five types of elementary comparisons: decompositions (e.g. in percentage), positions (ordering of elements), evolution, repartition in categories and correlation. He then gives some elementary rules to go from the comparisons to five typical kinds of graphics, illustrated in Fig. 4: pie charts, columns, bar, and point graphs, which form the basic blocks upon which we build PostGraphe.

Cleveland (1994) identifies problems in human perception of graphics and proposes interesting solutions. He shows that some graphic techniques can fool our visual system and can thus send the wrong message. For example, it is quite hard to distinguish differences in angles or length or variations in slopes or length. This has serious consequences on the perception of the information conveyed by a graphic; for example, he shows that the ‘ubiquitous’ pie chart is not often appropriate for what it is used for. We have implemented many of these guidelines in our system.

Table 1. Two-level decomposition of simple intentions: V is a variable and S is a set of variables

| Objective/structural <i>How to say?</i> | Subjective/contents <i>What to say?</i> |
|--|---|
| Presentation (V) | |
| Comparison (S_1, S_2) | |
| Comparison/Fractional (V, S) | |
| Evolution (V_1, V_2) | Increase Decrease Stability Recapitulation |
| Correlation (V_1, V_2) | |
| Distribution (V, S) | |
| Distribution/Fractional (S) | |

3. Intentions and their Effect on the Choice of Graphs

As is shown in Fig. 1, the writer's intentions have an important effect on the way information is presented. Indeed, the figure shows the same data presented in two different ways according to what the writer wants to convey. One part of the example shows how to present the evolution of the data and the other part shows how to compare its various elements.

The writer's intentions can be classified according to two basic criteria: structural differences and content differences. Figure 1 illustrates the structural difference between evolution and comparison. We refer to intentions derived from structural differences as *objective intentions* and intentions derived from contents differences as *subjective intentions*. This definition stems from the fact that when the difference between two intentions is more content-related than structure-related, the writer is choosing what to say and not how to say it. The writer is thus making a subjective choice as to what is more important.

We have devised a two-level classification of intentions shown in Table 1 which extends Zelazny's (1989) classification of messages in order to deal with more complex graphics. Contrarily to Zelazny, we distinguish *intentions* as a way of describing the communicative goal of the writer and *message* for its content.

At the first level, our classification contains five categories:

- *Presentation*: The values of a variable must be accurately read on the selected presentation medium.
- *Comparison*: The values of a variable or set of variables must be compared using another set as reference.
- *Evolution*: The evolution of a variable along another one must be presented.
- *Correlation*: The relationship between two variables must be presented.
- *Distribution*: The distribution of the values of a variable in a set of classes must be presented.

Two categories have subcategories obtained by using a fractional modifier. For comparison, the fractional modifier indicates that the comparison should be done on fractions of the whole instead of the actual values. For distribution, we obtain a specialized intention where the classes are presented according to their fraction of the total.

At the second level, the intentions become specialized according to subjective

| | |
|-------------------------------|------------------|
| SI | simple intention |
| $[GI_1, GI_2, \dots, GI_n]$ | composition |
| $\{GI_1, GI_2, \dots, GI_n\}$ | superposition |

Fig. 5. General intention (GI) as a combination of simple intentions (SI).

criteria. We have only studied subjective variations of evolution. In the first three, *increase*, *decrease*, and *stability*, the data is presented in a way that emphasizes the desired property. For example, it is possible to present only periods of increase from a data set or to group the data in longer intervals to lessen the effect of variations. The last subjective intention, *recapitulation*, is a rather unconventional way of presenting temporal data: a nontemporal evolution. Although strange at first glance, this method is very effective in a text and takes a more global approach to evolution, trying to summarize long periods nonsequentially. For example, one might say that during a given period profits have increased during two fiscal years and decreased during five without saying when and by how much. All of these subjective intentions, especially the last, could be criticized as being dishonest but this is how people prepare real reports; see Section 3.2 for an example of these subjective categories.

These simple intentions (SI) can then be combined either by composition or superposition. In composition, the order of the variables is important and there is a dominant intention; for example, the comparison of evolutions is quite different from the evolution of a comparison, as in the following sentences: *Sales figures of Xyz increased less quickly than the ones of Pqr between 1992 and 1994* compares evolutions while *Pqr always stayed at the top except between 1992 and 1994* shows the evolution of the comparison. In superposition, the intentions are merely expressed using the same graphic but the intentions do not interfere. This is expressed by the notation given in Fig. 5.

Many combinations are possible but we noticed that comparison is combined more often than the others, even as a ‘side effect’ because the reader tends to compare visible objects even though it is not the direct goal of the message.

We will now describe the effect of structural or objective intentions on the choice of graphs and tables.

3.1. Objective Intentions

These intentions deal with structural differences between kinds of graphics.

3.1.1. Presentation

Two elements have to be taken into account for the expression of this intention: speed and accuracy. Indeed, an efficient presentation must not take too much time to interpret and the interpreted value must be as close as possible to the actual one. Speed is affected by separation and captions. Graphs in which markers are tightly packed, the extreme case being line graphs, slow down the isolation of individual values. On the other hand, in tables and bar/column graphs, the markers are well separated and it is easier for the reader to separate individual values. Captions become important in complex graphs where it is not immediately obvious to a reader what part of the graph corresponds to the variable looked for.

Accuracy depends on the encoding method used to represent values visually. A table is accurate because the values are not encoded in any way. The second best method is encoding by position on an axis, such as with the main variables in bar, column, point, and line graphs. One of the least accurate methods is encoding by angles in a pie chart. Other methods include varying the size, color, or shape of markers to convey additional information. The accuracy of these methods varies tremendously according to the nature of the data and is often hard to classify. For example, varying the area of the circles in a point graph can be misleading for representing lengths but it is appropriate for representing actual areas. Still, it is less accurate than using a position on an axis, which is easier to measure with eyes and ruler.

3.1.2. *Evolution*

Evolution is best presented using positional information with the temporal variable on the horizontal axis. This explains the predominant use of lines and columns to express this goal. Most other methods are much less efficient and are seldom used. For example, the bottom part of Fig. 2 presented temporal information using a gray scale in a point graph. Although the temporal information can be read and be useful, the notion of evolution is almost nonexistent because it is not ordered.

3.1.3. *Comparison*

For an efficient comparison, the markers to be compared should be aligned and the relative values implied by the visual encoding should correspond to the relative values of the entities. This last constraint is both looser and stronger than the accuracy constraint of the *presentation* intention. Indeed, for comparison, exact values are not required but, for reading, the relative sizes of the visual representations of objects are not important as long as the values they represent are clear.

Bars and columns are very good for two variable comparisons such as comparing the profits of companies, but bars are almost always preferred for two reasons: the length of columns tends to be overestimated (Graham, 1937) and columns give a strong impression of evolution which can generate a false interpretation of the data. Line graphs are more efficient for comparing multiple sets of variables, even though they also give a strong impression of evolution.

In general, pie charts are rather weak for comparisons because their sectors are not aligned on a common reference point. This makes it harder to compare similar values. The use of angles is not as big a problem for comparison as it is for presentation because relative values are preserved. In the case of fractional representations, pie charts are almost always used because their shape is very suggestive of decomposition. When multiple sets are to be compared in fractional representations, partitioned columns are much more efficient as they are semi-aligned and values can be more easily compared from one column to the other, which would be impossible with two pie charts side by side.

3.1.4. *Correlation*

Correlation shows if two (or more) variables are related and, if possible, by which type of relation (linear, exponential, ...). This goal is best achieved for two

variables by using a point graph with an optional regression line. By varying the scales on either axis, it is possible to present nonlinear relations.

Although this method is the most efficient by far, some other graphs are also suitable for the presentation of certain types of correlation, especially for more than two variables. For example, a multiple-line graph can show that a set of variables follows roughly the same trends and are thus correlated to some extent.

3.1.5. *Distribution*

To show a distribution, one has to decompose an interval into subclasses and visually present the number of values per class in a way that gives a global view of the process. The decomposition into classes can be presented efficiently in a column graph with tightly packed columns (histogram). The width of each column shows the size of the class and the overall shape of the graph gives a nice global perspective of the data. Line graphs can also be used for distributions. They tend to give a better overall shape but the classes are less visible, which makes them quite useful when the number of classes is high.

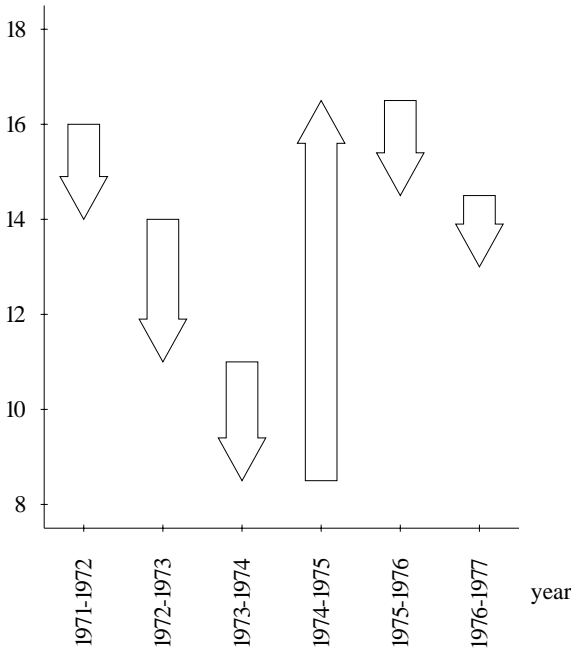
When a fractional representation is required, a pie chart can be used to convey the notion of parts. This should only be considered if the fractional aspect is very important, because a pie chart cannot show the width of the classes and the overall shape of the distribution, making it a rather weak substitute for the histogram.

3.2. Subjective Intentions

Subjective intentions deal with the content of the report and can modify the objective intentions presented in the previous subsection. The effects of a subjective intention can be much more subtle and reader dependent. A subjective graphic is more difficult to build than a subjective text because any omission is suspicious but, as we can see in Fig. 6, two similar graphics can convey two different messages. The upper part emphasizes variations and combines text and graphics in the usual way. But in the lower part, the message insists on the big increase and skips a series of decreases which in fact sum to a value greater than the increase. The choice of an appropriate graphic to accompany such a text is a bit tricky: if we omit the graph, then the reader might be dubious that we are hiding something; on the other hand, if this text is combined with the graphic shown in the upper part of Fig. 6 then we see that the text is dubious. But if we take the upper graphic and align all variations to 0, it is more difficult to sum all the decreases while showing all information so as not to raise any doubts. One might have serious reservations about the morality of such an approach but it should be clear that there are many situations in which the graphic and the text do give a subjective view of the data. In *PostGraphe*, we merely scratched the surface of this aspect but we are convinced that this aspect is a very important one that needs further investigation.

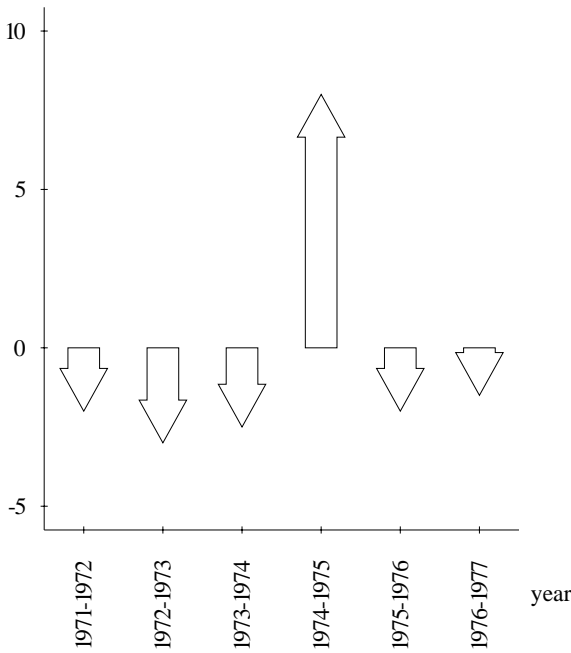
As we can see, there is no one-to-one correspondence between intentions and graphic strategies: many graphical ways of conveying a writer's intention and using a single graphic can combine more than one intention. In the next section, we describe how *PostGraphe*, our implementation of these ideas, chooses appropriate graphics and text to convey a group of writer's intentions.

profits



Globally, profits have decreased in spite of a strong comeback from 1974 to 1975.

profit variations



Profits had a strong increase from 1974 to 1975 and small decrease during the other years.

Fig. 6. Text and graphic integration for a subjective message.

4. An Automatic Report Generator: PostGraphe

We have built PostGraphe to test our model. The system is about 8000 lines of Prolog: 5000 for the graphics generation part and 3000 for the text generation module described in Section 4.4. The system produces a \LaTeX report with PostScript figures from annotated tabular data. We will now describe the major steps followed by the system in the generation of a report.

The input to PostGraphe consists of special annotations followed by the raw data. See Fig. 7 for an example of their use. These annotations indicate the names, the description, and the types of the variables, some of which may be of special interest to the user. There is also an indication of how to determine the relational keys for the data and a series of predicates describing the writer's intentions. The justification for these annotations and their Prolog syntax are presented in detail in Fasciano (1996). This format corresponds to columns in a spreadsheet having as first elements the name of the variable, as second its type, and as third an indication of whether the variable can be a key or not; the rest of the columns are the data values of the variable. Using a Prolog term with nine parameters is a rather crude way of inputting the data but this could be easily remedied using a graphical user interface or using a spreadsheet such as described in Section 5. But in this prototype, we decided to put more emphasis on the output part of our system.

4.1. Types

The type system associates a set of properties and a unit to every variable of the input. The properties are organized as a multiple inheritance graph divided into a number of subgraphs, each corresponding to a specific feature: organization (nominal, ordinal, quantitative, . . .), domain (enumeration, range, . . .), temporal (month, year, . . .), format (integer, real, . . .), measurements (distance, duration, . . .), and specific objects (countries, . . .). The properties have a variable number of parameters which can be used to further specify their function. For example, for an enumerated type (domain subgraph), a single parameter specifies the list of values for the enumeration.

In the input, the main type (or class) of each variable is specified, as well as a list of auxiliary types. The auxiliary properties override the ones that are inherited from the class, thus allowing the tailoring of built-in types. Also, a number of automatic type definitions are added according to the nature of the data (integers, labels, . . .).

Units are organized in a parallel inheritance graph. The inheritance mechanism is much simpler than the one used for types. A unit can be associated with a type (example: percentage \mapsto %). If a unit cannot be found using single inheritance, the name of the type is used as a unit. This process is described in more detail in Fasciano (1996).

4.2. Relational Keys

Relational keys are similar to the notion of the same name in relational databases (Date, 1988) and help determine which variables depend on which others. They

```

data(% names of the variables
  [année,compagnie,profits],      % [years, companies, profits]
  % 2-description of variables
  [année,compagnie,profits],      % [years, companies, profits]
  % 3- types of the variables
  [année/[symbolique],            % year/[symbolic]
  etiquette,                      % label
  dollar/[pluriel_de(profit)]],   % dollar/[plural_of(profit)]
  % 4-description of types of variables
  [année,compagnie,profits],      % [years, companies, profits]
  % 5- data of special interest to the user (none in this case)
  [ ],
  % 6- candidates for relational keys
  [année,compagnie],              % [years, company]
  % 7- non-candidates for relational keys
  [profits],                      % [years, companies, profits]
  % 8- the writer's intentions
  [ % section 1
  [presentation(année),           % presentation(year)
  presentation(compagnie),        % presentation(company)
  presentation(profits)],         % presentation(profits)
  % section 2
  [comparaison([profits],[compagnie]),
                                % comparaison([profits],[year])
  evolution(profits,année)],      % evolution(profits,year)
  % the raw data
  [[1987,'A',30],
  [1988,'A',35],
  [1989,'A',40],
  [1990,'A',35],
  [1987,'B',160],
  [1988,'B',165],
  [1989,'B',140],
  [1990,'B',155],
  [1987,'C',50],
  [1988,'C',55],
  [1989,'C',60],
  [1990,'C',95]])).

```

Fig. 7. Example of input to PostGraphe.

are also used for ordering variables in some graphics so that the more important ones (usually the keys) are given the more visible positions.

PostGraphe computes these keys from the data but the user can also add more constraints. Indeed, automatic calculation of keys can sometimes give strange results which do not fit the semantics of the variables. For example, a variable such as *profits* can wind up as a key if its values are all different but it is rarely desirable to express a set of variables such as *years* and *company names* as a function of *profits*. It is usually the other way around.

To solve this problem, two additional pieces of information can be specified in

the input: a list of variables that can be used as keys and a list of variables that should not be used as keys. Some control is thus maintained, while keeping the automatic calculation of keys. The ideal situation would be to have the system compute as much information as possible from the data in order to generate an appropriate graphic. But this goal clashes with another one, namely trying to use more complex relational information gathered from the database schema, for example, one to many or many to many. This would imply getting access to the database schema and, more important, finding appropriate graphic means of representing them. In PostGraphe, we limited ourselves to a spreadsheet type of data input for which we have studied the graphic implications for a statistical type report.

4.3. Writer's Intentions and Planning

The writer's intentions describe what to say and, up to a certain point, how to say it. This information is organized in lists that correspond to sections of the report. Intentions are constraints on the expressivity of the chosen text and graphics. PostGraphe tries to find the smallest set of graphics that covers the writer's intentions.

As was shown in Section 3, the following basic intentions are covered in our model: the *presentation* of a variable, the *comparison* of variables or sets of variables, the *evolution* of a variable along another one, the *correlation* of variables, and the *distribution* of a variable over another one. Some of these intentions are further divided into subjective subtypes.

PostGraphe uses a schema-based planning mechanism to generate both text and graphics. The planner uses the types and values of the data as well as the relational keys but it is mainly goal-driven. Because we have decided to use text and graphics for every message, planning is done in two phases: the graphical schemas are chosen and then the textual schemas are adapted to the contents and structure of the graphics. This separate planning of text and graphics might be questioned because it is often thought and said in the multimedia generation folklore and in some graphic generation texts that, to obtain a good interaction between text and graphics, text should give information that the graphics do not show. But Corio (1998; Corio and Lapalme, 1999) observed most often that the text merely reinforces what already appears in the graphic. For example, in 29% of texts associated with a comparison intention, there is a mention of the highest value as to say to the reader: 'Yes, what you see in this graphic is really what is important.'

From this information, our system determines which schemas best satisfy the writer's intentions. It starts from a set of intentions to satisfy; in doing so, no a priori ordering of the variables is given; all types of schemas have been assigned a weight for each intention; we can thus build a global quality function to be maximized. But instead of trying all groups of intentions to find the smallest subgroups of variables that best covers the writer's intentions, we use a set of heuristics.

We first find the intentions that are 'compatible' so that each schema takes into account as many intentions as possible while keeping each one 'readable'. Then we check if each group is feasible and determine the best schema to express it. This step is based on a table which associates the type of a variable with the most efficient graphical methods to express it. The table entries are weighted, and

| | | | | |
|-----------------------|----------|-----|--------|----------------------|
| comparison([A,B],[X]) | columns3 | 80 | [X Ys] | $A \in Ys, B \in Ys$ |
| comparison([Y],[X]) | bars1 | 100 | [X,Y] | |
| evolution(A, X) | curve2 | 100 | [X Ys] | $A \in Ys$ |

Fig. 8. Inference table excerpt.

the result of this phase is a list of candidates sorted from the most to the least efficient for the current goals. Figure 8 shows a three-entry excerpt from the table. The first entry indicates that the *columns3* schema (the third schema dealing with column graphs) satisfies the goal of comparing *A* and *B* along *X* with efficiency 80. The variable *X* and the list of variables *Ys* are involved in the process and *A* and *B* must be members of *Ys*.

The next step is the low-level generation of graphic primitives and text. If this stage determines that a figure cannot be generated because of physical reasons, such as being either too large or not having enough gray levels, the next best candidate is tested. This low-level work is quite involved because it has to take into account the 2-D constraints and the limitations of the media. For this we developed our own PostScript generation system in Prolog to determine and get access to the exact position of each element (character, line, axis, . . .) of a generated graph. This information is necessary for the references made within the text part of PostGraphe.

Finally, a postoptimization phase eliminates redundancies which can occur because the heuristics sometimes miss a compatible grouping of intentions.

4.4. Text Generation

The surface text generation is handled by a subsystem that we call SelTex (Corio, 1998; Corio and Lapalme, 1999), which captures the subtle nature of the interaction between text and graphics. Before designing SelTex, we did a corpus study of 411 French texts associated with graphics from such diverse sources as ‘Tendances sociales’ published every three months by Statistics Canada, books on statistics, investment funds reports, governmental reports, etc. We kept all texts associated with a graphic that could have been generated by PostGraphe but we rejected some generic titles giving no intentional message such as *Mean family income in 1995 constant dollars*; it is a shame to observe that these types of captions occur quite often in statistical reports. See Corio (1998) for more details on the process of collecting the corpus. For each text excerpt, we assigned a classification code indicating the kind of information that is transmitted and the way it is expressed. These codes were aggregated and we finally identified seven main themes for texts combined with graphics. Table 2 gives the frequencies of each theme for the intentions described in Table 1. We now briefly describe each theme.

Descriptive gives an overview of the graphic or identifies its main visual aspect: for example, using a title or a legend, it describes the data on the *X* or *Y* axis or the general tendency (increase or decrease).

Focusing messages (named quantitative in Corio, 1998) select the raw data that should interest the reader.

Domination expresses the highest or lowest values of the data such as *Which company made the most or the least profit?*

Table 2. Counts of themes and intentions of messages in our corpus of 411 French texts; some texts carry more than one intention and theme

| Theme | Intention | | | | | | Total | % |
|---------------|--------------|------------|-----------|-------------|--------------|----|-------|-----|
| | Presentation | Comparison | Evolution | Correlation | Distribution | | | |
| Descriptive | 98 | 6 | 30 | | | | 134 | 30 |
| Focusing | 23 | 46 | 30 | | | | 99 | 22 |
| Domination | | 65 | 3 | | | 17 | 85 | 19 |
| Deductive | 11 | 16 | 5 | 33 | 3 | | 68 | 15 |
| Discriminant | | 7 | 31 | | | | 38 | 9 |
| Qualitative | | 2 | 8 | 3 | | | 13 | 3 |
| Justificative | | 4 | | 1 | | | 5 | 1 |
| Total | 132 | 146 | 107 | 37 | 20 | | 442 | 100 |
| % | 30 | 33 | 24 | 8 | 5 | | 100 | |

Table 3. Selection rules for a comparison

| | Comparison |
|------------------------|--|
| Largest value | if its value is more than 10% higher than the second largest and there are more than 2 values |
| Two largest values | if the second highest value is more than 15% higher than the third largest and there are more than 4 values |
| Three largest values | if the third highest value is more than 20% higher than the fourth one and there are more than 6 values |
| Smallest value | if its value is at least 30% lower than the second smallest one and there are more than 2 values |
| Balance between values | if the ratio between the largest and smallest value is less than 1.2 and there are more than 3 values |
| Largest value | if none of the above |

Deductive messages draw a conclusion from the shape of the graphics or the values of the data.

Discriminant messages identify a particular fact that distinguishes this value from the others: we show an irregularity, a turning point in a curve or an exception in an otherwise constant situation.

Qualitative messages describe data in words such as *rare, weak, strong, frequent, high, low*; the shape of the curve can also be given. Here the judgment of the writer has the highest influence because the same value can qualified differently depending on the context.

Justificative messages identify causes for phenomena such as *Why is a bar the highest?, Why did the Canadian dollar fall?, Why does a given political party have more voting intentions?*

After studying the corpus, we have identified text type selection rules for a writer’s intentions. We also studied the comparison–evolution combination. We have implemented text schemas for 26 out of the 55 (47%) types of texts in our corpus but the ones we chose were the most frequent ones and they cover 71% of all texts in our corpus. All sentences given in this section were generated by SelTex, followed here by an English translation, done manually.

Table 4. Selection rules for showing an evolution

| Evolution | |
|-----------------------|---|
| Increase/decrease | if all values are greater/smaller than the preceding one |
| Most recent tendency | if there was a reversal of tendency |
| First and last values | if there were many reversals of tendency |

Table 5. Selection rules for showing the combination of comparison and evolution.

| Comparison and evolution | |
|---|---|
| Evolution of the gap between curves | if there are two groups of serial data or the user is interested in two specific items of data |
| Comparison of the last values of each curve | if there were many reversals of tendency |
| Enumeration of the evolutions of each group | if otherwise |

Comparison. From the information shown in Table 2, we can see that comparison is one of the most frequent uses of information graphics, so we studied it quite extensively.

Therefore, for the comparison intention, we determined from our corpus study the empirical rules given in Table 3.¹ The tests are done on the values of the data while excluding categories such as *others* and *total* that are often found in information graphics.

Evolution. Evolution occurs also quite often, and for this we use the rules given in Table 4 except when there are very big changes. For example, if the inflation rate in Canada from 1989 to 1995 were

{5.0%, 4.8%, 5.6%, 12.0%, 1.8%, 0.2%, 2.1%}

then it would not be appropriate to compare the values of 1989 and 1995 because of the extremely high value in 1992. In this case, we only mention the final value such as in *Le taux d'inflation au Canada se situe à 2.1% en 1995 (Inflation rate in Canada is at 2.1% in 1995)*. In the case of a reversal of tendencies, we only pinpoint this fact; but if there were many reversals, we only give the starting and ending values. This is justified by the fact that the graphic shows the evolution over a long period of time, while the text pinpoints what we think is of most interest to the reader: the last news.

Comparison and evolution combination. We distinguish two kinds of combination:

evolution of comparison such as in *Le pourcentage de ménages possédant un ordinateur au Québec se rapproche de celui de l'Ontario (The percentage of households with a home computer in Québec approaches that of Ontario)*

comparison of evolution such as in *Le taux de ménages possédant un ordinateur à domicile augmente plus rapidement à Terre-Neuve qu'à l'le*

¹ These rules are tried from top to bottom and SetTex outputs the first value on the left for which the conditions on the right are satisfied.

du Prince-Édouard (*The percentage of households with a home computer increases faster in Newfoundland than in Prince Edward Island*)

Correlation. To determine if there is a statistically significant correlation between two sets of data, we use a standard linear correlation computation and then indicate if there is a positive or negative relation between the variables. But great care must be used in indicating positive or negative relations between variables so that they do not give false implications. For example, the following was encountered in our corpus: L'opinion des femmes quant à l'importance d'occuper un emploi rémunéré pour être heureuses varie selon l'âge (*The opinion of women about the importance of having a paid job to be happy varies according to age*). Should we have written *change avec l'âge* (*changes with age*), we could have implied that women modify their opinion as they get older.

Distribution. In this case, we only implemented the case that indicates the interval where the values are highest.

Presentation. SelTex creates generic titles from the data description given by the user; if there are temporal data, then it also gives the starting and ending values such as in *Taux d'inflation au Canada de 1989 à 1995* (*Inflation rate in Canada from 1989 to 1995*).

4.5. An Automatically Generated Report

Figure 9 shows the output from our system PostGraphe from the input given in Fig. 7. The input is formulated as a Prolog term with nine arguments. As can be seen in the annotated input, the first two arguments are lists of variables and their description which will be used in the text part; the third and fourth arguments are lists of types/classes of their variables and their description in words; the fifth argument might be used for specifying a variable which would be of special interest to the user; the sixth and seventh specify the functional dependencies of the variables or how to derive them automatically, the eighth is the writer's intentions presented in groups and the last is the data itself in tabular format. The example presents the profits of companies *A*, *B* and *C* from 1987 to 1990. The intentions are divided in two sections: the first presents the three variables (*year*, *company* and *profits*). The second presents the comparison of the profits between companies and the evolution of the profits over the years.

PostGraphe generates French reports using L^AT_EX (Lamport, 1994) as its document description language and PostScript (Adobe Systems Inc., 1990) for figures. The output of the current system contains a number of special annotations that would not be present in a finished product. These annotations are mainly designed as debugging aids and they indicate what schema was used in each graphic and what intentions were covered by it. Every intention is given a score (0–100) indicating how well it was realized by the system. These scores are very helpful in analyzing the system's choices. We give here a manual translation of the text parts of the report.

In the first section of the report, PostGraphe chose a table to present the variables. Every value is perfectly readable but no analysis is implied, as was the intention of the writer. In the second section, the system chose to overlay three line graphs in the same figure in order to best satisfy the group of two

Nouvelle section (3 intentions à traiter).
(New section: (3 intentions to process))

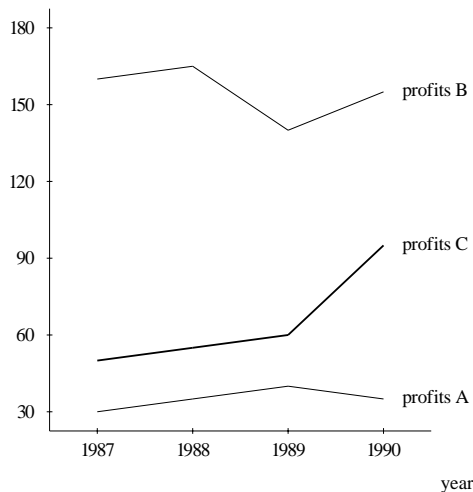
nouvelles intentions: présentation de année. présentation de compagnie.
 présentation de profits.(*new intentions: presentation of year. presentation of company.
 presentation of profits*)

| year | 1987 | 1988 | 1989 | 1990 |
|---------|---------|---------|---------|---------|
| company | profits | profits | profits | profits |
| A | 30 | 35 | 40 | 35 |
| B | 160 | 165 | 140 | 155 |
| C | 50 | 55 | 60 | 95 |

[Schéma: *table1*]. présentation de année (100). présentation de compagnie (100).
 présentation de profits (100).(presentation of year (100). presentation of company
 (100). presentation of profits (100).)

Nouvelle section (2 intentions à traiter).
(New section: 2 intentions to process)

nouvelles intentions: comparaison de profits entre compagnie. évolution de profits
 par rapport à année.(*new intentions: comparison of profits between companies.
 evolution of profits by year*)



[Schéma: *curve3*]. comparaison de profits entre compagnie (69). évolution de
 profits par rapport à année (95).

[Schéma: *evolution2*]. évolution de profits par rapport à année (100).

Les profits de la compagnie B se situent à 155 comparativement à 160 en 1987.
 Les profits de B sont plus élevés que les autres compagnies en 1990. (*Company
 B's profits are 155 compared to 160 in 1987. B's profits are higher than those of
 other companies in 1990.*)

Fig. 9. Report generated by PostGraphe from input of Fig. 7.

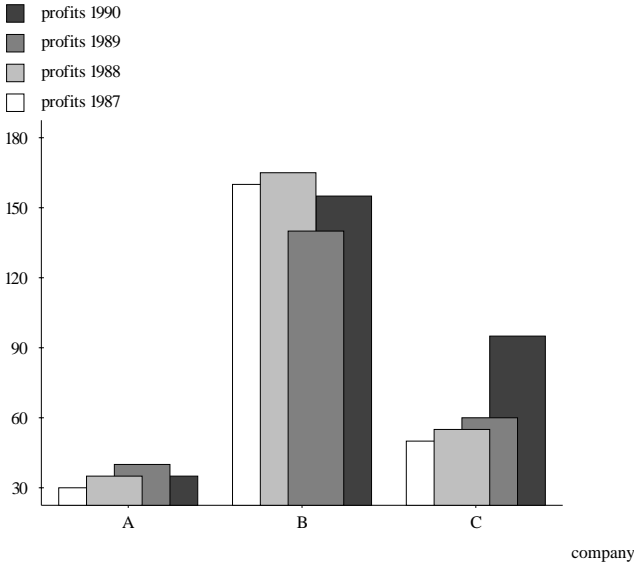


Fig. 10. [Schéma: *columns2*]. comparaison de profits entre compagnie (80). évolution de profits par rapport à année (60). (*comparison of profits by company (80). evolution of profits by year (60).*).

intentions specified by the user: comparison and evolution of the profits. The overlaid line graphs were chosen because, according to our knowledge base, they represent the best compromise between comparison and evolution. As can be seen in the figure’s caption, the two intentions were expressed with scores of 69 and 95 respectively. The score for comparison isn’t very good but the very high score for the other intention brings the average up to 72, which beats all other choices. The text part is generated according to the rules of Tables 4 and 5.

The weight system used in the choice of schemas allows the user to alter the behavior of PostGrappe by using special annotations in the input. These two annotations (> and *) are used to specify a threshold of acceptability and a weight to the expression of an intention. The default values used by the system are >50 for the threshold and *1 for the weight. Thus, the second group of intentions for Fig. 7 could be written as follows:

```
[comparaison
 ([profits],[compagnie])>50*1,
                               % comparison([profits],[company])>50*1,
 evolution(profits,année)>50*1] % evolution(profits,year)>50*1
```

By changing the weight of the first intention, we modify its importance in the calculation of the best schema. For example, if the weight is multiplied by 10, Fig. 10 will be produced instead of Fig. 9. The configuration of the columns in this figure facilitates comparison and still presents evolution adequately. The scores of 80 and 60 give it a global value of 78.2 under the specified weight distribution, whereas the score for Fig. 9 was 71.4. With the regular weights, the scores would have been 70 and 72, thus giving the line graphs a slight edge.

```
[comparaison
 ([profits],[compagnie])*10, % comparison([profits],[company])*10,
 evolution(profits,année)] % evolution(profits,year)
```

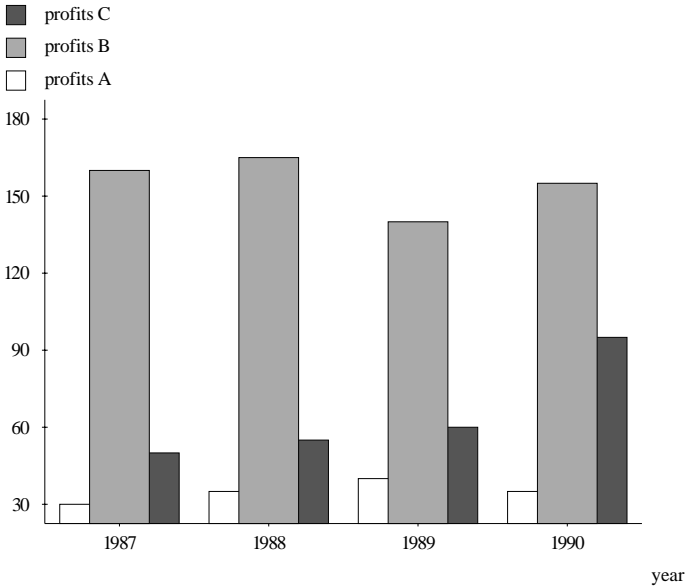


Fig. 11. [Schéma: columns2]. comparaison de profits entre compagnie (70). évolution de profits par rapport à année (85). (comparison of profits between companies (70). evolution of profits by year (85)).

In the following example, we keep a weight of 10 for comparison and add a threshold of 65 for evolution.

```
[comparaison
([profits],[compagnie])*10, % comparison([profits],[company])*10
evolution(profits,année)>65] % evolution(profits,year)>65
```

This makes Fig. 10 unsuitable because its score for evolution is 60. Figure 9 has already been eliminated because of the weight on comparison. The system will generate Fig. 11 in order to satisfy both weight and threshold.

If the weight of 10 is replaced by a threshold of 80, the system cannot generate a single figure that satisfies all constraints.

```
[comparaison
([profits],[compagnie])>80, % comparison([profits],[company])>80,
evolution(profits,année)>65] % evolution(profits,year)>65
```

It then produces two independent figures using the same graphs as Figs 10 and 9. Note that the high degree of freedom of the preceding examples would have been eliminated by working on the profit average instead of the profits. Thus, the following intentions

```
[comparaison % comparison of mean profits by company
(reduce(moyenne,profits),[compagnie]),
evolution % evolution of mean profits by year
(reduce(moyenne,profits),année)]
```

would have led to the generation of two independent figures (Figs 12 and 13). The grouping of these two intentions is impossible, which eliminates the variations shown in the preceding examples.

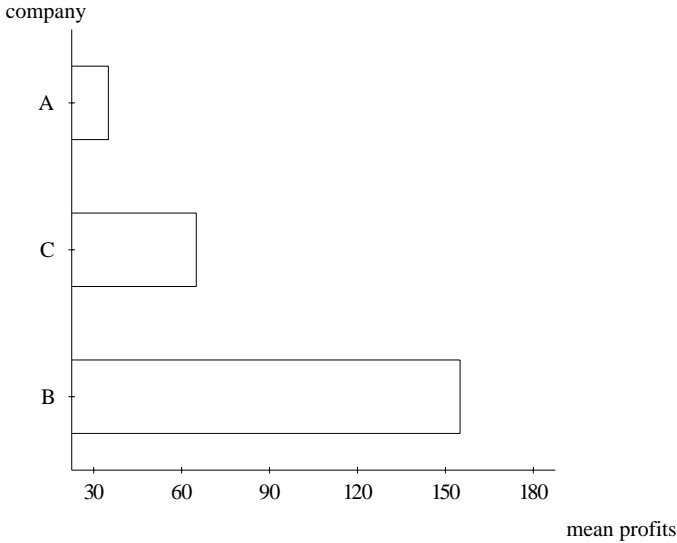


Fig. 12. [Schéma: bars1]. comparaison de la moyenne de profits entre compagnie (100). (*comparison of mean profits between companies (100)*).

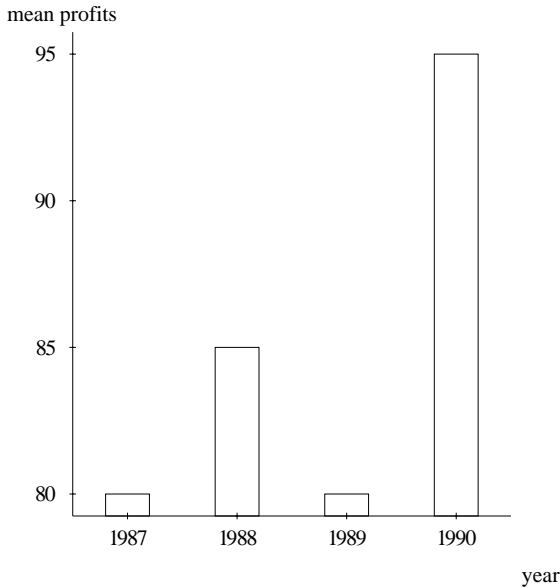


Fig. 13. [Schéma: columns1]. évolution de la moyenne de profits par rapport à année (94). (*evolution of mean profits by year (94)*).

This section has shown several uses of PostGraphe; see Fasciano (1996) for more examples. Even with its 21 graphical schemas, PostGraphe does not give all the possibilities of other statistical packages but the choices we have made do cover the graphics most often encountered in practice and transmit the writer's intentions clearly and precisely. PostGraphe only generates 2-D graphics because

we have not explored the consequences that choices of a perspective point and a projection have on the reader. This would have implied the development of new graphic schemas for this situation. Many of the so-called 3-D graphics are merely an embellishment of 2-D graphics by adding an artificial depth. These modifications can distort lengths and distances and thus can modify the intentions conveyed by the graphics. This is why we have decided to produce less flashy graphics but ones for which we know the messages they convey.

5. Adaptation to a Spreadsheet

In our project, we always had in mind the kind of tabular data found in spreadsheets. As we were aiming for an integrated text and graphic generation, we decided to use a Prolog environment for our experiments; six years ago, when we started this project, the macro languages of spreadsheets and their programming environments were also quite rudimentary. We were happy with this choice that gave us rapid prototyping, but now that we have a working system we have decided to adapt our intention-based paradigm to the context of a real spreadsheet. We have chosen Excel (Microsoft, 1997) because of its availability on both Macintosh and PCs and because of the facilities for adding new functionalities using Visual Basic, which now offers a very good programming environment.

Excel already has powerful tools for creating (too) many kinds of graphics from tabular data, but they focus on the form of the graphic; the choice of an appropriate one is left to the experience and good taste of the user. Normally, to create a graphic from a range of data cells, the user calls a *Chart Wizard* which asks questions about the data, the form of graphic the user wants and the title of the graphic. Once a graphic is generated, the user can go back and change almost every aspect of it: background, tick marks, legend, title and even the global form; for example, change a histogram to a pie chart. As the user selects the graphic, one could be tempted to think that there is some kind of user intention involved but in fact the user only selects the form and not the message. We now present how we adapted the ideas presented in this paper to build a new kind of *Chart Wizard* for Excel which is simpler to use once the user has determined the message to convey with the data. Figure 14 shows how the user indicates his intentions from which the Postgraphe Chart Wizard generates the appropriate graph shown in Fig. 15.

For this implementation, we made a few simplifications from the ideas described in this paper: Postgraphe Chart Wizard deals only with one intention at a time, and does not try to find a graphic covering more than one intention; the wizard only generates graphics but no accompanying text except for the title of the graphics, which transliterates the intention and the names of the variables. Using Excel enabled us to use its predefined graphic possibilities and we did not use the Prolog-based Postscript generator. One other very useful feature is the linking between the data and the graphic: a change in a value is immediately reflected in the graphic provided this change does not change the type of graphic.

The Postgraphe Chart Wizard focuses on the goal of the user, who only needs to specify his/her intentions and the data to be presented and not how it should be done. Because of this, the Postgraphe Chart Wizard validates the data and often catches errors that might have produced an inappropriate graphic or even go unnoticed. Our system does not use all types of graphics available in Excel

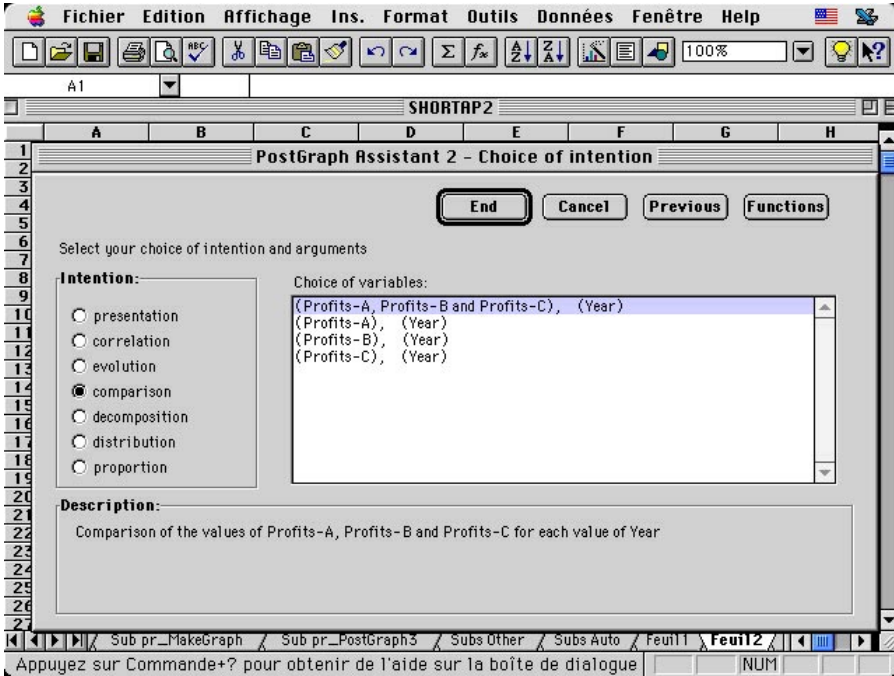


Fig. 14. Form for specifying the intention in the Postgraphe Excel Wizard.

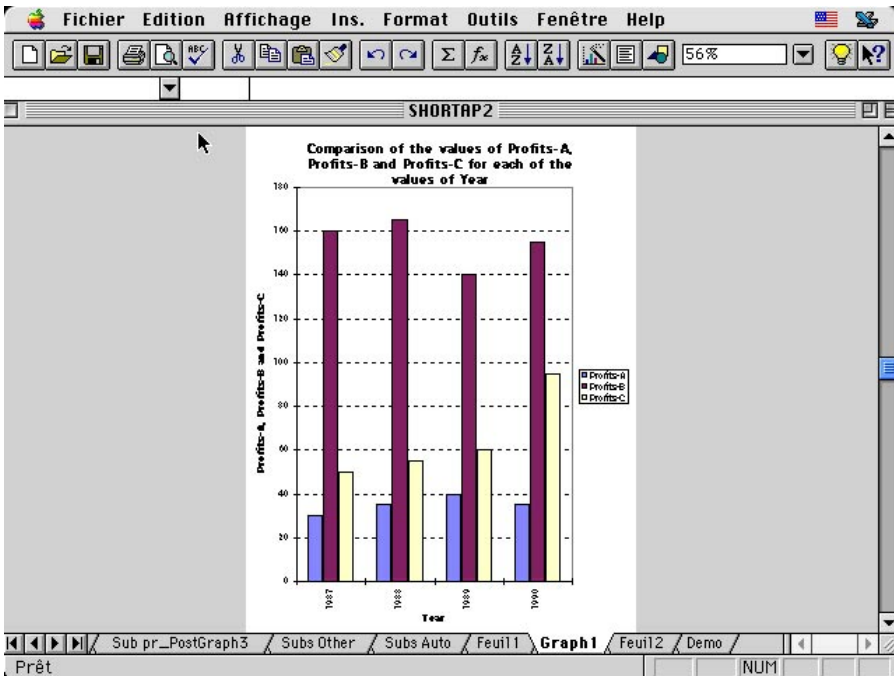


Fig. 15. Graph generated by the Postgraphe Excel Wizard.

(3-D, radar, picture shaped, . . .) because we have not studied the intentions they convey and their effects on the user. On the other hand, we added a few new features, namely a much better control on the shape and size of a tick mark for which a change in area or color can help to convey some intentions involving three or four variables. We also added the notion of series of variables of the same type which can be combined on the same graphic; PostGraphe detects these variables automatically and PostGraphe Chart Wizard also.

This experience has shown that the idea of taking the user's intention into account can be transposed from one research environment to a more widespread context. We are convinced that we could never have realized the importance of the user's intention in Excel should we not have developed PostGraphe beforehand.

6. Related Work

We now compare PostGraphe with previous work on automatic graphics and text generation which can be classified into three categories: systems that generate only graphics, those that deal with text only, and those which integrate both media. For a more complete survey, see Roth and Hefley (1993).

6.1. Automatic Graphic Generation

MacKinlay (1986a, 1986b) built the APT system, which generates statistical graphics from ground principles instead of choosing between predefined templates. He identified and classified a set of *graphic encodings* to represent the value of a variable. For example, the value of a market share of a company can be represented by the *position* of a point along an axis, by an *angle* in a pie chart or by the *area* of a circle. These encodings are a superset of the graphic properties of Bertin that we described in Section 2. MacKinlay then associated with these encodings some *primitive graphic languages* that are used to construct more complex graphics; for example, a graphic primitive associated with a position is an *axis*. He also determined a set of expressivity criteria for the encoding techniques; i.e., which types of relation and variables can be represented by a technique. To build a graphic, APT tries to use the most efficient encoding technique but sometimes there are conflicts between variables because they cannot all use the most efficient technique if they appear together; backtracking is then used to find an acceptable combination of encodings. This composition of graphics can thus extend the number of applicable methods and increase the efficiency of the result. APT can even align two graphics that share a compatible axis. APT only tries to optimize the graphic efficiency and does not take the user intention into account; it always gives the same graphics for the same input variables. On the other hand, PostGraphe can generate different graphics depending on the user's intention.

Feiner (1987) has built the APEX system, which generates one picture for each subtask of a graphic explanation of how to fix a piece of broken equipment. APEX determines automatically the objects to add to the picture and the style and the level of detail with which they should be rendered, the method by which the action should be indicated, and the camera's specification. This approach is quite involved and has served as a basis for future multimedia systems that we will describe later. APEX does take into account the writer's goal in the sense that it is based on a plan but it is a task plan and not the intention. Our system

has a quite different application area and is focused on tabular data which have quite different properties but they are not necessarily easier because the objects manipulated by APEX have already a predefined 3-D description and no choice has to be made in this respect.

Casner (1991) has built BOZ, a system that designs graphics based on an analysis of the task they are intended to support. For example, in the problem of using airline schedule information to locate flights obeying certain criteria, BOZ proposes customized graphic representations depending on whether we want to find the cheapest flight or the most direct one. The graphic design focuses on finding efficient perceptual procedures to be performed by humans. The input to BOZ can thus take into account some intention of the writer, who uses a Pascal-like procedural language for describing the task; this input is quite involved and is far from the raw data. In PostGraphe, we ask very little supplementary information for our graphics and only a declaration of the intention to be conveyed.

The graphic generator called IBIS (Seligmann and Feiner, 1991) generates pictures to fulfill a *communicative intent* such as showing the location of an object, its physical properties (size, form, color), its state, or its change of state. To build an illustration IBIS uses style strategies such as highlighting, hiding, or rotating objects that are defined by design rules that map the intent to the graphical means. This approach is one of the first attempts at taking into account the writer's intention. But the intention of the writer is defined in terms of the appearance of the objects on the illustration and not in terms of his/her goal (comparison, evolution, etc.) as in PostGraphe. IBIS has been integrated within the COMET system, to be described in Section 6.3.

6.2. Automatic Text Generation

Text generation is now well established and many systems consider the reader's model and the writer's intention within the 'what to say' module, the other module being 'how to say it'.

TEXT was developed by Kathleen McKeown (1985) to answer questions on the structure of a database. It is based on rhetorical schemas and has inspired many text generation systems, including ours. Some systems have focused on report generation, namely ANA by Karen Kukich (1983), which generates stock market reports directly from the Dow Jones indicators. LFS (Iordanskaja, *et al*, 1992) produces reports in both French and English on the *active population* of Canada; it has also been adapted for generating reports on detail sales and consumer report price index. It can generate good reports in the style expected from a government agency. The text plans are quite rigid and there is no adaptation to a goal or style of a reader. LFS can also generate a predefined format of graphics where only the values can change but not the type of graphics. In PostGraphe, we explicitly take the writer's intention into account to generate a customized graphic for each report.

6.3. Graphic and Text Integration

Some large-scale multimedia projects integrate text and graphic generation for applications such as maintenance explanations, changes in an evolving system,

or presentations of every-day appliances. Maybury (1993) is a good source of information for those systems.

COMET (Feiner and McKeown, 1991) generates multimedia explanations (graphics and text) answering user requests about the maintenance and repair of military radio equipment. The explanations are based on a knowledge base about an object, on possible actions, on diagnostics, and on geometrical informations; they also take into account the goal and level of knowledge of the user. The system uses rhetorical schemas (McKeown, 1985) to choose what is to be said and then decides what will be expressed graphically or textually: graphics are used to emphasize the spatial properties, while the text expresses abstract relations and actions. Text and graphics are then generated in parallel using a hierarchy of *logical forms* organized in an extended functional unification grammar formalism. Text generation takes into account the user's knowledge for choosing the level of language of the explanations. Graphics are generated using IBIS, described in Section 6.1. Then both text and graphics are put together for a final display; in PostGraphe we also generate text and graphics in parallel.

SAGE (Roth et al, 1991) was designed to analyse changes in an evolving system: it can deal with either models with hierarchical dependencies or spreadsheet type models. For example, in a business environment, it might be interesting to illustrate that a small department at the bottom of the hierarchy might be one major cause of profit decrease in the last semester. The first module of SAGE determines the content of the message as being the changes in the data that are relevant to the user's request. Roth and Mattis (1990) describe the rationale behind the graphic module, which is an extension of APT that takes into account the notion of goal (the user's request), but also new types of data (composite types and missing data), new graphic types such as the Gantt chart, showing the starting and ending dates of activities, and new composing capabilities: it can combine trees with statistical graphics. The text module of SAGE is inspired by the text schemata of TEXT (see Section 6.2). Emphasis was put on strategic generation (structure and content selection) instead of final text output. About 10 new rhetorical schemata were defined such as *change*, *cause-effect*, *reinforcement*, or *dependency*. Like in expert systems explanation systems (Weiner, 1980), SAGE must heavily prune the dependency trees to create understandable texts. SAGE often generates quite complicated graphics that are not immediately obvious to the user. Mittal, *et al.* (1998) recently extended SAGE to produce a caption to interpret the structure of the graphic. This is a quite different approach from multimedia in the sense that it describes the form instead of the content of the graphic. SAGE was one of the first systems to introduce the notion of user intention in the automatic generation of graphics but it does this in the context of graphics showing relations between parts of systems and not for the usual statistical graphics found in PostGraphe.

WIP (André et al, 1993) generates coordinated multimedia presentations of appliances from a common representation. These presentations are customized to a user's expertise and language and to the context of use (quick repair or general information). Content determination is done using RST rhetorical relations (Mann and Thompson, 1988) and the medium is dynamically chosen using *presentation strategies* indicating the chosen medium, its effect, its application conditions, and eventual substrategies. Planning is done by finding plans that unify with a goal to achieve. Subplans are dealt with recursively until primitive presentation strategies are found. Low-level modules are then used for the final realizations and display.

WIP's results and approach, like COMET's, are very important in the field of multimedia explanations. Their goal is the generation of graphic explanations starting from a given annotated picture in which the system decides which parts to emphasize and under which perspective its components should be visualized to satisfy the goal of the presentation. But in our case, PostGraphe generates a statistical report starting from raw data and builds the graphics from the ground up; it must not only decide which elements to present but also the best way to draw them and how to put them together to get the best possible result. COMET has a more precise user model than the one used in PostGraphe because it has to deal with less precise references to parts of graphics: it describes real objects with references to some parts. In PostGraphe, references are more simply to data series with marks and axes.

These systems have shown the importance of the coordination between text and graphics in the form of a conscious separation between the two modes directly from the writer's intentions. PostGraphe has managed this integration directly from a single declaration in a restricted but important area – tabular data encountered in many reports, many of which could be automatically generated.

7. Conclusion

In order to achieve the integrated generation of text and graphics in statistical reports, we have considered simultaneously the writer's goals, the types and values of the variables to be presented, and the relations between these variables. We also integrated good design rules that have a direct influence on the reader's perception of a report. Thus, one has to consider the writer's goals, the data itself and the reader's interpretation. These ideas have been implemented in a prototype system that generates reports from tabular data. We also adapted these ideas for a new type of Excel Chart Wizard that focuses on the user's intention instead of the form of graphics. This work has shown some of the complexities that arise even in the case of seemingly simple information graphics. We think that these principles are equally important, if not more so, in the case of more complex multimedia presentations or intermedia visualizations. For the moment, PostGraphe has shown its potential in information graphics but, in the future, we intend to apply the same principles to other domains where qualitative information is encountered or in the case of graphics for other kinds of application (e.g., instructional texts).

8. Acknowledgements

We would like to thank Marc Corio, who improved the text generation module that was originally in PostGraphe. We also thank Guy St-Denis and Francis Fauteux, who worked on the adaptation of the ideas of PostGraphe to build versions of the Postgraphe Excel Wizard. Thanks also to suggestions of anonymous referees which helped improve this text. This project has been partially funded by student grants from FCAR (Gouvernement du Québec) and NSERC (Government of Canada) and a research grant from NSERC.

References

- Adobe Systems Inc. (1990) PostScript language reference manual (2nd edn). Addison-Wesley, Reading, MA
- André E, Finkler W, Graf W, Rist T, Schauder A, Wahlster W (1993) WIP: the automatic synthesis of multimodal presentations. In Maybury MT (ed). *Intelligent multimedia interfaces*. AAAI Press, Cambridge, MA, pp 75–93
- Bertin J (1983) *Semiology of graphics* (transl Berg WJ). University of Wisconsin Press, Madison, WI
- Casner SM (1991) A task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics* 10(2): 111–151
- Cleveland WS (1994) *The elements of graphing data* (rev edn). Hobart Press, Summit, NJ
- Corio M (1998) *Sélection de l'information pour la génération de texte associé à un graphique statistique*. Master's thesis, Département d'informatique et de recherche opérationnelle, Université de Montréal
- Corio M, Lapalme G (1999) Generation of texts for information graphics. In 7th European workshop on natural language generation, Toulouse, May 1999, pp 49–58
- Date CJ (1998) *An introduction to database systems*, vol I (4th edn). Addison-Wesley, Reading, MA
- Ehrenberg ASC (1977) Rudiments of numeracy. *Journal of the Royal Statistical Society* 140(3): 277–297
- Fasciano M (1996) *Génération intégrée de textes et de graphiques statistiques*. PhD thesis, Université de Montréal
- Feiner SK (1987) *Computer generation of pictorial explanations*. Technical Report CS-87-30, Brown University, Providence, RI, December 1987
- Feiner S, McKeown K (1991) Automating the generation of coordinated multimedia explanations. *Multimedia Information Systems* 24(10): 33–41
- Graham JL (1937) Illusory trends in the observation of bar graphs. *Journal of Experimental Psychology* 6(20): 597–608
- Hovy EH (1993) Automated discourse generation using discourse structure relations. *Artificial Intelligence* 63: 341–385
- Iordanskaja L, Kim M, Kittredge R, Lavoie B, Polguère A (1992) Generation of extended bilingual statistical reports. In Proceedings of the 15th international conference on computational linguistics (COLING '92), pp 1019–1023
- Kukich K (1983) Design of a knowledge-based report generator. In Proceedings of the 21st annual meeting of the ACL, Massachusetts Institute of Technology, Cambridge, MA, 15–17 June 1983, pp 145–150
- Lampert L (1994) *L^AT_EX, a document preparation system* (2nd edn). Addison-Wesley, Reading, MA
- Mackinlay JD (1986a) *Automatic design of graphical presentations*. PhD thesis, Computer Science Department, Stanford University
- Mackinlay JD (1986b) Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics* 5(2): 110–141
- Mann W, Thompson S (1988) Rhetorical structure theory: towards a functional theory of text organization. *TEXT* 8(3): 243–281
- Maybury MT (ed) (1993) *Intelligent multimedia interfaces*. AAAI Press, Cambridge, MA
- McKeown KR (1985) Discourse strategies for generating natural language text. *Artificial Intelligence*, 27(1): 1–41
- Microsoft (1997) *Excel 97*. Microsoft Press, Redmond, WA
- Mittal VO, Moore JD, Carenini G, Roth S (1998) Describing complex charts in natural language: a caption generation system. *Computational Linguistics* 24(3): 431–467
- Roth SF, Hefley WE (1993) Intelligent multimedia presentation systems: research and principles. In Maybury MT (ed). *Intelligent multimedia interfaces*. AAAI Press, Cambridge, MA, pp 13–58
- Roth SF, Mattis J (1990) Data characterization for intelligent graphics presentation. In *Empowering people: CHI '90 conference proceedings*, April 1990, pp 193–200, ACM Seattle
- Roth SF, Mattis J, Mesnard X (1991) Graphics and natural language as components of automatic explanation. In Sullivan JW, Tyler SW (eds). *Intelligent user interfaces*, Frontier series, ch 10. ACM Press, New York, pp 207–239
- Seligmann D, Feiner, S (1991) Automated generation of intent-based 3D illustrations. *Computer graphics* 25(4): 123–132
- Tufte ER (1983) *The visual display of quantitative information*. Graphics Press, Cheshire, CT
- Tufte ER (1990) *Envisioning information*. Graphics Press
- Tufte ER (1997) *Visual explanations*. Graphics Press
- Weiner J (1980) BLAH, a system which explains its reasoning. *Artificial intelligence* 15: 19–48

Zelazny, G (1989) *Dites-le avec des graphiques*. InterÉditions, Paris

Zelazny G (1996) *Say it with charts: the executive's guide to visual communication* (3rd edn). McGraw-Hill, New York

Author Biographies



Massimo Fasciano completed his Ph.D. at the Université de Montréal in 1996, working on the automatic generation of graphics taking into account user intentions. He is now working as a system administrator in the Computer Science and Operations Research Department of the Université de Montréal.



Guy Lapalme is a professor in Computer Science at the Université de Montréal. He has been working in the field of natural language processing for more than 15 years, especially in text generation. He is also interested in functional programming and its applications to the field of computational biology.

Correspondence and offprint requests to: Guy Lapalme, Département d'informatique et de recherche opérationnelle, Université de Montréal, CP 6128, Succ Centre-Ville, Montréal, Québec H3C 3J7, Canada. Email: lapalme@IRO.UMontreal.CA