Integrating Planning and Task-based Design for Multimedia Presentation

Stephan Kerpedjiev[1], Giuseppe Carenini[2], Steven F. Roth[1], Johanna D. Moore[2]

[1]Robotics Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213 USA
kerpedjiev,roth@cs.cmu.edu

[2]Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260 USA
carenini,jmoore@cs.pitt.edu

ABSTRACT

We claim that automatic multimedia presentation can be modeled by integrating two complementary approaches to automatic design: hierarchical planning to achieve communicative goals, and task-based graphic design. The interface between the two approaches is a domain and media independent layer of communicative goals and actions. A planning process decomposes domain-specific goals to domain-independent goals, which in turn are realized by media-specific techniques. One of these techniques is task-based graphic design. We apply our approach to presenting information from large data sets using natural language and information graphics.

Keywords

Multimedia presentation, information seeking tasks, media allocation, information graphics, presentation planning.

INTRODUCTION

Understanding large data sets and explaining them to others via effective displays is a complex, laborious and time consuming activity. Analysts and other types of specialists on a daily and sometimes hourly basis explore complex data sets, prepare memos for themselves, brief their upper management, or inform peers of their observations, hypotheses, and conclusions. Their work would be greatly facilitated if a tool could extract the relevant pieces of information and present them in an appropriate way. This could take the form of a textual summary of the most important aspects of the data, a graphic elucidating an important trend, or a multimedia presentation combining text, graphics, photos, video, etc. What are the principles that guide people in choosing one or another method? How can we model presentation decisions so that they can be made by intelligent software? These questions have guided our design of a framework for generating multimedia (natural language and information graphics) presentations of quantitative and relational information.

In prior work, two complementary views to automatic presentation generation have emerged. Researchers from the natural language processing community [12,18] focus on the communicative intent of a presentation, and have modeled presentation design as a process of hierarchical planning to achieve communicative goals. In contrast, researchers in graphics emphasize the need to model the perceptual and logical tasks the user needs to perform [3,4,13], and have built computational systems that design a presentation that can support a given set of tasks. Designing effective multimedia presentations requires that both types of knowledge be used in the presentation design process, and our work seeks to integrate the planning and task views in a single coherent framework.

Our goal is to develop an approach to generating multimedia presentations that achieve communicative goals. In our framework, a planner is used to refine domain-specific communicative goals (e.g., *know-shortfalls*) into domain- and media-independent subgoals, such as *know-attribute*. These in turn are achieved by domain- and media-independent abstract actions (e.g., *assert*, *activate*) which are ultimately decomposed into media-specific actions (e.g., *inform*, *enable-lookup*). For example, one way to achieve the *know-attribute* goal is by decomposing the corresponding action *assert* into the graphical action *enable-lookup*. This decomposition embodies the rule that if there is a graphic on which the user can effectively look up the values of the attribute, then the goal *know-attribute* has been achieved. Media-specific actions are then executed by the various media generators to produce the actual text and graphics that make up the presentation.

The existence of a domain-independent layer of communicative goals from which we can generate multimedia presentations eases the process of developing systems in different domains. For a new application, the knowledge engineer need only provide decompositions from the domain-specific communicative goals to the domain and media independent goals. In addition to this practical benefit, capturing the communicative goals in both language and graphics provides a common descriptive model for studying multimedia presentations. We expect this model to enable us to provide a more general approach to media allocation and coordination than has previously been developed. In addition, this approach is the first attempt to model the communicative aspects of information graphics via general media-independent goals that are mapped to information-seeking tasks.

Our efforts continue a series of similar projects aimed at conceptualizing the design principles of multimedia presentations in a domain independent way. Among the applications previously addressed are instructions for operating physical devices [7, 18], explanations of quantitative models [14], route directions [10], and weather reports [9]. The system we are developing is intended to aid users in identifying important information that is contained in large data sets. Our system has neither complete knowledge of the domain, nor complete knowledge of what it is the user needs to know in order to solve the problems they are interested in. Therefore, the system plans its summaries, comparisons and correlations in a way that allows users to perform further analyses using the presentations.

In this paper, we briefly describe the domain in which we are developing our system. We then illustrate our approach with several sample presentations. We describe the communicative model and clarify its connections with both the planning process and the graphical tasks. We then work through an example of our system designing a sample presentation, and briefly describe heuristics for media allocation. Finally, we outline the graphics generator and relate our work to similar projects.

THE DOMAIN OF TRANSPORTATION PLANNING

The testbed for our system is the domain of transportation scheduling. The problem is to create a schedule with minimum lateness that transports cargo of certain types from a set of origins to a set of destinations during given intervals with given resources (capabilities): planes, ships and ports. Transportation analysts and planners routinely use automatic scheduling systems to produce numerous schedules that meet requirements with resources, analyze them with respect to causes for lateness and suggest workarounds. This process requires maintaining versions of the schedules, analyzing them, communicating with fellow analysts and members of upper management, and exploring changes to resource assumptions that would reduce the lateness (e.g., allocating additional crews to reduce a bottleneck). Our goal is to propose a new type of system-generated briefing that helps transportation planners organize their work and communicate the state of their analyses to others involved in the process.

For our experiments we use DITOPS, an automatic scheduling system developed at Carnegie Mellon University [17]. It applies constraint satisfaction methods to a set of requirements and transportation resources and produces a schedule, which may or may not have lateness.

To represent the output of DITOPS, our group developed a domain model, which is a Loom [8] KB consisting of about 60 concepts and 120 relations. Examples of concepts are *port* and *fleet*. Examples of relations are *has-destination* (a relation that specifies the arrival points for a schedule) and *cumulative-required-cargo* (a relation that specifies the total tons of cargo that need to be transported by a given date of a schedule).

Because of the novel character of the briefings we want to generate, we manually crafted a detailed scenario in HTML and Java and used it as a tool for communicating our approach to domain experts and soliciting feedback from them. The following examples, which are part of this scenario, illustrate our approach and show the complexity of the presentations we are modeling.

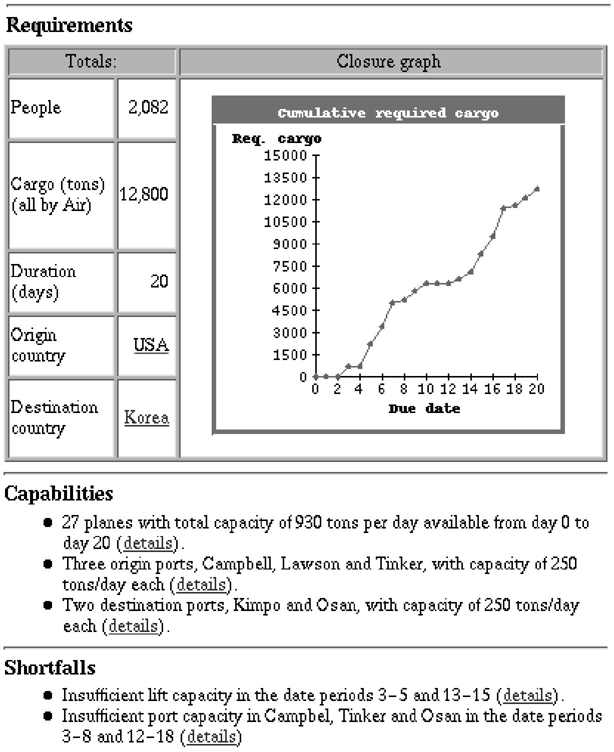
OUR APPROACH

Explaining large data sets requires a balance between presenting only relevant information, justifying its relevance, and supporting exploration of the excluded information. Thus, we characterize the explanation of large data sets as a combination of the following four aspects:

- **Content planning.** The system must select a limited amount of relevant information out of the potentially very large number of facts available in the KB.
- **Communicative goals.** In addition to selecting content, communicative goals and the strategies for achieving them direct the system in presenting this content in a way that emphasizes specific aspects. For example, in our domain shortfall is defined as a situation in which requirements exceed capabilities. Therefore, achieving the goal *know-shortfall* requires that we provide a presentation in which the user can identify and analyze these situations.
- **Perceptual tasks.** Some of the communicative goals can be better satisfied by enabling the user to perform certain perceptual tasks on a graphic, instead of simply being informed of the outcome of some automatically performed analysis.
- **Planning exploratory links.** Since our system is intended to support users in performing their analyses, it should provide convenient means for enabling them to request presentations of related information.

We now illustrate these aspects using our scenario presentations. Fig. 1 shows a presentation about a schedule. Since during the course of a single day analysts may produce numerous schedules, the first thing they typically want to know about a schedule is summary information about its requirements, capabilities, and possible shortfalls. This particular selection and organization of attributes is accomplished by a domain-specific strategy of achieving the goal *know-schedule.*

While most of the attributes in Fig. 1 are conveyed through simple summary statements (e.g., the total number of people), for the attribute *cumulative-required-cargo* the communicative goals are more complex. The user must be able to identify periods of rapid increase in the amount of required cargo as well as dates by which a certain portion of the required cargo is scheduled to arrive at the destination ports. Some of these communicative goals cannot be expressed in language as effectively as they can be expressed by the graphic in Fig. 1. The line graph not only enables the user to lookup the values of the attribute (a table could do this as well or even better), but also to scan the development of the graph for steep line segments indicative of rapid increase of the cumulative cargo or flat segments indicative of slow or no increase. Note that the system does not have to know whether there are rapid increases; by enabling users to scan the graphic, it enables them to discover this on their own. The user can also easily divide the y-axis by a certain portion of the total cargo, find the point where the imaginary horizontal line corresponding to this amount crosses the line graph, and check the x-position of that point, thus finding the date by which this amount of cargo should be at the destination ports. This presentation illustrates how different communicative goals can be assigned to an attribute and satisfied by enabling perceptual tasks such as search, scan and lookup.



Requirements

Totals:		Closure graph
People	2,082	
Cargo (tons) (all by Air)	12,800	
Duration (days)	20	
Origin country	USA	
Destination country	Korea	

Capabilities

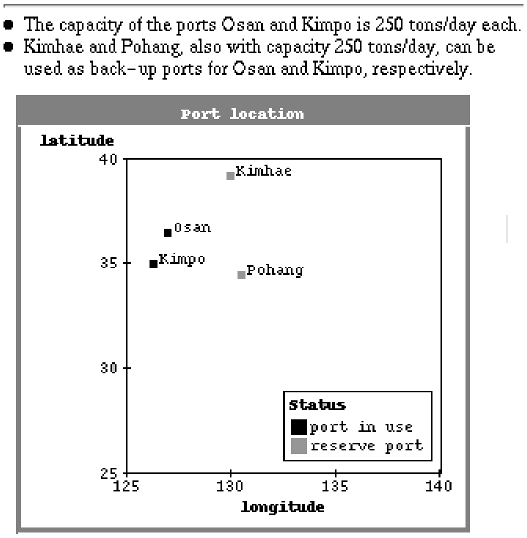
- 27 planes with total capacity of 930 tons per day available from day 0 to day 20 (details).
- Three origin ports, Campbell, Lawson and Tinker, with capacity of 250 tons/day each (details).
- Two destination ports, Kimpo and Osan, with capacity of 250 tons/day each (details).

Shortfalls

- Insufficient lift capacity in the date periods 3–5 and 13–15 (details).
- Insufficient port capacity in Campbel, Tinker and Osan in the date periods 3–8 and 12–18 (details)

Figure 1. A summary presentation of a schedule

In addition to providing information about various attributes of the schedule, the presentation in Figure 1 enables the user to request more information by making certain portions mouse sensitive (mouse sensitive phrases are underlined in Figures 1-3). Associated with each sensitive object, which can be a phrase or a graphical symbol, is a new goal. A mouse click on such an object is interpreted as a request by the user for a presentation that satisfies the goal associated with it. For example, the word "details" right after the sentence saying that the schedule has two destination ports (the third bullet in the capabilities section) is associated with the domain-specific goal of knowing all capability-related attributes for these ports. If the user clicks on this word, the system will plan the presentation shown in Fig. 2, which provides detailed information about the locations and capacities of the destination ports Kimpo and Osan, as well as of two reserve ports. Planning these hypertext-like links is an important element of our approach.

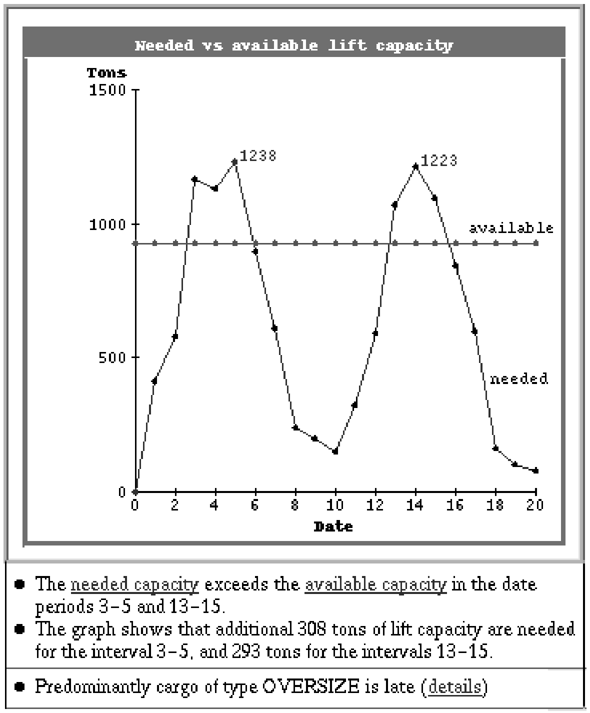


- The capacity of the ports Osan and Kimpo is 250 tons/day each.
- Kimhae and Pohang, also with capacity 250 tons/day, can be used as back-up ports for Osan and Kimpo, respectively.

Figure 2. A detailed presentation of the destination ports of a schedule

The following example shows how the combination of the basic features of our approach allows the user to perform a complete analysis of the cause for a shortfall. There are two common causes for shortfall: insufficient port capacity and insufficient lift capacity. Each of these types can be refined into more concrete ones, e.g., lack of aircraft suitable for cargo of a certain type. The goal of the analyst is to diagnose the shortfall and to suggest workarounds. Our domain model contains definitions of various causes for shortfall in terms of relations between certain attributes. Thus, if a schedule results in lateness, the system will prepare displays that convey aspects of the data suggesting the cause for the lateness.

The presentation in Fig. 1 summarizes two shortfalls in the current schedule. The short textual statements identify the shortfalls by their types and periods, and enable the user to explore them in more detail by providing mouse sensitive phrases. When the user clicks on the word “details” just after the statement about lift undercapacity (the first bullet under “Shortfalls” in Fig. 1) the system plans the presentation in Fig. 3, which satisfies the goal *know-lift-shortfall*. The system achieves this domain-specific goal with a presentation strategy that specifies the appropriate domain-independent communicative subgoals, and then applies media-specific techniques to achieve these. The line graphs allow the user to compare the amount of cargo that the fleet can carry on each date with the expected amount of cargo that needs to be transported on each date. The text makes specific points about the difference between the two attributes, e.g., the amount of additional lift capacity that is needed to eliminate the shortfall. This display helps the user answer the questions “How much additional capacity is needed and when it is needed?” The third bullet summarizes the distribution of the late cargo by the observation that predominantly cargo of type “oversize” is late, and enables the user to request a more detailed view. To diagnose the lift undercapacity the user clicks on the “details” phrase and a breakdown of the lateness by cargo type and date is presented graphically as in Fig. 4. This graphic shows that the major lateness occurs for cargo of type “oversize” immediately after the periods of lift shortage, and suggests that insufficient fleet capable of carrying oversize cargo could be the problem.



- The needed capacity exceeds the available capacity in the date periods 3-5 and 13-15.
- The graph shows that additional 308 tons of lift capacity are needed for the interval 3-5, and 293 tons for the intervals 13-15.

- Predominantly cargo of type OVERSIZE is late (details)

Figure 3. Comparison of two attributes

Thus with a sequence of three displays the system helped the analyst to obtain an overview of the schedule, to drill down into lift related information, to explore a sufficiently refined hypothesis for the cause of the lateness, and to come up with workarounds such as increasing the fleet that can carry oversize cargo in the day periods 3-5 and 13-15[1].

MODELING THE INTENT OF PRESENTATIONS

Our model of the generation process consists of different types of communicative actions and goals. Since these goals and actions are applied to sets, in this section we first describe how we represent sets, and then present the ontology of communicative goals and actions.

[1] In transportation planning the dates are relative with respect to the beginning of the schedule.

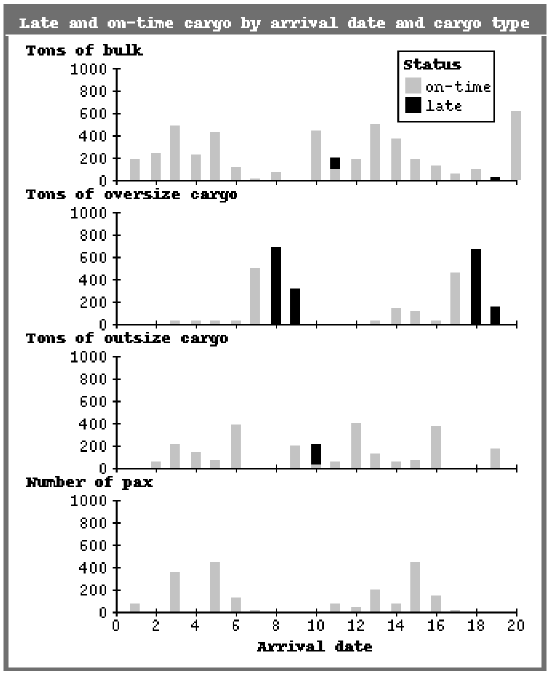


Figure 4. Correlation among tons of late and on-time cargo, cargo type and date.

Sets

The problem we are interested in is concerned with conveying relationships between sets of instances. In order to summarize such sets or to emphasize certain subsets, we often need to aggregate elements of a set. For example, in Fig. 1 all the "totals" are summary attributes for the aggregate of all the move requirements for the schedule.[2] A more complex form of aggregation is shown by the line graph in Fig. 1. The set of requirements for moving cargo have been aggregated cumulatively by date, and a new summary attribute that totals the cargo in each aggregate, *cumulative-required-cargo,* has been defined.

We represent a set by its intension (the condition satisfied by its members) and extension (the members of the set). The intension can be regarded as a query to the KB and the extension consists of the instances satisfying the query. In addition to intension and extension, the set can have summary attributes such as cardinality and total cargo.

We represent sets in this way for the following reasons:

- to explain/refer to a set or subset by its intension;
- to describe a set through its summary attributes;
- to post a goal that applies to all members of a set;
- to make decisions for media allocation based on the cardinality of a set.

Goals, Actions and Tasks

The structure of the goal and action space, part of which is shown in Fig. 5, stratifies into three layers: domain-specific presentation strategies that achieve domain-specific communicative goals; abstract actions that achieve domain- and media-independent communicative goals; and primitive actions that realize media-specific techniques. Primitive actions are those that can act as directives to the media generators. Thus, they are by definition media-specific. Abstract actions satisfy media-independent communicative goals and are realized by media-specific actions.

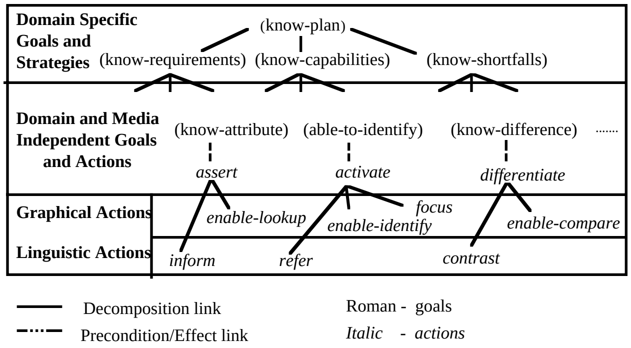


Figure 5. Goals and Actions

Domain-specific goals represent the domain knowledge that the users are expected to have after examining a presentation. The decomposition of these goals into domain-independent goals is accomplished by means of domain-specific strategies negotiated with domain experts. Such strategies define the content of the presentation in terms of concepts, relations between them, and the communicative intent associated with them. Two examples of domain-specific goals and their strategies are given below:

`know-schedule` - to know a schedule the user must know its requirements, capabilities, and shortfalls. In turn, each of these aspects requires knowing attributes of the schedule such as total number of passengers and total tons of cargo that need to be transported, the origin and destination countries, and so on.

`know-lift-shortfall` - to know a lift shortfall, the user must know the attributes daily needed lift capacity and daily available lift capacity, as well as the difference between them; must be able to identify the intervals when the needed capacity exceeds the available capacity; must know the maximum needed capacity in each of these periods, and the additional cargo needed to eliminate the shortfall in each of these periods; must be able to request information about the correlation among late and on-time cargo, cargo type, and date.

[2] A move requirement specifies the origin, destination, and time interval for moving a unit of people or a piece of cargo.

The important elements in these strategy descriptions are the names of the attributes and keywords such as “know,” “difference,” “identify,” and “request,” which convey the communicative intent of the presentation. Formally, these strategies translate the domain-specific goal into goals at the next level of the communicative model.

Domain and media independent goals are communicative goals that are typical in the genre of exploratory data analysis. Each of these goals is achieved by a media-independent abstract action. Some goals and the actions that satisfy them are given below:

`know-attribute` - the user knows the values of an attribute for a set of objects. Satisfied by `assert`.

`know-difference` - the user knows the differences between two attributes. Satisfied by `differentiate`.

`know-correlation` the user knows about the correlation of two or more attributes. Satisfied by `correlate`.

`able-to-identify` - the user can identify each element of a set or one of its subsets. Satisfied by `activate.` (Our model for this type of goal/action was motivated by the work of Andre and Rist who studied mechanisms for identifying objects in multimedia presentations [1].)

`able-to-request` - the user can pose another goal to the system. Satisfied by `enable-request`.

The `know-` type of goals can be annotated with a level of detail (*summary* or *detail*), which determines the amount of information through which the communicative goal needs to be achieved.

The action `enable-request` requires `activate` actions for the objects of the goal, which in turn can be realized in language or graphics. For example, to enable the user to request more information about a particular schedule, the user should be able to identify that schedule (e.g., through a referring expression) and be given a method for requesting the information (e.g., a mouse click on a mouse sensitive phrase associated with that referring expression).

Linguistic and graphical actions realize the media-independent actions using techniques from the corresponding medium. In text, `assert` is usually realized by `inform`, `differentiate` by `contrast,` and `activate` by building a referring expression (for brevity, `refer`).

In graphics, communicative goals are realized in two ways: by enabling the user to perform certain information-seeking tasks, or by focusing the user’s attention on a part of the graphic.

Asserting facts in graphics is realized by enabling the user to perceptually look up or compute the values of an attribute. (As described later, each task can be supported by various graphical techniques, which are selected by the graphics realization system.) The corresponding system actions are `enable-lookup` and `enable-compute`. In general, lookup is a more efficient task than compute. Depending on the specific graphical technique selected to support the corresponding task, the goal `know-attribute` can be achieved with different levels of accuracy [13]. For example, labels ensure very accurate lookup, while saturation is fairly inaccurate.

Activating objects can be realized graphically in two different ways. If each element of a set needs to be identified, then an attribute that uniquely identifies the individual elements is chosen and encoded by a graphical parameter (e.g., the proper name attribute for a set of people). The action corresponding to this method is `enable-identify`. If a subset must be identified as a whole, then its manifestation on the graphic must be highlighted in a certain way (e.g., using a color or a pointer). The corresponding action is `focus`. Examples of focusing can be found in Fig. 3 (the two maxima of the needed capacity are distinguished from the rest by the labels attached to them), in Fig. 4 (the late cargo is distinguished by the color-encoding technique), and in Fig. 6 (the amount of additional cargo needed on dates 5 and 14 are distinguished by the vertical line graphemes and the labels attached to them, 308 and 293).

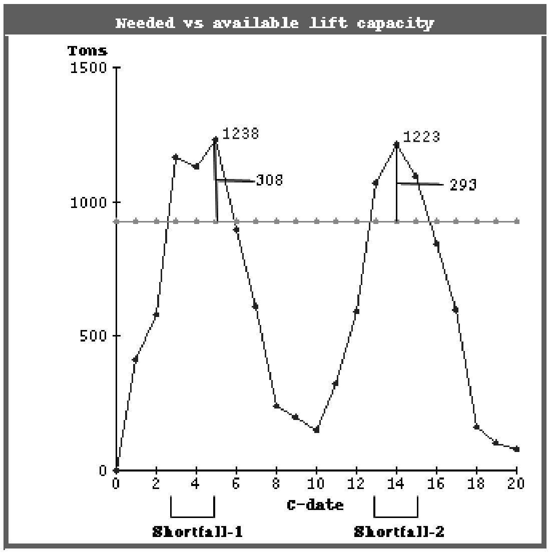


Figure 6. Asserting the needed additional cargo by graphical lookup and identifying the shortfall periods by focusing. The gray line shows the available-lift-capacity, whereas the black line shows the needed-lift-capacity.

Differentiating attributes is realized graphically by selecting a common encoding technique for those attributes. The corresponding action is `enable-compare`.

Similar techniques based on different information-seeking tasks exist for the other media-independent goals.

A DETAILED EXAMPLE

In this section we illustrate our approach by describing the automated design process that results in the presentation in Fig. 3. This presentation fulfills the domain-specific goal *know-lift-shortfall*. A domain-specific strategy decomposes it into the following domain-independent communicative goals:

- *know-attribute* for needed-lift-capacity;
- *know-attribute* for available-lift-capacity;
- *know-difference* between needed-lift-capacity and available-lift-capacity;
- *able-to-identify* the intervals of the lift shortfall (where the needed capacity exceeds the available capacity);
- for each interval of the shortfall, *know-attribute* for the maximum needed lift capacity;
- for each interval of the shortfall, *know-attribute* for the maximum additional lift capacity necessary to eliminate the lift shortfall.
- *able-to-request* for the goal *know-correlation* of tons of late and on-time cargo, cargo type and date.

The actions that can achieve these goals are *assert*, *compare*, *activate*, and *enable-request*. The next level of decomposition realizes these actions through media-specific ones. We will discuss the way these media independent actions are realized in Fig. 3 and also point to alternative methods of achieving the same goals.

The two assert and the *compare* actions are realized graphically in Fig. 3 by two *enable-lookup* and one *enable-compare* primitive actions. Since the two attributes needed-lift-capacity and available-lift-capacity are time series, they were visualized as two line graphs. The common encoding technique for the two attributes is *y-position*, which supports comparison very well. Alternatively, in language, the *assert* for available-lift-capacity could be realized by the sentence “The daily available capacity is 930 tons,” but the realization of the *assert* needed-lift-capacity would be awkward, resulting in the enumeration of 20 values. The differentiation is realized linguistically by the sentence in the first bullet in Fig. 3 "The needed capacity exceeds the available capacity in the date periods 3-5 and 13-15." Note that the graphical and linguistic action realize the same *differentiate* action at different levels of detail.

The identification of the two intervals is accomplished linguistically in Fig. 3 by the referring expressions in the sentence that differentiates the needed and available lift capacity, i.e., “periods 3-5 and 13-15”. Alternatively, it might have been realized graphically (action *focus*) by highlighting the points on the needed lift capacity graph, or by pointers to the two intervals as shown in the bottom part of Fig. 6.

The maximum needed capacity in the two shortfall intervals is asserted in Fig. 3 through the y-positions of the two high points on the line representing needed capacity. However, since high accuracy is needed, two labels were added to these points representing the maximum values, 1238 and 1223. Alternatively, the same *assert* actions could be realized linguistically by two *inform* actions.

The additional capacity needed to eliminate the shortfall is not directly encoded in Fig. 3. However, it can be evaluated by perceptually computing the difference between pairs of points on the two line graphs. Since this is a very inaccurate way to achieve the *know-attribute* goals, they were realized linguistically in the second bullet of Fig. 3 by the sentence "Additional 308 tons for the interval 3-5 and 293 tons for the interval 13-15 are needed to eliminate the shortfall." A possible way to achieve these goals by accurate *enable-lookup* actions is shown in Fig. 6. Two vertical line symbols have been added that represent the maximum differences between needed daily and available daily capacity for the two intervals, and labels have been added for accurate lookup of the values.

Finally, the *enable-request* action for the complex correlation has been realized through the summary statement in the third bullet (Fig. 3) and by appending the mouse-sensitive phrase “details” to the end.

MEDIA ALLOCATION

The above discussion shows how the same media-independent communicative goals could be achieved in both graphics and language. Media are allocated by rules, which consider the effectiveness of the media presentation techniques as a function of parameters such as level of detail, cardinality of the sets involved, and accuracy. For example, detailed *assert* and *compare* about large sets are less efficient in language than in graphics because the language expressions are sequential, whereas graphics are organized spatially. By compressing and indexing the information positionally and retinally, graphics provide compact, rapidly searchable presentations. Compare how effectively Fig. 3 realizes the comparison of the two line graphs, and imagine how cumbersome it would be to realize this comparison using text.

In contrast, language is superior to graphics when the communicative function must be explicitly conveyed to the user or when the abstract action involves sets with low cardinality. For instance, in Fig. 3 the linguistic statement in the first bullet explicitly and clearly conveys to the user the main point the presentation is intended to communicate (i.e. needed-lift-capacity exceeds available-lift-capacity in two intervals). Contrast this with a possible different communicative intent for the same chart that would be expressed by the statement: "while available-lift-capacity is constant, needed-lift-capacity varies considerably."

Table 1 shows some media allocation rules based on the cardinality of the set representing the objects of the attribute (the rows of the table) and the cardinality of the set representing the attribute values (the columns of the table). The table shows that our preference for presenting a large number of values is graphics, whereas our preference for presenting a small number of values is text.

Table 1. Media allocation for detailed assert (G - graphics only, T/G - prefer text but allow graphics).

	1	2-3	>3
1	T/G	T/G	G
2-3	T/G	T/G	G
>3	T/G	G	G

However, media allocation is also influenced by the interaction between goals. For instance, a goal may be efficiently expressed in graphics only if another goal is also expressed graphically and the two goals can share certain encoding techniques. In our example, the realization of the `assert` action for the maximum needed capacity in the two intervals was expressed efficiently in graphics because of the graphical realization of the `assert` action for the daily needed capacity. An explanation of this phenomenon is that interpreting graphics has the overhead of understanding the structure of the graphic and the encodings used. Once the user is situated in a graphic, assimilating additional information is easier.

GRAPHICS REALIZATION

For graphics realization we use SAGE. It incorporates design rules that apply encoding and composition techniques based on characteristics of the information to be presented [13]. In addition to automatic design, SAGE provides flexible tools for interactive design [15]. For the purpose of the current project, we have developed a new tool that designs graphics based on tasks that the users should be able to perform. This tool implemented in FUF [5] (the same formalism in which we are implementing the NL generator) performs a grammar-driven search of encoding and composition techniques. The result is a set of design directives, which are subsequently reconciled with the basic design rules of SAGE. In addition to generating the graphic, SAGE returns additional effects that this particular design achieves as well as any complexity involved with the interpretation of the graphic. The former is used for media coordination and follow-up questions while the latter spawns caption generation [11].

PREVIOUS WORK

Several projects have studied the problem of automatic generation of multimedia presentations. The COMET [7] and WIP [18] systems generate instructions for operating physical devices. Both systems use media allocation rules based on the distinction between concrete vs. abstract information. An attribute of an object is concrete if it can be perceived visually; analogously, an action is concrete if it causes visually perceptible changes. Generally, the graphical/pictorial medium is favored in the case of concrete information, whereas abstract information is preferably expressed in text. Although relevant for pictorial instructions, such rules are irrelevant in our genre, where the selection is between text and information graphics (e.g., charts, networks, and maps).

Arens and Hovy suggested some general principles and an abstract model of multimedia presentation planning called CICERO [2]. According to their principles, effective multimedia presentation requires models of the following elements: the media; the information to be displayed; the application task and interlocutors goals; the discourse and communicative context; and users' goals, interests, and abilities. Like the work of others [7,10,18], these principles ignore the task level, which we find important. Moreover, the application of these principles is not straightforward and requires additional research to specify them at a level where they can be used in real systems.

The PostGraphe system of Fasciano and Lapalme [6] generates multimedia statistical reports consisting of graphics and text. While their approach seems similar to ours, there are some significant differences. First, PostGraphe assumes that the set of communicative goals (what they call intentions) is given. Second, it is not clear whether the set of communicative goals they propose is intended to be media-independent or primarily graphical. In fact, the system goes directly from a set of intentions to graphics and text is only added subsequently. Finally, PostGraphe's schema-based mapping from intentions to graphic design is less flexible than the one allowed by our task-based graphic design model.

Previous work has shown how graphics design decisions influence the efficiency of different tasks that users can perform on graphical displays and how graphics should be designed to support certain types of tasks. For example, Casner's [4] system takes a procedural description of complex tasks and designs graphics that support them. Besher and Feiner's [3] system designs interactive techniques for 3D graphics that users can employ to accomplish certain tasks. Roth and Mattis [13] system, SAGE, incorporated information-seeking goals as one of the factors that influences graphics design. However, these systems assume that the set of tasks is given. Our approach extends prior work by showing how the tasks can be derived from higher-level communicative goals.

CONCLUSION

In this paper we proposed a framework for integrating decompositional planning and task-based graphic design. Our main emphasis was on defining a communicative model of multimedia presentations that accounts for content selection, realization of general communicative goals through media-specific techniques, and enabling users to request additional information. We also considered some media allocation rules that take into account the level of detail, the cardinality of the sets involved, and the interaction between goals. The model was partially implemented using existing AI systems: Longbow [19], FUF [5] and SAGE [12]. The system designs and generates multimedia briefings in the transportation scheduling domain.

In the short term, we plan to develop a media coordination mechanism and to devise techniques for generating subsequent presentations that take into account the context created by prior presentations [12]. In the longer term we plan to incorporate the system into Visage, an information-centric data exploration environment [16].

Acknowledgments: This project was supported by DARPA, contract DAA-1593K0005. We are grateful to Mark Derthick and Vibhu Mittal for the numerous discussions and the valuable comments they made on the draft of this paper. In addition, we are grateful for the comments of the anonymous reviewers.

REFERENCES

1. Andre, E., and Rist, T. Referring to World Objects with Text and Pictures. In *Proc. COLING-94*, 1994, 530-534.
2. Arens Y., and Hovy, E. The Design of a Model-Based Multimedia Interaction Manager. *AI Review, 8*(3), in print.
3. Beshers C., and Feiner, S. AutoVisual: Rule-Based Design of Interactive Multivariate Visualizations. *IEEE Computer Graphics and Applications*, July 1993, 41-49.
4. Casner, S.M. A Task-Analytic Approach to the Automated Design of Information Graphic Presentations. *ACM Transactions on Graphics, 10,* 2 (Apr. 1991), 111-151.
5. Elhadad, M. *Using Argumentation to Control Lexical Choice: A Functional Unification Implementation*. Ph.D. Dissertation, Columbia University, 1992.
6. Fasciano, M., and Lapalme, G. PostGraphe: a System for the Generation of Statistical Graphics and Text. In *Proc. 8th Int. Natural Language Generation Workshop*, June, 1996, Sussex, UK, 51-60.
7. Feiner, S., and McKeown, K. R. Automating the Generation of Coordinated Multimedia Explanations. *IEEE Computer, 24,* 10 (Oct. 1991), 33-40.
8. ISX Corporation. *Loom User's Guide*, 1991.
9. Kerpedjiev, S. Model-driven Assertion-based Generation of Multimedia Weather Information. *Bull. of the American Meteorological Society, 76,* 10 (Oct. 1995), 1791-1800.
10. Maybury, M. T. Planning Multimedia Explanations Using Communicative Acts. In *Proc. AAAI-91*, July 1991, 61-66.
11. Mittal, V. O., Roth, S. F., Moore, J. D., Mattis, J., and Carenini, G. Generating Explanatory Captions for Information Graphics. In *Proc. IJCAI'95*, Montreal, Canada, Aug. 1995, 1276-1283.
12. Moore J. D. *Participating in Explanatory Dialogues.* MIT Press, 1995.
13. Roth, S. F., and Mattis J. Data Characterization for Intelligent Graphics Presentation. *Proc. SIGCHI'90*, Seattle, WA, ACM, April, 1990, pp. 193-200.
14. Roth, S. F., Mattis, J., and Mesnard, X. Graphics and Natural Language Generation as Components of Automatic Explanation. In Sullivan and Tyler (Ed.), *Intelligent User Interfaces*, Addison-Wesley, Reading, MA, 1991, 207-239.
15. Roth, S. F., Kolojejchick, J., Mattis, J., and Goldstein, J. Interactive Graphic Design Using Automatic Presentation Knowledge, In *Proc. CHI'94*, Boston, MA, 1994, 24-28.
16. Roth, S. F., et. al. Visage: A User Interface Environment for Exploring Information. In *Proc. IEEE InfoVis'96*, San Francisco, CA, Oct. 1996.
17. Smith, S. F., Lassila, O., and Becker, M. Configurable, Mixed-Initiative Systems for Planning and Scheduling, In *Advanced Planning Technology*, (ed. A. Tate), AAAI Press, Menlo Park, CA, 1996.
18. Wahlster, W., Andre, E., Finkler, W., Profitlich, H.-J., and Rist, T. Plan-based Integration of Natural Language and Graphics Generation. *Artificial Intelligence, 63* (1993), 387-427.
19. Young, R. M., Pollack, M. E., and Moore, J. D. Decomposition and Causality in Partial Order Planning, In *Proc. 2nd Intern. Conf. on Artificial Intelligence and Planning Systems*, Chicago, IL, 1994.