

To appear in: Proceedings of the Fourth Bellcore/BCC Symposium on User-Centered Design, November 3-5, 1993, Piscataway, NJ

User-Needs Analysis and Design Methodology for an Automated Documentation Generator

*K. Kukich
N. Morgan and J. Phillips
Bellcore
445 South Street
Morristown, NJ 07960-6438*

*K. McKeown
J. Robin, J. Shaw and J. Lim
Computer Science Department
Columbia University
New York, NY 10027*

ABSTRACT

Telephone network planning engineers routinely study feeder routes within the telephone network in order to create and refine network capacity expansion (relief) plans. In doing so they make use of a powerful software tool called LEIS-PLAN™. We are developing an extension to PLAN called PLANDoc that will automatically generate natural language narratives documenting the engineers' use of PLAN. In this paper, we present the user-needs analysis and design methodology we have used in developing the PLANDoc system. We describe our interviews with various end users to determine if such a system would be desirable and what design factors would make it useful. We show how we model the system on a set of iteratively-revised human-generated narratives. The model narratives determine the function and architecture of the documentation system, and they inform the development of the system components.

1. Introduction

User-needs analysis has become a common practice in the development of computer-human interface systems and other end-user software.^{4,6} However, AI systems are frequently developed without an analysis of the end user's needs. Particularly in developing a large scale, practical AI system, the needs of the user should be studied if the resulting system is to be effectively used. In this paper, we report on a user-needs analysis and system development methodology that we are using in our ongoing development of an automated documentation system, PLANDoc, for telephone network planning operations. We are systematically modeling PLANDoc after a target set of manually-written narratives that have undergone successive revisions based on user critiques. We show how user input has influenced both the form and content of PLANDoc narratives and the design of the system itself.

The telephone network planning engineer's job is to derive a capacity expansion (relief) plan specifying when, where, and how much new copper, fiber, multiplexing and other equipment to install in the local network to avoid facilities exhaustion. Planning engineers have the benefit of a powerful software tool, the Bellcore LEIS-PLAN™ system, that helps them derive a 20-year relief plan that optimizes the timing, placement and cost of new facilities for a route in the network. Until now they have not had the benefit of a tool to help them with the equally important but often tedious task of documenting their planning decisions, so in many busy shops this crucial writing task has been upstaged by more pressing planning tasks. Documentation is needed primarily to provide a record of the planner's activities and reasoning to be used for future network studies, for informing managers who are responsible for authorizing project plans, and for justifying expenditures to internal auditors and external regulators.

Our work enhances PLAN by providing a natural language text generation system and a simple facility that prompts engineers to add their own manually-written comments at crucial points. Based on the

results of our user-needs analysis, PLANDoc is designed to combine human-generated and machine-generated text to produce natural language narratives. In the following sections we briefly describe the planning engineer's use of the PLAN system, our user-needs analysis and the resulting PLANDoc narrative format, the PLANDoc system architecture and the development of the text generator based on target narratives. We close with a discussion of future research challenges related to content planning for PLANDoc.

2. LEIS-PLAN™ Background

Voice and data service is carried to telephone customers through a complex network of routes consisting of copper or fiber cables supplemented by additional equipment such as Digital Loop Carrier (DLC) systems and fiber multiplexors. In order to be able to provide new services to customers on demand, network facilities must already be in place. It is the planning engineer's job to determine the optimum configuration of equipment needed to avoid facility exhaustion while minimizing costs.

The engineer uses PLAN to download a model of the route from a headquarters database, to enter forecasts of anticipated service demands, and then to have PLAN compute an optimum, cost-effective base relief plan. The base plan specifies the timing, placement and costs of new facilities needed to meet forecast demand over the next twenty years, with particular emphasis on the next five years.

Although PLAN employs sophisticated optimization algorithms to arrive at the most economical long-term base plan, the base plan may not always be realizable or desirable. It frequently needs to be refined to account for political, economical, practical and other factors known to the engineer but unknown to the computer. For example, the planning engineer may need to make changes to accommodate a management policy such as a mandate to achieve an all-fiber route by a certain date; she may need to work within the constraint of a fixed cap on initial installation cost; or she may be aware of practicalities such as the existence of small manholes that make equipment installation difficult in certain locations.

For this reason the PLAN system includes a powerful Interactive Refinement Module that allows the engineer to do 'what-if' modeling to explore the effects of various changes to the base plan. Some of the refinement actions an engineer might explore include requesting copper or fiber cable placements, requesting a DLC activation for a given site, changing a fiber activation time, or requesting a fiber service extension, among others. After comparing the effects of different refinement scenarios in terms of costs, timing and placements of equipment, the engineer ultimately decides on a realizable relief plan to recommend to management for project authorization.

PLAN keeps a somewhat cryptic trace of all the refinements the engineer tries in a refinement Tracking Report. Figure 1 shows a portion of a Tracking Report for one refinement together with its corresponding refinement paragraph generated by PLANDoc and its manually-entered engineering note.

3. User-Needs Analysis and Model PLANDoc Narratives

With the help of Bellcore Planning and Engineering staff* we formulated an initial proposal for PLANDoc and drafted preliminary target narratives. We then conducted a series of interviews with planning engineers and managers and PLAN support staff from several regional telephone companies in their home offices and at two PLAN training courses.† The work experience of the engineers we interviewed ranged from beginner to expert. Our goal was to determine 1) how engineers actually used the PLAN system, 2) whether they would find an automated documentation facility to be helpful, and, if so, 3) what the form and content of the narratives should be, and 4) how the documentation system should function. Finally, we compiled a corpus of target narratives written by an experienced planning engineer, incorporating feedback from the interviews.

Some of the things we learned about how engineers use PLAN include: a) novice planners often run 'bozo' refinements just to develop a feel for the effects of making certain changes in the route; b) experienced planners sometimes run refinements they know will be suboptimal just for the record, i.e., for the

* Many thanks to M. Horwath, D. Imhoff and L. Tener.

† Some of the helpful regional Planning and Engineering personnel included P. McNeill, J. Bruner, P. King, D. Kelly, I. McNeill, T. Smith, C. Lowe, and G. Giles, all from Pacific Bell, R. Riggs, D. Spiegel, S. Sweat, L. Doane, R. Tufts, and R. Ott, all from Southwestern Bell, S. Wasalinko from NYNEX, and C. Lazette from Ameritech.

PLANDoc Tracking Report

RUNID reg1 for DLC PLAN Changes [Activ]
CSA 3122 used ALL-DLC system idlc272
and DSS-DLC system idlc272
CSA 3122 for ALL-DLC activated 3 quarter 1994
for DSS-DLC activated 3 quarter 1994
CSA 3130 for ALL-DLC activated 3 quarter 1994
for DSS-DLC activated 3 quarter 1994
CSA 3134 for ALL-DLC activated 3 quarter 1994
for DSS-DLC activated 3 quarter 1994
CSA 3208 for ALL-DLC activated 3 quarter 1994
for DSS-DLC activated 3 quarter 1994
CSA 3420 for ALL-DLC activated 3 quarter 1994
for DSS-DLC activated 3 quarter 1994
Total 20 Year PWE (\$K) For The Route: \$2110
First 5 Year IFC (\$K) For The Route: \$1064

PLANDoc Refinement Paragraph with Engineering Note

RUNID Reg1: This refinement activated
CSA's 3122, 3130, 3134, 3208 and 3420
for DLC in the third quarter of 1994.
DLC system idlc272 was used for all
placements in CSA 3122. [The route
default system is 'idlc96'.] For this
refinement, the resulting 20-year route
PWE was \$2110K, a \$198K savings over
the base plan, and the 5-year IFC was \$1064K,
a \$65K penalty over the base plan.

ENGR NOTE: These CSA's are beyond 28 kf
and need range extenders to provide service
on copper. Moving them to 1994 will negate
a job adding a reg bay to the office.

Figure 1

benefit of managers, auditors and regulators who might ask "did you try such and such?". More critical to the need for documentation, we also learned that: c) experienced planners keep handwritten notes on paper listing their refinements and why they tried them; they asked for a way to enter their notes on-line to keep track of their reasoning; d) inexperienced planners tend to rely on the base plan because they lacked the expertise to refine it; they asked to see narratives written by experienced planners in order to learn from them; (unfortunately few such narratives exist); finally, e) all planners welcomed the idea of having the computer generate narratives that they could include in their documentation packages, especially if they could add to the narratives themselves.

These findings shaped the content of PLANDoc narratives and the design of the system. Specifically, they indicated that planners may not want all refinements that they tried to appear in the narrative. For example, novice planners do not want to include their 'bozo' refinements, while experienced planners do want to include the suboptimal refinements they ran to show that their final refinements were superior. Thus, PLANDoc includes a facility that lets the planner select a subset of refinements to be included in the final narrative. Planners made it clear that they use knowledge not included in PLAN to make their decisions (e.g., knowledge of other installation projects, corporate strategies, physical environment, etc.) and they wanted a way to record that knowledge on-line, while they were working. This gave rise to PLANDoc's facility to prompt for manually-written engineer's notes at crucial points. Throughout the PLANDoc design process, we took care to minimize and standardize changes to PLAN's original, successful interface. Thus, PLANDoc requires only two modifications to PLAN's interface, one to prompt for engineering notes and another to allow the engineer to request a narrative and select a subset of refinements to be included. Both options are presented using familiar PLAN interface commands and screen formats.

In addition to our interviews, we also arranged for an experienced retired planning engineer, Jim Phillips, who is also a PLAN expert, to write a corpus of target narratives based on PLAN runs of actual routes. We have been using this corpus to guide the development of the PLANDoc text generator. Based on the findings from our interviews and on successive revisions and critiques of target narratives, we arrived at the following general model for PLANDoc narratives that integrates machine-generated text with the engineer's manually-written notes.

Model PLANDoc Narrative Format

- PART 1: Tabular Route Input Data Summary
- PART 2: Base Plan Summary Paragraph

- PART 3: Alternating Paragraphs of Refinement
Descriptions and Engineer's Notes
- PART 4: Relief Plan Summary Paragraph

4. The PLANDoc Automatic Text Generator

PLANDoc's architecture, which is shown in Figure 2, draws on our previous report generation and text generation work.^{3,5}

PLANDoc System Architecture

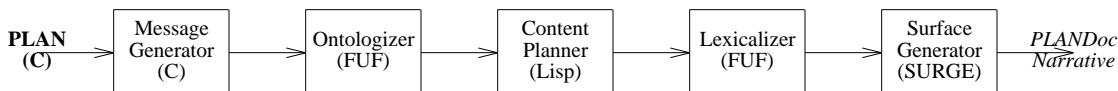


Figure 2

Since most of PLANDoc is written in LISP, a message generator * produces a LISP representation for each refinement action in a PLAN tracking report. This set of messages is first passed to an ‘ontologizer’ † that enhances each message with semantic knowledge from PLAN’s domain. The messages are then passed to a content planner, † whose job it is to determine which information in the messages should appear in the various summaries and to organize the overall narrative. For our initial prototype, we have focused only on the generation of paragraphs that describe refinement actions (Model Narrative, PART 3), and not on the generation of the summary paragraphs. Thus, currently the content planner’s task is limited to combining individual messages to produce the input for complex sentences.

We are using an existing surface generation system, the FUF/SURGE package,^{1,2} as a tool in the text generator. It consists of a unifier (FUF) and large grammar of English (SURGE) that enables it to generate a large variety of sentences. In order to generate the sentences required to “translate” the refinement actions, we built a lexical interface‡ which maps the tokens in the messages to the input required by FUF/SURGE. Since FUF/SURGE accepts a case role structure as input, with words already chosen, the lexical interface must determine the semantic roles and words in the sentence for each token in the input message.

Our user needs analysis directly influenced our system development. For example, an analysis of PLAN’s menu of refinement actions and the model narratives allowed us to specify the different possible message types. Furthermore, a systematic categorization of the sentences in our model narratives revealed all the different phrasings for each message type. This categorization showed that there is great variety in the possible sentences for each message type. For example, all message types that specify a type of refinement that the planning engineer carried out could be realized as “This refinement demanded <action np>”, as in “This refinement demanded DLC activation for CSA 2119 in 1994 Q1.” Alternatively, these sentences could be realized as “<Site X> <passive action verb>.” as in “CSA 2119 was activated for DLC in 1994 Q1.” These are but two of many possibilities. The model narratives indicate that which of these alternative phrasings is used depends on whether any refinements for this site have already been mentioned in the summary (i.e., the choice is based on previous discourse). If none had been mentioned, the summary used “This refinement demanded ...” and if the site was just discussed, the summary used the verb for the action in the passive form.

Thus, the use of the model narratives for system development has provided us with the data for lexi-con development, both in terms of vocabulary and sentence structure used. In addition, it provides us with

* written by N. Morgan

† both written by J. Robin

‡ written by J. Shaw, J. Lim and D. Horowitz

the constraints on alternative realizations. By examining the narratives, we can find the situations in which different forms and words were used. Currently, the implemented text generator can produce sentences for 24 out of a total of 32 different message types. It can produce an average of 150 different sentences for a single message and the lexicon contains 145 open class words in addition to equipment database terms and the closed class words which are selected by the grammar. It is able to produce complex sentences which combine several messages as shown in the output of Figure 1. The first sentence in this text combines 5 activations of DLC using conjunction.

5. Conclusion and Current Directions

The user-needs analysis we described here has influenced many aspects of system development. The model narrative format has determined the functions of the PLANDoc system by specifying the types of documentation required. Interviews with the planning engineers has influenced the design of PLANDoc's interface, resulting in facilities that allow engineers to select which refinements of the ones they tried should be included in the report and to add in their own notes which will be integrated with the automatically generated text. Finally, the manually-written target narratives have guided the development of the PLANDoc content-planner and lexicon. They have provided us with the data which allowed us to systematically identify the different sentences forms and words that are needed to describe each refinement action as well as to identify constraints that determine when different sentence types should be used.

Although the design of a content planner and a lexical grammar for managing the constraints of discourse structure lexical choice has posed many interesting research challenges, other difficult challenges remain to be addressed in the design of an even more intelligent narrative generator. These challenges are primarily related to content planning. One challenge is to enrich refinement paragraphs with relevant information drawn from PLAN's input data, base plan and previous refinements. An example of this is shown in the bracketed sentence included in Figure 1. Another challenge is to generate summary paragraphs for the base plan and relief plan. This challenge requires developing models for determining which few of the many facts included in the data are relevant to the readers' needs (where readers include engineers, managers, auditors and regulators.) As with the design of the current content planning and lexicalization modules, we intend to exploit the corpus of target narratives, supplemented by interviews with expert planners, to derive the models needed for generating more intelligent refinement paragraphs and summary paragraphs.

Acknowledgements

We are grateful for the contributions of Lynn Streeter, John Kaminski and Michael Elhadad to the PLANDoc project, and to Mary Leland for making the connection.

References

1. M. Elhadad, "FUF: The universal unifier - user manual, version 5.0," CUCS-038-91, Columbia University, 1991.
2. M. Elhadad, "Using argumentation to control lexical choice: a unification-based approach," Ph.D. Thesis, Computer Science Department, Columbia University, 1992.
3. Karen Kukich, "Design and Implementation of a Knowledge-Based Report Generator," in *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pp. 145-150, MIT, Cambridge, Massachusetts, 15-17 June 1983.
4. Thomas Landauer, Dennis Egan, Joel Remde, Michael Lesk, Carol Lochbaum, and Daniel Ketchum, "Enhancing the Usability of Text Through Computer Delivery and Formative Evaluation: The SuperBook Project," in *Hypertext: A Psychological Perspective*, ed. A. Dillon and C. McKnight, 1993.
5. K. R. McKeown, *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*, Cambridge University Press, Cambridge, England, 1985.

6. Jakob Nielsen, *Usability Engineering*, Academic Press, San Diego, CA, 1993.