Multilingual generation of administrative forms

Richard Power and Nico Cavallotto

ITRI, University of Brighton
Lewes Road
Brighton BN2 4AT UK
email: rjdp@itri.brighton.ac.uk

Abstract

We will demonstrate the GIST system, which generates social security forms in English, Italian and German. The system is intended for use by the technical authors and translators who design forms. A knowledge specification tool allows the author to build a model of the form in the knowledge representation language LOOM. From the LOOM model, a text drafter generates equivalent texts in the three supported languages, guided by some broad stylistic parameters which the author can control. The output texts serve as drafts which the authors and translators can modify or extend.

Keywords: multilingual generation, applications.

Type of submission: demonstration.

1 Background

The GIST system[1] produces drafts of social security forms in English, Italian, and German. It allows technical authors to model the content of a form by means of a knowledge specification tool; from this model, the system automatically generates draft texts.

Support for producing multilingual documentation has a twofold significance in Europe. First, the European Community (EC) has posed the long-term objective of producing official documentation in all the main languages of the community, so that workers migrating within the EC will be able to read essential documents, such as employment or pension forms, in their own languages; at present, this objective is realized only to a very limited degree, owing to translation costs. Secondly, many countries in Europe have multiple languages: GIST focusses on the Trentino Alto-Adige region of Northern Italy, in which all official documentation has to be produced in two languages, Italian and German, laid out side by side on the page. The GIST consortium includes two organizations that have to implement this requirement: the Italian social security institute (INPS), and the local government agency for the Bolzano province (PAB).

2 Requirements

To draw up requirements for the GIST system, we visited offices in Italy and Britain where social security forms are designed and translated. We are particularly grateful for the collaboration of the Document Design Unit (DDU) of the British Department of Social Security. From these meetings we drew three main conclusions.

1. *The specification tool should present the model of a form in a way that technical authors can easily understand.* The content of a form is modelled in the knowledge representation language LOOM [4]. Technical authors are not knowledge engineers: they cannot be expected to master quickly the concepts or syntax of a language like LOOM. An accessible interface between the author and the LOOM model is therefore essential. Moreover, when drafting a form, the author often refers to previous versions of the same form, or to other forms with overlapping content; thus it is important that a model defined by one author should easily be understood by another author, or by the same author several months later.

2. *In designing the text drafter, close attention should be paid to the stylistic preferences of authors.* Apart from their general training in
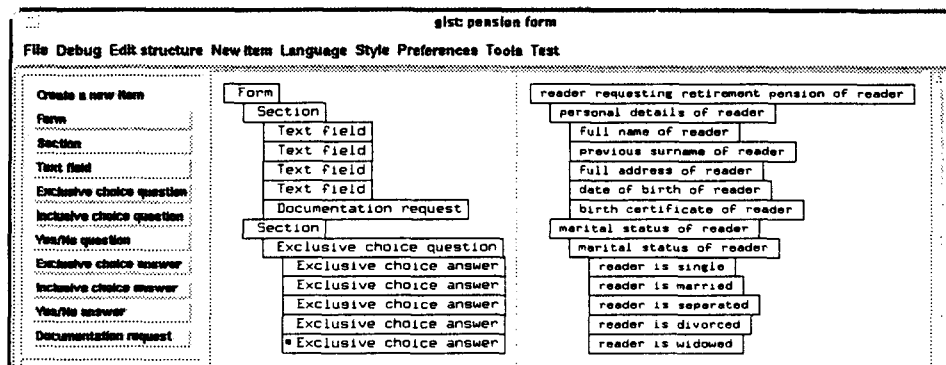
Figure 1: Modelling a pension form

writing and languages, the authors draw upon a great deal of expertise which has evolved in the department where they work. The DDU has for more than a decade employed independent market researchers to test its forms with typical users. Some results of these studies have been distilled in a written guide [3]; others are passed on by word of mouth or by imitation. Details of these stylistic requirements are given in [6].

3. *The system should be able to vary the style of the output texts to suit different languages or organizations.* Each organization that we studied had a clearly marked style which was applied consistently throughout its documents. The DDU forms were informal and concise; instructions and background information were kept to a minimum and integrated with the questions. By contrast the INPS forms were more formal; they also relied more on explicit instructions and other background notes, which were collected together on a separate sheet. To cover these variations, the GIST system includes a panel which allows the user to make some broad stylistic choices (e.g. formal vs informal; integrated instructions vs separate instructions).

3 Architecture

When specifying the content of the form, the author indirectly edits a knowledge base in the language LOOM. During generation, a text structurer consults the LOOM model in order to build a text plan [2] comprising a hierarchy of communicative goals. Microplanning rules are applied to this plan in order to obtain plans for individual sentences, expressed in extended SPL (Sentence Planning Language) [7].

Finally, tactical generators for English, Italian and German compute natural language texts from the SPL representations [1]. At each stage of planning, decisions may be influenced by the stylistic parameters, and plans for the three languages may diverge in accordance with cultural as well as linguistic variations.

4 Demonstration

Figure 1 shows part of the GIST main window during the definition of a simple pension form. Apart from the menu bar the window has three areas: the *button panel* on the left, followed by the *outline area* and the *content area*. By clicking on the buttons, the author can create various types of form part, including sections, text fields, and multiple choice questions; the whole form is also considered to be a form part. Each form part is presented on a single line of the model: its type is shown by a label in the outline area (e.g. Section); its content is shown by a sentence in the content area (e.g. personal details of reader). Hierarchical relationships among form parts are shown by indenting: thus the form is composed of two sections; the first section is composed of four text fields and a documentation request; and the second section is composed of a multiple choice question with five options.

Each form part is characterized by a set of *attributes*. The most important attribute, the Content, is shown in the main window; the other attributes can be viewed by double-clicking the relevant line of the model, and include the following:

• *Applicability condition:* A condition which determines whether a question or section applies to the form-filler – e.g. the question about the

18

reader's previous surname only applies to married women.

- *Information status:* An indication of whether the requested information is obligatory or optional. An applicable question may be optional if the requested information is inaccessible or sensitive.

- *Information source:* An indication of where to find the requested information.

All attributes are presented in a controlled natural language resembling English note form; Italian and German versions of this language are also supported. Although sometimes clumsy, sentences in this language are easily understood. To specify an attribute value, the user must create a sentence in the controlled language. Most systems using controlled languages allow users to enter sentences in free text (e.g. [5]); for our purposes, however, free text input is unsatisfactory because users would need training in the controlled language and might still make errors. We have therefore preferred an input mechanism in which sentences are built through a series of menu-guided choices.

As an illustration, we will consider the Content attribute for the form, *reader requesting retirement pension of reader*. Initially, this attribute is set to the pattern *[form title]*, the square brackets indicating an element to be expanded; in the interface, such elements are implemented as buttons. By clicking on the button, the user obtains a list of more specific patterns, including *[person] requesting [benefit]*. If selected, this becomes the current pattern in place of *[form title]*. Next, the user can click either on *[person]* or on *[benefit]* to expand the pattern further; this process continues until all expandable elements have been eliminated.

When the model is complete, the panel of style settings can be edited through the Style menu, and the output languages chosen through the Language menu; after these preliminaries, another option in the Language menu can be selected in order to generate draft texts. The drafts are displayed in text editing windows, one for each language, from which they can be saved as text files. From the model in figure 1 the system will generate the text shown in figure 2 along with equivalent versions in Italian and German.

References

[1] GIST consortium, 'Adaptation and extension of

Figure 2: Text of part of a form

the tactical generators', Technical Report LRE Project 062-09 Deliverable PR-2b, (1994).

[2] Erica Giorda, Elena Not, and Emanuele Pianta, 'Implementation of the text structurer', Technical Report LRE Project 062-09 Deliverable TSP-2b, IRST, (1995).

[3] The good forms guide. Department of Health and Social Security, 1983.

[4] Robert MacGregor and Raymond Bates, 'The LOOM knowledge representation language', in *Proceedings of the Knowledge-Based Systems Workshop*, St. Louis, April 21-23, (1987).

[5] Stephen Pulman, 'Controlled language for knowledge representation', in *Proceedings of the first international workshop on controlled language applications*, Katholieke Universiteit Leuven, Belgium, (1996).

[6] Donia Scott and Richard Power (eds), 'Characteristics of Administrative Forms in English, German and Italian', Technical Report LRE Project 062-09 Deliverable EV-1, (1994).

[7] Keith Vander Linden, 'Specification of the extended sentence planning language', Technical Report LRE Project 062-09 Deliverable TST-0, ITRI, (1994).