Appears in *Methods Inf Med*; Vol. 40 (3), pp. 253-258; 2001

# *Retrieval-oriented Design of Clinical Research Forms*

W. Gall[1], E. Eigenbauer[1], S. Rasoul-Rockenschaub[2]

[1]Department of Medical Computer Sciences
[2] Department of Surgery
University of Vienna

Address:

Walter Gall
Department of Medical Computer Sciences
University of Vienna

General Hospital
Spitalgasse 23
A-1090 Vienna
Austria
Tel.: +43-1-40400 / 6694 (6698)
Fax.: +43-1-40400 / 6697
E-mail : walter.gall@akh-wien.ac.at

# Retrieval-oriented Design of Clinical Research Forms

W. Gall[1], E. Eigenbauer[1], S. Rasoul-Rockenschaub[2]

[1]Department of Medical Computer Sciences
[2] Department of Surgery
University of Vienna

## Abstract

Computerized clinical forms are subject to a wide variety of different requirements. They have to allow detailed documentation and must be user-friendly. State-of-the-art applications for design permit even clinicians themselves to create their own forms as needed, with the various variables presented in different ways depending on their intended use. But often enough only aspects of clinical documentation are considered, with no thought being given to subsequent data retrieval. This article presents guidelines for the retrieval-oriented design of clinical forms. It discusses where anticipatory measures for structuring forms are easier to accomplish than complex data linkage at the time of retrieval and analysis.

## Keywords

Interface design, clinical research forms, form production, retrievability

# 1 Introduction

The importance of suitable user interfaces for clinical research is undisputed. General guidelines for human-computer interfaces have been developed [1] [2]. Many studies of user interfaces in the clinical field limit themselves to making interfaces user-friendly and to facilitate documentation. However, one thing frequently neglected is the functionality of a system in clinical practice [3].

Some clinical information systems include a wealth of detail but are nevertheless unsuitable for yielding results for simple clinical or cost-related inquiries. While detailed data can often be retrieved, simple aggregate queries such as those needed for annual summary statistics (e.g., “What organs were operated on?”) often require extensive preparations. In case of extensive and detailed documentation the risk of observer-related variance is increased and therefore enumeration errors can occur [4].

The focus of this article is on design guidelines related to the *retrievability* of clinical research forms. Forms used in clinical routine differ from research forms in some ways. Routine data management must compliance with legal regulations, the forms are designed for a more report-oriented way of documentation, etc. Clinical research forms can be designed more individually. On many clinical research forms, however, variables documented in patient records are used together with additionally collected research data.

To support clinicians in entering their research data themselves, various kinds of collecting structured data have been developed (e.g., knowledge based trees) [5]. However, interface demands for the entry of detailed data during clinical routine, differ from those for collecting research data. An important goal is, to support free text entry (to give clinicians the most freedom) in combination with structured data entry (to support analyses) [6].

Interface demands depend on the clinical situation. Various clinical settings require very divergent forms of documentation and representation, thus mandating a domain-specific approach to interface design [7]. As an example, if the objective is to facilitate rapid data entry amidst the mounting pressures of everyday hospital chores this might be achieved by using a single input medium such as a pen [8]. Successful administration of the extensive variable complexes on an intensive care ward has to meet criteria that differ from those that apply to an interface for an oncological ward for example [9].

There are of course many other criteria besides retrievability that need to be considered when designing an interface, such as usability and familiarity. Retrievability requirements must never be satisfied at the cost of neglecting other requirements. The many options offered by state-of-the-art design tools must be utilized in such a way that the various guidelines complement each other. And, retrieval-oriented design should also improve the quality of the acquired data.

Trying to include all modern interface options - such as clickable images for documenting tumors - is beyond the scope of this presentation. This article will focus on general guidelines for standard types of items such as *Text*, *Numeric* and *Time* along with the usual input widgets for graphic user interfaces such as *selection lists, radio buttons* and similar.

# 2 Methods

## 2.1 Designing Clinical Research Forms

An clinical *form* is an assortment of variables that can be arranged in various different ways. When a documentation system is first converted from paper-based to computerized format, users will accept the latter most readily if it resembles the paper forms they are already familiar with.

Each *item* in the database can be presented by means of several different widgets on a form as a form variable. In case of direct mapping of variables from a form to the database, large and detailed forms would lead to sparsely filled records [10]. Therefore, clinical documentation often relies on a relational database model using a patient/attribute/value model [11].

The core of the data model used in the ArchiMed system (described in detail in [12]) is build by the elements item, form, form version, data field, document and value. A form consists of a number of assiged items from the catalogue of items. The allocation is carried out using a form variable (data field). Each document relates to one form(version) and the stored values of this document relate to the corresponding form variable (data field).

The variables on a form can be characterized in several different ways: by *item type* (e.g., numeric, text), by *form of presentation* (e.g., checkbox, selection list), by *design details* (e.g., size, font, name) or by *data acquisition* (e.g., entered, generated).

- *Type of items* – In addition to the common data types (text, numeric, date, time, time stamp), code fields and their associated code text fields (which should be mutually generated), as well as graphical elements, play an ever more important role. In addition, modern systems support the integration of multimedia objects such as images or video clips (e.g., x-rays or ECG readings). These objects - which will not be addressed in detail within this article - may be used to assist in the documentation task, or might themselves be used as input objects (e.g., a clickable image).

- *Form of presentation* – Each item can be presented as form variable using different widgets. A text item can be presented as input field, editor window, radio button, selection list, checkbox etc. In addition to widgets for data, other widgets that serve control functions (e.g., buttons for branching to a different area of the form) and screen design widgets with no active function (e.g., lines) are also usually available.
- *Design details* – These include the various components to present a variable (e.g., a text variable can be composed of a label, a input area, a unit and a memo field) and visualization options such as fonts and font sizes.
- *Data acquisition* – Supplementary to data entry, automatic generation of specific values is important for the effectiveness and acceptance of documentation systems. There are different ways of generating new values: *calculated values (e.g.,* "days since surgery"), *compressions* of complex data (e.g., several techniques for reconstructing the mitral valve being condensed into a single variable "mitral valve surgery" that has just one of the values "yes" or "no") or *pre-filled values* derived as incorporations of existing data from previous examinations (e.g., the last medication prescribed).

In general, flexible adaptability of forms is an important consideration in clinical research. Unlike forms used in clinical routine, forms for clinical research are modified frequently. Studies often extend over long periods of time, and their forms are subject to frequent changes, as when interim findings give rise to new questions. It is therefore an important requirement that the documentation has to be amenable to adaptation. It must be possible to independently alter the content of a variable as well as its form of presentation [13].

Examples of such modifications are: changing the widget type of a numerical variable from a radio button to a selection list; grouping various variables within a table; removing an existing variable from a group; and in some cases it may even be desirable to change a variable type. The effects of changes to an existing form (creating a new version) must have no adverse effect on retrievability.

### 2.2 Data Model, Form Designer and Retrieval Tools

Guidelines for the design of clinical research forms must always be seen within the context of the options offered by the underlying *data model*, the *form designer tool*, and the *retrieval tools*.

For analytical processes data may be transferred to a system designed specifically for analytical processes [14], or the original documents are accessed directly, as in the ArchiMed system. In order to formulate logical queries, it is certainly desirable to access not only the medical data themselves, but also administrative and form information. The *data model* should therefore present the following data for query formulation [15]:

- *medical* data (e.g., numeric values, dates, text) to facilitate formulations such as "difference of surgery date and date of death (in month)";
- *administrative* data (e.g., patients, documents, cases) to facilitate formulations such as "last two periods of hospitalization";
- *form-related* data (e.g., pages, groups, or lines of a table) to facilitate formulations such as "on the medical history page" .

The retrievability of *form related data* is a very important issue in forming logical queries. When designing a form, it is not enough to consider easy presentation during documentation or patient-oriented requests. The design of the forms should reflect the fact that certain groups of variables are *logical related* (e.g., complication to a specific operation) and permit subsequent retrieval of the pertinent data as a group. The top level of organization is the *form* itself. Additionally, logical relationship should be emphasized by means of *pages*, *groups,* or *tables*.

Tables with multiple variables are characterized by the fact, that the functions of the individual variables are additionally augmented by the 'line function'. If exact relations can not be defined, counts may be to high. To analyze only related values together (e.g., variables "surgery" and "date of surgery") queries should be enabled to refer to lines in search and synchronization conditions (e.g., "and contained in the same line of the table").

Amongst others, a *form designer tool* should include the following options to support subsequent analysis:

- It should be possible to *pre-fill* variables. These can be derived from selected document fields or calculations (e.g., data from the last follow-up visit). The variables may be editable or read-only.
- A *formula editor* should offer comprehensive functions for generating additional variables. It should offer logical, arithmetic, statistical, and temporal functions for calculating new values.
- It should be possible to create *summary and query forms* automatically.

In any clinical documentation system, acquisition of patient-related data must be accompanied by a whole array of supplementary measures. Program tools are required for maintaining thesauri, for administering access rights, and for unit conversion (where the value for a variable can be documented in different units). The crucial quality requirement is to maintain a well-documented dictionary of all the variables used.

## 2.3 Design Guidelines for Enhancing Retrievability

The guidelines presented below, originate from our work with the system ArchiMed [12], designed to support clinical research. The most important input arose within a testing phase of 6 month, in close co-operation with the ‘power-users’. The guidelines are still evolving from experiences during the iterative process of *design, data entry* and *retrieval.*

Usually, the path from patient records to statistical evaluation is a long and thorny one [16]. Since most queries cannot be made directly, it is often necessary to conduct extensive data transformations involving extra software. The guidelines presented here, however, are primarily aimed at supporting direct and interactive evaluation by clinicians themselves with no programming involved.

*(1) Selecting list entries versus free-text*

When entering values for a text variable, the question arises whether - for reasons of data quality, standardization, or improved analysis - the user should be forced to select one of a number of predetermined texts/values. Otherwise synonym errors can occur during evaluation. This can be implemented by radio buttons (if the number of choices is small) or by a text field with a selection list.

The following options should be available for selection list elements:

- texts from a selection list only;
- texts from a selection list, where text may be entered but the final value must be an element of the selection list, with the remainder of the text entry being supplied automatically as soon as the first few letters entered are recognized as unique. This option is normally used when the selection list is long and the user knows the most common entries;

If it is necessary, to allow free text entry additionally to the selection list, following process should be considered: permit free text at the beginning, but evaluate these texts after some time, correct them as needed and add the approved values to the selection list canceling the free-text option.

*(2) Data reference vs. data entry*

Data already maintained outside the form (e.g., the patient’s date of death) should be taken over and displayed. These data must be available to any user, thus avoiding data inconsistencies. Multiple entries increase the probability of input errors (synonym errors). Note that it should be possible for the user to have wrong data corrected (e.g., by sending a message to the system administrator).

*(3) Analysis of unavailable/unknown values*

In all cases, where the value “unavailable/unknown” is of medical importance, this value must be provided. Checkboxes are relatively clear and easy to mark. But checkboxes used for apparently binary variables may cause problems. For example “yes” will be saved in the database if the checkbox is marked and “no” if it is not. When interpreting the data, no distinction can therefore be made between “no” and “not available/unknown.” If “no” has to be stored as a value distinct from “not available,” the widget type ‘radio button’ should be used instead. However, even in case of radio buttons the option for “unavailable/unknown” is often forgotten.

*(4) Clear identifiers for multiple items on a form*

Different variables on forms which refer to the same item in the data dictionary must be labeled according to their context already during the design process, otherwise it may be difficult to distinguish them later in retrieval sessions.

Design details are often treated sluttish, since the form designer is usually very familiar with his or her “own” research variables. If multiple variables on a form (e.g., sodium pre- and postoperatively) are not labeled clearly, undesirable values may be included (homonym errors) even in simple analyses like frequencies. It must be guaranteed that those variables can be distinguished referring to their location on the form (e.g., page or group) and to their names too, to analyze them jointly and separately.

*(5) Multiple Values within Tables*

The use of tables permits multiple data entries for variables. Each line of a table must have the same weight. Interpreting the order of lines (e.g., 1st line -> 1st surgeon, 2nd line -> 2nd surgeon”) may be incorrect. Manipulations of lines (insertions or deletions) will destroy additional interpretations.

*(6) Mapping to a different item-type*

Specific methods of analysis are available for each *type of item*. Mappings to other item types should be done during the design process, otherwise, it may be difficult in the retrieval phase to make the necessary assignments.

As an example, four different stages may be documented using a text variable with an ordered set of possible values. For analyses it might be desirable to create a rank order. If the text and the stages are used in analyses, a transformation mapping texts to numbers has to be performed (e.g., I → 11; II a → 21; II b → 22; III → 31, and so on).

This generates a new variable. However, if the texts are only needed to help users make their selection, radio buttons can be used to present texts on the user interface while storing a numeric value in the database, eliminating the need for an additional variable.

### *(7) Documentation of temporal uncertainty*

The input of imprecise temporal data should be supported. There are often events that cannot be dated exactly. For example, many patients will not remember the exact dates of vaccinations or complications. The clinician must therefore be offered the option to enter imprecise dates, e.g., by entering only the year or only the year and month. On the other hand, it would be desirable to analyze these imprecise dates, for instance to determine the temporal distance from other events.

Ideally, it should be possible to store and interpret dates in this fuzzy form. Although there are many database models with extended temporal representations [17], most common systems do not support them. However, there are also other options, such as using three fields for entering year, month, and day separately, where not every field has to be filled. Imprecise dates entered are converted to a ‘median’ date (e.g., 1993-07 becomes 1993-07-15), which is then stored and represented to the user in an additional field for control.

### *(8) Summary variables*

Creating summary variables is one of the most important preparation for data analysis. Complex information can be made more transparent, easier to understand, and easier to check by automatic summaries. Calculating *scores* and *aggregates* from several variables, where both, missing values and weights play a role, can be made in a more efficient way at the time of designing the form, so that programming complicated transformation routines prior to data analysis can be avoided.

Figure 1 shows a page with eight different anamnestic risk factors. They can be documented in detail, using different widgets. For retrieval, however, a quick overview is required. Therefore, an additional summary variable *Risk* is created. The form visualizes the values for this variable as a column of checkboxes (to the right).

If data referring to a specific risk is entered, (e.g., type and amount of nicotine consume), it is calculated if there is a risk. If this is the case, a value for the summary variable *Risk* is generated and the corresponding checkbox is marked. The right window of figure 1 shows the position information of the checkbox including the calculation if the risk ‘nicotine’. The summary variable makes it possible to answer simple questions like “Risk present?” or “Nicotine risk present?”, without further calculations in the queries, despite the fact that the documentation contains a considerable degree of detail.

Figure 1

Summary variables do not always refer to calculations designed at the time when the form is created. They can also, refer to subsequent documentation or new calculations based on requirements that were either unknown or neglected at the time the form was designed. To prevent data inconsistency during the post-documentation stage or when changing a calculation procedure, a decision process must be provided, if data already generated before should be protected.

# 3 Results

The long experience of our institute with applications for designing forms originates from the operation of the system DOKU of WAMIS) [18], WAMASTAT [19] and ArchiMed [12]. WAMIS and WAMASTAT are host applications that have been used at the Vienna General hospital for clinical research since the 1970s and 1980's respectively.

## 3.1 ArchiMed forms

ArchiMed is a platform-independent medical information and retrieval system used since 1997 by four University Clinics in Vienna and Graz, Austria. By way of a brief overview, we are presenting here the clinical research forms used by the Department of Surgery at the University of Vienna. At this point (March, 2000) 14.8 million values for 266,000 patients are stored in 762,000 documents. From these 14.8 million values, 10 % have been entered manually, 90 % have been imported.

There are 159 forms originates from 25 studies. Forms consist of an average 3.5 pages and 72 variables. In addition to the beautifully designed *clinical research forms*, there are approximately ten *system forms.* These consist mainly of a relative coarsely drawn list of variables. The values for the variables of these documents (e.g., laboratory results, diagnosis and therapies) are imported from external systems, mainly from the laboratory information system and the central HIS (Hospital Information System).

# Table 1

Table 1 shows the widgets used for the various item types on ArchiMed forms at the surgery clinic. In addition to the data fields, there are function buttons and functionless design elements. The overview also includes the variables summarized in tables. On the 159 forms used, single fields have been combined to form tables in 786 instances, for an average of 5 tables per form. The number of code items is included in the number of other forms of presentation. For example, a code wizard can be called to help select an ICD-10 code to be entered in a text field. Radio buttons and checkboxes often represent 'miniature coding systems'.

### *3.2 Evaluation*

The development of guidelines is an iterative process, where early, rapid prototyping plays an important role [20]. Our experiences have shown that the best results are in the areas, where a close cooperation between developers and users exists. The form design tool of the ArchiMed system, offers this possibility to develop a form quickly and easily during a session with a clinician and therefore supports useful interactions. However, over the course of the project, it was often necessary to revise the original procedure in order to ensure compatibility of documentation and retrieval options.

One of the main goal in assuring exact analyses is to avoid wrong values, caused by entering a value multiple times that already exists in the database. To evaluate guideline 2 (reference versus data entry), as an example, we examined the item 'date of death', which is important for many survival statistics. Analyzing the date of death for 1150 patients with kidney transplantation (relevant for survival statistics) produced the following result: subtracting 230 patients, which have only one date documented (80 by the user, 150 documented by the central patient administration), there are 920 patients with user- and administration-documented dates. The nonconformity for these 920 dates is 101 (11 %). Observing only patients with more than 1 day difference, there remain 64 (7 %). The median of the difference for those 64 patients was 4.5 days, with data varying from -413 to 533 days.

A common problem during retrieval is to distinguish variables on a form which refer to the same item in the dictionary (e.g., blood pressure). If there are no clear identifiers, users who are not familiar with the form (e.g., students) may use the wrong variable. To show the importance of clear identifiers (guideline 4), we examined, how often items occur multiple times on a single form. We found 203 cases with 1 repetition (165 variables) up to 10 repetitions (5 variables) on one form.

Guidelines have to be accompanied by technical and organizational arrangements. Whereas forms data originates from clinical routine rarely change over time (e.g., 43 new labor items in the last 1.5 years), research forms are frequently altered. As an example, it should be possible to change the set of input values of a variable (e.g., removing obsolete drugs from a selection list) but still allow to display and analyze existing values.

## 4 Discussion

This article describes a number of guidelines for improving the retrievability of clinical research forms. The criteria for designing forms must be seen in connection with the available tools for data retrieval and analysis. It is important to strike a balance between relations, functions, and aggregates already prepared by the form and those relegated to the queries. The query languages (e.g., SQL) of present-day database systems (e.g., Oracle) are powerful, but often still insufficient for evaluating complex patient histories. Clinical retrieval systems must be able to synchronize variables in time or to integrate individually developed and coded program steps. Even where there are exquisite tools for evaluation, pre-defined functions or summary sheets may be most serviceable. Nevertheless, this is of disadvantage to the data redundancy.

Guidelines must also be seen within the context of available design and documentation functions. If, for instance, the possibility exists to supply pre-filled values for variables, there should be the option to do this either *automatically* or *semi-automatically*. Pre-fillings such as “Data from the closest examination no sooner than three days before and no later than three days after” may implicate variables from a number of documents. ArchiMed uses a *Data Copying Wizard* that supports semi-automatic filling, where users can select among the possible documents from which the value can be taken. This can ensure, for instance, that all values are imported from the same document.

Furthermore, design criteria must be seen in the context of other criteria pertaining to the clinical situation. It is a good thing that a formula can be changed retroactively in the light of new findings, but the newly calculated values may very well turn out to be at odds with results already published.

While guidelines are necessary, there nevertheless remains a lot of creative freedom for form designers. Most ArchiMed forms consist of several pages the size of one screen, including tabs so that users can easily alternate between them. Some users, however, will surely prefer scrolling through larger pages.

## 5 Conclusion

Clinical forms are primarily patient-oriented. But high-quality clinical data have long been a basis of medical research and decision support [21]. Requirements of computer systems differ depending on whether they are used to support patient care or to analyze clinical data. Moreover, for each hospital department different aspects are important for the retrieval process. Assuring data quality is certainly a central focus when defining requirements for retrievability. A lot can be achieved by simple means, for instance by checking if a given value is within a predefined range. But organizational measures are also important, since they are the only way to obtain detailed, complete, and interpretable documentation.

## Acknowledgments

The authors wish to thank Gerald Boldt, Wolfgang Dorda, and Peter Sachs for their valuable suggestions.

## References

[1] Shneiderman B. *Designing the User Interface*. Reading, Mass: Addison-Wesley, 1992.

[2] Laurel B. *The art of human-computer interface design*. Reading, Mass: Addison-Wesley, 1999.

[3] Edmonds E. Human-computer interface evaluation: not user-friendliness but design for operation. Med Inform 1990; 15(3): 253-260.

[4] Williams JG, Morgan JM. The clinician-information interface. Proceedings Medinfo 1995; 801-805.

[5] van Ginneken AM, de Wilde M, van Mulligen EM, Stam H. Can data representation and interface demands be reconciled? Approach in ORCA. Proc AMIA Symp 1997; 779-83.

[6] van Mulligen EM, Stam H, van Ginneken AM. Clinical data entry. Proc AMIA Symp 1998; 81-5.

[7] Boralv E, Goransson B, Olsson E, Sandblad B. Usability and efficiency. The HELIOS approach to development of user interfaces. Comput Methods Programs Biomed 1994; 45 Suppl: 47-S64.

[8] Aisaka K, Tsutsui K, Murakami Y, et al. User interface design and evaluation for electronic medical record system. Proceedings Medinfo 1995; 781-784.

[9] Marin R, Taboada M, Mira J, Barreiro A, Otero RP. Design and integration of a graphic interface for an expert system in oncology. Int J Biomed Comput 1993; 33(1): 25-43.

[10] van Ginneken AM, Stam H, van Mulligen EM, de Wilde M, van Mastrigt R, van Bemmel JH. ORCA: the versatile CPR. Methods Inf Med 1999; 38(4-5): 332-8.

[11] Nadkarni PM. QAV: Querying entity-attribute-value metadata in a biomedical database. Comput Methods Programs Biomed 1997; 53(2): 93-103.

[12] Dorda W, Wrba T, Duftschmid G, et al. ArchiMed - A Medical Information- and Retrieval System, Methods Inf Med 1999; 38: 16-24.

[13] Singer SW, Meinert CL. Format-independent data collection forms. Control Clin Trials 1995; 16(6): 363-376.

[14] Canfield K, Silva M, Zhaohui S, Manchanda R, Abbott P. Transforms of the computerized patient record data model: from transactions to analytical processing. Proceedings Annu Symp Comput Appl Med Care 1995; 343-346.

[15] Gall W, Sachs P, Duftschmid G, Dorda W. A Retrieval System for the Selection and Statistical Analysis of Clinical Data, Medical Inf 1999; 24: 201-212.

[16] Pisanelli DM, Ricci FL. Patient records: from single events to elements for health planning. Methods Inf Med 1994; 33(5): 473-478.

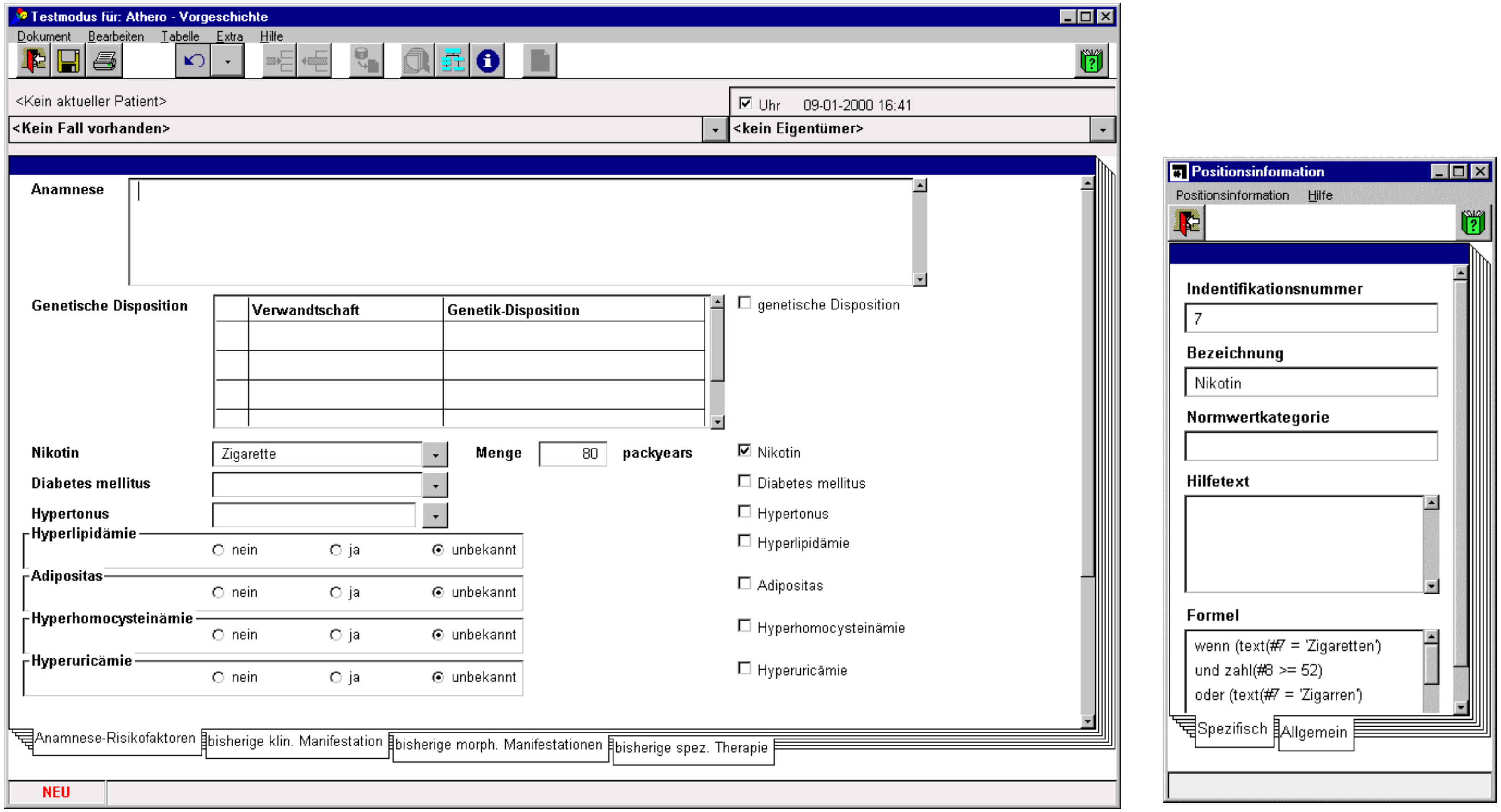
**[17]** Tansel A, Clifford J, Gadia S, Jajodia S, Segev A, Snodgrass R (Eds.). Temporal Databases: Theory, Design and Implementation. Benjamin/Cummings, 1993.

[18] Grabner G. WAMIS - Wiener allgemeines medizinisches Informations-System. Berlin, Springer, 1985.

[19] Laminger B, Dorda W, Reichetzeder C. WAMASTAT - ein System zur Benutzerfuehrung bei der Auswertung klinischer Studien. EDV in Medizin und Biologie 1986; 17 (1/2): 1-5.

[20] Patel VL, Kushniruk AW. Interface design for health care environments: the role of cognitive science. Proc AMIA Symp 1998; 29-37.

[21] Safran C, Chute CG. Exploration and exploitation of clinical databases. Int J Bio Med Comput 1995; 39(1): 151-156.


Testmodus für: Athero - Vorgeschichte
Dokument Bearbeiten Tabelle Extra Hilfe

<Kein aktueller Patient>

☑ Uhr 09-01-2000 16:41

<Kein Fall vorhanden>

<kein Eigentümer>

Anamnese

Genetische Disposition

| | Verwandtschaft | Genetik-Disposition |
|---|---|---|
| | | |
| | | |
| | | |
| | | |

□ genetische Disposition

| | | | | |
|---|---|---|---|---|
| Nikotin | Zigarette | Menge 80 packyears | | ☑ Nikotin |
| Diabetes mellitus | | | | □ Diabetes mellitus |
| Hypertonus | | | | □ Hypertonus |
| Hyperlipidämie | ○ nein | ○ ja | ● unbekannt | □ Hyperlipidämie |
| Adipositas | ○ nein | ○ ja | ● unbekannt | □ Adipositas |
| Hyperhomocysteinämie | ○ nein | ○ ja | ● unbekannt | □ Hyperhomocysteinämie |
| Hyperuricämie | ○ nein | ○ ja | ● unbekannt | □ Hyperuricämie |

Anamnese-Risikofaktoren
bisherige klin. Manifestation
bisherige morph. Manifestationen
bisherige spez. Therapie
NEU

Indentifikationsnummer

7

Bezeichnung

Nikotin

Normwertkategorie

Hilfetext

Formel

```
wenn (text(#7 = 'Zigaretten')
und zahl(#8 >= 52)
oder (text(#7 = 'Zigarren')
```

Fig. 1Page for identifying risk factors in the patient's medical history, form Atherohistory. In addition to a text edit field for entering the patient's relevant medical history, the form contains 8 different risk factors in various forms of presentation and a summary variable. The 8 checkboxes on the right represent the values for the summary variable Risk. The window on the right shows position information, including the relevant formula, for the checkbox related to the nicotine risk.

Table 1

Widgets for data entry and display, function buttons and design elements on the 159 ArchiMed forms at the Surgical Clinic, Vienna General Hospital (March, 2000).

| Form of presentation | Type | n | ∑ |
|---|---|---|---|
| Input field | Text | 1581 | |
| | Numeric | 3934 | |
| | Time stamp | 50 | |
| | Date | 781 | |
| | Time | 63 | 6409 |
| Edit field | Text | 137 | 137 |
| Selection list (Combo Box) | Text | 2412 | |
| | Numeric | 46 | 2458 |
| Radio button | Text | 985 | |
| | Numeric | 45 | 1030 |
| Checkbox | Text | 1403 | |
| | Numeric | 65 | 1468 |
| Buttons | Function | 124 | 124 |
| Frames and lines | Design | 162 | 162 |
| Text | Design | 265 | 265 |
| Graphics | Design | 12 | 12 |