

W. d'Hollosy, M. H. F. Walrave,  
B. Th. Hendriks,  
F. M. J. Debruyne, H. Wijkstra

Department of Urology, University  
Hospital Nijmegen, The Netherlands

# Semi-automated Database Design by the End-user

**Abstract:** An information system was developed to manage the data for a large number of research projects simultaneously. The system, called URIS, has facilitated the management of research data in an academic urological department. It enables end-users, who are not necessarily skilled computer scientists, to design their own databases semi-automatically, by supporting data entry screen design and the specification of research items. The system creates the database tables automatically after these activities. The specification of research items is the most important but also most difficult part in this process.

**Keywords:** Information Systems, Databases, Data Management, Semi-automated Database Design

## Introduction

The amount of data in modern hospitals is increasing. The technology to support data management has become widely available during the last decades [1-3] and is increasingly used to manage large amounts of medical data.

The need for computerized data management also increased in our department, for both clinical and scientific research in urology. Small amounts of patient data can be stored in the hospital information system (HIS), but a HIS is primarily meant to support daily patient care [4] and not to store research data. Therefore, temporary information systems were often developed as ad hoc solutions for data management problems in new research projects. The users of these systems were mainly physicians, medical students and data managers. Over time this ad hoc approach caused some difficulties:

- The development of new information systems became a very time consuming and expensive task of our biomedical engineering unit, especially because users could not adapt their system without the help of a system's designer. The database structures of these systems often had to be changed during the lifetime of a study, due to misinterpretation of

user requirements in item specification by the system designer, or due to items overlooked by the researchers.

- Difficulties also arose when a meta-analysis had to be performed, caused by data storage in different systems and non-standardized item definition. Meta-analysis is an attempt to combine the results from different studies for a common research topic [5].

- The temporary information systems did not enable to enter data from different sites. This possibility can be very helpful in performing multicenter studies, or processing data at different locations.

A standard system was needed to overcome these problems. System requirements were defined in consultation with the users. The main requirements were:

- The system should be able to transfer database design from computer scientists to end-users via a user-friendly interface.
- The system should be able to manage the data of a non-limited number of different research projects simultaneously.
- Real-time data verification should be possible during data input or modification.
- The system should be able to manage all kinds and types of data.

- It should be possible to manage data of one project at multiple sites that are not directly inter-connected. The possibility of data management should not depend on site location or the availability of other sites.

In the literature, several clinical and research information and database management systems are described [1, 6-8]. Most of them focus on the collection and use of clinical data for one or more clinical trials, and sometimes for individual therapy and research. No system, however, was found that could meet all the requirements described above.

We started the development of our **UR**ological Information System (URIS) in 1991. URIS enables to manage data for clinical usage and scientific research in urology, and involves end-users in the design of their own database structure. The system has been set up in a universal way, with potential applications outside urology.

We discuss the development and application of URIS, focussing on semi-automated database design by the end-user. In the next section, the development environment, the system structure and the method of semi-automated database design are described. The results and discussion section presents the evaluation of the current application of URIS and discusses the current pro-

and cons of the system with respect to database design and modification, data validation and data analysis.

## Material and Methods

### Development Environment

Personal computers were chosen as the platform for URIS, as is also done in many similar projects on data management [7, 9, 10]. URIS is based on the Microsoft® Foxpro® 2.5 for MS-DOS® database management system.

### System Structure

The URIS structure exists of a logical and physical part, strongly inter-related. The logical system structure describes the logical connection between the different research projects of which data are stored in URIS. The physical system structure describes the distribution of these research projects over different PCs. URIS is described in detail with emphasis on the logical structure. The physical system structure is only described briefly, as this is outside the scope of this article.

### Logical Structure

The logical structure of URIS exists of different parts. A part is called a *project* and is an environment used for data management of a single research project.

A tree structure, as shown in Fig. 1, is used to connect these projects. This tree defines the logical structure in which a parent project can access data stored in its children.

The root of the logical structure is also a project and is used to maintain the URIS structure. Only the URIS system manager has access to the root project to control this hierarchy. The tree structure can be expanded and pruned. If data management is required for a new research project, a new project is appended by the system manager to the tree with the help of the root project. A new project cannot immediately be used for data management. Project-specific research items have to be specified first. This is described below.

The root project can also be used to prune the tree by removing finished projects. Projects removed from the tree can be restored simply, provided that these projects are backed-up carefully. Restored projects can be used again for data management.

### Physical Structure

The physical system structure describes the distribution of the logical structure among different sites. Those sites are also organized in a tree structure in which the root is the main site. A project can be located at one or more sites, except for the URIS main project, without the need of a direct connection between those sites and the presence of

the total logical structure at the same site. The only restriction is that if a project is found at one site then all child projects should also be found at that site. The URIS main project, and consequently the total logical structure, is only found at the main site. A special update algorithm provides for data exchange between the different sites for which diskettes, telephone lines and network connections can be used.

### Database Design

End-users should specify the research items before their projects can be used for data management. URIS automatically creates the necessary database tables based on these specifications. This way of database design has several advantages:

- Misinterpretation of user requirements in item specification is avoided.
- The problem of time-consuming development of new information systems for new studies is solved, because the desired database tables will automatically be created by the system after item specification. This means that part of the time needed to design a totally new information system is shifted to the end-users. More time is saved by the automated creation of new projects by URIS.
- Uniformity is obtained in the development process of database design, also leading to uniformity in data storage, especially when standardized item specification can be enforced for recurring research items.

Data stored in URIS can be split into two groups: A distinction is made between *identification data* and *additional data*. Identification data are data to recognize, e. g., a single patient in a group of patients. In URIS several groups of *objects*, called *classes* can be used. This means that URIS is not restricted to patient-related data only.

Identification data are stored in global tables, one per class, accessible for all projects in the project tree. Patients in URIS are identified by the following items:

- unique number in URIS,
- unique number at the local site,
- hospital information system code,
- hospital code (in case of multi-center trials),

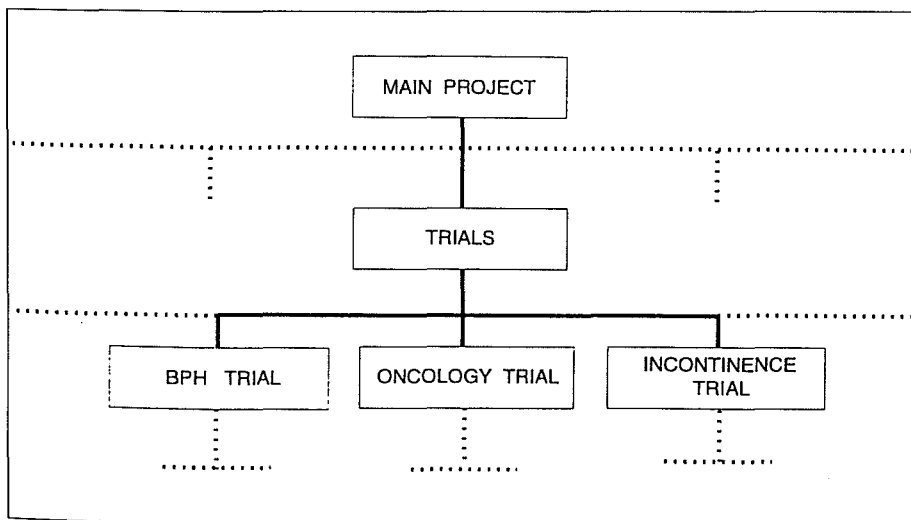


Fig. 1 Example of the logical URIS structure. Each part of URIS to be used for research data management is called a project. These projects are organized in a tree structure, which is the logical URIS structure.

- surname,
- first name,
- initials,
- date of birth,
- gender.

When a new patient is entered at one site, a local unique number is attached to that patient. The value of this number is the last distributed local unique number at that site to a patient, plus one. During updates, the patient is made known to URIS and also receives a unique URIS number from the main site. The need for two unique numbers instead of one is due to the indirect connection between the different sites.

Additional data are, for example, research data collected during a clinical trial and are stored in tables of the project databases. A project database can be divided into the following groups of tables: *item tables*, *screen tables*, and *data collection tables*. The item tables contain information about the specified project items. The screen tables contain information about the construction of the input screens in relation to the specified items. The structures of the data collection tables are based on the information in the item tables and contain the entered or edited data related to one class. Several data collection tables can be used within one project. If, for example, the project had been set up to manage data of a clinical trial, different data collection tables can be used to store data collected during different follow-up visits. No patient has more than one record in a data collection table.

If a patient is included in one or more trials, the identification data are stored centrally in the global patient table. The specific trial data are stored in the data collection tables of the corresponding project databases. The unique numbers provide relations between the global and data collection tables.

End-users are involved in the design of the data collection tables, consisting of three stages: (1) Input screen design; (2) Item specification; and (3) Automated table creation.

### Input Screen Design

During this phase the end-user constructs the input screens for his or her project. Screen design in URIS implies

**Table 1** Current list of item types with corresponding additional attributes.

Item Type	Additional Attributes	Description
String	None	A series of characters (max. 75) Examples: surname, address
Integer	Upper warning limit Lower warning limit Upper error limit Lower error limit	An integer number Example: age
Real	Upper warning limit Lower warning limit Upper error limit Lower error limit Position floating point	A real number with a floating decimal Examples: laboratory result, length
Date	None	A calendar date Example: date of birth
Memo	None	A series of free-text characters (no maximum) Example: comment
Choice	Valid characters	A character Example: married # (Y/N)?
List	Option list	A series of characters (max. 75). The value is chosen from a list of options. Example: medical treatment
Calc	Calculated expression	The value is calculated from values of other items. Example: score

only the creation of text. The input fields are indicated by special characters, and each input field corresponds to one of the research items. The number of characters specifies the length of the corresponding item.

The screens can be designed inside as well as outside URIS. Externally designed screens can be imported into URIS. This enables the screen designers to use the text editors they are accustomed to. Previously designed questionnaires can also be imported, which means that these paper forms and related data entry screens can be designed at once.

### Item Specification

The information on item specification is stored in the item table. An item behind an input field is specified by at least its *name*, *description*, and *type*. These item characteristics are the main *attributes* of an item. The item attribute *name* is a unique name that is used later as a field name in the data collection table. The item attribute *description* describes the meaning of this item in further detail. The item attribute *type* specifies the nature of an item.

Depending on the item type attrib-

ute, additional item attributes have to be specified. These additional attributes are used during data entry or modification, for example in data verification. Each item type has its own specific additional attributes and its own software procedures to deal with items of that type. New item types can be added to URIS by the system programmers. Table 1 gives an overview of the current item types and the additional attributes.

During this phase the user can also select standard items or already defined items. Standard items can be research items often used in research projects. Already defined items were already specified beforehand for that project. In clinical trials, for example, an item can recur during different follow-up visits.

### Database Creation

After item specification, the data collection tables are created automatically by the system. The item table is used as structure description. The field names of the record structures correspond with the names of the specified items. The first fields of a record contain the unique numbers that point to a record with identification data in the corresponding global table.

## Database Modification

Once the data collection tables have been created, data can be stored. The structure of these tables can be modified if necessary. This modification procedure is similar to the design procedure. The screen editor is used to add, modify, or remove items.

New items can be included by adding new input fields to the input screens. The items behind these fields have to be specified before the data collection tables can be modified. This then follows the same process as described above. The addition of new items is possible anytime, even if data have already been stored.

Editing items means the adjustment of item attributes. The adjustment of item attributes is not possible when this affects already stored data. In this way, modification of already specified items will not have consequences for the specific research results.

Items can be deleted by removing the corresponding input fields from the input screens. Items cannot be deleted if data related to these items have already been stored. In this case, the corresponding input fields can be removed from the input screens, but the data still remain in the data collection tables.

## Results and Discussion

At this moment URIS is managing 30 projects, including the URIS main

project. The latter project is not used for data management, but to maintain the system structure. URIS is most frequently used by researchers of the trial office of our department. This unit performs clinical and scientific research on medical treatments. Research topics include urological oncology, antibiotics, incontinence, impotence, and benign prostatic hyperplasia (BPH) [11, 12]. Other projects, used for data management outside the trial office, concern research topics such as pediatric urology and electrostimulation of the bladder [13]. Most data managed by URIS projects are patient-related data. At present, research data of more than 3400 patients are managed in 26 projects. Research data of about 10% of the patients are stored in two or more projects (Fig. 2). The current average number of patients in a project is 153 and increases up to 828.

The addition of a new project to the project tree by the system manager takes only a few minutes. Before this project can be used to manage data, the data collection tables have to be designed first by specifying the research items. The current number of specified research items for a project ranges between 21 and 1091 (average 363). The time for input screen design and item specification of an average project depends on the number and complexity of the research items and the experience of the user with the URIS interface. The time to specify an item is about one

minute. The name, type, description and additional attributes of the item are specified during this time. This specification is checked during data entry after the data collection tables have been created by the system. The average time for input screen design and item specification of an average project (360 items) is thus the time needed to design the data entry screens plus 360 times 1 minute. This means that about one or two working days are needed to prepare a project for data management. In comparison, the time to design a totally new information system, involving database and software design, usually took several weeks.

It should be noted that this method requires more effort by the end-users in database design. However, they can now design and modify the data entry screens and structures of the database tables without the help of the system manager. The time they had to wait for a computer expert for information system design and modification is now saved.

The time of data collection table design is more accelerated in projects dealing with storage of follow-up data, requiring the same items during different patient visits. In these projects, the input screens can be divided into subscreens that are duplicates of each other. By specifying the items for one subscreen, the items of the other subscreens are also specified. The data are stored in different data collection tables to keep data of different follow-ups apart. Each table corresponds to one subscreen.

It is also possible to "recycle" already designed data entry screens and corresponding items of an earlier project into a new project. This means that (adapted) copies of the screen table and the item table of an earlier project are transferred to the new project. A disadvantage is that this can only be done by the system manager of URIS, because the URIS software does not yet support this possibility.

Most data stored in URIS are clinical trials data. A problem with clinical trials is that often many items have to be specified and many complex hierarchical relationships exist between those items [14]. This is also a problem in URIS. A patient may visit the clinic

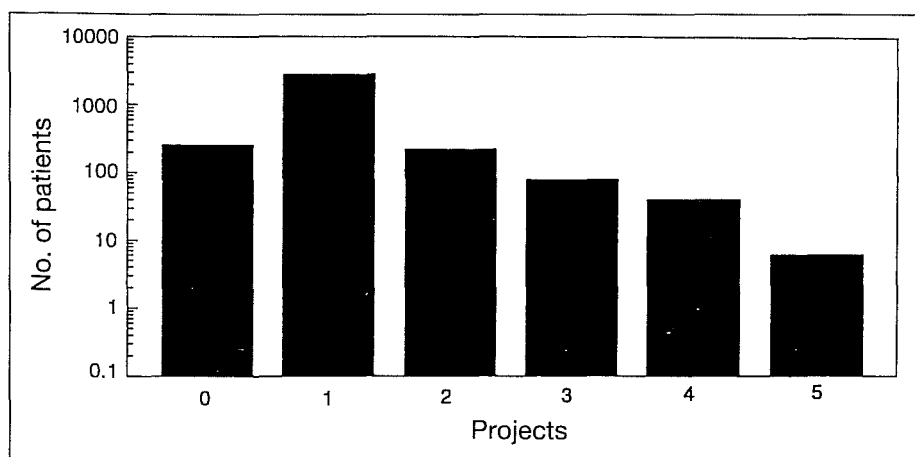


Fig. 2 Current distribution of patients among different projects. If research data of a patient are stored in a project, then this patient is said to be found in this project. Several patients are found in none of the projects, since these patients were present in projects that were removed from the logical structure of URIS, or were removed from current projects. About 10% of the patients were found in two or more projects.

one or more times during a clinical trial. During a visit an item can be determined several times, e. g., the bladder contents before and after urine flow.

URIS offers several possibilities to store this kind of data. Screens can be divided into subscreens in which each subscreen is used for data entry of, e. g., a visit, or a screen is used to enter data from different visits. The structure of the data entry screens has consequences for the item specification and, thus, the table structures.

In one screen, items cannot have the same name, because the item names correspond to the field names in the research data collection table. Two input fields, however, may relate to the same item, but the value entered in one field will also appear in the other field. Therefore, if the values for bladder contents before and after urine flow are entered into one data entry screen, the item names behind the input fields should have different names, for example "contents\_b" (bladder contents before) and "contents\_a" (bladder contents after). In fact, two different items are specified representing an item that is entered at two different points in time. The use of different item names for the same item should be considered during data analysis.

Item specification can be accelerated by using standard urological items. In URIS, an item can be specified by copying the attributes of an already specified item. If, for example, a standard item is "bladder contents", then the previously mentioned items "contents\_b" and "contents\_a" can be specified by copying the attributes (type, description, additional attributes) of this standard item. A table with standard items should be accessible for all projects during item specification. This table can be used to create new item tables, leading to a standardization in item specification of items frequently used. This may, for example, lead to a more reliable data correlation between different projects. At present, such a table has been constructed with only a limited number of standard items. We intend to analyze the currently specified items in URIS projects and the frequency of their occurrence in different projects.

The data collection tables are created automatically by the system after

item specification. The creation time of a table depends on the number of items and takes typically less than one minute. Subsequently, the user can start data entry.

Data structures in the medical domain often require complex checks to guarantee data integrity [15]. In medicine, data integrity is important to warrant safety. Data errors may lead to false conclusions during data interpretation, which may cause a wrong diagnosis or unfavorable treatment protocols. The possibility in URIS to increase the number of (complex) item types and their additional attributes enlarges data integrity, because data can be better checked with respect to their validity during data input and modification. This validity check, however, is still limited to data entry and editing only. There is no control on data consistency between two projects, except for identification data.

Output facilities are incorporated in URIS as well. Firstly, a report generator is available that is based on the Foxpro® report generator. Report frames can be created to generate reports with selected data. Simple statistical data analysis is available such as the computation of means, variances and standard deviations. The reports can either be printed or stored as ASCII files. The created report frames can be saved, modified and re-used. A problem is that the use of the report generator requires some experience. Secondly, the user can create tables that contain selected data from one or more projects. These tables are called *output tables*. The contents of these tables can be used as input for, e. g., statistical software packages. This also applies for data in the reports stored as ASCII files. Often, the output tables contain data that have to be analyzed by the medical statistics department of our hospital.

The URIS output facilities may seem to be limited, but appear to be sufficient for the current applications. In the future, we want to extend the system with graphical possibilities, e. g., to display time-dependent items.

URIS has facilitated data management of clinical and scientific research at our department although some limitations still exist. By using automated tools for data management, research

projects are realized within a reasonable time. There is no limitation to the number of projects that can be handled by URIS. Research data remain accessible for other researchers, because data management is standardized. End-users can design and adapt their own data collection tables, without intervention of computer experts. Although not described here, data can be entered from different locations without the need for a direct connection to those sites. Special methods avoid problems of inconsistency in the data collection table structures at different sites, which can be caused by allowing these structures to be modified by the end-users.

One problem has not yet been mentioned. It appears to be difficult for most researchers to determine the most relevant study items beforehand. Table structures are often modified because of this. Therefore, special attention should be paid to the importance of well-prepared research protocols, in which the determination of relevant study items is an important part. To standardize research design, a system will be developed, not restricted to clinical trials [16], to support the specification of the most important research items for URIS.

#### Acknowledgments

The authors would like to thank Prof. N. J. I. Mars (University of Twente, The Netherlands) and Prof. P. F. de Vries Robbé (University of Nijmegen, The Netherlands) for their helpful criticism in the preparation of this manuscript.

#### REFERENCES

1. Herson J. Personal computer software for clinical trials. In: Rotmensz N, ed. *Data Management and Clinical Trials*. Amsterdam: Elsevier Science Publ, 1989; 235-46.
2. Date CJ. *An Introduction to Database Systems*, Vol 1 (4th ed). New York: Addison-Wesley, 1986.
3. Wiederhold G. *Database Design* (2nd ed). Tokyo: McGraw-Hill, 1983.
4. Zwetsloot-Schonk JHM. Use a hospital database to determine the characteristics of diagnostic tests. *Meth Inform Med* 1990; 29: 205-12.
5. Gelber RD, Goldhirsch A. Statistics in clinical trials. In: Rotmensz N, ed. *Data Management and Clinical Trials*. Amsterdam: Elsevier Science Publ, 1989; 175-204.
6. Koval JJ, Kwarcia LM, Grace MGA, Lockwood BJ. A comprehensive database management system for a variety of clinical trials. *Meth Inform Med* 1987; 26: 24-30.

7. Choudhury MH, Shah MA, Bojanczyk MT. PC-based research-oriented clinical information system: a case study of Kuwait. *Med Inform* 1991; 16: 323-29.
8. Morrison LM, Small DR, Glen ES. Computer applications in a urology department. *Brit J Urol* 1991; 67: 257-62.
9. Mustard RA, Cosolo A, Fisher J, Pike T, Schouten BD, Swanson HT. PC-based system for collection and analysis of physiological data. *Comp Biol Med* 1990; 20: 65-74.
10. Carter PG, Adams D, Abrams P. Personal computers and urological research. *Brit J Urol* 1992; 69: 456-9.
11. de la Rosette JJ, Froeling FM, Debruyne FM. Clinical results with microwave thermotherapy of benign prostatic hyperplasia. *Eur Urol* 1993; 23 Suppl 1: 68-71.
12. Witjes WP, van der Meijden PM, Roos EP et al. BCG-RIVM versus BCG-Tice versus mitomycin-C in superficial bladder cancer. Rationale, design, and interim analysis of the trial of the South-East Cooperative Urological Group, The Netherlands. *Prog Clin Biol Res* 1992, 378: 59-67.
13. Koldewijn EL, Van Kerrebroeck PEV, Rossier P, Wijkstra H, Debruyne FMJ. Bladder compliance after posterio sacral rhizotomies and anterior sacral root stimulation. *J Urol* 1994; 15: 955-60.
14. Helms RW, McCanless I. The conflict between relational databases and the hierarchical structure of clinical trial data. *Controlled Clinical Trials* 1990; 11: 7-11.
15. Goble CA, Glowinski A, Jeffery KG. Semantic constraints in a medical information system. In: *Lecture Notes in Computer Science: Advances in Databases* 1993; 696: 40-57.
16. Wyatt J, Altman D, Pantin C et al. Design-a-Trial, a knowledge-based aid for writers of clinical protocols. In: Reichert A, Sadan B, Bengtsson S, Bryant J, Piccolo U, eds. *Proceedings Medical Informatics Europe '93*. London: Freund Ltd, 1993: 68-73.

Address of the authors:  
W. d'Hollosy, M. Sc.,  
Department of Urology,  
P.O. Box 9101,  
6500 HB Nijmegen,  
The Netherlands