

# Design and Implementation of a Query-By-Example Interface for Micro CDS/ISIS

Andrew Treloar  
School of Computing and Mathematics  
Rusden Campus  
Deakin University  
Clayton, Vic, 3168

Email: [treloarac@brt.deakin.edu.au](mailto:treloarac@brt.deakin.edu.au) (Internet)

NOTE: I now work at Monash University – current contact details at <http://andrew.treloar.net/>

## Abstract

The CDS/ISIS database software, developed by UNESCO, has been used to create a system for storing information about medicinal and aromatic plants. This system provides effective storage and retrieval of information for users in the developing world. The task of providing access to this information for novice computer users has been tackled by providing a new (for CDS/ISIS at least) interface to the database. The new system, a modification of Zloof's Query-By-Example, has been implemented using CDS/ISIS Pascal, a subset of the language with database extensions which is built into the CDS/ISIS program.

## Keywords

Query by Example, QBE, user-interface, ISIS, text-retrieval, database searching, medicinal plants, APINMAP

# 1. Setting the Scene

## 1.1 User-interfaces

A number of aspects contribute to the usefulness of an information system. These might include processing power, the quality, relevance, and currency of the underlying data, cost of the necessary hardware and software, and many more. In the early 1990's, the most important consideration is none of these. It is useability, and the associated focus on the user interface. Partially as a result of the long debate in the micro-computer world over whether a command line interface (CLI) – such as that for MS/PC-DOS as well as for most mainframes – or a graphical user interface (GUI) – such as that popularised by the Macintosh and now available in inferior form as Microsoft's Windows – is best, users of computers everywhere are sensitised to the importance of the user-interface.

Although the debate has largely been resolved in favour of the GUI on both major hardware platforms with the success of Windows, this does not address the needs of many users in the developing world. They do not have the funds to upgrade their hardware to satisfy the increasingly rapacious demands of GUI environments and their bloated software applications (dubbed 'fatware' in a recent issue of *Byte*) As a result, there is still a need to develop effective and enabling user-interfaces that will deliver information solutions on older hardware. This technical report describes one such interface, designed to provide access to information about medicinal and aromatic plants on hardware that was state-of-the-art in about 1985.

## 1.2 Information context

Worldwide, health care is an increasing drain on society's resources. The population available to be treated is increasing at an alarming rate, and the pressures on people's health are also becoming more widespread. Added to this is the increasing cost of hi-tech medicine in developing countries. Although modern medicine has had many successes (the eradication of smallpox and an increase worldwide in immunisation programs to name a few), a continuing trend towards urbanisation, coupled with environmental degradation and man-made/natural disasters are placing great strains on the health infrastructure of many developing nations. A significant health cost for such nations is the cost of imported pharmaceuticals, often amounting to tens of millions of dollars per year.

As a result, there has been a renewed interest in using medicinal preparations from indigenous plants in the prevention and treatment of certain ailments, particularly in rural areas where access to modern pharmaceuticals is more difficult. This interest has prompted international development organisations to stimulate the creation of information support mechanisms that will allow effective dissemination and use of research results in this field.

In response to requests from interested Member States, in July 1987 UNESCO formally launched the The Asia Pacific Information Network for Medicinal and Aromatic Plants (APINMAP), with 11 participating countries. These are Australia, the People's Republic of China, India, the Republic of Korea, Nepal, Pakistan, Papua New Guinea, the Philippines, Sri Lanka, Thailand, and the Socialist Republic of Vietnam. APINMAP is a voluntary cooperative program primarily designed to assist its member countries to improve their capability to collect, process,

disseminate, and use research information and data on medicinal and aromatic plants.

The network seeks to achieve its objectives by:

- making available and using to the maximum extent possible information in this field, either acquired from abroad or generated by research and development activities in the region;
- assisting in the development or strengthening of specialised information services and information handling capabilities of member states;
- providing guidance in the development of information products and services appropriate to various target user communities of the network;
- promoting resource sharing activities and services;
- providing linkages to other regional and international networks or services in the fields of medicinal and aromatic plants and chemistry of natural products.

APINMAP operates as a specialised information network within the framework of the UNESCO-supported Regional Network for the Exchange of Information and Experiences in Science and Technology in Asia and the Pacific (ASTINFO). With a structure that combines both centralised and decentralised approaches, APINMAP acquires its strength from the information activities of the National Nodes of each participating country, which are responsible for data input and the provision of information services.

Overall technical co-ordination is currently provided by a Network Centre based in Los Baños, the Philippines. This is the Agricultural Information Bank for Asia (AIBA), at the Southeast Asian Regional Centre for Graduate Study and Research in Agriculture (SEARCA)<sup>1</sup>. Policy-making, planning and review of the network's activities is provided by a Management Board with a Secretariat based in Manila.

### **1.3 The APINMAP Integrated Database**

Early in the life of the project, APINMAP developed two different information resources. The first was a referral database of information sources, research institutions, and experts. Complementing this was a bibliographic database on medicinal and aromatic plants prepared by the National Nodes in each of the participating countries, and consolidated as a regional database by AIBA, the Network Centre. It soon became clear that there was a significant overlap between these two databases, with a number of researchers appearing both in their own right, and as authors of articles. It was therefore decided to combine these two databases together. The resulting database, now simply called the APINMAP Integrated Database, provides an excellent traditional style of information resource for researchers in the area of medicinal and aromatic plants. Access to this database is managed using the standard ISIS search and retrieval procedures (essentially boolean search sets defined and combined using a conventional command-based interface).

---

<sup>1</sup> The author wishes to acknowledge the support of SEARCA/AIBA and the funding provided by Canada's International Development Research Centre in making possible the work described in this technical report.

## 2. The APINMAP Factual Database and PFBE

At about the same time as the Integrated Database was being developed, work commenced on quite a different information resource, intended from the start to meet quite a different market. It was realised that health workers and small local pharmaceutical companies needed solid useable information about particular plants and their application. They were not interested in questions like ‘Who else is working on applications of *Cocos nucifera*?’ or ‘What has been published about CNS-depressant alkaloids’. They were much more likely to want to know the answers to questions like ‘What plants can be grown in my area to help treat dysentery’ or ‘Are there local plants that have a medicinal use, and if so, how should they be prepared?’. This meant that the information contained in the Integrated Database, useful as it was and is for some categories of users, was quite inappropriate. What was needed was concrete, factual, immediately useable information. Keeping these considerations in mind, the APINMAP Network Centre began to design a database relating to plants, their active compounds, and their medicinal uses, including preparation and administration. The two critical issues that had to be resolved for this proposed database were how to deliver the information to the end-user, and how to make it easily accessible. Delivery and access were deliberately treated as quite separate issues during the planning process, although they are often erroneously regarded as synonymous.

### 2.1 Database software

The first decision to be made was the hardware and software platforms to be used to deliver the information. This decision was in fact fairly easy. Most of the National Nodes had extremely tight constraints on resources, so working with what was already in place (if possible) was a prime consideration.

Fortunately, most of the National Nodes already had access to the UNESCO-supplied<sup>2</sup> Micro CDS/ISIS software (hereafter shortened to ISIS). This is a database that runs on ISA machines<sup>3</sup> and provides particularly good facilities for working with textual data. The software is menu-driven, highly customisable, and comes with a built-in dialect of Pascal with added database functions that can be used to create custom extensions to the ISIS system. Importantly, the software makes minimal demands on the underlying hardware. ISIS will run perfectly well on a machine with an 8088 processor, a monochrome display, 512K of RAM, and a 10 Megabyte hard disk drive. Such low-specification hardware is very common in the developing world for a range of reasons: it is cheap, there is little to go wrong with it, the technologies involved are mature<sup>4</sup>, and there is an increasing reservoir of technical expertise in most cities.

ISIS is widely used throughout the developing and developed world for a very wide range of tasks. It excels at bibliographic information but has also been applied to library automation and map cataloguing. It provides good retrieval times (even on

---

<sup>2</sup> UNESCO provides this software free to approved organisations and in some cases has also supplied the hardware needed to run it.

<sup>3</sup> Industry Standard Architecture - what used to be called IBM PC-Compatibles, or IBM clones.

<sup>4</sup> Many would say over-mature or even in decline. However, I heard a PC consultant lamenting only recently the impossibility of purchasing a new Hercules-compatible monochrome monitor in Australia. Such hardware, although perfectly adequate for many applications, is no longer in fashion.

slow hardware), compact storage of data, and can deal with databases of up to 16 million records. It is possible to use ISIS to set up a system that largely insulates users from the underlying operations of the package and simply allows them to select from a range of pre-defined choices.

Although ISIS is not a relational system, it does provide more flexibility than a conventional flat-file database. It is possible to reference information held in other databases via the REF construct, and the record structures are particularly powerful. In contrast to many other textual databases, ISIS allows repeating fields within a record, and for each field to be broken down into subfields (with an appropriate mnemonic character tag). Fields can also be forced to conform to a pre-defined pattern, thus providing a check at data entry time. This means that it would be possible to have a field that stored citation information for a journal article defined as repeating (to allow for more than one citation) and with each repeat divided into subfields for Journal, Year, Volume, Number, and Page references. Each individual repeat/subfield occurrence is accessible through the ISIS formatting language. Variable length fields and records are both supported, meaning that data storage can be quite compact.

ISIS can also be used to store multiple different types of records within the one database, by defining an all-encompassing overall record type. Such a record type might have some fields that are common to all records and others that are specific to particular variant types. On output, only those fields that are relevant for each record are displayed. This was in fact the case with the APINMAP Integrated Database.

## **2.2 Structuring the database**

The specifications for the Factual Database records were developed in consultation with the APINMAP Management Board and the Technical Standards Committee. Due to the flexibility available in ISIS, implementing the specifications was quite straightforward. ISIS defines the structure of a database using a Field Definition Table (FDT). This in turn defines the logical record structure for each record of the database. Each field is provided with a unique tag (a 1-3 digit integer used to reference fields when working with the database), a name, a maximum length, and a data type. The field can be specified as repeating or not, and any sub-field delimiters can be defined. All of this field-level information is also stored in the FDT. The FDT for the APINMAP Factual database is attached as Appendix 1.

To improve performance when retrieving information, ISIS also allows the creation of a Field Select Table. This specifies which fields are to be used to build an Inverted file index for the database in question. It is possible to specify that only certain parts of fields are to be indexed. This allows a greater degree of control over the possible vocabulary to be used when searching the database.

## **2.3 Searching the database**

The most important issue to be resolved in developing the APINMAP Factual Database was how to provide easy access for the end-users. It was felt that likely clients for the database would want to (or have to) access the information themselves, rather than through a search intermediary. This meant that the search facility for the database needed to be as simple as possible to use, consistent with being able to provide sufficiently powerful and precise retrieval.

ISIS comes with an inbuilt search facility which is quite powerful, broadly comparable with that available for a typical information retrieval system such as DIALOG or ORBIT, or that provided with many CD-ROMS. The search command language allows searches on terms indexed in the pre-defined inverted file, or free-text searching on selected fields in the database. Terms can be right-truncated if required. Individual terms or search sets can be combined using the normal range of Boolean operators. Those interested in more detail can refer to an article on applying ISIS to a map catalogue in the January / February 1991 issue of *LASIE*<sup>5</sup>, which discusses the search facility at greater length. Unfortunately, the search language is less user-friendly than most others of its type. The search commands can be complicated, and the symbols used for the Boolean operators are not intuitive. In addition, it can take some time before users become confident with the principles underlying search sets and how to combine them.

All of this can be a problem if one wants people who are not information professionals to access the data without search intermediaries. The problem is exacerbated when users may only need to (or be able to) search the database infrequently, and thus lose the constant reinforcement in search techniques provided to regular users.

For this reason, it was necessary to devise an alternative way to access the information stored in the Factual database. It was clear that the easiest way to do this was to use the programming facility built into ISIS to write a separate retrieval program that would then be linked into the overall system, appearing in the search menu as just another option. This separate program could then provide a specially designed, easy to use way of accessing the database. As the full ISIS search language would still be available, both novice and expert users would be able efficiently to access the data.

Prior to the consultancy, a prototype query program had already been developed by staff in the Computer Services Unit at SEARCA. This prototype was based on a question and answer format, which operated as follows: the user would select one out of a number of different question types (for example, 'What plants can be used to treat an ailment?'); the software would then ask for the name of the ailment and retrieve some of the information for all the appropriate plants. The prototype had been demonstrated to the APINMAP Technical Standards Committee, and they had made a number of suggestions. Chief among these was the need to provide a wide range of possible ways to query the database. For instance, some users might want all plants in a particular geographical area, others might only be interested in chemical substances occurring in plants growing in a particular habitat, others again might want to know the different ways in which plants containing specific compounds are prepared. The possibility of expanding the prototype system to allow for this was considered, but the underlying 'question and answer' model was not adequate for this (without preparing literally tens of questions, each of which would need to lead to further questions). A completely new way to query the database and retrieve information was needed. The new method had to be intuitive, flexible, preferably not based on a command language with a rigid syntax, and above all easy to learn or relearn.

---

<sup>5</sup> Puntodewo, A, "Creation of a Map Catalogue using CDS/ISIS", *LASIE*, Vol. 21, No. 4, pp. 88 - 105.

## 2.4 Plant facts by example

What eventuated was an adaptation of the Query-By-Example (QBE) methodology, which was developed at IBM in the late 1960's and early 1970's<sup>6</sup>. Its best known commercial application in the micro world is the Paradox database management system from Borland. Put simply, one fills in a blank sample form on the computer screen with an example of the data one wishes to retrieve. For instance, if one wanted all plants that could be used to cure fever, one would put 'Fever' in the Ailments column on the screen. Once the screen has been completed, the program retrieves and prints or displays (or both) all the information that matches this example.

During the consultancy, a modified version of this system was designed and programmed. Mr Joey Agena from SEARCA assisted with the programming. The APINMAP implementation of QBE, which might be called Plant Facts by Example (PFBE) provides a slightly more limited search capability than the native ISIS search language, but is significantly easier for novice users. The system is very flexible, allows an almost infinite range of possible questions to be asked of the database, and can be mastered within 15 minutes.

What users interact with is a screen displaying a number of boxes, each corresponding to one or more database fields (see fig. 1). To create a search request, searchers use the cursor keys to navigate around the screen, placing examples of the information required in the appropriate boxes. If **P.** is placed in the box headed **??**, this indicates the entire record is to be **Printed**. A **D.** placed in the same box means **Display** on the screen. **P.** and **D.** can also be placed within individual boxes if the user is only interested in particular fields of the database. When the screen (or screens if using the two screen 'complex' option) has/have been completed to the user's satisfaction, they press the enter key to execute the search. The PFBE program 'parses' the screen(s) and converts the information that has been entered into a standard ISIS search request. If more than one box has been filled in, PFBE treats this as an implied AND. The search is then run against the database. As the results of the search are retrieved, the program again intervenes to provide the user with the opportunity to print out selected records, or the entire search. This ensures that users need never interact with the ISIS print module.

---

<sup>6</sup> See Zloof, M. M. "Query by Example", *Proc. NCC*, Vol. 44, May 1975 among others

APINMAP Factual Database: Simple Query Screen				
??	PLANT NAME/SYNONYMS	FAMILY	ENG. NAME(S)	LOCAL NAME(S)
HABITAT	DISTRIB'N	DESCRIP'N	COMPOUNDS	ACTIVITY
AILMENT NAME/SYNONYMS	PREPARATION	ADMINISTRATION	REFERENCES	

F1 = Help C = Clear this Screen X = Exit Query <CR> = Execute Search

Fig. 1 - The PFBE main query screen

The entire system is provided on a submenu reached from the normal ISIS retrieval menu. The new menu has three possible entry points:

*S - Formulate a Simple Search*

This option provides the single screen shown in Fig. 1 on which to create a search request. In order to fit the entire database on this single screen a number of simplifying assumptions have been made. Briefly summarised these are:

- it is not possible to search on Participating Centre Record Number, Contributing Institution, Date of Information, Raw Materials, or Market. The assumption here is that most users will not require these;
- Plant Name and Plant Name Synonyms have been combined into a single search field. This means that users searching for a particular plant do not have to remember to enter their search term in both fields (if they are not sure whether the name they are searching for is the primary name or a synonym);
- Ailment Name and Ailment Synonyms have been combined. This means that users searching for a particular ailment do not need to know in which of these two fields their search term has been placed;
- Chemical Substances and Active Compounds and Other Compounds have been combined into a single search field;
- The two preparation fields (Ethnomedicinal and Scientific) have been combined;
- The two administration fields (Ethnomedicinal and Scientific) have been combined.

Note that although a number of fields are combined for searching, the displayed results will show the individual fields separately. The omitted fields will not be displayed. The program also remembers the last search request submitted and displays it as the starting point for creating a new request. This is particularly useful when the user is making successive small modifications to a search as he/she refines a search strategy.

These changes do not significantly restrict the flexibility of searching. What they do provide is a simpler interface for most users, and less chance of missing some required results. As there is only a single screen, users do not have to learn how to navigate between the two different screens, and do not have to worry about



checking both before executing their search request. It is anticipated that most users will use this option.

### *C - Formulate a Complex Search*

This option provides the two screens shown in Appendix 2 on which to create a search request. Note that the user is now given the option at the base of each screen of moving between the two screens.

Displaying the screens and interacting with the user is handled by the same program as for the previous option. It provides the ability to search on almost all the fields separately, with only Ailment Name and Synonym combined. The two search screens are necessary because of the large number of fields in the APINMAP Factual Database. In all other respects this option operates in the same way as for the simpler search. This option is provided for users who wish to be able to search on all fields using the PFBE system.

### *D - Display the Search Results*

This branches to a separate program which allows the user to display or print the search results. There are a number of noteworthy features of this program:

- This program and the program to create a search request are linked automatically. The user can select the display option directly, but they will also be routed to it automatically after creating a search request. On completing the display of results, it is possible to automatically return to modify the original request. To the user, the two programs work together as one.
- The program provides for clearly formatted screen display of search results, and correctly handles display of part or all of a record.
- The program provides sophisticated printing facilities without the user having to interact with the ISIS print module (which is not easy to use).
- The program allows the user to display the results of a search and selectively print some or all of the retrieved records.

## **2.5 Helping the user**

To further simplify working with the system, the system provides extensive user help.

When the cursor is positioned in the box labelled ?? at the top left of the screen, pressing the F1 key will display a number of help screens explaining how to complete the query screen. These help screens are stored as an external text file, and so can quickly be customised.

When the cursor is positioned in one of the data entry boxes, pressing the F1 key will provide a list of the search terms available for the field(s) associated with that box. This is another refinement that needed to be specially programmed. ISIS usually allows the user to view the contents of the Inverted File for the entire database at any time and to select search terms from this file. The option now built into the PFBE system narrows the list of terms displayed to only those that appear in the field (or fields, if the box on the screen corresponds to more than one field) currently selected. This provides the user with much more tightly targeted assistance, rather than overwhelming him/her with lots of irrelevant information.

When the cursor is positioned on the command line at the base of the screen, pressing the F1 key will display a help screen explaining what each of the command options does. After a brief training period, it is anticipated that most users will not need to reference the user documentation.

Combined together, the program for creating a search request and the program for displaying the results of the search provide a highly integrated flexible environment which allows users easily to submit search requests, modify the requests indefinitely, and display or print the results. In many ways the style of working with the system reflects the work patterns typical of online database searching. The user can browse the database, finetune the search, sample the results, and then print out only those of interest. Again, ease of use is the prime concern.

## 2.6 Explanation by example: some searches

Assume that a user wants to retrieve all records that refer to the ailment Malaria. The user is working with the single screen search option and wishes to retrieve all fields. The screen would be completed as follows:

APINMAP Factual Database: Simple Query Screen				
??	PLANT NAME/SYNONYMS	FAMILY	ENG. NAME(S)	LOCAL NAME(S)
D.				
	HABITAT	DISTRIB'N	DESCRIP'N	COMPOUNDS
	AILMENT NAME/SYNONYMS	PREPARATION	ADMINISTRATION	REFERENCES
	malaria			
F1 = Help C = Clear this Screen X = Exit Query <CR> = Execute Search				

Now assume that the user is only interested in seeing the Plant name, Preparation and Administration fields (to reduce the space required, only the lines used will be shown):

APINMAP Factual Database: Simple Query Screen				
??	PLANT NAME/SYNONYMS	FAMILY	ENG. NAME(S)	LOCAL NAME(S)
D.				
	HABITAT	DISTRIB'N	DESCRIP'N	COMPOUNDS
	AILMENT NAME/SYNONYMS	PREPARATION	ADMINISTRATION	REFERENCES
	malaria	D.	D.	
F1 = Help C = Clear this Screen X = Exit Query <CR> = Execute Search				

The user now decides that they want to narrow down the search to only those plants that grow in the Philippines and in a mountain habitat:

APINMAP Factual Database: Simple Query Screen				
??	PLANT NAME/SYNONYMS	FAMILY	ENG. NAME(S)	LOCAL NAME(S)
	D.			
HABITAT	DISTRIB'N	DESCRIP'N	COMPOUNDS	ACTIVITY
mountain\$	philippine\$			
AILMENT NAME/SYNONYMS	PREPARATION	ADMINISTRATION	REFERENCES	
malaria	D.	D.		

F1 = Help C = Clear this Screen X = Exit Query <CR> = Execute Search

As should be clear, developing and refining searches is an extremely simple process.

### 3. Conclusion

The novel aspect of the APINMAP Factual Database is not the application of ISIS to the storage of a range of information relating to medicinal aspects of plants and aromatic substances. As was explained in the introduction, ISIS is very flexible and well suited to this type of variable length data storage. What is new is the search technique provided to enable novice users to create and run quite complex searches. The examples shown above are only a subset of the complexity available from the system. The Plant Facts by Example interface enables users to search using boolean sets without being aware of it, simply by filling in examples on the screen. This interface, while powerful, does not demand high-level hardware and software to make it work. In this way, it is much more appropriate for much of the developing world. The APINMAP Factual database and the associated PFBE interface thus provide both effective data storage and a retrieval system that is oriented firmly towards the needs of the end-user in the context of what is realistic for information technology in the developing world.

## Appendix 1: Factual database FDT

### Table Definition

Tag	Name	Len	Typ	Rep	Delimiters/Pattern
1	Participating Centre Record No.	11	P		AA-99-99999
5	Contributing Institution	18	X		
10	Date of Information	7	P		9999-99
15	Plant Scientific Name	50	X		
20	Synonyms	100	X	R	
25	Family	20	X		
27	English Common Name(s)	100	X	R	
30	Local Name(s)	100	X	R	
35	Habitat	100	X		
40	Plant Distribution	500	X		
45	Botanical Distribution	1000	X		
50	Chemical Substance	500	X		
55	Active Compound(s)	100	X	R	
60	Other Compound(s)	100	X	R	
70	Biological Activity	500	X	R	
72	Standards	100	X		
75	Name of Ailment(s)	200	X	R	
80	Synonyms (Ailments)	200	X	R	
85	Ethnomedicinal Preparation	1000	X	R	
90	Scientific Preparation	1000	X	R	
95	Ethnomedicinal Administration	1000	X	R	
100	Scientific Administration	1000	X	R	
105	Raw Material(s)	500	X	R	
110	Market	500	X	R	
115	References	500	X	R	

### Notes

1. The **Len** is in bytes.
2. The **Typ** is X for ?, P for pattern-controlled information.
3. If **Rep** is R then the field can be repeated.
4. The pattern for field 1 means 2 alphanumeric characters, followed by a -, followed by two digits, followed by another -, followed by five digits.

## Appendix 2: Two screen Query

\ APINMAP Factual Database: Query Screen 1 /  
\*\*\*\*\*

??	PCRN	CONTRIB.	INST.	DATE	PLANT NAME	SYNONYMS
FAMILY		ENG. COMMON NAME(S)			LOCAL NAME(S)	HABITAT
PLANT DISTRIB.		BOTANICAL DESCRIP.		CHEMICAL SUBSTANCE		ACTIVE COMP.

N = Next Screen C = Clear this Screen X = Exit Query <CR> = Execute Search

\ APINMAP Factual Database: Query Screen 2 /  
\*\*\*\*\*

OTHER COMPOUNDS	BIOLOGICAL ACTIVITY	STANDARDS	AILMENT NAME/SYNONYM
ETHNOMEDICAL PREPARATION	SCIENTIFIC PREPARATION	RAW MAT.	MARKET
ETHNOMEDICAL ADMIN.	SCIENTIFIC ADMIN.	REFERENCES	

B = Prev. Screen C = Clear this Screen X = Exit Query <CR> = Execute Search