Paper 022-29
A Complex Query within SAS/IntrNet

John Ranson, SAS Consultant, Ottawa, Canada
Wai Ming Chan, Senior Analyst, Health Canada, Ottawa, Canada

Abstract

Application developers and analysts know that the more complex the query the more likely for inconsistent and varied results. This reporting application was built to give one version of the truth. The application improves report generation turn around time and provides a user-friendly interface to retrieve data of interest for further statistical analysis, regardless of the users level of SAS and SQL expertise. The user is able to trace originality of the data through an audit trail.

This paper presents a web application using SAS/IntrNet that allows clients to extract health and other related data. The application shows the built-in functionality that provides the flexibility in complex subsetting and dynamic groupings of diseases. The power of the application lies in the integration of the reference codebooks (International Coding of Diseases) to facilitate data retrieval.

Diagrams and flow charts will be presented to give the audience an understanding of the data repository and the logic that has been built into the application.

Introduction

This web tool is a complete data extraction and analysis online web reporting system developed by Dansys Consultants and enhanced by the business requirements of the client. This presentation will focus on the Extract / Query component of the web tool.

The objective in building this application was to provide a single entry point to health data. The complexity of the user requirements and need for detailed data have resulted in the development of a flexible query tool. The tool is built around health data but applicable for many other reporting environments.

The complexity of the average query and size of the data has made it a challenge to build a reporting application that would convince users that there was another viable option other then writing the SAS program themselves. An application would save time, improve quality, optimized queries and control the extraction of data. In addition, non-SAS users could extract data and do further analysis as required.

The application runs on Unix using SAS/IntrNet and BASE SAS as the core modules for the application. SAS/STAT, SAS/GRAPH and SAS/ACCESS are options depending on user requirements. SAS/IntrNet is set up as a pool service.

The web tool is a thin-client application with a built in security and administration system.

The User

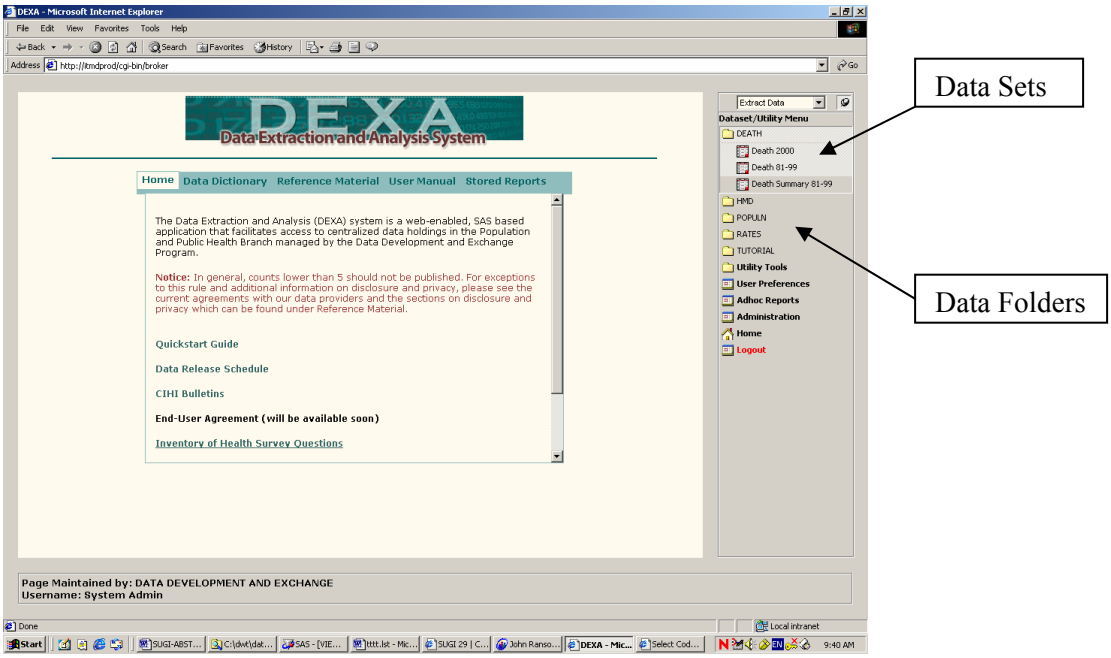
The users consist of several groups specializing in different areas of research in disease and prevention, for example, Cancer Research for Childhood Leukemia, Breast Cancer and Infectious

Disease for Sexually Transmitted Diseases. These groups all require different cuts of the same data. The client receives data from Statistics Canada, Canadian Institute for Health Information and other surveillance systems.

The users have a wide range of health-related experience and analysis experience. Users can have in-depth or no SAS knowledge.

The Data

The data is made available through the application by means of folders arranged by data area / subject. These are the data sets that have been made for the application stored and transformed in the SAS data repository.



All data sets have a similar structure with one significant variation, that being disease or diagnosis code. Demographics information is common throughout.

Data set sizes range from 10 Megs to 1 gig. Summarized data sets are available for querying. However, detailed datasets are used the most to give the user the querying flexibility and access to 'ALL THE DATA', the three scariest words when building a data warehouse or reporting application.

To understand the application and the built-in logic for this complex query, sample data are shown below. Two data layouts are shown, and the details about the integration of the international codebook for diseases are explained.

Mortality Data

Year	Sex	Age	Province	Cause of Death	Nature of Injury	Count
1997	F	80	ON	E887	8208	4
1998	M	20	ON	E8199	9598	1
1999	F	30	ON	E9530	9947	2
1996	F	70	ON	E887	8210	1
1997	F	0	BC	E912	9331	1
1997	F	40	ON	E8147	9598	1
1996	F	50	ON	E8902	986	1
1999	F	90	ON	E888	8208	3
1998	M	60	ON	E8147	8090	1
1999	F	80	ON	E8698	9878	1

Note: The values for the **Cause of Death** and **Nature of Injury** variable link to the International Coding of Disease (ICD).

Morbidity Data

Year	Sex	Age	Province	Diagnosis 01	-	Diagnosis 16	Operation 01	-	Operation 10	Accident 01	-	Accident 05
2000	M	70	QC	9964	-	V579		-			-	
2000	F	60	BC	V605	-			-			-	
2000	F	80	ON	V571	-	V498	0941	-			-	
2000	M	70	QC	7140	-	V572		-			-	
2000	F	90	ON	V578	-		0201	-			-	E8809
2000	M	20	QC	2959	-			-			-	
2000	F	80	QC	8208	-	V572		-		E888	-	
2000	F	60	QC	4341	-	V571	0276	-	0282		-	
2000	M	90	ON	V578	-	4281	0134	-	0282		-	
2000	M	60	NS	V578	-	2720	0276	-	0276		-	

Note: The values for **Diagnosis 01 to 16, Operation 01 to Operation10 and Accident 01 to 05** link to the international coding of disease (ICD).

International Coding of Disease (ICD)

International Coding of Disease is designed for the classification of morbidity and mortality information for analysis purposes and for identifying hospital records by disease and operation. Over time, new versions of ICD reference tables are released due to the expansion of disease classification.

What will be show is how the ICD codebooks are integrated into the query and used as lookup tables. The data layout for the standard codebook consists of a hierarchy of four levels of diseases. The four-digit disease code is the key with the 1st, 2nd and 3rd level codes all being generated from the four-digit code.

An example of the hierarchy is shown below. The first level is called Chapter, then level two, level three and then level four, the actual icd four level codes.

Example of Disease Hierarchy

Selected Values:
01 Infectious and parasitic diseases
042-044 Human immunodeficiency virus HIV infection
042 HIV infection with specified conditions
0420 Hiv infect w cert spec infect
0421 Hiv infect caus oth spec infect
0422 Hiv infect w spec malignant neo
0429 Aids w or no other conditions
043 HIV infection causing other specified conditions
0430 Hiv infect caus lymphadenopathy
0431 Hiv infect causing spec dis cns
0432 Hiv infect caus oth immune dis
0433 Hiv infect caus oth spec cond
0439 Aids rel complex w/no oth cond
044 Other HIV infection
0440 Hiv infect caus spec ac infect
0449 Hiv infection nos

Linking the Data and the ICD Codebooks

The variables **Cause of Death, Nature of Injury, Accident Code, Operation Code and Diagnosis Code**, as shown in the data layouts are all linked to ICD reference tables. Some of these variables use a subset of the same reference table and some data sets use more recent versions of the ICD reference tables.

For example, Chapter or the first level of the disease coding consists of 18 chapters. Chapter 18 represents all external injury codes. Therefore, when querying against **Nature of Injury**, only this chapter should be available to the user for selection. Also, many data sets that contain data for year 2000 and onward use a new version of the ICD coding scheme. These two versions of codebooks have to be synchronized with the selected data set. This flexibility is controlled by one text file read in by the application when needed. See below:

```
The following coding scheme is required for the varcid variable
1=diag1-daig16  6=accdcod1-accdcod5 7=oper1-oper10 8=injury 9=cause 10=14 11=accdcod1
```

DATASETS	VARCID	LOOKUP TABLE USED (ICD CODE BOOKS)
MORTALITY.DEDC8199	9	LKUP.ICD9CODE
MORTALITY.DEDC8199	8	LKUP.ICD9INJURY
MORTALITY.DEDC2000	9	LKUP.ICD10CODE
MORTALITY.DEDC2000	8	LKUP.ICD10INJURY
MORBIDITY.HMD2000C	1	LKUP.ICD9CODE
MORBIDITY.HMD2000C	6	LKUP.ICD9INJURY
MORBIDITY.HMD2000C	7	LKUP.CCPCODE
RATES.MORTALITY8199CAUSE	9	LKUP.ICD9CODE
RATES.MORTALITY8199INJURY	8	LKUP.ICD9INJURY
RATES.MORBIDITY9400	10	LKUP.ICD9CODE

This text file creates the relationship between data set, disease variable and ICD lookup table.

For example, **MORTALITY.DEDC8199** has two disease coding variables **cause** and **injury**. When the user queries against **Cause of Death, LKUP.ICD9CODE** will be used and when the user queries against **Nature of Injury, LKUP.ICD9INJURY** will be used. Similarly, you can see how the other data sets are linked with the respective codebook.

The Query

Now with an understanding about the data and how the International Coding of Diseases is integrated into the query / application, the remaining part of the paper will overview the interface in which the query is run.

What the query will do

- Selection of a data set
- Selection of variables
- Subset by disease code for multiple disease code variables and ICD codebooks
- Allow for the dynamic creation of disease groupings
- Build a standard where clause

One of the goals of any online reporting application is to have results in a matter of seconds. This web tool was built to do this. However, the larger the data set the more difficult this is. Can the server be upgraded? Can we optimize the data in some way? Can we optimize the application? All things considered, extracting data will take longer for the data sets that approach 1 Gig. Query time average is around 10 seconds with the most complex query against the largest table being close to the 60s.

Disease Groupings

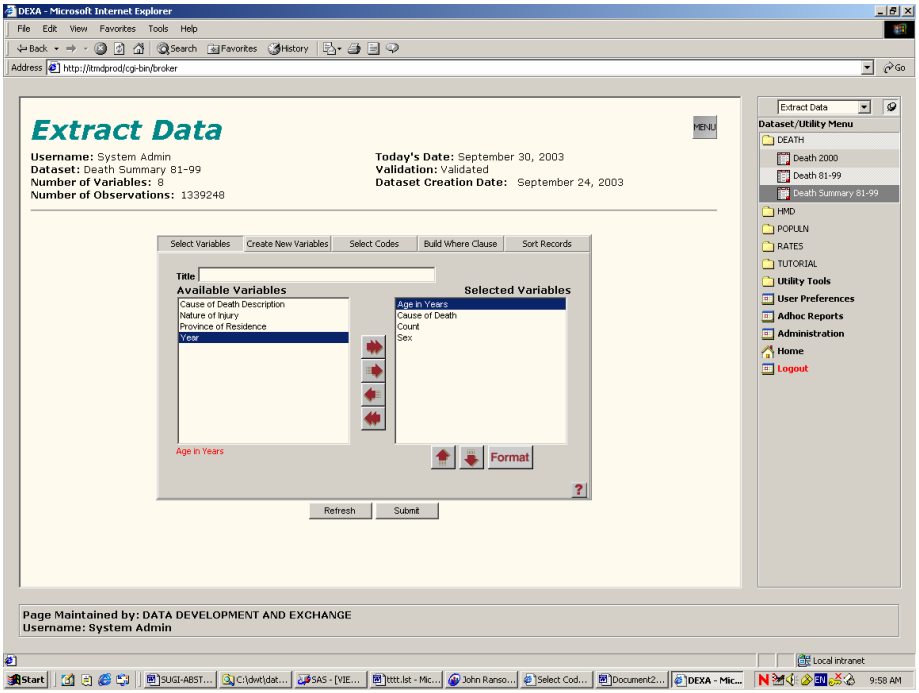
The ICD codebook attempts to create groupings for the roughly 9000 ICD codes via the codebook hierarchy. However, these groupings do not satisfy all reporting requirements.

A feature within the query component gives the user the ability to create their disease grouping and assign a group name. For example, Hepatitis A, B and C all have case definitions that a user may want to group and classify as Blood-borne diseases for reporting.

The Extraction Tool

The five tabs shown consist of all options the user has in building the query. The ***Select Codes*** tab is the main focus with some mention of the ***Select Variables*** and ***Build Where Clause*** Tab.

Select Variables screen

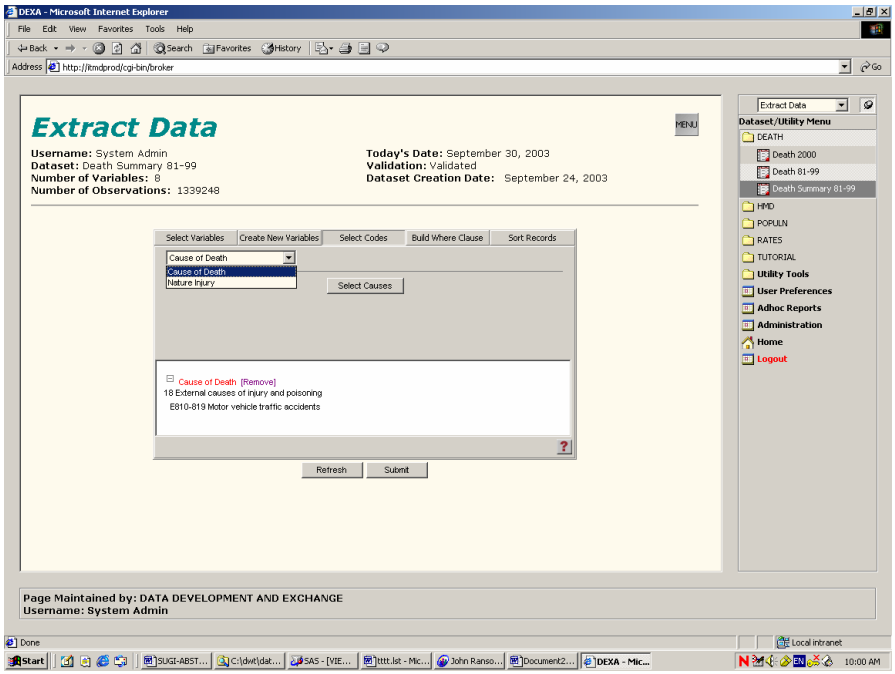


Note: Notice that the select variables tab contains a format option. This allows the user to create SAS formats and apply them to variables during extraction.

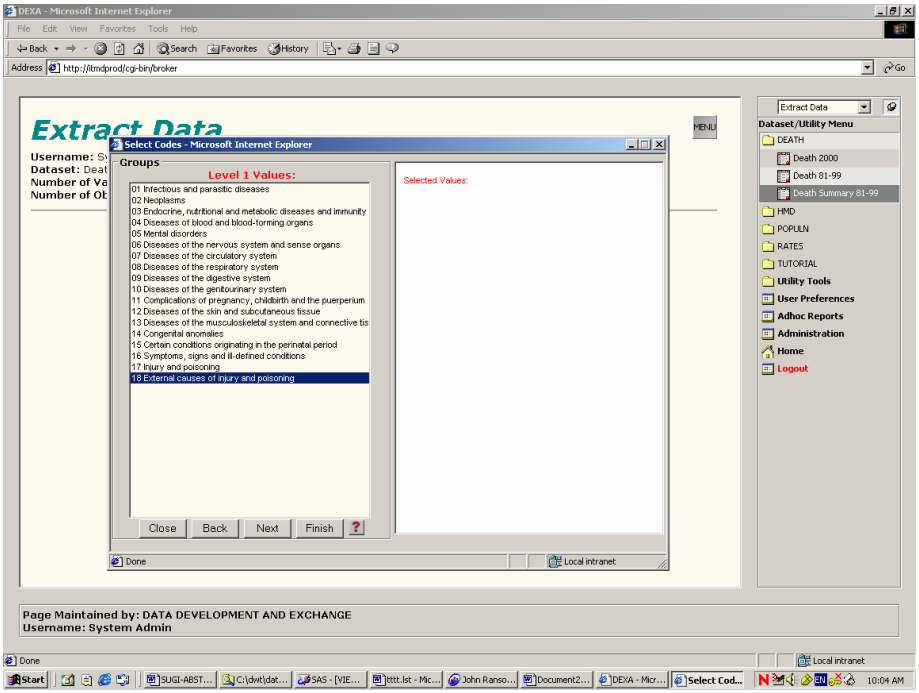
Select Codes Screen

The more complex the data set with relation to disease codes, the busier this tab becomes.

A simple view of the Select Codes Tab

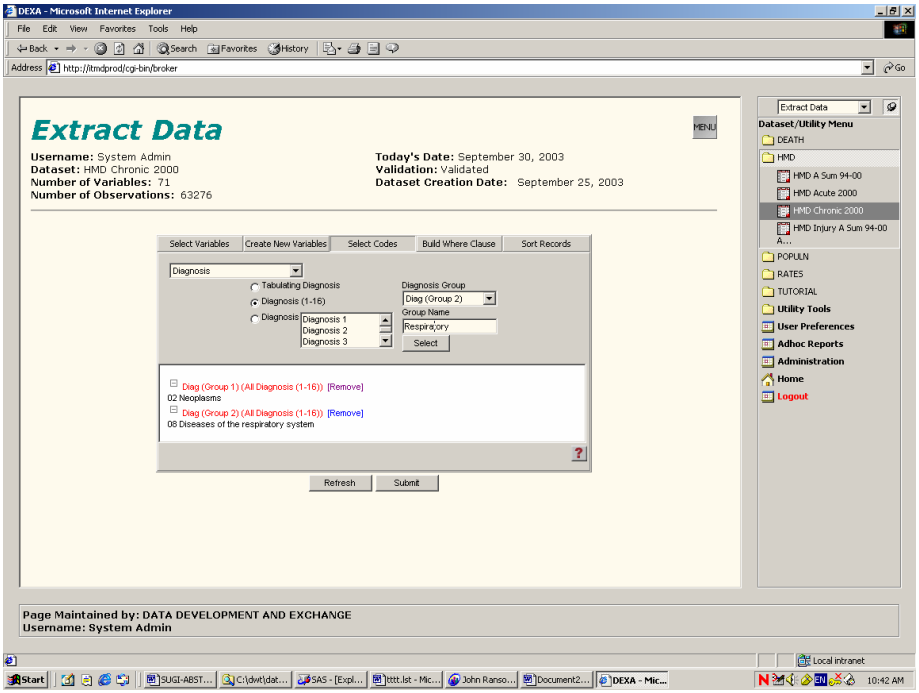


The disease hierarchy is activated when the ***Select Cause*** button is selected. The user then selects the ICD codes they want for the query. The disease hierarchy is shown below:



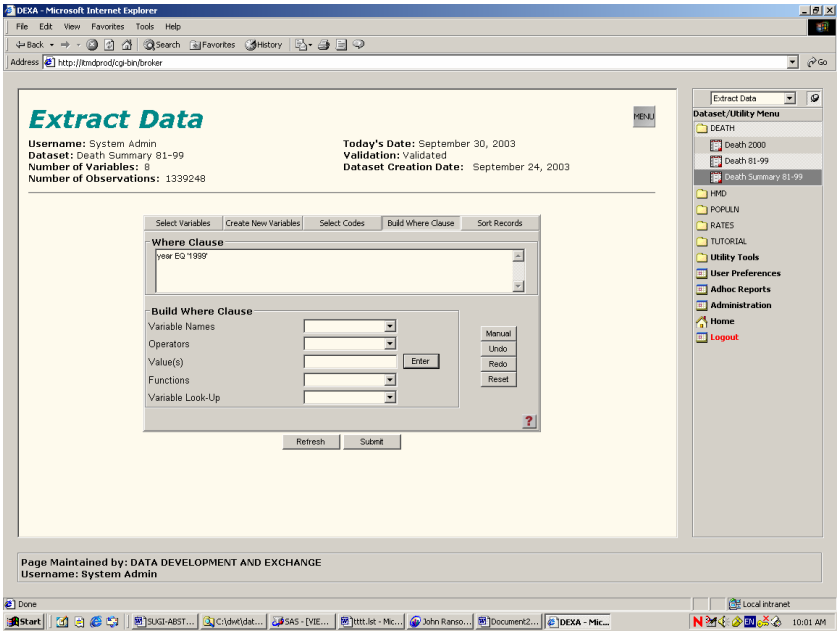
The user drills into the disease hierarchy to select the specific disease for their query. The disease descriptions are displayed along with the codes to make it more informative to the user.

A more complex ICD selection with Multiple Group Selection



The variables in the data set determine whether the grouping disease option is available. Shown above, two groupings have been created. The user also has the flexibility of which disease variables to query against.

Where Clause Screen



The Where clause is always available for standard subsetting, which can be used in conjunction with ***Select Codes***.

Conclusion

This paper shows a component of a full reporting system used for data extraction and analysis. The query component has been designed around the data in conjunction with the user requirements. The power lays in the application itself and the integration with the International Disease Codebook.

Acknowledgments