The Newsletter for SIR UK Users

SIR REPORTER

Contents

Editorial

Michael Staley

Why not visit SIR's web site at *www.sir.com.au*?

Note from the Chair

Since the last issue things have moved along at a great pace. Final details of the Conference in June are being put together, it looks like being a very interesting conference. Booking can be done

online through the web see ***http://www.soton.ac.uk/~_sir/edinburgh.html***

It has been announced that all support will now be done from Australia, by email, fax and phone. Forvus are no longer distributors and will not automatically provide support. I would like to give many thanks on behalf of all of you, to Forvus for the sterling work they have done for us all over the years and to welcome them to the user group as 'normal' users. The new arrangements will possibly give a better service especially to those with problems that have occurred late in the day - as an email to Australia gives them a day to work on it and get a reply to you before you arrive at work the next day.

Today (12/02/99) I sent an email an email to SIR in Australia at 9am and had a reply at 12:24pm. Three and a half hours in the middle of their night means they are trying.

As far as SIR itself is concerned, the UNIX versions of 4.0 have been delivered for Solaris, SCO and Linux.

Version 4.5 pre-beta has been with some of us for a month or so with some very nice new features. Multiple databases can be attached and accessed concurrently and then disconnected. There is ODBC access to and from SIR. I have imported data from SIR directly into Access, Paradox and Excel and queried a SIR database from Oracle. From SIR I have read data from Access, Paradox, Comma separated text, Oracle and dBase databases using PQL. There is an SQL server that is used to access SIR using ODBC, and this can be used to access data from SIR running on one platform from another. For example a PC version of SIR can access data in a UNIX SIR database. There is a spreadsheet view of records with update as a PQL command or as a procedure. There is also a new graphics procedure and a Comma separated file procedure along with correct handling of comma separated files through PQL read.

There is a new GUI interface from PQL rather than WDL which allows GUI interaction from PQL with such things as drag and drop, you could drop a data file on the SIR window and get SIR to load the data into your database.

Make sure you come to Edinburgh to see all this in action.

Dave Doulton

Events

16-18 June 1999 International SIR Conference, Edinburgh

Edinburgh 1999 International Conference

Draft Programme

Wednesday 16th June

14.00 – 17.30	Keynote and user papers
19.30	Dinner and ghost tour of Edinburgh

Thursday 17th June

09.00 – 17.30	Training and user papers, with parallel sessions for introductory and advanced training
19.30	Conference dinner and dance

Friday 18th June

09.00 – 12.30	User papers and tips and tricks
12.30	Lunch and close

So far we have finalised the following papers and training sessions, parallel sessions will be run so that delegates can choose between for example introductory and advanced training.

Keynote Speech		Mins
The Resurgence of SIR in the US	Nick Gawrit, SIR Americas	30
Training Sessions		
SIR 4.5	Tony Reardon, SIR Oz	60
Visual PQL	Tony Reardon & Dave Baxter	180
ODBC	Tony Reardon & Dave Baxter	30
Tailoring the SIR Environment	Tony Reardon & Dave Baxter	30
Introductory Training	Tom Shriver, ??	60
Advanced Training	Kathy Brooks, Forvus	30
SIR and the Y2K problems	Tony Reardon & Dave Baxter	30
SIRgery, The SIR Consulting Room	David Baxter	60
User Papers		
Heartbase - SIR is good for your Heart	Nick Gawrit	30
Scan your data into SIR	SIR Americas	30
Access reporting from SIR	SIR Americas	30
SIR and the DDI: can you teach an old dog new tricks?	Karen Brannen and Colin Smart, Centre for Educational Sociology	30
Using SIR as basis	Leif Spange Mortensen, ??	30

for Web based
statistical databases

Training Sessions

SIR 4.5

The large number of new features in SIR 4.5 arguably represent the biggest changes to SIR in the last 10 years. This session covers all new features including:

- A brief overview of Visual PQL and ODBC (These are covered in depth in separate sessions);
- The use of Multiple Databases and how to reference more than one database in PQL;
- The new Master Logon/Logoff capabilities;
- Using the new SIR spreadsheet interface through commands and through a procedure;
- Using the new Graph module and the updated procedures DESCRIPTIVE, GRAPH and PLOT.

VisualPQL

VisualPQL is a major new feature in 4.5 and is the name given to a set of new commands, functions and features in PQL to enable an application developer build a modern GUI style interface in PQL. This presentation/training course is in three sessions:

Session 1 Overview of VisualPQL - Introduction to all Commands and Functions - 1.5 hours

This session covers the concepts of message processing and VisualPQL. It then goes through the syntax and functionality of all VisualPQL commands and functions. The use of 'EXECUTE DBMS' is discussed.

Session 2 Building a menu, Building a dialog - 45 Minutes

This session covers the system menu, building a menu, processing menu messages and the DROPFILE message. It covers building a simple dialog directly in PQL

Session 3 Use of the DIALOG DESIGNER and DIALOG COMMANDER - 45 minutes

This session covers the use of the Designer and Commander to assist in building dialogs

ODBC

SIR 4.5 supports ODBC and can supply information to any ODBC client through the new SQL server, SirSQLs and SIR ODBC driver. The driver communicates with the SIRSQLs server and handles client ODBC requests. PQL can also use new features and functions to pass SQL commands through ODBC to get infomation from any ODBC data source or directly from the SirSQLs server without going through ODBC.

This session covers ODBC support in SIR 4.5 including installation issues and the new PQL commands.

Tailoring the SIR Environment

SIR saves a number of system settings, such as font and window size in a .INI file. This session gives details of these settings including new PQL functions to access this file.

SIR and the Y2k Problem

The year 2000 raises a number of issues and this session discusses how these may impact SIR operations with regard to specific versions of SIR and specific machines and operating systems. It includes details of any problems reported by customers as part of their Y2k testing.

User Papers

SIR and the DDI: can you teach an old dog new tricks?

Adhering to standards is one way of reducing the work involved in maintaining and exploiting the data which has been collected by an organisation. Recent work by the Essex Data Archive and others has lead to a draft of a framework for data documentation. This work goes under the heading of the Data Documentation Initiative (DDI). The framework is expressed in a meta-language known as XML (Extensible Meta-Language). XML bridges the gap between HTML (which is too simple and restrictive) and SGML (which is too complex and open) and provides the user with a simple, but flexible, way of defining the structure of data. The advantage of using a standard meta-language is that a number of tools will be available to manipulate the contents of an XML document and, hopefully, increase the ease with which the user can transfer their data and metadata between packages.

SIR has always been good at storing metadata (even before the term was acknowledged). While much has been done to ease the transfer of data between systems, the difficulty, in the past, has come in transferring that metadata to other users (or to a data archive). In this paper we show how the facilities within SIR can be exploited to create structured metadata which complies with an evolving standard for dataset description.

Karen Brannen and Colin Smart

Using SIR as basis for Web based statistical databases

As hardcore SIR users will know, the functional characteristics of SIR/DBMS and SIR/FORMS for building statistical databases and for entering and validating statistical data are second to none. If you want the same functionality from a Web based system you will have to do a lot more work, but here SIR can be of great help: It will be shown how use of SIR's schema functions can transfer a database defined in SIR into a default Web based form that, among others, includes range checks, and into a database definition for an SQL engine.

Leif Spange Mortensen

SIR Version 4.5

We are really excited about the upcoming release of 4.5 and think that new features such as Visual PQL, ODBC and multiple database support together with the new Graphics and Spreadsheet arguably represent the biggest changes to SIR in the last 10 years.

VisualPQL

One thing that you will notice in 4.5 straightaway is the new 'look and feel' of the product. All the menus and dialogs have been re-written from 4.0 and while these have lots of improvements in themselves, the biggest change is that they have all been written using the new Visual PQL.

The Visual PQL system starts with the main PQL program which defines the main window and the complete menu system. This program receives control when the user selects a lowest level menu item. It can deal directly with the requested function, call sub-routines, use sub-procedures or any PQL construct and can call other PQL programs and DBMS functions. The program can enable, disable, check or uncheck menu items as necessary. The program can also contain a *DROPFILE* processing block. This is given control if the user 'drags' a file and 'drops' it into the main window. The routine is given the filename and then decides how it is to be processed.

Once the system is running, any PQL program can output information into the main window (such as title and status) and can put text in the window. Programs can also save, print or clear the main window.

PQL programs can display and get information through dialogs. Any program can now create and run with dialogs and a program which creates a dialog remains active until it issues an *EXIT* command. A dialog is created by a program with a *DIALOG* block. Within this there are commands to define controls such as *BUTTON*, *LIST, RADIO, TEXT* which make up the dialog. Processing logic is contained in a series of *MESSAGE* blocks corresponding to the type of control. The logic in a *MESSAGE* block takes some appropriate action which may involve running further programs or invoking a further dialog which itself contains further message processing. The message processing blocks can contain any logic. *NEXT [MESSAGE]* returns control to the user and *EXIT [MESSAGE]* exits the dialog.

There are about 16 new commands and 25 functions to interact with the dialog which allow you to do things like inserting new lines into a list or finding the position of an item.

A key new feature is the *EXECUTE DBMS 'command_text'* which suspends the operation of the program that is executing and allows the execution of any other SIR command including running other PQL programs; these in turn may issue further *EXECUTE DBMS* commands building up a hierarchy of suspended programs. Control returns to the program when the input processing has finished and the suspended program then resumes execution at the next instruction.

Visual PQL also includes commands which directly pop-up boxes to which the user must respond and can be used to display an error message or to ask for an OK or Cancel response. It also has commands to display a file browse box appropriate to the operating system and to print files displaying a Print box to alter print specifications as necessary.

To assist with creating new Visual PQL dialogs, there are two new features called the *Dialog Designer* and the *Dialog Commander* respectively. These can be invoked from the new system menus and give a developer an interactive means of creating dialogs and of generating appropriate message processing blocks. The Designer helps create the physical appearance of the dialog and the Commander helps tie this to appropriate processing logic.

The complete source code for the user interface is supplied with the system (about 20,000 lines of PQL) and the menus and dialogs can be modified for specific environments or used as examples for application development.

ODBC

SIR version 4.5 also supports **Open Database Connectivity (ODBC)** for cross communication with other software packages

This allows SIR data to be accessed by packages such as Microsoft's Excel and Access and also allows PQL to get data from any ODBC source. ODBC is an SQL based standard i.e. data retrieval commands are expressed in SQL. In order to service requests from other software packages there are two new ODBC components:

The first is the SIR SQL server. This runs on the network and responds to applications using ODBC to access SIR data sources or any PQL programs using the new client/server features. The SIR SQL server runs under the various operating systems supported by SIR (Windows 95/98, NT, Unix, OpenVMS) and SIR handles all issues of cross machine compatibility. You can run multiple servers on a network if you need to for performance or security.

The second is the SIR ODBC driver. This communicates with the new server and handles client ODBC requests from third party software or from applications you write using ODBC. This has been tested with the Microsoft products ACCESS, QUERY and EXCEL and with MINITAB. It has been used for retrieving data by these products. The SIR ODBC driver can be installed on a machine which is not running SIR, however ODBC client services are only available on Windows.

There are new commands in PQL which allow programs to access any ODBC data sources using ODBC or directly access the new SIRSQL Server without going through ODBC. These allow a PQL program to establish a connection to a data source, pass the text of an SQL query and execute this. The program can then enquire as to the columns and rows available from the query and can get data from each column, stepping through the available rows.

Multiple Databases

A new feature was introduced in 4.0 so that you could run the system without a database and connect and disconnect to a single database as required. Now, in 4.5, you can have multiple databases connected at one time with the ability to switch between them and you can reference more than one database in a PQL program.

The system now has a table of connected databases one of which may be the current default database. By default, the last database connected is the default and is also the default procfile.

There are new PQL commands to connect and disconnect databases and to set the default. A program or retrieval can access a specified database with a new block construct *DATABASE IS*. Inside this block, all references are to variables in the new database. You can use any standard PQL commands in this block. When the block is exited, if there was an original database, it is made current.

Functions

The PQL functions which deal with database schema structures now operate in a *PROGRAM* as well as a *RETRIEVAL* and return information about the current default database. About forty new functions and twenty-seven new *SYSTEM(n)* functions have been added to return information about various aspects of the system. These include functions such as *CURDIR* which returns the current directory, *DBNAME* which returns the name of a database and *SYSTEM(40)* which returns the number of connected databases.

There are a couple of new DBMS commands such as *SHOW DATABASE*, which writes a viewable list of connected databases.

Procedures

There are a number of new and changed PQL Procedures in 4.5.

There is a New **CSV** Procedure (Comma Separated Variable file) plus support for reading CSV files.

The *CSV SAVE FILE* procedure writes a text file with the individual fields separated by commas. Numerous software packages read and/or write data in this format. Each record is on a single line in the file with up to 128 individual fields of variable length separated by commas. String values are enclosed in double quotes. Floating point values have decimal places as defined in the schema or as sufficient if not defined. Dates and times are formatted according to their maps. Strings are written for categorical variables.

The *READ* in PQL can now read files in this format. On the *READ* command, the free field input formats (represented by *) now take each comma as a field separator which is one field on the read input format. Two commas in the record result in blank input to that field.

There is a new SIR **spreadsheet** procedure plus a direct spreadsheet interface. This new feature allows you to examine and save output as a spreadsheet. The direct spreadsheet interface is used through a new DBMS command on database records or tables and gives retrieval and update capabilities if required. The spreadsheet itself can be saved, exported to Excel or printed. (N.B. The spreadsheet is only available on Windows systems)

There is a new **Graph** module which is used with a number of updated procedures. The SIRGRAPH module produces 35 styles of graph display including two-dimensional and three

dimensional options. It has a built in wizard to assist in making specific graphs and a wide variety of formatting options including saving and printing. SIRGRAPH is run as a stand-alone program (use ESCAPE 'SIRGRAPH sirxxxxx.srg to run from within SIR) and produces graphs from an input file. This file is a standard text file and there are three new procedures which produce SIRGRAPH files. SIRGRAPH is only available on the Windows platforms, however the procedures can be executed on any SIR platform and the files transferred to a Windows platform for graphing if necessary.

The new procedures are:

DESCRIPTIVE which analyses a single, numeric variable and produces descriptive statistics plus a frequency bar-chart. The statistics produced are a Weighted Count, Count of any missing observations, Mean, Standard Deviation, Variance, Coefficient of Variability, Skewness, Kurtosis, Minimum, 1st Quartile, Median, 3rd Quartile, Maximum and 95% confidence interval of mean. A frequency chart is produced. The default number of categories is the square root of the number of observations with a minimum of 7. The default interval and range are calculated depending on the values of the observations or can be specified as required. This procedure essentially combines the analysis done by the CONDESCRIPTIVE and FREQUENCIES procedures but these procedures are still available in there original form if required.

GRAPH which analyses an observation variable against either one or two categorical variables and produces either a 2D or 3D graph. It can also analyse multiple numeric variables against a single categorical variable as multiple series on a 2D graph. It reports summary results by category determined by the categorical variables. The summary of the observation variable may be counts, sums, averages, maximums, minimums or percentages by row, column or in total. This updates and replaces the old GRAPH procedure.

PLOT which analyses two numeric variables and produces a scatter plot or line plot together with correlation statistics. The statistics produced are the Mean and Standard Deviation of both variables plus the correlation (R), R squared, Student's T test, Degrees of Freedom (DF) and the probability of the significance of R. This updates and replaces the old PLOT procedure.

Other new features in 4.5 include:

Rewritten memory and table management so that there are now no restrictions on the size of a PQL program and no need to specify table space. Memory is allocated automatically as needed.

There are two new keywords on the *OPEN* command which are *BINARY* and *APPEND*. Binary files have no record structure and data is read from or written to the file as a stream of bytes corresponding to the format on the read or write. Opening an output file as *APPEND* means that data is added to the end of the file if it already exists. This can be done for text and binary files.

The system now supports wide output to the scrolled output window or default output file. The *PAGESIZE* command sets the default output width. For files, this can be any desired size. For the scrolled output window, this can be up to 4k, however lines longer than 1k wrap under Windows.

We will shortly be shipping beta releases of 4.5 on Windows and on Unix (simultaneous releases - another new feature!) and hope you take the opportunity to try it for yourselves. Please contact us at ***david@sir.com.au*** to participate in the test program.

Tony Reardon

Y2K

There is much concern and worry about the impact of the year 2000 on computer systems and we thought it appropriate to mention how this impacts SIR applications.

Within SIR, if a field is defined as a date, internally it is held as a count of days (Julian format) since the start of the Gregorian calendar (taken to be October 15th 1582). Conversions between this number and any other date representation happen when data is entered and when it is displayed.

This means that any dates already in a database are OK and there is no need for concerns about database layouts, etc. There are however issues when going between external dates and internal formats.

If an application has external date formats with only two digits for the year, the system has to 'guess' what century is meant. Before version 4.0, SIR's guess was that the century was what the operating system said the century was today. In other words, until Dec 31 1999, the century was 19.. and after that it should be 20... However, because of the TODAY(0) issue (see below), in every version we have tested, today is taken to be 19.. regardless as to the system date setting. This means that utilities wh …..***this bit got chopped again***

From 4.0 onwards, there is a bit more intelligence applied. If a year only has two digits, then it is assumed to be in a specific hundred year period. The default period is 1920 to 2019 so if '87' is entered it is taken to mean 1987 and if '12' is entered, it is taken to be 2012. The default 100 year period can be changed by parameter (CENY) at run time which means that you can set it to be the best fit for your application.

The algorithm which converts between external dates and the internal julian format must take account of leap years and this is handled specifically for given ranges of years. The practical effect of this is that the upper date correctly handled is limited. In 4.0 the upper date is Feb 28th 3000; in earlier versions it was Dec 31 2299.

So where might you hit problems with SIR applications after Jan 1 2000? If you are using 4.0 or later, we don't know of any problems. If using earlier versions, then there are some issues:

The SIR versions before 4.0 give problems with the functions that return today's date i.e. TODAY(0) in DBMS, Forms and SQL and the DATET function in DBMS and Forms. In most versions, including OpenVMS, Unix and the PC extended memory versions, it always returns 19 as the century (even if it is after the year 2000). This means that all 2 digit years will be guessed to be 19.., since that is what the system believes it is. This will give special problems on Feb 29 2000 since 2000 is being treated as 1900 and 1900 was not a leap year. ***Is there any missing here Tony?***

In the non extended memory version of SIR/PC, if the system date is after 1999, it returns -1 as the julian date!

Our strong advice to everyone is to upgrade to SIR version 4.0 or later if you have any concerns at all.

If you cannot change to 4.0 or later for some reason, what can you do? First, you should change all external date formats to four digit years. This will avoid any problems with entering new dates. If you use the TODAY(0) and DATET functions, we have a couple of small PQL sub-routines (and similar code for Forms) which return the correct date on those systems which return 19 as the century. You would need to alter calls to the old functions to use these. If you want further details, please contact our tech….. ***truncated again!***

One final possible nightmare. A few years ago, a CDC-based version of SIR stopped working because some converter had built in a date test and that date had been reached. This was very machine specific and not general practice but it is conceivable that some very old versions might have this type of problem.

So everyone should test their specific applications by setting the operating system date to 2000 and establishing exactly what happens.

Tony Reardon

Profile

Frances Williams is our newest member of the UK User Group Committee.

I have been a SIR user for the past six years, ever since I started working for the Institute for Social and Economic Research (ISER) based at Essex University, formerly known as the Interdisciplinary Research Centre for Micro Social Change in Britain (IRC), formerly known as the British Household Panel Study (BHPS).

The BHPS is a long-term panel study of around 10,000 individuals who are interviewed every year on topics such as demographics, health and welfare, employment, employment history, values and opinions and finance. We employ an agency, NOP, to interview our panel members and punch the data - we are still paper based. I am responsible for developing the SIR database of over 1500 variables into which the data is loaded and cleaning the data. Cleaning consists of assessing every variable and thus results in large amounts of PQL.

We still use SIR Version 2.2, partly because our database is developed under Unix but mainly because the amount of available compiler space was effectively reduced with the introduction of SIR version 3 and much of my PQL code only just compiles - roll on the Unix version of SIR 4.5.

I have 2 teenage children, and when I am not ferrying them backwards and forwards to one thing or another, I enjoy gardening and hill walking which can be somewhat of a problem as I live in the wilds of Essex which is not renowned for its mountains!

I'm looking forward to serving on the SIR User Group Committee and hope that it continues to be as effective in the future as it has been in the past.

Fran Williams

Forvus News

Steve Flack

Frequently Asked Questions

A tip for SIR 4.0 batch

For sirdbmsb on UNIX, and sirdbms in= on UNIX and PC as well.

Make sure that END TASK is the command before any others that access a database but after having specified the database. If you do not the commands in the input file will be executed with no database attached, this is probably not what you want if you are using a database.

Dave Doulton

How can I calculate an age in Years accurately?

This is more complex than just *aint((today(0)-birthday)/365.25)* as that is only approximate for leap years. The following program gets it exactly right:

```
| Calculate exact age from data of birth
| Age as at 31/8/97

program
compute INDOB = '31/08/80' | Should be 17,
   not 16
write INDOB 1x  [ aint ((
   cdate('31/08/97','DDiMMiYY')
   cdate(INDOB,'DDiMMiYY'))/365.25)] (i3) 1x

compute X_Year = numbr(sbst(INDOB,7,2))
compute AGE = 97 - X_YEAR
if (cdate(INDOB,'ddimmiyy') gt
   cdate(('31/08/'+format(X_Year)),'ddimmiyy'
   ))    AGE = AGE - 1
write AGE


| Count the leap years between DOB and cutoff
compute X_Year = numbr(sbst(INDOB,7,2))
set X_LEAP (0)
for X_IND = X_YEAR,97
. if (exists(
   cdate(('29/02/'+format(X_IND)),'ddimmiyy')
   )) X_LEAP = X_LEAP + 1
end for
write X_YEAR X_LEAP

| Is INDOB before 29 Feb in birth year ?
ifthen (cdate(INDOB,'ddimmiyy') le
   cdate(('28/02/'+format(X_Year)),'ddimmiyy'
   ))  write 'before 28 feb'
else
. write 'after 28 feb'
| subtract a leap year
. compute x_leap = x_leap - 1
end if

compute age = (cdate('31/08/97','ddimmiyy') -
   cdate(INDOB,'ddimmiyy') - x_leap) / 365
write INDOB age [aint(age)]
end program
```

Kathy Brooks

SIR UK User Group Committee Members 1998/99

Randy Banks
British Household Panel Survey
University of Essex
Wivenhoe Park
Colchester CO4 3SQ
Tel: 01206 873067
Fax: 01206 873151
email: randy@essex.ac.uk

Hilary Beedham
Economic and Social Research Council
Essex University
Wivenhoe Park
Colchester CO4 3SQ
Tel: 01206 872570
Fax: 01206 872003
email: beedh@essex.ac.uk

Karen Brannen
Centre for Educational Sociology
University of Edinburgh
7 Buccleugh Place
Edinburgh EH8 9LW
Tel: 0131 650 4203
Fax: 0131 668 3263
email: K.Brannen@ed.ac.uk

Kathy Brooks
Forvus
53 Clapham Common
South Side
London SW4 9BX
Tel: 0171 498 2602
Fax: 0171 498 1939
email: kathy@forvus.co.uk

Dave Doulton
University of Southampton
Computing Services
Highfield
Southampton SO17 1BJ
Tel: 01703 593541
Fax: 01703 593939
email: D.C.Doulton@soton.ac.uk

Dave Greatorex/Lisa Zaninetti
BBC, C307
Woodlands
80 Wood Lane
London W12 0TT
Tel: 0181 576 7060/
0181 743 8000 ext 62871
Fax: 0181 743 0906
email: Dave.Greatorex@bbc.co.uk
email: Lisa.Zaninetti@bbc.co.uk

Charlie Owen
Thomas Coram Research Unit
Institute of Education
27 Woburn Square
London WC1H 0AA
Tel: 0171 612 6942
Fax: 0171 612 6927
email: Owen@ioe.ac.uk

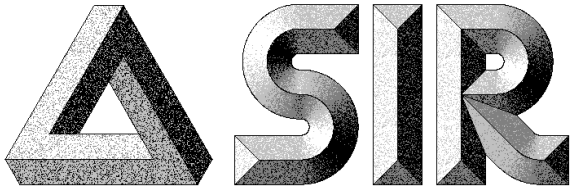
Peter Ritchie
Institute of Occupational Medicine
8 Roxburgh Place
Edinburgh EH8 9SU
Tel: 0131 667 5131
Fax: 0131 667 0136
email: PJ_Ritchie@iomhq.org.uk

Michael Staley
Office for National Statistics
1 Drummond Gate
London SW1V 2QQ
Tel: 0171 533 5346
Fax: 0171 533 5300
email: Michael.Staley@ons.gov.uk

Jenny Simpson
Department for Education and Employment
Mowden Hall
Staindrop Road
Darlington DL3 9BG
Tel: 01325 392 690
Fax: 01325 392989
email: jenny.simpson@dfee.gov.uk

Frances Williams
British Household Panel Survey
University of Essex
Wivenhoe Park
Colchester CO4 3SQ
Tel: 01206 873568
Fax: 01206 873151
email: fwill@essex.ac.uk

You can email the whole committee by addressing your message to **sug.comm@essex.ac.uk**



International SIR Users Conference
June 16-18, 1999
Edinburgh

Alter to a proper booking form

Organisation _____

Address _____

_____

_____

Contact name _____

Phone _____

Fax _____

Email _____

	I would like to register for the Conference
	I would like more details about the Conference
	I am interested in presenting a paper/tip & trick
	I am interested in display space for demonstrations/posters/other

Please send your completed forms to:

Kathy Brooks, Forvus House, 53 Clapham Common South Side,
London SW4 9BX, UK
Tel: +44 (0)171 498 2602 Fax: +44 (0)171 498 1939 Email: kathy@forvus.co.uk