

Converting From Earlier Versions

SIR Advanced Training New York 2005

David Baxter SIR

Introduction

This session will cover some of the issues faced when upgrading from an old version of sir to a more current one. In particular upgrading from console based versions before SIR 4.0 to windows based versions. It will also look at some strategies that will make future upgrading easier.

Databases

Databases are easily converted with Export/Import. Export from your old version and import with the new.

PQL Retrievals

Most non-fullscreen PQL programs will run on newer releases. Any stored executables and subroutines will need to be compiled.

Problems can occur with implied computes. An implied compute is one that looks like:

variable = expression

When this is being compiled, the SIR compiler recognizes it as an implied compute because the command starts with a word that is not a PQL or DBMS command word. You cannot have, for example:

$C = C + 1$

Because C is a DBMS Comment.

With new versions of SIR come new commands and keywords so what was an implied compute in one version may not be in future versions.

MESSAGE = "Hello World"

Will not compile in SIR2000 or beyond because MESSAGE is a PQL command word.

The solution is to use an implicit compute:

COMPUTE C = C + 1

COMPUTE MESSAGE = "Hello World"

Note Implied computes can occur in multipart commands:

IF (A EQ B) C = C + 1

COMPUTE MESSAGE = "Hello World"; C = C + 1

Fullscreen PQL programs

Fullscreen PQL Programs will still run in SIR2000/SIR2002 in the execution window (implemented especially for this) but they will not look or feel as nice as GUI programs).

Because of the inversion of logic between DOS and Windows, it is not trivial to write a conversion program for Fullscreen PQL. However the logic is much simpler in windows and the operating system takes a lot of the responsibility from the programmer.

ICE procedures

Most ICE commands are now either DBMS commands, redundant or obsolete. When converting an ICE command procedure to DBMS the clause delimiters need to be changed from \$ to /.

Some ICE commands conflict with PQL and so the DBMS command equivalent may be slightly different. For example ICE's **COMPUTE *global* = *expression*** is now DBMS **GCOMPUTE *global* = *expression***

SIRForms

SIRForms will continue to work in SIR2000/SIR2002 but will not have the look and feel of a windows application.

PQLForms is a GUI based data entry system based on SIRForms and VisualPQL.

There is a conversion program that will attempt to convert SIRForms to PQLForms but you will find that the output may need some tweaking. At the very least it will give you a good starting point for rewriting an old style form.

Be aware that there is some difference in the way logical expressions are handled in SIRForms and PQL. PQL expands IF (A EQ B OR C) to IF ((A EQ B) OR (A EQ C)) but forms do not.

User Menus (3.x)

User Menus written in 3.2 can be converted to SIR200x PQL menu programs using a conversion program. First the menu commands must be formatted nicely for the conversion program. DBMS3.2 will write a user menu definition to a text member in this nice format. In SIR32, run the user menu, Press F2, select Write and save as a member. Edit this member and press F2, select Save, to a file. This file will be in the correct format.

After conversion the menu may require some tidying up. In sir3.2 it was common to have an executable command in the top menu (e.g. Command and Quit) whereas this is discouraged under windows. The conversion program converts these top level commands to pull down menus with the executable command repeated as the only item in the list.

Help Tabfiles

In SIR 3.2 the help documentation was stored in specially formatted tabfiles. Users could create their own help tabfiles and produce application specific help.

From SIR4.0 SIR started using an HTML based help system – as this was common across all supported platforms. You can convert Tabfiles to HTML using PQL to navigate through the tabfile tree and write out the pages as HTML.

An example PQL conversion program is included.

Tips

Keep Exports of your old databases when archiving.

Always use explicit COMPUTE.

Maintain a makelist for subroutines and executables.

- e.g. SYSPROC.SYSTEM.COMPILE
CALL MENU.ABOUT
CALL MENU.ATTRIBS
CALL MENU.BUFFER2
....

Summary

Upgrading can appear daunting at first. Export/Import makes the first step much easier. The existence of the execution window means that most parts of your application are usable and you can produce more modern versions without stopping production.

Example Programs

The \Advanced Training\Converting From Earlier Versions\ directory contains some example programs:

FORM2PQL.pql and F2P.pql	SIRForms to PQLForms Conversion program
FANCYFOR.CMP FORM1.CMP	SIRForms examples
FANCYFOR.pql FORM1.pql	Forms examples converted
CONVERT_HELP.pql	HELP Tabfile conversion program
HELP.H32	Help tabfile
HELP\	Help tabfile converted
CONVERT_MENU.pql	3.2 User Menu conversion program
MENU32.PQL	3.2 User Menu example
MENU32.srm	Menu converted
LUNCH1.PQL	Fullscreen PQL example
LUNCH2.pql	GUI based version