SPIRE

SUBJECT: SPIRE QLA User Guide

PREPARED BY: Tanya Lim
Helen Bright

DOCUMENT No: SPIRE-RAL-DOC-001428

ISSUE: 0.1 Date: 27th November 2002

APPROVED BY:

DISTRIBUTION

Alain Abergel	IAS
Peter Ade	Cardiff
Jean-Paul Baluteau	Marseilles
Helen Bright	Imperial
Dave Clements	Imperial
Pierre Cox	IAS
Anna DiGiorgio	IFSI
Matt Fox	Imperial
Alberto Franceschini	Padova
Ken Ganaga	IPAC
Rene Gastaud	Saclay
Walter Gear	Cardiff
Matthew Graham	Imperial
Matt Griffin	Cardiff
Steve Guest	RAL
Peter Hamilton	Cardiff
Maohai Huang	Trieste
Sergio Molinari	IFSI
Ken King	RAL
Tanya Lim	RAL
Judy Long	RAL
Seb Oliver	Sussex
Mat Page	UCL
Paolo Saraceno	IFSI
Mark Sauvage	Saclay
Bernhard Schulz	IPAC
Sunil Sidher	RAL
Jason Stevens	UCL
Bruce Swinyard	RAL
Toshinobu Takagi	Imperial
Laurent Vigroux	Saclay
Gillian Wright	Edinburgh

CHANGE RECORD

ISSUE	DATE	Changes
0.1	27th November 2002	First Version

TABLE OF CONTENTS

1 Introduction 8
1.1 Scope 8
1.2 Structure of Document 8
1.3 Documents 8
1.3.1 Applicable Documents 8
1.3.2 Reference Documents 8
1.4 Conventions used in this document 9
1.5 Instructions for Problem Reporting 9

2 Setting up the QLA Session 10
2.1 Starting QLA 10
2.1.1 The Process Selection Menu 10
2.2 Selecting Input Data 11
2.2.1 Data Source 11
2.2.1.1 Local and Remote Databases 11
2.2.1.2 EGSE Router 12
2.2.2 Data Type 12
2.2.3 Time 12

3 Packet Dumps 12

4 Displaying Parameters 12
4.1 Parameter Selection 13
4.1.1 Selecting Bolometers From Array Images 13
4.1.2 Selecting All Parameters 14
4.2 Creating Displays 15
4.3 Clock Displays 16
4.4 Scrolling Lists 16
4.5 Time Series Plots 17
4.5.1 Compress 17
4.5.2 Follow 17
4.5.3 Fix 18
4.5.4 Other Uses of the Mouse 18

5 Image Displays 18

6 Adjusting Playback Speed 19

7 Saving Parameter Values to an ASCII File 20

8 Using the Console 21
8.1 Imports 21
8.2 Starting and Using Processes 21
8.2.1 Starting Packet Dump 21
8.3 Printing the Values of a Parameter 22
8.4 Saving Parameter Values to an ASCII File 22

APPENDIX A– THE ENGINEERING SIMULATOR 23

FIGURES

Figure 1: The SPIRE Quick Look Analysis window 10
Figure 2: Selecting the play button on the PacketReceiver 11
Figure 3: Selecting the data source from the PacketReceiver 11
Figure 4: Selecting parameters using a bolometer array display 14
Figure 5: Selecting parameters individually via the Parameters dropdown menu 15
Figure 6: Parameter Displayer windows for the TCREN parameter 16
Figure 7: Using the popup menu to select locking and unlocking of scroll bars 17
Figure 8: Imager window showing the Short PHOT array 19
Figure 9: The PacketReceiver menu in playback mode 20
Figure 10: The Save data and Data export to file windows 22
Figure 11: The Spire Engineering Simulator window 23

GLOSSARY

AVM	AVionics Model
DCU	Detector Control Unit
DPU	Digital Processing Unit
EGSE	Electrical Ground Support Equipment
MCU	Mechanism Control Unit
OBS	On-Board Software
QLA	Quick Look Analysis
SCU	Sub-system Control Unit
SID	Structure IDentifier
TFCS	Test Facility Control System

1 Introduction

This document is a user guide for *SPIRE Quick Look Analysis system* (*QLA*). This first version of the document describes use of the first version of *QLA* produced for the AVM model. For the AVM model *QLA* only has the basic functionality of:

- accessing and displaying parameters in a timeline
- packet dumping
- imaging displays of raw data

These functionalities can either be accessed via Graphical User Interfaces (GUIs) or via the *QLA* console. This guide focuses mainly on the former method, although the latter is briefly introduced.

1.1 Scope

This document describes basic use of *QLA*. It assumes that the required software to run *QLA* is fully installed and all environment variables are correctly set, and therefore it does not act as an installation guide. For installation instructions see the release notes or contact your system administrator. *QLA* is written in Java and can run on any operating system. This document only describes the use of *QLA* by normal users and mainly concerns itself with operating the GUIs, although a basic introduction to the console is also given. A normal user is defined as someone accessing *QLA* functionalities via the GUIs alone. Advanced users who wish to have direct access to the *QLA* public methods via the *QLA* console can get a description of this functionality directly from the Javadoc which should be available with the installation.

1.2 Structure of Document

Section 2, "Setting up the *QLA* session" is important to all users and should be read first. The following sections can then be accessed in any order as they are self-contained and describe different aspects of *QLA*. If *QLA* is being used with the engineering simulator, then Appendix A, "The Engineering Simulator" should be read and the engineering simulator started before the *QLA* session is set up.

1.3 Documents

1.3.1 Applicable Documents

SPIRE QLA Release Notes

1.3.2 Reference Documents

RD1	SPIRE Data ICD	SPIRE-RAL-PRJ-001078

1.4 Conventions used in this Document

Times New Roman Italic is used for file names, file paths, package names, variable names and application names.
`Courier bold` is used for user input at command lines.
`Courier plain` is used for user input in files.
Times New Roman Bold is used for window names, button names, menu names and tab names.
'Single quotes' are used for dropdown menu selections.

Where sections of file paths vary according to the installation location, an ellipsis is used, e.g. */.../herschel/spire/qla*. This document illustrates file paths using Linux file separators: Windows users will have to substitute "/"s with "\"s.

> is used to denote a command prompt. The commands given in this document are the same for Windows and Linux users.

1.5 Instructions for Reporting Problems

Problems detected in the *QLA* should be reported using the SPIRE SPR/SCR reporting system. This can be found at:

http://astro.esa.int/herschel_webapps/servletsuite/ProblemReportServlet?area=spire

Errors detected in this document should be reported to the author at helen.bright@ic.ac.uk

2 Setting up the *QLA* Session

The *QLA* can be run with two look and feel set-ups, defined in a file named *QLA.defaults* (located at */.../hercshel/spire/qla/QLA.defaults*), using the `qla.lookandfeel` value.

The multiple window set-up is the default setting. Select which setting you prefer by placing a comment symbol (#) in front of the setting you do not want.

`qla.lookandfeel=multiple`	the *QLA* windows will appear as separate windows
`qla.lookandfeel=single`	the *QLA* windows will appear as windows in a single frame

Screen shots in this document are taken using the multiple windows set-up.

2.1 Starting *QLA*

QLA can be started from the command prompt (in any directory) by typing:

```
> java herschel.spire.qla.QLA
```

This brings up two windows. **SPIRE Quick Look Analysis** is the main *QLA* window (see Figure 1), which is used to select *QLA* processes. If it is closed it ends the *QLA* session is ended. The **PacketReceiver** window is described below (section 2.1.1).

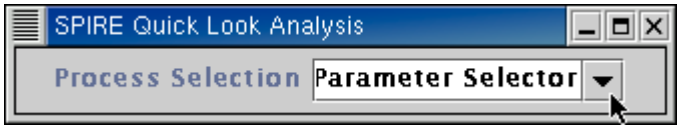

Figure 1: The **SPIRE Quick Look Analysis** window

2.1.1 The Process Selection Menu

The **Process Selection** menu on the **SPIRE Quick Look Analysis** window is used to select the main functions of the *QLA*. The options are introduced briefly below.

- 'PacketReceiver': allows the user to select input data type and source (see section 2.2).
- 'Parameter Selector': allows the user to select parameters in order to view them as they change over time (see section 4).
- 'Image Display': allows the user to view bolometer arrays of raw SPIRE data in the form of an image of the instrument projection on the sky (see section 5).
- 'Packet Dump': this dumps the contents of raw telemetry packets in Hex to the screen according to the structure identifier (SID) of the packets (see section 3).
- 'QlaConsole': this allows access to all *QLA* public methods via the Jython scripting language (see section 7).
- 'InteractiveQla': this is a legacy application which has been functionally replaced by the QlaConsole.
- 'EventTester': A developers' application for unit testing aspects of *QLA* functionality.
- 'PacketViewer': A demo application that shows when a packet is received by *QLA*.
- 'DataViewer': A demo application for viewing data value and saving data (see section 7).

2.2 Selecting Input Data

The input data type and source are selected using the **PacketReceiver**. First the data source and data type need to be selected, under the **Source** and **Data** tabs respectively. For playback, the required time period should also be set, under the **Time** tab. *QLA* will start to receive data once the **play** button is pressed, under the **Play** tab (see Figure 2). Note if the **play** button is pressed before an input stream is set up *QLA* is likely to timeout. The **Messages** tab can be used to check if this has occurred.

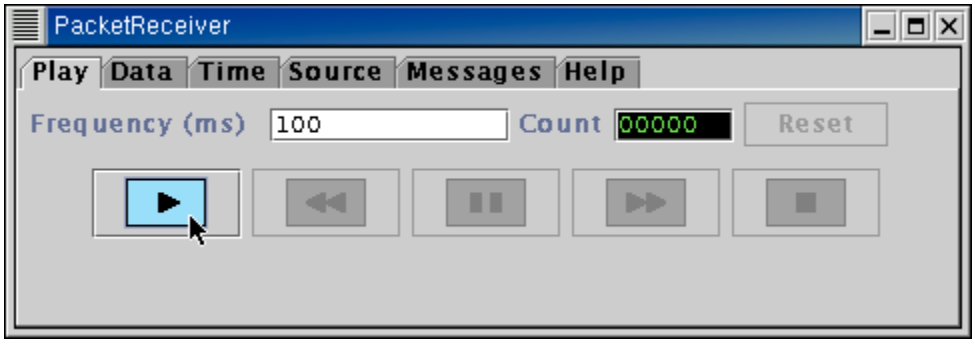

Figure 2: Selecting the **play** button on the **PacketReceiver**

2.2.1 Data Source

There are three sources from which *QLA* can retrieve data: a local or remote database for playback data, or the router for real time data. If the **Source** tab is pressed one of the three options can be selected (see Figure 3).

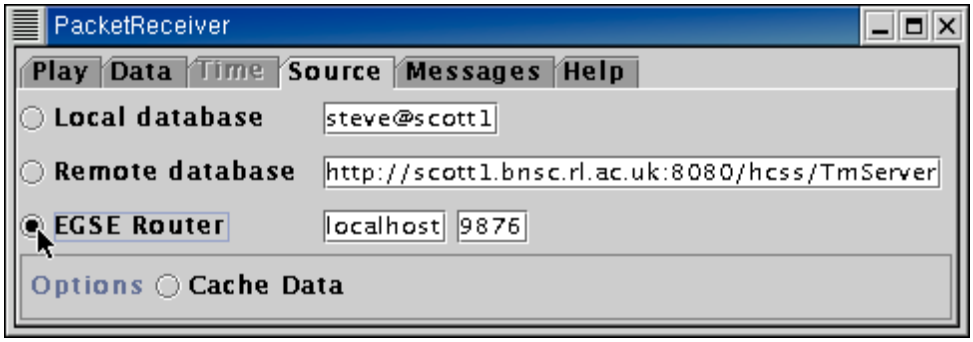

Figure 3: Selecting the data source from the **PacketReceiver**

2.2.1.1 Local and Remote Databases

A database has to be selected in order to run *QLA* in playback mode.

A local database is one that the user is able to access directly from their machine. This is simpler and provides better performance than a remote database, but means that Versant has to be installed on the machine running *QLA*. The text box contains the name of the database and may be edited.

A remote database can be installed anywhere that is network-accessible. Both TCP/IP sockets and HTTP protocols are supported. This does not require a Versant installation on the client, but does require a server to be running. Depending on the selected protocol, either a URL will be displayed (HTTP) or a host name and port number (TCP/IP).

The relevant configuration parameters are defined in the *QLA.defaults* file (see section 2). Refer to your system administrator for details.

2.2.1.2 EGSE Router

With the **EGSE Router** selection the first text box displays the host name, the default being **localhost**. Its value is determined by the property `hcss.access.router.host` in the *QLA.defaults* file (see section 2). The second box displays a four-digit port number, the default being 9876. This is set by the property `hcss.access.router.port` in the *QLA.defaults* file.

If the router is being used this should be started at the command prompt. The four-digit port number forms the last part of the command, and must be the same as that in the *QLA.defaults* file as described above. Speak to your system administrator if you have any problems.

```
> java nl.esa.herschel.egserouter.Router 9876
```

Note: Windows users will need to start another Command Prompt window in order to do this.

2.2.2 Data Type

The packet types received by *QLA* can be selected from under the **Data** tab. Although the option of **Data frames** is visible this has not yet been implemented as SPIRE data frames are still to be defined and this button will not work. At present the **Browse** button also does not do anything, but will eventually start the Test Execution browser.

2.2.3 Time

The **Time** tab allows a start and end time to be selected for playback purposes, with each time being entered as a day, month, year, hour, minute second (This is UT). This tab is only available if a local or remote database is selected under the **Source** tab.

3 Packet Dumps

The Packet Dump dumps the contents of raw telemetry packets in Hex to the screen according to the structure identifier (SID) of the packets. The default SID is 0x300 (Housekeeping), and a different SID can be selected from the pull-down menu on the **PacketDump** window. This window will only display one packet SID type and will ignore any packets that have a SID different from the selected one. The default display update rate is 500 milliseconds, and this can be changed by the `qla.DisplayRate` property in the *QLA.defaults* file (see section 2).

4 Displaying Parameters

In order to watch a parameter evolve with time, a user needs to select the parameter they are interested in, and then select how they wish to view it. Both of these selections are made via the

ParameterVisualisationSelector window which is started by selecting the 'Parameter Selector' option from the **SPIRE Quick Look Analysis** window. The ParameterVisualisationSelector consists of:

- a display area
- a **Parameters** menu
- an **Arrays** menu
- a **Help** button
- a **Display using** menu
- a **Clear Selected Parameters** button

4.1 Parameter Selection

Parameters can be selected in one of two ways. For specific bolometer parameters (PHOTOMETER or FTS), the user can click on the relevant bolometer array name displayed in the **Arrays** menu, which brings up a window with a visual display from which bolometers can be selected. Alternatively individual parameters can be selected from the **Parameters** menu, via the relevant submenu.

4.1.1 Selecting Bolometers From Array Images

The **Arrays** menu contains five buttons labelled with the five arrays in SPIRE. Pressing one of these buttons will bring up a window with a display of the bolometer projection on the sky (see Figure 4). On the left are two display areas, the top one showing a menu containing all the bolometer names in the array, and the bottom one displaying selected detectors. A mouse click on any bolometer, either in the array layout or upper display area, will put its identifier in the lower panel. A second click on the same bolometer will remove it. Once selections are completed, the **Confirm Selection** button displays the selected parameters in the **ParameterVisualisationSelector** window. If **Confirm Selection** is clicked twice for a particular parameter, it will be removed from the main display. To start a new list of parameters, the user must click on the **Clear Selected Parameters** button.

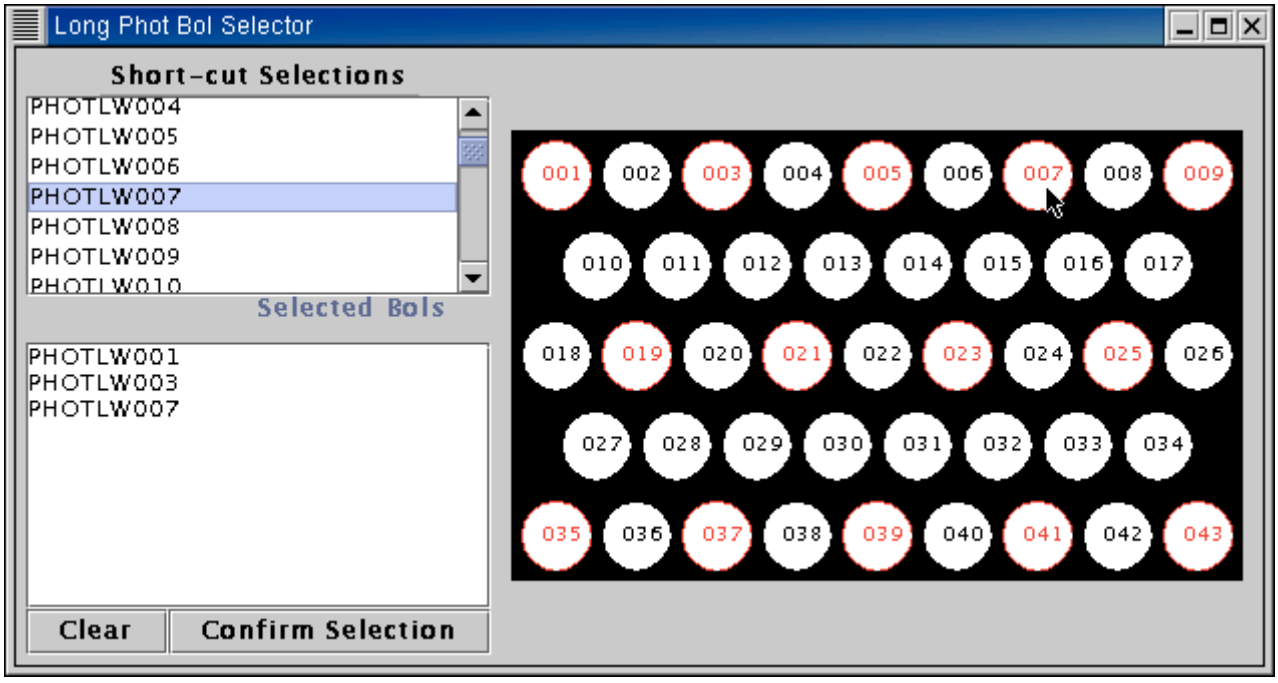

Figure 4: Selecting parameters using a bolometer array display

4.1.2 Selecting All Parameters

Any parameter can be selected via the pull down **Parameters** menu on the **ParameterVisualisationSelector** window. At present only the **Alphabetically** and **Subsystem** submenus have been defined and implemented, with the **Functionality** and **Operating Mode** ones still to be defined. The **Alphabetically** submenu lists all parameters in alphanumeric order, and the Subsystem submenu divides them according to their subsystem type (see Figure 5).

Once a parameter has been selected its name will appear in the display area. Selecting it again will remove it from the display area. To start a new list of parameters, the user must click on the Clear **Selected Parameters** button.

Descriptions of the parameter names can be found in the *SPIRE_Param_DB.txt* file, located at */.../herschel/spire/qla/tables/SPIRE_Param_DB.txt.*

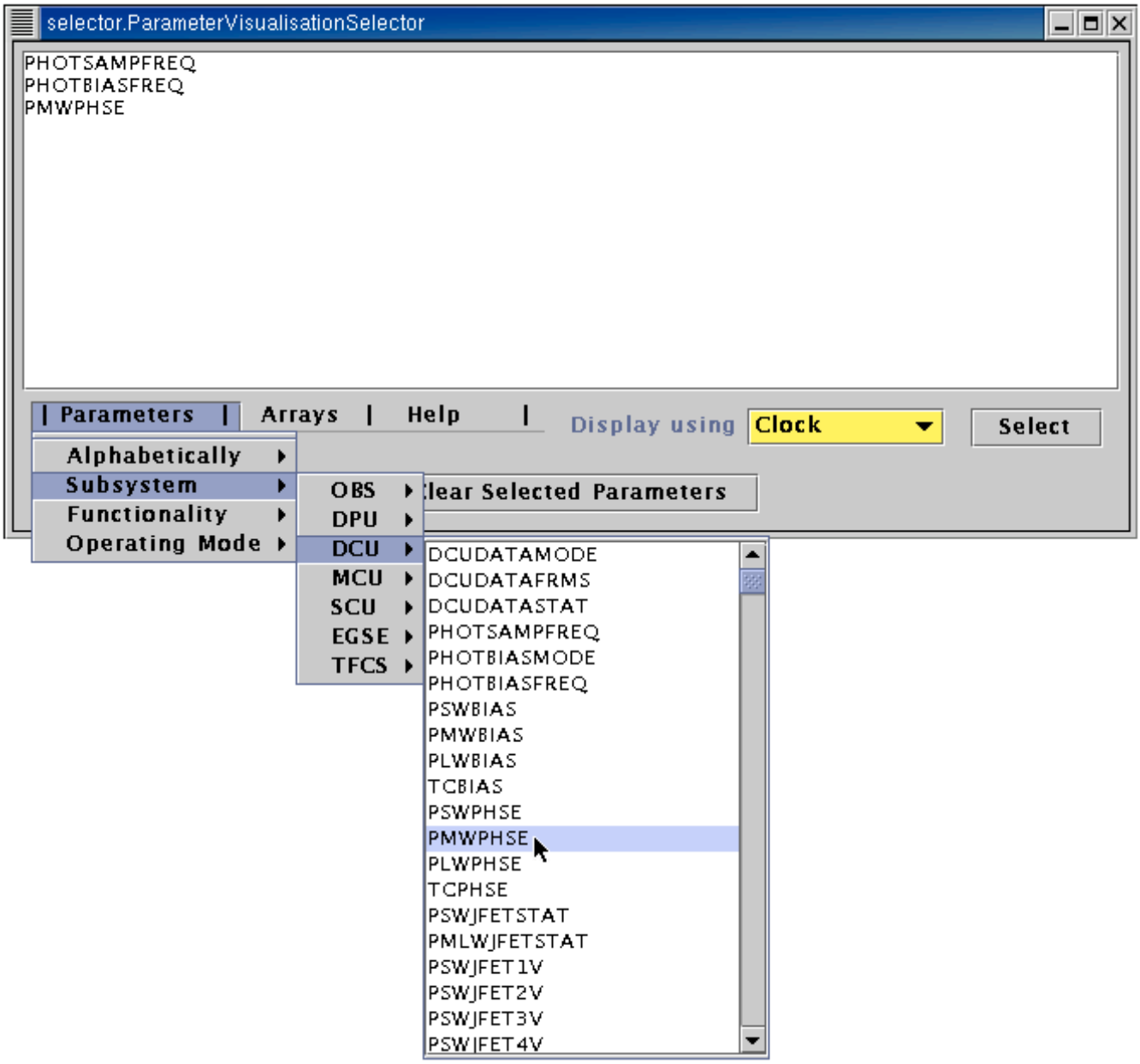

Figure 5 Selecting parameters individually via the **Parameters** dropdown menu.

4.2 Creating Displays

Once parameters have been selected, then the display type: 'Clock', 'Scroller', or 'Time Series' is chosen from the **Display using** dropdown menu. Clicking on the **Select** button will invoke a new **Parameter Displayer** window of the appropriate display type (see Figure 6). The selected parameters are viewed as separate components within the window. The behaviour of the selected parameters is shown from the time that the **Select** button is pressed.

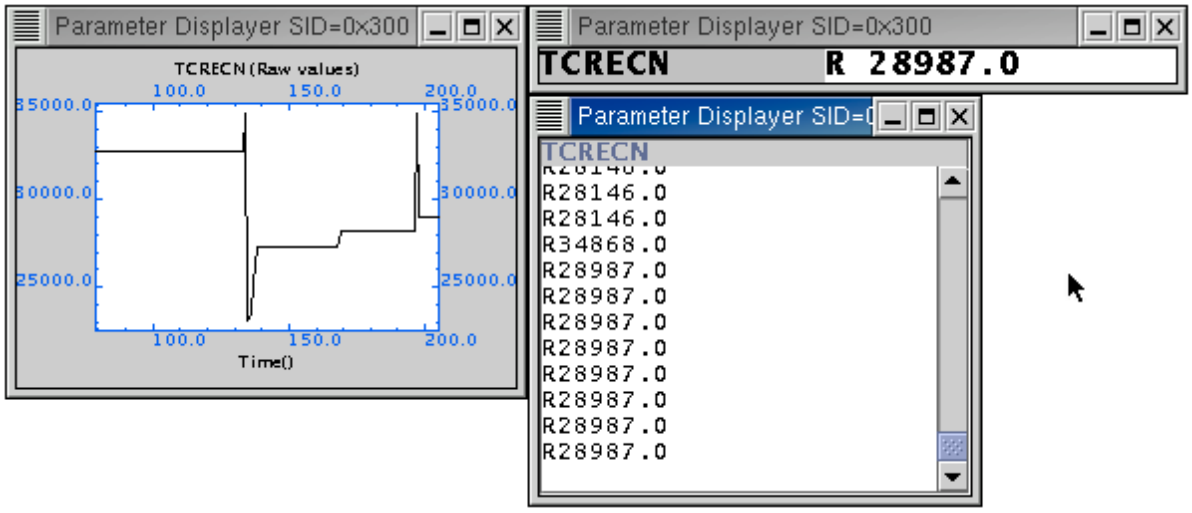

Figure 6: **Parameter Displayer** windows for the TCREN parameter, shown (clockwise from left) as a time series, clock and scroller.

To display a new set of parameters, press the **Clear Selected Parameters** button on the **ParameterVisualisationSelector** window. The already dispatched displays will continue to update. A new set of parameters can then be selected and new display will appear when **Select** is pressed again. Closing the **ParameterVisualisationSelector** window will not close the **Parameter Displayer** windows invoked by it. There is no limit to the number of times a parameter can be selected or the number of different ways it can be displayed.

4.3 Clock Displays

The Clock displays the latest values of parameters like a simple digital clock. Each clock will list the parameter names on the left and its value on the right. If both housekeeping and science values are selected, they will appear on different clock displays.

If the parameter has raw values, the raw value is displayed with a preceding "R"; if the parameter has converted values, the converted value will be displayed with a preceding "C". Clicking the popup menu mouse button (usually the right mouse button) will eventually allow the user to access the saving option although this is currently not functional.

4.4 Scrolling Lists

The Scroller displays the current and past values of the selected group of parameters in a set of scrollable lists. The lists display all data since the **Select** button on the **ParameterVisualisationSelector** was pressed. Like the clock displays, the science data and the housekeeping data are displayed in separate windows. When new data arrives, the display automatically switches to show the latest data at the bottom of the list.

Different parameters are scrolled independently of each other by default. Clicking the popup menu mouse button (usually the right mouse button) on any one of the lists brings up the option to **lock** scrolling with one or more other lists. This means moving one scroll bar will move the scroll bar of all other locked parameters at the same time. Clicking the popup menu mouse button again gives the

option to unlock from any previously locked parameters (see Figure 7). The format of every line is identical to that of the Clock (see section 4.3).

Clicking the popup menu mouse button also brings up the option to save the data (see section 7).

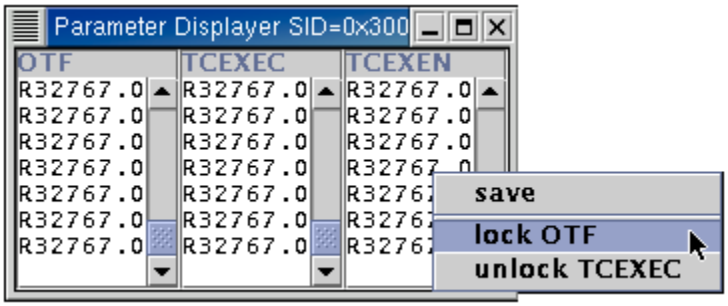

Figure 7: Using the popup menu to select locking and unlocking of scroll bars. In this example the mouse was clicked in the TCEXEN parameter window. The TCEXEN parameter had previously been locked to the TCEXEC parameter, which is why the option to unlock from it is shown here. It had not yet been locked to the OTF parameter, so the option to lock to it is shown.

4.5 Time Series Plots

The Time Series displays the values of a group of parameters as functions of time. The display has optional **compress** mode, **follow** mode, and **fix** mode. The display range and display size can be changed by the user.

4.5.1 Compress

The default mode for the Time Series display is the compress mode. This keeps the X coordinate of the left edge of the plot constant and compresses the X axis scale when new data become available. This means that new data are always within the display range.

- Double-click the mouse (left button) to return to the default plotting in compress mode including all data from the beginning to the newest data point.
- Control-double-click (left button) to plot in compress mode showing the current X range. This is only relevant if the user has zoomed in or out and changed the default X range (see section 4.5.4).

The y-axis scale varies to accommodate the highest and lowest data points.

4.5.2 Follow

The **follow** mode makes the display window follow the latest data point with a constant X coordinate span.

- Shift-double-click to plot in follow mode. The span will continue to display the same width as it had when shift-double-click occurred.

The y-axis scale varies to accommodate the highest and lowest data points.

4.5.3 Fix

The fix mode simply fixes the coordinates of all sides of the plot, both x and y axes, so that the displayed part of the data can be inspected without interference of new data. Fix mode only occurs automatically after a particular zoom action has taken place (see section 4.5.4 for details).

4.5.4 Other Use of the Mouse

Click the right mouse button to access the **saving** option; each data set can be saved separately (see section 7).

Drag the mouse holding down the left button to zoom. The release point of the mouse button can be outside the **Parameter Displayer** Time Series window, but the press point must be inside it. If the release point is outside the window, this has the effect of zooming out.

The X axis of all the parameter displays is adjusted when the user zooms. However, only one parameter display will have the Y axis adjusted.

Because several parameters can be displayed at once, a method is needed to determine which parameter display the user intends to update. If the user presses the mouse within a particular parameter display, then this is the display to have its Y axis updated. If the user clicks on the grey area either to the left or the right of the white displays, then the parameter display that the click is adjacent to will have its Y axis updated. If the user clicks on the grey area above or below the white display areas, then the top box will have its Y axis updated.

If the newest data point is not within range after zooming, plotting will switch into the fix mode to allow inspection of existing data without the interference of changing axes.

5 Image Displays

The **selector.BolometerImageDisplayer** window is launched when 'Image Display' is chosen from the **Process Selection** menu of the **SPIRE Quick Look Analysis** window. It can also be launched automatically when *QLA* is started by changing the values for `selector.BolometerImageDisplayer` in the *qla.processes* file (*/.../herschel/spire/qla/qla.processes*) to the following:

```
Selector.BolometerImageDisplayer true true Image Display
```

The user should select the desired bolometer array from the five buttons on the **selector.BolometerImageDisplayer** (e.g. **PHOT LONG**), which causes *QLA* to automatically register all the bolometer parameter names to be monitored. It also launches a new **Imager** window which displays the raw SPIRE data in the form of an image of the instrument projection on the sky (see Figure 8). When a science data event, from the simulator for instance, is received the bolometers with signal will "light up" using the colour table selected.

The current default colour table is a grey scale and pressing the 'Colour Table Selector' allows other monochromatic selections.

Holding the mouse over a bolometer will display the current signal value in the information box at the bottom of the imager panel.

The update rate of the display is set by the `qla.eventrate` value in the *QLA.defaults* file (see section 2). The default rate is an update once per second (a value of `1000` for the `qla.eventrate`).

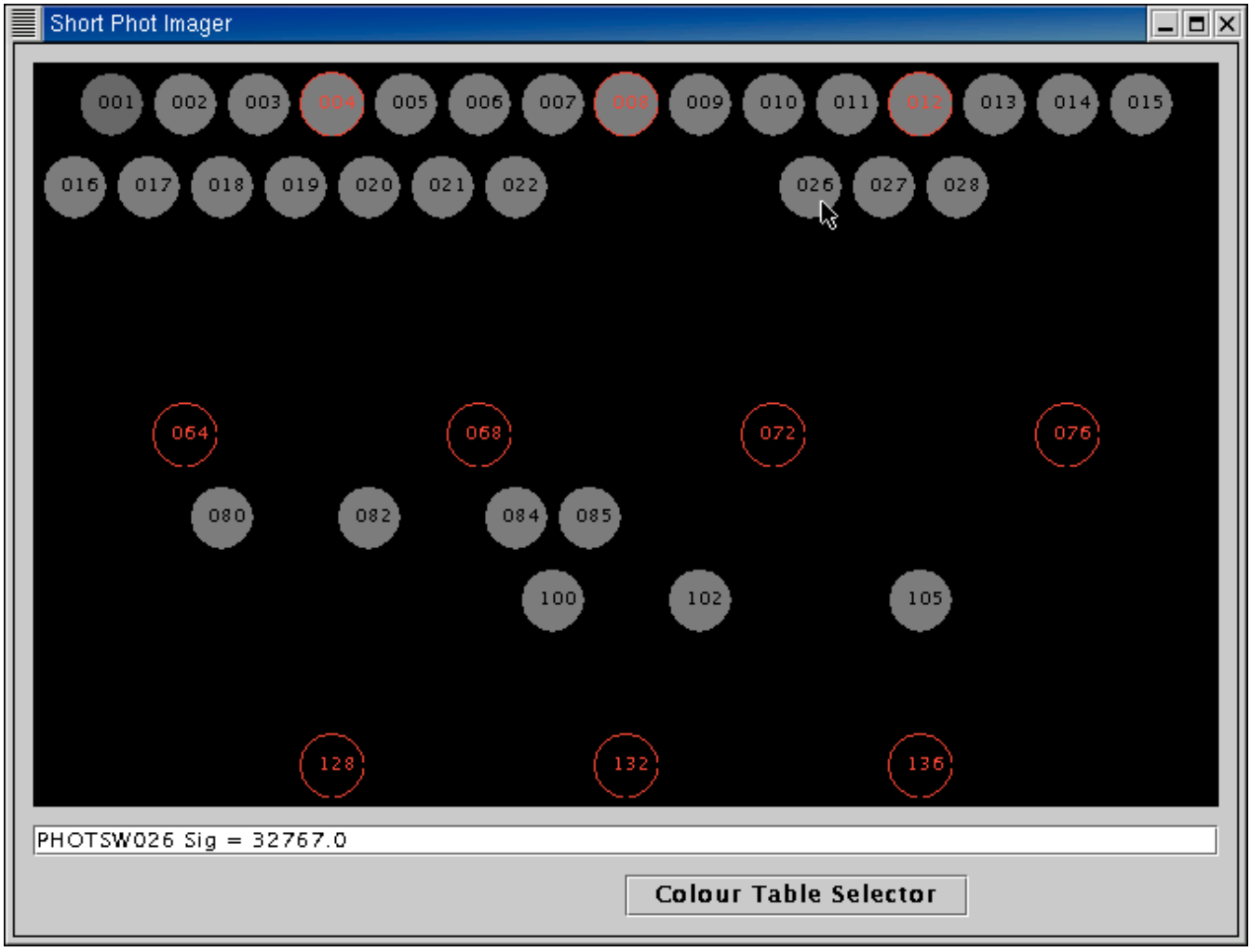

Figure 8: **Imager** window showing the Short PHOT array. The grey parameters are those receiving science events from the simulator. The user is checking the value of PHOTSW026 by holding the cursor over the relevant bolometer.

6 Adjusting Playback Speed

When *QLA* is used in playback mode the speed can be adjusted via the **fast forward** and **fast reverse** recorder buttons on the **PacketReceiver** window. In order to use playback mode select the **cache data** option from the **Source** tab of the **PacketReceiver** window (See Figure 3).

The pause button pauses playback, and causes two more buttons to become visible, which allow the user to step forwards and step backwards one packet at a time. This is intended to be used in conjunction with **PacketDump** for highly time refined error diagnostics (see Figure 9).

Figure 9: The **PacketReceiver** menu in playback mode, with the pause button having been pressed, and the step forward button being selected

7 Saving Parameter Values to an ASCII File

The **Save data** window can be accessed from the *DataViewer* demo application (accessed from the Process Selection menu on the **SPIRE Quick Look Analysis** window), or by pressing the popup menu mouse button in *QLA* windows which have a save function implemented, such as time series **Parameter Displayer** windows. To save data for a particular parameter to file the following instructions should be followed:

- If you are using the *DataViewer* application, enter the parameter name(s) in the **DataViewer** window (note: *QLA* is case-sensitive and all parameter names only use upper case letters)
- Press **Save data** – this brings up the **Save data** window showing the parameters in the current data buffer (i.e. the selected parameters that can be saved) and the various options for doing this (note: selecting a particular parameter in this window does nothing).
- Select **Export data to file**, and then click on the **launch** button – this brings up the **Data export to file** window showing the parameters in the current data buffer and the possible file formats that can be used (see Figure 10).
- Select which parameter(s) to save from the **Product Selection List** (note: multiple non-contiguous selection is supported).
- Select which data format to use from the **Data Format** selector. Note: **ASCII** refers to a structured flat-file format (i.e. the data is saved in human-readable form but with keywords describing the data); **Unstructured** refers to an unstructured flat-file format (i.e. the data is saved in human-readable form without any structure, e.g. plain columns of data).
- Press **Transfer Data** – this brings up a **FileChooser** window.
- Select the filename for the file to contain the parameter data and press **Save**. Note: If no extension or an inappropriate one for the selected data format is attached to the filename, the default extension for the selected data format is used. The configured values are:

Data format	Supported extensions	Default extension
ASCII	.dat, .DAT, .txt, .TXT	.txt
FITS	.fits, .FITS, .fit, .FIT	.fits
XDF	.xdf, .XDF	.xdf
Unstructured	.dat	

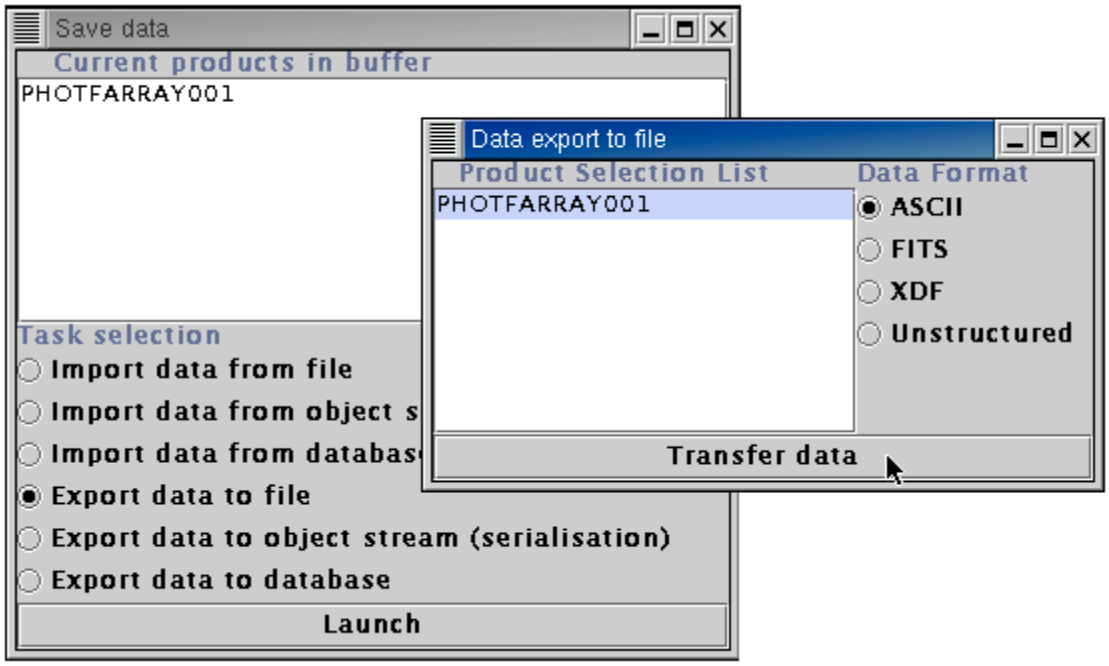

Figure 10: The **Save data** and **Data export to file** windows

8 Using The Console

The **QlaConsole** window is started by selecting the 'QlaConsole' option from the **Process Selection** menu in the **SPIRE Quick Look Analysis** window. The console allows access to all *QLA* public methods via the Jython scripting language and any Jython command is accepted. Advanced users can consult the Java doc for details on the public *QLA* methods available. Basic concepts are given in this section.

8.1 Imports

Several Java packages are imported automatically when the console is started. These are: *java.lang*, *herschel.spire.qla*, *herschel.spire.qla.viewer*, *herschel.spire.qla.selector* and *herschel.spire.qla.dataio*. The standard Jython syntax to import a package e.g. for java.lang is:

```
QLA> from java.lang import *
```

8.2 Starting and Using processes

8.2.1 Starting PacketDump

```
QLA> WindowManager.add("Packet Dump", PacketDump())
```

8.3 Printing the values of a parameter

There is a pre-defined variable of type *ParameterManager* called *pm* associated with the console.

This variable allows full access to the data, including the ability to start monitoring parameters. Full details are in the Javadoc.

```
QLA> t=pm.add(“THSK”)
QLA> print t
```

8.4 Saving Parameter Values to an ASCII File

For example, saving HKOBSID to a file

```
QLA> p=pm.add("HKOBSID")
QLA> from java.util import *
QLA> a=ArrayList()
QLA> a.add (p)
QLA> dt=DataTransferGUI(a,f)
QLA> from java.io import *
QLA> f=File(".")
QLA> v=Vector()
QLA> v.add(p)
QLA> dt=DataTransferGUI(v,f)
QLA> WindowManager.add("Save file", dt)
QLA> pm.add("HKBBID")
QLA> l2=Arrays.asList(pm.getParameters())
QLA> dt2=DataTransferGUI(l2,f
QLA> v2=Vector(l2)
QLA> dt2=DataTransferGUI(v2,f)
QLA> WindowManager.add("Save file", dt2)
```

APPENDIX A – THE ENGINEERING SIMULATOR

An engineering simulator has been produced for *QLA* test purposes. This produces simulated packets with the correct format at the correct rate and allows the user to insert values for testing purposes.

To start the simulator the following must be typed at the machine command prompt.

```
C:> java herschel.spire.engsim.Simulator
```

A **Spire Engineering Simulator** window will appear. The three buttons on the left, when pressed, bring up **Simulator Parameter Selector** windows, which are menus of the SPIRE parameters. When a parameter is selected from the menu a **Simulator Parameter Settings** window for that parameter appears with a slider. Further selections from the same **Simulator Parameter Selector** window will add more parameters to this same **Simulator Parameter Settings** window. Separate windows are generated for science and housekeeping data (See Figure 11). Closing a **Simulator Parameter Settings** window shuts down the whole simulator, but closing a **Simulator Parameter Selector** window does not.

Pressing **Data Write** will start the packet stream from the simulator to the router.

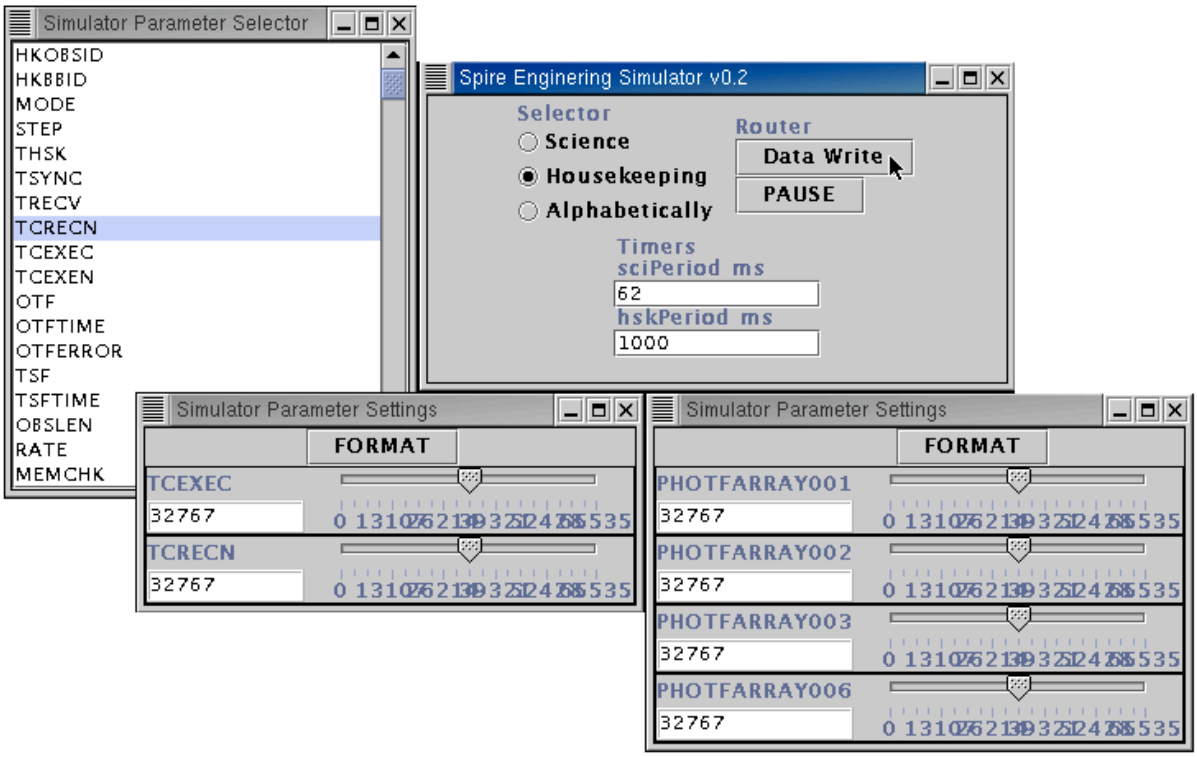

Figure 11: The **Spire Engineering Simulator** window, with 6 parameters selected (4 science and 2 housekeeping). The **Simulator Parameter Selector** housekeeping menu is also visible.