Report on the
Workshop on Computing Requirements for
the Neutron Beam Instrument Project at the
Australian Replacement Research Reactor

Building 55 Conference Room
Lucas Heights Science and Technology Centre
ANSTO
Sydney

9\textsuperscript{th}-10\textsuperscript{th} December 2002
Report on the Computing Workshop for the Neutron Beam Instruments Project, held at the Lucas Heights Science and Technology Centre, Sydney, 9th-10th December 2002

Prepared by the participants of the workshop

Edited: Paul Hathaway & Nick Hauser

1 INTRODUCTION

1.1 Executive Summary
1.1.1 Purpose of the workshop
1.1.3 Overview by the President of the Australian Neutron Beam Users Group. Ian Gentle.

1.2 Overseas Expert Reports
1.2.1 Mark Koennecke. SINQ. Paul Scherrer Institute. Villigen, Switzerland.
1.2.2 Chris Moreton-Smith. ISIS. Rutherton Appleton Laboratory. Didcot, England.

1.3 Outline of design process
1.3.1 Gathering information Neutron Beam Instruments Project (NBIP) scientists
1.3.2 International Review
1.3.3 Survey of Australian Neutron Beam Users Groups (ANBUG)

2 DETAILED REPORT

2.1 System
2.1.1 Systems: ISIS (Chris Moreton-Smith)
2.1.2 Systems. Paul Scherrer Institut (Mark Koennecke)
2.1.3 Distributed control using National Instruments DataSockets (Christopher Relf)
2.1.4 Motion Control Survey (Leo Cussen)
2.1.5 Detection and Data Acquisition Electronics Survey (Brett Hunter)
2.1.6 Discussion: User and Instrument Advisory Team (IAT) Requirements
2.1.7 Closeout

2.2 Data Formats and Data Access and an Integrated Proposal System
2.2.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)
2.2.2 Data Formats. Paul Scherrer Institut (Mark Koennecke)
2.2.3 Discussion: Data Formats
2.2.4 Remote Instrument and Data Access (Greg Doherty and Friedl Bartsch)
2.2.5 An Integrated Proposal System (Dennis Mather)
2.2.6 Closeout

2.3 User Interfaces - The instrument specific part of instrument control
2.3.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)
2.3.2 The ISIS Interfaces (Chris Moreton-Smith)
2.3.3 The Paul Scherrer Institut Interfaces (Mark Koennecke)
2.3.4 The ANSTO Interfaces (Andrew Studer)
2.3.5 Discussion: Instrument Control, Instrument Status and Interface Requirements
2.3.6 Closeout

2.4 Data Reduction and Analysis
2.4.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)
2.4.2 Data Analysis for Powder Diffraction Experiments (Brett Hunter)
2.4.3 Single-Crystal Data Analysis (Garry McIntyre) 33
2.4.4 Discussion: Data reduction and Visualisation (Andrzej Radlinski) 34
2.4.5 Discussion: Performance Requirements of Data Reduction and Analysis Packages 35
2.4.6 Closeout 36

3 CONCLUSIONS 39

Appendix A List of participants 40
Appendix B Program 42
Appendix C Workshop Report. Evan Gray. Griffith University. 45
Appendix D Workshop Report. Andrzej Radlinski. Geosciences Australia. 47
1 INTRODUCTION

1.1.1 Executive Summary

A major theme was a universally recognised need for remote access by users to both their data and their particular experiment, comparable with that provided by leading user facilities in North America and Europe. It seemed to the workshop that the best way to achieve this for the instrument suite to have its own network, segregated from the main ANSTO network. In addition it is highly desirable for ANSTO instrument scientists to have access to the instruments from home, or while on travel, so that swift remote diagnosis of problems can be performed.

It was agreed that there should be a common system for all 8 instruments, and that primary data storage should be for at least 10 years, via the emerging NeXus standard. The system should be functional in both GUI and command-line modes, simultaneously and in parallel. The common system should be able to handle all conceivable growth of the facility and should be flexible enough to accommodate a wide range of future needs and opportunities. A challenge for the coming months is to select such a system on the basis of capability, completeness and human resource requirements.

Reasonable round-the-clock operational support will be necessary, as is common elsewhere, though cost should be kept down via good design implemented in reliable systems.

The other remaining immediate choices are whether to go with VME crates or to a PC card-based data acquisition system and the choice of which computer control system to adopt, one that meets user and institutional requirements for reasonable capital and operational cost.
A detailed list of the attendees and affiliations is in Appendix C.

1.1.2 Purpose of the workshop

The purpose of the workshop was to gather the requirements of the stakeholders for a computer system for the 8 instruments of the Neutron Beam Instruments Project (NBIP) that provides instrument control, data acquisition, data analysis and data storage.

**Instrument control** means moving the instrument into a geometric configuration quickly (motion control), and to change and stabilise the sample environment quickly.

**Data acquisition** means providing the electronics to read the charge pulses that come from the neutron detectors, and converting this into a data format that records in some cases neutron position (x,y) and time.

**Data analysis** means to take this position and time data and apply a suitable reduction algorithm.

**Data storage** means to store the neutron data in a file that also contains sample data and instrument configuration.

Stakeholders in the Neutron Beam Instrument Project (NBIP) were invited to participate in the workshop. The stakeholders included the NBIP Instrument Scientists, members of each of the eight Instrument Advisory Teams, the NBIP Project Coordination Group, ANSTO Information Management, the Bragg Institute, the Australian Synchrotron Research Project, and the Australian Institute of Nuclear Science and Engineering (AINSE).

Through the course of the workshop, several key design issues were reviewed based on the requirements gathered.

The issues reviewed included:

- Whether to use proprietary (eg LabVIEW) or non-proprietary (eg open source) products for software and hardware development. This included the issue of writing device drivers for instrumentation. This can be a resource intensive task. This task both now and into the future is carried out for many devices by the software (eg LabVIEW) and hardware vendors. In the non-proprietary case, much of this work would be done in-house.

- Whether to use a heavyweight (in human resource required for development) distributed computing system eg. EPICS (Experimental Physics and Industrial Control System [http://wwwapsanlgov/epics/) or a lightweight system created specifically for neutron scattering instruments eg. SICS (SINQ Instrument Control System [http://lns00psi.ch/]). Systems that were reviewed also included TACO/TANGO ([http://wwwesrffr/taco/]), MX ([http://wwwimcaapsanlgov/mx/]), LabWindows/CVI DataSockets from National Instruments ([http://wwwnicomedevzonelibrariesdefaulthtm]) and the system used at Los Alamos Neutron Science Center ([http://lanscelanlgovlujanindex_lujanhtm])

- Should we develop a computer system from scratch, or was it possible to acquire an existing system that fulfils the requirements, or could be developed to do so?

- Were there any overlaps with the Australian Synchrotron where compatibility and technology sharing would lead to benefits to both facilities? Due to the synchrotron being in the early stages of its project life, there was no one to be an advocate for the synchrotron's requirements. Participants for this workshop are potential users of the Australian synchrotron, and gave their views.

- Data. Is there a common data format that we could use across the instruments? Where do we store the data? How much data will there be? How does a user retrieve the data?

- How do we design the user interfaces for the instruments and what are some of the common features?

- What are the requirements for data analysis and its integration into the control software?

Our invited international experts were selected based on their experiences of making decision on these issues, and having chosen significantly different solution strategies.
1.1.3 Overview by the President of the Australian Neutron Beam Users Group. Ian Gentle.

General points
- **The overriding principle should be that users gain efficient access to all aspects of instrument control and data visualisation and analysis. If this is not met, the facility will be a failure, and users will go elsewhere.**
- It is important that we meet “world’s best practice” on the new facility.
- By 2006 a lot of things will have changed, and we shouldn’t lock ourselves into ideas that will be out of date when they are actually used. For example, wireless networks will probably be universal, so “plug in” will mean something different, although the concept remains the same.
- Current restrictions such as not allowing instrument scientists to have administrator privileges on their own computers are counter-productive, and restrict their ability to do their jobs. There must be a certain level of trust given to staff and users of the new facility (as there is at overseas facilities).

Data storage and access
- Data should be stored in a common, agreed format. Nexus seems like a good choice.
- Individual user areas should be secure (although personally I don’t think this is a very serious issue – without knowing what data sets refer to, an intruder can’t do too much).
- Users must be able to easily plug their laptop computers into the local network, and gain Internet access. There must be ready network access to data collected on instruments, as many users will want to do analysis on their own computers. Because users need to be in contact with their home institutions while they are away, this must also include e-mail (POP3, SMTP, and IMAP) and web browsing. Other facilities such as ISIS allow this routinely, with DHCP used to assign temporary IP addresses.
- The easiest way to get data home (currently) is to copy onto your own computer using the fast local area network. By 2006 there will no doubt be new types of very high capacity removable storage media, and these should be available for machines that produce large data sets.
- Data collected on instruments needs to be backed up automatically, and retained indefinitely on some type of archive. This period should be a minimum of 5 years.

Proposal system
- Must be electronically based.
- The assessment process used at ISIS works well, with panels of experts for groups of instruments (eg small angle scattering and reflectivity) reviewing all proposals.
- The main criterion for acceptance is scientific excellence.
- There should be two proposal rounds per year.
- It works best if the scientific case, safety approval and funding are all handled in the one form like the Australian Synchrotron Research Project (ASRP).
- Instrument scientist comments on feasibility are important.

Instrument control
- A web-based graphical user interface for instruments should be implemented. This should be accessible from any computer (with appropriate security, such as password control), although it is probably undesirable to have control of the instrument too widely available. Possibly a “view-only” version available, anywhere. There must be Internet access available in the accommodation facility, so users can monitor (and control) their experiments from there.
- A common user interface should be implemented for all instruments, with a “tabbed-window” paradigm. This allows the use of some common windows for all instruments, and other windows to vary for instrument-specific purposes. Certain crucial windows should be visible at all times, with others varying depending on what’s happening. The main reason for commonality is to maximise efficiency of computing personnel who are responsible for support, and to present a professional approach to users.
- Demonstration paradigms should be trialed with users by the middle of 2003. User consultation should be frequent.
• There should be graphical representations of instrument functions for rapid status monitoring and ease of learning.
• New users should be able to operate the instrument after less than 30 minutes training.
• Real-time data visualisation is crucial.
• It should be possible to use a command line for expert users, in conjunction with the graphical user interface.
• Tables are a good way of handling scripting for batch runs.
• There should be basic and advanced modes of the graphical user interface available and they should be easily switchable.
• There should be a simulation mode to test scripts or tables.
• Certain reactor parameters should be continuously available to the system as required, such as reactor power, status of shutters, warnings etc.

Data reduction and analysis
• Basic data reduction (instrument dependent, but for example backgrounds, normalisation, dark current correction etc.) must be available within the main instrument control software. This should leave the data in a format suitable for further analysis such as modelling.
• A number of export filters should be available so that data can readily be loaded into analysis software. Most programs accept a simple ASCII format, but there are other common formats.
• Filenames for raw data should be allocated automatically, based on incremental run numbers. Names for reduced data can have more flexibility.
• Each instrument will have different requirements for analysis software, and different users will have their own preferences. The facility cannot be expected to provide all of these, but should provide one good package for new users. This may require supporting other platforms (Macintosh, Silicon Graphics) to a limited extent. This will be determined by consultation, closer to the startup date.
• Certain standard data analysis environments must be available generally. These may include IDL, Matlab, IGOR, etc.
• These must also be available by remote access.
1.2 Overseas Expert Reports

1.2.1 Mark Koennecke. SINQ. Paul Scherrer Institute. Villigen, Switzerland.
Views gained from running SINQ at the Paul Scherrer Institut, Switzerland

Instrument Control Systems

The Paul Scherrer Institut is running the client server system SINQ Instrument Control System (SICS) as its instrument control system for neutron scattering instruments. The requirements implemented by this system are:
- Common code base for all instruments
- Good remote accessibility from the office or from home
- Access control
- Provide for graphical user interfaces
- Portable and extensible.
- Server-side scripting language
- TCP/IP network as main instrument bus.

The experience with this system shows that it is worthwhile to have all these features. A common code base and standardisation of the hardware used reduces the maintenance effort required to support the system and increases overall system stability. Having the functionality required to control the instrument separated in a server facilitates debugging and allows us to run pretty much any user interface on top of it. The server-side scripting language is a very valuable tool for:
- Quick development of new commands (no compile/link cycle)
- Syntax adaptations on top of existing functionality
- Support of auxiliary or user supplied hardware.

Using the network as the main instrument bus allowed for easy spatial distributions of instrument components. We also experienced some instability induced by PCI-cards plugged into a PC. Using bus-TCP/IP bridges wherever possible isolates systems from each other and makes it much easier to exchange broken components. A broken instrument computer can typically be replaced in less than an hour remotely, just install the software, configure and run. This is far more complicated if PC-cards and drivers are involved.

One lesson we learned is that an instrument control system is rather a programming environment then a static application:
- Hardware is ever changing
- New instrument scientists come up with new ideas
- Instruments are upgraded
- Collaboration agreements require the inclusion of different hardware

DAQ software is a living thing! Moreover DAQ software is bound to have a very long lifetime (>= 20 Yrs) Usually one gets to build a new system once when taking a new system into operation or during a very major upgrade. The rest of the time there is no manpower to do anything but maintenance. With this given, it cannot be stressed enough that portability and extensibility are important issues. I also warn against locking into single vendor solutions as one Chief Executive Fraudster can ruin a company in no time and leave you high and dry.

As instrument computers I suggest to use Unix systems. This because such systems can be easily accessed from the command line which in turn allows support personnel to debug and fix problems from home. This is in some degree possible with Windows systems as well, using systems like VPN or remote desktop. However, such systems do not work well through modem lines.
Data Formats

A common data format for all instruments (possibly worldwide!) goes a long way to reduce maintenance costs and provides ease of use as users do not have to bother with conversion utilities. I recommend using NeXus based on HDF-5 as the common data format at Replacement Research Reactor Project (RRRP).

Remote Access and Control

Experimental data is stored at SINQ both at the instrument computer and on a central laboratory server. The protection is set to read only once writing the data file is finished. This data is freely visible within the SINQ hall and the offices associated with SINQ operation. Data is visible not only to Unix clients but also to Windows NT and Macintosh systems. Users can access their data from remote through a guest account and ssh/scp. A CD writer for writing CD is also provided. Users expect such a free level of data access.

Users may bring their own laptop and plug them into the Paul Scherrer Institut network without excessive administrative overhead. They then have access to all common network services (http, ftp, mail etc.) and, through the guest account, access to their data.

Today's facility users expect this kind of access. Experience shows that few problems arise from this kind of access. Users of neutron facilities are usually interested in keeping the facility running. They are not the enemy!

As a policy access to the instrument control programs is restricted to the Paul Scherrer Institut network.

Instrument scientists gain access through Paul Scherrer Institut dial-in accounts. This has been a policy decision. We need to provide some sort of virtual private network access though, when instrument scientists begin to use ADSL or cable modems at home.

User Interfaces

The Paul Scherrer Institut currently uses a command line interface as its mainstay for instrument control. The command line is augmented by graphical instrument specific status display applications. The command line interface has proven a very valuable debugging tool. First steps have been made to implement a WWW status and control application. At SINQ we have up to now been very busy to implement the required functionality and are only now starting to think about graphical user interfaces. Users clearly demand graphical user interfaces. Also today's students may have never encountered a normal (command line) interface in their lives.

Data Reduction and Analysis

For each instrument at least one complete chain of data reduction and analysis programs must be provided. Users like to evaluate their data on site with the help of the instrument scientist. Most instrument scientists at SINQ are either not willing or not capable to do serious programming work. Some help must be provided for instrument scientists during the installation and adaptation of data analysis software.

Concerning those general data analysis packages like IDL, Matlab etc: The laboratory should have a policy encouraging the use of a standard package. And provide support for it. However, at least one copy of each major package has to be around, as facility users become very unhappy when they cannot use their favourite package.

I also suggest investigating the public domain package Scilab for its use at Replacement Research Reactor Project (RRRP). University users may not have the money to buy licenses for the expensive commercial packages. Moreover the license administration for such systems increasingly becomes a pain in the neck.
1.2.2 Chris Moreton-Smith. ISIS. Rutherton Appleton Laboratory. Didcot, England.

General Comments

It seems sensible to give a quick summary of the ISIS computing and software development environment as comments here are based on experience in this environment.

At ISIS, a single, small (8-9 people) computing group is responsible for managing the computing hardware, software and networking for the beam lines, office systems and computing facilities provided to users and scientists; This comprises of the order of 20 scheduled instruments, 450 office machines and associated networking. The group makes good use of central, laboratory provided facilities for networking (to the rest of the laboratory and to the UK academic network) as well as central e-mail facilities. The group also contains expertise to provide advice on computing technology to the facility scientists and to form policy in consultation with the scientists. The group manages software development and maintenance for the majority of instrument control software, as well as developing some common middleware (Open GENIE for example). The group develops very little in the way of real scientific/analysis software. Most data analysis and some control software is programmed by the scientists and the group acts to support and encourage this software development; whilst doing this, attempts are made to spot and to abstract out useful infrastructure, for example data access routines, and then provide these centrally. In the ISIS environment staff effort is hard to control but finance to pay for contract work, maintenance or software is much easier to obtain.

There are some observations, which potentially have some relevance to the issues, commented on later in this report.

- Our instrument scientists and their collective (and sometimes individual) requirements ultimately drive the activities of the group.
- The scientists in general are largely uninterested in the computing infrastructure (as long as it works!) and would never provide a sufficient technological lead themselves to handle infrastructure development (in particular).
- The computing group therefore also provides a lead in developing requirements which may not been explicitly stated, often these are anticipated future requirements.
- The process of maintaining some non-trivial proportion of the original development effort to improve the systems and their design over and above normal maintenance and user support is valuable.

Comments on Session 1 – Instrument control computer systems.

Motion control, detectors + other sample environment control
- Never remains static and changes may be forced for variety of reasons
- Will always end up presenting diversity which saps hardware and software effort
- Need to operate a review process to keep reducing unnecessary diversity where possible
- Probably will end up being 80% or more of development cost over facility lifetime
- Setting up a good continuous development process and resources is important.

Systems for Instrument control
- Keeping office and Instrument platforms the same has real benefits.
- Uniform data access
- Familiarity with systems for users & staff, software and hardware support easier
- Desktops can take most of CPU, disk and memory load for analysis (fewer big servers).

Supporting the systems
- Ensure access to good, motivated and helpful support staff
- Need to remember primary function to help users - “can do” culture
- Attract and keep with challenging/interesting development work

Client/Server or Instrument/Remote access - or both?
- Is there are real split across a network or is it just a software layering?
- If there is a split can there be multiple clients and why/how?

Comments on Session 2 – Proposal systems, data formats and data access.

Data formats
- **Don’t invent any new ones, use NeXus instead**
  - If something really presents a problem, the NeXus team needs to address it!

Remote Access
- Assume broadband speeds as a minimum (because it will soon be pervasive)
- Assume access to full screen of instrument available as if at facility.
  - **For moving equipment, interlocks are only safe mechanism.**
  - Easy to mistype a number like 1.00 as 100 and break something, use limits.

Networks & Security
- **Allow users to bring notebooks on site and connect to instrument network.**
  - As a guideline, assume users want what a member of staff might want.
  - Networks are there to be as useful as possible (not as restricted as possible).
  - Establish record of real risk before putting in some draconian security measure.
  - Keep backups offline, checksum data etc. just in case an incident requires restoring data.

Proposal systems
- **If someone suggests a joint system look for political and software headaches.**

Comments on Session 3 – User Interfaces

Design process
- **Do deep prototyping early, as soon as there is any interface available.**
  - Test first bit of interface that can control a motor with a real scientist and a real motor (real motor must have backlash and be moved to a specific position!).
  - Could be wise to try hallway useability testing to simulate a new user?

Web interface
- **Make any web interface the local interface as well**
  - or people will just use remote screen software to use the real screen.
  - Reduced functionality via web or WAP most useful for status on small screens.

Interface
- **Taking user interface right through to reduction (and analysis) is innovative**
  - but user interfaces may really then have to be different for different machines.
  - Swapping a whole interface in for a specific machine use seems better than having “greyed” configuration options.

Comments on Session 4 – Data reduction and Analysis

Data Environment
- **Recommending one and integrating with analysis needs should save effort**
  - An environment with the “best of breed” analysis routines well integrated should be much better than a software or program library.

Developing Data Analysis Software
- **Putting some (computer science) effort into specific data analysis codes might pay.**
  - Even “tricky” porting across platforms could be much easier than scientists perceive.
  - Integrating known “best of breed” codes into a standard local environment could give a really good user experience. Data Seduction!
1.3 Outline of design process

1.3.1 Gathering information Neutron Beam Instruments Project (NBIP) scientists
This was in the form of 2 surveys
- Motion control survey. Presented by Dr Leo Cussen
- Detector electronics survey. Presented by Dr Brett Hunter

1.3.2 International Review
Information on computing systems was gathered from European institutes in September 2002
- Laboratoire Léon Brillouin (LLB)
  http://www-llb.cea.fr/index_e.html
- Institute Laue Langevin (ILL)
  http://www.ill.fr
- European Synchrotron Radiation Facility (ESRF)
  http://www.esrf.fr
- Forschungreaktor Muenchen II (FRM-II)
  http://www.frm2.tu-muenchen.de
- Paul Scherrer Institut
  http://www.psi.ch/index_e.shtml
- ISIS
  http://www.isis.rl.ac.uk/

Information on computing systems was gathered from US institutes in November 2002
- Advanced Photon Source (APS)
- Intense Pulsed Neutron Source (IPNS)
  http://www.pns.anl.gov
  http://www.ncnr.nist.gov
- High Flux Isotope Reactor (HFIR) Oak Ridge National Laboratory (ORNL)
  http://neutrons.ornl.gov/
- Spallation Neutron Source (SNS)
  http://www.sns.gov/
- Los Alamos Neutron Science Centre (LANSCE)
  http://www.lansce.lanl.gov/index_ext.htm

This information was summarised and presented at the workshop

1.3.3 Survey of Australian Neutron Beam Users Groups (ANBUG)
The membership of the Australian Neutron Beam Users Group (ANBUG) was surveyed in October & November 2002 for their responses to a group of questions. The questions were contained in
2 Detailed Report

2.1 Computer Control System

Each of the neutron beam instruments will be computer controlled. The computer system (fig 2.1) is the software and hardware that integrates the instrument. The system communicates to the devices (devices are histogram memory for detectors, sample environment, motion control and ancillaries), and provides for the requirements expressed by the participants of the workshop and by users through the ANBUG survey. These requirements are captured in this section. A review was made of computer control systems that are being used at neutron scattering and X-ray facilities. A recommendation will be made on which solution will fulfil user and ANSTO requirements.

2.1.1 Systems: ISIS (Chris Moreton-Smith)

In the past, ISIS failed to grasp the nettle of serious development and operation, successively choosing technology to keep things running and workable.

The design process was characterised by verbal specifications, intuitive design, trial and error in implementation within key constraints, continuous prototyping (important) and maintaining control of the systems and schedules at some cost in performance in the face of user pressures to adopt new features.

A few design pointers. Some goals:

- to maximise operational efficiency (fully utilise all resources)
- drive to improve the quality of science (fundamental improvements)
- to make experiments easier to perform/control/analyse
- to achieve better beam utilisation
- to accommodate smoother transition between experiments
- to enable greater flexibility in experiment design (control of multiple varying parameters)
- to utilise flexible data formats for the taking of generic data for later analysis
- to make the instrument the tool, rather than a fragile experiment itself
- to implement professional interfaces

Problems and Considerations:
- establish and control requirements well
- be aware of data rates
- be aware of electronics becoming outdated
- provide sufficient development time and tools
- tend to use crates rather than PCs for fast control requirements
- ISIS has one group to fully support the institution, with two staff dedicated to desktop PC support.

User Perspectives

Two levels of analysis required:
- sufficient analysis to properly guide control of the experiment
- detailed analysis of results

figure 2.1 This diagram represents the hardware and software layers used to control an instrument.

Communication of commands occurs between each of these layers. This architecture is known as client/server. The server layer (seen here in red) is designed to be common for all instruments, minimising development and maintenance costs. The primary function of the server is receiving and processing commands from the client and forwarding these to the devices. Additional functionality such as device access control resides on the server.

The client and user interface layers can be customised for each instrument. The type of control system chosen will determine the complexity of the device driver layer. The layers below the device driver are the hardware layers that convert software commands to electrical signals that control the devices.
Distinguish between survey and investigative experiments/instruments
Consider both hardware and software safety interlocks, especially with respect to remote control.
National Instruments (NI) cards have good support and have a number of third party card suppliers for virtual instrument or LabVIEW compatibility

2.1.2 Systems. Paul Scherrer Institut (Mark Koennecke)
Some general instrument control requirements - the systems must:
- run all instruments and hardware
- control any device
- have a common code base
- allow remote access for control
- maintain run buffers
- allow for graphical user interfaces
- support flexible (NeXus) data formats
- support any special user needs such as emulations
- be portable
- be extensible

Past design practices support:
- client-server architecture
- object oriented server design
- common protocol TCP/IP (and telnet) clients
- cooperative multi-tasking (portable implementation)
- server-side scripting
- access control with defined multiple levels of access
- server extensibility via scripting, batch files and implementation of new system commands
- a number of intelligent client types
- the easy addition of hardware

Lessons in Hardware at the Paul Scherrer Institut
- even home-grown electronics expire
- RS232 serial communications will be too slow in cases
- the provision of detector electronics was the biggest delay factor at SINQ due to competition for skilled human resources and handling of technical contractor problems

Lessons in Software at the Paul Scherrer Institut
- instrument control software is a more a programming environment than a static application
- the computing environment is always changing
- hardware is always changing
- requirements are always changing
- new and upgraded instruments need to be accommodated
- new scientists with new ideas need to be accommodated
- requirements of collaboration agreements need to be accommodated
- the system will have a long lifetime, but the initial design can only be addressed once
- extensibility and portability are real issues

Problems and Considerations with SINQ Instrument Control System (SICS)
- It works and does what we want!
- SINQ Instrument Control System (SICS) is not real time, but a coordinator
- cooperative multi-tasking can be problematic, and can be worked around
- a syntax checker is needed
- access to the client server environment can still confuse some users
- code clean-up is desirable
- on-line help is a priority - user manuals are frustrating, personal attention is preferred
- there are problems with the Java runtime
- there is a GPIB/Linux instability to attend to

Other Considerations
- Requirements are dynamic
- TCP/IP allows easy exchange of components
- command line remote administration is far more preferable than using graphical tools, due to their overheads and difficult extensibility
- the support group consists of two dedicated developer/maintainers and one administrative support person
- For SINQ Instrument Control System (SICS), approximately one third of commands are implemented as scripts
- security is maintained at server level

2.1.3 Distributed control using National Instruments DataSockets
(Christopher Relf)

Technical Input and Considerations:
- TCP/IP based, but server only runs on Windows platform
- maximum 1000 client connections, that can be permanently predefined or dynamically cached
- simple testing indicated 320kB/s throughput on 10BaseT connection
- maximum of one simultaneous server per machine
- no integrated encryption - third party software such as SnareNet (now owned by Computer Associates as eTrust VPN) available to encrypt at TCP/IP layer
- a common configuration is to establish a DataSocket Server outside the firewall to service remote users

2.1.4 Motion Control Survey (Leo Cussen)

International Facility Experience
A system for the Neutron Beam Instruments Project (NBIP) was described. This system is centred on a PC with a multi-pin I/O card used to drive stepper motors via some amplifier if necessary. The motor control intelligence resides in the software, which should be written in a widely used object oriented language. This proposal was
made to reduce the cost and complexity of the motor control system and to facilitate its long-term maintenance at the new facility. It was recognised that developing this system may take a significant effort and that a feasibility study must be made very soon.

The preferred commercial system option would be based on stand-alone controller with Ethernet port, as shown in fig 2.2.

Use of linear motors (self-positioning shaft in coil) is on the increase, with excellent performance and accuracy.

**Technical Input / Opinion**

Embedded microcontrollers for motion control, using DSP cores, are available to provide black-box functionality and allow easy upgrades. These are not the integrated controllers.

The (relatively) high radiation environment must be a consideration for the type of controller - this may favour less expensive, simple controllers using PC-based software intelligence.

Although the PC may not be real-time for all applications, it may be sufficient for slow control. The PC may be made real-time by use of an appropriate operating system.

**2.1.5 Detection and Data Acquisition Electronics Survey (Brett Hunter)**

**User Perspective**

Standardisation of equipment across a field rather than institution is not desirable, as it would multiply the support requirements by the number of instruments involved.

SNS and LANSCE are working toward storing each event \((x,y,t)\), potentially down to the shape of the event waveform.

Timing information (in raw data \(x,y,t\)) has been required for approximately half the current powder experiments due to changes in the sample eg. falling of a sample mount, during the run rendering part of the run void. When the information is histogrammed, it is impossible to remove the void section.

There are users that want only results and other experienced types that want to know all that happened during the experiment to aid their understanding.

Users want status information that makes sense to them eg histogram. This is not precluded by event storage as it can still be processed in real-time to update state information.

**International Facility Experience**

There are multi-week data capture requirements by some users - but they are generally not interested in the raw data and prefer to leave the technical side (of the instrument and team) to sort out events.

Typically, the detector is capable of greater resolution than the electronics. The software standpoint is to determine the resolution that we need and can use, and build that into the system.

When storing event data it is possible to use adaptive histogramming in firmware, setting bin size dynamically based on the data arriving in order to conserve resources.

**Technical Input / Opinion**
Event storage followed by processing can be flexible enough to allow:
- the same electronics whether time dependent or time independent
- the removal of time-stamp if not required
- histogramming to save memory resources if necessary

For standardisation, event storage could be implemented across all instruments.

If both event storage and histogrammed data storage is made available, make the capability switchable to avoid collecting all event data on all equipment at once, to conserve bandwidth.

Bandwidth is a concern. For HRPD 100Mb/s is okay, possibly over a dedicated line to storage before connection to Ethernet infrastructure.

There may be significant bandwidth and storage requirements for streaming data - on-line real-time analysis can evaluate whether the collected data is within parameters and the evaluation used to decide whether to continue, alter binning and how to store data.

One concern may be the long-term storage of unwanted and unreadable data.

2.1.6 Discussion: User and Instrument Advisory Team (IAT) Requirements

For the messaging system underpinning the instrument control ANSTO wishes to establish both a user perspective and a project perspective (for scheduling, lifetime etc). Due to the number of person years of development and testing it requires to produce a stable and reliable system, and that reliability is priority user requirement, the process of adapting an existing system was considered to be best. The system must fulfil the majority of user requirements.

User Perspective

Keep the Australian Neutron Beam Users Groups (ANBUG) requirements in mind.

The user primarily wants experimental results, reliably - to walk away with some experience and data in a useful format for the type of experiment in hand.

The user wants to be able to work with the system to run the experiment in an understandable way, preferably graphical and easy to learn.

The underlying operating system/platform may be very important if it impacts on the user interface. Ideally the interface will work, look good, provide help/assistance when required, and allow operations like the addition of ancillary equipment or application of supplied algorithms to data.

Using too many processes to achieve an end adds complexity and is frustrating. Many existing systems are like this. Possibly, the systems should be deemed useable by the experimenters before they start arriving at the facility.

Capability for web or remote access must be provided.

Capability for graphical user interface, batch mode and command line interfaces should be provided.

The user wants control over the experiment. We need a means to provide the necessary information to achieve it - whether on-line help or access to support, whenever and wherever they may need it. Faced with a problem, this may be as simple as the capability to restart the instrument and provide instructions.

There are educational issues:
- It is important to attract the first time user and assist them to become a sophisticated user. Control over the experiment, with its sense of achievement will aid this.
- The system should provide (with remote access) a simulated experience (or direct control of managed scenarios) to encourage students and prepare University participation by familiarity.
- Continually ask how the experience can be made more educational.
**International Facility Experience**

Instrument control decisions should not (be allowed to) lock you into a platform. Avoid the requirement to load more drivers than necessary, preferring to make use of TCP/IP connectivity for special devices. Users tend to not care about the underlying system platform.

Neutron instrument system interfaces are like programming activities - interactive, programmable - which may be difficult to maintain flexibly in a graphical interface. Scripting and batch modes must be made available.

Remote access is currently provided (at the Paul Scherrer Institut/SINQ) by SINQ Instrument Control System (SICS) specific clients, over basic/universal protocols. To provide additional web-based clients may take an additional person-year of development.

**Technical Input / Opinion**

Choice of platform brings to bear strong user opinions on interface. The system level is more flexible, but will affect the availability of developers, tools, and hence the maintainability of the system.

The choice needs to be flexible to accommodate growth and addition of new ancillary equipment.

NI DataSockets server is restricted to Windows platform.

For any operating system, server stability and flexibility is essential.

ANSTO Information Management ideal would be to restrict server platforms to combination of Windows and Linux, and to select web interface technology for clients. ANSTO Information Management already supports these.

There are two sets of user requirements: infrequent and inexperienced users need a simple interface, whereas instrument managers want full control and the ability to develop on that platform.

There are survey instruments with restricted, simple choice of action to run the experiment and investigative instruments that require significant interaction, and a more intimate knowledge of the instrument. A command line interface may be more efficient for those investigative users. The system needs to accommodate both but be flexible enough to allow the individual character of each type of instrument/experiment.

Web based solutions provide limited features that may either not be suitable, or need to be customised for each instrument.

Graphical programming is mature enough now to allow virtual instrument to appear/mimic actual equipment panels. The hierarchy of virtual instruments and control can support scripting, batch control, direct control, graphical user interface and command line.

Virtual Instrument front panels can be used to administer control servers. Control can be as at the equipment itself.

Web based interfaces are more scalable and allow interactive and batch modes combined with the provision of documentation to assist the experimenter. Support for a web browser client equates to universal access.
2.1.7 Closeout

Any of the listed systems may contribute to the solution system. This process is to establish some requirements that may eliminate some choices and then allow us to further refine our choices based on other factors such as maintenance, human resource requirements and reliability. Here we are looking at general user's perspective of requirements and user experiences of any of the systems surveyed.

User Perspective

It is a rare user that will have experience of all eight different types of instrument. Each user must try and convey what they want from their experiences to allow specialists to implement them.

Our requirements may be better expressed in terms of measurable criteria. For example, stability/reliability for SANS may require less than one failure a week on average, with down time limited to 15 minutes which may translate to 96% availability.

This type of figure assists the specification, quantifying our expectations. This is what we need to communicate - how the technical team meets those criteria is a separate problem.

Choice of platform or operating system should flow from the specification/requirements. Specify some degree of platform independence rather than specific platforms.

Criteria for response/assistance to a user difficulty depend upon the ease of diagnosis and the ability to rectify the problem in a realistic timeframe, possibly in absence of a software professional. The software should not only be robust, but fixable.

Platform selection should also consider the existing environs/culture of the facility.

A level of compatibility with future synchrotron systems should be driven by our desired neutron science capabilities at this facility, with provision for portable data formats.

If correctly designed and implemented, aspects of this system may be adopted there.

For a system of reasonable complexity, an infrequent or inexperienced user should become comfortable with the system within 30 minutes.

International Facility Experience

ISIS would not be able to report their statistical stability.

In four years the Paul Scherrer Institut has encountered only four problems caused by networking failures, and most of these were trivial. We do not consider this aspect of independence from the network a priority.

Institute Laue Langevin user reports that a separate instrument control network, independent of data processing and office facilities works well, unless instrument control feedback is a criterion.

Adopt one subnet per instrument, or at least one subnet for the guide hall network.

Technical Input / Opinion

It is highly likely from ANSTO Information Management standpoint that the expressed requirements for accessibility warrant an independent instrument network.
2.2 Data Formats and Data Access and an Integrated Proposal System

A discussion of data formats, system access methods and a request for requirements for an integrated proposal system

2.2.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)

User Perspective

The duration for data storage/archive may need to consider contractual requirements for commercial or sponsored work.

It is perceived that the cost of data archiving is negligible relative to the cost of running each instrument. Another perception is that despite the desire to keep older data, it is unlikely to be requested after a nominal term (in the vast majority of cases).

International Facility Experience

The option for laptop access was demanded by users and provided through little effort.

Technical Input / Opinion

There are issues to be dealt with for long term data archiving

- keeping unwanted and unreadable data
- on-line versus secondary storage in light of increasing raw data production rates
- costs and complexities in indexing to make the archive useful

2.2.2 Data Formats. Paul Scherrer Institut (Mark Koennecke)

Data formats

- There are many site-specific formats, mainly ASCII-based, that are never considered or suggested as a standard.
- CIF is an ASCII-based format specifically for crystallographic structure information. Despite some attempts, there is no significant move to further develop it.
- NeXus is a HDF based format proposed for neutron and X-ray data. The Paul Scherrer Institut is the only facility that has fully adopted this format, but there is a trend for new systems to implement NeXus. [http://www.neutron.anl.gov/nexus/](http://www.neutron.anl.gov/nexus/)

Data analysis

- SciLab is a Matlab-like public domain data analysis package that we have found useful and commend to you for further investigation.

2.2.3 Discussion: Data Formats

User Perspective

At least 5 years archive time is reasonable, longer if not limited by resources. Twenty years would be desirable.

The data is owned by the facility, unless the beam-time was paid for. Security sensitive data to be excluded from the archive on request.

It is essential that the data be self-describing (for meaningful long-term storage and indexing of data), and indexing and retrieval as automatic as possible.

There is no preference for storage format by users, as long as it can be translated into a format used at the instrument or used in the standard analysis packages. The single crystal community requires CIF for publication.
International Facility Experience

Data storage policy at existing facilities has tended to be to store every raw data file since the inception of the facility, regardless of its current practical value. To date, this has not caused a problem and in fact, close to all of it is available directly from on-line storage databases. These policies do not guarantee retrieval.

The Paul Scherrer Institut does not store event-by-event data.

Do consider that there is a storage overhead in indexing your data archive.

Ideally, the data archives should be reviewed every 5 years to determine whether formats or media are still acceptable, and transferred if necessary.

The Paul Scherrer Institut like to expect a 10 year archive of scientists' data, with no guarantee of retrieval beyond that.

HDF/NeXus formats are widely supported by tools, including free public domain packages that can be readily downloaded.

Technical Input / Opinion

There are costs in data retrieval from archives and management of archives.

ANSTO Information Management suggested a service level agreement specifying length of data storage time required, with no guarantees for further retrieval.

2.2.4 Remote Instrument and Data Access (Greg Doherty and Friedl Bartsch)

Any requests for types of access, types of accounts and access technology, needs to be run past a number of regulators/licensors (including Defence Signals Directorate, ARPANSA, ASIO, Audit Office) before ANSTO Information Management can offer to implement those requests.

The existing ANSTO network has strict security policies.

The microwave link to AARNET is being upgraded to 100Mb/s.

Voice, data and video services available on ANSTONET.

Consider cPCI cards and equipment as a replacement to VME, as it now provides a large choice of development tools, is faster, and is more resource efficient.

The Reactor Control and Monitoring System is on a separate network.

ASIO considers that all firewalls can be by-passed. Consequently, ANSTO networks are considered 'public', and do not have peer access to the Reactor Control and Monitoring System. If continuous network connectivity to this data is required, any network-connected devices selected need to have passed through a strict approval processes - effectively limiting the solution to an (expensive) data diode device or dedicated signal lines.

The Replacement Research Reactor Project (RRRP) network structure has been designed and planned, excluding the guide hall. Space is at a premium and has been assigned for a number of racks only.

ANSTO Information Management requires

- specification on the parameter data required from the Reactor Control and Monitoring System
- where access is required from,
- the approximate frequency of access to these parameters,
- an estimate of how much data is to be produced for archiving each year. Capacity will then be planned and purchased as necessary.
- requirements for voice, data and video
- whether a separate local area network is desirable for performance and security requirements
- specific cabling requirements (optical fibre, shielded copper etc)
  - consider the number of fibre ports desired,
  - consider wiring Ethernet directly to devices, and
  - connectivity required for racks located next to instruments
- Inform ANSTO Information Management of any further requirements in the Reactor Hall as soon as possible.

Equipment/wiring specification in the reactor hall has already been approved.

User Perspective

As an international user facility, this new facility needs very free access to both institution services and to the data. Current reactor/source parameters made visible on a status board in the guide hall are desirable.

International Facility Experience

Reactor parameters such as reactor power are highly desirable. Spallation sources require beam power information.

Technical Input / Opinion

The number of users may escalate for the new Replacement Research Reactor Project (RRRP) facilities. Estimates will need to be updated. An upper limit can be calculated based on beam utilisation time and throughput estimates. This workshop and the Instrument Advisory Teams (IAT) are expected to produce those estimates.

Any requests for open access need to be evaluated against ANSTO security requirements.

Consider the topology needs for the instrument network. There are costs associated with providing a separate local area network for differing security and performance needs, and a cost associated with providing information between this network and the secured ANSTO network.

Considering the stated desires for open visitor access, which may not meet ANSTO regulatory requirements, ANSTO Information Management is likely to insist on a separate instrument network.

2.2.5 An Integrated Proposal System (Dennis Mather)

- The application process should be electronic, on-line and interactive.
- Application stored to database for the evaluation process and future reference
- Interaction between the database by specialist committees, reporting processes and financial systems. Australian Institute of Nuclear Science and Engineering (AINSE) will work with the Bragg Institute to harmonise the two databases

User Perspective

Between the proposal system and systems for booking/scheduling instruments there may be an overlap of stored information

The Australian Institute of Nuclear Science and Engineering (AINSE) proposal vetting system is currently set up to service Australian University proposals, whereas these facilities will be opened to a larger community. We must consider a proposal system that manages proposals from the larger community. All proposals to this facility need a common review process. AINSE is ready to discuss with the Bragg Institute how its current system might be adapted to incorporate the larger group of applicants.

In effect this is an ANSTO policy decision that has not been made yet.
International Facility Experience

Overseas, it is unusual to see separated (proposal/management) systems.

Technical Input / Opinion

The AINSE proposal system might deal with university and non-university customers. AINSE is open to requests for extending the system and many have already been implemented.

2.2.6 Closeout

2.2.6.1 Internal access to network

User Perspective

An international user facility must have open access, characterised by

- visitor access, enabling access to data instantly
- able to plug in their own laptop
- access to instrument network and printers

Some scientists may wish their data to be private, not accessible by other users.

International Facility Experience

ISIS is relatively open internally, is open to external use via a firewall and is monitored by diagnostics. Ease of user access is a priority. ANSTO should consider establishing a policy document.

The Paul Scherrer Institut encourages relaxed access, automated as far as possible. There have been no malicious attacks in 4 years.

The consequences of any malicious action can be minimised - currently little risk to the instrument, user accounts required to do anything serious, accounts cannot destroy data that is backed-up in any case.

Privacy of user data (from other users) has not been a policy at several institutions.

Technical Input / Opinion

ANSTO Information Management would like the actual requirements to be defined. In the least, some sort of registration process is desirable; to know whom is connected, balancing freedom of information with quality of contact/data/security. Value your data - systems can be maliciously attacked.

Some systems may find it difficult to implement/guarantee truly read-only data that cannot be overwritten.

Mirror your data, it is cheap to do so.

Some users may be scared off using the facility if their data is not perceived to be secure. An additional capability to secure classified data needs to be implemented.

2.2.6.2 Remote access to network

Data processing via the web is exciting and full of possibilities, essential for getting the most out of the experiment - data analysis on the fly, from wherever the experimenter sits, is important when beam-time is limited.

User Perspective

Data transfer needs to be seamless and easy to use.

Some facilities provide remote data access immediately, to be able to compare data from different instruments (eg X-ray and neutron data). The distinction should be made between data reduction and data analysis.
International Facility Experience

Secure copy (scp) enforced by the Paul Scherrer Institut. Secure shell (ssh) and secure file transfer (sftp) available. Data is encrypted.

No authentication agents used. Users tend to reject any more sophisticated security features.

Institute Laue Langevin allows data processing off-site via the worldwide web for 3 axis spectrometers (TAS), time-of-flight (TOF), SANS and some diffraction instruments. (See http://barns.ill.fr).

Technical Input / Opinion

The main encryption concern is for passwords.

The Bragg Institute is expected to keep a read-only central repository, and the Commonwealth would want to reserve its rights to the data.

There would be a need for capability to divert commercially sensitive data, and the capability to prevent a user from secretly diverting data. (The metadata for those sessions could be maintained).

2.2.6.3 Flexible Data Format

The consensus opinion seems to be to adopt the NeXus data format for storage/archive purposes, whilst providing translation to major/common data formats. The data will be stored in raw format. Providing raw data to the end user would allow for reprocessing it at a later date.

These common data formats will be determined by the particular instruments and methods of analysis - a listing of these instrument/method-specific formats needs to be compiled by the Instrument Advisory Teams and provided to the Neutron Beam Instruments Project (NBIP) team.

International Facility Experience

Often the experimenters do not want the data in a raw format.

Technical Input / Opinion

A flexible data format would support good simulation functionality (such as at NIST).
2.3 User Interfaces - The instrument specific part of instrument control

2.3.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)

The user interface (UI) will make or break the success of the system from the users' point of view. We must consider and involve new/novice users in the user interface development.

User input and feedback will have an impact on the design, development and testing processes.

Components of the user interface
- instrument configuration (if needed)
- instrument control (general operation)
- instrument status

An instrument configuration user interface is required if the instrument is reconfigurable or user-supplied ancillaries are to be accommodated.

Some specific issues to resolve:
- include a macro language?
- Emulate which one/s? Multiple emulations? Graphical emulation?
- multi-window or menu driven operation
- capable of nested scans, with parameter changes possible on the fly?
- what would be the frequency of configuration change?

User Perspective

We should avoid the preconception that a command line interface (CLI) is the preferred interface of experienced users.

We expect web-based or functional draft graphical user interface for testing.

Features could include:
- configurable panes, allowing a decision as to whether numerous views are required
- pull down menus

2.3.2 The ISIS Interfaces (Chris Moreton-Smith)

The ISIS Control by Table Proposal

Under this table paradigm, widely available spreadsheet software is used to specify, monitor and interactively control the experiment.

Scripting is error prone. Tables are less so as they describe all actions explicitly. It could be described as an expanded script, unwinding each of the loops to allow description of complex sequences of events.

A script could be used to generate the table if necessary, which essentially tests the script. The user would also be free to adjust the script-generated table before use.

A sequencer (state machine) works through the table. The last or current values sourced from the instrument controller can be redisplayed within part of the table.

The immediate advantage is that the user can stop, restart and edit the settings interactively.

Check points are built into the table method.
Standard spreadsheet implementation of the table allows extra sheets to specify more detailed and additional parameter controls.

With the correctly formulated middle layers of calculation, the experiment could be described via the variation of scientific/physics parameters rather than instrument parameters.

**International Facility Experience:**

When given the option of basic or expert user interface, most users tend to display the most complex virtual instrument panels with the most parameters - whether or not those parameters are used.

From an educational point of view, the virtual instrument display should mimic the instrument in some way - even making use of photographs or web-cams.

Currently at ISIS there is no storage of user preferences or configurations. We aim to identify the user's produced files based on their login ID, but these files are not particularly secured.

The initial configuration aim at ISIS was to use only one PC to run the control, for reliability, but moving to an additional machine for running scripts and doing analysis, with the ability to transfer scripts/tables to the control machine. The second PC (script/table) would then simply prod (supply input/event to) the controls on the instrument control PC.

Using G as a scripting language has the advantage of allowing some analysis to be executed by the script. As to a historical comparison of the time required by a novice user for competent use of systems:

- past: required the reading of a short manual (less than an hour)
- present: new users need to be guided
- future: ideal use of a table-like user interface would allow the user to start immediately ask for assistance if something fails

**User Perspectives:**

Scripts/tables are good for the command line interface and batch mode, but may be insufficient support for the interactive start-up sequence of an experiment where a number of small commands are made based on immediate feedback from the most recent scan.

(An interactive table/spreadsheet is useful for even changing one parameter at a time and observing feedback via display of current values sourced from the instrument control.)

The table method useful to check a script in a simple experiment and display everything I need. Validation of script or table becomes important - especially at a late hour when concentration and support may not be optimal.

**Technical Input / Opinion**

The large number of instrument device parameters that need to be coordinated could be hidden by utilising the spreadsheet to translate scientific variables into device parameters and the appropriate manipulation of the spreadsheet view.

Validation of each parameter is possible on the fly, provided you can specify how to test it in the spreadsheet.

Syntax checking is necessary, but simulation is required to avoid destructive behaviour when dealing with complex combinations of axes. The only way to test control of a complex model is by experiment/simulation.

Consider making use of industrial control methods, including using in-built bounds checking on many/all device parameters. This is common practice on industrial 5-axis CNC machines.

**2.3.3 The Paul Scherrer Institut Interfaces (Mark Koennecke)**

There is no silver bullet for instrument control user interfaces (UI).

Consider instrument control as a programming task: desire to pre-program the experimental tasks and leave the experiment to run unattended.

- command line interfaces (CLI) are popular, as it is easy to add lines to scripts
- graphical user interfaces (graphical user interface) require enormous resources by comparison

SINQ Instrument Control System (SICS) makes use of several clients:
- Java-based command line interface client
- Several instrument specific status display clients were developed over a period of 6 months in collaboration with the relevant instrument scientists.
- Web interface (status display for each instrument, created by server side scripts, has the ability to view command/event log.)
- telnet interface

SINQ Instrument Control System (SICS) Web interface:
- status display for each instrument
- created by server side scripts
- can view command/event log
- integrate controls with documentation/assistance

SINQ Instrument Control System (SICS) Web experiment control page:
- instrument specific
- supports command line interface and batch mode execution
- batch files created via guide pages or direct edit
- configurable user parameters
- errors reported direct to page

SINQ Instrument Control System (SICS) telnet interface:
- minimal feedback/guidance/clutter
- can be operable via graphical or command line front end

International Facility Experience:
Users prefer the two-window solution: separate status and control panes.
Experienced scientists demand command line interface feature.
The time required to train a novice user depends on the complexity of the instrument:
- powder diffraction experiment requires reading a 10 page manual (20mins), the experimenter happy enough to be left alone after day one

Users have separate user accounts containing their own batch files. The SINQ Instrument Control System (SICS) server is configured to read those batch files.

Technical Input / Opinion
A graphical user interface could be created to operate standard methods via predefined batch files for survey instruments.
The web interface can have the look and feel of a training manual since it can combine simple controls with guiding documentation and textual feedback on the fly.

2.3.4 The ANSTO Interfaces (Andrew Studer)
MRPD - QuickBasic Interface
- features preview of batch commands, effective for debugging batch runs
- quick to modify and enhance functionality by coding
- not multi-tasking. This is an issue for slow control when feedback between the PC and controller is vital
- requires multiple data processing steps
- awkward to process raw data and plot for review

HRPD, Longpol (and Single Crystal Diff until recently) - graphical user interface
- instant plots
- immediate time status of current scan
- multi-window batch creation is more awkward than MRPD
- used the same library of interdependent modules - hard to modify functionality and prone to code breakage

SANS - IDL graphical user interface
- simple to use (passes the 10min to learn test)
- to date, the slow control is not integrated
- developed by independent attempt for an intuitive look and feel

TAS - developing VB graphical user interface
- code using C libraries and VB graphical user interface
- very simple, deliberately so but cannot continue to stay simple due to complexity of instrument
- desire to implement a clear process for alignment of samples ("wizard"). It is particularly important for external paying users that alignment is rapid and straightforward
- utilising a combined hardware/software solution for alignment and measurement of sample object. Samples are frequently complex shapes: the plan is to scan the object using a laser position measurement system into a common data format that would be used in residual stress measurements and finite element analysis.
- creation of object model can be outsourced locally for approximately $100
- other methods of model creation feasible - as long as user does not have to concern themselves with the process

Generally
- Four interfaces have been developed. An attempt to combine a user interface for three instruments failed.
- New interfaces are being planned for TAS and Reflectometer, which aim to make greater user of developed code and techniques.

2.3.5 Discussion: Instrument Control, Instrument Status and Interface Requirements

2.3.5.1 Beam Instruments

User Perspectives:
The aim has been to present concise information about the instrument (setting/status/configuration). The user interface developers should consider making effective use of the common elements of look and feel evolved by commercial developers over a considerable time.

Considerations for a solution:
- specific solution elements for each instrument
- user interface should be guiding and intuitive
- solution must be developed in consultation with users
- Use Spector (Oak Ridge National Laboratory) as a model for fully graphical solution for SANS application
- integrate data display and instrument status into user interface
- implement sufficient data analysis and reduction to allow decisions to shape the direction of the experiment
- common elements should be implemented such as standardised menus, and a STOP button in the bottom right-hand corner
- standardised and descriptive nomenclature should be used in all labelling

The overall look and feel should be common across instruments, without jeopardising the individual nature and complexity of each instrument. Some instruments will require entirely different functionality.

The user interface could be made modular, and easy to configure.

(One example of this design called Acid<br><http://www.sonicfoundary.com> )

Features:
- combined tabbed form approach with persistent window panes containing orthogonal information
- guidance as to important information without limiting flexibility
- common look and feel

All instrument configuration information should be available in some way, independent of what is stated as required by the user. Conversely, the availability of all the possible configuration parameters should not be thrust upon the user.

Access to infrequently used parameters is commonly achieved via command line interface.

The desire to be elegant and efficient should not divert effort away from the primary aim of controlling the experiment reliably.

Additional scripting/programmatic interfaces for users to customise control should be as lightweight as possible.

A graphic of the instrument showing its status is desirable, and may have training value, and be useful in the simulation mode. Graphic models displaying status should not be a priority.

Knowing the configuration of the instrument at a glance is important. This was achieved at Oak Ridge National Laboratory with the use of two computer screen, one to control the instrument, and one to display status fig 3.1. Many technologies exist to implement these displays.

Another paradigm to display instrument status is the display of a tree hierarchy in a left-hand pane, with selection displaying relevant information in the right-hand pane. Multiple views of an instrument are possible and are not mutually exclusive.

**International Facility Experience**

Keep distinct these aspects of user interfaces
- operational control by the user
- configuration by (or with the aid of the) instrument manager

The command line interface is the tool of choice for implementing a range of operations not available from the graphical user interface, such as uncommon configuration parameters, direct communication to component devices and commands that have not been implemented for the standard user.
Technical Input / Opinion

A web-based solution can be used for both local and remote user interfaces. This reduces requirements to distribute clients, support multiple clients and allows the delivery of documentation as part of the application.

Java Web Start is available for simplifying deployment of Java applications.

Standardisation can be accommodated at the modular level, the application selecting the modules it requires from a common set of independent modules.

The instruments will have common components. Functional analysis of each instrument requirements can be used to determine which modules can be shared between instrument user interfaces. eg 1D and 2D plots

An embedded programmatic interface, in the way that VBA is available to MS Office applications, may be desirable for the more sophisticated user.

A web user interface can allow user written code to interface your system. Provide an API for people to write their own clients.

Even basic server-side scripting is unpopular with scientists. Multiple interface technologies for development tools and debugging are a maintenance nightmare.

Expert systems (decisions made by the software to guide the user) may be a future feature to assist in ease of use (eg start when event count over limit, reverse series of scans when a condition is met.). This is seen as desirable but not practical to specify by scientists - not a priority.

2.3.5.2 Reactor Control and Monitoring System (RCMS)

Parameters from the Reactor Control and Monitoring System that are desirable at instrument user interface

- status of reactor
- reactor power
- primary shutter status
- warnings
- moderator temperature

Parameter should be available for instrument control software to read.

These parameters are also required in the instrument scientist offices.

For numerical parameters that could be logged alongside experimental data, they should be updated periodically - period in the order of seconds.

These parameters should also be separately logged and displayed on a status screen mounted in the guide hall.

2.3.5.3 Other Parameters and Interfaces

Other parameters desirable at instrument user interface

- background monitoring information
- status of other instruments in guide hall

2.3.5.4 Prototyping and the Development Process

Prototyping can be investigated by simulation of the interface in a number of stages, each with increased functionality.

Draft user interface images can be expected as part of the draft proposal on 10th February 2003.

The human resources to further develop these have yet to be hired.

User Perspectives
Simulation is the best way to test the usability. Draft images and later functional simulations can be published and surveyed for feedback.

Aim to find, learn about and adopt best practices from user experiences and existing/available software.

Adopt an iterative process of user interface development and feedback starting as soon as possible. The process should be recommended in the Workshop Report.

Web cams of the experiment are desirable.

**International Facility Experience**

In the user interface development process users will either ignore your efforts or flood you with change requests. Use a development system that allows rapid prototyping and changes. Respond quickly to real user feedback.

ISIS believe that prototyping should include some part of the solution system. Get the prototype to do something. This process will tend to uncover requirements for the user interface. Where possible, try and prototype the system on available components or instruments.

The Paul Scherrer Institut found that real problems and feedback come when the instrument is being worked.

### 2.3.6 Closeout

The priority of any user interface is that it should be functional.

There is particular concern as to the look and feel of this collection of graphical user interfaces.

**Graphical user interface requirements**

- Represent function of instrument
- The user interface should have transparency of purpose,
- Usability of user interface - its look and feel should be familiar - some even strongly guiding,
- Any user interface should have two modes of operation: direct/immediate and batch. It should also include a command line interface as an escape mechanism.
  - Real-time visualisation (as collected)
  - Real-time data reduction (sufficient to guide experiment)
  - Real-time data analysis (sufficient to guide experiment)
- Standardisation without compromising the functionality of individual instruments.
- Flexibility
- Accessibility to configurable parameters
- Expandable/extensible interface
- Ideally capable of following rule-based decisions
- Survey existing standards/styles/models in the production of prototype screens.
- Make simulations of user interface available for trialing.
- Iterate user interface development on basis of user feedback, quickly

**Implementation models to consider**

- web user interface
- menu driven user interface
- virtual instrument (including, but not limited to the LabVIEW paradigm)
- visual programming model
- command line interface
- multiple windowing model
- tabbed-form model
- graphical instrument (mimic)
- configurable user interface (text + graphical + command line interface changeable by user)
- fixed-pane orthogonal-information model
- tree model
- macro model
2.4 Data Reduction and Analysis

The primary importance of selecting appropriate data reduction and analysis software and supporting it is so that the users of the facility can be assured of taking useful data and experience away from their visit.

2.4.1 Site visits and the Australian Neutron Beam Users Groups (ANBUG) survey (Nick Hauser)

The workshop should aim to indicate:
- which data manipulation/analysis packages should be provided
- what are the requirements of these packages
- which packages should be integrated into user interfaces

User Perspectives

The range of major packages has to be made available, whether supported or not.

One package should be integrated (to some degree) into the user interface for each instrument to allow for simplified analysis.

For potential standardisation with the synchrotron facility, the provision of compatible data refinement software is most important.

International Facility Experience

As an international user facility you must provide access to the major packages - whether or not you adopt one of them as an internal standard.

Consider SciLab - a Matlab-like public domain package.

2.4.2 Data Analysis for Powder Diffraction Experiments (Brett Hunter)

Three types of analysis requirements
- Analysis away from the instrument
- Analysis near the instrument
- Analysis at the instrument

Analysis away from the instrument is driven by the science rather than the instrument, is not instrumentation specific and is ongoing development. These packages should be available now.

Analysis near the instrument is by software not necessarily incorporated into the instrument software, but provided by nearby access packages so that data reduction and visualisation can be done in parallel with experiment execution. The software may be on the same terminal.

Analysis at the instrument is primarily for data visualisation, real-time refinement of the experiment, and for monitoring the collection scheme including statistical parameters of the experiment.

User Perspectives

It is very important to have access to serious scientific analysis packages at this Institute - as a user facility it needs to:
- provide some familiarity for users who are not primarily physicists
- provide a library of packages, preferably with someone to guide the experimenter in their use
- facilitate the publishing of results
Each instrument scientist needs to establish and encourage a user community, and establish the analysis needs of that community. Each Instrument Advisory Team (IAT) should be made aware of this.

On-line access to materials/crystallographic databases should be considered. Visualisation software for crystal structures should be included.

Doing a degree of modelling whilst performing the experiment aids decision making during the experiment.

A common data format with access to methods for format conversion would facilitate these requirements transparently. The need is to support easy transportation of data.

**Technical Input / Opinion**

Away from the instrument, the software available should be whatever the user wants.

It is always more productive to do statistical analysis and visualisation of experiment data on the fly, in parallel with the experiment.

### 2.4.3 Single-Crystal Data Analysis (Garry McIntyre)

Presented principally were data-analysis methods appropriate to the Quasi-Laue Diffractometer (QLD), but mention was made of methods appropriate to 2TanA or to TAS when used for elastic scattering.

General types of single-crystal experiments:

**A.** Collection of many Bragg reflections under fixed conditions (T, P,...) for structure solution or refinement

**B.** Volumetric mapping of reciprocal space, especially to search for superlattice or incommensurate satellite reflections

**C.** Study of individual profiles in 1- 2- or 3-D.

**D.** Following a few reflections under an external influence through phase transitions

QLD will be especially suited for types **A** and **B**; 2TanA or TAS for **C** and **D**.

Flow of data reduction and analysis:

- Immediately after the start of the experiment we must visualise the reflections to check the crystal quality, determine the orientation matrix and the total neutron count for a given counting time.

- If the quality is acceptable, the orientation is fed back to determine the strategy of moving through reciprocal space, and the counting times adjusted according to the desired statistics. Such feedback may be made several times during the experiment, and it is important that such preliminary analysis be on-line and fast.

- At the end of the experiment, final reduction and analysis is made to obtain the most accurate structure factors, and then coordinates, bond lengths, spin directions as appropriate. As computing power continues to improve, the sophistication of the preliminary quantitative analysis should improve to the point that post-experiment analysis is rarely necessary.

Processing of ‘conventional’ quasi-Laue data is well catered for by the X-ray Laue suite from the Collaborative Crystallographic Project, CCP4 ([www.dl.ac.uk/SRS/PX/jwc_laue/laue_top.html](http://www.dl.ac.uk/SRS/PX/jwc_laue/laue_top.html)). Options presently missing include:

- Magnetic space groups
- Easy generation of positions of incommensurate reflections
- Pattern manipulations, eg. differences
- Easy lattice manipulations
- More interaction with diffraction pattern. eg spot identification
- Plotting of profiles along arbitrary lines
- Crystal shape and size from reflection forms
- Local multi-peak fitting to recover overlapping reflections

For volumetric mapping of reciprocal space, the present software for QLD only covers locating new reflections.

Full or even partial 2-D pattern fitting is not yet available, and in view of the total number of pixels (8 million) full 2-D pattern fitting is not even feasible.

Repeated analysis of multiple patterns (QLD) or scans (2TanA) to extract the integrated intensities, peak widths etc. is already quite straightforward now, often via common profile-fitting packages (Igor, Origin, Matlab, ...).

Post-experiment analysis should be facilitated by readily available crystallographic software (shelx, xtal, XD, GSAS for conventional structures; CCSL, FullProf for magnetic structures, CCSL, JANA for incommensurate structures).

In summary, single-crystal data analysis software must be:
- on-line, fast to provide guidance for the experiments' strategy
- intuitive to use
- offer input data formats (CIF) to standard crystallographic packages

We should aim for full multi-pattern refinement of QLD data available at or near the experiment.

User Perspectives

Powder experiment users would also like refinement on-line.

2.4.4 Discussion: Data reduction and Visualisation (Andrzej Radlinski)

The user wants to:
- walk out of the facility with useful data
- complete their experimental program
- form at least a preliminary analysis of the data
- be in control of their experiment (rather than endure the technicalities of measurement)
- have support available - before, during and after the experiment

The SANS process
1. Acquire 2D calibration/transmission data
2. Acquire 2D raw data
3. Reduce data to absolute units
4. Reduce absolute 2D data to 1D if appropriate
5. Analyse the data: plot, peruse and fit preliminary models
6. Decide whether the experiment is complete, or return to step 2.

It is desirable to perform this loop during the experiment.

SANS Issues
- access to useable graphical user interface
- availability of data reduction/analysis packages
- capability to do simple modelling

The Major Requirement: To have the software available to concentrate on interactive analysis (step 5) whilst data acquisition and reduction is intelligently automated to occur in real-time.

The Major Challenges
- To design software that will firmly guide the inexperienced or infrequent user status.
- To keep the system architecture subordinate to the experiment

From SANS Experience

- The use of absolute units should be enforced (at the user interface level) due to the diversity of the user community.
- The primary stumbling block to efficient operation of experiments and data analysis has been software integration.
- Endeavour to make sure that people walk out (of the facility) with useful data - no matter what!
- Make sure that the instrument configuration is accessible and can be easily changed.
- Data reduction to absolute units can be done in batch mode.
- There is a preference for simple data format: Three column ASCII format of 1D reduced files \{Q, I(Q), ΔI(Q)\}
- There are standard ways of looking at data and a library of simple routines should be built.

User Perspectives

- There is a tremendous degree of commonality and features between SANS and powder diffraction experiments.
- Orientation of SANS samples implemented by sample changing equipment would ideally be a feature of the control - it is a simpler process than in crystallography. (The new instrument specifies multi-sample handling.)
- SANS (low dimensional diffraction) does not necessarily require knowledge of crystallography. Larger dimensional crystallography software would be required for other instruments.
- The reduction of 2D samples (absolute units) to 1D data sets (which are analysed) could be automated for simple cases and would be useful. The process is more complex when interactively fitting models.
- We want many packages available in a library. As we gain experience, we may gravitate to a simpler set of tools.
- In the case of providing a choice between multiple packages on multiple experiments, make one package the default to avoid having to repeat the selection decision.
- Any file conversion utilities must be tested.

2.4.5 Discussion: Performance Requirements of Data Reduction and Analysis Packages

Some packages are available on only one platform.

Windows and Linux are supported on site, Silicon Graphics to a lesser extent. Some crystallographic software is available only on Apple platforms (ANSTO does support Apple platforms. Would require local support).

User Perspectives

All instruments should provide appropriate data reduction. Currently available packages will be used as the baseline for the specification.

Current ANSTO policy on supported platforms is seen as a practical limitation and is not acceptable to the users. Porting of hardware dependent packages should be explored.

Users and the Instrument Advisory Teams (IAT) should influence software development of analysis packages to be cross-platform software eg. the most common platform for TAS analysis is the Apple platform, and currently the best analysis is by Monte Carlo, which is slow.
Access to any third party software must be provided in some manner - even if this involves providing stand-alone machines for visitor use (For example, Macintosh and Silicon Graphics).

For most computer literate scientists, any perceived lack of freedom in computing environment is not desirable.

When there is little choice for the package/platform it may become the users' responsibility to analyse the data. The basic data formats still need to be defined and provided.

We need to identify if any particular ports of packages are required.

Obscure packages need not be fully supported.

Remote/visitor access to networks using the visitor's computer will allow visitors to use their own packages. Any ANSTO restrictions may deter use of the facility and not desirable to the community. The community is accustomed to "free" access, and insists on this type of access.

ANSTO direction for user support is toward centralised 24/7 support. If this is seen as too restrictive, we can make a case for alternatives in our proposal. Anything less than 24/7 support is not acceptable.

**International Facility Experience**

Silicon Graphics platforms pose unique development problems. Silicon Graphics needs Fortran. At the Institute Laue Langevin, the diffraction group and others, use Silicon Graphics for data reduction, but will be porting software to PC/Linux platforms to reduce costs.

At least one data reduction/analysis package will be required for each instrument. Also provide access to each of the major packages offline.

When considering remote access/control, the provision of on-line data reduction may be restricted. It is possible to make use of an analysis server, but this requires significantly more development.

Analysis, remotely or locally, cannot always be simultaneous with instrument control. Aim for transparent access to the data for analysis.

The Paul Scherrer Institut TAS uses Linux/octave. We see that restriction to platform is reasonable.

ANSTO will require either central 24/7 support for system & network management tasks, or a local group member who has the capability to administer the local network.

### 2.4.6 Closeout

#### 2.4.6.1 Data Reduction

Definition of data reduction is the manipulation of data, including

- standard normalisation
- background subtraction/correction
- calibration/transmission correction
- instrument geometry correction

Data reduction is the model-free process to produce data that can be readily accepted into analysis packages for standard methods of analysis.

Basic data reduction should be integrated into every instrument control user interface, transparently. This could also be applied to analysis packages where possible.

Community expects the capability in the user interface to import/export data to at least one adopted data reduction/analysis package, using the appropriate filters.

The data archive needs to retrieve the raw data at least. The Paul Scherrer Institut does not store processed data. Processed data can be generated from the raw data.
Where the instrument produces an inordinate amount of raw data, the NeXus format is still capable of storing the processed result sets. Therefore, if there is a good reason to implement it this way, do it.

**2.4.6.2 Data Visualisation**

Data visualisation
- is required for 0D, 1D, 2D, 3D and 4D data sets
- needs to be integrated into graphical user interface
- needs to be available post collection
- could provide preliminary fitting such as peak measurement

Writing data visualisation software is currently out of the scope of the Neutron Beam Instruments Project (NBIP).

**2.4.6.3 File Naming Schemes**

File naming schemes:
- needs to be defined, and implemented for automatic operation
- file names could be customisable to some extent
- names unique within the system, and across facilities if possible
- consider storage name versus user specified name

**2.4.6.4 Data Analysis**

Data analysis
- hardware platforms are required that support each of the major analysis packages, which currently includes non-MS Windows or Linux
- unique systems may be stand-alone
- state-of-the-art software is essential, whether provided at the facility or the user is allowed to supply on their own machine
- the range of packages for each instrument is to be recommended by the Instrument Advisory Teams (IAT)
- there is a standard set of analysis environments that should be provided - recommend at the workshop: IDL, Matlab, IGOR, SciLab, LAMP
- The ISIS experience. Pick one data analysis environment and stick to it.
- The Paul Scherrer Institut experience. Specify site-wide standard packages and also provide at least one licence for all of the major packages.

**2.4.6.5 Remote Access**

- Remote access and its impact on reduction, analysis and visualisation. Consider
- technical issues (see comments in Session 3)
- license issues

**2.4.6.6 Scientific software/programming support**

Scientific software/programming support
- external users should be able to expect some level of support for scientific software packages
- support from instrument scientist or an on-site computing group

At the Paul Scherrer Institut, support requests for scientific software/programming end up on the computing group's desk rather than the instrument scientist's. It is efficient from the instrument scientists' point of view to get a good programmer to fix a problem.

At ISIS, often the intractable problem for a scientist may be a simple programming fix.

The support person would deal with the instrument scientist rather than the user.

Having those software support resources could be very valuable, but those resources are personal-skill-based. Support is a question of communication between the scientist with the problem and the programmer or scientist with the knowledge to fix it. It would be good to have a systems programmer available to take care of systems problems, leaving the science-based problems to the scientists.
3 Conclusions

The user community at the workshop made the following list of requirements.

- Connection of the user's laptop to a network for data transfer and analysis, and Internet connection for retrieval of files and e-mail from their home institution. One way to achieve this is via a local area network physically separate from ANSTONET
- Remote access (Internet access) to instruments for instrument control, instrument status, data access and data analysis
- Read only access to some variables from the Reactor Control Management System. Reactor power and moderator temperature
- ANSTO administered user accounts with data security and privacy
- Data format will be NeXus with converters to a prescribed range of frequently used formats
- Computing support comparable to other neutron scattering facilities
- Data storage and accessibility via the Internet for 10 years or longer.
- Web cameras for instruments, accessible over the Internet
- At least one license of Matlab and one for IDL. Investigate Scilab and Octave as open-source alternatives
- Stable, reliable control systems
- Operating system independent client and server for instrument control (Linux or Windows)
- Customised interfaces for each instrument, with similar "look and feel". Things that are the same look the same eg. temperature controllers
- User involvement in the interface design
- Interfaces that guide new users
- Real-time data visualisation of data for some instruments
- Transparent proposal system and scheduling database for university and non-university organisations
- Provide indexing and retrieval mechanism for data files
- Simulation mode for instruments, for training and script verification
- User training
- A plethora of analysis packages that are working, with no requirement for training support.
## Appendix A

### List of participants

<table>
<thead>
<tr>
<th>Participant</th>
<th>Organisation</th>
<th>e-mail</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SANS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Andrzej Radlinski</td>
<td>Geoscience Australia</td>
<td><a href="mailto:Andrzej.Radlinski@ga.gov.au">Andrzej.Radlinski@ga.gov.au</a></td>
</tr>
<tr>
<td>Alan Hinde</td>
<td>Geoscience Australia</td>
<td><a href="mailto:Alan.Hinde@ga.gov.au">Alan.Hinde@ga.gov.au</a></td>
</tr>
<tr>
<td><strong>Quasi-Laue Diffractometer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wim Kloosters</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:wim@ansto.gov.au">wim@ansto.gov.au</a></td>
</tr>
<tr>
<td>Garry McIntyre</td>
<td>Institute Laue Langevin</td>
<td><a href="mailto:mcintyre@ill.fr">mcintyre@ill.fr</a></td>
</tr>
<tr>
<td><strong>Residual Stress Diffractometer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oliver Kirstein</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:oki@ansto.gov.au">oki@ansto.gov.au</a></td>
</tr>
<tr>
<td>Richard Blevins</td>
<td>ANSTO-Materials</td>
<td><a href="mailto:rbl@ansto.gov.au">rbl@ansto.gov.au</a></td>
</tr>
<tr>
<td><strong>3-axis Spectrometer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Leo Cussen</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:lcu@ansto.gov.au">lcu@ansto.gov.au</a></td>
</tr>
<tr>
<td>Trevor Finlayson</td>
<td>Monash University</td>
<td><a href="mailto:trevor.finlayson@spme.monash.edu.au">trevor.finlayson@spme.monash.edu.au</a></td>
</tr>
<tr>
<td>Andrew Studer</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:ajs@ansto.gov.au">ajs@ansto.gov.au</a></td>
</tr>
<tr>
<td><strong>Reflectometer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mike James</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:mja@ansto.gov.au">mja@ansto.gov.au</a></td>
</tr>
<tr>
<td>Jeremy Ruggles</td>
<td>University of Queensland</td>
<td><a href="mailto:j.ruggles@mailbox.uq.edu.au">j.ruggles@mailbox.uq.edu.au</a></td>
</tr>
<tr>
<td>Ian Gentle</td>
<td>University of Queensland</td>
<td><a href="mailto:gentle@chemistry.uq.edu.au">gentle@chemistry.uq.edu.au</a></td>
</tr>
<tr>
<td><strong>Powder Diffractometers</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Brett Hunter</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:bah@ansto.gov.au">bah@ansto.gov.au</a></td>
</tr>
<tr>
<td>Evan Gray</td>
<td>Griffith University</td>
<td><a href="mailto:E.Gray@sct.gu.edu.au">E.Gray@sct.gu.edu.au</a></td>
</tr>
<tr>
<td><strong>Polarisation Analysis Spectrometers</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Michel Ain</td>
<td>CEA Saclay</td>
<td><a href="mailto:ain@cea.fr">ain@cea.fr</a></td>
</tr>
<tr>
<td><strong>Project Coordination Group (NBIP)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shane Kennedy</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:sjk@ansto.gov.au">sjk@ansto.gov.au</a></td>
</tr>
<tr>
<td>Rob Robinson</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:rro@ansto.gov.au">rro@ansto.gov.au</a></td>
</tr>
<tr>
<td>Sungjoong Kim</td>
<td>ANSTO-Engineering</td>
<td><a href="mailto:skx@ansto.gov.au">skx@ansto.gov.au</a></td>
</tr>
<tr>
<td><strong>Neutron Beam Instruments Project (NBIP) Engineering</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peter Baxter</td>
<td>ANSTO-Engineering</td>
<td><a href="mailto:pab@ansto.gov.au">pab@ansto.gov.au</a></td>
</tr>
<tr>
<td>Glen Horton</td>
<td>ANSTO-Engineering</td>
<td><a href="mailto:gih@ansto.gov.au">gih@ansto.gov.au</a></td>
</tr>
<tr>
<td><strong>Other Bragg Institute personnel</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ross Piltz</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:rop@ansto.gov.au">rop@ansto.gov.au</a></td>
</tr>
<tr>
<td>Rob Knott</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:rbk@ansto.gov.au">rbk@ansto.gov.au</a></td>
</tr>
<tr>
<td>Jamie Schulz</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:jamie.schulz@ansto.gov.au">jamie.schulz@ansto.gov.au</a></td>
</tr>
<tr>
<td>Nick Hauser</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:nha@ansto.gov.au">nha@ansto.gov.au</a></td>
</tr>
<tr>
<td>Paul Hathaway</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:pvh@ansto.gov.au">pvh@ansto.gov.au</a></td>
</tr>
<tr>
<td>Yang Fei</td>
<td>ANSTO-Bragg Institute</td>
<td><a href="mailto:yfx@ansto.gov.au">yfx@ansto.gov.au</a></td>
</tr>
<tr>
<td><strong>VISITING SPEAKERS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mark Koennecke</td>
<td>Paul Scherrer Institut</td>
<td><a href="mailto:Mark.Koennecke@psi.ch">Mark.Koennecke@psi.ch</a></td>
</tr>
<tr>
<td>Chris Moreton-Smith</td>
<td>ISIS</td>
<td><a href="mailto:C.M.Moreton-Smith@rl.ac.uk">C.M.Moreton-Smith@rl.ac.uk</a></td>
</tr>
<tr>
<td>Chris Relf</td>
<td>crelf Consultants</td>
<td><a href="mailto:Christopher.Relf@mbox.com.au">Christopher.Relf@mbox.com.au</a></td>
</tr>
<tr>
<td><strong>Australian Institute of</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nuclear Science and Engineering (AINSE)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Dennis Mather</td>
<td>AINSE</td>
<td><a href="mailto:ainse@ansto.gov.au">ainse@ansto.gov.au</a></td>
</tr>
<tr>
<td>ANSTO Information Management Staff</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Greg Doherty</td>
<td>ANSTO-IM</td>
<td><a href="mailto:gdz@ansto.gov.au">gdz@ansto.gov.au</a></td>
</tr>
<tr>
<td>Glynn Peady</td>
<td>ANSTO-IM</td>
<td><a href="mailto:glynn@ansto.gov.au">glynn@ansto.gov.au</a></td>
</tr>
<tr>
<td>Jeff Horton</td>
<td>ANSTO-IM</td>
<td><a href="mailto:jph@ansto.gov.au">jph@ansto.gov.au</a></td>
</tr>
<tr>
<td>Friedl Bartsch</td>
<td>ANSTO-IM</td>
<td><a href="mailto:fkb@ansto.gov.au">fkb@ansto.gov.au</a></td>
</tr>
<tr>
<td>Jagoda Crawford</td>
<td>ANSTO-IM</td>
<td><a href="mailto:jc@ansto.gov.au">jc@ansto.gov.au</a></td>
</tr>
</tbody>
</table>
Appendix B  Program

Program for Workshop on Computing Requirements for the Neutron Beam Instruments Project
9-10th December 2002

Building 55 Conference Room.
Lucas Heights Science and Technology Centre.
Old Illawarra Road, Lucas Heights. Sydney.

Monday, 9th December 2002

<table>
<thead>
<tr>
<th>Time</th>
<th>Session Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.00-9.15am</td>
<td>Welcome: Rob Robinson.</td>
</tr>
<tr>
<td>Facilitator:</td>
<td>Instrument control computer systems - the generic part of instrument control.</td>
</tr>
<tr>
<td>Rob Robinson</td>
<td>What is to be controlled on the instrument?</td>
</tr>
<tr>
<td></td>
<td>relevant to this session Presentation &amp; discussion. Nick Hauser</td>
</tr>
<tr>
<td>9.45-10.45</td>
<td>Overview. The ISIS solution, past, present and future. Chris Moreton-Smith</td>
</tr>
<tr>
<td></td>
<td>The Paul Scherrer Institut solution, past, present and future. Mark Koennecke</td>
</tr>
<tr>
<td>10.45-11.00</td>
<td>Break</td>
</tr>
<tr>
<td>11.00-11.30</td>
<td>Motion control survey. Presentation &amp; discussion. Leo Cussen</td>
</tr>
<tr>
<td>11.30-12.00</td>
<td>Detection and data acquisition electronics survey. Presentation &amp; discussion. Brett Hunter</td>
</tr>
<tr>
<td>12.00-12.45</td>
<td>User and IAT requirements Discussion. Shane Kennedy.</td>
</tr>
<tr>
<td>12.45-13.30</td>
<td>Lunch</td>
</tr>
<tr>
<td>13.30-14.15</td>
<td>Closeout</td>
</tr>
<tr>
<td>Time</td>
<td>Session Details</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>14.15-14.45</td>
<td>Introduction and context&lt;br&gt;Collated information from visits&lt;br&gt;ANBUG Survey results relevant to this session&lt;br&gt;Presentation &amp; discussion. Nick Hauser</td>
</tr>
<tr>
<td>14.45-15.15</td>
<td>Overview.&lt;br&gt;Presentations &amp; discussion&lt;br&gt;Standard formats for neutron beam data. Mark Koennecke (20 mins)</td>
</tr>
<tr>
<td>15.15-15.35</td>
<td>What data format do you want?&lt;br&gt;Discussion. Facilitator: Shane Kennedy</td>
</tr>
<tr>
<td>15.35-15.50</td>
<td>Break (15mins)</td>
</tr>
<tr>
<td>15.50-16.20</td>
<td>Remote instrument &amp; data access.&lt;br&gt;Presentations &amp; discussion. Greg Doherty &amp; Mark Koennecke</td>
</tr>
<tr>
<td>16.20-16.40</td>
<td>ANSTO networks &amp; data storage.&lt;br&gt;Presentations &amp; discussion. Friedl Bartsch</td>
</tr>
<tr>
<td>16.40-16.50</td>
<td>What are the requirements for a proposal system?&lt;br&gt;Presentations &amp; discussion. Dennis Mather</td>
</tr>
<tr>
<td>17.00-17.45</td>
<td>Closeout. Facilitated by Mike James</td>
</tr>
</tbody>
</table>

**Monday Evening – Dinner at “Stapleton's”**

Stapleton's Restaurant<br>782 Pacific Highway, Sutherland.<br>9521 8747

**Tuesday, 10th December 2002**

<table>
<thead>
<tr>
<th>Time</th>
<th>Session Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.00-9.15</td>
<td>Welcome. Rob Robinson.</td>
</tr>
<tr>
<td>Facilitator:</td>
<td>User interfaces - the instrument specific part of instrument control</td>
</tr>
<tr>
<td>Evan Gray</td>
<td></td>
</tr>
<tr>
<td>9.15-9.45</td>
<td>Introduction and context&lt;br&gt;Collated information from visits&lt;br&gt;ANBUG Survey results relevant to this session.&lt;br&gt;Presentation &amp; discussion. Nick Hauser</td>
</tr>
<tr>
<td>9.45-10.45</td>
<td>Overview.&lt;br&gt;The ISIS interface. Chris Moreton-Smith&lt;br&gt;The Paul Scherrer Institut interface. Mark Koennecke&lt;br&gt;The ANSTO interfaces. Andrew Studer</td>
</tr>
<tr>
<td>10.45-11.00</td>
<td>Break</td>
</tr>
<tr>
<td>Time</td>
<td>Session Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>11.00-12.00</td>
<td>Instrument control Instrument status Interface requirements Discussion: Shane Kennedy</td>
</tr>
<tr>
<td>12.00-12.30</td>
<td>Process: How are we going to design user interface Discussion: Mike James</td>
</tr>
<tr>
<td>12.30-13.15</td>
<td>Lunch</td>
</tr>
<tr>
<td>13.15-14.00</td>
<td>Closeout Facilitated by Evan Gray</td>
</tr>
<tr>
<td>Facilitator:</td>
<td><strong>Data reduction and analysis</strong></td>
</tr>
<tr>
<td>Robert Knott</td>
<td></td>
</tr>
<tr>
<td>14.00-14.30</td>
<td>Introduction and context ANBUG Survey results relevant to this session Presentation &amp; discussion: Nick Hauser</td>
</tr>
<tr>
<td>14.30-15.10</td>
<td>Overview Rietveld Brett Hunter (20mins) Single Crystal Data Analysis Garry McIntyre (20mins) Presentation &amp; discussion</td>
</tr>
<tr>
<td>15.10-15.25</td>
<td>Break</td>
</tr>
<tr>
<td>15.25-15.55</td>
<td>Data reduction &amp; visualisation Presentation &amp; discussion: Andrzej Radlinski</td>
</tr>
<tr>
<td>15.55-16.15</td>
<td>What are the performance requirements of your analysis package Discussion: Wim Klooster</td>
</tr>
<tr>
<td>16.15-17.00</td>
<td>Closeout Facilitated by Robert Knott</td>
</tr>
<tr>
<td></td>
<td>CONCLUDING REMARKS AND WORKSHOP CLOSE: Rob Robinson</td>
</tr>
</tbody>
</table>
I thought the workshop went well and explored the most important issues from the point of view of the users.

As I wasn't able to attend on the first day, I wanted to make some comments about the access issue.

It is all too easy to build an excellent neutron beam instrument and then cripple it by not providing commensurately excellent services to support it.

Access, both physical and electronic, are always hot issues, and ANSTO needs to look very hard at its current method of operation if the new facility is to get anywhere near best practice in these areas.

To me, the following points define best practice in relation to electronic access, according to my experience of what is available to various degrees in other world-class facilities:

- Connection to the network (let's call it the neutron local area network) which hosts the computers for instrument control, data collection and data analysis must be possible from outside the ANSTO network.
- Authorised users off site or anywhere on site must be able to see their instrument control desktop and perform "safe" remote control functions.
- ANSTO-supported software to carry out data reduction (ie. preliminary treatment before analysis) and analysis must be accessible via the network from outside ANSTO.
- For the efficient conduct of an experiment is essential that users be able to connect their own computers to the neutron local area network while at the facility and have the functionality outlined above.
- This connection must be hassle-free.
- Because the facility will run 24 hours per day, there must be a connection point in every room of the on-site accommodation block (let's not dignify Stevens Hall by calling it a Motel) so that users can connect their own computers to continue working or monitoring/controlling an experiment away from the neutron beam facility.

I attended the sessions on 10th December and facilitated the session “User Interfaces – the instrument specific part of instrument control”

The success of the user interface with users was seen as crucial to the functioning of each instrument. There was a wide-ranging discussion of the models that are implemented elsewhere. There was no consensus on how the interface should be presented in detail, but there was wide support for the “look and feel” to be essentially similar across the facility. The general expectation is of a graphical user interface with some command-line capability for expert or non-standard commands. There was a lengthy discussion of the merits and problems of batch-mode operation via a scripting language, versus a table-based “unwound script” approach.

In the closeout session the following points were made to summarise the feeling of the meeting:

- Instrument function is paramount and must not be compromised by applying standardisation rules blindly.
- The usability of the user interface is crucial. The following aspects were highlighted:
  - the “look and feel”, which contains the ideas of visual quality and user-friendliness;
  - transparency of purpose, which contains the ideas of intuitive and logically sensible operation, and that the user should be guided to operate the instrument easily and correctly;
  - direct and batch modes should be available;
  - a command line should be available.
• The interface should provide data visualisation, data reduction and some level of data analysis (to be recommended by the Instrument Advisory Team (IAT)s) in real time.

• The interface should be flexible:
  □ it should be possible to easily add modules which might view data not normally associated with the instrument, such as goniometer settings on a powder diffractometer, or incorporate data generated by user-supplied instrumentation;
  □ there should be some ability to change the course of the experiment based on the evaluation of data against rules established by the user, such as collect data until a certain goodness of fit is reached in automatic Rietveld refinement or a threshold of temperature/pressure/strain etc is crossed;
  □ can the interface be made configurable by the user?

• Good interface models should be studied. These might be from completely unrelated areas, such as Microsoft Office, Macromedia, Adobe etc.

• A set of perhaps three simulated interfaces should be built on a common functioning layer and made available for user testing via the external network.

• Interface models recommended for consideration were Command line; “Windows”; Tabbed; Graphical instrument (“VI”); WWW interface with hyperlinked help; Fixed “orthogonal” windows (ref AJS); Tree model.
1. Regulatory safety and security software and hardware requirements should be met without compromising the instrumental capabilities as well as the requirements of user community, as benchmarked by leading overseas neutron scattering facilities. These requirements include an easy, web based access to the instrument controls and data by authorised users at any time from anywhere in the world, with data storage being active indefinitely. Raw data should be accessible in read-only format.

2. Overall ease of use and a robust, fool-proof design. The software architecture should be subordinate to the requirements of the experiment at hand in a user environment. System and mathematical routines should be automatic and transparent to the user. Instrument-specific features should be designed and programmed in close consultation with Instrument Advisory Team (IAT) and instrument scientists.

3. A single-stream system for peer review of proposals should be adopted, in line with similar overseas facilities. Free access to instruments should be granted based on scientific merit of proposals. Provisions should be made for cost-recovery industrial research, with appropriate data security measures in place.

In general, to be attractive to the neutron science user community, the facility needs to perform as well as the best of the similar overseas facilities. This will require some major changes in the way that computing is currently managed at ANSTO. The perceived dangers of computer security issues as expressed at the workshop are at best imaginary, as our colleagues from ISIS, ILL and PSI confirmed from their experience. At worst hackers will destroy data, which they could never interpret, so there is a limited attraction. Data archiving is achievable, and with the speed of change in the computer industry, this should be relatively cheap by the time of commissioning the facility in 2006. The instrument scientist should have administrative rights over instrument computers and ancillaries. Although it was explained that ANSTO has a ‘total control’ policy of 24/7/365 support for computing, users generally want to communicate directly with the instrument scientist. Many of the problems encountered on an instrument in the middle of the night require an intimate knowledge of the instrument, associated ancillary equipment, and the operating systems.

Instrument control

- A separate local area network for user environment would enable a different regime to coexist alongside the current ANSTO and Reactor Control and Monitoring System networks. This network should include instrument control, data storage and access to the WWW as well as some minimal information on reactor status.
- Unfettered remote access to this user local area network including use of laptops (auto IP addresses supplied), a login/password system for data and batch file storage and remote access is necessary. By 2006 things will have changed but essentially the network should be easily accessible.
- Unlimited access (no lifetime restriction) to data storage location including FTP/telnet access from outside the facility. Users should be able to login to their instrument from their on-site accommodation.
- On-line manuals necessary (on-screen help preferred)
- Seamless integration of ancillary equipment is essential as many experiments involve specialised vessels or environments.

Data

- Data reduction and some form of real-time data visualisation should be available at the instrument
- The platform (Linux/NT or?) is such that it is stable and flexible enough that communication with users laptop is possible for data transfer for real-time analysis. There must be variety of methods available for data transfer, either over the web or by CD, Zip or memory sticks etc.
- Data storage in the standard form, possibly NeXus, should be self-describing. This description should include time/date, instrument status, and other parameters such as temp, surface pressure etc are automatically stored with file for retrieval at a later date
- All data formats supported, ie. the conversion from NeXus to any recognised format for data analysis package
- One preferred/chosen data analysis package should be available on-line for real-time interactive analysis/experiment loops for each configuration of each instrument. This should be a choice of the Instrument Advisory Team (IAT) with combination of IS and user input.
- A library of commonly used data analysis packages should be available and supported by conversion routines
- Some non PC hardware will be necessary (Macintosh, SUN workstation, Silicon Graphics in a users room) and supported, preferably networked to instrument local area network and web
- Standard programming package available, eg. IDL, Matlab, Scilab etc
- A consistent and obvious file naming system eg. rrr1 – rrr100000000.dat

User Interface
• Consistency throughout facility – things the same look the same over ALL instruments, including the style of the graphical user interface and parts that are common should use the same symbolism. This gives a good impression and is useful for expansion of instrument and the facility.

• Both virtual instrument (graphical, click-to-alter) and command line instrument control should be available. Ideally the batch file should go through a filter which checks for obvious conflicts. This is the most common cause of lost data as the tired users plan a break while the experiment runs unattended.

• A ‘30 minute rule’ should apply for instrument interface

• Simulation mode should be available so that user may do a virtual experiment

• The instrument interface should have the facility to be expandable as the science produced on each instrument is always evolving

• Intuitive/obvious user interface was discussed, should be developed by users and instrument scientists together such that things make sense to a new user

Finally I would like to add some comments regarding personal fears raised by the input from the ANSTO information management representatives, from my past experience with using the current ANSTO facilities, and conflicting experience at several overseas facilities. Users will not tolerate being restricted by ANSTO rules. A minimum level of facilities such as user accounts, use of personal computers on-site, availability of remote access, and CD burners, which are available at other major facilities, MUST be available, otherwise users will be go elsewhere to perform experiments. The scientific community are in general sophisticated computer users/managers who as group tend to work cooperatively (sharing files, programs sometimes addresses). This system of trust is what enables good science as the openness encourages the sharing of ideas. The usual experience of being a user involves several 24 shifts, being a long way from home, one chance to perform work, which has taken several months to plan, means that the negative experiences are highlighted. The site already suffers from some disadvantages: isolated location, poor accommodation, no transport links, and security restrictions far beyond the sensible. The computing should be designed to easily connect the user with his instrument, the data, packages to make all tasks easy and the wider community via the web. The platform and the underlying design of the system needs to have an in-built flexibility so that it can change as the computing hardware and software evolve over the projected lifetime of the facility.

Dr Jeremy Ruggles
Department of Chemistry
University of Queensland