NPL Report CISE 28/99

Report to the National Measurement System Policy Unit, Department of Trade & Industry

INTERIM STATUS REPORT ON THE FOUNDATION METROLOGY AREA

FROM THE SOFTWARE SUPPORT FOR METROLOGY PROGRAMME

BY D. RAYNER
NPL

December 1999

ISSN 1361-407X

Authorised by Dr Dave Rayner,
Head of the Centre for Information Systems Engineering

National Physical Laboratory, Queens Road, Teddington, Middlesex, TW11 0LW

Contents

1. Introduction

The purpose of this status report is to inform the National Measurement System (NMS) Software Support for Metrology (SSfM) programme (1998-2001) about the status of the mathematics and software used to support the **Foundation Programme**. This and the companion status reports for the other metrology areas will guide the formulation of the next SSfM programme (2001-2004). It may also lead to links between the Foundation programme and the SSfM programme.

The SSfM programme is an underpinning programme that provides generic support in the use of software and mathematics to the NMS Programmes for each metrology area. For details of the programme, its expected deliverables and the results already produced, see the SSfM web site: http://www.npl.co.uk/ssfm/.

Those NMS Programmes for specific metrology areas provide metrological support to industry. The SSfM Programme in contrast has relatively little direct impact upon industry, although there is some as evidenced by the SSfM Club membership. The Foundation programme is also an underpinning programme, providing research input to the NMS Programmes for specific metrology areas and has little direct impact on industry. These relationships are depicted in Figure 1. It is because of these relationships that the Status Reports concentrate primarily on the use of software and mathematics in the NMS Programmes rather than in industry.

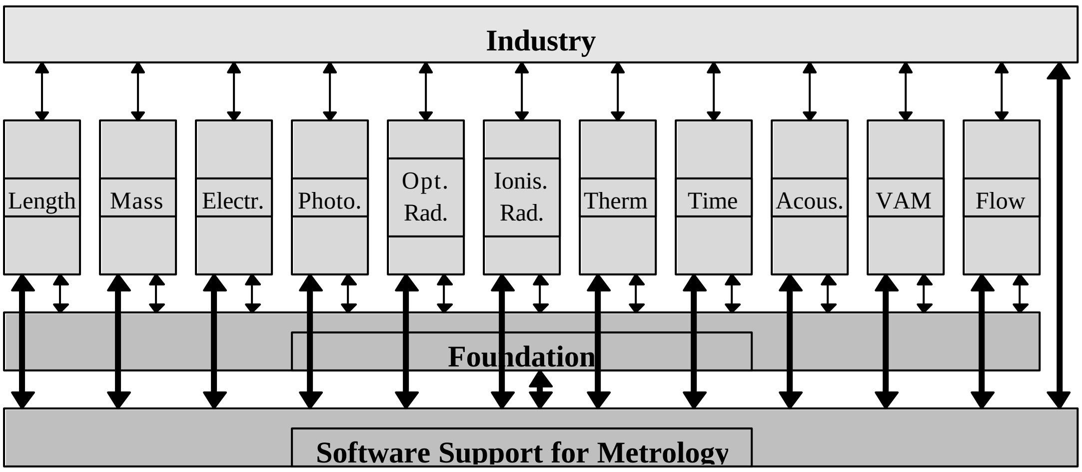

Figure 1. Relationship of SSfM to other NMS Programmes and Industry

In particular, this report addresses each of the themes within the SSfM programme and describes the status concerning the topics covered by each of the relevant projects. It also considers whether there are any important software or mathematics issues in the Foundation Programme which are not addressed by the current SSfM programme or which need to be taken further in the next SSfM programme.

This report is an update of an initial restricted status report produced in December 1998. That initial report was one of a set of restricted status reports which were synthesised into an overall status report for all metrology areas [5]. A summary of the differences from the initial Optical Radiation status report is provided at the end of this report.

2. Scope of Foundation Metrology

Foundation Metrology is a programme of underpinning research to support the top level infrastructure of the NMS and its ability to provide traceability to internationally agreed primary SI units of measurement. It is the area covered by the NMS Foundation Programme, which aims to provide innovative investigations into those aspects of fundamental physical phenomena that have potential impact on the NMS in the long term. It sustains a range of expertise on the science of measurement at the highest level of accuracy in order to meet the evolving NMS needs, contributes to international work on fundamental constants and SI units, and exploits atomic and quantum phenomena and related developments in materials and processes for the next generation of measurement standards.

The Foundation Programme includes work on new primary standards which will move into the mainstream NMS in 3 to 5 years, like the caesium fountain, and more speculative research with a 10 to 15 year lead time, like the quantum roulette noise thermometer and the use of spectral transitions in the hydrogen atom as frequency standards.

This is a programme area in which there are no measurement services.

The programme spans most of the metrology fields, particularly length, mass, time and frequency, electrical, optical radiation, and thermal. Therefore, in some respects there is a lack of homogeneity, with each project using software and mathematics in ways that are independent of the other projects in the Foundation Programme; on the other hand, the fact that it is all leading edge research could give rise to some commonality in the requirements for use of software and mathematics.

3. Modelling Techniques

3.1 Methods for Modelling Measurement Data

3.1.1 Types of data

There are three categories of measurement data sets:

1. discrete measurement data, which means:
   a) data obtained by sampling a discrete variable: e.g., a variable that may take only integer values such as in counting processes, and
   b) data obtained by sampling a continuous variable.
2. continuous measurement signal, which means an analogue signal prior to any analogue to digital conversion, which would result in data of type 1 b).
3. hybrid measurement data composed of both discrete and continuous data sets.

In the Foundation Programme, many of the quantities for which data are collected are continuous variables (such as temperature and time). Sometimes the data that are collected are discrete. However, for example, spectral techniques are used, in which case the raw data is analogue. This analogue data may be used directly as continuous data or may be fed into an A/D converter, resulting in large amounts of discrete data. In addition, occasionally, hybrid data may be needed for some applications.

3.1.2 Modelling

Types of model

There are three categories of model:

1. discrete models in which the outputs of the measurement system are related to measurements of its inputs by a system of algebraic equations;
2. continuous models in which the outputs of the measurement system are related to measurements of its inputs by a system of differential equations;
3. hybrid models in which the outputs of the measurement system are related to measurements of its inputs by a system composed of both algebraic and differential equations.

It is appropriate to distinguish here between the nature of the model and the approach to solving the model (which for a continuous model typically also involves solving algebraic equations).

Given infinite precision arithmetic, the set of equations derived from a discrete model can be solved exactly. This is not the case for a continuous model which typically requires approximations to be made, for example, in terms of domain discretisation.

Models used in the Foundation Programme

There is a wide range of modelling requirements in the Foundation Programme. In many cases the need is to model the basic physics of various processes, and this tends to imply the use of continuous modelling. Nevertheless, there are examples of discrete modelling, and possibly also some hybrid modelling.

The software package Mafia is used for solving Maxwell's equations using finite difference methods to model the resonant frequencies of microwave resonators. This is an example of very complex continuous models being solved by adaptive finite difference methods. Modelling of radiation transfer is performed using Monte Carlo simulation methods developed by Alexander Prokhorov; this could be viewed as continuous modelling solved using discrete methods (although some may prefer to call it hybrid modelling). Electromagnetic modelling, the relatively simple modelling of flow rates in a vacuum system, diffraction modelling using Fraunhofer equations, and the modelling of stresses are all further examples of continuous modelling. On the other hand, thermodynamic temperature, the behaviour of black bodies and the variations in electromagnetic flux in the moving coil kilogram project are modelled by discrete models.

3.2 Uncertainties and Statistical Modelling

In general, the emphasis in the Foundation Programme is not so much on the estimation of uncertainties, but rather on improving precision of measurement. There is nevertheless concern for uncertainties quoted with published results in scientific papers. Also, in work on measuring fundamental constants, there is a need to quote uncertainties, with the emphasis on being careful to identify and minimise systematic errors (type B uncertainties). On the other hand, projects developing novel measurement techniques concentrate on proof of principle, with the estimation of uncertainties tending to come later, when the technology is transferred from the realm of research into mainstream use in the appropriate specific metrology area.

3.2.1 Derivation of statistical models

The major concerns in the Foundation Programme regarding the estimation of uncertainties often do not lie in statistical modelling, but rather in getting a better understanding of the physics to be able to identify where errors are coming into the experiments. Nevertheless, there are concerns with both Type A and Type B uncertainties. There is always concern to achieve the right balance.

3.2.2 Determination of the statistical distribution of measurement results

Generally, the determination of the statistical distribution of measurement results is not a major concern in the Foundation Programme. However, a counter-example is where evidence of non-

Gaussian distributions gives clues to the presence of systematic errors which need to be minimised.

3.2.3 Levels of confidence other than 95%

Where a level of confidence is required, 95% is usually considered to be acceptable, but this is almost always quoted in terms of two standard deviations.

3.2.4 Supporting software

MathCad tends to be used in preference to spreadsheets.

3.2.5 Supporting documentation

The ISO Guide to the Expression of Uncertainty in Measurement is the base document in the area of uncertainty estimation.

UKAS document M 3003 is not used in the Foundation Programme because it is really an interpretation of the ISO Guide for calibration laboratories, but there are no calibration laboratories in the Foundation Programme.

3.2.6 Training

Even though the estimation of uncertainties is not a major activity within the Foundation Programme, some of the personnel, especially those working on the measurement of fundamental constants, are highly competent in the techniques of statistical modelling and the estimation of uncertainties.

3.3 Visual Modelling and Data Visualisation

In the Foundation Programme, data is usually presented visually in the form of graphs, but there is continual improvement to the standard of presentation used. In the moving coil kilogram work, data visualisation is particularly important for the understanding of what is going on. At present this is in the form of 2D graphs, each giving a 2D slice through multi-dimensional data. This is done using Origin, which has been chosen for its speed, enabling the scientist to switch quickly between different 2D views of the data. There is a need for major improvements in several aspects: data formatting, fusing data from a large number of sensors, 3D visualisation. Any proposed improved system would need to be able to cope with very noisy data, whilst retaining good usability, connectability, speed and the ability to handle real-time data. There are particular operating system problems to be overcome because currently the instrument control software runs on a transputer which is good for real-time working, whilst Origin runs under MS Windows.

In most other projects in the Foundation Programme, the visualisation techniques used are more for presentation purposes than understanding; the understanding tends to come from analysis of the data. Generally the needs are fully met by existing software packages (e.g. gnuplot, MatLab, LabView and Mafia). Mafia has a significant visualisation capability and is used, for example, to visualise 3D vector fields.

However, there is potential for further visual modelling to be used within the Foundation Programme. A major issue for those involved in the programme is keeping abreast of advances in commercial visualisation software, to know what software tools should be employed to achieve the objectives of a particular project most effectively.

3.4 Data Fusion

Within the Foundation Programme, data fusion is currently only used in the moving coil kilogram work. The data is fused by using time indicators in the data based on the modified Julian date

(MJD). Data reduction is performed by hand, using the knowledge that the scientist has regarding which data should be ignored (e.g. because there were unusual factors corrupting the data in a given time interval). It is considered that the automation of this data reduction would be too difficult, due to the difficulty of capturing the expert knowledge of what was happening during the experiment at specific times. However, there is scope for automated data fusion to look for hidden patterns in the data.

3.4.1 Applications

1. *Processing and reconciling measurements from an array of sensors*

   This is the normal use of data fusion in the Foundation Programme.

   There are circumstances in which data from a large number of thermometers and time sequences has to be reduced. At present this is done by looking for visible correlations and by using intuition rather than mathematical techniques. There seems to be scope for investigating whether data fusion techniques could be used to perform more effective data reduction in such applications. However, there remains some understandable scepticism from the physicists that any form of algorithm could replace scientific understanding in areas of fundamental research.

2. *Patching together measurements corresponding to different sets of parameters*

   Where measurements related to a set of different parameters need to be fused, the classical stepwise approach is used of isolating each cause of uncertainty in turn. In fundamental research it is important to do this in order to gain the resulting depth of understanding of what precisely is going on. Thus, there does not seem to be any scope for using data fusion to tackle such problems.

3. *Fault and error detection/diagnosis by multivariate statistical process control*

   This does not seem to be applicable in the Foundation Programme.

4. *Combining results from several very different measurement processes*

   This is only applicable in the Foundation Programme to the extent that it applies to the combination of data from two very different measurement processes in the moving coil kilogram work..

5. *Dealing with predictive data from models, or historical or archive data, in tandem with current measurement data*

   This is applicable to some extent in the moving coil kilogram work. In looking at the acceleration due to gravity, the effects of the sun and moon are predictable and corrections are made by using the appropriate model. However, the effects of the mass of air above the experiment and the water table underneath are not predictable and hence are not modelled. There is here a potential application for data fusion techniques.

6. *Handling numerical, image, and/or qualitative data within a single framework*

   This does not seem to be applicable in the Foundation Programme.

3.4.2 INTErSECT awareness

Although there is good awareness of INTErSECT within NPL, INTErSECT projects are inevitably too close to the market to be relevant to the long-term research being conducted in the Foundation Programme.

4. Validation and Testing

4.1 Spreadsheets and Other Mathematical Software Packages

4.1.1 'Off the shelf' software

Several different 'off the shelf' mathematical software packages are used in the Foundation Programme: particularly, MathCad, Maple, Mathematica, Mafia, MatLab, Origin, Measure, the NAG library, LabView and Sigmaplot.

4.1.2 Spreadsheets

Spreadsheets are not generally used for modelling or other mathematical computations in the Foundation Programme. However, on some projects data is recorded in Excel spreadsheets for archival purposes. Only simple calculations, such as averaging, are performed on the data.

4.1.3 Awareness of limitations

Within the Foundation Programme, there is some awareness of the potential limitations in spreadsheets and other software packages arising from inaccurate or unstable numerical algorithms. The main approach used to combat these limitations is to check the results using alternative methods.

4.1.4 Testing and validation

The extent of testing and validation performed on software packages depends on the importance of the package and the practicality of validation. In the case of the purchase of a new expensive package, thorough testing will be undertaken. There is a tendency to rely upon well established packages and to check results by using alternative methods. For example, when Mafia was used to model a simple analytically soluble case, the results were checked against those from an alternative method used by collaborators. In the case of Origin in the moving coil kilogram work validation is performed by using a simulation of the moving coil system. This showed that despite double precision arithmetic, it was only accurate to 1 part in 10^9 which is only just good enough.

4.1.5 Supplier advice

For low cost (< £1000) shrink wrapped software, users seldom seek advice from the suppliers about the mathematical performance characteristics of the package before purchase. They usually tend to rely on prior experience with the package or evaluation of demonstration versions coupled with expert reviews in the literature. Of course, should problems arise then the supplier will be contacted for advice. For low cost products advice from suppliers is variable. The general impression is that big name suppliers in particular are often reluctant to do anything about reported faults other than to thank the customer and say that the information will be passed to the development team. Such customer reports are certainly not always acted upon. Smaller or niche market suppliers are usually more inclined to respond positively to user concerns.

In the case of software/hardware systems representing large capital investments all aspects of the software will be scrutinised by the user. Where mathematical performance is an issue this will be tested to a level deemed appropriate by the user. There will normally be a contractual obligation on the supplier to co-operate in such testing and it will normally be able to offer much useful advice and information about an established product.

4.2 Model Validation

Experimental design is often the purpose of projects in the Foundation Programme. These are usually checked by the intuition and understanding of the scientists involved. Predictions are made and checked against experimental results. However, in the moving coil kilogram work validation of experimental design is performed against a simple simulation model of the physics involved. A similar example can be found in the scanning tunnelling microscope work, in the validation of a model of nanowires. In some cases, models are validated by using limit data for which the answer is known. In addition, the methods used for measurement system validation are also, where applicable, applied to model validation (see 4.3.2).

4.3 Measurement System Validation

4.3.1 Embedded software

Most modern instrumentation contains embedded software, but since this cannot generally be accessed by the user such instrumentation tends to be validated as complete systems. Within the Foundation Programme, mostly instruments are purchased with the software already embedded within them. For example, many oscilloscopes have Windows 95 interfaces and perform a lot of signal processing, but it is unclear to the user what functions are implemented in hardware and which are implemented in software. Some instruments switch between hardware and software control without telling the user. This makes the design of appropriate validation more difficult.

However, in some Foundation projects (e.g. the moving coil kilogram work and the magnetic moments project) embedded software is designed and developed using an unusual approach. The instruments are connected to the processor via a fibre-optic ring, which is a protocol-free bus, giving low-noise high-speed connectivity, whilst allowing the controlling software to be considered both as embedded and non-embedded. There is also some real embedded code in the form of programmable logic for time-critical processing.

4.3.2 System validation

Novel measurement systems are validated by checking their results against results from old systems using established methods. New measurements of fundamental constants are compared against the previous best practice (established by reviewed experiments) or accepted value. Sometimes, the results from direct methods of measurement are compared against those from indirect methods.

4.3.3 Reference data

Reference data is sometimes used in the validation of small subsystems. More often, typical data or simulated data is used instead of reference data in measurement system validation.

4.3.4 Validating embedded software

Where embedded software is supplied with the hardware, there is a general lack of user validation of such software and this means that there is no detection of any discontinuities in the outputs that might be caused by software errors. In those cases where the software is developed in-house, there is the potential to validate the software in isolation from the hardware, but generally this is not done; the software is instead usually only tested together with the hardware in full system validation. This is still in line with quality system requirements because the software is classed as prototype software which is for internal use only and which needs to be under continual development; however, if suspicious results indicate a problem with such software, then it can and should be tested in isolation from the hardware. In the case of the control software in the moving coil kilogram work, some parts of it need to be very thoroughly tested and validated. The fact that the software is readily separable from the instruments makes this possible.

4.3.5 Supplier guarantees

Users, as a rule, have to rely on the claims of manufacturers that embedded software has been adequately tested against supplier specifications. Users are nonetheless considered responsible for ensuring that purchased products meet specific needs.

Within NPL, software supplied by one part of NPL to another will normally be produced to a software quality standard forming part of the NPL quality system. This ensures that adequate testing is performed before the software is put into use. However, within the Foundation Programme, much of the software is developed by the scientists themselves for their own use and continual adaptation; such software is not generally tested as thoroughly (in quality system terms it is prototype software rather than production software).

4.3.6 Supplier validation of embedded software

Suppliers should validate embedded software as part of the design and implementation process. Current recommendations on software qualification require a 'design qualification' to verify that the design meets requirements; this is necessary but not considered sufficient. Most suppliers use a specification-design-code-test process through implementation; in conjunction with a design qualification, the specification/test pair constitutes validation. In addition, software needs to be tested in situ and under normal performance conditions.

Development and testing in general are almost certainly best carried out within recognised software development systems, such as TickIT, which allow suppliers to choose their own processes to achieve fitness for purpose.

4.3.7 Access

The scientists should have sufficient access to embedded software to satisfy themselves that it meets their own requirements. It is not often necessary to require access to software performing a purely control function for example but, where the software is operating on actual measurement data, it is important for the scientist to be able to test this independently of the manufacturer. Unfortunately, manufacturers do not generally provide such a facility. The approach used in the moving coil kilogram work and magnetic moments project (using the optical-fibre ring) could be applied to other areas of metrology where separation of the software from the hardware would be a significant advantage, without compromising speed or control.

4.4 Validation of Simulated Instruments

Monte Carlo simulations are used in some projects. In the scanning tunnelling microscope project there is some simulation software that came with a vacuum system that is used; this software is used for guidance purposes only. There is also a relatively simple simulation used in the moving coil kilogram work. It would be good to have a more comprehensive simulation, but it would need to be highly complex and it is unclear that it would be cost-effective to produce. There may well be more scope for simulation to be used in other Foundation Programme projects, but only in situations where the physics is well understood. There is little likelihood of being able to apply the ideas of the Virtual Co-ordinate Measuring Machine within the Foundation Programme.

5. Metrology Software Development Techniques

5.1 Software development methods

Within the Foundation Programme, bespoke software development is carried out in-house in a variety of different languages (including HT and HP Basic, C, C++, Turbo Pascal, Modula 2, Perl, and various command languages) and packages (e.g. MatLab or LabView). Bought-in software is only used when it is shrink-wrapped widely used software. When bespoke software is needed, suitable expertise tends to exist, so that in-house development is practical, especially where it is important for the research that the scientist has an in-depth understanding of the software and an ability to continually tune the software as the experimentation proceeds. Nevertheless, there is some scope for a professional software developer to work in conjunction with the relevant scientist, particularly where the software requires specialist software engineering expertise that the scientist may not have time to acquire.

The development methodologies employed for bespoke software have tended to be informal, somewhat quick and dirty, aimed at getting some working code in order to demonstrate feasibility, rather than the more formal approaches of a professional software house. The software development process used should reflect the risk associated with the software; so software used in prototyping is, in practice, often written before the design documentation and the documentation will not be comprehensive, whereas commercial packages should be developed using good software engineering practices.

5.1.1 Software Quality

1. Quality systems

NPL is certified to ISO 9001 plus TickIT, so any software NPL sells is produced in accordance with the requirements of TickIT. However, most of the software developed within the Foundation Programme would not meet the requirements for new production software (software integrity level 3), but only the lesser requirements of software integrity level 2, as is appropriate to research and prototyping.

2. Reliability records

No formal records are kept of the reliability of software used in the Foundation Programme. Such records are only kept within the context of the particular project, rather than being disseminated throughout the Foundation Programme or even wider, because the usage is very specific to the particular project.

3. Reliability

The methods used provide sufficient reliability for research purposes, but it is recognised that the reliability can always be improved.

4. Version Control

Configuration and version control of software are used consistent with the requirements of the NPL quality system. This is important if results need to be reproduced. Control information is put in the headers of data files to ensure that the data can be correctly traced to the exercise in which it was generated. For example, within the Basic Metrology Section, some use is made of the Revision Control System (RCS). However, experience shows that reversion to old versions is not always easy and indeed in a research environment it is not always sensible. Data capture software is usually to closely bound to equipment, so that reversion to an old version would be impractical unless the equipment could be similarly taken back into a previous state. Data analysis software, however, could revert to an old version if necessary to rerun a previous analysis.

Special attention is paid to version control in the moving coil kilogram work, using the SourceSafe package. At present this is run as a single-user system on a single machine, but it would be better to have a system which could be run over the network whilst maintaining a high level of security, and providing a fast and clear visual differencer (e.g. CVS).

5.1.2 Management issues

1. *Maintenance*

   Software developed in-house for research and prototyping purposes does not so much need maintenance as continual refinement as the research progresses; the development and refinement of such software can only be done effectively by the scientist or by a programmer working very closely with that scientist.

2. *Development costs*

   Current software development costs are considered to be reasonable.

3. *Problems*

   The main problem comes from the pace of IT developments: scientists in the Foundation Programme find it hard to keep pace with developments and find it difficult to make the time to climb the learning curve to mastery of each new language and package that would be useful. This is where use of professional software engineers may help, provided that they work closely with the scientists in effective multi-disciplinary teams.

   There are several problems with the use of LabView - these are documented below in Section 5.3.

4. *Quality attributes*

   The most important software quality attributes in Foundation Metrology are numerical accuracy, reliability and adaptability. It is interesting to note that, of these, only reliability features as a key quality attribute in the ISO standard on software quality.

5. *Risks*

   Being a research area, the risks from software errors in the Foundation Programme are not high. The main costs of software errors would be project delays and additional effort costs.

   Some risks come from the operating system environment. In general, Windows is less stable and less reliable that UNIX. UNIX also copes better with large files and large numbers of files. For this reason UNIX is the preferred platform (usually in its Linux variant) for many Foundation Programme projects.

5.1.3 Technical issues for software developers

1. *Development methods*

   Developers report using a combination of both the V model (design-code-test) approach and prototype refinement. The V method is used to develop a working prototype and then that prototype is refined as requirements change. The design may well get improved as a result of the code and testing phases. In some projects, there is an emphasis on good module design with simple interfaces, and on sufficiently thorough testing to give high reliability.

2. *Software testing methods*

   Software for experimental control, using multi-channel analogue and digital measurements, is tested by direct measurements, checking the results by using another instrument.

   Where processing occurs, the results are generally checked by hand.

Occasionally if appropriate a mathematical tool will be used to check the results produced by a piece of software.

In some projects, each command is tested in use with the apparatus.

3. *Independent testing*

   Some software can be tested in isolation of the hardware, but most is tested in combination with the hardware. In particular, a PC and the DAQ boards are usually needed in order to test experimental control software.

4. *Development problems*

   The scientists working in the Foundation Programme are generally satisfied with their current software development methods, reporting that they are not aware of any significant problems.

   In the moving coil kilogram work, the Modula 2 compiler for the Transputer gives problems that constantly require work-arounds. The solution must be to move away from the Transputer and indeed a new real-time platform is under evaluation.

5. *NPL Measurement Good Practice Guide 5, Software in Scientific Instruments*

   The scientists working in the Foundation Programme report that they do not have access to a copy of Guide 5; they were not even aware of its existence.

5.2 Software reuse

5.2.1 Management issues

1. *Importance*

   Software reuse is seen as an important means of reducing software maintenance costs. However, mostly re-engineering of software, rather than reuse, is the emphasis; existing software is re-engineered for use in a similar context or to solve a similar problem.

2. *Mixed language programming*

   Mixed language programming is used to some extent in the Foundation Programme, for example in the use of scripts with C with LabView. There is also likely to be a future requirement in some projects, e.g. where Modula 2 will need to be used together with C.

3. *Responsibility*

   The details of software reuse and mixed language programming tend to be left to the scientists concerned or their suppliers.

4. *Legacy software*

   Most of the legacy software is instrument control software, e.g., written in HP or HT Basic, which is used to control old instruments. There is little need for maintenance of such software.

5. *Year 2000*

   The Year 2000 problem has been recognised for some time. NPL has had a task group working on this issue, testing every piece of appropriate equipment. In the moving coil kilogram work, the use of the MJD makes all the important software millennium compliant. In other projects the UNIX epoch is used, which is not sensitive to the year 2000, although if faces a similar roll-over problem in 2038. However, there are problems in switching between the UNIX epoch and the MatLab date, because daylight saving time adjustment can get lost, but this is not really a Y2K problem. Those projects using LabView use its time-stamp which is millennium compliant. There is a general concern that there are too many different ways of doing time-stamping.

5.2.2 Technical issues for software developers

1. *Producing similar programs*

   When it is necessary to produce a program very similar to an existing one, any of the following approaches may be used depending upon circumstances:

   a) copy the existing program and make the changes needed;

   b) add an additional mode to the existing program to cater for small changes;

   c) start from scratch using LabView, for experimental control software which was produced in an old development environment.

2. *Software reuse strategies*

   The potential for software reuse in the Foundation Programme is generally low except within a single project. Attempts are sometimes made to minimise software maintenance costs by careful reuse of software. Since most of the software is developed by the scientist who will use it, there is no organisational barrier to reuse. However, reuse requires good documentation and version control, which in some projects is a low priority.

3. *Extent of mixed language programming*

   Some use is made of mixed-language programming in the Foundation Programme (e.g. scripts plus C plus LabView), and it is anticipated as a future requirement in some projects (e.g. Modula 2 plus C).

4. *Year 2000 problems*

   There are no known Year 2000 problems in the Foundation Programme. With the shrink-wrapped software that is used, the scientists concerned are relying upon the suppliers to fix any Year 2000 problems that those packages contain.

5.3 Virtual Instruments

The use of LabView at NPL is increasing; within the Foundation Programme, LabView is being used for most new instrument control applications, usually run under the Linux operating system for reliability. Several people have been on LabView training courses within the last year, but they are still not yet as efficient at creating instrument control software as they were with HT or HP Basic. Nevertheless, the use of LabView is sometimes a necessity because of interfacing problems. One difficulty with LabView is that it is poor at real time control, which gives problems in correlating data collected from two parallel systems monitored by separate computers. Precision real-time control requires the use of hardware clocks. Problems experienced in this area arise from LabView either not supporting or else duplicating the functionality of the relevant board software. In principle real-time control should be no more difficult with LabView than with other software packages. Further tips and hints would probably be helpful, although the scientists concerned doubt that they need a formal training course.

6. Support for Measurement and Calibration Processes

6.1 Automation of Measurement and Calibration Processes

There are no measurement or calibration services within the Foundation Programme, hence the automation of the processes through LIMS or other means is irrelevant.

6.2 Format Standards for Measurement Data

6.2.1 Need for standard data formats for measurement data

There is no apparent need for any standard formats for measurement data used in the Foundation Programme.

6.2.2 Need for sharing measurement data with other organisations

Measurement data from the Foundation Programme is shared with other NMIs conducting similar research, but this is really just for research purposes. In these cases, simple text files are used with headers to explain what each contains, or else instrument specific formats are used (e.g. for gravimetric data).

6.2.3 Need for a common format for calibration records

There is no need for a common format for calibration records, since there are no calibration records generated within the Foundation Programme.

6.2.4 Need for a common format for exchange of measurement data

There is no apparent need for a common format for electronic exchange of measurement data.

6.2.5 Different data formats

There are currently no identified problems with the different proprietary data formats found in different measuring instruments.

7. Suggestions for Future Activities

7.1 Expected Benefits of the SSfM Programme

The areas of the SSfM programme which are most likely to benefit the Foundation Programme are modelling techniques (particularly continuous modelling), model validation, visual modelling and data visualisation, data fusion, the results of testing relevant software packages, and virtual instruments.

It is possible that some improvements to the modelling techniques or model validation used in some Foundation Programme projects may be identified. Visual modelling of microwave fields could benefit the design of resonators and chokes. The moving coil kilogram work could benefit from improved data visualisation and from automated data fusion techniques. If significant errors are found in relevant software packages then these can be avoided or overcome in future. Further guidance on the development of virtual instruments should assist with the future use of LabView in instrument control.

As regards delivery mechanisms, there is doubt about the need for formal training courses. On the other hand, seminars, workshops, club meetings and awareness articles, plus intranet activities like the web site and news groups, are considered to be effective means of providing the tips, hints and other guidance that are needed.

7.2 Case Studies and Feasibility Studies

The continuous modelling undertaken within the Foundation Programme could be investigated to see whether it could provide an example of best practice to be discussed in the report on continuous modelling, although there are some concerns that it may be too specific to provide any useful transferable lessons.

In the moving coil kilogram work, there are potential case studies in model validation, measurement system validation, and uncertainty estimation.

7.3 Future SSfM Topics

A major issue is how to provide good quality, up to date, practical advice to scientists, allowing them to make use of relevant technological advances without causing undue disruption to their experimental work, e.g. in the area of visualisation tools. The judgement on when to make use of new technology has to be left to the scientist but the scientists need easy access to the relevant information. This can be seen as providing a challenge to the SSfM programme, which may be best met by widening the scope of MetroS (the SSfM web-based metrology software library). However, there is concern over the MetroS approach to providing validated numerical software by making DLLs available which can be coupled into packages as user supplied functions. The concern is that this job needs to be redone when a new version of the package comes out. The challenge for SSfM is therefore to press suppliers to incorporate such validated routines into their packages directly.

More work is needed on data fusion. Data fusion needs time-stamping, but there is a need for better guidance on time-stamping. The use of MJD is good, but a validated algorithm for calculating it is needed, as the one in Numerical Recipes is wrong. It would be desirable to have an MJD validated algorithm implemented in each language and package being used. Also regression tests for MJD conversion algorithms would be helpful. Furthermore, recommendations on authentication of such software would be useful.

More advice is called for on choice of programming languages. In some cases it is important that compilers and packages are available to run under UNIX, whereas in other cases platform independence is a requirement, but generally in the Foundation Programme there is a desire not to be required to use Windows PCs. There is one particular desire for a language which is platform independent, object-oriented, with both power and yet simplicity, and with a clean structure without backdoor traps so that it leads to reliable code which won't fall over.

The overlap between model validation and measurement system validation should be investigated. Sometimes these are two ways of looking at the same thing.

7.4 Future Foundation Programme Topics

None identified.

8. Acknowledgements

The author gratefully acknowledges the help provided by Ian Robinson and Jeff Flowers in providing input to this report and in reviewing drafts.

Appendix 1: Glossary of Abbreviations

Abbreviation	Expansion
GPIB	General Purpose Interface Bus
ISO	International Organization for Standardization
LIMS	Laboratory Information Management System
MJD	Modified Julian date
NMI	National Metrology Institute
NMS	National Measurement System
RCS	Revision Control System
SSfM	Software Support for Metrology
Y2K	Year 2000

Appendix 2: Bibliography

1. *Guide to the Expression of Uncertainty in Measurement*, International Organization for Standardization, 1993, ISBN 92-67-10188-9.
2. *ISO 9001: Quality systems - Model for quality assurance in design, development, production, installation and servicing*, 1994, Geneva, Switzerland.
3. *Measurement Good Practice Guide 5, Software in Scientific Instruments*, NPL, 1997.