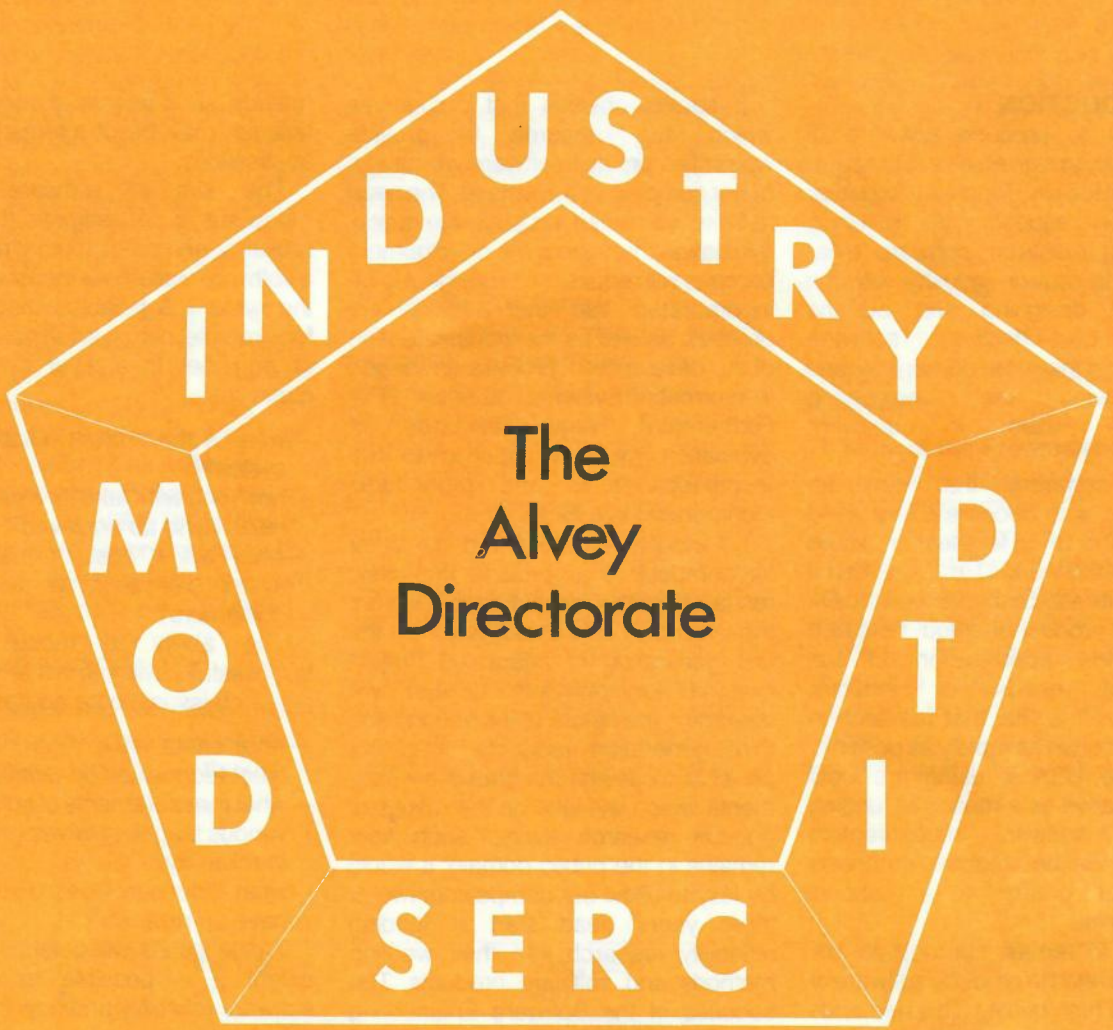
SOFTWARE RELIABILITY AND METRICS PROGRAMME

Overview



Prepared for the Software Engineering Directorate by the Centre for Software Reliability

Alvey Directorate
Dept of Trade and Industry
Millbank Tower
Millbank
London SW1P 4QU

April 1984

SOFTWARE RELIABILITY AND METRICS PROGRAMME
Overview

1. INTRODUCTION

The ability to produce software of known quality to a given cost is critical to the future of the UK IT industry. Together with other aspects of software engineering research, software reliability aims to provide industry with the means of designing and building reliable, cost-effective software systems. It is therefore primarily aimed at improving the engineering procedures used in software production.

It is apparent that Software Engineering is to become a key world industry. If the UK is to retain or extend its share of the world IT market, then it must back its established excellence in innovative product-oriented research with a new appreciation of the commercial needs of industry. Furthermore it is vital that research in this area is begun as rapidly as possible. Already the USA is using the DoD STARS initiative as a means of funding work on software measurement activities, while the Japanese emphasis on product quality is a national preoccupation.

In order to remain competitive, UK industry will need to produce software of guaranteed high quality. This demands the ability both to understand and improve the software development *process* (i.e. the techniques, tools, tasks and procedures by which software is produced) and to evaluate quantitatively the *product* produced by that process. Only then will software managers and production staff be able to introduce the vital commercial factors of cost and resources into the production process and to determine a profitable balance between production costs, maintenance overheads and customer requirements.

It seems obvious that companies which are prepared to provide warranties on their software will have a major competitive advantage. The need to base warranties on accurate quality evaluation throughout the development process will require measurement to be incorporated into the development process, as well as the industrialisation of the development process envisaged in Information Systems Factories (ISFs). Furthermore, new standards of evaluation must be applied to the techniques, tools and procedures incorporated into ISFs.

UK industry cannot afford to wait for its competitors to provide the basic research in this area. Feedback from measurement and evaluation from the first generation of Integrated Project Support Environments (IPSEs) will determine the nature of the second and third generation products. It is not possible to design integrated environments which will take up the results of non-UK research, even if such work remains in the public domain. It would be folly to allow our competitors two or three years head start in feeding reliability research into their working methods and software products. The success of the Software Engineering initiative of the Alvey Directorate rests on its ability to demonstrate the effectiveness of new methods of development, and research into software measurement and evaluation is essential to that goal.

2. SOFTWARE RELIABILITY

2.1 What is software reliability?

Software reliability may be formally defined as the probability of failure-free operation for a given time period in a specific environment. Alternative metrics, such as rate of occurrence of failures, may be of interest in certain applications.

The aim of software reliability modelling is to estimate the current reliability, and predict future reliability, of a software system via measurements of the system. A reliability model should, for example, permit a software manager or engineer to answer the following questions:

—what is the current reliability of the system?
—how long will it take for a given level of reliability to be achieved?
—what will it cost to maintain this system once it is released to customers?

The ability to model reliability successfully will permit a number of research issues to be addressed:

—what effect have different development techniques on reliability?
—what measurements of software in its various stages of development relate to reliability?
—what influence does customer use have on reliability?

These issues will determine to what extent it is possible to guarantee software reliability and how this may be achieved.

2.2 The importance of reliability

During the early years of programming, it was difficult enough to produce a system which provided the functionality required by its users, without worrying about reliability or any other quality considerations. Computers acquired the reputation of being unreliable and difficult to use. Members of the public ceased to be surprised by bills for ridiculous amounts, but the sphere of influence of computers and software

was sufficiently limited for them not to feel unduly worried.

However, the last few years have seen a massive increase in the use of computers throughout society. Familiarity with computers has led the general public to expect higher quality software, and moves to computerise life-critical and cost-critical systems have heightened this new awareness. Failure of a major company as a result of unreliable business software or any loss of life due to software failures in avionic systems or major chemical processing plants could cause a massive public reaction against the use of computers, which would have very serious consequences for all the industrialised countries. Events such as the software synchronisation failure which delayed the launch of the Space Shuttle, and the false report of an enemy attack proclaimed by the WWMCCS computerised defense control system, may be a harbinger of worse things to come if software reliability issues are not resolved.

It would be foolish to remain complacent about current levels of software reliability when advances in hardware and the requirements of industry are encouraging the development of ever larger and more complex software.

It is therefore clear that advances in the functional capability of software must be matched by improved methods of achieving reliability targets, and knowing that they have been attained.

3. SOFTWARE WARRANTY

3.1 Certification and Warranty

Guarantees of software quality will have much greater credibility if they are based upon independent assessment. The proposed National Quality Certification Centre is the obvious agency for such independent assessment and would be a major user of new measurement and modelling techniques resulting from the software reliability programme.

The NQCC would provide software producers with formal certification of software products. However, a producer would need to base a warranty on commercial considerations. When certification confirmed that a product exhibited a certain level of reliability, the producer could make informed decisions about maintenance costs before providing a warranty to customers.

Thus a software guarantee would have two components: independent certification by the NQCC and a cost-based warranty provided by the producer (which may also need be agreed with the NQCC).

3.2 Current Practice

Current practice in software quality assurance concentrates on the software development process. This involves agreeing the procedures necessary to complete the individual processes involved in each stage of software development (theoretically from requirements definition until replacement but in practice usually from specification until delivery), and checking that each process is completed in accordance with the agreed procedures. Such procedures usually involve conducting a series of design reviews or audits, adhering to agreed coding standards, producing a number of standard documents and performing certain classes of testing.

The two major deficiencies of this approach to quality assurance are:

(i) the relationship between system reliability, and the procedures and process at each stage of development, is completely unquantified and in some cases is no more than a pious hope;

(ii) there is no requirement to certify the product itself rather than the process that produces it: direct product measurement or assessment is ignored.

Accordingly, there is no way of assessing appropriate software development techniques for various reliability requirements and no way of evaluating whether reliability requirements have actually been achieved. The software reliability research programme outlined below will provide the basic information and techniques needed to resolve these problems and place software certification on a scientific and quantitative basis.

4 RESEARCH PROGRAMME SUMMARY

4.1 Aims of the Programme

The objective of the software reliability research programme is to provide the capability to design and build reliable software systems in the most cost-effective way. Thereby, software producers in the UK will be in a position to deliver and warrant software of competitive quality.

In order to achieve this objective a number of co-ordinated research activities need to be initiated:

Figure 1

Strategy	Innovation and Understanding	Integration and Implementation	Exploitation and Evaluation
Achievement	Models of the software development process	Procedures for management and production staff, e.g. Handbooks for reliable software development	—Database of software reliability and development —Procedures for software certification e.g. data collection from IPSEs
Assessment	Models of software reliability	—Cost and quality models —Procedures for quality assurance	—Software development technique and tool evaluation —Model evaluation

(a) to understand the causes of system failure, in order to develop and validate models of system reliability which permit software reliability to be specified, measured and controlled;
(b) to understand and measure the component processes of the software lifecycle in order to develop and validate quantitative models of software production;
(c) to improve and evaluate software development methods and techniques in terms of the reliability of resultant software;
(d) to produce software development rules and procedures such that given reliability requirements may be achieved;
(e) to develop resource models incorporating models of software development processes and software quality such that costs to the customer and producer can be predicted and controlled;
(f) to provide procedures and rules to enable the development of reliable software to be adequately managed.

The programme is divided into two themes, achievement and assessment, since it is important not only to *evaluate* the reliability of software but also to provide the techniques and tools which assist in the *production* of reliable software. The anticipated goals of these areas of research are summarised in Figure 1. It should be emphasised that the various categories that are used to partition the research goals will in practice be very closely interrelated as one part of the research programme feeds back information into another. For example, the results of technique evaluation should feed back into the development of handbooks to guide production and management; and cost and quality considerations and model validation exercises will refine the original models of reliability and the development process.

4.2 Results of Programme

(i) Short term (2½ years)
Within the first few years of the programme much of the fundamental research work will be done. The expected results in this time period are:
—NQCC using improved reliability models (such as adaptive models) for certification;
—mathematical formulation of new classes of reliability models incorporating additional information (project characteristics, development techniques and processes, and customer profile);
—handbooks for developers outlining best of current practice in producing reliable software;
—guidelines for project managers based on new models of the software development process incorporating cost and quality considerations;
—data collection processes established and working.

(ii) Medium term (5 years)
During this time period the theoretical models developed at the start of the time period will be evaluated and refined using data from a variety of software projects. In addition work will continue on developing reliability models tailored to special types of system (real time, embedded, distributed, etc). The expected results by the end of the programme will therefore be:
—NQCC using new classes of reliability models incorporating data collected from IPSEs for product certification;
—risk models of reliability identifying product characteristics and development methods most likely to achieve various levels of reliability (in principle these will be similar to medical risk models which identify the characteristics of groups at risk from various diseases);
—mathematical formulation of reliability models for special systems (validation may not be possible within these time scales);
—improved software engineering techniques to enable the construction of more reliable software;
—new handbooks incorporating new information resulting from the entire software engineering programme and validation exercises performed on new development techniques;
—cost and quality models incorporating and reconciling macro-level (strategic) and micro-level (project management) decision making;
—automatic collection of project data for certification and industry standardisation by IPSEs.

(iii) Long term (5 to 10 years)
After the completion of the current Alvey programme, work will still need to continue to incorporate the new methods of software development and novel system architectures developed by the Alvey collaborators into reliability modelling and certification procedures.

It can be anticipated that models of system and software reliability will need to be adapted and validated as new concepts of software engineering continue to emerge.

5. CONCLUSIONS

It is essential that the UK retain and if possible expand its share of the world IT market. In order to do this innovative product development will not be enough. Customers will demand high quality and, in particular, high levels of reliability from software-based systems. A considerable competitive advantage will accrue to software producers who are prepared to provide warranties to their customers backed by independent product certification. If UK companies are unable to provide this, there is no doubt that the US and Japanese companies will.

Therefore the Alvey Directorate will back a major research programme in software reliability as outlined in this document and detailed in the Software Reliability Research Programme. The objective of this research programme is to provide the capability to design and cost-effective reliable systems such that:
—a valid basis for independent software certification and warranty is achieved;
—software reliability can be measured, predicted and controlled;
—evaluation of development methodologies is based on objective measurements;
—the development of reliable software can be appropriately engineered and managed.

6. FURTHER DETAILS

Further details of the strategy and plan for the programme for software reliability are contained in the full document prepared by the Advisory Group—the Centre for Software Reliability. This will be published by the Alvey Directorate in mid-April.