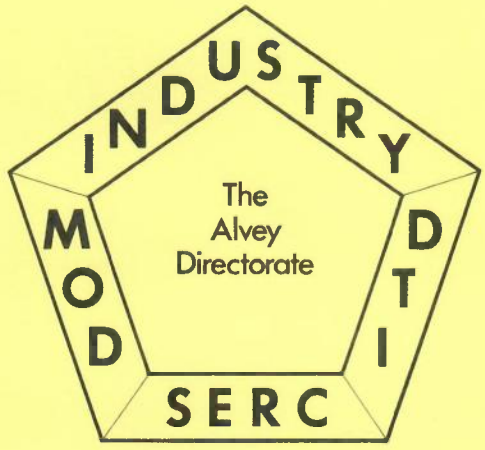
ALVEY PROGRAMME

SOFTWARE ENGINEERING

PROGRAMME FOR FORMAL METHODS IN SYSTEM DEVELOPMENT



April 1984

Prepared for the Software Engineering Directorate by the Advisory Group on Formal Methods

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

Abstract

The development and early exploitation of formal methods of system development is a key aspect of the Alvey Directorate strategy in software engineering. An Advisory Group on Formal Methods, with members from the industrial and academic sectors, has been formed to assist the Directorate in developing its strategy in this area. This document has been produced by the Advisory Group and sets out the proposed programme for formal methods in system development. The programme emphasises three areas of activity: the rapid exploitation of mature formal methods, the industrialisation of promising methods so that they can be exploited in the near future, and fundamental research to provide more powerful methods in the long term. The programme addresses systems in general (not just software) and will be closely coordinated with other Alvey programmes.

ALVEY PROGRAMME

SOFTWARE ENGINEERING

PROGRAMME FOR FORMAL METHODS IN SYSTEM DEVELOPMENT

Prepared for the Software Engineering Directorate by the Advisory Group on Formal Methods

Contents

1. Introduction

The Alvey Directorate has issued two documents which present, in differing degrees of detail, the Alvey Software Engineering Strategy (1, 2). In both documents, the need for development and early exploitation of formal methods of system development is identified as a key aspect of the programme.

The Alvey Programme Software Engineering Strategy (2) also identifies the need to establish specialist panels to assist in the development of programmes for specialist topic areas. This document represents the strategy and programme proposed by the panel so established by the Director, Software Engineering, for work in the area of Formal Methods. The composition of this panel appears as appendix 1.

Formal methods of systems engineering are rigorous engineering practices which are generally based on mathematical formal systems and which are to be applied to the development of engineering products such as software and hardware. These practices will often include the use of computer based tools. Work in Formal Methods aims to cover all aspects of the system life cycle. It may be the case that, in some areas, the determination of a totally formal system or method is infeasible. This does not mean that the programme will ignore such aspects, but rather that the degree of formality will be less.

Formal Methods are seen as the necessary basis for the accomplishment of the central goals of the Software Engineering Strategy. Work on Formal Methods represents the development of the technological foundations for the development of information systems. These foundations provide the base for the production of appropriate tools to be made available within the Integrated Project Support Environments and the Information Systems Factory envisaged in the Software Engineering Strategy. They are necessary for the techniques to be applied in software certification by the proposed National Quality Certification Centre. Work in Formal Methods will necessarily identify requirements for the measurement and management of the process of information systems development which have been identified as further specialist areas within the Software Engineering programme. The Formal Methods programme will seek to satisfy requirements from other major components of the Alvey Programme insofar as these requirements are related to the area of Formal Methods. The programme will also look to other components of the Alvey Programme for developments appropriate to achieving its own objectives.

2. Goals and Objectives

2.1 Objectives of the Alvey Software Engineering Programme

The two major goals identified in the Alvey Programme Software Engineering Strategy are:

> improved *quality* i.e. satisfying criteria such as performance, reliability, security, on-schedule delivery and meeting the needs of the user;
>
> improved *productivity* i.e. reducing cost, not just of the development, but of the life-cycle as a whole, including maintenance and future evolution.

The major objective of the overall software engineering programme is that "in 1989 the UK should be the world leader in Information System Factories (ISF)" (2).

Achievement of the main goals of the programme is seen in terms of a number of more general subgoals. These include the use of advanced methods, the establishment of measures of quality and productivity, research and, particularly, experimentation into emerging and new techniques, the use of distributed systems, the development of standards and the short term exploitation of tool developments. In particular the programme foresees the development of a succession of Integrated Project Support Environments embodying the emerging technology.

2.2 Objectives of the Formal Methods Programme

The Formal Methods programme relates directly to the goals and objectives of the overall software engineering programme. The specific objectives of the Formal Methods programme are given below under the headings of Methods, Skills and Tools.

2.2.1 Methods

Objective: To establish a coherent set of formal methods to support the life-cycle of system development as a whole.

The establishment of formal methods will consist both of the development of ideas and methods which are currently well advanced (in particular their evaluation and industrialisation) and innovative work where there is seen to be a need. Coherence of the set of methods is seen as being of primary importance. The scope of the programme will cover all aspects of the system life-cycle. It includes requirements definitions, designs, test programs and plans, prototyping, documentation, version and configuration control, maintenance, training and materials etc., and applies to a variety of devices and media.

2.2.2 Skills

Objective: To create a body of engineers skilled in these formal methods and to develop means by which the methods can be introduced into day to day use.

An urgent requirement is to create a general awareness of formal methods in the potential user population. The programme itself cannot address the problem of educating the total population of systems engineers. Rather it is necessary to ensure that software engineers in industry are sufficiently skilled in formal methods to apply them for the purpose of early exploitation and evaluation as part of developing the methods. A second need is to identify how the requirements and obstacles of the larger education problem might be overcome.

2.2.3 Tools

Objective: To create integrated sets of tools to support the methods.

The development of tools to support the application of formal methods will be critical to the achievement of the overall goals of the Software Engineering Programme. Thus it is an objective of the Formal Methods programme that such tools are produced for both IPSE1 and IPSE2.

3. The Role of Formal Methods

Perhaps the most important reason for needing formal methods is to raise the standard of software engineering to the level of the other engineering professions. Formal methods will provide the scientific basis underlying the construction of systems. There is a need for at least the same level of confidence in certain forms of software as, for example, in civil engineering where there are techniques and methods applied in the design process to give a quantitative measure of the suitability of a civil engineering design for its environment. The use of formal methods is the key to being able to certify software in safety-critical applications. The use of formal methods will also underlie the certification to be carried out by the proposed National Quality Certification Centre and lead to better defined software standards.

The precision, rigour and checking required by the use of formal methods during the specification and design phases force greater clarity of thought thus producing a product to which more thought has been given and which is usually much cleaner and more straightforward than the initial confused ideas.

The precision of formal methods will uncover many specification or design errors which otherwise might not have been detected until system test or operation when their repair would be very expensive (if not prohibitive).

The existence of a formal specification gives a firm basis for interaction between the supplier and client and should lead to superior product definition and a sounder contractual basis.

If system components are formally specified, there is a natural basis for rigorous interface descriptions. This potentially simplifies the problem of reusing components. The formal specification of individual components allows the use of formal techniques to show the effect of composition and decomposition of system components.

The use of formal methods in the specification and design phases will cause documents and other forms of output to be produced early in project development. These outputs, because of their nature, can be objectively assessed and will thus provide early and useful milestones in the project plan. This will improve project planning and control, improve the review processes and lead to greater quality assurance. The effect of changes in requirement on cost and timescale should be more clearly apparent.

Because formal methods are based on well-defined rules, there is considerable scope for the use of automatic aids. Thus one might expect increasing use of more sophisticated automated tools to assist in the construction of computer systems with consequential increased productivity and quality assurance.

In summary, these benefits should give rise to higher product quality and reliability and, consequently, improved productivity and reduced overall life cycle costs.

4. Strategy

4.1 The Current Situation

The current situation regarding the exploitation of formal methods is characterised by the problems given below:

(a) There is an extreme skill shortage in formal methods in both the industrial and academic sectors. For example, only a very small number of engineers have even a reading knowledge of any formal method.

(b) The UK software engineering industry is severely undercapitalised, making very inefficient use of its most skilled practitioners. This is manifested in the very low availability of even simple minded computer based tools.

(c) There is a great dearth of training materials to be used in raising skill levels in the industry.

(d) Formal Methods which have been identified as being useful have only received very few and generally small scale trials. There is a general lack of properly conducted case studies in the use of formal methods.

(e) Verification techniques and tools which have been shown to be promising in laboratory trials, have not been given any trials in an industrial setting because they have not been adapted so as to be usable in an industrial context.

(f) Industry in general is unaware of the potential benefits offered by formal methods and of the necessary level of investment required for their assimilation.

Industry cannot be expected to commit large resources to formal methods without sound evidence of their effectiveness.

4.2 The Future

The strategy outlined below must be seen in the light of what can be attained at various times in the future. Scenarios of two and a half years (half way through the Alvey programme), five years (the end of the Alvey programme), and ten years hence have been chosen as delimiting stages in the development and integration of formal methods in industry. The three stages will be called The Consolidation Stage, The Enhancement Stage, and The Future Foundations Stage, respectively.

Within each stage, three differing sets of activities are presented, corresponding to those of Section 3 (page I.11) of the Software Engineering Strategy statement (2): Exploitation and Evaluation, Integration and Implementation, and Innovation and Understanding. Again, within each of these sets of activities, three foci are identified for directing the programme: Methods, Skills and Tools.

4.2.1 Useful Characterisations

Before embarking on the three scenarios, the following characterisations are proposed of important concepts extensively used below.

A *Formal System* is a well-defined system of notation together with well-defined rules for manipulating that notation which are based on a sound mathematical theory.

A *Formal Method* is a set of rigorous engineering practices which are generally based on formal systems and which are applied to the development of engineering products such as software or hardware.

A *Mature Method* is characterised by the following attributes:

(a) Books, technical reports, journal articles are widely available and accessible to all sections of the community.

(b) Training programmes have been developed and given field trials in industrial contexts.

(c) Industrial case studies have been conducted and the results published. Evaluations of these case studies have been conducted and also published.

(d) The method has had some, perhaps small scale, production experience in industry.

(e) Some useful tool support, possibly of an experimental nature, should exist.

(f) The major strengths and weaknesses of the method are reasonably understood (probably as a result of (c) and (d) above).

A *Promising Method* is characterised by the following attributes:

(a) The theory underpinning the method is mature in the sense that published material is available for evaluation by the research community.

(b) Small scale laboratory exercises have been conducted with encouraging results, but no large case studies undertaken.

(c) Work on the method has spread beyond the originating community.

Fundamental Research in formal methods funded by the Alvey Programme will be of two kinds: *Directed* and *Speculative*. The former is characterised by initial investigation and development of ideas and methods which are judged as essential requirements for the fulfilment of the specific aims of the Alvey Programme. Speculative research is characterised by initial investigation and development of ideas and methods which are judged to have some promise of application in furthering the aims and objectives of the programme.

4.2.2 The Two and Half Year Scenario

(a) *Methods*
During the Consolidation Stage, two or three mature methods will have been identified and evaluated in a number of applications. Some promising methods will have been identified and industrialised for possible future exploitation. Foundational work will have been started for future formal methods which will address outstanding problems (including component reuse).

(b) *Skills*
A nucleus of engineers will have been trained in the use of some mature method to the point where they are able to conduct trials of these methods in industrial settings. In addition, a key group (about 3%) of practising engineers will have been made aware of the fundamental concepts of formal methods.

(c) *Tools*
Limited, book-keeping type of support will have been developed for mature methods undergoing industrial trials. These tools will be integrated into a 1st generation IPSE. Production quality versions of more sophisticated tools will have been developed for possible trial and evaluation in the Enhancement Stage and possible inclusion in a 2nd generation IPSE.

4.2.3 The Five Year Scenario
(End of the Alvey Programme)

(a) *Methods*
Other formal methods will be available with a wider range of applicability (e.g. handling concurrency) and will be used in more phases of the life cycle. Methods found successful in the previous stage will be routinely used in industry and their value already demonstrated. Experimentation and large scale case studies should have been undertaken on some of today's promising methods. The foundational work for supporting component reuse etc., should be well underway anticipating industrial experimentation after the Alvey programme.

(b) *Skills*
There will have been a substantial expansion of the section of the UK software engineering community with an awareness of the fundamental concepts of formal methods. A tenfold increase (to 30%) is envisaged. There will have been a similar increase in the proportion of software engineers who have acquired the facility for using mature methods in production environments. Training material will be available to support the dissemination of today's promising methods. Some training in the use of these methods should already have taken place.

(c) *Tools*
Aside from more sophisticated book-keeping tools to support mature methods, experimental tools to support more advanced use of these methods should be available. These tools will include general purpose theorem provers, special purpose theorem provers and proof checkers adapted to support mature methods, animation and prototyping tools, etc. Designs for tools to support component reuse should be available in readiness for the actual construction of experimental facilities.

4.2.4 The Ten Year Scenario

(a) *Methods*
A functioning ISF (or 3rd Generation IPSE) will exist supporting the whole life cycle. It will provide production quality tools based on developments in knowledge based systems, reusable components and semi-automatic transformation systems. The extent of feasibility of automating various parts of the life cycle will have been established. Libraries of components will have been built and be in experimental use in industry. There will be foundational support for integrated hardware/software design. The development of concurrent systems will be supported by mature methods.

(b) *Skills*
Almost all of the software engineering community will by now be familiar with the fundamental concepts of formal methods. The number of engineers skilled in the use of mature methods should see a commensurate increase. The training and awareness programmes will have been continually adapted to reflect changes in technology.

(c) *Tools*
As mentioned above, the ISF will be ready for wide scale use with early attempts at marrying the formal methods and IKBS developments.

4.3 The Way Forward

Given the above scenarios, a programme of work must be established to achieve the goals required to make the scenarios become the reality. The Consolidation Stage (the first two and a half years of the Alvey Programme) is summarised in figure 1, where the required activities are grouped under the three headings of Exploitation and Evaluation, Integration and Implementation, and Innovation and Understanding.

This classification of activities recurs in the two subsequent stages. The products of each stage are expected to feed into the next stage in a manner illustrated by the diagram in figure 2.

Fundamental research (Innovation) is intended to continue throughout the programme with increasing support as manpower becomes available to carry out the work.

All projects carried out under the formal methods programme will be subject to continual monitoring and the overall strategy will be periodically reviewed and amended in the light of progress.

A more detailed discussion of the activities required is given in the plan described in the next section.

Exploitation and Evaluation	Integration and Implementation	Innovation and Understanding
A— Identification and delivery of mature methods	D— Develop book-keeping tool support for mature methods	G— Short term directed research to support proposed activities in Enhancement Phase
B— Development of training for skilled practitioners in use of mature methods	E— Industrialisation of promising methods	H— Fundamental research for future
C— Awareness training for large segments of community	F— Prototype tools to support advanced use of mature methods	J— Application specific studies

Figure 1—Overview of the Consolidation Stage

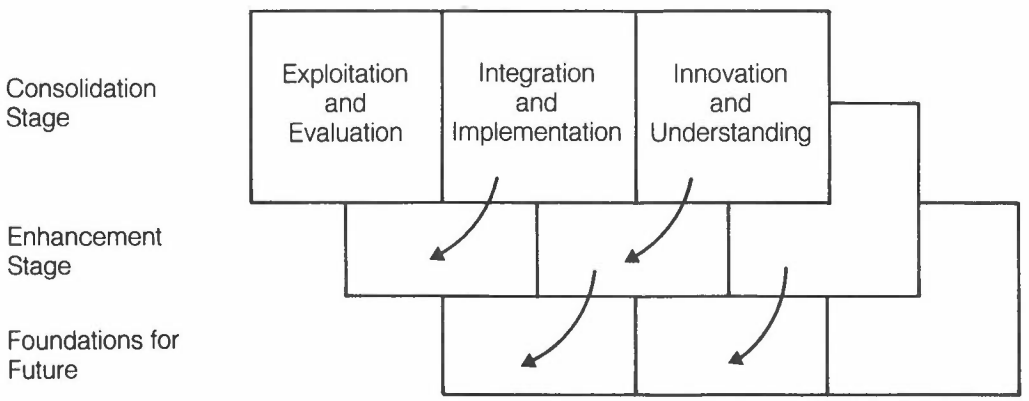


Figure 2—The Relationship Between Various Stages

4.4 Education and Training

It should be clear from the above that the success of the strategy depends on the creation of an environment in UK industry which is receptive to and capable of exploiting technical advances in the formal methods area brought about by the Alvey programme.

The difficulties of bringing about such an environment cannot be overestimated. Currently the fundamental mathematical and scientific concepts which underlie formal methods are understood by very few practising software engineers. Also, only a small proportion of academic computer science courses adequately address these concepts since the majority of academic staff are as ill-equipped as their colleagues in industry. Skills in applying particular formal methods to industrial problems are even more scarce.

These problems can only be tackled by a major initiative in education and training to prepare for the introduction of formal methods. Clearly the costs of this cannot be carried to any great extent by the Alvey programme and must be largely carried by industry itself. Government agencies and departments such as Training Boards, the Department of Education and Science, the Department of Trade & Industry, Universities and the Council for National Academic Awards must play a part by updating courses and curricula and by expanding the skill base in the educational sector. However, the Alvey programme must provide a strong lead to all concerned by:

(a) immediately establishing an awareness programme through publications, workshops and seminars;

(b) developing courses and curricula for the training and education programme, and enabling the provision of such a programme on a professional basis, adequate to meet demand;

(c) insisting that all engineers and their managers working within the Alvey programme undergo adequate education and training appropriate to their present and future needs, and assisting contractors to fund such education and training by allowing it to be charged as project costs.

4.5 Relationships to Other Alvey Areas

In this section the relationships between the Programme for Formal Methods in System Development and other parts of the Alvey Programme are outlined.

First, in section 4.5.1, the interaction with other aspects of the software engineering strategy are considered.

4.5.1 Within The Software Engineering Programme

(a) Reliability and Metrics.

The Formal Methods programme aims to identify successful mature and promising methods for the system life cycle. Thus there has to be close monitoring in the use and application of methods under trial. Such monitoring will be carried out in accordance with the Reliability and Metrics programme. It is also necessary that the reliability and metrics techniques are geared to handle formal methods being applied in the complete system development life cycle. The Metrics programme should provide a means by which critical areas in the project development can be identified, thus defining focal points for the application of more rigorous methods in future. Conversely, it is also likely that the Formal Methods programme will lead to many more objective checkpoints and outputs in the development process, which will in turn provide the basis for more sophisticated measurement and metrics techniques.

(b) Tools.

For the successful transfer of formal methods into industry, good tool support is essential. The technology necessary for tool development for the 1st Generation IPSE already exists, and in fact many book-keeping tools are already emerging.

For later generations of IPSE, the output from the Formal Methods programme will drive much of the tool development. Such tools will be developed through the use of formal methods. Standards will be produced for tools, interfaces, etc., and formal methods should be applied in the definition of such standards.

(c) National Quality Certification Centre and Software Production Centre.

The Formal Methods programme will have an impact upon both the National Quality Certification Centre (NQCC) and the Software Production Centre (SPC). Formal Methods techniques will lay the foundations for the methods used by the NQCC. Furthermore, the SPC will demonstrate the application of techniques developed in the Formal Methods programme.

4.5.2 Other Alvey Advanced Information Technology Areas

All the advanced information technology areas addressed by the Alvey programme are highly dependent on software development, so they will need to apply the methods, skills and tools developed in the Formal Methods programme. Conversely, they will provide a wide range of case studies to be pursued within the Formal Methods programme.

(a) VLSI.

The Formal Methods programme is aimed at system development in general, hardware and software being two possible implementation technologies. Formal methods are essential to the high quality design and production of both hardware and software, hence it is important to ensure that formal methods are transferred to the CAD systems for both VLSI and software. Strong links will be formed between the software engineering and VLSI communities to encourage and enable the transfer of formal methods.

(b) Man-Machine Interface.

Cooperation between formal methods aspects of the Software Engineering Programme and the Man-Machine Interface Programme will again be encouraged. Successful tool support for formal methods certainly requires good man-machine interfaces, for example, high resolution displays can be very beneficial. Again, rigorous development methods play an important role for MMI. Formal models of man-machine interaction may well prove useful in analysing cognitive aspects of the man-machine interface.

(c) Intelligent Knowledge-based Systems.

It is again natural that the output of the software engineering programme plays a most important part in the IKBS programme, not only in the use of formal methods in the development of tools required to support IKBS, but also because of the common interest in theorem proving. IKBS techniques will be applied, where appropriate, throughout the Formal Methods programme.

In the longer term, when sufficient expertise has been gained of the formal basis for system engineering (sufficient to build knowledge bases) and IKBS techniques have matured, it should be possible to apply IKBS techniques alongside formal methods to construct the 3rd Generation IPSE.

(d) Advanced Architectures.

Experimental toolsets for formal methods have made frequent use of novel languages (e.g. applicative or logic programming languages) to handle complex non-numeric data structures. The development of advanced architectures for such languages should therefore facilitate the development of production-quality toolsets for formal methods. Specialised processors to handle particular techniques (e.g. unification) could also be of potential value.

5. Plan

This section describes in more detail the activities which need to be carried out in order to achieve the strategic goals set out in section 4. It presents a refinement of the activities presented in section 4.3 and discusses their inter-relationship and how they should be scheduled.

5.1 The Consolidation Stage

5.1.1 Exploitation and Evaluation

Figure 1 represents an overview of the activities to be carried out in this phase. These are discussed in more detail below:

A—"Identification and delivery of mature methods".

B—"Development of training for skilled practitioners in the use of mature methods".

For this phase two or three mature methods need to be identified, supported and transferred into industry. This requires work to be carried out by two distinct types of organisation—suppliers and users.

Identification

The onus will be on the suppliers of methods to satisfy the Alvey Directorate that their methods are sufficiently mature for short term exploitation.

Support

The suppliers will carry out projects which package the mature methods in preparation for delivery by:

providing training materials to support the transfer of the technology into industry

developing standards for the methods in the areas of language definition and procedures for use

providing consultancy to support the introduction of new methods into industrial applications.

In addition, suppliers will carry out projects to produce support tools of a book-keeping nature, compatible with the 1st Generation of IPSE, though this activity will be carried out under the Integration and Innovation thread of the plan.

Transfer

Users of the mature methods, in collaboration with the suppliers, will carry out projects which:

conduct applications to evaluate the applicability of the methods in selected areas and which will serve as case studies for the community at large. Suitable applications might include, for example, the formal specification and rigorous development of some smaller scale self-contained analysis tools or the development of sequential parts of larger more complex systems. At this early stage in the programme it would be inappropriate to expect full formal verification of the development process but some analysis of consistency of development should be performed. The scale of these projects can be increased when support tools become available.

analyse the costs and benefits of using particular formal methods in particular applications, compared to using other formal methods and traditional approaches. These activities will be carried out in conjunction with the measurement and metrics programme.

C—"Awareness training for large segments of the community".

To achieve this, contracts should be placed with suppliers to:

develop an information dissemination programme through workshops and publications which informs the user community of the mature techniques which have been, or are ready to be, applied in industry

develop an education programme in fundamental concepts to support the transfer of formal methods in general.

In the first instance, study contracts should be placed to investigate the requirements for the awareness and education programme.

5.1.2 Integration and Implementation

D—"Develop book-keeping tool support for mature methods".

The methods identified as mature for this phase of the programme must be supported by automated tools compatible with IPSE1. Projects will be funded for suppliers to develop:

production quality tools which support formal methods in limited ways. Such tools include:

syntax-directed editors and pretty-printers
parsers
consistency and type checking aids
verification condition generators

Such tools must be robust and adequately documented. They will be developed using formal methods and may also serve as case study applications.

E—"Industrialisation of promising methods".

Projects will be funded with suppliers to develop emergent promising methods to the point where they can be considered mature. Such projects will:

conduct laboratory case studies to evaluate in principle the usefulness of the emergent methods

develop stable definitions, languages, documentation and procedures to ensure that the methods can deal with real projects involving teams of users

develop compatible management and measurement techniques (in conjunction with the management and metrics aspects of the Software Engineering programme).

develop prototype support tools to permit more extensive laboratory case studies and which will provide the basis for production quality tools to be developed in the Enhancement Stage of the programme.

F—"Prototype tools to support advanced use of mature methods".

As indicated above, suppliers must be encouraged to produce limited production quality support tools for mature methods as early as possible. For these methods, however, the R & D community must be supported to produce prototype versions of more sophisticated tools. Tools could be developed to assist, for example, in such areas as:

theorem proving
verification
animation
rapid prototyping
executable specifications
algebraic specifications
term rewriting systems
simulation
analysis
proof checking

These projects may well involve applying existing generic tools, of a research nature, to the mature methods.

5.1.3 Innovation and Understanding

G—"Short term directed research to support proposed activities in enhancement phase".

Projects should be placed with researchers to attack fundamental problems for which solutions will be tried out experimentally in the Enhancement Stage of the programme. Areas which are critical for the achievement of the 2.5 year scenario include:

the specification and development of concurrent systems

the technology for supporting reusable system components

the use of formal methods in system definition and requirements engineering and analysis

the use of formal methods in configuration management and system integration

verification and validation (to provide powerful techniques for the proposed National Quality Certification Centre)

proof techniques

privacy and security

H—"Fundamental Research for the Future".

It is important that fundamental research is started to lay the foundations for the long-term scenario described in the "Foundations for the Future" strategic stage.

In addition to further work on the problem areas described under 'short-term directed research' above, research should be started to provide a fundamental understanding of the processes involved and techniques needed in areas such as:

hardware/software integration

real-time and process control systems

performance aspects

transformation techniques

measurement of reliability and correctness using formal methods (in collaboration with the metrics and measurement aspects of the programme)

formal methods for product enhancement and evolution

new language concepts to realise the advantages of formal methods

the application of IKBS techniques (in collaboration with the Alvey IKBS programme)

J—"Application Specific Studies".

In addition to the research streams described above, complementary research streams need to be started to look at the place of formal methods in particular application areas (e.g. graphics, communication protocols). These studies will explore issues such as:

scenarios for introducing and applying formal methods

requirements for formalism, methods and tools

suitability of mature methods

migration paths from the existing scenario to carry over and develop the existing investment in designs, systems and software and the existing skill base.

5.2 The Enhancement Stage

This stage will be directed towards achieving the transition from the Two and Half Year Scenario to the Five Year Scenario described in section 3.2. During this stage methods identified as promising during the Consolidation Stage will have matured and be ready for exploitation.

Similarly, the directed research carried out in the Consolidation Stage will have laid the foundation for new promising methods which will undergo integration and implementation activities during this stage. The relationship between the activities of the Consolidation and Enhancement Stages is illustrated in Figure 2. Detailed plans for this stage will develop as the Consolidation Stage progresses. Overviews of the various streams of activity are given below:

5.2.1 Exploitation and Evaluation

The activities described in section 5.1.1 will be enhanced to handle the extended range of mature methods which will be capable of wider applicability in more phases of the life cycle. The range of case study applications will be extended and the training and education programmes scaled up to cope with the more advanced methods and the greater numbers of engineers which will need to be trained.

5.2.2 Integration and Implementation

Production quality versions of extended tool sets for the established mature methods and limited tool sets for the emergent mature methods will be developed and made available by suppliers. In this stage, tools will be integrated within the IPSE 2 framework. The emergent methods will be further industrialised to support their introduction into full-scale use.

5.2.3 Innovation and Understanding

Directed research will be carried out to meet requirements identified for activity beyond the five year scenario in order to lay the foundation for the Information System Factory (IPSE 3) goal. The results of research carried out in other parts of the Alvey Software Engineering Programme will be integrated in this phase.

Fundamental research into problem areas such as those outlined in section 5.1.3 will continue in order to produce more complete solutions.

6. Concluding Remarks

This strategy document will be periodically reviewed as the Alvey programme progresses and revised drafts will be issued. The Alvey Directorate and members of the panel will be pleased to receive comments and suggestions to assist in the updating process.

7. References

1. Alvey Directorate
   Alvey Software Engineering—A Strategy Overview DTI publication, November 1983.

2. Talbot D., Witty R.W.
   Alvey Programme—Software Engineering Strategy, November 1983.

Appendix—Names and Addresses of Members

Mr. M.I. Jackson (Chairman),
Standard Telecommunication Laboratories Ltd.,
London Road,
Harlow,
Essex CM17 9NA.

Mr. P.W. Dell (Acting Secretary),
British Telecom,
Research Laboratories,
Martlesham Heath,
Ipswich IP5 7RE.

Dr. H. Barringer,
Department of Computer Science,
University of Manchester,
Oxford Road,
Manchester M13 9PL.

Mr. J. Elmore,
Head of Division SES11,
British Telecom,
St. Vincent House,
1 Cutler Street,
Ipswich IP1 1UX.

Mr. S. Goodenough,
Royal Signals and Radar Establishment,
Ministry of Defence,
St. Andrews Road,
Great Malvern,
Worcs. WR14 3PS.

Professor C.A.R. Hoare,
Programming Research Group,
University of Oxford,
8-11 Keble Road,
Oxford.

Dr. T. Maibaum,
Department of Computing,
Imperial College of Science & Technology,
180 Queens Gate,
London SW7 2BZ.

Mr. D. Morgan,
Research Director (Software),
Plessey Electronic Systems Research Ltd.,
Roke Manor,
Romsey,
Hants. SO5 0ZN.

Dr. G. Plotkin,
Department of Computer Science,
University of Edinburgh,
Kings Buildings,
Mayfield Road,
Edinburgh EH9 3JZ.

Dr. R. Snowden,
ICL,
Westfields,
West Avenue,
Kidsgrove,
Stoke-on-Trent ST7 1TL.

Mr. P. Sizer,
Group Chief,
Software Research & Development Group,
Marconi Research Centre,
East Hanningfield Road,
Great Baddow,
Chelmsford,
Essex CM2 8HN.

Professor C.B. Jones (EARB C & C Committee Liaison),
Department of Computer Science,
University of Manchester,
Oxford Road,
Manchester M13 9PL.

Mr. D. Talbot (Alvey Directorate),
Department of Trade & Industry,
Millbank Tower,
Millbank,
London SW1P 4QU.

Dr. R.W. Witty (Alvey Directorate),
Science & Engineering Research Council,
Rutherford Appleton Laboratory,
Computing Division,
Atlas Centre,
Chilton,
Didcot,
Oxfordshire OX11 0QX.