IABSE
AIPC
IVBH

COLLOQUIUM on:
"INTERFACE BETWEEN COMPUTING AND DESIGN IN STRUCTURAL ENGINEERING"
August 30, 31 - September 1, 1978 - ISMES - BERGAMO (ITALY)

Strategy and Organisation for System Design

Stratégie et organisation du calcul de systèmes

Strategie und Organisation von System-Berechnung

A.I. LLEWELYN
Former Director of Computer Aided Design Centre, Cambridge, U.K.
Expert for CAD — Commission of the European Communities,
Brussels, Belgium

Summary
This paper briefly refers to the genesis and development of computers and against this background examines the needs of system development in the light of the lessons of experience. It is argued that the problems are in the main conceptual and organisational. The resulting implications are described and the importance of collective action is stressed. The impact on industry and the engineering professions, particularly in relation to education, is then discussed with some comments on the need for realism in respect of time scales and some suggestions for future progress.

Résumé
L'article retrace l'histoire du développement des ordinateurs et examine, à la lumière des expériences faites, les besoins pour un développement systématique. L' auteur prétend que les problèmes sont au niveau du concept et de l'organisation. Les conséquences qui en résultent sont décrites; l' importance d'une action commune est soulignée. L' influence de l'industrie sur les professions d' ingénieurs et particulièrement en relation avec leur éducation est présentée; quelques commentaires sont donnés sur la nécessité de rester réalistes par rapport à l'échelle du temps et quelques propositions sont suggérées pour des progrès futurs.

Zusammenfassung
Es wird über die Entwicklung von Computern berichtet und anhand der Erfahrungen Bedürfnisse für eine systematische Entwicklung geprüft. Es wird behauptet, dass die Probleme im Hauptkonzept und in der Organisation liegen. Die Folgen werden beschrieben und die Wichtigkeit einer gemeinsammen Aktion wird unterstrichen. Der Einfluss auf die Industrie und auf die Ingenieurberufe - besonders im Zusammenhang mit der Ausbildung - werden besprochen. Es wird mehr Realisums in Bezug auf Zeitskalen gefordert und es werden einige Vorschläge für zukünftige Fortschritte unterbreitet.

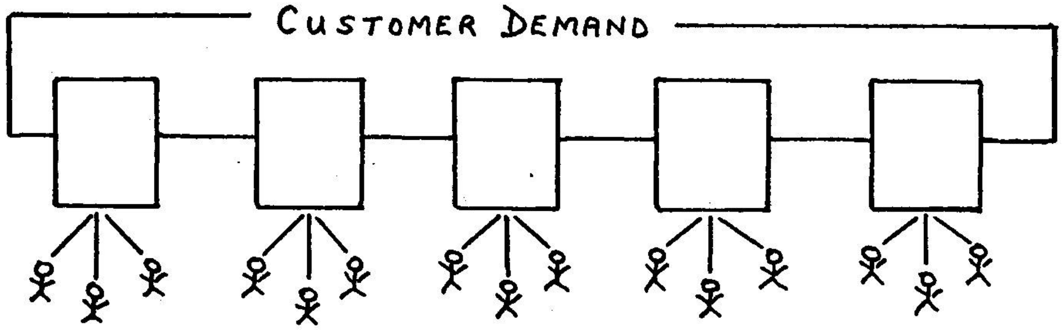
Historically, computers were born as a result of military need during a period of world-wide conflict. It was significant that this resulted from the first attempt in human history to apply science and technology on a large scale using mixed multi-discipline teams of scientists and engineers working with users in a collective and co-ordinated fashion. Being born out of the necessity of war, their initial development was for code-cracking and to meet the computational needs of atomic energy and nuclear weapons. For a decade and more, development was therefore set firmly on one path, that of "number-crunching" and analysis with continued emphasis on bigger and better hardware with funds and support coming from governments through defence contracts. As a result, all the early civil users of the computer struggled to make effective use of uneconomic and unsuitable hardware.

Although subsequent developments of specialised and advanced military systems produced dramatic improvements in hardware and software technology, conceptual understanding of the real use of the computer for linking people and processes and the overwhelming importance of software was set back by a generation. The legacy is with us today in the psychological bloc which makes engineers and firms analyse with great care marginal and trivial differences in hardware costs whilst ignoring and not understanding the implications of software and organisational costs which are greater by orders of magnitude. For example, hardware needs to be depreciated and replaced fairly rapidly but software has to last much longer and be capable of evolutionary enhancement in order to show a return on a much larger investment. Of course, a wider issue is also involved, namely the inability of cost-accounting methods to value software assets which leads to gross distortions in the balance sheet and affects general attitudes!

In the 35 years since the birth of the computer, relatively little organised funding has been devoted to developing either the understanding or the hardware and software technology best suited to civil and industrial use. Military developments and massive funding, sustained over a quarter of a century, have in contrast produced very powerful systems and major improvements in hardware and software technology together with a tremendous amount of "know-how". Many of these benefits have in the course of time been carried over and are responsible for the growth of the civil market but the full potential is far from being realised. However, there are still lessons to be learnt in the organisation of users and suppliers which, if properly applied, could help to transform industries and help to hold the balance between social and economic need. It would, however, be totally unrealistic to expect this to happen quickly, or cheaply, or within time-scales that politicians are constrained to work to.

The points to which I wish to draw attention are these: advanced systems of a modular kind with quick reaction capability were a result of user involvement in a mixed discipline team environment. Even so, they required support and funding over a long period of time, together with good project management and an organisation based on a common understanding. I want to return to these points at the end of my talk, after I have developed the main theme, when I will make some suggestions for future progress.

First, I want to use a simple illustration to introduce the nature of the organisational problem, the notion of a "system"and the implications for computer usage.



Any organised activity is broken down into a series of tasks with teams or people each contributing some specialist skill and experience. The overall objective is to produce something which satisfied customers and in this simple sense a total or integrated system links everything together with requirements as input and the end product as output. Computers are able to be used, and are being used, at many points in this total system - the problem is how to ensure that they link men and processes more effectively.

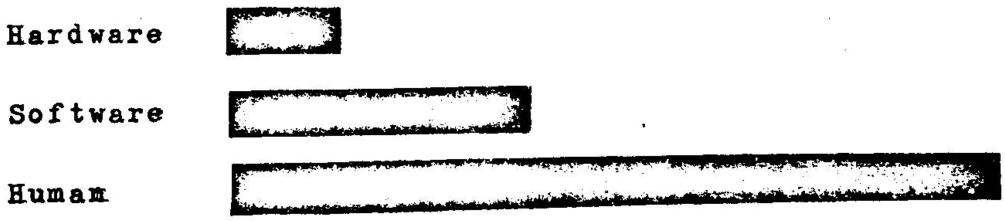
Clearly, if there is no co-ordination, the result is a lot of computer usage for isolated tasks which is only marginally productive since the major gains result from a common data-base and easy transfer of data without change of format from one computer to another. Many attempts to produce a total integrated system have, however, failed - the problems are too large and complex for this overall approach to be successful. The practical alternative, I suggest, is to visualise the integrated system as gradually evolving by the link-up of a series of manageable sub-systems. This of course requires some common agreement on the concept and a frame-work within which the separate developments can proceed so that they can subsequently be interfaced. The concept is attractive since it can be applied beyond one organisation to whole sectors of industry with the advantage that efficient data exchange between organisations would allow easier re-structuring of industry and facilitate new groupings and growth of specialised small firms.

More flexible groupings of men and machines would also be an attractive alternative to massive capital investment in automated production of dedicated high volume products.

The computer holds the key to these desirable goals provided we can arrive at a common understanding of the nature of the problems to be tackled and can combine resources and expertise in a sustained attack upon them. This is essential because many of the sub-systems we are talking about are 100 man-year projects even with the right team mix.

Let us now consider the nature of these problems and how they may be tackled. Any computer system is an assembly of hardware and software modules and it is now well understood that on a scale of cost or difficulty hardware is the least troublesome factor.

What is not understood is that the human or organisational factor in producing a usable and acceptable system is an even greater problem both in its own right and also because it interacts with and has to be incorporated in the software design. One might convey this point dramatically but not I think unrealistically by the following illustration:-



The real difficulty with software is not writing it but first deciding what the specification should be and where the system interfaces should be. This requires an in-depth understanding of the organisational procedures and professional practice in any given situation which of course have been gradually built up over a great many years. Naturally, for any application, this sort of experience resides with the users and not with system designers. In this situation there are three possible strategies:-

1. Hardware and software experts (and this includes manufacturers and suppliers) can try to gain sufficient experience of practical problems and organisational procedures to produce either general purpose systems or specialised application systems.

2. The users can try to develop expertise in programming and system design by forming in-house teams to assemble their own systems on basic hardware supplied by the manufacturers.

3. Joint development teams can be formed to deliberately bring different disciplines together, with users in project or consortium arrangements.

Attempts by independent research groups to produce integrated systems have failed as have similar attempts by suppliers of hardware and software systems but for different reasons. The former did not have user involvement and could not embrace enough application experience whilst the latter were additionally constrained by the need to produce general purpose systems in order to sell into the largest market and hence satisfied nobody. In fact, and this applies particularly to engineering design, they really sold expensive basic kits which required extensive in-house development before they were usable and productive. It has for example been estimated that between 70 and 90 per cent of all software in use has been produced by the user. (Ref. 1)

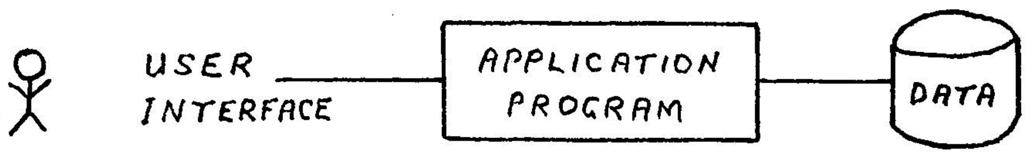
The first approach has, therefore, led to the break-up of the old computer industry and the formation of a software industry specialising in application systems and tailoring general purpose systems to suit individual requirements. The software industry has recently been given a new impetus by the emergence of new manufacturers offering mini-computers and micro-processors which with suitable software can be sold as systems and interfaced to computer systems already in use., It is encouraging to note these new market developments but at the same time it would be foolish not to recognise the danger that the new industry could repeat some of the mistakes originally made by the computer industry. This danger is the greater because of the enormous gap in understanding which runs right across industry, the professions, and even in education itself. In consequence market growth is still limited and this, together with low profit margins, dictates short term objectives and inhibits longer term investment in common software tools and improved system design techniques.

The second approach - in-house development has to date proved the most successful road to usable and productive systems. Unfortunately, whilst in the long term this strategy pays off for the particular firm, the resulting system is dedicated to the environment in which it was developed and not easy to transfer to other firms or other sectors of industry. This strategy, therefore, because of the cost, expertise and time scale involved, is only feasible for the larger organisations and cannot be regarded as a general or long-term solution - certainly not for the smaller firm or engineering practice.

The advent of the micro-processor might seem to change this situation but cheap hardware only extends the range of possibilities and does nothing to change the basic software and system problem. Indeed, the danger is that a surge of "do-it-yourself" applications could result in making it more difficult to use distributed computer power effectively. Analogies are always dangerous but one is tempted to compare the situation with the early growth of railways or use of electric power. No real progress could be made in either case until some common unifying standards were introduced.

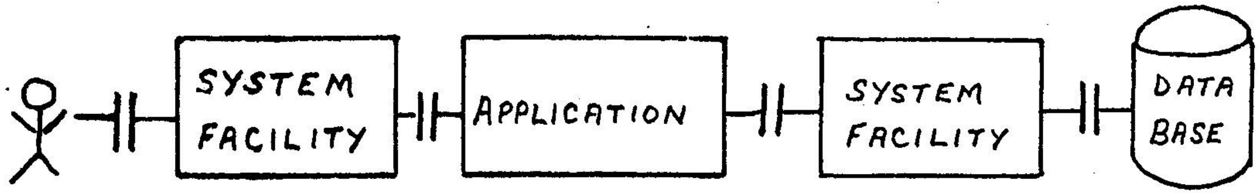
The third approach is a relatively recent development originating in the public sector or associated with professional institutions acting as collective centres. It is now beginning to attract the attention of some governments and industry as a cost-effective method of making progress and this could be the key to more rapid progress in future. I will later show that this approach has attractions as a possible strategy for Europe but first I want to consider the organisational implications of system design.

Traditionally, system design has taken either a "top-down" or a "bottom-up" approach reflecting control either by the system expert or the user. The latter tends to concentrate on an application and results in a dedicated solution with complex software which is difficult to maintain and not extendable.



No "bottom-up" solution can last and the lesson of experience points to the need to look at the total solution in order to home in on a particular solution which can then be organised with the benefit of evolution. (Ref. 2)

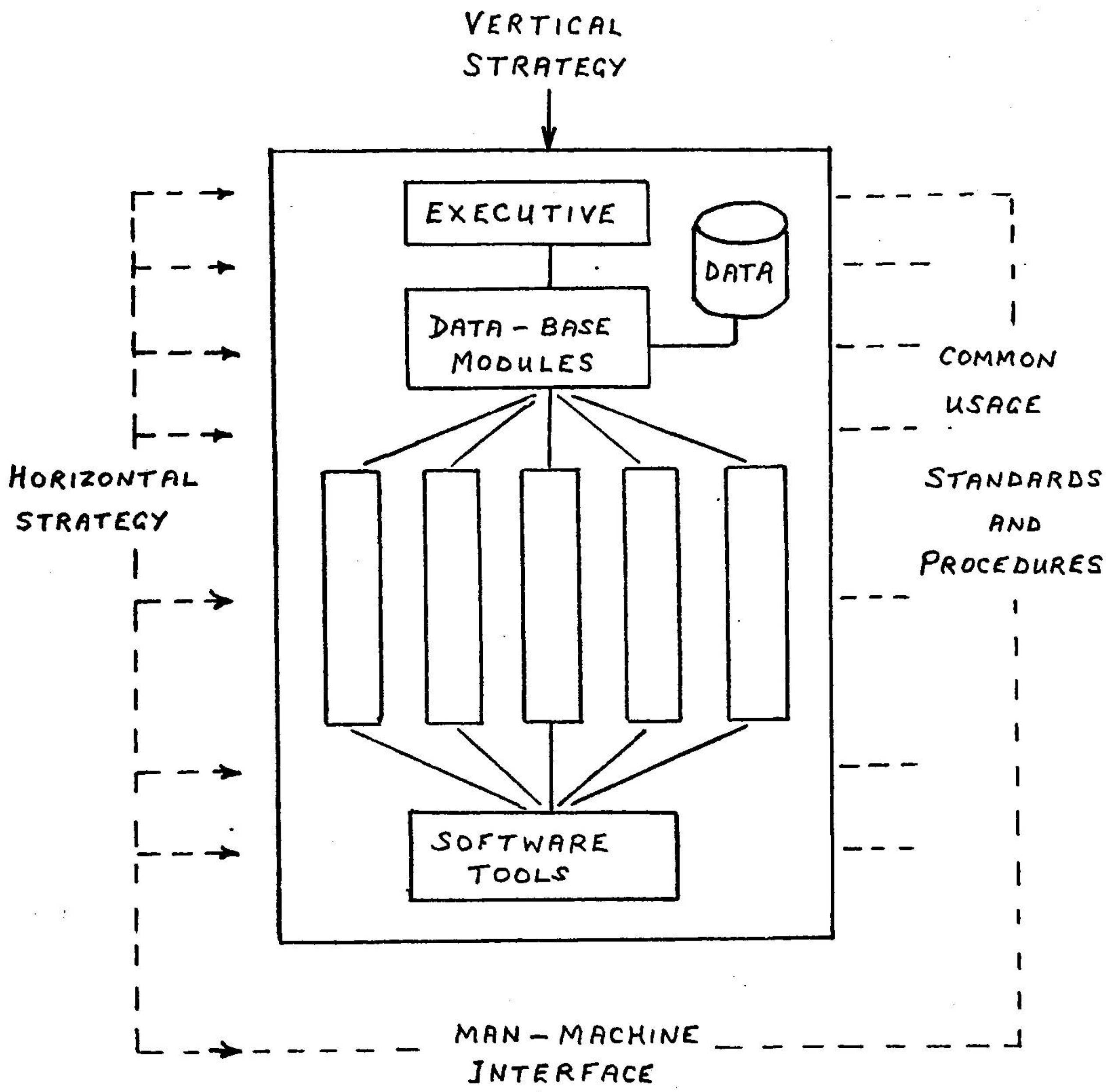
The "top-down" approach recognises this need and it is now generally agreed that a system design and structure which allows replacement and evolutionary enhancement of modular components is the best answer. The need for a software engineering approach was identified in the late sixties (Ref. 3) but it has taken almost a decade for a common understanding to develop.



This modular concept has the great advantage that common functions can be identified as system level facilities. Software tools can be provided to enable the user to modify and develop his own system as his needs change through time. Specifically to allow him to generate his own language and special application programmes (Ref. 4) which would fit the system structure and preserve total integrity. There is sufficient evidence to show that the technical software problems for interfacing modules and generating programs to meet application requirements within system design criteria are not likely to be barriers to progress.

The real problems are mainly organisational and educational. The structural design, positioning of interfaces and the content of the system facilities can only be successfully carried out on the basis of the actual engineering experience of many applications which by definition are only obtained at the point of user application. To be useful, this experience must be organised as user involvement in joint project development and later fed back through collective centres into improved system design using common infrastructure. This is not easy because there is no central discipline to take the responsibility. Experts in CAD, or interactive design systems, are normally labelled and operate under one of the traditional disciplines and only meet for intermittent exchanges of information. They may pass resolutions and agree common action but there is no mechanism for follow-up action and no cohesive strategy. These deficiencies are in part being remedied by collective strategies adopted by different countries and this now provides a new opportunity for European co-operation.

To illustrate what this means in practice, consider the following system structure:-



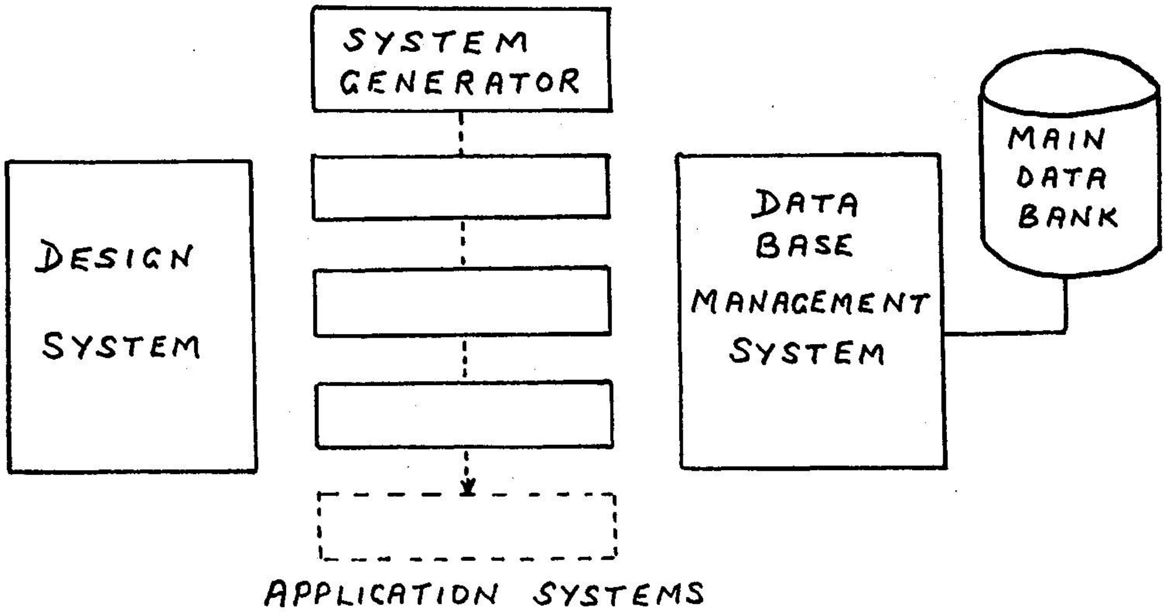
The block represents one of the sub-systems in an integrated system but is self-contained and can be used alone. For reasons which should by now be clear no one organisation has the necessary expertise to undertake all the design and development but many different groups can be chosen or organised into project groups to undertake those aspects of the work for which they are especially suited. Provided there is some central focus for overall design strategy, and preferably a lubricating source of funding, such co-operative activity has been tested and proven successful. Indeed it is a very flexible and powerful method which consolidates and makes maximum use of all available application software (in a recognised high level language) and allows virtually open-ended development.

An important feature is the ability to assemble systems to suit different requirements and cater for different stages in development and use in firms and also countries. For example, different countries might wish to standardise on particular application programs or system modules or an individual firm to carry on using existing programs. Provided there is agreement on interface and data linking rules this type of alternative substitution would be possible and one can avoid the major conflicts which have caused previous attempts at co-operation to fail.

This represents the vertical strategy, to make effective use of whatever is available at any point in time, to meet immediate needs, and cater for change and growth.

The horizontal strategy is to agree and disseminate a common understanding of the basic rules for interfacing and data-flow which define the system infra-structure and allow independent generation of application programs with multiple use of common system facilities. For example, to ensure that data output from any program can be input to another and that data can flow both within the sub-system and to other linked systems, between private data-bases and shared data-banks.

Although the solution to these desirable aims will take many years to accomplish there is a growing awareness of what needs to be done. (Refs. 5, 6 & 7). Fortunately, this is consistent with the needs of engineering and industrial users. The larger user is now confronted with the problem of linking his separate computer systems and task-oriented application teams together in a more productive way and controlling further software development so that it conforms to data and system design standards. (Ref. 8). A simple representation of this concept is shown below:



The pressure to achieve a common framework, largely independent of hardware and application, is rendered more acute by cheap hardware and increasing costs of software. The cost however in time and resources (up to 100 man-years) is not trivial, even for the largest organisation, and although the benefits and return on the investment may be high there are other considerations. For example, no organisation can act independently - sub-contractors and specialist suppliers are also involved in the process of design and manufacture and for this reason common data interchange on an industry-wide basis has many attractions. Indeed, it is interesting to note the gradual evolution of project and consortium type activity directed to this end. Some early attempts took place in the U.K. but more powerful undertakings are now under way in Germany (Ref. 9), Japan and the U.S.A. (Ref.10). In all cases an external catalyst has seemed necessary in the shape of government or a public-funded organisation.

Whilst advanced projects and the pace setting activities of the larger organisations tend to dominate discussion the uncommitted sectors of industry, the non-users, are by far the majority and in every country. To meet their needs a sustained educational and re-training program should be formulated based on the simplest start with a skeletal system capable of gradual development and benefiting from the results of a co-ordinated horizontal strategy program. In this way, technology can be transferred from the advanced sectors and limited resources deployed to the greatest effect.

All countries seem to be in agreement on the need for positive action to create greater awareness and understanding at all levels. (Ref. 11) and recognise the yawning gap which exists between the practical design engineering experience in industry and the output from educational establishments with only a small fraction having any CAD knowledge or capability.

Suggestions to remedy this situation include:

- Appreciation courses for senior people in industry, the professions and education.

- Series of workshop "teach-in" seminars to disseminate "know-how".

- Re-training facilities to allow experienced engineers to become familiar, in their own pace and time, with usage of CAD systems.

- The incorporation of CAD as part of fundamental teaching in depth in every discipline.

- Establishment of courses with an agreed syllabus to turn out competent system designers.

- Establishment of courses of vocational training to turn out good practitioners and expert users of design systems.

- Arrangements to allow selected commercial systems to be used in education for practical use in real life situations.

- Greater involvement of industry and education in joint project work.

- Use of T.V. media to create a climate of understanding and facilitate general acceptance.

A comprehensive programme on these lines should form the foundation of a horizontal strategy organised for the whole community and sustained with continuity over many years. It would be a great mistake to think only in the short term.

Another important element in the horizontal strategy would be forward looking research of a collaborative kind aimed at removing known deficiencies which are obstacles to future progress. One such item can be identified - indeed no discussion of design systems would be complete without some reference to their current limitations and the need for further research and development to provide the logic rules and search strategies to allow problem solving to be carried out in the full creative sense. Currently interactive systems do a good job of evaluation and analysis and generation of data when most of the important decisions are taken. The real problem, however, is to explore the extreme limits of possibility in use of new materials and processes outside previous experience. Normally the designer is constrained by his own limited experience and does not venture, unless prompted, into unfamiliar regions which may, however, hold the most innovative solution (Ref. 12). This is a limitation on man's own creative processes and hence a combination of man and computer, properly organised, should be a definitive improvement in this kind of conceptual thinking. Suitable techniques are likely to be derived from work in operational research (Ref. 14) and artificial intelligence (Ref. 15).

In conclusion, let me summarise my main thesis:

- Major advances occur when time and circumstances provide the opportunity for a multi-discipline effort toward an agreed objective.

- The opportunity arises for a new endeavour in Europe because the use of computers for new systems of design and manufacture is in the formative stage, the dominant computer industry is breaking up, new technology is available, old industries are dying and need transformation and new growth to ensure their survival and provide future employment.

- Most of the historic examples to date are in the military rather than the civil field but the lessons apply.

- The key prerequisites for success are a coherent strategy, a clear programme, good organisation, user involvement, and a catalyst in the shape of a public benefactor.

- The European Commission can act as the catalyst and provide a focus for a co-operative programme to link up and re-inforce the strategies now being employed by member countries.

- The dissemination and growth of a common understanding on strategy, the promotion of basic system design conventions and standards together with a sustained drive to improve education in industry and educational institutions would form the initial priorities.

Finally, since this colloquium is concerned with one branch of engineering, let me answer one obvious question.

The engineering professions stand to gain from participation in co-operative action of the kind I have described in a number of ways. Involvement helps gain acceptance. The status of the engineer would be firmly established. Control of systems vital to future design and manufacture in industry would be regained.

As I have explained, the engineering contribution is essential to determine the software content and how the individual modules should be designed and fitted together. The system design rules and basic standards are essentially engineering considerations and unless theyare established by the engineer the temptation will be for systems to become automated and not aided! This is a real danger which would be disastrous not only for the engineering professions but for industry and society. The need to agree basic standards should not be seen as limiting choice but rather as establishing a framework to allow users maximum flexibility to create their own systems.

Another aspect which the engineering professions should appreciate is the need to capture past experience and pass on hard-won knowledge. In the past this was done through documentation and apprenticeships. Now it is possible to embody and pass on this expertise through software facilities and data-banks which will only be accessible and meaningful if the engineering professions seize the opportunity to play a leading role in designing the systems and establishing a new infra-structure for the future.

REFERENCES

1. QUITTNER, P.: The International Software Market - European Computing Conference on Software Systems Engineering - Online 1976.

2. LEHMAN, M.M. & PARR, F.N.: Program evolution dynamics and Project Management - European Conference on Software Systems Engineering - Online 1976.

3. Reports on Software Engineering - International Conferences 1969 and 1970 sponsored by NATO Science Committee.

4. PRYWES, N.S. - Automatic Generation of Programs - European Computing Conference on Software Systems Engineering - Online 1976.

5. WINTER, P. & NEWELL, R.G. - Data Bases in Design Management - Chemical Engineering Progress(Vol.73, No.6.).

6. WINTER, P. - An integrated approach to Chemical Plant Design - US-JAPAN Joint Seminar - June 1975 - Kyoto, Japan.

7. ABEL, L.C. Structure and Foundations of a large multi-user, multi-task CAD system - European Computing Conference on Software Systems Engineering - Online 1976.

8. BISHOP, A.W. - A Framework for distributed design computing - CAD 78 - IPC Science & Technology Press.

9. CAD - Berichte - das Forderprojekt CAD Staad Juli 1978, Kernforschungszentrum, Karlsruhe.

10. MILLER, R.E. - Integrated programs for Aerospace - Vehicle Design (IPAD) - Interactive Design Systems Conference, April 1977 - IPC Science & Technology Press for the CAD Centre.

11. Proceedings of International Conference on Computer Aided Design Education held at Teeside Polytechnic - July 1977 - to be published by IPC Science and Technology Press.

12. NEVILLE, G.E. & CROWE, R.A. - Computer Augmented Conceptual Design - Proceedings of the Symposium on Basic Questions of Design Theory - organised by National Science Foundation (USA) 1974 published by North Holland/American Elsevier

13. WESTERBERG, A.W., STEPHANOPOULUS, G. & SHAH, J. - The Synthesis Problem with some thoughts on Evolutionary Synthesis in the Design of Engineering Systems - as for 12.

14. BRENNECKE, W. - Optimisation through "Branch and Judge" - Main features and application to Roof Structures - CAD 78 - IPC Science and Technology Press.

15. Proceedings of IFIP Symposium (Grenoble) - CAD and Artificial Intelligence - to be published by North Holland.