Zeitschrift: IABSE reports of the working commissions = Rapports des commissions de travail AIPC = IVBH Berichte der Arbeitskommissionen

Band: 31 (1978)

Artikel: The computer usage environment for structural engineers

Autor: Tomino, H.

DOI: https://doi.org/10.5169/seals-24908

IABSE
A I P C
I V B H

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

The Computer Usage Environment for Structural Engineers

Utilisation des ordinateurs par les ingénieurs civils

Anwendungsbereich von Computern im Bauingenieurwesen

H. TOMINO

Director
Kozo Keikaku Eng. Inc.
Tokyo, Japan

Summary

The paper presents the observation of recent computer usage in the field of structural design, and discusses the problems involved in development of software in this area and its usage. Discussion also covers a comparison of trends in various usage environments, such as traditional batch, on-line TSS, mini computer systems and combinations thereof, and suggests further pursuance of better computer usage environment for the professions.

Résumé

L'article présente les observations faites récemment sur l' utilisation des ordinateurs dans le domaine du projet des constructions de génie civil, et discute les problèmes impliqués par le développement du software en ce domaine et par son utilisation. La discussion couvre aussi une comparaison des tendances dans les domaines tels que "traditional batch", "on-line TSS", des mini ordinateurs et des combinations de ceux-ci, suggère l' extension de l'utilisation des ordinateurs dans la profession.

Zusammenfassung

Der Artikel stellt Beobachtungen in der Computeranwendung vor, die auf dem Gebiet des Bauentwurfs in der letzten Zeit gemacht worden sind. Er untersucht Probleme, die sich aus der Entwicklung und der Anwendung der Software auf diesem Gebiet ergeben. Die Diskussion erstreckt sich auch auf den Vergleich von Tendenzen in verschiedenen Anwendungsbereichen, wie z. B. "traditional batch", "on-line TSS", kleincomputer systeme und Kombinationen davon, und schlägt die weitere Entwicklung für bessere Anwendungsbereiche für Computer in diesem Berufszweig vor.

INTRODUCTION

Our structural engineering practices in Japan has merely 15 years of experience in computer usage and no commercial sector in the world seems to have far more than 20 years of practical use of electronic digital computer in structural engineering field. However, with tremendous development of computer hardware and remarkable improvement of its usage environment, this powerful tool has already hold definite position in dayly structural engineer's works. These days, throughout the world, structural design work can hardly ever be performed productively without electronic digital computer.

The development of computer usage environment in our country has been promoted remarkably by the introduction of remote accessing in late 1960s and dedicated systems using mini computer in early 70s. On the other hand, in the field of structural analysis method, an introduction of matrix methods represented by Displacement method, Force method and particularly Finite Element Method had brought remarkable changes into our dayly structural analysis work. Moreover dynamic analysis concept introduced early 1960s with above mentioned analysis methods has given tremendous impact to our structural design principles.

Certainly, the introduction of computer has made changes of our structural engineers work. But it does not necessarily result simple man power saving or design cost cut down, although it has released structural engineers from tiring manual calculation to the great extent. It offers better opportunity for us of repetitive design trial for more careful design by fast turn arround. It has made us possible to investigate behaviors of structure under various conditions more closely using advanced analytical method, which we can hardly accomplish without this powerful tool. It is naturally conceivable that computer use in such manner often ends up with increase of structural design cost, however, we can enjoy such an opportunity for better accomplishment of structural engineers professional responsibility.

However, to the contrary, we cannot neglect certain adverse effect. Namely, some engineers has become too heavily numerical analysis oriented. Unexperienced engineers tends to design structures arround their knowledge of codes or regulations and computer analysis then care less to the past valuable experience of our profession.

Another problems we will be facing are increase in software development and maintenance cost, and promotion for proper usage of programs. This paper present observation of various computer usage environments and address to above mentioned problems.

SOFTWARE DEVELOPMENT COST

Hardware cost-performance has been greatly improved and still being improved. However, software development cost which is highly dependent to human cost has been increased and followings the extension of computer usage, application systems has become larger in size and more complex. Also it has become more clear that maintenance cost of generally used software is far more than anticipated. It is said that the maintenance cost of widely used application system may be estimated as 30 - 50% of total development cost.

Structural analysis and design program can not be the exception on this regard;

Rather, our past experience in our firm, although we do not have good statistical support at present yet, suggest that structural analysis program are more costly in maintenance. Because structural analysis programs often requires large memory area and CPU time, computer cost tends to become higher for investigation of bug, confirmation of correction and anti-degrading varidation.

It is also true that because of unexpected slow progress of software engineering, reliability and quality of software is still quite difficult to evaluate clearly and this makes maintenance problem more complex. Another problem regarding software development for private organization is increasing technical complexity of software system and organizational difficulty to keep such maintenance support personell always available.

Now a days, it is true that almost every young structural engineer can write computer programs. However, by the development of computer hardware and progress of software technique including numerical or other analysis technique, wide variety of high technical knowledge and more sophistication has been required to develop good structural engineering software, i.e., system building in this field also has become more professionalized. And to keep such sophisticated software alive and useful, there should be capable maintenance supporting personell. Because if maintenance work is not in any reason properly performed then such software will gradually become dead. No one can make good application software by one shot. [1] It is our experience that good application software only become available after certain period of practical use by engineers and changes, alternations, upgrade made there of.

It has been a particular nature of structural engineers society in the past, that being quite agressive in computer usage, to develop all necessary software by structural engineer's own effort, just with minor exceptions. There are so many structural engineers who are merely developing programs, but structural analysis and design programs are merely a tool regardless whatever capability it has and how useful they are. Then why we structural engineers should make tools by ourselves? We have already spent so much of the best resources of the profession to do present days job. [2]

TABLE - 1 Man Power distribution of software development Project

	Program development phase					
	System design	Functional design	Detail design	Coding debugging & module test	system test	project management, etc.
man-power distribution * 1	20 %	24 %		35 %	10 %	11 %
* 2	20 - 25 %	10 %	15 %	25 - 30 %	15 %	10 %
expected contribution of structural engineers	essential	desirable			desirable	
	current general coverage by structural engineers					

Note: * 1 average from 11 Kozo's Fortran project (total 90K steps) in 1974, 75.

* 2 currently used value for project scheduling of engineering application program development in KOZO.

Table-1 shows some data gathered in our firm regarding man power distribution. In most cases of the past, it was common that system design group was also responsible for implementation, however, for more effective use of technical resources, KOZO has been trying to segment development phases and distribute responsibility of corresponding work to several groups. It is my understanding through such trial that over 50% of software development burden better be carried by software people. I believe we structural engineers should get more cooperation from computer and software industry, although we carry rather little weight with the computer industry. We should define our requirements more clearly, establish communication and let software professions develop our tools.

Our another current software development problem is the duplicated investment and this will be discussed in the latter part of the paper.

PROGRAM USAGE PROMOTION AND CONSULTING SUPPORT

As program in structural engineering field become more sophisticated and covers various features and complex functions, some kinds of back up support for "how to use xxx program" becomes necessary. There are so many program, developed but not effectively used in the organization because of poor usage promotion and support. Naturally, engineers are reluctant to use what they can not fully understand. And, in the past, program manuals and other supporting materials for use were generally very poor.

Sometimes it is not an easy job to understand exactly what features a program has and how to prepare input data productively from available documents and this fact discourages engineers from using new programs. Really, it often means too much for a practical engineer to study the usage of certain program which he uses once in a while. As the matter of fact, even for a small program, usage consulting support is undoubtedly needed for the effective use of such software resources in the organization, more attention should be paid in this regard. These consulting support cost is often covert in the organization and then neglected, however, such cost should seriously be considered together with software development and maintenance cost for a realistic feasibility analysis. If such cost is not properly budgeted and materialized such development investment will have very high possibility of becoming developers personal property.

TABLE - 2 Average man-day for data input including modeling and data preparation

(in case of certain batch FEM program with simple mesh generation feature)

structure type	average man-day/100 nodes
2 dimensional (plate)	2 - 2.7
3 dimensional solid - isoparametric	3 - 4
shell - ditto	3 - 4

Note; above figures were derived from Kozo's experience in 1977 - 78.

Even after program features and data input are reasonably understood, still input data preparation is time consuming job for certain problems. Table-2 shows some experience in our firm regarding average data input man power. Herewith input data preparation includes modeling of structures for analysis and corresponding data preparation. Such man power needed for data preparation is heavily dependent to the engineer's skill of the program usage and his capability for the problem solving of the particular problem to be analyzed. And computer time required for the solution is even more dependent to such skill or capability;

In any cases such amount of effort required for data preparation as shown in Table-2 is just too much and average practical engineers cannot afford to spend his own time for this purpose. However, it is obvious that suitable assumption for analysis modeling can decrease computer time to a great extent yet the result can be more consistent and reliable. Then application consulting of structural analysis or design programs will become the new sector of our job.

Those engineers who will work in this area has to have sufficient knowledge and experience for the usage of particular program and should perform following services;

- Suitable modeling of the structure along analysis or design objectives and economically feasible.
- Decision of corresponding various engineering constants or coefficients.
- Adaptability check and interpretation of the computer processed result, and engineering evaluation.

In certain area such as stress analysis of complex structure, modeling of structure is awfully important because even with these powerful tool an analysis of full structure model can hardly be justified economically then idealization of structure with certain acceptable engineering assumption should be applied. We should also recognize that these work is not necessarily academic as often misunderstood but must be practical. And we already need such personell who really bridges between computer systems and practical design engineers for more extensive use of computer in our profession. Especially in commercial TSS, this kind of program usage support will become essential to keep customer's hands on the system and in private organizations as well, not to waste software development investment.

Needless to say, functional coverage and correctness of program are very important, however, availability of program usage consulting support will soon become the key factor of deciding what particular program usage environment we engineer should choose as user.

TIME SHARING SERVICE (TSS) ENVIRONMENT

First commercial real time TSS in Japan has started back in 1971 by Nippon Telephone and Telegram Public Corporations (NTTP) as DEMOS system. Since then over ten commercial TSS Companies are offering services in the market.

It is often explained that the merit of TSS for users is :
"Users can utilize very large computer system which any single user can hardly afford to use as his own, through terminal devices from very remote location at any time, as if he occupies such large scale computer exclusively and yet with reasonable cost."

At the early stage of TSS it seems like emphases was put on hardware resource-sharing but in realty the merit of TSS lies more in software resource-sharing.

TABLE - 3 Cost-wise quick comparison of usage environment for users

cost items / type or servise	Available core area	Initial system investment	Software development or procurement	Running cost incl. operational	Software maintenance	Dissemination or consulting	Remarks
Traditional in house batch	Medium	Medium (lease or rental)	Medium or High	Medium	High	High	non interactive. Inhouse consulting support is often unsatisfactory
Inhouse TSS	Medium, or Large	High	High	High	High	High	Interactive/remote batch. Inhouse consulting support is often unsatisfactory.
Commercial TSS	Large	Low (terminal installation)	Very Low	High (machine time, line, storage and Prog. charge)	almost none	Very Low	Interactive/remote batch. Consulting service charges are included in machine or Prog. charge.
Dedicated minis computer systems	Small	High (system purchase)	High	Very Low (self operation)	High	(High)	Interactive/non interavtive. Consulting is often unexpected.

Especially, as recently, by the ever ending increase of human cost, software related cost occupies over 70% of total computer usage cost, the merit of software resource-sharing has become more significant. Users have to consider somehow to share software development and maintenance cost or investment with other users and TSS offers such opportunities in a sense. You need not necessarily develop your own application but you can use library application software on usage cost basis. Also you need not worry about maintenance cost and if you are engaged with suitable TSS organization, you should be able to get technical support on request basis mostly with free of charge.

Computer usage cost is generally composed of following elements, i.e..

Hardware related cost;
Hardware investment. (Hardware purchase, lease, rental expense)
Operational cost. (System operation cost including personell & hardware maintenance)

Software related cost;
Software development or procurement.
Software maintenance including upgrade.
Dissemination or consulting support.

Table-3 is a very general comparison of several computer usage environment in this respect.

On the other hand, computer processing of our job as structural engineers is generally categorized into two types, i.e..

1. Large CPU time for processing:
   Relatively light user interruptions in the middle of processing.
   ---- Analysis type application
   ---- Automated design type application
2. Small CPU time for processing:
   Heavy requirement of user interruptions.
   ---- Input data preparation and validation type application
   ---- Output editing type application
   ---- CAD type application

Also for some kinds of application, size of available memory area dominates the throughout. Then, since many kinds of computer usage environment are available today and more in tomorrow, we structural engineers have to consider organizing our total computer usage environment with combination of every possible means.

DEDICATED MINI-COMPUTER SYSTEM

As for mini-computer hardware, minis are no more mini-computer and they are already highly competitive with general purpose medium computers in certain throughput. Then dedicated system for certain application has become quite common in various field. (Currently over 4,000 minis are shipped yearly in Japanese market) [3] [4] [5]

Scientific and engineering data processing has two different needs for the system, i.e.,

- Super high speed system with huge core memory.
  ---- ex. CDC "STAR", Illiac IV
- Application oriented system, higher cost/performance than general purpose system.

The latter, dedicated system, is expected to be more common in future.

In last several years, the U.S. mini-computer manufactures has announced high performance minis by newest electronic technology. At present time, still available memory area is relatively small and those minis are often weak in operating system and language processor, but they are already powerful enough to be a component for dedicated engineering computing system. For instance, we can build up relatively easily a satelite system using mini arround large on line system. And if we are able to materialize our requirements for dedicated system adequately and good hardware systems design support is available, we should be able to have more powerful application oriented system in our hands in near future.

For instance, instead of the development of various terminal system in commercial field, we can hardly find good engineering terminal system. For more extensive use of computer system in our field, the improvement of Input-Output system is now essential. May be I put too much emphasis on this concern because of our particular language problem, if not, untill we have low-cost terminal system, more familiar to engineer, possibly with graphic capability, our expectation may not be fulfilled.

Fundamentaly, what we practical engineers expect to computer system is to be a excellent assistant who is able to respond to our engineers questions properly and quickly. And also process what ever calculation we ask very quickly and precisely. But no more than that, we do structural design and we make decision. Computer Aided Design concept is quite common and for its achivement, Data Base Concept is very important of course but also good communication media with computer system is essential. Engineers should be able to give instruction to the system easily and the result of processing following the instruction should be presented in proper from and manner acceptable to engineers. It is my feeling that we structural engineers should pay more attentions to this area and should insist our requirements to computer industry.

SOFTWARE PORTABILITY

Among many problems in software development one significant problem is duplicated software investment. There are so many programs with similar function developed [6] and many of them are left unused on a shelf. This trends will be continued another years to come if we sit still. Tremendous effort to use existing modules or subprograms in creating new software systems has been payed, however, the result has not been fully satisfactory. There will be many reasons for this.
Namely; insufficient module informations, lack of reliable document, programmers particular nature and management problem.

However, one major reason lies simply in programming language. Programmers tend to design data structure by their knowledge of programming language such as FORTRAN, PL1 and because of this, portability of modules among systems can hardly be maintenanced.

Generally speaking, in application programs of our field, data file structures are quite dependent to each application systems. It is common that application programs are created from many modules or subprograms and those modules have to communicate with each other through data files which mostly reside in secondary storage and common communication area which is generally allocated on main memory. Using aforesaid programming language, programmers have to define these

communication, or in other words, programmers have to design data transfer among modules and build data transfer logic into each modules.

For such data transfer to be programmed precisely, each module must know exactly about structure of data files which has been left for it by earlier modules before it can use such data, and must generate data that will be needed by latter modules. Then once data structure for an application has been designed, modules which access to such data structure become dependent to it even module algorithm for the problem solving is identical.

This fact also makes maintenance work very costly. Simply because data structure is unique by application and then processing modules are dependent to it, it becomes just very costly to make significant changes in the system or add new features for handling unanticipated needs. To avoid those problems and improve software productivity the separation of data transfer and problem solving algorithm should be provided.

In this respect, I would suggest to promote more research and development tried in several systems such as GENESYS [7] and POLO [8], etc; which, to my understanding, facilitates data file management among modules and module handling feature, as a supporting subsystem around application modules and release programmers burden from data transfer description to an extent.

In near future, we will not be able to afford to build large unique user oriented software because of development cost and maintenance burden. We should pursue the possibility of developing application support utility system which makes engineers possible to joint application subsystems easily together from time to time as desired, flexibly enough to accomodate practical engieers dayly needs under TSS environment.

CONCLUSION

Computer Technology has been extensively developed then various types of computer usage are now available for structural engineers and further development of user oriented systems and services with the utilization of continually emerging electronic and electro-mechanical devices such as mini/micro computers, fast large volume disk storages, etc, are expected.

On the other hand, some problems in software development has been disclosed. As it is quite conceivable that software problem will dominate the future computer usage in our profession, we should consider the problem, namely the problem of software development, maintenance and program application promotion or consulting survices, more seriously from today.

As for the current computer usage, it is suggested not to stick to solo computer resources because it has become relatively easier to utilise various computer resources in combination along objectives. Then it is required to reevaluate available computer usage environment and reestablish the direction of our computer use taking advantage of various computer resources in the organization or outside for enjoying best possible, economically justifiable services.

REFERENCES

1. BROOKS Jr, F.P.: The Mythical man month, Addison-Wesley, 1975.

2. FENVES, S.J.: Senario for a Third Computer Revolution in Structural Engineering, Journal of the STRUCTURAL DIVISION, ASCE, JAN., 1971.

3. JIPDEC (Japan Information Processing Development Center): Computer White Paper, 1971 - 75.

4. QUANTUM SCIENCE CORP.: The Computer Services Industry, 1975.

5. JIPDEC: Survey on Oversea's Data Processing Industry, 1977.

6. MINI (Ministry of International Trade and Industry): Listing of Computer Programs, 1974.

7. ALLWOOD R.J., MAXWELL T.O'N.: 'GENESYS; A Machine Independent System, The Genesys Center, 1975.

8. LOPEZ L.A.: "POLO" Problem Oriented Language Organizer, Journal of Computers and Structure, Vol. 2, 1972.