DOCUMENT RESUME

ED 084 838 EM 011 659

AUTHOR	Frederick, Terry J.
TITLE	Test-Site Evaluation of ICU/PLANIT.
INSTITUTION	Purdue Univ., Lafayette, Ind. Dept. of Computer Science.
SPONS AGENCY	National Science Foundation, Washington, D.C.
PUB DATE	Aug 73
NOTE	11p.; Paper presented at the ACM Annual Conference (Atlanta, Georgia, August 1973)
EDRS PRICE	MF-$0.65 HC-$3.29
DESCRIPTORS	*Computer Assisted Instruction; *Computer Programs; Computer Science; Input Output; *Man Machine Systems; Program Descriptions; *Program Evaluation; *Programing Languages
IDENTIFIERS	Central Processor; CP; ICU; ICUPLANIT; Instructors Computer Utility; Peripheral Processor; *Programing Language of Interactive Teaching

ABSTRACT

Test-site evaluation of the Instructor's Computer Utility/Programing Language of Interactive Teaching (ICU/PLANIT) was conducted. Goals included: 1) analysis of the operation of ICU/PLANIT; 2) development of two PLANIT. Modifications were made in a distrubuted version, cost analyses were in man hours and quantities of machine resources consumed, and performance was measured in response time and machine resources consumed. Preliminary results included the findings that: 1) ICU/PLANIT is machine independent; 2) PLANIT can be quickly and inexpensively installed with medium or large scale hardware; 3) PLANIT does not show a negative effect on the throughput of jobs in the host environment; 4) computer operational costs of PLANIT are not prohibitive; 5) response time under the one-copy-per-user version is poor with only one interactive job in core and in this situation demands are low on the central processor (CP) but high on the peripheral processor; and 6) authoring demands more CP time but less input/output than student use, although the costs of each are almost equal. (PB)

TEST-SITE EVALUATION OF ICU/PLANIT

Terry J. Frederick [1]
Department of Computer Science
Purdue University

HISTORICAL PERSPECTIVE

PLANIT (Programming LANguage of Interactive Teaching) is a language used by authors to generate instructional sequences which are accessed by students via a computer. The Instructor's Computer Utility or ICU/PLANIT is the complete software system which makes PLANIT operational. This system is intended to function either as the sole operating system for the target machine or in co-operation with other operating systems.

In 1968, the National Science Foundation awarded the System Development Corporation a contract to redesign their own statistics-oriented version of PLANIT as a general-purpose language for computer-assisted instruction (CAI) to run on a variety of computers. By project termination in 1970, SDC had a partially debugged system including a new version of PLANIT and the ICU/PLANIT program which included a new method for adapting the program to different computers utilizing a Generator program.

Subsequently, several installations throughout the academic community and industry attempted to implement PLANIT with somewhat limited success. Then in 1972 Dr. Charles Frye, who headed the original design work at SDC, spent six months at the University of Freiburg in Germany debugging and cleaning up much of the system. In addition, he documented to NSF those portions of ICU/PLANIT needing further development and debugging.

In August 1972, the National Science Foundation selected Purdue University as a test-site for an analysis and evaluation of ICU/PLANIT.

[1] This work was supported by NSF Contract No. GJ-35633

Near the end of 1972, the Northwest Regional Educational Laboratory and Dr. Frye contracted with NSF for further PLANIT development and interaction between the test-site and PLANIT development was established.

TEST-SITE GOALS

Initially, the Purdue test-site was to take a copy of ICU/PLANIT as originally designed but much improved by the debugging efforts in Freiburg (version ∅) and implement it on the CDC 6500 with as few changes as possible. The project's goals included (1) the analysis and evaluation of the implementation and maintenance of ICU/PLANIT, (2) the development of course-ware and teaching of two courses by PLANIT (Computer Science 414 Numerical Analysis and Education 249 Case Studies), (3) the analysis of the consequent impact on ICU/PLANIT and the host computational environment, and (4) the production of a hierarchical package of test programs designed to demonstrate PLANIT features.

After the PLANIT development contract was awarded, both projects and the funding agency mutually agreed to release a distributable version of ICU/PLANIT in July, 1973, that would be the best version available within the time constraint. As a result, the stated goals were then reoriented toward the distributed version of ICU/PLANIT (version 1.0).

APPROACH

The test-site evaluation of ICU/PLANIT is viewed from two levels. The first of these represents systems programming aspects and is termed the internal level. The second emphasizes the user-apparent aspects and is called the external level. Although it is convenient to resolve the investigation into these levels, they are concurrent and have interdependent aspects. Some questions raised in light of the project goals are categorized by these two levels as follows.

Internal Level

1. What are the difficulties and costs involved in installing an operable ICU/PLANIT system?
2. What is the impact of this system on the interactive computing environment in which it will be placed?
3. What are the maintenance costs of the ICU/PLANIT system?
4. What modifications will be required to produce an ICU/PLANIT system with which External Level investigations may be carried out? Which of the modifications are relevant to the general ICU/PLANIT user?

External Level

5. To what extent does the ICU/PLANIT system meet its documented objectives as a CAI facility?
6. How effective is PLANIT for teaching diverse material to several simultaneous users.

To begin to answer such questions, several basic principles were adopted. First, any modifications to ICU/PLANIT would have to be reflected in a distributed version and not just patches that were local in nature. Secondly, cost analyses would be in terms of man hours and quantities of distinct machine resources consumed; the latter being resolved into three categories: (1) central processor time, (2) information transfer between central and peripheral storage (I/O activity), and (3) central memory. Finally, performance of the system during execution would be measured in terms of response time and machine resources consumed.

A specific direction materialized in an effort to answer these questions and meet the project goals. The local interactive environment is oriented toward a concept of one copy of an interactive processor program per user with the main operating system controlling time-slicing. The ICU operating system is designed for an environment where one copy of the processor program services all users and consequently it performs its own scheduling services. This type of processor, referred to as a multi-terminal processor, can operate in the local environment. However, some additional local software extension is required. As a result, it was decided to initially install ICU/PLANIT on a one-copy-per-user basis by bypassing the multi-terminal features through appropriate parameterization of version ∅. Once PLANIT was operating, a pilot study would be run whereby several students would take lessons from the two kinds of courseware being developed. Data would then be collected and analyzed from both the internal and external levels. Subsequently, ICU/PLANIT utilizing its multi-terminal features would be installed and the pilot study repeated. The resulting data would then be compared with that for the one-copy-per-user. The results should aid in fine tunning ICU/PLANIT and the host computational environment for production run teaching two courses.

SOME PRELIMINARY RESULTS

The installation of the ICU/PLANIT system on a target computer involves three major steps. These include (1) the writing of three subroutines in FORTRAN and/or local assembly language to provide machine dependent services, (2) installing the PLANIT SYSTEM GENERATOR program on the target machine and using it to create the FORTRAN statements from the META-FORTRAN code which constitute PLANIT, and (3) combining the locally written subroutines and the generated PLANIT FORTRAN into an executable PLANIT system. One of three subroutines under item (1) called MIOP which performs all I/O operations and operating system services is by far the most difficult single step in the installation process.

With less than twenty (20) lines changed out of about 13,000 lines in the original META-FORTRAN of version ϕ, PLANIT was installed at Purdue parameterized as a one-copy-per-user system. Two systems programmers worked part time using a powerful, high speed graphic display terminal (IMLAC PDS-I with 8K memory) and a sophicated remote job entry system (Purdue's PROCSY System). Excluding the time and cost for creation of test material for checking out PLANIT, the installation was accomplished in 163 man hours and a computer expense of $1,239. Local billing rates are based on $275 per hour of central processor time on the CDC 6500 and $.26 per 1,000 I/O units (approximately 100 peripheral processor seconds).

Major steps of the installation effort are described as follows.

1. Conversion of version ϕ system files to local files ($118, 4 hours)
2. Installation of the generator ($47, 5 hours)
3. Generation, compilation and construction of a loadable PLANIT system ($391, 50 hours)
4. Designing and coding of a minimal initial MIOP (8 hours)
5. Debugging of the initial system to obtain a LOG IN message ($225, 24 hours)
6. A second generation, compilation and construction of a loadable PLANIT system and preparation of author access material ($133, 16 hours)
7. Debugging of console I/O, PRESTORE/BUILD commands, lesson access, the random number generator ($295, 40 hours)
8. Development of a trace facility for recording internal flow of control in PLANIT ($15, 8 hours)
9. Development of documentation for author access to the current PLANIT system ($15, 8 hours)

It deserves mentioning that System Development Corporation demonstrated the working portion of ICU/PLANIT on IBM 360 equipment back in 1970 and the version ∅ that Purdue received was running in Freiburg on Siemens equipment. Thus, the relative ease with which PLANIT was installed on a CDC 6500 with virtually no change is a positive evidence for the machine independent claim.

PLANIT maintenance efforts for a period of three and a half months subsequent to the installation are described in the following remarks. During this period three system programmers were employed with two working half-time and one a quarter-time (total of 700 man hours and $4,200 computer expense). The work involved (1) performance improvement in the one-copy-per-user version, (2) debugging efforts to the distributed META-FORTRAN, and (3) development of a local MIOP for a multi-terminal version of PLANIT.

A variety of local modifications were made in order to improve the performance of the Generator program as well as the PLANIT system itself. Some of the modifications which realized substantial improvement are detailed below.

After installing the Generator program (Gp) in a straight forward manner, the following facts were observed through the use of program performance evaluation packages available locally. First, 85% of the central processor time consumed by Gp was attributed to the use of FORTRAN-formatted I/O statements. Secondly, 3% of the central processor time was consumed by one particular subroutine (LASTCH). By spending one-half man week of effort, each of these was recoded in local assembly language with the normal FORTRAN read, write and format statements being replaced by subroutine calls. The result of this was that the operations described above now consumed 29% and 0.5% of the central processor respectively. Another improvement was realized by compiling the Gp with a highly optimized FORTRAN compiler. An improvement in run-time efficiency occurred and several non-ANSI FORTRAN items in the distributed Generator were detected. The final version of the optimized Generator used one-sixth of central processor time consumed by the initial version and in the local environment shows a 10 to 1 decrease in real time operation.

The initial version of MIOP was constructed in FORTRAN. The MIOP used with the one-copy-per-user system was re-coded to obtain efficient code and local data storage organization, and to incorporate a number of local diagnostic facilities.

The PLANIT group at Michigan State University provided an overlay organization determined by them to be optimal for student use of PLANIT. This required a reordering of the PLANIT procedures and consequently a renumbering of the procedures. The improvement in PLANIT performance in terms of non-resident partition faults was substantial. Note that the new organization did not appreciably change the amount of code in the resident partition which in the original configuration was about 35K (octal) and 1/3 of the resident code used in version ø in Freiburg. Note also that the partition reorganization was the only modification to affect the original Freiburg system.

In order to achieve the project goal of teaching two courses using PLANIT, it is not only necessary to have an operable system but also the system must adapt to policies and capabilities inherent in interactive processor environment available through the Computing Center's remote terminal system. As a result, a pilot study was designed whereby students would take lessons under varying conditions from the two kinds of courseware being developed. The study would employ both the one-copy-per-user and multi-terminal systems competing for the CDC 6500 with the regular work load. The resulting data should be extremely useful in preparing for the production run. As of this writing, the pilot study using the one-copy-per-user PLANIT system has been completed.

The one-copy-per-user pilot study design was based on one week of lesson presentation using a maximum of eight students on PLANIT at any one time. At the time of the pilot study, only sixty-four ports were available on the remote entry system used by PLANIT and it was decided that eight would provide the information needed and still permit a reasonably normal load by other users. Hardware and software work is underway that will provide 128 available ports in the very rear future with more later. Also, the design utilized both a stratification and a mix of courseware, students and authors. Moreover, the study was conducted from 9:30 to 12:00 noon daily during a week in the middle of the semester in order to involve a time representative of the regular computing load.

The computing facility at Purdue is oriented toward batch computing and normally only one interactive job is in central memory at one time. Moreover, such jobs have a 15K (octal) limit and larger jobs such as PLANIT at about 35K (octal) must be specially approved and compete with the small ones for core. In order to determine the difference, the number of interactive jobs permitted in memory was varied between one and two during the pilot study. Since there is about 230K (octal) storage available, PLANIT consumed about 1/5 or 2/5 of central memory, respectively, when one or two jobs were in core. The average number of jobs run daily during the week of the pilot study was 6,185 over 17.4 hours of production time. On the average, 434 jobs were completed per hour during the time period the pilot study was running. While it is not known how many interactive jobs were running during this period an average of 43 terminals were active at log on time for the PLANIT terminals.

Statistics collected the week before and after the pilot study show that PLANIT did not alter the computing throughput. For the week prior to the study, an average of 5,874 jobs were run per day over 17.4 hours of production time. Also, on the average, 421 jobs were completed per hour during the corresponding test time period. Similarly for the week following, 5,860 jobs were run per day over 17.4 hours of production time while 445 jobs were completed per hour during the 9:30 to 12:00 noon period.

Several internal level measures were taken. The pilot study consumed a total of .0574 hours of central processor time ($15.79) and 1.1838 hours of perpheral processor time ($110.81) for a total computer expense of $126.60. There were fifty-six users, half with the numerical analysis (NA) material and half with education case studies. The average terminal time for the users was 49.8 minutes and for education 46.1. Purdue classes operate on a fifty minute time period so using that as a norm, the average NA student needed 3.87 seconds of CP time ($.30) and 6,057 I/O units ($1.57) for a total computer expense of $1.87. There were two kinds of education case study courseware; one was text oriented and the other was dialogue oriented; hereafter termed Ed and EdD, respectively. Surprisingly, the Ed students used more CP time than the NA students, 4.16 seconds at a cost of $.32, but considerably less I/O, 2,034 units at $.53 for a total of $.85. However, the EdD user needed more CP time, 4.42 seconds at $.34 and considerably more I/O units, 5,490 at $1.43 for a total computer expense of $1.77. Some authoring on EdD material was done during the study using an average of 5.74 CP seconds ($.44) and 4,949 I/O units ($1.29) for a computer cost ($1.73) nearly identical to the average EdD student user. The final internal measure involved a trace routine to check on the percent of faults in overlay calls for the two types of courseware. A sample from the numerical analysis material showed 1,600 total transfers and 85 total faults for a percentage of 5.31. Similarly for the education material, it was 133 total faults or 4.02% on 3,305 total transfers.

The only external level measure used during the pilot study besides a subjective report from the users was response time. Response time was defined in the normal way--the elapsed time from the instant the user hit carriage return until the instant the print head was activated by the computer. As expected, given the computing load and the way PLANIT had to compete for central memory, the response times were poor. There is some evidence that the response time for the numerical analysis students was considerably worse than that for the education students. During one period with only one interactive job in core at once and four simultaneous PLANIT users (2 NA, 2 Ed) the response time was 17.6 seconds for the NA users and 11.6 seconds for the Ed students. During another period under the same conditions except that all four users were taking NA material, the response time degraded to 19.2 seconds where as when all four were accessing Ed material, it dropped to 10.3 seconds. The effect of another interactive job being allowed in central memory was significant. With eight simultaneous PLANIT users (four in NA, four in Ed) the average response time was 20.1 seconds for the students taking NA and 12.7 for those taking Ed. Under the same conditions except with two interactive jobs in core at the same time, the response time dropped to 6.7 seconds and 6.4 seconds, respectively. Finally, the response time for the EdD users was about the same as that for NA with little difference whether the user was a student or an author.

SOME COMMENTS

Some preliminary results obtained at the Purdue test site for ICU/PLANIT can be summarized by the following comments.

1. The original claims of machine independence for ICU/PLANIT received affirmative support.
2. Using medium or large scale hardware, it is reasonable to expect to be able to install a distributed version of PLANIT in a relatively short period of time without great expense.

3. PLANIT does not show a negative effect on the throughput of jobs in the host computational environment.

4. Computer costs to run PLANIT are not prohibitive at Purdue based on the current billing process and prices.

5. In general, response time under the one-copy-per-user version is not good especially with only one interactive job in core at a time. Given the same number of users, it gets significantly better when the job limit is increased to two. Significant is used in the sense of a factor of two or better with eight simultaneous users.

6. One-copy-per-user PLANIT using both the numeric and text oriented courseware makes fairly low demands on the central processor but considerably greater demands on the peripheral processor.

7. Authoring demands more central processor time but less I/O than student users of the same type of material yet ends up costing about the same.

It is expected that these preliminary results and statistics will be even more interesting when they are compared with those from the multi-terminal version of the pilot study.

The author is indebted to the Purdue PLANIT project staff, particularly Dr. R. Roman and to Mr. J. Steele of the Computing Center for their assistance in the preparation of this paper.