

CATER: AN ONLINE PROBLEM TRACKING FACILITY FOR SLC*

Robert C. Sass, Hamid Shoaee

Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309 USA

Abstract

An online facility has been developed for SLC to organize and simplify the management of all problems encountered in the operation of the accelerator. CATER (Computer Aided Trouble Entry and Reporting) may be used to make the initial entry of a problem, to enter one or more solutions to a problem, to modify or closeout a problem, to generate a variety of pre-defined reports giving status and statistical summaries, and to allow anyone to browse the database. All phases of CATER can take place on the operator console, workstations, or on any ANSI compatible terminal. The user interface is designed around a menu driven windowed environment with a large amount of context sensitive help information to alleviate the need for consulting user documentation. Currently, the CATER database contains information on more than 30,000 problems entered since it went online in January of 1988. The features of the software and some implementation details will be presented.

INTRODUCTION

In the early days of SLC operation, hardware and software problems were reported and tracked by the "yellow-sticky" and other paper-based methods. It was apparent early on that some more reliable method of tracking the many machine problems was needed. The initial attempt to implement a problem tracking software system failed mostly because of a lack of user acceptance. For the second attempt, it became clear that like a good business, the system had to cater to the needs of the users first if it was to be accepted. CATER was thus designed with the following general requirements:

1. Above all it had to be easy to use with a minimum of instruction and keystrokes.
2. It had to run on any of the terminals then in use at SLAC.
3. It had to keep all problems in a database for historical analysis.
4. It had to be fairly easy to modify so it could adapt to changes in the physical accelerator and management structure.

GENERAL FEATURES

At any point, a given problem is either Unsolved (has no solution), Solved (has one or more solutions) or Closed (solved and a supervisor agrees that it's fixed). Thus from the main menu there are separate CATER functions to report, solve and close a given problem. Also from the main menu are additional functions to modify existing unclosed problems or solutions, generate canned reports and browse the database.

For consistency, the operation of all screens in CATER is as similar as possible. Figure 1 shows the hardware problem report screen as it appears when reporting a new problem which we'll use to show the operations common to all screens.

* Work supported by Department of Energy contract DE-AC03-76SF00515.

Figure 1. Hardware problem report screen

1. All allowable control options are highlighted at the bottom of the screen. Control key sequences are used for most control options because there is little overlap in function key mappings between the various terminals and emulators on which CATER can be run.
2. Online help is available for every field. This help information includes any validation which is performed. If the field is limited to a set of specific entries, the help lists them and the minimum keystrokes required for each entry.
3. Within a field the user can edit the text using the arrow keys and switching the entry mode to insert or overstrike.
4. The user can move back and forth to the next or previous field as many times as desired until all data is entered satisfactorily. Until the user enters CTRL-Z to execute the function, the cancel option (CTRL-C) is always available which returns to the previous menu without making any database changes.

PROBLEM REPORTING

Again refer to Figure 1, the hardware problem report screen. The software screen is similar but with different fields after "Urgency". For the problem and solution entry forms there are some additional items of note:

1. Required fields are enclosed in brackets. CATER insists that you make a valid entry in these fields.
2. Initial default values are entered. For the hardware problem report this includes the users id, name and problem priority.
3. Both the problem report and solution forms have a 10 line free form description field used to describe the problem or solution. As in single line fields, the arrow keys and entry mode can be used to do simple editing. For this multi-line field, carriage-return goes to the next line as you would expect. The TAB or CTRL-B must be used to go to the next or previous field respectively.

An interesting historical anecdote reveals how important it is for any widely used system to adapt to the user's needs. The initial system had two lists of supervisors; one for hardware problems and one for software. After a problem was entered, it was mailed to the appropriate list. It turned out that the hardware people were usually in the field and rarely read electronic mail and so their problems were just stacking up! While the specific printer details have evolved over the years, we automatically print new hardware problems, email software ones and everybody's happy.

Except for the distribution list, the email of software problems has remained unchanged since CATER's initial release. Hardware problem distribution on the other hand, has been modified several times, reflecting organizational and personnel changes. At the present time a hardware problem is assigned to a default shop based on several problem criteria. The reporter can change the default if desired and when the problem is entered it is immediately printed on that shop's printer. This has served to expedite the solution of hardware problems since most problems go directly to those responsible for fixing them without the necessity of logging in to the computer system.

SOLVING A PROBLEM

Once a problem has been completely or partially fixed, the solver enters a solution into the database. Any number of solutions can be entered for a given problem. Figure 2 shows the solution entry of a previously solved hardware problem. As with the problem entry, there are a set of required fields and some default values are supplied.

```

CATER - Hardware Problem Report Information

CATER - Hardware Problem Solution Information

CATER - Solution to Hardware Problem
                                09:13PM

Your Userid   : [PHIPAT ]      Date/Time   6-MAY-1993 09:13
Your Name    : [Patrick Conroy ]
Hours to Solve : [ ]          Problem Number 31659
Solution was  : [ ]          Solution Number
(A) Adjust/repair, (R) Replace, (E) Enhance/Modify, (N) No trouble found
Exchange Unit-FRU :
Old serial number :
New serial number :
Documentation    : [Unknown ]
Schematic number :
Ready-to-Close Y/N: [Yes]
Solution description. <CR> next line and <TAB> to validate description.

[PF1] Buries top window  [PF2] Toggles thru solutions

[PF1] Cancel menu      Entry Mode : Overstrike  [TAB] Next field
[PF6] Get help        [PF4] Insert/Overstrike [PF5] Previous field
[PF4] rewrite screen  [ARROW] Move in field  [PF2] Record solution
  
```

Figure 2. Hardware problem solution entry screen

In this case there have been previous attempts to solve the problem and while entering yet another solution you have immediate access to the initial problem entry and all solutions to date. By using the PF1 and PF2 keys as indicated, you can expose the buried problem and solution windows and scroll through all previous solutions. If this is the first solution entry to a previously unsolved problem, the problem status automatically changes from Unsolved to Solved.

MODIFYING PROBLEMS AND SOLUTIONS

It is frequently useful to modify the fields of an existing problem or solution. Additionally, not all fields in a problem description (such as who is assigned to fix it) are available to the initial reporter. The modify function allows supervisors to change any field in an existing, Un-closed problem. Typical reasons for modifications include:

- Add to the problem description.
- Change the person or shop to which the problem is assigned.
- Change the problem's urgency

When the modified problem is entered into the database, the modifier has the option of re-distributing the modified problem in the same way as if it were initially entered. This again allows the immediate notification of maintenance personnel of any change in a problem's status.

CLOSING PROBLEMS

When a problem has been solved to everyone's satisfaction, a supervisor is responsible for officially closing it. Figure 3 shows the closeout screen.

```

CATER - Hardware Problem Report Information

CATER - Hardware Problem Solution Information

Solved by      PATR          PAT REARDON
Hours to solve 1.00         Date solved    4-MAY-1993 23:00
Solution type  Adjust/Repair Problem number  31654
Exchange module MDHE       Solution number  1
Old serial number
New serial number
Documentation   Unknown
Schematic Number

Solution description.
-----
20-B FOUND KLYS FIL AT MAX CURRENT, MIN VOLTAGE. TESTED MODULATOR WITH
KLYS LOAD TEST UNIT, FOUND KLYS FAULTY. INFORMED AMRF.

CATER - Closeout a Problem

Your Userid   : [JHM ]      Closeout Date  6-MAY-1993 09:00
Your Name    : [Justy Murphy ]

[PF1] Buries top window  [PF2] Toggles thru solutions

[PF1] Cancel menu      Entry Mode : Overstrike  [TAB] Next field
[PF6] Get help        [PF4] Insert/Overstrike [PF5] Previous field
[PF4] rewrite screen  [ARROW] Move in field  [PF2] Close problem
  
```

Figure 3. Problem closeout screen

As before, the report and all solutions are available for immediate review by burying windows and scrolling through multiple solutions. The user's id and name are filled in and validated against a list of authorized closers before the status of the problem is officially changed to Closed in the database.

REPORTS & DATABASE BROWSING

Figure 4 shows the first level report screen.

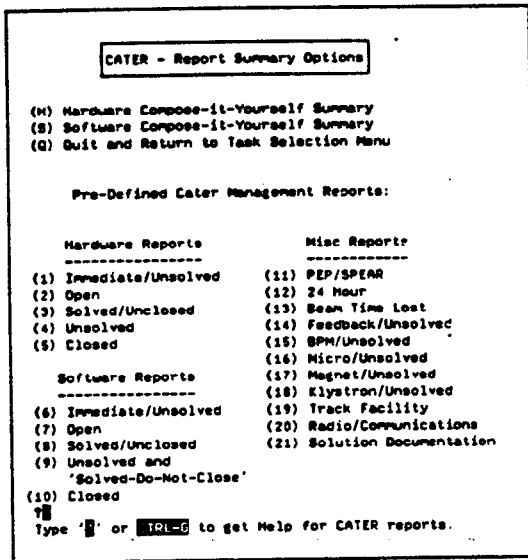


Figure 4. First level report screen

If you enter one of the numbers, CATER generates the appropriate pre-defined report on a printer of your choice. This is submitted as a batch job and returns immediately so you can do other CATER work while it's printing. If you enter 'H' or 'S', you get a "compose-it-yourself" screen for browsing the database. Figure 5 shows the hardware browsing screen. There are several things to note about this screen:

1. You can direct the output to a printer, have it displayed on your screen or written to a file for disposition as you choose.
2. The output format can be an abbreviated one liner for each selected problem, a full display of the problem and all solutions or the data can be written to a file in an 'export' format, suitable for incorporation into a PC spreadsheet or database.
3. You can enter selection criteria for any of the problem or solution fields augmented with the operators listed at the bottom of the screen. This allows almost unlimited read access to the database in a simple manner.

IMPLEMENTATION

CATER was implemented in late 1987 before workstations and GUI interfaces were available or popular at SLAC. Indeed, many of the CATER users still use VTXXX compatible terminals or emulators which are located throughout the accelerator to access the system. The basic tools used to construct CATER were:

1. SMG, a set of screen management routines.

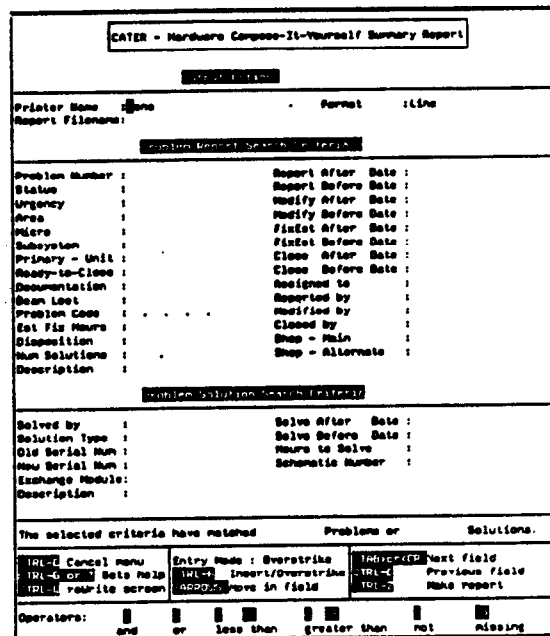


Figure 5. Hardware browsing screen

2. Rdb, DEC's relational database and associated tools, precompilers etc..
3. VAX "C" programming language.

The SMG routines are fairly low level so the CATER program has a set of data structures which define the screen layouts, fields and their validation. This means that the screens are decoupled from the database so when new database fields are added, they must also be manually added to the appropriate screens. This is the most tedious and error prone aspect of CATER software maintenance. We considered migrating to Oracle a few years ago to make maintenance easier. We ultimately decided against this approach since it would have substantially changed CATER's "look-and-feel" and some of the fine control we exercise over the screen was difficult to reproduce under Oracle.

In general, Rdb has been satisfactory for our purposes. It offers a precompiler for executing fixed queries like we use for problem & solution entry or modification. For the browsing screens, we compose a query "on-the-fly" from the fields and operators entered and pass that to Rdb for interpretation. This is somewhat slower but gives us complete freedom to formulate queries at run time.

Retrieval performance for the browsing screens has been a bit of a problem as the database has grown. The indexes are defined around the set of fields most often referred to when scanning open (Unsolved or Solved) problems and with a little care, retrieval performance is generally satisfactory averaging 1-10 seconds depending on the complexity of the query. At the present time, of the more than 30000 problems in the database, only about 1% are open so historical queries which look at a large number of closed problems can take a minute or more. For this reason, those people compiling historical statistics usually export a large selection of records into a PC tool and do the analysis there.

To date our reliability has been excellent; we have not lost a single problem or solution in CATER's operating history.