

A SIMPLE TIME-TRACKING SYSTEM USING SAS SOFTWARE

Richard A. Henkle, Consolidated Diesel Company

BACKGROUND

Consolidated Diesel Company (CDC), based in Whitakers, North Carolina, is a partnership owned in equal shares by Cummins Engine Company, a manufacturer of diesel engines based in Columbus, Indiana and J.I. Case Company, a manufacturer of construction and agricultural equipment based in Racine, Wisconsin. CDC began building diesel engines for its partners in the spring of 1983.

Like the plant it serves, CDC's data processing department was built from the ground up. Not surprisingly for a start-up operation, the development staff, which eventually rose to 12 people, quickly found itself faced with over 60 open projects at any given point. At the same time, there was no computer-based software available to aid the staff in tracking time spent either on these development projects or on the various administrative activities which also demanded their attention.

PROBLEM

There were several constraints which had to be overcome in the development of a workable system:

1. With critically important projects such as MRP, Order Entry, and Accounts Payable to be implemented, the priority of a time-tracking system was not high. It clearly had to be developed with a minimal investment of time and effort, and the design therefore had to be a simple one. As time passed, the reporting sub-system would develop a number of add-on features, but the data-entry portion of the system was to remain streamlined and functional.
2. For the system to be of any value, it could not demand large amounts of the users' time just to enter data. Quick access to the system from a terminal was essential. It was also necessary that the user be required to enter as little data as possible for each transaction.
3. At the time that the system was developed, there was little knowledge of SAS* software within

the development staff. The system had to be usable even by those who were unfamiliar with the SAS system.

4. As the data entry screen calls only for a short mnemonic in identifying a project, the reporting sub-system would require a PROC FORMAT type of table to expand the identifier into a meaningful project name and code. With each member of the staff confronted by an ever-changing set of projects and activities which to one degree or another is different from that of anyone else in the department, it was unrealistic to try to maintain a single project table for everyone's use. Rather, each user needed the ability to maintain his own, and this maintenance had to be kept as simple as possible, particularly given that most of these users were not familiar with the SAS system.
5. Different reporting requirements would demand that the user be able to produce data either in discrete time units or summarized for any given time span. It would also be necessary to produce the data in the form of totals by project or groupings by day, week, or month.
6. As the system was intended to be used and maintained on an individual basis with no overall database administrator, it had to be as close to maintenance-free as possible. In addition, with an average of four activities (and therefore four transactions) a day for each user, it was important to keep the database as compact as possible to minimize mass-storage space requirements.

ENVIRONMENT

The hardware and software available at CDC consist of an IBM 4381 mainframe running under the MVS operating system. TSO/SPF is the interactive medium. Base SAS and SAS/FSP* software are available and each user has access to an IBM 3278 or similar terminal.

DATA DEFINITION

In order to track time spent on a project or other activity, there are three items of information which are necessary, and these became the variables for the SAS dataset forming the heart of the system.

1. DATE. The date on which the work was done is stored in SAS date format (the number of days since January 1, 1960). As four bytes is more than enough to hold any date value for the foreseeable future, the date became a four-byte numeric variable.
2. PROJECT. In order to record the activity that was worked on as efficiently as possible, a five-byte mnemonic identifier was chosen. It would permit the efficient use of space necessary in such a system while still providing the user with an alphanumeric code which he himself could select and might therefore more easily remember. In a moment we will see that the data entry portion of the system provided a means to help him remember.
3. HOURS. The time spent on a given date working on one project had to be recorded in tenths of hours. This time would be stored as a three-byte variable.

The job used to create the SAS dataset and the output of a PROC CONTENTS on the dataset are shown in Figures 1 and 2. Note the number of observations which can be stored on one track. Few users would fill a track in the course of an entire year.

DATA ENTRY

Once the dataset was created, a means of entering data interactively was needed. In the environment described above, a TSO CLIST represented the logical mechanism for allocating the SAS library, invoking base SAS software interactively, and executing PROC FSEDIT. This CLIST can be written to perform an automatic backup of the dataset, or an individual user might choose to leave out the backup if, for example, a nightly system backup is available. However, care must be taken that the system utility used in the

data center is capable of restoring SAS data libraries.

Once the CLIST is executed, the user is taken directly to the default FSEDIT screen. By pressing PF3 and entering Option 6 of FSEDIT, he can create a screen similar to the one in Figure 3. The mnemonics to be used for the PROJECT variable can be listed on the screen for reference whenever data is to be entered. The same procedure can later be used to modify the list of mnemonics as projects are added or removed. The screen specified in the PROC FSEDIT statement is permanently saved in the SAS library and automatically updated whenever Screen Modification mode is used.

When entering data, the FORMAT and INFORMAT statements used in the definition phase permit a good deal of flexibility in entering the DATE and HOURS variables and will reformat each. Other features available in PROC FSEDIT which can be helpful are PF6 and the CURSOR command. The first is used to avoid the need to re-enter the date when several transactions for a single day are entered at once. The second allows the user to force automatic positioning of the cursor at the PROJECT variable.

REPORTING

The reporting sub-system had to address three separate issues:

1. PROJECT MNEMONICS. Each mnemonic represents a development project or administrative activity. For either one, there is a description which should appear on any report generated by the system. In addition, each development project is tracked within the department by a project code in the form YY-NNN (e.g., 84-001 was the first new project defined in 1984), which should also appear on the report.

The obvious solution was PROC FORMAT. Defining one format for titles and one for codes and using the PUT function to generate each from the mnemonic permitted the reporting software to generate meaningful identifiers. At first, the system used permanently-stored formats, requiring a format

library and the need to submit a job to rebuild the formats each time a new project was added. But with new projects added on an ongoing basis, it became apparent that the time saved in not executing the FORMAT procedure for each report was more than offset by the complications involved in maintaining a separate library. It was therefore abandoned in favor of a partitioned dataset member containing an entire FORMAT procedure with two VALUE statements, one for titles and one for codes. A %INCLUDE makes them available for each of the reports. Figure 4 is an example of a FORMAT procedure used by this system.

2. REPORT SOFTWARE. PROC PRINT was used to generate all of the reports for this system, but as a simple printout of the database would be of little value for most reporting functions, six short SAS programs were written to sort and summarize the data into a variety of useful reports. Sample printouts are as follows (the name in column 1 is preceded by 'TIME'):

PROG NAME	PROGRAM TITLE	SAMPLE OUTPUT
R1	Individual Time Records Grouped by Project.	Fig 5
R1S	Time Totals by Project.	Fig 6
R2S	Monthly Data Summarized by Project.	Fig 7
R3S	Weekly Data Summarized by Project.	Fig 8
R4	Individual Time Records Grouped by Date.	Fig 9
R4S	Time Totals by Date.	Fig 10

3. INTERACTIVE REPORT GENERATION. With six different reports which could be generated, it would make sense for the system to provide a menu-driven means of selection. Additionally, among the criteria for the system that were stated at the outset was the requirement that it be usable by someone who possessed no knowledge of the SAS system. Without help from within the Time-tracking system, such a user could not easily limit the time period to be reported or report on only a single project.

To alleviate these potential problems, a second CLIST was written to invoke PROC FSEDIT on a SAS dataset whose variables would provide the user with a means to specify one of the above reports and to limit the time period to be covered (an excellent paper on this use of PROC FSEDIT was presented at SUGI '84. Entitled "Use of PROC FSEDIT to generate SAS code", it was written by C. D. Hardison and R. D. Muller of Eli Lilly and Company). This CLIST takes the user directly into a default FSEDIT screen, which can be modified to look something like the one shown in Figure 11. When this screen is filled in and PF2 is pressed, the CLIST will automatically pass the requested parameters to the batch program and submit it.

SYSTEM MAINTENANCE

To set up this system, a SAS library must be available to hold the database, a backup database (optional), and the screens for the data entry and reporting functions. On an IBM 3350 disk pack, 10 tracks should be more than adequate.

A partitioned dataset (PDS) is necessary to hold the formats, JCL, and report programs used by the system. Five tracks on the same device type are recommended.

The two CLISTS should be stored in a CLIST library.

Other than modification of the mnemonics in both the PROC FORMAT member of the PDS and in the data entry screen, the only ongoing maintenance in the system would be an occasional COMPRESS of the PDS. Also, if the space made available by deleted observations is not reused, a special run will have to be set up to remove the emptied observations.

ADDITIONAL COMMENTS

When the SAS system stores a decimal fraction in anything less than a full 8 bytes, the decimal places will not be represented exactly (see pp 92-93 of SAS USER'S GUIDE: BASICS - 1982 EDITION). This should normally cause no problem in

the system, but if maximum accuracy is necessary, the user may allow the default length for this variable.

The only report program with any complexity is Report 3S, WEEKLY DATA SUMMARIZED BY PROJECT. It goes to some trouble to avoid becoming misaligned when the first observation encountered does not represent a Monday activity or when there is no Monday observation for a week, but when both of these factors are present in the first two weeks processed, the 2nd 'week of' designation will become the 7th day after the first.

While the system is intended to be used in a way that it can be tailored to a user's individual workload and reporting needs, the reporting capability was modified by one project team at CDC to permit each member's data to be merged with that of the others, producing a report of the entire team's activities during the reporting period. One team member was designated to maintain one data entry screen and a single set of format tables so that each team member would use the same mnemonics.

This system performs the basic task of tracking and reporting time. It is not a true project management system in the sense that it does not allow for

tracking of time against milestones or any other form of planned activity measurement. Additionally, it does not permit time to be recorded for sub-project activity, though the addition of another variable would be simple enough. For the purposes of the CDC users, however, this was not considered necessary.

Source code for this system is available on request. Questions or comments may be addressed to the author at:

Consolidated Diesel Company
P.O. Box 670
Whitakers, North Carolina 27891
(919) 437-9221

REFERENCES

SAS User's Guide: Basics, 1982 Edition. SAS Institute Inc., Cary, NC
Hardison, C. D., and Muller, R. D., Use of PROC FSEDIT to Generate SAS code. Proceedings of the Ninth Annual SAS Users Group International. March 18-21, 1984. Hollywood, Florida. SAS Institute Inc., Cary, NC

* SAS and SAS/FSP are registered trademarks of SAS Institute Inc., Cary, North Carolina, USA

```
//JOBNAME JOB (XXX,YYY), 'CREATE TIME DS', MSGCLASS=T,
//      MSGLEVEL=(1,1), CLASS=A
/**
/** NEW TIME-TRACKING DATABASE
/**
//STEP10 EXEC SAS
//SASDS DD DSN=MY.TIME.LIBRARY, DISP=(NEW,CATLG), UNIT=SYSDA,
//      VOL=SER=PROD99, SPACE=(TRK,(5,2))
//SYSIN DD *
DATA SASDS.TIME;
  LENGTH DATE 4 PROJECT # 5 HOURS 3;
  INFORMAT DATE MMDDYY6.;
  FORMAT DATE MMDDYY8.;
  CARDS; /* INSERT A DUMMY RECORD ON THE NEXT LINE */
010385 SAS 1.0
PROC CONTENTS;
/*
```

FIGURE 1

```
CONTENTS OF SAS DATA SET SASDS.TIME
TRACKS USED=1 SUBEXTENTS=1 OBSERVATIONS=1 CREATED BY OS JOB JOBNAME
ON CPUID 00-4381-001582 AT 10:02 MONDAY, FEBRUARY 4, 1985 BY SAS
RELEASE 82.4 DSN=MY.TIME.LIBRARY BLKSIZE=19069 LRECL=16
OBSERVATIONS PER TRACK=1167 GENERATED BY DATA
```

ALPHABETIC LIST OF VARIABLES						
#	VARIABLE	TYPE	LENGTH	POSITION	FORMAT	INFORMAT LABEL
2	DATE	NUM	4	9	MMDDYY8.	MMDDYY
3	HOURS	NUM	3	13	3.1	
1	PROJECT	CHAR	5	4		

FIGURE 2

```

Edit SAS data set: IN.TIME          | SCREEN 1
Command ===)                       |-----
                                      | Obs   1

```

```

LABOR: WEEKLY LABOR CHANGES      ADMIN: ADMINISTRATIVE ACTIVITIES
INVEN: PHYSICAL INVENTORY         MEET: SYSTEM & TEAM MEETINGS
MCT: MATERIAL COST TRACKING       PROCU: PROCURE - HDWARE, SFWARE
APPL: APPLICANT FLOW SYSTEM       SICK: SICK LEAVE
SAS: SAS USER SUPPORT            VAC: VACATION
GEN: ORDER ENTRY COMPLETION       XFUN: CROSS-FUNCTIONAL ACTIV.
CONSL: CONSULT ON PROSPECTIVE PROJ HOL: COMPANY HOLIDAY

```

DATE: 01/03/85 PROJECT: SAS HOURS: 1.0

FIGURE 3

```

PROC FORMAT;
VALUE $NAME SAS='SAS USER SUPPORT
MCT='MATERIAL COST TRACKING
INVEN='PHYSICAL INVENTORY
VAC='VACATION
APPL='APPLICANT FLOW SYSTEM
* ;
VALUE $CODE SAS='84-536'
MCT='82-005'
APPL='83-241';

```

FIGURE 4

INDIVIDUAL TIME RECORDS GROUPED BY PROJECT - (TIMER1)

ALL RECORDS FROM 05/07/84 THROUGH 05/09/84
(ALL PROJECTS PROCESSED)

NAME	CODE	DATE	HOURS
ADMINISTRATIVE	ADMIN	05/07/84	0.5
	ADMIN	05/09/84	3.0
-----			3.5
MRP RUN-TIME PROBLEM	83-749	05/08/84	12.0
MAINTENANCE SYSTEM - TERA	82-002	05/07/84	3.0
	82-002	05/08/84	1.0
-----			4.0
MAINTENANCE SYSTEM - TERA			4.0
=====			19.5

FIGURE 5

TIME TOTALS BY PROJECT - (TIMERIS)

ALL RECORDS FROM 05/07/84 THROUGH 05/27/84
(ALL PROJECTS PROCESSED)

NAME	CODE	HOURS
ORDER ENTRY	82-036	24.0
ADMINISTRATIVE	ADMIN	21.0
SAS USER SUPPORT	83-536	11.0
MAINTENANCE TO INSTALLED SYSTEMS	MAINT	9.5
=====		65.5

FIGURE 6

MONTHLY DATA SUMMARIZED BY PROJECT - (TIMER2S)

ALL RECORDS FROM 05/01/84 THROUGH 06/30/84
(ALL PROJECTS PROCESSED)

MONTH	NAME	CODE	HOURS
MAY	ORDER ENTRY	82-036	47.0
	ADMINISTRATIVE	ADMIN	25.0
	SAS USER SUPPORT	85-536	26.5
	HOLIDAY	HOL	8.0
-----			106.5
JUNE	VACATION	VAC	40.0
	PHYSICAL INVENTORY	84-478	30.0
	SAS USER SUPPORT	85-536	20.0
	MAINTENANCE TO INSTALED SYSTEMS	MAINT	20.0
-----			110.0
=====			216.5

FIGURE 7

WEEKLY DATA SUMMARIZED BY PROJECT - (TIMER3S)

ALL RECORDS FROM 05/07/84 THROUGH 05/20/84
(ALL PROJECTS PROCESSED)

WEEK	NAME	CODE	HOURS
05/07/84	ORDER ENTRY	82-036	20.5
	ADMINISTRATIVE	ADMIN	6.5
	MAINTENANCE SYSTEM - TERA	82-002	5.0
	SAS USER SUPPORT	83-536	12.0
-----			44.0
05/14/84	MAINTENANCE SYSTEM - TERA	82-036	26.0
	ADMINISTRATIVE	ADMIN	4.0
	SAS USER SUPPORT	83-536	16.0
-----			46.0
=====			90.0

FIGURE 8

INDIVIDUAL TIME RECORDS GROUPED BY DATE - (TIMER4)

ALL RECORDS FROM 05/07/84 THROUGH 05/08/84
(ALL PROJECTS PROCESSED)

DATE	DAY	NAME	CODE	HOURS
05/07/84	MON	MAINTENANCE SYSTEM - TERA	B2-002	5.0
		SAS USER SUPPORT	B3-536	2.0
		ADMINISTRATIVE	ADMIN	1.0
-----				8.0
05/08/84	TUE	ORDER ENTRY	B2-036	4.0
		MAINTENANCE TO INSTALLED SYSTEMS	MAINT	3.0
		ADMINISTRATIVE	ADMIN	2.0
-----				9.0
=====				18.0

FIGURE 9

TIME TOTALS BY DATE - (TIMER4S)

ALL RECORDS FROM 05/07/84 THROUGH 05/08/84
(ALL PROJECTS PROCESSED)

DATE	DAY	HOURS
05/07/84	MON	8.0
05/08/84	TUE	9.0

FIGURE 10

```

Edit SAS data set: WORK.PARMS          | SCREEN 1
Command ==>                            |-----|
                                          | Obs   1
    
```

<<<<<<<<<< TIME-TRACKING SYSTEM >>>>>>>>>

REPORT TO BE RUN: 1 - INDIVIDUAL TIME RECORDS GROUPED BY PROJECT
 1S - TIME TOTALS BY PROJECT
 2S - MONTHLY DATA SUMMARIZED BY PROJECT
 3S - WEEKLY DATA SUMMARIZED BY PROJECT
 4 - INDIVIDUAL TIME RECORDS GROUPED BY DATE
 4S - TIME TOTALS BY DATE *** REPORT NUMBER: __

DATES TO BE PROCESSED IN THIS REPORT.
 - ENTER BOTH DATES : ALL DATA WITHIN THE SPECIFIED RANGE.
 - LEAVE OFF ENDING DATE : ALL DATA STARTING WITH BEGINNING DATE.
 - LEAVE OFF BEGINNING DATE: ALL DATA THROUGH ENDING DATE.
 - LEAVE OFF BOTH DATES : ALL DATA IN THE DATABASE.

DATES (FORM MMDDYY): BEGINNING _____ ENDING _____

ENTER A SINGLE PROJECT OR LEAVE BLANK FOR ALL PROJECTS: _____

((ENTER DATA ABOVE AND PRESS 'PF2' TO SUBMIT REPORT))

FIGURE 11