SPS/ACC/PB/Technical Note/77-8

28 April 1977

GENERATING SYSTEM SOFTWARE

P. Brummer

1. Introduction

An organization has been set up to generate software for the SPS control computers. The reliability of a newly created system should be higher than or equal to that of the preceding version. This can be done by limiting the number of coding changes involved in each step. The assembly method is described below.

2. The task of installing new software

In order to obtain an updated software system version in an SPS controls computer, the following actions are taken.

a. Check of the justification of the proposed change
b. Planning of the different steps listed below
c. System software assembly
d. Testing on software lab. computer
e. Repeating assembly and test, if needed
f. Back-up data-tables (data-tables of working system)
g. Installation and checking (reload data-tables)
h. Updating system initialisation program (SYS.GO)
i. Decision to use the new version, or to reload the preceding one.

The above procedures all aim at obtaining a computer working more reliably or at least as well as before the change.

The procedures are performed by personnel of the computer-management section, with the help of experts only when a particular problem arises.

The rest of this note describes the process of system software assembly, which has to follow certain rules in order to obtain the required reliability.

3. Reliability and fault tracing

The software of the SPS control computers covers in a normal case, about 20,000 words of resident coding, and another 30,000 words on core-loads (drum-memory).

In order to obtain a system, a total of about 20 program-files are assembled to obtain the resident system and another 50 program-files are used for additional functions on the 16 core-loads.

The program-files mentioned are derived from the files of about 17 different programmers.

If from one system version to the next only one program-file is changed, there is a chance that one or more errors are introduced in the new version. These should be found during testing.

If, however, two programs are changed, the chance of faults is more than double, due to the additional chance of faults caused by the fact that one program assumes that the other is unchanged. This gives rise to fault probability curves of the type shown in fig. 1.

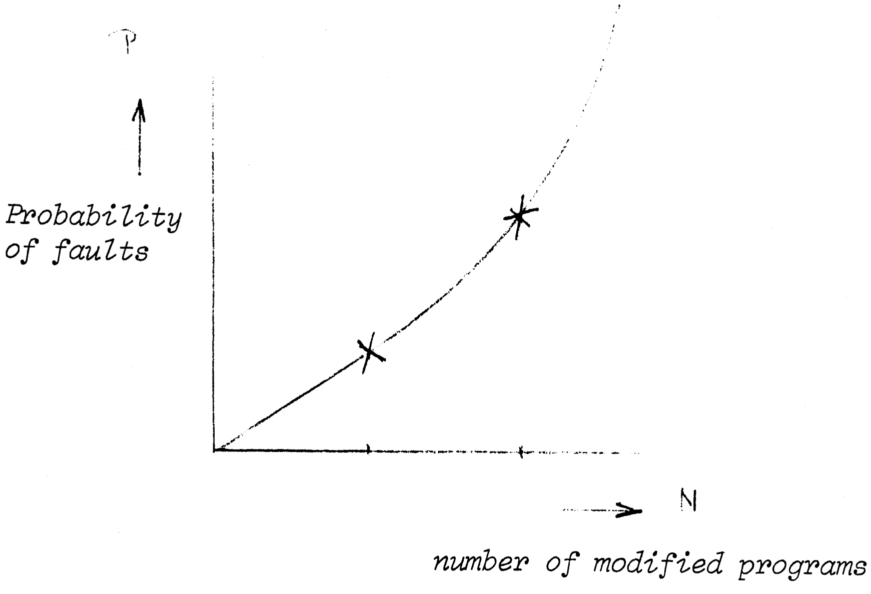


Fig. 1

The amount of non-linearity differs of course from one program set to the other. Also certain changes imply a change elsewhere, and non-combined changes cannot be avoided. In practice it works out best, if only programs that occupy themselves with one certain part of the system's activity are changed at a time.

Testing is then more effective as the source of possible errors is identified.

Experience gained during 1976 has led to the above practice, and to the assembly method below.

4. Sofware assembly

A method of software assembly which exists in essential points since January 1977 is presented in fig. 2. Two generation streams are shown. They are initiated by programs capable of assembling other programs (sometimes called 'mode-files').

The first stream results in a basic system and in a core-load that contains system support routines.

There are several other layouts of the SPS control computers. The ones for which standardised generation is proposed here are summarized in table 1.

The different types of layout are indicated with a letter; examples are given for each type. Each example can have an individual set of core-loads which makes, for instance, GP1 and GP2 to be identical systems but with certain individual characteristics in core-load layout.

The second stream of figure 2 presents the production of additional functions to be put in the resident memory; and a set of 15 additional core-loads.

During the operation of the assembly program the different program-files - that make up the system - are assembled in correct order and linked together. At the end the assembly paper-tape is punched in binary code. The tape is then used to load the computer in question. Back-up of the system can be taken on a disk file or on magnetic tape.

Table 1

System types of satellite computers*

System type	Properties	Examples (present & future)
A	128K drum, 64K memory	WEA (WEXPT) NEA (NEXPT)
B	128K drum, 32K memory	GP1-GP6, INJ, RF, GP80, GP81, GP61
C	128K drum., 48K memory	PS
D	128K drum, 40K memory	WEXTR, NEXTR

*) computers at the Control Centre have been omitted

Transport of information from the system generation to the core-load generation process is done via labels stored on common system files (RLIST and NLIST). The labels in those lists may be used only for reference by the different programs. These names are forbidden for use by individual programs.

In figure 2, the different sets of programs composing a system are indicated globally in blocks. Each block represents a user area of the TSS computer. During the last few months, most of the individual program-files have been written to so-called 'buffer-user areas'. The files in these areas can be updated at a predetermined time. In this way system changes can be designed to happen in steps of the desired size.

5. Some consequences

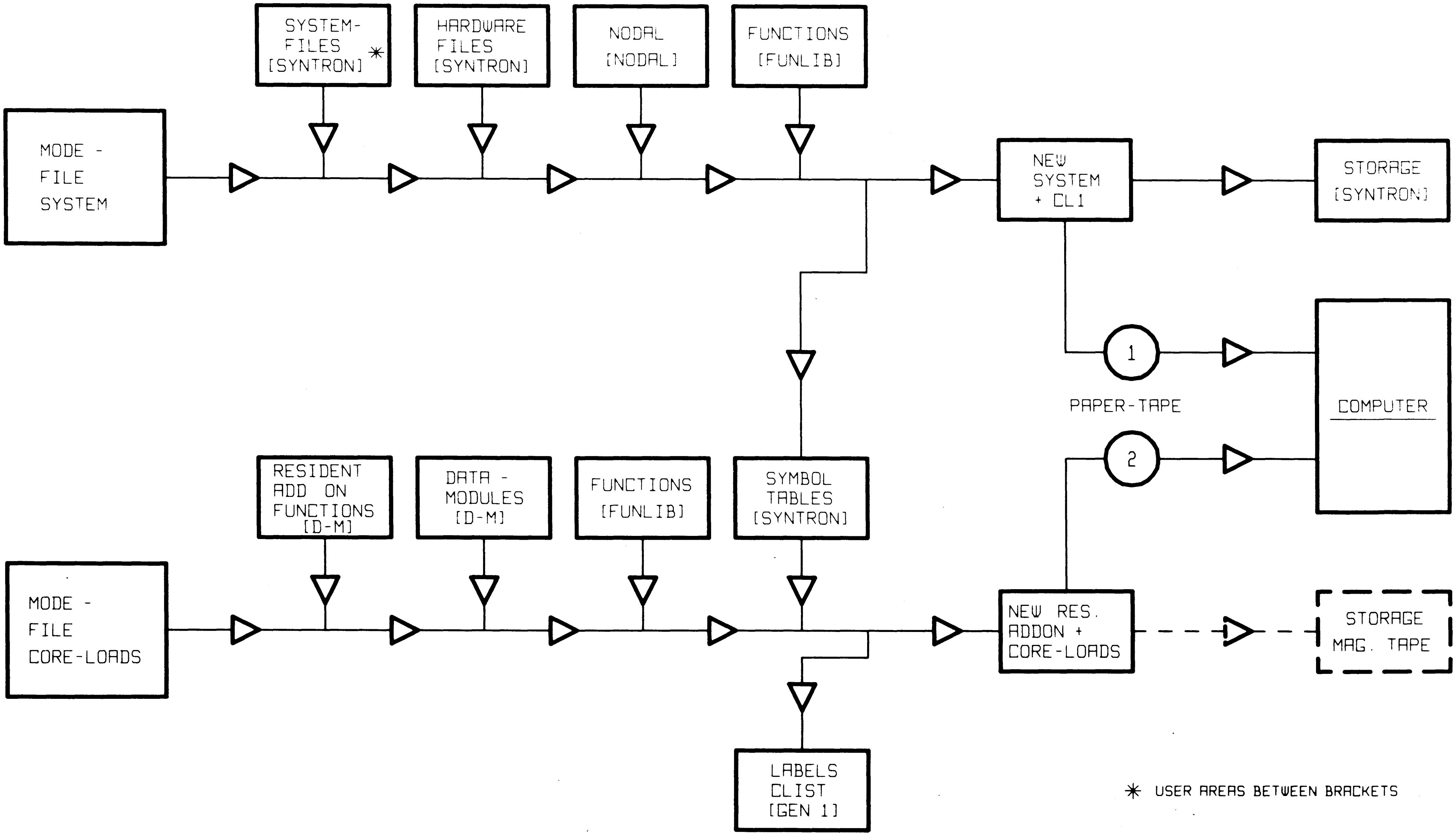
In order to have an overview of the procedure, when installing new software, there must be one team responsible for this task. A similar organization exists for hardware.

For programmers who want to test routines on a working system a facility exists to generate core-loads and resident add on parts. Testing can be done on the software lab.computer and on a computer which is temporarily available in block 2.

The core-load tapes for testing purposes must not be used for installation in the SPS controls system as this will circumvent the procedure mentioned under 2.

It is clear that new system versions can only be introduced during machine shutdown periods. Testing in the lab is done more severely than before and takes 2 to 4 days depending on the amount of faults discovered.

The above reasons tend to reduce the speed of system changes compared with the situation in 1976, though this should be considered as the price put for obtaining reliability of an operational system.



* USER AREAS BETWEEN BRACKETS

FIG. 2 ASSEMBLY STREAMS

P.BRUMMER 29/3/77
D.PERON 12/4/77