Chapter 5

PTS APPLICATION DEVELOPMENT

The development of a complete PTS application such as was outlined in Chapter 4 involves the creation of three distinct parts:

- the application program itself
- the TOSS Monitor
- the Configuration Data file

All three parts are created under the development operating system, DOS-PTS, then transferred to a medium suitable for loading under TOSS. The resulting application software is tested under TOSS. Any errors detected may be corrected by repeating the process until the application works correctly.

Development is usually carried out on a large hardware configuration, as the system software exists on hard disk and requires such things as a line printer to be available. Testing may be carried out on the same machine, or another machine may be available for this purpose, supporting only the terminal devices necessary for the application.

The development process is summarised in figure B.1 in Appendix B. This may be folded out for use with the more detailed explanation given in the rest of this chapter.

5.1 CREDIT Application Development

The CREDIT application program is written as a series of modules, one of which contains the global Data Division. All the other modules in the application refer to this Data Division for their data definitions. Every module contains a Procedure Division, where the CREDIT instructions to perform the various functions required by the application are coded.

Once they have been coded, these modules have to be input to the system. Input is normally via the console typewriter, although punched card or magnetic tape input can also be used. Each module is then translated to obtain the object code to be processed at run-time, and finally all the object modules are linked together to form the complete application program.

5.1.1 CREDIT Translator

Each module must be individually translated to check for syntax errors and to generate the intermediate object code. This is the code which will ultimately be interpreted at run time.

The translator produces three listings. The first is the source listing, containing all the source code as entered to the system, and details of any syntax errors it may contain. The other listings are of all the data items defined in the Data Division and all labels declared in the Procedure Division.

If no syntax errors are present, the intermediate object code is produced and may be kept for use in the next stage of development.

5.1.2 Updating CREDIT Source Modules

If any syntax errors occur during translation, or if any logic errors are discovered, the source modules must be updated. This can be done by using one of two processors, the Line Editor or the Text Editor.

The Line Editor is a sequential processor for updating source or data files. Each line of the file has a unique sequence number associated with it, and the Line Editor processes these lines in the sequence of their numbers.

The Text Editor is a more complex processor which offers greater flexibility in updating the file. The lines may still be identified by their sequence number, if required, but they may also be referred to by any character string they contain. This allows updates to the file to be made in any sequence.

Once the source modules have been updated, they may be kept and retranslated.

5.1.3 CREDIT Linker

The CREDIT Linker takes the Intermediate Object Modules produced by the CREDIT Translator and links them together to form one final object module. At the same time, it solves some of the references the Translator cannot solve for itself.

The Intermediate Object Modules produced by the Translator may contain two types of unresolved references: those to other CREDIT routines and those to system routines. The CREDIT Linker solves the first type of reference, references to system routines are solved by the Linkage Editor.

5.1.4 Linkage Editor

The output object file from the CREDIT Linker is then input to the Linkage Editor. The Linkage Editor will link in any system routines the application requires.

The major system routine included by the Linkage Editor is the CREDIT Interpreter. This is the part of the application program which `decodes´ the object code at run-time and executes each instruction. It must therefore be present in every CREDIT application.

Also included at this stage, unless it is specifically excluded, will be the CREDIT debugger to enable the application to be tested.

All these modules will be linked together to form one program, the application Load Module.

5.2 Generating the TOSS Monitor

The second major component of the complete system is the TOSS Monitor. The complete Monitor is supplied as part of the system software. If it were to be used complete in every application, it would be very wasteful of memory, as many of the functions would never be needed. Therefore, the relevant parts of the Monitor for the application in question are selected and linked together to form the Monitor required. So if the application does not make use of a general terminal printer, for example, the software for this type of printer will not be included and the size of the Monitor will be reduced.

The program to perform this selection and linking procedure is a standard DOS-PTS system program, SYSGEN (SYStem GENeration). This is an interactive program that is run from the console typewriter. It uses the responses to a series of questions to generate the parameters necessary to select the relevant Monitor modules.

Output from the SYSGEN program is a Monitor Load Module.

5.3 Generating the Configuration Data File

The final element needed to create a running system under TOSS is the Configuration Data file. This contains details of the numbers of terminals connected for each terminal class within the application, and other information necessary to run the system successfully, such as the priority (the relative `importance´) of each task.

The Configuration Data file is created as an ordinary data file under DOS-PTS. If necessary, it can be updated by using the Line or Text Editor, in much the same way as a source file.

5.4 Transferring from DOS-PTS to TOSS

All the procedures outlined so far in this chapter take place under the DOS-PTS operating system, but the resulting files have to be used under TOSS. As the two operating systems use different disk formats, some method must be available to effect the transfer from one operating system to the other.

Loading a TOSS system is performed from disk, either hard disk or flexible disk, and a DOS-PTS procedure ($PDISC) is available to make the transfer.

All that is then necessary is to load the device into the appropriate drive and follow the IPL (Initial Program Load) procedure for the machine. The application program will be loaded. If the CREDIT Debugger has been included, this will gain control, otherwise the application itself will begin.

5.5 Testing

When the complete application has been transferred to a TOSS-formatted disk, it may be loaded into a machine and tested. To make testing easier, the CREDIT debugger is available. The debugger is included automatically by the CREDIT Linker, unless it is specifically excluded (e.g. for a production system).

If any errors are found during testing, the source modules must be updated and the application relinked before a further test is carried out.