from MAC Research
3/7/69
Copies: Daley
Saltzer
Clingen
- Corbató

TIP/III - A Text Management System

This brief document indicates several areas in which the staff of Project TIP has demonstrated special competence. After five years of subsystems development experience on the CTSS system at Project MAC, Project TIP is now engaged in the extension of these areas and the implementation of more capable subsystems on newer machinery. These interest areas are:

1. Structuring
2. Retrieval
3. Organization
4. Editing
5. Presentation

The goal of this effort is to establish a set of harmoniously related subsystems. Initially, the subsystems will be implemented on the 360/50, 65, 67 machinery under OS/360 and the GE 645 under MULTICS. The creation of such a text management environment will considerably extend the software of these machines. Some elaboration of implementations under consideration follows.

Structuring

File systems have grown considerably in sophistication and capability in the past few years. It is now possible for a computer user to conveniently store and access fairly large data bases. Elaborate schemes make the sharing of programs practical. Similarly, several persons can read and share a data file simultaneously. Although access to files in this manner has been made possible, sharing of programs has been more successful than sharing of data. In practice, structuring of data is seldom general and flexible enough to allow the simultaneous use of a data base for two or more purposes. For example, a person computing family size for certain cities can be working from the same census data file as a person investigating the relationship between the school years completed and the change in median income. Traditionally, however, in such situations the original data base is reformatted into one or more subsidiary data bases, and these secondary sources are used for calculations and inferences. This is unsatisfactory for a number of reasons. Propagation of changes or corrections in the original data becomes tortuous and nearly impossible. The task of reformatting is itself not a very interesting or challenging intellectual occupation. The resulting data base is often a degradation, not an enhancement, of the original data.

A unified and comprehensive approach to the structuring of data will make reformatting of source data unnecessary and will remove the present barriers to effective sharing of data in a user community. Three aspects of this problem deserve major consideration:

a. Representation--The investigation of the primary structuring problems in data bases, including the nesting and encapsuling of other structures. Representation of many-leveled or tree-structured

information, for example, in a succinct and operationally useful manner, in the same structural form as list structured data.

b. Description--The development of machine usable descriptions for such structured data bases so that accessing of data elements is straightforward.

c. Transformation--The implementation of methods for altering data into equivalent forms so that the same data may be referred to quite differently by different users.

Retrieval

Project TIP has had considerable experience in text retrieval. In the TIP command now on CTSS, selection criteria may be established over specific, named subelements and related by various boolean functions. More recently, some experimentation has been done in retrieval by structure as well as content. For example, it is possible to select those elements having three or more of one type of subelement, two or less of another type, and, further, containing some specific text. Our most extensive application of retrieval by relationship has been in the field of bibliographic coupling but we forsee many important general innovations in this area. Specifically, we propose to make available a text retrieval system in which it will be possible to retrieve:

a. By Content--Where specific text may be used as a selection criterion. Indefinite ways of identifying string patterns will also be available.

b. By Structure--Where the overall pattern of elements and subelements may be used as a selection criterion.

c. By Relationship--Where a relationship of similarity of contents or structure may be used as a selection criterion. Retrieval of text pointed to by other text will be possible.

Organization

The data bases maintained by Project TIP are now growing at the rate of eighteen million characters per month. To cope with this mass of data and to organize it rationally we have developed: 1. the most capable and efficient file sorting subsystem available on CTSS and the only such subsystem with no size limitations on the length of files; 2. a powerful merging and updating system; and, 3. a tape storage controlling system which maintains disk-based file directories for tape files and allows safe archival preservation of data. We believe the following will be essential functions in the TIP/III text management system:

a. Sorting--The massive re-arrangement of elements by content or structure.

b. Merging--The massive updating of elements into another collection of elements and the partitioning collections.

c. Meshing--The aggregation of elements into new structural arrangements and the concatenation of structural parts.

?
I would hope that proper file organization might eliminate them — or lessen at the user level — they are techniques for efficient retrieval from serial devices.

Editing

The editing subsystems on CTSS are among the most advanced ever implemented but they are tuned to the small user. Editing large files is nearly impossible without breaking the files down and then reconstituting them. The economics of such editing is also intolerable. In response to this, Project TIP developed the subsystem EDIT, a flexible programmable editor; and QEDIT, by far the fastest large file editor on CTSS. The QEDIT subsystem provides substantial reporting of all changes made in a file and took away our hand-to-mouth attitude about editing. We forsee a need

What is the fundamental issue? Is it a file organization problem rather than a tool problem?

for two subsystems:

a. A Programmable Editor that can react in an interpretive or compiler mode.

b. A Reporting Editor that economically alters large files and makes available a record of changes to a file.

Presentation

We have experimented with table-driven methods of creating complex output presentations. Headings, indentations, underlining, and many other functions were adequately handled. In spite of this, interaction with CTSS formatting subsystems such as RUNOFF was often necessary to provide the final output desired by the book publishing industry. We believe still stronger presentation methods are needed. Also, higher-grade output devices must be interfaced with to satisfy the needs of the community. Subsystems must be developed with special reference to the accommodation of three types of displays:

a. Soft Copy Display Consoles

b. High-Speed Film Printers

c. Graphic-Arts Quality Typesetting Machines

TIP/III - A Text Management Language

Two other interest areas must be considered since reasonable implementation rests upon sound judgement in choosing basic construction tools and obtaining consistent feedback about the effectiveness of those tools. These areas support the implementation of each subsystem:

1. Language Definition
2. Subsystem Tuning

Language Definition

In the development of CTSS subsystems, we have gained experience with a precompiler called ASEMBL and a highly modified FAP called TAP. ASEMBL constructs tables, dictionaries and pointers to ASCII text streams. TAP allows the writing of highly stylized machine language code. Coding in assembly language is not desirable for TIP/III, yet coding in a much higher level language such as PL/1 leaves much to be desired. We feel that efficiency of operation, clarity of code, and exportability can all be served. To do so we propose to create:

a. TML--TIP Macro Language or Text Management Language. A machine independent assembler-like language with text manipulative features. A system programmer's programming language compatible with higher level languages.

b. A TML Bootstrap--A higher-level bootstrap compiler for TML. The bootstrap will be coded in AED or PL/1. The actual compiler for TML will be written in TML.

SubSystem Tuning

Our ability to create subsystems has outstripped our ability to analyze and understand how they operate once they are running. It is now possible to construct subsystems that are unmaintainable--even the designer has no way of monitoring or understanding them.

Project TIP has substantially reworked FAPDBG to create the command DEBUG and redesigned STRACE to produce TRACE. These debugging tools, together with our subsystem TIME form a basic tuning aid for designers. Without such aids, powerful

information handling subsystems cannot be confidently built. We intend to investigate:

a. A Time Use Monitor which gives load factors and usage statistics for each component of software code.

b. A Space Use Monitor which gives a dynamic profile of storage requirements for each component of software copy.

Delivery date for the first TIP/III system is July 1970.

for everything ?

Emphasis on tools instead of on problems