

AUBRY A.M. et al., 1984.- From POSEIDON to NEPTUNE: Software for environmental surveys. In Burrough P.A. & Bie S.W. (Eds). Soil information systems technology. Proc. 6 th meeting ISSS working group on soil information systems. Bolkesj , 28 Feb.-4 March 1983, pp. 41-45

## From POSEIDON to NEPTUNE - Software for environmental surveys

A.M. Aubry et al.<sup>1</sup>

After the major change in host computers that occurred after our meeting in Druzhba, the project to convert the system from Fortran to Fortran 4 appeared inappropriate. Therefore, a more sophisticated system was designed, flow-charted, programmed in PL/1 on an Amdahl 470/V7 CNRS-CIRCE computer. Consequently, the new software was applied to factual data from two divergent terminologies.

The experiment currently under way is with descriptors from the second terminology (on exhibit: Recherche d'un langage transdisciplinaire pour l' tude du milieu naturel. Travaux et documents Orstom, No 91, 143 pp., 1978). No doubt, the system can as well be implemented on any terminology, as expressly stated through the acronym NEPTUNE Nouvel Ensemble de Programmes pour Terminologie Usuelle Non-format e   l'Entr e. Users preferring the first acronym need not change, as it is spelt out PL/1 Oversimplified System Ensuring Immediate Databank Operation in Natural Language or Programmes Organis s de Saisie, d'Edition et d'Interrogation pour Descriptions Ordinairement Non-ordonn es.

### Data input

By data input, character strings only are keyed in. Character strings as defined by the system are of 3 types: enumerated strings, equality strings and ordinary strings.

(1) *Enumerated strings* are 1 to 56 characters in length, with digits, capital letters, blank spaces, and a few EBCDIC special characters % \* / : ; , ' - \_ \$ and a terminating period. The remaining characters are system oriented. At least one character is not a digit. Blank spaces before the string are not read. The name 'enumerated string' reminds the user that it is to be found in the

1. A.M./Aubry, A./Beaudou, Y./Boulvert, E./Braudeau, Y./Chatelin, J./Collinet, J.F./Dupon, J.C. Filleron\*, M./Fromaget, A. Garcia G mez, A. Giey\*\*, Ph. Horent\*, B. Koli Bi\*, R./Poss, J.F./Richard, G./Riou and R./Van den Driessche;

Office de la Recherche Scientifique et Technique Outre-Mer, research unit 'Etudes Int gr es et cartographie des milieux naturels et humanis s par les m thodes de la typologie et de l'informatique', 70 route d'Aulnay, F 93140 Bondy, France;

Universit  Nationale de C te d'Ivoire (\*), Abidjan;

Sociologociels, 20 rue Ste Croix Bretonnerie, 75004, Paris (\*\*), France



nomenclature, which, let us stress the point, can be updated instantaneously. By the way, the nomenclature is split into a (sampling) unit section, whose descriptors are used once in a record, and a subunit section, whose descriptors are used repeatedly. Within each section, the descriptors are aggregated into variables. All data are of the ordinal or nominal type. Names of the variables are not used by data input, only by search.

(2) *Equality strings* contain a simple equals sign = followed by a signed integer in the range -32767 to 32767 (optional positive sign) and a period. Left of the = character is a string 1 to 56 characters long (same as above) or the abbreviated 1 to 4 characters synonym explicitly stated in the nomenclature (at least one character not a digit, no blanks within the synonym). Both strings may be spelt the same way when this seems preferable. Data are of the interval type in the great majority of cases.

(3) *Ordinary strings* are introduced by a plus sign + and terminated upon a period. All EBCDIC characters are used here, except . and @. Such an ordinary string may be unique, 1 to 400 characters in length, or split into unequal strings dispersed within the unit description. To avoid lengthy records, the current system excludes ordinary strings from the subunits as there are 20 subunits in a unit. Comments are the most frequent type of ordinary string. However, an uncertain, biased, or even highly significant descriptor may enter as an ordinary string with the appropriate remark, duplicating the enumerated or equality string. Also when there are subunits in excess, the additional ones may enter as ordinary strings.

Just a hint about keywords; 2 logical keywords are mandatory: the names given to unit and subunit in the form of equality strings, e.g. H=7.L=4. --- .L=2. ---. L=1. -----. Before input, a visual check of key presence and accuracy is advocated. Subunits always follow their unit and may be entered in any order. Moreover, units need not follow the order of their integer identification number.

#### Validation

Validation is omnipresent as storage occurs solely upon validation of the strings found in the nomenclature. By validation, 3 situations occur:

1. The *enumerated string* is read and then located in the nomenclature by scanning, if necessary, the whole list; if not found, it is printed and annotated by an interrogation mark.
2. For an *equality string*, both string and integer are subject to control: string must be found and integer must lie within the specified range; here again, verification is concluded by printing the input string either plus a period or plus an interrogation mark.
3. An *ordinary string* cannot be validated; total length, however, must comply with limits; with overflow, an arrobas @ sign replaces characters in excess that had nevertheless printed and were truncated.

When two or more descriptors from the same variable enter the same subunit or unit, the last one only is validated and stored, although all are output on the lineprinter. What occurs when a period is not keyed in? The resulting concatenated string is rejected, printed and annotated, even if both descriptors are valid!

### *Nomenclature*

Within the nomenclature, strings for the unit description are presented first; the second part, under the title subunit, covers all strings available repeatedly. Enumerated strings are not presented alphabetically, but in a logical order suited for later editing (see below). They are aggregated under a fixed number of variables. The variables too receive names: the usual name and an abbreviated name, both available by search, never by data input (the two names need not be different). Such a variable is followed by a list of strings, each one being matched with an integer value. To a given integer can be associated more than one string: synonyms, abbreviations, codes . . . , all available upon data input and upon search. Such a list is permanently open to receive new strings and its content may be reordered (first string is privileged by editing). A descriptor never has the same spelling as a variable; the difference may be in a single character. For equality strings, the nomenclature contains only the two strings and the anticipated minimum and maximum, the narrower the range the better the validation. Obviously, all strings of the nomenclature must differ to permit recognition and non-redundancy is checked.

The limits set on a nomenclature file and the current status are as follows.

	Maximum	No 6	No 7
within-unit			
number of variables	151	151	151
number of strings	1000	311	416
number of characters	32752	8965	6319
within-subunit			
number of variables	18	16	16
number of strings	1000	730	333
number of characters	32752	4814	3119

Nomenclatures No 6 and No 7 contain 16 subunit variables (\*) and 2 dummy variables for later use. When upper limits are reached, it may prove advantageous to build the system with several homologous nomenclatures, as explained in Orstom Init. Doc. Tech. No 25, 1974. Number 7 contains numerous redundancies and has, nevertheless, been stored, on the same disk, by a modified storage programme lacking the 'non-redundancy' constraint and that for editing purposes only.

### *Storage*

The host computer is an AMDAHL 470/V7 linked to an IBM 370/168

under MVS+JES3 at CNRS-CIRCE, on the Orsay campus. At least 320 terminals cover the country. Furthermore, a second computer, an IBM 3033/U08, at CNRS-CNUSC, on the Montpellier campus, has been introduced in the same network. One of the ORSTOM remote batch terminals is a CII-HA-MINI6 computer with a dozen implemented keyboard+video consoles. For the files, indexed data are the key to organization on a resident 3350 disk. Integers from the nomenclature are stored in binary fixed, the ordinary strings are concatenated and stored in the same record as character string. The fixed length record has following structure:

- a non-embedded key 5 bytes;
- the 151 integers from unit 302 bytes,
- the character string 400 bytes,
- the 20 subunits of 18 integers each 720 bytes;

with a total record length of 1422 bytes, or 711 half-words of 16 bits. Missing values are stored as an integer specified in the nomenclature (-32768). Integers are ordered as in the nomenclature.

#### *Update*

Update may be local or global. The latest read descriptor of a given variable is the one to get storage. We take advantage of this by LOCAL update. For a misspelling, for instance, unit and subunit (when relevant) are identified and the descriptor typed correctly. Local update permits also erasure of a given descriptor by a single \* keystroke (key £ on ASCII console). By GLOBAL update, an externally stored record (on floppy) is modified on the video, read, and stored, erasing completely the previous record bearing the same number.

#### *Edit*

To obtain lineprinter output in plain language of individual records or of a series of records, we give their identification numbers, e.g. 1801, 1609 A 1622. Editing a complete retrieved subfile is by giving its own number.

#### *Search*

All strings currently in the nomenclature are available for search purposes. Ordinary strings cannot intervene. Enumerated strings and equality strings are used simultaneously, in any number, in short or long spelling and in any order. When appropriate, a subscript (in parenthesis) is added to the string in order to restrict search within the corresponding subunits. There is no need to learn a search command language: the terminology is familiar to the user, and the number of symbols (system-oriented and nomenclature) has been reduced.

Applicability of the operators  $\neg$  &  $|$  on operands type  $A > B$ ,  $A < B$ ,  $A = B$ , where A and B are strings or integers, under normal priority rules, or in parenthesis, offers as wide a spectrum of search capabilities. On local keyboards, press ^ key for logical negation  $\neg$

(not), & for logical conjunction (and), and ! for logical disjunction | (or). Parentheses are available to modify priority. The negation is used as follows:  $\neg$  string = integer  $\neg$  string = string (with = > or < ). All integers, even out of range, can be used. No periods are allowed within the search formula, only one final period. The search strings are validated and the logic of the syntax is checked.

The search formula is converted to reverse-polish form:

if        A ,    B ,    C , . . . are operands  
and       a ,    b ,    c , . . . are operators  
a ( ( A b ( B c C ) ) b ( ( B c C ) d D ) )  
is written  
A B C c b B C c D d b a

Search is in batch mode and output is on the lineprinter, or on console CRT via a local disk, as an ordered list of record numbers. These records may be retrieved in totality as a subfile bearing the same physical and logical characteristics as the main file. This enables the user to follow the search on the subfile and this, of course, may be repeated as NESTED SEARCH. Lineprinter output in plain language of the retrieved records may be requested at any step of the nested search.