



```

SSSSSSSS MM MM GGGGGGGG TTTTTTTTTT AAAAAA BBBB BBBB DDDDDDDD EEEEEEEEE EEEEEEEEE FFFFFFFF
SSSSSSSS MM MM GGGGGGGG TTTTTTTTTT AAAAAA BBBB BBBB DDDDDDDD EEEEEEEEE FFFFFFFF
SS MM MM MM GG GG GG TTT TTT AA AA BB BB DD DD EE EE FF
SS MM MM MM GG GG GG TTT TTT AA AA BB BB DD DD EE EE FF
SS MM MM MM GG GG GG TTT TTT AA AA BB BB DD DD EE EE FF
SSSSSS MM MM GG GG GGGGGG TTT TTT AA AA BBBB BB DD DD EEEEEEE FFFFFFFF
SSSSSS MM MM GG GG GGGGGG TTT TTT AA AA BBBB BB DD DD EEEEEEE FFFFFFFF
SS MM MM GG GG GG TTT TTT AA AA BBBB BB DD DD EE EE FF
SS MM MM GG GG GG TTT TTT AA AA BBBB BB DD DD EE EE FF
SS MM MM GG GG GG TTT TTT AA AA BBBB BB DD DD EE EE FF
SSSSSS MM MM GGGGGG TTT TTT AA AA BBBB DDDDDDD EEEEEEEEE FF
SSSSSS MM MM GGGGGG TTT TTT AA AA BBBB DDDDDDD EEEEEEEEE FF

```

```

....
....
....
....

```

```

RRRRRRRR EEEEEEEEE QQQQQQ
RRRRRRRR EEEEEEEEE QQQQQQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RRRRRRRR EEEEEEEEE QQ QQ
RRRRRRRR EEEEEEEEE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EE QQ QQ
RR RR EEEEEEEEE QQQQ QQ
RR RR EEEEEEEEE QQQQ QQ

```

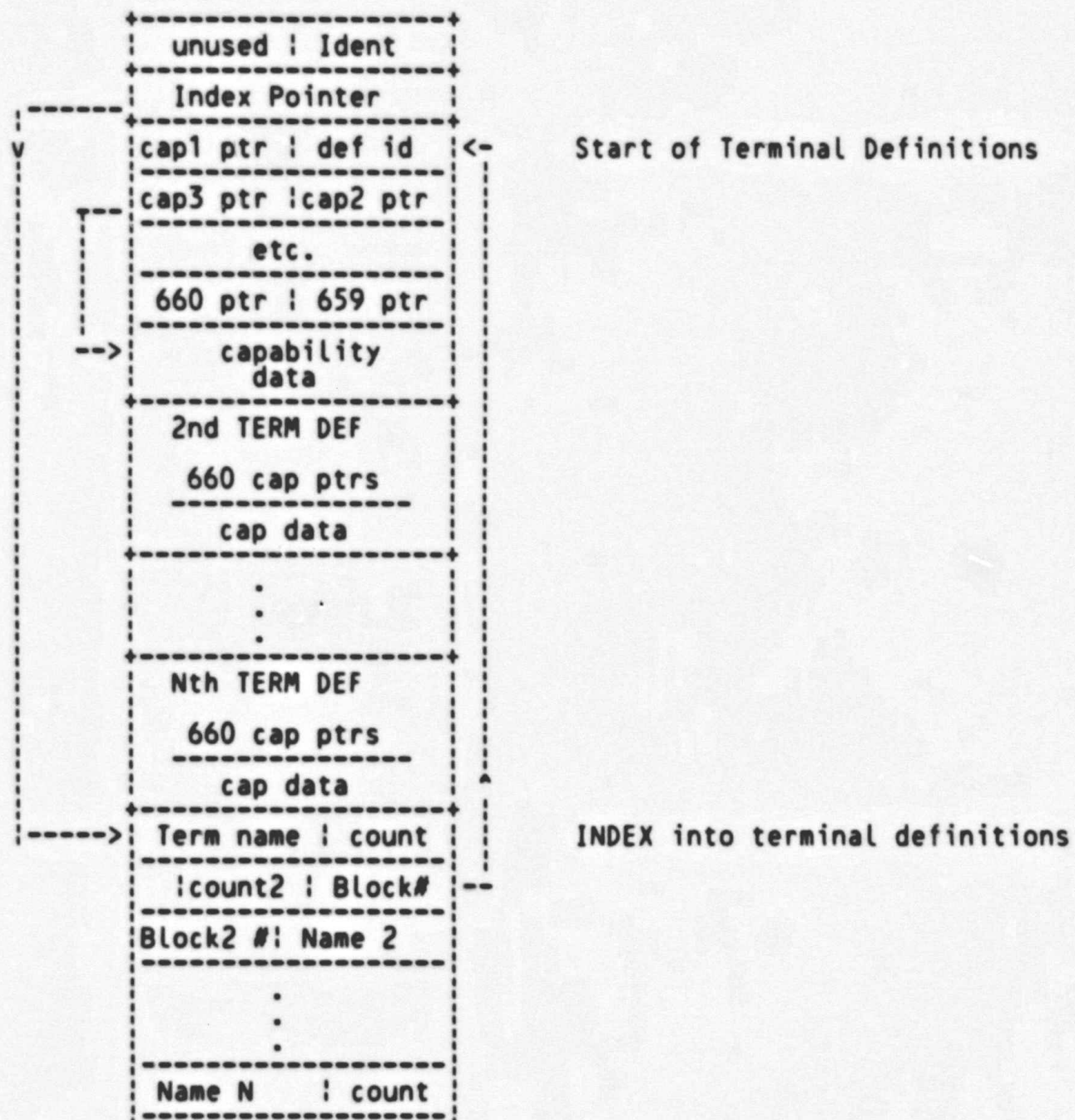






TERMTABLE.EXE also contains an index so that individual terminal definitions can be located. The index will consist of a count, the ascii name for the terminal, and the block number where the terminal definition begins. The index will actually be located at the end of TERMTABLE.EXE. This is because the number of terminal definitions is unpredictable, and we don't want to impose an arbitrary size constraint that would restrict the number of terminals defined.

TERMTABLE.EXE will begin with an ident number, to allow for future changes. These future changes would probably allow more terminal capabilities (greater than 660).





```

| | 0 | BlockN# | count = 0 --> end of terminal def ptrs
|-----|

```

```

|+ Offsets used to access a terminal table.
|-

```

```

MACRO
TTB_W_IDENT          = 0, 0, 16, 0%, | ident field (to allow for
TTB_W_unused         = 2, 0, 16, 0%, | future changes)
TTB_L_INDEX_OFFSET   = 4, 0, 32, 0%, | not used
                       | offset from here to terminal
                       | index

```

```

|+ Define constants needed to build terminal table
|-

```

```

LITERAL
SMGSK_HEADER_SIZE    = 512,          | 1 block for header info
SMGSK_TERM_INDEX_SIZE = 5000,        | arbitrary size for index
SMGSK_CAP_PTRS_SIZE  = 1536,        | 3 blocks to hold ptrs
SMGSK_CAP_DATA_SIZE  = 5120,        | 10 blocks to hold data
SMGSK_TERM_DEF_SIZE  = 6656,        | ptrs + data (3+10 blocks)
SMGSK_TERM_DEF_ID    = 137,        | random number to identify
SMGSK_TERM_DEF_ID_OFFSET = 0,        | a valid terminal def
SMGSK_MIN_REQUEST_CODE = 1,          | location of id - 1st
SMGSK_MAX_REQUEST_CODE = 660;       | word in ptr vector
                                   | lowest capability number
                                   | highest capability number

```

```

|+ The following are used when parsing arithmetic expressions.
|- Information about expressions is stored via these encodings.

```

```

LITERAL
SMGSK_FAO_STRING      = %X'FFFFFFFF',
SMGSK_ARITH_STRING    = %X'FFFFFFFE',
SMGSK_OPERAND         = %X'FFFFFFFD',
SMGSK_SUBSTITUTE      = %X'FFFFFFFC',
SMGSK_ADD              = %X'FFFFFFFB',
SMGSK_SUBTRACT        = %X'FFFFFFFA',
SMGSK_MULTIPLY        = %X'FFFFFFF9',
SMGSK_DIVIDE          = %X'FFFFFFF8',
SMGSK_STORE           = %X'FFFFFFF7';

```



