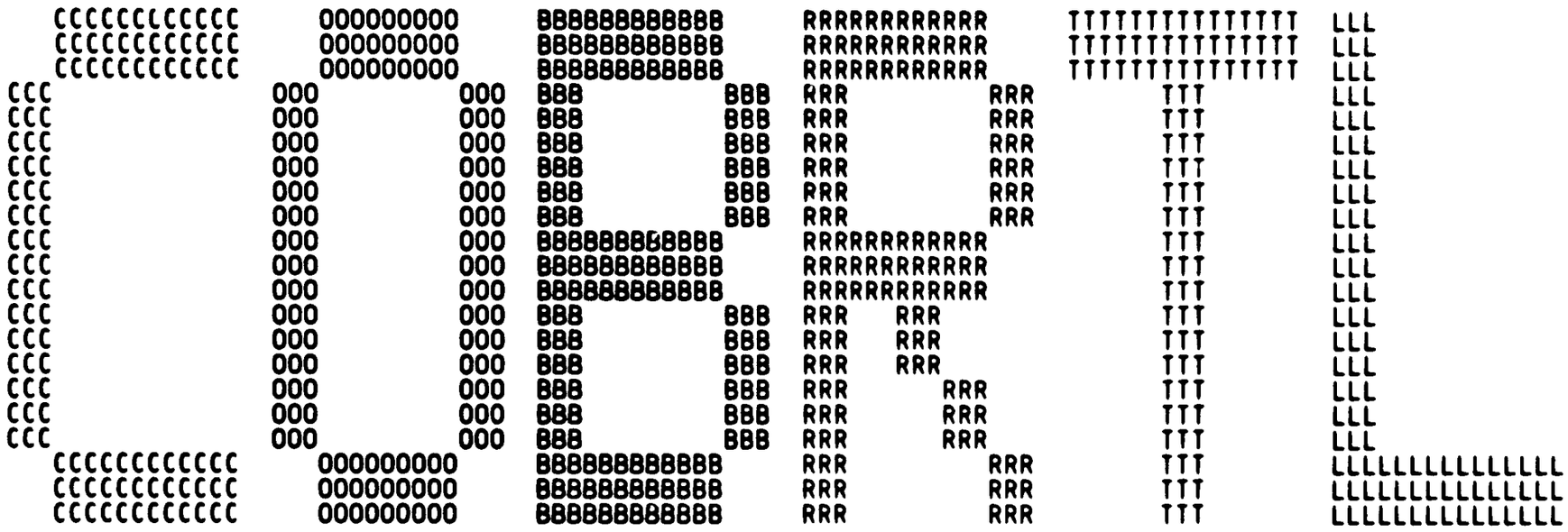
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL

FILEID**COBVECTOR

```
  CCCCCCCC    000000    BBBBBBBB    VV      VV  EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT    000000    RRRRRRRR
  CCCCCCCC    000000    BBBBBBBB    VV      VV  EEEEEEEEEE    CCCCCCCC  TTTTTTTTTT    000000    RRRRRRRR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR      RR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR      RR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR      RR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR      RR
CC          00      00  BBBBBBBB    VV      VV  EEEEEEEE    CC              TT      00      00  RRRRRRRR
CC          00      00  BBBBBBBB    VV      VV  EEEEEEEE    CC              TT      00      00  RRRRRRRR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR  RR
CC          00      00  BB      BB  VV      VV  EE          CC              TT      00      00  RR  RR
CC          00      00  BB      BB    VV  VV    EE          CC              TT      00      00  RR    RR        ....
CC          00      00  BB      BB    VV  VV    EE          CC              TT      00      00  RR    RR        ....
  CCCCCCCC    000000    BBBBBBBB        VV      EEEEEEEEEE    CCCCCCCC      TT        000000    RR      RR      ....
  CCCCCCCC    000000    BBBBBBBB        VV      EEEEEEEEEE    CCCCCCCC      TT        000000    RR      RR      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

Table of contents

```
0000     1         .TITLE  COB$VECTOR - Entry vectors for COBRTL.EXE
0000     2         .IDENT  /1-009/                 ; File: COBVECTOR.MAR Edit: LGB1009
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28
0000    29 ;++
0000    30 ; FACILITY: Run-Time Library - COBOL Language Support
0000    31 ;
0000    32 ; ABSTRACT:
0000    33 ;
0000    34 ;       This module contains the entry vector definitions for the
0000    35 ;       VAX-11 Run-Time Library shareable image COBRTL.EXE
0000    36 ;
0000    37 ; ENVIRONMENT: User mode, AST Reentrant
0000    38 ;
0000    39 ; AUTHOR: Steven B. Lionel, CREATION DATE: 29-October-1982
0000    40 ;
0000    41 ; MODIFIED BY:
0000    42 ;
0000    43 ; 1-001 - Original. SBL 29-October-1982
0000    44 ; 1-002 - Add entry for COB$$HANDLER so that VMSRTL can reference the mask.
0000    45 ;         SBL 11-May-1983
0000    46 ; 1-003 - Add new entries for COB$DISPLAY and COB$POS_ERASE.  LEB 20-May-1983
0000    47 ; 1-004 - Add all COBOL code that can be shared.  LEB 23-May-1983
0000    48 ; 1-005 - Take out COB$$RESTVA (found during an RTL test build, so no affect
0000    49 ;         on programs).  LEB 24-May-1983
0000    50 ; 1-006 - Change reference from COB$DISPLAY_SCR to COB$DISP_SCR as well as
0000    51 ;         COB$DISPLAY_SCR_NO_ADV to COB$DISP_SCR_NO_ADV.  LEB 25-May-1983
0000    52 ; 1-007 - Add entry COB$ACC_SCR to module COB$ACCEPT.  LEB 22-Aug-1983
0000    53 ; 1-008 - add entry COB$$RET_A_AB_PREV to module COB$DISPLAY.  MDL 29-Aug-1983
0000    54 ; 1-009 - Add entries COB$POS_ACCEPT and COB$POS_DISPLAY to COB$POS_ERASE.
0000    55 ;                                                                 LGB 27-Feb-1984
0000    56 ;--
```

```
                        58          .SBTTL  DECLARATIONS
                        59 ;
                        60 ; LIBRARY MACRO CALLS:
                        61 ;
                        62 ;       NONE
                        63 ;
                        64 ; EXTERNAL DECLARATIONS:
                        65 ;
0000                    66          .DSABL  GBL             ; Force all external symbols to be declared
                        67 ;
                        68 ; MACROS:
                        69 ;
0000                    70
0000                    71 ;+
0000                    72 ; Macro to define an entry vector for a CALL entry point
0000                    73 ;-
0000                    74
0000                    75          .MACRO  VCALL   NAME, ALTMSK
0000                    76          .EXTRN          NAME
0000                    77          .TRANSFER       NAME
0000                    78          .IF B ALTMSK
0000                    79          .MASK           NAME
0000                    80          .IFF
0000                    81          .MASK           ALTMSK
0000                    82          .ENDC
0000                    83          JMP             NAME+2
0000                    84          .ENDM
0000                    85
0000                    86 ;+
0000                    87 ; Macro to define an entry vector for a JSB entry point
0000                    88 ;-
0000                    89
0000                    90          .MACRO  VJSB    NAME
0000                    91          .EXTRN          NAME
0000                    92          .TRANSFER       NAME
0000                    93          JMP             NAME
0000                    94          .BLKB           2
0000                    95          .ENDM
0000                    96
0000                    97 ;+
0000                    98 ; Macro to define an entry vector for a condition handler whose actual
0000                    99 ; routine address has a different name from the vector entry.
0000                   100 ;-
0000                   101
0000                   102          .MACRO  VHANDL  NAME, INTNAME
0000                   103          .EXTRN          INTNAME
0000                   104 NAME::
0000                   105          .TRANSFER       NAME
0000                   106          .MASK           INTNAME
0000                   107          JMP             INTNAME+2
0000                   108          .ENDM
0000                   109
0000                   110 ;+
0000                   111 ; Macro to define an alias for the next vectored entry point
0000                   112 ;-
0000                   113
0000                   114          .MACRO  ALIAS   NAME
```

```
                        115          .TRANSFER       NAME
                        116          .ENDM
                        117
                        118
                        119 ;
                        120 ; EQUATED SYMBOLS:
                        121 ;
                        122 ;        NONE
                        123 ;
                        124 ; OWN STORAGE:
                        125 ;
                        126 ;        NONE
                        127 ;
                        128 ; PSECT DECLARATIONS:
                        129 ;
                  00000000  130          .PSECT $COB$VECTOR PIC, USR, CON, REL, LCL, SHR, -
                        131                                EXE, RD, NOWRT, LONG
                        132
```

```
                        .SBTTL  COBRTL Vector

;+
; Define vectored entry points for the COBOL Language Support procedures
; by module in alphabetical order.
;
; Any additions to this file should be reflected in
; COM$:COBRTLVEC.DAT.  All new entry points must be appended to the end
; of the list.  NEVER change existing entries unless you are sure that
; what you do won't break existing programs.
;-

; Module COB$$HANDLER
        VHANDL  COB$HANDLER     COB$$HANDLER

; Module COB$ACCEPT
        VCALL   COB$ACCEPT

; Module COB$ACC_DATE
        VCALL   COB$ACC_DATE

; Module COB$ACC_DAY
        VCALL   COB$ACC_DAY

; Module COB$ACC_DAYWEEK
        VCALL   COB$ACC_DAYWEEK

; Module COB$ACC_TIME
        VCALL   COB$ACC_TIME

; Module COB$CVTPQ_R9
        VJSB    COB$CVTPQ_R9

; Module COB$CVTQP_R9
        VJSB    COB$CVTQP_R9

; Module COB$CVTRPQ_R9
        VJSB    COB$CVTRPQ_R9

; Module COB$CVTRQP_R9
        VJSB    COB$CVTRQP_R9

; Module COB$DISPLAY
        VCALL   COB$DISPLAY
        VCALL   COB$DISP_NO_ADV

; Module COB$DIVQ_R8
        VJSB    COB$DIVQ_R8

; Module COB$ERROR
        VCALL   COB$ERROR

; Module COB$INTARI
        VCALL   COB$ADDI
        VCALL   COB$CMPI
        VCALL   COB$DIVI
        VCALL   COB$DIVI_OSE
```

```
0090   191          VCALL   COB$MULI
0098   192          VCALL   COB$SUBI
00A0   193
00A0   194 ; Module COB$INTER
00A0   195          VJSB    COB$CVTDI_R7
00A8   196          VJSB    COB$CVTFI_R7
00B0   197          VJSB    COB$CVTID_R7
00B8   198          VJSB    COB$CVTIF_R7
00C0   199          VJSB    COB$CVTIL_R8
00C8   200          VJSB    COB$CVTIP_R9
00D0   201          VJSB    COB$CVTIQ_R8
00D8   202          VJSB    COB$CVTIW_R8
00E0   203          VJSB    COB$CVTLI_R8
00E8   204          VJSB    COB$CVTPI_R9
00F0   205          VJSB    COB$CVTQI_R8
00F8   206          VJSB    COB$CVTRIL_R8
0100   207          VJSB    COB$CVTRIP_R9
0108   208          VJSB    COB$CVTRIQ_R8
0110   209          VJSB    COB$CVTRIW_R8
0118   210          VJSB    COB$CVTTI_R8
0120   211          VJSB    COB$CVTWI_R8
0128   212
0128   213 ; Module COB$IOEXCEPTION
0128   214          VCALL   COB$IOEXCEPTION
0130   215
0130   216 ; Module COB$MULQ_R8
0130   217          VJSB    COB$MULQ_R8
0138   218
0138   219 ; Module COB$PAUSE
0138   220          VCALL   COB$PAUSE
0140   221
0140   222 ;+
0140   223 ; End of initial COBRTL vector.  All subsequent additions must be made
0140   224 ; after this point.
0140   225 ;-
0140   226
0140   227 ;+
0140   228 ; The following entry is present so that the vector of VMSRTL can reference
0140   229 ; the mask of COB$$HANDLER.
0140   230 ;-
0140   231
0140   232 ; Module COB$$HANDLER
0140   233          VCALL   COB$$HANDLER
0148   234
0148   235 ;+
0148   236 ; Add the following entries for the new version of COBOL.
0148   237 ;-
0148   238
0148   239 ; Module COB$DISPLAY
0148   240          VCALL   COB$DISP_SCR
0150   241          VCALL   COB$DISP_SCR_NO_ADV
0158   242
0158   243 ; Module COB$POS_ERASE
0158   244          VCALL   COB$POS_ERASE
0160   245
0160   246
0160   247 ;+
```

```
                     ; Add all remaining COBOL specific code that can be shared (note that
                     ; the exceptions are tables, COB$FIND_NAME, COB$CALL, COB$CANCEL,
                     ; COB$RMS_BLOCKS, COB$DHANDL, COB$$RESTVA.
                     ;-

                     ; Module COB$CVTDQ_R8

                             VJSB    COB$CVTDQ_R8

                     ; Module COB$CVTFQ_R8

                             VJSB    COB$CVTFQ_R8

                     ; Module COB$CVTQD_R8

                             VJSB    COB$CVTQD_R8

                     ; Module COB$CVTQF_R8

                             VJSB    COB$CVTQF_R8

                     ; Module COB$CVTRDQ_R8

                             VJSB    COB$CVTRDQ_R8

                     ; Module COB$CVTRFQ_R8

                             VJSB    COB$CVTRFQ_R8

                     ; Module COB$DBEXCEPTION

                             VCALL   COB$DBEXCEPTION

                     ; Module COB$EXPI

                             VCALL   COB$EXPI
                             VCALL   COB$EXPI_OSE

                     ; Module COB$LINAGE

                             VCALL   COB$INIT_LINAGE
                             VCALL   COB$LINAGE
                             VCALL   COB$TERM_LINAGE

                     ; Module COB$SET_SWITCH

                             VCALL   COB$SET_SWITCH

                     ; Module COB$SWITCH

                             VCALL   COB$SWITCH

                     ; Module COB$ACCEPT (new entry to an already existing shared module)

                             VCALL   COB$ACC_SCR
```

```
01D8    305 ; Module COB$DISPLAY (new entry to an already existing shared module)
01D8    306
01D8    307         VCALL   COB$$RET_A_AB_PREV
01E0    308
01E0    309 ; Module COB$POS_ERASE (new entries to an already existing shared module)
01E0    310
01E0    311         VCALL   COB$POS_ACCEPT
01E8    312         VCALL   COB$POS_DISPLAY
01F0    313
01F0    314         .END                                    ; End of module COB$VECTOR
```

COB$$HANDLER	********	X	01	COB$POS_ERASE	********	X	01
COB$$RET_A_AB_PREV	********	X	01	COB$SET_SWITCH	********	X	01
COB$ACCEPT	********	X	01	COB$SUBI	********	X	01
COB$ACC_DATE	********	X	01	COB$SWITCH	********	X	01
COB$ACC_DAY	********	X	01	COB$TERM_LINAGE	********	X	01
COB$ACC_DAYWEEK	********	X	01				
COB$ACC_SCR	********	X	01				
COB$ACC_TIME	********	X	01				
COB$ADDI	********	X	01				
COB$CMPI	********	X	01				
COB$CVTDI_R7	********	X	01				
COB$CVTDQ_R8	********	X	01				
COB$CVTFI_R7	********	X	01				
COB$CVTFQ_R8	********	X	01				
COB$CVTID_R7	********	X	01				
COB$CVTIF_R7	********	X	01				
COB$CVTIL_R8	********	X	01				
COB$CVTIP_R9	********	X	01				
COB$CVTIQ_R8	********	X	01				
COB$CVTIW_R8	********	X	01				
COB$CVTLI_R8	********	X	01				
COB$CVTPI_R9	********	X	01				
COB$CVTPQ_R9	********	X	01				
COB$CVTQD_R8	********	X	01				
COB$CVTQF_R8	********	X	01				
COB$CVTQI_R8	********	X	01				
COB$CVTQP_R9	********	X	01				
COB$CVTRDQ_R8	********	X	01				
COB$CVTRFQ_R8	********	X	01				
COB$CVTRIL_R8	********	X	01				
COB$CVTRIP_R9	********	X	01				
COB$CVTRIQ_R8	********	X	01				
COB$CVTRIW_R8	********	X	01				
COB$CVTRPQ_R9	********	X	01				
COB$CVTRQP_R9	********	X	01				
COB$CVTTI_R8	********	X	01				
COB$CVTWI_R8	********	X	01				
COB$DBEXCEPTION	********	X	01				
COB$DISPLAY	********	X	01				
COB$DISP_NO_ADV	********	X	01				
COB$DISP_SCR	********	X	01				
COB$DISP_SCR_NO_ADV	********	X	01				
COB$DIVI	********	X	01				
COB$DIVI_OSE	********	X	01				
COB$DIVQ_R8	********	X	01				
COB$ERROR	********	X	01				
COB$EXPI	********	X	01				
COB$EXPI_OSE	********	X	01				
COB$HANDLER	00000000	RG	01				
COB$INIT_LINAGE	********	X	01				
COB$IOEXCEPTION	********	X	01				
COB$LINAGE	********	X	01				
COB$MULI	********	X	01				
COB$MULQ_R8	********	X	01				
COB$PAUSE	********	X	01				
COB$POS_ACCEPT	********	X	01				
COB$POS_DISPLAY	********	X	01				

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
COBVECTOR	000001F0 (496.)	01 (1.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	40	00:00:00.09	00:00:01.95
Command processing	134	00:00:00.33	00:00:02.23
Pass 1	101	00:00:01.05	00:00:05.40
Symbol table sort	0	00:00:00.03	00:00:00.03
Pass 2	70	00:00:00.50	00:00:02.35
Symbol table output	9	00:00:00.04	00:00:00.04
Psect synopsis output	2	00:00:00.01	00:00:00.01
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	358	00:00:02.05	00:00:12.04

The working set limit was 900 pages.
7783 bytes (16 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 62 non-local and 0 local symbols.
314 source lines were read in Pass 1, producing 19 object records in Pass 2.
4 pages of virtual memory were used to define 4 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/LIS=LIS$:COBVECTOR/OBJ=OBJ$:COBVECTOR MSRC$:COBVECTOR/UPDATE=(ENH$:COBVECTOR)

COBRMSBLO
LIS

COBSPANC
LIS

COBSWITCH
LIS

COBRESTVA
LIS

CONVSHR
MAP

CONVCOMIO
LIS

COBSETSWI
LIS

CONVDEF
MDL

CONVLINK
REQ

COBSPANC2
LIS

CONV

CONVERT
MAP

CONVCALL
LIS

COBVECTOR
LIS

CONVERT
REQ

RECLAIM
MAP

ADDKEY
LIS