```cobol
       Identification Division.
       Program-ID.    J7200521.
*      *
*      * Copyright Wisconsin Department of Transportation
*      *
*      * Permission is hereby granted, free of charge, to any person or
*      * organisation to use this software and its associated files
*      * subject to the following conditions:
*      *
*      * The software may be redistributed free of charge to any other
*      * person or organisation provided that the above copyright
*      * notice, this permission notice and the disclaimer shall be
*      * included with all copies of the Software.
*      *
*      * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF
*      * ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED
*      * TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
*      * PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT
*      * SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR
*      * ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
*      * ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
*      * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
*      * OTHER DEALINGS IN THE SOFTWARE. IF THIS DISCLAIMER
*      * CONTRADICTS LOCALLY APPLICABLE LAW THEN USE OF THIS
*      * SOFTWARE IS PROHIBITED.
*      *
       Author.        Craig Schneiderwent.
*
* Encapsulate the CICS WRITE OPERATOR function.  Caller must
* specify a message number, which will be prefixed with 'UDOT'
* and a message type of 'I' (Informational), 'W' (Warning),
* 'E' (Error), or 'S' (Severe).
*
* Good practice is to document the message, currently the place
* to do that is in MVS/QuickRef.  These messages will be used
* by on-call and operations staff in problem resolution
* situations - do your best to help them out with meaningful
* diagnostics in the message and good documentation of the
* message.
*
* In order to avoid issues with the WRITE OPERATOR CICS API
* not being threadsafe, it is encapsulated here in its own
* program object and we LINK to it instead of dynamically
* CALLing it.
*
* The CICS Programming Guide has a good explanation of
```

```
* threadsafe and its implications.
*
*
* Sample Use:
*
*Working-Storage Section.
*01  CONSTANTS.
*    05  MYNAME                 PIC X(008) VALUE 'J8675309'.
*    05  CICS-ERR-TXT-PGM       PIC X(008) VALUE 'J7200501'.
*    05  CICS-WTO-PGM           PIC X(008) VALUE 'J7200521'.
*    05  MSG-NB                 PIC X(004) VALUE '0005'.
*
*01  WORK-AREAS.
*    05  CICS-RESP              PIC S9(008) COMP-5.
*    05  CICS-RESP2             PIC S9(008) COMP-5.
*    05  WTO-CA-LN              PIC 9(008)  COMP-5.
*    05  DATA-TXT               PIC X(025)
*        VALUE 'Test message'.
*
*01  WTO-CA.
*    COPY J7200521 REPLACING ==:PRFX:== BY ==WTO-==.
*
*Procedure Division.
*    MOVE LENGTH OF DATA-TXT TO WTO-TXT-LN
*    MOVE DATA-TXT TO WTO-TXT
*    MOVE '0005' TO WTO-MSG-NB
*    SET WTO-MSG-TY-ERR TO TRUE
*    MOVE MYNAME TO WTO-CALLER
*    MOVE LENGTH OF WTO-CA TO WTO-CA-LN
*    EXEC CICS LINK
*        PROGRAM(CICS-WTO-PGM)
*        COMMAREA(WTO-CA)
*        LENGTH(WTO-CA-LN)
*        RESP(CICS-RESP)
*        RESP2(CICS-RESP2)
*    END-EXEC
*
*    IF WTO-RC-NORMAL AND CICS-RESP = DFHRESP(NORMAL)
*        CONTINUE
*    ELSE
*        ...error handling
*    END-IF
*
* Return Codes:
* 0  - Success
* 4  - Invalid parameter(s)
```

```cobol
*       An error message will be DISPLAYed
* 16 - Failure
*
*
 Environment Division.
 Configuration Section.
 Data Division.
 Working-Storage Section.
 01  CONSTANTS.
     05  MYNAME                PIC X(008) VALUE 'J7200521'.
     05  CICS-ERR-PGM          PIC X(008) VALUE 'J7200501'.
     05  ERR-MSG-PRFX          PIC X(004) VALUE 'UDOT'.

 01  WORK-AREAS.
     05  ERR-MSG-LN            PIC 9(008) COMP-5 VALUE 1.
     05  CICS-APPL-ID          PIC X(008) VALUE SPACES.
     05  THIS-USERID           PIC X(008) VALUE SPACES.
     05  ERR-MSG-WTO.
         10                    PIC X(050) OCCURS 13.
         10                    PIC X(040).
     05  CICS-API-FAILED       PIC X(030) VALUE SPACES.
     05  CICS-API-FAILED-LOC   PIC X(004) VALUE SPACES.
     05  CICS-RESP-TXT         PIC X(025) VALUE SPACES.
     05  CICS-RESP-DSPL        PIC 9(010) VALUE ZEROES.
     05  CICS-RESP-DSPL-X
         REDEFINES
         CICS-RESP-DSPL        PIC X(010).
     05  CICS-RESP2-DSPL       PIC 9(010) VALUE ZEROES.
     05  CICS-RESP2-DSPL-X
         REDEFINES
         CICS-RESP2-DSPL       PIC X(010).


 Linkage Section.
 01  DFHCOMMAREA.
     COPY J7200521 REPLACING ==:PRFX:== BY ==LS-==.

 Procedure Division Using
     DFHCOMMAREA

     .


     SET LS-RC-NORMAL TO TRUE
     INITIALIZE ERR-MSG-WTO
     PERFORM 0100-EDIT-PARMS
```

```
            IF LS-RC-NORMAL
                CONTINUE
            ELSE
                EXEC CICS RETURN END-EXEC
            END-IF

            EXEC CICS ASSIGN
                APPLID(CICS-APPL-ID)
                USERID(THIS-USERID)
                NOHANDLE
            END-EXEC

            MOVE EIBRESP  TO LS-RESP
            MOVE EIBRESP2 TO LS-RESP2

            IF EIBRESP = DFHRESP( NORMAL )
                PERFORM 9040-FMT-MSG-01
                PERFORM 9100-WTO
            ELSE
*           The documentation for CICS TS 2.2 doesn't list any
*           way for the above CICS ASSIGN call to fail, but this
*           routine is likely to be called as an application is
*           failing.  Hence the "belt _and_ suspenders" technique.
                MOVE 'ASSIGN' TO CICS-API-FAILED
                MOVE 'MAIN'   TO CICS-API-FAILED-LOC
                PERFORM 9200-DSPL-CICS-ERR
                SET LS-RC-BAD TO TRUE
            END-IF

            EXEC CICS RETURN END-EXEC
            .

     0100-EDIT-PARMS.
            IF LS-MSG-TY-VALID
                CONTINUE
            ELSE
                DISPLAY
                    MYNAME
                    ' message type ('
                    LS-MSG-TY
                    ') is invalid.  Valid values are I W E S.'
                SET LS-RC-INVD-PARM TO TRUE
            END-IF

            IF LS-TXT-LN-VALID
                CONTINUE
```

```
          ELSE
              DISPLAY
                  MYNAME
                  ' message length ('
                  LS-TXT-LN
                  ') is invalid.  0 < length <= 690.'
              SET LS-RC-INVD-PARM TO TRUE
          END-IF
          .

  9040-FMT-MSG-01.

      MOVE SPACES TO ERR-MSG-WTO
      MOVE +1     TO ERR-MSG-LN
      STRING
          ERR-MSG-PRFX
            DELIMITED BY SIZE
          LS-MSG-NB
            DELIMITED BY SIZE
          LS-MSG-TY
            DELIMITED BY SIZE
          SPACE
            DELIMITED BY SIZE
          CICS-APPL-ID
            DELIMITED BY SIZE
          SPACE
            DELIMITED BY SIZE
          EIBTRNID
            DELIMITED BY SIZE
          SPACE
            DELIMITED BY SIZE
          LS-CALLER
            DELIMITED BY SIZE
          SPACE
            DELIMITED BY SIZE
          THIS-USERID
            DELIMITED BY SIZE
          SPACE
            DELIMITED BY SIZE
          LS-TXT(1:LS-TXT-LN)
            DELIMITED BY SIZE
        INTO
          ERR-MSG-WTO
        WITH POINTER
          ERR-MSG-LN
      END-STRING
```

```cobol
            .

 9100-WTO.
*
*    The NOHANDLE is here because if an error occurs during
*    error processing, an abend is too likely to create an
*    abend loop.
*
     EXEC CICS
         WRITE OPERATOR
          TEXT(ERR-MSG-WTO)
          TEXTLENGTH(ERR-MSG-LN)
          NOHANDLE
     END-EXEC

     MOVE EIBRESP  TO LS-RESP
     MOVE EIBRESP2 TO LS-RESP2
     IF EIBRESP = DFHRESP( NORMAL )
         CONTINUE
     ELSE
         MOVE 'WRITE OPERATOR' TO CICS-API-FAILED
         MOVE '9100'           TO CICS-API-FAILED-LOC
         SET LS-RC-BAD TO TRUE
         PERFORM 9200-DSPL-CICS-ERR
     END-IF
          .

 9200-DSPL-CICS-ERR.
*
* This is really a last ditch effort at error determination.  If
* the ASSIGN or WRITE OPERATOR calls fail, this paragraph will
* provide at least some indication of the problem.
*
     CALL CICS-ERR-PGM USING
         LS-RESP
         CICS-RESP-TXT
     END-CALL
     MOVE LS-RESP    TO CICS-RESP-DSPL
     MOVE LS-RESP2   TO CICS-RESP2-DSPL

     DISPLAY
         MYNAME
         SPACE
         CICS-API-FAILED
         ' FAILED AT '
         CICS-API-FAILED-LOC
```

```
SPACE
' RESP = '
CICS-RESP-DSPL
' RESP TEXT = '
CICS-RESP-TXT
' RESP2 = '
CICS-RESP2-DSPL
' CALLER = '
LS-CALLER
' MESSAGE # = '
LS-MSG-NB
' MESSAGE TYPE = '
LS-MSG-TY
' WTO TEXT LENGTH = '
LS-TXT-LN
' WTO TEXT = '
LS-TXT(1:LS-TXT-LN)
.
```