

**PUBLICATIONS
UPDATE**

Operating System/3 (OS/3)

Data Base Management
System (DMS)
Data Manipulation
Language

User Guide/
Programmer Reference

UP-8036 Rev. 4-B

This Library Memo announces the release and availability of Updating Package B to "SPERRY UNIVAC Operating System/3 (OS/3) Data Base Management System (DMS) Data Manipulation Language User Guide/ Programmer Reference", UP-8036 Rev. 4.

This update for OS/3 release 8.0 adds new rollback error codes and changes the text and description of several DML preprocessor diagnostic messages.

Other changes in this update are applicable to the software prior to the current release. These include a description of partially deleted, or "DEFERRED-DELETED" record occurrences, and additional DML preprocessor diagnostic messages.

Copies of Updating Package B are now available for requisitioning. Either the updating package only or the complete manual with the updating package may be requisitioned by your local Sperry Univac representative. To receive only the updating package, order UP-8036 Rev. 4-B. To receive the complete manual, order UP-8036 Rev. 4.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS
<p>Mailing Lists BZ, CZ and MZ</p>	<p>Mailing Lists A00, A09, B00, B09, 18, 18U, 19, 19U, 20, 20U, 21, 21U, 28U, 29U, 75, 75U, 76, and 76U (Package B to UP-8036 Rev. 4, 18 pages plus Memo)</p>	<p>Library Memo for UP-8036 Rev. 4-B</p> <hr/> <p>RELEASE DATE: September, 1982</p>



PAGE STATUS SUMMARY

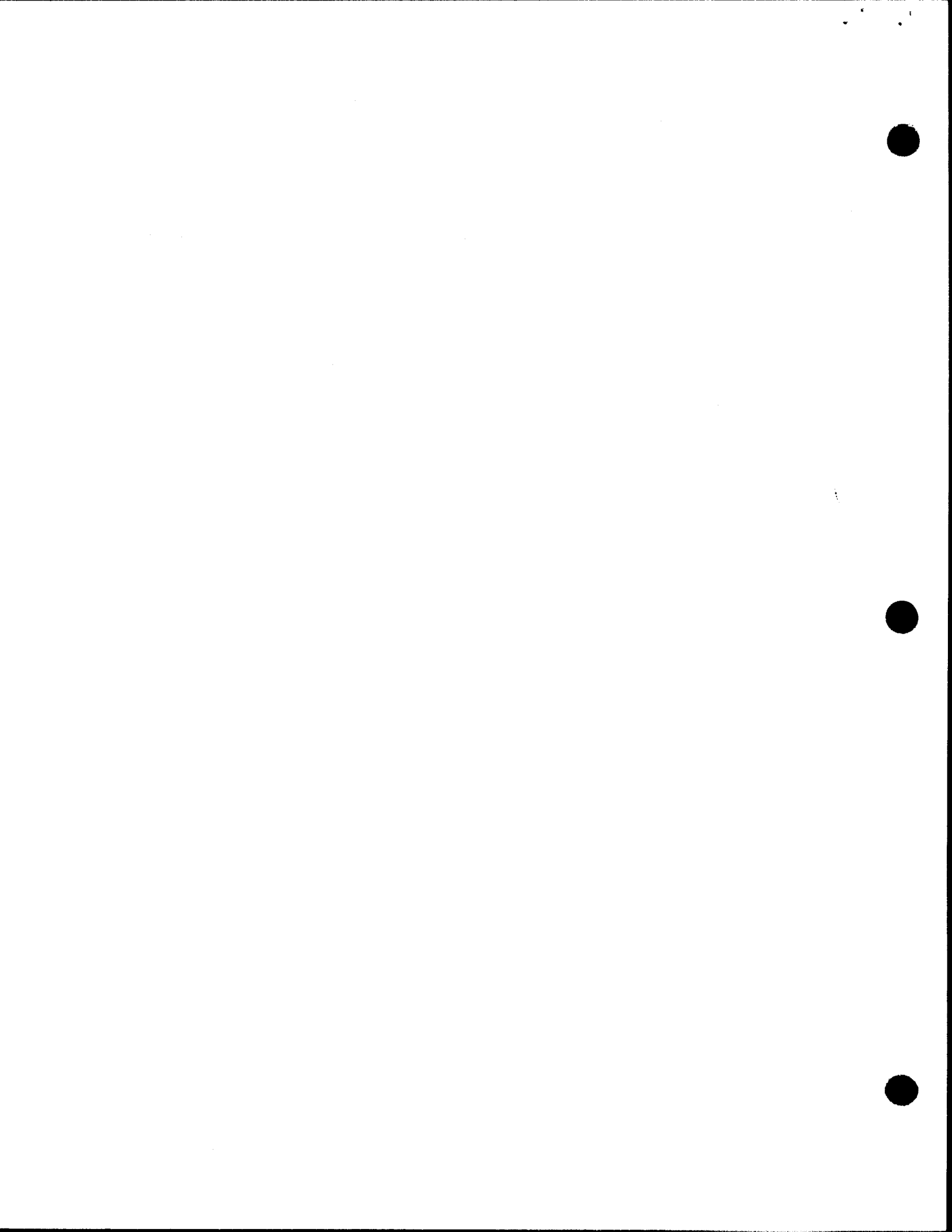
ISSUE: Update B – UP-8036 Rev. 4

RELEASE LEVEL: 8.0 Forward

Part/Section	Page Number	Update Level	Part/Section	Page Number	Update Level	Part/Section	Page Number	Update Level
Cover/Disclaimer		Orig.						
PSS	1	B						
Acknowledgment	1	Orig.						
Preface	1	Orig.						
Contents	1 thru 5	Orig.						
1	1 thru 5	Orig.						
2	1 thru 18 18a 19 thru 25	Orig. Orig. Orig.						
3	1 thru 27 28 thru 30 30a 31 thru 44 45 46 thru 68	Orig. B B* Orig. A Orig.						
4	1 thru 25	Orig.						
5	1 thru 4	Orig.						
Appendix A	1 thru 3	Orig.						
Appendix B	1 thru 17	Orig.						
Appendix C	1 thru 10 11 thru 13	Orig. B						
Appendix D	1 2 thru 7	Orig. B						
Index	1 thru 9	Orig.						
User Comment Sheet								

* New Pages

All the technical changes are denoted by an arrow (→) in the margin. A downward pointing arrow (↓) next to a line indicates that technical changes begin at this line and continue until an upward pointing arrow (↑) is found. A horizontal arrow (→) pointing to a line indicates a technical change in only that line. A horizontal arrow located between two consecutive lines indicates technical changes in both lines or deletions.



Examples:■ **Format 1**

Assuming the end of area or set has been reached, all data manipulations have been performed, and you are ready to end the job, the CLOSE statement number 024200 closes all areas.

1	8	12
024200	CLOSE	ALL.
024300	CLOSE AREA	CUSTOMER-AREA, ORDER-AREA, PRODUCT-AREA.

■ **Format 2**

Statement number 024300 closes only those areas you specify (up to 37 areas).

3.5.3. DELETE Statement**Function:**

The DELETE statement:

- makes the object record occurrence unavailable for further processing (the object record is the record being acted upon by the DELETE verb);
- removes the object record from all set occurrences of the particular subschema in which it participates as a member;
- deletes all record occurrences that are mandatory members of set occurrences owned by the object record;
- removes or optionally deletes all record occurrences that are optional members of set occurrences owned by the object record; and
- optionally prevents deletion of the object record if the data base contains any nonempty set occurrences of which the object record is the owner.

Format:

```
DELETE { record-name } [ ALL
      { identifier-1 } [ ONLY
                      [ SELECTIVE ] ] .
```

Rules:

1. When you use record-name, it must refer to a record that was included in the invoked subschema. When you use identifier-1, it must have already been established as a generalized name entry with a USAGE IS RECORD-NAME clause, and initialized with the name of a record included in the subschema invoked by the program.

The DMLP assumes the word DELETE to be a DML verb when:

- a. the program is preprocessed for American National Standard 68 level COBOL; or
- b. ALL, ONLY, or SELECTIVE appears correctly in the DELETE statement; or
- c. identifier-1 is a valid generalized name, or record-name is found in the subschema.

When none of these conditions are true, the DMLP assumes that DELETE is a COBOL statement and no translation is done.

NOTE:

A misspelled generalized name or subschema record name isn't detected until compilation by the COBOL compiler.

2. The object record occurrence of the DELETE statement must be the current record of the run unit. If the current record of run unit is not an occurrence of record-name, an error condition exists.
3. The area from which the record is deleted, as well as any areas indirectly updated by the deletion, must be open for an update usage mode before the DELETE statement can be executed.
4. The object record is removed from all set occurrences of the particular subschema in which it participates as a member, and it is then deleted, that is, made unavailable for further processing by any DML statement. The space and the data base key used by the object record are made available for reuse.

NOTE:

A distinction is made between a removed record occurrence and a deleted record occurrence. Deletion is a 2-step process that first cancels (removes) the existing membership of the object record in specific set occurrences and then releases the space occupied by the object record for reuse. The space formerly used by the record occurrence and its assigned data base key may be used subsequently to store a different record type.

- ↓
5. In some cases, a deleted record occurrence is only partially deleted, or *deferred-deleted*, from the data base. If the occurrence does not have prior pointers in all sets in which it participates, efficient delinking and removal are not always possible. In this case, the data portion of the occurrence is deleted, but the set pointers and line-entry associated with the occurrence are maintained until all information is available for an efficient deletion. The maintained line-entry is marked (with X'80' in the most significant bit field) to indicate a deferred-deleted occurrence.

A deferred-deleted occurrence can eventually be delinked from a set if a trace through the set is made in update mode. This may occur on a retrieval operation (FIND/FETCH) if a deferred-deleted occurrence is found in an area opened for update. If the occurrence can be efficiently delinked from all sets, the final delete is performed.

↑

6. The unqualified form of the DELETE statement

$$\underline{\text{DELETE}} \left\{ \begin{array}{l} \text{record-name} \\ \text{identifier-1} \end{array} \right\} .$$

deletes the object record only if it has no member records. If the object record is the owner of a nonempty set occurrence, the DELETE statement is not successfully executed and an error condition exists. An error condition can be avoided by use of format 1 of the IF statement (3.5.8) before execution of the unqualified DELETE statement.

7. The DELETE ONLY form of the statement deletes the object record and all of its mandatory members. It removes, but does not delete, its optional members.

If any of the deleted mandatory members are themselves the owners of any set occurrences, then the DELETE statement is executed on such records as if they were the object record of a DELETE ONLY statement. Thus, all mandatory members of such sets are also deleted, and in turn this process continues down the structure until a record occurrence participates only as an optional member in a set.

8. The DELETE SELECTIVE form of the statement has the same results as the DELETE ONLY statement, with the following exceptions:

- optional members are deleted only if they do not currently participate as members in other set occurrences; and
- all deleted records that are the owners of any set occurrences are treated as if they were the record-name of a DELETE SELECTIVE statement.

9. The DELETE ALL form of the statement deletes the object record and all its member records, both mandatory and optional. As with the DELETE ONLY form of the statement, the delete process continues down the hierarchy, with the difference that all deleted records that are the owners of any set occurrences are treated as if they were the object record of a DELETE ALL statement.

10. Following successful execution of a DELETE statement, the current record of the run unit becomes null. A null value is indicated by a minus 1 in the DBKEY system status location. All other currency information remains unchanged. Retrieving deleted records by using FIND format 2 results in a value of 0317 being placed in ERROR-STATUS. Retrieving deleted records by using other formats results in a value of 0326 being placed in ERROR-STATUS.

11. The DELETE statement is not executed for any record that is:

- an owner or member of any set that has not been specified as part of the subschema;
- a member of a set that has one or more members not specified as part of the subschema;

- an owner of a set that has one or more members not specified as part of the sub-schema; or
- an owner of a set whose members cannot be deleted.

NOTE:

The third and fourth restrictions don't apply to unqualified DELETE statements.

12. When an error occurs, the ERROR-STATUS location contains a nonzero value, which is explained in Appendix C. In addition, the data base and working storage remain in the state existing prior to the attempted execution of the DELETE statement.

Example 1:

This example uses the SPSUBS subschema in Figure 4-1. Assume that an occurrence of a CUST-ORDER record is current of run unit.

```

1      8      12
-----
602020  DELETE CUST-ORDER ALL.

```

Remember that the DELETE ALL form of the statement deletes the object record and all its member records, mandatory and optional. Consequently, executing statement number 602020:

- Deletes all occurrences of the ORD-REMARK record in the current SPEC-REMARK set occurrence.
- Removes each ORDER-ITEM record occurrence from the related PROD-ORD set occurrence.
- Deletes all ORDER-ITEM record occurrences in the current ITEM set occurrence.
- Removes the CUST-ORDER record from the ORDOR set.
- Deletes the CUST-ORDER record.

Example 2:

Referring to the data structure in Figure 4-3, assume that an occurrence of an MFG-LOT record is current of run unit.

```

609120  DELETE MFG-LOT ONLY.

```

Remember that the DELETE ONLY form of the statement deletes the object record and all its mandatory members. It removes, but does not delete, its optional members. Consequently, executing statement number 609120:

- Removes every occurrence of the LOT-INV record that participates in the current LOT-OH set occurrence from the LOT-OH set.

- Removes the MFG-LOT record from the current occurrence of the MFG-LOT-DET set (provided the current MFG-LOT is a member in that set).
- Deletes the MFG-LOT record.

The effect of these operations is to delete the owner of the LOT-OH set (MFG-LOT record) without also deleting the LOT-INV member records.



Table C-5. Rollback Error Codes (Part 1 of 3)

Rollback Code (first or last two bytes of RB-ERROR-CODE)*	Explanation
01	DMS internal error
02	Data base write error
03	Data base read error
04	Program check or unrecoverable error (in DBMS)
05	I/O error in concurrent run unit
06	Run unit cancelled (while in DBMS)
07	Journal file write error
08	Program termination before depart
09	Invalid object subschema
10	Main storage not available
11	Space inventory write error
12	Space inventory read error
13	Invalid verb number
14	Area unavailable for update
15	Record update conflict
16	Not used
17	Not used
18	Resource wait time exceeded
19	Deadlock avoidance
20	Delete hierarchy error
21	QBL file allocation error
22	Maximum updating run units exceeded for DMCL
23	Rollback failure - QBL file was shut down.
24	QBL file read error
25	QBL record error
26	Not used
27	Not used
28	Not used
29	Requested QBL write rejected; IMS QBL file shut down



Table C-5. Rollback Error Codes (Part 2 of 3)

Rollback Code (first or last two bytes of RB-ERROR-CODE)*	Explanation
30	Requested DMCL not loaded
31	Requested subschema not found
32	Insufficient main storage to load subschema
33	Invalid subschema
34	Invalid DBA name for DMCL
35	Invalid DBA name for subschema
36	Invalid schema name for DMCL
37	Invalid schema name for subschema
38	Different DMCL not allowed on BIND
39	Subschema compilation date/time mismatch
40	IMS terminal limit exceeded
41	Not used
42	Not used
43	Data base is shut down.
44	Not used
45	Area is shut down. Possible causes are: <ul style="list-style-type: none"> <li data-bbox="553 1346 915 1373">■ DBA shut down an area; or <li data-bbox="553 1398 1143 1425">■ data base file had read/write or open/close error.
46	Application is shut down.
47	Journal file is shut down. Journal file had open/close or read/write error under another task.
48	File access time-out
49	Not used
50	Not used
51	New impart is inhibited for DBMS.
52	New impart is inhibited within a DBA system.
53	New impart is inhibited for a data base.

Table C—5. Rollback Error Codes (Part 3 of 3)

Rollback Code (first or last two bytes of RB-ERROR-CODE)*	Explanation
54	Not used
55	New OPEN is inhibited for an area.
56	Maximum run units exceeded for DBMS
57	Maximum run units exceeded for DMCL
58	IMS access inhibited to DMCL for this session
59	IMS access inhibited to DBMS for this session
67	IMS QBL file write rejected
68	IMS QBL file space exhausted
88	Unidentified DMS error
97	DMS SVC processing error
99	First DBMS call not imparted

* RB-ERROR-CODE is a 4-byte field. The last two bytes indicate the cause of the error. The first two bytes give the status of the rollback. If the rollback is successful, the first two bytes contain zeros. If the rollback is unsuccessful, the first two bytes indicate the cause of the rollback.



Appendix D. DML Preprocessor Messages

D.1. INFORMATIONAL MESSAGES

The messages in Table D-1 are displayed either during a DML preprocessor run or after a run is completed successfully or terminated abnormally.

Table D-1. DML Preprocessor Informational Messages

Message Text	Explanation
DML ERROR(S) DETECTED DURING PREPROCESSING nnINVALID DML STATEMENTS DETECTED nn WARNING MESSAGES ISSUED	Invalid DML statements have been detected during DML processing. Any attempt at a COBOL compilation would be meaningless.
DML PREPROCESSING INITIATED	DML processing has started.
DML PREPROCESSING COMPLETED WITHOUT ERRORS	DML processing has been completed successfully.
END OF FILE REACHED IN DATA DIVISION	DMLP has encountered an end-of-file while reading the input file, and neither a procedure division nor a definition division (IMS only) header has been encountered.
INVALID ID DIVISION OR COBRUN STATEMENT	Identification division statement missing or syntactically incorrect. DML processing terminates.
QUOTE=DOUBLE IS APPLIED TO BOUND NUMERIC LITERALS	Nonnumeric literals generated by the DML preprocessor are bound by double quotes. Users may specify this explicitly by using the QUOTES ARE DOUBLE option of the DUPL syntax. QUOTES ARE DOUBLE is the default option.
QUOTE=SINGLE IS APPLIED TO BOUND NUMERIC LITERALS	Nonnumeric literals generated by the DML preprocessor are bound by single quotes. Users must specify this explicitly by using the QUOTES ARE SINGLE option of the DUPL syntax.
SCHEMA/INVOKE STATEMENT MISSING OR INVALID	Schema section or INVOKE statement is syntactically invalid. DML processing is aborted.

D.2. DIAGNOSTIC MESSAGES

The messages in Table D-2 appear in the DML preprocessor output listing. The diagnostic severity codes are:

W (Warning)

The specified DML statement must be reexamined; however, the COBOL CALL statement is generated appropriately.

NOTE:

The UPSI byte is set to X '20'.

E (Severe)

The specified DML statement is invalid either syntactically or semantically. The COBOL CALL statement is not generated, but the specified DML statement is converted to a COBOL comment statement.

NOTE:

The UPSI byte is set to X '40'.

F (Fatal)

A fatal error has occurred; a dump is given. This type of error should be reported to the data base administrator.

NOTE:

The UPSI byte is set to X '80'.

Table D-2. DML Preprocessor Diagnostic Messages (Part 1 of 6)

Severity Code	Message Text	Explanation	Recovery Action
E	AREA-NAME area-name NOT DEFINED IN sub- schema-name SUB-SCHEMA	The specified area-name is not included in the invoked subschema.	Check the source documentation of the subschema.
F	COPYING LINKAGE AND NO LINKAGE HEADER	The linkage section header is required if linkage or linkage common is specified in the INVOKE statement.	Correct and rerun.
F	DATA DIVISION HEADER MISSING	The source program does not contain a data division.	Correct and rerun.
W	DEFAULT DMCL CONTAINS ERRORS	The default DMCL in the data dictionary contains errors.	Check with the data base administrator.
W	DEPART STATEMENT NOT PRESENT IN THE PROGRAM	DMLP has not found a DEPART statement. This is acceptable only for a subprogram.	Check the DML/COBOL program.

Table D-2. DML Preprocessor Diagnostic Messages (Part 2 of 6)

Severity Code	Message Text	Explanation	Recovery Action
E	DML STATEMENT SIZE INVALID	The DML statement contains too many operands or is missing its terminating period.	Check for valid format and rerun.
W	FREE WITH CHECKPOINT ASSUMED	Three formats of the FREE verb exist. FREE WITH CHECKPOINT is the default.	Check statement syntax format.
F	FILE DEFINITION MISSING FOR SUB-SCHEMA INVOKED. DML PRE-PROCESSOR CANNOT CONTINUE.	01-level record descriptions of the subschema invoked could not be found in the data base.	Check with the data base administrator.
E	IDENTIFIER NAME IS NOT UNIQUE	The data item name has already been used for a previous DML identifier statement.	Make the name unique and rerun.
E	IDENTIFIER NAME IS GREATER THAN 16 CHARACTERS	The data item name is too long.	Shorten the name to 16 or fewer characters and rerun.
W	IMPART OR BIND STATEMENT NOT FOUND IN PROGRAM	DMLP has not found an IMPART or BIND statement. This is acceptable only for a sub-program.	Check the DML/COBOL program.
E	INVALID BIND STATEMENT	DML statement is syntactically incorrect.	Check the statement syntax format.
E	INVALID CLOSE STATEMENT		
E	INVALID DELETE STATEMENT		
	INVALID DEPART STATEMENT		
E	INVALID DML SYNTAX	The DML statement is syntactically invalid.	Check for valid format and resubmit.
E	INVALID DML SYNTAX. DUPLICATES NOT ALLOWED FOR record-name RECORD	For the DML format, FIND NEXT DUPLICATE record-name RECORD. The record name is not allowed to have the same contents of the CALC key data item as the record type previously stored in the data base.	Check the source documentation of the schema invoked.
E	INVALID DML SYNTAX. RECORD NAME record-name IS NOT MEMBER OF CALC SET	This is pertinent to the following DML statements: 1. FIND record-name RECORD. 2. FIND NEXT DUPLICATE record-name RECORD. The location mode of the record-name is not CALC mode.	Check the source documentation of the schema invoked to determine the location mode.
E	INVALID DML SYNTAX SET NAME set-name IS NOT LAST PROCESSABLE	The set specified in FIND LAST [record-name] RECORD OF set-name SET does not have a prior pointer.	Check the source documentation of the schema invoked to determine what pointers are in the specified set.

Table D-2. DML Preprocessor Diagnostic Messages (Part 3 of 6)

Severity Code	Message Text	Explanation	Recovery Action
E	INVALID DML SYNTAX SET NAME set-name IS NOT PRIOR PROCESSABLE	This is pertinent to the following DML statements: 1. FIND PRIOR record-name RECORD OF set-name SET. 2. FIND PRIOR RECORD OF set-name SET. The specified set does not have a prior pointer.	Check the source documentation of the schema invoked to determine what pointers are in the specified set.
F	INVALID DMS-STATUS SOURCE FILE	The file containing DMS-STATUS source could not be read.	Check file. Check with the data base administrator.
E	INVALID FIND/FETCH STATEMENT	DML statement is syntactically incorrect.	Check the statement syntax format.
E	INVALID FREE STATEMENT		
E	INVALID GET STATEMENT		
E	INVALID IDENTIFIER STATEMENT SYNTAX	Possible causes are: 1. Additional words appeared in the statement. 2. An invalid level number was assigned or an invalid margin for the level number was detected.	1. Verify that a period terminates the statement. 2. Ensure that the level number field is valid.
E	INVALID IF STATEMENT	DML statement is syntactically incorrect.	Check the statement syntax format.
E	INVALID IMPART STATEMENT		
E	INVALID INSERT STATEMENT		
E	INVALID KEEP STATEMENT		
E	INVALID MODIFY STATEMENT		
E	INVALID MOVE STATEMENT		
E	INVALID OPEN STATEMENT		
E	INVALID REMOVE STATEMENT		
E	INVALID ROLLBACK STATEMENT		
E	INVALID STORE STATEMENT		
E	INVALID QUALIFICATION OF IDENTIFIER identifier-name	A DML identifier is the object of data name qualification.	Remove the applicable qualifiers or correct the discrepant identifier-name.
F	INVALID SUB-SCHEMA COPY KEY	The DUPL input did not contain the proper authorization key for the defined subschema.	Check with the data base administrator.
E	INVALID UNBIND STATEMENT	DML statement is syntactically incorrect.	Check the statement syntax format.

Table D-2. DML Preprocessor Diagnostic Messages (Part 4 of 6)

Severity Code	Message Text	Explanation	Recovery Action
E	INVOKE/SCHEMA SECTION MISSING OR SYNTACTICALLY INVALID	Schema section or INVOKE statement is missing or syntactically invalid. DML processing stops at this point.	Check the COBOL/DML program.
E	INVOKED DMCL dmcl-name CONTAINS ERRORS.	The DMCL in the data dictionary contains errors.	Check with the data base administrator.
W	KEYWORD 'RECORD' MISSING	The work 'RECORD' is part of the syntax from a previous DMS release.	Check the statement syntax format.
F	LEVEL NUMBER GREATER THAN 49	Level numbers greater than 49 are not allowed if working-common or linkage-common is specified in the INVOKE statement.	Correct and rerun.
E	MAXIMUM NUMBER OF IDENTIFIER NAMES EXCEEDED	Only 100 identifier names may be defined within any given DML program.	Correct and rerun.
W	NO DEFAULT DMCL MODULE HAS BEEN SPECIFIED	There is no default DMCL defined for the schema. Implicit initialization of DMCL-NAME location in DMCA by DMLP cannot be done.	If the correct DMCL name is not known to the user, check with the data base administrator.
E	NO DMCL MODULE HAS BEEN DEFINED FOR THIS DATA BASE	DMLP cannot find any DMCL defined for the schema within the data dictionary. DMLP cannot do implicit initialization of DMCL-NAME location in DMCA.	Check with the data base administrator.
F	NO DMS-STATUS SECTION COPIED	COBOL images of the DMS-STATUS section could not be copied because the file assigned was empty, or the rollback section/paragraph was not placed last.	Check with the data base administrator.
W	NO SCHEMA CROSS-REFERENCE RECORD	DMLP could not find a schema cross-reference record.	Check with the data base administrator.
W	NO SUBSCHEMA CROSS-REFERENCE RECORD	DMLP could not find a subschema cross-reference record.	Check with the data base administrator.
W	OLD FORMAT OF INVOKE STATEMENT DETECTED	Conversion code has been included to upgrade this program from single-thread operation.	1. Review the OPEN ALL/ CLOSE ALL statement expansions to determine adequacy of the conversion aids; or 2. Upgrade the INVOKE statement to the current format and rerun.
F	PROGRAM ID NOT FOUND	The program ID statement is required.	Correct and rerun.

Table D-2. DML Preprocessor Diagnostic Messages (Part 5 of 6)

Severity Code	Message Text	Explanation	Recovery Action
W	QUOTE=SINGLE/DOUBLE NO LONGER SUPPORTED	The user now uses the DUPL syntax to indicate choice of quotes.	Correct when convenient.
W	QUOTE PARAM DOES NOT MATCH DUPL COMMAND	The DUPL command overrides the no-longer supported QUOTE parameter.	Correct if necessary when convenient.
E	RECORD-NAME record-name IS NOT A MEMBER OF area-name AREA	The specified record-name is not included in the named area.	Check the source documentation of the subschema.
E	RECORD-NAME record-name IS NOT A MEMBER OF set-name SET	The specified record-name is not included in the named set.	Check the source documentation of the subschema.
E	RECORD-NAME record-name NOT DEFINED IN subschema- name SUB-SCHEMA	The specified record-name is not included in the invoked subschema.	Check the source documentation of the subschema.
W	RECORD/SET/DB-IDENTIFIER ASSUMED CONSISTENT	DMLP cannot validate that the specified data-base-identifier is correct with respect to the target record and set (i.e., because the record and/or set names are unknown).	Refer to rule 27 of the FIND/FETCH statement (format 6) for a description of this precautionary warning (3.5.5).
W	ROLLBACK PARAGRAPH MISSING FROM INVOKE STANDARD ONE ASSUMED	The user did not include ROLLBACK clause in the INVOKE.	None.
F	SCHEMA NAME schema-name HAS NOT BEEN DEFINED VIA DDL PROCESSOR. THE DML PRE-PROCESSOR CANNOT CONTINUE	<ol style="list-style-type: none"> 1. The schema has not been compiled; the data dictionary is in an initialized state. 2. The spelling of the schema-name is incorrect. 	<ol style="list-style-type: none"> 1. The DBA must recompile the schema and subschema. 2. Check with the DBA for correct spelling.
F	SECTION HEADER INVALID FOR PROGRAM TYPE	A section header has been encountered that conflicts with program type given in DUPL syntax.	Correct and rerun.
E	SET-NAME set-name NOT DEFINED AS SORTED SET	Set named in format 6 of FIND/FETCH statement is not sorted.	Check with the data base administrator.
E	SET-NAME set-name NOT DEFINED IN subschema-name SUB-SCHEMA	The specified set-name is not included in the invoked subschema.	Check the source documentation of the subschema.
F	SRNAME, RECORD-NAME EQUIVALENCE MISSING. DML PRE-PROCESSOR CANNOT CONTINUE	01-level record of record names, set names, or area names of the invoked subschema could not be found in the data base.	Check with the data base administrator.
F	SUB-SCHEMA NAME subschema- name HAS NOT BEEN DEFINED VIA SUBSCHEMA PROCESSOR. THE DML PRE-PROCESSOR CANNOT CONTINUE	The subschema invoked has not been compiled, or the spelling of the subschema name is incorrect.	Check with the data base administrator.

Table D-2. DML Preprocessor Diagnostic Messages (Part 6 of 6)

Severity Code	Message Text	Explanation	Recovery Action
E	THE DATA-BASE-IDENTIFIER IS NOT ASCENDING OR DESCENDING SET KEY OF THE set-name SET	A field that is not the key of a sorted set is being used as such, or the field is not within the specified record.	Check the source documentation of the schema invoked.
E	THE INVOKED DMCL dmcl-name HAS NOT BEEN DEFINED VIA THE DMCL PROCESSOR	The DMCL has not been placed in the data dictionary by the DMCLP, or the spelling of dmcl-name is incorrect.	Check with the data base administrator.
F	THE INVOKED SCHEMA schema-name CONTAINS ERRORS. THE DML PRE-PROCESSING CANNOT CONTINUE	The schema in the data dictionary is invalid.	Check with the data base administrator.
F	THE INVOKED SUB-SCHEMA subschema-name CONTAINS ERRORS. THE DML PRE-PROCESSING CANNOT CONTINUE	The subschema in the data dictionary contains errors.	Check with the data base administrator.
E	THE QUALIFIER record-name IS INVALID	The record name used to qualify the data item does not match the first record name in the statement.	Change the wrong record name to match the valid record name.
E	THE USAGE OF IDENTIFIER identifier-name IS NOT CONSISTENT WITH ITS DEFINITION.	The intended usage of the specified identifier does not agree with its defined usage within the data division of the program.	Correct the identifier-name being referenced to match its intended usage; or, correct the USAGE clause in the definition so it supports the type of reference.
E	TOO MANY COMMENTS. ONLY 50 IN A ROW ARE ALLOWED	A DML statement may not span more than 50 consecutive statements. This includes embedded comments and blank statements.	Correct the DML statement in error and rerun.
W	WARNING DML SYNTAX	The DML statement is not correct but can be interpreted correctly.	Check for valid format.
W	WARNING DML SYNTAX. SET NAME set-name IS NOT OWNER PROCESSABLE	The set specified in FIND OWNER RECORD OF set-name SET does not have an owner pointer.	The DML preprocessor generates the proper DML CALL statement. This message is a warning that this type DML accessing would be poorly performed.
F	WORKING STORAGE SECTION HEADER MISSING	The working storage section header is required.	Correct and rerun.



USER COMMENT SHEET

Your comments concerning this document will be welcomed by Sperry Univac for use in improving subsequent editions.

Please note: This form is not intended to be used as an order blank.

(Document Title)

(Document No.)

(Revision No.)

(Update No.)

Comments:

ong line.

From:

(Name of User)

(Business Address)

Fold on dotted lines, and mail. (No postage stamp is necessary if mailed in the U.S.A.)
Thank you for your cooperation

FOLD



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

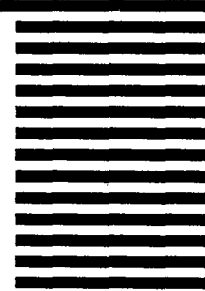
FIRST CLASS PERMIT NO. 21 BLUE BELL, PA.

POSTAGE WILL BE PAID BY ADDRESSEE

SPERRY UNIVAC

ATTN.: SYSTEMS PUBLICATIONS

P.O. BOX 500
BLUE BELL, PENNSYLVANIA 19424



CUT

FOLD