

**PUBLICATIONS
UPDATE**

Series 90

Data Management System
(DMS/90) Data Description
Language

Programmer Reference

UP-8022 Rev. 1-B

This Library Memo announces the release and availability of Updating Package B to "SPERRY UNIVAC Series 90 Data Base Management System (DMS/90) Data Description Language Programmer Reference," UP-8022 Rev. 1.

This updating package provides a default for the CALC clause of the LOCATION MODE sentence.

Copies of Updating Package B are now available for requisitioning. Either the updating package alone, or the complete manual with the updating package may be requisitioned by your local Sperry Univac Representative. To receive the updating package alone, order UP-8022 Rev. 1-B. To receive the complete manual, order UP-8022 Rev. 1.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS:
Mailing Lists 217, 630 and 692	Mailing Lists 67, 68, 71, 72, 75, 76, H7 and H8 (Package B to UP-8022 Rev. 1, 14 pages plus Memo)	Library Memo for UP-8022 Rev. 1-B RELEASE DATE: July, 1975



**PUBLICATIONS
UPDATE**

SERIES 90
**Data Management
System (DMS/90)
Data Description
Language**
Programmer Reference
UP-8022 Rev. 1-A

This Library Memo announces the release and availability of Updating Package A to "SPERRY UNIVAC Series 90 Data Base Management system (DMS/90) Data Description Language Programmer Reference," UP-8022 Rev. 1.

This updating package adds Appendix E, DMS/90 Clock Rule, to the original manual. This appendix describes the procedure to be used for assigning data base key position numbers to a given record type.

Copies of Update Package A are now available for requisitioning. Either the updating package alone, or the complete manual with the updating package may be requisitioned by your SPERRY UNIVAC representative. To receive the updating package alone, order UP-8022 Rev. 1-A. To receive the complete Manual, order UP-8022 Rev. 1.

<p>Mailing Lists 217, 630 and 692</p>	<p>Mailing Lists 67, 68, 71 and 72 (Package A to UP-8022 Rev. 1, 12 pages plus Memo)</p>	<p>Library Memo for UP-8022 Rev. 1-A</p> <p>October, 1974</p>
--	---	---



SPERRY UNIVAC

Series 90

**Data Base Management
System (DMS/90)**

**Data Description
Language**

Programmer Reference

This document contains the latest information available at the time of publication. However, Sperry Univac reserves the right to modify or revise its contents. To ensure that you have the most recent information, contact your local Sperry Univac representative.

Sperry Univac is a division of Sperry Rand Corporation.

FASTRAND, PAGERWRITER, SPERRY UNIVAC, UNISCOPE, UNISERVO, and UNIVAC are trademarks of the Sperry Rand Corporation.

COPYRIGHT, 1972 BY THE B.F. GOODRICH COMPANY
REPRODUCED WITH PERMISSION

SPERRY UNIVAC Series 90

**Data Base Management
System (DMS/90)**

Data Description Language

Programmer Reference

This document contains the latest information available at the time of publication. However, Sperry Univac reserves the right to modify or revise its contents. To ensure that you have the most recent information, contact your local Sperry Univac representative.

Sperry Univac is a division of Sperry Rand Corporation.

FASTRAND, PAGEWRITER, SPERRY UNIVAC, UNISCOPE, UNISERVO, and UNIVAC are trademarks of the Sperry Rand Corporation.





Acknowledgment

Sperry Univac wishes to acknowledge the efforts of the CODASYL Programming Language Committee (PLC) Data Base Task Group (DBTG). The DBTG produced two reports containing specifications for a standardized data management facility containing a data description language and a data manipulation language. The DBTG issued its first report in October 1969; the second, a revised and expanded version of the first, was released in April 1971. Sperry Univac is a member of the DBTG and participated in the development of the data management specifications. Sperry Univac implementation of Series 90 Data Base Management System (DMS/90) is based upon the April 1971 DBTG specifications.



Preface

This document is the Programmer Reference Manual for the SPERRY UNIVAC Series 90 Data Base Management System (DMS/90). It is one of a series of manuals covering DMS/90. Because this manual contains the schema and subschema definition languages, it is a prerequisite to the use of the other reference manuals in the series covering DMS/90. For introductory concepts to DMS/90, refer to the data manipulation language manual, UP-8036 (current version).





Contents

PAGE STATUS SUMMARY

ACKNOWLEDGMENT

PREFACE

CONTENTS

1. INTRODUCTION

- | | | |
|------|---|-----|
| 1.1. | GENERAL | 1-1 |
| 1.2. | SYMBOLS, RULES, AND NOTATIONS USED IN THIS MANUAL | 1-1 |

2. LANGUAGE CONSIDERATIONS

- | | | |
|--------|-------------------------------------|-----|
| 2.1. | DATA DESCRIPTION LANGUAGE STRUCTURE | 2-1 |
| 2.1.1. | DDL Character Set | 2-1 |
| 2.1.2. | DDL Words | 2-2 |
| 2.1.3. | Types of Words | 2-2 |
| 2.2. | DDL ORGANIZATION | 2-3 |
| 2.3. | DDL CODING FORM | 2-3 |

3. SCHEMA DATA DESCRIPTION LANGUAGE

- | | | |
|--------|--------------------------------|-----|
| 3.1. | GENERAL | 3-1 |
| 3.2. | ORGANIZATION OF THE SCHEMA DDL | 3-1 |
| 3.3. | SCHEMA DESCRIPTION STATEMENTS | 3-1 |
| 3.3.1. | SCHEMA NAME Sentence | 3-2 |
| 3.3.2. | AUTHOR Paragraph | 3-3 |
| 3.3.3. | DATE Paragraph | 3-3 |
| 3.3.4. | INSTALLATION Paragraph | 3-3 |
| 3.3.5. | REMARKS Paragraph | 3-4 |

3.4.	AREA DESCRIPTION STATEMENTS	3-4
3.4.1.	AREA NAME Sentence	3-4
3.5.	RECORD DESCRIPTION STATEMENTS	3-5
3.5.1.	RECORD NAME Sentence	3-6
3.5.2.	RECORD ID Sentence	3-7
3.5.3.	LOCATION MODE Sentence	3-7
3.5.4.	WITHIN Sentence	3-8
3.5.5.	Data-Item Sentence	3-8
3.5.5.1.	Level Number and Data Name/FILLER	3-9
3.5.5.2.	REDEFINES Clause	3-10
3.5.5.3.	USAGE Clause	3-10
3.5.5.4.	SYNCHRONIZED Clause	3-11
3.5.5.5.	PICTURE Clause	3-11
3.5.5.6.	VALUE Clause	3-13
3.5.5.7.	OCCURS Clause	3-13
3.5.5.8.	INDEXED Clause	3-14
3.6.	SET DESCRIPTION STATEMENTS	3-14
3.6.1.	SET NAME Sentence	3-16
3.6.2.	ORDER Sentence	3-16
3.6.3.	MODE Sentence	3-17
3.6.4.	OWNER Sentence	3-18
3.6.5.	MEMBER Sentence	3-18
3.7.	PROCESSOR REPORTS AND SELECTION OPTIONS	3-19
3.7.1.	Processor Reports	3-19
3.7.2.	Report Selection Options	3-23
4.	SUBSCHEMA DATA DESCRIPTION LANGUAGE	
4.1.	GENERAL	4-1
4.2.	ORGANIZATION OF THE SUBSCHEMA DDL	4-1
4.3.	SUBSCHEMA IDENTIFICATION DIVISION STATEMENTS	4-1
4.3.1.	SUBSCHEMA NAME Sentence	4-2
4.3.2.	AUTHOR Paragraph	4-3
4.3.3.	DATE Paragraph	4-3
4.3.4.	INSTALLATION Paragraph	4-3
4.3.5.	REMARKS Paragraph	4-4
4.4.	SUBSCHEMA DATA DIVISION STATEMENTS	4-4
4.4.1.	AREA SECTION Statements	4-5
4.4.2.	RECORD SECTION Statements	4-5
4.4.3.	SET SECTION Statements	4-6
4.5.	PROCESSOR REPORTS	4-7

APPENDIXES

A. DMS/90 LIST OF RESERVED WORDS

A.1.	RESERVED WORD LIST	A-1
A.2.	RESTRICTED DATA NAMES AND PROCEDURE NAMES	A-2

B. SCHEMA DDL SYNTAX

C. SUBSCHEMA DDL SYNTAX

D. SAMPLE SCHEMA AND SUBSCHEMA DEFINITIONS

D.1.	GENERAL	D-1
D.2.	THE SCHEMA DEFINITION (DMSSCHM)	D-2
D.3.	THE SUBSCHEMA DEFINITION (DMSSUBS)	D-5

E. DMS/90 CLOCK RULE

E.1.	GENERAL	E-1
E.2.	ASSIGNING DATA BASE KEY POSITION NUMBERS	E-1

INDEX

USER COMMENT SHEET

FIGURES

2-1.	COBOL Programming Form	2-3
3-1.	Sample Schema DDL Input Statement Listing	3-20
3-2.	Sample Range Map Listing	3-21
3-3.	Sample Record Description Listing	3-22
3-4.	Sample Set Description Listing	3-23
3-5.	Sample Warning and Error Message Listings	3-24

4-1.	Sample Listing of Subschema DDL Input Statements	4-8
4-2.	Sample Listing of Subschema Data Records	4-9
4-3.	Sample Listing of Record Definition Entries	4-9
4-4.	Sample Listing of Area Definition Entries	4-9
4-5.	Sample Listing of Set Definition Entries	4-10
4-6.	Sample Listing of DMS/90 Bind Calls	4-10

→ E-1. Clock Rule Example for Assigning Data Base Key Position Numbers E-2

TABLES

2-1.	Programming Form Column Usage	2-4
------	-------------------------------	-----

1. Introduction

1.1. GENERAL

This manual describes the schema and subschema data description language (DDL) of the SPERRY UNIVAC Series 90 Data Base Management System (DMS/90).

The schema is defined as the total description of a data base, including the names and descriptions of all areas, records, and sets. The language used in describing the schema is called the schema DDL.

The schema DDL compiler generates a description of the data base that is stored in the first 500 pages of the data base itself. This data base description, or output schema, consists of records organized in a network data structure. In fact, the schema compiler uses the run time data base management system (DBMS) routines of DMS/90 and a predefined object subschema to create the output schema. The output schema is later referenced by the subschema DDL compiler, which also stores records in the same network data structure to describe a particular subschema.

A subschema is defined as that portion of the total schema (selected areas, records, and sets) which is of interest to one or more specific application programs. The language used in describing a subschema is called the subschema DDL.

The subschema DDL compiler generates a description of a portion of the data base. This description takes two forms. One form is assembler source code that is subsequently assembled and added to an object module library file. This form, the object subschema, is later link edited with any COBOL (DML) application programs that invoke the subschema. The other form consists of records stored in the same network data structure as the output schema, i.e., in the area consisting of the first 500 pages of the data base. This form is subsequently referenced by the DML preprocessor when a COBOL (DML) application program that invokes the subschema is processed.

The maximum number of subschema descriptions that can be stored in these first 500 pages can vary, depending on the size of the schema and subschema, and the physical size of each page.

One schema and one or more subschemas may be described for a given data base. The resulting descriptions must be logically consistent with one another.

1.2. SYMBOLS, RULES, AND NOTATIONS USED IN THIS MANUAL

The various language elements that comprise a schema and subschema DDL program must be written in formats that adhere to fixed and precise rules of presentation. Each format statement indicates the following information:

- order of presentation;
- words that are requisite to the proper functioning of the statement;

- words that are optional and are included at the discretion of the user;
- information which must be supplied by the user;
- elements in the statement that involve a choice by the user;
- functions of the statement that are optional.

In accordance with the foregoing, the following conventions are used in this manual:

1. The order of presentation is indicated by the format statement itself.
2. All DDL reserved words appear in all capitals. They are also listed in Appendix A.
3. Words in all capitals that are underlined are key words. Key words must be present when the functions in which they appear are used except in the case of default options (rule 5). Those capitalized words not underlined are optional and may be included at the user's discretion to improve readability; there is no compiler action. All completely capitalized words, whether underlined or not, are part of the DDL language and must be spelled exactly as indicated.
4. All lowercase words represent generic terms to be supplied by the user when the functions of which they are a part are used.
5. Elements of a statement involving a choice, one of which must be chosen, are enclosed in braces { }. If one of the choices within the braces has no key words, it is a default option; i.e., if none of the elements within the braces is specified, the action will be the same as if the default option had been specified.
6. Optional functions, which may be included or omitted at the user's discretion, are enclosed in brackets []. When two or more items are stacked within brackets, one or none of them may be specified.
7. In some statements, certain portions may be used as many times as needed by the programmer. The ellipsis (. . .) indicates this repeatability. If there is a choice to be made from stacked options, or if there is only a single possibility, brackets or braces are used as delimiters to indicate that portion of the statement which is repeatable.
8. Periods must be used where shown and must also appear at the end of each paragraph.

2. Language Considerations

2.1. DATA DESCRIPTION LANGUAGE STRUCTURE

The structure of both the schema and subschema data description language (DDL) has its foundation in the DDL character set where one or more characters form a word, one or more words form a clause, one or more clauses form a sentence, and one or more sentences form a paragraph.

2.1.1. DDL Character Set

The complete DDL character set for the SPERRY UNIVAC Series 90 Data Base Management System (DMS/90) is:

0,1,...,9

A,B,...,Z

Blank or space (written on coding form as Δ , or a blank space)

.

Period

<

Less than

(

Left parenthesis

+

Plus sign

\$

Currency sign

*

Asterisk

)

Right parenthesis

;

Semicolon

-

Minus sign or hyphen

,

Comma

>

Greater than

'

Apostrophe (alternate character for quotation mark)

- = Equal sign
- " Quotation mark (see apostrophe)
- / Slash

The period, comma, semicolon, and space characters may be used for punctuation where the period terminates a sentence or paragraph, and the other characters denote separation between words or clauses. The period, comma, and semicolon are recognized as punctuation only when followed by the space character.

2.1.2. DDL Words

DDL words are composed of 1 to 30 characters from the preceding character set. Allowable combinations of characters are dependent on the type of word. A word is terminated by one of the punctuation characters given in 2.1.1.

2.1.3. Types of Words

There are three types of words in the DDL: reserved, schema and subschema names, and constants.

■ Reserved Words

Reserved words are used for syntactical purposes and may not be used as user-supplied words. There are two types of reserved words: key and optional.

- Key reserved words are required in a DDL statement. These words are capitalized and underlined in the syntactical formats shown in Sections 3 and 4.
- Optional reserved words are those which may be used in a DDL statement as desired for syntax clarity and are capitalized but not underlined in the syntactical formats.

Appendix A contains a complete list of reserved words used by DMS/90.

→ ■ Schema and Subschema Symbolic Names

Schema and subschema names must be constructed from the letters, digits, and hyphen characters only. The name must contain at least 1 letter, be no more than 16 characters in length, and not begin or end with a hyphen. The hyphen must be preceded and followed by a letter or digit. Exceptions are noted in the text where applicable.

■ Constants

Constants are used to establish initial data values for data items defined by the schema DDL. Constants are classified into two categories: literals and figurative constants.

Literals may further divide into numeric and nonnumeric.

- Numeric literals are associated with numeric data items and are not enclosed in quotation marks.
- Nonnumeric literals are associated with nonnumeric data items and are enclosed in quotation marks.

Figurative constants are reserved words which are equated to specific values and may be associated with numeric or nonnumeric data items. The following figurative constants are recognized by the schema DDL processor: ZERO, ZEROS, ZEROES, SPACE, SPACES, HIGH-VALUE, HIGH-VALUES, LOW-VALUE, and LOW-VALUES.

2.2. DDL ORGANIZATION

The organization of the schema and subschema DDL is discussed in detail with the syntax description in Sections 3 and 4 of this manual.

2.3. DDL CODING FORM

Schema and subschema DDL processor input record format is 80 characters equatable to one line of information from the standard COBOL coding form. Figure 2-1 shows the layout of the COBOL coding form. On this form, the programmer, using the rules of format and content defined in this manual, enters all the information needed by the DDL compiler. Table 2-1 explains the divisions of the form.

UNIVAC **COBOL**

PROGRAMMING FORM

PROGRAM I.D. 72 80

PROGRAM _____ PROGRAMMER _____ DATE _____ PAGE _____

SEQUENCE NUMBER	A	B	TEXT	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	IDENTIFICATION
1			TEXT																									
2																												
3																												
4																												
5																												
6																												
7																												
8																												
9																												
10																												

Figure 2-1. COBOL Programming Form

Table 2-1. Programming Form Column Usage

Columns	Designation	Contents
1-6	SEQUENCE NUMBER	A numeric entry, used only by the programmer (not the DDL processor) to establish a sequence among the various lines of coding (optional).
7	CONTINUATION	A hyphen (-) is used when an entry extends past one noncomment line. A break is used in the middle of a word, and the hyphen is written in column 7 of the next contiguous line on which the word is completed. A word may be interrupted in any column, the rest of the line space-filled, and completed on the next line. If the continued line contains a nonnumeric literal without a closing quotation mark, the first nonblank character in Area B of the continuation line must be a quotation mark, and the continuation starts with the character immediately after that quotation mark.
7	COMMENT	An asterisk (*) in column 7 signifies a comment line which will be printed but ignored by the compiler. A comment may appear anywhere in the program and can contain any printable combination of characters including reserved words. If a comment entry extending past one line has a break occurring in the middle of a word, the continuation line must contain an asterisk in column 7. (The hyphen is only used for noncomment continuation lines.)
8-72	TEXT	All DDL-formatted information, in the form of names, statements, information, instructions, etc., that is to be compiled into the object program. Note that two left-margin limits designated "A" and "B" are shown. These are needed for program alignment. Major definitive names are begun at margin A (column 8). Margin B (column 12) is used for subordinate items and for continuations of entries from the last preceding line.
73-80	IDENTIFICATION	Card deck information (optional)

3. Schema Data Description Language

Rules:

AREA NAME IS area-name RANGE IS integer-1 THRU integer-2

Format:

The AREA NAME sentence identifies an area of the total schema, and defines the page range of that area.

Function:

3.1. GENERAL

The schema data description language (schema DDL) is a subset of the April 1971 CODASYL Data Base Task Group language specification, and is used to describe an integrated data base which is destined to reside on disc.

3.4.1. AREA NAME Sentence

2. A minimum of one AREA NAME sentence must be specified.

3.2. ORGANIZATION OF THE SCHEMA DDL

1. Area description statements begin with a paragraph name composed of the words AREA DESCRIPTION. The schema DDL is composed of the following four groups of statements:

Rules:

1. Schema description

2. Area description

AREA NAME IS area-name RANGE IS integer-1 THRU integer-2

3. Record type description

AREA DESCRIPTION

4. Set type description

Area	A or B	A or B
Area	A or B	A or B
Area	A or B	A or B

Each statement group must appear in the sequence shown, each describes a specific aspect of the total schema, and each relies on descriptive statements from the prior group or groups.

Format:

Each statement group begins with a descriptive paragraph name. Processing of statements for a specific group begins with the associated paragraph name and ends either with the recognition of the paragraph name or the end of the statement input stream.

In addition to these statement groups, optional processor report control and input sequence check control parameter statements (3.7) may be entered in the statement input stream prior to the schema description statements.

3.4. AREA DESCRIPTION STATEMENTS

3.3. SCHEMA DESCRIPTION STATEMENTS

REMARKS, [comments].

Function:

Format:

Schema description statements identify the schema, and may define the page range of the schema (at the user's option), identify the schema author, define the date of schema creation, identify the installation for the schema, and present any desired remarks about the schema on the statement input listing.

Function:

3.3.5. REMARKS Paragraph

5. The integer-1 word must be an unsigned number composed of one to eight numeric digits, greater than or equal to 500, representing the lowest page number of the schema (not including page zero). This number must be higher than the DMS/90 reserved area, which includes the page numbers 0 through 499.
4. If the RANGE clause is specified, subsequent processor auditing of area and record ranges is performed on the range specified.
3. The integer-2 word must be an unsigned number composed of one to eight numeric digits representing the highest page number of the schema and may be omitted if the integer-1 word is specified.
2. If the RANGE clause is specified, the word RANGE begins in the B area of the same or subsequent line.

3.3.2. AUTHOR Paragraph

Function:

1. The word SCHEMA begins in the A area.

Rules:

The AUTHOR paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

Format:

AUTHOR. [comments].

schema.

The SCHEMA NAME sentence identifies the schema and, at the user's option, defines the page range of the

3.3.3. DATE Paragraph

Function:

Function:

3.3.1. SCHEMA NAME Sentence

The DATE paragraph may be used to specify a schema date of creation other than the computer date. If the date override feature is not desired, the DATE paragraph must be omitted entirely and the computer date is assumed.

Format:

and must be the only occurrence of the sentence within the schema DDL input stream.

DATE. date-entry.

DESCRIPTION followed by a period and a space. Additional entries in the same line are ignored.

1. Schema description statements begin with a paragraph name composed of the words SCHEMA
2. The SCHEMA NAME sentence must immediately follow the SCHEMA DESCRIPTION paragraph name.
3. The remaining optional paragraphs and associated entries may appear in any order.

Rules:

1. The word DATE begins in the A area and must be followed by a period and a space.
2. The date-entry word must accompany the DATE paragraph name and begin in the B area of the same line. The date-entry word must be eight characters in the form mm/dd/yy where mm, dd, and yy are the month, day, and year numbers, respectively.

[DATE, date-entry.]

[AUTHOR, [comments].]

3.3.4. INSTALLATION Paragraph

Function:

SCHEMA NAME IS schema-name [RANGE IS integer-1 THRU integer-2].

The INSTALLATION paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

INSTALLATION. [comments].

Area	A or B	Area
Area	A or B	Area
Area	A or B	Area

Format:

5. The integer-1 word must be an unsigned number composed of one to eight numeric digits, greater than or equal to 500, representing the lowest page number of the schema (not including page zero). This number must be higher than the DMS/90 reserved page numbers, which include the page numbers 0 through 499.

4. If the RANGE clause is specified, subsequent processor auditing of area and record page ranges is

6. The integer-2 word must be an unsigned number composed of one to eight numeric digits representing the highest page number of the schema and must be greater than the integer-1 word. The schema name and record range must be specified and must be in the same or subsequent line.

3. If the RANGE clause is specified, the word RANGE begins in the B area of the same or subsequent line.

2. If the RANGE clause is specified, the word RANGE begins in the B area of the same or subsequent line.

3.3.2. AUTHOR Paragraph

Function:

1. The word SCHEMA begins in the A area.

Rules:

The AUTHOR paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

Format:

AUTHOR. [comments].

schema.

The SCHEMA NAME sentence identifies the schema and, at the user's option, defines the page range of the

3.3.3. DATE Paragraph

Function:

Function:

3.3.1. SCHEMA NAME Sentence

The DATE paragraph may be used to specify a schema date of creation other than the computer date. If the date override feature is not desired, the DATE paragraph must be omitted entirely and the computer date is assumed.

3. The remaining optional paragraphs and associated entries may appear in any order and are assumed to be in the same or subsequent line.

Format:

and must be the only occurrence of the sentence within the schema DDL input stream.

2. The SCHEMA NAME sentence must immediately follow the SCHEMA DESCRIPTION paragraph name.

DATE. date-entry.

DESCRIPTION followed by a period and a space. Additional entries in the same line are ignored.

1. Schema description statements begin with a paragraph name composed of the words SCHEMA

Rules:

1. The word DATE begins in the A area and must be followed by a period and a space.

2. The date-entry word must accompany the DATE paragraph name and begin in the B area of the same line. The date-entry word must be eight characters in the form mm/dd/yy where mm, dd, and yy are the month, day, and year numbers, respectively.

[DATE, date-entry.]

3.3.4. INSTALLATION Paragraph

[AUTHOR, [comments].]

Function:

SCHEMA NAME IS schema-name [RANGE IS integer-1 THRU integer-2].

The INSTALLATION paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

A Area	A Area	A Area
A or B	A Area	B

INSTALLATION. [comments].

Format:

3. Schema Data Description Language

Rules:

AREA NAME IS area-name RANGE IS integer-1 THRU integer-2.

Format:

The AREA NAME sentence identifies an area of the total schema, and defines the page range of that area.

Function:

3.1. GENERAL

3.4.1. AREA NAME Sentence

The schema data description language (schema DDL) is a subset of the April 1971 CODASYL Data Base Task Group language specification, and is used to describe an integrated data base which is destined to reside on disc.

2. A minimum of one AREA NAME sentence must be specified.

3.2. ORGANIZATION OF THE SCHEMA DDL

1. Area description statements begin with a paragraph name composed of the words AREA DESCRIPTION followed by a period and a space. Additional entries in the same line are separated by a period and a space. The schema DDL is composed of the following four groups of statements:

Rules:

1. Schema description

AREA NAME IS area-name RANGE IS integer-1 THRU integer-2.

2. Area description

AREA DESCRIPTION.

3. Record type description

4. Set type description

Area	A or B	Area
Area	A or B	Area
Area	A or B	Area

Each statement group must appear in the sequence shown, each describes a specific aspect of the total schema, and each relies on descriptive statements from the prior group or groups.

Format:

Each statement group begins with a descriptive paragraph name. Processing of statements for a specific group begins with the associated paragraph name and ends either with the recognition of the paragraph name for the next group or the area description statements identify one or more areas of the total schema, and define the page range of each area identified.

In addition to these statement groups, optional processor report control and input sequence check control parameter statements (3.7) may be entered in the statement input stream prior to the schema description statements.

3.4. AREA DESCRIPTION STATEMENTS

3.3. SCHEMA DESCRIPTION STATEMENTS

REMARKS. [comments].

Function:

Format:

Schema description statements identify the schema, and may define the page range of the schema (at the user's option), identify the schema author, define the date of schema creation, identify the installation for the listing. The REMARKS paragraph is created as a comment and is presented on the processor input statement schema, and present any desired remarks about the schema on the statement input listing.

Function:

3.3.5. REMARKS Paragraph

2. If multiple areas are defined, the associated AREA NAME sentences may be placed in any desired sequence.
3. The area-name word must be a unique name and conform to the rules for formation of schema and subschema symbolic names (2.1.3).
4. Schema pages assigned to one area may not be assigned to any other area. An area may not be partially or wholly contained within another area. Schema pages may be left unassigned to any area.
5. The integer-1 word must be an unsigned number composed of one to eight numeric digits representing the lowest page number assigned to the named area.
6. The integer-2 word must be an unsigned number composed of one to eight numeric digits representing the highest page number assigned to the named area. The integer-2 word must be of equal or greater value than integer-1.
7. The integer-1 word and the integer-2 word must be consistent with the range specified in the schema name sentence.

3.5. RECORD DESCRIPTION STATEMENTS

Function:

Record description statements identify one or more records of the total schema, and define the mode of storage, the area of storage, and the data content for each record.

Format:

Area A	Area A or B	Area B
-----------	----------------	-----------

RECORD DESCRIPTION.

RECORD NAME IS record-name.

RECORD ID IS integer.

LOCATION MODE IS

$$\left\{ \begin{array}{l} \text{DIRECT} \\ \text{CALC USING identifier} \\ \text{VIA set-name SET} \end{array} \right\} \text{ DUPLICATES ARE } \left\{ \begin{array}{l} \text{FIRST} \\ \text{LAST} \\ \text{NOT ALLOWED} \end{array} \right\} .$$

WITHIN area-name AREA.

level-number $\left\{ \begin{array}{l} \text{data-name-1} \\ \text{FILLER} \end{array} \right\}$

[REDEFINES data-name-2]

[USAGE IS {
DISPLAY
COMP
COMPUTATIONAL
COMP-1
COMPUTATIONAL-1
COMP-2
COMPUTATIONAL-2
COMP-3
COMPUTATIONAL-3 }]

[SYNCHRONIZED]
[SYNC]

[PICTURE] IS character-string
[PIC]]

[VALUE IS literal]

[OCCURS integer TIMES]

[INDEXED BY index-name] _

Rules:

1. Record description statements begin with a paragraph name composed of the words RECORD DESCRIPTION followed by a period and a space. Additional entries in the same line are ignored. A RECORD NAME sentence must immediately follow the RECORD DESCRIPTION paragraph name, and a minimum of one record must be described for a schema.
2. The description of a schema record is accomplished by entry of one RECORD NAME, RECORD ID, LOCATION MODE, and WITHIN sentence, and a minimum of one data-item sentence. The RECORD NAME sentence must be the first for each record described, and the data-item sentence or sentences must be the last for each record.

3.5.1. RECORD NAME Sentence

Function:

The RECORD NAME sentence alphanumerically identifies, by name, a schema record type.

Format:

RECORD NAME IS record-name_

Rules:

1. The word RECORD may begin in the A or B area.

- The record-name word must be a unique name and conform to the rules for formation of schema/subschema symbolic names (2.1.3).

3.5.2. RECORD ID Sentence

Function:

The RECORD ID sentence numerically identifies, by number, a schema record type.

Format:

RECORD ID IS integer.

Rules:

- The word RECORD may begin in the A or B area.
- The integer word must be an unsigned number composed of three or four numeric digits from the range 100 through 9999. Numeric identifiers from 1 through 99 are reserved for use by DMS/90.
- The integer word values on subsequent RECORD ID sentences may be assigned arbitrarily.

3.5.3. LOCATION MODE Sentence

Function:

The LOCATION MODE sentence defines the selection or storage criteria for a record occurrence in an area.

Format:

LOCATION MODE IS

$$\left\{ \begin{array}{l} \underline{\text{DIRECT}} \\ \underline{\text{CALC USING}} \text{ identifier } \underline{\text{DUPLICATES ARE}} \left\{ \begin{array}{l} \underline{\text{FIRST}} \\ \underline{\text{LAST}} \\ \underline{\text{NOT ALLOWED}} \end{array} \right\} \\ \underline{\text{VIA}} \text{ set-name SET} \end{array} \right\} .$$

Rules:

- The word LOCATION may begin in the A or B area. All subsequent clauses and words of the sentence must be entered in the B area of the same or following lines.
- Only one of the modes DIRECT, CALC, or VIA may be selected for a specific schema record.
- DIRECT mode is specified for a schema record which is to be stored into a specific data base area or page dependent on user direction through the data manipulation language (DML) preprocessor supplied currency-status field DIRECT-DBK. See the DMS/90 data manipulation language programmer reference manual, UP-8036 (current version).

- ↓
- ↑
4. CALC mode is specified for a schema record which is to be stored into a specific data base page based on the randomized contents of an identifier described within the record, and the page range associated with the record. A standard randomizing module, XR7CALC, is provided with DMS/90 and is used by default. If the user wishes to provide his own randomizing algorithm, however, he may create his own CALC module and link it with all application programs. Details of this process are given in DMS/90 system support functions, UP-8037 (current version).
 5. The identifier word must be the name of a data item defined by a data-item sentence for the schema record. Any group or elementary data item may be used as an identifier with the exception of the following: data items named FILLER, data items which redefine or are subordinate to a data item which redefines, and data items which are occurred or are subordinate to a data item which is occurred (3.5.5).
 6. A DUPLICATES clause is associated with the CALC mode, and is a required entry. A duplicate record will be stored in a first-in, first-out (FIFO) or last-in, first-out (LIFO) sequence or not stored at all depending on the clause entry.
 7. VIA mode is specified for a schema record which is to be stored into a data base as a member record in a set relationship, and as such should be located as close as possible to owner record of the set.
 8. The set-name word must be the name of a set which is defined within the schema.

3.5.4. WITHIN Sentence

Function:

The WITHIN sentence is used to specify the area of the schema into which a record may be stored.

Format:

WITHIN area-name AREA.

Rules:

1. The word WITHIN may begin in the A or B area; all other clauses or words for the sentence must begin in the B area of the same or subsequent lines.
2. The area-name word must be the name of an area that is defined within the schema. The page range associated with the named area is assigned to the schema record.

3.5.5. Data-Item Sentence

Function:

The data-item sentence defines group or elementary data items of a schema record.

Format:

level-number { data-name-1
FILLER }

[REDEFINES data-name-2]

[USAGE IS { DISPLAY
COMP
COMPUTATIONAL
COMP-1
COMPUTATIONAL-1
COMP-2
COMPUTATIONAL-2
COMP-3
COMPUTATIONAL-3 }]

{ SYNCHRONIZED
SYNC }

[{ PICTURE
PIC } IS character-string]

[VALUE IS literal]

[OCCURS integer TIMES]

[INDEXED BY index-name]_

Rules:

1. The level-number word may begin in the A or B area; all other clauses or words for the sentence must begin in the B area of the same or subsequent lines.
2. The structure of and the rules governing the formation and use of the data-item sentence within the schema DDL are similar to those of the data description entries of COBOL. (See COBOL supplementary reference manual, UP-7973 (current version).) Exceptions and restrictions to COBOL are noted where applicable.

3.5.5.1. Level Number and Data Name/FILLER

Function:

The level number identifies the group or elementary relationship of a data item to the total record and to other data items within the record.

Format:

level-number { data-name-1
FILLER }

Rules:

1. The level-number word must be a 2-digit number from the series 02 through 49. Note that special level numbers (such as 66, 77, 88) and their associated functions are not recognized by the schema DDL processor.
2. The data-name-1 word must be a unique name and conform to the rules for formation of schema and subschema names (2.1.1).
3. FILLER may be used to specify elementary or group items that are never referred to directly in the program, and therefore need not be named. These data items that are never referred to in the program do occupy space within the record; hence, this space must be defined to ensure proper alignment of data items that are used.

3.5.5.2. REDEFINES Clause

Function:

The REDEFINES clause permits the redefinition of storage previously defined in a data-item sentence for the record.

Format:

REDEFINES data-name-2

Rules:

1. The level number associated with the redefining data item must be equal to that of the redefined item.
2. The data-name-2 word must be the name of a previously defined data item.
3. A data item which redefines or which is subordinate to one which redefines may not be used as a control identifier for CALC location mode storage or for a sorted set. In addition, no VALUE clause may be specified for the item.

3.5.5.3. USAGE Clause

Function:

1. The USAGE clause specifies the manner in which a data item is represented in storage, and may only be specified at the elementary level. A usage of DISPLAY is assumed in the absence of the clause.

Format:

[USAGE IS] {
DISPLAY
COMP
COMPUTATIONAL
COMP-1
COMPUTATIONAL-1
COMP-2
COMPUTATIONAL-2
COMP-3
COMPUTATIONAL-3

Rules:

1. DISPLAY usage specifies data-item storage in character format (one character = eight bits = one byte). At the elemental level, a PICTURE clause must accompany this usage.
2. COMPUTATIONAL usage specifies data-item storage in binary format and may only be used at the elemental level. A PICTURE clause must accompany this usage. The amount of storage required by a computational data item is dependent on the number of decimal digits represented by its picture.

<u>Number of Decimal Digits</u>	<u>Storage Required</u>
1 through 4	2 bytes
5 through 9	4 bytes
10 through 18	8 bytes

3. COMPUTATIONAL-1 or COMP-1 usage specifies data-item storage in internal floating-point (short-precision) format, and requires four bytes of storage. It may only be specified at the elemental level, and no PICTURE or VALUES clause may accompany this usage.
4. COMPUTATIONAL-2 or COMP-2 usage specifies data-item storage in internal floating-point (long-precision) format, and requires eight bytes of storage. It may only be specified at the elemental level, and no PICTURE or VALUE clause may accompany this usage.
5. COMPUTATIONAL-3 or COMP-3 usage specifies data-item storage in internal decimal (packed decimal) format, and may only be used at the elemental level. A PICTURE clause must accompany this usage. The amount of storage required by a COMPUTATIONAL-3 data item is equal to 1/2 of a byte (four bits), plus 1/2 of a byte for each decimal digit represented by the PICTURE, plus 1/2 of a byte (as required) to round the total required storage to the next whole byte.

3.5.5.4. SYNCHRONIZED Clause

Function:

The schema DDL processor treats the SYNCHRONIZED clause as comments. DML preprocessor-generated record and data-item statements will not include synchronization.

Format:

```
{ SYNCHRONIZED
  SYNC }
```

3.5.5.5. PICTURE Clause

Function:

The PICTURE clause describes the general characteristics of an elementary data item, and is required in all cases except where USAGE IS COMPUTATIONAL-1 or COMPUTATIONAL-2.

Format:

```
{ PIC
  PICTURE } IS character-string
```

Rules:

1. The character-string word represents the actual PICTURE and may be no more than 30 characters in length.
2. The combination of PICTURE characters determines the category of data described by the PICTURE clause. The schema DDL processor recognizes two categories of data: alphanumeric and numeric. Alphabetic, alphanumeric-edited, and numeric-edited data categories are not recognized.
3. Alphanumeric items are those whose character strings are restricted to the symbol X and an integer enclosed in parentheses representing consecutive appearances of X. Data items in this category must have a usage of DISPLAY, and if a VALUE clause is specified, the literal must be nonnumeric or a figurative constant. For example, the character string XXX describes three alphanumeric characters, and the character string X(6) describes six alphanumeric characters.
4. Numeric items are further categorized into two types: fixed point and floating point.
 - Fixed-point numeric items are those whose character strings are restricted to the symbols 9, V, P, S, and an integer enclosed in parentheses to represent consecutive appearances of 9 or P. Data items in this category must have a usage of DISPLAY, COMPUTATIONAL, or COMPUTATIONAL-3. If a VALUE clause is specified, the literal must be numeric or one of the figurative constants ZERO, ZEROS, or ZEROES. Examples are:

999

Describes three numeric characters.

99V99

Describes four numeric characters; V designates the position of the decimal point.

S9(6)V99

Describes eight numeric characters preceded by an operational sign; V designates the decimal point ($\{\begin{smallmatrix} + \\ - \end{smallmatrix}\}$ 999999.99).

SVP(4)9(6)

Describes 10 numeric characters preceded by an operational sign; V designates the decimal point; P(4) indicates that zeros are to be the four most significant characters following the decimal point ($\{\begin{smallmatrix} + \\ - \end{smallmatrix}\}$.0000999999).

S9(4)VP(6)

Describes 10 numeric characters preceded by an operational sign; V designates the decimal point; P(6) indicates the number of characters following the decimal point ($\{\begin{smallmatrix} + \\ - \end{smallmatrix}\}$ 9999.999999).

- Floating-point numeric items are those whose character strings are restricted to the symbols +, -, 9, V, .(period), E, and an integer enclosed in parentheses to represent consecutive appearances of 9.

A floating-point display item has a picture string in the form:

$\{\begin{smallmatrix} + \\ - \end{smallmatrix}\}$ mantissa E $\{\begin{smallmatrix} + \\ - \end{smallmatrix}\}$ exponent

Data items in this category must have a usage of DISPLAY, and no VALUE clause may be specified. Only external floating-point items have variable picture formats; consequently, these items require a PICTURE clause. For internal floating-point data-item requirements, see the USAGE clause (3.5.5.3).

3.5.5.6. VALUE Clause

Function:

The VALUE clause specifies an initial value of an elementary data item, and may be stated whenever a PICTURE clause is specified except when the data item is of the external floating-point numeric category. In addition, no VALUE clause may be present on a data item which is of COMPUTATIONAL-1 or COMPUTATIONAL-2 usage, or a data item which redefines or is subordinate to one which redefines, or a data item which is occurred or is subordinate to one which is occurred.

Format:

VALUE IS literal

Rules:

1. The literal word represents the actual initial value desired and may be no more than 30 characters in length. The schema DDL processor recognizes figurative constants, nonnumeric literals, and numeric literals.
2. A figurative constant may be substituted for a literal with the following restrictions:
 - a. The literal may be any figurative constant when the data-item PICTURE is alphar.umeric.
 - b. The literal may be only one of the figurative constants ZERO, ZEROS, or ZEROES when the data-item PICTURE is numeric.
3. Nonnumeric literals may be specified only when the data-item PICTURE is alphanumeric. The literal must be enclosed in quotation marks (or apostrophes), and the number of enclosed characters must not exceed the number of characters represented by the PICTURE.
4. Numeric literals may be specified only when the data-item PICTURE is numeric. The schema DDL processor does not validate the VALUE literal against the PICTURE character-string for correct format, decimal alignment, etc. The numeric literal is accepted as stated.

3.5.5.7. OCCURS Clause

Function:

The OCCURS clause defines repetitive occurrences of group or elementary data items.

Format:

OCCURS integer TIMES

Rules:

1. Data items that are occurred and data items subordinate to ones that are occurred may not be named as control identifiers for CALC location mode storage (3.5.3) or for sorted sets.
2. The integer word must be an unsigned number composed of from one to four numeric digits of the range 2 through 9999.

3.5.5.8. INDEXED Clause

Function:

The INDEXED clause may be specified only when an occurred data item and/or data items subordinate to one which is occurred are to be referenced by indexing, and for the same data item which bears the OCCURS clause.

Format:

INDEXED BY index-name

Rule:

1. The index-name word must be a unique name and conform to the rules for formation of schema/subschema names.

3.6. SET DESCRIPTION STATEMENTS

Function:

The set description statements define the relationship between two or more record types of a schema.

Format:

Area A	Area A or B	Area B
-----------	----------------	-----------

SET DESCRIPTION.

```

SET NAME IS set-name.

ORDER IS { FIRST
           LAST
           NEXT
           PRIOR
           SORTED } .

MODE IS CHAIN [LINKED TO PRIOR].

OWNER IS record-name

           NEXT DBKEY POSITION IS integer-1

           [PRIOR DBKEY POSITION IS integer-2] .

MEMBER IS record-name

           { MANDATORY } { AUTOMATIC }
           { OPTIONAL }  { MANUAL }

           [LINKED TO OWNER]

           NEXT DBKEY POSITION IS integer-1

           [PRIOR DBKEY POSITION IS integer-2]

           [OWNER DBKEY POSITION IS integer-3]

           [ { ASCENDING } KEY IS identifier
             { DESCENDING }
             DUPLICATES ARE { FIRST
                               LAST
                               NOT ALLOWED } .
    
```

Rules:

1. Set description statements begin with a paragraph name composed of the words SET DESCRIPTION followed by a period and a space. Additional entries in the same line are ignored. The paragraph name is a required entry regardless of the presence or absence of other set type description statements.
2. The description of a schema set is accomplished by entry of one each of the SET NAME, ORDER, MODE, and OWNER sentences, and a minimum of one MEMBER sentence. The SET NAME sentence must be the first for each schema set described, and the MEMBER sentence must be the last for each schema set.

3.6.1. SET NAME Sentence

Function:

The SET NAME sentence identifies by alphanumeric name, the set type being defined.

Format:

SET NAME IS set-name_

Rules:

1. The word SET may begin in the A or B area.
2. The word set-name must be a unique name and conform to the rules for formation of schema and subschema symbolic names (2.1.3).

3.6.2. ORDER Sentence

Function:

The ORDER sentence specifies the insertion point of a member record occurrence within a set occurrence.

Format:

ORDER IS { FIRST
LAST
NEXT
PRIOR
SORTED } .

Rules:

1. The word ORDER may begin in the A or B area.
2. FIRST is specified when the desired insertion point is immediately following the owner record occurrence.
3. LAST is specified when the desired insertion point is immediately preceding the owner record occurrence.
4. NEXT is specified when the desired insertion point is immediately following the current record occurrence of the set.
5. PRIOR is specified when the desired insertion point is immediately preceding the current record occurrence of the set.
6. SORTED is specified when the desired insertion point is to be determined by the value contained within a KEY control data item of the member record.
7. When ORDER is PRIOR or LAST, prior linkage and data base key positions must also be specified.

8. If ORDER is SORTED, each set type member record definition must contain an ASCENDING or DESCENDING KEY identifier and DUPLICATES clause (3.5.5).
9. If ORDER is FIRST, LAST, NEXT, or PRIOR, no MEMBER sentence in the same SET DESCRIPTION can contain the ASCENDING or DESCENDING clause.
10. When ORDER is SORTED is specified for a given set, each MEMBER record must have an ASCENDING or DESCENDING clause to determine the order of MEMBER record occurrences of that type within a given set occurrence.
11. When format 3 of the FIND statement, which is described in the DMS/90 data manipulation language manual, UP-8036 (current version), is used in the form

FIND NEXT record-name RECORD OF set-name SET

to pass through the MEMBER record occurrences in a set, only the records of the specified type are located, and they are located in the order specified by the ASCENDING or DESCENDING clause.

12. When format 3 of the FIND statement is used in the form

FIND NEXT RECORD OF set-name SET

to pass through the member record occurrences in a set, all record occurrences of all types are located.

13. The order of records of a given type is as specified in rules 11 and 12 when other record types are ignored. The order of a record of one type relative to a record of another depends upon the order in which each occurrence of each record was inserted into the set. MEMBER record occurrences are not grouped by record type.

3.6.3. MODE Sentence

Function:

The MODE IS CHAIN sentence specifies that all records participating in the named set are to be linked to the next record.

Format:

MODE IS CHAIN [LINKED TO PRIOR].

Rules:

1. The word MODE may begin in the A or B area.
2. If the optional LINKED PRIOR clause is entered, all records participating in the named set may be linked to the prior record.
3. The LINKED PRIOR clause must be stated if the set ORDER is defined as PRIOR or LAST.

3.6.4. OWNER Sentence

Function:

The OWNER sentence specifies the name of a record, the occurrence of which establishes the existence of a set and the relative position of data base keys within the control portion of the owner record of a set. Data base key values, which are inserted into the control portion of an owner record occurrence by DMS/90 at run-time, serve to link the owner to the first and, optionally, the last member record occurrence.

Format:

OWNER IS record-name
NEXT DBKEY POSITION IS integer-1
[PRIOR DBKEY POSITION IS integer-2].

Rules:

1. The word OWNER may begin in the A or B area; additional clauses may be entered only in the B area.
2. The record-name word must be the name of a record type previously defined in the record-type description statements.
3. The integer-1 and integer-2 words must be unsigned numbers composed of no more than two numeric digits valued from 1 through 99.
4. Data base key position numbers must be assigned according to the clock rule described in Appendix E.

3.6.5. MEMBER Sentence

Function:

The MEMBER sentence specifies the name of a record, the occurrence of which may be a member of the named set. It also specifies the relative position of data base keys within the control portion of the member record. Data base key values, which are inserted into the control portion of a member record occurrence by DMS/90 at run-time, serve to link the member to the next member record occurrence and, optionally, to the prior member record occurrence and also, optionally, to the owner record occurrence.

Format:

MEMBER IS record-name

{ MANDATORY } { AUTOMATIC }
 { OPTIONAL } { MANUAL }

[LINKED TO OWNER]

NEXT DBKEY POSITION IS integer-1
[PRIOR DBKEY POSITION IS integer-2]
[OWNER DBKEY POSITION IS integer-3]

[{ ASCENDING } KEY IS identifier
 { DESCENDING }
DUPLICATES ARE { FIRST
LAST
NOT ALLOWED }] .

Rules:

1. The word MEMBER may begin in the A or B area; additional clauses may be entered only in the B area.
2. The record-name word must be the name of a record type previously defined in the record-type description statements (3.5).
3. Specification of the set membership type is accomplished by selection of the membership termination control parameter (MANDATORY or OPTIONAL) and the membership establishment control parameter (AUTOMATIC or MANUAL). See the DMS/90 data manipulation language programmer reference manual, UP-8036 (current version) for a detailed explanation of the use of the set membership type parameters.
4. When the optional LINKED OWNER clause is entered, the named member record is linked directly to its associate owner record. If selected, the owner data base key position must also be specified for the member record.
5. The integer-1, integer-2, and integer-3 words must be unsigned numbers composed of no more than two numeric digits valued from 1 through 99.
6. Data base key position numbers must be assigned according to the clock rule described in Appendix E. ←
7. The sorted set key identifier is specified by entry of the ASCENDING or DESCENDING KEY and DUPLICATES clauses.
8. The identifier word must be the name of a data item defined within the named member record (group or elementary level), and may not be FILLER, redefining, or occurred, or subordinate to a data item that redefines or is occurred.
9. Each occurrence of a member record can participate in only one occurrence of a given set type. ←

3.7. PROCESSOR REPORTS AND SELECTION OPTIONS

3.7.1. Processor Reports

The schema DDL processor can produce five separate reports, all of which are generated to a single report output file. The following, is a list of reports in order of creation, with a brief description of each.

- Schema DDL Input Statement Listing

The schema DDL input statement listing contains a full listing of all statements input to the processor (schema DDL and processor control statements). A processor assigned sequence number appears to the left of each printed input statement and is used in relating subsequent warning and error messages to specific statements. Figure 3-1 illustrates a sample schema DDL input statement listing.

- Range Map Listing

The range map listing shows, in sequence from low to high data base page number, the page range of the total schema, each area within the schema, and each record within their assigned area. This report may be produced or suppressed by user control entries. Figure 3-2 illustrates a sample range map listing.

COMPILED BY UNIVAC SERIES 90 SCHEMA COMPILER VERSION 01.02 DATE 740425

-- SCHEMA DDL INPUT STATEMENT LISTING --

LINE	SEQ.	SOURCE STATEMENT	IDENT.
000001	000100	INCLST RNGMAP RECDSE SETDES	
000002	000200*		
000003	000300*	*****	
000004	000400*	* SCHEMA DESCRIPTION STATEMENTS *	
000005	000500*	*****	
000006	000600*		
000007	000700	SCHEMA DESCRIPTION.	
000008	000800		
000009	000900	SCHEMA NAME IS DMSSCHM RANGE IS 502 THRU 540.	
000010	001000		
000011	001100	AUTHOR. DMS/90 PROGRAMMERS.	
000012	001200		
000013	001300	DATE. 04/10/73.	
000014	001400		
000015	001500	INSTALLATION. SPERRY*UNIVAC BLUE BELL.	
000016	001600		
000017	001700	REMARKS. THIS IS THE DMS/90 SAMPLE SCHEMA.	
000018	001800		
000019	001900*		
000020	002000*	*****	
000021	002100*	* AREA DESCRIPTION STATEMENTS *	
000022	002200*	*****	
000023	002300*		
000024	002400	AREA DESCRIPTION.	
000025	002500		
000026	002600	AREA NAME IS CUSTOMER-AREA RANGE IS 502 THRU 510.	
000027	002700	AREA NAME IS ORDER-AREA RANGE IS 511 THRU 530.	
000028	002800	AREA NAME IS PRODUCT-AREA RANGE IS 531 THRU 540.	
000029	002900		
000030	003000*		
000031	003100*	*****	
000032	003200*	* RECORD DESCRIPTION STATEMENTS *	
000033	003300*	*****	
000034	003400*		
000035	003500	RECORD DESCRIPTION.	
000036	003600		
000037	003700	RECORD NAME CUSTOMER .	
000038	003800	RECORD ID 611 .	
000039	003900	LOCATION MODE CALC USING CUST-NO-611 DUPPLICATES NOT ALLOWED.	
000040	004000	WITHIN CUSTOMER-AREA AREA.	
000041	004100		
000042	004200	05 CUST-NO-611 PIC X(11).	
000043	004300	05 CUST-NAME-S-611 PIC X(35).	
000044	004400	05 CUST-ADDR-S-611 PIC X(30).	
000045	004500	05 CUST-CREDIT PIC XXX.	
000046	004600	05 FILLER PIC X(125).	
000047	004700		
000048	004800	RECORD NAME CUST-ORDER .	
000049	004900	RECORD ID 620.	

Figure 3-1. Sample Schema DDL Input Statement Listing

```

----- RANGE MAP LISTING -----
SCHEMA..... DMSSCHM          00000502  0000054n
  AREA..... CUSTOMER-AREA    00000502  0000051n
    RECORD... CUSTOMER        00000502  0000051n
  AREA..... ORDER-AREA      00000511  0000053n
    RECORD... CUST-ORDER     00000511  0000053n
    RECORD... ORDER-ITEM    00000511  0000053n
    RECORD... ORD-REMARK    00000511  0000053n
  AREA..... PRODUCT-AREA    00000531  0000054n
    RECORD... PRODUCT       00000531  0000054n
----- RANGE MAP LISTING ----- END

```

Figure 3-2. Sample Range Map Listing

Record Description Listing

The record description listing restates all input entries for each record described; it also shows all data base key positions (3.5.5) that are assigned to the named record type. Special notation is also inserted on any data item which is used as a control identifier for CALC or a sorted set. Numbers appearing in the right margin are processor calculated values (number of characters or bytes), where RLGTH is the total record length (data base keys and data); KLGTH is the total length of all data base keys assigned to the record; DLGTH is the total length of the data portion of the record; and DSTRT is the data starting position relative to the beginning of the record (first character position is ZERO). For each data item, STRT is the data-item starting position relative to the beginning of the record and LGTH is the length of the data item.

Figure 3-3 shows a sample record description listing.

Set Description Listing

The set description listing, for each set defined (CALC included), shows the owner and member record types assigned, the linkage, and for sorted sets, the control identifier and duplicate control statement. Figure 3-4 shows a sample set description listing.

Warning and Error Message Listing

The warning and error message listing comprises two parts; the first is a presentation of the actual messages where each message may be composed as follows:

1. A processor-assigned sequence number of the statement for which the message was developed.
2. An error severity code. Codes and associated descriptions follow:

<u>Code</u>	<u>Description</u>
E	Indicates an error that results in an unusable schema compilation.
W	Indicates a warning of a possible error that will not hinder the resolution of the schema.

----- RECORD DESCRIPTION LISTING -----

RECORD NAME.....	CUSTOMER								RLGTH=0220
RECORD ID.....	0611								KLGTH=0016
LOCATION MODE.....	CALC USING	CUST-NO-611							DLGTH=0204
WITHIN.....	CUSTOMER-AREA			FROM 0000502		THRU 0000510			DSTART=0016
DBKEY POSITIONS....	SET.....	TYPE..	NEXT	PRIOR	OWNER				
	CALC	MEMBER	1	2					
	ORDOR	OWNER	3	4					
DATA ITEM.....	REDEFINES.....	USAGE.....	VALUE.....			PICTURE.....		STRT	LGTH
05 CUST-NO-611		DISPLAY				X(11)		0016	0011
			SET CONTROL ITEM FOR	-----		CALC			
05 CUST-NAME-S-611		DISPLAY				X(35)		0027	0036
05 CUST-ADDR-S-611		DISPLAY				X(30)		0062	0030
05 CUST-CREDIT		DISPLAY				XXX		0092	0003
05 FILLER		DISPLAY				X(125)		0095	0126

RECORD NAME.....	CUST-ORDER								RLGTH=0240
RECORD ID.....	0620								KLGTH=0036
LOCATION MODE.....	CALC USING	FO-NO-620							DLGTH=0204
WITHIN.....	ORDER-AREA			FROM 0000511		THRU 0000530			DSTART=0036
DBKEY POSITIONS....	SET.....	TYPE..	NEXT	PRIOR	OWNER				
	CALC	MEMBER	1	2					
	ORDOR	MEMBER	3	4	5				
	SPEC-REMARK	OWNER	6	7					
	ITEM	OWNER	8	9					
DATA ITEM.....	REDEFINES.....	USAGE.....	VALUE.....			PICTURE.....		STRT	LGTH
05 FO-NO-620		DISPLAY				X(18)		0036	0008
			SET CONTROL ITEM FOR	-----		CALC			
			SET CONTROL ITEM FOR	-----		ORDOR			
05 CUST-PO-NO-620		DISPLAY				X(10)		0044	0010
05 FILLER		DISPLAY				X(27)		0062	0027
05 DATE-SHIP-620		DISPLAY				X(6)		0089	0006
05 DATE-REQ-620		DISPLAY				X(6)		0095	0006
05 DATE-PROM-620		DISPLAY				X(6)		0101	0006
05 FILLER		DISPLAY				X(133)		0107	0133

RECORD NAME.....	ORDER-ITEM								RLGTH=0072
RECORD ID.....	0621								KLGTH=0024
LOCATION MODE.....	VIA SET	ITEM		DISPLACEMENT		0000 PAGES			DLGTH=0066
WITHIN.....	ORDER-AREA			FROM 0000511		THRU 0000530			DSTART=0024
DBKEY POSITIONS....	SET.....	TYPE..	NEXT	PRIOR	OWNER				
	ITEM	MEMBER	1	2	3				
	PROD-ORD	MEMBER	4	5	6				

Figure 3-3. Sample Record Description Listing

----- SET DESCRIPTION LISTING -----

```

SET..... CALC                MODE IS CHAIN                ORDER IS SORTED
OWNER.... SRI                0001 LINKED NEXT PRIOR
MEMBER... CUSTOMER          0611 LINKED NEXT PRIOR
MEMBER... CUST-ORDER        0620 LINKED NEXT PRIOR
MEMBER... PRODUCT           0631 LINKED NEXT PRIOR
                                MANDATORY AUTOMATIC ASC CUST-NO=411   DUP NOT ALLOWED
                                MANDATORY AUTOMATIC ASC FO-NO=620    DUP NOT ALLOWED
                                MANDATORY AUTOMATIC ASC PROD-NO=631   DUP NOT ALLOWED

SET..... ORDR                MODE IS CHAIN                ORDER IS SORTED
OWNER.... CUSTOMER          0611 LINKED NEXT PRIOR
MEMBER... CUST-ORDER        0620 LINKED NEXT PRIOR OWNER
                                MANDATORY AUTOMATIC ASC FO-NO=620    DUP NOT ALLOWED

SET..... SPEC-REMARK        MODE IS CHAIN                ORDER IS LACT
OWNER.... CUST-ORDER        0620 LINKED NEXT PRIOR
MEMBER... ORD-REMARK        0622 LINKED NEXT PRIOR
                                OPTIONAL MANUAL

SET..... ITEM                MODE IS CHAIN                ORDER IS NEXT
OWNER.... CUST-ORDER        0620 LINKED NEXT PRIOR
MEMBER... ORDER-ITEM        0621 LINKED NEXT PRIOR OWNER
                                MANDATORY AUTOMATIC

SET..... PROD-ORD           MODE IS CHAIN                ORDER IS SORTED
OWNER.... PRODUCT           0631 LINKED NEXT PRIOR
MEMBER... ORDER-ITEM        0621 LINKED NEXT PRIOR OWNER
                                OPTIONAL AUTOMATIC ASC LOT-NO=621   DUP NOT ALLOWED

```

----- SET DESCRIPTION LISTING ----- END

Figure 3-4. Sample Set Description Listing

3. An S/message. This message relates what the processor was scanning or searching for.
4. An F/message. This message relates what the processor found.
5. A D/message. This is the actual diagnostic message (what was wrong).
6. An A/message. This message indicates the action to be taken by the processor or, in some cases, the possible user action to be taken.

The second part of the listing totals the number of processor messages by severity code and issues a statement concerning the usefulness of the schema for subschema processing. Figure 3-5 shows two sample warning and error message listings. In Figure 3-5a, the printout indicates that there are no error or warning messages. Figure 3-5b lists E-level and W-level messages in the first part and an explanation in the second part.

3.7.2. Report Selection Options

The schema DDL processor generates all five reports described in 3.7.1 unless E-level errors are detected or unless input statements described as follows are specified prior to the SCHEMA DESCRIPTION paragraph name. If E-level errors are detected, the range map, record description, and set description listings are not printed.

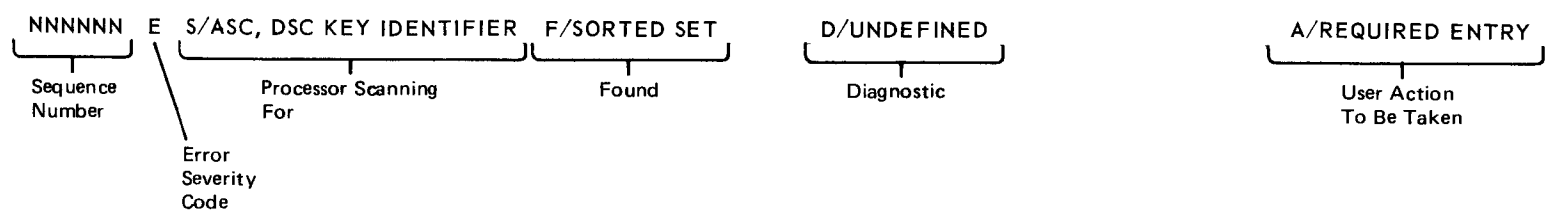
-- WARNING AND ERROR MESSAGE LISTING ---

NO ERROR AND/OR WARNING MESSAGES ISSUED
SUBSCHEMA PROCESSING ALLOWED

-- WARNING AND ERROR MESSAGE LISTING --- END

a. No error or warning messages issued

-- WARNING AND ERROR MESSAGE LISTING ---



NOTE:

In this sample listing, the schema processor was searching for the ascending or descending key identifier in the SET DESCRIPTION and did not find it. Because this is a required entry, an E-level diagnostic is produced, resulting in an unusable schema compilation.

-- WARNING AND ERROR MESSAGE LISTING ---

LINE	SEQ.	SOURCE STATEMENT	IDENT.
00023n	E	LEVEL MESSAGES ISSUED	
000018	W	LEVEL MESSAGES ISSUED	
		SUBSEQUENT JOBSTEPS CANCELED	
		SUBSCHEMA PROCESSING DISALLOWED	

-- WARNING AND ERROR MESSAGE LISTING --- END

b. E-level and W-level messages issued

Figure 3-5. Sample Warning and Error Message Listings

The following features are available to alter the report selection defaults:

- **Sequence Check Feature**

This feature may be nullified by specifying the word NOSEQ beginning in column 8.

- **E-level Error Message Listing**

To override report suppression when E-level errors occur, specify INCLST beginning in column 8 followed by a 6-character abbreviated name for each of the desired reports to be included in printing. Acceptable abbreviated report names are:

RNGMAP – for the range map listing

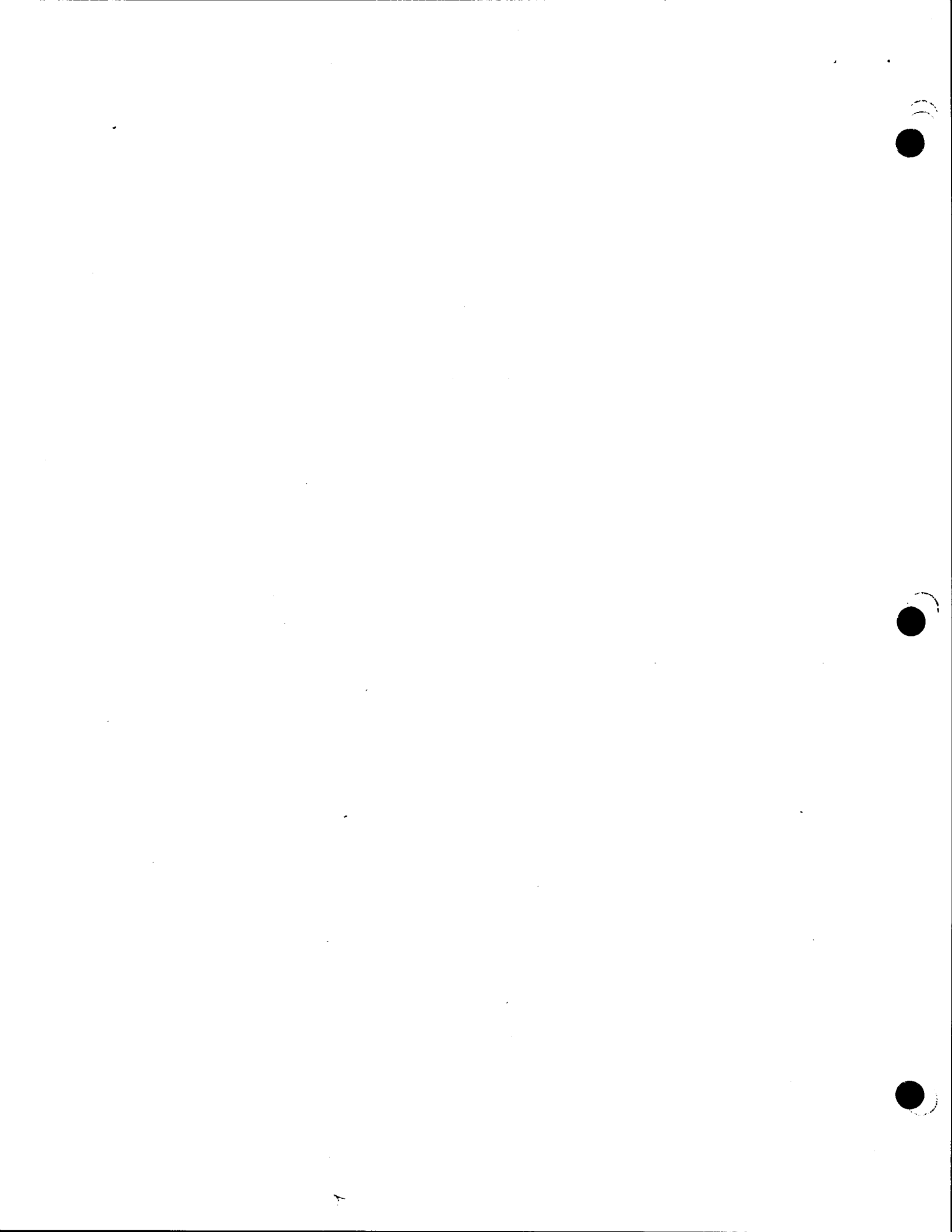
RECDES – for the record description listing

SETDES – for the set description listing

- **Suppressing Report Listings**

The programmer can suppress the listing of a range map listing, a record description listing, or a set description listing, provided no E-level severity-code errors occur. This suppression is accomplished by specifying STPLST beginning in column 8 of an input statement followed by the 6-character abbreviated name for each report to be omitted from the printing. Acceptable abbreviated report names are the same as those given in the preceding paragraph.

Examples of the use of input statements are given in Appendix D.



4. Subschema Data Description Language

4.1. GENERAL

The subschema data description language (subschema DDL) is a subset of the April 1971 CODASYL Data Base Task Group language specification, and is used to describe a portion of the total schema (selected areas, records, and sets) which is of interest to one or more application programs. In addition, the resultant subschema is in a form compatible with the application program.

The subschema DDL described in the following text is for writing a subschema compatible with application programs written in COBOL.

4.2. ORGANIZATION OF THE SUBSCHEMA DDL

The subschema DDL is composed of the following two groups of statements:

1. Subschema identification division statements
2. Subschema data division statements

Each statement group must appear in the sequence shown; each describes a specific aspect of the subschema. The latter group relies on the first for identification.

Each group begins with a divisional paragraph name. The processing of statements for a group begins with its paragraph name and is terminated by the next paragraph name or the end of the statement input stream.

In addition to DDL statements within the two groups mentioned, optional input sequence check and report control parameter statements (4.5) may be entered in the statement input stream prior to the subschema identification division statements.

4.3. SUBSCHEMA IDENTIFICATION DIVISION STATEMENTS

Function:

Subschema identification statements identify the subschema and, at the user's option, may identify the subschema author, the date of its creation, the user installation, and may present any other desired remarks.

Format:

Area A	Area A or B	Area B
-----------	----------------	-----------

SUBSCHEMA IDENTIFICATION DIVISION.

SUBSCHEMA NAME IS *subschema-name* ← [PCN 5]

OF SCHEMA NAME *schema-name.*

[AUTHOR. [comments] .]

[DATE. [comments] .]

[INSTALLATION. [comments] .]

[REMARKS. [comments] .]

Rules:

1. Subschema identification division statements begin with a paragraph name composed of the words SUBSCHEMA IDENTIFICATION DIVISION followed by a period and a space. Additional entries in the same line are ignored.
2. The SUBSCHEMA NAME sentence must immediately follow the words SUBSCHEMA IDENTIFICATION DIVISION and must be the only occurrence of the sentence within the subschema DDL input stream.
3. The remaining optional paragraphs and associated entries may appear in any desired sequence.

4.3.1. SUBSCHEMA NAME Sentence

Function:

The SUBSCHEMA NAME sentence identifies the subschema being described.

Format:

SUBSCHEMA NAME IS *subschema-name* OF SCHEMA NAME *schema-name.*

Rules:

1. The word SUBSCHEMA begins in the A area; subsequent words of the sentence ~~may~~ continue in the B area of the same ~~or subsequent lines.~~ *LINES, BUT MAY NOT CONTINUE INTO SUBSEQUENT LINES.* [PCN 5]
2. The subschema-name word must be a unique name composed of one to eight alphanumeric characters, the first of which must be a letter.

NOTE:

If a subschema-name is not unique (another subschema was previously defined and identified with the same name), the DDL processor assumes that a redescription of the subschema is desired and deletes the existing subschema replacing it with the new one. If errors are detected in the new subschema DDL, the system is left with no usable subschema identified by subschema-name.

3. The schema-name word must be the identifying name of the schema.

4.3.2. AUTHOR Paragraph**Function:**

The AUTHOR paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

AUTHOR. [comments].

4.3.3. DATE Paragraph**Function:**

The DATE paragraph may be used to specify a subschema date of creation other than the computer date. If the date override feature is not desired, the DATE paragraph must be omitted entirely and computer date will be assumed for subschema date of creation.

Format:

DATE. date-entry.

Rules:

1. The word DATE begins in the A area and must be followed by a period and a space.
2. The date-entry must accompany the DATE paragraph name and begin in the B area of the same line. The date-entry must be eight characters in the form mm/dd/yy, where mm, dd, and yy are the month, day, and year numbers, respectively.

4.3.4. INSTALLATION Paragraph**Function:**

The INSTALLATION paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

INSTALLATION.[comments].

4.3.5. REMARKS Paragraph

Function:

The REMARKS paragraph is treated as a comment entry and is presented on the processor input statement listing.

Format:

REMARKS. [comments].

4.4. SUBSCHEMA DATA DIVISION STATEMENTS

Function:

The subschema data division statements name the areas, records, and sets that are to be contained in the subschema.

Format:

Area A	Area A or B	Area B
-----------	----------------	-----------

SUBSCHEMA DATA DIVISION.

AREA SECTION.

COPY area-name AREA.

RECORD SECTION.

COPY record-name RECORD.

SET SECTION.

[COPY set-name SET.]

Rules:

1. Subschema data division statements begin with a paragraph name composed of the words SUBSCHEMA DATA DIVISION followed by a period and a space. Additional entries in the same line are ignored.
2. The remaining statements of this division are divided into three sections: AREA, RECORD, and SET. Each section is composed of a paragraph name and associated COPY statements.
3. The sections of the data division must be coded in the sequence shown, each naming component parts of the subschema, and each relying on named parts from the prior section or sections.
4. Processing of statements for a specific section begins with its associated paragraph name and is terminated by the following paragraph name or end of statement input stream.

4.4.1. AREA SECTION Statements

Function:

AREA SECTION statements are used to name those areas of the total schema that are to be included in the subschema being described. These statements comprise one AREA SECTION statement and one or more COPY AREA statements.

Format:

AREA SECTION.

COPY area-name AREA.

Rules:

1. The section begins with a paragraph name composed of the words AREA SECTION followed by a period and a space. Additional entries in the same line are ignored.
2. The COPY AREA statement names a schema area that is to be included in the subschema being described.
3. A minimum of one COPY AREA sentence must be entered for a subschema; if more than one is entered, only one sentence per input line is permitted.
4. The word COPY may begin in the A or B area.
5. The area-name word must be the name of a previously defined area of the schema.

4.4.2. RECORD SECTION Statements

Function:

RECORD SECTION statements are used to name those record types of the total schema that are to be included in the subschema being described. These statements comprise one RECORD SECTION statement and one or more COPY RECORD statements.

Format:

RECORD SECTION.

COPY record-name RECORD.

Rules:

1. The section begins with a paragraph name composed of the words RECORD SECTION followed by a period and a space. Additional entries in the same line are ignored.
2. The COPY RECORD statement names a schema record type that is to be included in the subschema being described.
3. A minimum of one COPY RECORD sentence must be entered for a subschema; if more than one is entered, only one sentence per input line is permitted.

4. The word COPY may begin in the A or B area.
5. The record-name word must be the name of a previously defined schema record type, the actual data base occurrence of which was described as being within an area named in the AREA SECTION statements.

4.4.3. SET SECTION Statements

Function:

SET SECTION statements are used to name those set types of the total schema which are to be included in the subschema being described. These statements comprise one SET SECTION statement and, at the programmer's option, one or more COPY SET statements.

Format:

SET SECTION.

[COPY set-name SET.]

Rules:

1. The section begins with a paragraph name composed of the words SET SECTION followed by a period and a space. Additional entries in the same line are ignored.
2. The SET SECTION paragraph name is required regardless of the presence or absence of COPY SET sentences.
3. The COPY SET statement names a schema set type that is to be included in the subschema being described.
4. The COPY SET statement may be omitted if the set relationships between record types are not desired or needed by the application program using the subschema. If entered, only one COPY SET sentence per input line is permitted.
5. The word COPY may begin in the A or B area.
6. The set-name word must be the name of a previously defined schema set. In addition, the owner record (3.6.4) and at least one member record (3.6.5) of the set-name type must have been named in the RECORD SECTION statements.
7. All set relationships in which a specific data base record is involved must be defined in the subschema being run, if in that run, the modification (STORE, MODIFY, DELETE) of the specific record is desired.
8. It is possible and often desirable to describe a subschema where all set relationships for a given record or all records for a given set relationship are not included in the subschema. The subschema processor recognizes either of these conditions and marks the object subschema record description as protected from run modification.

4.5. PROCESSOR REPORTS

The following is a list of reports, in order of occurrence, that can be produced during, and resulting from, the processing of subschema DDL statements. A brief description of each report is given and, where required, conditions determining the presence or absence of a report are noted.

■ Source DDL Input Listing

As input statements to the processor are processed and checked for proper sequence, a source DDL input listing is generated. Any errors detected at this point are acknowledged by a message appearing to the right of the statement in error. The sequence check feature can be overridden by specifying NOSEQ beginning in column 8 of an input statement placed before the subschema identification division paragraph name statement.

Figure 4-1 illustrates a sample subschema DDL input statement listing.

■ Errors Detected During Network Analysis

A listing of errors detected during network analysis may result if complete analysis of the defined subschema reveals that erroneous relationships between areas, records, or sets were attempted. Further processing is stopped and an error severity level associated with the subschema DDL processor is printed. There are three levels of error severity:

<u>Level</u>	<u>Description</u>
1	Warning level error; requires user investigation. Processing continues.
3	Serious level error; the compilation continues, but usually more serious errors result during subsequent processing. Execution of the object program, in general, gives unpredictable results.
9	Terminal level error; processing is halted.

If no errors or only minor errors are detected, processing would continue resulting in the creation of statements that are to be inserted by the data manipulation language (DML) preprocessor into the working storage and procedure division areas of COBOL programs invoking the named subschema.

If no errors or only minor errors are detected, the following reports are produced:

■ Subschema Data Records

This report lists all records (01 level entries) and associated group and elemental data-item statements included in the named subschema. Figure 4-2 shows a sample listing of subschema data records.

■ Record Definition Entries

This report lists all record name constant statements included in the named subschema. Figure 4-3 shows a sample listing of record definition entries.

COMPILED BY UNIVAC SERIES 90 SUBSCHEMA COMPILER VERSION 02/00 DATE 740425
SOURCE DDL INPUT LISTING.

SEQ.	SOURCE STATEMENT	IDEN.
000100		
000200	
000300	* SUBSCHEMA IDENTIFICATION STATEMENTS *	
000400	
000500		
000600	SUBSCHEMA IDENTIFICATION DIVISION.	
000700		
000800	SUBSCHEMA NAME IS DMSSUBS OF SCHEMA NAME DMSSCHM.	
000900		
001000	AUTHOR. DMS/90 PROGRAMMERS.	
001100		
001200	DATE. 04/10/73.	
001300		
001400	INSTALLATION. SPERRY*UNIVAC BLUE BELL.	
001500		
001600	REMARKS. THIS IS THE DMS/90 SAMPLE SUBSCHEMA.	
001700		
001800		
001900	
002000	* SUBSCHEMA DATA STATEMENTS *	
002100	
002200		
002300	SUBSCHEMA DATA DIVISION.	
002400		
002500	AREA SECTION.	
002600		
002700	COPY CUSTOMER-AREA AREA.	
002800	COPY ORDER-AREA AREA.	
002900	COPY PRODUCT-AREA AREA.	
003000		
003100	RECORD SECTION.	
003200		
003300	COPY CUSTOMER RECORD.	
003400	COPY CUST-ORDER RECORD.	
003500	COPY ORDER-ITEM RECORD.	
003600	COPY ORD-REMARK RECORD.	
003700	COPY PRODUCT RECORD.	
003800		
003900	SET SECTION.	
004000		
004100	COPY ORDR SET.	
004200	COPY ITEM SET.	
004300	COPY SPEC-REMARK SET.	
004400	COPY PROD-ORD SET.	
004500		

Figure 4-1. Sample Listing of Subschema DDL Input Statements

SUBSCHEMA DATA RECORDS.

```

01 CUSTOMER.
05 CUST-NO-611 PIC X(11).
05 CUST-NAME-S-611 PIC X(35).
05 CUST-ADDR-S-611 PIC X(30).
05 CUST-CREDIT PIC XXX.
05 FILLER PIC X(125).

01 CUST-ORDER.
05 FO-NO-620 PIC X(8).
05 CUST-PO-NO-620 PIC X(16).
05 FILLER PIC X(27).
05 DATE-SHIP-620 PIC X(6).
05 DATE-REQ-620 PIC X(6).
05 DATE-PROM-620 PIC X(6).
05 FILLER PIC X(133).

01 ORDER-ITEM.
05 PROD-NO-621 PIC X(12).
05 FILLER PIC XXX.
05 LOT-NO-621 PIC X(7).
05 FILLER PIC X(4).
05 QTY-ORD-621 COMP-1 PIC S9(7).
05 QTY-SHIP-621 COMP-3 PIC S9(7).
05 FILLER PIC X(34).

01 ORD-REMARK.
03 ORD-REM-CD-622.
05 REMARK-CD-622 PIC X.
05 REMARK-SEQ-622 PIC X.
05 REMARK-622 PIC X(75).
05 FILLER PIC XXX.

01 PRODUCT.
05 PROD-NO-631 PIC X(12).
05 FILLER PIC XX.
05 PROD-DES-INT-631 PIC X(15).
05 PROD-DES-EXT-631 PIC X(30).
05 FILLER PIC X(53).

```

Figure 4-2. Sample Listing of Subschema Data Records

RECORD DEFINITION ENTRIES.

```

01 DMSUBS-RECORDS.
03 SR611 PIC X(16) VALUE *CUSTOMER *.
03 SR620 PIC X(16) VALUE *CUST-ORDER *.
03 SR621 PIC X(16) VALUE *ORDER-ITEM *.
03 SR622 PIC X(16) VALUE *ORD-REMARK *.
03 SR631 PIC X(16) VALUE *PRODUCT *.

```

Figure 4-3. Sample Listing of Record Definition Entries

■ Area Definition Entries

This report lists all area name constant statements included in the named subschema. Figure 4-4 shows a sample listing of area definition entries.

AREA DEFINITION ENTRIES.

```

01 DMSUBS-AREAS.
03 CUSTOMER-AREA PIC X(16) VALUE *CUSTOMER-AREA *.
03 ORDER-AREA PIC X(16) VALUE *ORDER-AREA *.
03 PRODUCT-AREA PIC X(16) VALUE *PRODUCT-AREA *.

```

Figure 4-4. Sample Listing of Area Definition Entries

Set Definition Entries

This report lists all set name constant statements included in the named subschema. Figure 4-5 shows a sample listing of set definition entries.

SET DEFINITION ENTRIES.

```
01 DMS<UBS=SETS.
03 ORDR          PIC X(16) VALUE *ORDR          *
03 ITEM          PIC X(16) VALUE *ITEM          *
03 SPEC-REMARK  PIC X(16) VALUE *SPEC-REMARK   *
03 PROD-ORD     PIC X(16) VALUE *PROD-ORD     *
03 CALC         PIC X(16) VALUE *CALC         *
```

Figure 4-5. Sample Listing of Set Definition Entries

DMS/90 Bind Calls

This report lists all procedural call statements used in establishment of address linkage. Figure 4-6 shows a sample listing of DMS/90 bind calls.

DBMS BIND CALLS.

```
CALL 'XR7DMS' USING IDBMSCOM (59) DMSUBS-CTRL.
CALL 'XR7DMS' USING IDBMSCOM (48) SR611 CUSTOMER.
CALL 'XR7DMS' USING IDBMSCOM (1) CUST-NO-611
    SR611 CALC.
CALL 'XR7DMS' USING IDBMSCOM (48) SR620 CUST-ORDER.
CALL 'XR7DMS' USING IDBMSCOM (1) FU-NO-620
    SR620 CALC.
CALL 'XR7DMS' USING IDBMSCOM (1) FU-NO-620
    SR620 ORDRN.
CALL 'XR7DMS' USING IDBMSCOM (48) SR621 ORDER-ITEM.
CALL 'XR7DMS' USING IDBMSCOM (1) LOT-NO-621
    SR621 PROD-ORD.
CALL 'XR7DMS' USING IDBMSCOM (48) SR622 ORD-REMARK.
CALL 'XR7DMS' USING IDBMSCOM (48) SR631 PRODUCT.
CALL 'XR7DMS' USING IDBMSCOM (1) PROD-NO-631
    SR631 CALC.
```

Figure 4-6. Sample Listing of DMS/90 Bind Calls

Normally, a list of assembler input statements from the subschema DDL processor is suppressed, but, at the programmer's option, this list may be printed by specifying the words LIST NETWORK beginning in column 8 of an input statement. This input statement must be placed before the subschema identification division paragraph name statement.

The assembler input statements listing is the last report resulting from the processing of subschema DDL statements, and bears the title SUBSCHEMA subschema-name NETWORK DEFINITION TABLES.

Appendix A. DMS/90 List of Reserved Words

A.1. RESERVED WORD LIST

ACTUAL	DATA	HIGH-VALUE
ALIAS	DATA-BASE-KEY (DBKEY)	HIGH-VALUES
ALL	DATE	
ALLOWED	DECIMAL (DEC)	IDENTIFICATION (ID)
ALTER	DECODING	IDENTIFIED
ALWAYS	DEFINED	IMMATERIAL
AND	DELETE	IMPLEMENTOR
ARE	DEPENDING	IN
AREA	DESCENDING (DESC)	INDEX
AREAS	DESCRIPTION	INDEXED
AREA-CODE	DIRECT	INSERT
AREA-ID	DIRECT-DBK	INSERTION
ASCENDING (ASC)	DISPLACEMENT	INSTALLATION
AUTHOR	DISPLAY	INTO
AUTOMATIC (AUTO)	DIVISION	INVOKE
	DUPLICATE (DUP)	IS
BINARY (BIN)	DUPLICATES (DUPS)	
BIT	DURING	KEY
BY	DYNAMIC	KEYS
CALC		
CALC-KEY	EMPTY	LAST
CALL	ENCODING	LEADING
CHAIN	EQUAL	LINKED
CHANGED	ERROR	LOCATION (LOC)
CHARACTER (CHAR)	EXCLUSIVE (EXCL)	LOCK
CHECK		LOCKS
CLOSE	FILLER	LOW-VALUE
COMMENT	FIND	LOW-VALUES
COMPILE	FIRST	
COMPLEX	FIXED	MANDATORY (MAND)
COMPUTATIONAL (COMP)	FLOAT	MANUAL
COMPUTATIONAL-1 (COMP-1)	FOR	MEMBER
COMPUTATIONAL-2 (COMP-2)	FROM	MEMBERS
COMPUTATIONAL-3 (COMP-3)		MODE
COPY	GET	MODIFY
CURRENT		MOVE

NAME	REAL	TEMPORARY (TEM)
NEXT	RECORD	THEN
NON-EXCLUSIVE (NEXCL)	RECORD-NAME	THIS
NOT	REDEFINES	THROUGH
NULL	REMARKS	THRU
	REMOVE	TIMES
OBTAIN	RESULT	TO
OCCURRENCE	RETRIEVAL (RETR)	TRAILING
OCCURS	RUN-UNIT	TYPE
OF		
ON	SCHEMA	UPDATE
ONLY	SEARCH	USAGE
OPEN	SECTION	USAGE-MODE
OPTIONAL (OPT)	SELECTION	USING
OR	SELECTIVE	
ORDER	SEPARATE	VALUE
OWNER	SET	VALUES
	SETS	VIA
PAGES	SIGN	VIRTUAL
PERMANENT	SORTED	
PICTURE (PIC)	SOURCE	WHERE
POINTER-ARRAY (PTR)	SPACE	WITHIN
POSITION	SPACES	
PRIOR	SR1	ZERO
PRIVACY	SR6	ZEROS
PROCEDURE (PROC)	STATUS	ZEROES
PROCESSABLE	STORE	
PROTECTED (PROT)	SUBSCHEMA	
	SYNCHRONIZED (SYNC)	
RANGE	SYSTEM	

A.2. RESTRICTED DATA NAMES AND PROCEDURE NAMES

The following words are restricted from arbitrary usage by the user. The words are data names and procedure names with special meanings described in the data manipulation language programmer reference manual, UP-8036 (current version).

AREA-NAME
 CURRENCY
 DMS-ABORT
 DMS-STATUS
 DMS-SUCCESS
 ERROR-AREA
 ERROR-RECORD
 ERROR-SET
 ERROR-STATUS

Appendix B. Schema DDL Syntax

Area A	Area A or B	Area B
-----------	----------------	-----------

SCHEMA DESCRIPTION.

SCHEMA NAME IS schema-name

[RANGE IS integer-1 THRU integer-2].

[AUTHOR. [comments].]

[DATE. date-entry.]

[INSTALLATION. [comments].]

[REMARKS. [comments].]

AREA DESCRIPTION.

AREA NAME IS area-name

RANGE IS integer-1 THRU integer-2.

Area A	Area A or B	Area B
-----------	----------------	-----------

RECORD DESCRIPTION.

RECORD NAME IS record-name.

RECORD ID IS integer.

LOCATION MODE IS

{ DIRECT
CALC USING identifier DUPLICATES ARE { FIRST
VIA set-name SET { LAST
NOT ALLOWED } } ±

WITHIN area-name AREA.

level-number { data-name-1
FILLER }

[REDEFINES data-name-2]

[USAGE IS { DISPLAY
COMP
COMPUTATIONAL
COMP-1
COMPUTATIONAL-1
COMP-2
COMPUTATIONAL-2
COMP-3
COMPUTATIONAL-3 }]

{ SYNCHRONIZED
SYNC }

{ PICTURE } IS character-string
PIC

[VALUE IS literal]

[OCCURS integer TIMES]

[INDEXED BY index-name].

Area A	Area A or B	Area B
-----------	----------------	-----------

SET DESCRIPTION.

SET NAME IS set-name.

ORDER IS {
FIRST
LAST
NEXT
PRIOR
SORTED
}

MODE IS CHAIN [LINKED TO PRIOR].

OWNER IS record-name

NEXT DBKEY POSITION IS integer-1

[PRIOR DBKEY POSITION IS integer-2].

MEMBER IS record-name

{MANDATORY} {AUTOMATIC}
{OPTIONAL} {MANUAL}

[LINKED TO OWNER]

NEXT DBKEY POSITION IS integer-1

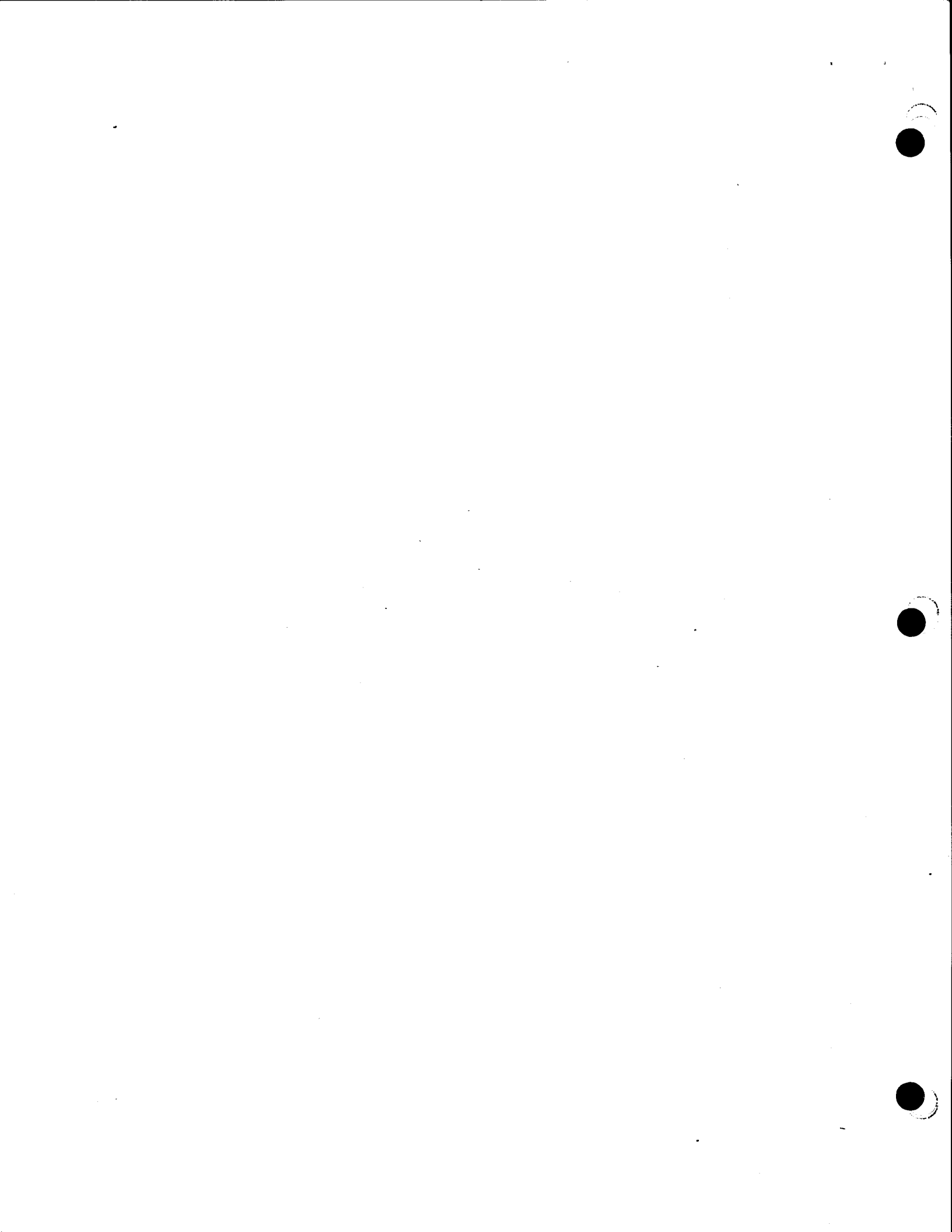
[PRIOR DBKEY POSITION IS integer-2]

[OWNER DBKEY POSITION IS integer-3]

{ASCENDING} KEY IS identifier
{DESCENDING}

DUPLICATES ARE {FIRST
LAST
NOT ALLOWED}





Appendix C. Subschema DDL Syntax

Area A	Area A or B	Area B
-----------	----------------	-----------

SUBSCHEMA IDENTIFICATION DIVISION.

SUBSCHEMA NAME IS subschema-name

OF SCHEMA NAME schema-name.

[AUTHOR.] [comments].]

[DATE.] [comments].]

[INSTALLATION.] [comments].]

[REMARKS.] [comments].]

SUBSCHEMA DATA DIVISION.

AREA SECTION.

COPY area-name AREA.

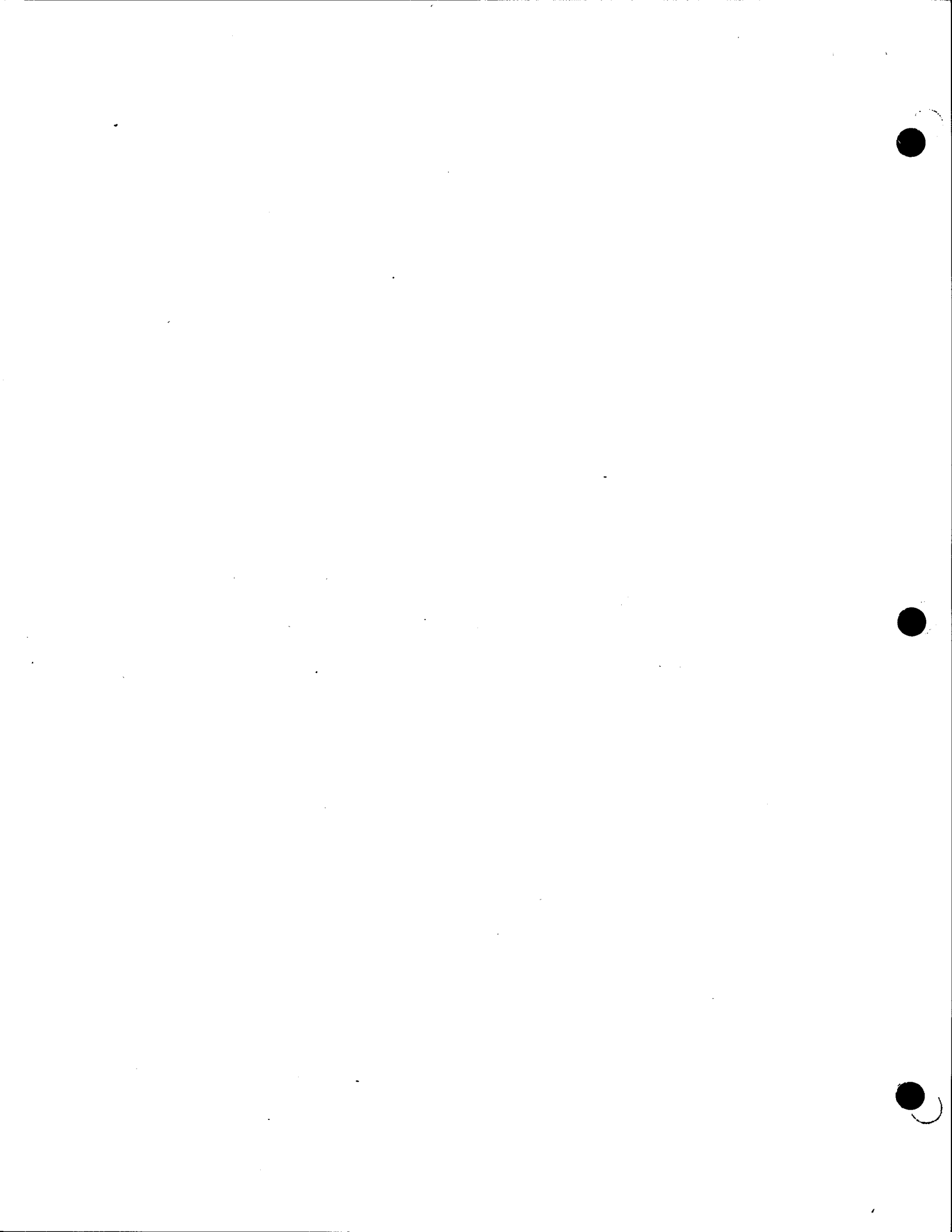
RECORD SECTION.

COPY record-name RECORD.

SET SECTION.

[COPY set-name SET.]





Appendix D. Sample Schema and Subschema Definitions

D.1. GENERAL

Sample schema (DMSSCHM) and subschema (DMSSUBS) data definitions for an inventory-oriented data base are shown in this appendix. The sample data manipulation language (DML) program appropriate to the subschema is shown and explained in the data manipulation language programmer reference manual, UP-8036 (current version).



D.2. THE SCHEMA DEFINITION (DMSSCHM)

```

000100 INCLST RNGMAP RECDER SETDES
000200*
000300* .....
000400*          SCHEMA DESCRIPTION STATEMENTS          *
000500* .....
000600*
000700 SCHEMA DESCRIPTION.
000800
000900 SCHEMA NAME IS DMSSCHM RANGE IS          502 THRU          540.
001000
001100 AUTHOR.          DMS/90 PROGRAMMERS.
001200
001300 DATE.          04/10/73.
001400
001500 INSTALLATION. SPERRY•UNIVAC BLUE BELL.
001600
001700 REMARKS.          THIS IS THE DMS/90 SAMPLE SCHEMA.
001800
001900*
002000* .....
002100*          AREA DESCRIPTION STATEMENTS          *
002200* .....
002300*
002400 AREA DESCRIPTION.
002500
002600 AREA NAME IS CUSTOMER-AREA RANGE IS          502 THRU          510.
002700 AREA NAME IS ORDER-AREA RANGE IS          511 THRU          530.
002800 AREA NAME IS PRODUCT-AREA RANGE IS          531 THRU          540.
002900
003000*
003100* .....
003200*          RECORD DESCRIPTION STATEMENTS          *
003300* .....
003400*
003500 RECORD DESCRIPTION.
003600
003700 RECORD NAME CUSTOMER          .
003800 RECORD ID 611 .
003900 LOCATION MODE CALC USING CUST-NO-611          DUPLICATES NOT ALLOWED.
004000 WITHIN          CUSTOMER-AREA AREA,
004100
004200 05 CUST-NO-611          PIC X(11).
004300 05 CUST-NAME-S-611          PIC X(35).
004400 05 CUST-ADDR-S-611          PIC X(30).
004500 05 CUST-CREDIT          PIC XXX.
004600 05 FILLER          PIC X(125).
004700
004800 RECORD NAME CUST-ORDER          .
004900 RECORD ID 620.
005000 LOCATION MODE CALC USING FO-NO-620          DUPLICATES NOT ALLOWED.
005100 WITHIN          ORDER-AREA AREA,

```

```

005200
005300      05  FO-NO-620                PIC X(8).
005400      05  CUST-PO-NO-620          PIC X(18).
005500      05  FILLER                  PIC X(27).
005600      05  DATE-SHIP-620           PIC X(6).
005700      05  DATE-REQ-620            PIC X(6).
005800      05  DATE-PROM-620           PIC X(6).
005900      05  FILLER                  PIC X(133).
006000
006100 RECORD  NAME ORDER-ITEM
006200 RECORD  ID   621 .
006300 LOCATION MODE VIA ITEM          SET.
006400 WITHIN   ORDER-AREA            AREA.
006500
006600      05  PROD-NO-621              PIC X(12).
006700      05  FILLER                  PIC XXX.
006800      05  LOT-NO-621               PIC X(7).
006900      05  FILLER                  PIC X(4).
007000      05  QTY-ORD-621              COMP-3  PIC S9(7).
007100      05  QTY-SHIP-621             COMP-3  PIC S9(7).
007200      05  FILLER                  PIC X(34).
007300
007400 RECORD  NAME ORD-REMARK
007500 RECORD  ID   622 .
007600 LOCATION MODE DIRECT.
007700 WITHIN   ORDER-AREA            AREA.
007800
007900      03  ORD-REM-CD-622.
008000      05  REMARK-CD-622            PIC X.
008100      05  REMARK-SEQ-622            PIC X.
008200      05  REMARK-622               PIC X(75).
008300      05  FILLER                  PIC XXX.
008400
008500 RECORD  NAME PRODUCT
008600 RECORD  ID   631 .
008700 LOCATION MODE CALC USING PROD-NO-631  DUPLICATES NOT ALLOWED.
008800 WITHIN   PRODUCT-AREA            AREA.
008900
009000      05  PROD-NO-631              PIC X(12).
009100      05  FILLER                  PIC XX.
009200      05  PROD-DES-INT-631           PIC X(15).
009300      05  PROD-DES-EXT-631          PIC X(30).
009400      05  FILLER                  PIC X(53).
009500
009600.
009700. ....
009800. *          SET DESCRIPTION STATEMENTS          .
009900. ....
010000.
010100 SET DESCRIPTION.
010200
010300 SET NAME ORDOR

```



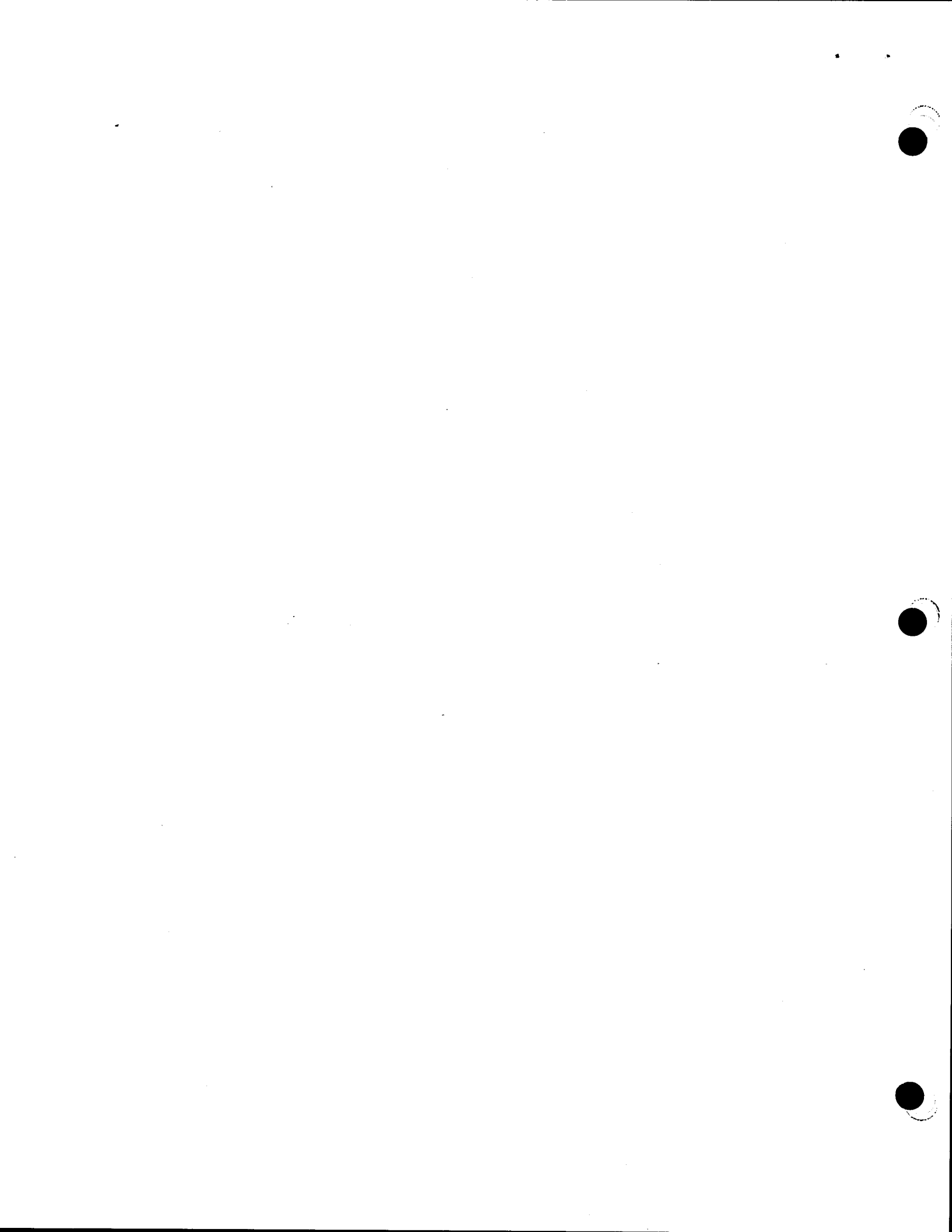
D.3. THE SUBSCHEMA DEFINITION (DMSSUBS)

```

000100*
000200* .....
000300*      SUBSCHEMA IDENTIFICATION STATEMENTS      *
000400* .....
000500*
000600 SUBSCHEMA IDENTIFICATION DIVISION.
000700
000800 SUBSCHEMA NAME IS DMSSUBS OF SCHEMA NAME DMSSCHM.
000900
001000 AUTHOR.          DMS/90 PROGRAMMERS.
001100
001200 DATE.           04/10/73.
001300
001400 INSTALLATION.   SPERRY*UNIVAC   BLUE BELL.
001500
001600 REMARKS.        THIS IS THE DMS/90 SAMPLE SUBSCHEMA.
001700
001800*
001900* .....
002000*      SUBSCHEMA DATA STATEMENTS      *
002100* .....
002200*
002300 SUBSCHEMA DATA DIVISION.
002400
002500 AREA SECTION.
002600
002700     COPY CUSTOMER-AREA   AREA.
002800     COPY ORDER-AREA     AREA.
002900     COPY PRODUCT-AREA   AREA.
003000
003100 RECORD SECTION.
003200
003300     COPY CUSTOMER       RECORD.
003400     COPY CUST-ORDER    RECORD.
003500     COPY ORDER-ITEM    RECORD.
003600     COPY ORD-REMARK    RECORD.
003700     COPY PRODUCT       RECORD.
003800
003900 SET SECTION.
004000
004100     COPY ORDOR         SET.
004200     COPY ITEM         SET.
004300     COPY SPEC-REMARK   SET.
004400     COPY PROD-ORD     SET.
004500

```





Appendix E. DMS/90 Clock Rule

E.1. GENERAL

Record occurrences in a DMS/90 data base must be linked together as specified in the data base schema. An occurrence of a record type has two portions, control and data. The control portion is under the exclusive control of DMS/90. It is not accessible to any application program. It contains links in the form of 1-word data base keys that point to other record occurrences. DMS/90 maintains the links in response to DML requests made by application programs and in accordance with the description of the data base contained in object subschemas.

E.2. ASSIGNING DATA BASE KEY POSITION NUMBERS

The location of each link within the control portion of a given record type must be specified in the schema by using the DBKEY POSITION clauses in the OWNER and MEMBER sentences of the set entries.

The following rules must be observed when assigning data base key position numbers for a given record type:

1. The numbers must be unique within record type.
2. They must be assigned as consecutive integers beginning with 1 and ending with a number less than 100.
3. All sets within which the record type is a member must be considered first. The sets within this group may be considered in any order. For a given set, data base key position numbers must be assigned in the order N, NP, NO, or NPO, depending upon which linkage option is required.
4. Once positions have been assigned for all sets in item 3, then all sets within which the record type is an owner must be considered. The sets within this group may be considered in any order. For a given set, data base key position numbers must be assigned in the N or NP order, depending upon which linkage option is required. An owner data base key position is never assigned for the owner record type.
5. Data base key positions associated with the CALC location mode are accounted for by the system and should not be numbered in the schema.

It may be convenient to use the clock rule to assign data base key positions. The clock rule is a procedure that uses the network data structure diagram (refer to the data manipulation language manual, UP-8036 (current version)), an example of which is shown in Figure E-1. This procedure is a systematic way of following the preceding rules.

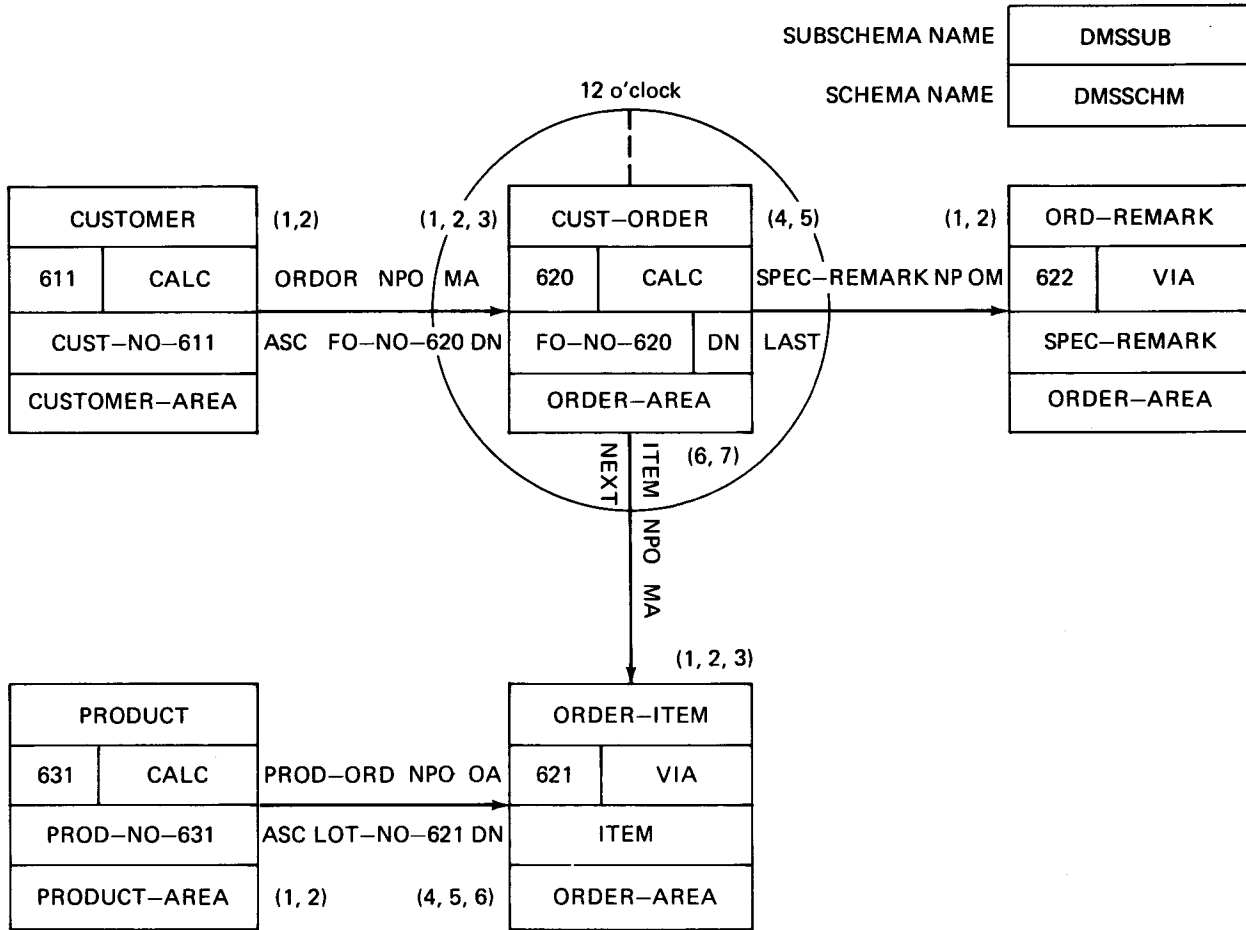


Figure E-1. Clock Rule Example for Assigning Data Base Key Position Numbers

Consider the problem of assigning data base key position numbers for the CUST-ORDER record type shown in Figure E-1. Imagine the record representation superimposed on the face of a clock. Beginning at the 12 o'clock position and moving clockwise, the clock is circled twice, the first time to number all data base key positions for sets in which the record participates as a member, and the second time to number all data base key positions for sets in which the record participates as an owner. The first data base key position encountered is assigned the number 1, others being consecutively numbered from 1, incrementing by 1 for each.

Returning to Figure E-1, note that the CUST-ORDER record is a member of the ORDOR set, and NEXT, PRIOR, and OWNER (NPO) linkage is specified to which the numbers 1, 2, and 3 are assigned the first time around the clock. Continuing around the clock, the second time, for owner linkages the numbers 4 and 5 are assigned to the NEXT and PRIOR (NP) linkage required for the SPEC-REMARK set, and the numbers 6 and 7 are assigned to the NEXT and PRIOR linkage required for the ITEM set. No number is assigned for the OWNER linkage of the ITEM set because that data base key position is only defined in the ORDER-ITEM member record, i.e., the owner does not need to be linked to itself.

Figure E-1 is a network data base diagram of the DMS/90 sample schema/subschema. Schema DDL statements and resultant calculations from the clock rule as applied to the sample can be found in the processor report examples in Appendix D.2.

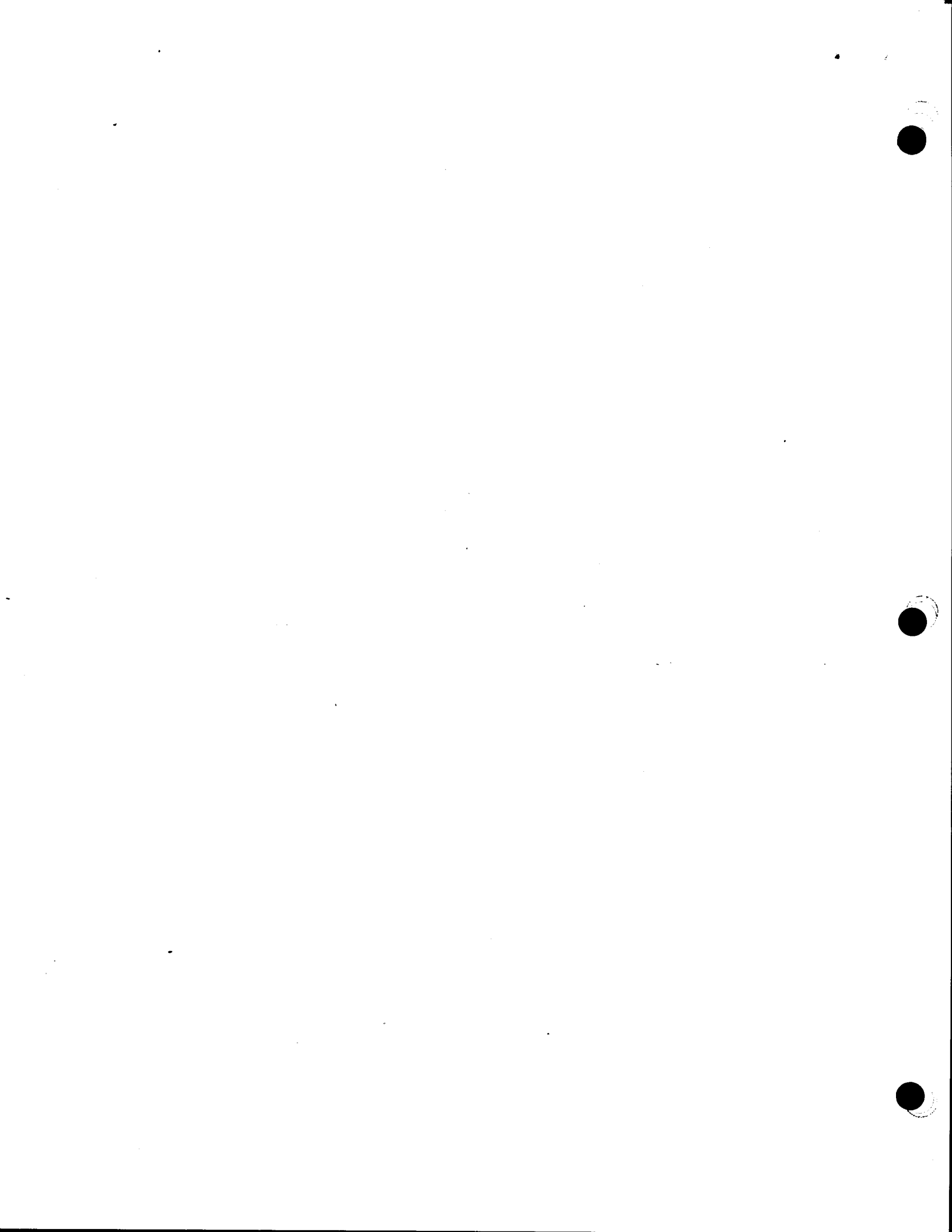
Index

Term	Reference	Page	Term	Reference	Page
A			C		
"A" area	Table 2-1	2-4	CALC control identifier	3.7.1	3-21
A/message	3.7.1	3-23	Categories of data		
Alphanumeric items	3.5.5.5	3-12	alphanumeric	3.5.5.5	3-12
Area definition entries			numeric	3.5.5.5	3-12
description	4.5	4-9	COBOL programming form	Figure 2-1	2-3
sample listing	Figure 4-4	4-9	COMMENTS	Table 2-1	2-4
Area description statements			Computational data item	3.5.5.3	3-11
AREA NAME sentence	3.4.1	3-4	CONTINUATION	Table 2-1	2-4
format	3.4	3-4	Control identifier	3.7.1	3-21
function	3.4	3-4	COPY RECORD statement	4.4.2	4-5
AREA NAME sentence			COPY SET statement	4.4.3	4-6
page range definition	3.4.1	3-4	D		
total schema	3.4.1	3-4	D/message	3.7.1	3-23
Area of storage, defining	3.5	3-5	Data base key positions	3.6.4	3-18
Assembler input statement list	4.5	4-10		3.7.1	3-21
AUTHOR paragraph	3.3.2	3-3	Data base keys	3.6.4	3-18
	4.3.2	4-3		3.7.1	3-21
B			Data description language (DDL)		
"B" area	Table 2-1	2-4	structure		
Braces	1.2	1-2	character set	2.1.1	2-1
Brackets	1.2	1-2	types of words	2.1.3	2-2
			words	2.1.2	2-2

Term	Reference	Page	Term	Reference	Page
Data item			DMSSCHM	D.2	D-2
computational	3.5.5.3	3-11	DMSSUBS	D.3	D-5
elementary	3.5.3	3-8	Duplicate control statement	3.7.1	3-21
	3.5.5	3-8	Duplicate records	3.5.3	3-8
	3.5.5.5	3-11			
	3.5.5.6	3-13			
group	3.5.3	3-8			
	3.5.5	3-8			
Data-item sentence			E		
format	3.5.5	3-9	E-level error message listing	3.7.2	3-25
function	3.5.5	3-8	Elementary data item	3.5.3	3-8
INDEXED clause	3.5.5.8	3-14		3.5.5	3-8
level number and data name/FILLER	3.5.5.1	3-9		3.5.5.5	3-11
OCCURS clause	3.5.5.7	3-13		3.5.5.6	3-13
PICTURE clause	3.5.5.5	3-11	Ellipsis	1.2	1-2
record description statement rule	3.5	3-6	Error severity code	3.7.1	3-21
REDEFINES clause	3.5.5.2	3-10	Error severity level	4.5	4-7
SYNCHRONIZED clause	3.5.5.4	3-11	Errors detected during network analysis	4.5	4-7
USAGE clause	3.5.5.3	3-10			
VALUE clause	3.5.5.6	3-13			
Data name	3.5.5.1	3-9	F		
Date-entry word	3.3.3	3-3	F/message	3.7.1	3-23
	4.3.3	4-3	Figurative constants	3.5.5.6	3-13
DATE paragraph	3.3.3	3-3	FILLER	3.5.5.1	3-9
	4.3.3	4-3	Fixed-point numeric items	3.5.5.5	3-12
DDL character set	2.1.1	2-1			
DDL coding form			G		
COBOL programming	Figure 2-1	2-3	Group data item	3.5.3	3-8
programming form column usage	Table 2-1	2-4			
DDL organization	2.2	2-3	I		
DDL reserved words	1.2	1-2	INDEXED clause	3.5.5.8	3-14
DDL words	2.1.2	2-2	Input sequence check	3.2	3-1
DMS/90 reserved words	Appendix A			4.2	4-1
Default option	1.2	1-2	INSTALLATION paragraph	3.3.4	3-3
Diagnostic message	3.7.1	3-23		4.3.4	4-3
Divisional paragraph name	4.2	4-1	Internal decimal format	3.5.5.3	3-11
DMS/90 bind calls					
description	4.5	4-10			
sample listing	4.5	4-10			

Term	Reference	Page	Term	Reference	Page
Internal floating-point format	3.5.5.3	3-11			
Inventory-oriented data base	D.1	D-1			
	K				
Key words	1.2	1-2			
	L				
Language elements	1.2	1-1			
Level number	3.5.5.1	3-9			
Linkage	3.7.1	3-21			
LOCATION MODE sentence	3.5.3	3-7			
Lowercase words	1.2	1-2			
	M				
MEMBER sentence	3.6.5	3-18			
Minor errors	4.5	4-7			
Mode of storage, defining	3.5	3-5			
MODE sentence	3.6.3	3-17			
	N				
Nonnumeric literals	3.5.5.6	3-13			
Notations	1.2	1-1			
Numeric items					
fixed-point	3.5.5.5	3-12			
floating-point	3.5.5.5	3-12			
Numeric literals	3.5.5.6	3-13			
	O				
OCCURS clause	3.5.5.7	3-13			
Optional functions	1.2	1-2			
ORDER sentence	3.6.2	3-16			
OWNER sentence	3.6.4	3-18			
			P		
			Packed decimal format	3.5.5.3	3-11
			Page range definition	3.4.1	3-4
			Periods	1.2	1-2
			PICTURE clause	3.5.5.5	3-11
			Processor-assigned sequence number	3.7.1	3-19
			Processor report control	3.2	3-1
			Processor reports		
			area definition entries	4.5	4-9
			DMS/90 bind calls	4.5	4-10
			errors detected during		
			network analysis	4.5	4-7
			record definition entries	4.5	4-7
			record description listing	3.7.1	3-21
			schema DDL input statement		
			listing	3.7.1	3-19
			set definition entries	4.5	4-10
			set description listing	3.7.1	3-21
			source DDL input listing	4.5	4-7
			subschema data records	4.5	4-7
			warning and error message		
			listing	3.7.1	3-21
			R		
			RANGE clause	3.3.1	3-2
			Range map listing		
			description	3.7.1	3-19
			RNGMAP	3.7.2	3-25
			sample	Figure 3-2	3-21
			RECDES	3.7.2	3-25
			Record data content, defining	3.5	3-5
			Record definition entries		
			description	4.5	4-7
			sample listing	Figure 4-3	4-9
			Record description listing		
			description	3.7.1	3-21
			RECDES	3.7.2	3-25
			sample	Figure 3-3	3-22

Term	Reference	Page	Term	Reference	Page
Record description statements			Schema description statements		
data-item sentence	3.5.5	3-8	AUTHOR paragraph	3.3.2	3-3
format	3.5	3-5	DATE paragraph	3.3.3	3-3
function	3.5	3-5	INSTALLATION paragraph	3.3.4	3-3
LOCATION MODE sentence	3.5.3	3-7	RANGE clause	3.3.1	3-2
RECORD ID sentence	3.5.2	3-7	REMARKS paragraph	3.3.5	3-4
RECORD NAME sentence	3.5.1	3-6	schema	3.3	3-1
WITHIN sentence	3.5.4	3-8	SCHEMA NAME sentence	3.3.1	3-2
RECORD ID sentence	3.5.2	3-7	schema page range	3.3	3-1
RECORD NAME sentence	3.5.1	3-6	SCHEMA NAME sentence	3.3.1	3-2
RECORD SECTION statements	4.4.2	4-5	Schema page range	3.3	3-1
REDEFINES clause	3.5.5.2	3-10	Schema sample definition	D.2	D-2
REMARKS paragraph	3.3.5	3-4	Sequence check feature	3.7.2	3-25
	4.3.5	4-4		4.5	4-7
Report control	4.2	4-1	SEQUENCE NUMBER	Table 2-1	2-4
Report listings, suppressing	3.7.2	3-25	Set definition entries		
Report selection options			description	4.5	4-10
E-level error message			sample listing	Figure 4-5	4-10
listing	3.7.2	3-25	Set description listing		
sequence check feature	3.7.2	3-25	description	3.7.1	3-21
suppressing report listings	3.7.2	3-25	sample	Figure 3-4	3-23
RNGMAP	3.7.2	3-25	SETDES	3.7.2	3-25
Rules	1.2	1-1	Set description statements		
			format	3.6	3-15
			function	3.6	3-14
			MEMBER sentence	3.6.5	3-18
			MODE sentence	3.6.3	3-17
			ORDER sentence	3.6.2	3-16
			OWNER sentence	3.6.4	3-18
			SET NAME sentence	3.6.1	3-16
S/message	3.7.1	3-23	Set membership type parameters	3.6.5	3-19
Schema DDL input statement			SET NAME sentence	3.6.1	3-16
listing			SET SECTION statements	4.4.3	4-6
description	3.7.1	3-19	SETDES	3.7.2	3-25
sample	Figure 3-1	3-20	Short-precision format	3.5.5.3	3-11
Schema DDL organization			Source DDL input listing	4.5	4-7
area description	3.2	3-1		Figure 4-1	4-8
schema description	3.2	3-1	Statement input stream	4.2	4-1
record type description	3.2	3-1			
set type description	3.2	3-1			
Schema DDL syntax	Appendix B				
Schema definition	1.1	1-1			



Comments concerning this manual may be made in the space provided below. Please fill in the requested information.

System: _____

Manual Title: _____

UP No: _____ Revision No: _____ Update: _____

Name of User: _____

Address of User: _____

Comments:

CUT



CUT



FIRST CLASS
PERMIT NO. 21
BLUE BELL, PA.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

SPERRY  **UNIVAC**

P.O. BOX 500
BLUE BELL, PA.
19422

ATTN: SYSTEMS PUBLICATIONS DEPT.



FOLD

FOLD

2. 4. 6.



