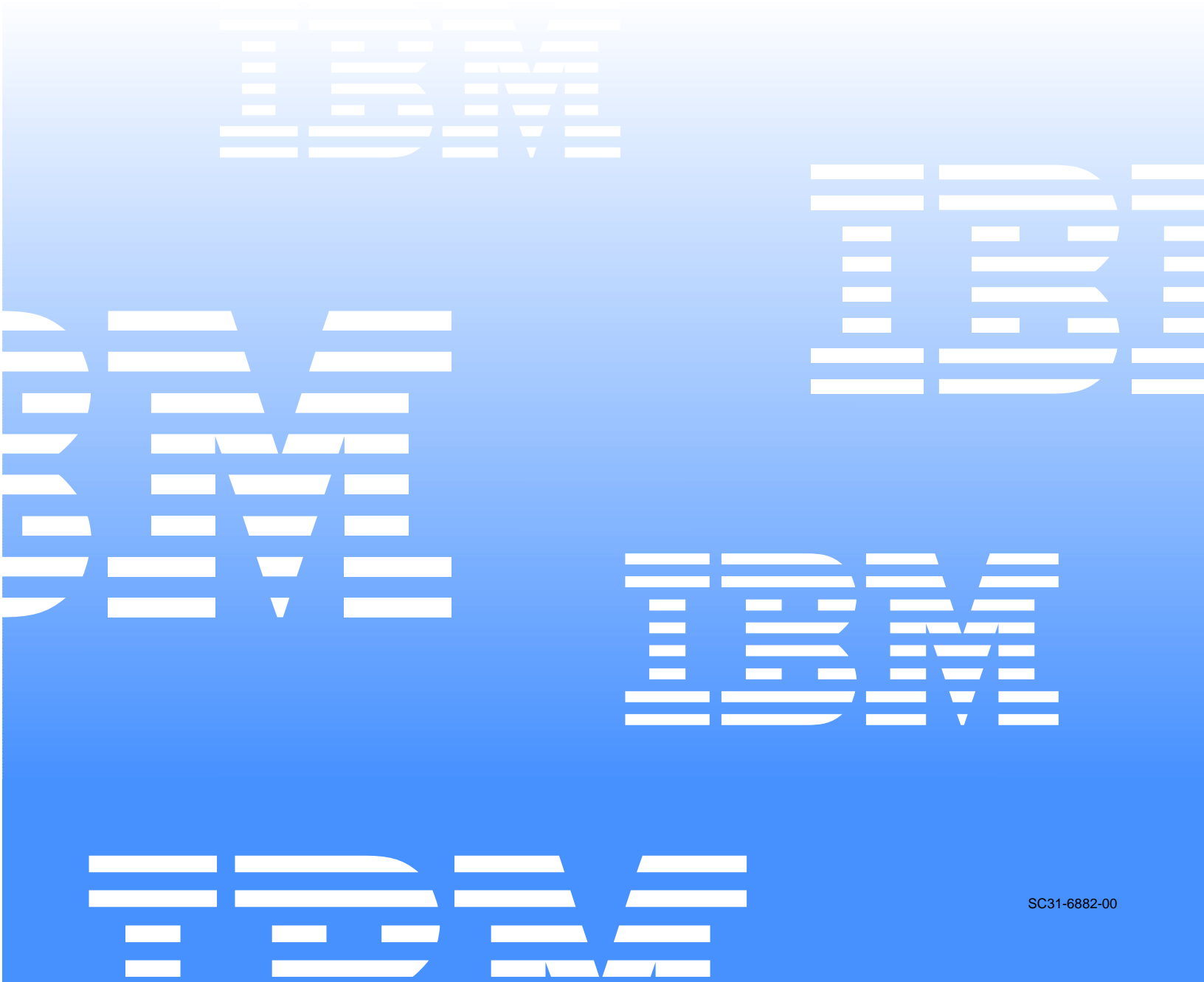


IBM Asset Transformation Workbench  
v1.1



# Parser Reference Manual





IBM Asset Transformation Workbench  
v1.1



# Parser Reference Manual

Note:

Before using this information and the product it supports, read the information in “Notices.”

**First Edition (February 2005)**

This edition applies to IBM Asset Transformation Workbench (product number 5724-L54) and to all subsequent releases and modifications until otherwise indicated in new editions.

For the latest information about this product, please refer to the Release Notes.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of IBM. Information in this document is subject to change without notice and is not guaranteed to be error-free.

You can order publications through your IBM representative or the IBM branch office serving your locality. When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Licensed Materials - Property of IBM.

Product Reference: IBM Asset Transformation Workbench v1.1

Document Reference: REL7.3.07.DOC08.A

**© 2005 Copyright International Business Machines Corporation. All rights reserved.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

**© 2004, 2005 Relativity Technologies, Inc. All rights reserved.**

RescueWare is a registered trademark of Relativity Technologies, Inc. All other brands mentioned in this document are trademarks or registered trademarks of their respective holders.

# Contents

## Preface

<i>Audience</i> . . . . .	xi
<i>Organization</i> . . . . .	xii
<i>Conventions</i> . . . . .	xiii
<i>Related Manuals</i> . . . . .	xiii
<i>Online Help</i> . . . . .	xiv

## 1 Supported COBOL Statements

<i>Key to tables</i> . . . . .	1-1
<i>Supported VS COBOL II statements</i> . . . . .	1-4
<i>COBOL language structure</i> . . . . .	1-4
<i>Literals</i> . . . . .	1-5
<i>Referencing names</i> . . . . .	1-6
<i>COBOL program structure</i> . . . . .	1-8
<i>Identification Division</i> . . . . .	1-8
<i>Environment Division</i> . . . . .	1-9
<i>Data Division</i> . . . . .	1-11
<i>Procedure Division</i> . . . . .	1-14
<i>Compile-directing statements and directives</i> . . . . .	1-28

- Supported COBOL for OS/390 statements. . . . . 1-30
  - COBOL language structure . . . . . 1-30
  - Literals . . . . . 1-32
  - Referencing names . . . . . 1-33
  - COBOL program structure . . . . . 1-34
  - Identification Division . . . . . 1-35
  - Environment Division . . . . . 1-36
  - Data Division . . . . . 1-38
  - Procedure Division . . . . . 1-42
  - Intrinsic functions . . . . . 1-59
  - Compile-directing statements and directives . . . . . 1-61
- Supported COBOL/400 statements . . . . . 1-63
  - COBOL language structure . . . . . 1-63
  - Literals . . . . . 1-64
  - Referencing names . . . . . 1-64
  - COBOL program structure . . . . . 1-65
  - Identification Division . . . . . 1-66
  - Environment Division . . . . . 1-66
  - Data Division . . . . . 1-68
  - Procedure Division . . . . . 1-72
  - Compile-directing statements and directives . . . . . 1-89
- Supported Unisys ASCII COBOL statements. . . . . 1-90
  - COBOL language structure . . . . . 1-90
  - Literals . . . . . 1-91
  - Referencing names . . . . . 1-91
  - Control Division . . . . . 1-92
  - Identification Division . . . . . 1-93
  - Environment Division . . . . . 1-93
  - Data Division . . . . . 1-97
  - Procedure Division . . . . . 1-105
- Supported Unisys UCS COBOL statements. . . . . 1-119
  - COBOL language structure . . . . . 1-119
  - Literals . . . . . 1-120
  - Referencing names . . . . . 1-120
  - COBOL program structure . . . . . 1-122
  - Control Division . . . . . 1-122
  - Identification Division . . . . . 1-122

<i>Environment Division</i> . . . . .	1-123
<i>Data Division</i> . . . . .	1-126
<i>Procedure Division</i> . . . . .	1-133
<i>Intrinsic functions</i> . . . . .	1-146
<i>Unisys ASCII/UCS implementation and limitations</i> . . . . .	1-148
<i>Data Types summary</i> . . . . .	1-156
<i>Supported Unisys CDML COBOL statements</i> . . . . .	1-156
<i>Data Division</i> . . . . .	1-156
<i>Procedure Division</i> . . . . .	1-158
<i>Update Commands</i> . . . . .	1-161
<i>Support Commands</i> . . . . .	1-162
<i>Unisys support</i> . . . . .	1-162
<i>Supported Micro Focus COBOL statements</i> . . . . .	1-165
<i>COBOL language structure</i> . . . . .	1-165
<i>Literals</i> . . . . .	1-166
<i>Referencing names</i> . . . . .	1-166
<i>Identification Division</i> . . . . .	1-168
<i>Environment Division</i> . . . . .	1-168
<i>Data Division</i> . . . . .	1-171
<i>Procedure Division</i> . . . . .	1-176
<i>Intrinsic functions</i> . . . . .	1-198
<i>COBOL language structure</i> . . . . .	1-201
<i>Literals</i> . . . . .	1-202
<i>COBOL program structure</i> . . . . .	1-202
<i>Identification Division</i> . . . . .	1-203
<i>Environment Division</i> . . . . .	1-204
<i>Data Division</i> . . . . .	1-205
<i>Procedure Division</i> . . . . .	1-208
<i>Supported HP COBOL II/XL statements</i> . . . . .	1-218
<i>COBOL language structure</i> . . . . .	1-218
<i>Literals</i> . . . . .	1-219
<i>Referencing names</i> . . . . .	1-219
<i>COBOL program structure</i> . . . . .	1-221
<i>Identification Division</i> . . . . .	1-221
<i>Environment Division</i> . . . . .	1-221
<i>Data Division</i> . . . . .	1-224
<i>Procedure Division</i> . . . . .	1-226

<i>Intrinsic functions</i> . . . . .	1-240
<i>Preprocessor commands</i> . . . . .	1-242
<i>Supported Fujitsu COBOL85 (M Series) statements</i> . . . . .	1-244
<i>COBOL language structure</i> . . . . .	1-244
<i>Literals</i> . . . . .	1-246
<i>Referencing names</i> . . . . .	1-247
<i>COBOL program structure</i> . . . . .	1-248
<i>Identification Division</i> . . . . .	1-248
<i>Environment Division</i> . . . . .	1-248
<i>Data Division</i> . . . . .	1-252
<i>Procedure Division</i> . . . . .	1-257
<i>Intrinsic functions</i> . . . . .	1-271
<i>Compile-directing statements and directives</i> . . . . .	1-271
<i>Supported AIM Network Database Statements</i> . . . . .	1-273
<i>Supported SIEMENS COBOL statements</i> . . . . .	1-275
<i>COBOL language structure</i> . . . . .	1-275
<i>Literals</i> . . . . .	1-277
<i>Referencing names</i> . . . . .	1-278
<i>SIEMENS COBOL program structure</i> . . . . .	1-279
<i>Identification Division</i> . . . . .	1-280
<i>Environment Division</i> . . . . .	1-281
<i>Data Division</i> . . . . .	1-284
<i>Procedure Division</i> . . . . .	1-289
<i>Intrinsic functions</i> . . . . .	1-306
<i>Compile-directing statements and directives</i> . . . . .	1-308

## 2 Supported Natural Statements

<i>Key to tables</i> . . . . .	2-1
<i>Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows</i>	
<i>statements</i> . . . . .	2-4
<i>Constants</i> . . . . .	2-4
<i>System variables</i> . . . . .	2-5
<i>Date and time system variables</i> . . . . .	2-8
<i>Session parameters</i> . . . . .	2-9
<i>System functions</i> . . . . .	2-10
<i>Conditional expressions</i> . . . . .	2-11
<i>Statements</i> . . . . .	2-12



**3 Supported PL/I Statements**

<i>Key to tables</i> . . . . .	3-1
<i>Supported IBM Visual Age PL/I 2.0 statements</i> . . . . .	3-3
<i>Problem data types</i> . . . . .	3-3
<i>Program-control data types</i> . . . . .	3-3
<i>Aggregate types and attributes</i> . . . . .	3-4
<i>Expressions and references</i> . . . . .	3-5
<i>Statements</i> . . . . .	3-7
<i>Built-in functions and pseudovariables</i> . . . . .	3-12

**4 Supported SQL Statements**

<i>Supported DDL statements</i> . . . . .	4-1
<i>Supporting DDL in ATW</i> . . . . .	4-1
<i>Supported DDL statements</i> . . . . .	4-2
<i>Constraints of DDL Support in ATW</i> . . . . .	4-4
<i>Supported SQL Statements</i> . . . . .	4-8
<i>Supported ANSI SQL-92 Statements</i> . . . . .	4-9
<i>Supported RDMS 2200 SQL Statements</i> . . . . .	4-28

**5 Supported JCL Statements**

<i>Supported JCL statements</i> . . . . .	5-1
---	-----

**6 Supported CICS Statements**

<i>Supported CICS command statements</i> . . . . .	6-1
<i>Supported CICS/BMS statements</i> . . . . .	6-4

**7 Supported CA-IDMS DML Statements**

<i>Supported CA-IDMS DML statements</i> . . . . .	7-1
---	-----

**8 Supported IMS Statements**

<i>Supported IMS/Exec DLI statements</i> . . . . .	8-1
<i>Supported IMS/MFS statements</i> . . . . .	8-2

## **9 Supported UNISYS Statements**

<i>Supported ECL statements</i> . . . . .	9-1
<i>Supported DPS (FLDP) statements</i> . . . . .	9-3
<i>Supported DMS (DDL) statements</i> . . . . .	9-4

## **10 Supported Complexity Metrics**

<i>Supported Cobol Complexity Metrics</i> . . . . .	10-2
<i>Supported PLI Complexity Metrics</i> . . . . .	10-5
<i>Supported Natural Complexity Metrics</i> . . . . .	10-6

## **Bibliography**

## **Notices**

## ***Preface***

**T**he IBM Asset Transformation Workbench (ATW) is a suite of PC-based software products for analyzing, re-architecting, and transforming legacy applications. The products are deployed in an integrated environment with access to a common repository of program objects. Repository models serve as the basis for a rich set of diagrams, reports, and other documentation.

The ATW suite consists of customizable modules that together address the needs of organizations at every stage of legacy application evolution — maintenance/enhancement, renovation, and modernization.

### ***Audience***

This manual assumes that you are a corporate Information Technology (IT) professional with a working knowledge of the legacy platforms you are using the product to analyze. If you are transforming a legacy application, you should also have a working knowledge of the target platform.

## Organization

This manual contains the following chapters:

- Chapter 1, “Supported COBOL Statements,” describes the COBOL statements supported by the ATW Application Analyzer product.
- Chapter 2, “Supported Natural Statements,” describes the Natural statements supported by the ATW Application Analyzer product.
- Chapter 3, “Supported PL/I Statements,” describes the PL/I statements supported by the ATW Application Analyzer product.
- Chapter 4, “Supported SQL Statements,” describes SQL and DDL statements supported by the ATW Application Analyzer product.
- Chapter 5, “Supported JCL Statements,” describes JCL statements supported by the ATW Application Analyzer product.
- Chapter 6, “Supported CICS Statements,” describes CICS (BMS and Exec CICS) statements supported by the ATW Application Analyzer product.
- Chapter 7, “Supported CA-IDMS DML Statements,” describes IDMS Data Manipulation Language statements supported by the ATW Application Analyzer product.
- Chapter 8, “Supported IMS Statements,” describes IMS (MFS and Exec DLI) statements supported by the ATW Application Analyzer product.
- Chapter 9, “Supported UNISYS Statements,” describes UNISYS (DMS DDL, DPS and ECL) statements supported by the ATW Application Analyzer product.
- Chapter 10, “Supported Complexity Metrics,” contains language-specific details for the complexity metrics supported by the ATW legacy estimation tools.

## Conventions

This guide uses the following typographic conventions:

- **Bold type** — Indicates a specific area within the graphical user interface, such as a button on a screen, a window name, or a command or function.
- *Italic type* — Indicates a new term. Also indicates a document title. Occasionally, italic type is used for emphasis.
- `Monospace type` — Indicates computer programming code.
- **Bold monospace type** — Indicates input you type on the computer keyboard.
- **1A/1B, 2A/2B** — In task descriptions, indicates mutually exclusive steps: perform step A or step B, but not both.

## Related Manuals

This document is part of a complete set of ATW manuals. Together they provide all the information you need to get the most out of the system.

- *Getting Started* introduces ATW. This guide provides an overview of the workbench tools and discusses basic concepts. It describes how to install the product and how to manage licenses. It also describes how to use common product features.
- *Preparing Projects* describes how to set up ATW projects. This guide describes how to load applications in the repository and how to use reports and other tools to ensure that the entire application is available for analysis.
- *Analyzing Projects* describes how to analyze applications at the project level. This guide describes how to create diagrams of applications and how to perform impact analysis across applications. It also describes how to estimate project complexity and effort, and how to create a project dictionary.
- *Analyzing Programs* describes how to analyze applications at the program level. This guide describes how to use HyperView tools to

view programs interactively and perform program analysis in stages. It also describes how to analyze procedure and data flows, search the repository, and extract business rules with HyperView.

- *Profiling Projects* describes how to create and browse web-generated views of the repositories in your organization.
- *Creating Components* describes how to extract program components from a legacy application.
- *Architecture Reference Manual* describes legacy constructions supported by Application Architect in reference format.

## Online Help

In addition to the manuals provided with the system, you can learn about the product using the integrated online help. All GUI-based tools include a standard Windows **Help** menu.

You can display:

- The entire help system, with table of contents, index, and search tool, by selecting **Help: Help Topics**.
- Help about a particular ATW window by clicking the window and pressing the **F1** key.

Many ATW tools have *guides* that you can use to get started quickly in the tool. The guides are help-like systems with hyperlinks that you can use to access functions otherwise available only in menus and other program controls.

To open the guide for a tool, choose **Guide** from the **View** menu. Use the table of contents in the **Page** drop-down to navigate quickly to a topic.

# Supported COBOL Statements



This chapter contains detailed information on ATW parser, HyperView, Change Analyzer, and Global Data Element Flow support for Common Business-Oriented Language (COBOL) statements. For a list of supported legacy versions, refer to the *Release Notes*.

## Key to tables

The tables below contain detailed information on ATW parser, HyperView, Change Analyzer, Global Data Element Flow, and repository support. The first column of each table shows COBOL keywords. Next, if needed, in the 'Format' column, the possible forms of usage for each keyword are explained. These forms are represented in the standard grammar notation, where

- the optional arguments are written in square brackets '[arguments]';
- the arguments which cannot be omitted are listed in angle brackets <arguments>, and
- the vertical line '|' means 'or' logical connective.

In this way, the notation <arg1 | arg2 | arg3> means that one of the arg1, arg2 or arg3 is required. Other columns provide information for each particular tool.

- **Pars.** — ATW Parser. The possible values in this column are:
  - (V)erified — parsed and prepared for further processing. Possibly is supported by other ATW tools. Does not initiate any errors or warnings during the verification phase.
  - (V)erification (O)nly — parsed and skipped immediately. Further processing is not possible, i.e. not supported by ATW tools. Verification warning is initiated.
  - (N)ot Verified — if the ATW parser meets such an entity it stops and verification is considered as erroneous.
- **IA** — Interactive Analysis. Interactive Analysis (S)upport means that the entity will be displayed in the HyperView window and all the HyperView analysis tools will be available. In fact, every entity which is verified by ATW parser is supported in HyperView. If the entity is (N)ot supported then it is either (N)ot Verified or is skipped by parser without creating an appropriate node in the parser tree. In this way, 'N' in the HyperView column corresponds to (N) or (VO) in the parser column.
- **CA/GDF** — Change Analyzer and Global Data Element Flow. The 'Sup.' subcolumn in this column describes general support in CA and GDF. Similarly to HyperView, it contains 'S' for (S)upported statements and 'N' for the ones, which are (N)ot supported. Next subcolumn labeled 'Pairs' describes the possible pairs of the arguments, which can be joined by some program level relationship. The last column describes the relationship itself. The possible relationships and the explanation what they are given below:
  - 'const.move' — for literal assignments  

```
MOVE literal TO var
```
  - 'const.init' — for initial values at declarations  

```
PIC 99 VALUE 10.
```
  - 'const.comp' — for other manipulations with literals  

```
STRING 'str1' 'str2' INTO var
```
  - 'move' — for memory-to-memory assignments



```
MOVE var1 TO var2
```

- ‘cast’ — for assignments with datatype conversion

```
MOVE num TO str
```

- ‘comp’ — for operations, where data flow exists but operands are not synonyms:

```
SET ptr TO address of var
```

```
STRING s1, s2 INTO ss
```

```
exponentiation, bit (&, |) operations
```

- ‘comp+’ — for ‘+’ and ‘-’ operations

```
COMPUTE c = a + b
```

- ‘comp\*’ — for ‘\*’ and ‘/’ operations

```
COMPUTE c = a / b
```

- ‘cond’ — for conditions, which are synonyms but no data flow exist

```
if a = b
```

- ‘like’ — for declarations using LIKE statements

**Note:** For more detailed information about relationships in CA/GDF, refer to *Analyzing Projects* in the ATW document set.

- **Repository** — describes the entities that can occur in repository and possible relationships between them.

There is also the column Extraction for JCL and ECL statements. The occurrence of meaning ‘S’ in this column denotes that the supplementary information for this statement is extracted. Such information is required for Update Data Flow method being used in Diagrammer tool.

## Supported VS COBOL II statements

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### Special registers

Registers	Parser	IA	CA/GDF
ADDRESS OF	V	S	S
DEBUG ITEM	V	S	S
LENGTH OF	V	S	S
LINAGE-COUNTER	V	S	S
RETURN-CODE	V	S	S
SHIFT OUT / SHIFT IN	V	S	S

Registers	Parser	IA	CA/GDF
SORT-CONTROL	V	S	S
SORT-CORE-SIZE	V	S	S
SORT-FILE-SIZE	V	S	S
SORT-MESSAGE	V	S	S
SORT-MODE-SIZE	V	S	S
SORT-RETURN	V	S	S
TALLY	V	S	S
WHEN COMPILED	V	S	S

### Literals

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literals	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3: with double-byte characters</b> "EBCDIC-data&lt;D1D2&gt;EBCDIC-data"</li> </ul>	N	N	N
	<ul style="list-style-type: none"> <li><b>Format 4: hexadecimal notation</b> X"hexadecimal-digits"</li> </ul>	V	S	S

**1-6** Supported COBOL Statements  
Supported VS COBOL II statements

Literal type	Format	Parser	IA	CA/GDF
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; mantissa E &lt;+   -&gt; exponent</li> </ul>	V	S	S
DBCS literals	<ul style="list-style-type: none"> <li><b>Format</b> G"&lt;D1D2D3&gt;"</li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

**Referencing names**

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	S

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: subscripting</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li><b>Format 1: Data Division</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: Special-Names paragraph</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>mnemonic-name-1</i>]</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>integer-1</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>&gt;</li> </ul>	V	S	S treated as reference to the whole array
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>data-name-1</i>   FUNCTION <i>function-name-1</i> (<i>arguments</i>)&gt; (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S treated as reference to the whole <i>data-name-1</i>
	FUNCTION <i>function-name-1</i> ( <i>arguments</i> )>	VO	N	
	( <i>leftmost-character-position: [length]</i> )	V	S	

1-8 Supported COBOL Statements  
Supported VS COBOL II statements

**COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i> .	V	S
Nested program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> <i>nested source program</i> END-PROGRAM <i>program-name-1</i> .	N	N

**Identification Division**

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name</i> [[IS] <RECURSIVE   COMMON [INITIAL]   INITIAL [COMMON]> [PROGRAM]]. [AUTHOR. <i>comment-entry</i> ] [INSTALLATION. <i>comment-entry</i> ] [DATE-WRITTEN. <i>comment-entry</i> ] [DATE-COMPILED. <i>comment-entry</i> ] [SECURITY. <i>comment-entry</i> ]	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point

## Environment Division

### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: QSAM/SAM/VSAM sequential file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL] [PADDING [CHARACTER] [IS] < <i>data-name-5</i>   <i>literal-2</i> >] [RECORD DELIMITER [IS] <STANDARD-1   <i>assignment-name-2</i> >] [ACCESS [MODE] [IS] SEQUENTIAL] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: VSAM indexed file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   <i>literal-1</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] INDEXED [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD [KEY] [IS] <i>data-name-2</i> [PASSWORD [IS] <i>data-name-6</i> ALTERNATE RECORD [KEY] [IS] <i>data-name-3</i> [PASSWORD [IS] <i>data-name-7</i> ] [[WITH] DUPLICATES] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: VSAM relative file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   <i>literal-1</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC> RELATIVE [KEY] [IS] <i>data-name-4</i> >] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: QSAM/SAM I-O</b></li> </ul> I-O-CONTROL. <RERUN ON < <i>assignment-name-1</i>   <i>file-name-1</i> > [EVERY] < <i>integer-1</i> RECORDS   END [OF] <REEL   UNIT>> [OF] <i>file-name-1</i>   SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i>   MULTIPLE FILE [TAPE] [CONTAINS] <i>file-name-5</i> [POSITION] <i>integer-2</i>   APPLY WRITE-ONLY [ON] <i>file-name-2</i> >.	VO	N		

**1-10** Supported COBOL Statements  
Supported VS COBOL II statements

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 2: VSAM I-O</b> I-O-CONTROL. &lt;RERUN ON &lt;assignment-name-1   file-name-1&gt; [EVERY] integer-1 RECORDS [OF] file-name-1   SAME [RECORD] [AREA] [FOR] file-name-3 file-name-4&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: sort-merge I-O</b> I-O-CONTROL. [RERUN [ON] assignment-name-1] SAME &lt;RECORD   SORT   SORT-MERGE&gt; [AREA] [FOR] file-name-3 file-name-4.</li> </ul>	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA
OBJECT-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> OBJECT-COMPUTER. [computer-name [MEMORY [SIZE] integer &lt;WORDS   CHARACTERS   MODULES&gt;] [[PROGRAM] [COLLATING] SEQUENCE [IS] alphabet-name] [SEGMENT-LIMIT [IS] priority-number].]</li> </ul>	VO	N
SOURCE-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE-COMPUTER. [computer-name [[WITH] DEBUGGING MODE].]</li> </ul>	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [environment-name-1 [IS] mnemonic-name-1   environment-name-2 &lt;[IS] mnemonic-name-2 entry-1   entry-1&gt;] where entry-1 is: entry-1: &lt;ON [STATUS] [IS] condition-1 [OFF [STATUS] [IS] conditional-2]   OFF [STATUS] [IS] condition-2 [ON [STATUS] [IS] conditional-1]&gt;</li> </ul>	V	S
	[ALPHABET alphabet-name-1 [IS] <STANDARD-1   STANDARD-2   NATIVE   EBCDIC   literal-1 [<THROUGH   THRU> literal-2   ALSO literal-3]>]	VO	N
	[SYMBOLIC [CHARACTERS] symbolic-character [ARE   IS] integer-1 [IN] alphabet-name-2]	VO	N
	[CLASS class-name-1 [IS] literal-4 [<THROUGH   THRU> literal-5]]	VO	N
	[CURRENCY [SIGN] [IS] literal-6 [[WITH] PICTURE SYMBOL literal-7]]	VO	N
	[DECIMAL-POINT [IS] COMMA] [.]	V	S



**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Data Division	<ul style="list-style-type: none"> <li><b>Format: program and method DATA DIVISION</b></li> </ul> DATA DIVISION. [FILE SECTION. [ <i>file-description-entry</i> <i>record-description-entry</i> ]] [WORKING-STORAGE SECTION. [ <i>record-description-entry</i>   <i>data-item-description-entry</i> ]] [LINKAGE SECTION. [ <i>record-description-entry</i>   <i>data-item-description-entry</i> ]]	V	S	S			
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b></li> </ul> FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL] [BLOCK [CONTAINS] [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i> ]->] [LABEL <RECORD [IS]   RECORDS [ARE] > <STANDARD   OMITTED   <i>data-name-2</i> >] [VALUE OF <i>system-name-1</i> [IS] < <i>data-name-3</i>   <i>literal-1</i> >] [DATA <RECORD [IS]   RECORDS [ARE] > <i>data-name-4</i> ] [LINAGE [IS] < <i>data-name-5</i>   <i>integer-8</i> > [LINES] [[WITH] FOOTING [AT] < <i>data-name-6</i>   <i>integer-9</i> >] [[LINES] [AT] TOP < <i>data-name-7</i>   <i>integer-10</i> >] [[LINES] [AT] BOTTOM < <i>data-name-8</i>   <i>integer-11</i> >]] [RECORDING [MODE] [IS] <i>mode</i> ] [CODE-SET [IS] <i>alphabet-name</i> ].	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: relative and indexed files</b></li> </ul> FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL] [BLOCK [CONTAINS] [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i> ]->] [LABEL <RECORD [IS]   RECORDS [ARE] > <STANDARD   OMITTED   <i>data-name-2</i> >] [VALUE OF <i>system-name-1</i> [IS] < <i>data-name-3</i>   <i>literal-1</i> >] [DATA <RECORD [IS]   RECORDS [ARE] > <i>data-name-4</i> ].	V	S	S			

1-12 Supported COBOL Statements  
Supported VS COBOL II statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
	<ul style="list-style-type: none"> <li><b>Format 3: sort/merge files</b>  SD <i>file-name-1</i>  [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i>] [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;]  [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]  [BLOCK [CONTAINS] <i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]  [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]  [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]  [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]  [CODE-SET [IS] <i>alphabet-name</i>].</li> </ul>	V	S	S			
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b>  <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt;  [<i>redefines-clause</i>] [<i>blank-when-zero-clause</i>] [<i>external-clause</i>] [<i>global-clause</i>] [<i>justified-clause</i>] [<i>occurs-clause</i>] [<i>picture-clause</i>] [<i>sign-clause</i>] [<i>synchronized-clause</i>] [<i>usage-clause</i>] [<i>value-clause</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b>  66 <i>data-name-1</i> <i>renames-clause</i>.</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3</b>  88 <i>condition-name</i> <i>value-clause</i>.</li> </ul>	V	S	S			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b>  BLANK [WHEN] &lt;ZERO   ZEROS   ZEROES&gt;</li> </ul>	V	S	N			
DATE FORMAT clause	<ul style="list-style-type: none"> <li><b>Format</b>  DATE FORMAT [IS] <i>date-pattern</i></li> </ul>	VO	N	N			
EXTERNAL clause		V	S	N			
GLOBAL clause		V	S	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b>  &lt;JUSTIFIED   JUST&gt; [RIGHT]</li> </ul>	VO	N	N			

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> OCCURS <i>integer-2</i> [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> OCCURS <i>integer-1</i> TO <i>integer-2</i> [TIMES] DEPENDING [ON] <i>data-name-1</i> [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; [IS] <i>character-string</i></li> </ul>	V	S	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; REDEFINES <i>data-name-2</i></li> </ul>	V	S	S			
RENAME clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> USAGE [IS] &lt;BINARY   COMP   COMP-1   COMP-2   COMP-3   COMP-4   COMPUTATIONAL   COMPUTATIONAL-1   COMPUTATIONAL-2   COMPUTATIONAL-3   COMPUTATIONAL-4   DISPLAY   DISPLAY-1   INDEX   PACKED-DECIMAL   POINTER &gt;</li> </ul>	V	S	S			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> 88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: NULL value</b> VALUE [IS] &lt;NULL   NULLS&gt;</li> </ul>	V	S	S			

**1-14** Supported COBOL Statements  
Supported VS COBOL II statements

**Procedure Division**

**Arithmetic operators**

Operation	Meaning	Format	Parser	IA	CA/GDF			
					Sup.	Pair	Relation	
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

**Conditional Expressions**

Conditions	Format	Parser	IA	CA/GDF		
				Sup.	Pair	Relation
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1 [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><i>  class-name   DBCS   KANJI &gt;</i></li> </ul>	VO	N	N		

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pair		Relation
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> operand-1 [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; operand-2</li> </ul>	V	S	S	operand-1	operand-2	cond
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> operand-1 [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> NOT condition-1</li> </ul>	V	S	S			
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-1 &lt;AND   OR&gt; condition-2</li> </ul>	V	S	S			
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> relation-condition &lt;AND   OR&gt; [NOT] [relational-operator] object</li> </ul>	V	S	S			

1-16 Supported COBOL Statements  
Supported VS COBOL II statements

Statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pair	Relation	Entities	Relations
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern.using	
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. <i>sentence</i>] END DECLARATIVES.</li> </ul>	V	S	S				
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>   <i>environment-name</i>&gt;]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.screen	
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE   DAY   DAY-OF-WEEK   TIME&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.screen	
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal</i>&gt; TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-1</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+	
					<i>id-2</i>	<i>id-3</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER <i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i></li> </ul>	V	S	S				

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relations
CALL	<ul style="list-style-type: none"> <li><b>Format 1: with ON OVERFLOW</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   id-3   file-name-1&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2&gt;&gt;] [ON] OVERFLOW imperative-stmt-1] [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
				port	id-2, id-5	extern.call			
	<ul style="list-style-type: none"> <li><b>Format 2: with ON EXCEPTION</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   id-3   file-name-1&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2&gt;&gt;] [[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
				port	id-2, id-5	extern.call			
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S					
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE file-name-1 [&lt;REEL   UNIT&gt; [[FOR] REMOVAL   WITH NO REWIND]   [WITH] &lt;NO REWIND   LOCK&gt;]</li> </ul>	V	S	S					
		VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE file-name-1 [[WITH] LOCK]</li> </ul>	V	S	S					
		VO	N	N					
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE id-1 [ROUNDED] [EQUAL] arithmetic-expr [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-COMPUTE]</li> </ul>	V	S	S	identifiers of the arithmetic-expr	id-1	comp+ or comp* (according to the operation performed)		
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S					

1-18 Supported COBOL Statements  
Supported VS COBOL II statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair	Relation	Entities	Relations	
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph) Program Deletes From File	
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [UPON &lt;<i>mnemonic-name-1</i>   <i>environment-name-1</i>&gt;] [[WITH] NO ADVANCING]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.screen		
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-1</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-3</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-4</i>	comp*		
					<i>id-2</i>	<i>id-4</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-4</i>	comp*		
					<i>id-2</i>	<i>id-4</i>	comp*		
ENTRY	<ul style="list-style-type: none"> <li><b>Format</b> ENTRY <i>literal-1</i> [USING <i>id-1</i>]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.using	defines <i>literal-1</i> (Program Entry Point)	Program Has Program Entry Point
					<i>port</i>	<i>id-1</i>	extern.using		



Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relations
					id-1, expr-1	condition-1			
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;] WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1 [WHEN OTHER imperative-stmt-2] [END-EVALUATE] Phrases phrase-1 and phrase-2 should be represented in the following form: &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-3   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S	id-1, expr-1	condition-1	cond		
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> paragraph-name. EXIT.</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b> GOBACK</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] procedure-name-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] procedure-name-1 DEPENDING [ON] id-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: altered</b> paragraph-name. GO [TO].</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: MORE-LABELS</b> GO [TO] MORE-LABELS</li> </ul>	N	N	N					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF condition-1 THEN &lt;stmt-1   NEXT SENTENCE&gt; [ELSE &lt;stmt-2   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					

1-20 Supported COBOL Statements  
Supported VS COBOL II statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair	Relation	Entities	Relations	
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED   DBCS   EGCS&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S					
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair		Relation	Entities	Relations
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i></li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>USING <i>file-name-2 file-name-3</i> &lt;OUTPUT PROCEDURE [IS] <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>file-name-4</i>&gt;</li> </ul>	V	S	S					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair		Relation	Entities	Relations
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]   imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] &lt;id-1   integer-1&gt; TIMES imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with UNTIL phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: with VARYING phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 6: with VARYING phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 [AFTER &lt;id-5   index-name-3&gt; FROM &lt;id-6   index-name-4   literal-3&gt; BY &lt;id-7   literal-4&gt; UNTIL condition-2]   &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-2</li> </ul>	N	N	N					

1-24 Supported COBOL Statements  
Supported VS COBOL II statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relations	
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b> READ <i>file-name-1</i></li> </ul>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	[NEXT   PREVIOUS] [RECORD]	VO	N	N					
	[INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-READ]	V	S		<i>port</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	extern.file	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b> READ <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [KEY [IS] <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-3</i>] [ NOT INVALID [KEY] <i>imperative-stmt-4</i>] [END-READ]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S	S					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b> RETURN <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [[AT] END <i>imperative-stmt-1</i>] [NOT [AT] END <i>imperative-stmt-2</i>] [END-RETURN]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format</b> REWRITE <i>file-name-1</i> [FROM <i>id-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-REWRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Updates File
					<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	<i>port</i>	extern.file		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relations	
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   integer-1&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> integer-1	move		
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   integer-2&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET <i>condition-name-1</i> TO TRUE</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	move		
	<ul style="list-style-type: none"> <li><b>Format 5: USAGE IS POINTER data items</b> SET &lt;<i>id-4</i>   ADDRESS OF <i>id-5</i>&gt; TO &lt;<i>id-6</i>   ADDRESS OF <i>id-7</i>   NULL   NULLS&gt;</li> </ul>	V	S	S	<i>id-4</i> <i>id-4</i>	<i>id-6</i> <i>id-7</i>	move comp		

1-26 Supported COBOL Statements  
Supported VS COBOL II statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relations	
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i> [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>&lt;USING <i>file-name-2</i>   INPUT PROCEDURE [IS] <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]&gt;</li> <li>&lt;GIVING <i>file-name-3</i>   OUTPUT PROCEDURE [IS] <i>procedure-name-3</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-4</i>]&gt;</li> </ul>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		



Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relations
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED [BY] [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER [IN] id-5] [COUNT [IN] id-6] [[WITH] POINTER id-7] [TALLYING [IN] id-8] [[ON] OVERFLOW imperative-stmt-1] [NOT [ON] OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

1-28 Supported COBOL Statements  
Supported VS COBOL II statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relations
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: VSAM sequential files</b></li> </ul> WRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [END-WRITE]	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
					<i>record-name-1</i>	<i>port</i>	extern.file		
	<ul style="list-style-type: none"> <li><b>Format 2: QSAM/SAM sequential files</b></li> </ul> WRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [<<BEFORE   AFTER> [ADVANCING] << <i>id-1</i>   <i>integer-1</i> > [LINE   LINES]   <i>mnemonic-name-1</i>   PAGE>] [[AT] <END-OF-PAGE   EOP> <i>imperative-stmt-3</i> ] [NOT [AT] <END-OF-PAGE   EOP> <i>imperative-stmt-4</i> ]   [INVALID [KEY] <i>imperative-stmt-1</i> ] [NOT INVALID [KEY] <i>imperative-stmt-2</i> ] [END-WRITE]	VO	N	N				refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
	<ul style="list-style-type: none"> <li><b>Format 3: VSAM indexed and relative files</b></li> </ul> WRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [INVALID [KEY] <i>imperative-stmt-1</i> ] [NOT INVALID [KEY] <i>imperative-stmt-2</i> ] [END-WRITE]	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move		
					<i>record-name-1</i>	<i>port</i>	extern.file		

**Compile-directing statements and directives**

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
BASIS	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [ <i>sequence-number</i> ] BASIS < <i>basis-name</i>   <i>literal-1</i> >	VO	N		
CBL (PROCESS)	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <CBL   PROCESS> [ <i>options-list</i> ]	VO	N		
*CONTROL (*CBL)	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <*CONTROL   *CBL> <SOURCE   NOSOURCE   LIST   NOLIST   MAP   NOMAP>	VO	N		
COPY	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> COPY < <i>text-name-1</i>   <i>literal-1</i> > [<OF   IN> < <i>library-name</i>   <i>literal-2</i> >] [SUPPRESS] [REPLACING <i>operand-1</i> BY <i>operand-2</i> ]	V	S	refers to <i>text-name-1</i> (Copybook)	Cobol Includes Copybook

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] DELETE sequence-number-field</li> </ul>	VO	N		
EJECT	<ul style="list-style-type: none"> <li><b>Format</b> EJECT [.]</li> </ul>	VO	N		
ENTER	<ul style="list-style-type: none"> <li><b>Format</b> ENTER language-name-1 [routine-name-1].</li> </ul>	VO	N		
INSERT	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] INSERT sequence-number-field</li> </ul>	VO	N		
READY or RESET TRACE	<ul style="list-style-type: none"> <li><b>Format</b> &lt;READY   RESET&gt; TRACE.</li> </ul>	VO	N		
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b> REPLACE ==pseudo-text-1== BY ==pseudo-text-2==.</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPLACE OFF.</li> </ul>	V	S		
SERVICE LABEL	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE LABEL</li> </ul>	VO	N		
SERVICE RELOAD	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE RELOAD id-1</li> </ul>	VO	N		
SKIP1/2/3	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SKIP1   SKIP2   SKIP3&gt;.</li> </ul>	VO	N		
TITLE	<ul style="list-style-type: none"> <li><b>Format</b> TITLE <i>literal</i>.</li> </ul>	VO	N		

**1-30** Supported COBOL Statements  
*Supported COBOL for OS/390 statements*

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
USE	<ul style="list-style-type: none"> <li><b>Format 1: EXCEPTION ERROR declarative</b>            USE [GLOBAL] AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: LABEL declarative</b>            USE [GLOBAL] AFTER [STANDARD] [BEGINNING   ENDING] [FILE   REEL   UNIT] LABEL PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: DEBUGGING declarative</b>            USE [FOR] DEBUGGING [ON] &lt;procedure-name-1   ALL PROCEDURES&gt;</li> </ul>				

## Supported COBOL for OS/390 statements

**Note:** Object-oriented COBOL statements are not supported.

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
NULL / NULLS	V	S	S

***Special registers***

<b>Registers</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ADDRESS OF	V	S	S
DEBUG ITEM	V	S	S
LENGTH OF	V	S	S
LINAGE-COUNTER	V	S	S
RETURN-CODE	V	S	S
SHIFT OUT / SHIFT IN	V	S	S
SORT-CONTROL	V	S	S
SORT-CORE-SIZE	V	S	S
SORT-FILE-SIZE	V	S	S
SORT-MESSAGE	V	S	S
SORT-MODE-SIZE	V	S	S
SORT-RETURN	V	S	S
TALLY	V	S	S
WHEN COMPILED	V	S	S

1-32 Supported COBOL Statements  
Supported COBOL for OS/390 statements

**Literals**

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literals	<ul style="list-style-type: none"> <li>• <b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 3: with double-byte characters</b> "EBCDIC-data&lt;D1D2&gt;EBCDIC-data"</li> </ul>	N	N	N
	<ul style="list-style-type: none"> <li>• <b>Format 4: hexadecimal notation</b> X"hexadecimal-digits"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 5: null-terminated</b> Z"dddd"</li> </ul>	VO	N	N
Numeric literals	<ul style="list-style-type: none"> <li>• <b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 2: floating-point</b> &lt;+   -&gt; mantissa E &lt;+   -&gt; exponent</li> </ul>	V	S	S
DBCS literals	<ul style="list-style-type: none"> <li>• <b>Format 1</b> G"&lt;D1D2D3&gt;"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 2</b> N"&lt;D1D2D3&gt;"</li> </ul>	N	N	N
PICTURE character string	<ul style="list-style-type: none"> <li>• <b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

## Referencing names

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	S
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: subscripting</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li><b>Format 1: Data Division</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: Special-Names paragraph</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>mnemonic-name-1</i>]</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>integer-1</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>&gt;</li> </ul>	V	S	S treated as reference to the whole array

**1-34** Supported COBOL Statements  
*Supported COBOL for OS/390 statements*

Division	Format	Parser	IA	CA/GDF
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;data-name-1              FUNCTION <i>function-name-1</i> (<i>arguments</i>)&gt;            (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S treated as reference to the whole <i>data-name-1</i>
		VO	N	
		V	S	

**COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i>            ENVIRONMENT DIVISION. <i>env-division-content</i>            DATA DIVISION. <i>data-division-content</i>            PROCEDURE DIVISION. <i>proc-division-content</i>            END-PROGRAM <i>program-name-1</i>.</li> </ul>	V	S
Nested program	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i>            ENVIRONMENT DIVISION. <i>env-division-content</i>            DATA DIVISION. <i>data-division-content</i>            PROCEDURE DIVISION. <i>proc-division-content</i>  <i>nested source program</i> END-PROGRAM <i>program-name-1</i>.</li> </ul>	N	N



**Identification Division**

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name</i> [[IS] <RECURSIVE   COMMON [INITIAL]   INITIAL [COMMON]> [PROGRAM]]. [AUTHOR. <i>[comment-entry]</i> ] [INSTALLATION. <i>[comment-entry]</i> ] [DATE-WRITTEN. <i>[comment-entry]</i> ] [DATE-COMPILED. <i>[comment-entry]</i> ] [SECURITY. <i>[comment-entry]</i> ]	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point
Class Identification Division	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. CLASS-ID. <i>class-name-1</i> INHERITS <i>class-name-2</i> [METAClass [IS] <i>class-name-3</i> ]. [AUTHOR. <i>[comment-entry]</i> ] [INSTALLATION. <i>[comment-entry]</i> ] [DATE-WRITTEN. <i>[comment-entry]</i> ] [DATE-COMPILED. <i>[comment-entry]</i> ] [SECURITY. <i>[comment-entry]</i> ]	N	N		
Method Identification Division	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. METHOD-ID. <i>method-name-1</i> [[IS] [METHOD] OVERRIDE]. [AUTHOR. <i>[comment-entry]</i> ] [INSTALLATION. <i>[comment-entry]</i> ] [DATE-WRITTEN. <i>[comment-entry]</i> ] [DATE-COMPILED. <i>[comment-entry]</i> ] [SECURITY. <i>[comment-entry]</i> ]	N	N		

## Environment Division

### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL] [PADDING [CHARACTER] [IS] < <i>data-name-5</i>   <i>literal-2</i> >] [RECORD DELIMITER [IS] <STANDARD-1   <i>assignment-name-2</i> >] [ACCESS [MODE] [IS] SEQUENTIAL] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] INDEXED [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD [KEY] [IS] <i>data-name-2</i> [PASSWORD [IS] <i>data-name-6</i> ALTERNATE RECORD [KEY] [IS] <i>data-name-3</i> [[WITH] DUPLICATES] [PASSWORD [IS] <i>data-name-7</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC> RELATIVE [KEY] [IS] <i>data-name-4</i> >] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: line-sequential file-control entries (all platforms except VM)</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[ORGANIZATION [IS]] LINE SEQUENTIAL [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li> <b>Format 1: sequential I-O</b>                      I-O-CONTROL.                      &lt;RERUN ON &lt;assignment-name-1   file-name-1&gt; [EVERY] &lt;integer-1 RECORDS   END [OF] &lt;REEL   UNIT&gt;&gt; [OF] file-name-1   SAME [RECORD] [AREA] [FOR] file-name-3 file-name-4   MULTIPLE FILE [TAPE] [CONTAINS] file-name-5 [POSITION] integer-2   APPLY WRITE-ONLY [ON] file-name-2&gt;.                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 2: relative and indexed I-O</b>                      I-O-CONTROL.                      &lt;RERUN ON &lt;assignment-name-1   file-name-1&gt; [EVERY] integer-1 RECORDS [OF] file-name-1   SAME [RECORD] [AREA] [FOR] file-name-3 file-name-4&gt;                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 3: line-sequential I-O</b>                      I-O-CONTROL. SAME [RECORD] [AREA] [FOR] file-name-3 file-name-4.                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 4: sort-merge I-O</b>                      I-O-CONTROL. [RERUN [ON] assignment-name-1]                      SAME &lt;RECORD   SORT   SORT-MERGE&gt; [AREA] [FOR] file-name-3 file-name-4.                 </li> </ul>	VO	N		

### Configuration section

Paragraphs and entries	Format	Parser	IA
OBJECT-COMPUTER paragraph	<ul style="list-style-type: none"> <li> <b>Format</b>                      OBJECT-COMPUTER.                      [computer-name [MEMORY [SIZE] integer &lt;WORDS   CHARACTERS   MODULES&gt;] [[PROGRAM] [COLLATING] SEQUENCE [IS] alphabet-name] [SEGMENT-LIMIT [IS] priority-number].]                 </li> </ul>	VO	N
REPOSITORY paragraph	<ul style="list-style-type: none"> <li> <b>Format</b>                      REPOSITORY. [CLASS class-name-1 [[IS] external-class-name-1]].                 </li> </ul>	N	N
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li> <b>Format</b>                      SOURCE-COMPUTER. [computer-name [[WITH] DEBUGGING MODE].]                 </li> </ul>	VO	N

**1-38** Supported COBOL Statements  
Supported COBOL for OS/390 statements

Paragraphs and entries	Format	Parser	
		Parser	IA
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [<i>environment-name-1</i> [IS] <i>mnemonic-name-1</i>   <i>environment-name-2</i> &lt;[IS] <i>mnemonic-name-2</i> <i>entry-1</i>   <i>entry-1</i>&gt;] where <i>entry-1</i> is: <i>entry-1</i>: &lt;ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i>]   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i>]&gt;</li> </ul>	V	S
	[ALPHABET <i>alphabet-name-1</i> [IS] <STANDARD-1   STANDARD-2   NATIVE   EBCDIC   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> ]>]	VO	N
	[SYMBOLIC [CHARACTERS] <i>symbolic-character</i> [ARE   IS] <i>integer-1</i> [IN] <i>alphabet-name-2</i> ]	VO	N
	[CLASS <i>class-name-1</i> [IS] <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]]	VO	N
	[CURRENCY [SIGN] [IS] <i>literal-6</i> [[WITH] PICTURE SYMBOL <i>literal-7</i> ]]	VO	N
	[DECIMAL-POINT [IS] COMMA] [.]	V	S

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format: program and method DATA DIVISION</b> DATA DIVISION. [FILE SECTION. [<i>file-description-entry</i> <i>record-description-entry</i>]] [WORKING-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [LOCAL-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [LINKAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format: class DATA DIVISION</b> [WORKING-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]</li> </ul>	N	N	N		

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] <i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]            [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]            [RECORDING [MODE] [IS] <i>mode</i>] [CODE-SET [IS] <i>alphabet-name</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: relative and indexed files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] <i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] <i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 4: sort/merge files</b></li> </ul> <p>SD <i>file-name-1</i>            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]            [BLOCK [CONTAINS] <i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]            [CODE-SET [IS] <i>alphabet-name</i>].</p>	V	S	S			

1-40 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number &lt;data-name-1   FILLER&gt;</i> <i>[redefines-clause] [blank-when-zero-clause] [external-clause] [global-clause] [justified-clause] [occurs-clause] [picture-clause] [sign-clause] [synchronized-clause] [usage-clause] [value-clause] [data-format-clause]</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>66 data-name-1 renames-clause.</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>88 condition-name value-clause.</i></li> </ul>	V	S	S		
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>BLANK [WHEN] &lt;ZERO   ZEROS   ZEROES&gt;</i></li> </ul>	V	S	N		
DATE FORMAT clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>DATE FORMAT [IS] date-pattern</i></li> </ul>	VO	N	N		
EXTERNAL clause		V	S	N		
GLOBAL clause		V	S	N		
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;JUSTIFIED   JUST&gt; [RIGHT]</i></li> </ul>	VO	N	N		
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> <i>OCCURS integer-2 [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> <i>OCCURS integer-1 TO integer-2 [TIMES] DEPENDING [ON] data-name-1 [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S	S		
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;PICTURE   PIC&gt; [IS] character-string</i></li> </ul>	V	S	S		
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number &lt;data-name-1   FILLER&gt; REDEFINES data-name-2</i></li> </ul>	V	S	S		

Entries and clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: standard</b> USAGE [IS] &lt;BINARY   COMP   COMP-1   COMP-2   COMP-3   COMP-4   COMP-5   COMPUTATIONAL   COMPUTATIONAL-1   COMPUTATIONAL-2   COMPUTATIONAL-3   COMPUTATIONAL-4   COMPUTATIONAL-5   DISPLAY   DISPLAY-1   INDEX    PACKED-DECIMAL   POINTER &gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: with 'NATIVE'</b> USAGE [IS] &lt;BINARY NATIVE   COMP   COMP-1 NATIVE   COMP-2 NATIVE   COMP-3   COMP-4 NATIVE   COMP-5   COMPUTATIONAL   COMPUTATIONAL-1 NATIVE   COMPUTATIONAL-2 NATIVE   COMPUTATIONAL-3   COMPUTATIONAL-4 NATIVE   COMPUTATIONAL-5   DISPLAY NATIVE   DISPLAY-1 NATIVE   INDEX   PACKED-DECIMAL   POINTER   PROCEDURE-POINTER&gt;</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3: PROCEDURE-POINTER</b> USAGE [IS] PROCEDURE-POINTER</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 4: OBJECT REFERENCE</b> USAGE [IS] OBJECT REFERENCE [[METAClass [OF]] <i>class-name-1</i>]</li> </ul>	N	N	N			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> 88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: NULL value</b> VALUE [IS] &lt;NULL   NULLS&gt;</li> </ul>	V	S	S			

## Procedure Division

### Arithmetic operators

Operation	Meaning	Format	Parser	IA	Sup.	CA/GDF		
						Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			



### Conditional Expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pair	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   <i>class-name</i>   DBCS   KANJI&gt;</li> </ul>	V	S	S			
		VO	N	N			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>operand-1</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	cond
	<ul style="list-style-type: none"> <li><b>Format 2: pointer data items</b> &lt;ADDRESS OF <i>id-1</i>   <i>id-2</i>   NULL   NULLS&gt; [IS] [NOT] EQUAL [TO] &lt;ADDRESS OF <i>id-3</i>   <i>id-4</i>   NULL   NULLS&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: procedure-pointer data items</b> <i>id-2</i>   NULL   NULLS] [IS] [NOT] EQUAL [TO] [ <i>id-4</i>   NULL   NULLS]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 4: object reference data items</b> &lt;<i>object-reference-id-1</i>   SELF   NULL   NULLS&gt; [IS] [NOT] EQUAL [TO] &lt;<i>object-reference-id-2</i>   SELF   NULL   NULLS&gt;</li> </ul>	N	N	N			
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>condition-1</i></li> </ul>	V	S	S			
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			

**1-44** Supported COBOL Statements  
Supported COBOL for OS/390 statements

Conditions	Format	Parser	IA	Sup.	CA/GDF		
					Pair		Relation
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S			

**Statements**

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format 1: programs and methods (with BY REFERENCE)</b> PROCEDURE DIVISION [USING [[BY] REFERENCE] <i>data-name-1</i>] [RETURNING <i>data-name-2</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern. using		
					<i>data-name-1</i>	<i>port</i>	extern. using		
						<i>data-name-1</i>	<i>port</i>	extern. using	
	<ul style="list-style-type: none"> <li><b>Format 2: programs and methods (with BY VALUE)</b> PROCEDURE DIVISION [USING [[BY] VALUE] <i>data-name-1</i>] [RETURNING <i>data-name-2</i>].</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: classes</b> PROCEDURE DIVISION.</li> </ul>	N	N	N					
PROCEDURE DIVISION structure	<ul style="list-style-type: none"> <li><b>Format 1: program and method</b> procedure division header [DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.] <i>section-name</i> SECTION [<i>priority-number</i>]. [<i>paragraph-name</i>. [<i>sentence</i>]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: classes</b> PROCEDURE DIVISION. [<i>method-definition</i>]</li> </ul>	N	N	N					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.</li> </ul>	V	S	S					
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>   <i>environment-name</i>&gt;]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen		
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE [YYYYMMDD]   DAY [YYYYDDD]   DAY-OF-WEEK   TIME&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen		
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal</i>&gt; TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
					<i>id-2</i>	<i>id-2</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-1</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+		
					<i>id-2</i>	<i>id-3</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
					<i>id-2</i>	<i>id-2</i>	comp+		
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER</li> </ul>	V	S	S					
	<i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i>	VO	N	N					
CALL	<ul style="list-style-type: none"> <li><b>Format 1</b> CALL &lt;<i>id-1</i>   <i>literal-1</i>&gt; [USING &lt;[[BY] REFERENCE] [ADDRESS OF] <i>id-2</i>   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] <i>id-3</i>   <i>literal-2</i>&gt;&gt;] [RETURNING <i>id-5</i>] [[[ON] EXCEPTION <i>imperative-stmt-1</i>] [NOT [ON] EXCEPTION <i>imperative-stmt-2</i>]   [ON] OVERFLOW <i>imperative-stmt-3</i>] [END-CALL]</li> </ul>	V	S	S	<i>id-2, id-3</i>	<i>port</i>	extern.call	refers to <i>literal-1</i> (program Entry Point), defines <i>id-1</i> (Decision)	Program Calls Program Entry Point, Program Calls Decision
					<i>port</i>	<i>id-2, id-5</i>	extern.call		

1-46 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 2: with OMITTED</b> CALL &lt;id-1 literal-1   procedure-ptr-1&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   file-name-1   OMITTED&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2   OMITTED&gt;   [BY] VALUE &lt;[ADDRESS OF   LENGTH OF] id-4   literal-3&gt;&gt;] [RETURNING id-5] [[[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2]   [ON] OVERFLOW imperative-stmt-3] [END-CALL]</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: with file-name-1</b> CALL &lt;id-1 literal-1   procedure-ptr-1&gt; [USING [[BY] REFERENCE] file-name-1] [RETURNING id-5] [[[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2]   [ON] OVERFLOW imperative-stmt-3] [END-CALL]</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 4: with USING BY VALUE</b> CALL &lt;id-1 literal-1   procedure-ptr-1&gt; [USING [BY] VALUE &lt;[ADDRESS OF   LENGTH OF] id-4   literal-3&gt;] [RETURNING id-5] [[[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2]   [ON] OVERFLOW imperative-stmt-3] [END-CALL]</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 5: with procedure-ptr-1</b> CALL procedure-ptr-1 [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   file-name-1   OMITTED&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2   OMITTED&gt;   [BY] VALUE &lt;[ADDRESS OF   LENGTH OF] id-4   literal-3&gt;&gt;] [RETURNING id-5] [[[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2]   [ON] OVERFLOW imperative-stmt-3] [END-CALL]</li> </ul>	VO	N	N					
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					
	[<REEL   UNIT> [[FOR] REMOVAL   WITH NO REWIND]   [WITH] <NO REWIND   LOCK>]	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					
	[[WITH] LOCK]	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b> CLOSE <i>filename-1</i></li> </ul>	V	S	S					
	[<REEL   UNIT> [[FOR] REMOVAL   WITH NO REWIND]   [WITH] <NO REWIND   LOCK>]	VO	N	N					
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE <i>id-1</i> [ROUNDED] [EQUAL] <i>arithmetic-expr</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-COMPUTE]</li> </ul>	V	S	S	<i>identifiers of the arithmetic-expr</i>	<i>id-1</i>	comp+ or comp* (according to the operation performed)		
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S					
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Deletes from File
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [UPON &lt;<i>mnemonic-name-1</i>   <i>environment-name-1</i>&gt;] [[WITH] NO ADVANCING]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.screen		

1-48 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;id-1   literal-1&gt; INTO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-1	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-3	comp*		
					id-1	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
					id-1	id-4	comp*		
					id-2	id-4	comp*		
<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*			
				id-2	id-3	comp*			
				id-1	id-4	comp*			
				id-2	id-4	comp*			
ENTRY	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTRY literal-1 [USING [[BY] REFERENCE] id-1] [BY] VALUE] id-1]</li> </ul>	V	S	S	id-1	port	extern.using	defines literal-1 (Program Entry Point)	Program Has Program Entry Point
					port	id-1	extern.using		
	<ul style="list-style-type: none"> <li><b>Format 2</b> ENTRY literal-1 [USING [[BY] VALUE] id-1]</li> </ul>	VO	N	N					

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;] WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1 [WHEN OTHER imperative-stmt-2] [END-EVALUATE] Phrases phrase-1 and phrase-2 should be represented in the following form: &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-3   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S	id-1, expr-1	condition-1	cond		
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> paragraph-name. EXIT.</li> </ul>	V	S	S					
EXIT METHOD	<ul style="list-style-type: none"> <li><b>Format</b> EXIT METHOD.</li> </ul>	N	N	N					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b> GOBACK</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] procedure-name-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] procedure-name-1 DEPENDING [ON] id-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: altered</b> paragraph-name. GO [TO].</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: MORE-LABELS</b> GO [TO] MORE-LABELS</li> </ul>	N	N	N					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF condition-1 THEN &lt;stmt-1   NEXT SENTENCE&gt; [ELSE &lt;stmt-2   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					

1-50 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED   DBCS   EGCS&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S				
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp	
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp	
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp	



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
INVOKE	<ul style="list-style-type: none"> <li><b>Format</b>            INVOKE &lt;id-1   class-name-1   SELF   [class-name-2 OF] SUPER&gt;            &lt;literal-1   id-2&gt;            [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-3   OMITTED&gt;              [[BY] CONTENT] &lt;[ADDRESS OF   LENGTH OF] id-4   literal-2              OMITTED&gt;   [[BY] VALUE] &lt;[ADDRESS OF   LENGTH OF] id-5   lit-            eral-3&gt;&gt;]            [RETURNING id-6]            [[ON] EXCEPTION imperative-stmt-1]            [NOT [ON] EXCEPTION imperative-stmt-2] [END-INVOKE]         </li> </ul>	N	N	N					
MERGE	<ul style="list-style-type: none"> <li><b>Format</b>            MERGE file-name-1 [ON] &lt;ASCENDING   DESCENDING&gt; [KEY]            data-name-1            [[COLLATING] SEQUENCE [IS] alphabet-name-1]         </li> </ul>	V	S	N					
	<ul style="list-style-type: none"> <li>USING file-name-2 file-name-3            &lt;OUTPUT PROCEDURE [IS] procedure-name-1 [&lt;THROUGH              THRU&gt; procedure-name-2]   GIVING file-name-4&gt;         </li> </ul>	VO	N	N					
		V	S	N					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b>            MOVE &lt;id-1   literal-1&gt; TO id-2         </li> </ul>	V	S	S	id-1	id-2	move or cast (depending on data types)		
	<ul style="list-style-type: none"> <li><b>Format 2</b>            MOVE &lt;CORRESPONDING   CORR&gt; id-1 TO id-2         </li> </ul>	V	S	S	id-1	id-2	move or cast (depending on data types)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b>            MULTIPLY &lt;id-1   literal-1&gt; BY id-2 [ROUNDED]            [[ON] SIZE ERROR imperative-stmt-1]            [NOT [ON] SIZE ERROR imperative-stmt-2] [END-MULTIPLY]         </li> </ul>	V	S	S	id-1	id-2	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b>            MULTIPLY &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt;            GIVING id-3 [ROUNDED]            [[ON] SIZE ERROR imperative-stmt-1]            [NOT [ON] SIZE ERROR imperative-stmt-2] [END-MULTIPLY]         </li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		

**1-52** Supported COBOL Statements  
*Supported COBOL for OS/390 statements*

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b>  OPEN &lt;INPUT <i>file-name-1</i>  [REVERSED   WITH NO REWIND]    OUTPUT <i>file-name-2</i>  [WITH NO REWIND]    I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					
		VO	N	N					
		V	S	S					
		VO	N	N					
		V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b>  OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   I-O <i>file-name-3</i>    EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					
<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b>  OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S						

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]   imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] &lt;id-1   integer-1&gt; TIMES imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with UNTIL phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1</li> </ul>	S	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: with VARYING phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 6: with VARYING phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 [AFTER &lt;id-5   index-name-3&gt; FROM &lt;id-6   index-name-4   literal-3&gt; BY &lt;id-7   literal-4&gt; UNTIL condition-2]   &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-2</li> </ul>	N	N	N					

**1-54** Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b> READ <i>file-name-1</i></li> </ul>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	[NEXT   PREVIOUS] [RECORD]	VO	N	N					
	[INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-READ]	V	S	S	<i>port</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	extern.file	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b> READ <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [KEY [IS] <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-3</i>] [ NOT INVALID [KEY] <i>imperative-stmt-4</i>] [END-READ]</li> </ul>	V	S	S					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S	S					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b> RETURN <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [[AT] END <i>imperative-stmt-1</i>] [NOT [AT] END <i>imperative-stmt-2</i>] [END-RETURN]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format</b> REWRITE <i>file-name-1</i> [FROM <i>id-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-REWRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Updates File
					<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	<i>port</i>	extern.file		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> <i>integer-1</i>	move		
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET <i>condition-name-1</i> TO TRUE</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	move		
	<ul style="list-style-type: none"> <li><b>Format 5: USAGE IS POINTER data items</b> SET &lt;<i>id-4</i>   ADDRESS OF <i>id-5</i>&gt; TO &lt;<i>id-6</i>   ADDRESS OF <i>id-7</i>   NULL   NULLS&gt;</li> </ul>	V	S	S	<i>id-4</i> <i>id-4</i>	<i>id-6</i> <i>id-7</i>	move comp		
	<ul style="list-style-type: none"> <li><b>Format 6: USAGE IS PROCEDURE-POINTER data items</b> SET <i>procedure-pointer-data-item-1</i> TO &lt;<i>procedure-pointer-data-item-2</i>   ENTRY &lt;<i>id-8</i>   <i>literal-1</i>&gt;   NULL   NULLS   <i>pointer-data-item-3</i>&gt;</li> </ul>	VO	N	N					

1-56 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 7: USAGE OBJECT REFERENCE data items</b> SET <i>object-reference-id-1</i> TO &lt;<i>object-reference-id-2</i>   NULL   SELF&gt;</li> </ul>	N	N	N					
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i> [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<USING <i>file-name-2</i>   INPUT PROCEDURE [IS] <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> ]> <GIVING <i>file-name-3</i>   OUTPUT PROCEDURE [IS] <i>procedure-name-3</i> [<THROUGH   THRU> <i>procedure-name-4</i> ]>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED [BY] [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER [IN] id-5] [COUNT [IN] id-6] [[WITH] POINTER id-7] [TALLYING [IN] id-8] [[ON] OVERFLOW imperative-stmt-1] [NOT [ON] OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

1-58 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>]   [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
				<i>record-name-1</i>	<i>port</i>	extern.file			
	<ul style="list-style-type: none"> <li><b>Format 2: sequential files with 'BEFORE/AFTER'</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [&lt;&lt;BEFORE   AFTER&gt; [ADVANCING] &lt;&lt;<i>id-1</i>   <i>integer-1</i>&gt; [LINE   LINES]   <i>mnemonic-name-1</i>   PAGE&gt;] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>]   [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	VO	N	N				refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
	<ul style="list-style-type: none"> <li><b>Format 3: indexed and relative files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
				<i>record-name-1</i>	<i>port</i>	extern.file			
<ul style="list-style-type: none"> <li><b>Format 4: line-sequential files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File	
			<i>record-name-1</i>	<i>port</i>	extern.file				
<ul style="list-style-type: none"> <li><b>Format 5: line-sequential files with 'AFTER'</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [AFTER [ADVANCING] &lt;<i>id-2</i>   <i>integer-1</i>&gt; [LINE   LINES]   PAGE&gt;] [END-WRITE]</li> </ul>	VO	N	N				refer to File (see FILE-CONTROL paragraph)	Program Inserts into File	



***Intrinsic functions***

Functions	Parser	IA	CA/GDF			
			Sup.	Pairs		Relation
ACOS	V	S	S	operands	result	comp
ANNUITY	V	S	S	operands	result	comp
ASIN	V	S	S	operands	result	comp
ATAN	V	S	S	operands	result	comp
CHAR	V	S	S	operands	result	comp
COS	V	S	S	operands	result	comp
CURRENT-DATE	V	S	S	operands	result	comp
DATE-OF-INTEGER	V	S	S	operands	result	comp
DATE-TO-YYYYMMDD	V	S	S	operands	result	comp
DATEVAL	V	S	S	operands	result	comp
DAY-OF-INTEGER	V	S	S	operands	result	comp
DAY-TO-YYYYDDD	V	S	S	operands	result	comp
FACTORIAL	V	S	S	operands	result	comp
INTEGER	V	S	S	operands	result	comp
INTEGER-OF-DATE	V	S	S	operands	result	comp
INTEGER-OF-DAY	V	S	S	operands	result	comp
INTEGER-PART	V	S	S	operands	result	comp
LENGTH	V	S	S	operands	result	comp
LOG	V	S	S	operands	result	comp
LOG10	V	S	S	operands	result	comp
LOWER-CASE	V	S	S	operands	result	comp

**1-60** Supported COBOL Statements  
*Supported COBOL for OS/390 statements*

Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
MAX	V	S	S	<i>operands</i>	<i>result</i>	comp	
MEAN	V	S	S	<i>operands</i>	<i>result</i>	comp	
MEDIAN	V	S	S	<i>operands</i>	<i>result</i>	comp	
MIDRANGE	V	S	S	<i>operands</i>	<i>result</i>	comp	
MIN	V	S	S	<i>operands</i>	<i>result</i>	comp	
MOD	V	S	S	<i>operands</i>	<i>result</i>	comp	
NUMVAL	V	S	S	<i>operands</i>	<i>result</i>	comp	
NUMVAL-C	V	S	S	<i>operands</i>	<i>result</i>	comp	
ORD	V	S	S	<i>operands</i>	<i>result</i>	comp	
ORD-MAX	V	S	S	<i>operands</i>	<i>result</i>	comp	
ORD-MIN	V	S	S	<i>operands</i>	<i>result</i>	comp	
PRESENT-VALUE	V	S	S	<i>operands</i>	<i>result</i>	comp	
RANDOM	V	S	S	<i>operands</i>	<i>result</i>	comp	
RANGE	V	S	S	<i>operands</i>	<i>result</i>	comp	
REM	V	S	S	<i>operands</i>	<i>result</i>	comp	
REVERSE	V	S	S	<i>operands</i>	<i>result</i>	comp	
SIN	V	S	S	<i>operands</i>	<i>result</i>	comp	
SQRT	V	S	S	<i>operands</i>	<i>result</i>	comp	
STANDARD-DEVIATION	V	S	S	<i>operands</i>	<i>result</i>	comp	
SUM	V	S	S	<i>operands</i>	<i>result</i>	comp	
TAN	V	S	S	<i>operands</i>	<i>result</i>	comp	
UNDATE	V	S	S	<i>operands</i>	<i>result</i>	comp	
UPPER-CASE	V	S	S	<i>operands</i>	<i>result</i>	comp	

Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
				operands	result		
VARIANCE	V	S	S	operands	result	comp	
WHEN-COMPILED	V	S	S	operands	result	comp	
YEAR-TO-YYYY	V	S	S	operands	result	comp	
YEARWINDOW	V	S	S	operands	result	comp	

### Compile-directing statements and directives

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
BASIS	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] BASIS &lt;basis-name   literal-1&gt;</li> </ul>	V	N		
CBL (PROCESS)	<ul style="list-style-type: none"> <li><b>Format</b> &lt;CBL   PROCESS&gt; [options-list]</li> </ul>	V	N		
CALLINTER-FACE directive	<ul style="list-style-type: none"> <li><b>Format</b> &lt;&gt;&gt;CALLINTERFACE   &gt;&gt;CALLINT &gt; [SYSTEM   OPTLINK   FAR16   PASCAL16   CDECL] [DESC   DESCRIPTOR   NODESC   NODESCRIPTOR]</li> </ul>	V	N		
*CONTROL (*CBL)	<ul style="list-style-type: none"> <li><b>Format</b> &lt;*CONTROL   *CBL&gt; &lt;SOURCE   NOSOURCE   LIST   NOLIST   MAP   NOMAP&gt;</li> </ul>	V	N		
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; &lt;library-name   literal-2&gt;] [SUPPRESS] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S	refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] DELETE sequence-number-field</li> </ul>	V	N		
EJECT	<ul style="list-style-type: none"> <li><b>Format</b> EJECT [.]</li> </ul>	V	N		

1-62 Supported COBOL Statements  
Supported COBOL for OS/390 statements

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
ENTER	<ul style="list-style-type: none"> <li><b>Format</b> ENTER <i>language-name-1</i> [<i>routine-name-1</i>].</li> </ul>	VO	N		
INSERT	<ul style="list-style-type: none"> <li><b>Format</b> [<i>sequence-number</i>] INSERT <i>sequence-number-field</i></li> </ul>	VO	N		
READY or RESET TRACE	<ul style="list-style-type: none"> <li><b>Format</b> &lt;READY   RESET&gt; TRACE.</li> </ul>	VO	N		
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b> REPLACE ==<i>pseudo-text-1</i>== BY ==<i>pseudo-text-2</i>==.</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPLACE OFF.</li> </ul>	V	S		
SERVICE LABEL	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE LABEL</li> </ul>	VO	N		
SERVICE RELOAD	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE RELOAD <i>id-1</i></li> </ul>	VO	N		
SKIP1/2/3	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SKIP1   SKIP2   SKIP3&gt;.</li> </ul>	VO	N		
TITLE	<ul style="list-style-type: none"> <li><b>Format</b> TITLE <i>literal</i>.</li> </ul>	VO	N		
USE	<ul style="list-style-type: none"> <li><b>Format 1: EXCEPTION ERROR declarative</b> USE [GLOBAL] AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;<i>file-name-1</i>   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: LABEL declarative</b> USE [GLOBAL] AFTER [STANDARD] [BEGINNING   ENDING] [FILE   REEL   UNIT] LABEL PROCEDURE [ON] &lt;<i>file-name-1</i>   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		

## Supported COBOL/400 statements

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### Special registers

Registers	Parser	IA	CA/GDF
ADDRESS OF	V	S	S
DB-FORMAT-NAME	N	N	N
DEBUG -ITEM	N	N	N
LENGTH OF	V	S	S
LINAGE-COUNTER	V	S	N
WHEN COMPILED	V	S	N

**1-64** Supported COBOL Statements  
*Supported COBOL/400 statements*

**Literals**

Literal type	Format	Parser	IA	CA/GDF
Boolean literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: B"0"</li> </ul>	V	S	S
Nonnumeric literal	<ul style="list-style-type: none"> <li><b>Format 1: with quotes (can contain any allowable character from the EBCDIC character set)</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: hexadecimal notation</b> <i>X"hexadecimal-digits"</i></li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

**Referencing names**

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1 [&lt;IN   OF&gt; library-name-1]</i></li> </ul>	V	S	
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1 [&lt;IN   OF&gt; section-name-1]</i></li> </ul>	V	S	
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	
	<ul style="list-style-type: none"> <li><b>Format 2</b> &lt;condition-name-1   data-name-1&gt; [&lt;IN   OF&gt; data-name-2]</li> </ul>	V	S	
	<ul style="list-style-type: none"> <li><b>Format 3</b> LINAGE-COUNTER [&lt;IN   OF&gt; file-name-2]</li> </ul>	V	S	
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;integer-1   ALL   data-name-3 [&lt;+   -&gt; integer-2]   index-name-1 [&lt;+   -&gt; integer-3]&gt;</li> </ul>	V	S	
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b> &lt;data-name-1&gt; [&lt;IN   OF&gt; data-name-2] (leftmost-character-position: [length])</li> </ul>	V	S	

### COBOL program structure

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b> &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i>.</li> </ul>	V	S

1-66 Supported COBOL Statements  
Supported COBOL/400 statements

### Identification Division

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b> &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name</i>. [AUTHOR. [<i>comment-entry</i>]] [INSTALLATION. [<i>comment-entry</i>]] [DATE-WRITTEN. [<i>comment-entry</i>]] [DATE-COMPILED. [<i>comment-entry</i>]] [SECURITY. [<i>comment-entry</i>]]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point

### Environment Division

#### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN &lt;[TO] <i>assignment-name-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S	defines program- name.assignment- name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL] [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN &lt;[TO] <i>assignment-name-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S		
	[ORGANIZATION [IS]] INDEXED	V	S		
	[RESERVE <i>integer</i> [AREA   AREAS]] [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC->] RECORD [KEY] [IS] < <i>data-name-2</i>   EXTERNALLY-SCRIBED-KEY-> [[WITH] DUPLICATES] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		



Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   <i>literal-1</i> >	V	S	defines program-name.assignment-name-1 (File)	Program Reads Data Port, Writes Data Port depending on OPEN statement
	[[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC> RELATIVE [KEY] [IS] <i>data-name-4</i> >] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: sort or merge</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   <i>literal-1</i> >	V	S		
	<ul style="list-style-type: none"> <li><b>Format 5: transaction</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   <i>literal-1</i> >	V	S		
	[[ORGANIZATION [IS]] TRANSACTION] [ACCESS [MODE] [IS] <SEQUENTIAL   DYNAMIC RELATIVE [KEY] [IS] <i>data-name-1</i> >] [[FILE] STATUS [IS] <i>data-name-2</i> [ <i>data-name-3</i> ]] [CONTROL-AREA [IS] <i>data-name-4</i> ].	VO	N		
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential files I-O</b></li> </ul> I-O-CONTROL. [RERUN ON < <i>assignment-name-1</i>   <i>file-name-1</i> > [EVERY] <i>integer-1</i> RECORDS [OF] <i>file-name-2</i> ] [SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i> ] [MULTIPLE FILE [TAPE] [CONTAINS] <i>file-name-5</i> [POSITION <i>integer-2</i> ]] [COMMITMENT CONTROL [FOR] <i>file-name-6</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files I-O</b></li> </ul> I-O-CONTROL. [RERUN ON < <i>assignment-name-1</i>   <i>file-name-1</i> > [EVERY] <i>integer-1</i> RECORDS [OF] <i>file-name-2</i> ] [SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i> ] [COMMITMENT CONTROL [FOR] <i>file-name-6</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: sort-merge I-O</b></li> </ul> I-O-CONTROL. SAME <RECORD   SORT   SORT-MERGE> [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i> .	VO	N		

**1-68** Supported COBOL Statements  
Supported COBOL/400 statements

**Configuration section**

Paragraphs and entries	Format	Parser	IA
OBJECT-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> OBJECT-COMPUTER. [<i>computer-name</i> [MEMORY [SIZE] <i>integer</i> &lt;WORDS   CHARACTERS   MODULES&gt;] [[PROGRAM] [COLLATING] SEQUENCE [IS] <i>alphabet-name</i>] [SEGMENT-LIMIT [IS] <i>priority-number</i>].]</li> </ul>	VO	N
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE-COMPUTER. [<i>computer-name</i> [[WITH] DEBUGGING MODE].]</li> </ul>	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [<i>environment-name-1</i> [IS] <i>mnemonic-name-1</i>   <i>environment-name-2</i> &lt;[[IS] <i>mnemonic-name-2</i> <i>entry-1</i>   <i>entry-1</i>&gt;] where <i>entry-1</i> is: <i>entry-1</i>: &lt;ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i>]   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i>&gt;</li> </ul>	V	S
	[ALPHABET <i>alphabet-name-1</i> [IS] <STANDARD-1   STANDARD-2   NATIVE   EBCDIC   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> >]	VO	N
	[CLASS <i>class-name-1</i> [IS] <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]]	VO	N
	[CURRENCY [SIGN] [IS] <i>literal-6</i> [[WITH] PICTURE SYMBOL <i>literal-7</i> ]]	VO	N
	[DECIMAL-POINT [IS] COMMA]	V	S
	<[CONSOLE [IS] CRT   CURSOR [IS] <i>data-name-1</i>   CRT STATUS [IS] <i>data-name-2</i> ] [.]	V	S

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format: program and method DATA DIVISION</b> DATA DIVISION. [FILE SECTION. [<i>file-description-entry</i> <i>record-description-entry</i>]] [WORKING-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [LINKAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]</li> </ul>	V	S			

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: Formatfile, Database, Disk</b></li> </ul> <p>FD <i>file-name-1</i>            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>].</p>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 2: Diskette, Tapefile</b></li> </ul> <p>FD <i>file-name-1</i>            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]            [CODE-SET [IS] <i>alphabet-name</i>].</p>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 3: Printer</b></li> </ul> <p>FD <i>file-name-1</i>            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]            [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]            [CODE-SET [IS] <i>alphabet-name</i>].</p>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 4: sort/merge</b></li> </ul> <p>SD <i>file-name-1</i>            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>].</p>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 5: transaction</b></li> </ul> <p>FD <i>file-name-1</i>            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>].</p>	V	S			

1-70 Supported COBOL Statements  
Supported COBOL/400 statements

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number &lt;data-name-1   FILLER&gt;</i> <i>[redefines-clause   like-clause] [blank-when-zero-clause] [justified-clause] [occurs-clause] [picture-clause] [sign-clause] [synchronized-clause] [usage-clause] [value-clause]</i></li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>66 data-name-1 renames-clause.</i></li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>88 condition-name value-clause.</i></li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 4: Boolean data</b> <i>level-number &lt;data-name-1   FILLER&gt;</i> <i>[redefines-clause   like-clause] [picture-clause] [usage-clause] [occurs-clause] [&lt;INDICATOR   INDICATORS   INDIC&gt; integer-1] [synchronized-clause] [justified-clause] [value-clause]</i></li> </ul>	V	S			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>BLANK [WHEN] ZERO</i></li> </ul>	V	S			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;JUSTIFIED   JUST&gt; [RIGHT]</i></li> </ul>	VO	N			
LIKE clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>LIKE data-name [(integer)]</i></li> </ul>	V	S			
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> <i>OCCURS integer-2 [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> <i>OCCURS integer-1 TO integer-2 [TIMES] DEPENDING [ON] data-name-1 [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;PICTURE   PIC&gt; [IS] character-string</i></li> </ul>	V	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number &lt;data-name-1   FILLER&gt; REDEFINES data-name-2</i></li> </ul>	V	S			

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> USAGE [IS] &lt;BINARY   COMP   COMP-3   COMP-4   COMPUTATIONAL   COMPUTATIONAL-3   COMPUTATIONAL-4   DISPLAY   INDEX   PACKED-DECIMAL   POINTER &gt;</li> </ul>	V	S			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> 88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S			
	<ul style="list-style-type: none"> <li><b>Format 3: NULL value</b> VALUE [IS] &lt;NULL   NULLS&gt;</li> </ul>	V	S			

1-72 Supported COBOL Statements  
Supported COBOL/400 statements

## Procedure Division

### Arithmetic operators

Operation	Meaning	Format	Parser	IA	Sup.	CA/GDF		
						Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S	<i>oper</i>		
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S	<i>oper</i>		

### Conditional Expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   <i>class-name</i>&gt;</li> </ul>	V	S	S			
		VO	N	N			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	cond
	<ul style="list-style-type: none"> <li><b>Format 2: ADDRESS comparison</b> &lt;ADDRESS OF <i>id-1</i>   <i>id-2</i>   NULL   NULLS&gt; [IS] [NOT] &lt;EQUAL [TO]   =&gt; &lt;ADDRESS OF <i>id-3</i>   <i>id-4</i>   NULL   NULLS&gt;</li> </ul>	V	S	S			
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Negated simple conditions	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>condition-1</i></li> </ul>	V	S	S			
Combined conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			
Abbreviated combined relation conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S			

1-74 Supported COBOL Statements  
Supported COBOL/400 statements

**Statements**

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern.using		
PROCEDURE DIVISION structure	<ul style="list-style-type: none"> <li><b>Format:</b> procedure division header [DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.] <i>section-name</i> SECTION [<i>priority-number</i>]. [<i>paragraph-name</i>. [<i>sentence</i>]]</li> </ul>	V	S	S					
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.screen		



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>   <i>environment-name</i>&gt;]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE   DAY   TIME&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: feedback, local data area</b> ACCEPT <i>id</i> FROM <i>mnemonic-name</i> [FOR &lt;<i>file-name</i>   <i>id-2</i>   <i>literal</i>&gt;]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: PIP data area</b> ACCEPT <i>id</i> FROM <i>mnemonic-name</i> [[ON] EXCEPTION <i>imperative-statement-1</i>] [NOT [ON] EXCEPTION <i>imperative-statement-1</i>] [END-ACCEPT]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: attribute data</b> ACCEPT <i>id</i> FROM <i>mnemonic-name</i> [FOR &lt;<i>id-2</i>   <i>literal</i>&gt; [FOR <i>file-name-1</i>]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 6: workstation I/O</b> ACCEPT <i>id</i> FROM <i>mnemonic-name</i> [FROM CRT   MODE [IS] BLOCK   [AT] &lt;LINE [NUMBER] &lt;<i>id-1</i>   <i>integer-1</i>&gt;   &lt;COLUMN   COL [NUMBER] &lt;<i>id-2</i>   <i>integer-2</i>&gt;&gt;   [AT] &lt;<i>id-3</i>   <i>integer-3</i>&gt;   WITH &lt;&lt;AUTO   AUTO-SKIP&gt;   &lt;BELL   BEEP&gt;   BLINK   &lt;FULL   LENGTH-CHECK&gt;   HIGHLIGHT   &lt;REQUIRED   EMPTY-CHECK&gt;   REVERSE-VIDEO   &lt;SECURE   NO-ECHO&gt;   UNDERLINE   LEFT-JUSTIFY   RIGHT-JUSTIFY   SPACE-FILL   TRAILING-SIGN   UPDATE   ZERO-FILL   SIZE [IS] &lt;<i>id-4</i>   <i>integer-4</i>&gt;   PROMPT [CHARACTER IS] &lt;<i>id-5</i>   <i>literal</i>&gt;   FOREGROUND-COLOR [IS] <i>integer-5</i>   BACKGROUND-COLOR [IS] <i>integer-6</i>&gt;] [[ON] EXCEPTION <i>imperative-statement-1</i>] [NOT [ON] EXCEPTION <i>imperative-statement-1</i>] [END-ACCEPT]</li> </ul>	V	S	S					
ACQUIRE	<ul style="list-style-type: none"> <li><b>Format</b> ACQUIRE &lt;<i>id</i>   <i>literal</i>&gt; FOR <i>file-name</i></li> </ul>	V	S	S					

1-76 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;id-1   literal&gt; TO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;id-1   literal-1&gt; TO &lt;id-2   literal-1&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; id-1 TO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S				
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER procedure-name-1 TO [PROCEED TO] procedure-name-2</li> </ul>	V	S	S				
CALL	<ul style="list-style-type: none"> <li><b>Format 1: with ON OVERFLOW</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   file-name-1&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2&gt;&gt;] [ON] OVERFLOW imperative-stmt-1] [END-CALL]</li> </ul>	V	S	S	id-2, id-3 port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
	<ul style="list-style-type: none"> <li><b>Format 2: with ON EXCEPTION</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] id-2   id-3   file-name-1&gt;   [BY] CONTENT &lt;[ADDRESS OF   LENGTH OF] id-3   literal-2&gt;&gt;] [[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 3: graphics support</b> CALL "GDDM" USING routine-name [data-name-1]</li> </ul>	V	S	S				
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					
	[<REEL   UNIT> [[FOR] REMOVAL   WITH NO REWIND]   [WITH] <NO REWIND   LOCK>]	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files, transaction</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					
	[[WITH] LOCK]	VO	N	N					
COMMIT	<ul style="list-style-type: none"> <li><b>Format</b> COMMIT</li> </ul>	V	S	S					
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE <i>id-1</i> [ROUNDED] = <i>arithmetic-expr</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-COMPUTE]</li> </ul>	V	S	S	<i>identifiers of the arithmetic-expr</i>	<i>id-1</i>	comp+ or comp* (according to the operation performed)		
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S					
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [[INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Deletes From File

1-78 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	CA/GDF				Repository	
				Sup.	Pairs		Relation	Entities	Relations
DISPLAY	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> DISPLAY &lt;id-1   literal-1&gt; [UPON &lt;mnemonic-name-1   environment-name-1&gt;]</li> </ul>	V	S	S	id-1	port	extern.screen		
	<ul style="list-style-type: none"> <li><b>Format 2: local data area</b> DISPLAY &lt;id-1   literal-1&gt; UPON mnemonic-name-1 [[FOR] &lt;id-2   literal-2&gt;]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: workstation I/O</b> DISPLAY &lt;&lt;id-1   literal-1&gt;   UPON &lt;CRT   CRT-UNDER&gt;   MODE [IS]   BLOCK   [AT] &lt;LINE [NUMBER] &lt;id-1   integer-1&gt;   &lt;COLUMN   COL&gt; [NUMBER] &lt;id-2   integer-2&gt;&gt;   [AT] &lt;id-3   nteger-3&gt;   WITH &lt;&lt;BELL   BEEP&gt;   BLINK   HIGHLIGHT   REVERSE-VIDEO   UNDERLINE   BLANK &lt;SCREEN   LINE&gt;   SIZE [IS] &lt;id-4   integer-4&gt;   FOREGROUND-COLOR [IS] integer-5   BACKGROUND-COLOR [IS] integer-6&gt;&gt;</li> </ul>	V	S	S					
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;id-1   literal-1&gt; INTO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-1	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b>            DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt;            GIVING id-3 [ROUNDED] REMAINDER id-4            [[ON] SIZE ERROR imperative-stmt-1]            [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S					
DROP	<ul style="list-style-type: none"> <li><b>Format</b>            DROP &lt;id   literal&gt; FROM file-name</li> </ul>	V	S	S					
ENTER	<ul style="list-style-type: none"> <li><b>Format</b>            ENTER language-name [routine-name] [.]</li> </ul>	V	S	S					
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b>            EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;            WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1            [WHEN OTHER imperative-stmt-2] [END-EVALUATE]            Phrases phrase-1 and phrase-2 should be represented in the following form:            &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-3   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S					
EXIT	<ul style="list-style-type: none"> <li><b>Format</b>            paragraph-name. EXIT.</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b>            EXIT PROGRAM.</li> </ul>	V	S	S					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b>            GOBACK</li> </ul>	V	S	S					

1-80 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] <i>procedure-name-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] <i>procedure-name-1</i> DEPENDING [ON] <i>id-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: altered</b> <i>paragraph-name</i>. GO [TO].</li> </ul>	V	S	S					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF <i>condition-1</i> THEN &lt;<i>stmt-1</i>   NEXT SENTENCE&gt; [ELSE &lt;<i>stmt-2</i>   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S					
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i></li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	USING <i>file-name-2</i> <i>file-name-3</i> <OUTPUT PROCEDURE [IS] <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> ]   GIVING <i>file-name-4</i> >	V	S	S					

1-82 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;id-1   literal-1&gt; TO id-2</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; id-1 TO id-2</li> </ul>	V	S	S					
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;id-1   literal-1&gt; BY id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-MULTIPLY]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-MULTIPLY]</li> </ul>	V	S	S					
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT file-name-1  </li> </ul>	V	S	S					
	[REVERSED   [WITH] NO REWIND]	VO	N	N					
	OUTPUT file-name-2	V	S	S					
	[[WITH] NO REWIND]	VO	N	N					
	I-O file-name-3   EXTEND file-name-4>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> OPEN &lt;INPUT file-name-1   OUTPUT file-name-2   I-O file-name-3   EXTEND file-name-4&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: transaction</b> OPEN I-O file-name</li> </ul>	V	S	S					



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]   imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] &lt;id-1   integer-1&gt; TIMES   &lt;id-1   integer-1&gt; TIMES imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with UNTIL phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: with VARYING phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 6: with VARYING phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 [AFTER &lt;id-5   index-name-3&gt; FROM &lt;id-6   index-name-4   literal-3&gt; BY &lt;id-7   literal-4&gt; UNTIL condition-2]   &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-2</li> </ul>	N	N	N					

1-84 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b> READ <i>file-name-1</i></li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
	[NEXT   FIRST   LAST   PRIOR] [RECORD]	VO	N	N					
	[[WITH] NO LOCK] [FORMAT [IS] < <i>id-1</i>   <i>literal-1</i> >]	V	S	S					
	[INTO <i>id-2</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-READ]	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b> READ <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [[WITH] NO LOCK] [KEY [IS] <i>data-name-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [INVALID [KEY] <i>imperative-stmt-3</i>] [ NOT INVALID [KEY] <i>imperative-stmt-4</i>] [END-READ]</li> </ul>	V	S	S					
<ul style="list-style-type: none"> <li><b>Format 3: transaction (non-subfile)</b> READ <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [TERMINAL [IS] &lt;<i>id-3</i>   <i>literal-2</i>&gt;] [&lt;INDICATOR   INDICATORS   INDIC&gt; &lt;IS   ARE&gt; <i>id-4</i>] [NO DATA <i>imperative-stmt-1</i>] [INVALID [KEY] <i>imperative-stmt-2</i>] [ NOT INVALID [KEY] <i>imperative-stmt-3</i>] [END-READ]</li> </ul>	V	S	S				refer to literal-1 (Screen)	Program Receives Screen	
<ul style="list-style-type: none"> <li><b>Format 4: transaction (subfile)</b> READ <i>file-name-1</i> [[NEXT] MODIFIED] [RECORD] [INTO <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [TERMINAL [IS] &lt;<i>id-3</i>   <i>literal-2</i>&gt;] [&lt;INDICATOR   INDICATORS   INDIC&gt; &lt;IS   ARE&gt; <i>id-4</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [ NOT INVALID [KEY] <i>imperative-stmt-2</i>] [[AT] END <i>imperative-stmt-3</i>] [NOT [AT] END <i>imperative-stmt-4</i>] [END-READ]</li> </ul>	V	S	S				defines id-2 (Decision)	Program Receives Decision	
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
RETURN	<ul style="list-style-type: none"> <li><b>Format</b> RETURN <i>file-name-1</i> [RECORD] [INTO <i>id-1</i>] [[AT] END <i>imperative-stmt-1</i>] [NOT [AT] END <i>imperative-stmt-2</i>] [END-RETURN]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph) Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format 1:</b> REWRITE <i>file-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-REWRITE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph) Program Updates File
	<ul style="list-style-type: none"> <li><b>Format 2: transaction (subfile)</b> REWRITE SUBFILE <i>file-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [TERMINAL [IS] &lt;<i>id-3</i>   <i>literal-2</i>&gt;] [&lt;INDICATOR   INDICATORS   INDIC&gt; &lt;IS   ARE&gt; <i>id-4</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-REWRITE]</li> </ul>	V	S	S				<i>refer to literal-1 (Screen)</i> <i>defines id-2 (Decision)</i> Program Sends Screen Program Sends Decision
ROLLBACK	<ul style="list-style-type: none"> <li><b>Format</b> ROLLBACK</li> </ul>	V	S	S				
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S				
SEARCH	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S				

1-86 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;index-name-1   id-1&gt; TO &lt;index-name-2   id-2   integer-1&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET index-name-3 &lt;UP BY   DOWN BY&gt; &lt;id-3   integer-2&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET mnemonic-name-1 TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET condition-name-1 TO TRUE</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: USAGE IS POINTER data items</b> SET &lt;id-4   ADDRESS OF id-5&gt; TO &lt;id-6   ADDRESS OF id-7   NULL   NULLS&gt;</li> </ul>	V	S	S					
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT file-name-1 [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] data-name-1 [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] alphabet-name-1]	VO	N	N					
	<ul style="list-style-type: none"> <li>&lt;USING file-name-2   INPUT PROCEDURE [IS] procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]&gt; &lt;GIVING file-name-3   OUTPUT PROCEDURE [IS] procedure-name-3 [&lt;THROUGH   THRU&gt; procedure-name-4]&gt;</li> </ul>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START file-name-1 [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; &lt;EXTERNALLY-DESCRIBED-KEY   data-name-1&gt;] [FORMAT [IS] &lt;id-1   literal-1&gt;] [FORMAT [IS] &lt;id-2   literal-1&gt;] [INVALID [KEY] imperative-stmt-1] [NOT INVALID [KEY] imperative-stmt-2] [END-START]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S				
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S				
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b> SUBTRACT &lt;<i>id-1</i>   <i>literal-1</i>&gt; FROM <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-SUBTRACT]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> SUBTRACT &lt;<i>id-1</i>   <i>literal-1</i>&gt; FROM &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-SUBTRACT]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 3</b> SUBTRACT &lt;CORRESPONDING   CORR&gt; <i>id-1</i> FROM <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-SUBTRACT]</li> </ul>	V	S	S				
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b> UNSTRING <i>id-1</i> [DELIMITED [BY] [ALL] &lt;<i>id-2</i>   <i>literal-1</i>&gt; [OR [ALL] &lt;<i>id-3</i>   <i>literal-2</i>&gt;]] INTO <i>id-4</i> [DELIMITER [IN] <i>id-5</i>] [COUNT [IN] <i>id-6</i>] [[WITH] POINTER <i>id-7</i>] [TALLYING [IN] <i>id-8</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-UNSTRING]</li> </ul>	V	S	S				

1-88 Supported COBOL Statements  
Supported COBOL/400 statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files (basic)</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [END-WRITE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
	<ul style="list-style-type: none"> <li><b>Format 2: sequential files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] &lt;&lt;BEFORE   AFTER&gt; [ADVANCING] &lt;&lt;<i>id-1</i>   <i>integer-1</i>&gt; [LINE   LINES]   <i>mnemonic-name-1</i>   PAGE&gt;] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>] [END-WRITE]</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: indexed and relative files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: format file</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [&lt;INDICATOR   INDICATORS   INDIC&gt; [IS   ARE] <i>id-3</i>] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>] [END-WRITE]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: transactions</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [TERMINAL [IS] &lt;<i>id-3</i>   <i>literal-2</i>&gt;] [STARTING [AT] [LINE] &lt;<i>id-4</i>   <i>literal-3</i>&gt;] [&lt;BEFORE   AFTER&gt; ROLLING [LINE   LINES] &lt;<i>id-5</i>   <i>literal-4</i>&gt; [THROUGH   THRU] &lt;<i>id-6</i>   <i>literal-5</i>&gt; &lt;UP   DOWN&gt; &lt;<i>id-7</i>   <i>literal-6</i>&gt; [LINE   LINES]] [&lt;INDICATOR   INDICATORS   INDIC&gt; [IS   ARE] <i>id-3</i>] [END-WRITE]</li> </ul>	V	S	S				refer to <i>literal-1</i> (Screen)  <i>defines id-2</i> (Decision)	Program Sends Screen  Program Sends Decision
	<ul style="list-style-type: none"> <li><b>Format 5: transactions</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [FORMAT [IS] &lt;<i>id-2</i>   <i>literal-1</i>&gt;] [TERMINAL [IS] &lt;<i>id-3</i>   <i>literal-2</i>&gt;] [&lt;INDICATOR   INDICATORS   INDIC&gt; [IS   ARE] <i>id-3</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S					

**Compile-directing statements and directives**

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
*CONTROL (*CBL)	<ul style="list-style-type: none"> <li><b>Format</b> &lt;*CONTROL   *CBL&gt; &lt;SOURCE   NOSOURCE   LIST   NOLIST   MAP   NOMAP&gt;</li> </ul>	VO	N		
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; library-name [file-name]] [SUPPRESS] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2: DDS translate</b> COPY &lt;DD-format-name   DD-ALL-FORMATS   DDR-format-name   DDR-ALL-FORMATS   DDS-format-name   DDS-ALL-FORMATS   DDSR-format-name   DDSR-ALL-FORMATS&gt; [-I   -O   -I-O] [-INDICATOR   -INDICATORS   -INDIC] [&lt;OF   IN&gt; library-name&gt;] file-name [SUPPRESS] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S		
EJECT	<ul style="list-style-type: none"> <li><b>Format</b> EJECT [.]</li> </ul>	VO	N		
SKIP1/2/3	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SKIP1   SKIP2   SKIP3&gt;.</li> </ul>	VO	N		
TITLE	<ul style="list-style-type: none"> <li><b>Format</b> TITLE <i>literal</i>.</li> </ul>	VO	N		
USE	<ul style="list-style-type: none"> <li><b>Format 1: EXCEPTION ERROR declarative</b> USE AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: LABEL declarative</b> USE [GLOBAL] AFTER [STANDARD] [BEGINNING   ENDING] [FILE   REEL   UNIT] LABEL PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: DEBUGGING declarative</b> USE [FOR] DEBUGGING [ON] &lt;procedure-name-1   ALL PROCEDURES   file-name   ALL [REFERENCES OF] id-1&gt;</li> </ul>	VO	N		

## **Supported Unisys ASCII COBOL statements**

### **COBOL language structure**

#### **Figurative constants**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### **Special registers**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
DEBUG ITEM	V	S	S
LINAGE-COUNTER	V	S	S
LINE-COUNTER	V	S	S
PAGE-COUNTER	V	S	S



### Literals

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literal	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; <i>mantissa</i> E &lt;+   -&gt; <i>exponent</i></li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

### Referencing names

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S

**1-92** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li> <b>Format 1: subscripting</b>  <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>) </li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li> <b>Format 2</b>  &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] </li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li> <b>Format 3</b>  LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>] </li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li> <b>Format 4</b>  &lt;PAGE-COUNTER   LINE-COUNTER&gt; [&lt;IN   OF&gt; <i>report-name</i>] </li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li> <b>Format: Data Division</b>  <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) </li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li> <b>Format</b>  &lt;<i>integer-1</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>&gt; </li> </ul>	V	S	S treated as reference to the whole array
Reference modification	<ul style="list-style-type: none"> <li> <b>Format</b>  &lt;<i>data-name-1</i>&gt; (<i>leftmost-character-position: [length]</i>) </li> </ul>	V	S	S treated as reference to whole <i>data-name-1</i>

**Control Division**

Names	Format	Parser	IA
Control Division Entry	<ul style="list-style-type: none"> <li> <b>Format</b>  [CONTROL DIVISION].  [ALPHABET SECTION].  [SOURCE-ALPHABET CHARACTERS ARE <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>]] </li> </ul>	N	N

Names	Format	Parser	IA
Inline Section	<ul style="list-style-type: none"> <li><b>Format</b> APPLY INLINE &lt;ADD   ALL   CONVERSION   IF   INSPECT   MOVE   SIGN&gt; FOR &lt;ALL PROCEDURES   <i>procedure-name-1</i>&gt; [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]</li> </ul>	N	N

### Identification Division

Names	Format	Parser	IA	Repository	
				Entities	Relation
Identification Division	<ul style="list-style-type: none"> <li><b>Format</b> IDENTIFICATION DIVISION. PROGRAM-ID <i>program-name</i>. [AUTHOR. [<i>comment-entry</i>]] [INSTALLATION. [<i>comment-entry</i>]] [DATE-WRITTEN. [<i>comment-entry</i>]] [DATE-COMPILED. [<i>comment-entry</i>]] [SECURITY. [<i>comment-entry</i>]]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point
	[REMARKS. [ <i>comment-entry</i> ]]	VO	N		

### Environment Division

#### Input-output section

Paragraphs and clauses	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN TO <i>assignment-name-1</i></li> </ul>	V	S	defines program- name.assignment- name-1 (File)	
	[; RESERVE <i>integer</i> [AREA   AREAS]] [; ORGANIZATION IS SEQUENTIAL] [; ACCESS MODE IS SEQUENTIAL] [FILE STATUS IS <i>data-name-1</i> ].	VO	N		

**1-94** Supported COBOL Statements  
*Supported Unisys ASCII COBOL statements*

	<ul style="list-style-type: none"> <li><b>Format 2: indexed files</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN TO <i>assignment-name-1</i>	V	S	defines program-name.assignment-name-1 (File)	
	[; RESERVE <i>integer</i> [AREA   AREAS]] ; ORGANIZATION IS INDEXED [; ACCESS MODE IS <SEQUENTIAL   RANDOM   DYNAMIC-> ; RECORD KEY IS <i>data-name-2</i> ; ALTERNATE RECORD KEY IS <i>data-name-3</i> [WITH DUPLICATES] [; FILE STATUS IS <i>data-name-1</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: relative files</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN TO <i>assignment-name-1</i>	V	S	defines program-name.assignment-name-1 (File)	
	[; RESERVE <i>integer</i> [AREA   AREAS]] ; ORGANIZATION IS RELATIVE [; ACCESS MODE IS <SEQUENTIAL [, RELATIVE KEY IS <i>data-name-1</i> ]   <RANDOM   DYNAMIC> RELATIVE KEY IS <i>data-name-1</i> >] [; FILE STATUS IS <i>data-name-2</i> ].	VO	S		
	<ul style="list-style-type: none"> <li><b>Format 4: sort-merge</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN TO <i>assignment-name-1</i> .	V	S		
	<ul style="list-style-type: none"> <li><b>Format 5: IBM tape processing</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name</i> ASSIGN TO <i>assignment-name</i>	V	S	defines program-name.assignment-name (File)	
	[; RESERVE < <i>integer</i>   NO> ALTERNATE [AREA   AREAS]] [; ACCESS MODE IS SEQUENTIAL] [; ORGANIZATION IS SEQUENTIAL].	VO	N		

I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li>• <b>Format 1: sequential files</b></li> </ul> <p>I-O CONTROL.                  [: RERUN ON &lt;assignment-name-1   file-name-1&gt; EVERY &lt;&lt;integer-1 RECORDS   END OF &lt;REEL   UNIT&gt;&gt; OF file-name-1   integer-2 CLOCK-UNITS   condition-name&gt;]                  [: SAME [RECORD] AREA FOR file-name-3]                  [: MULTIPLE FILE TAPE CONTAINS file-name-5 [POSITION integer-3]].</p>	VO	N		
	<ul style="list-style-type: none"> <li>• <b>Format 2: indexed and relative files</b></li> </ul> <p>I-O CONTROL.                  [RERUN ON &lt;assignment-name-1   file-name-1&gt; EVERY &lt;integer-1 RECORDS OF file-name-1   integer-2 CLOCK-UNITS   condition-name&gt;]                  [: SAME [RECORD] AREA FOR file-name-3].</p>	VO	N		
	<ul style="list-style-type: none"> <li>• <b>Format 3: sort-merge</b></li> </ul> <p>I-O CONTROL.                  [: SAME &lt;RECORD   SORT   SORT-MERGE&gt; AREA FOR file-name-1].</p>	VO	N		
	<ul style="list-style-type: none"> <li>• <b>Format 4: IBM tape processing</b></li> </ul> <p>I-O CONTROL.                  [: RERUN ON assignment-name-1 EVERY &lt;integer-1 RECORDS   [END OF] REEL&gt; OF file-name-1]                  [: SAME [RECORD] AREA FOR file-name-2]</p>	VO	N		
	<ul style="list-style-type: none"> <li>• <b>Format 5: direct I-O</b></li> </ul> <p>I-O CONTROL.                  [: APPLY INTERLOCK ON file-name-1]                  [: APPLY PROTECT ON file-name-2 [FOR integer-3 RECORDS]]</p>	VO	N		
	<ul style="list-style-type: none"> <li>• <b>Format 6: indexed sequential I-O</b></li> </ul> <p>I-O CONTROL.                  [: APPLY INTERLOCK ON file-name-1]                  [: APPLY PROTECT ON file-name-2 [FOR integer-3 RECORDS]]                  [: APPLY CORE-INDEX ON file-name-3]</p>	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA
OBJECT-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> OBJECT-COMPUTER. computer-name [, MEMORY SIZE <i>integer</i> &lt;WORDS   CHARACTERS   MODULES&gt;] [, PROGRAM COLLATING SEQUENCE IS <i>alphabet-name</i>].</li> </ul>	VO	N
SOURCE-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE-COMPUTER. computer-name.</li> </ul>	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [, <i>implementor-name</i> &lt;IS <i>mnemonic-name</i> [, ON STATUS IS <i>condition-name-1</i>, [OFF STATUS IS <i>condition-name-2</i>]]   IS <i>mnemonic-name</i> [, OFF STATUS IS <i>condition-name-2</i>, [ON STATUS IS <i>condition-name-1</i>]]   ON STATUS IS <i>condition-name-1</i>, [OFF STATUS IS <i>condition-name-2</i>]   OFF STATUS IS <i>condition-name-2</i>, [ON STATUS IS <i>condition-name-1</i>]-&gt;]</li> </ul>	V	S
	[, <i>alphabet-name</i> IS <STANDARD-1   NATIVE   <i>implementor-name</i>   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> ]->]	VO	N
	[, CURRENCY SIGN IS <i>literal-4</i> ]	VO	N
	[, DECIMAL POINT IS COMMA]	V	S
	[, CLASS NAME. IS [, <i>mnemonic-name</i> ] [VALUE IS <i>literal-5</i> [<THROUGH   THRU> <i>literal-6</i> ]]]	VO	N
ACTUAL KEY clause	<ul style="list-style-type: none"> <li><b>Format</b> ACTUAL KEY IS <i>data-name</i></li> </ul>	VO	N
APPLY clause	<ul style="list-style-type: none"> <li><b>Format</b> APPLY &lt;EXREF   EXDEF&gt; ON <i>id-1</i></li> </ul>	VO	N
APPLY WRITE-ONLY clause	<ul style="list-style-type: none"> <li><b>Format</b> APPLY WRITE-ONLY ON <i>file-name-1</i></li> </ul>	VO	N
ASSIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> ASSIGN TO <i>assignment-name</i></li> </ul>	V	S
FILE-LIMITS clause	<ul style="list-style-type: none"> <li><b>Format</b> [; &lt;FILE-LIMIT IS   FILE-LIMITS ARE&gt; &lt;<i>data-name-1</i>   <i>literal-1</i>&gt; &lt;THROUGH   THRU&gt; &lt;<i>data-name-2</i>   <i>literal-2</i>&gt;] [<i>data-name-3</i>   <i>literal-3</i>] &lt;THROUGH   THRU&gt; &lt;<i>data-name-4</i>   <i>literal-4</i>&gt;]</li> </ul>	VO	N

PROCESSING MODE clause	<ul style="list-style-type: none"> <li><b>Format</b> PROCESSING MODE IS SEQUENTIAL</li> </ul>	VO	N
RERUN clause	<ul style="list-style-type: none"> <li><b>Format</b> RERUN [ON &lt;file-name-1   UNISERVO   UNISERVOS   MASS-STORAGE&gt;] EVERY &lt;&lt;integer-1 RECORDS   END OF &lt;REEL   UNIT&gt;&gt; OF file-name-1</li> </ul>	VO	N
RESERVE clause	<ul style="list-style-type: none"> <li><b>Format</b> RESERVE &lt;integer-1   NO&gt; ALTERNATE [AREA   AREAS]</li> </ul>	VO	N
SEGMENT LIMIT clause	<ul style="list-style-type: none"> <li><b>Format</b> SEGMENT LIMIT IS <i>segment-number</i></li> </ul>	VO	N
WITH DEBUGGING MODE clause	<ul style="list-style-type: none"> <li><b>Format</b> WITH DEBUGGING MODE</li> </ul>	VO	N
WITH RANK clause	<ul style="list-style-type: none"> <li><b>Format</b> WITH RANK OF <i>integer</i></li> </ul>	VO	N

### Data Division

Entries and Clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Communication Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> CD <i>cd-name</i>; FOR [INITIAL] INPUT [; SYMBOLIC QUEUE IS <i>data-name-1</i>] [; SYMBOLIC SUB-QUEUE-1 IS <i>data-name-2</i>] [; SYMBOLIC SUB-QUEUE-2 IS <i>data-name-3</i>] [; SYMBOLIC SUB-QUEUE-3 IS <i>data-name-4</i>] [; MESSAGE DATE IS <i>data-name-5</i>] [; MESSAGE TIME IS <i>data-name-6</i>] [; SYMBOLIC SOURCE IS <i>data-name-7</i>] [; TEXT LENGTH IS <i>data-name-8</i>] [; END KEY IS <i>data-name-9</i>] [; STATUS KEY IS <i>data-name-10</i>] [; MESSAGE COUNT IS <i>data-name-11</i>]</li> </ul>	N	N	N		

**1-98** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Entries and Clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
	<ul style="list-style-type: none"> <li><b>Format 2</b> CD <i>cd-name</i>; FOR OUTPUT [; DESTINATION COUNT IS <i>data-name-1</i>] [; TEXT LENGTH IS <i>data-name-2</i>] [; STATUS KEY IS <i>data-name-3</i>] [; DESTINATION TABLE OCCURS <i>integer-2</i> TIMES [; INDEXED BY <i>index-name-1</i>]] [; ERROR KEY IS <i>data-name-4</i>] [; SYMBOLIC DESTINATION IS <i>data-name-5</i>]</li> </ul>	N	N	N			
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; [; REDEFINES <i>data-name-2</i>] [; &lt;PICTURE   PIC&gt; IS <i>character-string</i>] [[USAGE IS] &lt;COMPUTATIONAL   COMP   DISPLAY   COMPUTATIONAL-1   COMP-1   COMPUTATIONAL-2   COMP-2   DISP&gt;] [; [SIGN IS] &lt;LEADING   TRAILING&gt; [SEPARATE CHARACTER]] [; &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]] [; &lt;JUSTIFIED   JUST&gt; RIGHT] [; BLANK WHEN ZERO] [; VALUE IS <i>literal</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> 66 <i>data-name-1</i>; RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3</b> 88 <i>condition-name</i>; &lt;VALUE IS   VALUES ARE&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>] [, <i>literal-3</i> [&lt;THROUGH   THRU&gt; <i>literal-4</i>]].</li> </ul>	V	S	S			



Entries and Clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> FD <i>file-name</i></li> </ul>	V	S	S			
	<pre>[; BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;] [; RECORD CONTAINS [<i>integer-3</i> TO] <i>integer-4</i> CHARACTERS] ; LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt; [; VALUE OF <i>system-name-1</i> IS &lt;<i>data-name-1</i>   <i>literal-1</i>&gt;] [; DATA &lt;RECORD IS   RECORDS ARE&gt; <i>data-name-2</i>] [; LINAGE IS &lt;<i>data-name-3</i>   <i>integer-5</i>&gt; LINES [, WITH FOOTING AT &lt;<i>data-name-4</i>   <i>integer-6</i>&gt;] [, LINES AT TOP &lt;<i>data-name-6</i>   <i>integer-7</i>&gt;] [, LINES AT BOTTOM &lt;<i>data-name-7</i>   <i>integer-8</i>&gt;]] [; CODE-SET IS <i>alphabet-name</i>] DENSITY IS &lt;6   8   12&gt; LINES.</pre>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> FD <i>file-name</i></li> </ul>	V	S	S			
	<pre>[; BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;] [; RECORD CONTAINS [<i>integer-3</i> TO] <i>integer-4</i> CHARACTERS] ; LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt; [; VALUE OF <i>system-name-1</i> IS &lt;<i>data-name-1</i>   <i>literal-1</i>&gt;] [; DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-2</i>].</pre>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3: IBM tape processing</b> FD <i>file-name</i></li> </ul>	V	S	S			
	<pre>[; BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;] [; RECORD CONTAINS [<i>integer-3</i> TO] <i>integer-4</i> CHARACTERS] ; LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt; [; VALUE OF <i>system-name-1</i> IS &lt;<i>data-name-1</i>   <i>literal-1</i>&gt;] [; DATA &lt;RECORD [IS]   RECSORDS [ARE]&gt; <i>data-name-2</i>.] ; RECORDING MODE IS &lt;E   U   V&gt; [AN]</pre>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 4: sort-merge</b> SD <i>file-name</i></li> </ul>	V	S	S			
	<pre>[; RECORD CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> CHARACTERS] [; DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-1</i>].</pre>	VO	N	N			

**1-100** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
	<ul style="list-style-type: none"> <li><b>Format 5: report writer</b> FD <i>file-name</i></li> </ul>	V	S	S			
	<pre>[; BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;] [; RECORD CONTAINS [<i>integer-3</i> TO] <i>integer-4</i> CHARACTERS] [; LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt;] [; VALUE OF <i>system-name-1</i> IS &lt;<i>data-name-1</i>   <i>literal-1</i>&gt;] [; CODE-SET IS <i>alphabet-name</i>] [; &lt;REPORT IS   REPORTS ARE&gt; <i>report-name</i> DENSITY IS &lt;6   8   12&gt; LINES.</pre>	VO	N	N			
Report Description (RD) Entry	<ul style="list-style-type: none"> <li><b>Format</b> RD <i>report-name</i> [; CODE <i>literal-1</i>] [; &lt;CONTROL IS   CONTROLS ARE&gt; &lt;<i>data-name-1</i>   FINAL [; <i>data-name-1</i>]&gt;] [; PAGE [LIMIT IS   LIMITS ARE] <i>integer-1</i> [LINE   LINES] [, HEADING <i>integer-2</i>] [, FIRST DETAIL <i>integer-3</i>] [, LAST DETAIL <i>integer-4</i>] [, FOOTING <i>integer-5</i>].</li> </ul>	V	S	N			
Report Group Description	<ul style="list-style-type: none"> <li><b>Format 1</b> 01 [<i>data-name-1</i>] [; LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [; NEXT GROUP IS &lt;<i>integer-3</i>   PLUS <i>integer-4</i>   NEXT PAGE&gt;] [; TYPE IS &lt;REPORT HEADING   RH   PAGE HEADING   PH   &lt;CONTROL HEADING   CH&gt; &lt;<i>data-name-2</i>   FINAL&gt;   DETAIL   DE   &lt;CONTROL FOOTING   CF&gt; &lt;<i>data-name-3</i>   FINAL&gt;   PAGE FOOTING   PF   REPORT FOOTING   RF&gt; [; [USAGE IS] &lt;DISPLAY   DISP&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>level-number</i> [<i>data-name-1</i>] [; LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [; [USAGE IS] &lt;DISPLAY   DISP&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>level-number</i> [<i>data-name-1</i>] [; BLANK WHEN ZERO] [; GROUP INDICATE] [; &lt;JUSTIFIED   JUST&gt; RIGHT] [; LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [; COLUMN NUMBER IS <i>integer-3</i>] [; &lt;PICTURE   PIC&gt; IS <i>character-string</i>; SOURCE IS <i>id-1</i> &lt;; VALUE IS <i>literal</i>   ; SUM <i>id-2</i> [UPON <i>data-name-2</i>] [; RESET ON &lt;<i>data-name-3</i>   FINAL&gt;] [; [USAGE IS] &lt;DISPLAY   DISP&gt;]</li> </ul>	V	S	N			

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Saved-Area Description	<ul style="list-style-type: none"> <li><b>Format</b> SA <i>area-name</i> ; AREA CONTAINS <i>integer-1</i> RECORDS [; WITH RANK OF <i>integer-2</i>]</li> </ul>	N	N	N			
AREA CONTAINS clause	<ul style="list-style-type: none"> <li><b>Format</b> AREA CONTAINS <i>integer-1</i> RECORDS</li> </ul>	N	N	N			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> BLANK WHEN ZERO</li> </ul>	V	S	N			
BLOCK CONTAINS clause	<ul style="list-style-type: none"> <li><b>Format</b> BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;</li> </ul>	N	N	S			
CODE clause	<ul style="list-style-type: none"> <li><b>Format</b> CODE <i>literal-1</i></li> </ul>	V	S	N			
CODE-SET clause	<ul style="list-style-type: none"> <li><b>Format</b> CODE-SET IS <i>alphabet-name</i></li> </ul>	V	S	N			
COLUMN NUMBER clause	<ul style="list-style-type: none"> <li><b>Format</b> COLUMN NUMBER IS <i>integer-1</i></li> </ul>	V	S	N			
CONTROL clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;CONTROL IS   CONTROLS ARE&gt; <i>data-name-1</i>   FINAL&gt;</li> </ul>	V	S	N			
Data-name or FILLER clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>data-name</i>   FILLER&gt;</li> </ul>	V	S	S			
DATA RECORDS clause	<ul style="list-style-type: none"> <li><b>Format</b> DATA &lt;RECORD IS   RECORDS ARE&gt; <i>data-name</i></li> </ul>	V	S	S			
DENSITY clause	<ul style="list-style-type: none"> <li><b>Format</b> DENSITY IS &lt;6   8   12&gt; LINES</li> </ul>	VO	N	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;JUSTIFIED   JUST&gt; RIGHT</li> </ul>	V	S	N			

**1-102** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
LABEL RECORDS clause	<ul style="list-style-type: none"> <li><b>Format</b> LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt;</li> </ul>	V	S	S		
LINAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> LINAGE IS &lt;data-name-1   integer-1&gt; LINES [, WITH FOOTING AT &lt;data-name-2   integer-2&gt;] [, LINES AT TOP &lt;data-name-3   integer-3&gt;] [, LINES AT BOTTOM &lt;data-name-4   integer-4&gt;]</li> </ul>	N	N	N		
LINE NUMBER clause	<ul style="list-style-type: none"> <li><b>Format</b> LINE NUMBER IS &lt;integer-1 [ON NEXT PAGE]   PLUS integer-2&gt;</li> </ul>	V	S	N		
NEXT GROUP clause	<ul style="list-style-type: none"> <li><b>Format</b> NEXT GROUP IS &lt;integer-1   PLUS integer-2   NEXT PAGE&gt;</li> </ul>	V	S	N		
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> OCCURS integer-2 TIMES [&lt;ASCENDING   DESCENDING&gt; KEY IS data-name-2] [INDEXED BY index-name-1]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> OCCURS integer-1 TO integer-2 TIMES DEPENDING ON data-name-1 [&lt;ASCENDING   DESCENDING&gt; KEY IS data-name-2] [INDEXED BY index-name-1]</li> </ul>	V	S	S		
PAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> PAGE [LIMIT IS   LIMITS ARE] integer-1 [LINE   LINES] [, HEADING integer-2] [, FIRST DETAIL integer-3] [, LAST DETAIL integer-4] [, FOOTING integer-5]</li> </ul>	V	S	N		
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; IS character-string</li> </ul>	V	S	S		
POINT LOCATION clause	<ul style="list-style-type: none"> <li><b>Format</b> POINT LOCATION IS &lt;LEFT   RIGHT&gt; integer-1 &lt;PLACE   PLACES&gt;</li> </ul>	N	N	N		
RECORD CONTAINS clause	<ul style="list-style-type: none"> <li><b>Format</b> RECORD CONTAINS integer-1 TO integer-2 CHARACTERS</li> </ul>	VO	N	N		

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
RECORDING MODE clause	<ul style="list-style-type: none"> <li><b>Format 1: IBM tape processing</b> RECORDING MODE IS &lt;E   U   V&gt; [AN]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: sequential I-O</b> RECORDING MODE IS &lt;&lt;LION   CFH&gt; [AN]   &lt;FORMO1   FORMO2   FORMO3&gt; [U]   BLANK [SIGN]   COMPACT   INTERNAL   SIGN   SDF&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: direct I-O</b> RECORDING MODE IS &lt;BLANK [SIGN]   COMPACT   INTERNAL   SIGN   SDF&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 4: Information Interchange tape processing</b> RECORDING MODE IS &lt;D   F   S   U&gt;</li> </ul>	V	S	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number data-name-1; REDEFINES data-name-2</i></li> </ul>	V	S	S			
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>66 data-name-1; RENAMES data-name-2 [&lt;THROUGH   THRU&gt; data-name-3].</i></li> </ul>	V	S	S			
REPORT clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;REPORT IS   REPORTS ARE&gt; report-name</i></li> </ul>	VO	N	N			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>[SIGN IS] &lt;LEADING   TRAILING&gt; [SEPARATE CHARACTER]</i></li> </ul>	V	S	S			
SOURCE clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>SOURCE IS id-1</i></li> </ul>	V	S	N			
SUM clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>SUM id-1 [UPON data-name-2] [RESET ON &lt;data-name-3   FINAL&gt;]</i></li> </ul>	V	S	N			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;SYNCHRONIZE   SYNC&gt; [LEFT   RIGHT]</i></li> </ul>	V	S	N			

**1-104** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
TYPE clause	<ul style="list-style-type: none"> <li><b>Format</b> TYPE IS &lt;REPORT HEADING   RH   PAGE HEADING   PH   &lt;CONTROL HEADING   CH&gt; &lt;data-name-2   FINAL&gt;   DETAIL   DE   &lt;CONTROL FOOTING   CF&gt; &lt;data-name-3   FINAL&gt;   PAGE FOOTING   PF   REPORT FOOTING   RF&gt;</li> </ul>	V	S	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: general</b> [USAGE IS] &lt;COMPUTATIONAL   COMP   DISPLAY   COMPUTATIONAL-1   COMP-1   COMPUTATIONAL-2   COMP-2   DISP&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: table handling</b> [USAGE IS] INDEX</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: asynchronous processing</b> [USAGE IS] LOCK</li> </ul>	N	N	N			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1</b> VALUE IS <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li><b>Format 2</b> &lt;VALUE IS   VALUES ARE&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>] [, <i>literal-3</i> [&lt;THROUGH   THRU&gt; <i>literal-4</i>]</li> </ul>	V	S	S			
VALUE OF clause	<ul style="list-style-type: none"> <li><b>Format 1</b> VALUE OF <i>system-name-1</i> IS &lt;data-name-1   <i>literal-1</i>&gt;</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2: sequential and direct I-O</b> VALUE OF &lt;<i>system-name-1</i>   FILE-ID   SET-ID   CREATION-DATE   PURGE-DATE   FILE-ACCESS   FILE-QUALIFIER&gt; IS &lt;data-name-1   <i>literal-1</i>&gt;</li> </ul>	VO	N	N			

## Procedure Division

### Arithmetic operators

Operation	Meaning	Format	Parser	IA	Sup.	CA/GDF		
						Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

### Conditional Expressions

Conditions	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id IS [NOT] &lt;NUMERIC   ALPHABETIC   CLASS-NAME mnemonic-name&gt;</i></li> </ul>	V	S	S			

**1-106** Supported COBOL Statements  
*Supported Unisys ASCII COBOL statements*

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs		Relation
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 IS [NOT] &lt;GREATER THAN   LESS THAN   EQUAL TO&gt; oper-2</i> <i>oper-1: &lt;id-1   literal-1   arithmetic-expr-1&gt;</i> <i>oper-2: id-2   literal-2   arithmetic-expr-2&gt;</i></li> </ul>	V	S	S	<i>oper-1</i>	<i>oper-2</i>	<i>cond</i>
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1 IS [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</i></li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S			
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>NOT simple-condition</i></li> </ul>	V	S	S			
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1 &lt;AND   OR&gt; condition-2</i></li> </ul>	V	S	S			
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition &lt;AND   OR&gt; [NOT] [relational-operator] object</i></li> </ul>	V	S	S			

**Statements**

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs		Relation	Entities
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>] [WITH ENTRY POINTS <i>procedure-name-1</i>]</li> </ul>	V	S					



Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
DECLARATIVES	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.</li> </ul>	V	S					
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM <i>mnemonic-name</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE [YYYYMMDD]   DAY [YYYYDDD]   TIME   DATE-TIME [MMDDYYYY]&gt;</li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 3:</b> ACCEPT <i>id</i> [FROM &lt;CARD-READER   CONSOLE   DATE-TIME&gt;]</li> </ul>	VO	N					
ACCEPT MESSAGE COUNT	<ul style="list-style-type: none"> <li><b>Format</b> ACCEPT <i>cd-name</i> MESSAGE COUNT</li> </ul>	N	N					
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i> [ROUNDED] [: ON SIZE ERROR <i>imperative-stmf</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt;, &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [:ON SIZE ERROR <i>imperative-stmf</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [:ON SIZE ERROR <i>imperative-stmf</i>]</li> </ul>	V	S					
ALL	<ul style="list-style-type: none"> <li><b>Format</b> ALL &lt;INACTIVE   ACTIVE&gt; FOR &lt;<i>area-name-1</i>   <i>section-name-1</i>&gt;</li> </ul>	N	N					
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER <i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i></li> </ul>	VO	N					
ANY	<ul style="list-style-type: none"> <li><b>Format</b> ANY &lt;INACTIVE   ACTIVE&gt; FOR &lt;<i>area-name-1</i>   <i>section-name-1</i>&gt;</li> </ul>	N	N					

**1-108** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
CALL	<ul style="list-style-type: none"> <li><b>Format 1: general</b> CALL &lt;id-1   literal-1&gt; [USING &lt;data-name-1   literal-2&gt;] [; ON OVERFLOW <i>imperative-stmt</i>]</li> </ul>	V	S		id-2, id-3   port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
	<ul style="list-style-type: none"> <li><b>Format 2: CALL debug</b> CALL 'PAD\$\$'</li> </ul>	V	S		port   id-2, id-5	extern.call		
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S					
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE <i>file-name-1</i></li> </ul>	V	S					
	[<REEL   UNIT> [FOR REMOVAL   WITH NO REWIND]   WITH <NO REWIND   LOCK>]	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE <i>file-name-1</i></li> </ul>	V	S					
	[WITH LOCK]	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 3: IBM tape processing</b> CLOSE <i>file-name-1</i></li> </ul>	V	S					
[<REEL   UNIT> [WITH NO REWIND]   WITH <NO REWIND   LOCK>]	VO	N						
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE <i>id-1</i> [ROUNDED] = <i>arithmetic-expr</i> [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY <i>text-name</i> [&lt;OF   IN&gt; <i>library-name</i>] [REPLACING, &lt;==<i>pseudo-text-1</i>==   <i>id-1</i>   <i>literal-1</i>   <i>word-1</i>&gt; BY &lt;==<i>pseudo-text-2</i>==   <i>id-2</i>   <i>literal-2</i>   <i>word-2</i>&gt;]</li> </ul>	V	S				refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format: indexed and relative files</b> DELETE <i>file-name</i> RECORD [; INVALID KEY <i>imperative-stmt</i>]</li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	Program Deletes From File

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
DISABLE	<ul style="list-style-type: none"> <li><b>Format</b>                      DISABLE &lt;INPUT [TERMINAL]   OUTPUT&gt; <i>cd-name</i> WITH KEY                      &lt;<i>id-1</i>   <i>literal-1</i>&gt;</li> </ul>	N	N					
DISPLAY	<ul style="list-style-type: none"> <li><b>Format 1</b>                      DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt;                      [UPON &lt;CARD-PUNCH   CONSOLE   PRINTER&gt;]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b>                      DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [UPON <i>mnemonic-name</i>]</li> </ul>	V	S					
		VO	N					
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b>                      DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO <i>id-2</i> [ROUNDED]                      [: ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b>                      DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt;                      GIVING <i>id-3</i> [ROUNDED]                      [: ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b>                      DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt;                      GIVING <i>id-3</i> [ROUNDED]                      [: ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b>                      DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt;                      GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i>                      [: ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b>                      DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt;                      GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i>                      [: ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					

**1-110** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
ENABLE	<ul style="list-style-type: none"> <li><b>Format</b> ENABLE &lt;INPUT [TERMINAL]   OUTPUT&gt; <i>cd-name</i> WITH KEY &lt;<i>id-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S					
ENTER	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTER &lt;MASM   ASM   PL1   FTN&gt; <i>literal</i> [USING <i>argument</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> ENTER MASM "C\$INFO" USING <i>file-name</i>, <i>data-name</i></li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> ENTER &lt;FORTRAN   FD [ASM]&gt; <i>literal</i> [USING <i>argument</i>]</li> </ul>	V	S					
EXAMINE	<ul style="list-style-type: none"> <li><b>Format</b> EXAMINE <i>id-1</i> &lt;TALLYING &lt;UNTIL FIRST   ALL   LEADING   TRAILING&gt; &lt;<i>id-2</i>   <i>literal-1</i>&gt; [REPLACING BY &lt;<i>id-3</i>   <i>literal-2</i>&gt;]   REPLACING &lt;[UNTIL] FIRST   ALL   LEADING   TRAILING&gt; &lt;<i>id-2</i>   <i>literal-1</i>&gt; BY &lt;<i>id-3</i>   <i>literal-2</i>&gt;&gt;</li> </ul>	V	S					
EXHIBIT	<ul style="list-style-type: none"> <li><b>Format</b> EXHIBIT &lt;NAMED   CHANGED   CHANGED NAMED&gt; &lt;<i>data-name-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S					
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> <i>paragraph-name</i>. EXIT.</li> </ul>	V	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S					
FREE	<ul style="list-style-type: none"> <li><b>Format</b> FREE <i>file-name</i> RECORD</li> </ul>	V	S					
GENERATE	<ul style="list-style-type: none"> <li><b>Format</b> GENERATE &lt;<i>data-name</i>   <i>report-name</i>&gt;</li> </ul>	V	S					

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO TO <i>procedure-name-1</i></li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO TO <i>procedure-name-1</i> DEPENDING ON <i>id-1</i></li> </ul>	V	S					
HOLD	<ul style="list-style-type: none"> <li><b>Format</b> HOLD &lt;ALL   <i>are-name-1</i>   <i>section-name-1</i>&gt;</li> </ul>	N	N					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF <i>condition</i>; &lt;<i>stmt-1</i>   NEXT SENTENCE&gt;; ELSE &lt;<i>stmt-2</i>   NEXT SENTENCE&gt;</li> </ul>	V	S					
INITIATE	<ul style="list-style-type: none"> <li><b>Format</b> INITIATE <i>report-name-1</i></li> </ul>	V	S					
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING , <i>id-2</i> FOR, &lt;CHARACTERS   &lt;TRAILING   ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   , &lt;TRAILING   ALL   LEADING   FIRST&gt; , &lt;<i>id-5</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING , <i>id-2</i> FOR, &lt;CHARACTERS   &lt;TRAILING   ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   , &lt;TRAILING   ALL   LEADING   FIRST&gt; , &lt;<i>id-5</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S					

**1-112** Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
LOCK	<ul style="list-style-type: none"> <li><b>Format</b> LOCK <i>id-1</i></li> </ul>	N	N					
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> ON &lt;ASCENDING   DESCENDING&gt; KEY <i>data-name-1</i> [COLLATING SEQUENCE IS <i>alphabet-name-1</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li>USING <i>file-name-2 file-name-3</i> &lt;OUTPUT PROCEDURE IS <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>file-name-4</i>&gt;</li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format</b> MONITOR &lt;<i>procedure-name-1</i>   <i>id-1</i>   ALL&gt; [UNTIL <i>procedure-name</i>]</li> </ul>	V	S					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i></li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i></li> </ul>	V	S					
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
NOTE	<ul style="list-style-type: none"> <li><b>Format</b> NOTE <i>character-string</i>.</li> </ul>	VO	N					
ON	<ul style="list-style-type: none"> <li><b>Format</b> ON <i>integer-1</i> [AND EVERY <i>integer-2</i>] [UNTIL <i>integer-3</i>] ; &lt;<i>imperative-stmt-1</i>   NEXT SENTENCE&gt; [; &lt;OTHERWISE   ELSE &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt;]</li> </ul>	V	S					

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT <i>file-name-1</i> [REVERSED   WITH NO REWIND]   OUTPUT <i>file-name-2</i> [WITH NO REWIND]   I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S					
		VO	N					
		V	S					
		VO	N					
		V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   I-O <i>file-name-3</i>&gt;</li> </ul>	V	S					
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] &lt;<i>id-1</i>   <i>integer-1</i>&gt; TIMES</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] UNTIL <i>condition-1</i></li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with VARYING phrase</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt; FROM &lt;<i>id-3</i>   <i>index-name-2</i>   <i>literal-1</i>&gt; BY &lt;<i>id-4</i>   <i>literal-2</i>&gt; UNTIL <i>condition-1</i> [AFTER &lt;<i>id-5</i>   <i>index-name-3</i>&gt; FROM &lt;<i>id-6</i>   <i>index-name-4</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-4</i>&gt; UNTIL <i>condition-2</i>]</li> </ul>	V	S					
PROCESS	<ul style="list-style-type: none"> <li><b>Format</b> PROCESS <i>section-name</i> [FROM <i>id</i> [USING <i>area-name</i>]]</li> </ul>	N	N					

**1-114** Supported COBOL Statements  
*Supported Unisys ASCII COBOL statements*

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b>            READ <i>file-name-1</i>            [NEXT] RECORD</li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Reads File</i>
	[INTO <i>id-1</i> ] [: AT END <i>imperative-stmt</i> ]	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b>            READ <i>file-name-1</i> RECORD [INTO <i>id-1</i>]            [: INVALID KEY <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3: direct I-O WITH PROTECT</b>            READ <i>file-name-1</i> [NEXT] RECORD [INTO <i>id-1</i>]            [WITH [NO] PROTECT]            [: INVALID KEY <i>imperative-stmt</i>]</li> </ul>	V	S					
	[WITH [NO] PROTECT]	VO	N					
	[: INVALID KEY <i>imperative-stmt</i> ]	V	S					
RECEIVE	<ul style="list-style-type: none"> <li><b>Format</b>            RECEIVE <i>cd-name</i> &lt;MESSAGE   SEGMENT&gt; INTO <i>id-1</i>            [: NO DATA <i>imperative-stmt</i>]</li> </ul>	N	N					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b>            RELEASE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b>            RETURN <i>file-name-1</i> RECORD [INTO <i>id-1</i>]            ; AT END <i>imperative-stmt</i></li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Reads File</i>
REWRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b>            REWRITE <i>record-name</i> [FROM <i>id</i>]</li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Updates File</i>
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b>            REWRITE <i>record-name</i> [FROM <i>id</i>]            [: INVALID KEY <i>imperative-stmt</i>]</li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Updates File</i>



Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [; AT END <i>imperative-stmt-1</i>] ; WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [; AT END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> &lt;IS EQUAL TO   IS = &gt; &lt;<i>id-3</i>   <i>literal-1</i>   <i>arith-metic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> &lt;IS EQUAL TO   IS = &gt; &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt;</li> </ul>	V	S					
SEND	<ul style="list-style-type: none"> <li><b>Format 1</b> SEND <i>cd-name</i> FROM <i>id-1</i></li> </ul>	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b> SEND <i>cd-name</i> [FROM <i>id-1</i>] WITH &lt;<i>id-2</i>   ESI   EMI   EGI&gt; [&lt;BEFORE   AFTER&gt; ADVANCING &lt;&lt;<i>id-3</i>   <i>integer</i>&gt; [LINE   LINES]   <i>mnemonic-name</i>   PAGE&gt;]</li> </ul>	N	N					
SET	<ul style="list-style-type: none"> <li><b>Format 1: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 4: asynchronous processing</b> SET RANK FOR &lt;<i>area-name</i>   <i>section-name</i>&gt; &lt;UP BY   DOWN BY   TO&gt; &lt;<i>id-1</i>   <i>literal-1</i>&gt;</li> </ul>	N	N					

1-116 Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> ON &lt;ASCENDING   DESCENDING&gt; KEY <i>data-name-1</i> [COLLATING SEQUENCE IS <i>alphabet-name-1</i>] &lt;USING <i>file-name-2</i>   INPUT PROCEDURE IS <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]&gt; &lt;GIVING <i>file-name-3</i>   OUTPUT PROCEDURE IS <i>procedure-name-3</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-4</i>]&gt;</li> </ul>	V	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY IS &lt;EQUAL TO   =   GREATER THAN   &gt;   NOT LESS [THAN]   NOT &lt; &gt; <i>data-name-1</i>] [; INVALID KEY <i>imperative-stmt</i>]</li> </ul>	V	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED BY &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [WITH POINTER <i>id-4</i>] [; ON OVERFLOW <i>imperative-stmt</i>]</li> </ul>	V	S					
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b> SUBTRACT &lt;<i>id-1</i>   <i>literal-1</i>&gt; FROM <i>id-2</i> [ROUNDED] [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> SUBTRACT &lt;<i>id-1</i>   <i>literal-1</i>&gt; FROM &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> SUBTRACT &lt;CORRESPONDING   CORR&gt; <i>id-1</i> FROM <i>id-2</i> [ROUNDED] [; ON SIZE ERROR <i>imperative-stmt</i>]</li> </ul>	V	S					
SUPPRESS	<ul style="list-style-type: none"> <li><b>Format</b> SUPPRESS PRINTING</li> </ul>	V	S					

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
TERMINATE	<ul style="list-style-type: none"> <li><b>Format</b> TERMINATE <i>report-name-1</i></li> </ul>	V	S					
TRANSFORM	<ul style="list-style-type: none"> <li><b>Format</b> TRANSFORM <i>id-3</i> CHARACTERS FROM <i>&lt;figurative-constant-1   nonnumeric-literal-1   id-1&gt;</i> TO <i>&lt;figurative-constant-2   nonnumeric-literal-2   id-2&gt;</i></li> </ul>	VO	N					
UNLOCK	<ul style="list-style-type: none"> <li><b>Format</b> UNLOCK <i>id-1</i></li> </ul>	N	N					
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b> UNSTRING <i>id-1</i> [DELIMITED BY [ALL] <i>&lt;id-2   literal-1&gt;</i> [, OR [ALL] <i>&lt;id-3   literal-2&gt;</i>]] INTO <i>id-4</i> [, DELIMITER IN <i>id-5</i>] [, COUNT IN <i>id-6</i>] [WITH POINTER <i>id-7</i>] [TALLYING IN <i>id-8</i>] [, ON OVERFLOW <i>imperative-stmf</i>]</li> </ul>	V	S					

1-118 Supported COBOL Statements  
Supported Unisys ASCII COBOL statements

Conditions and statements	Format	Parser	IA	CA/GDF			Repository	
				Sup	Pairs	Relation	Entities	Relation
USE	<ul style="list-style-type: none"> <li><b>Format 1: file I-O</b> USE AFTER STANDARD &lt;EXCEPTION   ERROR&gt; PROCEDURE ON &lt;file-name   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: sequential file I-O</b> USE &lt;BEFORE   AFTER&gt; STANDARD [BEGINNING   ENDING] [REEL   FILE] LABEL PROCEDURE ON &lt;file-name   INPUT   OUTPUT   I-O&gt;</li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 3: report writer</b> USE BEFORE REPORTING <i>id</i></li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 4: debug mode</b> <i>section-name</i> SECTION [<i>segment-number</i>] USE FOR DEBUGGING ON &lt;<i>cd-name-1</i>   [ALL REFERENCES OF] <i>id-1</i>   <i>file-name-1</i>   <i>procedure-name-1</i>   ALL PROCEDURES&gt;</li> </ul>	VO	N					
	<ul style="list-style-type: none"> <li><b>Format 5: table error</b> USE FOR TABLE ERROR REFERENCES ON &lt;ALL   <i>id-1</i>&gt;</li> </ul>	N	N					
	<ul style="list-style-type: none"> <li><b>Format 6: asynchronous processing</b> USE FOR RANDOM PROCESSING [USING &lt;<i>file-name-1</i>   <i>cd-name-1</i>&gt;] [WITH RANK OF <i>integer-1</i>] [FOR <i>integer-2</i> CYCLES]</li> </ul>	VO	N					
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S				refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
	[<BEFORE   AFTER> ADVANCING << <i>id-1</i>   <i>integer-1</i> > [LINE   LINES]   <i>mnemonic-name</i>   PAGE>	VO	N					
	[; AT <END-OF-PAGE   EOP> <i>imperative-stmf</i> ]	V	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [; INVALID KEY <i>imperative-stmf</i>]</li> </ul>	V	S					

## **Supported Unisys UCS COBOL statements**

### **COBOL language structure**

#### **Figurative constants**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### **Special registers**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
DEBUG ITEM	V	S	S
LINAGE-COUNTER	V	S	S
LINE-COUNTER	V	S	S
PAGE-COUNTER	V	S	S

**1-120** Supported COBOL Statements  
*Supported Unisys UCS COBOL statements*

**Literals**

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literal	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; <i>mantissa</i> E &lt;+   -&gt; <i>exponent</i></li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

**Referencing names**

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li>• <b>Format 1: subscripting</b>  <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 2</b>  <i>&lt;condition-name-1   data-name-1&gt;</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 3</b>                      LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li>• <b>Format 4</b>  <i>&lt;PAGE-COUNTER   LINE-COUNTER&gt;</i> &lt;IN   OF&gt; <i>report-name</i></li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li>• <b>Format: Data Division</b>  <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>)</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li>• <b>Format</b>  <i>&lt;integer-1   ALL   data-name-3 [ &lt;+   -&gt; integer-2 ]   index-name-1 [ &lt;+   -&gt; integer-3 ]&gt;</i></li> </ul>	V	S	S treated as reference to the whole array
Reference modification	<ul style="list-style-type: none"> <li>• <b>Format</b>  <i>&lt;data-name-1   FUNCTION function-name-1 (arguments)&gt;</i> (<i>leftmost-character-position: [length]</i>)</li> </ul>	S	S	S treated as reference to whole <i>data-name-1</i>

**1-122** Supported COBOL Statements  
*Supported Unisys UCS COBOL statements*

**COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> [[IS] <COMMON   INITIAL> [PROGRAM]]. <i>id-division-content</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i> .	V	S
Nested program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> [[IS] <COMMON   INITIAL> [PROGRAM]]. <i>id-division-content</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> <i>nested source program</i> END-PROGRAM <i>program-name-1</i> .	N	N

**Control Division**

Names	Format	Parser	IA
Control Division Entry	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [CONTROL DIVISION]. [ALPHABET SECTION]. [SOURCE-ALPHABET CHARACTERS ARE <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i> ]]	N	N
Inline Section	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> APPLY INLINE <ADD   ALL   CONVERSION   IF   INSPECT   MOVE   SIGN> FOR <ALL PROCEDURES   <i>procedure-name-1</i> > [<THROUGH   THRU> <i>procedure-name-2</i> ]	N	N

**Identification Division**

Names	Format	Parser	IA	Repository	
				Entities	Relation



Identification Division	<ul style="list-style-type: none"> <li><b>Format</b> IDENTIFICATION DIVISION. PROGRAM-ID <i>program-name</i>. [AUTHOR. [<i>comment-entry</i>]] [INSTALLATION. [<i>comment-entry</i>]] [DATE-WRITTEN. [<i>comment-entry</i>]] [DATE-COMPILED. [<i>comment-entry</i>]] [SECURITY. [<i>comment-entry</i>]]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Program Entry Point
-------------------------	--	---	---	---	---

### Environment Division

#### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN TO &lt;<i>assignment-name-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION IS] SEQUENTIAL] [PADDING CHARACTER IS < <i>data-name-1</i>   <i>literal-2</i> >] [RECORD DELIMITER IS STANDARD-1] [ACCESS MODE IS SEQUENTIAL] [FILE STATUS IS <i>data-name-2</i> ] [CODE TYPE IS <i>data-name-3</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed files</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN TO &lt;<i>assignment-name-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION IS] INDEXED [ACCESS MODE IS <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD KEY IS <i>data-name-1</i> [ALTERNATE RECORD KEY IS <i>data-name-2</i> [WITH DUPLICATES]] [FILE STATUS IS <i>data-name-3</i> ] [LOCALE IS <i>data-name-4</i> ].	VO	N		

**1-124** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

	<ul style="list-style-type: none"> <li> <b>Format 3: relative files</b>            FILE-CONTROL.            SELECT <i>file-name-1</i> ASSIGN TO &lt;<i>assignment-name-1</i>   <i>literal-1</i>&gt;         </li> </ul>	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION IS] RELATIVE [ACCESS MODE IS <SEQUENTIAL [RELATIVE KEY IS <i>data-name-1</i> ]   <RANDOM   DYNAMIC> RELATIVE KEY IS <i>data-name-1</i> >] [FILE STATUS IS <i>data-name-2</i> ] [CODE TYPE IS <i>data-name-3</i> ].	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 4: sort-merge</b>            FILE-CONTROL.            SELECT <i>file-name-1</i> ASSIGN TO <i>assignment-name-1</i>.         </li> </ul>	V	S		
	<ul style="list-style-type: none"> <li> <b>Format 5: report writer</b>            FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN TO &lt;<i>assignment-name-1</i>   <i>literal-1</i>&gt;         </li> </ul>	V	S		
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION IS] SEQUENTIAL] [PADDING CHARACTER IS < <i>data-name-1</i>   <i>literal-2</i> >] [RECORD DELIMITER IS STANDARD-1] [ACCESS MODE IS SEQUENTIAL] [FILE STATUS IS <i>data-name-2</i> ].	VO	N		
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li> <b>Format 1: sequential files</b>            I-O CONTROL.            [RERUN [ON &lt;<i>assignment-name-1</i>   <i>file-name-1</i>&gt;] &lt;[END OF] &lt;REEL   UNIT&gt;   <i>integer-1</i> RECORDS   <i>integer-2</i> CLOCK-UNITS   <i>condition-name-1</i>&gt; OF <i>file-name-2</i>]            [SAME [RECORD] AREA FOR <i>file-name-3</i>]            [MULTIPLE FILE TAPE CONTAINS <i>file-name-5</i> [POSITION <i>integer-3</i>]].         </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 2: indexed and relative files</b>            I-O CONTROL.            [RERUN ON &lt;<i>assignment-name-1</i>   <i>file-name-1</i>&gt; EVERY &lt;<i>integer-1</i> RECORDS OF <i>file-name-1</i>   <i>integer-2</i> CLOCK-UNITS   <i>condition-name</i>&gt;]            [SAME [RECORD] AREA FOR <i>file-name-3</i>].         </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 3: sort-merge</b>            I-O CONTROL.            [SAME &lt;RECORD   SORT   SORT-MERGE&gt; AREA FOR <i>file-name-1</i>].         </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 4: report writer</b>            I-O CONTROL.            [SAME AREA FOR <i>file-name-1</i>]            [MULTIPLE FILE TAPE CONTAINS <i>file-name-3</i> [POSITION <i>integer-1</i>]].         </li> </ul>	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA
OBJECT-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> OBJECT-COMPUTER. [<i>computer-name</i> [MEMORY SIZE <i>integer</i> &lt;WORDS   CHARACTERS   MODULES&gt;] [PROGRAM COLLATING SEQUENCE IS <i>alphabet-name</i>] [SEGMENT-LIMIT IS <i>segment-number</i>].]</li> </ul>	VO	N
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE-COMPUTER. [<i>computer-name</i> [WITH DEBUGGING MODE].]</li> </ul>	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [<i>implementor-name</i> &lt;IS <i>mnemonic-name</i> [ON STATUS IS <i>condition-name-1</i> [OFF STATUS IS <i>condition-name-2</i>]]   IS <i>mnemonic-name</i> [OFF STATUS IS <i>condition-name-2</i> [ON STATUS IS <i>condition-name-1</i>]]   ON STATUS IS <i>condition-name-1</i> [OFF STATUS IS <i>condition-name-2</i>]   OFF STATUS IS <i>condition-name-2</i> [ON STATUS IS <i>condition-name-1</i>]-&gt;]</li> </ul>	V	S
	[ALPHABET <i>alphabet-name</i> IS <STANDARD-1   STANDARD-2   NATIVE   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> ]->]	VO	N
	[SYMBOLIC CHARACTERS <i>symbolic-character-1</i> <IS   ARE> <i>integer-1</i> [[IN <i>alphabet-name-2</i> ]]	VO	N
	[CLASS <i>class-name</i> IS <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]]	VO	N
	[CURRENCY SIGN IS <i>literal-6</i> ]	VO	N
	[DECIMAL POINT IS COMMA.]	V	S
APPLY clause	<ul style="list-style-type: none"> <li><b>Format</b> APPLY &lt;EXREF   EXDEF&gt; ON <i>id-1</i></li> </ul>	VO	N
MULTIPLE FILE TAPE clause	<ul style="list-style-type: none"> <li><b>Format</b> MULTIPLE FILE TAPE CONTAINS <i>file-name-1</i> [POSITION <i>integer-1</i>]</li> </ul>	VO	N
RERUN clause	<ul style="list-style-type: none"> <li><b>Format</b> RERUN [ON &lt;<i>assignment-name-1</i>   <i>file-name-1</i>&gt;] &lt;[END OF] &lt;REEL   UNIT&gt;   <i>integer-1</i> RECORDS   <i>integer-2</i> CLOCK-UNITS   <i>condition-name-1</i>&gt; OF <i>file-name-2</i></li> </ul>	VO	N
SAME clause	<ul style="list-style-type: none"> <li><b>Format</b> SAME [RECORD] AREA FOR <i>file-name-1</i></li> </ul>	VO	N

**1-126** Supported COBOL Statements  
*Supported Unisys UCS COBOL statements*

Paragraphs and entries	Format	Parser	IA
SEGMENT-LIMIT clause	<ul style="list-style-type: none"> <li><b>Format</b>            SEGMENT-LIMIT IS <i>segment-number</i></li> </ul>	VO	N

**Data Division**

Entries and Clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format</b>            DATA DIVISION.            [FILE SECTION. [<i>file-description-entry record-description-entry</i>]]            [WORKING-STORAGE SECTION. [<i>record-description-entry   data-item-description-entry</i>]]            [LINKAGE SECTION. [<i>record-description-entry   data-item-description-entry</i>]]</li> </ul>	V	S	S		
LINKAGE section	<ul style="list-style-type: none"> <li><b>Format</b>            LINKAGE SECTION.            [<i>77-level-description-entry   record-description-entry</i>]</li> </ul>	V	S	S		
WORKING-STORAGE section	<ul style="list-style-type: none"> <li><b>Format</b>            WORKING-STORAGE SECTION.            [<i>77-level-description-entry   record-description-entry</i>]</li> </ul>	V	S	S		

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Data Description Entry	<ul style="list-style-type: none"> <li> <b>Format 1</b>  <i>level-number</i> [<i>data-name-1</i>   FILLER]                      [REDEFINES <i>data-name-2</i>] [IS EXTERNAL] [IS GLOBAL]                      [&lt;PICTURE   PIC&gt; IS <i>character-string</i>]                      [[USAGE IS] &lt;BINARY   BINARY-1   BYTE-118N   COMPUTATIONAL   COMP   DISPLAY   COMPUTATIONAL-1   COMP-1   COMPUTATIONAL-2   COMP-2   DISP   DISP-118N   INDEX   PACKED-DECIMAL&gt;]                      [[SIGN IS] &lt;LEADING   TRAILING&gt; [SEPARATE CHARACTER]]                      [OCCURS &lt;<i>integer-2</i> TIMES   <i>integer-1</i> TO <i>integer-2</i> TIMES DEPENDING ON <i>data-name-4</i>&gt; [&lt;ASCENDING   DESCENDING&gt; KEY IS <i>data-name-3</i>] [INDEXED BY <i>index-name-1</i>]]                      [&lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]]                      [&lt;JUSTIFIED   JUST&gt; RIGHT]                      [BLANK WHEN ZERO]                      [VALUE IS <i>literal</i>].                 </li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li> <b>Format 2</b>                      66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>].                 </li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li> <b>Format 3</b>                      88 <i>condition-name</i> &lt;VALUE IS   VALUES ARE&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].                 </li> </ul>	V	S	S			

**1-128** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Entries and Clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> <i>FD file-name</i></li> </ul>	V	S	S			
	[BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   IS VARYING IN SIZE [[FROM <i>integer-4</i> ] [TO <i>integer-5</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ]   CONTAINS [ <i>integer-6</i> TO] <i>integer-7</i> CHARACTERS>] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF <i>implementor-name-1</i> IS < <i>data-name-2</i>   <i>literal-1</i> >] [DATA <RECORD IS   RECORDS ARE> <i>data-name-3</i> ] [LINAGE IS < <i>data-name-4</i>   <i>integer-8</i> > LINES [WITH FOOTING AT < <i>data-name-5</i>   <i>integer-9</i> >] [LINES AT TOP < <i>data-name-6</i>   <i>integer-10</i> >] [LINES AT BOTTOM < <i>data-name-7</i>   <i>integer-11</i> >]] [LINAGE IS SYSTEM LINES] [CODE-SET IS <i>alphabet-name</i> ] DENSITY IS < <i>data-name-8</i>   <i>integer-12</i> > LINES	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> <i>FD file-name</i></li> </ul>	V	S	S			
	[BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   IS VARYING IN SIZE [[FROM <i>integer-4</i> ] [TO <i>integer-5</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ]   CONTAINS [ <i>integer-6</i> TO] <i>integer-7</i> CHARACTERS>] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF <i>implementor-name-1</i> IS < <i>data-name-2</i>   <i>literal-1</i> >] [DATA <RECORD [IS]   RECORDS [ARE]> <i>data-name-3</i> ]	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3: sort-merge</b> <i>SD file-name</i></li> </ul>	V	S	S			
	[RECORD <CONTAINS <i>integer-1</i> CHARACTERS   IS VARYING IN SIZE [[FROM <i>integer-2</i> ] [TO <i>integer-3</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ]   CONTAINS [ <i>integer-4</i> TO] <i>integer-5</i> CHARACTERS>] [DATA <RECORD [IS]   RECORDS [ARE]> <i>data-name-2</i> ].	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 4: report writer</b> <i>FD file-name</i></li> </ul>	V	S	S			
	[BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   CONTAINS [ <i>integer-4</i> TO] <i>integer-5</i> CHARACTERS>] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF <i>system-name-1</i> IS < <i>data-name-2</i>   <i>literal-2</i> >] [CODE-SET IS <i>alphabet-name</i> ] <REPORT IS   REPORTS ARE> <i>report-name</i> DENSITY IS < <i>data-name-2</i>   <i>integer-6</i> > LINES.	VO	N	N			

Entries and Clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
Report Description (RD) Entry	<ul style="list-style-type: none"> <li><b>Format</b> RD <i>report-name</i> [IS GLOBAL] [CODE <i>literal-1</i>] [&lt;CONTROL IS   CONTROLS ARE&gt; &lt;<i>data-name-1</i>   FINAL [<i>data-name-1</i>&gt;] PAGE [LIMIT IS   LIMITS ARE] <i>integer-1</i> [LINE   LINES] [HEADING <i>integer-2</i>] [FIRST DETAIL <i>integer-3</i>] [LAST DETAIL <i>integer-4</i>] [FOOTING <i>integer-5</i>].</li> </ul>	V	S	N			
Report Group Description	<ul style="list-style-type: none"> <li><b>Format 1</b> 01 [<i>data-name-1</i>] [LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [NEXT GROUP IS &lt;<i>integer-3</i>   PLUS <i>integer-4</i>   NEXT PAGE&gt;] TYPE IS &lt;REPORT HEADING   RH   PAGE HEADING   PH   &lt;CONTROL HEADING   CH&gt; &lt;<i>data-name-2</i>   FINAL&gt;   DETAIL   DE   &lt;CONTROL FOOTING   CF&gt; &lt;<i>data-name-3</i>   FINAL&gt;   PAGE FOOTING   PF   REPORT FOOTING   RF&gt; [[USAGE IS] &lt;DISPLAY   DISP&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>level-number</i> [<i>data-name-1</i>] [LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [[USAGE IS] &lt;DISPLAY   DISP&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>level-number</i> [<i>data-name-1</i>] &lt;PICTURE   PIC&gt; IS <i>character-string</i> [[USAGE IS] &lt;DISPLAY   DISP&gt;] [[SIGN IS] &lt;LEADING   TRAILING&gt; SEPARATE CHARACTER] [&lt;JUSTIFIED   JUST&gt; RIGHT] [BLANK WHEN ZERO] [LINE NUMBER IS &lt;<i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [COLUMN NUMBER IS <i>integer-3</i>] SOURCE IS <i>id-1</i> VALUE IS <i>literal</i> SUM <i>id-2</i> [UPON <i>data-name-2</i>] [RESET ON &lt;<i>data-name-3</i>   FINAL&gt;] [GROUP INDICATE].</li> </ul>	V	S	N			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> BLANK WHEN ZERO</li> </ul>	V	S	N			
BLOCK CONTAINS clause	<ul style="list-style-type: none"> <li><b>Format</b> BLOCK CONTAINS [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;</li> </ul>	N	N	N			
CODE clause	<ul style="list-style-type: none"> <li><b>Format</b> CODE <i>literal-1</i></li> </ul>	V	S	N			

**1-130** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
CODE-SET clause	<ul style="list-style-type: none"> <li><b>Format</b> CODE-SET IS <i>alphabet-name</i></li> </ul>	V	S	N			
COLUMN NUMBER clause	<ul style="list-style-type: none"> <li><b>Format</b> COLUMN NUMBER IS <i>integer-1</i></li> </ul>	V	S	N			
CONTROL clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;CONTROL IS   CONTROLS ARE&gt; <i>&lt;data-name-1   FINAL&gt;</i></li> </ul>	V	S	N			
Data-name or FILLER clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;data-name   FILLER&gt;</i></li> </ul>	V	S	S			
DATA RE-CORDS clause	<ul style="list-style-type: none"> <li><b>Format</b> DATA &lt;RECORD IS   RECORDS ARE&gt; <i>data-name</i></li> </ul>	V	S	S			
DENSITY clause	<ul style="list-style-type: none"> <li><b>Format</b> DENSITY IS <i>&lt;data-name-8   integer-12&gt;</i> LINES</li> </ul>	VO	N	N			
GROUP INDICATE clause	<ul style="list-style-type: none"> <li><b>Format</b> GROUP INDICATE</li> </ul>	V	S	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;JUSTIFIED   JUST&gt; RIGHT</li> </ul>	V	S	N			
LABEL RE-CORDS clause	<ul style="list-style-type: none"> <li><b>Format</b> LABEL &lt;RECORD IS   RECORDS ARE&gt; &lt;STANDARD   OMITTED&gt;</li> </ul>	V	S	N			
LINAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> LINAGE IS <i>&lt;data-name-1   integer-1&gt;</i> LINES [WITH FOOTING AT <i>&lt;data-name-2   integer-2&gt;</i>] [LINES AT TOP <i>&lt;data-name-3   integer-3&gt;</i>] [LINES AT BOTTOM <i>&lt;data-name-4   integer-4&gt;</i>]</li> </ul>	N	N	N			
LINE NUMBER clause	<ul style="list-style-type: none"> <li><b>Format</b> LINE NUMBER IS <i>&lt;integer-1 [ON NEXT PAGE]   PLUS integer-2&gt;</i></li> </ul>	V	S	N			
NEXT GROUP clause	<ul style="list-style-type: none"> <li><b>Format</b> NEXT GROUP IS <i>&lt;integer-1   PLUS integer-2   NEXT PAGE&gt;</i></li> </ul>	V	S	N			



Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> OCCURS <i>integer-2</i> TIMES [&lt;ASCENDING   DESCENDING&gt; KEY IS <i>data-name-2</i>] [INDEXED BY <i>index-name-1</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> OCCURS <i>integer-1</i> TO <i>integer-2</i> TIMES DEPENDING ON <i>data-name-1</i> [&lt;ASCENDING   DESCENDING&gt; KEY IS <i>data-name-2</i>] [INDEXED BY <i>index-name-1</i>]</li> </ul>	V	S	S			
PAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> PAGE [LIMIT IS   LIMITS ARE] <i>integer-1</i> [LINE   LINES] [HEADING <i>integer-2</i>] [FIRST DETAIL <i>integer-3</i>] [LAST DETAIL <i>integer-4</i>] [FOOTING <i>integer-5</i>]</li> </ul>	V	S	N			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; IS <i>character-string</i></li> </ul>	V	S	S			
RECORD CONTAINS clause	<ul style="list-style-type: none"> <li><b>Format</b> RECORD CONTAINS <i>integer-1</i> TO <i>integer-2</i> CHARACTERS</li> </ul>	VO	N	N			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number</i> [<i>data-name-1</i>   FILLER] [REDEFINES <i>data-name-2</i>]</li> </ul>	V	S	S			
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>].</li> </ul>	V	S	S			
REPORT clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;REPORT IS   REPORTS ARE&gt; <i>report-name</i></li> </ul>	VO	N	N			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> [SIGN IS] &lt;LEADING   TRAILING&gt; SEPARATE CHARACTER</li> </ul>	V	S	S			
SOURCE clause	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE IS <i>id-1</i></li> </ul>	V	S	N			
SUM clause	<ul style="list-style-type: none"> <li><b>Format</b> SUM <i>id-1</i> [UPON <i>data-name-2</i>] [RESET ON &lt;<i>data-name-3</i>   FINAL&gt;]</li> </ul>	V	S	N			

**1-132** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZE   SYNC&gt; [LEFT   RIGHT]</li> </ul>	V	S	N			
TYPE clause	<ul style="list-style-type: none"> <li><b>Format</b> TYPE IS &lt;REPORT HEADING   RH   PAGE HEADING   PH   &lt;CONTROL HEADING   CH&gt; &lt;data-name-2   FINAL&gt;   DETAIL   DE   &lt;CONTROL FOOTING   CF&gt; &lt;data-name-3   FINAL&gt;   PAGE FOOTING   PF   REPORT FOOTING   RF&gt;</li> </ul>	V	S	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: general</b> [USAGE IS] &lt;BINARY   BINARY-1   BYTE-118N   COMPUTATIONAL   COMP   DISPLAY   COMPUTATIONAL-1   COMP-1   COMPUTATIONAL-2   COMP-2   DISP   DISP-118N   INDEX   PACKED-DECIMAL&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: report writer</b> [USAGE IS] DISPLAY</li> </ul>	V	S	S			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1</b> VALUE IS <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	cost. unit
	<ul style="list-style-type: none"> <li><b>Format 2</b> &lt;VALUE IS   VALUES ARE&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>]</li> </ul>	V	S	S			
VALUE OF clause	<ul style="list-style-type: none"> <li><b>Format</b> VALUE OF <i>system-name-1</i> IS &lt;data-name-1   <i>literal-1</i>&gt;</li> </ul>	VO	N	N			

## Procedure Division

### Arithmetic operators

Operation	Meaning	Format	Parser	IA	CA/GDF			
					Sup.	Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

### Conditional Expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs		Relation
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id IS [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   class-name&gt;</i></li> </ul>	V	S	S			

**1-134** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> IS &lt;[NOT] &lt;GREATER THAN   LESS THAN   EQUAL TO&gt;   GREATER THAN OR EQUAL TO   LESS THAN OR EQUAL TO   &gt;   &lt;   =   &gt;=   &lt;= &gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	cond
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>arithmetic-expr</i> IS [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i>.</li> </ul>	V	S	S			
Negated simple conditions	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>simple-condition</i></li> </ul>	V	S	S			
Combined conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			
Abbreviated combined relation conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S			

**Statements**

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pair	Relation	Entities	Relation
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>] [WITH ENTRY POINTS <i>procedure-name-1</i>]</li> </ul>	V	S	S	<i>data-name-1</i>	<i>port</i>	extern. using	
					<i>port</i>	<i>data-name-1</i>	extern. using	
DECLARATIVES	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION. USE <i>stmt</i>. [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.</li> </ul>	V	S	S				
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM <i>mnemonic-name</i>]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen	
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE [YYYYMMDD]   DAY [YYYYDDD]   TIME   DAY-OF-WEEK&gt;</li> </ul>	VO	N	N				
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i> [ROUNDED] [ON SIZE ERROR <i>imperative-stmt-1</i>] [NOT ON SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [ON SIZE ERROR <i>imperative-stmt-1</i>] [NOT ON SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+	
					<i>id-2</i>	<i>id-3</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [ON SIZE ERROR <i>imperative-stmt-1</i>] [NOT ON SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER <i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i></li> </ul>	VO	N	N				

1-136 Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relation
CALL	<ul style="list-style-type: none"> <li><b>Format 1</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[BY REFERENCE] &lt;procedure-name-1   id-2&gt;   BY CONTENT &lt;literal-2   id-2&gt;   [BY CONTENT] literal-2&gt;] [ON OVERFLOW imperative-stmt-1] [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
				port	id-2, id-5	extern.call			
	<ul style="list-style-type: none"> <li><b>Format 2</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[BY REFERENCE] &lt;procedure-name-1   id-2&gt;   BY CONTENT &lt;literal-2   id-2&gt;   [BY CONTENT] literal-2&gt;] [ON EXCEPTION imperative-stmt-1] [NOT ON EXCEPTION imperative-stmt-1] [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
			port	id-2, id-5	extern.call				
<ul style="list-style-type: none"> <li><b>Format 3: suppression of SORT console and log messages</b> CALL &lt;'SUPCONS\$'   'SUPLOGS\$'   'SUPCNLOGS\$'   'ENACONS\$'   'ENALOGS\$'   'ENACONLG\$'&gt;</li> </ul>	V	S	S						
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S					
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential and report writer</b> CLOSE file-name-1</li> </ul>	V	S	S					
	<[<REEL   UNIT> [FOR REMOVAL]   WITH <NO REWIND   LOCK>]	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE file-name-1</li> </ul>	V	S	S					
	[WITH LOCK]	VO	N	N					
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE id-1 [ROUNDED] = arithmetic-expr [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-COMPUTE]</li> </ul>	V	S	S	identifiers of the arithmetic-expr	id-1	comp+ or comp* (according to the operation performed)		
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relation	
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY text-name [&lt;OF   IN&gt; library-name] [REPLACING &lt;==pseudo-text-1==   id-1   literal-1   word-1&gt; BY &lt;==pseudo-text-2==   id-2   literal-2   word-2&gt;]</li> </ul>	V	S	N				refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format: indexed and relative files</b> DELETE file-name RECORD [INVALID KEY imperative-stmt-1] [NOT INVALID KEY imperative-stmt-2] [END-DELETE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Deletes From File
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;id   literal&gt;</li> </ul>	V	S	S	id-1	port	extern. screen		
	[UPON mnemonic-name]	VO	N	N					
	[WITH NO ADVANCING]	V	S	S					

1-138 Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pair	Relation	Entities	Relation
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;id-1   literal-1&gt; INTO id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2   id-1	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2   id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1   id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1   id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1   id-3	comp*		
					id-2   id-3	comp*		
					id-1   id-4	comp*		
					id-2   id-4	comp*		
					id-1   id-4	comp*		
					id-2   id-4	comp*		
ENTER	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTER MASM literal [USING argument]</li> </ul>	V	S	N				
	<ul style="list-style-type: none"> <li><b>Format 2</b> ENTER MASM "C\$INFO" USING file-name, data-name</li> </ul>	V	S	N				



Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relation
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;] WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1 [WHEN OTHER imperative-stmt-2] [END-EVALUATE] Phrases phrase-1 and phrase-2 should be represented in the following form: &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-1   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-2   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S	id-1, expr-1	condition-1	cond		
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> EXIT.</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GENERATE	<ul style="list-style-type: none"> <li><b>Format</b> GENERATE &lt;data-name-1   report-name-1&gt;</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO TO procedure-name-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO TO procedure-name-1 DEPENDING ON id-1</li> </ul>	V	S	S					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF condition THEN &lt;stmt-1   NEXT SENTENCE&gt; [ELSE &lt;stmt-2   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE id-1 [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED&gt; DATA BY &lt;id-2   literal-1&gt;]</li> </ul>	V	S	S					
INITIATE	<ul style="list-style-type: none"> <li><b>Format</b> INITIATE report-name-1</li> </ul>	V	S	S					

1-140 Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair	Relation	Entities	Relation	
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt;&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt; ALL   LEADING   FIRST&gt; , &lt;<i>id-5</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt;&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;] REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-5</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [BEFORE   AFTER] [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp		
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> ON &lt;ASCENDING   DESCENDING&gt; KEY <i>data-name-1</i></li> </ul>	V	S						
	[COLLATING SEQUENCE IS <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>USING <i>file-name-2</i> <i>file-name-3</i> &lt;OUTPUT PROCEDURE IS <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>file-name-4</i>&gt;</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair		Relation	Entities	Relation
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;id-1   literal-1&gt; TO id-2</li> </ul>	V	S	S	id-1	id-2	move, cast (depending on data type)		
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; id-1 TO id-2</li> </ul>	V	S	S	id-1	id-2	move, cast (depending on data type)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;id-1   literal-1&gt; BY id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-MULTIPLY]</li> </ul>	V	S	S	id-1	id-2	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-MULTIPLY]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT file-name-1 [REVERSED   WITH NO REWIND]   OUTPUT file-name-2 [WITH NO REWIND]   I-O file-name-3   EXTEND file-name-4&gt;</li> </ul>	V	S	S					
		VO	N	N					
		V	S	S					
		VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> OPEN &lt;INPUT file-name-1   OUTPUT file-name-2   I-O file-name-3   EXTEND file-name-4&gt;</li> </ul>	V	S	S					
		V	S	S					
		VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: report writer</b> OPEN &lt;OUTPUT file-name-1 [WITH NO REWIND]   EXTEND file-name-2&gt;</li> </ul>	V	S	S					
		VO	N	N					
		V	S						

1-142 Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relation	
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [<i>imperative-stmt</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] &lt;<i>id-1</i>   <i>integer-1</i>&gt; TIMES [<i>imperative-stmt</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [WITH TEST &lt;BEFORE   AFTER&gt;] UNTIL <i>condition-1</i> [<i>imperative-stmt</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with VARYING phrase</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [WITH TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt; FROM &lt;<i>id-3</i>   <i>index-name-2</i>   <i>literal-1</i>&gt; BY &lt;<i>id-4</i>   <i>literal-2</i>&gt; UNTIL <i>condition-1</i> [AFTER &lt;<i>id-5</i>   <i>index-name-3</i>&gt; FROM &lt;<i>id-6</i>   <i>index-name-4</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-4</i>&gt; UNTIL <i>condition-2</i>] [<i>imperative-stmt</i> END-PERFORM]</li> </ul>	V	S	S					
PROCESS	<ul style="list-style-type: none"> <li><b>Format</b> PROCESS <i>section-name</i> [FROM <i>id</i> [USING <i>area-name</i>]]</li> </ul>	N	N	N					
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b> READ <i>file-name-1</i> [NEXT] RECORD [INTO <i>id-1</i>] [AT END <i>imperative-stmt-1</i>] [NOT AT END <i>imperative-stmt-2</i>] [END-READ]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Reads File</i>
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b> READ <i>file-name-1</i> RECORD [INTO <i>id-1</i>] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-READ]</li> </ul>	V	S	S					
RE-LEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE <i>record-name-1</i> [FROM <i>id-1</i>]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relation	
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b> REPLACE ==pseudo-text-1== BY ==pseudo-text-2==</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPLACE OFF</li> </ul>	V	S	S					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b> RETURN file-name-1 RECORD [INTO id-1] AT END imperative-stmt-1 [NOT AT END imperative-stmt-2] [END-RETURN]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> REWRITE record-name [FROM id] [END-REWRITE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Updates File
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> REWRITE record-name [FROM id] [INVALID KEY imperative-stmt-1] [NOT INVALID KEY imperative-stmt-2] [END-REWRITE]</li> </ul>	V	S	S					
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH id-1 [VARYING &lt;id-2   index-name-1&gt;] [AT END imperative-stmt-1] WHEN condition-1 &lt;imperative-stmt-2   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL id-1 [AT END imperative-stmt-1] WHEN &lt;data-name-1 &lt;IS EQUAL TO   IS = &gt; &lt;id-3   literal-1   arithmetic-expr-1&gt;   condition-name-1&gt; [AND &lt;data-name-2 &lt;IS EQUAL TO   IS = &gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;   condition-name-2&gt;] &lt;imperative-stmt-2   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					

**1-144** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relation	
SET	<ul style="list-style-type: none"> <li><b>Format 1: external switches</b> SET <i>mnemonic-name</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> <i>integer-1</i>	move		
	<ul style="list-style-type: none"> <li><b>Format 3: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> SET <i>condition-name</i> TO TRUE</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name</i> )	<i>condition value</i> (associated with <i>condition-name</i> )	move		
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> ON &lt;ASCENDING   DESCENDING&gt; KEY <i>data-name-1</i> [COLLATING SEQUENCE IS <i>alphabet-name-1</i>] &lt;USING <i>file-name-2</i>   INPUT PROCEDURE IS <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>&gt; &lt;GIVING <i>file-name-3</i>   OUTPUT PROCEDURE IS <i>procedure-name-3</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-4</i>&gt;</li> </ul>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY IS &lt;EQUAL TO   =   GREATER THAN   &gt;   NOT LESS THAN   NOT &lt;   GREATER THAN OR EQUAL TO   &gt;= &gt; <i>data-name-1</i>] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED BY &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [WITH POINTER <i>id-4</i>] [ON OVERFLOW <i>imperative-stmt-1</i>] [NOT ON OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pair		Relation	Entities	Relation
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
SUPPRESS	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> SUPPRESS PRINTING	V	S	S					
TERMINATE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> TERMINATE report-name-1	V	S	S					
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED BY [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER IN id-5] [COUNT IN id-6] [WITH POINTER id-7] [TALLYING IN id-8] [ON OVERFLOW imperative-stmt-1] [NOT ON OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

**1-146** Supported COBOL Statements  
Supported Unisys UCS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pair	Relation	Entities	Relation	
USE	<ul style="list-style-type: none"> <li><b>Format 1: file I-O</b> USE [GLOBAL] AFTER STANDARD &lt;EXCEPTION   ERROR&gt; PROCEDURE ON &lt;file-name   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: sequential file I-O</b> USE &lt;BEFORE   AFTER&gt; STANDARD [BEGINNING   ENDING] [REEL   FILE] LABEL PROCEDURE ON &lt;file-name   INPUT   OUT- PUT   I-O&gt;</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: report writer</b> USE [GLOBAL] BEFORE REPORTING <i>id</i></li> </ul>	VO	N	N					
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [&lt;BEFORE   AFTER&gt; ADVANCING &lt;&lt;<i>id-2</i>   <i>integer-1</i>&gt; [LINE   LINES]   <i>mnemonic-name</i>   PAGE&gt;] [AT &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-1</i>] [NOT AT &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	<i>Program Inserts Into File</i>
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S					

**Intrinsic functions**

Functions	Parser	IA	Sup.	CA/GDF		
				Pairs		Relation
ACOS	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
ANNUITY	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
ASIN	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
ATAN	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>



Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
CHAR	V	S	S	operands	result	comp	
COS	V	S	S	operands	result	comp	
CURRENT-DATE	V	S	S	operands	result	comp	
DATE-OF-INTEGER	V	S	S	operands	result	comp	
DAY-OF-INTEGER	V	S	S	operands	result	comp	
FACTORIAL	V	S	S	operands	result	comp	
INTEGER	V	S	S	operands	result	comp	
INTEGER-OF-DATE	V	S	S	operands	result	comp	
INTEGER-OF-DAY	V	S	S	operands	result	comp	
INTEGER-PART	V	S	S	operands	result	comp	
LENGTH	V	S	S	operands	result	comp	
LOG	V	S	S	operands	result	comp	
LOG10	V	S	S	operands	result	comp	
LOWER-CASE	V	S	S	operands	result	comp	
MAX	V	S	S	operands	result	comp	
MEAN	V	S	S	operands	result	comp	
MEDIAN	V	S	S	operands	result	comp	
MIDRANGE	V	S	S	operands	result	comp	
MIN	V	S	S	operands	result	comp	
MOD	V	S	S	operands	result	comp	
NUMVAL	V	S	S	operands	result	comp	
NUMVAL-C	V	S	S	operands	result	comp	
ORD	V	S	S	operands	result	comp	

**1-148** Supported COBOL Statements  
*Unisys ASCII/UCS implementation and limitations*

Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
ORD-MAX	V	S	S	operands	result	comp	
ORD-MIN	V	S	S	operands	result	comp	
PRESENT-VALUE	V	S	S	operands	result	comp	
RANDOM	V	S	S	operands	result	comp	
RANGE	V	S	S	operands	result	comp	
REM	V	S	S	operands	result	comp	
REVERSE	V	S	S	operands	result	comp	
SIN	V	S	S	operands	result	comp	
SQRT	V	S	S	operands	result	comp	
STANDARD-DEVIATION	V	S	S	operands	result	comp	
SUM	V	S	S	operands	result	comp	
TAN	V	S	S	operands	result	comp	
UNDATE	V	S	S	operands	result	comp	
UPPER-CASE	V	S	S	operands	result	comp	
VARIANCE	V	S	S	operands	result	comp	
WHEN-COMPILED	V	S	S	operands	result	comp	

***Unisys ASCII/UCS implementation and limitations***

COPY libraries are not supported:

- Copybook names should be the same as PROC name.
- Each copybook should contain only one PROC.

- When COPY PRC statement is found, the parser looks for PRC.CPY file and copies text between the first pair of PROC / END directives.

Format 1 of COPY statements for ADMLP:

- ADMLP (CDML preprocessor of ASCII compiler) ignores the first two words of the procedure, leaving the original dataitem/paragraph name.

EXHIBIT and EXAMINE statements are allowed in both Unisys ASCII and UCS (they are prohibited in UCS by UCS Cobol specifications).

ATW always allocates 8-bit bytes, and all data are aligned on byte boundaries.

- The Unisys model has 9-bit bytes and 36-bit words (or 6-bit bytes, 36-bit words in some cases under ASCII COBOL).

PIC H9 is allowed in both Unisys ASCII and UCS (there is no H in UCS picture strings):

- H is converted to USAGE COMP phrase.
- The parser checks that no other USAGE clause was specified.

USAGE COMP-3 is allowed in both Unisys ASCII and UCS, equivalent to PACKED-DECIMAL.

PIC 1 — exact binary dataitems (bits).

- In Unisys subsequent bit dataitems are allocated contiguously. For instance, the following defines structure which occupies one 9-bit byte:

```
01 one-byte.  
    03 a pic 1(4).  
    03 b pic 1(5).
```

The ATW parser will perform the following actions:

- Detect groups of contiguous bit fields
- Count the memory size of the whole group

**1-150** Supported COBOL Statements  
*Unisys ASCII/UCS implementation and limitations*

- Set the same length and offset for each item (length = total length of memory area; offset = offset of the first item in the given group).
- For each allocated item, fill the exact bit offset and bit length.

BRE analyzes the correct length and offset. However, generation will not.

Synthesis converts each exact binary item separately; each item occupies integer number of bytes. Exact binary is converted to PIC 9(*n*) BINARY, where *n* is number of digits, required by binary-decimal equivalence to represent the same (not less) number of bits than original PIC 1(*m*).

$$n = 1 + \text{trunc} (\lg (2^{**} (m-1)))$$

USAGE BINARY-1 phrase is assumed but not required (USAGE for PIC 1 must not be specified in ASCII, and must be BINARY-1 in UCS).

**USAGE BINARY**

The table below shows the number of bytes needed for each dataitem:

Number of 9's	Number of bytes
1-2	1
3-5	2
6-7	3
8-10	4
11-13	5
14-15	6
16-18	7

Generation and Runtime do not support byte-mode storage allocation for binary dataitems. This may lead to unpredictable results on execution.

**USAGE COMP-1, USAGE COMP-2**

Allowed in both dialects.

**USAGE COMP-3**

- Allowed in ASCII (Does not exist in ASCII by Unisys ASCII COBOL specifications).

**USAGE COMP-4 (Fieldata 6-bit binary)**

- Allowed only in ASCII, warning message is issued for each item
- The table below shows the number of bytes needed for each dataitem:

Number of 9s	Number of bytes
1	1
2-3	2
4-5	3
6	4
7-8	5
9-10	6
11-12	7
13-14	8
15	9
16-17	10
18	11

Again, this may cause problems with REDEFINES in generation and BRE, when redefining item is ASCII. However, this eliminates the need to convert data on MOVES between Fieldata and ACSII.

**1-152** Supported COBOL Statements  
*Unisys ASCII/UCS implementation and limitations*

**USAGE COMP-5**

- Allowed in UCS .
- Only S9(4) or S9(9) pictures are allowed. Converted to USAGE BINARY, warning message issued.
- Truncation and SIZE ERROR behavior is exactly as for BINARY items (according to PICTURE string).

**USAGE COMP**

- The internal representation depends on option 'Unisys Cobol'.
- For ASCII: converted to USAGE BINARY
- For UCS: converted to USAGE DISPLAY, sign is trailing.

**USAGE DISP-2**

- Always allowed, converted to DISP-1 of standard parser (DBCS string)

**USAGE DISP-1 (Fielddata 6-bit characters)**

- Allowed only in ASCII, warning message is issued for each item.
- Each dataitem starts on byte boundary and occupies number of bytes equal to number of 'X' in PICTURE clause. For instance,

```
03 B redefines A PIC XX DISP.
```

Two 8-bit bytes are allocated for B and dependencies between A and B do not reflect Unisys hardware.

- In ATW 3 8-bit bytes will be allocated, when in Unisys it was 2 9-bit bytes (and in its turn would have been represented as 2 8-bit bytes in ATW).
- This may cause problems with REDEFINES in generation and BRE, when redefining item is ASCII. However, this eliminates the need to convert data on MOVES between Fielddata and ACSII.

***USAGE IS DISP-I18N***

- Allowed in both dialects (in ASCII it does not exist by Unisys ASCII COBOL specifications).
- Warning message is issued for each item. Converted to USAGE DISPLAY.
- In UCS comparison of DISP-I18N items is done based on locale setting, which is not used in ATW.

***USAGE IS BYTE-I18N***

- Allowed in both dialects (in ASCII it does not exist by Unisys ASCII COBOL specifications).
- Warning message is issued for each item. Converted to USAGE DISPLAY.
- In UCS when compare BYTE-I18N items of different length the smaller item is padded with binary zeroes, in ATW (being converted to USAGE DISPLAY) they are padded with blanks.
- In UCS some restriction on using BYTE-I18N in statements apply, but they are not forced in ATW.

***SYNCRONIZED and data alignment***

- In ASCII COMP-1 and COMP-2 are always aligned on word boundaries (4-bytes).
- This may cause problems with REDEFINES and group MOVES in generation and BRE.

***USAGE IS AREA-KEY, ...***

The table below describes the conversion of special usage clauses

<b>Usage</b>	<b>New usage</b>
01 data-item USAGE IS AREA-NAME.	01 data-item USAGE IS DISPLAY PIC X(12).
01 data-item USAGE IS RECORD-NAME.	01 data-item USAGE IS DISPLAY PIC X(30).
01 data-item USAGE IS SET-NAME.	01 data-item USAGE IS DISPLAY PIC X(30).

**1-154** Supported COBOL Statements  
*Unisys ASCII/UCS implementation and limitations*

<b>Usage</b>	<b>New usage</b>
01 data-item USAGE IS DATABASE-KEY.	01 data-item USAGE IS COMP PIC 9(10). 02 PAGE-NUM PIC 1(18). 02 RECORD-NUM PIC 1(18).
01 data-item USAGE IS AREA-KEY.	01 data-item USAGE IS BINARY-1 02 PAGE-NUM PIC 1(18). 02 RECORD-NUM PIC 1(18).
01 data-item USAGE IS ACCESS-KEY.	01 data-item USAGE IS DISPLAY PIC X(12).
01 data-item USAGE IS ACQUIRE-LIST <integer> DATABASE-KEYS.	01 data-item USAGE IS BINARY. 02 DBK-COUNT PIC 9(10). 02 DBK-LIST PIC 9(10) OCCURS 1 TO <integer> TIMES DEPENDING ON DBK-COUNT OF data-item
02 data-item USAGE IS DATE.	02 data-item USAGE IS DISPLAY. 03 DT-YEAR PIC 9(4). 03 DT-MONTH PIC 9(2). 03 DT-DAY PIC 9(2).
02 data-item USAGE IS TIME.	02 data-item USAGE IS DISPLAY. 03 DT-HOUR PIC 9(2). 03 DT-MINUTE PIC 9(2). 03 DT-SECOND PIC 9(2). 03 DT-PARTIAL-SECOND PIC 9(2).



Usage	New usage
02 data-item USAGE IS DATE-TIME.	02 data-item USAGE IS DISPLAY. 03 DT-YEAR PIC 9(4). 03 DT-MONTH PIC 9(2). 03 DT-DAY PIC 9(2). 03 DT-HOUR PIC 9(2). 03 DT-MINUTE PIC 9(2). 03 DT-SECOND PIC 9(2). 03 DT-PARTIAL-SECOND PIC 9(2).

**INVOKE SUBSCHEMA**

- Two copybooks should exist for each SubSchema:
  - *<schema-name>* \$*<subschema-name>* \$\$P\$PROC - with record definitions
  - *<schema-name>* \$*<subschema-name>* \$\$D\$PROC - with data names definitions
- These copybooks will be included automatically (without any COPY statements).
- Three clauses are processed to detect where to include data declarations:
  - DMCA INTO [ WORKING | COMMON | LINKAGE ]
  - COPYING RECORDS INTO [ WORKING | COMMON | LINKAGE ]
  - COPYING DATA-NAMES INTO [ WORKING | COMMON | LINKAGE ]
- Data Copybooks should be created manually from SDDL definition or extracted from Unisys environment.
- At present, all other clauses in the SubSchema section are ignored silently.

**COMMON SECTION**

- Common section is processed as part of Linkage section.

1-156 Supported COBOL Statements  
Supported Unisys CDML COBOL statements

### Data Types summary

Unisys Usage/Picture	ASCII/UCS	ATW Usage	ATW Picture	Warning	Notes
PIC H9(n)		BINARY	9(n)		
PIC 1(n) BINARY-1		BINARY	9(m)	Yes	Parser and BRE uses exact bit sizes
COMP	ACSII	BINARY	As is		
COMP	UCS	DISPLAY	As is		
COMP-5		BINARY	As is	Yes	
COMP-4	ASCII	BINARY	As is	Yes	
DISP-1	ASCII	DISPLAY	As is	Yes	
DISP-2		DISP-1	As is		
DISP-I18N		DISPLAY	As is	Yes	
BYTE-I18N		DISPLAY	As is	Yes	

### Supported Unisys CDML COBOL statements

#### Data Division

Clauses	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relations
DMCA	<ul style="list-style-type: none"> <li><b>Format</b> [DMCA [AND RUN-UNIT-STATISTICS] &lt;IS   ARE&gt; &lt; COMMON   LINKAGE   WORKING &gt; [ NUMBER OF ERROR-ITEMS IS <i>int-4</i> ]]</li> </ul>	V	S	S		
ERROR	<ul style="list-style-type: none"> <li><b>Format</b> [ ERROR RECOVERY IS <i>general-error-para</i> ]</li> </ul>	V	S	S		

Clauses	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relations
INVOKE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> INVOKE SUBSCHEMA <i>subschema-name</i> IN FILE <i>file-name</i> OF SCHEMA <i>schema-name</i> [KEY FOR INVOKE IS <i>lit-1</i> ] [ENVIRONMENT IS HVTIP] [COPYING RECORDS INTO < COMMON   LINKAGE   WORKING > [COPYING DATA-NAMES INTO < COMMON   LINKAGE   WORKING >	V	S	S		
OVERLAY	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [OVERLAY <i>rec-delivery-area</i> WITH < ALL   <i>rec-name-4</i> [ , <i>rec-name-5</i> ]>] [OVERLAY <i>rec-name-6</i> WITH <i>rec-name-7</i> [ , <i>rec-name-8</i> ]]	V	S	S		
POINTER	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [POINTER < AREAS > FOR INITIAL LOAD < IS   ARE > <i>int-3</i> ]	V	S	S		
PRIORITY	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [PRIORITY IS <i>int-1</i> ]	V	S	S		
RECORD DELIVERY AREA	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [RECORD DELIVERY-AREA IS <i>rec-delivery-area</i> [< WORKING   COMMON   LINKAGE >] [LENGTH IS <i>int-2</i> WORDS]]	V	S	S		
ROLLBACK	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [ ROLLBACK IS <i>rollback-para</i> ]	V	S	S		
RUN-UNIT-ID	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [ RUN-UNIT-ID IS <i>run-unit-id</i> ]	V	S	S		
SAVE DATA	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [SAVE DATA INCLUDES << COMMAND   RUN-UNIT > QUIT-BEFORE-LOOKS   DEFFERED-UPDATES >]]	VO	N	S		

1-158 Supported COBOL Statements  
Supported Unisys CDML COBOL statements

**Procedure Division**

**Control Commands**

Command	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relation
CLOSE	<ul style="list-style-type: none"> <li><b>Format</b> CLOSE &lt; ALL   a-name-1 [ ,a-name-2]   id-1 [ ,id-2 &gt; [ON ERROR GO TO <i>error-para</i>].</li> </ul>	V	S	S		
DEPART	<ul style="list-style-type: none"> <li><b>Format</b> DEPART [WITH &lt; MESSAGE &lt; ADVANCE   TERMINATE &gt;   ROLLBACK [AND MESSAGE &lt; DISCARD   REQUEUE   RETAIN &gt; ] &gt;] [ON ERROR GO TO <i>error-para</i>].</li> </ul>	V	S	S		
IMPART	<ul style="list-style-type: none"> <li><b>Format</b> IMPART [ON ERROR GO TO <i>error-para</i>].</li> </ul>	V	S	S		
OPEN	<ul style="list-style-type: none"> <li><b>Format 1</b> OPEN ALL [USAGE-MODE IS &lt; [&lt;EXCLUSIVE   PROTECTED&gt;] &lt; RETIRIEVAL   UPDATE &gt;   INITIAL LOAD &gt; ] .</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> OPEN &lt; a-name-1   id-1 &gt; [USAGE-MODE IS &lt; [&lt;EXCLUSIVE   PROTECTED&gt;] &lt; RETIRIEVAL   UPDATE &gt;   INITIAL LOAD &gt; ] &lt; a-name-2   id-2 &gt; [USAGE-MODE IS &lt; [&lt;EXCLUSIVE   PROTECTED&gt;] &lt; RETIRIEVAL   UPDATE &gt;   INITIAL LOAD &gt; ] ] [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		

**Retrieval Commands**

Command	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relation
ACQUIRE	<ul style="list-style-type: none"> <li><b>Format</b>                      ACQUIRE &lt; int-1   id-1 &gt; DATABASE-KEYS [BEGINNING AT id-2] TO id-3                      FROM &lt; set-name-1   id-4 &gt; [USING DEFINED KEYS]                      [AT END GO TO <i>end-para</i>] [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		
FIND/FETCH	<ul style="list-style-type: none"> <li><b>Format</b>                      &lt; FIND   FETCH &gt; <i>rse</i> [SUPPRESS &lt; ALL   [&lt; RECORD   AREA   &lt; SETS   <i>set-name-1</i> [ ,<i>set-name-2</i>]   <i>id-1</i> [ ,<i>id-2</i>] &gt;] &gt; CURRENCY UPDATES]                      [AT AND GO <i>end-para</i>]                      [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		Program Reads NetDBRecord (*for FETCH only)  Program Reads NetDBDecision
rse as FIND statement argu- ment	<ul style="list-style-type: none"> <li><b>Format 1</b>                      [&lt; <i>rec-name</i>   <i>id-3</i> &gt; RECORD] <i>id-4</i> [,<i>id-5</i>]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>                      CURRENT RECORD WITHIN &lt; <i>rec-name-2</i>   <i>id-6</i> &gt; RECORD</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b>                      &lt; OWNER   CURRENT   NEXT   PRIOR   FIRST   LAST   <i>id-7</i> &gt; RECORD WITHIN &lt; <i>set-name-3</i> SET   <i>id-8</i> SET   <i>a-name-1</i> AREA   <i>id-9</i> AREA &gt;</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 4</b>                      &lt; NEXT   PRIOR   FIRST   LAST   <i>id-10</i> &gt; &lt; <i>rec-name-4</i>   <i>id-11</i> &gt; WITHIN &lt; <i>set-name-4</i> SET   <i>id-12</i> SET   <i>a-name-2</i> AREA   <i>id-13</i> AREA &gt;</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 5</b>                      [&lt; NEXT [DUPLICATE]   PRIOR   FIRST   LAST &gt; WITHIN] &lt; <i>rec-name-4</i>   <i>id-14</i> &gt; RECORD</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 6</b>                      &lt; <i>rec-name-5</i>   <i>id-15</i> &gt; VIA &lt; <i>set-name-5</i>   <i>id-16</i> &gt; [USING <i>db-id-1</i> [ ,<i>db-id-2</i>]]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 7</b>                      NEXT DUPLICATE [ <i>rec-name-6</i> RECORD] WITHIN &lt; <i>set-name-6</i>   <i>id-17</i> &gt; USING <i>db-id-3</i> [ ,<i>db-id-4</i>]</li> </ul>	V	S	S		

**1-160** Supported COBOL Statements  
Supported Unisys CDML COBOL statements

Command	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relation
rse as FETCH statement argument	<ul style="list-style-type: none"> <li><b>Format 1</b> [&lt; rec-name   id-3 &gt; RECORD] id-4 [,id-5]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> CURRENT RECORD WITHIN &lt; rec-name-2   id-6 &gt; RECORD</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt; OWNER   CURRENT   NEXT   PRIOR   FIRST   LAST   id-7 &gt; RECORD WITHIN &lt; set-name-3 SET   id-8 SET   a-name-1 AREA   id-9 AREA &gt;</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 4</b> &lt; NEXT   PRIOR   FIRST   LAST   id-10 &gt; &lt; rec-name-4   id-11 &gt; WITHIN &lt; set-name-4 SET   id-12 SET   a-name-2 AREA   id-13 AREA &gt;</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 5</b> [&lt; NEXT [DUPLICATE]   PRIOR   FIRST   LAST &gt; WITHIN] &lt; rec-name-4   id-14 &gt; RECORD</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 6</b> &lt; rec-name-5   id-15 &gt; VIA &lt; set-name-5   id-16 &gt; [USING db-id-1 [ ,db-id-2]]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 7</b> NEXT DUPLICATE [ rec-name-6 RECORD] WITHIN &lt; set-name-6   id-17 &gt; USING db-id-3 [ ,db-id-4]</li> </ul>	V	S	S		
GET	<ul style="list-style-type: none"> <li><b>Format</b> GET [&lt; rec-name   id-1 &gt; RECORD] [ON ERROR GO TO error-para]</li> </ul>	V	S	S	refer to rec-name (NetDbRecord)  defines id-1 (Decision)	Program Reads NetDbRecord  Program Reads NetDbDecision

### Update Commands

Command	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relation
DELETE	<ul style="list-style-type: none"> <li><b>Format</b>                      DELETE [&lt; rec-name   id-1 &gt; RECORD] [&lt; ONLY   ALL &gt;]                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S	refer to rec-name (NetDbRecord)  defines id-1 (Decision)	Program Deletes NetDbRecord  Program Deletes NetDbDecision
INSERT	<ul style="list-style-type: none"> <li><b>Format 1</b>                      INSERT [&lt; rec-name   id-1 &gt; RECORD] INTO &lt; set-name -1   id-2 &gt; [, INTO &lt; set-name-2   id-3 &gt;]                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>                      INSERT [&lt; rec-name   id-1 &gt; RECORD] INTO ALL SETS                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S		
MODIFY	<ul style="list-style-type: none"> <li><b>Format</b>                      MODIFY [&lt; rec-name   id-1 &gt; RECORD] [db-id-1 [ , db-id-2]]                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S	refer to rec-name (NetDbRecord)  defines id-1 (Decision)	Program Updates NetDbRecord  Program Updates NetDbDecision
RE-MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b>                      REMOVE [&lt; rec-name   id-1 &gt; RECORD] FROM &lt; set-name -1   id-2 &gt; [, &lt; set-name-2   id-3 &gt;]                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>                      REMOVE [&lt; rec-name   id-1 &gt; RECORD] FROM ALL SETS                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S		
STORE	<ul style="list-style-type: none"> <li><b>Format</b>                      STORE &lt; rec-name   id-1 &gt; [SUPPRESS &lt; ALL   [&lt; RECORD   AREA   &lt; SETS   set-name-1 [ , set-name-2]   id-2 [ , id-3 ] &gt;] &gt; CURRENCY UPDATES]                      [ON ERROR GO TO error-para]</li> </ul>	V	S	S	refer to rec-name (NetDbRecord)  defines id-1 (Decision)	Program Inserts NetDbRecord  Program Inserts NetDbDecision

**1-162** Supported COBOL Statements  
*Supported Unisys CDML COBOL statements*

**Support Commands**

Command	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relation
FREE	<ul style="list-style-type: none"> <li><b>Format</b>            FREE [WITH MESSAGE &lt; ADVANCE   TERMINATE &gt;]            [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		
IF	<ul style="list-style-type: none"> <li><b>Format 1</b>            IF &lt; <i>set-name-1</i>   <i>id-1</i> &gt; SET [NOT] EMPTY; &lt; <i>statement-1</i>   NEXT SENTENCE &gt; [ELSE &lt; <i>statement-2</i>   NEXT SENTENCE &gt;]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>            IF RECORD [NOT] &lt; OWNER   MEMBER &gt; OF &lt; <i>set-name-1</i>   <i>id-1</i>   ANY &gt; SET &lt; <i>statement-3</i>   NEXT SENTENCE &gt; [ELSE &lt; <i>statement-4</i>   NEXT SENTENCE &gt;]</li> </ul>	V	S	S		
KEEP	<ul style="list-style-type: none"> <li><b>Format</b>            KEEP [&lt; <i>rec-name</i>   <i>id-1</i>&gt; RECORD]            [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		
LOG	<ul style="list-style-type: none"> <li><b>Format</b>            KEEP <i>id-1</i> WORDS [ FROM <i>id-2</i> ] [FOR RECOVERY POINT]            [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b>            MOVE CURRENCY STATUS FOR &lt; <i>RUN-UNIT</i>   <i>set-name</i> SET   <i>id-3</i> SET   <i>a-name</i> AREA   <i>id-2</i> AREA   <i>id-1</i> RECORD   <i>rec-name</i> RECORD &gt; TO <i>id-4</i>            [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>            MOVE &lt; AREA-KEY   AREA-NAME &gt; FOR &lt; <i>RUN-UNIT</i>   <i>set-name</i> SET   <i>id-7</i> SET   <i>a-name</i> AREA   <i>id-6</i> AREA   <i>id-5</i> RECORD   <i>rec-name</i> RECORD   <i>id-8</i> &gt; TO &lt; <i>id-4</i>   <i>id-10</i> &gt;            [ON ERROR GO TO <i>error-para</i>]</li> </ul>	V	S	S		

**Unisys support**

**Problems:**



- Byte-storage mode for USAGE COMP
- USAGE BINARY-1
- 9-bit memory model

**Solution:**

Unisys program memory should be converted into memory model supported by existing RW run-time support. It can be achieved by allocating extra memory for “bad” data items.

Reallocating program memory will have an influence on redefined/redefining data items and group assignments. Dependent data items should be correspondingly modified to reflect changes. Such modifications are very complicated when dependent data items have incongruous internal structure bounds.

**Implementation**

Data items with COMP usage should be enlarged to have one of the nearest supported size values \_2, 4 or 8 bytes, which is large enough to hold original maximum value (e.g., 4 bytes for PIC 9(5) COMP).

Data items with BINARY-1 usage should be converted to have ‘9’ picture:

```
01 A PIC 1. => 01 A PIC 9.
```

Warning messages are issued in these cases:

- 1 structure with corrected data items is redefined or redefining data item (REDEFINES clause);
- 2 structure with corrected data items is renamed or renaming data item (RENAMES clause);
- 3 structure with corrected data items is used in MOVE statement (group move).

All tree modifications should be performed before Cobol synthesis during business logic transition methods; verification methods should not depend on them to save old memory allocation for BRE, GDF, etc. Additional exported method is added to CobolUtil.dll:

```
DllExport CNode* ConvertUnisysTypes(CNode *n);
```

**1-164** Supported COBOL Statements  
*Supported Unisys CDML COBOL statements*

It should be called immediately before Cobol synthesis.

Complicated cases:

```
-----  
01 A PIC 1(18) .  
01 B PIC 1 OCCURS 18 REDEFINES A .  
-----  
01 A PIC 1(18) .  
01 B PIC X(2) REDEFINES A .
```

***Other problems:***

Common-Storage is added to Linkage Section with warning (during Cobol synthesis).

## Supported Micro Focus COBOL statements

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### Special registers

Registers	Parser	IA	CA/GDF
ADDRESS OF	V	S	S
LENGTH OF	V	S	S
RETURN-CODE	V	S	S
SORT-CONTROL	V	S	S
SORT-RETURN	V	S	S

### Literals

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literals	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN""T WRONG"</li> </ul>	V	V	S
	<ul style="list-style-type: none"> <li><b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	V	S
	<ul style="list-style-type: none"> <li><b>Format 3: hexadecimal notation</b> <i>X"hexadecimal-digits"</i></li> </ul>	V	V	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	V	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> <i>&lt;+   -&gt; mantissa E &lt;+   -&gt; exponent</i></li> </ul>	V	V	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	V	S

### Referencing names

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1 [&lt;IN   OF&gt; library-name-1]</i></li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1 [&lt;IN   OF&gt; section-name-1]</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	S

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: subscripting</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li><b>Format 1: Data Division</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: Special-Names paragraph</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>mnemonic-name-1</i>]</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>integer-1</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>]&gt;</li> </ul>	V	S	S
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>data-name-1</i>   FUNCTION <i>function-name-1</i> (<i>arguments</i>)&gt; (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S

1-168 Supported COBOL Statements  
Supported Micro Focus COBOL statements

### Identification Division

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
PROGRAM-ID paragraph	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <pre>[ &lt;ID   IDENTIFICATION&gt; DIVISION ] [ PROGRAM-ID. <i>program-name</i> [ &lt; IS &lt;COMMON   INITIAL&gt; PROGRAM   IS EXTERNAL PROGRAM &gt; ] ]. [AUTHOR. [<i>comment-entry</i>]] [INSTALLATION. [<i>comment-entry</i>]] [DATE-WRITTEN. [<i>comment-entry</i>]] [DATE-COMPILED. [<i>comment-entry</i>]] [SECURITY. [<i>comment-entry</i>]]</pre>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Program Entry Point

### Environment Division

#### Input-Output Section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b></li> </ul> <pre>SELECT [&lt;OPTIONAL   NOT OPTIONAL&gt;] <i>file-name-1</i> ASSIGN TO &lt; [&lt; EXTERNAL   DYNAMIC&gt; ] [&lt;&lt; LINE ADVANCING   MULTIPLE &lt; REEL   UNIT &gt;&gt; FILE   DISK &gt; ] &lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i> &gt;   [&lt; EXTERNAL   DYNAMIC&gt; ] &lt; DISK   KEYBOARD   DISPLAY   PRINTER   PRINTER-1 &gt; [&lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i> &gt; ] DISK FROM <i>data-name-1</i></pre>	V	S	defines program-name.assignment-name-1 (File)	
	<pre>[RESERVE <i>integer-1</i> &lt; AREA   AREAS &gt;] [[ORGANISATION IS] [&lt;RECORD&gt;] SEQUENTIAL] [PADDING [CHARACTER] [IS] &lt;<i>data-name-5</i>   <i>literal-2</i>&gt;] [RECORD DELIMITER [IS] &lt;STANDARD-1   character-string&gt;] [ACCESS MODE IS SEQUENTIAL] [LOCK MODE IS &lt;&lt; MANUAL   AUTOMATIC&gt; [WITH &lt;LOCK ON [MULTIPLE] &lt; RECORD   RECORDS&gt;   ROLL-BACK&gt;]   EXCLUSIVE&gt;]</pre>	VO	N		

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b></li> </ul> <p>SELECT [&lt; OPTIONAL   NOT OPTIONAL&gt;] <i>file-name-1</i> ASSIGN TO &lt; [&lt; EXTERNAL   DYNAMIC&gt;] DISK &lt; <i>external-file-reference</i>   <i>literal-1</i>&gt;   [&lt; EXTERNAL   DYNAMIC&gt;]   DISK FROM <i>data-name-1</i>&gt;</p>	V	S	defines program- name.assignment- name-1 (File)	
	<p>[RESERVE <i>integer-1</i> &lt; AREA   AREAS &gt;] [ORGANIZATION IS] INDEXED [ACCESS MODE IS &lt;SEQUENTIAL   RANDOM   DYNAMIC&gt;] [LOCK MODE IS &lt;&lt; MANUAL   AUTOMATIC&gt; [WITH &lt;LOCK ON [MULTIPLE] &lt; RECORD   RECORDS&gt;   ROLL- BACK&gt;]   EXCLUSIVE&gt;] RECORD KEY IS &lt; <i>data-name-5</i>   <i>split-key-name-1</i> = <i>data-name-6</i> [<i>data-name-7</i>]&gt; [ALTERNATE [RECORD] KEY IS &lt; <i>data-name-8</i>   <i>split-key-name-2</i> = <i>data-name-9</i> [<i>data-name-10</i>]&gt; [WITH DUPLI- CATES] [SUPPRESS [WHEN] &lt; ZERO S   ZEROES   SPACES   [ALL] <i>literal</i> &gt;]] [FILE STATUS IS <i>data-name-2</i>]</p>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b></li> </ul> <p>SELECT [&lt; OPTIONAL   NOT OPTIONAL&gt;] <i>file-name-1</i> ASSIGN TO &lt; [&lt; EXTERNAL   DYNAMIC&gt;] DISK &lt; <i>external-file-reference</i>   <i>literal-1</i>&gt;   [&lt; EXTERNAL   DYNAMIC&gt;]   DISK FROM <i>data-name-1</i>&gt;</p>	V	S	defines program- name.assignment- name-1 (File)	
	<p>[RESERVE <i>integer-1</i> &lt; AREA   AREAS &gt;] [ORGANIZATION IS] RELATIVE [ACCESS MODE IS &lt;SEQUENTIAL [RELATIVE KEY IS <i>data-name-5</i>]   &lt;RANDOM   DYNAMIC&gt; RELATIVE KEY IS <i>data-name-5</i>&gt;] [LOCK MODE IS &lt;&lt; MANUAL   AUTOMATIC&gt; [WITH &lt;LOCK ON [MULTIPLE] &lt; RECORD   RECORDS&gt;   ROLL- BACK&gt;]   EXCLUSIVE&gt;] [FILE STATUS IS <i>data-name-2</i>]</p>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: line-sequential file-control entries</b></li> </ul> <p>SELECT [&lt;OPTIONAL   NOT OPTIONAL&gt;] <i>file-name-1</i> ASSIGN TO &lt; [&lt; EXTERNAL   DYNAMIC&gt;] [&lt;FILE   DISK &gt;] &lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i>&gt;   [&lt; EXTERNAL   DYNAMIC&gt;] &lt; DISK   KEYBOARD   DISPLAY   PRINTER   PRINTER-1&gt; [&lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i>&gt;]   DISK FROM <i>data-name-1</i>&gt;</p>	V	S	defines program- name.assignment- name-1 (File)	
	<p>[[ORGANISATION IS] [LINE] SEQUENTIAL] [ACCESS MODE IS SEQUENTIAL] [LOCK MODE IS &lt;&lt; MANUAL   AUTOMATIC&gt; [WITH &lt;LOCK ON [MULTIPLE] &lt; RECORD   RECORDS&gt;   ROLL- BACK&gt;]   EXCLUSIVE&gt;] [FILE STATUS IS <i>data-name-2</i>] [PASSWORD IS <i>data-name-4</i>]</p>	VO	N		

**1-170** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 5: sort-merge file-control entries</b> SELECT <i>file-name</i> ASSIGN TO &lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i> &gt; [&lt;SORT   FILE&gt; STATUS IS <i>data-name-2</i>]</li> </ul>	V	S	defines program-name.assignment-name-1 (File)	
	<ul style="list-style-type: none"> <li>SELECT <i>file-name</i> ASSIGN TO &lt;EXTERNAL   DYNAMIC &gt; &lt; <i>external-file-reference</i>   <i>data-name-1</i>   <i>literal-1</i> &gt; [&lt;SORT   FILE&gt; STATUS IS <i>data-name-2</i>]</li> </ul>	VO	N		

**Configuration Section**

Paragraphs and entries	Format	Parser	IA
ENVIROMENT DIVISION paragraph	<ul style="list-style-type: none"> <li><b>Format</b> [EVIROMENT DIVISION] [CONFIGURATION SECTION] [SOURCE-COMPUTER. [<i>source-computer-name</i> [WITH DEBUGGING MODE].]] [OBJECT-COMPUTER. <i>object-computer-name</i> [MEMORY SIZE <i>integer</i> &lt;WORDS   CHARACTERS   MODULES&gt;] [PROGRAM COLLATING SEQUENCE IS <i>alphabet-name</i>] [SEGMENT-LIMIT IS <i>segment-number</i>.] [SPECIAL-NAMES. [ &lt; SWITCH-0   SWITCH-1   SWITCH-2   SWITCH-3   SWITCH-4   SWITCH-5   SWITCH-6   SWITCH-7   SWITCH-8   <i>function-name</i> &gt; &lt;[IS] <i>mnemonic-name</i> [ON STATUS [IS] <i>condition-name-1</i> [OFF STATUS [IS] <i>condition-name-2</i>]]   [IS] <i>mnemonic-name</i> [OFF STATUS [IS] <i>condition-name-2</i> [ON STATUS [IS] <i>condition-name-1</i>]]   [ON STATUS [IS] <i>condition-name-1</i> [OFF STATUS [IS] <i>condition-name-2</i>]]   [OFF STATUS [IS] <i>condition-name-2</i> [ON STATUS [IS] <i>condition-name-1</i>]]&gt;] [ALPHABET <i>alphabet-name-1</i> IS &lt;&lt;STANDARD-1   STANDARD-2   NATIVE   ASCII   EBCDIC&gt;   &lt;<i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>   ALSO <i>literal-3</i>&gt;&gt;] [SYMBOLIC CHARACTERS &lt;&lt;<i>symbolic-character</i> &lt;ARE   IS&gt; &lt;<i>integer-1</i>&gt;&gt; [IN <i>alphabet-name-2</i>]] [CLASS <i>class-name-1</i> IS &lt;<i>literal-4</i> [&lt;THROUGH   THRU&gt; <i>literal-5</i>&gt;] [CURRENCY SIGN IS <i>literal-6</i>] [DECIMAL-POINT [IS] COMMA] [NUMERIC SIGN IS TRAILING SEPARATE] [CALL-CONVENTION <i>integer-2</i> IS <i>mnemonic-name</i>] [CONSOLE IS CRT] [INPUT-OUTPUT SECTION.] [FILE-CONTROL.] &lt;<i>file-control-entry</i>&gt; [I-O-CONTROL. [RERUN [ON&lt; <i>file-name-1</i>   <i>character-string</i>&gt;] EVERY &lt;&lt; [END OF] &lt; REEL   UNIT &gt;   <i>integer-1</i> RECORDS&gt; OF <i>file-name-2</i>   <i>integer-2</i> CLOCK-UNITS   <i>condition-name</i> &gt;] [SAME [&lt; RECORD   SORT   SORT-MERGE] AREA FOR <i>file-name-3</i> &lt;<i>file-name-4</i>&gt;] [MULTIPLE FILE TAPE CONTAINS &lt;<i>file-name-5</i> [POSITION <i>integer-3</i>&gt;]</li> </ul>	VO	N



ENVIROMENT DIVISION paragraph	<ul style="list-style-type: none"> <li><b>Format</b>            [EVIROMENT DIVISION]            [SPECIAL-NAMES. [ &lt; SWITCH-0   SWITCH-1   SWITCH-2   SWITCH-3   SWITCH-4   SWITCH-5   SWITCH-6   SWITCH-7   SWITCH-8   <i>function-name</i> &gt; &lt;[IS] <i>mnemonic-name</i> [ON STATUS [IS] <i>condition-name-1</i> [OFF STATUS [IS] <i>condition-name-2</i>]   [IS] <i>mnemonic-name</i> [OFF STATUS [IS] <i>condition-name-2</i> [ON STATUS [IS] <i>condition-name-1</i>]   [ON STATUS [IS] <i>condition-name-1</i> [OFF STATUS [IS] <i>condition-name-2</i>]   [OFF STATUS [IS] <i>condition-name-2</i> [ON STATUS [IS] <i>condition-name-1</i>]]&gt;]            [DECIMAL-POINT [IS] COMMA]]         </li> </ul>	V	S
-------------------------------	--	---	---

**Data Division**

Entries and Clauses	Format	Parser	IA	Sup:	CA/GDF	
					Pairs	Relation
Data Division paragraph	<ul style="list-style-type: none"> <li><b>Format</b>            [DATA DIVISION.]            [[FILE SECTION.] [&lt; <i>file-description-entry</i> &lt; <i>record-description-entry</i> &gt;   &gt;<i>sort-merge-file-description-entry</i> &lt;<i>record-description-entry</i>&gt;   <i>report-file-description-entry</i> &gt;]]            [WORKING-STORAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [LOCAL-STORAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [LINKAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [REPORT SECTION [ <i>report-description-entry</i> &lt; <i>report-group-description-entry</i>&gt;]]            [SCREEN SECTION [ <i>screen-description-entry</i> ]]]         </li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format</b>            [DATA DIVISION.]            [[FILE SECTION.] [&lt; <i>file-description-entry</i> &lt; <i>record-description-entry</i> &gt;   &gt;<i>sort-merge-file-description-entry</i> &lt;<i>record-description-entry</i>&gt;   <i>report-file-description-entry</i> &gt;]]            [WORKING-STORAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [LOCAL-STORAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [LINKAGE SECTION. [77-level-description-entry   <i>record-description-entry</i>]]            [COMMUNICATION SECTION. [ <i>communication-description-entry</i> [ <i>record-description-entry</i> ]]]            [REPORT SECTION [ <i>report-description-entry</i> &lt; <i>report-group-description-entry</i>&gt;]]            [SCREEN SECTION [ <i>screen-description-entry</i> ]]]         </li> </ul>	VO	N	N		

1-172 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Entries and Clauses	Format	Parser	IA	Sup:	CA/GDF		
					Pairs		Relation
File Description (FD) paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> FD <i>file-name-1</i> &lt;[IS EXTERNAL]   [IS GLOBAL]&gt;</li> </ul>	V	S	S			
	<[IS EXTERNAL]   [IS GLOBAL]> [BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   CONTAINS <i>integer-6</i> TO <i>integer-7</i> CHARACTERS   IS VARYING IN SIZE [[FROM] <i>integer-4</i> ] [TO] <i>integer-5</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF < <i>implementor-name-1</i> IS < <i>data-name-3</i>   <i>literal-1</i> >   FILE-ID IS < <i>data-name-6</i>   <i>literal-3</i> >>] [RECORDINFG MODE IS < F   V   U   S   FIXED VARIABLE >] [DATA <RECORD IS   RECORDS ARE> <i>data-name-7</i> [ <i>data-name-8</i> ]] [LINAGE IS < <i>data-name-9</i>   <i>integer-8</i> > LINES [WITH FOOTING AT < <i>data-name-10</i>   <i>integer-9</i> >] [LINES AT TOP < <i>data-name-11</i>   <i>integer-10</i> >] [LINES AT BOTTOM < <i>data-name-12</i>   <i>integer-11</i> >] [CODE-SET IS <i>alphabet-name</i> [FOR < <i>id-1</i> >]]. < <i>record-description-entry</i> >	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2: relative/indexed files</b> FD <i>file-name-1</i></li> </ul>	V	S	S			
	<[IS EXTERNAL]   [IS GLOBAL]> [BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   CONTAINS [ <i>integer-6</i> TO] <i>integer-7</i> CHARACTERS   IS VARYING IN SIZE [[FROM] <i>integer-4</i> ] [TO] <i>integer-5</i> ] CHARACTERS] [DEPENDING [ON] <i>data-name-1</i> ] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF < <i>implementor-name-1</i> IS < <i>data-name-3</i>   <i>literal-1</i> >   FILE-ID IS < <i>data-name-6</i>   <i>literal-3</i> >>] [RECORDINFG MODE IS < F   V   U   S   FIXED VARIABLE >] [DATA <RECORD IS   RECORDS ARE> <i>data-name-7</i> [ <i>data-name-8</i> ]] < <i>record-description-entry</i> >	VO	N	N			
File Description (FD) paragraph	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b> FD <i>file-name-1</i></li> </ul>	V	S	S			
	< [IS EXTERNAL]   [IS GLOBAL]> [BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   IS VARYING IN SIZE [FROM <i>integer-4</i> ] [TO] <i>integer-5</i> ] CHARACTERS [DEPENDING ON <i>data-name-1</i> ]   CONTAINS [ <i>integer-6</i> TO] <i>integer-7</i> CHARACTERS >] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF < <i>implementor-name-1</i> IS < <i>data-name-3</i>   <i>literal-1</i> >   FILE-ID IS < <i>data-name-6</i>   <i>literal-3</i> >>] [RECORDING MODE IS < F   V   FIXED VARIABLE >] [DATA <RECORD IS   RECORDS ARE> <i>data-name-7</i> [ <i>data-name-8</i> ]] [CODE-SET IS <i>alphabet-name</i> [FOR < <i>id-1</i> >]]. < <i>record-description-entry</i> >	VO	N	N			

Entries and Clauses	Format	Parser	IA	Sup:	CA/GDF		
					Pairs		Relation
	<ul style="list-style-type: none"> <li><b>Format 4: sort/merge files</b> SD <i>file-name-1</i></li> </ul>	V	S	S			
	[RECORD <CONTAINS <i>integer-3</i> CHARACTERS   CONTAINS [ <i>integer-4</i> TO] <i>integer-5</i> CHARACTERS   IS VARYING IN SIZE [[FROM <i>integer-6</i> ] [TO <i>integer-7</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ]->] [DATA <RECORD IS   RECORDS ARE> < <i>data-name-3</i> >] [VALUE OF FILE-ID IS < <i>data-name-4</i>   <i>literal-1</i> >]	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 5: report files</b> FD <i>file-name-1</i></li> </ul>	V	S	S			
	<[IS EXTERNAL]   [IS GLOBAL]FD <i>file-name-1</i> <[IS EXTERNAL]   [IS GLOBAL]> [BLOCK CONTAINS [ <i>integer-1</i> TO] <i>integer-2</i> <CHARACTERS   RECORDS>] [RECORD <CONTAINS <i>integer-3</i> CHARACTERS   CONTAINS <i>integer-6</i> TO <i>integer-7</i> CHARACTERS   IS VARYING IN SIZE [[FROM] <i>integer-4</i> ] [TO <i>integer-5</i> ] CHARACTERS] [DEPENDING ON <i>data-name-1</i> ]->] [LABEL <RECORD IS   RECORDS ARE> <STANDARD   OMITTED>] [VALUE OF << <i>data-name-3</i> IS < <i>data-name-4</i>   <i>literal-1</i> >>   FILE-ID IS < <i>data-name-3</i>   <i>literal-1</i> >>] [RECORDINFG MODE IS < F   V   FIXED VARIABLE >] [DATA <RECORD IS   RECORDS ARE> <i>data-name-4</i> [ <i>data-name-5</i> ]] [CODE-SET IS <i>alphabetic-name-1</i> .] [< REPORT IS   REPORT ARE > < <i>report-name-1</i> >]	VO	N	N			
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level number</i> [&lt; <i>data-name-1</i>   FILLER &gt;] [REDEFINES <i>data-name-2</i>] [IS GLOBAL] [IS TYPEDEF] [&lt; PICTURE   PIC &gt; IS <i>character-string</i>] [[USAGE IS] &lt; BINARY   COMPUTATIONAL   COMP   COMPUTATIONAL-1   COMP-1   COMPUTATIONAL-2   COMP-2   COMPUTATIONAL-3   COMP-3   COMPUTATIONAL-4   COMP-4   COMPUTATIONAL-5   COMP-5   COMPUTATIONAL-X   COMP-X   <i>typedef-name-1</i>   INDEX   PACKED-DECIMAL   DISPLAY   POINTER   DISPLAY-1   PROCEDURE-POINTER &gt;] [&lt; OCCURS <i>integer-2</i> TIMES [&lt; ASCENDING   DESVENDING &gt; KEY IS &lt;<i>data-name-3</i>&gt;] [INDEXED BY &lt;<i>index-name-1</i>&gt;]   OCCURS [<i>integer-1</i> TO] <i>integer-2</i> TIMES DEPENDING ON <i>data-name-4</i> [&lt; ASCENDING   DESCENDING &gt; KEY IS &lt;<i>data-name-3</i>&gt;] [INDEXED BY &lt;<i>index-name-1</i>&gt;] &gt;] [SIGN IS &lt; LEADING   TRAILING &gt; [SEPARATE CHARACTER]] [&lt; SYNCHRONIZED   SYNC &gt; [&lt;LEFT   RIGHT&gt;]] [&lt; JUSTIFIED   JUST &gt; RIGHT] [BLANK WHEN &lt; ZERO   ZEROS   ZEROES &gt;] [VALUE IS <i>literal-1</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt; THROUGH   THRU &gt; <i>data-name-3</i>].</li> </ul>	V	S	S			

**1-174** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup:	Pairs	Relation	
	<ul style="list-style-type: none"> <li><b>Format 3</b> 88 <i>condition-name</i> &lt; VALUE IS   VALUE ARE &gt; [<i>literal-2</i> &lt; THROUGH   THRU &gt; <i>literal-3</i>] [WHEN SET TO FALSE <i>literal-4</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 4</b> 78 <i>constant-name</i> VALUE IS &lt; <i>literal-5</i>   NEXT   START OF <i>data-name-1</i>   LENGHT OF <i>data-name-2</i> &gt; [&lt; +   -   *   /   AND   OR &gt; &lt; <i>integer-1</i>   NEXT   START OF <i>data-name-3</i>   LENGHT OF <i>data-name-4</i> &gt;].</li> </ul>	V	S	S			
Communication Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> CD <i>cd-name</i> FOR [INITIAL] INPUT [&lt; [SYMBOLIC QUEUE IS <i>data-name-1</i>] &lt; [SYMBOLIC SUB-QUEUE IS <i>data-name-2</i>]   [SYMBOLIC SUB-QUEUE IS <i>data-name-3</i>]   [SYMBOLIC SUB-QUEUE IS <i>data-name-4</i>]   [MESSAGE DATE IS <i>data-name-5</i>]   [MESSAGE TIME IS <i>data-name-6</i>]   [SYMBOLIC SOURCE IS <i>data-name-7</i>]   [TEXT LENGHT IS <i>data-name-8</i>]   [END KEY IS <i>data-name-9</i>]   [STATUS KEY IS <i>data-name-10</i>]   [MESSAGE COUNT IS <i>data-name-11</i>] &gt;]   [<i>data-name-1 data-name-2...data-name-11</i>] &gt;]</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2</b> CD <i>cd-name</i> FOR OUTPUT [ DESTINATION COUNT IS <i>data-name-1</i> ] [TEXT LENGHT IS <i>data-name-2</i>] [STATUS KEY IS <i>data-name-3</i>] [DESTINATION TABLE OCCURS <i>integer-2</i> TIMES [INDEXED BY <i>index-name-1</i> [<i>index-name-2</i>]]</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3</b> CD <i>cd-name</i> FOR [INITIAL] I-O [&lt;&lt; [MESSAGE DATE IS <i>data-name-1</i>]   [MESSAGE TIME IS <i>data-name-2</i>]   [SYMBOLIC TERMINAL IS <i>data-name-3</i>]   [TEXT LENGHT IS <i>data-name-4</i>]   [END KEY IS <i>data-name-5</i>]   [STATUS KEY IS <i>data-name-6</i>] &gt;   [<i>data-name-1 data-name-2...data-name-6</i>] &gt;]</li> </ul>	VO	N	N			
Report Description Entry	RD <i>report-name-1</i> [IS GLOBAL] [CODE < <i>literal-1</i> >] [< CONTROL IS   CONTROL ARE > < <i>data-name-1</i> >   FINAL [ <i>data-name-1</i> ] >] [PAGE [< LIMIT IS   LIMITS ARE >] <i>integer-1</i> [< LINE   LINES >] [HEADING <i>integer-2</i> ] [FIRST DETAIL <i>integer-3</i> ] [LAST DETAIL <i>integer-4</i> ] [FOOTING <i>integer-5</i> ]	VO	N	N			

Entries and Clauses	Format	Parser	IA	CA/GDF			
				Sup:	Pairs	Relation	
Report Group Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> 01 [<i>data-name-1</i>]</li> </ul>	V	S	N			
	[LINE NUMBER IS < <i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i> >] [NEXT GROUP IS < <i>integer-3</i>   PLUS <i>integer-4</i>   NEXT PAGE >] TYPE IS < < REPORT HEADING   RH >   < PAGE HEADING   PH >   < CONTROL HEADING   CH > < <i>data-name-2</i>   FINAL >   < DETAIL   DE >   < CONTROL FOOTING   CF > < <i>data-name-3</i>   FINAL >   < PAGE FOOTING   PF >   < REPORT FOOTING   RF > > [[USAGE IS] < DISPLAY   DISPLAY-1 >.	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>level-number</i> [<i>data-name-1</i>]</li> </ul>	V	S	N			
	[LINE NUMBER IS < <i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i> >] [[USAGE IS] < DISPLAY   DISPLAY-1 >.	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>level-number</i> [<i>data-name-1</i>]</li> </ul>	V	S	N			
< PICTURE   PIC > IS <i>character-string</i> [[USAGE IS] < DISPLAY   DISPLAY-1 > [SIGN IS < LEADING   TRAILING > [SEPARATE CHARACTER]] [< JUSTIFIED   JUST > RIGHT] [BLANK WHEN < ZERO   ZEROS   ZEROES >] [LINE NUMBER IS < <i>integer-1</i> [ON NEXT PAGE]   PLUS <i>integer-2</i> >] [COLUMN NUMBER IS <i>integer-3</i> ] < SOURCE IS <i>id-1</i>   VALUE IS <i>literal-1</i>   < SUM < <i>id-2</i> >> [UPON < <i>data-name-2</i> >]   [RESET ON < <i>data-name-3</i>   FINAL-> > [GROUP INDICATE]	VO	N	N				

**1-176** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

Entries and Clauses	Format	Parser	IA	CA/GDF		
				Sup:	Pairs	Relation
Screen Description Entry	<code>level-number [ screen-name   FILLER ]</code> <code>[BLANK &lt; SCREEN   LINE &gt;]</code> <code>&lt; BELL   BEEP &gt;</code> <code>[BLINK]</code> <code>[ERASE &lt; EOL   EOS &gt;]</code> <code>&lt; HIGHLIGHT   LOWLIGHT &gt;</code> <code>[GRID]</code> <code>[LEFTLINE]</code> <code>[OVERLINE]</code> <code>[REVERSE-VIDEO]</code> <code>[UNDERLINE]</code> <code>[SIZE IS &lt; id-1   integer-1 &gt;]</code> <code>[LINE [NUMBER IS [&lt; PLUS   +   - &gt;] &lt; id-2   integer-2&gt;]]</code> <code>&lt; COLUMN   COL &gt; [NUMBER IS [&lt; PLUS   +   - &gt;] &lt; id-3   integer-3&gt;]]</code> <code>&lt; FOREGROUND-COLOR   FOREGROUND-COLOUR &gt; IS &lt; integer-4   id-4 &gt;</code> <code>&lt; BACKGROUND-COLOR   BACKGROUND-COLOUR &gt; IS &lt; integer-5   id-5 &gt;</code> <code>[CONTROL IS id-6]</code> <code>[[VALUE IS] literal-6]</code> <code>&lt; PICTURE   PIC &gt; IS character-string ]</code> <code>&lt; [FROM &lt; id-6   literal-2&gt;] [TO id-7]   USING id-8 &gt;</code> <code>[[USAGE IS] &lt; DISPLAY   DISPLAY-1 &gt;]</code> <code>[BLANK WHEN ZERO]</code> <code>&lt; JUSTIFIED   JUST &gt; RIGHT]</code> <code>[[SIGN IS] &lt; LEADING   TRAILING &gt; [SEPARATE CHARACTER]]</code> <code>&lt; AUTO   AUTO-SKIP &gt; ]</code> <code>&lt; SECURE   NO-ECHO &gt;</code> <code>&lt; REQUIRED   EMPTY-CHECK &gt;</code> <code>PROMPT [CHARACTER IS &lt; id-9   literal-3 &gt;]]</code> <code>[OCCURS integer-6 TIMES]</code> <code>&lt; FULL   LENGHT-CHECK &gt;</code> <code>[ZERO-FILL]</code>	VO	N	N		

**Procedure Division**

Entries and Clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation

Procedure Division paragraph	<ul style="list-style-type: none"> <li><b>Declarative Format</b> <i>procedure-division-header</i> [DECLARATIVES. &lt; <i>section-name</i> SECTION [segment-number] [<i>declarative-sentence</i>] [<i>paragraph-name</i> [<i>sentence</i>]] &gt; END DECLARATIVES.] [[<i>sentence</i>] [<i>paragraph-name</i> [<i>sentence</i>]] &lt; <i>section-name</i> SECTION [segment-number] [<i>sentence</i>] [<i>paragraph-name</i> [<i>sentence</i>]] &gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Non-Declarative Format</b> <i>[procedure-division-header]</i> <i>[sentence]</i> [<i>paragraph-name</i> &lt; <i>sentence</i> &gt;]</li> </ul>	V	S	S			

**Arithmetic operators**

Operation	Meaning	Format	Parser	IA	CA/GDF			
					Sup.	Pairs	Relation	
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

1-178 Supported COBOL Statements  
Supported Micro Focus COBOL statements

**Conditional Expressions**

Condition	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> IS [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   <i>class-name-1</i> &gt;</li> </ul>	V	S	S			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> &lt; <i>id-1</i>   <i>literal-1</i>   <i>arithmetic-expression-1</i>   <i>index-name-1</i> &gt; &lt; IS &lt; [NOT] &lt; GREATER THAN   LESS THAN   EQUAL TO   &lt;   &gt;   = &gt;   GREATER THAN EQUAL TO   &gt;=   LESS THAN OR EQUAL TO   &lt;= &gt;   IS UNEQUAL TO   IS &lt;&gt;   EQUALS   EXCEEDS &gt; &lt; <i>id-2</i>   <i>literal-2</i>   <i>arithmetic-expression-2</i>   <i>index-name-2</i> &gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	cond
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> operand-1 [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>simple-condition-1</i></li> </ul>	V	S	S			
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;&lt;AND   OR&gt; <i>condition-1</i>&gt;</li> </ul>	V	S	S			
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;&lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i>&gt;</li> </ul>	V	S	S			
Pointer condition	<ul style="list-style-type: none"> <li><b>Format</b> &lt; ADDRESS OF <i>id-1</i>   <i>id-2</i>   NULL &gt; &lt; IS [NOT] EQUAL TO   IS [NOT] =   IS UNEQUAL TO   IS &lt;&gt;   EQUALS &gt; &lt; ADDRESS OF <i>id-3</i>   <i>id-4</i>   NULL &gt;</li> </ul>	V	S	N			
Procedure-Pointer Condition	<ul style="list-style-type: none"> <li><b>Format</b> &lt; <i>id-1</i>   NULL &gt; &lt; IS [NOT] EQUAL TO   IS [NOT] =   IS UNEQUAL TO   IS &lt;&gt;   EQUALS &gt; &lt; <i>id-2</i>   NULL &gt;</li> </ul>	N	N	N			



Condition	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
Object Relation Condition	<ul style="list-style-type: none"> <li><b>Format</b> &lt; <i>object-id-1</i>   NULL &gt; &lt; IS [NOT] EQUAL TO   IS [NOT] =   IS UNEQUAL TO   IS &lt;&gt;   EQUALS &gt; &lt; <i>object-id-2</i>   NULL &gt;</li> </ul>	N	N	N			

**Statements**

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format 1</b> PROCEDURE DIVISION [<i>mnemonic-name</i>] [&lt; USING   CHAINING &gt; &lt; [BY REFERENCE] &lt; <i>data-name-1</i> &gt;   BY VALUE &lt;<i>data-name-2</i>&gt; &gt;]</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	<i>extern.using</i>		
					<i>data-name-1</i>	<i>port</i>	<i>extern.using</i>		
	<ul style="list-style-type: none"> <li><b>Format 2</b> PROCEDURE DIVISION [<i>mnemonic-name</i>] [&lt; USING   CHAINING &gt; &lt; [BY REFERENCE] &lt; <i>data-name-1</i> [DELIMITED [BY SIZE]]   <i>typedef-name-1</i>   ANY &gt;   BY VALUE &lt; <i>data-name-2</i>   <i>typedef-name-2</i>   ANY &gt; &gt; [REPEATED [<i>integer-1</i> TO <i>integer-2</i>]]] [&lt; GIVING   RETURNING &gt; &lt; <i>data-name-3</i>   <i>typedef-name-3</i> &gt; ]</li> </ul>	N	N	N					

**1-180** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
ACCEPT	<p><b>Format 1</b></p> <p>ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>   <i>function-name</i>&gt;] &lt; [ON EXCEPTION <i>imperative-statement-1</i>]   [NOT ON EXCEPTION <i>imperative-statement-2</i>]   [END-ACCEPT] &gt;</p>	V	S	S	<i>port</i>	<i>id</i>	<i>extern.screen</i>	
	<p><b>Format 2</b></p> <p>ACCEPT <i>id</i> FROM &lt; DATE   DAY   DAY-OF-WEEK   TIME&gt; [END-ACCEPT]</p>	V	S	S	<i>port</i>	<i>id</i>	<i>extern.screen</i>	
	<p><b>Format 3</b></p> <p>ACCEPT <i>id</i> FROM &lt; LINE NUMBER   USER NAME   ESCAPE KEY   EXCEPTION STATUS &gt; [END-ACCEPT]</p>	V	S	S	<i>port</i>	<i>id</i>	<i>extern.screen</i>	
	<p><b>Format 4</b></p> <p>ACCEPT <i>screen-name</i> [ AT &lt; <i>id-4</i>   <i>Integer-3</i> &gt; ] [ON &lt; EXCEPTION   ESCAPE &gt; <i>imperative-statement-1</i>] [NOT ON &lt; EXCEPTION   ESCAPE &gt; <i>imperative-statement-2</i> ] [END-ACCEPT]</p>	V	S	S	<i>port</i>	<i>screen-name</i>	<i>extern.screen</i>	
	<p><b>Format 5</b></p> <p>ACCEPT <i>id-1</i> [&lt; AT [LINE NUMBER &lt; <i>id-2</i>   <i>integer-1</i> &gt;] [&lt; COLUMN   COL &gt; NUMBER &lt; <i>id-3</i>   <i>integer-2</i> &gt;]   AT &lt; <i>id-4</i>   <i>integer-3</i> &gt; ] [FROM CRT] [MODE IS BLOCK] [WITH &lt; &lt; AUTO   AUTO-SKIP &gt;   &lt; BELL   BEEP &gt;   BLINK   &lt; FULL   LENGHT-CHECK &gt;   GRID   [&lt; HIGHLIGHT   LOWLIGHT &gt; ]   &lt; LEFTLINE   OVERLINE   PROMPT [CHARACTER IS [<i>literal-1</i>]] &gt;   &lt; REQUIRED   EMPTY-CHECK &gt;   REVERSE-VIDEO   &lt;SECURE   NO-ECHO &gt;   SIZE IS &lt; <i>id-6</i>   <i>integer-4</i> &gt;   UNDERLINE   &lt; FOREGROUND-COLOR   FOREGROUND-COLOUR &gt; IS <i>integer-5</i>   &lt; BACKGROUND-COLOR   BACKGROUND-COLOUR &gt; IS <i>integer-6</i>   CONTROL IS &lt; <i>id-7</i>   <i>literal-2</i> &gt;   &lt; TIME-OUT   TIMEOUT &gt; AFTER &lt; <i>integer-7</i>   <i>id-8</i> &gt;   &lt; LEFT-JUSTIFY   RIGHT-JUSTIFY   SPACE-FILL   TRAILING-SIGN   UPDATE   UPPER   LOWER   ZERO-FILL &gt; &gt; ] [ON &lt; EXCEPTION   ESCAPE &gt; <i>imperative-statement-1</i>] [NOT ON &lt; EXCEPTION   ESCAPE &gt; <i>imperative-statement-2</i> ] [END-ACCEPT]</p>	V	S	S	<i>port</i>	<i>id-1</i>	<i>extern.screen</i>	
	<p><b>Format 6</b></p> <p>ACCEPT <i>cd-name</i> MESSAGE COUNT</p>	V	S	S	<i>port</i>	<i>cd-name</i>	<i>extern.screen</i>	

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;id-1   literal&gt; TO &lt;id-2 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2</b> ADD &lt;id-1   literal-1&gt; TO &lt;id-2   literal-1&gt; GIVING &lt;id-3 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-3	comp+		
				id-2	id-3	comp+			
				id1	id2				
	<ul style="list-style-type: none"> <li><b>Format 3</b> ADD &lt;CORRESPONDING   CORR&gt; id-1 TO id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER procedure-name-1 TO [PROCEED TO] procedure-name-2 [procedure-name-3 TO [PROCEED TO] procedure-name-4]</li> </ul>	V	S	S					
CALL	<ul style="list-style-type: none"> <li><b>Format</b> CALL &lt;id-1   literal-1   procedure-pointer-1&gt; [USING [&lt; [BY REFERENCE] &lt; id-2   literal-2   ADDRESS OF id-3   OMITTED &gt;   BY CONTENT &lt; id-4   literal-3   LENGH OF id-5&gt;   BY VALUE &lt; integer-1 [SIZE IS integer-2]   LENGH OF id-7&gt;&gt;]] [&lt; GIVING   RETURNING &gt; &lt; INTO id-8   ADDRESS OF id-9 &gt;] &lt; [ON EXCEPTION imperative-stmt-1]   [NOT ON EXCEPTION imperative-stmt-2]   ON OVERFLOW imperative-stmt-1 &gt; [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
					port	id-2, id-5	extern.call		
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S					

1-182 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	CA/GDF			Repository	
				Sup*	Pairs	Relation	Entities	Relations
CHAIN	<ul style="list-style-type: none"> <li><b>Format</b> CHAIN &lt;id-1   literal-1&gt; CALL &lt;id-1   literal-1   procedure-pointer-1&gt; [USING [&lt; [BY REFERENCE] &lt; id-2   literal-2   ADDRESS OF id-3   OMITTED &gt;   BY CONTENT &lt; id-4   literal-3   LENGH OF id-5&gt;   BY VALUE &lt; integer-1 [SIZE IS integer-2]   LENGH OF id-7&gt;&gt;]] [END-CHAIN]</li> </ul>	V	S	N				
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1</b> CLOSE file-name-1</li> </ul>	V	S	S				
	<< REEL   UNIT > WITH LOCK   < REEL   UNIT > [< FOR REMOVAL   WITH NO REWIND>   WITH < NO REWIND   LOCK >]	VO	N	N				
	<ul style="list-style-type: none"> <li><b>Format 2</b> CLOSE filename-1</li> </ul>	V	S	S				
	[<REEL   UNIT> [[FOR] REMOVAL   WITH NO REWIND]   [WITH] <NO REWIND   LOCK>	VO	N	N				
COMMIT	<ul style="list-style-type: none"> <li><b>Format</b> COMMIT</li> </ul>	V	S	S				
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE &lt;id-1 [ROUNDED]&gt; &lt; EQUAL   = &gt; arithmetic-expression [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-COMPUTE]</li> </ul>	V	S	S	ids of the arithmetic-expr	id-1	comp+ or comp* (according to the operation performed)	
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S				
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY &lt;text-name   external-file-name-literal &gt; [&lt; OF   IN &gt; &lt;library-name   library-name-literal&gt;] [SUPPRESS] [REPLACING &lt;&lt; ==pseudo-text-1==   id-1   literal-1   word-1 &gt; BY &lt; ==pseudo-text-2==   id-2   literal-2   word-2 &gt;&gt;]</li> </ul>	V	S	S			refers to text-name (Copy-book)	Cobol Includes Copybook

Statements	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> RECORD [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)  <i>Program Deletes From File</i>
DELETE FILE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE FILE &lt; <i>file-name</i> &gt;</li> </ul>	V	S	S				
DISABLE	<ul style="list-style-type: none"> <li><b>Format</b> DISABLE &lt; INPUT [TERMINAL]   I-O TERMINAL   OUTPUT &gt; <i>cd-name</i> WITH KEY &lt; <i>id-1</i>   <i>literal-1</i> &gt;</li> </ul>	V	S	N				
DISPLAY	<ul style="list-style-type: none"> <li><b>Format 1</b> DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [&lt; <i>id-2</i>   <i>literal-2</i> &gt;] [UPON &lt;<i>mnemonic-name-1</i>   <i>function-name-1</i>&gt;] [WITH NO ADVANCING] &lt; [ON EXCEPTION <i>imperative-stmt-1</i>]   [NOT ON EXCEPTION <i>imperative-stmt-2</i>]   [END-DISPLAY] &gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	<i>extern.screen</i>	
	<ul style="list-style-type: none"> <li><b>Format 2</b> DISPLAY <i>screen-name</i> [ &lt; AT [LINE NUMBER &lt; <i>id-2</i>   <i>integer-1</i> &gt;] [&lt;COLUMN   COL&gt; NUMBER &lt; <i>id-3</i>   <i>integer-2</i>&gt;]   AT &lt; <i>id-4</i>   <i>integer-3</i> &gt; &gt;] [END-DISPLAY]</li> </ul>	V	S	S	<i>screen-name</i> <i>id-2</i> <i>id-3</i>	<i>port</i>	<i>extern.screen</i>	
	<ul style="list-style-type: none"> <li><b>Format 3</b> DISPLAY &lt;&lt; <i>id-1</i>   <i>literal-1</i> &gt; &gt; AT [LINE NUMBER &lt; <i>id-2</i>   <i>integer-1</i> &gt;] [&lt; COLUMN   COL &gt; NUMBER &lt; <i>id-3</i>   <i>integer-2</i> &gt;]   AT &lt; <i>id-4</i>   <i>integer-3</i> &gt; &gt;] [ UPON &lt; CRT   CRT-UNDER &gt;] [MODE IS BLOCK] [FROM CRT] [MODE IS BLOCK] [WITH &lt; &lt; BELL   BEEP &gt;   BLINK   GRID   HIGHLIGHT   LOW-LIGHT   LEFTLINE   OVERLINE   REVERSE-VIDEO   SIZE IS &lt; <i>id-5</i>   <i>integer-4</i> &gt;   UNDERLINE   &lt; FOREGROUND-COLOR   FOREGROUND-COLOUR &gt; IS <i>integer-5</i>   &lt; BACKGROUND-COLOR   BACKGROUND-COLOUR &gt; IS <i>integer-6</i>   CONTROL IS &lt; <i>id-6</i>   <i>literal-2</i> &gt;   BLANK &lt; SCREEN   LINE &gt; &gt; &gt;] [END-DISPLAY]</li> </ul>	V	S	S	<i>id-1</i> <i>id-2</i> <i>id-3</i> <i>id-5</i> <i>id-6</i>	<i>port</i>	<i>extern.screen</i>	

1-184 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-1	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING &lt;id-3 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-3	comp*		
					id-1	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING &lt;id-3 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
				id-2	id-3	comp*			
<ul style="list-style-type: none"> <li><b>Format 4</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING &lt;id-3 [ROUNDED]&gt; REMAINDER id-4 [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*			
				id-2	id-3	comp*			
				id-1	id-4	comp*			
				id-2	id-4	comp*			
<ul style="list-style-type: none"> <li><b>Format 5</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING &lt;id-3 [ROUNDED]&gt; REMAINDER id-4 [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*			
				id-2	id-3	comp*			
				id-1	id-4	comp*			
				id-2	id-4	comp*			
ENABLE	<ul style="list-style-type: none"> <li><b>Format</b> ENABLE &lt; INPUT [TERMINAL]   I-O TERMINAL   OUTPUT&gt; cd-name WITH KEY &lt; id-1   literal-1&gt;</li> </ul>	V	S	N					
END	<ul style="list-style-type: none"> <li><b>Format:</b> END</li> </ul>	V	S	S					

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
ENTER	<ul style="list-style-type: none"> <li><b>Format:</b> ENTER <i>language-name</i> [<i>routine-name</i>]</li> </ul>	V	S	N					
ENTRY	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTRY <i>literal-1</i> [USING &lt; [BY REFERENCE] &lt; <i>data-name-1</i> &gt;   BY VALUE &lt; <i>data-name-2</i> &gt; ]</li> </ul>	V	S	S				defines <i>literal-1</i> (Program Entry Point)	Program Has Program Entry Point
	<ul style="list-style-type: none"> <li><b>Format 2</b> ENTRY <i>literal-1</i> [<i>mnemonic-name</i>] [USING &lt; [BY REFERENCE] &lt; <i>data-name-1</i> [DELIMITED [BY SIZE]]   <i>typedef-name-1</i>   ANY &gt;   BY VALUE &lt; <i>data-name-2</i>   <i>typedef-name-2</i>   ANY &gt; &gt; [REPEATED [<i>integer-1</i> TO <i>integer-2</i>]]] [&lt; GIVING   RETURNING &gt; &lt; <i>data-name-3</i>   <i>typedef-name-3</i> &gt; ]</li> </ul>	VO	N	N					
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;<i>id-1</i>   <i>literal-1</i>   <i>expr-1</i>   TRUE   FALSE&gt; [ALSO &lt;<i>id-2</i>   <i>literal-2</i>   <i>expr-2</i>   TRUE   FALSE&gt;] &lt;&lt; WHEN &lt; ANY   <i>condition-1</i>   TRUE   FALSE   [NOT] &lt;&lt;<i>id-3</i>   <i>literal-5</i>   <i>arithmetic-expression-3</i> &gt; [&lt; THROUGH   THRU &gt; &lt; <i>id-6</i>   <i>literal-6</i>   <i>arithmetic-expression-4</i>&gt;] &gt;   <i>partial-expression-2</i> &gt;] <i>imperative statement-1</i> [WHEN OTHER <i>imperative-statement-2</i>] [END-EVALUATE]</li> </ul>	V	S	S	<i>id-1, expr-1</i>	<i>condition-1</i>	<i>cond</i>		
EXAMINE	<ul style="list-style-type: none"> <li><b>Format 1</b> EXAMINE <i>id</i> REPLACING &lt; ALL   LEADING   FIRST   UNTIL FIRST&gt; <i>literal-1</i> BY <i>literal-2</i></li> </ul>	V	S	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b> EXAMINE <i>id</i> TALLYNG &lt; ALL   LEADING   UNTIL FIRST&gt; <i>literal-1</i> [REPLACING BY] <i>literal-2</i></li> </ul>	V	S	N					
EXECUTE	<ul style="list-style-type: none"> <li><b>Format</b> EXEC[UTE] <i>text-name text-data</i> END-EXEC</li> </ul>	V	S	S					
EXHIBIT	<ul style="list-style-type: none"> <li><b>Format 1</b> EXHIBIT &lt; NAMED   CHANGED NAMED   CHANGED &gt; &lt; <i>id-1</i>   <i>literal-1</i> &gt;</li> </ul>	V	S	N					

**1-186** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> EXIT.</li> </ul>	V	S	S					
EXIT ME-THOD	<ul style="list-style-type: none"> <li><b>Format</b> EXIT METHOD.</li> </ul>	V	S	N					
EXIT PARAGRAPH	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PARAGRAPH</li> </ul>	V	S	S					
EXIT PERFORM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PERFORM [CYCLE]</li> </ul>	V	S	S					
EXIT SECTION	<ul style="list-style-type: none"> <li><b>Format</b> EXIT SECTION</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM [&lt; GIVING   RETURNING &gt; &lt; [ADDRESS OF] <i>id-1</i>   <i>integer-1</i> &gt;].</li> </ul>	V	S	S					
GENERATE	<ul style="list-style-type: none"> <li><b>Format</b> GENERATE &lt; <i>data-name-1</i>   <i>report-name-1</i> &gt;</li> </ul>	VO	N	N					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b> GOBACK [&lt; GIVING   RETURNING &gt; &lt; [ADDRESS OF] <i>id-1</i>   <i>integer-1</i> &gt;]</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1</b> GO TO [<i>procedure-name-1</i>]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> GO TO <i>procedure-name-1</i> [<i>procedure-name-2</i>] DEPENDING ON <i>id</i></li> </ul>	V	S	S					



Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
IF	<ul style="list-style-type: none"> <li><b>Format 1</b> IF <i>condition-1</i> THEN &lt;<i>stmt-1</i>   NEXT SENTENCE&gt; [ ELSE &lt;<i>stmt-2</i>   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> IF <i>constant-name-1</i> [NOT] &lt;&lt;   &gt;   = &gt; <i>literal-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> IF <i>constant-name-2</i> [NOT] DENIED</li> </ul>	N	N	N					
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED   NATIONAL   NATIONAL-EDITED   DBCS &gt; DATA BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S					
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING &lt;<i>id-2</i> FOR &lt; CHARACTERS [&lt;BEFORE   AFTER&gt; INITIAL &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; INITIAL &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; INITIAL &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; INITIAL &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>			
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;] REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>			

1-188 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING <i>&lt;id-6   literal-4&gt;</i> TO <i>&lt;id-7   literal-5&gt;</i> [&lt;BEFORE   AFTER&gt;] INITIAL <i>&lt;id-4   literal-2&gt;</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>		
INVOKE	<ul style="list-style-type: none"> <li><b>Format</b> INVOKE <i>&lt;object-id-1   id-1 AS &lt;template-1   OBJECT&gt;&gt;</i> <i>&lt;literal-1   id-2&gt;</i> [USING <i>&lt;[BY REFERENCE] &lt;[ADDRESS OF] id-4   id-3   literal-2&gt;   [BY CONTENT] &lt; id-5] LENGTH OF id-6   literal-3 &gt;   [BY VALUE] &lt;id-7   LENGTH OF id-6   integer-1 [SIZE IS integer-2]&gt;&gt;</i>] [&lt; RETURNING   GIVING &gt; <i>&lt;INTO id-9   ADDRESS OF id-10&gt;</i>] [[ON] EXCEPTION <i>imperative-stmt-1</i>] [NOT [ON] EXCEPTION <i>imperative-stmt-2</i>] [END-INVOKE]</li> </ul>	N	N	N				
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [&lt;ON&gt;] [&lt;ASCENDING   DESCENDING&gt;] KEY <i>&lt;data-name-1&gt;</i> [COLLATING SEQUENCE IS <i>alphabet-name-1</i>] USING <i>file-name-2</i> <i>&lt;file-name-3&gt;</i> [&lt;OUTPUT PROCEDURE IS <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>&lt;file-name-4&gt;</i> &gt;]</li> </ul>	V	S	N				
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE <i>&lt;id-1   literal-1&gt;</i> TO <i>&lt;id-2&gt;</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	<i>move or cast (depending on data types)</i>	
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE [&lt;CORRESPONDING   CORR&gt;] <i>id-1</i> TO <i>&lt;id-2&gt;</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	<i>move or cast (depending on data types)</i>	

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b>                      MULTIPLY &lt;id-1   literal-1&gt; BY &lt;id-2 [ROUNDED]&gt;                      [ON SIZE ERROR <i>imperative-stmt-1</i>]                      [NOT ON SIZE ERROR <i>imperative-stmt-2</i>]                      [END-MULTIPLY]</li> </ul>	V	S	S	id-1	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2</b>                      MULTIPLY &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt;                      GIVING &lt;id-3 [ROUNDED]&gt;                      [ON SIZE ERROR <i>imperative-stmt-1</i>]                      [NOT ON SIZE ERROR <i>imperative-stmt-2</i>]                      [END-MULTIPLY]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-2	comp*		
					id-2	id-3	comp*		
NEXT SENTENCE	<ul style="list-style-type: none"> <li><b>Format</b>                      NEXT SENTENCE</li> </ul>	V	S	S					
NOTE	<ul style="list-style-type: none"> <li><b>Format</b>                      NOTE <i>character-string</i></li> </ul>	V	S	S					
ON	<ul style="list-style-type: none"> <li><b>Format</b>                      ON &lt; literal-1   id-1 &gt; [AND EVERY &lt; literal-2   id-2 &gt;] [UNTIL &lt; literal-3   id-3 &gt;] &lt; imperative-statement-1   NEXT SENTENCE &gt; [                      ELSE   OTHERWISE &gt; &lt; imperative-statement-2   NEXT SENTENCE &gt;]</li> </ul>	V	S	N					
OPEN	<ul style="list-style-type: none"> <li><b>Format</b>                      OPEN &lt;&lt; INPUT &lt;file-name-1 [WITH LOCK]&gt;   OUTPUT &lt;file-name-2 [WITH LOCK]   I-O &lt;file-name-3 [WITH LOCK]&gt;   EXTEND &lt; file-name-4 [WITH LOCK]&gt;&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li>OPEN &lt;&lt; INPUT &lt;file-name-1 &gt;   OUTPUT &lt;file-name-2   I-O &lt;file-name-3 &gt;   EXTEND &lt; file-name-4 &gt;&gt;</li> </ul>	VO	N	N					

1-190 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [<i>imperative-stmt-1</i> [END-PERFORM]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] &lt;<i>id-1</i>   <i>integer-1</i>&gt; TIMES [<i>imperative-stmt-1</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [WITH TEST &lt;BEFORE   AFTER&gt;] UNTIL &lt;<i>condition-1</i>   EXIT &gt; [<i>imperative-stmt-1</i> [END-PERFORM]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [WITH TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt; FROM &lt;<i>id-3</i>   <i>index-name-2</i>   <i>literal-1</i>&gt; BY &lt;<i>id-4</i>   <i>literal-2</i>&gt; UNTIL &lt;<i>condition-1</i>&gt; [AFTER &lt;<i>id-5</i>   <i>index-name-3</i>&gt; FROM &lt;<i>id-6</i>   <i>index-name-4</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-4</i>&gt; UNTIL <i>condition-2</i>] [<i>imperative-stmt-1</i> [END-PERFORM]]</li> </ul>	V	S	S					
PURGE	<ul style="list-style-type: none"> <li><b>Format</b> PURGE <i>cd-name</i></li> </ul>	V	S	N					
READ	<ul style="list-style-type: none"> <li><b>Format 1</b> READ <i>file-name</i> [NEXT] RECORD [INTO <i>id</i>] [WITH &lt; [&lt; KEPT   NO &gt;] LOCK   WAIT&gt;] [AT END <i>imperative-stmt-1</i>] [NOT AT END <i>imperative-stmt-2</i>] [END-READ]</li> </ul>	V	S	S	<i>id</i>	<i>file-rec (corresponding to file-name)</i>	<i>move</i>	refer to File (see FILE-CONTROL paragraph)	<i>Program Reads File</i>
					<i>port</i>	<i>file-rec (corresponding to file-name)</i>	<i>extern.file</i>		
	<ul style="list-style-type: none"> <li><b>Format 2</b> READ <i>file-name</i> [NEXT] RECORD [INTO <i>id</i>] [AT END <i>imperative-stmt-1</i>] [NOT AT END <i>imperative-stmt-2</i>] [END-READ]</li> </ul>	V	S	S					

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 3</b>            READ <i>file-name</i> [&lt; NEXT   PREVIOUS &gt;] RECORD [INTO <i>id</i>]            [WITH &lt; [&lt; KEPT   NO   IGNORE &gt;] LOCK   WAIT&gt;]            [AT END <i>imperative-stmt-1</i>]            [NOT AT END <i>imperative-stmt-2</i>]            [END-READ]         </li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b>            READ <i>file-name</i> RECORD [INTO <i>id-1</i>] [WITH &lt; [&lt; KEPT   NO &gt;]            LOCK   WAIT&gt;]            [INVALID KEY <i>imperative-stmt-3</i>]            [NOT INVALID KEY <i>imperative-stmt-4</i>]            [END-READ]         </li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5</b>            READ <i>file-name</i> RECORD [INTO <i>id-1</i>] [WITH &lt; [&lt; KEPT   NO              IGNORE &gt;] LOCK   WAIT&gt;]            [KEY IS &lt; <i>data-name</i>   <i>split-key-name</i> &gt;]            [INVALID KEY <i>imperative-stmt-3</i>]            [NOT INVALID KEY <i>imperative-stmt-4</i>]            [END-READ]         </li> </ul>	V	S	S					
READY TRACE	<b>Format</b> READY TRACE	VO	N	N					
RECEIVE	<b>Format</b> RECEIVE <i>cd-name</i> <MESSAGE   SEGMENT>	V	S	N					
	INTO <i>id-1</i> [NO DATA <i>imperative stmt-1</i> ] [WITH DATA <i>imperative stmt-2</i> ] [END-RECEIVE]	VO	N	N					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b>            RELEASE <i>record-name</i> [FROM <i>id</i>]         </li> </ul>	V	S	S					
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b>            REPLACE &lt;==<i>pseudo-text</i>==BY==<i>pseudo-text-2</i>==&gt;         </li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b>            REPLACE OFF         </li> </ul>	V	S	S					

1-192 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
RESET TRACE	<ul style="list-style-type: none"> <li><b>Format</b> RESET TRACE</li> </ul>	VO	N	N					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b> RETURN <i>file-name</i> RECORD [INTO <i>id-1</i>] AT END <i>imperative-stmt-1</i> [NOT AT END <i>imperative-stmt-2</i>] [END-RETURN]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format 1</b> REWRITE <i>record-name</i> [FROM <i>id</i>] [END-REWRITE]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Updates File
	<ul style="list-style-type: none"> <li><b>Format 2</b> REWRITE <i>record-name</i> [FROM <i>id</i>] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-REWRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	<i>move</i>		
					<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	<i>port</i>	<i>extern.file</i>		
ROLLBACK	<ul style="list-style-type: none"> <li><b>Format</b> ROLLBACK</li> </ul>	VO	N	N					
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [AT END <i>imperative-stmt-1</i>] &lt;WHEN <i>condition-1</i> [&lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt;]&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> SEARCH ALL <i>id-1</i> [AT END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> IS &lt; EQUAL TO   = &gt; &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> IS &lt; EQUAL TO   = &gt; &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] [&lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt;] [END-SEARCH]</li> </ul>	V	S	S					

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SEND	<ul style="list-style-type: none"> <li><b>Format 1</b> SEND <i>cd-name</i> FROM <i>id-1</i></li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b> SEND <i>cd-name</i> [FROM <i>id-1</i>] &lt; WITH &lt; <i>id-2</i>   ESI   EMI   EGI &gt; [&lt; BEFORE   AFTER &gt; ADVANCING &lt; &lt;<i>id-1</i>   <i>integer</i>&gt; [LINE   LINES]   &lt; <i>mnemonic name</i>   PAGE &gt; &gt;]</li> </ul>	VO	N	N					
SERVICE	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE &lt; LABEL   RELOAD <i>id</i> &gt;</li> </ul>	VO	N	N					
SET	<ul style="list-style-type: none"> <li><b>Format 1</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-6</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-6</i> <i>integer-2</i>	<i>move</i>		
	<ul style="list-style-type: none"> <li><b>Format 2</b> SET &lt;<i>pointer-name-3</i>&gt; &lt;UP   DOWN &gt; BY &lt;<i>id-3</i>   <i>integer-1</i>   LENGTH OF <i>id-4</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> SET &lt;&lt;<i>mnemonic-name-1</i>&gt;TO &lt;ON   OFF&gt;&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> SET &lt;&lt;<i>condition-name-1</i>&gt; TO &lt; TRUE   FALSE &gt;&gt;</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	<i>move</i>		
	<ul style="list-style-type: none"> <li><b>Format 5</b> SET &lt;<i>pointer-name-1</i>   ADDRESS OF <i>id-5</i>&gt; TO &lt;<i>pointer-name-2</i>   ADDRESS OF <i>id-7</i>   NULL   NULLS&gt;</li> </ul>	V	S	S	<i>pointer-name-1</i>	<i>pointer-name-2</i>	<i>move</i>		
					<i>pointer-name-1</i>	<i>id-7</i>	<i>comp</i>		
	<ul style="list-style-type: none"> <li><b>Format 6</b> SET &lt;<i>procedure-pointer-name-1</i>&gt; TO &lt;<i>procedure-pointer-name-2</i>   ENTRY &lt;<i>id-8</i>   <i>literal-1</i>&gt;   NULL   NULLS &gt;</li> </ul>	V	S	N					
<ul style="list-style-type: none"> <li><b>Format 7</b> SET &lt;<i>index-name-3</i>&gt; &lt; UP   DOWN &gt; BY &lt;<i>id-7</i>   <i>integer-3</i>&gt;</li> </ul>	V	S	S						

**1-194** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

Statements	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 8</b>            SET &lt;object-id-1&gt;TO &lt;object-id-2   NULL&gt;</li> </ul>	V	S	N					
SORT	<ul style="list-style-type: none"> <li><b>Format 1</b>            SORT <i>file-name-1</i> ON &lt;&lt;ASCENDING   DESCENDING&gt; KEY            &lt;<i>data-name-1</i>&gt;&gt;            [WITH DUPLICATES IN ORDER]            &lt;USING &lt;<i>file-name-2</i>&gt;   INPUT PROCEDURE IS <i>procedure-name-1</i>            [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]&gt;            &lt;GIVING &lt;<i>file-name-3</i>&gt;   OUTPUT PROCEDURE IS <i>procedure-name-3</i>            [&lt;THROUGH   THRU&gt; <i>procedure-name-4</i>]&gt;</li> </ul>	V	S	S					
	SORT <i>file-name-1</i> ON <<ASCENDING   DESCENDING> KEY < <i>data-name-1</i> >> [WITH DUPLICATES IN ORDER] [COLLATING SEQUENCE IS <i>alphabet-name</i> ] <USING < <i>file-name-2</i> >   INPUT PROCEDURE IS <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> ]> <GIVING < <i>file-name-3</i> >   OUTPUT PROCEDURE IS <i>procedure-name-3</i> [<THROUGH   THRU> <i>procedure-name-4</i> ]>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b>            SORT <i>data-name-2</i> ON &lt;&lt;ASCENDING   DESCENDING&gt; KEY            &lt;<i>data-name-1</i>&gt;&gt;            [WITH DUPLICATES IN ORDER]</li> </ul>	V	S	S					
	SORT <i>data-name-2</i> ON <<ASCENDING   DESCENDING> KEY < <i>data-name-1</i> >> [WITH DUPLICATES IN ORDER] [COLLATING SEQUENCE IS <i>alphabet-name</i> ]	VO	N	N					



Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
START	<ul style="list-style-type: none"> <li><b>Format 1</b> START <i>file-name-1</i> [KEY IS &lt;EQUAL TO   =   LESS THAN   &lt;   GREATER THAN   &gt;   NOT LESS THAN   NOT &lt;   NOT GREATER THAN   NOT &gt;   LESS THAN OR EQUAL TO   &lt;=   GREATER THAN OR EQUAL TO   &gt;= &gt; &lt; <i>data-name-1</i>   <i>split-key-name-1</i> &gt; [WITH SIZE &lt; <i>id-1</i>   <i>literal-1</i>&gt;] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> START <i>file-name-1</i> [KEY IS &lt;EQUAL TO   =   LESS THAN   &lt;   GREATER THAN   &gt;   NOT LESS THAN   NOT &lt;   NOT GREATER THAN   NOT &gt;   LESS THAN OR EQUAL TO   &lt;=   GREATER THAN OR EQUAL TO   &gt;= &gt; <i>data-name-1</i>] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID KEY <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format 1</b> STOP &lt;RUN   <i>literal-1</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> STOP RUN [&lt; GIVING   RETURNING &gt; &lt; [ADDRESS OF] <i>id-1</i>   <i>integer-1</i> [SIZE IS <i>integer-2</i>&gt;]</li> </ul>	VO	N	N					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;&lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt;&gt; INTO <i>id-3</i> [WITH POINTER <i>id-4</i>] [ON OVERFLOW <i>imperative-stmt-1</i>] [NOT ON OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	<i>comp</i>		
					<i>id-2</i>	<i>id-3</i>	<i>comp</i>		

1-196 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b> SUBTRACT &lt;id-1   literal-1&gt; FROM &lt;id-2 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2</b> SUBTRACT &lt;id-1   literal-1&gt; FROM &lt;id-2   literal-2&gt; GIVING &lt;id-3 [ROUNDED]&gt; [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]</li> </ul>	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b> SUBTRACT &lt;CORRESPONDING   CORR&gt; id-1 FROM id-2 [ROUNDED] [ON SIZE ERROR imperative-stmt-1] [NOT ON SIZE ERROR imperative-stmt-2] [END-SUBTRACT]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
SUPPRESS	<ul style="list-style-type: none"> <li><b>Format</b> SUPPRESS PRINTING</li> </ul>	VO	N	N					
TERMINATE	<ul style="list-style-type: none"> <li><b>Format</b> TERMINATE &lt;report-name-1&gt;</li> </ul>	VO	N	N					
TRANSFORM	<ul style="list-style-type: none"> <li><b>Format</b> TRANSFORM id-3 CHARACTERS FROM &lt; figurative-constant-1   nonnumeric literal-1   id-1 &gt; TO &lt; figurative-constant-2   nonnumeric-literal-2   id-2 &gt;</li> </ul>	VO	N	N					
UNLOCK	<ul style="list-style-type: none"> <li><b>Format</b> UNLOCK file-name &lt;RECORD   RECORDS&gt;</li> </ul>	VO	N	N					

Statements	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
					id-1	id-4			
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b> UNSTRING <i>id-1</i> [DELIMITED BY [ALL] &lt;<i>id-2</i>   <i>literal-1</i>&gt; [OR [ALL] &lt;<i>id-3</i>   <i>literal-2</i>&gt;]] INTO &lt;<i>id-4</i> [DELIMITER IN <i>id-5</i>] [COUNT IN <i>id-6</i>&gt; [WITH POINTER <i>id-7</i>] [TALLYING IN <i>id-8</i>] [ON OVERFLOW <i>imperative-stmt-1</i>] [NOT ON OVERFLOW <i>imperative-stmt-2</i>] [END-UNSTRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-4</i>	<i>comp</i>		
USE	<ul style="list-style-type: none"> <li><b>Format 1</b> USE [GLOBAL] AFTER STANDARD &lt; EXCEPTION   ERROR &gt; PROCEDURE ON &lt;[<i>file-name-1</i>]   INPUT   OUTPUT   I-O   EXTEND &gt;</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2</b> USE [GLOBAL] BEFORE REPORTING <i>id-1</i></li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3</b> USE FOR DEBUGGING ON &lt; <i>cd-name-1</i>   [ALL REFERENCES OF] <i>id-1</i>   <i>file-name-1</i>   <i>procedure-name-1</i>   ALL PROCEDURES]&gt;</li> </ul>	VO	N	N					
WRITE	<ul style="list-style-type: none"> <li><b>Format 1</b> WRITE <i>record-name</i> [&lt; FROM <i>id-1</i>   FROM <i>literal</i> &gt;] [INVALID KEY <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>&gt; [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name</i>	<i>move</i>	refer to File (see FILE-CONTROL paragraph)	<i>Program Inserts into File</i>
	<ul style="list-style-type: none"> <li><b>Format 2</b> WRITE <i>record-name</i> [&lt; FROM <i>id-1</i>   FROM <i>literal</i> &gt;] [&lt; BEFORE   AFTER &gt; ADVANCING &lt; &lt; <i>id-2</i>   <i>integer</i> &gt; [LINE   LINES]   <i>mnemonic-name</i>   PAGE   TAB   FRMFEED &gt;] [AT &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-1</i>] [NOT AT &lt;END- OF-PAGE   EOP&gt; <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	N	N	N					

**1-198** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

***Intrinsic functions***

Functions	Parser	IA	CA/GDF			
			Sup:	Pairs		Relation
ABS	V	S	S	operands	result	comp
ACOS	V	S	S	operands	result	comp
ANNUITY	V	S	S	operands	result	comp
ASIN	V	S	S	operands	result	comp
ATAN	V	S	S	operands	result	comp
CHAR	V	S	S	operands	result	comp
CHAR-NATIONAL	V	S	S	operands	result	comp
COS	V	S	S	operands	result	comp
CURRENT-DATE	V	S	S	operands	result	comp
DATE-OF-INTEGER	V	S	S	operands	result	comp
DAY-OF-INTEGER	V	S	S	operands	result	comp
DAY-TO-YYYYDDD	V	S	S	operands	result	comp
DISPLAY-OF	V	S	S	operands	result	comp
E	V	S	S	operands	result	comp
EXP	V	S	S	operands	result	comp
EXP10	V	S	S	operands	result	comp
FACTORIAL	V	S	S	operands	result	comp
FRACTION-PART	V	S	S	operands	result	comp
INTEGER	V	S	S	operands	result	comp
INTEGER-OF-DATE	V	S	S	operands	result	comp
INTEGER-OF-DAY	V	S	S	operands	result	comp

Functions	Parser		CA/GDF				
			IA	Sup:	Pairs		Relation
					operands	result	
INTEGER-PART	V	S	S	operands	result	comp	
LENGTH	V	S	S	operands	result	comp	
LENGTH-AN	V	S	S	operands	result	comp	
LOG	V	S	S	operands	result	comp	
LOG10	V	S	S	operands	result	comp	
LOWER-CASE	V	S	S	operands	result	comp	
MAX	V	S	S	operands	result	comp	
MEAN	V	S	S	operands	result	comp	
MEDIAN	V	S	S	operands	result	comp	
MIDRANGE	V	S	S	operands	result	comp	
MIN	V	S	S	operands	result	comp	
MOD	V	S	S	operands	result	comp	
NATIONAL-OF	V	S	S	operands	result	comp	
NUMVAL	V	S	S	operands	result	comp	
NUMVAL-C	V	S	S	operands	result	comp	
ORD	V	S	S	operands	result	comp	
ORD-MAX	V	S	S	operands	result	comp	
ORD-MIN	V	S	S	operands	result	comp	
P1	V	S	S	operands	result	comp	
PRESENT-VALUE	V	S	S	operands	result	comp	
RANDOM	V	S	S	operands	result	comp	
RANGE	V	S	S	operands	result	comp	
REM	V	S	S	operands	result	comp	

**1-200** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

Functions	Parser	IA	Sup:	CA/GDF		
				Pairs		Relation
REVERSE	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
SIGN	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
SIN	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
SQRT	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
STANDARD-DEVIATION	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
SUM	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
TAN	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
UPPER-CASE	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
VARIANCE	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
WHEN-COMPILED	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>
YEARWINDOW	V	S	S	<i>operands</i>	<i>result</i>	<i>comp</i>

## **COBOL language structure**

### **Figurative constants**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S

### **Reserved data names**

<b>Data names</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
STATION-ID	V	S	S
ATTENTION-INDEX	V	S	S
BACK-TAB	V	S	S
FUNCTION-INDEX	V	S	S
CURRENT-FIELD	V	S	S
FIELD-CHANGED-FLAG	V	S	S
LINE-INDEX	V	S	S
COLUMN-INDEX	V	S	S
FIELD-EXIT-KEY	V	S	S

**1-202** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

**Literals**

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literal	<ul style="list-style-type: none"> <li><b>Format</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: -2.71828</li> </ul>	V	S	S
Hexadecimal literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: H"FC1D"</li> </ul>	V	S	S
Octal literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: O"7063"</li> </ul>	V	S	S
Binary literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: B"0100101101010"</li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

**COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i>            ENVIRONMENT DIVISION. <i>env-division-content</i>            DATA DIVISION. <i>data-division-content</i>            PROCEDURE DIVISION. <i>proc-division-content</i>            END-PROGRAM <i>program-name-1</i>.</li> </ul>	V	S



**Identification Division**

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b>  IDENTIFICATION DIVISION.  PROGRAM-ID. <i>program-name</i>.  [AUTHOR. <i>comment-entry</i>]  [INSTALLATION. <i>comment-entry</i>]  [DATE-WRITTEN. <i>comment-entry</i>]  [SECURITY. <i>comment-entry</i>]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point

1-204 Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

## Environment Division

### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential organization</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN <i>assignment-name-1</i>	V	S	defines program-name.assignment-name-1 (File)	
	[; ORGANIZATION [IS] SEQUENTIAL] [; NO DELETE [OF <i>literal</i> ]] [; [FILE] STATUS [IS] <i>data-name-1</i> ] [; RESERVE [NO] [ALTERNATE] AREA] [; ALTERNATE [INPUT AREA] STATUS [IS] <i>data-name-3</i> ]	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: relative organization</b></li> </ul> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN <i>assignment-name-1</i>	V	S	defines program-name.assignment-name-1 (File)	
	[; ORGANIZATION [IS] SEQUENTIAL] [; NO DELETE [OF <i>literal</i> ]] [; [FILE] STATUS [IS] <i>data-name-1</i> ] [; RESERVE [NO] [ALTERNATE] AREA] [; ALTERNATE [INPUT AREA] STATUS [IS] <i>data-name-3</i> ]	VO	N		
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> I-O-CONTROL. [; SAME [RECORD] [AREA] [FOR] <i>file-name-1</i> [, <i>file-name-4</i> ]]	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	
		VO	IA
OBJECT-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b>                      OBJECT-COMPUTER. &lt;UTS40   UTS4000   UTS30&gt;                      [, SEGMENT LIMIT IS <i>integer-1</i>]                      [MEMORY SIZE <i>integer</i> &lt;CHARACTRERS   MODULES   WORDS&gt;]                      [PROGRAM COLLATING SEQUENCE IS <i>alphabet-name</i>]                      [SUSPEND-KEY IS DISABLED]</li> </ul>	V	N
SOURCE-COM-PUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b>                      SOURCE-COMPUTER. [<i>computer-name</i> [WITH DEBUGGING MODE].]</li> </ul>	V	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b>                      SPECIAL-NAMES.                      [<i>alphabet-name-1</i> [IS] &lt;STANDARD-1   NATIVE   ASCII&gt;]</li> </ul>	V	S
	<ul style="list-style-type: none"> <li>[, CURRENCY [SIGN] [IS] <i>literal-6</i> [[WITH] PICTURE SYMBOL <i>literal-7</i>]</li> </ul>	V	N
	<ul style="list-style-type: none"> <li>[, DECIMAL-POINT [IS] COMMA] [.]</li> </ul>	V	S

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format</b>                      DATA DIVISION.                      [FILE SECTION. [<i>file-description-entry</i> <i>record-description-entry</i>]]                      [SCREEN SECTION. [<i>screen-description-entry</i>]]                      [WORKING-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]                      [SHARED-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]                      [LINKAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]                      [COMMON-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]]</li> </ul>	V	S	S		

**1-206** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format</b> FD <i>file-name-1</i> [; BLOCK [CONTAINS] <i>integer-1</i> &lt;CHARACTERS   RECORDS&gt;] [; RECORD [CONTAINS] <i>integer-2</i> TO <i>integer-3</i> [CHARACTERS]] [; LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED&gt; VALUE OF &lt;FILE-ID   PERIPHERAL-ID&gt; [IS] &lt;<i>data-name-1</i>   <i>literal-1</i>&gt; [; DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-2</i> [, <i>data-name-3</i> ...]] [; CODE-SET [IS] <i>alphabet-name</i>].</li> </ul>	V	S	S			
Screen Description (SR) Entry	<ul style="list-style-type: none"> <li><b>Format</b> SR <i>file-name</i> [; DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-1</i> [, <i>data-name-2</i> ...]] [; RECORD [CONTAINS] <i>integer-1</i> TO <i>integer-2</i> [CHARACTERS]] [; LINAGE IS <i>integer-3</i> LINES] [; LINE SIZE IS <i>integer-4</i> CHARACTERS]</li> </ul>	V	S	S			
SCREEN DATA description	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; [<i>occurs-clause</i>] [<i>picture-clause</i>] [<i>usage-clause</i>] [<i>justified-clause</i>] [<i>value-clause</i>] [<i>sign-clause</i>] [<i>protected-clause</i>] [<i>intensity-clause</i>] [<i>emphasis-clause</i>] [<i>line-clause</i>] [<i>column-clause</i>] [<i>alternate-character-clause</i>] [<i>upper-case-clause</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> 88 <i>condition-name</i> <i>value-clause</i>.</li> </ul>	V	S	S			
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; [<i>redefines-clause</i>] [<i>blank-when-zero-clause</i>] [<i>justified-clause</i>] [<i>occurs-clause</i>] [<i>picture-clause</i>] [<i>sign-clause</i>] [<i>synchronized-clause</i>] [<i>usage-clause</i>] [<i>value-clause</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> 88 <i>condition-name</i> <i>value-clause</i>.</li> </ul>	V	S	S			
ALTERNATE CHARACTER clause	<ul style="list-style-type: none"> <li><b>Format</b> ALTERNATE [CHARACTER] REPRESENTATION</li> </ul>	V	S	N			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> [; BLANK [WHEN] ZERO</li> </ul>	V	S	N			

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
COLUMN clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;COL   COLUMN&gt; NUMBER [IS] &lt;integer-1   PLUS integer-2&gt;</li> </ul>	VO	N	N			
EMPHASIS clause	<ul style="list-style-type: none"> <li><b>Format</b> EMPHASIS [IS] &lt;UNDERSCORE   STRIKE-THROUGH   FRAMED   COLUMN-SEPARATOR&gt;</li> </ul>	VO	N	N			
INTENSITY clause	<ul style="list-style-type: none"> <li><b>Format</b> INTENSITY [IS] &lt;BLINK   HIGH   LOW   NON-DISPLAYED   REVERSE&gt;</li> </ul>	VO	N	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;JUSTIFIED   JUST&gt; [RIGHT]</li> </ul>	VO	N	N			
LINE clause	<ul style="list-style-type: none"> <li><b>Format</b> LINE [NUMBER] [IS] &lt;integer-1   PLUS integer-2&gt;</li> </ul>	VO	N	N			
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format</b> OCCURS integer-2 [TIMES] [INDEXED [BY] index-name-1]</li> </ul>	V	S	S			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; [IS] character-string</li> </ul>	V	S	S			
PROTECTED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PROT   PROTECTED&gt;</li> </ul>	VO	N	N			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> level-number data-name-1 REDEFINES data-name-2</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt; [SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			
UPPER-CASE clause	<ul style="list-style-type: none"> <li><b>Format</b> UPPER-CASE</li> </ul>	VO	N	N			

**1-208** Supported COBOL Statements  
Supported Micro Focus COBOL statements

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
USAGE clause	<ul style="list-style-type: none"> <li><b>Format</b> USAGE [IS] &lt;COMP   COMP-1   COMP-2   COMPUTATIONAL   DISPLAY   INDEX   LOCK&gt;</li> </ul>	V	S	S		
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> 88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S	S		

**Procedure Division**

**Arithmetic operators**

Operation	Meaning	Format	Parser	IA	CA/GDF		
					Sup.	Pair	Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S		
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S		
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S		
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S		
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S		
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S		

-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> - oper</li> </ul>	V	S	S			
---	----------------------	--	---	---	---	--	--	--

### Conditional Expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC&gt;</li> </ul>	V	S	S			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL   &gt;   &lt;   =&gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	cond
Combined conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			

1-210 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern.usin g		
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. <i>paragraph-name</i>. [<i>sentences</i>] END DECLARATIVES.</li> </ul>	V	S	S					
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: elementary</b> ACCEPT <i>id</i> [NON-WAIT]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.scre en		
	<ul style="list-style-type: none"> <li><b>Format 2: group</b> ACCEPT &lt;<i>id</i>   SCREEN&gt; AT <i>id-2</i></li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.scre en		
	<ul style="list-style-type: none"> <li><b>Format 3: sequenced</b> ACCEPT <i>id-1</i>, <i>id-2</i> [, <i>id-3</i> ...]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 4</b> ACCEPT <i>id</i> FROM &lt;DATE   DAY   TIME&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 5</b> ACCEPT DATE FROM <i>id-1</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal</i>&gt; TO <i>id-2</i> [ROUNDED] [; [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern.usin g		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-1</i>&gt; GIVING <i>id-3</i> [ROUNDED] [; [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.scre en		
ALERT	<ul style="list-style-type: none"> <li><b>Format</b> ALERT [ON   OFF]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern.scre en		



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER <i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i></li> </ul>	V	S	S					
CALL	<ul style="list-style-type: none"> <li><b>Format 1</b> CALL <i>literal-1</i> [USING <i>data-name-1</i>]</li> </ul>	V	S	S	<i>id-2, id-3</i>	<i>port</i>	extern.call	refers to <i>literal-1</i> (program Entry Point), defines <i>id-1</i> (Decision)	Program Calls Program Entry Point, Program Calls Decision
					<i>port</i>	<i>id-2, id-5</i>	extern.call		
CHANGE	<ul style="list-style-type: none"> <li><b>Format 1</b> CHANGE <i>id-1</i> TO &lt;HIGH   LOW   BLINK   NON-DISPLAYED&gt; [INTENSITY [WITH [NO] TABSET]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> CHANGE <i>id-1</i> TO &lt;NO   UNDERSCOPE   STRIKE-THROUGH   COLUMN-SEPARATOR   FRAMED   ALTERNATE&gt; EMPHASIS</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> CHANGE <i>id-1</i> TO [NO] SUBSET</li> </ul>	V	S	S					
CLEAR	<ul style="list-style-type: none"> <li><b>Format 1</b> CLEAR &lt;<i>file-name</i>   <i>record-name</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> CLEAR SCREEN</li> </ul>								
	<ul style="list-style-type: none"> <li><b>Format 3</b> CLEAR ACCEPT</li> </ul>	V	S	S					
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					
	[<REEL   UNIT> [WITH NO REWIND]]	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: relative</b> CLOSE <i>file-name-1</i></li> </ul>	V	S	S					

1-212 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE <i>id-1</i> [ROUNDED] = <i>arithmetic-expr</i> [: [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>identifiers of the arithmetic-expr</i>	<i>id-1</i>	comp+ or comp* (according to the operation performed)	
COPY	<ul style="list-style-type: none"> <li><b>Format (preprocessor statement)</b> COPY <i>text-name</i>.</li> </ul>	V	N	N				
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [INVALID [KEY] <i>imperative-stmt-1</i>]</li> </ul>	V	S	S				<i>refer to File (see FILE-CONTROL paragraph)</i>  <i>Program Deletes From File</i>
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [AT LINE <i>literal-2</i> [, COLUMN <i>literal-4</i>]]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.screen	
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO <i>id-2</i> [ROUNDED] [: [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-1</i>	comp*	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; &lt;BY   INTO&gt; &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [REMAINDER <i>id-4</i>] [: [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-3</i>	comp*	
EXIT	<ul style="list-style-type: none"> <li><b>Format 1</b> EXIT.</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2</b> EXIT <i>procedure-name</i>.</li> </ul>	V	S	S				
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S				
FORMAT	<ul style="list-style-type: none"> <li><b>Format</b> FORMAT <i>record-name</i></li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
FUNCTION NUMVAL	<ul style="list-style-type: none"> <li><b>Format</b> FUNCTION NUMVAL (<i>alphanumeric-argument</i>)</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] <i>procedure-name-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] <i>procedure-name-1</i> DEPENDING [ON] <i>id-1</i></li> </ul>	V	S	S					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF <i>condition-1</i> THEN &lt;<i>stmt-1</i>   NEXT SENTENCE&gt; [; ELSE &lt;<i>stmt-2</i>   NEXT SENTENCE&gt;]</li> </ul>	V	S	S					
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING, <i>id-2</i> FOR, &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   , &lt;ALL   LEADING   FIRST&gt;, &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING, <i>id-2</i> FOR, &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   , &lt;ALL   LEADING   FIRST&gt;, &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
LOAD	<ul style="list-style-type: none"> <li><b>Format</b> LOAD &lt;<i>id-1</i>   <i>literal-1</i>&gt; FROM &lt;<i>id-2</i>   <i>literal-2</i>&gt;</li> </ul>	V	S	S					

**1-214** Supported COBOL Statements  
*Supported Micro Focus COBOL statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
LOCK	<ul style="list-style-type: none"> <li><b>Format</b> LOCK <i>id-1</i> [<i>id-2</i>]</li> </ul>	V	S	S				
MOVE	<ul style="list-style-type: none"> <li><b>Format</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)	
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [: [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp*	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [: [ON] SIZE ERROR <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*	
OPEN	<ul style="list-style-type: none"> <li><b>Format</b> OPEN &lt;INPUT <i>file-name-1</i>  </li> </ul>	V	S	S				
	[[WITH] NO REWIND]	VO	N	N				
	OUTPUT <i>file-name-2</i> [	V	S	S				
	[[WITH] NO REWIND]	VO	N	N				
	I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i> >	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] &lt;<i>id-1</i>   <i>integer-1</i>&gt; TIMES</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] UNTIL <i>condition-1</i></li> </ul>	V	S	S					
READ	<ul style="list-style-type: none"> <li><b>Format 1</b> READ <i>file-name-1</i> RECORD [INTO <i>id</i>] [; [AT] END <i>imperative-stmf</i>]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 2</b> READ <i>file-name-1</i> RECORD [INTO <i>id</i>] ; SEARCH KEY IS <i>data-name</i> [; INVALID [KEY] <i>imperative-stmf</i>]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 3</b> READ <i>file-name-1</i> NEXT RECORD [INTO <i>id</i>] [; [AT] END <i>imperative-stmf</i>]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> READ <i>file-name-1</i> RECORD [INTO <i>id</i>] [; INVALID [KEY] <i>imperative-stmf</i>]</li> </ul>	V	S	S					
REWRITE	<ul style="list-style-type: none"> <li><b>Format</b> REWRITE <i>file-name-1</i> [FROM <i>id-1</i>] [; INVALID [KEY] <i>imperative-stmf</i>]</li> </ul>	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Updates File
SEEK	<ul style="list-style-type: none"> <li><b>Format</b> SEEK <i>file-name</i> [RECORD] KEY [IS] <i>data-name</i> [; INVALID [KEY] <i>imperative-stmf</i>]</li> </ul>	V	S	N					

1-216 Supported COBOL Statements  
Supported Micro Focus COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
SET	<ul style="list-style-type: none"> <li><b>Format 1</b> SET &lt;index-name-1   id-1&gt; TO &lt;index-name-2   id-2   integer-1&gt;</li> </ul>	V	S	S	index-name-1 id-1	index-name-2 id-2 integer-1	move	
	<ul style="list-style-type: none"> <li><b>Format 2</b> SET index-name-3 &lt;UP BY   DOWN BY&gt; &lt;id-3   integer-2&gt;</li> </ul>	V	S	S	index-name-1 id-1	id-3 integer-2	move	
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT file-name-1 [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] data-name-1 [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S				
	[[COLLATING] SEQUENCE [IS] alphabet-name-1]	VO	N	N				
	<USING file-name-2   INPUT PROCEDURE [IS] procedure-name-1 [<THROUGH   THRU> procedure-name-2]> <GIVING file-name-3   OUTPUT PROCEDURE [IS] procedure-name-3 [<THROUGH   THRU> procedure-name-4]>	V	S	S				
START	<ul style="list-style-type: none"> <li><b>Format</b> START file-name-1 [KEY [IS] &lt;EQUAL [TO]   =   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt; &gt; data-name] ; [INVALID [KEY] imperative-stmt-1]</li> </ul>	V	S	S				
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   literal&gt;</li> </ul>	V	S	S				
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b> SUBTRACT &lt;id-1   literal-1&gt; FROM id-2 [ROUNDED] ; [ON] SIZE ERROR imperative-stmt-1]</li> </ul>	V	S	S	id-1	id-2	comp+	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> SUBTRACT &lt;id-1   literal-1&gt; FROM &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] ; [ON] SIZE ERROR imperative-stmt-1]</li> </ul>	V	S	S	id-1	id-3	comp+	
UNLOCK	<ul style="list-style-type: none"> <li><b>Format</b> UNLOCK id-1 [, id-2 ...]</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
USE	<ul style="list-style-type: none"> <li><b>Format 1</b> USE AFTER [STANDARD] &lt;ERROR   EXCEPTION&gt; PROCEDURE [ON] &lt;file-name   INPUT   OUTPUT   I-O&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> USE [FOR] SCREEN PROCESSING ON <i>data-name</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3</b> USE [FOR] ATTENTION-KEY PROCESSING</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4</b> USE [FOR] INTERRUPT PROCESSING [ON] <i>file-name</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5</b> USE [FOR] FUNCTION-KEY PROCESSING</li> </ul>	V	S	S					
WAIT	<ul style="list-style-type: none"> <li><b>Format 1</b> WAIT &lt;<i>literal</i>   <i>id</i> [SECOND[S]]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> WAIT ON &lt;<i>lock-variable</i>   <i>file-name</i>   I-O&gt;</li> </ul>	V	S	S					
WRITE	<ul style="list-style-type: none"> <li><b>Format 1</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [&lt;BEFORE   AFTER&gt; [ADVANCING] &lt;<i>id-1</i> [LINE   LINES]   <i>integer-1</i>   PAGE&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
					<i>record-name-1</i>	<i>port</i>	extern.file		
	<ul style="list-style-type: none"> <li><b>Format 2</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [; INVALID [KEY] <i>imperative-stmt-1</i>]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
	<ul style="list-style-type: none"> <li><b>Format 3</b> WRITE <i>record-name-1</i> [FROM ATTENTION-INDEX]</li> </ul>	V	S	S					
<ul style="list-style-type: none"> <li><b>Format 4</b> WRITE FUNCTION-INDEX</li> </ul>	V	S	S						

1-218 Supported COBOL Statements  
*Supported HP COBOL II/XL statements*

## Supported HP COBOL II/XL statements

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
ZERO / ZEROS / ZEROES	V	S	S
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S

#### Special registers

Registers	Parser	IA	CA/GDF
CURRENT-DATE	V	S	S
DEBUG-ITEM	V	S	S
LINAGE-COUNTER	V	S	S
RETURN-CODE	V	S	S
TALLY	V	S	S
TIME-OF-DAY	V	S	S
WHEN-COMPILED	V	S	S



### Literals

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literal	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: -2.71828</li> </ul>	V	S	S
Numeric octal literals	<ul style="list-style-type: none"> <li><b>Format</b> %456</li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$99,999,999.99</li> </ul>	V	S	S

### Referencing names

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	S

1-220 Supported COBOL Statements  
Supported HP COBOL II/XL statements

Division	Format	Parser	IA	CA/GDF
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: subscripting</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>integer-1</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>&gt;</li> </ul>	V	S	S treated as reference to the whole array
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>data-name-1</i>  </li> </ul>	V	S	S treated as reference to the whole <i>data-name-1</i>
	FUNCTION <i>function-name-1</i> ( <i>arguments</i> )>	VO	N	
	( <i>leftmost-character-position: [length]</i> )	V	S	

### COBOL program structure

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i> .	V	S
Nested program	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> <i>nested source program</i> END-PROGRAM <i>program-name-1</i> .	N	N

### Identification Division

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <IDENTIFICATION   ID> DIVISION. PROGRAM-ID. <i>program-name</i> [[IS] <COMMON [INITIAL]   INITIAL [COMMON]> [PROGRAM]]. [AUTHOR. <i>comment-entry</i> ] [INSTALLATION. <i>comment-entry</i> ] [DATE-WRITTEN. <i>comment-entry</i> ] [DATE-COMPILED. <i>comment-entry</i> ] [SECURITY. <i>comment-entry</i> ] [REMARKS. <i>comment-entry</i> ]	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Define-sProgram, Program Has Program Entry Point

### Environment Division

1-222 Supported COBOL Statements  
Supported HP COBOL II/XL statements

**Input-output section**

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program ReadsDataPort, WritesDataPort depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>stat-item</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program ReadsDataPort, WritesDataPort depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] INDEXED [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD [KEY] [IS] <i>data-name-2</i> [WITH DUPLICATES] [ALTERNATE RECORD [KEY] [IS] <i>data-name-3</i> [[WITH] DUPLICATES]] [[FILE] STATUS [IS] <i>stat-item</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program ReadsDataPort, WritesDataPort depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC>] RELATIVE [KEY] [IS] <i>data-name-4</i> ] [[FILE] STATUS [IS] <i>stat-item</i> ].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: random-access file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program ReadsDataPort, WritesDataPort depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] ACCESS [MODE] [IS] RANDOM ACTUAL KEY IS <i>data-name-1</i> [[FILE] STATUS [IS] <i>stat-name</i> ].	VO	N		

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential I-O</b></li> </ul> I-O-CONTROL. [SAME [RECORD   SORT   SORT-MERGE] AREA FOR <i>file-name-1</i> { <i>file-name-2</i> } ...] [MULTIPLE FILE [TAPE] [CONTAINS] <i>file-name-3</i> [POSITION] <i>integer-1</i> ].	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA
OBJECT-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> OBJECT-COMPUTER. [ <i>computer-name</i> [MEMORY [SIZE] <i>integer</i> <WORDS   CHARACTERS   MODULES>] [[PROGRAM] [COL-LATING] SEQUENCE [IS] <i>alphabet-name</i> ] [SEGMENT-LIMIT [IS] <i>priority-number</i> ].]	VO	N
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> SOURCE-COMPUTER. [ <i>computer-name</i> [[WITH] DEBUGGING MODE].]	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> SPECIAL-NAMES. [[< <i>feature-name</i>   <i>switch-name</i>   <i>device-name</i> > <[IS] <i>mnemonic-name-1</i> <ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i> ]   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i> ]>   <ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i> ]   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i> ]> >	V	S
	[ALPHABET <i>alphabet-name-1</i> [IS] <STANDARD-1   STANDARD-2   NATIVE   EBCDIC   EBCDIK   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> ]>]	VO	N
	[SYMBOLIC [CHARACTERS] <i>symbolic-character</i> [ARE   IS] <i>integer-1</i> [IN] <i>alphabet-name-2</i> ]	VO	N
	[CLASS <i>class-name-1</i> [IS] <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]	VO	N
	[CURRENCY [SIGN] [IS] <i>literal-6</i> ]	VO	N
	[DECIMAL-POINT [IS] COMMA] [,]	V	S

**1-224** Supported COBOL Statements  
Supported HP COBOL II/XL statements

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Data Division	<ul style="list-style-type: none"> <li><b>Format: program and method DATA DIVISION</b> DATA DIVISION. [FILE SECTION. [<i>file-description-entry record-description-entry</i>]] [WORKING-STORAGE SECTION. [<i>record-description-entry   data-item-description-entry</i>]] [LINKAGE SECTION. [<i>record-description-entry   data-item-description-entry</i>]]</li> </ul>	V	S	S			
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL] [BLOCK [CONTAINS] <i>integer-1</i> TO <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;] [RECORDING MODE [IS] &lt;F   V   U   S&gt;] [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;] [LABEL &lt;[RECORD [IS]   RECORDS [ARE]]&gt; &lt;STANDARD   OMITTED&gt;] [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;] [DATA &lt;[RECORD [IS]   RECORDS [ARE]]&gt; <i>data-name-4</i>] [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;] [CODE-SET [IS] <i>alphabet-name</i>].</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: sort/merge files</b> SD <i>file-name-1</i> [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]&gt;] [DATA &lt;[RECORD [IS]   RECORDS [ARE]]&gt; <i>data-name-4</i>].</li> </ul>	V	S	S			
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; [<i>redefines-clause</i>] [<i>external-clause</i>] [<i>global-clause</i>] [<i>picture-clause</i>] [<i>usage-clause</i>] [<i>sign-clause</i>] [<i>occurs-clause</i>] [<i>synchronized-clause</i>] [<i>justified-clause</i>] [<i>blank-when-zero-clause</i>] [<i>value-clause</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> 66 <i>data-name-1</i> <i>renames-clause</i>.</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3</b> 88 <i>condition-name</i> <i>value-clause</i>.</li> </ul>	V	S	S			

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> BLANK [WHEN] &lt;ZERO   ZEROS   ZEROES&gt;</li> </ul>	V	S	N			
EXTERNAL clause	<ul style="list-style-type: none"> <li><b>Format</b> [IS] EXTERNAL</li> </ul>	V	S	N			
GLOBAL clause	<ul style="list-style-type: none"> <li><b>Format</b> [IS] EXTERNAL</li> </ul>	V	S	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;JUSTIFIED   JUST&gt; [RIGHT]</li> </ul>	VO	N	N			
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> OCCURS <i>integer-2</i> [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> OCCURS <i>integer-1</i> TO <i>integer-2</i> [TIMES] DEPENDING [ON] <i>data-name-1</i> [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; [IS] <i>character-string</i></li> </ul>	V	S	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; REDEFINES <i>data-name-2</i></li> </ul>	V	S	S			
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			

**1-226** Supported COBOL Statements  
Supported HP COBOL II/XL statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: standard</b> USAGE [IS] &lt;BINARY   COMP   COMP-3   COMPUTATIONAL   COMPUTATIONAL-3   DISPLAY   INDEX   PACKED-DECIMAL&gt;</li> </ul>	V	S	S			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> 88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S	S			

**Procedure Division**

**Arithmetic operators**

Operation	Meaning	Format	Parser	IA	CA/GDF		
					Sup.	Pair	Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S		
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S		
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S		
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S		
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S		



+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

### Conditional Expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   <i>class-name</i>&gt;</li> </ul>	V	S	S			
		VO	N	N			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	<i>cond</i>
Intrinsic relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>mnemonic-name</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; 0</li> </ul>	V	S	S			
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S			
Negated simple conditions	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>condition-1</i></li> </ul>	V	S	S			
Combined conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			

**1-228** Supported COBOL Statements  
Supported HP COBOL II/XL statements

Conditions	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
Abbreviated combined relation conditions	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S			

**Statements**

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE DIVISION [USING <i>data-name-1</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern. using		
PROCEDURE DIVISION structure	<ul style="list-style-type: none"> <li><b>Format</b> <i>procedure division header</i> [DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE <i>statement</i> [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.] <i>section-name</i> SECTION [<i>priority-number</i>]. [<i>paragraph-name</i>. [<i>sentence</i>]]</li> </ul>	V	S	S					
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE <i>sentence</i> [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.</li> </ul>	V	S	S					
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FREE] [FROM &lt;<i>mnemonic-name</i>   SYSIN   CONSOLE&gt;] [[ON] INPUT ERROR <i>imperative-stmt-1</i>] [NOT [ON] INPUT ERROR <i>imperative-stmt-2</i>] [END-ACCEPT]</li> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE   DAY   DAY-OF-WEEK   TIME&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;id-1   literal&gt; TO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;id-1   literal-1&gt; TO &lt;id-2   literal-1&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; id-1 TO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-ADD]</li> </ul>	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER</li> </ul>	V	S	S					
	procedure-name-1 TO [PROCEED TO] procedure-name-2	VO	N	N					
CALL	<ul style="list-style-type: none"> <li><b>Format 1: with ON OVERFLOW</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] id-2   [BY] CONTENT id-3&gt;] [[ON] OVERFLOW imperative-stmt-1] [END-CALL]]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
					port	id-2, id-5	extern.call		
	<ul style="list-style-type: none"> <li><b>Format 2: with ON EXCEPTION</b> CALL &lt;id-1   literal-1&gt; [USING &lt;[[BY] REFERENCE] id-2   [BY] CONTENT id-3&gt;] [ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	V	S	S	id-2, id-3	port	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
					port	id-2, id-5	extern.call		
	<ul style="list-style-type: none"> <li><b>Format 3: with GIVING</b> CALL &lt;id-1   [INTRINSIC] literal-1&gt; [USING &lt;@   @id-2   id-2   id-2   literal-2&gt;] [GIVING id-4] [ON OVERFLOW imperative-stmt-1] [END-CALL]</li> </ul>	V	S						

1-230 Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
	<ul style="list-style-type: none"> <li><b>Format 4: with GIVING</b> CALL &lt;id-1   [INTRINSIC] literal-1&gt; [USING &lt;\   @id-2   id-2   \id-2   \literal-2&gt;] [GIVING id-4] [ON] EXCEPTION imperative-stmt-1 [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	V	S					
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S				
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE file-name-1</li> </ul>	V	S	S				
	[<REEL   UNIT> [[FOR] REMOVAL   WITH NO REWIND]   [WITH] <NO REWIND   LOCK>]	VO	N	N				
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE file-name-1</li> </ul>	V	S	S				
	[[WITH] LOCK]	VO	N	N				
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE id-1 [ROUNDED] = arithmetic-expr [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-COMPUTE]</li> </ul>	V	S	S	identifiers of the arithmetic-expr	id-1	comp+ or comp* (according to the operation performed)	
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S				
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; &lt;library-name   literal-2&gt;] [SUPPRESS] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S	S			refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE file-name-1 [RECORD] [INVALID [KEY] imperative-stmt-1] [NOT INVALID [KEY] imperative-stmt-2] [END-DELETE]</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;id-1   literal-1&gt; [UPON &lt;mnemonic-name-1   SYSOUT   CONSOLE&gt;] [[WITH] NO ADVANCING]</li> </ul>	V	S	S	id-1	port	extern.scre en		
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;id-1   literal-1&gt; INTO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-1	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-3	comp*		
					id-1	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
					id-1	id-4	comp*		
					id-2	id-4	comp*		
<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*			
				id-2	id-3	comp*			
				id-1	id-4	comp*			
				id-2	id-4	comp*			
ENTER	<ul style="list-style-type: none"> <li><b>Format</b> ENTER language-name [routine-name]</li> </ul>	VO	N	N					
ENTRY	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTRY literal-1 [USING id-1]</li> </ul>	V	S	S	id-1	port	extern.usin g	defines literal-1 (Program Entry Point)	Program Has Program Entry Point
					port	id-1	extern.usin g		

1-232 Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;] WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1 [WHEN OTHER imperative-stmt-2] [END-EVALUATE] Phrases phrase-1 and phrase-2 should be represented in the following form: &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-3   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S	id-1, expr-1	condition-1	cond		
EXAMINE	<ul style="list-style-type: none"> <li><b>Format</b> EXAMINE id &lt;TALLYING &lt;UNTIL FIRST   ALL   LEADING&gt; literal-1 [REPLACING BY literal-2]   REPLACING &lt;UNTIL FIRST   ALL   LEADING&gt; literal-3 BY literal-4&gt;</li> </ul>	V	S	N					
EXCLUSIVE	<ul style="list-style-type: none"> <li><b>Format</b> EXCLUSIVE file-name [CONDITIONALLY]</li> </ul>	V	S	N					
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> paragraph-name. EXIT.</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b> GOBACK</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] procedure-name-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] procedure-name-1 DEPENDING [ON] id-1</li> </ul>	V	S	S					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF condition-1 THEN &lt;stmt-1   NEXT SENTENCE&gt; [ELSE &lt;stmt-2   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S					
INITIATE	<ul style="list-style-type: none"> <li><b>Format</b> INITIATE <i>report-name-1</i></li> </ul>	V	S	S					
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp		

**1-234** Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i></li> </ul>	V	S	N					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>USING <i>file-name-2 file-name-3</i> &lt;OUTPUT PROCEDURE [IS] <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>file-name-4</i>&gt;</li> </ul>	V	S	N					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT <i>file-name-1</i></li> </ul>	V	S	S					Program Reads DataPort, Program Writes DataPort
	[REVERSED   WITH NO REWIND]	VO	N	N					
	OUTPUT <i>file-name-2</i>	V	S	S					
	[WITH NO REWIND]	VO	N	N					
	I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i> >	V	S	S					



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b>                      PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]   imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b>                      PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] &lt;id-1   integer-1&gt; TIMES imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b>                      PERFORM [procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]] [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: out-of-line PERFORM ... VARYING</b>                      PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1                      AFTER &lt;id-5   index-name-3&gt; FROM &lt;id-6   index-name-4   literal-3&gt; BY &lt;id-7   literal-4&gt; UNTIL condition-2</li> </ul>	N	N	N					
	<ul style="list-style-type: none"> <li><b>Format 5: in-line PERFORM ... VARYING</b>                      PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					

1-236 Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
READ	<ul style="list-style-type: none"> <li><b>Format 1: all files</b></li> </ul> READ <i>file-name-1</i>	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move		
	[NEXT] [RECORD]	VO	N	N					
	[INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-READ]	V	S	S	<i>port</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	extern.file		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed files</b></li> </ul> READ <i>file-name-1</i> RECORD [INTO <i>id-1</i> ] [KEY [IS] <i>data-name-1</i> ] [INVALID [KEY] <i>imperative-stmt-3</i> ] [ NOT INVALID [KEY] <i>imperative-stmt-4</i> ] [END-READ]	V	S	S					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> RELEASE <i>record-name-1</i> [FROM <i>id-1</i> ]	V	S	S					
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> REPLACE == <i>pseudo-text-1</i> == BY == <i>pseudo-text-2</i> ==.	V	S	N/A					
	<ul style="list-style-type: none"> <li><b>Format 2</b></li> </ul> REPLACE OFF.	V	S	N/A					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> RETURN <i>file-name-1</i> [RECORD] [INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-RETURN]	V	S	S					
REWRITE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> REWRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [INVALID [KEY] <i>imperative-stmt-1</i> ] [NOT INVALID [KEY] <i>imperative-stmt-2</i> ] [END-REWRITE]	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move		
					<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	<i>port</i>	extern.file		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
SEEK	<ul style="list-style-type: none"> <li><b>Format</b> SEEK <i>file-name</i> [RECORD]</li> </ul>	N	N	N					
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> <i>integer-1</i>	move		
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET <i>condition-name-1</i> TO TRUE</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	move		

1-238 Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i> [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>&lt;USING <i>file-name-2</i>   INPUT PROCEDURE [IS] <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]&gt;</li> <li>&lt;GIVING <i>file-name-3</i>   OUTPUT PROCEDURE [IS] <i>procedure-name-3</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-4</i>]&gt;</li> </ul>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
UN-EXCLUSIVE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UN-EXCLUSIVE file-name	V	S						
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED [BY] [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER [IN] id-5] [COUNT [IN] id-6] [[WITH] POINTER id-7] [TALLYING [IN] id-8] [[ON] OVERFLOW imperative-stmt-1] [NOT [ON] OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

**1-240** Supported COBOL Statements  
Supported HP COBOL II/XL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
USE	<ul style="list-style-type: none"> <li><b>Format 1: EXCEPTION ERROR declarative</b> USE [GLOBAL] AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	V	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: LABEL declarative</b> USE [GLOBAL] AFTER [STANDARD] BEGINNING [FILE] LABEL PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	V	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: debugging</b> USE [FOR] DEBUGGING [ON] &lt;procedure-name   ALL PROCEDURES&gt;</li> </ul>	V	S						
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> WRITE record-name-1 [FROM id-1] [&lt;BEFORE   AFTER&gt; [ADVANCING] &lt;&lt;id-2   integer-1&gt; [LINE   LINES]   PAGE   mnemonic-name&gt;] [[AT] &lt;END-OF-PAGE   EOP&gt; imperative-stmt-3] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; imperative-stmt-4] [END-WRITE]</li> </ul>	V	S	S	id-1	record-name-1	move		
					record-name-1	port	extern.file		
	<ul style="list-style-type: none"> <li><b>Format 2: relative, indexed or random-access files</b> WRITE record-name-1 [FROM id-1] [INVALID [KEY] imperative-stmt-1] [NOT INVALID [KEY] imperative-stmt-2] [END-WRITE]</li> </ul>	V	S	S	id-1	record-name-1	move		
					record-name-1	port	extern.file		

**Intrinsic functions**

Functions	Parser	IA	Sup.	CA/GDF		
				Pairs		Relation
ACOS	V	S	S	operands	result	comp
ANNUITY	V	S	S	operands	result	comp

Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
ASIN	V	S	S	operands	result	comp	
ATAN	V	S	S	operands	result	comp	
CHAR	V	S	S	operands	result	comp	
COS	V	S	S	operands	result	comp	
CURRENT-DATE	V	S	S	operands	result	comp	
DATE-OF-INTEGER	V	S	S	operands	result	comp	
DAY-OF-INTEGER	V	S	S	operands	result	comp	
FACTORIAL	V	S	S	operands	result	comp	
INTEGER	V	S	S	operands	result	comp	
INTEGER-OF-DATE	V	S	S	operands	result	comp	
INTEGER-OF-DAY	V	S	S	operands	result	comp	
INTEGER-PART	V	S	S	operands	result	comp	
LENGTH	V	S	S	operands	result	comp	
LOG	V	S	S	operands	result	comp	
LOG10	V	S	S	operands	result	comp	
LOWER-CASE	V	S	S	operands	result	comp	
MAX	V	S	S	operands	result	comp	
MEAN	V	S	S	operands	result	comp	
MEDIAN	V	S	S	operands	result	comp	
MIDRANGE	V	S	S	operands	result	comp	
MIN	V	S	S	operands	result	comp	
MOD	V	S	S	operands	result	comp	
NUMVAL	V	S	S	operands	result	comp	

**1-242** Supported COBOL Statements  
*Supported HP COBOL II/XL statements*

Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
				operands	result		
NUMVAL-C	V	S	S	operands	result	comp	
ORD	V	S	S	operands	result	comp	
ORD-MAX	V	S	S	operands	result	comp	
ORD-MIN	V	S	S	operands	result	comp	
PRESENT-VALUE	V	S	S	operands	result	comp	
RANDOM	V	S	S	operands	result	comp	
RANGE	V	S	S	operands	result	comp	
REM	V	S	S	operands	result	comp	
REVERSE	V	S	S	operands	result	comp	
SIN	V	S	S	operands	result	comp	
SQRT	V	S	S	operands	result	comp	
STANDARD-DEVIATION	V	S	S	operands	result	comp	
SUM	V	S	S	operands	result	comp	
TAN	V	S	S	operands	result	comp	
UPPER-CASE	V	S	S	operands	result	comp	
VARIANCE	V	S	S	operands	result	comp	
WHEN-COMPILED	V	S	S	operands	result	comp	

**Preprocessor commands**

Commands	Parser	IA
\$COMMENT	V	S



<b>Commands</b>	<b>Parser</b>	<b>IA</b>
\$CONTROL	V	S
\$COPYRIGHT	V	S
\$DEFINE	V	S
\$EDIT	V	S
\$IF	V	S
\$INCLUDE	V	S
\$PAGE	V	S
\$PREPROCESSOR	V	S
\$SET	V	S
\$TITLE	V	S
\$VERSION	V	S

**1-244** Supported COBOL Statements  
*Supported Fujitsu COBOL85 (M Series) statements*

## Supported Fujitsu COBOL85 (M Series) statements

**Note:** Object-oriented COBOL statements are not supported.

### COBOL language structure

#### Figurative constants

Constants	Parser	IA	CA/GDF
[ALL] <ZERO   ZEROS   ZEROES>	V	S	S
[ALL] <SPACE   SPACES>	V	S	S
[ALL] <HIGH-VALUE   HIGH-VALUES>	V	S	S
[ALL] <LOW-VALUE   LOW-VALUES>	V	S	S
[ALL] NULL	V	S	S
[ALL] <QUOTE   QUOTES>	V	S	S
ALL literal	V	S	S
[ALL] symbolic-character	V	S	S

#### Special registers

Registers	Parser	IA	CA/GDF
AQLCA	N	N	N
EDIT-COLOR	N	N	N
EDIT-CURSOR	N	N	N

<b>Registers</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
EDIT-MODE	N	N	N
EDIT-OPTION	N	N	N
EDIT-STATUS	N	N	N
FCOM	V	S	S
LINAGE-COUNTER	V	S	S
LINE-COUNTER	V	S	S
PAGE-COUNTER	V	S	S
PROGRAM-STATUS	V	S	S
RETURN-CODE	V	S	S
SHIFT OUT / SHIFT IN	V	S	S
SORT-CONTROL	V	S	S
SORT-CORE-SIZE	V	S	S
SORT-FILE-SIZE	V	S	S
SORT-MESSAGE	V	S	S
SORT-MODE-SIZE	V	S	S
SORT-RETURN	V	S	S
SORT-STATUS	V	S	S
TALLY	V	S	S
WHEN COMPILED	V	S	S

**1-246** Supported COBOL Statements  
*Supported Fujitsu COBOL85 (M Series) statements*

**Literals**

Literal type	Format	Parser	IA	CA/GDF
Nonnumeric literals	<ul style="list-style-type: none"> <li><b>Format</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; mantissa E &lt;+   -&gt; exponent</li> </ul>	V	S	S
Hexadecimal nonnumeric literals	<ul style="list-style-type: none"> <li><b>Format</b> X"AB123FE9236"</li> </ul>	V	S	S
National literals	<ul style="list-style-type: none"> <li><b>Format</b> N"national-string" NC"national-string" NN"national-string" NA"national-string" NK"national-string" NH"national-string" G"national-string" NX"AB123FE9236"</li> </ul>	V	S	S
Boolean literals	<ul style="list-style-type: none"> <li><b>Format</b> B"010010111010"</li> </ul>	V	S	S
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

### Referencing names

Division	Format	Parser	IA	CA/GDF
Qualification	<ul style="list-style-type: none"> <li><b>Format 1</b> &lt;data-name-1   condition-name-1   index-name-1&gt; &lt;&lt;IN   OF&gt; data-name-2 ... [&lt;IN   OF&gt; file-name-1]   &lt;IN   OF file-name-1&gt;&gt;</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> paragraph-name-1 &lt;IN   OF&gt; section-name-1</li> </ul>	V		
	<ul style="list-style-type: none"> <li><b>Format 3</b> text-name-1 &lt;IN   OF&gt; library-name-1</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER &lt;IN   OF&gt; file-name-1</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 5</b> &lt;PAGE-COUNTER   LINE-COUNTER&gt; &lt;IN   OF&gt; report-name-1</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 6</b> data-name-1 &lt;&lt;IN   OF&gt; data-name-2 [&lt;IN   OF&gt; report-name-1]   &lt;IN   OF&gt; report-name-1&gt;</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 7</b> &lt;EDIT-MODE   EDIT-OPTION   EDIT-COLOR   EDIT-STATUS   EDIT-CURSOR&gt; &lt;IN   OF&gt; id-1</li> </ul>	N	N	N
Subscripting	<ul style="list-style-type: none"> <li><b>Format</b> &lt;condition-name-1   data-name-1&gt; (&lt;ALL   integer-1   data-name-2 [&lt;+   -&gt; integer-2]   index-name-1&gt; [&lt;+   -&gt; integer-3] ...)</li> </ul>	V	S	S
Function-identifier	<ul style="list-style-type: none"> <li><b>Format</b> FUNCTION function-name-1 [(argument-1 ...)]</li> </ul>	V	S	S
Reference modifier	<ul style="list-style-type: none"> <li><b>Format</b> &lt;data-name-1   (leftmost-character-position: [length])&gt;</li> </ul>	V	S	S

**1-248** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Division	Format	Parser	IA	CA/GDF
Identifier	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>function-identifier-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] ... [&lt;IN   OF&gt; &lt;<i>file-name-1</i>   <i>report-name-1</i>&gt;] [(<i>subscript</i> ...)] [<i>reference-modifier</i>]</li> </ul>	V	S	S

**COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b> IDENTIFICATION DIVISION. PROGRAM-ID. <i>program-description-entry</i> <i>id-division-entries</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i>.</li> </ul>	V	S

**Identification Division**

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b> IDENTIFICATION DIVISION. PROGRAM-ID. <i>program-name</i>. [AUTHOR. [<i>comment-entry</i>]] [INSTALLATION. [<i>comment-entry</i>]] [DATE-WRITTEN. [<i>comment-entry</i>]] [DATE-COMPILED. [<i>comment-entry</i>]] [SECURITY. [<i>comment-entry</i>]]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point

**Environment Division**

**Input-output section**

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN [TO] < <i>file-id-1</i>   <i>file-id-literal-1</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL] [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-3</i> ]] [FORMAT [IS] <i>data-name-4</i> [GROUP [IS] <i>data-name-5</i> ]] [PADDING [CHARACTER] [IS] < <i>data-name-6</i>   <i>literal-1</i> >] [RECORD DELIMITER [IS] STANDARD-1]	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN [TO] < <i>file-id-1</i>   <i>file-id-literal-1</i> >	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] INDEXED [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD [KEY] [IS] <i>data-name-2</i> [[WITH] DUPLICATES] [ALTERNATE RECORD [KEY] [IS] <i>data-name-3</i> ]] [[WITH] DUPLICATES]] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		

**1-250** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN [TO] &lt;<i>file-id-1</i>   <i>file-id-literal-1</i>&gt;</li> </ul>	V	S	defines program-name.assignment-name-1 (File)	
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC> RELATIVE [KEY] [IS] <i>data-name-4</i> >] [[FILE] STATUS [IS] <i>data-name-2</i> [ <i>data-name-3</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: presentation file-control entries</b> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN [TO] <i>file-id-1</i></li> </ul>	V	S		
	[ORGANIZATION [IS]] SEQUENTIAL [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-2</i> ]] [FORMAT [IS] <i>data-name-3</i> ] [GROUP [IS] <i>data-name-4</i> ] [[SYMBOLIC] DESTINATION [IS] < <i>data-name-5</i>   <i>literal-1</i> >] [DESTINATION-1 [IS] <i>data-name-6</i> ] [DESTINATION-2 [IS] <i>data-name-7</i> ] [DESTINATION-3 [IS] <i>data-name-8</i> ] [PROCESSING [MODE] [IS] <i>data-name-9</i> ] [[SELECTED] FUNCTION [IS] <i>data-name-10</i> ] [UNIT CONTROL [IS] <i>data-name-11</i> ] [MESSAGE MODE [IS] <i>data-name-12</i> ] [END KEY [IS] <i>data-name-13</i> ] [SESSION CONTROL [IS] <i>data-name-14</i> ] [PROCESSING CONTROL [IS] <i>data-name-15</i> ] [MESSAGE CLASS [IS] <i>data-name-16</i> ] [PROCESSING TIME [IS] <i>data-name-17</i> ] [MESSAGE CODE [IS] <i>data-name-18</i> ] [MESSAGE OWNER [IS] <i>data-name-19</i> ]	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 5: direct file-control entries</b> FILE-CONTROL. SELECT <i>file-name-1</i> ASSIGN [TO] <i>file-id-1</i> [[FOR] MULTIPLE UNIT]</li> </ul>	V	S		
	[ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM>] [ACTUAL KEY [IS] <i>data-name-1</i> ] [<FILE-LIMIT [IS]   FILE-LIMITS [ARE]> < <i>data-name-1</i>   <i>literal-1</i> > <THROUGH   THRU> < <i>data-name-2</i>   <i>literal-2</i> >] [PROCESSING [MODE] [IS] SEQUENTIAL] [RESERVE <NO   <i>integer-1</i> >] [ALTERNATE] [AREA   AREAS]] [TRACK-LIMIT [IS] <i>integer-1</i> ] [TRACK   TRACKS]]	VO	N		



Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li> <b>Format 1: sequential I-O</b>                      I-O-CONTROL.                      &lt;RERUN ON <i>file-id-1</i> [EVERY] &lt;<i>integer-1</i> RECORDS   END [OF] &lt;REEL   UNIT&gt;&gt; [OF] <i>file-name-1</i>   SAME [RECORD] [AREA] [FOR] <i>file-name-3 file-name-4</i>   MULTIPLE FILE [TAPE] [CONTAINS] <i>file-name-5</i> [POSITION] <i>integer-2</i>&gt;.                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 2: relative and indexed I-O</b>                      I-O-CONTROL.                      &lt;RERUN ON <i>file-id-1</i> [EVERY] <i>integer-1</i> RECORDS [OF] <i>file-name-1</i>   SAME [RECORD] [AREA] [FOR] <i>file-name-3 file-name-4</i>&gt;                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 3: sort-merge I-O</b>                      I-O-CONTROL.                      [SAME &lt;RECORD   SORT   SORT-MERGE&gt; [AREA] [FOR] <i>file-name-3 file-name-4</i>.]                 </li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li> <b>Format 4: presentation file I-O</b>                      I-O-CONTROL.                      [SAME [RECORD] [AREA] [FOR] <i>file-name-3 file-name-4</i>] [APPLY &lt;MULTICONVERSATION-MODE   MULTICON&gt;] [APPLY SAVED-AREA [TO] <i>data-name-1</i>]                 </li> </ul>	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li> <b>Format</b>                      SOURCE-COMPUTER. [<i>computer-name</i> [[WITH] DEBUGGING MODE].]                 </li> </ul>	VO	N

**1-252** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Paragraphs and entries	Format	Parser	
		IA	IA
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [<i>environment-name-1</i> [IS] <i>mnemonic-name-1</i>   <i>environment-name-2</i> &lt;[IS] <i>mnemonic-name-2</i> <i>entry-1</i>   <i>entry-1</i>&gt;] where <i>entry-1</i> is: <i>entry-1</i>: &lt;ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i>]   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i>]&gt;</li> </ul>	V	S
	[ALPHABET <i>alphabet-name-1</i> [IS] <STANDARD-1   STANDARD-2   NATIVE   <i>function-name</i>   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> ]>]	VO	N
	[SYMBOLIC [CHARACTERS] <i>symbolic-character</i> [ARE   IS] <i>integer-1</i> [[IN] <i>alphabet-name-2</i> ] [SYMBOLIC CONSTANT <i>symbolic-constant-1</i> IS <i>literal-8</i> ...] ...	VO	N
	[CLASS <i>class-name-1</i> [IS] <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]	VO	N
	[CURRENCY [SIGN] [IS] <i>literal-6</i> [[WITH] PICTURE SYMBOL <i>literal-7</i> ]	VO	N
	[DECIMAL-POINT [IS] COMMA].	V	S

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format</b> DATA DIVISION. [BASED-STORAGE SECTION. [<i>77-level-description-entry</i>   <i>record-description-entry</i>] ... ] [FILE SECTION. [<i>file-description-entry</i> <i>record-description-entry</i>   <i>sort/merge-file-description-entry</i> <i>record-description-entry</i>   <i>report-file-description-entry</i>]] [WORKING-STORAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [CONSTANT SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [LINKAGE SECTION. [<i>record-description-entry</i>   <i>data-item-description-entry</i>]] [REPORT SECTION. [<i>report-description-entry</i> <i>report-group-description-entry</i>]]</li> </ul>	V	S	S		

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL   [IS] GLOBAL]            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;[CHARACTERS]   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]-&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]            [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]            [CODE-SET [IS] <i>alphabet-name</i>] [CONTROL &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-9</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: relative and indexed files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL   [IS] GLOBAL]            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;[CHARACTERS]   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]-&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>data-name-2</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: sort/merge files</b></li> </ul> <p>SD <i>file-name-1</i>            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i> [TO] <i>integer-7</i> [CHARACTERS] [DEPENDING [ON] <i>data-name-1</i>]-&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]&gt; <i>data-name-4</i>]</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 4: report files</b></li> </ul> <p>FD <i>file-name-1</i>            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;[CHARACTERS]   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [CHARACTERS]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [CHARACTERS]-&gt;]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;]            [VALUE OF <i>system-name-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [CODE-SET [IS] <i>alphabet-name</i>] [&lt;REPORT [IS]   REPORTS [ARE]&gt; <i>report-name-1</i>].</p>	V	S	S			

**1-254** Supported COBOL Statements  
*Supported Fujitsu COBOL85 (M Series) statements*

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b>  <i>level-number &lt;data-name-1   FILLER&gt;</i>  <i>[redefines-clause] [picture-clause] [usage-clause] [sign-clause] [occurs-clause] [synchronized-clause] [justified-clause] [blank-when-zero-clause] [value-clause] [character-type-clause] [printing-position-clause]</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b>  <i>66 data-name-1 renames-clause.</i></li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b>  <i>88 condition-name value-clause.</i></li> </ul>	V	S	S		
Report Description Entry	<ul style="list-style-type: none"> <li><b>Format</b>  <i>RD report-name-1</i>  <i>[CODE literal-1] [&lt;CONTROL [IS]   CONTROLS [ARE]&gt; &lt;data-name-1   FINAL data-name-1&gt;]</i>  <i>[PAGE [LIMIT [IS]   LIMITS [ARE]] integer-1 [LINE   LINES]] [HEADING integer-2] [FIRST DETAIL integer-3]</i>  <i>[LAST DETAIL integer-4] [FOOTING integer-5]</i></li> </ul>	V	S	N		

Entries and clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
Report Group Description	<ul style="list-style-type: none"> <li><b>Format 1</b> 01 [<i>data-name-1</i>] [LINE NUMBER [IS] &lt;<i>integer-1</i> [[ON] NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [NEXT GROUP [IS] &lt;<i>integer-3</i>   PLUS <i>integer-4</i>   NEXT PAGE&gt;] TYPE [IS] &lt;REPORT HEADING   RH   PAGE HEADING   PH   &lt;CONTROL HEADING   CH&gt; &lt;<i>data-name-2</i>   FINAL&gt;   DETAIL   DE   &lt;CONTROL FOOTING   CF&gt; &lt;<i>data-name-3</i>   FINAL&gt;   PAGE FOOTING   PF   REPORT FOOTING   RF&gt; [[USAGE [IS]] DISPLAY] [CHARACTER TYPE [IS] &lt;&lt;MODE-1   MODE-2   MODE-3&gt; [BY] <i>mnemonic-name-1</i>   <i>mnemonic-name-2</i>&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>level-number</i> [<i>data-name-1</i>] [LINE NUMBER IS &lt;<i>integer-1</i> [[ON] NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [[USAGE [IS]] DISPLAY] [CHARACTER TYPE [IS] &lt;&lt;MODE-1   MODE-2   MODE-3&gt; [BY] <i>mnemonic-name-1</i>   <i>mnemonic-name-2</i>&gt;]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>level-number</i> [<i>data-name-1</i>] &lt;PICTURE   PIC&gt; IS <i>character-string</i> [[USAGE [IS]] DISPLAY] [[SIGN [IS]] &lt;LEADING   TRAILING&gt; SEPARATE [CHARACTER]] [&lt;JUSTIFIED   JUST&gt; RIGHT] [BLANK WHEN ZERO] [LINE NUMBER [IS] &lt;<i>integer-1</i> [[ON] NEXT PAGE]   PLUS <i>integer-2</i>&gt;] [COLUMN NUMBER IS <i>integer-3</i>] [[PRINTING] POSITION [IS] <i>integer-4</i>] SOURCE IS <i>id-1</i> &lt;VALUE IS <i>literal</i>   SUM <i>id-2</i> [UPON <i>data-name-2</i>]&gt; [RESET ON &lt;<i>data-name-3</i>   FINAL&gt;] [GROUP INDICATE] [CHARACTER TYPE [IS] &lt;&lt;MODE-1   MODE-2   MODE-3&gt; [BY] <i>mnemonic-name-1</i>   <i>mnemonic-name-2</i>&gt;]</li> </ul>	V	S	N			
BLANK WHEN ZERO clause	<ul style="list-style-type: none"> <li><b>Format</b> BLANK [WHEN] &lt;ZERO   ZEROS   ZEROES&gt;</li> </ul>	V	S	N			
CHARACTER TYPE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;[CHARACTER TYPE [IS]] &lt;MODE-1   MODE-2   MODE-3&gt; [[BY] <i>mnemonic-1</i>]   CHARACTER TYPE [IS] <i>mnemonic-2</i>   CHARACTER TYPE [IS] &lt;<i>printing-mode-name-1</i>   <i>printing-mode-name-2</i> ... DEPENDING [ON] <i>data-name-5</i>&gt; [OR &lt;<i>printing-mode-name-3</i>   <i>printing-mode-name-4</i> ... DEPENDING [ON] <i>data-name-5</i>&gt;]</li> </ul>	VO	N	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;JUSTIFIED   JUST&gt; [RIGHT]</li> </ul>	VO	N	N			

**1-256** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> OCCURS <i>integer-2</i> [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> OCCURS <i>integer-1</i> TO <i>integer-2</i> [TIMES] DEPENDING [ON] <i>data-name-1</i> [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] <i>data-name-2</i>] [INDEXED [BY] <i>index-name-1</i>]</li> </ul>	V	S	S			
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; [IS] <i>character-string</i></li> </ul>	V	S	S			
PRINTING POSITION clause	<ul style="list-style-type: none"> <li><b>Format</b> [PRINTING] POSITION [IS] <i>integer</i> [[BY] <i>positioning-unti-name-1</i>]</li> </ul>	V	S	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; REDEFINES <i>data-name-2</i></li> </ul>	V	S	S			
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>66 data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: standard</b> USAGE [IS] &lt;BINARY   BIT   COMP   COMP-5   COMPUTATIONAL   COMPUTATIONAL-5   DISPLAY   INDEX   PACKED-DECIMAL   POINTER&gt;</li> </ul>	V	S	S			
VALUE clause	<ul style="list-style-type: none"> <li><b>Format 1: literal value</b> VALUE [IS] <i>literal</i></li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li><b>Format 2: condition-name value</b> <i>88 condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].</li> </ul>	V	S	S			

## Procedure Division

### Arithmetic operators

Operation	Meaning	Format	Parser	IA	Sup.	CA/GDF		
						Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1 oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

**1-258** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

**Conditional Expressions**

Conditions	Format	Parser	IA	CA/GDF		
				Sup.	Pair	Relation
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   BOOLEAN   KANJI   DBCS   <i>class-name</i>&gt;</li> </ul>	V	S	S		
		VO	N	N		
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S		
Relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO]   &gt;   &lt;   =&gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]   &gt;=   &lt;= &gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i> cond
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZERO&gt;</li> </ul>	V	S	S		
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-name</i></li> </ul>	V	S	S		
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>condition-1</i></li> </ul>	V	S	S		
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S		
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S		



**Statements**

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
PROCEDURE DIVISION structure	<ul style="list-style-type: none"> <li><b>Format 1</b> PROCEDURE DIVISION [USING <i>data-name-1</i> ...]. [DECLARATIVES section] <i>section-name</i> SECTION [<i>paragraph-name</i>. [<i>sentence</i> ...] ...]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2</b> PROCEDURE DIVISION [USING <i>data-name-1</i> ...]. <i>paragraph-name</i>. [<i>sentence</i> ...] ...</li> </ul>	V	S	S				
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION. USE [<i>paragraph-name</i> [<i>section</i> ...] ...] ... END-DECLARATIVES.</li> </ul>	V	S	S				
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>&gt;]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen	
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE   DAY   DAY-OF-WEEK   TIME&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen	
	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> FROM &lt;CONSOLE   SYSIN&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen	
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal</i>&gt; TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-1</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+	
					<i>id-2</i>	<i>id-3</i>	comp+	
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+	
					<i>id-2</i>	<i>id-2</i>	comp+	

**1-260** Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER <i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i></li> </ul>	V	S	S					
		VO	N	N					
CALL	<ul style="list-style-type: none"> <li><b>Format 1</b> CALL &lt;<i>id-1</i>   <i>literal-1</i>&gt; [USING &lt;[[BY] REFERENCE] <i>id-2</i>   [BY] CONTENT &lt;<i>id-3</i>   <i>literal-2</i>&gt;&gt;] [ON] OVERFLOW <i>imperative-stmt-1</i>] [END-CALL]</li> </ul>	V	S	S	<i>id-2, id-3</i>	<i>port</i>	extern.call	refers to <i>literal-1</i> (program Entry Point), defines <i>id-1</i> (Decision)	Program Calls Program Entry Point, Program Calls Decision
					<i>port</i>	<i>id-2, id-5</i>	extern.call		
	<ul style="list-style-type: none"> <li><b>Format 2</b> CALL &lt;<i>id-1</i>   <i>literal-1</i>&gt; [USING &lt;[[BY] REFERENCE] <i>id-2</i>   [BY] CONTENT &lt;<i>id-3</i>   <i>literal-2</i>&gt;&gt;] [[ON] EXCEPTION <i>imperative-stmt-1</i>] [NOT [ON] EXCEPTION <i>imperative-stmt-2</i>] [END-CALL]</li> </ul>	VO	N	N					
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;<i>id-1</i>   <i>literal-1</i>&gt;</li> </ul>	V	S	S					
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE <i>file-name-1</i> [&lt;REEL   UNIT&gt; [[FOR] REMOVAL]   [WITH] &lt;NO REWIND   LOCK&gt;]</li> </ul>	V	S	S					
		VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE <i>file-name-1</i> [[WITH] LOCK]</li> </ul>	V	S	S					
		VO	N	N					

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
COMPUTE	<ul style="list-style-type: none"> <li><b>Format 1</b> COMPUTE <i>id-1</i> [ROUNDED] = <i>arithmetic-expr</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-COMPUTE]</li> </ul>	V	S	S	<i>identifiers of the arithmetic-expr</i>	<i>id-1</i>	comp+ or comp* (according to the operation performed)	
	<ul style="list-style-type: none"> <li><b>Format 2</b> COMPUTE <i>id-1</i> = <i>boolean-expr</i></li> </ul>	V	S	S	<i>identifiers of the arithmetic-expr</i>	<i>id-1</i>	comp+ or comp* (according to the operation performed)	
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S				
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S				<i>refer to File (see FILE-CONTROL paragraph)</i> Program Deletes From File
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;<i>id-1</i>   <i>literal-1</i>&gt; [UPON &lt;<i>mnemonic-name-1</i> &gt;] [[WITH] NO ADVANCING]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.screen	

1-262 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;id-1   literal-1&gt; INTO id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-1	comp*		
					id-2	id-2	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-2	id-3	comp*		
					id-1	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
	<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; INTO &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*		
					id-2	id-3	comp*		
					id-1	id-4	comp*		
					id-2	id-4	comp*		
<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;id-1   literal-1&gt; BY &lt;id-2   literal-2&gt; GIVING id-3 [ROUNDED] REMAINDER id-4 [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-DIVIDE]</li> </ul>	V	S	S	id-1	id-3	comp*			
				id-2	id-3	comp*			
				id-1	id-4	comp*			
				id-2	id-4	comp*			
ENTER	<ul style="list-style-type: none"> <li><b>Format</b> ENTER language-name-1 [procedure-name-1]</li> </ul>	V	S	N					
ENTRY	<ul style="list-style-type: none"> <li><b>Format</b> ENTRY literal-1 [USING id-1]</li> </ul>	V	S	S	id-1	port	extern.using	defines literal-1 (Program Entry Point)	Program Has Pro- gram Entry Point
					port	id-1	extern.using		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;id-1   literal-1   expr-1   TRUE   FALSE&gt; [ALSO &lt;id-2   literal-2   expr-2   TRUE   FALSE&gt;] WHEN phrase-1 [ALSO phrase-2] imperative-stmt-1 [WHEN OTHER imperative-stmt-2] [END-EVALUATE] Phrases phrase-1 and phrase-2 should be represented in the following form: &lt;ANY   condition-1   TRUE   FALSE   [NOT] &lt;id-3   literal-1   arithmetic-expr-1&gt; [&lt;THROUGH   THRU&gt; &lt;id-4   literal-2   arithmetic-expr-2&gt;]&gt;</li> </ul>	V	S	S	id-1, expr-1	condition-1	cond		
EXAMINE	<ul style="list-style-type: none"> <li><b>Format</b> EXAMINE id &lt;TALLYING &lt;UNTIL FIRST   ALL   LEADING&gt; literal-1 [REPLACING BY literal-2]   REPLACING &lt;UNTIL FIRST   ALL   LEADING&gt; literal-3 BY literal-4&gt;</li> </ul>	V	S	N					
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> EXIT.</li> </ul>	V	S	S					
EXIT PERFORM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT [[TO] TEST [OF]] PERFORM.</li> </ul>	V	S	S					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GENERATE	<ul style="list-style-type: none"> <li><b>Format</b> GENERATE &lt;data-name-1   report-name-1&gt;</li> </ul>	V	S	S					
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] procedure-name-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] procedure-name-1 DEPENDING [ON] id-1</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: MORE-LABELS</b> GO [TO] MORE-LABELS</li> </ul>	VO	N	N					

1-264 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF <i>condition-1</i> THEN &lt;<i>stmt-1</i>   NEXT SENTENCE&gt; &lt;ELSE <i>stmt-2</i> [END-IF]   ELSE NEXT SENTENCE   END-IF&gt;</li> </ul>	V	S	S				
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED   NATIONAL   NATIONAL-EDITED   BOOLEAN   EGCS&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S				
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp	
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp	
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp	

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i></li> </ul>	V	S	N					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	USING <i>file-name-2 file-name-3</i> <OUTPUT PROCEDURE [IS] <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> ]   GIVING <i>file-name-4</i> >	V	S	N					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		

1-266 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT <i>file-name-1</i> [REVERSED   WITH NO REWIND]   OUTPUT <i>file-name-2</i> [WITH NO REWIND]   I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					
		VO	N	N					
		V	S	S					
		VO	N	N					
		V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] [<i>imperative-stmt-1</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]] &lt;<i>id-1</i>   <i>integer-1</i>&gt; TIMES [<i>imperative-stmt-1</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase</b> PERFORM [<i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] [(WITH) TEST &lt;BEFORE   AFTER&gt;] UNTIL <i>condition-1</i> [<i>imperative-stmt-1</i> END-PERFORM]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with VARYING phrase</b> PERFORM <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>] [(WITH) TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt; FROM &lt;<i>id-3</i>   <i>index-name-2</i>   <i>literal-1</i>&gt; BY &lt;<i>id-4</i>   <i>literal-2</i>&gt; UNTIL <i>condition-1</i> [AFTER &lt;<i>id-5</i>   <i>index-name-3</i>&gt; FROM &lt;<i>id-6</i>   <i>index-name-4</i>   <i>literal-3</i>&gt; BY &lt;<i>id-7</i>   <i>literal-4</i>&gt; UNTIL <i>condition-2</i>] [<i>imperative-stmt-1</i> END-PERFORM]</li> </ul>	V	S	S					



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
READ	<ul style="list-style-type: none"> <li><b>Format 1: sequential retrieval</b></li> </ul> READ <i>file-name-1</i> [NEXT] [RECORD]	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	[INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-READ]	VO	N	N	<i>port</i>	<i>file-rec-1</i> (corresponding to <i>file-name-1</i> )	extern.file	refer to File (see FILE-CONTROL paragraph)	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 2: random retrieval</b></li> </ul> READ <i>file-name-1</i> [RECORD] [INTO <i>id-1</i> ] [KEY [IS] <i>data-name-1</i> ] [INVALID [KEY] <i>imperative-stmt-3</i> ] [ NOT INVALID [KEY] <i>imperative-stmt-4</i> ] [END-READ]	V	S	S					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> RELEASE <i>record-name-1</i> [FROM <i>id-1</i> ]	V	S	S					
RETURN	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> RETURN <i>file-name-1</i> [RECORD] [INTO <i>id-1</i> ] [[AT] END <i>imperative-stmt-1</i> ] [NOT [AT] END <i>imperative-stmt-2</i> ] [END-RETURN]	V	S	S				refer to File (see FILE-CONTROL paragraph)	Program Reads File
REWRITE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> REWRITE <i>rec-name-1</i> [FROM <i>id-1</i> ] [INVALID [KEY] <i>imperative-stmt-1</i> ] [ NOT INVALID [KEY] <i>imperative-stmt-2</i> ] [END-REWRITE]	V	S	S	<i>id-1</i>	<i>file-rec-1</i> (corresponding to <i>rec-name-1</i> )	move	refer to File (see FILE-CONTROL paragraph)	Program Updates File
					<i>file-rec-1</i> (corresponding to <i>rec-name-1</i> )	<i>port</i>	extern.file		

1-268 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN <i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt;   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
SEEK	<ul style="list-style-type: none"> <li><b>Format</b> SEEK <i>file-name-1</i> [RECORD]</li> </ul>	N							
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> <i>integer-1</i>	move		
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET <i>condition-name-1</i> TO TRUE</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	move		
	<ul style="list-style-type: none"> <li><b>Format 5: pointer data items</b> SET &lt;<i>id-4</i>   ADDRESS OF <i>id-5</i>&gt; TO &lt;<i>id-6</i>   ADDRESS OF <i>id-7</i>   <i>literal-1</i>&gt;</li> </ul>	V	S						

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [KEY] <i>data-name-1</i> [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<USING <i>file-name-2</i>   INPUT PROCEDURE [IS] <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> ]> <GIVING <i>file-name-3</i>   OUTPUT PROCEDURE [IS] <i>procedure-name-3</i> [<THROUGH   THRU> <i>procedure-name-4</i> ]>	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; DELIMITED [BY] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		

1-270 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
SUPPRESS	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> SUPPRESS [PRINTING]	V	S	S					
TERMINATE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> TERMINATE report-name-1	V	S	S					
TRANSACTION	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> TRANSACTION <START   END   CANCEL   NOTE id-1>	VO	N	N					
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED [BY] [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER [IN] id-5] [COUNT [IN] id-6] [[WITH] POINTER id-7] [TALLYING [IN] id-8] [[ON] OVERFLOW imperative-stmt-1] [NOT [ON] OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
					id-1	record-name-1	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b></li> </ul> WRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [<BEFORE   AFTER> [ADVANCING] << <i>id-1</i>   <i>integer-1</i> > [LINE   LINES]   <i>mnemonic-name-1</i>   PAGE>] [[AT] <END-OF-PAGE   EOP> <i>imperative-stmt-3</i> ] [NOT [AT] <END-OF-PAGE   EOP> <i>imperative-stmt-4</i> ] [END-WRITE]	VO	N	N	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
				<i>record-name-1</i>	<i>port</i>	extern.file			
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b></li> </ul> WRITE <i>record-name-1</i> [FROM <i>id-1</i> ] [[INVALID [KEY] <i>imperative-stmt-1</i> ] [NOT INVALID [KEY] <i>imperative-stmt-2</i> ] [END-WRITE]	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move	refer to File (see FILE-CONTROL paragraph)	Program Inserts into File
				<i>record-name-1</i>	<i>port</i>	extern.file			

### Intrinsic functions

Functions	Parser	IA	Sup.	CA/GDF		
				Pairs		Relation
				operands	result	comp
ADDR	V	S	S	<i>operands</i>	<i>result</i>	comp
LENG	V	S	S	<i>operands</i>	<i>result</i>	comp

### Compile-directing statements and directives

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
*CONTROL (*CBL)	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <*CONTROL   *CBL> <SOURCE   NOSOURCE   LIST   NOLIST   MAP   NOMAP>	VO	N		
BASIS	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> [ <i>sequence-number</i> ] BASIS < <i>basis-name</i>   <i>literal-1</i> >	VO	N		

1-272 Supported COBOL Statements  
Supported Fujitsu COBOL85 (M Series) statements

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
COPY	<ul style="list-style-type: none"> <li><b>Format 1</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; library-name] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S	refers to text-name-1 (Copybook)	Cobol Includes Copybook
	<ul style="list-style-type: none"> <li><b>Format 2</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; library-name] DISJOINING word-3 JOINING word-4 AS &lt;PREFIX   SUFFIX&gt;</li> </ul>	V	S	refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] DELETE sequence-number-field</li> </ul>	VO	N		
EJECT	<ul style="list-style-type: none"> <li><b>Format</b> EJECT</li> </ul>	VO	N		
INSERT	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] INSERT sequence-number-field</li> </ul>	VO	N		
INCLUDE	<ul style="list-style-type: none"> <li><b>Format</b> INCLUDE &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; library-name]</li> </ul>	V	S		
NOTE	<ul style="list-style-type: none"> <li><b>Format</b> NOTE character-string</li> </ul>	V	S		
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b> REPLACE ==pseudo-text-1== BY ==pseudo-text-2==.</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPLACE OFF.</li> </ul>	V	S		
SKIP1/2/3	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SKIP1   SKIP2   SKIP3&gt;.</li> </ul>	VO	N		
TITLE	<ul style="list-style-type: none"> <li><b>Format</b> TITLE literal.</li> </ul>	VO	N		

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
USE	<ul style="list-style-type: none"> <li><b>Format 1</b> USE [GLOBAL] AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;file-name-1   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> USE BEFORE REPORTING <i>id-1</i></li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3</b> USE FOR DEAD-LOCK</li> </ul>	V	S		

### Supported AIM Network Database Statements

Clauses	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relations
CONNECT	<ul style="list-style-type: none"> <li><b>Format</b> CONNECT [<i>literal-1</i>   <i>id-1</i>] TO &lt;ALL   &lt;<i>literal-2</i>   <i>id-2</i>&gt; [NEXT   PRIOR   <i>integer</i>   AT <i>id-3</i>&gt;</li> </ul>	V	S	SS		
DISCONNECT	<ul style="list-style-type: none"> <li><b>Format</b> DISCONNECT [<i>literal-1</i>   <i>id-1</i>] FROM &lt;ALL   [<i>literal-2</i>   <i>id-2</i>&gt;</li> </ul>	V	S	S		
ERASE	<ul style="list-style-type: none"> <li><b>Format</b> ERASE &lt;<i>literal-1</i>   <i>id-1</i>&gt; [SELECTIVE   ALL]</li> </ul>	V	S	S	refer to <i>literal-1</i> (NetDbRecord)  defines <i>id-1</i> (Decision)	Program Deletes NetDbRecord  Program Deletes NetDbDecision
FIND	<ul style="list-style-type: none"> <li><b>Format</b> FIND <i>record-selection-expression</i> [[AT] END GO [TO] <i>procedure-name-1</i>]</li> </ul>	V	S	S		
FINISH	<ul style="list-style-type: none"> <li><b>Format</b> FINISH [<i>literal-1</i>]</li> </ul>	V	S	S		

**1-274** Supported COBOL Statements  
Supported AIM Network Database Statements

Clauses	Format	Parser	IA	CA/GDF	Repository	
					Entities	Relations
GET	<ul style="list-style-type: none"> <li><b>Format</b> GET <i>record-selection-expression</i> [[AT] END GO [TO] <i>procedure-name-1</i>]</li> </ul>	V	S	S		<i>Program Reads NetDbRecord</i>  <i>Program Reads NetDbDecision</i>
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF DB-EXCEPTION &lt;IS   [NOT] &lt;=   &lt;   &gt; &gt;&gt; <i>integer-1</i> [&lt;AND   OR&gt; [DB-EXCEPTION] &lt;IS   [NOT] &lt;=   &lt;   &gt; &gt;&gt; <i>integer-2 ...</i>] &lt;GO TO <i>procedure-name-1</i>   PERFORM <i>procedure-name-2</i> [THRU <i>procedure-name-3</i>]   CALL <i>literal-1</i> USING <i>literal-2</i>&gt; [END-IF]</li> </ul>	V	S	S		
MODIFY	<ul style="list-style-type: none"> <li><b>Format</b> MODIFY &lt;<i>literal-1</i>   <i>id-1</i>&gt;</li> </ul>	V	S	S	<i>refer to literal-1 (NetDbRecord)</i>  <i>defines id-1 (Decision)</i>	<i>Program Updates NetDbRecord</i>  <i>Program Updates NetDbDecision</i>
READY	<ul style="list-style-type: none"> <li><b>Format</b> READY [<i>subschema-name-1</i>]</li> </ul>	V	S	S		
STORE	<ul style="list-style-type: none"> <li><b>Format 1</b> STORE &lt;<i>literal-1</i>   <i>id-1</i>&gt; [TO &lt;ALL   &lt;<i>literal-2</i>   <i>id-2</i>&gt; [NEXT   PRIOR   <i>integer</i>   AT <i>id-3</i>&gt;]</li> </ul>	V	S	S	<i>refer to literal-1 (NetDbRecord)</i>	<i>Program Inserts NetDbRecord</i>
	<ul style="list-style-type: none"> <li><b>Format 2</b> STORE &lt;<i>literal-1</i>   <i>id-1</i>&gt; TO &lt;<i>literal-2</i>   <i>id-2</i>&gt; [AT] END GO [TO] <i>procedure-name-1</i></li> </ul>	V	S	S	<i>defines id-1 (Decision)</i>	<i>Program Inserts NetDbDecision</i>
USE	<ul style="list-style-type: none"> <li><b>Format</b> USE FOR DB-EXCEPTION</li> </ul>	V	S	S		



## **Supported SIEMENS COBOL statements**

**Note:** Object-oriented COBOL statements are not supported.

### **COBOL language structure**

#### **Figurative constants**

<b>Constants</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ZERO / ZEROS / ZEROES	N	N	N
SPACE / SPACES	V	S	S
HIGH-VALUE / HIGH-VALUES	V	S	S
LOW-VALUE / LOW-VALUES	V	S	S
QUOTE / QUOTES	V	S	S
ALL literal	V	S	S
NULL / NULLS	V	S	S

#### **Special registers**

<b>Registers</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ADDRESS OF	V	S	S
DEBUG ITEM	V	S	S

**1-276** Supported COBOL Statements  
*Supported SIEMENS COBOL statements*

<b>Registers</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
LENGTH OF	V	S	S
LINAGE-COUNTER	V	S	S
RETURN-CODE	V	S	S
SHIFT OUT / SHIFT IN	V	S	S
SORT-CONTROL	V	S	S
SORT-CORE-SIZE	V	S	S
SORT-FILE-SIZE	V	S	S
SORT-MESSAGE	V	S	S
SORT-MODE-SIZE	V	S	S
SORT-RETURN	V	S	S
TALLY	V	S	S
WHEN COMPILED	V	S	S
SORT-EOW	V	S	S
CBL-CTR	V	S	S

## Literals

Literal type	Format	Parser	IA	CA/GDF
Floating-point literals	<ul style="list-style-type: none"> <li><b>Format: floating-point</b> Example: &lt;+   -&gt; <i>mantissa</i> E &lt;+   -&gt; <i>exponent</i></li> </ul>	V	S	S
Nonnumeric literals	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3: with double-byte characters</b> "EBCDIC-data&lt;D1D2&gt;EBCDIC-data"</li> </ul>	N	N	N
	<ul style="list-style-type: none"> <li><b>Format 4: hexadecimal notation</b> X"hexadecimal-digits"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 5: null-terminated</b> Z"dddd"</li> </ul>	VO	N	N
Numeric literals	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; <i>mantissa</i> E &lt;+   -&gt; <i>exponent</i></li> </ul>	V	S	S
DBCS literals	<ul style="list-style-type: none"> <li><b>Format 1</b> G"&lt;D1D2D3&gt;"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> N"&lt;D1D2D3&gt;"</li> </ul>	N	N	N
PICTURE character string	<ul style="list-style-type: none"> <li><b>Format</b> Example: \$9(5).9(2)CR</li> </ul>	V	S	S

1-278 Supported COBOL Statements  
Supported SIEMENS COBOL statements

**Referencing names**

Division	Format	Parser	IA	CA/GDF
to COPY libraries	<ul style="list-style-type: none"> <li><b>Format</b> <i>text-name-1</i> [&lt;IN   OF&gt; <i>library-name-1</i>]</li> </ul>	V	S	S
to Procedure Division	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>paragraph-name-1</i> [&lt;IN   OF&gt; <i>section-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>section-name-1</i></li> </ul>	V	S	S
to Data Division	<ul style="list-style-type: none"> <li><b>Format 1: simple data reference</b> <i>data-name-1</i></li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: subscripting</b> <i>data-name-1</i> [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>) (<i>leftmost-character-position: [length]</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> &lt;<i>condition-name-1</i>   <i>data-name-1</i>&gt; [&lt;IN   OF&gt; <i>data-name-2</i>] [&lt;IN   OF&gt; <i>file-name-1</i>]</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 4</b> LINAGE-COUNTER [&lt;IN   OF&gt; <i>file-name-2</i>]</li> </ul>	V	S	S
Condition names	<ul style="list-style-type: none"> <li><b>Format 1: Data Division</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>data-name-1</i>] [&lt;IN   OF&gt; <i>file-name-1</i>] (<i>subscript</i>)</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: Special-Names paragraph</b> <i>condition-name-1</i> [&lt;IN   OF&gt; <i>mnemonic-name-1</i>]</li> </ul>	V	S	S
Subscript	<ul style="list-style-type: none"> <li><b>Format</b> &lt;<i>integer-1</i>   <i>arithmetic-expr</i>   ALL   <i>data-name-3</i> [&lt;+   -&gt; <i>integer-2</i>]   <i>index-name-1</i> [&lt;+   -&gt; <i>integer-3</i>]&gt;</li> </ul>	V	S	S treated as reference to the whole array

Division	Format	Parser	IA	CA/GDF
Reference modification	<ul style="list-style-type: none"> <li><b>Format</b> &lt;data-name-1   FUNCTION <i>function-name-1</i> (<i>arguments</i>)&gt; (<i>leftmost-character-position</i>: [<i>length</i>])</li> </ul>	V	S	S treated as reference to the whole <i>data-name-1</i>
		VO	N	
		V	S	

### **SIEMENS COBOL program structure**

Program type	Format	Parser	IA
Single program	<ul style="list-style-type: none"> <li><b>Format</b> &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> END-PROGRAM <i>program-name-1</i>.</li> </ul>	V	S
Nested program	<ul style="list-style-type: none"> <li><b>Format</b> &lt;IDENTIFICATION   ID&gt; DIVISION. PROGRAM-ID. <i>program-name-1</i> ENVIRONMENT DIVISION. <i>env-division-content</i> DATA DIVISION. <i>data-division-content</i> PROCEDURE DIVISION. <i>proc-division-content</i> <i>nested source program</i> END-PROGRAM <i>program-name-1</i>.</li> </ul>	N	N

**1-280** Supported COBOL Statements  
Supported SIEMENS COBOL statements

**Identification Division**

Paragraphs	Format	Parser	IA	Repository	
				Entities	Relation
Program Identification Division	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION.            PROGRAM-ID. <i>program-name</i> [[IS] &lt;RECURSIVE   COMMON [INITIAL]   INITIAL [COMMON]&gt; [PROGRAM]].            [AUTHOR. <i>[comment-entry]</i>]            [INSTALLATION. <i>[comment-entry]</i>]            [DATE-WRITTEN. <i>[comment-entry]</i>]            [DATE-COMPILED. <i>[comment-entry]</i>]            [SECURITY. <i>[comment-entry]</i>]</li> </ul>	V	S	defines program-name (Program) program-name (Program Entry Point)	Cobol File Defines Program, Program Has Pro- gram Entry Point
Class Identification Division	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION.            CLASS-ID. <i>class-name-1</i> INHERITS <i>class-name-2</i> [METACLASS [IS] <i>class-name-3</i>].            [AUTHOR. <i>[comment-entry]</i>]            [INSTALLATION. <i>[comment-entry]</i>]            [DATE-WRITTEN. <i>[comment-entry]</i>]            [DATE-COMPILED. <i>[comment-entry]</i>]            [SECURITY. <i>[comment-entry]</i>]</li> </ul>	N	N		
Method Identification Division	<ul style="list-style-type: none"> <li><b>Format</b>            &lt;IDENTIFICATION   ID&gt; DIVISION.            METHOD-ID. <i>method-name-1</i> [[IS] [METHOD] OVERRIDE].            [AUTHOR. <i>[comment-entry]</i>]            [INSTALLATION. <i>[comment-entry]</i>]            [DATE-WRITTEN. <i>[comment-entry]</i>]            [DATE-COMPILED. <i>[comment-entry]</i>]            [SECURITY. <i>[comment-entry]</i>]</li> </ul>	N	N		

## Environment Division

### Input-output section

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
FILE-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program Reads Data Port, Writes Data Port depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [[ORGANIZATION [IS]] SEQUENTIAL] [PADDING [CHARACTER] [IS] < <i>data-name-5</i>   <i>literal-2</i> >] [RECORD DELIMITER [IS] <STANDARD-1   <i>assignment-name-2</i> >] [ACCESS [MODE] [IS] SEQUENTIAL] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: indexed file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program Reads Data Port, Writes Data Port depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] INDEXED [ACCESS [MODE] [IS] <SEQUENTIAL   RANDOM   DYNAMIC>] RECORD [KEY] [IS] <i>data-name-2</i> [PASSWORD [IS] <i>data-name-6</i> ] ALTERNATE RECORD [KEY] [IS] <i>data-name-3</i> [[WITH] DUPLICATES] [PASSWORD [IS] <i>data-name-7</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: relative file-control entries</b></li> </ul> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN <[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i> >	V	S	defines program-name.assignment-name-1 (File)	Program Reads Data Port, Writes Data Port depending on OPEN statement
	[RESERVE <i>integer</i> [AREA   AREAS]] [ORGANIZATION [IS]] RELATIVE [ACCESS [MODE] [IS] <SEQUENTIAL [RELATIVE [KEY] [IS] <i>data-name-4</i> ]   <RANDOM   DYNAMIC> RELATIVE [KEY] [IS] <i>data-name-4</i> >] [PASSWORD [IS] <i>data-name-6</i> ] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		

**1-282** Supported COBOL Statements  
Supported SIEMENS COBOL statements

Paragraphs and entries	Format	Parser	IA	Repository	
				Entities	Relation
	<ul style="list-style-type: none"> <li><b>Format 4: line-sequential file-control entries (all platforms except VM)</b> FILE-CONTROL. SELECT [OPTIONAL] <i>file-name-1</i> ASSIGN &lt;[TO] <i>assignment-name-1</i>   USING <i>data-name-9</i>&gt;</li> </ul>	N	N	defines program-name.assignment-name-1 (File)	Program Reads Data Port, Writes Data Port depending on OPEN statement
	[ORGANIZATION [IS]] LINE SEQUENTIAL [ACCESS [MODE] [IS] SEQUENTIAL] [[FILE] STATUS [IS] <i>data-name-1</i> [ <i>data-name-8</i> ]].	VO	N		
I-O-CONTROL paragraph	<ul style="list-style-type: none"> <li><b>Format 1: sequential I-O</b> I-O-CONTROL. &lt;RERUN [ON &lt;<i>assignment-name-1</i>   <i>file-name-1</i>&gt;] [EVERY] &lt;<i>integer-1</i> RECORDS   END [OF] &lt;REEL   UNIT&gt;&gt; [OF] <i>file-name-1</i>   SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i>   MULTIPLE FILE [TAPE] [CONTAINS] <i>file-name-5</i> [POSITION] <i>integer-2</i>   APPLY WRITE-ONLY [ON] <i>file-name-2</i>&gt;.</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: relative and indexed I-O</b> I-O-CONTROL. &lt;RERUN ON &lt;<i>assignment-name-1</i>   <i>file-name-1</i>&gt; [EVERY] <i>integer-1</i> RECORDS [OF] <i>file-name-1</i>   SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i>&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential I-O</b> I-O-CONTROL. SAME [RECORD] [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i>.</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 4: sort-merge I-O</b> I-O-CONTROL. [RERUN [ON] &lt;<i>file-name-1</i>   <i>assignment-name-1</i>&gt; [[EVERY] SORT [OF] &lt;<i>file-name-2</i>   <i>assignment-name-2</i>&gt;]] SAME &lt;RECORD   SORT   SORT-MERGE&gt; [AREA] [FOR] <i>file-name-3</i> <i>file-name-4</i>.</li> </ul>	VO	N		

**Configuration section**

Paragraphs and entries	Format	Parser	IA



Paragraphs and entries	Format	Parser	IA
REPOSITORY paragraph	<ul style="list-style-type: none"> <li><b>Format</b> REPOSITORY. [CLASS <i>class-name-1</i> [[IS] <i>external-class-name-1</i>].</li> </ul>	N	N
SOURCE-COMPUTER paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SOURCE-COMPUTER. [&lt;<i>computer-name</i>   <i>id id</i>   <i>id literal</i> &gt; [[WITH] DEBUGGING MODE].]</li> </ul>	VO	N
SPECIAL-NAMES paragraph	<ul style="list-style-type: none"> <li><b>Format</b> SPECIAL-NAMES. [<i>environment-name-1</i> [IS] <i>mnemonic-name-1</i>   <i>environment-name-2</i> &lt;[IS] <i>mnemonic-name-2 entry-1</i>   <i>entry-1</i>&gt;] where <i>entry-1</i> is: <i>entry-1</i>: &lt;ON [STATUS] [IS] <i>condition-1</i> [OFF [STATUS] [IS] <i>conditional-2</i>   OFF [STATUS] [IS] <i>condition-2</i> [ON [STATUS] [IS] <i>conditional-1</i>]&gt; <i>ARGUMENT-NUMBER</i> [IS] <i>mnemonic-name-3</i> <i>ARGUMENT-VALUE</i> [IS] <i>mnemonic-name-4</i> <i>ENVIRONMENT-NAME</i> [IS] <i>mnemonic-name-5</i> <i>ENVIRONMENT-VALUE</i> [IS] <i>mnemonic-name-6</i></li> </ul>	V	S
	[ALPHABET <i>alphabet-name-1</i> [IS] <STANDARD-1   STANDARD-2   NATIVE   EBCDIC   <i>literal-1</i> [<THROUGH   THRU> <i>literal-2</i>   ALSO <i>literal-3</i> >]	VO	N
	[SYMBOLIC [CHARACTERS] <i>symbolic-character</i> [ARE   IS] <i>integer-1</i> [IN] <i>alphabet-name-2</i> ]	VO	N
	[CLASS <i>class-name-1</i> [IS] <i>literal-4</i> [<THROUGH   THRU> <i>literal-5</i> ]]	VO	N
	[CURRENCY [SIGN] [IS] <i>literal-6</i> [[WITH] PICTURE SYMBOL <i>literal-7</i> ]]	VO	N
	[DECIMAL-POINT [IS] COMMA] [.]	V	S

1-284 Supported COBOL Statements  
Supported SIEMENS COBOL statements

**Data Division**

Entries and clauses	Format	Parser	IA	CA/GDF		
				Sup.	Pairs	Relation
Data Division	<ul style="list-style-type: none"> <li><b>Format: program and method DATA DIVISION</b>            DATA DIVISION.            [FILE SECTION. [file-description-entry record-description-entry]]            [WORKING-STORAGE SECTION. [record-description-entry   data-item-description-entry]]            [LOCAL-STORAGE SECTION. [record-description-entry   data-item-description-entry]]            [LINKAGE SECTION. [record-description-entry   data-item-description-entry]]</li> </ul>	V	S	S		
	<ul style="list-style-type: none"> <li><b>Format: class DATA DIVISION</b>            [WORKING-STORAGE SECTION. [record-description-entry   data-item-description-entry]]</li> </ul>	N	N	N		

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
File Description (FD) Entry	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [&lt;CHARACTER   CHARACTERS&gt;]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [&lt;CHARACTER   CHARACTERS&gt;]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i>] [TO] <i>integer-7</i>] [&lt;CHARACTER   CHARACTERS&gt;] [DEPENDING [ON] <i>data-name-1</i>]]&gt;            [LABEL &lt;RECORD [IS]   RECORDS [ARE]]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;            [VALUE OF <i>ID-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]]&gt; <i>data-name-4</i>]            [LINAGE [IS] &lt;<i>data-name-5</i>   <i>integer-8</i>&gt; [LINES] [[WITH] FOOTING [AT] &lt;<i>data-name-6</i>   <i>integer-9</i>&gt;] [[LINES] [AT] TOP &lt;<i>data-name-7</i>   <i>integer-10</i>&gt;] [[LINES] [AT] BOTTOM &lt;<i>data-name-8</i>   <i>integer-11</i>&gt;]]            [RECORDING [MODE] [IS] <i>mode</i>] [CODE-SET [IS] <i>alphabet-name</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: relative and indexed files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [&lt;CHARACTER   CHARACTERS&gt;]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [&lt;CHARACTER   CHARACTERS&gt;]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i>] [TO] <i>integer-7</i>] [&lt;CHARACTER   CHARACTERS&gt;] [DEPENDING [ON] <i>data-name-1</i>]]&gt;            [LABEL &lt;RECORD [IS]   RECORDS [ARE]]&gt; &lt;STANDARD   OMITTED            [VALUE OF <i>ID-1</i> [IS] &lt;<i>data-name-3</i>   <i>literal-1</i>&gt;]            [DATA &lt;RECORD [IS]   RECORDS [ARE]]&gt; <i>data-name-4</i>].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b></li> </ul> <p>FD <i>file-name-1</i> [[IS] EXTERNAL] [[IS] GLOBAL]            [BLOCK [CONTAINS] [<i>integer-1</i> TO] <i>integer-2</i> &lt;CHARACTERS   RECORDS&gt;]            [RECORD &lt;[CONTAINS] <i>integer-3</i> [&lt;CHARACTER   CHARACTERS&gt;]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i>] [TO] <i>integer-7</i>] [&lt;CHARACTER   CHARACTERS&gt;] [DEPENDING [ON] <i>data-name-1</i>]]&gt;].</p>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 4: sort/merge files</b></li> </ul> <p>SD <i>file-name-1</i>            [RECORD &lt;[CONTAINS] <i>integer-3</i> [&lt;CHARACTER   CHARACTERS&gt;]   [CONTAINS] <i>integer-4</i> TO <i>integer-5</i> [&lt;CHARACTER   CHARACTERS&gt;]   [IS] VARYING [IN] [SIZE] [[FROM] <i>integer-6</i>] [TO] <i>integer-7</i>] [&lt;CHARACTER   CHARACTERS&gt;] [DEPENDING [ON] <i>data-name-1</i>]]&gt;            [DATA &lt;RECORD [IS]   RECORDS [ARE]]&gt; <i>data-name-4</i>]            [LABEL &lt;RECORD [IS]   RECORDS [ARE]]&gt; &lt;STANDARD   OMITTED   <i>data-name-2</i>&gt;            [RECORDING [MODE] [IS] <i>mode</i></p>	V	S	S			

1-286 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Data Description Entry	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>level-number &lt;data-name-1   FILLER&gt; [redefines-clause] [blank-when-zeros-clause] [dynamic-clause] [external-clause] [global-clause] [justified-clause] [occurs-clause] [picture-clause] [sign-clause] [synchronized-clause] [usage-clause] [value-clause] [data-format-clause] [dynamic-clause]</i></li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>66 data-name-1 renames-clause.</i></li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>88 condition-name value-clause.</i></li> </ul>	V	S	S			
BLANK WHEN ZEROS/ZEROES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>BLANK [WHEN] &lt;ZERO   ZEROS   ZEROES&gt;</i></li> </ul>	N	N	N			
DATE FORMAT clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>DATE FORMAT [IS] date-pattern</i></li> </ul>	VO	N	N			
EXTERNAL clause	<ul style="list-style-type: none"> <li><b>Format: [IS] EXTERNAL</b></li> </ul>	V	S	N			
DYNAMIC clause	<ul style="list-style-type: none"> <li><b>Format: [IS] DYNAMIC</b></li> </ul>	N	N	N			
GLOBAL clause		V	S	N			
JUSTIFIED clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>&lt;JUSTIFIED   JUST&gt; [RIGHT]</i></li> </ul>	VO	N	N			
OCCURS clause	<ul style="list-style-type: none"> <li><b>Format 1: fixed-length tables</b> <i>OCCURS integer-2 [TIMES] [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: variable-length tables</b> <i>OCCURS integer-1 TO integer-2 [TIMES] DEPENDING [ON] data-name-1 [&lt;ASCENDING   DESCENDING&gt; [KEY] [IS] data-name-2] [INDEXED [BY] index-name-1]</i></li> </ul>	V	S	S			

Entries and clauses	Format	Parser	IA	Sup.	CA/GDF		
					Pairs		Relation
PICTURE clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;PICTURE   PIC&gt; [IS] <i>character-string</i></li> </ul>	V	S	S			
REDEFINES clause	<ul style="list-style-type: none"> <li><b>Format</b> <i>level-number</i> &lt;<i>data-name-1</i>   FILLER&gt; REDEFINES <i>data-name-2</i></li> </ul>	V	S	S			
RENAMES clause	<ul style="list-style-type: none"> <li><b>Format</b> 66 <i>data-name-1</i> RENAMES <i>data-name-2</i> [&lt;THROUGH   THRU&gt; <i>data-name-3</i>]</li> </ul>	V	S	S			
SIGN clause	<ul style="list-style-type: none"> <li><b>Format</b> SIGN [IS] &lt;LEADING   TRAILING&gt;[SEPARATE CHARACTER]</li> </ul>	V	S	S			
SYNCHRONIZED clause	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SYNCHRONIZED   SYNC&gt; [LEFT   RIGHT]</li> </ul>	VO	N	N			
USAGE clause	<ul style="list-style-type: none"> <li><b>Format 1: standard</b> USAGE [IS] &lt;BINARY   COMP   COMP-1   COMP-2   COMP-3   COMP-4   COMP-5   COMPUTATIONAL   COMPUTATIONAL-1   COMPUTATIONAL-2   COMPUTATIONAL-3   COMPUTATIONAL-4   COMPUTATIONAL-5   DISPLAY   DISPLAY-1   INDEX    PACKED-DECIMAL   POINTER &gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 2: with 'NATIVE'</b> USAGE [IS] &lt;BINARY NATIVE   COMP   COMP-1 NATIVE   COMP-2 NATIVE   COMP-3   COMP-4 NATIVE   COMP-5   COMPUTATIONAL   COMPUTATIONAL-1 NATIVE   COMPUTATIONAL-2 NATIVE   COMPUTATIONAL-3   COMPUTATIONAL-4 NATIVE   COMPUTATIONAL-5   DISPLAY NATIVE   DISPLAY-1 NATIVE   INDEX   PACKED-DECIMAL   POINTER   PROCEDURE-POINTER&gt;</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 3: PROCEDURE-POINTER</b> USAGE [IS] PROCEDURE-POINTER</li> </ul>	VO	N	N			
	<ul style="list-style-type: none"> <li><b>Format 4: OBJECT REFERENCE</b> USAGE [IS] OBJECT REFERENCE [[METACLASS [OF]] <i>class-name-1</i>]</li> </ul>	N	N	N			

**1-288** Supported COBOL Statements  
Supported *SIEMENS COBOL* statements

Entries and clauses	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
VALUE clause	<ul style="list-style-type: none"> <li> <b>Format 1: literal value</b>            &lt;VALUE   VALUES&gt; [FROM <i>subscript-1</i>] [&lt;IS   ARE&gt;] <i>literal-1</i> [REPEATED &lt;<i>integer-1</i> TIMES   [TO] END&gt;]         </li> </ul>	V	S	S	<i>literal</i>	<i>corresponding variable</i>	const.init
	<ul style="list-style-type: none"> <li> <b>Format 2: condition-name value</b>            88 <i>condition-name-1</i> &lt;VALUE [IS]   VALUES [ARE]&gt; <i>literal-1</i> [&lt;THROUGH   THRU&gt; <i>literal-2</i>].            88 <i>condition-name-2</i> &lt;[when] set to false&gt;         </li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li> <b>Format 3: NULL value</b>            VALUE [IS] &lt;NULL   NULLS&gt;         </li> </ul>	V	S	S			

**Procedure Division**

**Arithmetic operators**

Operation	Meaning	Format	Parser	IA	Sup.	CA/GDF		
						Pairs		Relation
+	Addition	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 + oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
-	Subtraction	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 - oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp+
*	Multiplication	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 * oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
/	Division	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 / oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
**	Exponentiation	<ul style="list-style-type: none"> <li><b>Format</b> <i>oper-1 ** oper-2</i></li> </ul>	V	S	S	<i>oper-1</i> <i>oper-2</i>	<i>result</i>	comp*
+	Multiplication by +1	<ul style="list-style-type: none"> <li><b>Format</b> <i>+ oper</i></li> </ul>	V	S	S			
-	Multiplication by -1	<ul style="list-style-type: none"> <li><b>Format</b> <i>- oper</i></li> </ul>	V	S	S			

1-290 Supported COBOL Statements  
Supported SIEMENS COBOL statements

**Conditional Expressions**

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pair	Relation	
Class condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>id-1</i> [IS] [NOT] &lt;NUMERIC   ALPHABETIC   ALPHABETIC-LOWER   ALPHABETIC-UPPER   <i>class-name</i>   DBCS   KANJI&gt;</li> </ul>	V	S	S			
		VO	N	N			
Condition-name condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Relation condition	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>operand-1</i> [IS] &lt;[NOT] &lt;GREATER [THAN]   LESS [THAN]   EQUAL [TO] &gt;   GREATER [THAN] OR EQUAL [TO]   LESS [THAN] OR EQUAL [TO]&gt; <i>operand-2</i></li> </ul>	V	S	S	<i>operand-1</i>	<i>operand-2</i>	cond
	<ul style="list-style-type: none"> <li><b>Format 2: pointer data items</b> &lt;ADDRESS OF <i>id-1</i>   <i>id-2</i>   NULL   NULLS&gt; [IS] [NOT] EQUAL [TO] &lt;ADDRESS OF <i>id-3</i>   <i>id-4</i>   NULL   NULLS&gt;</li> </ul>	V	S	S			
	<ul style="list-style-type: none"> <li><b>Format 3: procedure-pointer data items</b> <i>[id-2</i>   NULL   NULLS] [IS] [NOT] EQUAL [TO] [ <i>id-4</i>   NULL   NULLS]</li> </ul>	V	S	N			
	<ul style="list-style-type: none"> <li><b>Format 4: object reference data items</b> &lt;<i>object-reference-id-1</i>   SELF   NULL   NULLS&gt; [IS] [NOT] EQUAL [TO] &lt;<i>object-reference-id-2</i>   SELF   NULL   NULLS&gt;</li> </ul>	N	N	N			
Sign condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [IS] [NOT] &lt;POSITIVE   NEGATIVE   ZEROS   ZEROES&gt;</li> </ul>	V	S	S			
Switch-status condition	<ul style="list-style-type: none"> <li><b>Format</b> condition-name</li> </ul>	V	S	S			
Negated simple condition	<ul style="list-style-type: none"> <li><b>Format</b> NOT <i>condition-1</i></li> </ul>	V	S	S			
Combined condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>condition-1</i> &lt;AND   OR&gt; <i>condition-2</i></li> </ul>	V	S	S			



Conditions	Format	Parser	IA	Sup.	CA/GDF		
					Pair		Relation
Abbreviated combined relation condition	<ul style="list-style-type: none"> <li><b>Format</b> <i>relation-condition</i> &lt;AND   OR&gt; [NOT] [<i>relational-operator</i>] <i>object</i></li> </ul>	V	S	S			

### Statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PROCEDURE DIVISION header	<ul style="list-style-type: none"> <li><b>Format 1: programs and methods (with BY REFERENCE)</b> PROCEDURE DIVISION [USING [[BY] REFERENCE] <i>data-name-1</i>] [RETURNING <i>data-name-2</i>].</li> </ul>	V	S	S	<i>port</i>	<i>data-name-1</i>	extern. using		
					<i>data-name-1</i>	<i>port</i>	extern. using		
					<i>data-name-1</i>	<i>port</i>	extern. using		
	<ul style="list-style-type: none"> <li><b>Format 2: programs and methods (with BY VALUE)</b> PROCEDURE DIVISION [USING [[BY] VALUE] <i>data-name-1</i>] [RETURNING <i>data-name-2</i>].</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: classes</b> PROCEDURE DIVISION.</li> </ul>	N	N	N					
PROCEDURE DIVISION structure	<ul style="list-style-type: none"> <li><b>Format 1: program and method</b> procedure division header [DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. [<i>sentence</i>]] END DECLARATIVES.] <i>section-name</i> SECTION [<i>priority-number</i>]. [<i>paragraph-name</i>. [<i>sentence</i>]]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: classes</b> PROCEDURE DIVISION. [<i>method-definition</i>]</li> </ul>	N	N	N					

1-292 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DECLARATIVES section	<ul style="list-style-type: none"> <li><b>Format</b> DECLARATIVES. <i>section-name</i> SECTION [<i>priority-number</i>]. USE [<i>paragraph-name</i>. <i>sentence</i>] END DECLARATIVES.</li> </ul>	V	S	S					
ACCEPT	<ul style="list-style-type: none"> <li><b>Format 1: data transfer</b> ACCEPT <i>id</i> [FROM &lt;<i>mnemonic-name</i>   <i>environment-name</i>&gt;] [&lt;[ON] EXCEPTION <i>imperative-statement-1</i>   NOT [ON] EXCEPTION <i>imperative-statement-2</i>&gt;] [END-ACCEPT]</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen		
	<ul style="list-style-type: none"> <li><b>Format 2: system information transfer</b> ACCEPT <i>id</i> FROM &lt;DATE [YYYYMMDD]   DAY [YYYYDDD]   DAY-OF-WEEK   TIME&gt;</li> </ul>	V	S	S	<i>port</i>	<i>id</i>	extern. screen		
ADD	<ul style="list-style-type: none"> <li><b>Format 1</b> ADD &lt;<i>id-1</i>   <i>literal</i>&gt; TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
					<i>id-2</i>	<i>id-2</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> ADD &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO &lt;<i>id-2</i>   <i>literal-1</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp+		
					<i>id-2</i>	<i>id-3</i>	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3: with CORRESPONDING</b> ADD &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-ADD]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp+		
					<i>id-2</i>	<i>id-2</i>	comp+		
ALTER	<ul style="list-style-type: none"> <li><b>Format</b> ALTER</li> </ul>	V	S	S					
	<i>procedure-name-1</i> TO [PROCEED TO] <i>procedure-name-2</i>	VO	N	N					
CALL	<ul style="list-style-type: none"> <li><b>Format 1: with ON OVERFLOW</b> CALL &lt;<i>id-1</i>   <i>literal-1</i>&gt; [USING [BY] &lt;REFERENCE [&lt;<i>id-2</i>   <i>file-name-1</i>&gt;]   VALUE [<i>id-3</i>]   CONTENT [&lt;<i>id-4</i>   <i>literal-2</i>&gt;]] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [END-CALL]</li> </ul>	V	S	S	<i>id-2, id-3</i>	<i>port</i>	extern.call	refers to literal-1 (program Entry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
					<i>port</i>	<i>id-2, id-5</i>	extern.call		

Statement	Format	Parser	IA	Sup.	CA/GDF		Repository	
					Pairs	Relation	Entities	Relations
	<ul style="list-style-type: none"> <li><b>Format 2: with ON EXCEPTION</b> CALL &lt;id-1   literal-1&gt; [USING [BY] &lt;REFERENCE [&lt;id-2   file-name-1&gt;]   VALUE [id-3]   CONTENT [&lt;id-4   literal-2&gt;] ] [[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	VO	N	N				
	<ul style="list-style-type: none"> <li><b>Format 3: with CALL UPON SYSTEM</b> CALL UPON SYSTEM USING &lt;id-1   literal-1&gt; [id-2] [STATUS id-3] [[ON] EXCEPTION imperative-stmt-1] [NOT [ON] EXCEPTION imperative-stmt-2] [END-CALL]</li> </ul>	VO	N	N				
CANCEL	<ul style="list-style-type: none"> <li><b>Format</b> CANCEL &lt;id-1   literal-1&gt;</li> </ul>	V	S	S				
CLOSE	<ul style="list-style-type: none"> <li><b>Format 1: sequential</b> CLOSE file-name-1</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li>[&lt;REEL   UNIT&gt; [[FOR] REMOVAL   WITH NO REWIND]   [WITH] &lt;NO REWIND   LOCK&gt;]</li> </ul>	VO	N	N				
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> CLOSE file-name-1</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li>[[WITH] LOCK]</li> </ul>	VO	N	N				
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b> CLOSE filename-1</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li>[&lt;REEL   UNIT&gt; [[FOR] REMOVAL   WITH NO REWIND]   [WITH] &lt;NO REWIND   LOCK&gt;]</li> </ul>	VO	N	N				
COMPUTE	<ul style="list-style-type: none"> <li><b>Format</b> COMPUTE id-1 [ROUNDED] [EQUAL] arithmetic-expr [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-COMPUTE]</li> </ul>	V	S	S	identifiers of the arithmetic-expr	id-1	comp+ or comp* (according to the operation performed)	
CONTINUE	<ul style="list-style-type: none"> <li><b>Format</b> CONTINUE</li> </ul>	V	S	S				

1-294 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE <i>file-name-1</i> [RECORD] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-DELETE]</li> </ul>	V	S	S					
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISPLAY &lt;<i>id-1</i>   <i>integer-1</i>&gt; [UPON <i>mnemonic-name-1</i>] [[ON] EXCEPTION <i>imperative-stmt-1</i>] [NOT [ON] EXCEPTION <i>imperative-stmt-2</i>] [END-DISPLAY]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.screen		
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: INTO</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-1</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: INTO with GIVING</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-3</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 3: BY</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
				<i>id-2</i>	<i>id-3</i>	comp*			
<ul style="list-style-type: none"> <li><b>Format 4: INTO with REMAINDER</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; INTO &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*			
					<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-4</i>	comp*		
					<i>id-2</i>	<i>id-4</i>	comp*		
<ul style="list-style-type: none"> <li><b>Format 5: BY with REMAINDER</b> DIVIDE &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] REMAINDER <i>id-4</i> [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-DIVIDE]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*			
					<i>id-2</i>	<i>id-3</i>	comp*		
					<i>id-1</i>	<i>id-4</i>	comp*		
					<i>id-2</i>	<i>id-4</i>	comp*		

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
ENTRY	<ul style="list-style-type: none"> <li><b>Format 1</b> ENTRY <i>literal-1</i> [USING [[BY] REFERENCE] <i>id-1</i>][ [BY] VALUE] <i>id-1</i>]</li> </ul>	V	S	S	<i>id-1</i>	<i>port</i>	extern.using	defines <i>literal-1</i> (Program Entry Point)	Program Has Program Entry Point
	<ul style="list-style-type: none"> <li><b>Format 2</b> ENTRY <i>literal-1</i> [USING [[BY] VALUE] <i>id-1</i>]</li> </ul>	VO	N	N	<i>port</i>	<i>id-1</i>	extern.using		
EVALUATE	<ul style="list-style-type: none"> <li><b>Format</b> EVALUATE &lt;<i>id-1</i>   <i>literal-1</i>   <i>expr-1</i>   TRUE   FALSE&gt; [ALSO &lt;<i>id-2</i>   <i>literal-2</i>   <i>expr-2</i>   TRUE   FALSE&gt;] WHEN <i>phrase-1</i> [ALSO <i>phrase-2</i>] <i>imperative-stmt-1</i> [WHEN OTHER <i>imperative-stmt-2</i>] [END-EVALUATE] Phrases <i>phrase-1</i> and <i>phrase-2</i> should be represented in the following form: &lt;ANY   <i>condition-1</i>   <i>subcondition-1</i>   TRUE   FALSE   [NOT] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>&gt; [&lt;THROUGH   THRU&gt; &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1, expr-1</i>	<i>condition-1</i>	cond		
EXIT	<ul style="list-style-type: none"> <li><b>Format</b> EXIT [[TO] TEST [OF]] PERFORM</li> </ul>	V	S	S					
EXIT METHOD	<ul style="list-style-type: none"> <li><b>Format</b> EXIT METHOD.</li> </ul>	N	N	N					
EXIT PROGRAM	<ul style="list-style-type: none"> <li><b>Format</b> EXIT PROGRAM.</li> </ul>	V	S	S					
GOBACK	<ul style="list-style-type: none"> <li><b>Format</b> GOBACK</li> </ul>	V	S	S					

1-296 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
GO TO	<ul style="list-style-type: none"> <li><b>Format 1: unconditional</b> GO [TO] <i>procedure-name-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: conditional</b> GO [TO] <i>procedure-name-1</i> DEPENDING [ON] <i>id-1</i></li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: altered</b> <i>paragraph-name</i>. GO [TO].</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: MORE-LABELS</b> GO [TO] MORE-LABELS</li> </ul>	N	N	N					
IF	<ul style="list-style-type: none"> <li><b>Format</b> IF <i>condition-1</i> THEN &lt;<i>stmt-1</i>   NEXT SENTENCE&gt; [ELSE &lt;<i>stmt-2</i>   NEXT SENTENCE&gt;] [END-IF]</li> </ul>	V	S	S					
INITIALIZE	<ul style="list-style-type: none"> <li><b>Format</b> INITIALIZE <i>id-1</i> [REPLACING &lt;ALPHABETIC   ALPHANUMERIC   NUMERIC   ALPHANUMERIC-EDITED   NUMERIC-EDITED   DBCS   EGCS&gt; [DATA] BY &lt;<i>id-2</i>   <i>literal-1</i>&gt;]</li> </ul>	V	S	S					

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs		Relation	Entities	Relations
INSPECT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSPECT <i>id-1</i> REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSPECT <i>id-1</i> TALLYING <i>id-2</i> FOR &lt;CHARACTERS [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]   &lt;ALL   LEADING&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]&gt; REPLACING &lt;CHARACTERS BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]   &lt;ALL   LEADING   FIRST&gt; &lt;<i>id-3</i>   <i>literal-1</i>&gt; BY &lt;<i>id-5</i>   <i>literal-3</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-6</i>   <i>literal-4</i>&gt;]&gt;</li> </ul>	V	S	S	<i>id-1</i>	<i>id-5</i>	comp		
	<ul style="list-style-type: none"> <li><b>Format 4</b> INSPECT <i>id-1</i> CONVERTING &lt;<i>id-6</i>   <i>literal-4</i>&gt; TO &lt;<i>id-7</i>   <i>literal-5</i>&gt; [&lt;BEFORE   AFTER&gt; [INITIAL] &lt;<i>id-4</i>   <i>literal-2</i>&gt;]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-7</i>	comp		
INVOKE	<ul style="list-style-type: none"> <li><b>Format</b> INVOKE &lt;<i>id-1</i>   <i>class-name-1</i>   SELF   [<i>class-name-2</i> OF] SUPER&gt; &lt;<i>literal-1</i>   <i>id-2</i>&gt; [USING &lt;[[BY] REFERENCE] &lt;[ADDRESS OF] <i>id-3</i>   OMITTED&gt;   [[BY] CONTENT] &lt;[ADDRESS OF   LENGTH OF] <i>id-4</i>   <i>literal-2</i>   OMITTED&gt;   [[BY] VALUE] &lt;[ADDRESS OF   LENGTH OF] <i>id-5</i>   <i>literal-3</i>&gt;&gt;] [RETURNING <i>id-6</i>] [[ON] EXCEPTION <i>imperative-stmt-1</i>] [NOT [ON] EXCEPTION <i>imperative-stmt-2</i>] [END-INVOKE]</li> </ul>	N	N	N					

1-298 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
MERGE	<ul style="list-style-type: none"> <li><b>Format</b> MERGE <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [&lt;KEY   KEY-YY&gt;] <i>data-name-1</i></li> </ul>	V	S	N					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<ul style="list-style-type: none"> <li>USING <i>file-name-2 file-name-3</i> &lt;OUTPUT PROCEDURE [IS] <i>procedure-name-1</i> [&lt;THROUGH   THRU&gt; <i>procedure-name-2</i>]   GIVING <i>file-name-4</i>&gt;</li> </ul>	V	S	N					
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE &lt;<i>id-1</i>   <i>literal-1</i>&gt; TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE &lt;CORRESPONDING   CORR&gt; <i>id-1</i> TO <i>id-2</i></li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	move or cast (depending on data types)		
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY <i>id-2</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-2</i>	comp*		
					<i>id-2</i>	<i>id-2</i>	comp*		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b> MULTIPLY &lt;<i>id-1</i>   <i>literal-1</i>&gt; BY &lt;<i>id-2</i>   <i>literal-2</i>&gt; GIVING <i>id-3</i> [ROUNDED] [[ON] SIZE ERROR <i>imperative-stmt-1</i>] [NOT [ON] SIZE ERROR <i>imperative-stmt-2</i>] [END-MULTIPLY]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp*		
					<i>id-2</i>	<i>id-3</i>	comp*		



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
OPEN	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> OPEN &lt;INPUT <i>file-name-1</i> [REVERSED   WITH NO REWIND]   OUTPUT <i>file-name-2</i> [WITH NO REWIND]   I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					Program Reads Data Port, Program Writes Data Port
		VO	N	N					
		V	S	S					
		VO	N	N					
		V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: indexed and relative files</b> OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   I-O <i>file-name-3</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					Program Reads Data Port, Program Writes Data Port
	<ul style="list-style-type: none"> <li><b>Format 3: line-sequential files</b> OPEN &lt;INPUT <i>file-name-1</i>   OUTPUT <i>file-name-2</i>   EXTEND <i>file-name-4</i>&gt;</li> </ul>	V	S	S					Program Reads Data Port, Program Writes Data Port

1-300 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
PERFORM	<ul style="list-style-type: none"> <li><b>Format 1: basic</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2]   imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: with TIMES phrase</b> PERFORM &lt;procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] &lt;id-1   integer-1&gt; TIMES imperative-stmt-1 END-PERFORM&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: with UNTIL phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: with UNTIL phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] UNTIL condition-1</li> </ul>	S	S	S					
	<ul style="list-style-type: none"> <li><b>Format 5: with VARYING phrase and END-PERFORM</b> PERFORM [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 imperative-stmt-1 END-PERFORM</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 6: with VARYING phrase, without END-PERFORM</b> PERFORM procedure-name-1 [&lt;THROUGH   THRU&gt; procedure-name-2] [[WITH] TEST &lt;BEFORE   AFTER&gt;] VARYING &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-1 [AFTER &lt;id-5   index-name-3&gt; FROM &lt;id-6   index-name-4   literal-3&gt; BY &lt;id-7   literal-4&gt; UNTIL condition-2]   &lt;id-2   index-name-1&gt; FROM &lt;id-3   index-name-2   literal-1&gt; BY &lt;id-4   literal-2&gt; UNTIL condition-2</li> </ul>	N	N	N					



1-302 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
SEARCH	<ul style="list-style-type: none"> <li><b>Format 1: serial search</b> SEARCH <i>id-1</i> [VARYING &lt;<i>id-2</i>   <i>index-name-1</i>&gt;] [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>condition-1</i> &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: binary search</b> SEARCH ALL <i>id-1</i> [[AT] END <i>imperative-stmt-1</i>] WHEN &lt;<i>data-name-1</i> [IS] EQUAL [TO] &lt;<i>id-3</i>   <i>literal-1</i>   <i>arithmetic-expr-1</i>   <i>condition-name-1</i>&gt; [AND &lt;<i>data-name-2</i> [IS] EQUAL [TO] &lt;<i>id-4</i>   <i>literal-2</i>   <i>arithmetic-expr-2</i>   <i>condition-name-2</i>&gt;] &lt;<i>imperative-stmt-2</i>   NEXT SENTENCE&gt; [END-SEARCH]</li> </ul>	V	S	S					
SET	<ul style="list-style-type: none"> <li><b>Format 1: basic table handling</b> SET &lt;<i>index-name-1</i>   <i>id-1</i>&gt; TO &lt;<i>index-name-2</i>   <i>id-2</i>   <i>integer-1</i>   ZEROS   SEROES&gt;</li> </ul>	V	S	S	<i>index-name-1</i> <i>id-1</i>	<i>index-name-2</i> <i>id-2</i> <i>integer-1</i>	move		
	<ul style="list-style-type: none"> <li><b>Format 2: adjusting indexes</b> SET <i>index-name-3</i> &lt;UP BY   DOWN BY&gt; &lt;<i>id-3</i>   <i>integer-2</i>&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: external switches</b> SET <i>mnemonic-name-1</i> TO &lt;ON   OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 4: condition-names</b> SET <i>condition-name-1</i> [<i>condition-name-2</i>] TO &lt;TRUE   FALSE&gt;</li> </ul>	V	S	S	<i>condition variable</i> (associated with <i>condition-name-1</i> )	<i>condition value</i> (associated with <i>condition-name-1</i> )	move		
	<ul style="list-style-type: none"> <li><b>Format 5: USAGE IS POINTER data items</b> SET &lt;<i>id-4</i>   ADDRESS OF <i>id-5</i>&gt; TO &lt;<i>id-6</i>   ADDRESS OF <i>id-7</i>   NULL   NULLS&gt;</li> </ul>	V	S	S	<i>id-4</i> <i>id-4</i>	<i>id-6</i> <i>id-7</i>	move comp		
	<ul style="list-style-type: none"> <li><b>Format 6: USAGE IS PROCEDURE-POINTER data items</b> SET <i>procedure-pointer-data-item-1</i> TO &lt;<i>procedure-pointer-data-item-2</i>   ENTRY &lt;<i>id-8</i>   <i>literal-1</i>&gt;   NULL   NULLS   <i>pointer-data-item-3</i>&gt;</li> </ul>	VO	N	N					

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
	<ul style="list-style-type: none"> <li><b>Format 7: USAGE OBJECT REFERENCE data items</b> SET <i>object-reference-id-1</i> TO &lt;<i>object-reference-id-2</i>   NULL   SELF&gt;</li> </ul>	N	N	N					
SORT	<ul style="list-style-type: none"> <li><b>Format</b> SORT <i>file-name-1</i> [ON] &lt;ASCENDING   DESCENDING&gt; [&lt;KEY   KEY-YY&gt;] <i>data-name-1</i> [[WITH] DUPLICATES [IN] [ORDER]]</li> </ul>	V	S	S					
	[[COLLATING] SEQUENCE [IS] <i>alphabet-name-1</i> ]	VO	N	N					
	<USING <i>file-name-2</i>   INPUT PROCEDURE [IS] <i>procedure-name-1</i> [<THROUGH   THRU> <i>procedure-name-2</i> > <GIVING <i>file-name-3</i>   OUTPUT PROCEDURE [IS] <i>procedure-name-3</i> [<THROUGH   THRU> <i>procedure-name-4</i> >	V	S	S					
START	<ul style="list-style-type: none"> <li><b>Format</b> START <i>file-name-1</i> [WITH NO LOCK] [KEY [IS] &lt;EQUAL [TO]   =   LESS [THAN]   &lt;   GREATER [THAN]   &gt;   NOT LESS [THAN]   NOT &lt;   NOT GREATER [THAN]   NOT &gt;   LESS [THAN] OR EQUAL [TO]   &lt;=   GREATER [THAN] OR EQUAL [TO]   &gt;= &gt; <i>data-name-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-START]</li> </ul>	V	S	S					
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP &lt;RUN   <i>literal</i>&gt;</li> </ul>	V	S	S					
STRING	<ul style="list-style-type: none"> <li><b>Format</b> STRING &lt;<i>id-1</i>   <i>literal-1</i>&gt; [DELIMITED [BY]] &lt;<i>id-2</i>   <i>literal-2</i>   SIZE&gt; INTO <i>id-3</i> [[WITH] POINTER <i>id-4</i>] [[ON] OVERFLOW <i>imperative-stmt-1</i>] [NOT [ON] OVERFLOW <i>imperative-stmt-2</i>] [END-STRING]</li> </ul>	V	S	S	<i>id-1</i>	<i>id-3</i>	comp		
					<i>id-2</i>	<i>id-3</i>	comp		

1-304 Supported COBOL Statements  
Supported SIEMENS COBOL statements

Statement	Format	Parser	IA	CA/GDF			Repository		
				Sup.	Pairs	Relation	Entities	Relations	
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b></li> </ul> SUBTRACT <id-1   literal-1> FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: with GIVING</b></li> </ul> SUBTRACT <id-1   literal-1> FROM <id-2   literal-2> GIVING id-3 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-3	comp+		
					id-2	id-3	comp+		
	<ul style="list-style-type: none"> <li><b>Format 3</b></li> </ul> SUBTRACT <CORRESPONDING   CORR> id-1 FROM id-2 [ROUNDED] [[ON] SIZE ERROR imperative-stmt-1] [NOT [ON] SIZE ERROR imperative-stmt-2] [END-SUBTRACT]	V	S	S	id-1	id-2	comp+		
					id-2	id-2	comp+		
UNSTRING	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> UNSTRING id-1 [DELIMITED [BY] [ALL] <id-2   literal-1> [OR [ALL] <id-3   literal-2>]] INTO id-4 [DELIMITER [IN] id-5] [COUNT [IN] id-6] [[WITH] POINTER id-7] [TALLYING [IN] id-8] [[ON] OVERFLOW imperative-stmt-1] [NOT [ON] OVERFLOW imperative-stmt-2] [END-UNSTRING]	V	S	S	id-1	id-4	comp		

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: sequential files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>]   [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move		
					<i>record-name-1</i>	<i>port</i>	extern.file		
	<ul style="list-style-type: none"> <li><b>Format 2: sequential files with 'BEFORE/AFTER'</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [&lt;&lt;BEFORE   AFTER&gt; [ADVANCING] &lt;&lt;<i>id-1</i>   <i>integer-1</i>&gt; [LINE   LINES]   <i>mnemonic-name-1</i>   PAGE&gt;] [[AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-3</i>] [NOT [AT] &lt;END-OF-PAGE   EOP&gt; <i>imperative-stmt-4</i>]   [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	VO	N	N					
	<ul style="list-style-type: none"> <li><b>Format 3: indexed and relative files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [INVALID [KEY] <i>imperative-stmt-1</i>] [NOT INVALID [KEY] <i>imperative-stmt-2</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move		
					<i>record-name-1</i>	<i>port</i>	extern.file		
<ul style="list-style-type: none"> <li><b>Format 4: line-sequential files</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [END-WRITE]</li> </ul>	V	S	S	<i>id-1</i>	<i>record-name-1</i>	move			
				<i>record-name-1</i>	<i>port</i>	extern.file			
<ul style="list-style-type: none"> <li><b>Format 5: line-sequential files with 'AFTER'</b> WRITE <i>record-name-1</i> [FROM <i>id-1</i>] [AFTER [ADVANCING] &lt;<i>id-2</i>   <i>integer-1</i>&gt; [LINE   LINES]   PAGE&gt;] [END-WRITE]</li> </ul>	VO	N	N						

**Intrinsic functions**

Functions	Parser	IA	CA/GDF			
			Sup.	Pairs		Relation
ACOS	V	S	S	operands	result	comp
ANNUITY	V	S	S	operands	result	comp
ASIN	V	S	S	operands	result	comp
ATAN	V	S	S	operands	result	comp
CHAR	V	S	S	operands	result	comp
COS	V	S	S	operands	result	comp
CURRENT-DATE	V	S	S	operands	result	comp
DATE-OF-INTEGER	V	S	S	operands	result	comp
DATE-TO-YYYYMMDD	V	S	S	operands	result	comp
DATEVAL	V	S	S	operands	result	comp
DAY-OF-INTEGER	V	S	S	operands	result	comp
DAY-TO-YYYYDDD	V	S	S	operands	result	comp
FACTORIAL	V	S	S	operands	result	comp
INTEGER	V	S	S	operands	result	comp
INTEGER-OF-DATE	V	S	S	operands	result	comp
INTEGER-OF-DAY	V	S	S	operands	result	comp
INTEGER-PART	V	S	S	operands	result	comp
LENGTH	V	S	S	operands	result	comp
LOG	V	S	S	operands	result	comp
LOG10	V	S	S	operands	result	comp
LOWER-CASE	V	S	S	operands	result	comp



Functions	Parser	IA	CA/GDF				
			Sup.	Pairs		Relation	
				operands	result		
MAX	V	S	S	operands	result	comp	
MEAN	V	S	S	operands	result	comp	
MEDIAN	V	S	S	operands	result	comp	
MIDRANGE	V	S	S	operands	result	comp	
MIN	V	S	S	operands	result	comp	
MOD	V	S	S	operands	result	comp	
NUMVAL	V	S	S	operands	result	comp	
NUMVAL-C	V	S	S	operands	result	comp	
ORD	V	S	S	operands	result	comp	
ORD-MAX	V	S	S	operands	result	comp	
ORD-MIN	V	S	S	operands	result	comp	
PRESENT-VALUE	V	S	S	operands	result	comp	
RANDOM	V	S	S	operands	result	comp	
RANGE	V	S	S	operands	result	comp	
REM	V	S	S	operands	result	comp	
REVERSE	V	S	S	operands	result	comp	
SIN	V	S	S	operands	result	comp	
SQRT	V	S	S	operands	result	comp	
STANDARD-DEVIATION	V	S	S	operands	result	comp	
SUM	V	S	S	operands	result	comp	
TAN	V	S	S	operands	result	comp	
UNDATE	V	S	S	operands	result	comp	
UPPER-CASE	V	S	S	operands	result	comp	

**1-308** Supported COBOL Statements  
Supported SIEMENS COBOL statements

Functions	Parser	IA	Sup.	CA/GDF		
				Pairs		Relation
				operands	result	
VARIANCE	V	S	S	operands	result	comp
WHEN-COMPILED	V	S	S	operands	result	comp
YEAR-TO-YYYY	V	S	S	operands	result	comp
YEARWINDOW	V	S	S	operands	result	comp

**Compile-directing statements and directives**

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
BASIS	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] BASIS &lt;basis-name   literal-1&gt;</li> </ul>	VO	N		
CBL (PROCESS)	<ul style="list-style-type: none"> <li><b>Format</b> &lt;CBL   PROCESS&gt; [options-list]</li> </ul>	VO	N		
CALLINTER-FACE directive	<ul style="list-style-type: none"> <li><b>Format</b> &lt; &gt;&gt;CALLINTERFACE   &gt;&gt;CALLINT &gt; [SYSTEM   OPTLINK   FAR16   PASCAL16   CDECL] [DESC   DESCRIPTOR   NODESC   NODESCRIPTOR]</li> </ul>	VO	N		
*CONTROL (*CBL)	<ul style="list-style-type: none"> <li><b>Format</b> &lt;*CONTROL   *CBL&gt; &lt;SOURCE   NOSOURCE   LIST   NOLIST   MAP   NOMAP&gt;</li> </ul>	VO	N		
COPY	<ul style="list-style-type: none"> <li><b>Format</b> COPY &lt;text-name-1   literal-1&gt; [&lt;OF   IN&gt; &lt;library-name   literal-2&gt;] [SUPPRESS] [REPLACING operand-1 BY operand-2]</li> </ul>	V	S	refers to text-name-1 (Copybook)	Cobol Includes Copybook
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> [sequence-number] DELETE sequence-number-field</li> </ul>	VO	N		
EJECT	<ul style="list-style-type: none"> <li><b>Format</b> EJECT [.]</li> </ul>	VO	N		

Statements and directives	Format	Parser	IA	Repository	
				Entities	Relations
ENTER	<ul style="list-style-type: none"> <li><b>Format</b> ENTER <i>language-name-1</i> [<i>routine-name-1</i>].</li> </ul>	VO	N		
INSERT	<ul style="list-style-type: none"> <li><b>Format</b> [<i>sequence-number</i>] INSERT <i>sequence-number-field</i></li> </ul>	VO	N		
READY or RESET TRACE	<ul style="list-style-type: none"> <li><b>Format</b> &lt;READY   RESET&gt; TRACE.</li> </ul>	VO	N		
REPLACE	<ul style="list-style-type: none"> <li><b>Format 1</b> REPLACE ==<i>pseudo-text-1</i>== BY ==<i>pseudo-text-2</i>==.</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPLACE OFF.</li> </ul>	V	S		
SERVICE LABEL	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE LABEL</li> </ul>	VO	N		
SERVICE RELOAD	<ul style="list-style-type: none"> <li><b>Format</b> SERVICE RELOAD <i>id-1</i></li> </ul>	VO	N		
SKIP1/2/3	<ul style="list-style-type: none"> <li><b>Format</b> &lt;SKIP1   SKIP2   SKIP3&gt;.</li> </ul>	VO	N		
TITLE	<ul style="list-style-type: none"> <li><b>Format</b> TITLE <i>literal</i>.</li> </ul>	VO	N		
USE	<ul style="list-style-type: none"> <li><b>Format 1: EXCEPTION ERROR declarative</b> USE [GLOBAL] AFTER [STANDARD] &lt;EXCEPTION   ERROR&gt; PROCEDURE [ON] &lt;<i>file-name-1</i>   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		
	<ul style="list-style-type: none"> <li><b>Format 2: LABEL declarative</b> USE [GLOBAL] AFTER [STANDARD] [BEGINNING   ENDING] [FILE   REEL   UNIT] LABEL PROCEDURE [ON] &lt;<i>file-name-1</i>   INPUT   OUTPUT   I-O   EXTEND&gt;</li> </ul>	VO	N		

**1-310** Supported COBOL Statements  
*Supported SIEMENS COBOL statements*

# Supported Natural Statements



This chapter contains detailed information on ATW parser, HyperView, Change Analyzer, and Global Data Element Flow support for Natural Programming Language statements. For a list of supported legacy versions, refer to the *Release Notes*.

## Key to tables

The tables below contain detailed information on ATW parser, HyperView, Change Analyzer, Global Data Element Flow, and repository support. The first column of each table shows Natural keywords. Next, if needed, in the 'Format' column, the possible forms of usage for each keyword are explained. These forms are represented in the standard grammar notation, where

- the optional arguments are written in square brackets '[arguments]';
- the arguments which cannot be omitted are listed in angle brackets <arguments>, and
- the vertical line '|' means 'or' logical connective.

In this way, the notation <arg1 | arg2 | arg3> means that one of the arg1, arg2 or arg3 is required. Other columns provide information for each particular tool.

- **Pars.** — ATW Parser. The possible values in this column are:
  - (V)erified — parsed and prepared for further processing. Possibly is supported by other ATW tools. Does not initiate any errors or warnings during the verification phase.
  - (V)erification (O)nly — parsed and skipped immediately. Further processing is not possible, i.e. not supported by ATW tools. Verification warning is initiated.
  - (N)ot Verified — if the ATW parser meets such entity, it stops and verification is considered as erroneous.
- **IA** — Interactive Analysis. Interactive Analysis (S)upport means that the entity will be displayed in the HyperView window and all the HyperView analysis tools will be available. In fact, every entity which is verified by the ATW parser is supported in HyperView. If the entity is (N)ot supported then it is either (N)ot Verified or is skipped by parser without creating an appropriate node in the parser tree. In this way, 'N' in the IA column corresponds to (N) or (VO) in the parser column.
- **CA and GDF** — Change Analyzer and Global Data Element Flow. The 'Sup.' subcolumn in this column describes general support in CA and GDF. Similarly to HyperView, it contains 'S' for (S)upported statements and 'N' for the ones, which are (N)ot supported. Next subcolumn labeled 'Pairs' describes the possible pairs of the arguments, which can be joined by some program level relationship. The last column describes the relationship itself. The possible relationships and the explanation what they are given below:
  - 'const.move' — for literal assignments: MOVE literal TO var
  - 'const.init' — for initial values at declarations: 1 var (A3) INIT <'ABC'>
  - 'const.comp' — for other manipulations with literals: COMPRESS 'str1' 'str2' INTO var
  - 'const.cond' — for conditions with literals: IF a = 0
  - 'move' — for memory-to-memory assignments: MOVE var1 TO var2
  - 'cast' — for assignments with datatype conversion: MOVE num TO str

- ‘comp’ — for operations, where data flow exists, but operands are not synonyms: COMPRESS s1, s2 INTO ss; exponentiation operation
- ‘comp+’ — for statements like +, -: COMPUTE c = a + b
- ‘comp\*’ — for statements like \*, /: COMPUTE c = a \* b
- ‘comp@’ — for operations with array elements:

```
01 arr (1:5)
    02 elem (A1)
MOVE val TO elem(idx)
```

Two relations should be generated: Val - move(cast) -> Elem; Val - comp@ -> Arr

- ‘cond’ — for conditions, where no data flow exist, but operands are synonyms: IF a = b
- ‘cond\*’ — for conditions using \*, / operations: IF a \* x = b

**Note:** For more detailed information about relationships in CA and GDF refer to *Analyzing Projects* in the ATW document set.

- **Repository** — describes the entities that can occur in repository and possible relationships between them.

## Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements

### Constants

Constant type	Format	Parser	IA	CA/GDF
Alphanumeric constants	<ul style="list-style-type: none"> <li><b>Format 1: with quotes</b> Example: "THIS ISN'T WRONG"</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: with apostrophes</b> Example: 'THIS ISN'T WRONG'</li> </ul>	V	S	S
Numeric constants	<ul style="list-style-type: none"> <li><b>Format 1: fixed-point</b> Example: -2.71828</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2: floating-point</b> &lt;+   -&gt; mantissa E &lt;+   -&gt; exponent</li> </ul>	V	S	S
Hexadecimal constants	<ul style="list-style-type: none"> <li><b>Format</b> Example: H'A1A2A3'</li> </ul>	V	S	S
Logical constants	<ul style="list-style-type: none"> <li><b>Format</b> &lt;TRUE   FALSE&gt;</li> </ul>	V	S	S
Date and time constants	<ul style="list-style-type: none"> <li><b>Format 1</b> Example: D'DD.MM.YYYY'</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> Example: T'hh:mm:ss'</li> </ul>	V	S	S
	<ul style="list-style-type: none"> <li><b>Format 3</b> E'DD.MM.YYYY hh:mm:ss'</li> </ul>	N	N	N
Attribute constants	<ul style="list-style-type: none"> <li><b>Format</b> ([AD-D   AD-B   AD-I   AD-N   AD-V   AD-U   AD-C   AD-Y   AD-P] [CD-BL   CD-GR   CD-NE   CD-PI   CD-RE   CD-TU   CD-YE])</li> </ul>	N	N	N



**System variables**

Variable	Parser	IA	CA/GDF
*APPLIC-ID	V	S	S
*APPLIC-NAME	V	S	S
*COM	V	S	S
*CONTROL	N	N	N
*CONVID	V	S	S
*COUNTER	V	S	S
*CPU-TIME	N	N	N
*CURS-COL	V	S	S
*CURS-FIELD	V	S	S
*CURS-LINE	V	S	S
*CURSOR	V	S	S
*DATA	V	S	S
*DEVICE	V	S	S
*DIALOG-ID	V	S	S
*ERROR-LINE	V	S	S
*ERROR-NR	V	S	S
*ERROR-TA	V	S	S
*ETID	V	S	S
*EVENT	V	S	S
*GROUP	V	S	S
*HARDCOPY	V	S	S

**2-6** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

<b>Variable</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
*HARDWARE	N	N	N
*HOSTNAME	N	N	N
*INIT-ID	V	S	S
*INIT-PROGRAM	V	S	S
*INIT-USER	V	S	S
*ISN	V	S	S
*LANGUAGE	V	S	S
*LENGTH	N	N	N
*LEVEL	V	S	S
*LIBRARY-ID	V	S	S
*LINE-COUNT	V	S	S
*LINESIZE	V	S	S
*LOG-LS	N	N	N
*LOG-PS	N	N	N
*MACHINE-CLASS	N	N	N
*NATVERS	N	N	N
*NET-USER	N	N	N
*NUMBER	V	S	S
*OCCURRENCE	V	S	S
*OPSYS	N	N	N
*OS	N	N	N
*OSVERS	N	N	N

Variable	Parser	IA	CA/GDF
*PAGESIZE	V	S	S
*PAGE-NUMBER	V	S	S
*PARM-USER	N	N	N
*PATCH-LEVEL	N	N	N
*PF-KEY	V	S	S
*PF-NAME	V	S	S
*PID	N	N	N
*PROGRAM	V	S	S
*ROWCOUNT	N	N	N
*SCREEN-IO	N	N	N
*SERVER-TYPE	N	N	N
*STARTUP	V	S	S
*STEPLIB	V	S	S
*SUBROUTINE	N	N	N
*THIS-OBJECT	N	N	N
*TPSYS	N	N	N
*UI	N	N	N
*USER	V	S	S
*USER-NAME	V	S	S
*WINDOW-LS	N	N	N
*WINDOW-POS	N	N	N
*WINDOW-PS	N	N	N

**2-8** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Variable	Parser	IA	CA/GDF
*WINMGR	N	N	N
*WINMGRVERS	N	N	N

***Date and time system variables***

Variable	Parser	IA	CA/GDF
*DATD	V	S	S
*DAT4D	N	N	N
*DATE	V	S	S
*DAT4E	V	S	S
*DATG	V	S	S
*DATI	V	S	S
*DAT4I	N	N	N
*DATJ	V	S	S
*DAT4J	N	N	N
*DATN	V	S	S
*DATU	V	S	S
*DAT4U	N	N	N
*DATV	N	N	N
*DATVS	N	N	N
*DATX	V	S	S

**Session parameters**

Parameter	Parser	IA	CA/GDF
AD	V	S	S
AL	V	S	S
BX	V	S	S
CC	V	S	S
CD	V	S	S
CF	V	S	S
CO	V	S	S
CV	V	S	S
DC	V	S	S
DF	V	S	S
DFOUT	V	S	S
DFSTACK	V	S	S
DFTITLE	V	S	S
DU	V	S	S
DY	V	S	S
EJ	V	S	S
EM	V	S	S
ES	V	S	S
FC	V	S	S
FCDP	V	S	S

Parameter	Parser	IA	CA/GDF
FL	V	S	S
FS	V	S	S
GC	V	S	S
HC	V	S	S
HE	V	S	S
HW	V	S	S
IA	V	S	S
IC	V	S	S
ID	V	S	S
IM	V	S	S
IP	V	S	S
IS	V	S	S
KD	V	S	S
LC	V	S	S
LE	V	S	S
LS	V	S	S
LT	V	S	S
MC	V	S	S
MP	V	S	S
MS	V	S	S

Parameter	Parser	IA	CA/GDF
MT	V	S	S
NC	V	S	S
NL	V	S	S
OPF	V	S	S
PC	V	S	S
PD	V	S	S
PM	V	S	S
PS	V	S	S
REINP	V	S	S
SA	V	S	S
SF	V	S	S
SG	V	S	S
SL	V	S	S
SM	V	S	S
TC	V	S	S
TS	V	S	S
UC	V	S	S
WH	V	S	S
ZD	V	S	S
ZP	V	S	S

**2-10** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

**System functions**

System function	Parser	IA	CA/GDF
AVER	V	S	S
COUNT	V	S	S
MAX	V	S	S
MIN	V	S	S
NAVER	V	S	S
NCOUNT	V	S	S
NMIN	V	S	S
OLD	V	S	S
POS	V	S	S
RET	V	S	S
SORTKEY	N	N	N
SUM	V	S	S
TOTAL	V	S	S

Mathematical function	Parser	IA	CA/GDF
ABS	V	S	S
ATN	V	S	S
COS	V	S	S
EXP	V	S	S
FRAC	V	S	S
INT	V	S	S
LOG	V	S	S
SGN	V	S	S
SIN	V	S	S
SQRT	V	S	S
TAN	V	S	S
VAL	V	S	S

### Conditional expressions

Conditions	Format	Parser	IA	CA/GDF			
				Sup.	Pairs	Relation	
Relation condition	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>operand-1</i> &lt;=   EQ   EQUAL [TO]   &lt;&gt;   NE   NOT =   NOT EQ   NOT EQUAL [TO]   &lt;   LT   LESS THAN   &lt;=   LE   LESS EQUAL   &gt;   GT   GREATER THAN   &gt;=   GE   GREATER EQUAL &gt; <i>operand-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>const.cond</i> <i>or cond+</i> <i>or cond*</i>
	<ul style="list-style-type: none"> <li><b>Format 2: with SUBSTRING</b> &lt;SUBSTRING (<i>operand-1</i>, <i>op-3</i>, <i>op-4</i>)   <i>operand-1</i>&gt; relation condition &lt;SUBSTRING (<i>operand-2</i>, <i>op-5</i>, <i>op-6</i>)   <i>operand-2</i>&gt;</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>const.cond</i> <i>or cond+</i> <i>or cond*</i>
	<ul style="list-style-type: none"> <li><b>Format 3: extended</b> <i>op-1</i> &lt;=   EQ   EQUAL [TO]&gt; <i>op-2</i> &lt;OR &lt;=   EQ   EQUAL [TO]&gt; <i>op-3</i>   THRU <i>op-4</i> [BUT NOT <i>op-5</i> [THRU <i>op-6</i>]]&gt;</li> </ul>	V	S	S	<i>op-1</i> <i>op-1</i> <i>op-1</i> <i>op-1</i> <i>op-1</i>	<i>op-2</i> <i>op-3</i> <i>op-4</i> <i>op-5</i> <i>op-6</i>	<i>const.cond</i> <i>or cond+</i> <i>or cond*</i>
MASK option	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> &lt;=   EQ   EQUAL [TO]   NE   NOT EQUAL &gt; MASK (<i>mask-definition</i>) [<i>operand-2</i>]</li> </ul>	V	S	N			
SCAN option	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> &lt;=   EQ   EQUAL [TO]   NE   NOT EQUAL &gt; SCAN <i>operand-2</i></li> </ul>	V	S	N			
IS option	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> IS (<i>format</i>)</li> </ul>	V	S	N			
Logical variable evaluation	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i></li> </ul>	V	S	N			
Modified control variables	<ul style="list-style-type: none"> <li><b>Format</b> <i>operand-1</i> [NOT] MODIFIED</li> </ul>	V	S	N			
SPECIFIED option	<ul style="list-style-type: none"> <li><b>Format</b> <i>parameter-name</i> [NOT] SPECIFIED</li> </ul>	V	S	N			
Complex logical expressions	<ul style="list-style-type: none"> <li><b>Format</b> [NOT] &lt;<i>logical-condition-criterion</i>   (<i>logical-expression</i>)&gt; [&lt;OR   AND&gt; <i>logical-expression</i>]</li> </ul>	V	S	N			

**2-12** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

**Statements**

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
ACCEPT	<ul style="list-style-type: none"> <li><b>Format</b> ACCEPT [IF] <i>logical-condition</i></li> </ul>	V	S	S					
ADD	<ul style="list-style-type: none"> <li><b>Format</b> ADD [ROUNDED] <i>operand-1</i> ... TO <i>operand-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp+ const.comp</i>		
ASSIGN	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> &lt;ASSIGN [ROUNDED] <i>op-1</i> = &lt;<i>arithmetic-expression</i>   <i>op-2</i>&gt;   <i>op-1</i> := &lt;<i>arithmetic-expression</i>   <i>op-2</i>&gt;&gt;</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move, cast, comp+, comp*, comp</i>		
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [ASSIGN [ROUNDED]] <i>op-1</i> = &lt;<i>arithmetic-expression</i>   <i>op-2</i>&gt;</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move, cast, comp+, comp*, comp</i>		
AT BREAK	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> [AT] BREAK [(<i>r</i>)] [OF] <i>op-1</i> [/n/] <i>statement</i>... END-BREAK</li> </ul>	V	S	N					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [AT] BREAK [(<i>r</i>)] [OF] <i>op-1</i> [/n/] &lt;<i>statement</i>   DO <i>statement</i>... DOEND&gt;</li> </ul>	V	S	S					
AT END OF DATA	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> [AT] END [OF] DATA [(<i>r</i>)] <i>statement</i>... END-ENDDATA</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [AT] END [OF] DATA [(<i>r</i>)] &lt;<i>statement</i>   DO <i>statement</i>... DOEND&gt;</li> </ul>	V	S	S					
AT END OF PAGE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> [AT] END [OF] PAGE [(<i>rep</i>)] <i>statement</i>... END-ENDPAGE</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [AT] END [OF] PAGE [(<i>rep</i>)] &lt;<i>statement</i>   DO <i>statement</i>... DOEND&gt;</li> </ul>	V	S	S					



Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
AT START OF DATA	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> [AT] START [OF] DATA [(r)] <i>statement...</i> END-START</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [AT] START [OF] DATA [(r)] &lt;<i>statement</i>   DO <i>statement...</i> DOEND&gt;</li> </ul>	V	S	S					
AT TOP OF PAGE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> [AT] TOP [OF] PAGE [(rep)] <i>statement...</i> END-TOPPAGE</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> [AT] TOP [OF] PAGE [(rep)] &lt;<i>statement</i>   DO <i>statement...</i> DOEND&gt;</li> </ul>	V	S	S					
BACKOUT TRANSACTION	<ul style="list-style-type: none"> <li><b>Format</b> BACKOUT [TRANSACTION]</li> </ul>	V	S	S					
BEFORE BREAK PROCESSING	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> BEFORE [BREAK] [PROCESSING] <i>statement...</i> END-BEFORE</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> BEFORE [BREAK] [PROCESSING] &lt;<i>statement</i>   DO <i>statement...</i> DOEND&gt;</li> </ul>	V	S	S					
CALL	<ul style="list-style-type: none"> <li><b>Format</b> CALL &lt;id-1  literal-1&gt; [USING] [op-2]...</li> </ul>	V	S	S				refers to literal-1 (programEntry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
CALL FILE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> CALL FILE 'program-name' op-1 op-2 <i>statement...</i>, END-FILE</li> </ul>	V	S	S				refers program-name (programEntry Point)	Program Calls Program Entry Point
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> CALL FILE 'program-name' op-1 op-2 <i>statement...</i>, [LOOP]</li> </ul>	V	S	S					

**2-14** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
CALL LOOP	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> CALL LOOP &lt;id-1   literal-1&gt; [op-2] statement., END-LOOP</li> </ul>	V	S	S				refers to literal-1 (programEntry Point), defines id-1 (Decision) Program Calls Program Entry Point, Program Calls Decision
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> CALL LOOP &lt;id-1   literal-1&gt; [op-2] statement., [LOOP]</li> </ul>	V	S	S				refers to literal-1 (programEntry Point), defines id-1 (Decision) Program Calls Program Entry Point, Program Calls Decision
CALLNAT	<ul style="list-style-type: none"> <li><b>Format</b> CALLNAT &lt;id-1   literal-1&gt; [[USING] &lt;op-2 [(AD-&lt;M   O   A&gt;)]   nX&gt;]</li> </ul>	V	S	S				refers to literal-1 (programEntry Point), defines id-1 (Decision) Program Calls Program Entry Point, Program Calls Decision
CLOSE CONVERSATION	<ul style="list-style-type: none"> <li><b>Format</b> CLOSE CONVERSATION &lt;op-1   *CONVID   ALL&gt;</li> </ul>	V	S	S				
CLOSE DIALOG	<ul style="list-style-type: none"> <li><b>Format</b> CLOSE DIALOG [USING] [DIALOG-ID] &lt;op-1   *DIALOG-ID&gt;</li> </ul>	V	S	S				
CLOSE PRINTER	<ul style="list-style-type: none"> <li><b>Format</b> CLOSE PRINTER &lt;(logical-printer-name)   (printer-number)&gt;</li> </ul>	V	S	S				
CLOSE WORK FILE	<ul style="list-style-type: none"> <li><b>Format</b> CLOSE WORK [FILE] work-file-number</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
COMPOSE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <p>COMPOSE  [ RESETTING [DATAAREA   TEXTAREA   MACROAREA   ALL] ]  [ MOVING [op-1 ...] [TO DATAAREA] [LAST] [STATUS [TO] op-2 [op-3 [op-4 [op-5]]]] ]  MOVING &lt;op-1 ... [TO DATAAREA]   LAST&gt; [OUTPUT] TO VARIABLES op-6 ... [STATUS [TO] op-2 [op-3 [op-4 [op-5]]]] ]  MOVING [OUTPUT] TO VARIABLES op-6 ... [STATUS [TO] op-2 [op-3 [op-4 [op-5]]]] ]  [ ASSIGNING [TEXTVARIABLE] op-7 = op-8 ... ]  [ FORMATTING  [OUTPUT &lt;(rep)   SUPPRESSED   CALLING op-9   TO VARIABLES [CONTROL op-10 op-11] op-12 ...   DOCUMENT-option&gt; ]  [INPUT &lt;DATAAREA [FROM &lt;EXIT op-14   CABINETS op-14 [PASSW=op-15]&gt; ...]   op-13 FROM &lt;EXIT op-14   CABINETS op-14 [PASSW=op-15]&gt; ...&gt; ]  [STATUS op-16 [op-17 [op-18 [op-19]]]] [PROFILE op-16] [MESSAGES &lt;[LISTED] [ON] (rep)   SUPPRESSED&gt; ]  [ERRORS &lt;[LISTED] [ON] (rep)   INTERCEPTED&gt; ] [ENDING &lt;[AT] [PAGE] op-20   AFTER op-20 [PAGES]&gt; ]  [STARTING [FROM] [PAGE] op-21] [EXTRACTING [TEXTVARIABLE] op-22 = op-23 ...]]</p>	N	N	N					
COMPRESS	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <p>COMPRESS [NUMERIC] [FULL] &lt;op-1 [(parameter)]   SUBSTRING (op-1, op-3, op-4) [(parameter)]&gt; INTO &lt;op-2   SUBSTRING (op-2, op-5, op-6)&gt; [LEAVING SPACE   LEAVING NO [SPACE]   WITH [ALL] DELIMITERS [op-7]]</p>	V	S	S	op-1	op-2	comp or const.comp		
COMPUTE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b></li> </ul> <p>&lt;COMPUTE [ROUNDED] op-1 = &lt;arithmetic-expression   op-2&gt;   op-1 := &lt;arithmetic-expression   op-2&gt;&gt;</p>	V	S	S	op-1	op-2	move, cast, comp+, comp*, comp		
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b></li> </ul> <p>[COMPUTE [ROUNDED]] op-1 = &lt;arithmetic-expression   op-2&gt;</p>	V	S	S	op-1	op-2	move, cast, comp+, comp*, comp		
CREATE OBJECT	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul> <p>CREATE OBJECT op-1 OF [CLASS] op-2 [ON [NODE] op-3] [GIVING [op-4]]</p>	V	S	S					

2-16 Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
DECIDE FOR	<ul style="list-style-type: none"> <li><b>Format</b>            DECIDE FOR &lt;FIRST   EVERY&gt; CONDITION WHEN <i>logical-condition-stmt</i> ... [WHEN ANY <i>stmt</i> ...] [WHEN ALL <i>stmt</i> ...] WHEN NONE <i>stmt</i> ... END-DECIDE</li> </ul>	V	S	S				
DECIDE ON	<ul style="list-style-type: none"> <li><b>Format</b>            DECIDE ON &lt;FIRST   EVERY&gt; [<u>VALUES</u>] [OF] <i>op-1 VALUES</i> &lt;<i>op-2</i>   <i>op-2</i> ...   [<i>op-2</i>]... <i>op-3</i> : <i>op-4</i>&gt; <i>stmt</i>... [ANY [<u>VALUES</u>] <i>stmt</i> ...] [ALL [<u>VALUES</u>] <i>stmt</i> ...] NONE [<u>VALUES</u>] <i>stmt</i> ... END-DECIDE</li> </ul>	V	S	S	<i>op-1</i> <i>op-1</i> <i>op-1</i>	<i>op-2</i> <i>op-3</i> ...	<i>cond</i>	
DEFINE CLASS	<ul style="list-style-type: none"> <li><b>General format (see all definitions below)</b>            DEFINE CLASS <i>classname</i> [[WITH] ACTIVATION [POLICY] &lt;ES   IM   EM&gt;] [OBJECT &lt;USING &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>data-definition</i> ...&gt;] ... [LOCAL &lt;USING &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>data-definition</i> ...&gt;] ... [ID <i>class-GUID</i>] [INTERFACE USING <i>copycode</i>   INTERFACE <i>stmt</i>] ... [PROPERTY <i>stmt</i>] ... [METHOD <i>stmt</i>] ... END-CLASS</li> </ul>	V	S	S				
DEFINE DATA	<ul style="list-style-type: none"> <li><b>General format (see all definitions below)</b>            DEFINE DATA <i>classname</i> [GLOBAL USING <i>global-data-area</i> [WITH <i>block</i> [, <i>block</i>] ...]] [PARAMETER &lt;USING <i>parameter-data-area</i>   <i>parameter-data-definition</i>&gt;] ... [OBJECT &lt;USING &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>data-definition</i> ...&gt;] ... [LOCAL &lt;USING &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>data-definition</i> ...&gt;] ... [INDEPENDENT AIV-<i>data-definition</i> ...] [CONTEXT &lt;USING &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>context-data-definition</i> ...&gt;]</li> </ul>	V	S	S				

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
Definition types	<ul style="list-style-type: none"> <li><b>data-definition</b>  <i>level &lt;group-name [(array-definition)]   view-definition   redefinition   variable-definition   handle-definition&gt;</i> </li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>parameter-data-definition</b>  <i>level &lt;group-name [(array-definition)]   redefinition   variable-name &lt;(format-length)   (format-length / array-definition)   (&lt;A   B&gt;) DYNAMIC&gt; [BY VALUE [RESULT] [OPTIONAL]]   handle-definition&gt;</i> </li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>handle-definition</b>  <i>handle-name &lt;HANDLE OF &lt;dialog-element-type   OBJECT&gt; [&lt;CONSTANT   INIT&gt; init-definition]   (array-definition) HANDLE OF &lt;dialog-element-type   OBJECT&gt; [&lt;CONSTANT   INIT&gt; array-init-definition]&gt;</i> </li> </ul>	N	S	S				
	<ul style="list-style-type: none"> <li><b>view-definition</b>  <i>view-name VIEW [OF] ddm-name [level &lt;ddm-field [(format-length) [emhdpm]]   (&lt;A   B&gt;) DYNAMIC]   redefinition&gt;</i> </li> </ul>	V	S	S				<i>refers to ddm-name (Natural DDM file)</i>
	<ul style="list-style-type: none"> <li><b>redefinition</b>  <i>REDEFINE field-name level &lt;rfield (format-length)   FILLER nX&gt; ...</i> </li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>variable-definition</b>  <i>variable-name &lt;(format-length) [&lt;CONSTANT   INIT&gt; init-definition] [emhdpm]   (format-length / array-definition) [&lt;CONSTANT   INIT&gt; array-init-definition] [emhdpm]   (&lt;A   B&gt;) DYNAMIC&gt;</i> </li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>init-definition</b>  <i>&lt; &lt;constant&gt;   &lt;system-variable&gt;   FULL LENGTH &lt;character-s&gt;   LENGTH n &lt;character-s&gt; &gt;</i> </li> </ul>	V	S	S				

2-18 Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
	<ul style="list-style-type: none"> <li><b>array-definition</b> <i>index [:index], ...</i></li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>array-init-definition</b> <i>[ALL   (&lt;index [:index]   V&gt;, ...)] &lt;&lt;FULL LENGTH   LENGTH n&gt; &lt;character-s, ...&gt;   &lt;constant, ...&gt;   &lt;system-variable, ...&gt;&gt;</i></li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>emhdpm</b> <i>((EM = value) [HD = 'value'] [PM=value])</i></li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>AIV-data-definition</b> <i>level &lt;AIV-definition   REDEFINE field-name&gt;</i></li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>AIV-definition</b> <i>variable-name &lt;(format-length) [INIT init-definition]   (format-length / array-definition) [INIT array-init-definition]&gt; [emhdpm]</i></li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>context-data-definition</b> <i>level &lt;variable-definition   redefinition   handle-definition&gt;</i></li> </ul>	V	S	S				
DEFINE PRINTER	<ul style="list-style-type: none"> <li><b>Format</b> <i>DEFINE PRINTER ((logical-printer-name =) n) [OUTPUT op-1] [PROFILE op-2   FORMS op-2   NAME op-2   DISP op-2   CLASS op-2   COPIES op-2   PRTY op-2]</i></li> </ul>	V	S	S				
DEFINE SUB-ROUTINE	<ul style="list-style-type: none"> <li><b>Format</b> <i>DEFINE [SUBROUTINE] subroutine-name statement ... &lt;END-SUBROUTINE   RETURN&gt;</i></li> </ul>	V	S	S				
DEFINE WINDOW	<ul style="list-style-type: none"> <li><b>Format</b> <i>DEFINE WINDOW window-name [SIZE &lt;AUTO   QUARTER   op-1 * op-2&gt;] [BASE &lt;CURSOR   &lt;TOP   BOTTOM&gt; &lt;LEFT   RIGHT&gt;   op-3 / op-4&gt;] [REVERSED [(CD = background-color)]] [TITLE op-5] [CONTROL &lt;WINDOW   SCREEN&gt;] [FRAMED &lt;OFF   ON&gt;] [(CD = frame-color)] [POSITION &lt;SYMBOL [TOP   BOTTOM] [AUTO] [SHORT] [LEFT   RIGHT]   TEXT [MORE] [LEFT   RIGHT]   OFF&gt;]&gt;</i></li> </ul>	V	S	S				

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
DEFINE WORK FILE	<ul style="list-style-type: none"> <li><b>Format</b> DEFINE WORK FILE <i>n op-1</i> [TYPE <i>op-2</i>]</li> </ul>	V	S	S				
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE [RECORD] [IN] [STATEMENT] [(<i>r</i>)]</li> </ul>	V	S	S				Program Deletes from Adabas File
DISPLAY	<ul style="list-style-type: none"> <li><b>Format</b> DISLPAY [(<i>rep</i>)] [NOTITLE] [NOHDR] [[AND] [GIVE] [SYSTEM] FUNCTIONS] [(<i>stmt-parameters</i>)] [<i>l</i> ...] [<i>nX</i>   <i>nT</i>   <i>x / y</i>   <i>T*field-name</i>   <i>P*field-name</i>] [<i>text</i> [(<i>attributes</i>)]   '<i>c</i>' (<i>n</i>) [(<i>attributes</i>)]] [VERTICALLY [AS &lt;<i>text</i>' [(<i>attributes</i>)] [CAPTIONED]   CAPTIONED&gt; [<i>l</i>...]]   HORIZONTALLY [&lt;<i>text</i>' [(<i>attributes</i>)]   '<i>c</i>' (<i>n</i>) [(<i>attributes</i>)]&gt;   <i>nX</i>   <i>nT</i>   <i>x / y</i>   '='] <i>op-1</i> [(<i>parameters</i>)]</li> </ul>	V	S	S				
DIVIDE	<ul style="list-style-type: none"> <li><b>Format 1: with GIVING</b> DIVIDE [ROUNDED] <i>op-1</i> INTO <i>op-2</i> [GIVING <i>op-3</i>]</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i> ( <i>op-3</i> )	<i>comp</i> *	
	<ul style="list-style-type: none"> <li><b>Format 2: with REMAINDER</b> DIVIDE <i>op-1</i> INTO <i>op-2</i> [GIVING <i>op-3</i>] REMAINDER <i>op-4</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i> ( <i>op-3</i> )	<i>comp</i> *	
DO / DOEND	<ul style="list-style-type: none"> <li><b>Format</b> DO <i>stmt</i> ... DOEND</li> </ul>	V	S	S				
EJECT	<ul style="list-style-type: none"> <li><b>Format 1</b> EJECT &lt;ON   OFF&gt; [(<i>rep</i>)]</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2</b> EJECT [(<i>rep</i>)] [[IF   WHEN] LESS [THAN] <i>op-1</i> [LINES] [LEFT]]</li> </ul>	V	S	S				
END	<ul style="list-style-type: none"> <li><b>Format</b> &lt;END   &gt;</li> </ul>	V	S	S				
END TRANSACTION	<ul style="list-style-type: none"> <li><b>Format</b> END [OF] TRANSACTION [<i>op-1</i> ...]</li> </ul>	V	S	S				
ESCAPE	<ul style="list-style-type: none"> <li><b>Format</b> ESCAPE &lt;TOP   BOTTOM [(<i>r</i>)] [IMMEDIATE]   ROUTINE [IMMEDIATE]&gt;</li> </ul>	V	S	S				

**2-20** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF				Repository	
				Sup.	Pairs	Relation	Entities	Relations	
EXAMINE	<ul style="list-style-type: none"> <li><b>Format</b>  EXAMINE [FULL [VALUE [OF]]] &lt;op-1   SUBSTRING (op-1, op-2, op-3)&gt; [FOR] [FULL [VALUE [OF]]] [PATTERN] op-4 [ABSOLUTE   WITH [DELIMITERS]   WITH [DELIMITERS] op-5] [[AND] &lt;DELETE [FIRST]   REPLACE [FIRST] [WITH] [FULL [VALUE [OF]]] op-6&gt;] [[GIVING] [NUMBER   POSITION   LENGTH   INDEX] [IN] op-7]</li> </ul>	V	S	S	op-1	op-1	comp		
EXAMINE TRANSLATE	<ul style="list-style-type: none"> <li><b>Format</b>  EXAMINE &lt;op-1   SUBSTRING (op-1, op-2, op-3)&gt; [AND] TRANSLATE &lt;INTO &lt;UPPER   LOWER&gt; CASE   USING [INVERTED] op-4&gt;</li> </ul>	V	S	S	op-1	op-1	comp		
EXPAND	<ul style="list-style-type: none"> <li><b>Format</b>  EXPAND [SIZE OF] DYNAMIC VARIABLE op-1 TO op-2</li> </ul>	V	N	N					
FETCH	<ul style="list-style-type: none"> <li><b>Format</b>  FETCH &lt;REPEAT   RETURN&gt; &lt;id-1   literal-1&gt; [op-2 [(parameter)]] ...</li> </ul>	V	S	S				refers to literal-1 (programEntry Point), defines id-1 (Decision)	Program Calls Program Entry Point, Program Calls Decision
FIND	<ul style="list-style-type: none"> <li><b>Format</b>  FIND [ALL   (op-1)   FIRST   NUMBER   UNIQUE] [RECORDS] [IN] [FILE] view-name [PASSW = op-2] [CIPHER = op-3] [WITH] [[LIMIT] (op-4)] basic-search-criterion [&lt;AND   OR&gt; COUPLED [TO] [FILE] view-name [VIA descriptor-1 &lt;=   EQ   EQUAL [TO]&gt; descriptor-2] [WITH] basic-search-criterion] [STARTING WITH ISN = op-5] [SORTED [BY] descriptor-3 [DESCENDING]] [RETAIN AS op-6] [WHERE logical-condition] [IF NO [RECORDS] [FOUND] &lt;ENTER   stmt-2&gt; &lt;END-NOREC   DO stmt-2 DOEND&gt;] stmt-1 ... &lt;END-FIND   [LOOP]&gt;</li> </ul>	V	S	S					Program Reads Adabas File
FOR	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b>  FOR op-1 [[:]=   EQ   FROM] op-2 [TO   THRU] op-3 [[STEP] op-4] stmt ... END-FOR</li> </ul>	V	S	S	op-1 op-1 op-1	op-2 op-3 op-4	move or cast cond comp+		
	<ul style="list-style-type: none"> <li><b>Format 2: reported mode</b>  FOR op-1 [[:]=   EQ   FROM] op-2 [TO   THRU] op-3 [[STEP] op-4] stmt ... [LOOP]</li> </ul>	V	S	S	op-1 op-1 op-1	op-2 op-3 op-4	move or cast cond comp+		



Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
FORMAT	<ul style="list-style-type: none"> <li><b>Format</b> FORMAT [(rep)] parameter ...</li> </ul>	V	S	S				
GET	<ul style="list-style-type: none"> <li><b>Format</b> GET [IN] [FILE] view-name [PASSWORD = op-1] [CIPHER = op-2] [RECORD] &lt;op-3   *ISN [(r)]&gt; op-4 ...</li> </ul>	V	S	S				Program Reads Adabas File
GET SAME	<ul style="list-style-type: none"> <li><b>Format</b> GET SAME [(r)] [op-1 ...]</li> </ul>	V	S	S				
GET TRANS- ACTION DATA	<ul style="list-style-type: none"> <li><b>Format</b> GET TRANSACTION [DATA ] op-1 ...</li> </ul>	V	S	S				
HISTOGRAM	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> HISTOGRAM [ALL   (op-1)] [VALUE] [IN] [FILE] view-name [PASSWORD = op-2] [(IN) &lt;ASCENDING   DESCENDING   VARIABLE op-3&gt; [SEQUENCE]] [VALUE] [FOR] [FIELD] op-4 [[[STARTING] [WITH]   FROM] [VALUES] op-5] [[THRU   ENDING AT] op-6]] [WHERE logical-condition] stmt ... END-HISTOGRAM</li> </ul>	V	S	S				Program Reads Adabas File
	<ul style="list-style-type: none"> <li><b>Format 2: reported mode</b> HISTOGRAM [ALL   (op-1)] [VALUE] [IN] [FILE] view-name [PASSWORD = op-2] [(IN) &lt;ASCENDING   DESCENDING   VARIABLE op-3&gt; [SEQUENCE]] [VALUE] [FOR] [FIELD] op-4 [[[STARTING] [WITH]   FROM] [VALUES] op-5] [[THRU   ENDING AT] op-6]] [WHERE logical-condition] stmt ... [LOOP]</li> </ul>	V	S	S				
IF	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> IF logical-condition [THEN] stmt ... [ELSE stmt ...] END-IF</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: reported mode</b> IF logical-condition [THEN] &lt;stmt ...   DO stmt ... DOEND&gt; [ELSE &lt;stmt ...   DO stmt ... DOEND&gt;]</li> </ul>	V	S	S				

**2-22** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
IF SELECTION	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b>  IF SELECTION [NOT UNIQUE [IN [FIELDS]]] <i>op-1</i> [THEN] <i>stmt ...</i> [ELSE <i>stmt ...</i>] END-IF</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: reported mode</b>  IF SELECTION [NOT UNIQUE [IN [FIELDS]]] <i>op-1</i> [THEN] &lt;<i>stmt ...</i>   DO <i>stmt ...</i> DOEND&gt; [ELSE &lt;<i>stmt ...</i>   DO <i>stmt ...</i> DOEND&gt;]</li> </ul>	V	S	S				
IGNORE	<ul style="list-style-type: none"> <li><b>Format</b>  IGNORE</li> </ul>	V	S	S				
INCLUDE	<ul style="list-style-type: none"> <li><b>Format</b>  INCLUDE <i>copycode-name</i> [<i>op-1 ...</i>]</li> </ul>	V	S	S				<i>refers to copycode-name (Natural Include file)</i>
INPUT	<ul style="list-style-type: none"> <li><b>Format 1: dynamic screen layout specification</b>  INPUT [WINDOW = '<i>window-name</i>'] [NO ERASE] [(<i>statement-parameters</i>)] [[WITH] TEXT [<i>op-1</i>   <i>op-2</i>] [(<i>attributes</i>)] [, <i>op-3 ...</i>]] [MARK [POSITION <i>op-4</i> [IN]] [FIELD] &lt;<i>op-1</i>   *<i>field-name</i>&gt;] [[AND] [SOUND] ALARM] [<i>nX</i>   <i>nT</i>   <i>x / y</i>] [<i>text</i>] [(<i>attributes</i>)]   '<i>c</i>'(<i>n</i>) [(<i>attributes</i>)]   '<i>'</i>   '='   /...]] [*IN   *OUT   *OUTIN] <i>op-1</i>(<i>parameter</i>) ...</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: using predefined map layout</b>  INPUT [WINDOW = '<i>window-name</i>'] [[WITH] TEXT [<i>op-1</i>   <i>op-2</i>] [(<i>attributes</i>)] [, <i>op-3 ...</i>]] [MARK [POSITION <i>op-4</i> [IN]] [FIELD] &lt;<i>op-1</i>   *<i>field-name</i>&gt;] [[AND] [SOUND] ALARM] [USING] MAP &lt;<i>id-map</i>   <i>lit-map</i>&gt; [NO ERASE] [<i>op-1</i>   NO PARAMETER]</li> </ul>	V	S	S				<i>refers to lit-map (Natural Map), defines id-map (Decision)</i>  <i>Program Input Using Map, Program Input Using Decision</i>
INTERFACE	<ul style="list-style-type: none"> <li><b>Format</b>  INTERFACE <i>interface-name</i> [ID <i>interface-GUID</i>]  [PROPERTY <i>property-name</i> [(<i>format-length/array-definition</i>)] [READONLY] [IS <i>oper</i>] END-PROPERTY]  [PROPERTY <i>method-name</i> [IS <i>subprogram-name</i>] [PARAMETER &lt;<i>local-data-area</i>   <i>parameter-data-area</i>&gt;   <i>data-definition ...</i>&gt;] END-METHOD] END-INTERFACE  See the "Definition types" row for details about '<i>data-definition</i>' and '<i>array-definition</i>'</li> </ul>	V	S	S				
LIMIT	<ul style="list-style-type: none"> <li><b>Format</b>  LIMIT <i>n</i></li> </ul>	V	S	S				

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
LOOP	<ul style="list-style-type: none"> <li><b>Format</b> [CLOSE] LOOP [(r)]</li> </ul>	V	S	S				
METHOD	<ul style="list-style-type: none"> <li><b>Format</b> METHOD <i>method-name</i> OF [INTERFACE] <i>interface-name</i> IS <i>sub-program-name</i> END-METHOD</li> </ul>	V	S	S				
MOVE	<ul style="list-style-type: none"> <li><b>Format 1</b> MOVE [ROUNDED] <i>op-1</i> [(parameter)] TO <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
	<ul style="list-style-type: none"> <li><b>Format 2</b> MOVE [ROUNDED] &lt;<i>op-1</i>   SUBSTRING (<i>op-1</i>, <i>op-3</i>, <i>op-4</i>)&gt; [(parameter)] TO &lt;<i>op-2</i>   SUBSTRING (<i>op-2</i>, <i>op-5</i>, <i>op-6</i>)&gt;</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
	<ul style="list-style-type: none"> <li><b>Format 3</b> MOVE BY &lt;[NAME]   POSITION&gt; <i>op-1</i> TO <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
	<ul style="list-style-type: none"> <li><b>Format 4</b> MOVE EDITED <i>op-1</i> TO <i>op-2</i> (EM = <i>value</i>)</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
	<ul style="list-style-type: none"> <li><b>Format 5</b> MOVE EDITED <i>op-1</i> (EM = <i>value</i>) TO <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
	<ul style="list-style-type: none"> <li><b>Format 6</b> MOVE &lt;LEFT   RIGHT&gt; [JUSTIFIED] <i>op-1</i> [(parameter)] TO <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>move or const.move</i>	
MOVE ALL	<ul style="list-style-type: none"> <li><b>Format</b> MOVE ALL <i>op-1</i> TO <i>op-2</i> [UNTIL <i>op-3</i>]</li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp or const.comp</i>	
MULTIPLY	<ul style="list-style-type: none"> <li><b>Format 1</b> MULTIPLY [ROUNDED] <i>op-1</i> BY <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp* or const.comp</i>	
	<ul style="list-style-type: none"> <li><b>Format 2</b> MULTIPLY [ROUNDED] <i>op-1</i> BY <i>op-2</i> GIVING <i>op-3</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp* or const.comp</i>	

**2-24** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
NEWPAGE	<ul style="list-style-type: none"> <li><b>Format</b> NEWPAGE [(rep)] [EVEN [IF] TOP [OF] [PAGE]   [IF] [WHEN] LESS [THAN] op-1 [LINES] [LEFT]] [[WITH] TITLE]</li> </ul>	V	S	S				
OBTAIN	<ul style="list-style-type: none"> <li><b>Format</b> OBTAIN op-1 ...</li> </ul>	V	S	S				
ON ERROR	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> ON ERROR stmt ... END-ERROR</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> ON ERROR &lt;stmt ...   DO stmt ... DOEND&gt;</li> </ul>	V	S	S				
OPEN CON- VERSATION	<ul style="list-style-type: none"> <li><b>Format</b> OPEN CONVERSATION USING [SUBPROGRAMS] op-1 ...</li> </ul>	V	S	S				
OPEN DIALOG	<ul style="list-style-type: none"> <li><b>Format</b> OPEN DIALOG op-1 [USING] [PARENT] op-2 [[GIVING] [DIALOG-ID] op-3] [WITH &lt;op-4 [(AD = &lt;M   O   A&gt;)]   nX   PARAMETERS parameter-name = op-4 ... END-PARAMETERS]</li> </ul>	V	S	S				
PASSW	<ul style="list-style-type: none"> <li><b>Format</b> PASSW = op-1</li> </ul>	V	S	S				
PERFORM	<ul style="list-style-type: none"> <li><b>Format</b> PERFORM op-1 [op-2 [(AD = &lt;M   O   A&gt;)]   nX] ...</li> </ul>	V	S	S			refers to op-1 (program Entry Point) in case external sub-routine	Program Calls Program Entry Point
PERFORM BREAK PRO- CESSING	<ul style="list-style-type: none"> <li><b>Format</b> PERFORM BREAK [PROCESSING] [(r)] AT BREAK stmt ...</li> </ul>	V	S	S				
PRINT	<ul style="list-style-type: none"> <li><b>Format</b> PRINT [(rep)] [NOTITLE] [NOHDR] [(statement-parameters)] [nX   nT   /] ... &lt;text [(attributes)]   'c' (n) [(attributes)]   [=] op-1 [(parameters)]&gt; ...</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
PROCESS	<ul style="list-style-type: none"> <li><b>Format</b> PROCESS <i>view-name</i> USING <i>op-1</i> = <i>op-2</i> GIVING <i>op-3</i> ...</li> </ul>	N	N	N					
PROCESS COMMAND	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> PROCESS COMMAND ACTION &lt;CLOSE   &lt;CHECK   EXEC   TEXT   HELP&gt; USING PROCESSOR-NAME = <i>op-1</i> COMMAND-LINE (<i>index</i> [<i>index</i>]) = <i>op-2</i>   GET USING PROCESSOR-NAME = <i>op-1</i> GETSET-FIELD-NAME = <i>op-3</i>   SET USING PROCESSOR-NAME = <i>op-1</i> GETSET-FIELD-NAME = <i>op-3</i> GETSET-FIELD-VALUE = <i>op-4</i>&gt;</li> </ul>	N	N	N					
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> PROCESS COMMAND ACTION &lt;CLOSE [GIVING NATURAL-ERROR]   &lt;CHECK   EXEC   TEXT   HELP&gt; USING PROCESSOR-NAME = <i>op-1</i> COMMAND-LINE (<i>index</i> [<i>index</i>]) = <i>op-2</i> GIVING RESULT-FIELD (<i>index</i> [<i>index</i>]) RETURN-CODE [NATURAL-ERROR]   GET USING PROCESSOR-NAME = <i>op-1</i> GETSET-FIELD-NAME = <i>op-3</i> GIVING GETSET-FIELD-VALUE [NATURAL-ERROR]   SET USING PROCESSOR-NAME = <i>op-1</i> GETSET-FIELD-NAME = <i>op-3</i> GETSET-FIELD-VALUE = <i>op-4</i> [GIVING NATURAL-ERROR]&gt;</li> </ul>	N	N	N					
PROCESS GUI	<ul style="list-style-type: none"> <li><b>Format</b> PROCESS GUI ACTION <i>action-name</i> WITH &lt;<i>op-1</i> ...   <i>nX</i> ...   PARAMETERS <i>parameter-name</i> = <i>op-1</i> ... END-PARAMETERS&gt; [GIVING <i>op-2</i>]</li> </ul>	N	N	N					
PROCESS REPORTER	<ul style="list-style-type: none"> <li><b>Format</b> PROCESS REPORTER ACTION &lt;INITIALIZE   TERMINATE   &lt;OPEN   CLOSE   REPLACE-TABLE   SET-PRINTER   SET-PRINT-OPTIONS   PRINT   PREVIEW&gt; WITH &lt;<i>op-1</i> ...   PARAMETERS <i>parameter-name</i> = <i>op-1</i> ... END-PARAMETERS&gt;   EDIT [WITH &lt;<i>op-1</i> ...   PARAMETERS <i>parameter-name</i> = <i>op-1</i> ... END-PARAMETERS&gt;]&gt; [GIVING <i>op-2</i>]</li> </ul>	N	N	N					
PROPERTY	<ul style="list-style-type: none"> <li><b>Format</b> PROPERTY <i>property-name</i> OF [INTERFACE] <i>interface-name</i> IS <i>operand</i> END-PROPERTY</li> </ul>	V	S	S					

2-26 Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
READ	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b>            &lt;READ   BROWSE&gt; [ALL   (op-1)] [RECORDS] [IN] [FILE] view-name [PASSWORD = op-2] [CIPHER = op-3] [WITH REPOSITION] [[IN] [PHYSICAL] [ASCENDING   DESCENDING   VARIABLE op-5] [SEQUENCE]] [BY   WITH &gt; &lt;ISN   descriptor&gt; [&lt;=   EQ   EQUAL TO]   [STARTING] FROM&gt; op-6] [[THRU   ENDING AT] op-7]] [STARTING WITH ISN = op-4] [WHERE logical-condition] stmt ... END-READ</li> </ul>	V	S	S				Program Reads Adabas File
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b>            &lt;READ   BROWSE&gt; [ALL   (op-1)] [RECORDS] [IN] [FILE] view-name [PASSWORD = op-2] [CIPHER = op-3] [WITH REPOSITION] [[IN] [PHYSICAL] [ASCENDING   DESCENDING   VARIABLE op-5] [SEQUENCE]] [BY   WITH &gt; &lt;ISN   descriptor&gt; [&lt;=   EQ   EQUAL TO]   [STARTING] FROM&gt; op-6] [[THRU   ENDING AT] op-7]] [STARTING WITH ISN = op-4] [WHERE logical-condition] stmt ... [LOOP]</li> </ul>	V	S	S				Program Reads Adabas File
READ WORK FILE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b>            READ WORK [FILE] work-file-number [ONCE] &lt;RECORD op-1   [AND] [SELECT] [OFFSET n   FILLER nX] ... op-2 ...&gt; [GIVING LENGTH op-3] AT [END] [OF] [FILE] stmt ... END-ENDFILE stmt... END-WORK</li> </ul>	V	S	S			defines program-name.CMWKF<work-file-number>	Program Reads File
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b>            READ WORK [FILE] work-file-number [ONCE] &lt;RECORD op-1   [AND] [SELECT] [OFFSET n   FILLER nX] ... op-2 ...&gt; [GIVING LENGTH op-3] AT [END] [OF] [FILE] &lt;stmt ...   DO stmt... DOEND&gt; stmt ... [LOOP]</li> </ul>	V	S	S			DATAPORT	Program Reads File
REDEFINE	<ul style="list-style-type: none"> <li><b>Format</b>            REDEFINE op-1 (&lt;nX   op-2&gt; ...) ...</li> </ul>	V	S	S	op-1	op-2	redefine	
REDUCE	<ul style="list-style-type: none"> <li><b>Format</b>            REDUCE [SIZE OF] DYNAMIC [VARIABLE] op-1 TO op-2</li> </ul>	N	N	N				
REINPUT	<ul style="list-style-type: none"> <li><b>Format</b>            REINPUT [FULL] [(statement-parameters)] &lt;USING HELP   [WITH] [TEXT] [*op-1   op-2] [(attributes)] [, op-3 ...]&gt; [MARK [POSITION op-4 [IN]] [FIELD] &lt;op-1   *field-name&gt;] [[AND] [SOUND] ALARM]</li> </ul>	V	S	S				

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
REJECT	<ul style="list-style-type: none"> <li><b>Format</b> REJECT [IF] <i>logical-condition</i></li> </ul>	V	S	S					
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE &lt;STACK   <u>SETS</u> [<i>set-name ...</i>]   VARIABLES&gt;</li> </ul>	V	S	S					
REPEAT	<ul style="list-style-type: none"> <li><b>Format 1</b> REPEAT <i>stmt ...</i> [&lt;UNTIL   WHILE&gt; <i>logical-condition</i>] END-REPEAT</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2</b> REPEAT [&lt;UNTIL   WHILE&gt; <i>logical-condition</i>] <i>stmt ...</i> END-REPEAT</li> </ul>	V	S	S					
RESET	<ul style="list-style-type: none"> <li><b>Format</b> RESET [INITIAL] <i>op-1</i></li> </ul>	V	S	S					
RETRY	<ul style="list-style-type: none"> <li><b>Format</b> RETRY</li> </ul>	V	S	S					
RUN	<ul style="list-style-type: none"> <li><b>Format</b> RUN [REPEAT] &lt;id-1  literal-1&gt; [<i>op-2...]</i></li> </ul>	V	S	S				<i>refers to literal-1 (program Entry Point), defines id-1 (Decision)</i>	<i>Program Calls Program Entry Point, Program Calls Decision</i>
SEND EVENT	<ul style="list-style-type: none"> <li><b>Format</b> SEND EVENT <i>op-1</i> TO [DIALOG-ID] &lt;<i>op-2</i>   *DIALOG-ID&gt; [WITH &lt;<i>op-3</i> [(AD =&lt;M   O   A&gt;)]   <i>nX</i>&gt; ...   USING [DIALOG] '<i>dialog-name</i>' WITH PARAMETERS <i>parameter-name</i> = <i>op-3 ...</i> END-PARAMETERS]</li> </ul>	V	S	S					
SEND METHOD	<ul style="list-style-type: none"> <li><b>Format</b> SEND [METHOD] <i>op-1</i> TO [OBJECT] <i>op-2</i> [WITH &lt;<i>op-3</i> [(AD =&lt;M   O   A&gt;)]   <i>nX</i>&gt; ... ] [RETURN <i>op-4</i>] [GIVING <i>op-5</i>]</li> </ul>	V	S	S					

**2-28** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
SEPARATE	<ul style="list-style-type: none"> <li><b>Format</b>  SEPARATE &lt;op-1   <u>SUBSTRING</u> (op-1, op-2, op-3)&gt; [LEFT [JUSTIFIED]] INTO op-4 ... [[IGNORE   REMAINDER op-5] [WITH [RETAINED] &lt;{ANY} DELIMITERS   INPUT DELIMITERS   DELIMITERS op-6&gt;] [[GIVING] NUMBER [IN] op-7]</li> </ul>	V	S	S	op-1 op-1	op-4 op-7	comp or const.comp		
SET CONTROL	<ul style="list-style-type: none"> <li><b>Format</b>  SET CONTROL op-1</li> </ul>	V	S	S					
SET GLOBALS	<ul style="list-style-type: none"> <li><b>Format</b>  SET GLOBALS parameter ...</li> </ul>	V	S	S					
SET KEY	<ul style="list-style-type: none"> <li><b>Format 1: affecting all keys</b>  SET KEY &lt;ALL   ON   OFF   COMMAND &lt;ON   OFF&gt;   NAMED OFF&gt;</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: affecting individual keys</b>  SET KEY &lt;PA<sub>n</sub>   PF<sub>n</sub>   CLR   DYNAMIC op-1&gt; [= &lt;ON   OFF   DISABLED   COMMAND &lt;ON   OFF&gt;&gt;] ...</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 3: affecting individual keys</b>  SET KEY &lt;&lt;PA<sub>n</sub>   PF<sub>n</sub>   CLR   DYNAMIC op-1&gt; [= &lt;PGM   op-2   HELP   DATA op-3&gt;] [NAMED &lt;op-4   OFF&gt;]   ENTR NAMED &lt;op-4   OFF&gt;&gt;</li> </ul>	V	S	S					
SET TIME	<ul style="list-style-type: none"> <li><b>Format</b>  &lt;SET TIME   SETTIME&gt;</li> </ul>	V	S	S					
SET WINDOW	<ul style="list-style-type: none"> <li><b>Format</b>  SET WINDOW &lt;'window-name'   OFF&gt;</li> </ul>	V	S	S					
SKIP	<ul style="list-style-type: none"> <li><b>Format</b>  SKIP [(rep)] op-1 [LINES]</li> </ul>	V	S	S					



Statement	Format	Parser	IA	CA/GDF			Repository	
				Sup.	Pairs	Relation	Entities	Relations
SORT	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> END-ALL [AND] SORT [THEM   RECORDS] [BY] <i>op-1</i> [ASCENDING   DESCENDING] ... &lt;USING <i>op-2</i>   USING KEYS&gt; [GIVE &lt;MAX   MIN   NMIN   COUNT   NCOUNT   OLD   AVER   NAVER   SUM   TOTAL&gt; ... [OF] &lt;(<i>op-3</i> ...)   <i>op-3</i> ...&gt; [(NL=<i>nn</i>)] ...] <i>stmt</i> ... END-SORT</li> </ul>	V	S	S				
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> SORT [THEM   RECORDS] [BY] <i>op-1</i> [ASCENDING   DESCENDING] ... &lt;USING <i>op-2</i>   USING KEYS&gt; [GIVE &lt;MAX   MIN   NMIN   COUNT   NCOUNT   OLD   AVER   NAVER   SUM   TOTAL&gt; ... [OF] &lt;(<i>op-3</i> ...)   <i>op-3</i> ...&gt; [(NL=<i>nn</i>)] ...] <i>stmt</i> ...</li> </ul>	V	S	S				
STACK	<ul style="list-style-type: none"> <li><b>Format</b> STACK [TOP] &lt;COMMAND <i>op-1</i> [<i>op-2</i> [(<i>parameter</i>)] ...]   [DATA] [FORMATTED] <i>op-1</i> [(<i>parameter</i>)] ...&gt;</li> </ul>	V	S	S				
STOP	<ul style="list-style-type: none"> <li><b>Format</b> STOP</li> </ul>	V	S	S				
STORE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> STORE [RECORD] [IN] [FILE] <i>view-name</i> [PASSWORD = <i>op-1</i>] [CIPHER = <i>op-2</i>] [[USING   GIVING] NUMBER <i>op-3</i>] [(<i>r</i>)]</li> </ul>	V	S	S				TABLE <i>Program Inserts into Adabas File</i>
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> STORE [RECORD] [IN] [FILE] <i>view-name</i> [PASSWORD = <i>op-1</i>] [CIPHER = <i>op-2</i>] [[USING   GIVING] NUMBER <i>op-3</i>] [(<i>r</i>)] &lt;[USING] SAME [RECORD] [AS] [STATEMENT [(<i>r</i>)]   [SET   WITH] [ <i>op-4</i> = <i>op-5</i>] ...&gt;</li> </ul>	V	S	S				TABLE <i>OUT</i>
SUBTRACT	<ul style="list-style-type: none"> <li><b>Format 1</b> SUBTRACT [ROUNDED] <i>op-1</i> ... FROM <i>op-2</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp+ or const.comp</i>	
	<ul style="list-style-type: none"> <li><b>Format 2</b> SUBTRACT [ROUNDED] <i>op-1</i> ... FROM <i>op-2</i> GIVING <i>op-3</i></li> </ul>	V	S	S	<i>op-1</i>	<i>op-2</i>	<i>comp+ or const.comp</i>	
SUSPEND IDENTICAL SUPPRESS	<ul style="list-style-type: none"> <li><b>Format</b> SUSPEND IDENTICAL [SUPPRESS] [(<i>rep</i>)]</li> </ul>	V	S	S				

**2-30** Supported Natural Statements  
*Supported Natural 3.1.3 for Mainframes / 4.1.2 for Windows statements*

Statement	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs	Relation	Entities	Relations	
TERMINATE	<ul style="list-style-type: none"> <li><b>Format</b> TERMINATE [op-1 [op-2]]</li> </ul>	V	S	S					
UPDATE	<ul style="list-style-type: none"> <li><b>Format 1: structured mode</b> UPDATE [RECORD] [IN] [STATEMENT] [(r)]</li> </ul>	V	S	S					Program Updates Adabas File
	<ul style="list-style-type: none"> <li><b>Format 2: reporting mode</b> UPDATE [RECORD] [IN] [STATEMENT] [(r)] [SET   WITH   USING] &lt;SAME [RECORD]   op-1 = op-2 ...&gt;</li> </ul>	V	S	S					Program Updates Adabas File
WRITE	<ul style="list-style-type: none"> <li><b>Format 1: dynamic formatting</b> WRITE [(rep)] [NOTITLE] [NOHDR] [(statement-parameters)] [nX   nT   x/y   T*field-name   P*field-name   /] ... &lt;text' [(attributes)]   'c' (n) [(attributes)]   [=] op-1 [(parameters)]&gt; ...</li> </ul>	V	S	S					
	<ul style="list-style-type: none"> <li><b>Format 2: using predefined map</b> WRITE [(rep)] [NOTITLE] [NOHDR] [USING] &lt;FORM   MAP&gt; &lt;id-map   lit-map&gt; [op-2 ...]</li> </ul>	V	S	S				refers to lit-map (Natural Map), defines id-map (Decision)	Program Uses Map, Program Uses Map Decision
WRITE TITLE	<ul style="list-style-type: none"> <li><b>Format</b> WRITE [(rep)] TITLE [LEFT [JUSTIFIED]] [UNDERLINED] [(statement-parameters)] [nX   nT   x/y] ... &lt;text' [(attributes)]   'c' (n) [(attributes)]   [=] op-1 [(parameters)]&gt; ... [SKIP op-2 [LINES]]</li> </ul>	V	S	S					
WRITE TRAILER	<ul style="list-style-type: none"> <li><b>Format</b> WRITE [(rep)] TRAILER [LEFT [JUSTIFIED]] [UNDERLINED] [(statement-parameters)] [nX   nT   x/y] ... &lt;text' [(attributes)]   'c' (n) [(attributes)]   [=] op-1 [(parameters)]&gt; ... [SKIP op-2 [LINES]]</li> </ul>	V	S	S					
WRITE WORK FILE	<ul style="list-style-type: none"> <li><b>Format</b> WRITE WORK [FILE] work-file-number [VARIABLE] op-1 ...</li> </ul>	V	S	S				defines program-name.CMWKF<work-file-number>	Program Inserts into File

## Supported PL/I Statements



This chapter contains detailed information on ATW parser, HyperView, Change Analyzer, and Global Data Element Flow support for PL/I statements. For a list of supported legacy versions, refer to the *Release Notes*.

### Key to tables

The tables below contain detailed information on ATW parser, HyperView, Change Analyzer, Global Data Element Flow, and repository support. The first column of each table shows PL/I keywords.

- **Parser** — ATW Parser. The possible values in this column are:
  - (V)erified — parsed and prepared for further processing. Possibly is supported by other ATW tools. Does not initiate any errors or warnings during the verification phase.

**3-2** Supported PL/I Statements  
*Key to tables*

- (V)erification (O)nly — parsed and skipped immediately. Further processing is not possible, i.e. not supported by ATW tools. Verification warning is initiated.
- (N)ot Verified — if the ATW parser meets such an entity it stops and verification is considered as erroneous.
- **IA** — Interactive Analysis. Interactive Analysis (S)upport means that the entity will be displayed in the HyperView window and all the HyperView analysis tools will be available. In fact, every entity which is verified by the ATW parser is supported in HyperView. If the entity is (N)ot supported then it is either (N)ot Verified or is skipped by parser without creating an appropriate node in the parser tree. In this way, ‘N’ in the HyperView column corresponds to (N) or (VO) in the parser column.
- **CA/GDF** — Change Analyzer and Global Data Element Flow. Similarly to HyperView, it contains ‘S’ for (S)upported statements and ‘N’ for the ones, which are (N)ot supported.

## Supported IBM Visual Age PL/I 2.0 statements

### Problem data types

PL/I language construction	Parser	IA	CA/GDF
BIN FIXED	V	V	S
BIN FLOAT	V	V	S
DEC FIXED	V	V	S
DEC FLOAT	V	V	S
CHAR	V	V	S
BIT	V	V	S
PIC	V	V	S
DATE	N	N	N
GRAPHIC	VO	N	S
COMPLEX	VO	N	S
VARYING	VO	N	S

### Program-control data types

PL/I language construction	Parser	IA	CA/GDF
----------------------------	--------	----	--------

**3-4** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

LABEL attribute	V	V	N
Label data	PS <sup>a</sup>	PS	N
GENERIC attribute	PS <sup>b</sup>	PS	S
Entry constants (functions and procedures)	V	V	S
Entry variables	V	V	N
OPTIONS option	V	V	S
Returns option and attribute	V	V	S

- a. The in-place initialization of an array of labels is not supported.
- b. Incorrect work in the case of using ENTRY, POINTER in the generic-descriptor

***Aggregate types and attributes***

<b>PL/I language construction</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
Arrays	V	V	S
Structures	V	V	S
Structures with array field	V	V	S
Array of structures	V	V	S
Select from structure of arrays	V	V	S

LIKE attribute	V	V	S
----------------	---	---	---

**Expressions and references**

PL/I language construction		Parser	IA	CA/GDF	
Pointer operations	+	VO	N	N	
	-	VO	N	N	
	compare	V	V	S	
	built-ins	VO	N	N	
Arithmetic operations	+	Unary	V	V	S
		Binary	V	V	S
	-	Unary	V	V	S
		Binary	V	V	S
	*	V	V	S	
	/	V	V	S	
	**	V	V	S	
	+=	N	N	N	
	-=	N	N	N	

**3-6** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

	* =	N	N	N
	/ =	N	N	N
	** =	N	N	N
Bit operations		V	V	S
	&	V	V	S
	^	V	V	S
	=	N	N	N
	&=	N	N	N
Comparison operations	<	V	V	S
	^<	V	V	S
	<=	V	V	S
	=	V	V	S
	^=	V	V	S
	>=	V	V	S
	>	V	V	S
	^>	V	V	S
Concatenation		V	V	S
	=	N	N	N



Array expressions	+	Unary	V	V	S
		Binary	V	V	S
	-	Unary	V	V	S
		Binary	V	V	S
	*		V	V	S
	/		V	V	S
	**		V	V	S
			V	V	S
	&		V	V	S
	^		V	V	S
	<		V	V	S
	^<		V	V	S
	<=		V	V	S
	=		V	V	S
	^=		V	V	S
	>=		V	V	S
Structure expressions	=		V	V	S
	= by name		V	V	S

**Statements**

**3-8** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

PL/I language construction	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
ALLOCATE	• <b>Format</b>	V	V	S	<i>op1</i>	<i>op2</i>	<i>cast, comp, comp+, comp*, move, const.move, const.cast</i>		
Assignment	• <b>Format</b>	V	V	S					
BEGIN	• <b>Format</b>	V	V	S					
CALL	• <b>Format</b> <i>CALL &lt;id-1   literal1&gt;</i>	V	V	S	<i>port</i>	<i>id-1</i>	<i>extern.call</i>	<i>refers to id-1  literal entry point or decision in case entry variable</i>	<i>Program Calls Program Entry Point, Program Calls Decision</i>
CLOSE	• <b>Format</b>	VO	N	S					
DECLARE	• <b>Format</b>	V	V	S					
DELAY	• <b>Format</b>	VO	N	S					
DELETE	• <b>Format</b> <i>DELETE FILE &lt;file-ref&gt;</i>	VO	N	S	<i>port</i>	<i>file-ref</i>		<i>Refers to file-ref DataPort or Decision</i>	<i>Program Deletes dataport, program Deletes decision</i>
DISPLAY	• <b>Format</b>	V	V	S					
DO (type 1: DO;)	• <b>Format</b>	V	V	S					

PL/I language construction	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
DO (type 2: DO WHILE/UNTIL)	• Format	V	V	S					
DO (type 3: DO <.>=.. TO ... BY ... WHILE .. UNTIL ...REPEAT ...)	• Format	V	V	S					
DO (type 4: DO <.>=...,... TO.. BY.., WHILE ..., UNTIL .., REPEAT ...)	• Format	V	V	S					
END	• Format	V	V	S					
ENTRY	• Format	V	V	S					
EXIT	• Format	VO	N	S					
FETCH	• Format	VO	N	S					
FLUSH	• Format	VO	N	S					
FORMAT	• Format	VO	N	S					
FREE	• Format	V	V	S					

**3-10** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

PL/I language construction	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
GET	<ul style="list-style-type: none"> <li><b>Format</b> <i>GET FILE &lt;file-ref&gt;</i></li> </ul>	V	V	S	<i>port</i>	<i>file-ref</i>	<i>extern.get</i>	<i>Refers to file-ref DataPort or Decision</i>	<i>Program reads dataport, program reads decision</i>
GOTO	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
IF	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
LEAVE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
LOCATE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	VO	N	S					
null statement	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
ON	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
OPEN	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	VO	N	S					
OTHERWISE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S					
PROCEDURE	<ul style="list-style-type: none"> <li><b>Format</b></li> </ul>	V	V	S			<i>uses parameters</i>	<i>Refers to PROGRAM in case of upper level of procedure</i>	
PUT	<ul style="list-style-type: none"> <li><b>Format</b> <i>PUT FILE &lt;file-ref&gt;</i></li> </ul>	V	V	S	<i>port</i>	<i>file-ref</i>	<i>extern.put</i>	<i>Refers to file-ref DataPort or Decision</i>	<i>Program Inserts dataport, program Inserts decision</i>

PL/I language construction	Format	Parser	IA	Sup.	CA/GDF			Repository	
					Pairs		Relation	Entities	Relations
READ	<ul style="list-style-type: none"> <li>Format</li> </ul> <i>READ FILE &lt;file-ref&gt;</i>	V	S	S	<i>port</i>	<i>file-ref</i>		<i>Refers to file-ref DataPort or Decision</i>	<i>Program Reads dataport, program Reads decision</i>
RELEASE	<ul style="list-style-type: none"> <li>Format</li> </ul>	VO	N	S					
RETURN	<ul style="list-style-type: none"> <li>Format</li> </ul>	V	V	S					
REVERT	<ul style="list-style-type: none"> <li>Format</li> </ul>	VO	N	S					
REWRITE	<ul style="list-style-type: none"> <li>Format</li> </ul> <i>REWRITE FILE &lt;file-ref&gt;</i>	V	S	S	<i>port</i>	<i>file-ref</i>		<i>Refers to file-ref DataPort or Decision</i>	<i>Program Updates dataport, program Updates decision</i>
SELECT	<ul style="list-style-type: none"> <li>Format</li> </ul>	V	V	S					
SIGNAL	<ul style="list-style-type: none"> <li>Format</li> </ul>	V	V	S					
STOP	<ul style="list-style-type: none"> <li>Format</li> </ul>	V	V	S					
WHEN	<ul style="list-style-type: none"> <li>Format</li> </ul>	V	V	S					
WRITE	<ul style="list-style-type: none"> <li>Format</li> </ul> <i>WRITE FILE &lt;file-ref&gt;</i>	V	S	S	<i>port</i>	<i>file-ref</i>		<i>Refers to file-ref DataPort or Decision</i>	<i>Program Inserts dataport, program Inserts decision</i>

**3-12** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

***Built-in functions and pseudovariables***

<b>PL/I language construction</b>	<b>Parser</b>	<b>IA</b>	<b>CA/GDF</b>
ABS	V	V	S
ACOS	V	V	S
ADD	V	V	S
ADDR	V	V	S
ALL	V	V	S
ALLOCATION	VO	N	S
ANY	V	V	S
ASIN	V	V	S
ATAN	V	V	S
ATAND	V	V	S
ATANH	V	V	S
BINARY	V	V	S
BINARYVALUE	VO	N	S
BIT	V	V	S
BOOL	V	V	S
CEIL	V	V	S

CHARACTER	V	V	S
COMPLEX	VO	N	S
CONJG	VO	N	S
COS	V	V	S
COSD	V	V	S
COSH	V	V	S
COTAN	VO	N	S
COTAND	VO	N	S
COUNT	VO	N	S
CURRENTSTORAGE	V	V	S
DATAFIELD	V	V	S
DATE	V	V	S
DATETIME	V	V	S
DECIMAL	V	V	S
DIMENSION	V	V	S
DIVIDE	V	V	S
EMPTY	VO	N	S
ENTRYADDR	VO	N	S
ERF	V	V	S
ERFC	V	V	S

**3-14** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

EXP	V	V	S
FIXED	V	V	S
FLOAT	V	V	S
FLOOR	V	V	S
GRAPHIC	VO	N	S
HBOUND	V	V	S
HIGH	V	V	S
IMAG	VO	N	S
INDEX	V	V	S
LBOUND	V	V	S
LENGTH	V	V	S
LINENO	VO	N	S
LOG	V	V	S
LOG2	V	V	S
LOG10	V	V	S
LOW	V	V	S
MAX	V	V	S
MIN	V	V	S
MOD	V	V	S
MPSTR	V	V	S



MULTIPLY	V	V	S
NULL	V	V	S
OFFSET	VO	N	S
ONCHAR	V	V	S
ONCODE	V	V	S
ONCOUNT	VO	N	S
ONFILE	VO	N	S
ONKEY	VO	N	S
ONLOC	V	V	S
ONSOURCE	V	V	S
ONSUBCODE	VO	N	S
PLIDUMP	VO	N	S
PLIRETC	VO	N	S
PLIRETV	V	V	S
PLISRTA	VO	N	S
PLISRTB	VO	N	S
PLISRTC	VO	N	S
PLISRTD	VO	N	S
POINTER	VO	N	S
POINTERADD	VO	N	S

**3-16** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

POINTVALUE	VO	N	S
PRECISION	V	V	S
PRESENT	VO	N	S
PROD	V	V	S
REAL	VO	N	S
REPEAT	V	V	S
ROUND	V	V	S
SAMEKEY	VO	N	S
SIGN	V	V	S
SIN	V	V	S
SIND	V	V	S
SINH	V	V	S
SQRT	V	V	S
STORAGE	V	V	S
STRING	V	V	S
SUBSTR	V	V	S
SUM	V	V	S
SYSNULL	V	V	S
SYSTEM	VO	N	S
TAN	V	V	S

TAND	V	V	S
TANH	V	V	S
TIME	V	V	S
TRANSLATE	V	V	S
TRUNC	V	V	S
UNSPEC	V	V	S
VERIFY	V	V	S

**3-18** Supported PL/I Statements  
*Supported IBM Visual Age PL/I 2.0 statements*

# Supported SQL Statements



**T**his chapter contains detailed information SQL and DDL support. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported DDL statements

### Supporting DDL in ATW

DDL is static part of SQL used to describe the data structure and the references between data. ATW supports such DDL statements as: CREATE TABLE, CREATE INDEX, ALTER TABLE.

During DDL verification process, ATW performs filling DBSchema. DBSchema is the special database to store information about user tables, indices, keys, and relationships.

**4-2** Supported SQL Statements  
*Supported DDL statements*

**Supported DDL statements**

The following table contains the list of DDL statements supported by ATW.

Statement	Parsed	Analyzed	Java
COMMIT	Yes	Yes	Yes
DELETE	Yes	Yes	Yes
INSERT	Yes	Yes	Yes
PROCESS SQL	Yes	Yes	No
ROLLBACK	Yes	Yes	Yes
SELECT	Yes	Yes	Yes
UPDATE	Yes	Yes	Yes

Statement	Parser	IA	Notes
ALLOCATE DESCRIPTOR	VO	N	
ALTER TABLE	V	S	
BEGIN/END DECLARE SECTION	VO	N	
CHARTOROWID function	VO*	N	
CLOSE CURSOR	V	S	
COMMENT ON COLUMNS	V	S	
COMMENT ON TABLE	V	S	
COMMIT	V	S	
CONNECT	V	S	
CREATE CHARACTER SET	VO	N	

CREATE DATABASE	VO	N	
CREATE INDEX	V	S	
CREATE SCHEMA	VO	N	
CREATE TABLE	V	S	
CREATE TABLESPACE	VO	N	
CREATE VIEW	V	S	
DECLARE for dynamic statement	VO*	N	Dynamic SQL
DECLARE CURSOR for dynamic statement	VO*	N	Dynamic SQL
DECLARE CURSOR for select statement	V	S	
DECLARE TABLE	V	S	
DECODE function	VO*	N	
DELETE	V	S	
DROP INDEX	V	S	
DROP SCHEMA	VO	N	
DROP TABLE	V	S	
DROP TABLESPACE	VO	N	
DROP VIEW	VO	N	
EXECUTE	VO*	N	Dynamic SQL
FETCH	V	S	
GRANT	V*	S	
INCLUDE	VO	N	Copybook expansion - synonym to COBOL COPY
INSERT	V	S	

**4-4** Supported SQL Statements  
*Supported DDL statements*

JOIN in SELECT statement	V*	S	
LOCK TABLE	V*	S	
NVL function	V*	N	
OPEN CURSOR	V	S	
Oracle specific data types	V	S	
REVOKE	V*	S	
ROLLBACK	V	S	
ROWIDTOCHAR function	V	N	
SELECT	V	S	
SET SESSION AUTHORIZATION	V*	N	
SET TRANSACTION LEVEL	V*	N	
SET TRANSACTION OPTION	V*	N	
SET VARIABLE	V*	N	
TO_DATE function	V*	N	
TO_NUMBER function	V*	N	
UNLOCK TABLE	V*	S	
UPDATE	V	S	
WHENEVER	V	S	

**Note:** In this table, “\*” (asterisk) after “V” or “VO” in the “Parser” column means that the nature of the statement may suggest a more full support, which is being implemented.

***Constraints of DDL Support in ATW***

***Unsupported DB2 functions***

ABS	ACOS	ASCII	ASIN	ATAN
-----	------	-------	------	------



ATAN2	BLOB	CEILING	CHAR	CHR
CLOB	COALESCE	COS	COT	COUNT_BIG
DAY	DAYNAME	DAYOFWEEK	DAYOFYEAR	DAYS
DBCLOB	DECIMAL	DEGREES	DIFFERENCE	DIGITS
DOUBLE	EVENT_MON_STATE	EXP	FLOAT	FLOOR
GENERATE_UNIQUE	GRAPHIC	GROUPING	HEX	INSERT
ULIAN_DAY	LCASE	LEFT	LENGTH	LN
LOCATE	LOG	LOG10	LONG_VARCHAR	LONG_VARGRAPHIC
LTRIM	MICROSECOND	MIDNIGHT_SECONDS	MINUTE	MOD
MONTH	MONTHNAME	NODENUMBER	NULLIF	PARTITION
POSSTR	POWER	QUARTER	RADIANS	RAISE_ERROR
RAND	REAL	REPEAT	REPLACE	RIGHT
ROUND	RTRIM	SECOND	SIGN	SIN
SMALLINT	SOUNDEX	SPACE	STDDEV	TABLE_NAME
TABLE_SCHEMA	TAN	TIME	TIMESTAMP	TIMESTAMP_ISO
TIMESTAMPDIFF	TRANSLATE	TRUNCATE	UCASE	VALUE
VARCHAR	VARIANCE	VARGRAPHIC	WEEK	YEAR

***Constraints of supporting DDL statements***

<b>DDL statement</b>	<b>Constraints of supporting</b>
Alter Bufferpool	not supported
Alter Nodegroup	not supported

**4-6** Supported SQL Statements  
Supported DDL statements

Alter Table	<p><b>UNIQUE CONSTRAINT</b> in column definition — no info in DBSchema  <b>CHECK CONSTRAINT</b> in column definition — no info in DBSchema  <b>PARTITIONIG KEY</b> — not supported  <b>DATA CAPTURE</b> clause — not supported  <b>Lob options clause</b> in column definition — not supported  Default value in column definition — constant only</p>
Alter Tablespace	not supported
Begin Declare Section	Parsing stage is only supported
Call	not supported
Close Cursor	<b>WITH RELEASE</b> clause — not supported
Comment	<b>Comment On Column</b> and <b>Comment On Table</b> statements are only supported
Compound SQL	not supported
Connect	<p>In current version the next syntax is supported:</p> <pre>CONNECT TO <i>host-variable-1</i> (IDENTIFIED BY <i>host-variable-2</i>),</pre> <p>where <i>host-variable-1</i> is database name, and <i>host-variable-2</i> is user name.</p>
Create Alias	not supported
Create Bufferpool	not supported
Create Distinct Type	not supported
Create Event Monitor	not supported
Create Function	not supported
Create Nodegroup	not supported
Create Procedure	not supported
Create Schema	<p>Parsing stage is only supported.  The next syntax is supported:</p> <pre>CREATE SCHEMA <i>schema-name</i></pre>

Create Table	<b>DATE CAPTURE</b> clause — not supported <b>PARTITIONIG KEY</b> — not supported <b>Tablespace options</b> clause — not supported <b>Lob options clause</b> in column definition — not supported Default value in column definition — constant only
Create Trigger	not supported
Create View	<b>WITH</b> clause — not supported
Describe	not supported
Disconnect	not supported
Drop	The next cases are supported: <ul style="list-style-type: none"> <li>• <b>DROP TABLE</b></li> <li>• <b>DROP INDEX</b></li> <li>• <b>DROP TABLESPACE</b></li> <li>• <b>DROP VIEW</b></li> <li>• <b>DROP SCHEMA</b></li> </ul>
End Declare Section	Parsing stage is only supported
Execute	<b>USING DESCRIPTOR</b> clause — not supported
Execute Immediate	Parsing stage is only supported
Explain	not supported
Fetch	<b>USING DESCRIPTOR</b> construction — not supported
Free Locator	not supported
Grant	Parsing stage is only supported. <b>GRANT ON TABLE</b> supported only
Insert	<b>WITH</b> clause — not supported
Lock Table	Parsing stage is only supported
Open Cursor	For Open statement with <b>USING</b> clause — parsing stage is only supported

**4-8** Supported SQL Statements  
*Supported SQL Statements*

Prepare	<b>INTO descriptor</b> construction — not supported
Release	not supported
Rename Table	not supported
Revoke	Parsing stage is only supported. <b>REVOKE ON TABLE</b> supported only
Select	<b>name AS new-name</b> construction — parsing stage is only supported <b>grouping sets</b> in <b>group by</b> — not supported <b>ROLLUP</b> in <b>group by</b> — not supported <b>CUBE</b> in <b>group by</b> — not supported select with <b>JOIN</b> — parser only <b>SELECT FROM subselect</b> construction — parsing stage is only supported select with <b>UNION</b> — parsing stage is only supported select with <b>EXCEPT</b> — parsing stage is only supported select with <b>INTERSECT</b> — parsing stage is only supported <b>for read only</b> construction — parsing stage is only supported <b>for update</b> construction — parsing stage is only supported <b>optimize for</b> construction — parsing stage is only supported <b>VALUES</b> clause in select — not supported <b>WITH</b> construction — not supported <b>sort-key-expression in order by</b> — not supported
Update	The following format of Update statement is not supported:  UPDATE table SET (<sequence of columns>) = (<sequence of values>)

## **Supported SQL Statements**

The following table contains the list of SQL statements supported by ATW.

**Supported ANSI SQL-92 Statements****Data types**

<b>Data type</b>	<b>Host variable data type</b>	<b>Parser</b>
BLOB	01 name USAGE IS SQL TYPE IS BLOB(n)	N
CHARACTER	01 name PIC X(n)	V
CLOB	01 name USAGE IS SQL TYPE IS CLOB(n)	N
DATE	01 name PIC X(n)	V
DECIMAL	01 name PIC S9(m)V9(n) COMP-3 (or PACKED DECIMAL)	V
DOUBLE	01 name COMP-2	V
INTEGER	01 name PIC S9(9) COMP (or COMP-4)	V
LONG VAR-CHAR	01 name 49 length PIC S9(4) COMP-5 49 data PIC X(n) 4001<=n<=32700	V
REAL	01 name COMP-1	V
SMALLINT	01 name PIC S9(4) COMP (or COMP-4)	V
TIME	01 name PIC X(n)	V
TIMESTAMP	01 name PIC X(n)	V
VARCHAR	01 name 49 length PIC S9(4) COMP-5 49 data PIC X(n) 1<=n<=4000	V

**4-10** Supported SQL Statements  
Supported SQL Statements

**Special registers**

Registers	Parser
CURRENT DATE	V
CURRENT DEGREE	V
CURRENT EXPLAIN MODE	V
CURRENT EXPLAIN SNAPSHOT	V
CURRENT FUNCTION PATH	V
CURRENT NODE	V
CURRENT QUERY OPTIMIZATION	V
CURRENT SERVER	V
CURRENT TIME	V
CURRENT TIMESTAMP	V
CURRENT TIMEZONE	V
USER	N

**Comparisons**

Predicates	Format	Parser
BASIC	<ul style="list-style-type: none"> <li><b>Format</b> <i>expr-1</i> &lt;=   &lt;&gt;   &gt;   &lt;   &gt;=   &lt;= &gt; <i>expr-2</i></li> </ul>	V
QUANTIFIED	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>expr</i> &lt;=   &lt;&gt;   &gt;   &lt;   &gt;=   &lt;= &gt; &lt; SOME   ANY   ALL &gt; (fullselect)</li> </ul>	V
	<ul style="list-style-type: none"> <li><b>Format 2</b> (<i>expr-1, expr-2,...</i>) &lt;=   &lt;&gt;   &gt;   &lt;   &gt;=   &lt;= &gt; &lt; SOME   ANY   ALL &gt; (fullselect1)</li> </ul>	N

Predicates	Format	Parser
BETWEEN	<ul style="list-style-type: none"> <li><b>Format</b> <i>expr-1 [NOT] BETWEEN expr-2 AND expr-3</i></li> </ul>	V
EXISTS	<ul style="list-style-type: none"> <li><b>Format</b> [NOT] EXISTS (fullselect)</li> </ul>	V
IN	<ul style="list-style-type: none"> <li><b>Format 1-</b> <i>expr-1 [NOT] IN (expr-2, expr-3,...)</i></li> </ul>	V
	<ul style="list-style-type: none"> <li><b>(Format 2)</b> <i>(expr-1, expr-2,...) [NOT] IN (expr-3, expr-4,...)</i></li> </ul>	N
LIKE	<ul style="list-style-type: none"> <li><b>Format 1</b> <i>&lt; constant   host-variable   column-name &gt; [NOT] LIKE pattern-expr</i></li> </ul>	V
	<ul style="list-style-type: none"> <li><b>Format 2</b> <i>special register [NOT] LIKE pattern-expr</i></li> </ul>	V*
	<ul style="list-style-type: none"> <li><b>Format 3</b> <i>scalar function [NOT] LIKE pattern-expr</i></li> </ul>	V**
	<ul style="list-style-type: none"> <li><b>Format 4</b> <i>large object locator [NOT] LIKE pattern-expr</i></li> </ul>	N
	<ul style="list-style-type: none"> <li><b>Format 5</b> <i>expr LIKE scalar function whose operands are constant / special register / host variable</i></li> </ul>	V*
	<ul style="list-style-type: none"> <li><b>Format 6</b> <i>expr LIKE special register</i></li> </ul>	V*
	<ul style="list-style-type: none"> <li><b>Format 7</b> <i>expr-1 LIKE &lt; constant   host-variable   expr-2 concatenating any of the above &gt;</i></li> </ul>	V
NULL	<ul style="list-style-type: none"> <li><b>Format</b> <i>expr IS [NOT] NULL</i></li> </ul>	V

\* Works only with supported special registers

**4-12** Supported SQL Statements  
Supported SQL Statements

\*\* Works only with supported scalar functions

**Essential Statements**

Statement	Format	Parser	IA	Repository	
				Entities	Relations
BEGIN DECLARE SECTION	<ul style="list-style-type: none"> <li><b>Format</b> BEGIN DECLARE SECTION</li> </ul>	V	S		
CLOSE CUR- SOR	<ul style="list-style-type: none"> <li><b>Format 1</b> CLOSE <i>cursor-name</i></li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> CLOSE <i>cursor-name</i> WITH RELEASE</li> </ul>	N	N		
COMMIT	<ul style="list-style-type: none"> <li><b>Format</b> COMMIT</li> </ul>	V	S		
CONNECT	<ul style="list-style-type: none"> <li><b>Format</b> CONNECT TO &lt;<i>server-name</i>   <i>host-variable-1</i>&gt; USER &lt;<i>authorization-name</i>   <i>host-variable-2</i>&gt; USING &lt;<i>password</i>   <i>host-variable-3</i>&gt;</li> </ul>	N	N		
DECLARE CURSOR	<ul style="list-style-type: none"> <li><b>Format</b> DECLARE <i>cursor-name</i> CURSOR FOR &lt;<i>select-stmt</i>   <i>stmt-name</i>&gt;</li> </ul>	V	S		
DELETE	<ul style="list-style-type: none"> <li><b>Format 1</b> DELETE FROM <i>table-name</i> WHERE <i>search-condition</i></li> </ul>	V	S	refers to table-name (Table)	Program Deletes from Table
	<ul style="list-style-type: none"> <li><b>Format 2</b> DELETE FROM <i>table-name</i> WHERE CURRENT OF <i>cursor-name</i></li> </ul>	V	S	refers to table-name (Table)	Program Deletes from Table
DISCONNECT	<ul style="list-style-type: none"> <li><b>Format</b> DISCONNECT <i>server-name/host-variable/ALL/CURRENT</i></li> </ul>	N	N		
END DECLARE SECTION	<ul style="list-style-type: none"> <li><b>Format</b> END DECLARE SECTION</li> </ul>	V	S		



Statement	Format	Parser	IA	Repository	
				Entities	Relations
FETCH	<ul style="list-style-type: none"> <li><b>Format 1</b> FETCH <i>cursor-name</i> INTO <i>host-variable</i></li> </ul>	V	S	refers to table-name (Table) from corresponding Declare Cursor statement	Program Reads Table
	<ul style="list-style-type: none"> <li><b>Format 2</b> FETCH <i>cursor-name</i> USING DESCRIPTOR <i>descriptor-name</i></li> </ul>	N	N		
INCLUDE	<ul style="list-style-type: none"> <li><b>Format 1</b> INCLUDE SQLCA</li> </ul>	V	S	refers to SQLCA (Copybook)	Cobol Includes Copybook
	<ul style="list-style-type: none"> <li><b>Format 2</b> INCLUDE SQLDA</li> </ul>	V	S		
INSERT	<ul style="list-style-type: none"> <li><b>Format 1</b> INSERT INTO <i>table-name</i> (<i>col-name1</i>, <i>col-name2</i>,...) VALUES [&lt; <i>expr-11</i>   NULL &gt;]</li> </ul>	V	S	refers to table-name (Table)	Program Inserts into Table
	<ul style="list-style-type: none"> <li><b>Format 2</b> INSERT INTO <i>table-name1</i> SELECT <i>expr</i> FROM <i>table-name2</i></li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b> INSERT INTO <i>table-name</i> (<i>col-name1</i>, <i>col-name2</i>,...) VALUES [&lt; <i>expr</i>   NULL ,...&gt;] WITH <i>common-table-expr</i></li> </ul>	N	N		
OPEN CURSOR	<ul style="list-style-type: none"> <li><b>Format 1</b> OPEN <i>cursor-name</i></li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 2</b> OPEN <i>cursor-name</i> USING <i>host-variable1</i>, <i>host-variable2</i>,...</li> </ul>	V	S		
	<ul style="list-style-type: none"> <li><b>Format 3</b> OPEN <i>cursor-name</i> USING DESCRIPTOR <i>descriptor-name</i></li> </ul>	N	N		
ROLLBACK	<ul style="list-style-type: none"> <li><b>Format</b> ROLLBACK</li> </ul>	V	S		

**4-14** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA	Repository	
				Entities	Relations
SELECT	<ul style="list-style-type: none"> <li>• <b>Format 1</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 2</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> AS <i>correlation-name</i> WHERE <i>expr</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 3</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr</i> GROUP BY <i>grouping-expr</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 4</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr</i> ORDER BY <i>column-name</i>1, ... &lt; ASC   DESC &gt;</li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 5</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE [NOT] &lt; <i>predicate</i>   <i>search-condition</i> &gt;</li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 6</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE &lt; <i>predicate-1</i>   <i>search-condition</i> &gt; &lt; AND   OR &gt; &lt; <i>predicate-2</i>   <i>search-condition2</i> &gt;</li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 7</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr-1</i> &lt;=   &lt;&gt;   &lt;   &gt;   &lt;=   &gt;= &gt; <i>expr-2</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 8</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr-1</i> BETWEEN <i>expr-2</i> AND <i>expr-3</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 9</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>expr</i> [NOT] IN (<i>expr-2</i>, ...)</li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li>• <b>Format 10</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i> WHERE <i>match-expr</i> [NOT] LIKE <i>pattern-expr</i></li> </ul>	V	S	refers to table-name (Table)	Program Reads Table

Statement	Format	Parser	IA	Repository	
				Entities	Relations
	<ul style="list-style-type: none"> <li> <b>Format 11</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-ref</i>  WHERE <i>expr</i> IS [NOT] NULL </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 12</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> [&lt; UNION   UNION ALL &gt;] [<i>subselect</i>   <i>fullselect</i>] </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 13</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> [&lt; EXCEPT   EXCEPT ALL &gt;] [<i>subselect</i>   <i>fullselect</i> &gt;] </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 14</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> [&lt; INSPECT   INSPECT ALL &gt;] [<i>subselect</i>   <i>fullselect</i> &gt;] </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 15</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> [&lt; INNER   LEFT   RIGHT   FULL &gt;] JOIN <i>expr-2</i> ON </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 16</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> FOR [&lt;UPDATE   READ ONLY   OPTIMIZE&gt;] FOR </li> </ul>	V	S	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 17</b>  SELECT <i>column-name</i> AS <i>new-column-name</i> INTO <i>host-variable</i> FROM <i>table-def</i> WHERE <i>expr</i> </li> </ul>	N	N	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 18</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> GROUP BY [&lt;GROUPING SET   ROLLUP   CUBE&gt;](...) </li> </ul>	N	N	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 19</b>  SSELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> VALUES [&lt;expression   NULL&gt;] </li> </ul>	N	N	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li> <b>Format 20</b>  SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i>  WHERE <i>expr</i> ORDER BY <i>sort-key-expression</i> </li> </ul>	N	N	refers to table-name (Table)	Program Reads Table

**4-16** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA	Repository	
				Entities	Relations
	<ul style="list-style-type: none"> <li><b>Format 21</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i> WHERE <i>expr</i> &lt; =   &gt;   &lt;   &lt;&gt;   &gt;= &gt; &lt; SOME   ANY   ALL &gt;</li> </ul>	N	N	refers to table-name (Table)	Program Reads Table
	<ul style="list-style-type: none"> <li><b>Format 22</b> SELECT &lt; <i>table-name</i>   <i>view-name</i>   <i>correlation-name</i>   <i>column-name</i> &gt; INTO <i>host-variable</i> FROM <i>table-def</i> WHERE EXISTS <i>expr</i></li> </ul>	N	N	refers to table-name (Table)	Program Reads Table
UPDATE	<ul style="list-style-type: none"> <li><b>Format 1</b> UPDATE <i>table-name</i> SET <i>column-name1</i> = &lt; <i>expression1</i>   NULL, ... &gt; WHERE <i>search-condition</i></li> </ul>	V	S	refers to table-name (Table)	Program Updates Table
	<ul style="list-style-type: none"> <li><b>Format 2</b> UPDATE <i>table-name</i> WHERE CURRENT OF <i>cursor-name</i></li> </ul>	V	S	refers to table-name (Table)	Program Writes Table
	<ul style="list-style-type: none"> <li><b>Format 3</b> UPDATE <i>table-name</i> SET (<i>column-name1</i>, <i>column-name2</i>,...) = (&lt;<i>expression1</i>   NULL &gt;, &lt;<i>expression2</i>   NULL&gt;, ...) WHERE <i>search-condition</i></li> </ul>	N	N	refers to table-name (Table)	Program Writes Table
WHENEVER	<ul style="list-style-type: none"> <li><b>Format</b> WHENEVER &lt; <i>Not found</i>   <i>SQLerror</i>   <i>SQLwarning</i> &gt; &lt;CONTINUE   GOT <i>host-label</i> &gt;</li> </ul>	V	S		

**Essential dynamic SQL statements**

Statement	Format	Parser	IA
PREPARE	<ul style="list-style-type: none"> <li><b>Format 1</b> PREPARE ... INTO <i>descriptor-name</i> FROM...</li> </ul>	N	N
	<ul style="list-style-type: none"> <li><b>Format 2</b> PREPARE <i>statement-name</i> FROM <i>host-variable</i></li> </ul>	V	S

Statement	Format	Parser	IA
EXECUTE	<ul style="list-style-type: none"> <li><b>Format 1</b> EXECUTE <i>statement-name</i> [USING <i>host-variable</i>]</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> EXECUTE <i>statement-name</i> USING DESCRIPTOR <i>descriptor-name</i></li> </ul>	N	N
EXECUTE IMMEDIATE	<ul style="list-style-type: none"> <li><b>Format</b> EXECUTE IMMEDIATE <i>host-variable</i></li> </ul>	V	S

**Not essential SQL statements**

Statement	Format	Parser	IA
ALTER BUFFERPOOL	<ul style="list-style-type: none"> <li><b>Format</b> ALTER BUFFERPOOL <i>bufferpool-name</i> &lt;[ NODE <i>node-integer</i>] SIZE <i>number-of-pages</i>   [NOT] EXTENDED STORAGE   ADD NODEGROUP <i>nodegroup-name</i>&gt;</li> </ul>	N	N
ALTER NODEGROUP	<ul style="list-style-type: none"> <li><b>Format</b> ALTER NODEGROUP <i>nodegroup-name</i> &lt; ADD &lt;NODE   NODES&gt; <i>nodes-clause</i> [&lt;LIKE NODE <i>node-number</i>   WITHOUT TABLESPACES&gt;]   DROP &lt;NODE   NODES&gt; <i>nodes-clause</i>&gt;</li> </ul>	N	N
ALTER TABLE	<ul style="list-style-type: none"> <li><b>Format 1</b> ALTER TABLE <i>table-name</i> &lt; ADD &lt; [COLUMN] <i>column definition</i>   <i>unique-constraint</i>   <i>check-constraint</i> &gt;   DROP &lt;PRIMARY KEY   &lt; FOREIGN KEY   UNIQUE   CHECK   CONSTRAINT &gt; <i>constraint - name</i> &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format 2</b> ALTER TABLE <i>table-name</i> &lt; ADD PARTITIONING KEY   DROP PARTIONING KEY &gt; [DATA CAPTURE &lt;NONE   CHANGES&gt;] [<i>column definition</i>] [[&lt; LOGGED   NOT LOGGED &gt;]. [NOT COMPACT   COMPACT]]\</li> </ul>	N	N
ALTER TABLESPACE	<ul style="list-style-type: none"> <li><b>Format</b> ALTER-TABLESPACE <i>tablespace-name</i> &lt; ADD <i>database-container-clause</i> [<i>on-nodes-clause</i>]   <i>system-container-clause on-nodes-clause</i>   PREFETCHSIZE <i>number-of pages</i>   BUFFERPOOL <i>bufferpool-name</i>   OVERHEAD <i>number-of-milliseconds</i>   TRANSFERRATE <i>number-of-milliseconds</i>&gt;</li> </ul>	N	N

**4-18** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA
CALL	<ul style="list-style-type: none"> <li><b>Format</b> CALL &lt;procedure-name   host-variable&gt; &lt; [ (host-variable) ]   USING DESCRIPTOR descriptor-name &gt;</li> </ul>	N	N
COMMENT ON	<ul style="list-style-type: none"> <li><b>Format 1</b> COMMENT ON &lt; &lt; ALIAS alias-name   CONSTRAINT table.name constraint.name   DISTINCT TYPE distinct-type-name   FUNCTION function-name [ data-type ]   SPECIFIC FUNCTION specific-name   INDEX index-name   NODEGROUP nodegroup-name   PACKAGE package-name   PROCEDURE procedure-name   SPECIFIC PROCEDURE specific-name   SCHEMA schema-name   TABLESPACE tablespace.name   TRIGGER trigger-name &gt; IS string-constant   &lt; table-name   view-name&gt; ( column-name IS string-constant) &gt;</li> </ul>	N	N
	<ul style="list-style-type: none"> <li><b>Format 2</b> COMMENT ON &lt; COLUMN &lt;table-name. column-name   view-name. column-name&gt;   TABLE &lt;table-name   view-name&gt; &gt; ( column-name IS string-constant) &gt;</li> </ul>	V	S
COMPOUND SQL	<ul style="list-style-type: none"> <li><b>Format</b> BEGIN COMPOUND &lt;ATOMIC   NOT ATOMIC&gt; STATIC [STOP AFTER FIRST host-variable STATEMENTS] [sql-statement] END COMPOUND</li> </ul>	N	N
CREATE ALIAS	<ul style="list-style-type: none"> <li><b>Format</b> CREATE &lt; ALIAS   SYNONYM &gt; alias-name FOR &lt; table-name   view-name   alias-name2&gt;</li> </ul>	N	N
CREATE BUFFERPOOL	<ul style="list-style-type: none"> <li><b>Format</b> CREATE BUFFERPOOL bufferpool-name [&lt;ALL NODES   NODEGROUP nodegroup-name&gt;] SIZE number-of-pages [except-on-nodes-clause] [&lt; NOT EXTENDED STORAGE   EXTENDED STORAGE &gt;]</li> </ul>	N	N
CREATE DISTINCT TYPE	<ul style="list-style-type: none"> <li><b>Format</b> CREATE DISTINCT TYPE <i>distinct-type-name</i> AS <i>source-data-type</i> WITH COMPARISONS</li> </ul>	N	N
CREATE EVENT MONITOR	<ul style="list-style-type: none"> <li><b>Format</b> CREATE EVENT MONITOR create-monitor-name FOR &lt; &lt; DATABASE   TABLES   DEADLOCKS   TABLESPACES   BUFFERPOOLS &gt;   &lt; CONNECTIONS   STATEMENTS   TRANSACTIONS &gt; [WHERE Event Condition]&gt; WRITE TO &lt;PIPE pipe-name   FILE path-name file options [&lt; MANUALSTART   AUTOSTART &gt;] [ON NODE <i>node-number</i>] [&lt;LOCAL   GLOBAL&gt;]</li> </ul>	N	N

Statement	Format	Parser	IA
CREATE FUNCTION	<ul style="list-style-type: none"> <li> <b>Format: external scalar</b>            CREATE FUNCTION function-name ( [data-type1 [AS LOCATOR]] ).            RETURNS &lt; data-type2 [AS LOCATOR]   data-type3 CAST FROM data-type4 [AS LOCATOR]&gt;.            [SPECIFIC specific-name]. EXTERNAL [NAME &lt;'string'   id&gt;].            LANGUAGE &lt;C   JAVA   OLE&gt;. PAARETER STYLE &lt;DB2SQL   DB2GENERAL&gt;. &lt;DETERMINISTIC   NOT DETERMINISTIC&gt;.            [&lt;FENCED   NOT FENCED&gt;]. [&lt;NOT NULL CALL   NULL CALL&gt;]. NO SQL. &lt;NO EXTERNAL ACTION   EXTERNAL ACTION&gt;. [&lt;NO SCRATCHPAD   SCRATCHPAD&gt;]. [&lt; NO FINAL CALL   FINAL CALL&gt;]. [&lt;ALOOW PARALLEL   DISALLOW PARALLEL&gt;]. [&lt; NO DBINFO   DBINFO &gt;]         </li> </ul>	N	N
	<ul style="list-style-type: none"> <li> <b>Format: external table</b>            CREATE FUNCTION function-name ( [data-type1 [AS LOCATOR]] ).            RETURNS TABLE column-name data-type2 [AS LOCATOR].            [SPECIFIC specific-name]. EXTERNAL [NAME &lt;'string'   id&gt;].            LANGUAGE &lt;C   JAVA   OLE&gt;. PARAMETER STYLE &lt;DB2SQL   DB2GENERAL&gt;. &lt;DETERMINISTIC   NOT DETERMINISTIC&gt;.            [&lt;FENCED   NOT FENCED&gt;]. [&lt;NOT NULL CALL   NULL CALL&gt;]. NO SQL. &lt;NO EXTERNAL ACTION   EXTERNAL ACTION&gt;. [&lt;NO SCRATCHPAD   SCRATCHPAD&gt;]. FINAL CALL. DISALLOW PARALLEL. [&lt; NO DBINFO   DBINFO &gt;]            [CARDINALITY integer]         </li> </ul>	N	N
	<ul style="list-style-type: none"> <li> <b>Format: external table</b>            CREATE FUNCTION function-name ( [data-type1] ).            RETURNS data-type2.            [SPECIFIC specific-name]. SOURCE &lt;function-name   SPECIFIC specific-name   function-name ([ data-type ])&gt;.         </li> </ul>	N	N
CREATE INDEX	<ul style="list-style-type: none"> <li> <b>Format</b>            CREATE [UNIQUE] INDEX index-name            ON table-name (column-name [&lt;ASC   DESC&gt;])         </li> </ul>	V	S
CREATE NODEGROUP	<ul style="list-style-type: none"> <li> <b>Format</b>            CREATE NODEGOUP nodegroup-name            [&lt; ON ALL NODES   ON &lt;NODE   NODES&gt; (node-number1 [TO node-number2])]         </li> </ul>	N	N
CREATE PRO- CEDURE	<ul style="list-style-type: none"> <li> <b>Format</b>            CREATE PROCEDURE procedure-name ( [ [&lt; IN   OUT   INOUT &gt;] parameter-name-data-type ] )            [SPECIFIC specific-name]. [ &lt; RESULT SETS 0   RESULT SETS integer &gt;].            EXTERNAL [NAME &lt;'string'   id&gt;].            LANGUAGE &lt;C   JAVA &gt;. PARAMETER STYLE &lt;DB2SQL   DB2GENERAL&gt;. &lt;DETERMINISTIC   NOT DETERMINISTIC&gt;.            [&lt;FENCED   NOT FENCED&gt;]. [ NULL CALL ].         </li> </ul>	N	N
CREATE SCHEMA	<ul style="list-style-type: none"> <li> <b>Format</b>            CREATE SCHEMA &lt; schema-name   AUTHORIZATION authorization-name   schema-name- AUTHORIZATION authorization-name&gt;            [schema-SQL-statement]         </li> </ul>	V	S

**4-20** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA
CREATE TABLE	<ul style="list-style-type: none"> <li><b>Format</b> CREATE TABLE table-name (&lt; column-definition   unique constraint   referential-constraint   check constraint &gt; ) [&lt; DATA CAPTURE NONE   DATA CAPTURE CHANGES &gt;] [IN tablespace-name1 tablespace options] [PARTIONING KEY ('column') [USING HASHING]] [NOT LOGGED INITIALLY]</li> </ul>	V	S
CREATE TABLESPACE	<ul style="list-style-type: none"> <li><b>Format</b> CREATE [&lt; REGULAR   LONG   TEMPORARY &gt;] TABLESPACE tablespace-name [IN [NODEGROUP] nodegroup-name] MANAGED BY [&lt;SYSTEM system-containers   DATABASE database-containers&gt;] [EXTENTSIZE number-of-pages] [PREFETCHSIZE number-of pages] [BUFFERPOOL bufferpool-name] [OVERHEAD [&lt; 24.1   number-of-milliseconds &gt;] [TRANSFERRATE [&lt; 0.9   number-of milliseconds &gt;]</li> </ul>	V	S
CREATE TRIGGER	<ul style="list-style-type: none"> <li><b>Format</b> CREATE TRIGGER trigger-name &lt; NO CASCADE BEFORE   AFTER &gt; &lt;INSERT   DELETE   UPDATE [ OF column-name ] &gt; ON table-name [REFERENCING &lt;OLD [AS] correlation-name   NEW [AS] correlation-name   OLD-TABLE [AS] id   NEW_TABLE [AS] id &gt; ] &lt; FOR EACH ROW   FOR EACH STATEMENT &gt; MODE DB2SQL triggered action</li> </ul>	N	N
CREATE VIEW	<ul style="list-style-type: none"> <li><b>Format</b> CREATE VIEW view-name [( column-name )] AS WITH common-table-expression ] fullselect WITH [&lt;CASCADED   LOCAL&gt;] CHECK OPTION]</li> </ul>	N	N
DESCRIBE	<ul style="list-style-type: none"> <li><b>Format</b> DESCRIBE statement-name INTO descriptor-name</li> </ul>	N	N
DROP	<ul style="list-style-type: none"> <li><b>Format 1</b> DROP &lt; ALIAS alias-name   BUFFERPOOL bufferpool-name   DISTINCT TYPE distinct-type-name   EVENT MONITOR event-monitor-name   FUNCTION function-name [ data-type ]   SPECIFIC FUNCTION specific-name   NODEGROUP nodegroup-name   PACKAGE package-name   PROCEDURE procedure-name   SPECIFIC PROCEDURE specific-name   TRIGGER trigger-name &gt; IS string-constant   &lt; table-name   view-name &gt; ( column-name IS string-constant ) &gt;</li> </ul>	N	N
	<ul style="list-style-type: none"> <li><b>Format 2</b> DROP &lt; INDEX index-name   TABLE table-name   VIEW view-name   TABLESPACE tablespace-name   SCHEMA schema-name &gt;</li> </ul>	V	S
EXPLAIN	<ul style="list-style-type: none"> <li><b>Format</b> EXPLAIN &lt;PLAN SELECTION   ALL   PLAN&gt; [&lt;FOR   WITH &gt; SNAPSHOT] [SET QUERYND integer] [[SET QUERYND string-constant] FOR explainable-sql-statement</li> </ul>	N	N



Statement	Format	Parser	IA
FREE LOCATOR	<ul style="list-style-type: none"> <li><b>Format</b> FREE LOCATOR variable-name</li> </ul>	N	N
GRANT	<ul style="list-style-type: none"> <li><b>Format: Database Authorities</b> GRANT &lt; BINDADD   CONNECT   CREATETAB   CREATE_NOT_FENCED   IMPLICIT_SCHEMA   DBADM &gt; ON DATABASE TO &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format: Index Privileges</b> GRANT CONTROL ON INDEX index-name TO &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format: Package Privileges</b> GRANT &lt; BIND   CONTROL   EXECUTE &gt; ON PACKAGE package-name TO &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format: Schema Privileges</b> GRANT &lt; ALTERIN   CREATEIN   DROPIN &gt; ON SCHEMA schema-name TO &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt; [WITH GRANT OPTION]</li> </ul>	V	S
	<ul style="list-style-type: none"> <li><b>Format: Table or View Privileges</b> GRANT &lt; ALL [PRIVILEGES]   ALTER   CONTROL   DELETE   INDEX   INSERT   REFERENCES [(‘column-name’)]   SELECT   UPDATE [(‘column-name’)] &gt; ON SCHEMA schema-name ON [TABLE] &lt;table-name   view-name&gt; TO &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
LOCK TABLE	<ul style="list-style-type: none"> <li><b>Format</b> LOCK TABLE table-name IN &lt; SHARE   EXCLUSIVE &gt; NODE</li> </ul>	V	S
RELEASE	<ul style="list-style-type: none"> <li><b>Format</b> RELEASE &lt;server-name   host-variable   CURRENT   ALL [SQL] &gt;</li> </ul>	N	N
RENAME TABLE	<ul style="list-style-type: none"> <li><b>Format</b> RENAME [TABLE] source-table-name TO target-identifier</li> </ul>	V	S

**4-22** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA
REVOKE	<ul style="list-style-type: none"> <li>• <b>Format 1: Database Authorities</b> REVOKE &lt; BINDADD   CONNECT   CREATETAB   CREATE_NOT_FENCED   IMPLICIT_SCHEMA   DBADM &gt; ON DATABASE FROM &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li>• <b>Format: Index Privileges</b> REVOKE CONTROL ON INDEX index-name FROM &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li>• <b>Format: Packages Privileges</b> REVOKE &lt; BIND   CONTROL   EXECUTE &gt; ON PACKAGE package-name FROM &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li>• <b>Format: Schema Privileges</b> REVOKE &lt; ALTERIN   CREATEIN   DROPIN &gt; ON SCHEMA schema-name FROM &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
	<ul style="list-style-type: none"> <li>• <b>Format: Table or View Privileges</b> REVOKE &lt; ALL [PRIVILEGES]   ALTER   CONTROL   DELETE   INDEX   INSERT   REFERENCES [(column-name)]   SELECT   UPDATE [(column-name)]&gt; &gt; ON SCHEMA schema-name ON [TABLE] &lt;table-name   view-name&gt; FROM &lt;[ &lt;USER   GROUP &gt; authorization name   PUBLIC &gt;</li> </ul>	V	S
SET CONNECTION	<ul style="list-style-type: none"> <li>• <b>Format</b> SET CONNECTION &lt;server-name   host-variable&gt;</li> </ul>	N	N
SET CONSTRAINTS	<ul style="list-style-type: none"> <li>• <b>Format</b> SET CONSTRAINTS &lt;FOR table-name &lt;OFF   IMMEDIATE CHECKED [exception-clause]   FOR table-name &lt; ALL   [ &lt; FOREIGN KEY   CHECK &gt; ] &gt; IMMEDIATE UNCHECKED</li> </ul>	N	N
SET CURRENT DEGREE	<ul style="list-style-type: none"> <li>• <b>Format</b> SET CURRENT DEGREE &lt; string-constant   host-variable &gt;</li> </ul>	N	N
SET CURRENT EXPLAIN MODE	<ul style="list-style-type: none"> <li>• <b>Format</b> SET CURRENT EXPLAIN MODE &lt; NO   YES   EXPLAIN   host-variable &gt;</li> </ul>	N	N
SET CURRENT EXPLAIN SNAPSHOT	<ul style="list-style-type: none"> <li>• <b>Format</b> SET CURRENT EXPLAIN SNAPSHOT &lt; NO   YES   EXPLAIN   host-variable &gt;</li> </ul>	N	N

Statement	Format	Parser	IA
SET CURRENT FUNCTION PATH	<ul style="list-style-type: none"> <li><b>Format</b> SET CURRENT FUNCTION PATH &lt; schema-name   SYSTEM PATH   USER   CURRENT FUNCTION PATH   host-variable   string-variable &gt;</li> </ul>	N	N
SET CURRENT PACKAGESET	<ul style="list-style-type: none"> <li><b>Format</b> SET CURRENT PACKAGESET &lt; string-constant   host-variable &gt;</li> </ul>	N	N
SET CURRENT QUERY OPTIMIZATION	<ul style="list-style-type: none"> <li><b>Format</b> SET CURRENT QUERY OPTIMIZATION &lt; 0   1   2   3   5   7   9   host-variable &gt;</li> </ul>	N	N
SET EVENT MONITOR STATE	<ul style="list-style-type: none"> <li><b>Format</b> SET EVENT MONITOR event-monitor-name STATE &lt; 0   1   host-variable ?</li> </ul>	N	N
SET	<ul style="list-style-type: none"> <li><b>Format</b> SET transition-variable &lt; expression   NULL   DEFAULT &gt;   transition-variable &lt; &lt; expression   NULL   DEFAULT &gt;   row-fullselect &gt;</li> </ul>	N	N
SIGNAL SQLSTATE	<ul style="list-style-type: none"> <li><b>Format</b> SIGNAL SQLSTATE string-constant (diagnostic -string)</li> </ul>	N	N
VALUES INTO	<ul style="list-style-type: none"> <li><b>Format</b> VALUES &lt; expression   (expression,..) &gt; INTO host-variable</li> </ul>	N	N

### Column Functions

Functions	Parser	IA
AVG	V	S
COUNT	V	S
COUNT_BIG	N	N
GROUPING	N	N

**4-24** Supported SQL Statements  
*Supported SQL Statements*

<b>Functions</b>	<b>Parser</b>	<b>IA</b>
MAX	V	S
MIN	V	S
STDDEV	N	N
SUM	V	S
VARIANCE	N	N

**Scalar Functions**

<b>Functions</b>	<b>Parser</b>	<b>IA</b>
ABS or ABSVAL	N	N
ACOS	N	N
ASCII	N	N
ASIN	N	N
ATAN	N	N
ATAN2	N	N
BLOB	N	N
CEIL or CEILING	N	N
CHAR	N	N
CHR	N	N
CLOB	N	N
COALESCE	N	N
CONCAT	V	S
COS	N	N

<b>Functions</b>	<b>Parser</b>	<b>IA</b>
COT	N	N
DATE	V	S
DAY	N	N
DAYNAME	N	N
DAYOFWEEK	N	N
DAY OF YEAR	N	N
DAYS	N	N
DBCLOB	N	N
DECIMAL	N	N
DEGREES	N	N
DIFFERENCE	N	N
DIGITS	N	N
DOUBLE	N	N
EVENT_MON_STATE	N	N
EXP	N	N
FLOAT	N	N
FLOOR	N	N
GENERATE_UNIQUE	N	N
GRAPHIC	N	N
HEX	N	N
HOUR	V	S
INSERT	N	N
INTEGER	N	N

**4-26** Supported SQL Statements  
*Supported SQL Statements*

<b>Functions</b>	<b>Parser</b>	<b>IA</b>
JULIAN_DAY	N	N
LCASE	N	N
LEFT	N	N
LENGTH	N	N
LN	N	N
LOCATE	N	N
LOG	N	N
LOG10	N	N
LONG_VARCHAR	N	N
LONG_VARGRAPHIC	N	N
LTRIM	N	N
MICROSECOND	N	N
MIDNIGHT_SECOND	N	N
MINUTE	N	N
MOD	N	N
MONTH	N	N
MONTHNAME	N	N
NODENUMBER	N	N
NULLIF	N	N
PARTITION	N	N
POSSTR	N	N
POWER	N	N
QUARTER	N	N

<b>Functions</b>	<b>Parser</b>	<b>IA</b>
RADIANS	N	N
RAISE_ERROR	N	N
RAND	N	N
REAL	N	N
REPEATE	N	N
REPLACE	N	N
RIGHT	N	N
ROUND	N	N
RTRIM	N	N
SECOND	N	N
SIGN	N	N
SIN	N	N
SMALLINT	N	N
SOUNDEX	N	N
SPACE	N	N
SQRT	V	S
SUBSTR	V	S
TABLE_NAME	N	N
TABLE_SCHEMA	N	N
TAN	N	N
TIME	N	N
TIMESTAMP	N	N
TIMESTAMP_ISO	N	N

**4-28** Supported SQL Statements  
Supported SQL Statements

Functions	Parser	IA
TIMESTAMPDIFF	N	N
TRANSLATE	N	N
TRUNC or TRUNCATE	N	N
UCASE	N	N
VALUE	N	N
VARCHAR	N	N
VARGRAPHIC	N	N
WEEK	N	N
YEAR	N	N

**Supported RDMS 2200 SQL Statements**

Statement	Format	Parser	IA
ALLOCATE CURSOR	<ul style="list-style-type: none"> <li><b>Format</b> <i>ALLOCATE cursor-name CURSOR [FOR RETENTION] FOR statement-name</i></li> </ul>	VO	N
ALTER TABLE	<ul style="list-style-type: none"> <li><b>Format</b> <i>ALTER TABLE table-specification edit-specification-list</i></li> </ul>	V	N
BEGIN DECLARE	<ul style="list-style-type: none"> <li><b>Format</b> BEGIN DECLARE SECTION</li> </ul>	VO	N
BEGIN THREAD	<ul style="list-style-type: none"> <li><b>Format</b> <i>BEGIN THREAD [thread-name] FOR [APPLICATION] application-name [READ   RETRIEVE   UPDATE [(DEFERRED   QUICKLOOKS   COMMANDLOOKS   NONE)]] [UDSMMSG   UDSMESSAGE   UDSMESSAGES]</i></li> </ul>	VO	N
CALL	<ul style="list-style-type: none"> <li><b>Format</b> <i>CALL [schema-name.]routine-name (parameter, ...)</i></li> </ul>	VO	N



Statement	Format	Parser	IA
CLOSE	<ul style="list-style-type: none"> <li><b>Format</b> &lt;CLOSE   RELEASE [CURSOR]&gt; <i>cursor-name</i></li> </ul>	V	S
COMMIT	<ul style="list-style-type: none"> <li><b>Format</b> COMMIT [WORK   THREAD] [ADVANCE   TERMINATE]</li> </ul>	V	S
CREATE INDEX	<ul style="list-style-type: none"> <li><b>Format</b> CREATE INDEX <i>index-name</i> ON <i>table-specification</i> (<i>sort-column-list</i>) [IN [ <i>schema-name</i>.]<i>storage-area-name</i>]</li> </ul>	V	S
CREATE SCHEMA	<ul style="list-style-type: none"> <li><b>Format</b> CREATE SCHEMA [<i>schema-name</i>] AUTHORIZATION <i>authorization-id</i> [<i>table-definition</i>   <i>view-definition</i>   <i>privilege-definition</i>]</li> </ul>	VO	N
CREATE TABLE	<ul style="list-style-type: none"> <li><b>Format</b> CREATE TABLE <i>table-specification</i> ([<i>storage-area-specification</i>, ] <i>column-definition</i> [, &lt;<i>column-definition</i>   <i>table-constraint-specification</i>   <i>storage-area-specification</i>   <i>index-specification</i>&gt;] ...)</li> </ul>	V	S
CREATE VIEW	<ul style="list-style-type: none"> <li><b>Format</b> CREATE VIEW <i>view-specification</i> [(<i>column-name</i> ...)] AS <i>query-specification</i> [WITH CHECK OPTION] [WITH ACCESS CONTROL]</li> </ul>	V	S
DEBUG	<ul style="list-style-type: none"> <li><b>Format</b> DEBUG &lt;DUMP <i>dump-option</i>   NODUMB <i>dumb-option</i>   STUB   NOSTUB   PARSE   EXECUTE&gt; <i>where dump-option is</i> &lt;COMMAND MODULE   DATA PACKET   REQUEST PACKET   RDMCA   RSP PACKET   WORK AREA   ALL   SIZES   INPUT   ECHO&gt;</li> </ul>	V	S
DECLARE CURSOR	<ul style="list-style-type: none"> <li><b>Format</b> &lt;DECLARE <i>cursor-name</i> CURSOR   DEFINE CURSOR <i>cursor-name</i>&gt; [FOR &lt;SEQUENTIAL   RANDOM   DIRECT&gt; [ACCESS]] [[FOR] RETENTION] [FOR] <i>query-expression</i> [ORDER BY <i>sort-specification-list</i>   FOR UPDATE OF <i>column-name-list</i>] [WITH DESCRIPTION]</li> </ul>	V	S
DELETE	<ul style="list-style-type: none"> <li><b>Format</b> DELETE [FROM] <i>table-specification</i> [ALL [ROWS]   WHERE &lt;<i>boolean-expr</i>   CURRENT OF <i>cursor-name</i>&gt;]</li> </ul>	V	S
DROP CURSOR	<ul style="list-style-type: none"> <li><b>Format</b> DROP CURSOR <i>cursor-name</i></li> </ul>	VO	N
DROP INDEX	<ul style="list-style-type: none"> <li><b>Format</b> DROP INDEX <i>index-name-list</i> FROM [TABLE] <i>table-specification</i></li> </ul>	VO	N

**4-30** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA
DROP PROCEDURE/FUNCTION	<ul style="list-style-type: none"> <li><b>Format</b> DROP &lt;PROCEDURE   FUNCTION&gt; [<i>schema-name</i>.]<i>routine-name</i> &lt;CASCADE   RESTRICT&gt;</li> </ul>	VO	N
DROP TABLE	<ul style="list-style-type: none"> <li><b>Format</b> DROP [PERMANENT] TABLE <i>table-list</i></li> </ul>	V	S
DROP VIEW	<ul style="list-style-type: none"> <li><b>Format</b> DRO VIEW [<i>schema-name</i>.]<i>routine-name</i> ...</li> </ul>	V	S
END DECLARE	<ul style="list-style-type: none"> <li><b>Format</b> END DECLARE SECTION</li> </ul>	VO	N
END THREAD	<ul style="list-style-type: none"> <li><b>Format</b> END THREAD [ADVANCE   TERMINATE]</li> </ul>	VO	N
EXECUTE	<ul style="list-style-type: none"> <li><b>Format</b> EXECUTE <i>statement-name</i> [USING <i>host-variable-list</i>]</li> </ul>	V	S
EXECUTE IMMEDIATE	<ul style="list-style-type: none"> <li><b>Format</b> EXECUTE IMMEDIATE &lt;:<i>embedded-variable</i>   <i>string-literal</i>&gt;</li> </ul>	VO	N
EXPLAIN	<ul style="list-style-type: none"> <li><b>Format</b> EXPLAIN <i>cursor-name</i></li> </ul>	VO	N
FETCH	<ul style="list-style-type: none"> <li><b>Format</b> FETCH [NEXT   FIRST   PRIOR   LAST   CURRENT] [FROM] <i>cursor-name</i> INTO <i>variable-specification-list</i></li> </ul>	V	N
FETCH NEXT	<ul style="list-style-type: none"> <li><b>Format</b> FETCH NEXT <i>number-of-rows</i> [FROM] <i>cursor-name</i> INTO <i>variable-specification-list</i></li> </ul>	V	N
FUNCTION	<ul style="list-style-type: none"> <li><b>Format</b> FUNCTION [<i>schema-name</i>.]<i>routine-name</i> (<i>parameter</i> ...) RETURNS <i>data-type</i> <i>stored-procedure-statement</i></li> </ul>	VO	N
GET DESCRIPTION	<ul style="list-style-type: none"> <li><b>Format</b> GET DESCRIPTION INTO <i>placeholder-list</i></li> </ul>	VO	N

Statement	Format	Parser	IA
GETERROR	<ul style="list-style-type: none"> <li><b>Format</b> GETERROR INTO <i>variable-list</i></li> </ul>	VO	N
GET PARAMETERS	<ul style="list-style-type: none"> <li><b>Format</b> GET PARAMETERS FOR [<i>schema-name</i>].<i>routine-name</i> INTO <i>placeholder-list</i></li> </ul>	VO	N
GRANT	<ul style="list-style-type: none"> <li><b>Format</b> GRANT &lt;ALL [PRIVILEGES]   <i>privilege-list</i>&gt; ON [TABLE] <i>table-specification</i> TO <i>user-id-list</i> [WITH GRANT OPTION]</li> </ul>	V	S
INSERT	<ul style="list-style-type: none"> <li><b>Format</b> INSERT [INTO] <i>table-specification</i> &lt;[(<i>column-name-list</i>)] &lt;VALUES (<i>value-list</i>)   <i>query-specification</i>&gt;   COLUMNS (<i>column-and-value</i>)&gt;</li> </ul>	V	S
LEVEL	<ul style="list-style-type: none"> <li><b>Format</b> LEVEL [RSA   RDMS] INTO <i>placeholder</i></li> </ul>	VO	N
LOCATE	<ul style="list-style-type: none"> <li><b>Format</b> LOCATE <i>cursor-name</i> ON <i>table-specification</i> USING VALUES (<i>primary-key-value</i>)</li> </ul>	VO	N
LOCK	<ul style="list-style-type: none"> <li><b>Format</b> LOCK [TABLE] <i>table-specification-list</i> [IN <i>lock-specification</i> [MODE]] [ON CONFLICT &lt;RETURN   QUEUE&gt;]</li> </ul>	VO	N
OPEN	<ul style="list-style-type: none"> <li><b>Format</b> &lt;OPEN   READY [CURSOR]&gt; <i>cursor-name</i> [USING <i>variable-list</i>]</li> </ul>	V	S
PREPARE	<ul style="list-style-type: none"> <li><b>Format</b> PREPARE <i>statement-name</i> FROM &lt;.<i>embedded-variable</i>   <i>string-literal</i>&gt;</li> </ul>	V	S
PROCEDURE	<ul style="list-style-type: none"> <li><b>Format</b> PROCEDURE [<i>schema-name</i>].<i>routine-name</i> (<i>parameter ...</i>) <i>stored-procedure-statement</i></li> </ul>	VO	N
REVOKE	<ul style="list-style-type: none"> <li><b>Format</b> REVOKE &lt;ALL [PRIVILEGES]   <i>privilege-list</i>&gt; ON <i>table-specification</i> FROM <i>user-id-list</i></li> </ul>	V	S
ROLLBACK	<ul style="list-style-type: none"> <li><b>Format</b> &lt;ROLLBACK [WORK]   OMIT THREAD&gt; [KEEP   REQUEUE   DISCARD]</li> </ul>	V	S

**4-32** Supported SQL Statements  
Supported SQL Statements

Statement	Format	Parser	IA
SELECT	<ul style="list-style-type: none"> <li><b>Format</b> SELECT [ALL   DISTINCT] &lt;select-list   *&gt; FROM table-specification-list [WHERE boolean-expression] [GROUP BY column-specification-list] [HAVING boolean-expression] INTO variable-specification-list</li> </ul>	V	S
SET	<ul style="list-style-type: none"> <li><b>Format</b> SET return-variable = [schema-name.]routine-name (parameter, ...)</li> </ul>	VO	N
SET STATISTICS	<ul style="list-style-type: none"> <li><b>Format</b> SET STATISTICS &lt;ON   OFF&gt;</li> </ul>	VO	N
UNLOAD	<ul style="list-style-type: none"> <li><b>Format</b> UNLOAD TO FILE file-name &lt;[FROM] CURSOR cursor-name   query-expression [ORDER BY sort-specification-list]&gt; [FORMAT &lt;EXTERNAL [position-and-null-list]   INTERNAL   MAPPER&gt;] [WITHOUT DESCRIPTION   WITH DESCRIPTION [ONLY]]</li> </ul>	VO	N
UNLOCK	<ul style="list-style-type: none"> <li><b>Format</b> UNLOCK table-specification-list</li> </ul>	VO	N
UPDATE ALL	<ul style="list-style-type: none"> <li><b>Format</b> UPDATE table-specification [CHANGE] [IN] ALL [ROWS] change-specification-list [RETRIEVE INTO embedded-variable-specification-list]</li> </ul>	V	S
UPDATE Positioned	<ul style="list-style-type: none"> <li><b>Format</b> UPDATE table-specification [CHANGE] &lt;SET   COLUMNS&gt; change-specification-list WHERE CURRENT OF cursor-name [RETRIEVE INTO embedded-variable-specification-list]</li> </ul>	V	S
UPDATE Searched	<ul style="list-style-type: none"> <li><b>Format</b> UPDATE table-specification [CHANGE] &lt;SET   COLUMNS&gt; change-specification-list [WHERE boolean-expression] [RETRIEVE INTO embedded-variable-specification-list]</li> </ul>	V	S
UPDATE VALUES	<ul style="list-style-type: none"> <li><b>Format</b> UPDATE table-specification [CHANGE] VALUES (value-list) [WHERE boolean-expression] [RETRIEVE INTO embedded-variable-specification-list]</li> </ul>	V	S
USE DEFAULT	<ul style="list-style-type: none"> <li><b>Format</b> USE DEFAULT &lt;&lt;QUALIFIER   SCHEMA&gt; &lt;qualifier-name   placeholder&gt;   VERSION &lt;version-name   placeholder&gt;&gt;</li> </ul>	VO	N
WHENEVER	<ul style="list-style-type: none"> <li><b>Format</b> WHENEVER &lt;SQLERROR   NOT FOUND&gt; &lt;CONTINUE   &lt;GOTO   GO TO&gt; paragraph-name&gt;</li> </ul>	V	S

# Supported JCL Statements



This chapter contains detailed information on JCL statements support. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported JCL statements

Statement	Parser	Extraction
JOB	V	S
EXEC	V	S
DD	V	S
PROC	V	S

**5-2** Supported JCL Statements  
*Supported JCL statements*

PEND	VO	N
JCLLIB	VO	N
OUTPUT	VO	N
SET	V	S
IF	VO	N
ELSE	VO	N
ENDIF	VO	N

# Supported CICS Statements



This chapter contains detailed information on BMS and Exec CICS statements support. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported CICS command statements

Statement	Parser	IA	Repository	
			Entities	Relations
ADDRESS	V	V		
ABEND	V	V		
ALLOCATE	V	V		
ASKTIME	V	V		
ASSIGN	V	V		

**6-2** Supported CICS Statements  
*Supported CICS command statements*

CANCEL	V	V		
CONNECT PRO- CESS	V	V		
CONVERSE	V	V		
DELETE	V	V	defines program-name. literal (File) defines program-name.id (Decision)	Deletes from File Deletes File Decision
DELETEQ TD	V	V		
DELETEQ TS	V	V		
DUMP	V	V		
ENDBR	V	V		
FORMATTIME	V	V		
FREE	V	V		
FREEMAIN	V	V		
GETMAIN	V	V		
HANDLE ABEND	V	V		
HANDLE AID	V	V		
HANDLE CONDI- TION	V	V		
IGNORE CONDI- TION	V	V		
INQUIRE	V	V		
JOURNAL	V	V		
LINK	V	V	refers to literal (Program Entry Point) defines program-name.Links.id (Decision)	Links Links Decision
LOAD	V	V		
POP HANDLE	V	V		



PURGE MESSAGE	V	V		
PUSH HANDLE	V	V		
READ	V	V	defines program-name. literal (File) defines program-name.id (Decision)	Reads File Reads File Decision
READNEXT	V	V		
READPREV	V	V		
READQ TD	V	V		
READQ TS	V	V		
RECEIVE	V	V		
RECEIVE MAP	V	V	defines literal (Screen) defines program-name.Receives.id (Decision)	Program Receives Program ReceivesDecision
RELEASE	V	V		
RESETBR	V	V		
RETRIEVE	V	V		
RETURN	V	V	refers to literal (Transaction) defines program-name.Starts.id (Decision)	Starts Starts Decision
REWRITE	V	V	defines program-name. literal (File) defines program-name.id (Decision)	Updates File Updates File Decision
ROUTE	V	V		
SEND	V	V		
SEND MAP	V	V	defines literal (Screen) defines program-name.Sends.id (Decision)	Program Sends Program SendsDecision
SEND TEXT	V	V		
SET	V	V		
SIGNOFF	V	V		

**6-4** Supported CICS Statements  
*Supported CICS/BMS statements*

SIGNON	V	V		
SPOOLCLOSE	V	V		
SPOOLOPEN INPUT	V	V		
SPOOLOPEN OUT-PUT	V	V		
SPOOLREAD	V	V		
SPOOLWRITE	V	V		
START	V	V	refers to literal (Transaction) defines program-name.Starts.id (Decision)	Starts Starts Decision
STARTBR	V	V		
SYNCPOINT	V	V		
TRACE	V	V		
UNLOCK	V	V		
WAIT JOURNAL	V	V		
WRITE	V	V	defines program-name. literal (File) defines program-name.id (Decision)	Inserts into File Inserts File Decision
WRITEQ TD	V	V		
WRITEQ TS	V	V		
XCTL	V	V	refers to literal (Program Entry Point) defines program-name.Xctl.id (Decision)	Xctls XctlsDecision

***Supported CICS/BMS statements***

Statement	Parser	IA	Repository	
			Entities	Relations

DFHMSD	V	S		
DFHMDI	V	S		
DFHMDF	V	S		

**Note:** For each of these statements, some options are not supported.

**6-6** Supported CICS Statements  
*Supported CICS/BMS statements*

## Supported CA-IDMS DML Statements



This chapter contains detailed information on CA-IDMS Data Manipulation Language statements support. For a list of supported legacy versions, refer to the *Release Notes*.

### Supported CA-IDMS DML statements

Statement	Parser	IA	Relation
ABEND	V	S	
ACCEPT	V	S	
ACCEPT BIND ADDRESS	V	S	
ACCEPT DATABASE STATISTICS	V	S	
ACCEPT DB-KEY FROM CURRENCY	V	S	

**7-2** Supported CA-IDMS DML Statements  
*Supported CA-IDMS DML statements*

ACCEPT DB-KEY RELATIVE TO CURRENCY	V	S	
ACCEPT page-info-location	V	S	
ACCEPT PROCEDURE CONTROL LOCATION	V	S	
ACCEPT TRANSACTION STATISTICS	V	S	
ATTACH	V	S	
BIND MAP	V	S	
BIND PROCEDURE	V	S	
BIND RECORD	V	S	
BIND RUN-UNIT	V	S	
BIND TASK	V	S	
BIND TRANSACTION STATISTICS	V	S	
CHANGE PRIORITY	V	S	
CHECK TERMINAL	V	S	
COMMIT	V	S	
CONNECT	V	S	
DC RETURN	V	S	
DELETE QUEUE	V	S	Program Deletes NetDbRecord Program Deletes NetDbRecord Decision
DELETE SCRATCH	V	S	
DELETE TABLE	V	S	
DEQUEUE	V	S	
DISCONNECT	V	S	
END LINE TERMINAL SESSION	V	S	
END TRANSACTION STATISTICS	V	S	
ENDPAGE	V	S	

ENQUEUE	V	S	
ERASE	V	S	
ERASE (LRF)	V	S	
FIND/OBTAIN	V	S	Program Reads NetDbRecord Program Reads NetDbRecord Decision
FIND/OBTAIN CALC/DUPLICATE	V	S	
FIND/OBTAIN CURRENT	V	S	
FIND/OBTAIN DB-KEY	V	S	
FIND/OBTAIN OWNER	V	S	
FIND/OBTAIN WITHIN SET USING SORT-KEY	V	S	
FIND/OBTAIN WITHIN SET/AREA	V	S	
FINISH	V	S	
FREE STORAGE	V	S	
GET	V	S	Program Reads NetDbRecord Program Reads NetDbRecord Decision
GET QUEUE	V	S	
GET SCRATCH	V	S	
GET STORAGE	V	S	
GET TIME	V	S	
IF	V	S	
INQUIRE MAP	V	S	
KEEP CURRENT	V	S	
KEEP LONGTERM	V	S	
LOAD TABLE	V	S	
MAP IN	V	S	
MAP OUT	V	S	

**7-4** Supported CA-IDMS DML Statements  
*Supported CA-IDMS DML statements*

MAP OUTIN	V	S	
MODIFY	V	S	Program Updates NetDbRecord Program Updates NetDbRecord Decision
MODIFY (LRF)	V	S	
MODIFY MAP	V	S	
OBTAIN (LRF)	V	S	
POST	V	S	
PUT QUEUE	V	S	
PUT SCRATCH	V	S	
READ LINE FROM TERMINAL	V	S	
READ TERMINAL	V	S	
READY	V	S	
RETURN	V	S	
ROLLBACK	V	S	
SEND MESSAGE	V	S	
SET ABEND EXIT	V	S	
SET TIMER	V	S	
SNAP	V	S	
STARTPAGE	V	S	
STORE	V	S	Program Inserts NetDbRecord Program Inserts NetDbRecord Decision
STORE (LRF)	V	S	
TRANSFER CONTROL	V	S	
WAIT	V	S	
WRITE JOURNAL	V	S	
WRITE LINE TO TERMINAL	V	S	



WRITE LOG	V	S	
WRITE PRINTER	V	S	
WRITE TERMINAL	V	S	
WRITE THEN READ TERMINAL	V	S	

**7-6** Supported CA-IDMS DML Statements  
*Supported CA-IDMS DML statements*

# Supported IMS Statements



**T**his chapter contains detailed information on IMS (MFS and Exec DLI) statements support. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported IMS/Exec DLI statements

Statement	Parser	IA
GET (GU, GN, GNP)	V	S
TERM	V	S
SCHEDULE	V	S
ISRT	V	S
REPL	V	S
DLET	V	S

**8-2** Supported IMS Statements  
*Supported IMS/MFS statements*

## **Supported IMS/MFS statements**

<b>Statement</b>	<b>Parser</b>	<b>IA</b>
DEV	V	S
DFLD	V	S
DIV	V	S
DO	VO	N
DPAGE	V	S
FMT	V	S
FMTEND	V	S
LPAGE	VO	N
MFLD	VO	N
MSG	VO	N
MSGEND	VO	N
PASSWORD	VO	N
SEG	VO	N

# Supported UNISYS Statements



This chapter contains detailed information on UNISYS (DMS DDL, DPS and ECL) statements support. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported ECL statements

Statement	Parser	Extraction
@ADD	VO	N
@ASG	VO	S
@BRKPT	VO	N
@CAT	VO	N

**9-2** Supported UNISYS Statements  
*Supported ECL statements*

@CHG	VO	N
@CKPT	VO	N
@CLOSE	VO	N
@COPIN	VO	N
@COPOUT	VO	N
@COPIN	VO	N
@COPY	VO	N
@CYCLE	VO	N
@DELETE	VO	N
@ENABLE	VO	N
@ENDIF	VO	N
@EOF	VO	N
@ERS	VO	N
@FILE	VO	N
@FIN	VO	N
@FIND	VO	N
@FREE	VO	N
@HDG	VO	N
@JUMP	VO	N
@LOG	VO	N
@MARK	VO	N
@MODE	VO	N
@MOVE	VO	N
@MSG	VO	N
@PACK	VO	N

@PCH	VO	N
@PERFEV	VO	N
@PREP	VO	N
@PRT	VO	N
@QUAL	VO	N
@REWIND	VO	N
@RSTRT	VO	N
@RUN	V	S
@SETC	VO	N
@START	VO	N
@SYM	VO	N
@TEST	VO	N
@USE	V	S
@XQT	V	S
@process	VO	N

***Supported DPS (FLDP) statements***

Statement	Parser	IA
FORM	V	S
IMAGE	V	S
TEMPLATE	V	S
FIELD	V	S
BACKGROUND_ATTRIBUTES	VO	N

**9-4** Supported UNISYS Statements  
*Supported DMS (DDL) statements*

CHARACTER_ATTRIBUTES	VO	N
CHECKNUMBER_TEST	VO	N
CONVERSATION	VO	N
CURSOR	VO	N
EDIT_SYMBOL	VO	N
BACKGROUND_ATTRIBUTES	VO	N
FUNCTION_KEYS	VO	N
GENERAL_EDIT	VO	N
HIGHLIGHT	VO	N
INPUT_ATTRIBUTES	VO	N
OUTPUT_ATTRIBUTES	VO	N
PREDEFINED_VALUE	VO	N
POSITION_IMAGE	VO	N
REPEAT_IMAGE	VO	N
SECURITY	VO	N
STORAGE	VO	N
TAB	VO	N
TEST	VO	N

***Supported DMS (DDL) statements***

<b>Statement</b>	<b>Parser</b>	<b>IA</b>
ACCESS CONTROL	V	S
ALLOCATE	V	S



AREA CODE	V	S
AREA CONTROL	V	S
AREA LOOKS	V	S
AREA NAME	V	S
AREA SECTION	V	S
ASC/DESCENDING	V*	S
CALC	V	S
CHECK	V	S
CHECK IS	V	S
CODE	V	S
DATA DIVISION	V	S
DATA NAME SECTION	V	S
ENCODING/DECODING	V	S
IDENTIFICATION DIVISION	V	S
LEVEL NUMBER	V	S
LOAD	V	S
LOCALE	V	S
LOCATION MODE	V	S
LOCK FOR COPY	V	S
LOOKS INCLUDE	V	S
MEMBER	V*	S
MODE	V	S
OCCURS	V	S
ORDER	V	S
OWNER	V	S

**9-6** Supported UNISYS Statements  
*Supported DMS (DDL) statements*

PAGES	V	S
PICTURE	V	S
RECORD CODE	V	S
RECORD MODE	V	S
RECORD NAME	V	S
RECORD SECTION	V	S
RESERVE	V	S
RESULT	V*	S
SCHEMA NAME	V	S
SCHEMA QUALIFIER	V	S
SET CODE	V	S
SET NAME	V	S
SET OCCURENCE SELECTION	V*	S
SET SECTION	V	S
TABLE NAME	V*	S
TABLE SECTION	V	S
TRAINING	V	S
USAGE	V	S
WITHIN	V	S

**Note:** In this table, “\*” (asterisk) after “V” in the “Parser” column means that some particular options for the statement may be not supported by the parser.

## *Supported Complexity Metrics*



**T**his chapter contains language-specific details for the complexity metrics supported by the ATW legacy estimation tools. For more information on the legacy estimation tools, see *Analyzing Projects* in the ATW document set. For a list of supported legacy versions, refer to the *Release Notes*.

## Supported Cobol Complexity Metrics

Metric	Description
Executable Statements	All Procedure Division statements (not compiler-directing). CONTINUE NEXT SENTENCE is regarded as a statement.
Operators	All Executable statements, +, -, *, /, **, NOT, AND, OR, <, <=, >, >=, =, IS, (subscript), (reference:modification), FUNCTION.
Operands	All Procedure Division references to data items and literals.
Binary Decisions	IF, EVALUATE (# of WHEN except WHEN OTHER), PERFORM...TIMES, PERFORM...UNTIL, PERFORM...VARYING, PERFORM...VARYING...AFTER (# of AFTER phrases + 1), statements with ON/NOT ON, AT END/NOT AT END, INVALID/NOT INVALID (1 decision per statement), GO TO...DEPENDING ON (# of alternatives), SEARCH (# of WHEN, AT END). CDML: IF. IDMS: IF. Fujitsu: IF DB-EXCEPTION.
Logical Operators in Conditions	AND, OR in conditions of conditional statements. See also notes below.
Unique Operands in Conditions	Unique operands in conditions of conditional statements. See also notes below.
Nesting Level	Maximum nesting of conditional statements.
Include Statements	COPY, ++INCLUDE, -INC, EXEC SQL INCLUDE.
Data Elements	All declared data names.
Call Statements	CALL.
GoTo Statements	GO TO, GO TO...DEPENDING ON.

IO Statements	OPEN, CLOSE, READ, WRITE, REWRITE, DELETE, START, SORT, MERGE, RETURN, RELEASE, ACCEPT, DISPLAY, STOP literal. SQL: INSERT, FETCH, SELECT, UPDATE, DELETE, EXECUTE; CICS: CONVERSE, SEND, SEND MAP, SEND TEXT, RECEIVE, RECEIVE MAP, RECEIVE TEXT, READQ, WRITEQ, DELETEQ, READ, READNEXT, READPREV, WRITE, REWRITE, DELETE. CDML: OPEN, CLOSE, FETCH, GET, DELETE, MODIFY, STORE, INSERT, REMOVE. IDMS: ERASE, OBTAIN, GET, MODIFY, STORE. Fujitsu AIM/DB: ERASE, GET, MODIFY, STORE.
Computational Statements	ADD, SUBTRACT, DIVIDE, MULTIPLY, COMPUTE.
Loop Statements	PERFORM...TIMES, PERFORM...UNTIL, PERFORM...VARYING PERFORM...VARYING...AFTER (# of AFTER + 1).
Pointers	Data items with USAGE IS POINTER, PROCEDURE-POINTER.
Entry Points	PROCEDURE DIVISION, ENTRY.
Inner Procedures	Paragraphs (including nameless).
Inner Call Statements	PERFORM procedure-name.
Parameters	PROCEDURE DIVISION USING ... RETURNING ... parameters.
Conditional Statements	IF, EVALUATE, SEARCH, PERFORM...UNTIL, PERFORM...VARYING...UNTIL, statements with ON/NOT ON, AT END/NOT AT END, INVALID/NOT INVALID. CDML: IF. IDMS: IF. Fujitsu: IF DB-EXCEPTION.

**Notes:**

- Abbreviated conditions are expanded before calculations.
- DECLARATIVEs content and other exception-handling statements are counted once, as ordinary statements.
- EVALUATE

**10-4** Supported Complexity Metrics  
*Supported Cobol Complexity Metrics*

EVALUATE ... [ALSO ...]    condition  
  WHEN ... [ALSO ...]    binary decision, condition  
    statements...  
  WHEN ... [ALSO ...]    binary decision, condition  
  WHEN ... [ALSO ...]    binary decision, condition  
    statements...  
  WHEN OTHER  
    statements...  
END-EVALUATE

- **SEARCH**

SEARCH ...  
  AT END ...            binary decision  
  WHEN ...            binary decision, condition  
    statements...  
  WHEN ...AND...        binary decision, condition  
    statements...  
END-SEARCH

**Supported PLI Complexity Metrics**

<b>Metric</b>	<b>Description</b>
Executable Statements	All statements (not compiler-directing), except DECLARE, END, ENTRY, BEGIN, DO(block), PACKAGE, PROCEDURE.
Operators	All executable statements, +, -, *, /, **,   , ^, &,  , NOT, AND, OR, <, <=, >, >=, =, (subscript).
Operands	All references to data items and literals in a procedure.
Binary Decisions	IF, SELECT, DO.
Logical Operators in Conditions	AND, OR in conditions of conditional statements.
Unique Operands in Conditions	Unique operands in conditions of conditional statements.
Nesting Level	Maximum nesting of conditional statements.
Include Statements	%INCLUDE, EXEC SQL INCLUDE.
Data Elements	All declared data names.
Call Statements	CALL.
GoTo Statements	GO TO.
IO Statements	DISPLAY, PUT, GET, READ, REWRITE, WRITE, DELETE, OPEN, CLOSE, LOCATE, UNLOCK.
Computational Statements	Assignment.
Loop Statements	DO.
Pointers	Data items with type POINTER.
Entry Points	PROCEDURE (top level), ENTRY.
Inner Procedures	Procedures.
Inner Call Statements	Call inner procedures.
Parameters	PROCEDURE parameters.
Conditional Statements	IF, DO, SELECT.

## Supported Natural Complexity Metrics

Metric	Description
Executable Statements	All Natural statements, except DEFINE DATA and IGNORE.
Operators	All executable statements, +, -, *, /, **, NOT, AND, OR, OR EQUAL, <, <=, >, >=, =, !=, IS (format), THRU, BETWEEN, LIKE, IN, ESCAPE (in SQL statements), MODIFIED, IS NULL, (subscript), etc.
Operands	All references to data items and literals in program logic (session parameters are not operands).
Binary Decisions	ACCEPT, AT BREAK, AT END OF DATA, AT END OF PAGE, AT START OF DATA, AT TOP OF PAGE, BEFORE BREAK PROCESSING, IF, IF SELECTION, DECIDE FOR (number of WHEN except WHEN NONE), DECIDE ON (number of VALUES except NONE VALUES), FOR, ON ERROR, REJECT, REPEAT.
Logical Operators in Conditions	AND, OR in conditions of conditional statements.
Unique Operands in Conditions	Unique operands in conditions of conditional statements.
Nesting Level	Maximum nesting of conditional statements.
Include Statements	LOCAL USING, GLOBAL USING, PARAMETER USING, OBJECT USING, CONTEXT USING, INCLUDE.
Data Elements	All declared data names.
Call Statements	CALL, CALLNAT, CALL FILE, CALL LOOP, CALLDBPROC, FETCH, PERFORM (in case of external subroutine), PROCESS, PROCESS COMMAND, PROCESS GUI, RUN, SEND METHOD.
IO Statements	CLOSE WORK FILE, DELETE, DISPLAY, EJECT, FIND, GET, GET SAME, HISTOGRAM, INPUT, NEW PAGE, OBTAIN, OPEN CONVERSATION, OPEN DIALOG, PRINT, READ, READ WORK FILE, REINPUT, RETRY, SKIP, SORT, STORE, UPDATE, WRITE, WRITE TITLE, WRITE TRAILER, WRITE WORK FILE. SQL: DELETE, INSERT, SELECT, UPDATE.
Computational Statements	ADD, COMPRESS, SUBTRACT, DIVIDE, MULTIPLY, ASSIGN/COMPUTE, MOVE, RESET, SEPARATE, EXAMINE TRANSLATE.



Loop Statements	CALL LOOP, FIND (without options FIRST, NUMBER, UNIQUE), FOR, HISTOGRAM, READ, READ WORK FILE (without option ONCE), REPEAT, SORT; SQL: SELECT.
Entry Points	= 1
Inner Procedures	DEFINE SUBROUTINE.
Inner Call Statements	PERFORM (in case of inner subroutine).
Parameters	PARAMETER, PARAMETER USING.
Conditional Statements	ACCEPT, IF, IF SELECTION, DECIDE ON, DECIDE FOR, REJECT, REPEAT (with UNTIL or WHILE).

**10-8** Supported Complexity Metrics  
*Supported Natural Complexity Metrics*

---

# Bibliography

- *IBM Asset Transformation Workbench v1.1 Getting Started (SC31-6877-00)*
- *IBM Asset Transformation Workbench v1.1 Preparing Projects (SC31-6879-00)*
- *IBM Asset Transformation Workbench v1.1 Analyzing Projects (SC31-6880-00)*
- *IBM Asset Transformation Workbench v1.1 Analyzing Programs (SC31-6878-00)*
- *IBM Asset Transformation Workbench v1.1 Profiling Projects (SC31-6881-00)*
- *IBM Asset Transformation Workbench v1.1 Creating Components (SC31-6876-00)*
- *IBM Asset Transformation Workbench v1.1 Parser Reference (SC31-6882-00)*
- *IBM Asset Transformation Workbench v1.1 Architecture Reference (SC31-6898-00)*



---

# Notices

This information was developed for products and services offered in the U.S.A. IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation

Licensing

2-31 Roppongi 3-chome, Minato-ku

Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION •AS IS• WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you. This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you. Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
P.O. Box 12195, Dept. TL3B/B503/B313  
3039 Cornwallis Rd.  
Research Triangle Park, NC 27709-2195  
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only. This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

---

## Trademarks

The following terms are trademarks of the IBM Corporation or its subsidiaries in the United States or other countries or both:

*Table 1. Trademarks*

IBM	MVS
AS400	CICS
IMS	DB/2
Database 2	OS/390
S/390	z/OS

The following terms are trademarks of other companies:

Java and JavaScript are registered trademarks and Sun Solaris and Solaris are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product and service names may be trademarks or service marks of others.









Product Number: 5724-L54

SC31-6882-00



(1P) P/N:5724-L54

