ASG-Manager Products™
Tools Support: Integration with ADW/IEW

Version 2.5

Publication Number: MPR0200-25-IEW
Publication Date: December 2000

ASG Worldwide Headquarters Naples Florida USA | asg.com | info@asg.com
1333 Third Avenue South, Naples, Florida 34102 USA Tel: 239.435.2200 Fax: 239.263.3692 Toll Free: 800.932.5536 (USA only)

ASG Documentation/Product Enhancement Fax Form

Please FAX comments regarding ASG products and/or documentation to (239) 263-3692.

Company Name	Telephone Number	Site ID	Contact name

Product Name/Publication	Version #	Publication Date
Product:		
Publication:		
Tape VOLSER:		

Enhancement Request:

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)
- Verify whether you received an ASG Service Pack for this product. It may include information to help you resolve questions regarding installation of this ASG product. The Service Pack instructions are in a text file on the distribution media included with the Service Pack.

If You Receive a Voice Mail Message:

1. Follow the instructions to report a production-down or critical problem.
2. Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
3. Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578	239.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.222	49.89.45716.400	support.de@asg.com
Singapore	65.6332.2922	65.6337.7228	support.sg@asg.com
All other countries:	1.239.435.2200		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578		
Asia	65.6332.2922	**Japan/Telecom**	0041.800.9932.5536
Australia	0011.800.9932.5536	**Netherlands**	00.800.3354.3578
Denmark	00.800.9932.5536	**New Zealand**	00.800.9932.5536
France	00.800.3354.3578	**Singapore**	001.800.3354.3578
Germany	00.800.3354.3578	**South Korea**	001.800.9932.5536
Hong Kong	001.800.9932.5536	**Sweden/Telia**	009.800.9932.5536
Ireland	00.800.9932.5536	**Switzerland**	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536	**Thailand**	001.800.9932.5536
Japan/IDC	0061.800.9932.5536	**United Kingdom**	00.800.9932.5536
		All other countries	1.239.435.2200

ASG Web Site

Visit http://www.asg.com, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at http://www.asg.com/asp/emailproductsuggestions.asp.

If you do not have access to the web, FAX your suggestions to product management at (239) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Preface v

About this Publication v

Publication Conventions vi

1 Introduction 1

2 Installation 3

Introduction 3

Choosing an Environment 3

Allocating MP-AIDS 4

UDR Clauses 4

Alias Clauses 4

3 Importing and Exporting 7

Overview 7

Importing to Manager Products from ADW/IEW 8

Import Preparations 8

The ADW and IEW IMPORT Commands 8

Introduction to Commands 8

Populating the WBTA 9

Generating and Reporting Proposed Members 10

Overriding the Reconcile Defaults 12

The Reconciliation Report 12

Generating Proposed Member Definitions 14

Overriding the Preview Defaults 15

Populating the Repository 16

Overriding the Populate Defaults 17

Resetting the WBTA 18

ADW/IEW IMPORT Command Syntax 18

Notes On Special Processing Used Within the Import Facility 19

Tailoring the Import Facility . . . 25

Exporting from Manager Products to ADW/IEW . . . 33
The ADW and IEW EXPORT Commands . . . 33
Tailoring the Export Facility . . . 37
ADW/IEW Tokens . . . 43

4 Member Types . . . 47

Introduction . . . 47
IEW-Application Syntax . . . 51
IEW-Attribute-Type Syntax . . . 51
IEW-CRITICAL-ASSUMPTION Syntax . . . 53
IEW-CRITICAL-SUCCESS-FACTOR Syntax . . . 54
IEW-DATA-COLLECTION Syntax . . . 55
IEW-DATA-SCHEMA Syntax . . . 58
IEW-DATA-STRUCT-REP-BLOCK Syntax . . . 59
IEW-DATA-STRUCTURE-OR-BLOCK Syntax . . . 60
IEW-DATA-Type Syntax . . . 61
IEW-DATA-TYPE-SET Syntax . . . 63
IEW-DATAFLOW Syntax . . . 64
IEW-DATASTORE Syntax . . . 65
IEW-DATASTORE-ACCESS Syntax . . . 65
IEW-DB2-DATABASE Syntax . . . 66
IEW-DB2-INDEX-PARTITION Syntax . . . 67
IEW-DB2-STOGROUP Syntax . . . 67
IEW-DB2-SUBSYSTEM Syntax . . . 68
IEW-DB2-TABLE Syntax . . . 68
IEW-DB2-TABLESPACE-PARTITION Syntax . . . 70
IEW-DB2-VIEW Syntax . . . 70
IEW-ENTITY-TYPE Syntax . . . 71
IEW-EXTERNAL-AGENT Syntax . . . 73
IEW-FIELD Syntax . . . 74
IEW-FILE-DATABASE Syntax . . . 74
IEW-FILE-RECORD Syntax . . . 74
IEW-FOREIGN-KEY Syntax . . . 75
IEW-FUNCTION Syntax . . . 76
IEW-GLOBAL-DATA-record Syntax . . . 77
IEW-GLOBAL-DATA-STRUCTURE Syntax . . . 78
IEW-GLOBAL-DATA-TYPE Syntax . . . 79
IEW-GOAL Syntax . . . 81
IEW-INDEX Syntax . . . 82
IEW-INFORMATION-NEED Syntax . . . 85
IEW-INFORMATION-TYPE Syntax . . . 86
IEW-JUNCTION Syntax . . . 87
IEW-LIBRARY Syntax . . . 88
IEW-LOCAL-DATA-RECORD Syntax . . . 88

IEW-LOCAL-DATA-STRUCTURE Syntax 89
IEW-LOCAL-DATA-TYPE Syntax. 90
IEW-LOCATION Syntax 92
IEW-MECHANISM Syntax 93
IEW-MFS-SCREEN Syntax 96
IEW-MODELLING-SOURCE Syntax 98
IEW-MODULE Syntax. 99
IEW-MODULE-DATA-AREA Syntax 100
IEW-ORGANIZATIONAL-UNIT Syntax 101
IEW-PARAMETER Syntax 104
IEW-PROBLEM Syntax. 104
IEW-PROCESS Syntax 107
IEW-PROGRAM Syntax 109
IEW-PROJECT Syntax. 112
IEW-PSB Syntax. 114
IEW-RELATION Syntax 115
IEW-RELATIONAL-DATABASE Syntax 117
IEW-RELATIONSHIP-TYPE Syntax 117
IEW-REPORT Syntax 118
IEW-SCREEN Syntax 119
IEW-SCREEN-OBJECT Syntax 121
IEW-SEQUENTIAL-PROCESS Syntax 122
IEW-SUBJECT-AREA Syntax. 123
IEW-SUBSET-INFO-TYPE Syntax. 124
IEW-SUBTYPE-SET Syntax 125
IEW-TBSPACE Syntax 126
IEW-UNIQUE-IDENTIFIER Syntax. 129
IEW-VALUE Syntax 129
IEW-VALUE-RESTRICTION Syntax. 129
IEW-VALUE-SET Syntax 130
IMS-DATABASE HDAM Syntax 131
IMS-DATABASE HIDAM Syntax 134
IMS-DATABASE HISAM Syntax. 137
IMS-DATABASE HSAM Syntax 138
IMS-DATABASE INDEX Syntax 140
IMS-DATABASE LOGICAL Syntax 141
PCB Syntax. 142
SEGMENT INDEX-POINTER Syntax 145
SEGMENT LOGICAL Syntax. 147
SEGMENT PHYSICAL Syntax. 147
The Udr1 and Udr2 Clauses Syntax 151
Common Clauses Syntax 152

Appendix A

ADW/IEW Associations Mapped Against Manager Products Repository Relationship Clauses 155

Appendix B
ADW/IEW Object-Type Properties Mapped Against Manager Products Repository Attributes 171

Appendix C
ADW/IEW Member Type Naming Prefixes 217

Index 221

Preface

This *ASG-Manager Products Tools Support: Integration with ADW/IEW* publication is one of a series describing the Manager Family of Program Products developed for organizations seeking to automate and manage their application development and maintenance effort. This publication describes how ASG-Manager Products (herein called Manager Products) enables integration with other applications. It provides support for ADW/IEW that is available in MVS and VM installations.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

This publication consists of these chapters:

- Chapter 1, "Introduction," introduces Manager Products and the benefits this support provides.
- Chapter 2, "Installation," provides installation information specific to Manager Products.
- Chapter 3, "Importing and Exporting," documents how to populate your repository with members generated from imported ADW or IEW objects.
- Chapter 4, "Member Types," documents the syntax of the Manager Products member types supporting ADW/IEW.

Publication Conventions

The following conventions apply to syntax diagrams that appear in this publication.

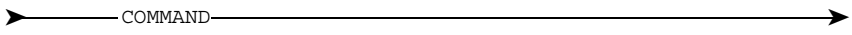
Diagrams are read from left to right along a continuous line (the main path). Keywords and variables appear on, above, or below the main path.

Convention	Represents
➤➤	At the beginning of a line indicates the start of a statement.
➤◄	At the end of a line indicates the end of a statement.
——→	At the end of a line indicates that the statement continues on the line below.
➤——	At the beginning of a line indicates that the statement continues from the line above.

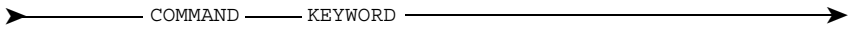
Keywords are in upper-case characters. Keywords and any required punctuation characters or symbols are highlighted. Permitted truncations are not indicated.

Variables are in lower-case characters.

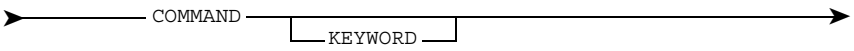
Statement identifiers appear on the main path of the diagram:



A required keyword appears on the main path:

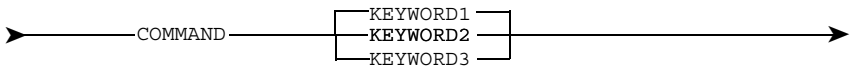


An optional keyword appears below the main path:

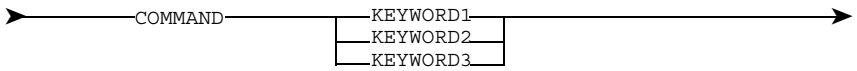


Where there is a choice of required keywords, the keywords appear in a vertical list; one of them is on the main path:

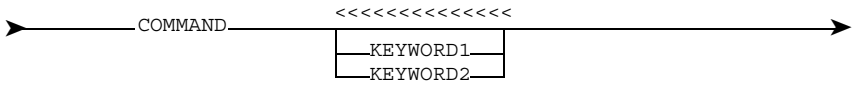
Convention	Represents



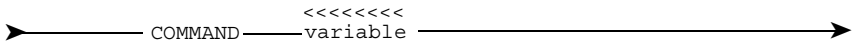
or



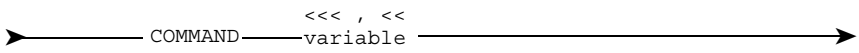
Where there is a choice of optional keywords, the keywords appear in a vertical list, below the main path:



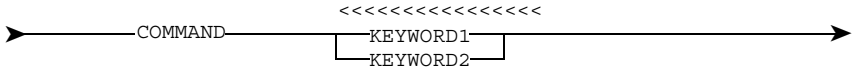
The repeat symbol, <<<<<<, above a keyword or variable, or above a whole clause, indicates that the keyword, variable, or clause may be specified more than once:



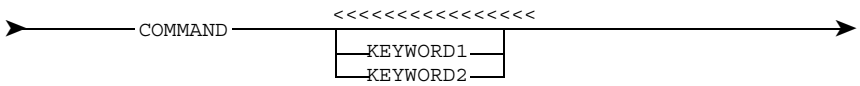
A repeat symbol broken by a comma indicates that if the keyword, variable, or clause is specified more than once, a comma must separate each instance of the keyword, variable, or clause:



The repeat symbol above a list of keywords (one of which appears on the main path) indicates that any one or more of the keywords may be specified; at least one must be specified:



Convention	Represents
The repeat symbol above a list of keywords (all of which are below the main path) indicates that any one or more of the keywords may be specified, but they are all optional:	



Allen Systems Group, Inc. uses these conventions in publications ASG uses these conventions in technical publications:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
lowercase italic monospace	Information that you provide according to your particular situation. For example, you would replace *filename* with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax.
	Also used for denoting brief examples in a paragraph.
Vertical Separator Bar (|) with underline	Options available with the default value underlined (e.g., Y|N).

1 Introduction

This product enables integration between Manager Products and Knowledge Ware's Application Development Workbench/Information Engineering Workbench (ADW/IEW), which enables the import and/or export of data between the two environments.

You can import logical/physical model definitions from the ADW/IEW Planning, Analysis, and Design workstations to your Manager Products repository, and reconcile these with existing definitions. Once the information has been imported, using the import facility, repository controllers can perform change/version control and impact analysis. The repository controllers can also locate and obtain the definitions of the data necessary for the generation of record layouts, storage definitions, or DBMS required DDL statements.

You can also export definitions from Manager Products to the ADW/IEW Planning, Analysis, Design, and Construction workstations, using the integration facilities of both product sets. This provides the maximum compatibility and most effective transfer and usage of information between the two environments. Information originally imported to the repository can be managed to ensure compliance with the corporate business model. The export facilities enable the validated information to be exported subsequently to any of the ADW/IEW workstations. Furthermore, you can populate ADW/IEW workstations with existing member definitions, which allows you to capitalize on past investment in the repository. (This feature is not readily available to ADW/IEW workstation users.)

All the functionality of ADW/IEW integration is available for both MethodManager and command interface users.

In the command interface, integration with ADW/IEW is implemented through the ADW IMPORT, ADW EXPORT, IEW IMPORT, and EXPORT commands. MethodManager integration is implemented through ToolSet SERVICES import and export panels.

2 Installation

Introduction

For general information on how Manager Products are installed, refer to your installation manual. This chapter documents the ADW/IEW specific information not included in that manual.

Choosing an Environment

You must decide how you want the software to operate. You have these two choices:

- Operate the software in an isolated environment. This may be necessary if you want integration between various ADW/IEW users and workstations by means of a common Manager Products repository and import/export interfaces. This is the recommended choice for your initial testing.
- Integrate the software into an existing Manager Products environment. This may be necessary if you want to use existing Manager Products repository definitions to populate ADW/IEW workstations through the export interface.

Depending on your requirements, you may also need to adjust the UDS-TABLE and the environment. This is especially important if you do not plan to use the ADW/IEW member types as supplied in the MDRIM (UDSTABLE DU016). You must load the supplied administration definitions into a dependent status. You should then freeze this status and perform any tailoring in another dependent status.

Note:

If integration is made as complete as possible, MMT-IEW-GOAL will be replaced by MMT-GOAL. If so, you have to update MPDYIITAB1 and any UDS-TABLE references to MMT-IEW-GOAL. You may want this level of integration if you want to use the supplied ManagerView diagram types for ADW/IEW repository definitions.

The process of integrating the supplied UDS-TABLE with an existing table is complex. Care must be taken when performing this task. If in doubt, please consult the ASG Service Desk for guidance.

Allocating MP-AIDS

Since the tailoring components of the software are supplied as EXECUTIVE-ROUTINE repository members, it is possible to have multiple configurations that are different. If such a requirement exists, then one option is to populate more than one MP-AID with EXECUTIVE members constructed from the source. Another option is to supply the EXECUTIVE source code for individual User Executives. In any case, ASG recommends that the source for all such executives is retained by the Systems Administrator for future reference.

UDR Clauses

Certain ADW/IEW associations with properties are mapped onto UDR clauses. These associations are given in Appendix A,"ADW/IEW Associations Mapped Against Manager Products Repository Relationship Clauses" on page 155. By default the UDR clauses are named IEW-ADW-XREF and IEW-ADW-XREF-DSTRUCT. The default names can be tailored using the iew_xref and iew_xref_dstruct variables in "MPDYIITAB0" on page 26.

The UDR clauses must be controlled through a CONTROL UDR command or RULE 130 of Method Manager's enable ToolSet SERVICES functions (panel A70000).

Refer to the *ASG-Manager Product Controller Manual* for details of the UDR clause and the CONTROL command.

Refer to of *ASG-MethodManager Administration* for details of RULE 130.

Alias Clauses

Although extra aliases are not a prerequisite for installation of the software, ASG recommends that you add at least one alias to ensure that the exact format of ADW/IEW names (including any embedded spaces or lower case characters) is maintained. This alias should correspond with the one specified in the tailoring variables iew_import_alias in "MPDYIITAB0" on page 26 and iew_export_alias in "MPDYIXTAB0" on page 37.

The alias table may be tailored (macro DALIAS module DMUO6) to support these alias types:

ALIAS FORMAL-NAME name
ALIAS GAMMA name
ALIAS IEW name
ALIAS IMS name
ALIAS LOCAL-NAME name
ALIAS SQL name

ALIAS FORMAL-NAME. This is used by the IEW-PARAMETER member type.

ALIAS GAMMA. This is used to hold a local name that may be utilized by the mainframe product GAMMA supplied by KnowledgeWare. Users who do not utilize the Design workstation or the GAMMA mainframe product may wish to omit the definition of this alias when defining the alias table.

ALIAS IEW. This is used to hold full, unchanged object instance entries for workstation object names. It is not mandatory for export, but import always generates an IEW ALIAS. Users of ADW may wish to use ALIAS ADW; tailoring iew_import_alias (see "MPDYIXTAB0" on page 37) and iew_export_alias type (see "MPDYIXTAB0" on page 37), accordingly.

ALIAS LOCAL-NAME. This is used for name, local name, and short name.

Several of these aliases may already exist (for example, IMS), and an existing alias may also have an equivalent meaning to one of these. If so, you should consider using the ALIAS SYNONYM capability.

After re-linking DMU06, the systems administrator needs to enter a CONTROL NEW-ALIASES command for the relevant repository.

Refer to the *ASG-Manager Products Installation Manual*for details of the ALIAS tailoring process.

Refer to the *ASG-Manager Products Controllers Manual* for details of the CONTROL NEW-ALIASES command.

3 Importing and Exporting

Overview

The ADW and IEW IMPORT commands are the primary commands for importing to Manager Products from ADW and IEW.

The ADW and IEW EXPORT commands are the primary commands for exporting to ADW and IEW from Manager Products.

The import and export facilities are tailorable through Executive Routines. These Executive Routines are supplied as Corporate Executives on the administration repository and MP-AID. It is also possible to execute them as User Executives, enabling the administrator to tailor the interface according to multiple user requirements with relative ease. Use of concatenated MP-AIDs can also achieve this objective, if required. Refer to the *ASG-Manager Products System Administrator Manual* for details of concatenated MP-AIDs. Refer to the *ASG-Manager Products Procedures Language* for details of Executive Routines.

These routines make extensive use of Manager Products variables in the form of arrays, that are used for tailoring the input and output functions. As a result, depending largely on the volume of data being imported/exported, you may need to allocate a larger region to run Manager Products than normal. You may therefore wish to invoke the export and import facilities in batch.

Note:

Throughout this text the colon character (:) is used as a literal delimiter in variable assignment statements.

Importing to Manager Products from ADW/IEW

Import Preparations

In order to import to Manager Products, you must first make resident on the mainframe the ADW/IEW Export files produced by the IEW consolidate and export process (also known as encyclopedia data transfer to formatted text files for ADW). This process should result in these four files:

- The object instances file (OI.EXP)
- The associations file (AI.EXP)
- The short properties file (PI.EXP)
- The long properties file (TI.EXP)

You can achieve this by using SEND software.

Note:

TI.EXP must be uploaded with LRECL = 98. Some file-transfer packages default to a record length of 80).

The ddnames to be associated with these files can be allocated/tailored in table MPDYIITAB0 (see "MPDYIITAB0" on page 26). In all instances you must specify ASCII to EBCDIC translation in the relevant file transfer command.

The ADW and IEW IMPORT Commands

The ADW or IEW IMPORT commands populate your repository with members generated from imported ADW or IEW objects.

Refer to "The ADW and IEW IMPORT Commands" on page 8 for the syntax of the ADW and IEW IMPORT commands.

Introduction to Commands

Use the ADW and IEW IMPORT commands to:

- *Extract* information from the ADW or IEW export files resident on your mainframe and to populate global variables on the Workbench Translation Area (WBTA).
- *Reconcile* them with any existing members that share the same name by generating proposed members from the extracted information and producing a report that enables this reconciliation.
- *Preview* them prior to their entry in the repository by generating member definitions for the proposed members.
- *Populate* the repository with the generated member definitions.

Each of the stages in the import process can be taken individually or combined with one another but must follow the sequence specified in "Introduction to Commands" on page 8. For example, you could enter:

```
ADW IMPORT EXTRACT;
ADW IMPORT RECONCILE;
ADW lMPORT PREVIEW;
ADW IMPORT POPULATE;
```

Or

```
ADW IMPORT EXTRACT RECONCILE PREVIEW;
ADW IMPORT POPULATE;
```

Or

```
ADW IMPORT EXTRACT RECONCILE PREVIEW POPULATE;
```

You can abandon the import process after any of the stages prior to the populate stage. The information on the WBTA is automatically replaced by the next ADW or IEW IMPORT EXTRACT command. You can also clear the WBTA at any stage by entering an ADW or IEW IMPORT RESET command.

You can repeat any of the stages in the import process. You can also return to an earlier stage. For example, you could return to *reconcile* after *preview* and then repeatedly reconcile the proposed members until you are satisfied that you want to generate member definitions from them.

You can execute any command during the import process that does not impact the global variables on the WBTA. For example, you may want to use the WHICH command to interrogate the existing members listed on the reconciliation report before continuing the import process.

The systems administrator can tailor the ADW and IEW IMPORT commands by altering the ASG-supplied defaults specified in Corporate Executive Routine MPDYIITAB0. You can also override certain of the defaults active in your environment by specifying particular keywords in an ADW or IEW IMPORT command. Refer to "MPDYIITAB0" on page 26 for details on how the systems administrator can tailor the ASG-supplied defaults.

Populating the WBTA

Use the ADW and IEW IMPORT EXTRACT commands to extract information from the ADW or IEW export files resident on your mainframe. These commands are allso used to populate Procedures Language global variables on the WBTA.

To extract information onto the WBTA, enter:

```
tool IMPORT EXTRACT;
```

where *tool* is ADW or IEW depending on whether the imported data has been extracted from an ADW or IEW export file.

The ASG-supplied default is that the export files have these file names:

ASSOC (THE AIEXP associations file)
OBJECT (the OI.EXP object instances file)
PROP (the TI.EXP long properties file)
TEXT (the PI.EXP short properties file).

The ADW and IEW IMPORT EXTRACT commands output a message specifying the number of objects for which information has been extracted. Any previously extracted information held on the WBTA is automatically replaced.

You can subsequently generate proposed members from the extracted information using the ADW or IEW IMPORT RECONCILE command.

Generating and Reporting Proposed Members

The ADW and IEW IMPORT RECONCILE commands generate:

- Proposed members from the information placed on the WBTA by the preceding ADW or IEW IMPORT EXTRACT command
- A reconciliation report that reports both the proposed members and any existing members in the current or visible statuses that have the same name

To generate proposed members and a reconciliation report, enter:

```
tool IMPORT RECONCILE;
```

where *tool* is ADW or IEW.

Depending on whether ASG-supplied or user-defined defaults are in place in your environment, the ADW and IEW IMPORT EXTRACT command will determine whether proposed members:

- Are subsequently entered into the repository with an ADD or a REPLACE command. The ASG-supplied default is to ADD proposed members.
- Include in their definition these common clauses of the existing members they are to replace:
  - ADMIMSTRATIVE-DATA
  - CATALOG
  - COMMENT
  - DESCRIPTION
  - NOTE

  The ASG-supplied default is to exclude the common clauses of existing members.You can override the defaults by specifying these keywords in the command:
  - ADDING or REPLACING
  - COMMON-CLAUSES or NO-COMMON-CLAUSES

You can also use these commands in combination with the reconciliation report:

RADD. This command specifies that a proposed member is entered in the repository by an ADD command.

RREP. This command specifies that a proposed member will be entered in the repository by a REPLACE command.

RIGN. This command specifies that a proposed member will not be entered in the repository.

RREN. This command renames a proposed member.

RUPD. This command updates an existing member with the same name as a proposed member.

For example, if you do not want a proposed member to replace an existing member use the RREN command to rename the proposed member or the RIGN command to specify that it is not to be entered in the repository.

You can subsequently generate command and member definition statements for the proposed members using the ADW or IEW IMPORT PREVIEW command.

Proposed IEW-DATAFLOW members are only reconciled with existing repository members if their names are prefixed by IF1- (or a user-defined alternative).

Objects that are not supported by Manager Products are listed on the reconciliation report, but member definitions are not generated for them. The unsupported objects are indicated by comments in the output of the subsequent ADW or IEW IMPORT PREVIEW command.

Refer to the *ASG-Manager Products Dictionary/Repository User's Guide* for details of the RADD, RREN, RREP, RIGN, and RUPD commands.

Refer to the *ASG-Manager Products Dictionary/Repository User's Guide* for details of the ADD and REPLACE commands.

Overriding the Reconcile Defaults

These commands enable you to override certain defaults active in your Manager Products environment:

To specify that you want the proposed members entered in the repository by REPLACE commands, enter:

```
tool IMPORT RECONCILE REPLACING;
```

where *tool* is ADW or IEW.

To specify that you want the proposed members entered in the repository by ADD commands, enter:

```
tool IMPORT RECONCILE ADDING;
```

To include the common clauses of existing members in the definition of the proposed members replacing them, enter:

```
tool IMPORT RECONCILE COMMON-CLAUSES;
```

To exclude the common clauses of existing members, enter:

```
tool IMPORT RECONCILE NO-COMMON-CLAUSES;
```

The Reconciliation Report

The list of the proposed members documenting the external objects that you have imported information about is the reconciliation report.

The report is divided into different sections for each external object. Each proposed member has a unique identification number.

The entries that follow *Extracted* give the name and object type code of the external object. Each object type has a unique object type code that corresponds with the proposed member type.

The entries that follow *Proposed* give the member name and member type of the proposed member documenting the external object. (You can abbreviate information if necessary.) Indicate if the member is entered in the repository by using an ADD or REPLACE command.

The entries after *Dictionary* give information about the condition and member type of any repository member that has the same name as the proposed member. (The condition may be encoded, dummy, or unverified.) *NO AUTH displays if you do not have the authority to access the existing member.

The entries after *refers to* gives the member name and member type of the members to which the existing member refers.

Figure 1 • Example of Reconciliation Report:

```
************************************************************************
                    Reconciliation Detailed Report
            for extract of IEW Model Data from EXTERNAL FILE.

************************************************************************
1  ExtractedProduct10007
   ProposedNT-PRODUCTIEW-ENTITY-TYPEADD

************************************************************************
2  ExtractedCustomer10007
   ProposedNT-CUST0MERIEW-ENTITY-TYPEADD
   Dictionary  SCE ENC IEW-ENTITY-TYPE
   refers to
          1  AT-CUSTOMER_NAME IEW-ATTRIBUTE-TYPE

************************************************************************
3  ExtractedOrder10003
   ProposedAT-ORDERIEW-ATTRIBUTE-TADD

************************************************************************
4  ExtractedProduct_Description10003
   ProposedAT-PRODUCT_DESCRIPTIONIEW-ATTRIBUTE-TADD

************************************************************************
5  ExtractedCustomer_Name10003
   ProposedAT-CUSTOMER_NAMEIEW-ATTRIBUTE-TADD
   Dictionary  SCE ENC IEW-ATTRIBUTE-TYPE

************************************************************************
6  ExtractedCredit_Rating10016
   ProposedGT-CREDIT_RATINGIFF-GLOBAL-DATA ADD

************************************************************************
7  ExtractedCustomer_Order10044
   ProposedRT-CUSTOMER_ORDERIEW-RELATIONSHI ADD

************************************************************************
```

Generating Proposed Member Definitions

Use the ADW and IEW IMPORT PREVIEW commands to generate ADD or REPLACE command and member definition statements for the proposed members generated by the preceding ADW or IEW IMPORT RECONCILE command.

To generate the statements, enter:

```
tool IMPORT PREVIEW;
```

where *tool* is ADW or IEW.

Depending on the ASG-supplied or user-defined defaults in place in your environment, the above command will:

- File the generated statements in either a new or an existing public USER-MEMBER. If the USER-MEMBER already exists, then the statements replace its contents or are appended to it. The ASG-supplied default is for the statements to replace the contents of a USER-MEMBER named IEWRUN. The filed statements can be edited, retained and referenced using all the facilities for handling MP-AID members.
- Either print or not print the statements. The ASG-supplied default is not to print statements.

You can override the above defaults by specifying this information in the command:

- An alternative USER-MEMBER name
- APPEND, NEW, or REPLACE
- NOPRINT, NO-PRINT, or PRINT

Definitions are generated for members in the same order they are listed on the reconciliation report.

Objects that are not supported by Manager Products cannot have a member definition generated for them. These objects are indicated by comments. The comments help you to relate the output with any previous reconciliation report. Generated IEW-TBSPACE and IEW-INDEX member definitions are also indicated by comments.

You can subsequently enter the statements in the repository using the ADW or IEW IMPORT POPULATE command.

Overriding the Preview Defaults

These commands enable you to override certain of the defaults active in the Manager Products environment:

To file generated output in a USER-MEMBER, enter:

```
tool IMPORT PREVIEW member-name option;
```

where:

tool is ADW or IEW.

member-name is the name of the USER-MEMBER.

option is either:

- NEW to file the output in a new member
- APPEND to append the output to the contents of an existing member
- REPLACE (the default) to replace the contents of an existing member

If you specify NEW, and the member already exists, the output is not generated. If you specify APPEND or REPLACE, and the member does not already exist, a new member is created.

To print or not print the generated statements, enter:

```
option IMPORT PREVIEW member-name PRINT;
```

or

```
option IMPORT PREVIEW member-name NOPRINT;
```

Specifying the keyword NO-PRINT has the same effect as NOPRINT.

Populating the Repository

Use the ADW or IEW IMPORT POPULATE commands to populate the repository with the definitions generated by the ADW or IEW IMPORT PREVIEW command.

To populate the repository, enter:

```
tool IMPORT POPULATE;
```

where *tool* is ADW or IEW.

Depending on which defaults are in place in your environment (ASG-supplied or user-defined), the ADW or IEW IMPORT PREVIEW command will:

- Only execute statements filed in a public USER-MEMBER or alternatively displayed in the current buffer. The ASG-supplied default is for the ADW or IEW IMPORT PREVIEW command to file statements in a USER-MEMBER named IEWRUN and for the ADW or IEW IMPORT POPULATE command to execute them.
- Print or not print the output generated by the executed statements. The ASG-supplied default is to print output.
- Execute all the statements separately or as one Logical Unit of Work (LUW). The ASG-supplied default is for all the statements to form a LUW that will either update the repository or be rolled back in its entirety, leaving the repository unchanged. This occurs if, for any reason, any of the statements are unsuccessful.

You can override the above defaults by specifying these keywords in the command:

- BUFFER or USER
- ROLLBACK
- PRINT or either NO-PRINT or NOPRINT

You can execute each of the statements filed in a USER-MEMBER by entering the member's name in the command area or by specifying it in a batch job. Do this, if by default, all statements in your environment are executed as an LUW, but space limitations on the recovery data set are restrictive in a particular instance. Generally, each imported object requires 6K of space on the recovery dataset. The actual amount of space required by an individual object varies according to its size.

Overriding the Populate Defaults

These commands enable you to override certain defaults active in your Manager Products environment:

To execute statements filed in a specified USER-MEMBER, enter:

```
tool IMPORT POPULATE USER user-member;
```

where:

tool is ADW or IEW.

user-member is the name of a public USER-MEMBER.

To execute statements displayed in the current buffer, enter:

```
tool IMPORT POPULATE BUFFER;
```

To print or not print any output generated in response to the executed statements, enter:

```
tool IMPORT POPULATE PRINT;
```

Or

```
tool IMPORT POPULATE NOPRINT;
```

The NO-PRINT keyword can be specified instead of NOPRINT and has the same effect.

To specify that all the executed statements will form one LUW, enter:

```
tool IMPORT POPULATE ROLLBACK;
```

Resetting the WBTA

If you have completed or abandoned the import process, reset the WBTA by clearing it of any extracted information. Resetting the WBTA increases the amount of virtual storage available in your environment.

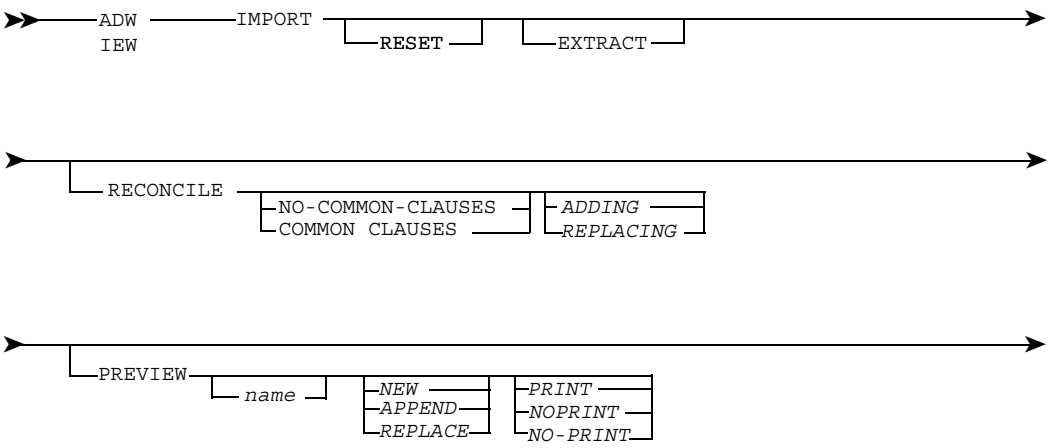
To reset the WBTA, enter:

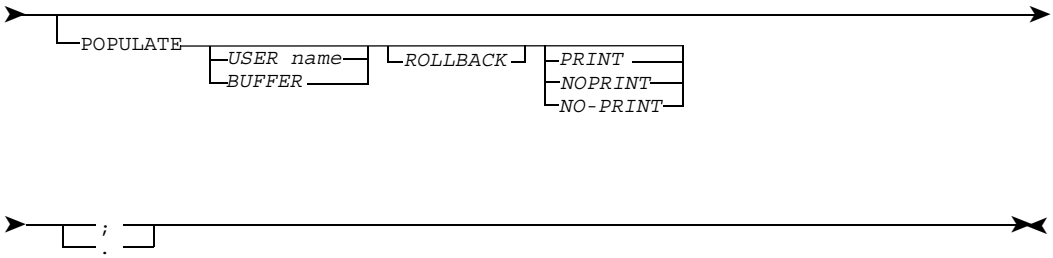
```
tool IMPORT RESET;
```

where *tool* is ADW or IEW.

If you do not reset the WBTA, the information held there is replaced by a subsequent ADW or IEW IMPORT EXTRACT command.

ADW/IEW IMPORT Command Syntax





where *name* is the name of a public USER-MEMBER.

Notes On Special Processing Used Within the Import Facility

In some cases special processing is employed within the import process to handle differences in the ways in which ADW/IEW and Manager Products represent certain data. This applies almost exclusively to IMS and dataflow definitions used within the Design Workstation. Other instances where special processing is employed are when error conditions arise within the export files. For all known circumstances, import routines have additional code added to handle the error and the error report, if applicable. This was implemented in an effort to ensure that data generated for population to the repository is as complete and error free as possible. The extra processing is required for these routines to generate complete and error-free definitions.

Manager Products Repository Member Names

Generally, member names are generated from the entries in the object instances file (OI.EXP). After folding a given object to upper case, any full stop characters are translated to #. This is necessary because the full stop character is regarded as a terminator. Similarly, if blanks are found, they are translated to underscores. Assuming the member name has a three-character prefix, it is then shortened to 29 characters using the REDUCE function and a tailorable prefix is added to it. Full OI.EXP name entries are held unchanged in the member's IEW ALIAS clause, which allows you to regenerate the OI.EXP entry upon export.

Special cases are IEW-RELATIONSHIP-TYPEs, IEW-INDEXs, and IEWTBSPACEs. IEW-RELATIONSHIP-TYPE member names are generated as a concatenation of the names of the two IEW-ENTITY-TYPE members it links, this name is then as described above. If two IEW-ENTITYs are linked by more than one IEW-RELATIONSHIP-TYPE, the same basis for generating the name is used, but a suffix, beginning -2, appends to the member name and a message is issued. You may use the RREN command to change the proposed name. IEW-TBSPACE member names are similarly generated by concatenating IEW-RELATIONALDATABASE and IEW-RELATION member names. IEW-INDEX member names are derived from the index-name property for the relation (code 30137).

It is possible to find some objects in the OI.EXP file without any name entry. In such cases the objects are named from other sources, as shown here:

Objects with Names Derived from their Properties:	
ATTRIBUTE-TYPE	Name taken from NAME (property type 3001) in the PI file
DATA-STRUCTURE-REPETITION-BLOCK	Name taken from LOCAL-NAME (property type 300116) in the PI file
LOCAL-DATE-STRUCTURE	Name taken from LOCAL-NAME (property type 30116) in the PI file
SCREEN-OBJECT	Name taken from SHORT-NAME (property type 30151) in the PI file
DATA-STRUCTURE-OR-BLOCK	Name taken from SHORT-NAME (property type 30151) in the PI file
PARAMETER	Name taken from LOCAL-NAME (property type 30116) in the PI file

Objects with Names Derived by Other Methods:	
LOCAL-DATA-TYPE	Name derived from the name of the structure or attribute that contains the Local Data Type, prefixed with the declared prefix value.
FIELDS	In the case of IMS Fields, the property type field name in segment (number 30294) of the association <data area> has local description<data structure> (number 20076) from the segment containing the field, when used in conjunction with association Field is implemented by <data structure> in Segment (number 20149).

In cases where a suitable name cannot be derived by any of the above methods, the token from the OI.EXP file, prefixed by the declared value is used, as shown here:

- JUNCTION
- DB PCB
- TP PCB
- DATA-STORE-ACCESS

Duplicate Names

Using short, local, or formal names can cause potential name duplication, since ADW/IEW does not prevent two or more objects from having the same short name. If you encounter this problem during reconcile, ADW/IEW automatically appends a numeric suffix to the duplicate name to maintain uniqueness.

It is possible for some objects to appear twice in the OI.EXP file, which results in duplicate names and duplicate tokens appearing in the export files. Detecting duplicates here implies that the most recently encountered instance is the correct definition added to the repository and all other instances will be ignored.

Limitations in the Support for Screen and Module Associations

To map as closely as possible to the Manager Products data model, there are several cases where import is forced to disregard information. This is because of limitations imposed by the currently available Manager Products definition. An example is the *module calls* association. Other examples are the (*screen gets/puts module before/after)* associations. By inspecting MPDYIITAB3 you can identify the limitations and tailor the interface at your installation.

ADW/IEW recognizes three types of module calls, plus the association (*module includes module)*. All of these associations may have parameters, but since the *includes* case is mapped to a Manager Products CONTAINS clause, no parameters can be associated with it.

The (*screen gets/puts module before/after)* associations (codes 20124 to 20127) are all mapped to the USER-EXIT clause of the IEW-SCREEN member type. A subsequent export only generates an association code 20124 from the member type.

The only associations that have their properties generated are those associations with properties mapped to Manager Products clauses for which there is an associated INFORMATION or DATA sub-clause. In these cases the delimited string consists of one or more occurrences of:

```
(string-tag> <attrib-vatue> <delimiter>
```

where:

string-tag is the name of the attribute as defined in MPDYIITAB2.
attrib-value is the value extracted from the PI.EXP file.
delimiter is the data-delimiter defined in MPDYIITAB0.

Two special cases of such associations are code 20076:

```
<data area> "has Local description" <data structure>
```

and code 20080:

```
<data area> "references" <data area>
```

Both of these associations can have data added to their INFORMATION sub-clauses through the code 20077 association:

```
<data-area> "Local description has context" <data~structure>
```

and code 20081 association:

```
<system component> "contains" <Data Area Reference>
```

respectively.

Note:

The 20155 associations (screen contains screen object) can have a source token that is itself all association. This will be a 20076 association. In this case a `CONTEXT-SO screen-object-name` string tag will be added to the INFORMATION clause.

Local and Global Data Types

In order to facilitate source language generation, data type attributes are mapped directly onto IEW-GLOBAL-DATA-TYPE and IEW-LOCAL-DATA-TYPE member type clauses. The translation of attributes are:

Translation of Attributes		
30307	Data-Type Type	This will be translated using the iew-data-type entry in table MPDYIITAB0 to give the appropriate form description.
30105	Format	This is translated to upper-case, delimited with quotes and used as the PICTURE clause in ENTERED-AS.
30119	Internal Length	This is used along with Data-Type Type to give the HELD-AS clause.
30120	External Length	This is used along with Data-Type Type to give the REPORTED-AS clause.

IMS Databases

Association Code 20077:

```
<data-area> "local description has context" <data-structure>
```

This association effectively modifies a specific 20076 association. The string that is appended to any other attributes for the 20076 association is:

```
CONTEXT <data-structure> <relationship> <data-area>
```

where:

data-structure is the name that has been assigned to the data-structure in this association.

relationship is the entry iew_rel_name(78) in the import table MPDYIITAB3.

data-area is the name that has been assigned to the data-area in this association.

Association Code 20081:

```
<system-component> "contains" <data-area reference>
```

This association effectively modifies a specific 20080 association by appending this information to the INFORMATION sub-clause on the 20080 association:

```
DATABASE dbname
```

where *dbname* is the name assigned to the database referenced by this association.

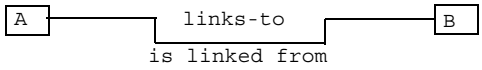
Association Code 20149:

```
Field is implemented by <data-structure> in Segment
```

Although this association is not mapped into Manager Products, it is used in conjunction with the properties of the associated 20076 association to name fields contained in segments.

Relationship Types (Association Code 20044)

In ADW/IEW, this association always has properties. This kind of information cannot currently be generated directly from a repository relationship clause, and so this particular association has been allocated a discrete member type in the Manager Products repository information model. Two entities linked by an association generate three members on the repository. Consider this diagram on ADW/IEW:



This process generates these member definitions in a repository:

A
IEW-ENTITY-TYPE
CONTAINS RT_A_B

RT_A_B
IEW-RELATIONSHIP-TYPE
TO-FROM-NAME "*links-to*"
FROM-TO-NAME "*is-linked-from*"
CONTAINS B

B
IEW-ENTITY-TYPE

Note:

The IEW-RELATIONSHIP-TYPE member name is generated from a concatenation of the two entities it is linking, and may be reduced as a result.

Dataflow Processing

A dataflow can have more than one set of SOURCE and DESTINATION object associations. This is as a result of dataflows used at different decomposition levels in dataflow diagrams.

An IEW-DATAFLOW member definition can only contain one SOURCE and one DESTINATION clause. A separate IEW-DATAFLOW member is therefore generated during import for each set of SOURCE and DESTINATION associations. The set of members defining the dataflow is generated with names prefixed by IF*n*-.

where *n* is an integer in the range 1 to 99.

For example, if four members are generated they will have names prefixed by IF1-, JF2-, JF3-, and 1F4-. Repository member names can be a maximum of 32 characters long. Imported dataflow names are therefore reduced to 27 characters, where necessary, in order to allow them to be prefixed.

Only members with names prefixed by IF1- are reconciled with the current contents of the repository during import.

The dataflow vector association (code 20040) also carries two dataflow diagram positional properties (codes 30070 and 30071). During import of the dataflow vector from/to junction, associations (codes 20023/20026) are required in order to retain these properties. Furthermore, these codes must be properly matched in order for any subsequent export to recreate the original set of associations and properties. For example, an indication of dataflow direction (FROM or TO) is required to indicate the type of position (INPUT or OUTPUT)). If these requirements are not met during import for a dataflow diagram, a warning message (DM06056W) is issued advising that the property has not been imported.

During export the clauses in all the members documenting the dataflow are used to generate a single dataflow that is named by the IEW-ALIAS of IF1-dataflow-name. For this reason, renaming a dataflow may cause unpredictable results during export. For the same reason, if you add IEW-DATAFLOW members to the repository by methods other than the import functions they must follow the import naming convention and must have relevant clauses (SOURCE and DESTINATION etc.) if they are exported.

The default naming prefix can be tailored using the variable iew_object_prefrx(9) in MPDYIITAB1, but will always have one or two digits added to it, if there are greater than 9 levels of decomposition (even if the prefix is set to null).

Relative Positions

Some of the relative-positions exported by ADW/IEW will not be imported and cannot be generated by export. Property codes falling into this category are 30060, 30061, 30065, and 30066.

Positional information is always meaningful in the context of a single instance of an export from ADW/IEW. This is not always the case where multiple sets of data transfer files are consolidated in a single repository. In cases where the members selected for export are derived from a separate encyclopedia, with conflicting or overlapping positional information, you must resolve such conflicts after export from the repository using the facilities of ADW/IEW.

Common clauses include only one relative position attribute. This means that only one positional property may currently be held. If an object appears in more than one diagram (for example, IEW-PROCESS in dataflow and/or decomposition diagrams) then the decomposition diagram position is not retained.

Tailoring the Import Facility

The import facility utilizes five look-up tables in order to determine a range of keyword and generation options. It should be noted that the look-up tables perform a major role in the control of the import facility and that the contents, even if not tailored, should be checked and verified.

MPDYIITAB0

This table contains general values pertaining to the import environment and these variables may be tailored:

Default Command Option Variables:	
iew_pre_cc_preserve	To include the common clauses of existing repository members in the definition of the members replacing them. To exclude existing common clauses specify N. To include existing common clauses specify Y. The default is N.
iew_pre_func	Generation option used when making additions to the repository. Values are ADDING or REPLACING. The default is ADDING.
iew_pre_print_option	To display preview output as well as filing it in a USER-MEMBER (values PRINT or NOPRINT). The default is NOPRINT.
iew_pre_user_name	Name of the public USER-MEMBER created by export. The default name is IEWRUN.
iew_pre_user_option	Disposition of the above USER-MEMBER (NEW, REPLACE or APPEND). The default is REPLACE.
iew_pop_from_option	To allow populate to act on either the current BUFFER or on a USER-MEMBER (NEW, REPLACE, or APPEND). The default is REPLACE.
iew_pop_print_option	NO-PRINT to suppress the primary populate output, else null. The default is null.
iew_pop_print_option	NO-PRINT to suppress the primary populate output, else null. The default is null.
iew_pop_rollback_option	Either ROLLBACK or null. In the event of an error, ROLLBACK will allow dynamic backout (treating the member definition statements as one LUW) whereas null will immediately cease processing at the point that the error is encountered. The default is ROLLBACK.

Extract ddnames:	
iew_ext_ddname(l)	The logical file name of the OI.EXP object instances file. The default is OBJECT.
iew_ext_ddname(2)	The logical file name of the AI.EXP associations file. The default is ASSOC.
iew_ext_ddname(3)	The logical file name of the TI.EXP long properties file. The default is TEXT.
iew_ext_ddname(4)	The logical file name of the PI.EXP short properties file. The default is PROP.

Preview Options	
iew_adw_currency	The currency character that is valid within ADW/IEW data type formats. The default is $.
iew_mpr_currency	The currency character that is allowed in the ENTERED-AS PICTURE clause. The default is £.
iew_ix_tab_expand	To expand index columns. The default is EXPAND. Set to null if you do not want to expand index columns.
iew_tab_tab_expand	To expand table columns. The default is EXPAND. Set to null if you do not want to expand table columns.
iew_ix_col_clause	The clause used for indexes. The default is CONTAINS.
iew_pre_delim	Character used to delimit proposed member names. The default is double quotes (").
iew_data_delim	Character used to delimit attributes appearing in INFORMATION or DATA sub-clauses. Care must be taken to ensure that this character should not appear as part of an attributes name or value and that (if applicable) it matches the value in variable iew_export_data_delim in table MPDYIITAB0 (see "MPDYIITAB0" on page 26). The default is \.
iew_import_alias	Alias type used by import. May be tailored as required but must specify a valid alias type. The default is IEW.

Preview Options	
iew_data_type (x)	Possible values of data-type type for an ADW/IEW Local or Global Data Type.
iew_dmr_form_desc (x)	IEW-LOCAL-DATA-TYPE and IEW-GLOBAL-DATA-TYPE member type form descriptions corresponding to iew_data_type entries. F as the first character indicates that a: HELD-AS form-description length clause is to be generated. U as the first character indicates that a HELD-AS CHARACTER length and a USAGE clause are to be generated.
iew_xref	Name of the general purpose UDR clause used to map ADW/IEW associations with properties. The default is IEW-ADW-XREF-DSTRUCT. Refer to "MPDYIXTAB0" on page 37.
iew_xref_dstruct	Name of the data structure UDR clause used to map ADW/IEW associations with properties. The default is IEW-ADW-XREF-DSTRUCT. Refer to "MPDYIXTAB0" on page 37.

MPDYIITAB1

This table provides a look-up to determine the member type proposed by the reconcile process, and allows the specification of a member name prefix (variables may be tailored):

Variable	Description
iew_object_type (x)	The member type encode keyword for the specific object identified by the index value of the array.
iew_object_prefix (x)	The prefix string inserted before the member name proposed by reconcile.

Note:

The array index values (cell numbers) specified for variables in this table are important and must not be changed. The index number corresponds to the ADW/IEW object-code represented by the repository member type.

For example, an ADW/IEW ENTITY object has the code 1007 in the OI.EXP file. To represent this object in the repository, ASG has created an IEW-ENTITY member type. The variable requiring the value IEW-ENTITY is iew_object_type and its index number is calculated as IEW object-code-9999 (i.e. 10007- 999=8). So the entry becomes:

```
iew_object_type (8) = IEW-ENTITY:
```

Likewise, its prefix, set in iew_object_prefix will also have an index value of 8.

Prefix Limitations

ASG recommends using prefixes during import.

If you use MethodManager, you need to ensure compatibility with NAMING clauses for existing RIM definitions, especially if integrating this facility into an existing MethodManager environment. As supplied, the table is compatible with the MDRIM as supplied by *ASG-Manager Products Tools Support: Integration with ADW/IEW*.

Even as a non-Manager Products user there will be problems if you attempt to export with null-prefix tailoring in place. One example is IEW-DATAFLOWs, which will still require a one to two digit numeric to identify decomposed dataflows (cross level).

The other example is IMS-DATABASEs and SEGMENTs. To integrate the ADW/IEW RIM with the MethodManager MDRIM, iew_object_prefix is supplied with a null value in two instances. These are IMS-DATABASE and SEGMENT, where the MDRIM distinguishes these types further. Refer to "MPDYIITAB4" on page 33. Inserting a value here will apply to *all* databases or segments, and the values in MPDYIITAB4 will have no effect.

MPDYIITAB2

This table provides a look-up to determine attribute names that are proposed by preview (variables may be tailored):

Variable	Description
iew_attrib_name	The attribute name generated by preview for this particular ADW/IEW property code including a Y or N character in column one signifying whether or not delimiters are required for the attribute value. Null indicates this is an attribute for a relationship, and so delimiters will automatically be generated and inserted into the INFORMATION or DATA sub-clause.
iew_make_alias	If an ADW/IEW property is mapped to an alias type, then iew_make_alias for the corresponding iew_attrib_name entry should be set to Y. In this case iew_attrib_name should be set to the alias type to be generated. For example, iew_attrib_name(l2) = :YLOCAL-NAME: iew_make_alias(l2) = :Y:

Note:

The array index value for variables in this table are important and must not be changed. The index number corresponds to the ADW/IEW property code represented by the member attribute value.

For example, the ADW/IEW property FROM-TO-NAME has the code 30034 in the PI.EXP file. To represent this property in the repository a FROM-TO- NAME attribute type was created. The variable requiring the value FROM-TO-NAME is iew_attrib_name and its index number is calculated as IEW property code 29999 (i.e., 30034 - 29999 = 35). So the entry becomes:

```
iew_attrib_name(35)= :YFROM-TO-NAME:
```

The Y in column 1 of the variable value indicates that a delimiter is required for this attribute. An N in this column will ensure that no delimiter is generated by preview. If this column is blank, it indicates that the data is associated with an INFORMATION or DATA sub-clause, and so delimiters are automatically assumed, and take the value specified in `iew_data_delim`.

MPDYIITAB3

This table provides a look-up to determine the relationship clauses generated by preview. These variables may be tailored:

Variable Names Relating Only to ADW/IEW Association Code 20044 (RELATIONSHIP-TYPE)	
iew_rel_mem(45)	The name of the member type proposed by reconcile created for the ADW/IEW association code 20044 (IEW-RELATIONSHIP-TYPE).
iew_rel_prefix(45)	The prefix generated by reconcile that is inserted in front of the member name.
iew_rel_name(45)	The name of the repository relationship used within IEW-RELATIONSHIP-TYPE to link two ENTITY-TYPES. The default is CONTAINS.

Variable Names Relating to the Generation of Qualification Data for Relationships (i.e., INFORMATION and DATA Sub-Clauses):	
iew_rel_data	This variable denotes whether or not a relationship carries information that is included in the DATA or INFORMATION sub clauses, which helps avoid unnecessary processing. Applicable values are Y or N signifying that the sub clauses are generated or omitted respectively.

Variable Names Relating to the Omission of Particular Relationships:	
iew_ignore_rel	Some relationships within ADW/IEW are superfluous and can be safely ignored without the loss of any semantic integrity. Applicable value is Y indicating that this relationship can be ignored.

Variable Names Relating to the Generation of Member Type Relationship Clauses:	
iew_rel_name	The name of the member type clause generated for this particular association code.
iew_db2_name	A particular member type clause used only in the case of DB2 objects.
iew_ims_rel	An indicator used to signify that this relationship relates to IMS. Used for performance reasons, this variable only has one value, Y.

Note:

There is a special entry in this table for the repository member type IEW-RELATIONSHIP-TYPE, generated from the ADW/IEW object code 20044. Generally, all ADW/IEW objects that generate a member definition have codes beginning with 1. Since it has significant data associated with it, IEW-RELATIONSHIP-TYPE is processed as an object, and for this reason is included within this table.

The array index value (specified for variables in this table) is important and must not be changed. The index number corresponds to the ADW/IEW association code represented by the member type relationship clause. For example, an ADW/IEW association IS SCOPE OF has the code *20006* in the AI.EXP file. To represent this relationship on the repository Manager Products uses the RELIES-ON relationship clause. The variable requiring the value RELIES-ON is iew_rel_name and its index number is calculated as IEW association-code -19999 (i.e., 20006 - 19999= 7). The entry becomes:

```
iew_rel _name(7) = :RELIES-ON:
```

MPDYIITAB4

This table gives a look-up reference for the second level of member type qualification of IMS (DL/1) database and segment member types. As these are not distinguished by ADW/IEW object type codes, a separate table from MPDYIITAB1 is needed. The specification of member name prefixes is also allowed here:

These Variables May be Tailored:	
iew_ims_mbr_type	The high level IMS member type.
iew_ims_type	The second level encode keyword for the IMS member type.
iew_ims_prefix	The prefix string inserted before the member name proposed by reconcile.

Exporting from Manager Products to ADW/IEW

The ADW and IEW EXPORT Commands

The ADW and IEW EXPORT commands populate ADW or IEW with objects generated from exported repository members.

Refer to "ADW EXPORT Command Syntax" on page 36 for the syntax of the ADW and IEW EXPORT commands.

Exporting Members

Use the ADW and IEW EXPORT commands to generate ADW or IEW data transfer file records from member definitions in the repository. The file records can be used to populate ADW and IEW with the objects that the members represent.

The exported members must be held in a KEPT-DATA list, which according to the ASG-supplied default is named EXPORT. You can catalog the members appropriately and then keep them with a KEEP IN EXPORT WHAT FORMS command. The systems administrator can define an alternative default name for the KEPT-DATA list but it must not be the same as any of the keywords available in the command.

To generate data transfer file records, enter:

```
tool EXPORT;
```

where *tool* is ADW or IEW, depending on whether members are being exported to ADW or IEW import files.

Depending on the ASG-supplied or user-defined defaults in place in your environment, the above command will perform these functions:

- Lock or not lock the exported members. The ASG-supplied default is not to lock the members. Locking a member prevents other users from updating it for a predefined period offtime. Non-MethodManager users can only lock members if they have the Workstation Interface facility installed (selectable unit CMR-WS0l). Specifying that exported members are locked will also RESERVE the entire repository for updates while the export is in progress.
- Export or not export the information held in the LAST-SAVE-DATE, CREATION-DATE, and LAST-UPDATE clauses of exported members. The ASG-supplied default is to export the clauses.
- Generate or not generate object names from the ALIAS clauses of the exported members or from the members' names. The ASG-supplied default is to use the IEW alias type.
- Print or not print the generated data transfer file records. The ASG-supplied default is to print them.
- Generate or not generate external files on your mainframe containing the data transfer file records. The ASG-supplied default is to create files with these logical file names:

  IEWXOI (tile OI.EXP object instance file)
  IEWXAI (the AI.EXP associations file)
  IEWXTI (the TI.EXP long properties file)
  IEWXPI (the PI.EXP short properties file)

- Display or not display DM5900I messages generated by the ADW or IEW EXPORT commands. The ASG-supplied default displays them.

You can override the above defaults by specifying these keywords in the command:

LOCK or NO-LOCK
PRINT or either NO-PRINT or NOPRINT
LSDATE or NO-LSDATE
CRDATE or NO-CRDATE
UPDATE or NO-UPDATE
FILE or NO-FILE
ALIAS or NO-ALIAS
MESSAGE or NO-MESSAGE

The systems administrator can tailor the ADW and IEW EXPORT commands by altering the ASG-supplied defaults specified in Corporate Executive Routine MPDYIXTAB0.

Refer to "Tailoring the Export Facility" on page 37 for details of tailoring export.

Refer to the *ASG-Manager Products Dictionary/Repository User's Guide* (MPR-DRUG) for details of the KEEP and LOCK commands.

Refer to the *ASG-Manager Products Procedures Language* for details of the RESERVE command.

Overriding the Export Defaults

These commands enable you to override certain of the defaults active in your environment:

To export members in a specified KEPT-DATA list, enter:

```
tool EXPORT list-name
```

where:

tool is ADW or IEW.

list-name is a KEPT-DATA list name and must not be the same as any of the keywords available in the command.

To lock or not lock the exported members, enter:

```
tool EXPORT LOCK;
```

Or

```
tool EXPORT NO-LOCK;
```

To print or not print the generated data transfer file records, enter:

```
tool EXPORT PRINT;
```

Or

```
tool EXPORT NO-PRINT;
```

To generate or not generate external files, enter:

```
tool EXPORT FILE;
```

Or

```
tool EXPORT NO-FILE;
```

To export or not export information in LAST-SAVE-DATE clauses, enter:

```
tool EXPORT LSDATE;
```

Or

```
tool EXPORT NO-LSDATE;
```

To export or not export information in CREATION-DATE clauses, enter:

```
tool EXPORT CRDATE;
```

Or

```
tool EXPORT NO-CRDATE;
```

To export or not export information in LAST-UPDATE clauses, enter:

```
tool EXPORT UPDATE;
```

Or

```
tool EXPORT NO-UPDATE;
```

To generate object names from ALIAS clauses or from member names, enter:

```
tool EXPORT ALIAS;
```

Or

```
tool EXPORT NO-ALIAS;
```

To display or not display DM59001 messages, enter:

```
tool EXPORT MESSAGE;
```

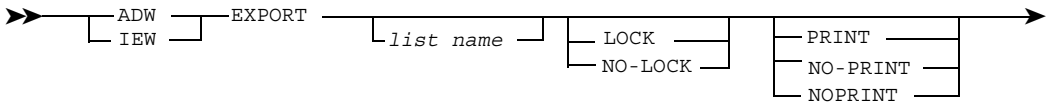
Or

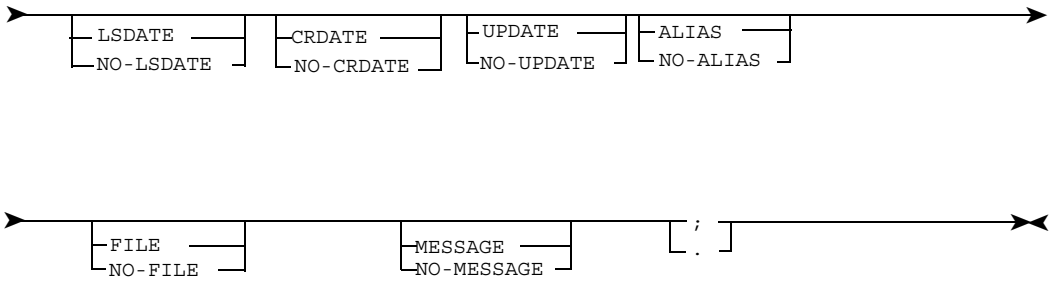
```
tool EXPORT NO-MESSAGE;
```

Note:

The NO-PRINT or NOPRINT keyword has no effect if you also specify the NO-FILE keyword in an ADW or IEW EXPORT command.

ADW EXPORT Command Syntax





where *list-name* is a KEPT-DATA list name and must not be the same as any of the keywords available in the command.

Tailoring the Export Facility

Like import, the export facility uses a number of look-up tables to determine default values and keywords that are generated at run time. Because of the complexity of the routines within the software, some of the table entries themselves appear complex. This particular table approach was taken in the interest of improving processing and performance of the software while ensuring tailorability. The simplest, most direct route was taken in building the table entries, but those entries for attributes (properties) and relationships (associations) may appear complex.

Whereas entries in the import tables had positional significances (the index value being derived from the ADW/IEW code), there is no such significance in the ordering of entries in the export table. Performance may be slightly improved by positioning the most frequently referenced entries at the start of the table, for tables other than MPDYIXTAB0.

MPDYIXTAB0

This table contains general values pertaining to the export environment. The numerics, 1, 2, 3, and 4, suffixing some of the variables relating to external datasets, refer to OI.EXP, AI.EXP, TI.EXP, and PI.EXP, respectively.These variables may be tailored:

Default Command Option Variables:	
iew_export_file_supp	If set to FILE, then external files containing generated record layouts are created on your mainframe. If set to NO-FILE, then the external files are not created. The default is FILE.
iew_export_kdl	The name of the KEPT-DATA list containing the members to be processed. The default name is EXPORT.

Default Command Option Variables:	
iew_export_lock_key	If set to LOCK then exported members are locked on the repository. If set to NO-LOCK then the members are not locked. The default is NO-LOCK.
iew_export_mess_supp	If set to MESSAGE then export messages display. If set to NO-MESSAGE then export messages do not display. The default is MESSAGE.
iew_export_ print_option	Whether or not primary output is sent to the primary print device. Permitted values are PRINT, NOPRINT, or NO-PRINT. The default is PRINT.
iew_export_crdate_supp	If set to CRDATE, then the CREATION-DATE clause is exported. If set to NO-CRDATE, then the clause is not exported. The default is CRDATE.
iew_export_lsdate_supp	If set to LSDATE, then the LAST-SAVE-DATE clause is exported. If set to NO-LSDATE, then the clause is not exported. The default is LSDATE.
iew_export_update supp	If set to UPDATE, then the LAST-UPDATE clause is exported. If set to NO-UPDATE, then the clause is not exported. The default is UPDATE.

General Variables:	
iew_export_string_del	The character used to delimit names and descriptive text when writing export files. The default is double quotes (").

The generation of ADW/IEW export object instance names can be from either a repository member's ALIAS clause or from the repository member's name. This is governed by the setting of the variables

iew_export_alias_supp	When set to NO-ALIAS, signifies that all object instance names are derived from repository member names. When set to ALIAS, all object instance names are generated from the alias type specified. The default is ALIAS.
iew_export_alias_type	The alias type from which object instance names are derived, if alias names are used. The default is IEW.

Variables Used for Output to a Sequential File:	

The names of the sequential file DD cards are specified through these fields:

iew_export_ddname(1)	The logical file name of the OI.EXP object instances file.
iew_export_ddname(2)	The logical file name of the AI.EXP associations.
iew_export_ddname(3)	The logical file name of the TI.EXP long properties file.
iew_export_ddname(4)	The logical file name of the PI.EXP short properties file.

The default logical file names (ddnames) used are:

File Names	
IEWXOI	(OI.EXP)
IEWXAI	(AI.EXP)
IEWXTI	(TI.EXP)
IEWXPI	(PI.EXP)

The file formats are specified through these variables:

Variables	
iew_export_file_format(1)	(OI.EXP)
iew_export_file_format(2)	(AI.EXP)
iew_export_file_format (3)	(TI.EXP)
iew_export_file_format(4)	(PI.EXP)

Permissible values are FIXED or VARIABLE. The default is FIXED.
The blocksizes are specified through these variables:

Variables	
iew_export_file_blksze(1)	(OI.EXP)
iew_export_file_blksze(2)	(AI.EXP)
iew_export_file_blksze(3)	(TI.EXP)
iew_export_file_blksze(4)	(PI.EXP)

The record-lengths are specified through the variables:

iew_export_file_lrecl(1)	(OI.EXP)
iew_export_file_lrecl(2)	(AI.EXP)
iew_export_file_lrecl(3)	(TI.EXP)
iew_export_file_lrecl(4)	(PI.EXP)

These are the record lengths as supplied, and the blocksizes default to the same values:

Record Lengths	
1 = 52	(OI.EXP)
2 = 41	(AI.EXP)
3 = 98	(TI.EXP)
4 = 58	(PI.EXP)

Note:
ASG recommends using the supplied default values. This ensures that the generated files need not be reformatted on downloading.

Other Variables	
iew_export_data_ delim	Character used to delimit attributes appearing in INFORMATION or DATA sub clauses. This character should not appear as part of an attributes name or value and it should match the value in variable iew_data_delim in table MPDYIITAB0. The default is \.
iew_export_data_type (x)	Possible values of data-type type for an ADW/IEW Local or Global Data Type.
iew_export_dmr_form-desc (x)	IEW-GLOBAL-DATA-TYPE and IEW-LOCAL- DATA-TYPE repository member form descriptions corresponding to iew_export_data_type entries.

Other Variables	
iew_export_drop_kdl	If set to 1, DUMMY and SOURCE members are not processed (i.e., they are dropped from the export KEPT-DATA list). If set to 0 the KEPT-DATA list will not be preprocessed. The default is 1.
iew_export_gdt_suffix	A suffix that will be used to generate a unique name for a GLOBAL-DATA-TYPE where one is generated for ADW/IEW. An example of such a situation is where a FILE directly contains an ITEM with no intermediate GROUP, which may occur where the data being exported did not originate in ADW/IEW. The default suffix is -GDT.
iew_export_sepa	The character that separates data in MPDYIXTAB*n* tables. The default is @. The character must not also be used as a string delimiter. Use the QUERY STRING- DELIMITER command to display string delimiters.
iew_export_xref	The name of the general purpose UDR clause holding properties of associations. The default name is IEW-ADW-XREF (see "MPDYIITAB0" on page 26).
iew_export_xref_dstruct	The name of the data structure UDR clause holding properties of associations. The default name is IEW-ADW-XREF-DSTRUCT (see "MPDYIITAB0" on page 26).
iew_tbspce_type	The member type generated for table spaces during import processing (ignored during export). The default is IEW-TBSPACE. See the iew_ tbspace_type variable in "MPDYIITAB0" on page 26.
iew_index-type	The member type generated for indexes during import processing (ignore during export). The default is IEW-INDEX.

ADW/IEW Tokens

Use Tokens as unique identifiers linking an object instance with its associations and properties. Tokens have no intrinsic meaning either within ADW/IEW or for integration of ADW/IEW. They are generated purely as reference labels for export and consolidation purposes and are not a concern for ADW/IEW users or for users of this integration facility.

You can randomly generate tokens as long as they are unique. You have the opportunity to declare a value where any generated tokens will begin. Subsequently generated tokens are incremented by 1.

The start values for tokens are tailorable through these variables:

Variable	Description
iew_ export_obj_token	For the OI.EXP token. The default is 0 (zero).
iew_export_ass_token	For the AI.EXP token. The default is 0 (zero).

Take care in the allocation of these values, considering the limitation of a maximum 10 digits in the Procedures Language arithmetic function. ASG recommends that you do not exceed a maximum of 7 digits as a start number.

MPDYIXTAB1

This table contains the values for all attributes in the UDS-TABLE and specifies a corresponding ADW/IEW object code linking the attribute with the member type. An example entry is:

```
iew_xtab_004(2)= :30004CENTRAL-TRANSFORM1000010004:
```

where:

30004 is the property code for the ADW/IEW property.

CENTRAL-TRANSFORM is the repository attribute representing the ADW/IEW property.

10000 is the object code for the ADW/IEW Object to which this attribute applies.

10004 is the object code for a further ADW/IEW object to which this attribute also applies.

Therefore, if the UDS-TABLE is changed and CENTRAL-TRANSFORM is altered, only the text part of the variable value requires changing. Alteration of any other part may render the export mechanism inoperable.

Note:

The repository attribute should not contain a suffix of more than four numeric characters (or it will be confused with an object code).

MPDYIXTAB2

This table contains the member type keyword for all ADW/IEW members represented in the Manager Products UDS-TABLE. It ties the member type (represented by the STANDARD-LITERAL clause) to its corresponding ADW/IEW object code. For example, an ADW/IEW object code of 10007 (ENTITY) is represented in the repository as an IEW- ENTITY-TYPE. The STANDARD-LITERAL specified for an IEW-ENTITY-TYPE is IEW-ENTITY-TYPE. Therefore, an appropriate entry in the table in this example is:

```
iew_xtab_002(x) = :10007@IEW-ENTITY-TYPE@:
```

Delimiters are used when there is a possibility that the string IEW-ENTITY-TYPE may also form part of another string such as IEW-ENTITY-TYPE-EVENT-MATRIX-NODE. Therefore @IEW-ENTITY-TYPE@ and @IEW-ENTITY-TYPE-EVENT-MATRIX-NODE@ cannot be confused when an exact match is requested by the software.

Note:

The @ character or any other chosen character may only be used if it is not currently employed as a string delimiter. You can check this possibility by using the QUERY STRING-DELIMITER command.

MPDYIXTAB3

This table contains the values for most relationship clauses (not IMS) in the UDS table and specifies a corresponding IEW association code. Refer to Appendix A, "ADW/IEW Associations Mapped Against Manager Products Repository Relationship Clauses" on page 155. Each repository member type has a list of its applicable clauses and corresponding codes. For example, the repository member type IEW-PROCESS may have the Manager Products relationship HAS to an IEW-FUNCTION (object code 10058). The corresponding ADW/IEW association for HAS in the context of an IEW-PROCESS is CONSISTS OF (for which the ADW/IEW code is 20034). Therefore in the IEW-PROCESS/IEW-FUNCTION example, an appropriate entry in the table would be:

```
iew_xtab_003(x) = :Y2003410058HAS@IEW-PROCESS:
```

where:

Y signifies that this association can have properties (held in the DATA or INFORMATION sub-clauses). An N in this column signifies that the association cannot have any properties.

20034 is the association code representing the ADW/IEW association.

10058 is the object code representing the ADW/IEW object from which this association is valid.

HAS is the repository relationship representing the ADW/IEW association specified in the code given above.

IEW-FUNCTION is the repository member type that this association can refer to. Users may wish to tailor the keyword IEW-PROCESS should they change the supplied UDS structure. It is unlikely (and inadvisable) that they should need to change any other part of the value.

The subscript value (*x*) has no significance, although it should be unique and it is useful to group one association code in a range of values for ease of reference.

Multiple table entries are required when more than one member type can be the subject of the relationship. For example, IEW-FUNCTION can refer to another IEW-FUNCTION, in this case another table entry would be required as:

```
iew_xtab_003(x)= :Y2003410058HAS@IEW-FUNCTION:
```

@ symbols are used in the circumstances and for the reasons described with regard to MPDYJXTAB2 (see "MPDYIITAB2" on page 30).

4 Member Types

Introduction

This chapter gives the syntax of the Manager Products member types supporting ADW/IEW. The syntax in this document is not exhaustive (for complete documentation refer to the documentation of the BASED-ON member type). All clauses are illustrated that may be present as a result of import and that may be referenced during export. In some cases export may reference clauses not shown, since only clauses common to both import and export are illustrated.

All objects that are found in the ADW/IEW meta model relevant to the Planning, Analysis, and Design Workstations are supported (depending on their relevance and presence within the Export files).

Refer to Figure 2 on page 48 for a list of the member types supporting ADW/IEW Planning, Analysis, and Design Workstations.

Refer to Appendix A, "ADW/IEW Associations Mapped Against Manager Products Repository Relationship Clauses" on page 155, for details of member type clauses that document ADW/IEW associations.

Refer to Appendix B, "ADW/IEW Object-Type Properties Mapped Against Manager Products Repository Attributes" on page 171, for details of member types that document ADW/IEW object types and their properties.

Note:

Any changes to the UDS-TABLE require changes to the supplied tables for import/export (MPDYIITAB*n* and MPDYIXTAB*n*). While the ordering of the clauses in the syntax diagrams should correspond directly with imported data, the order of the clauses may differ. This will not affect the operation of import or export. MethodManager's assisted update facility provides a consistent interface with respect to the ordering of clauses, irrespective of import.

Figure 2 • Member Types Supporting Planning, Analysis, and Design.

Planning	Analysis	Design
		IEW-APPLICATION
IEW-ATTRIBUTE-TYPE	IEW-ATTRIBUTE-TYPE	
IEW-CRITICAL-ASSUMPTION		
IEW-CRITICAL-SUCCESS-FACTOR		
IEW-DATA-COLLECTION		
		IEW-DATA-SCHEMA
		IEW-DATA-STRUCT-REP-BLOCK
		IEW-DATA-STRUCTURE-OR-BLOCK
		IEW-DATA-TYPE
		IEW-DATA-TYPE-SET
	IEW-DATAFLOW	
	IEW-DATASTORE	
	IEW-DATASTORE-ACCESS	
		IEW-DB2-DATABASE
		IEW-DB2-INDEX-PARTITION
		IEW-DB2-STOGROUP
		IEW-DB2-SUBSYSTEM
		IEW-DB2-TABLE
		IEW-DB2-TABLESPACE-PARTITION
		IEW-DB2-VIEW
IEW-ENTITY-TYPE	IEW-ENTITY-TYPE	IEW-ENTITY-TYPE
	IEW-EXTERNAL-AGENT	
		IEW-FIELD
		IEW-FILE-DATABASE
		IEW-FILE-RECORD

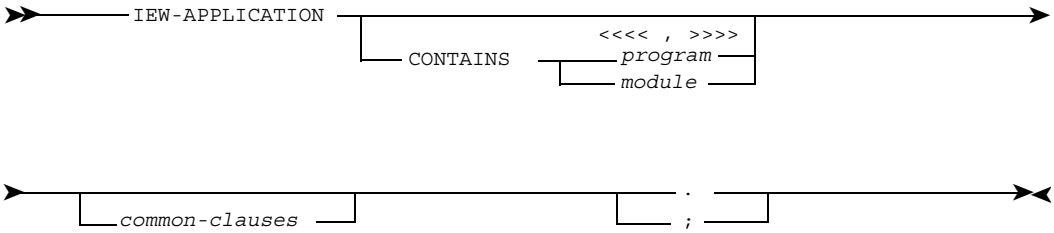
Planning	Analysis	Design
		IEW-FOREIGN-KEY
IEW-FUNCTION		
		IEW-GLOBAL-DATA-RECORD
		IEW-GLOBAL-DATA-STRUCTURE
	IEW-GLOBAL-DATA-TYPE	IEW-GLOBAL-DATA-TYPE
IEW-GOAL		
		IEW-INDEX
IEW-INFORMATION-NEED		
	IEW-INFORMATION-TYPE	
	IEW-JUNCTION	
		IEW-LIBRARY
		IEW-LOCAL-DATA-RECORD
		IEW-LOCAL-DATA-STRUCTURE
	IEW-LOCAL-DATA-TYPE	IEW-LOCAL-DATA-TYPE
IEW-LOCATION		
IEW-MECHANISM		
		IEW-MFS-SCREEN
IEW-MODELLING-SOURCE		
		IEW-MODULE
		IEW-MODULE-DATA-AREA
IEW-ORGANIZATIONAL-UNIT	IEW-ORGANIZATIONAL-UNIT	
		IEW-PARAMETER
IEW-PROBLEM		
IEW-PROCESS	IEW-PROCESS	
		IEW-PROGRAM
IEW-PROJECT		

Planning	Analysis	Design
		IEW-PSB
		IEW-RELATION
		IEW-RELATIONAL-DATABASE
IEW-RELATIONSHIP-TYPE	IEW-RELATIONSHIP-TYPE	
		IEW-REPORT
		IEW-SCREEN
		IEW-SCREEN-OBJECT
	IEW-SEQUENTIAL-PROCESS	IEW-SEQUENTIAL-PROCESS
IEW-SUBJECT-AREA	IEW-SUBJECT-AREA	
	IEW-SUBJECT-INFO-TYPE	
IEW-SUBTYPE-SET	IEW-SUBTYPE-SET	
		IEW-TBSPACE
		IEW-UNIQUE-IDENTIFIER
	IEW-VALUE	
	IEW-VALUE-RESTRICTION	
	IEW-VALUE-SET	
		IMS-DATABASE HDAM
		IMS-DATABASE HIDAM
		IMS-DATABASE HISAM
		IMS-DATABASE HSAM
		IMS-DATABASE INDEX
		IMS-DATABASE LOGICAL
		PCB
		SEGMENT INDEX-POINTER

Planning	Analysis	Design
		SEGMENT LOGICAL
		SEGMENT PHYSICAL

IEW-Application Syntax

The syntax of the member types supporting ADW/IEW are documented in this section in alphabetical order of member type name.



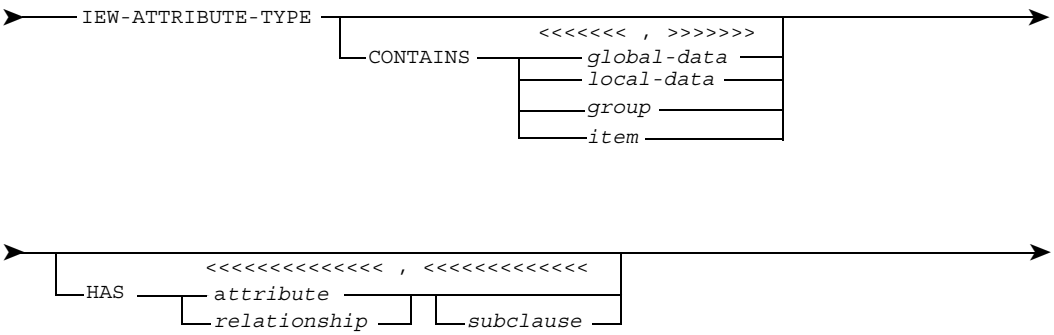
where:

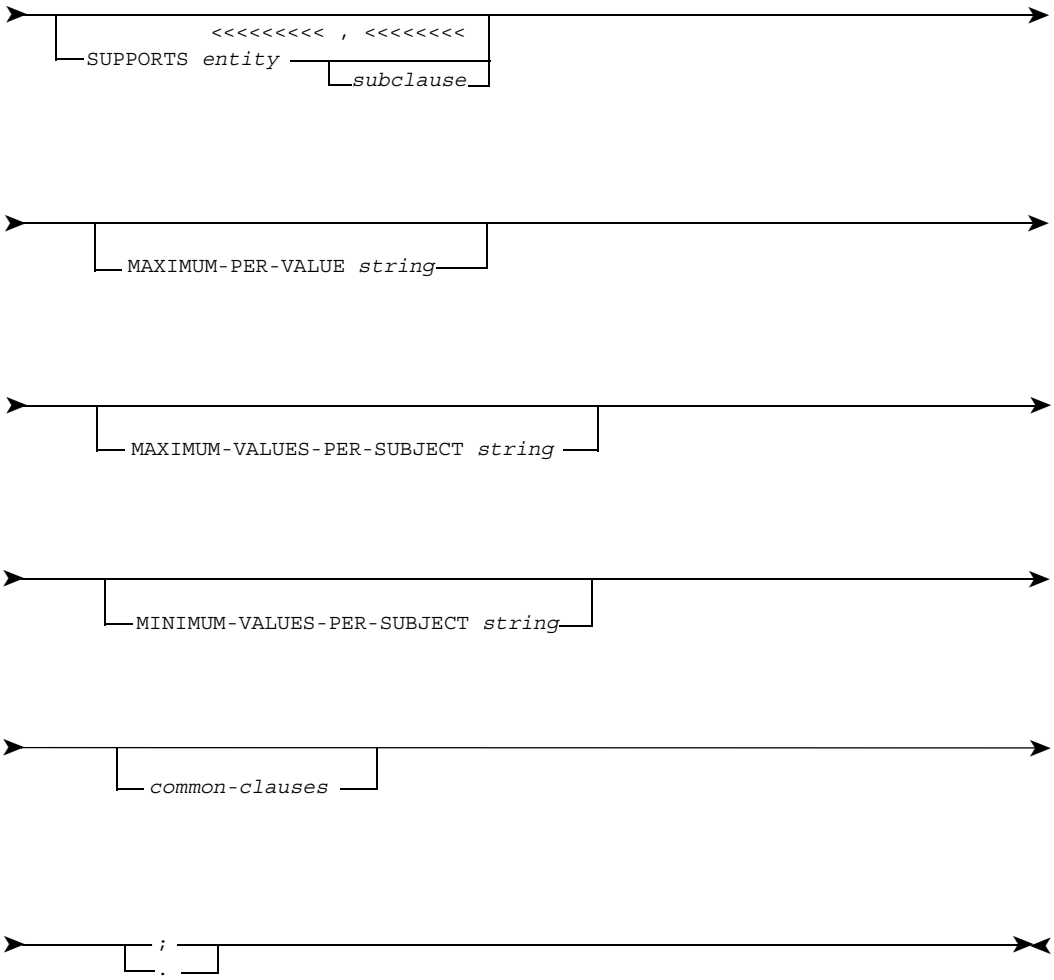
program is an IEW-PROGRAM member name.

module is an IEW-MODULE member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-Attribute-Type Syntax





where:

global-data is an IEW-GLOBAL-DATA-TYPE member name.

local-data is an IEW-LOCAL-DATA-TYPE member name.

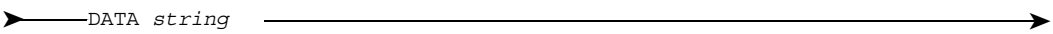
group is a GROUP member name.

item is an ITEM member name.

attribute is an IEW-ATTRIBUTE-TYPE member name.

relationship is an IEW-RELATIONSHIP-TYPE member name.

subclause is:

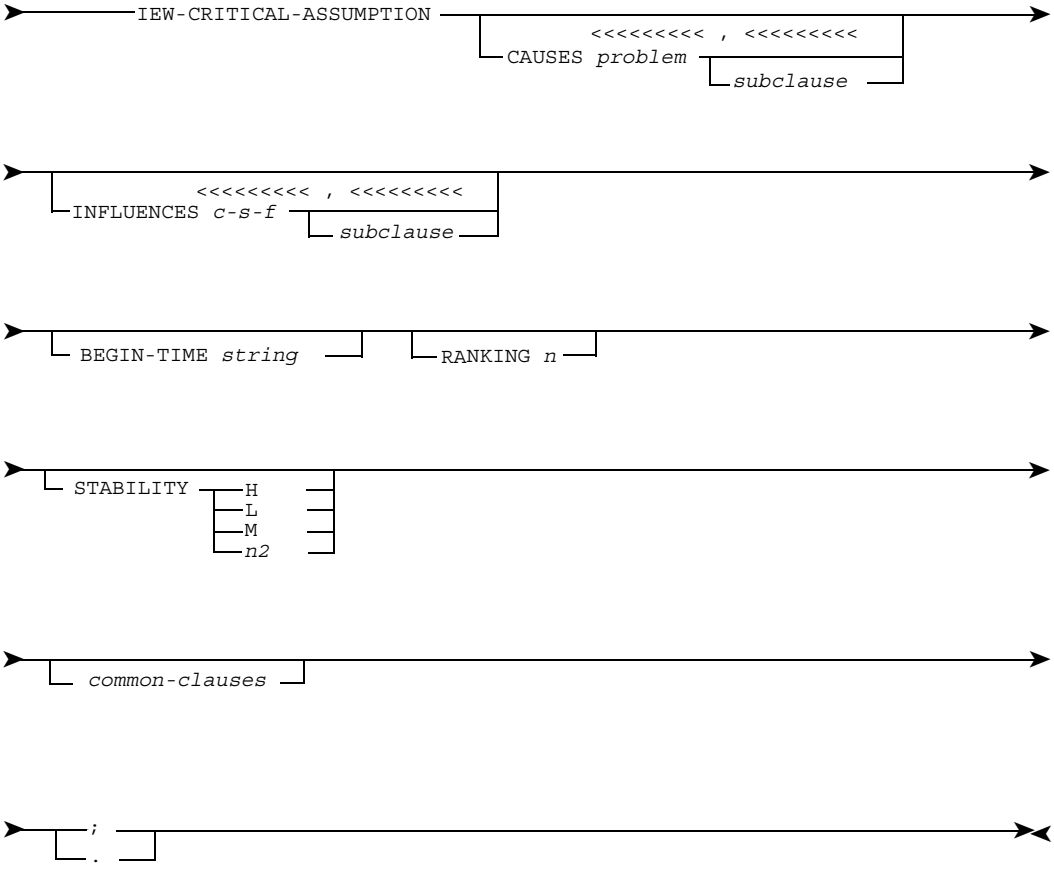


string is the data for import from, or export to, ADW/IEW.

entity is an IEW-ENTITY-TYPE member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-CRITICAL-ASSUMPTION Syntax

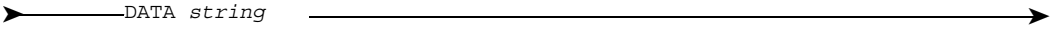


where:

problem is an IEW-PROBLEM member name.

c-s-f is an IEW-CRITICAL-SUCCESS-FACTOR member name.

subclause is



where:

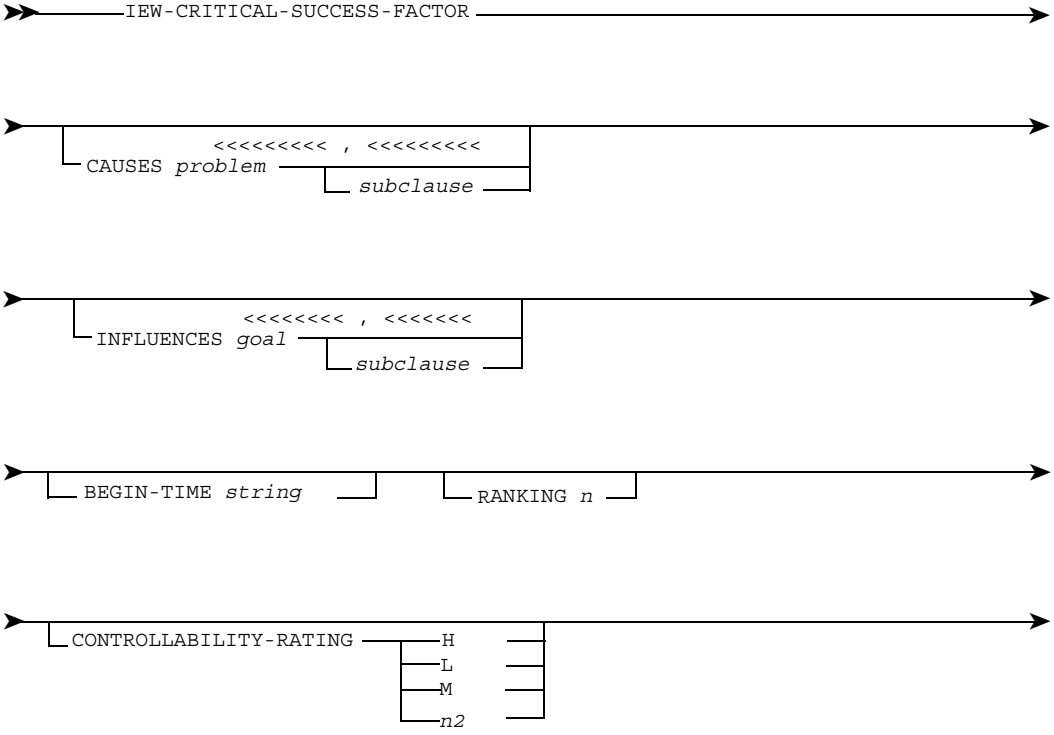
string is the data for import from, or export to, ADW/IEW.

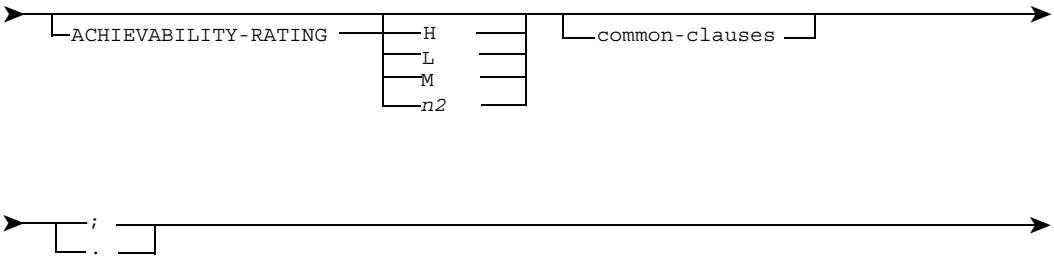
n is an unsigned numeric value in the range 1 to 9,999.

n2 is an unsigned numeric value in the range 1 to 999.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-CRITICAL-SUCCESS-FACTOR Syntax

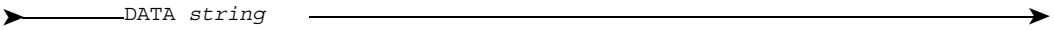




where:

problem is an IEW-PROBLEM member name.

subclause is:



string is the data for import from, or export to, ADW/IEW.

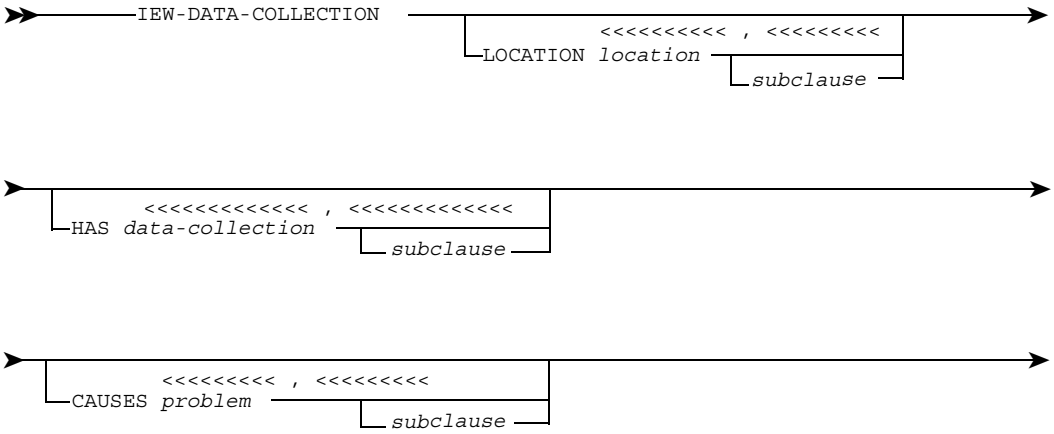
goal is an IEW-GOAL member name.

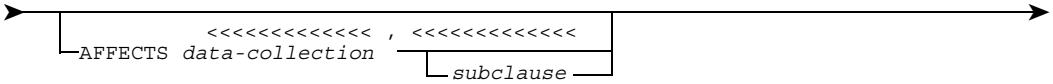
n is an unsigned numeric value in the range 1 to 9,999.

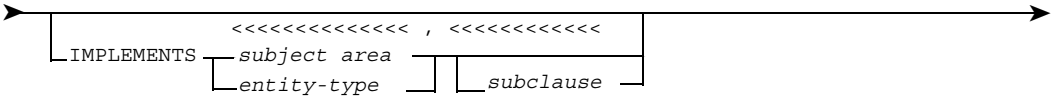
n2 is an unsigned numeric value in the range 1 to 999.

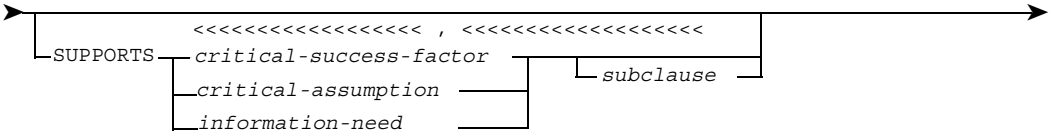
common-clauses are defined in "Common Clauses Syntax" on page 152.

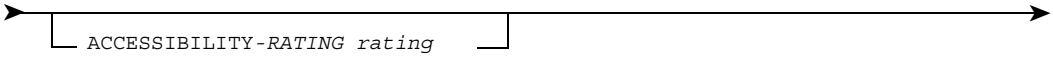
IEW-DATA-COLLECTION Syntax

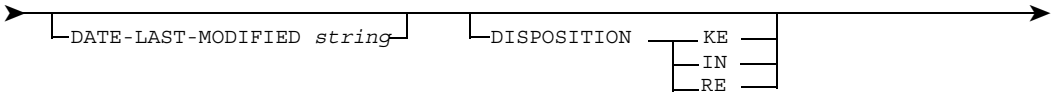


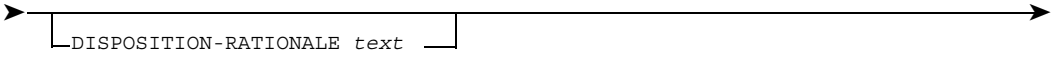

AFFECTS data-collection
<<<<<<<<<<<<< , <<<<<<<<<<<<<
subclause

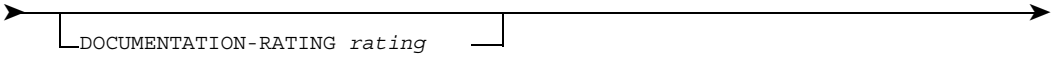

IMPLEMENTS
subject area
entity-type
<<<<<<<<<<<<<< , <<<<<<<<<<<<
subclause


SUPPORTS
critical-success-factor
critical-assumption
information-need
<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<
subclause

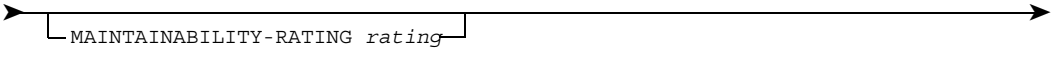

ACCESSIBILITY-RATING rating

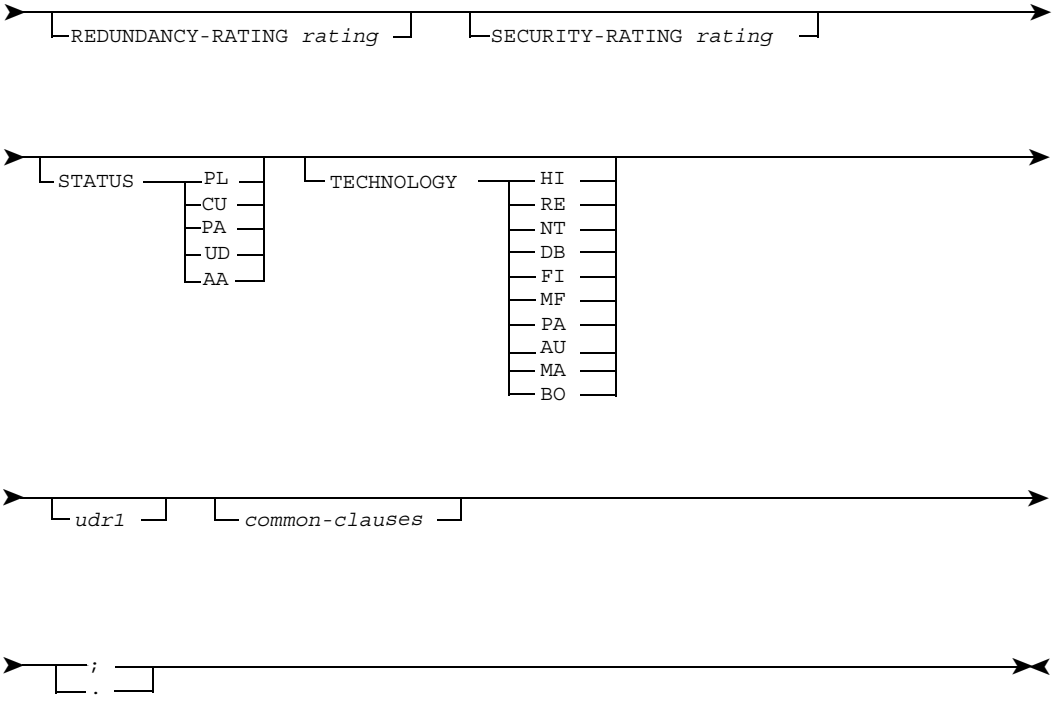

DATE-LAST-MODIFIED string
DISPOSITION
KE
IN
RE


DISPOSITION-RATIONALE text


DOCUMENTATION-RATING rating


EFFICIENCY-RATING rating
INTEGRITY-RATING rating

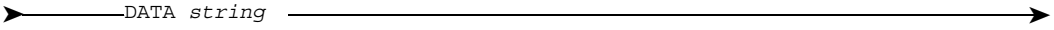

MAINTAINABILITY-RATING rating



where:

location is an IEW-LOCATION member name.

subclause is:



string is the data to be imported from, or exported to, ADW/IEW.

data-collection is an IEW-DATA-COLLECTION member name.

problem is an IEW-PROBLEM member name.

subject-area is an IEW-SUBJECT-AREA member name.

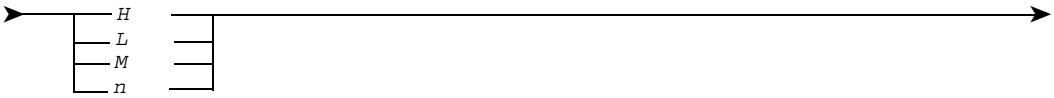
entity-type is an IEW-ENTITY-TYPE member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

information-need is an IEW-INFORMATION-NEED member name.

rating is:



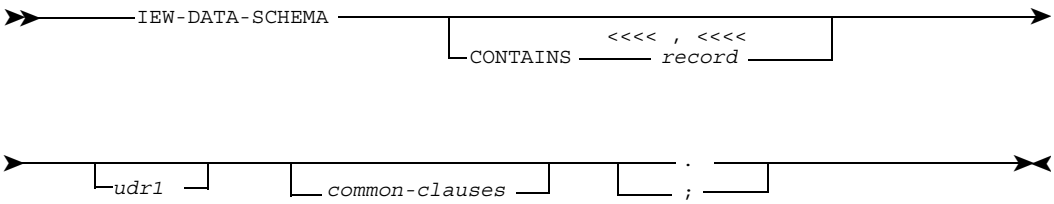
n is an integer in the range 1 to 999.

text is up to 32,767 delimited strings, each string being a maximum of 60 characters long.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATA-SCHEMA Syntax



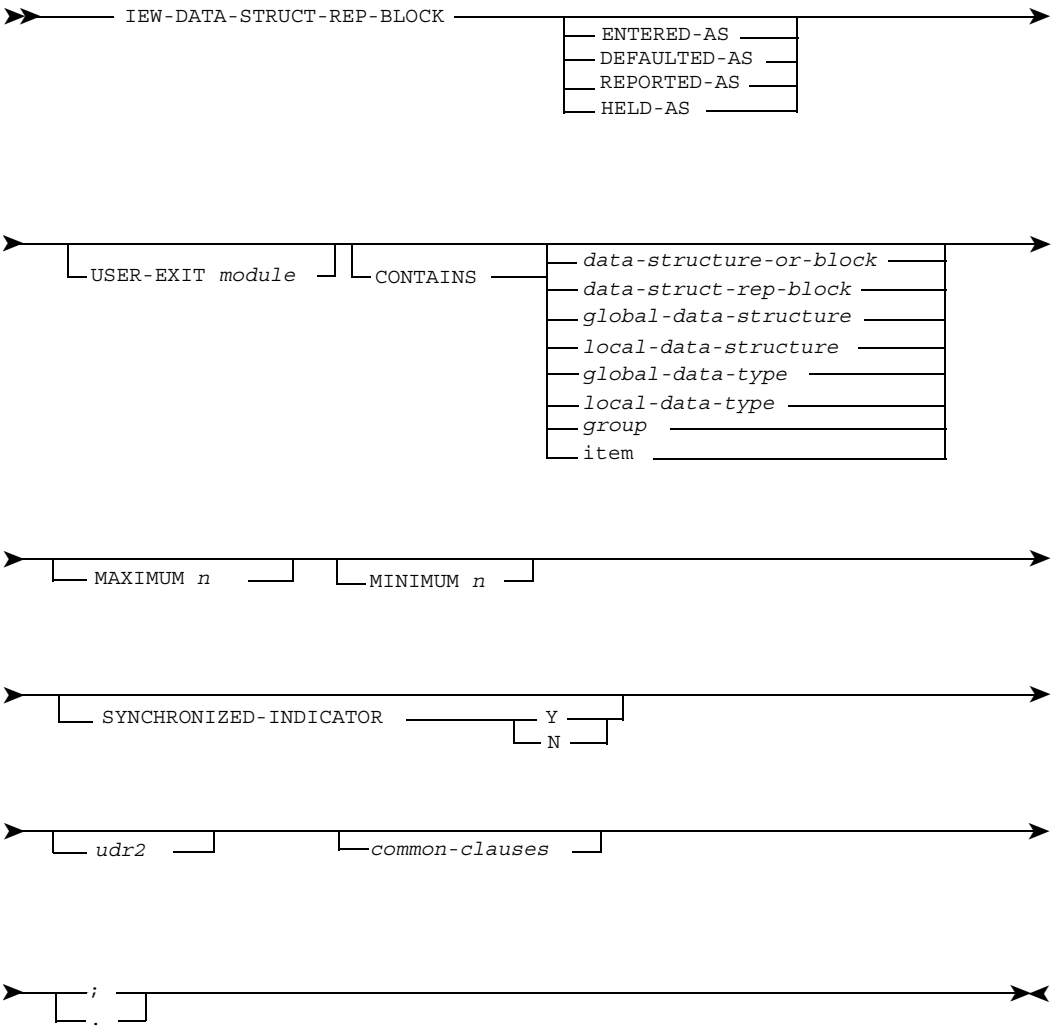
where:

record is an IEW-GLOBAL-DATA-RECORD member name.

udr1 is defined in section "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATA-STRUCT-REP-BLOCK Syntax



where:

module is an IEW-MODULE member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-type is an IEW-GLOBAL-DATA-TYPE member name.

local-data-type is an IEW-LOCAL-DATA-TYPE member name.

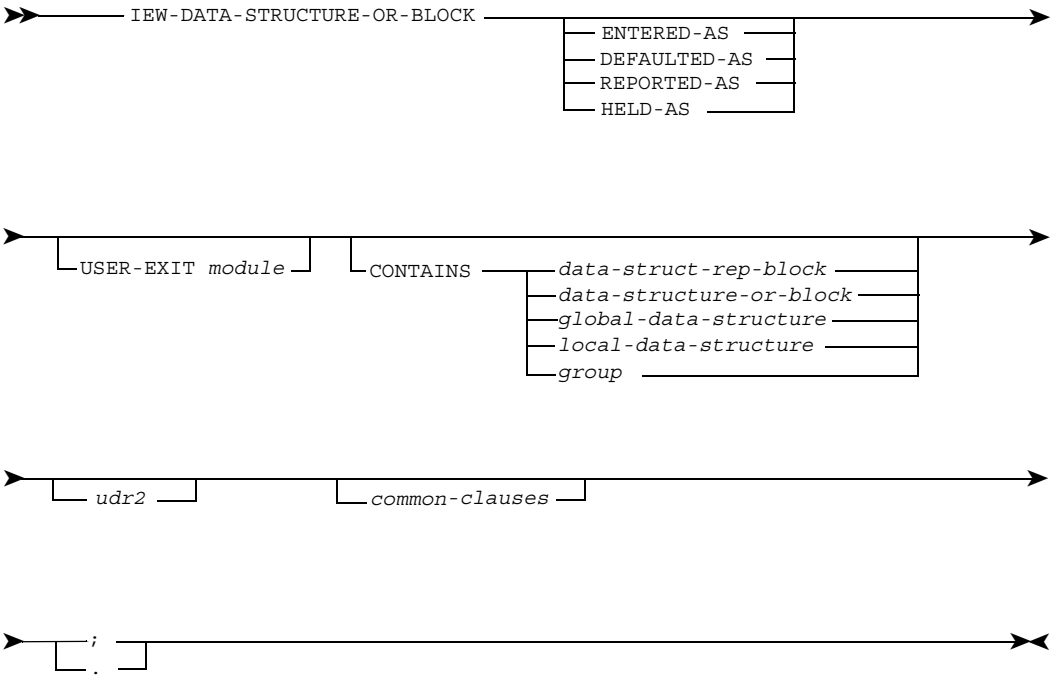
group is a GROUP member name.

n is an integer in the range 1 to 32,767.

udr2 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATA-STRUCTURE-OR-BLOCK Syntax



where:

module is an IEW-MODULE member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK

member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK

member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE membe name.

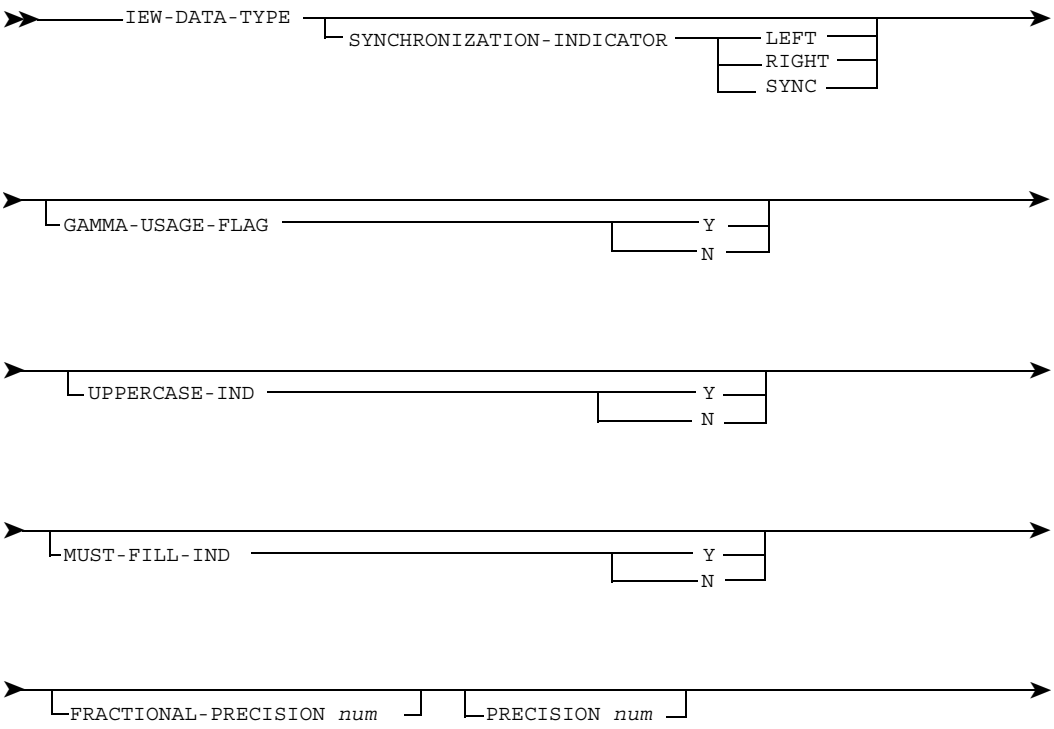
global-data-type is an IEW-GLOBAL-DATA-STRUCTURE member name.

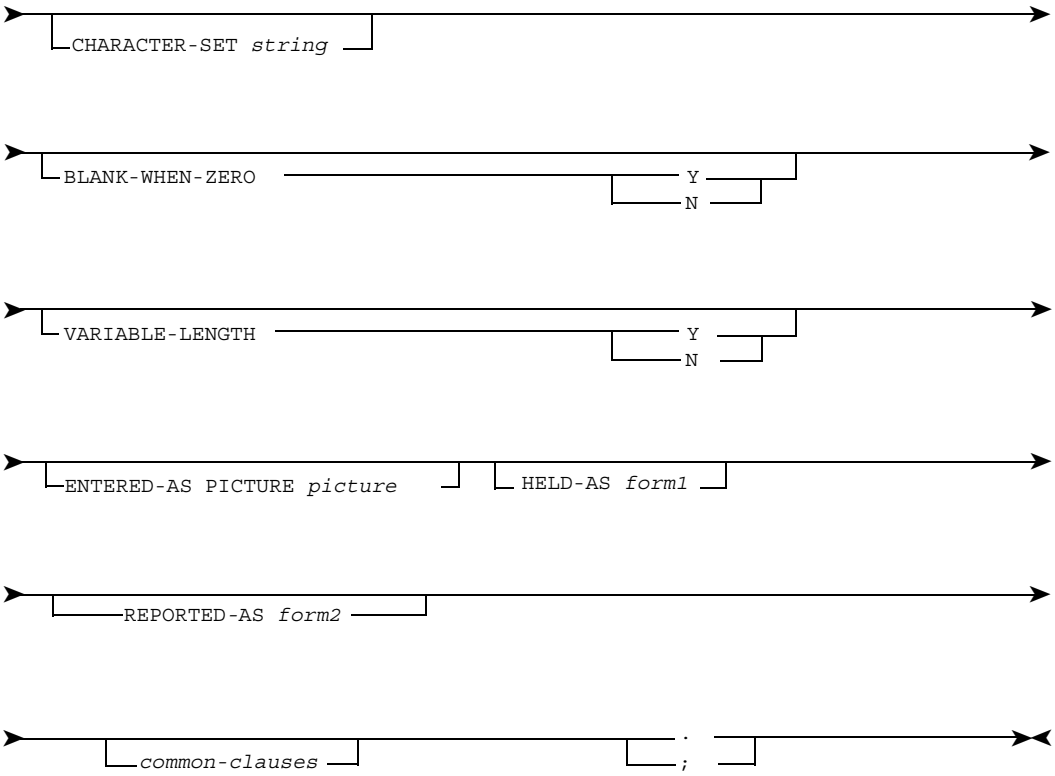
group is a GROUP member name.

udr2 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATA-Type Syntax





where:

num is a small unsigned integer.

string is data to be imported from, or exported to, ADW/IEW.

picture is a valid data-type format.

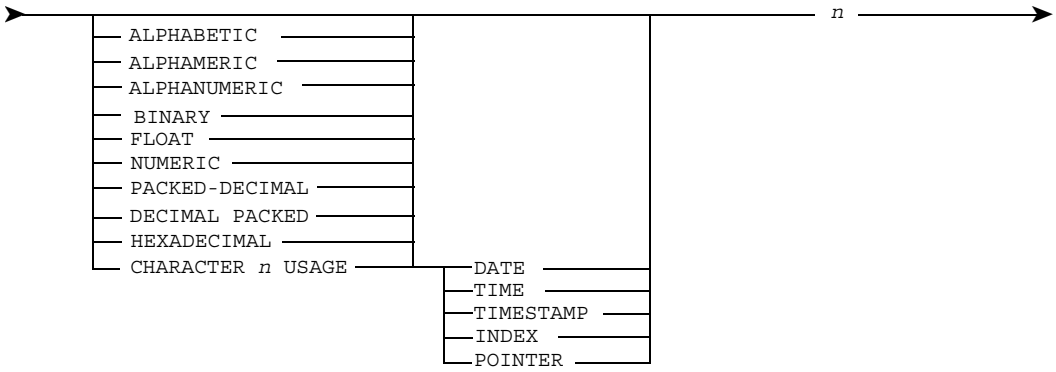
common-clauses are defined in "Common Clauses Syntax" on page 152.

form1 is:

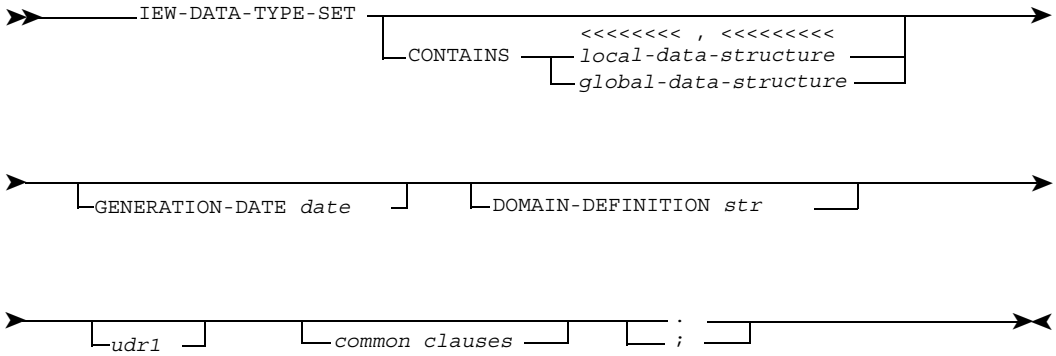
where:

n is an unsigned decimal number specifying a maximum length.

form2 is:



IEW-DATA-TYPE-SET Syntax



where:

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

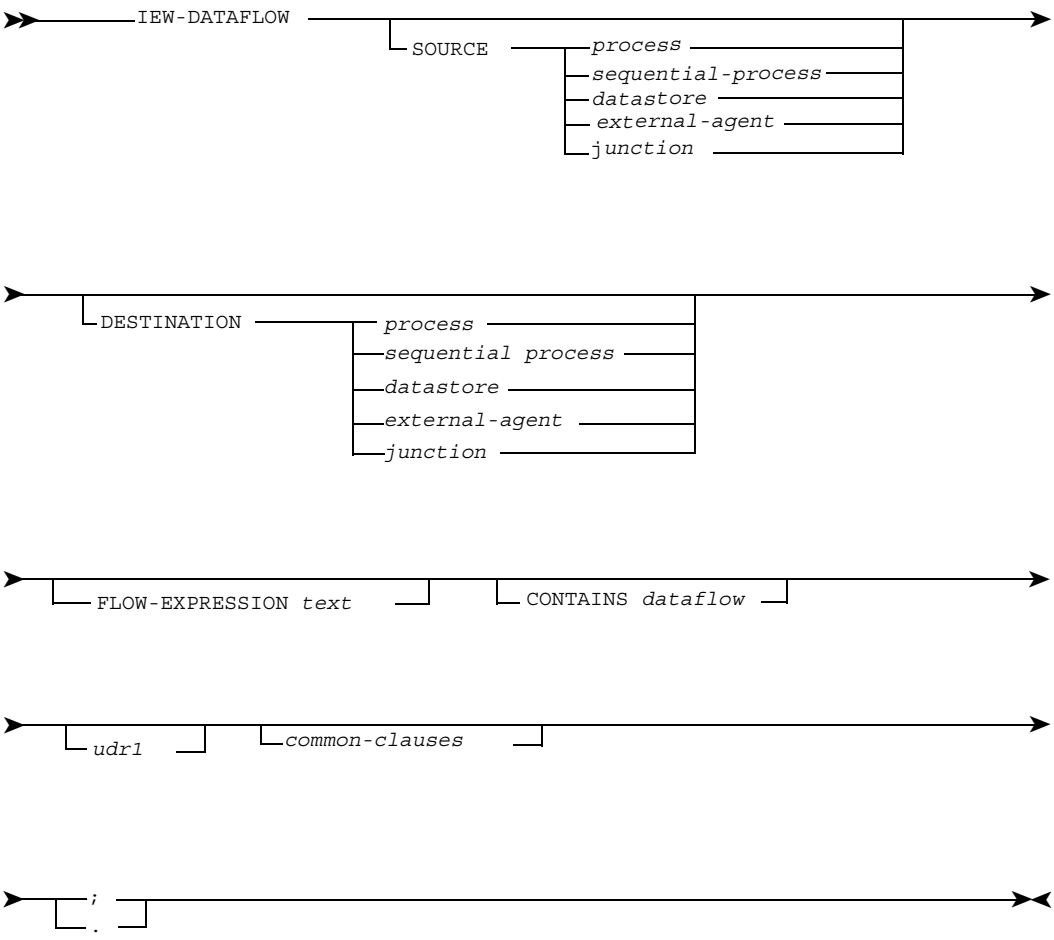
date is a string in date format.

str is data to be imported from, or exported to, ADW/IEW.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATAFLOW Syntax



where:

process is an IEW-PROCESS member name.

sequential-process is an IEW-SEQUENTIAL-PROCESS member name.

datastore is an IEW-DATASTORE member name.

external-agent is an IEW-EXTERNAL-AGENT member name.

junction is an IEW-JUNCTION member name.

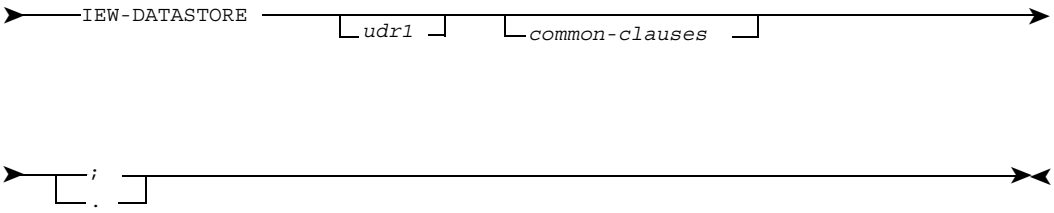
text is up to 32,767 delimited strings, each string being a maximum of 60 characters long.

dataflow is an IEW-DATAFLOW member name.

udr1 is defined "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined "Common Clauses Syntax" on page 152.

IEW-DATASTORE Syntax

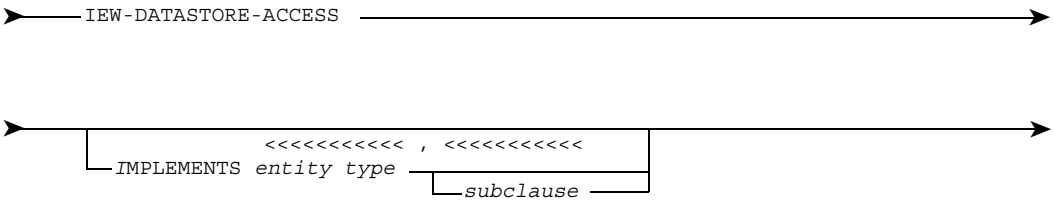


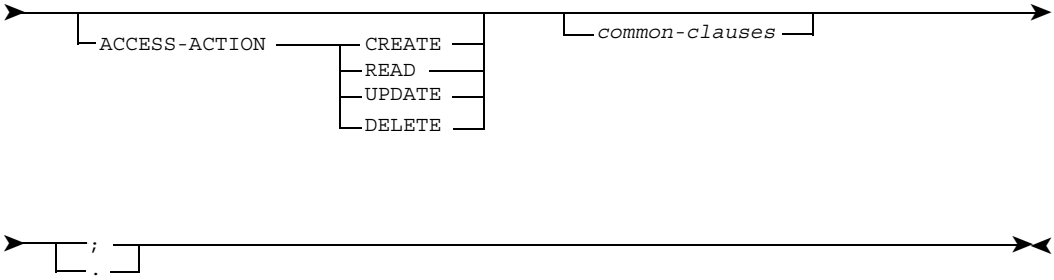
where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DATASTORE-ACCESS Syntax

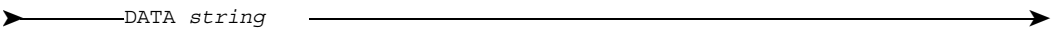




where:

entity-type is an IEW-ENTITY-TYPE member name.

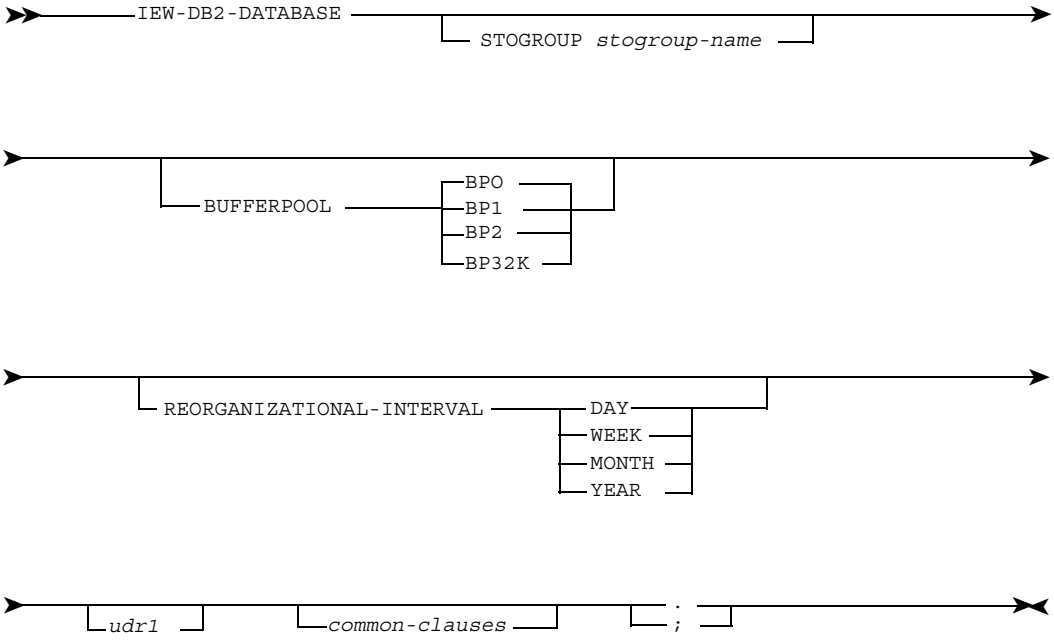
subclause is:



string is the data to be imported from, or exported to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-DATABASE Syntax



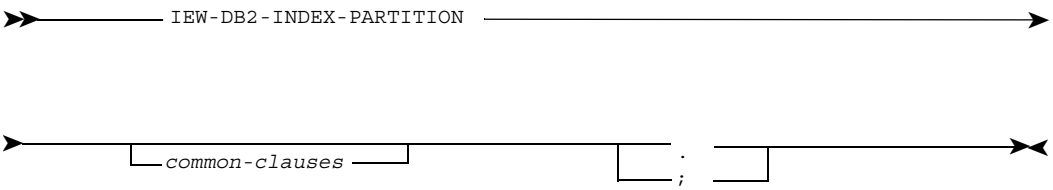
where:

stogroup-name is the name of an IEW-DB2-STOGROUP member.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

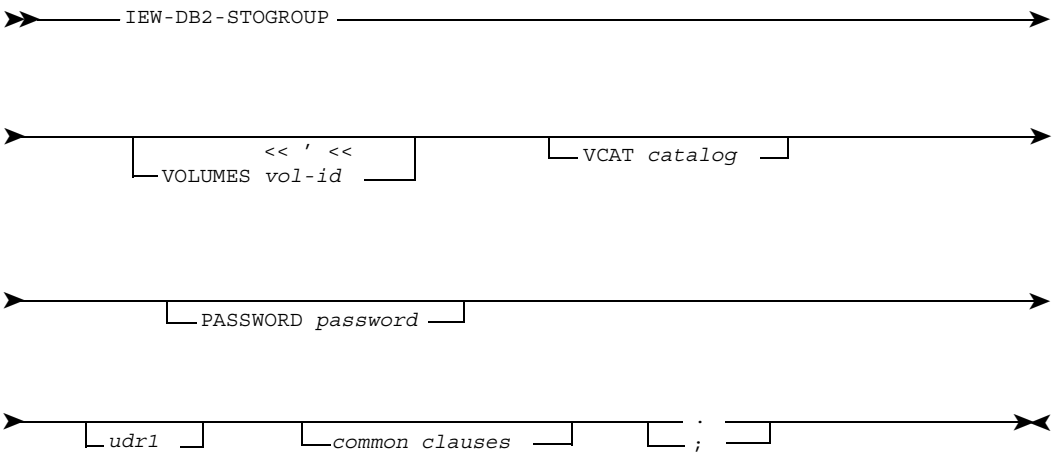
common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-INDEX-PARTITION Syntax



where *common-clauses* are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-STOGROUP Syntax



where:

vol-id is a storage volume name, of no more than 6 characters.

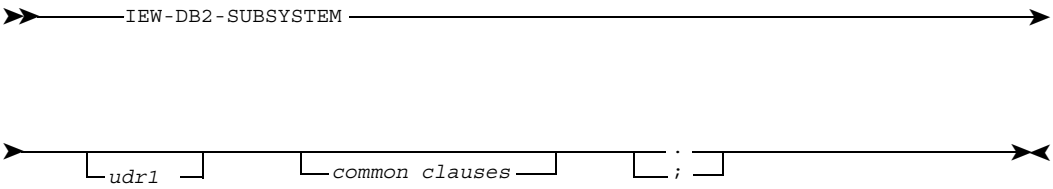
catalog is a VSAM catalog password, of no more than 8 characters.

password is a VSAM catalog password, of no more than 8 characters.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-SUBSYSTEM Syntax

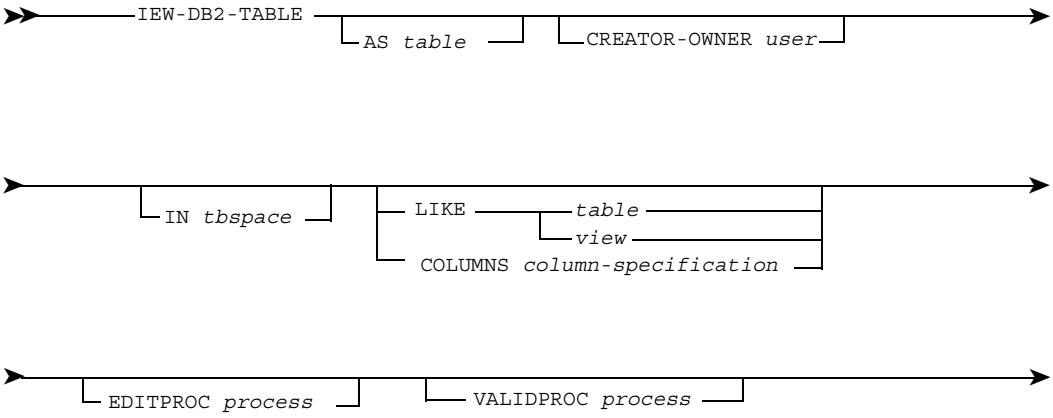


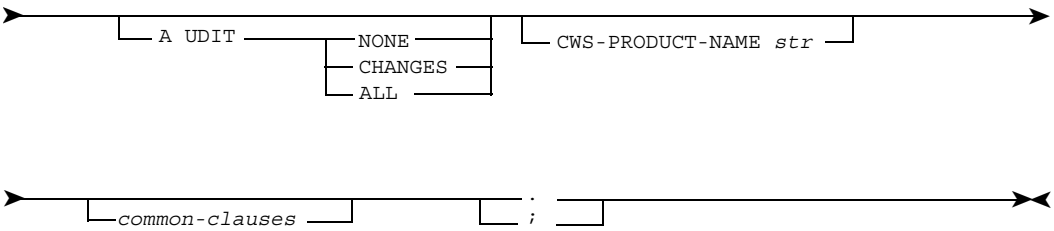
where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-TABLE Syntax





where:

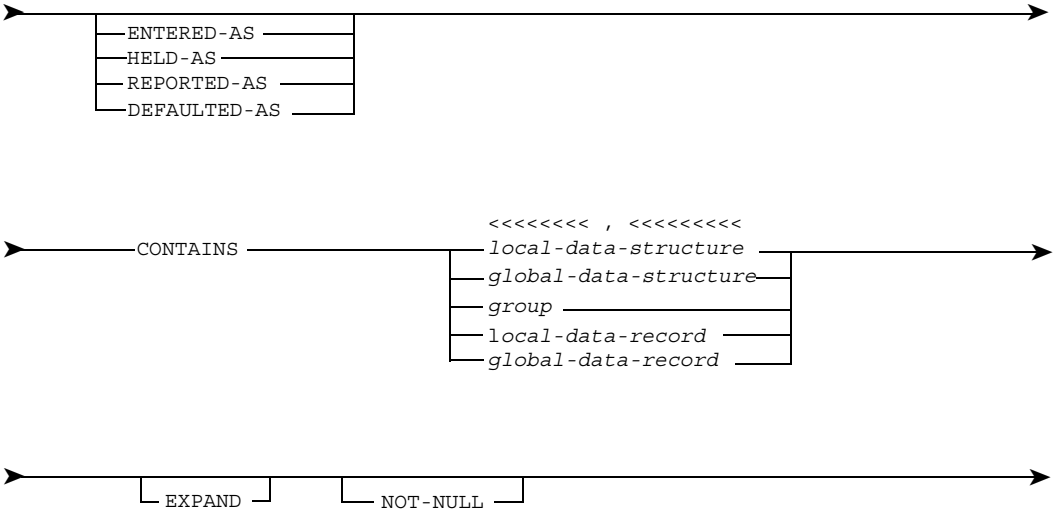
table is an IEW-DB2-TABLE member name.

user is a DB2-USER member name.

tbspace is an IEW-TBSPACE member name.

view is an IEW-DB2-VIEW member name.

column-specification is:



where:

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

group is a GROUP member name.

local-data-record is an IEW-LOCAL-DATA-RECORD member name.

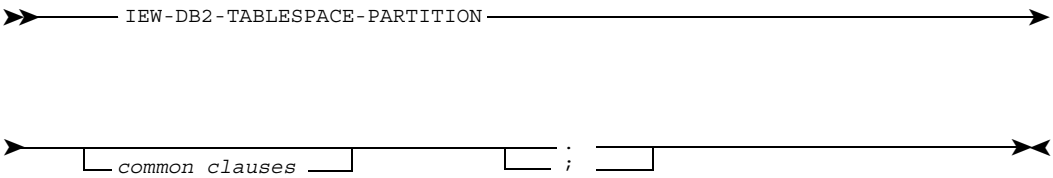
global-data-record is an IEW-GLOBAL-DATA-RECORD member name.

process is the name of a SYSTEM, PROGRAM, or MODULE member name.

str is data to be imported from, or exported to, ADW/IEW.

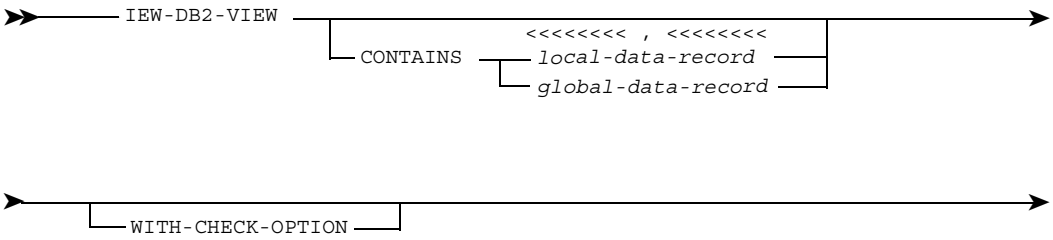
common-clauses are defined in "Common Clauses Syntax" on page 152.

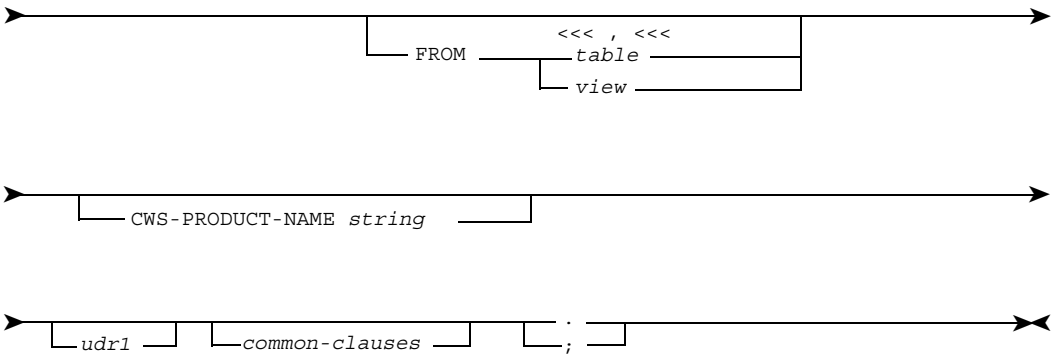
IEW-DB2-TABLESPACE-PARTITION Syntax



where *common-clauses* are defined in "Common Clauses Syntax" on page 152.

IEW-DB2-VIEW Syntax





where:

local-data-record is an IEW-LOCAL-DATA-RECORD member name.

global-data-record is an IEW-GLOBAL-DATA-RECORD member name.

table is an IEW-DB2-TABLE member name.

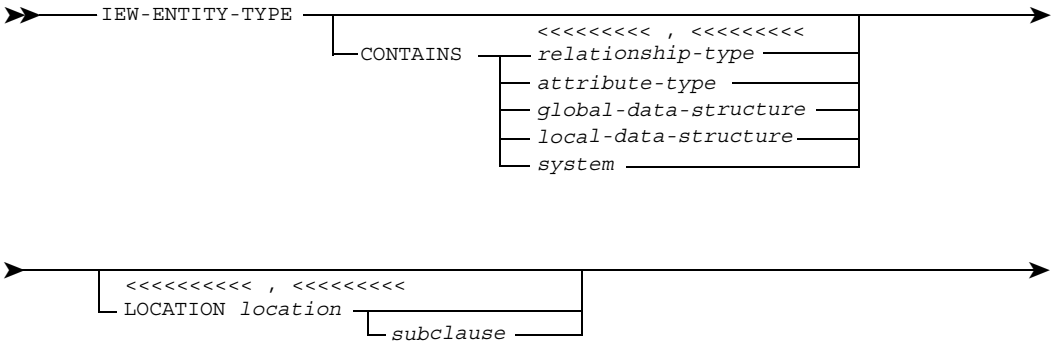
view is an IEW-DB2-TABLE member name.

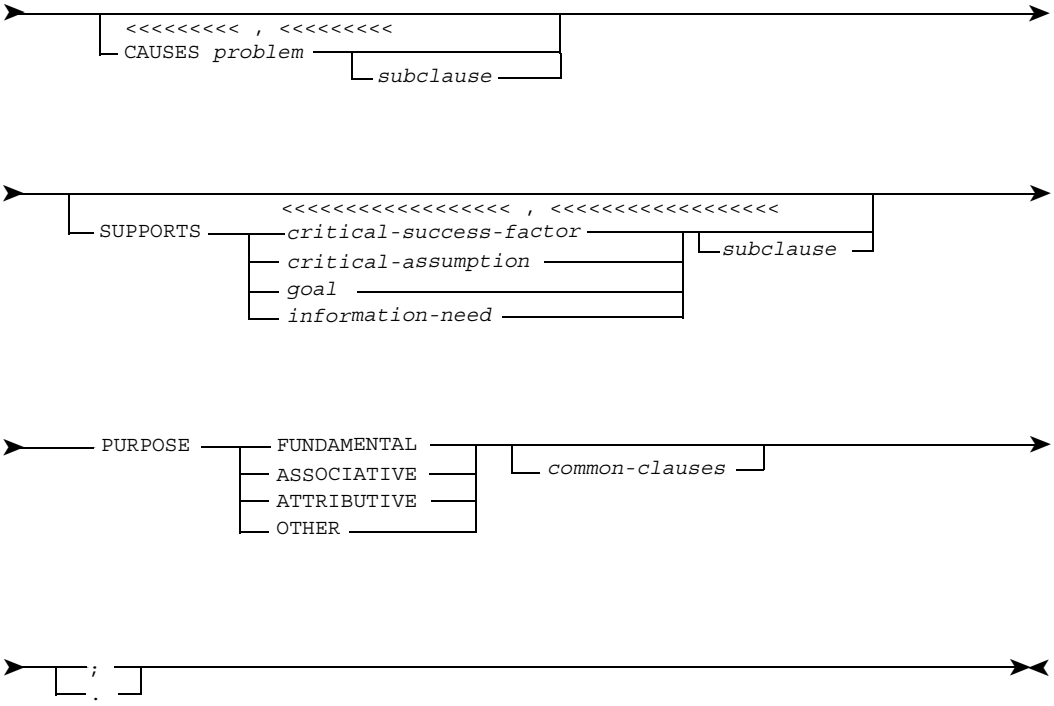
string is data to be imported from, or exported to, ADW/IEW.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

Common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-ENTITY-TYPE Syntax





where:

relationship-type is an IEW-RELATIONSHIP-TYPE member name.

attribute-type is an IEW-ATTRIBUTE-TYPE member name.

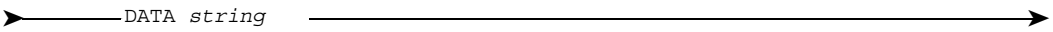
global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

system is a SYSTEM member name.

location is an IEW-LOCATION member name.

subclause is:



string is data to be imported from, or exported to, ADW/IEW.

problem is an IEW-PROBLEM member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

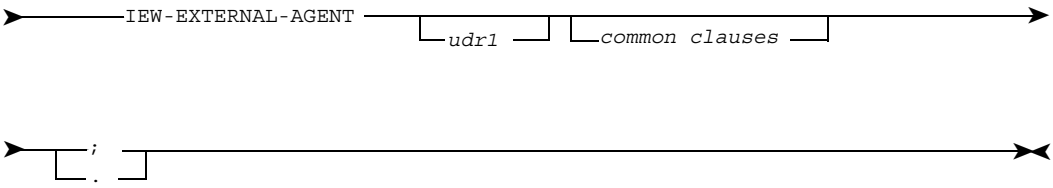
critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

goal is an IEW-GOAL-NEED member name.

information-need is an IEW-INFORMATION-NEED member name.

Common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-EXTERNAL-AGENT Syntax

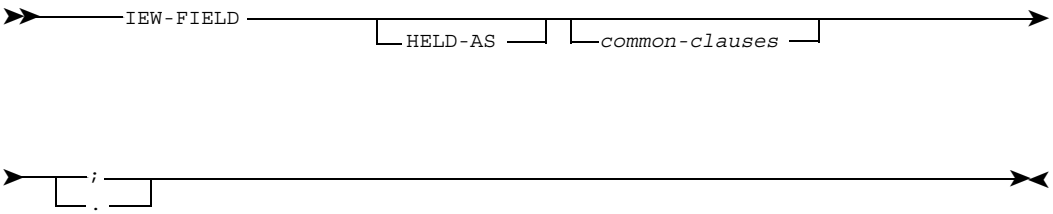


where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

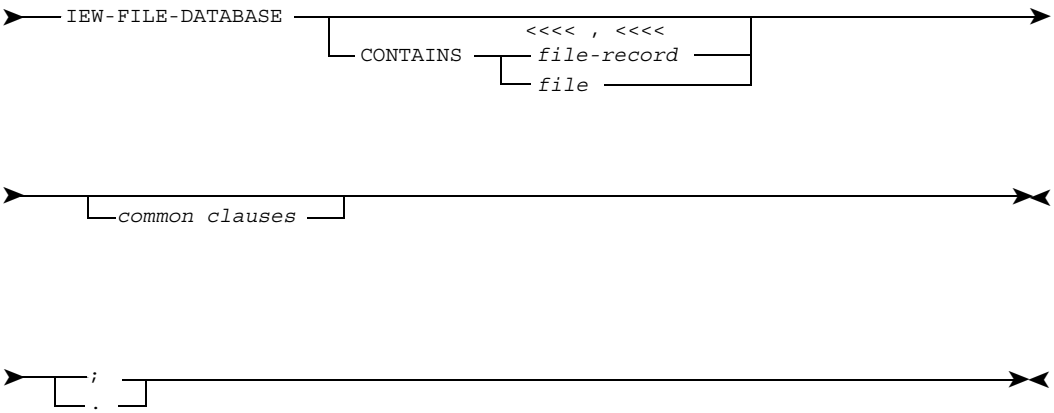
common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-FIELD Syntax



where *common-clauses* are defined in "Common Clauses Syntax" on page 152.

IEW-FILE-DATABASE Syntax



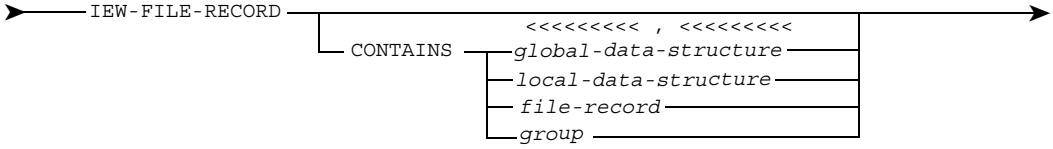
where:

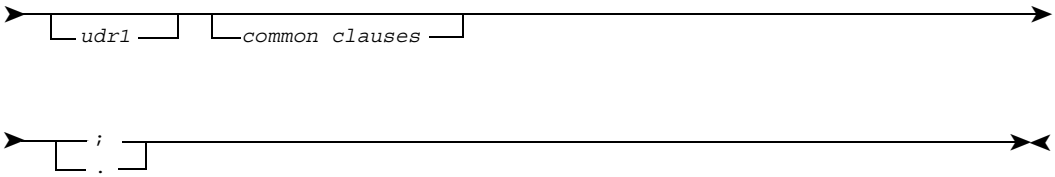
file-record is an IEW-FILE-RECORD member name.

file is a FILE member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-FILE-RECORD Syntax





where:

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

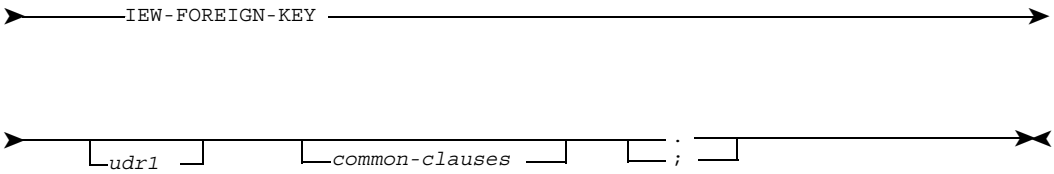
file-record is an IEW-FILE-RECORD member name.

group is a GROUP member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in"Common Clauses Syntax" on page 152.

IEW-FOREIGN-KEY Syntax

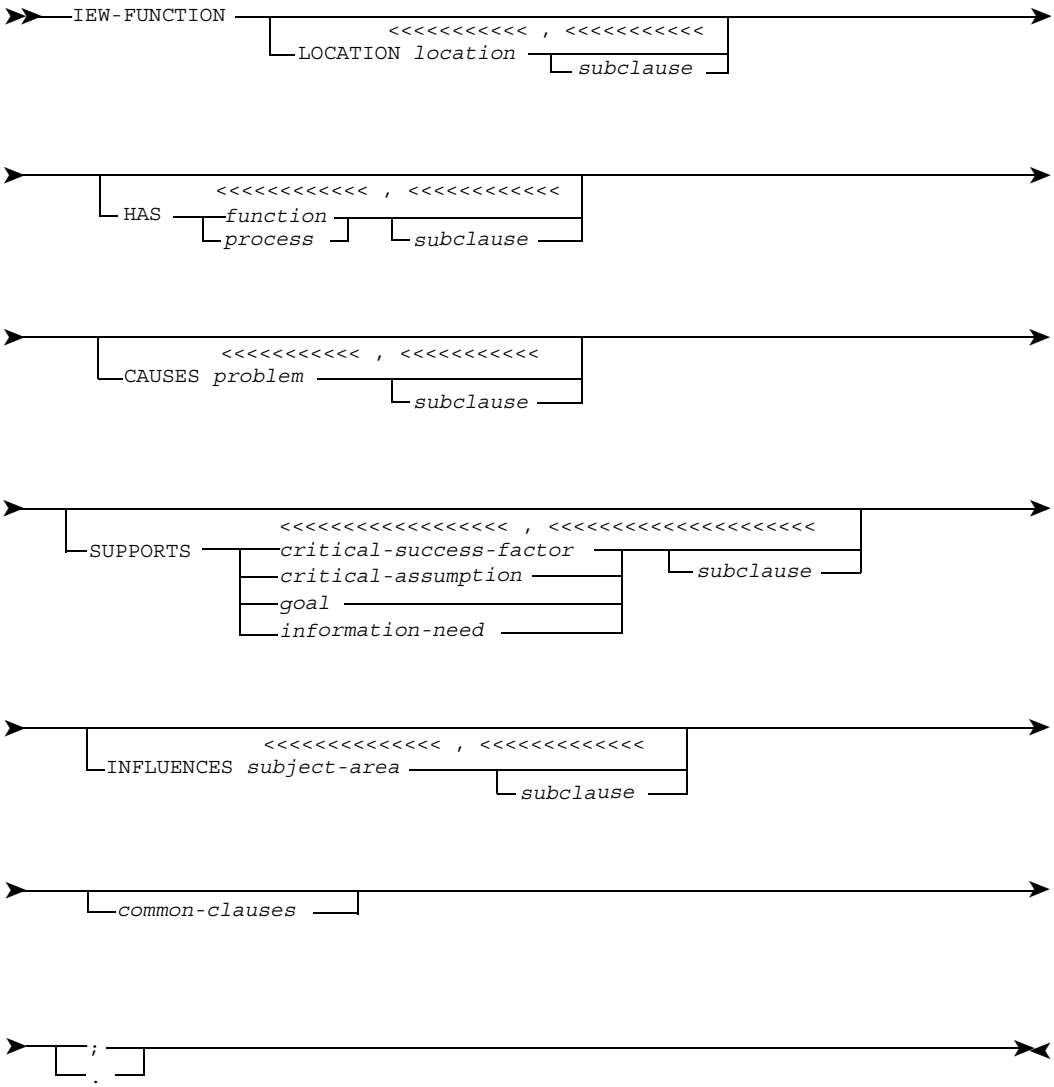


where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

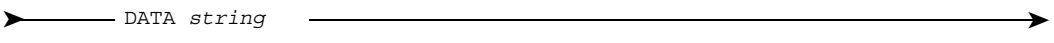
IEW-FUNCTION Syntax



where:

location is an IEW-LOCATION member name.

subclause is:



string is the data to be imported from, or exported to, ADW/IEW.

function is an IEW-FUNCTION member name.

process is an IEW-PROCESS member name.

problem is an IEW-PROBLEM member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

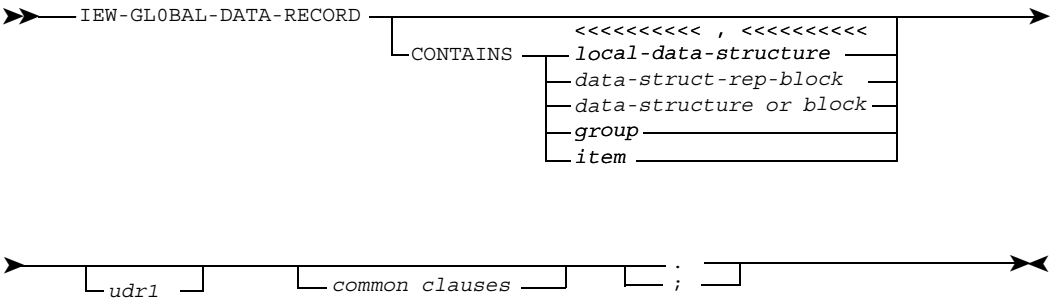
goal is an IEW-GOAL member name.

information-need is an IEW-INFORMATION-NEED member name.

subject-area is an IEW-SUBJECT-AREA member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-GLOBAL-DATA-RECORD Syntax



where:

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

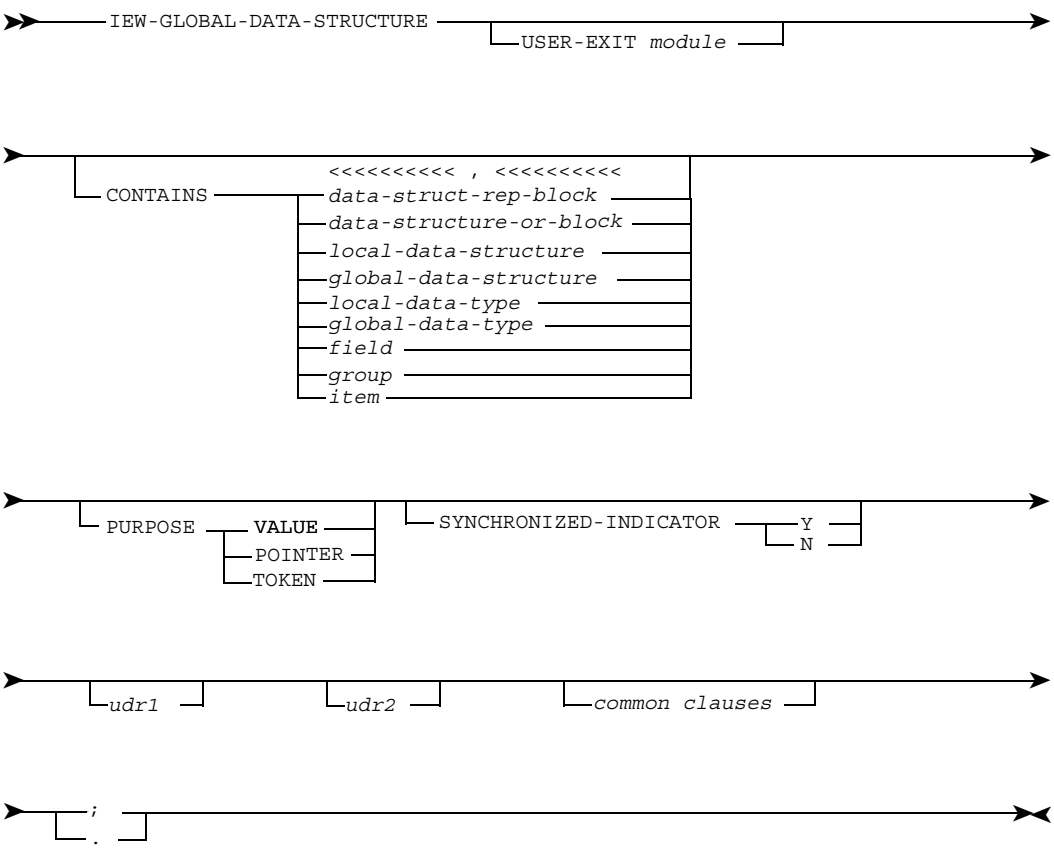
group is a GROUP member name.

item is an ITEM member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-GLOBAL-DATA-STRUCTURE Syntax



where:

module is an IEW-MODULE member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-type is an IEW-GLOBAL-DATA-TYPE member name.

local-data-type is an IEW-LOCAL-DATA-TYPE member name.

field is an IEW-FIELD member name.

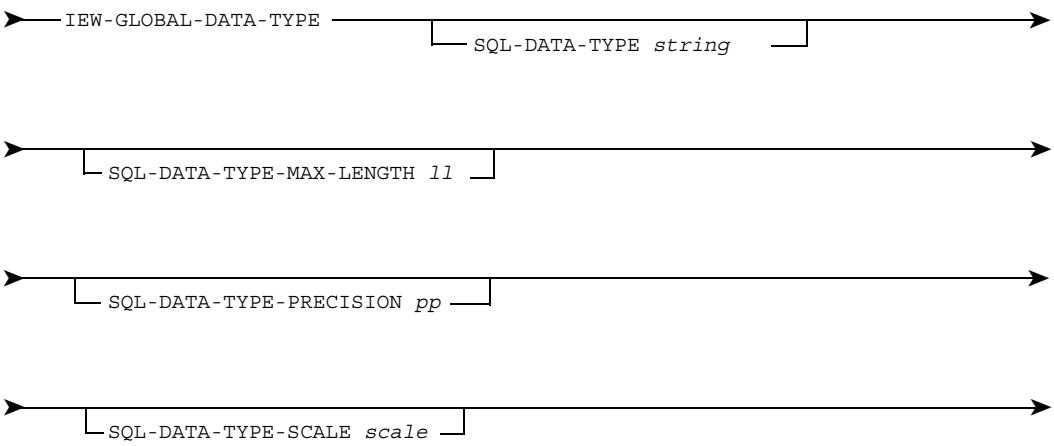
group is a GROUP member name.

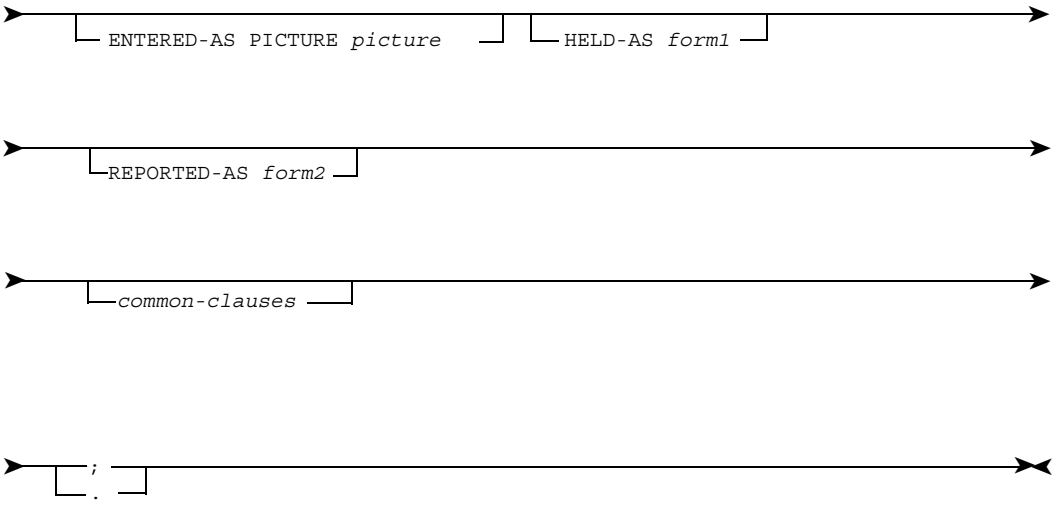
item is an ITEM member name.

udr1 and *udr2* are defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-GLOBAL-DATA-TYPE Syntax





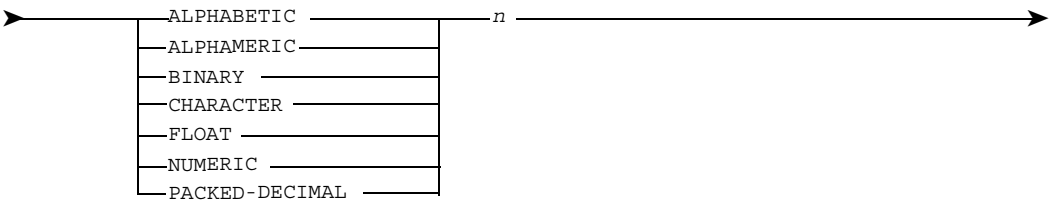
where:

string is data for import from, or export to, ADW/IEW.

ll, *pp*, and *scale* are unsigned integers.

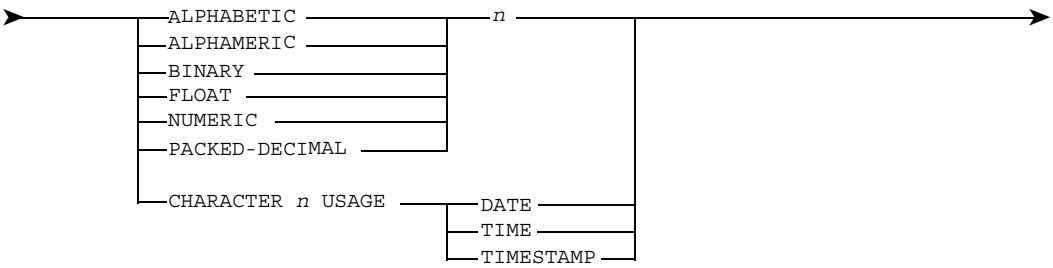
picture is a valid data-type format.

form1 is:



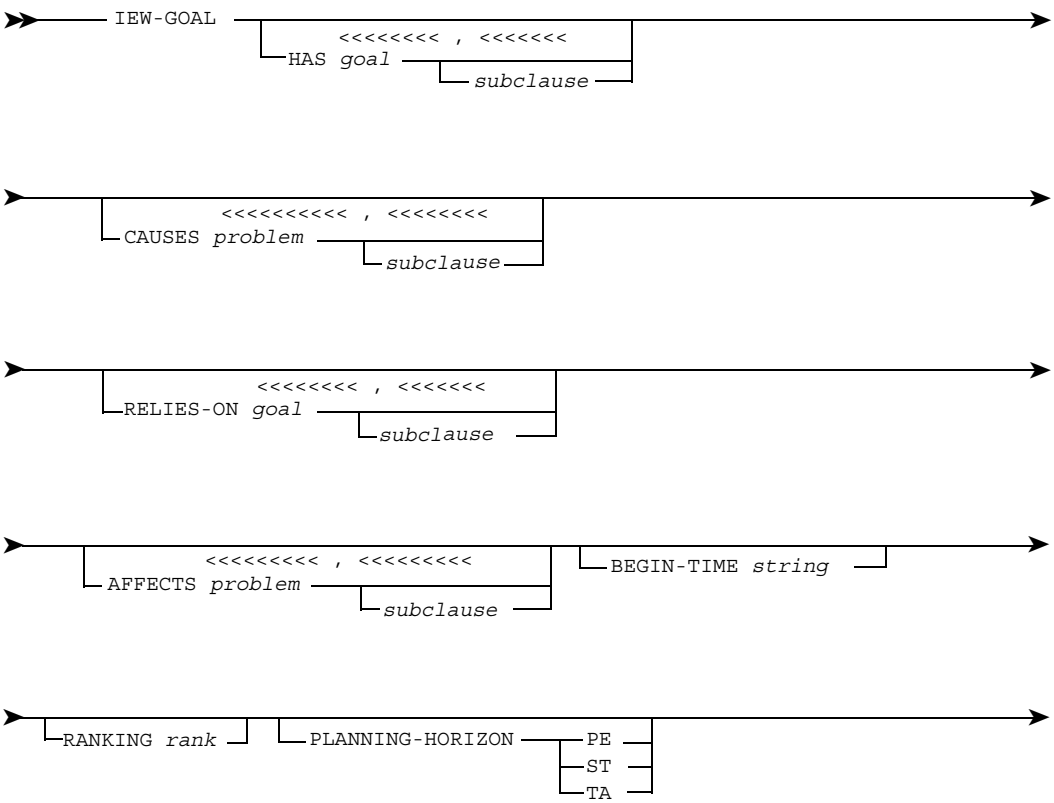
n is an unsigned decimal number specifying a maximum length.

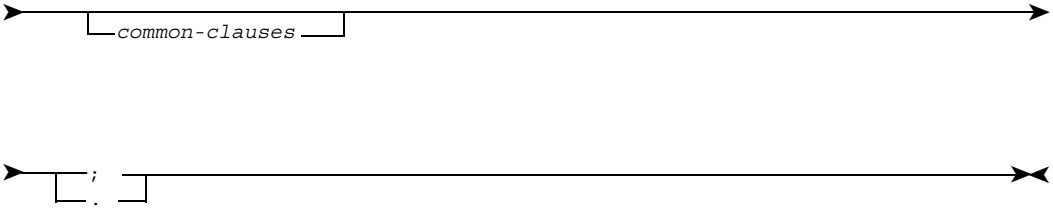
form2 is:



common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-GOAL Syntax

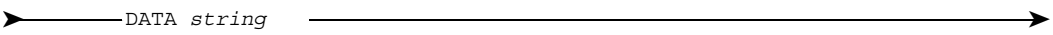




where:

goal is an IEW-GOAL member name.

subclause is:



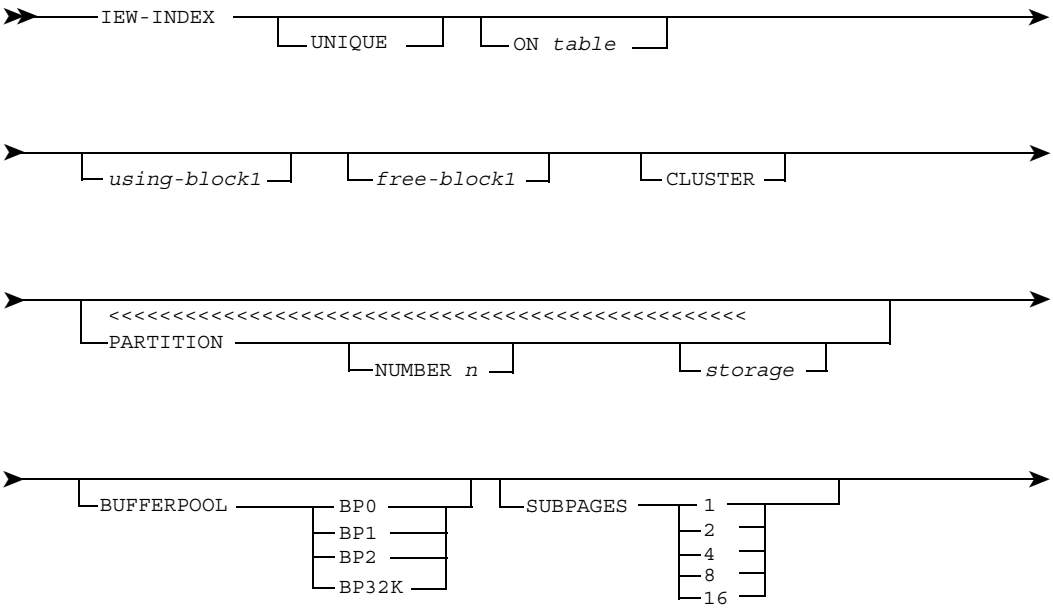
string is the data to be imported from, or exported to, ADW/IEW.

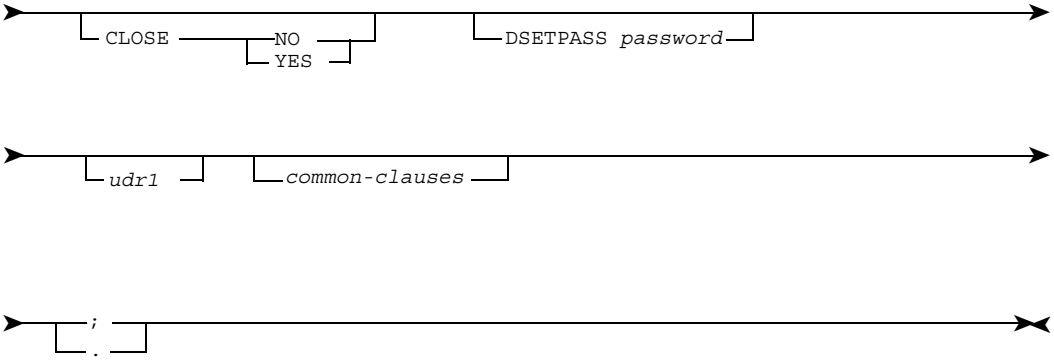
problem is an IEW-PROBLEM member name.

rank is an unsigned integer in the range 1 to 9,999.

common-clauses are defined in"Common Clauses Syntax" on page 152.

IEW-INDEX Syntax

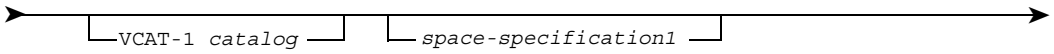




where:

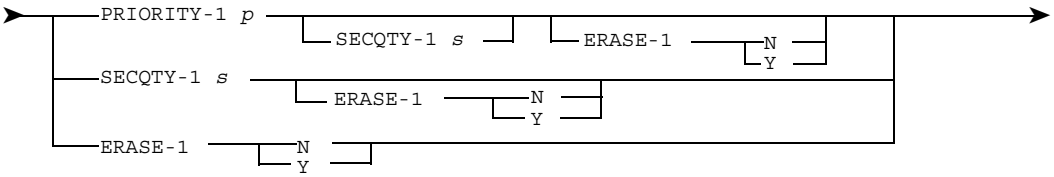
table is an IEW-DB2-TABLE member name.

using-block1 is:



catalog is a VSAM catalog name of no more than 8 characters.

space-specification1 is:



where:

p is an integer in the range 3 to 4,194,304.

s is an integer in the range 0 to 131,068.

free-block1 is:

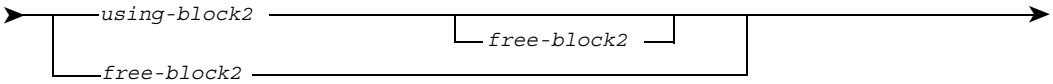
where:

fn is an integer in the range 0 to 255.

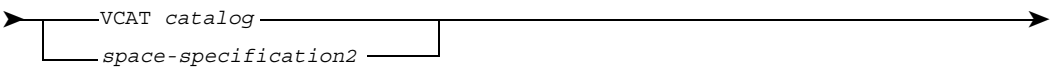
pn is an integer in the range 0 to 99.

*n i*s an integer in the range 1 to 64.

storage is:

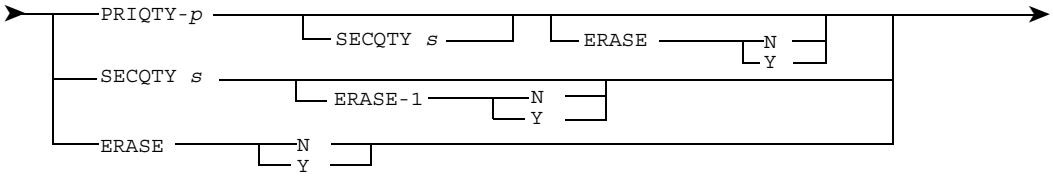


using-block2 is:



catalog is as defined above.

space-specification2 is:

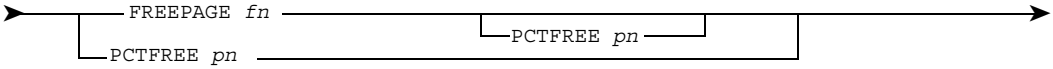


where:

p is as defined above.

s is as defined above.

free-block2 is:



fn is as defined above.

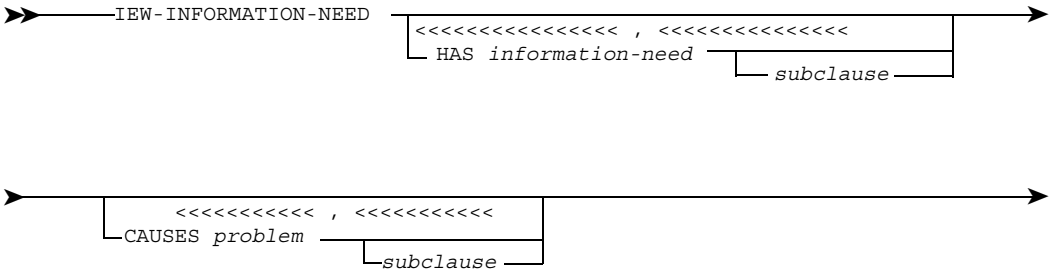
pn is as defined above.

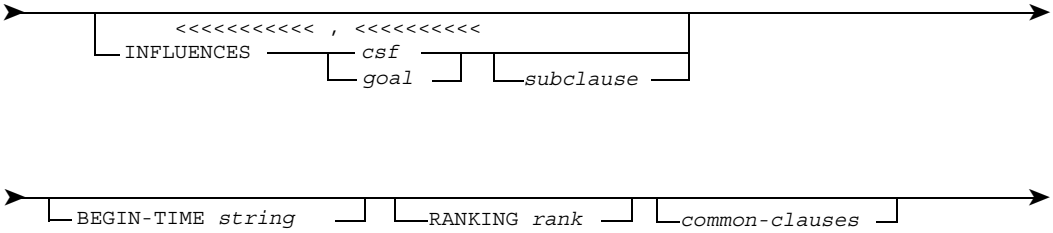
password is a VSAM data set password, of no more than 8 characters.

common-clauses are defined in "Common Clauses Syntax" on page 152.

udr1 is defined in f.

IEW-INFORMATION-NEED Syntax

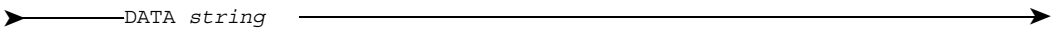




where:

information-need is an IEW-INFORMATION-NEED member name.

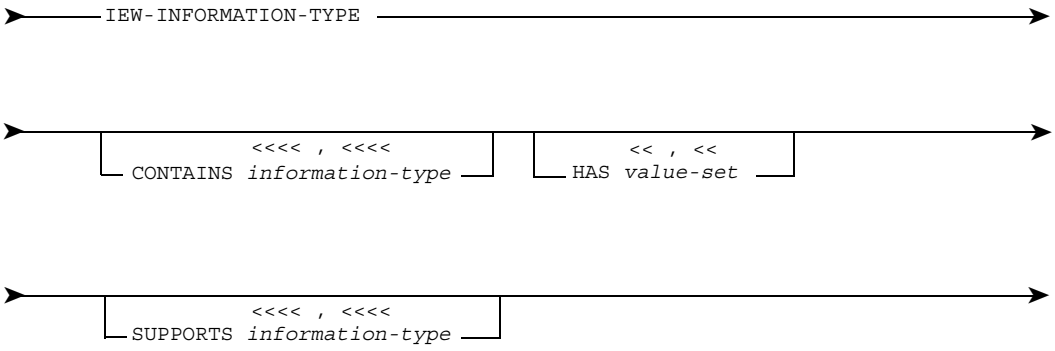
subclause is:

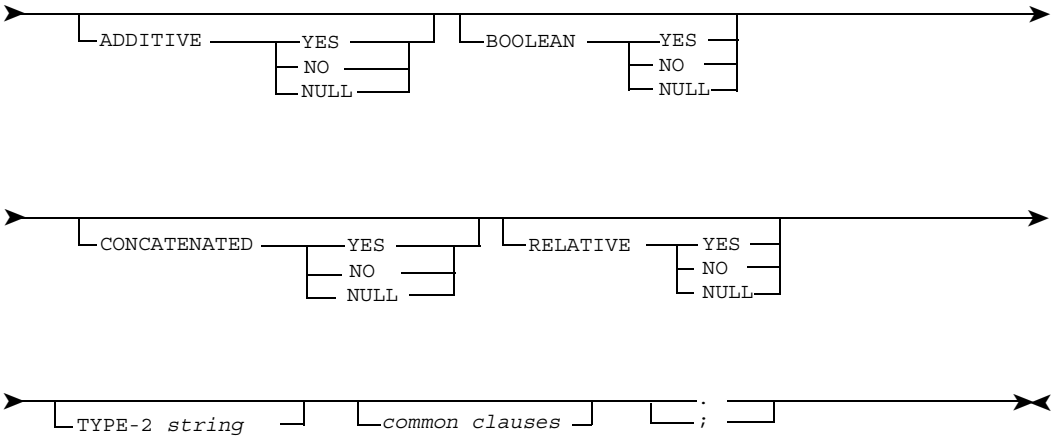


- *string* is the data to be imported from, or exported to, ADW/IEW.
- *problem* is an IEW-PROBLEM member name.
- *csf* is an IEW-CRITICAL-SUCCESS-FACTOR member name.
- *goal* is an IEW-GOAL member name.
- *rank* is an unsigned integer in the range 1 to 9,999.

common-clauses are defined in section"Common Clauses Syntax" on page 152.

IEW-INFORMATION-TYPE Syntax





where:

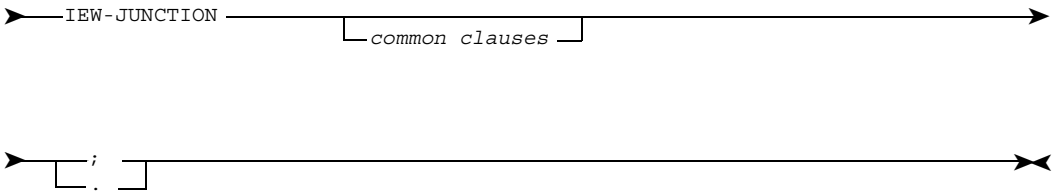
information-type is an IEW-INFORMATION-TYPE member name.

value-set is an IEW-VALUE-SET member name.

string is data to be imported from, or exported to, ADW/IEW.

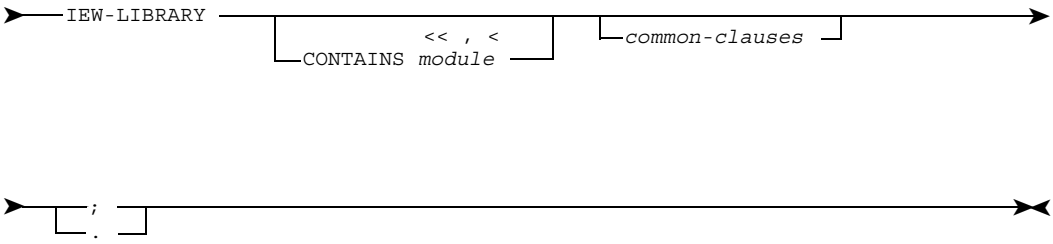
common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-JUNCTION Syntax



where *common-clauses* are defined in "Common Clauses Syntax" on page 152.

IEW-LIBRARY Syntax

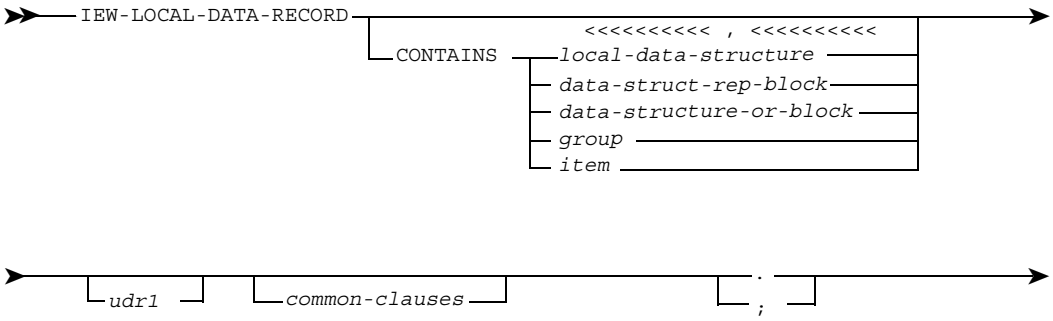


where:

module is an IEW-MODULE member name.

common-clauses are defined "Common Clauses Syntax" on page 152.

IEW-LOCAL-DATA-RECORD Syntax



where:

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

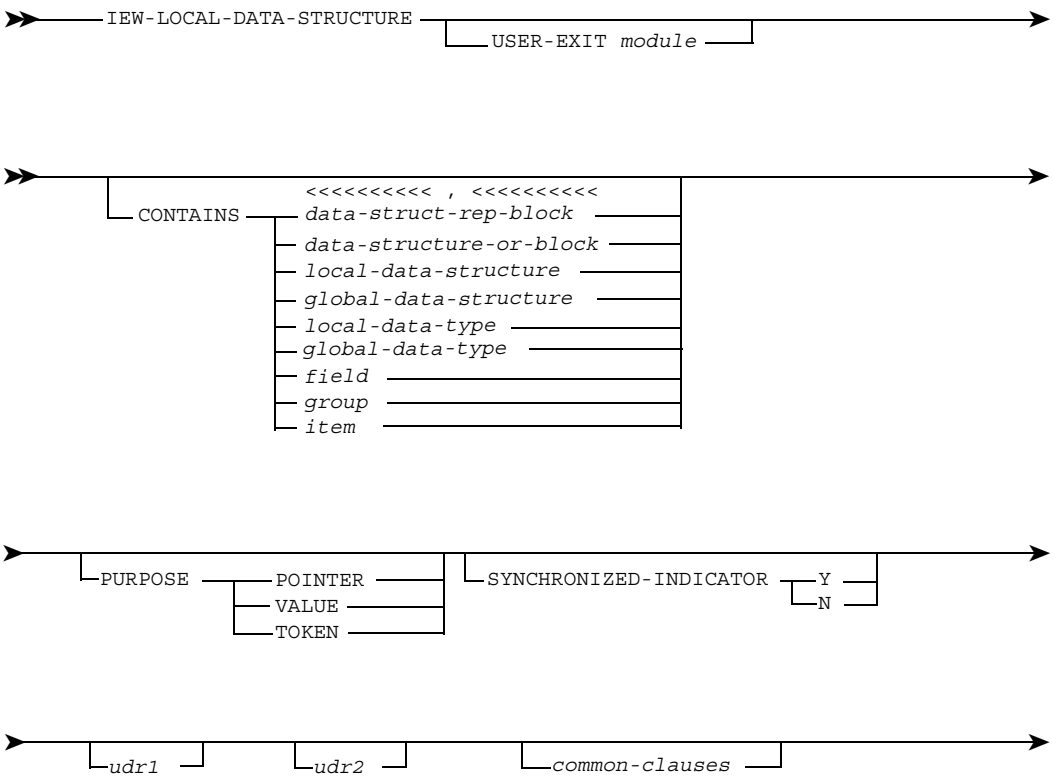
group is a GROUP member name.

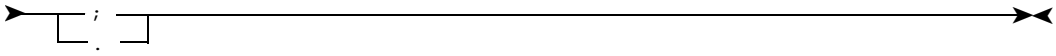
item is an item member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined "Common Clauses Syntax" on page 152.

IEW-LOCAL-DATA-STRUCTURE Syntax





where:

module is an IEW-MODULE member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-type is an IEW-GLOBAL-DATA-TYPE member name.

local-data-type is an IEW-LOCAL-DATA-TYPE member name.

field is a IEW-FIELD member name.

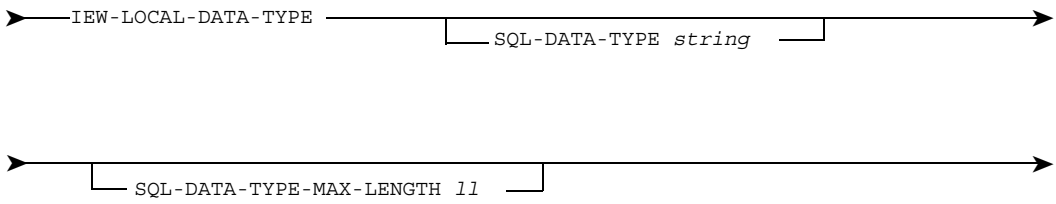
group is a GROUP member name.

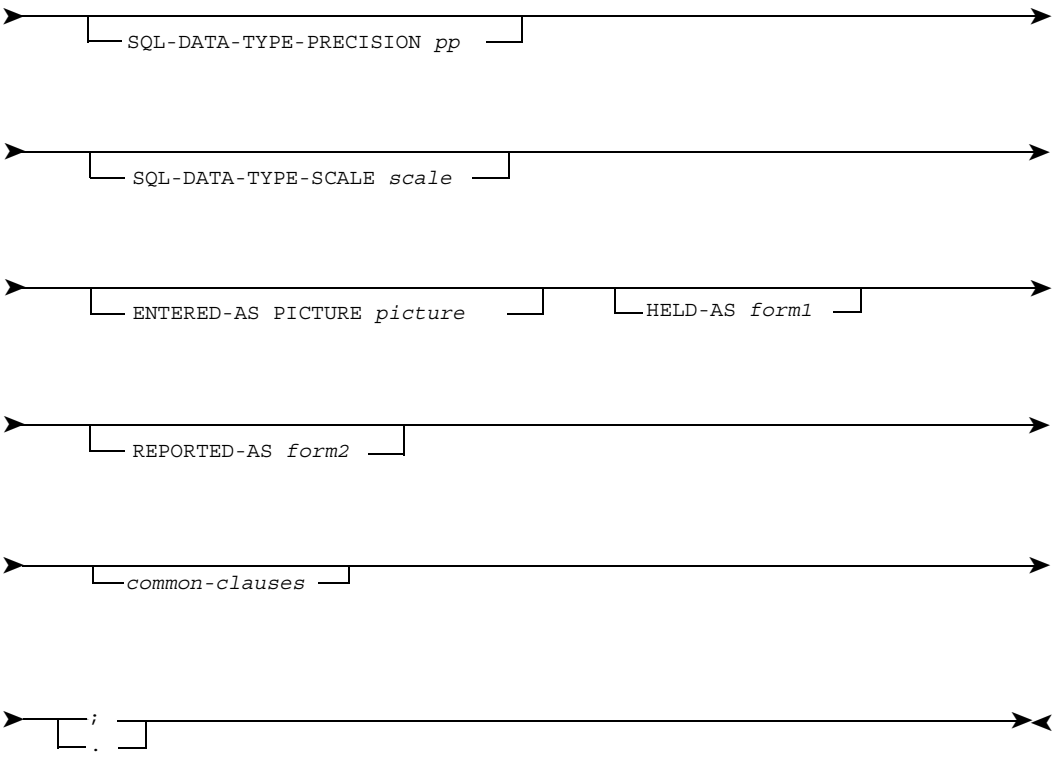
item is an ITEM member name.

udr1 and *udr2* are defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-LOCAL-DATA-TYPE Syntax





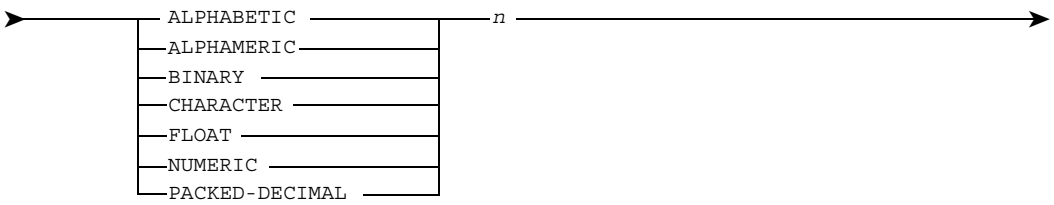
where:

string is data for import from, or export to, ADW/IEW.

ll, *pp*, and *scale* are unsigned integers.

picture is a valid data-type format.

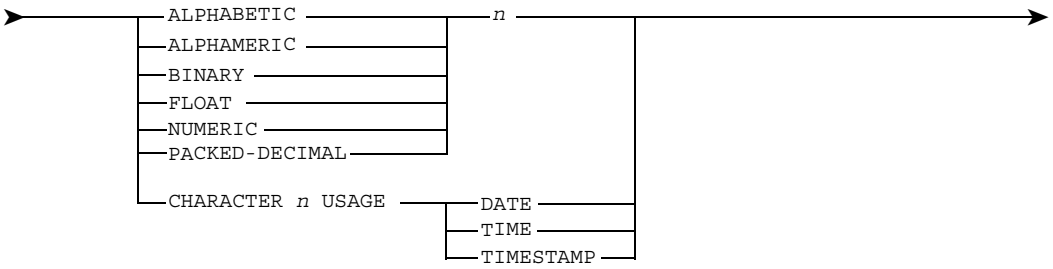
form1 is:



where:

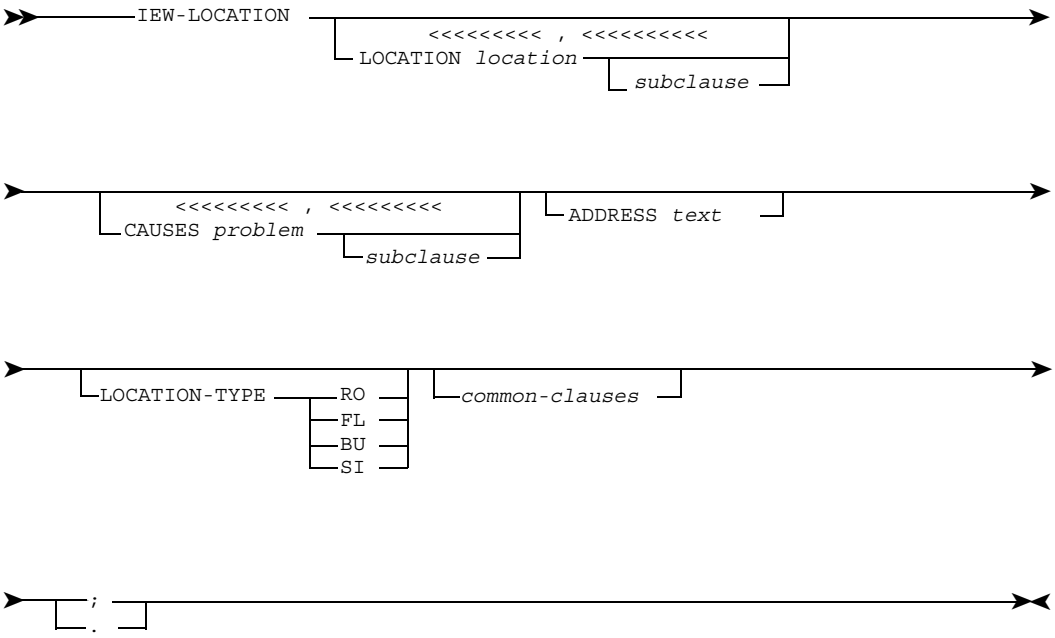
n is an unsigned decimal number specifying a maximum length.

form2 is:



common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-LOCATION Syntax



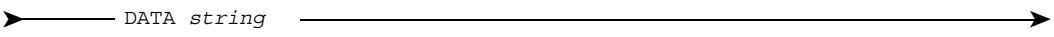
where:

location is an IEW-LOCATION member name.

problem is an IEW-PROBLEM member name.

text is up to 32,767 delimited strings, each string having a maximum length of 60 characters.

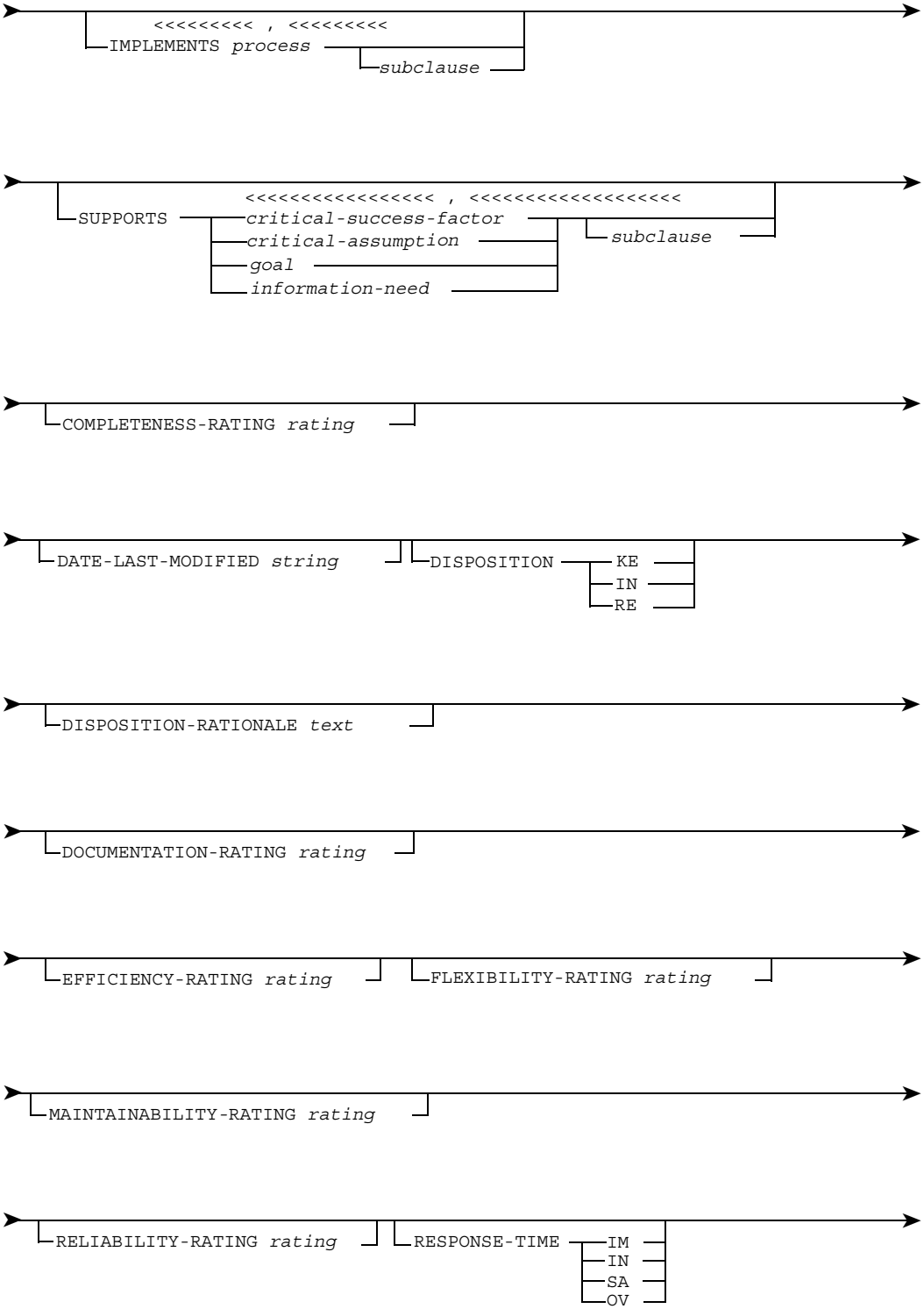
subclause is:

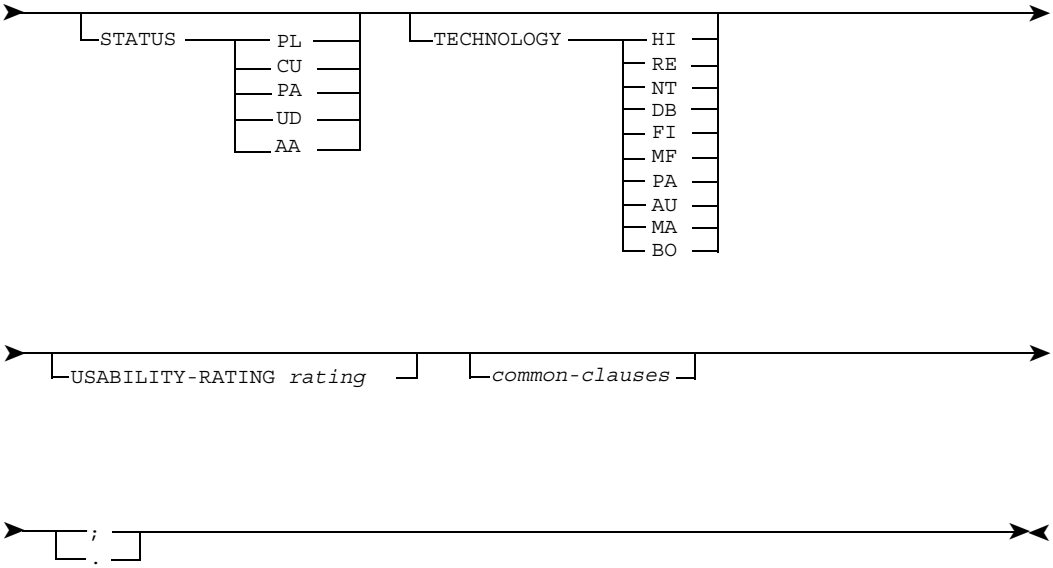


string is the data to be imported from, or exported to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-MECHANISM Syntax

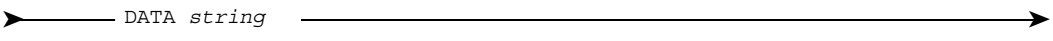

<<<<<<<<< , <<<<<<<<<
IMPLEMENTS process
subclause
<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<<
SUPPORTS
critical-success-factor
critical-assumption
goal
information-need
subclause
COMPLETENESS-RATING rating
DATE-LAST-MODIFIED string
DISPOSITION
KE
IN
RE
DISPOSITION-RATIONALE text
DOCUMENTATION-RATING rating
EFFICIENCY-RATING rating
FLEXIBILITY-RATING rating
MAINTAINABILITY-RATING rating
RELIABILITY-RATING rating
RESPONSE-TIME
IM
IN
SA
OV



where:

data-collection is an IEW-DATA-COLLECTION member name.

subclause is:



string is data to be imported from, or exported to, ADW/IEW.

location is an IEW-LOCATION member name.

mechanism is an IEW-MECHANISM member name.

problem is an IEW-PROBLEM member name.

process is an IEW-PROCESS member name.

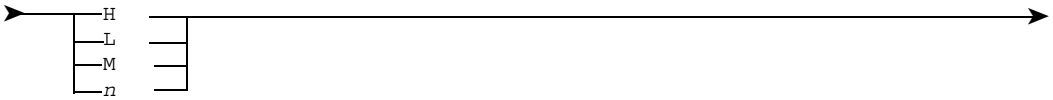
critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

goal is an IEW-GOAL member name.

information-need is an IEW-INFORMATION-NEED member name.

rating is:



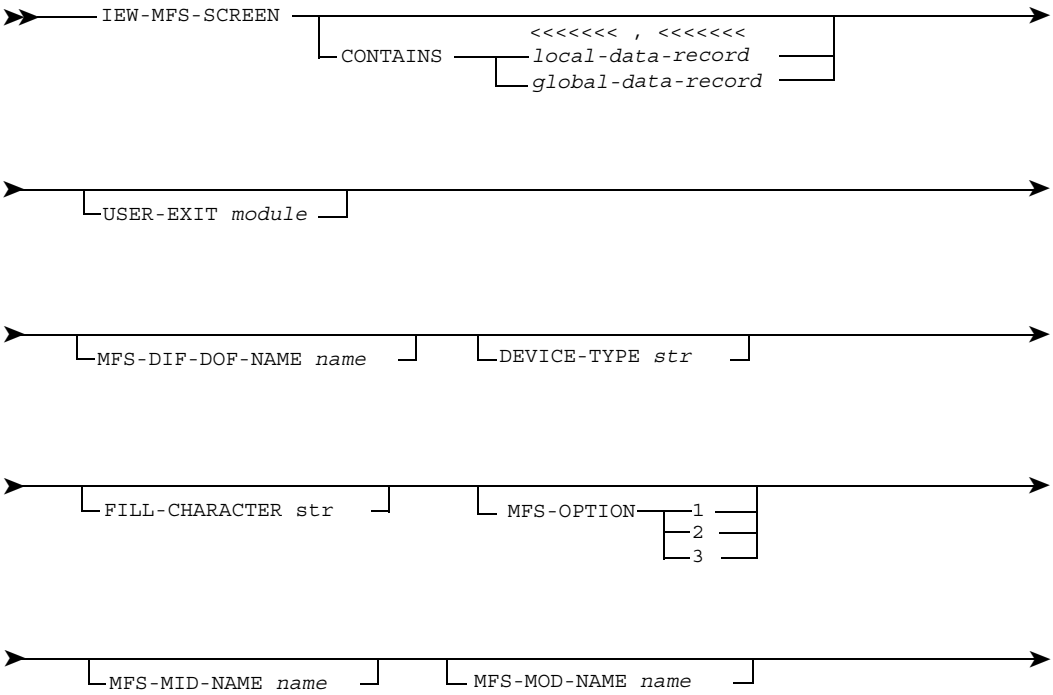
n is an integer in the range 1 to 999.

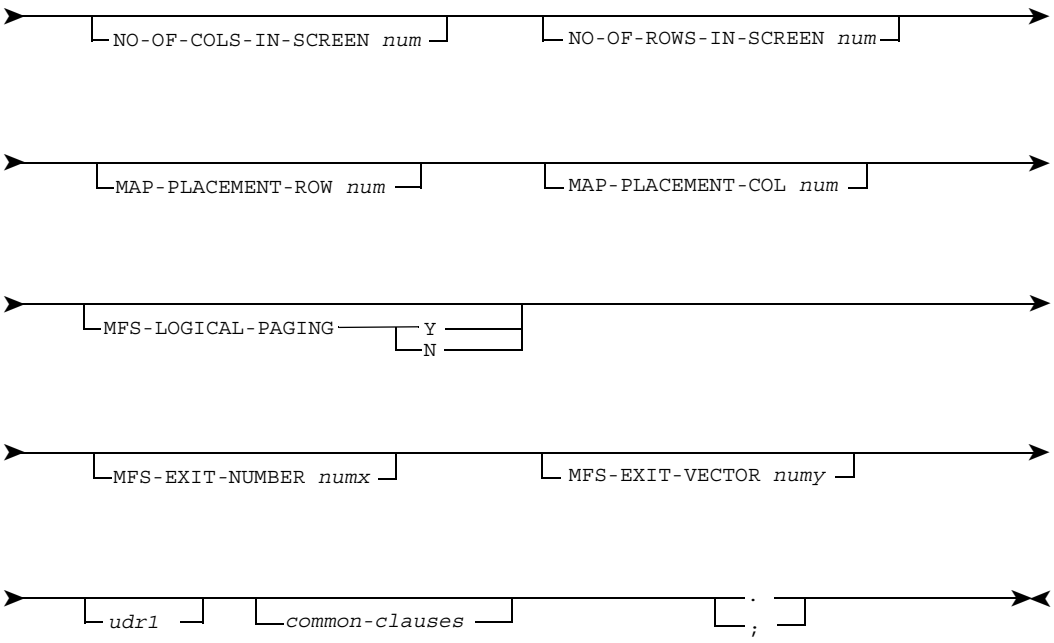
string is as defined above.

text is up to 32,767 delimited strings, each string having a maximum length of 60 characters.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-MFS-SCREEN Syntax





where:

local-data-record is an IEW-LOCAL-DATA-RECORD member name.

global-data-record is an IEW-GLOBAL-DATA-RECORD member name.

module is an IEW-MODULE member name.

num is a positive integer.

numx is a positive integer between 0 and 127.

numy is a positive integer between 0 and 255.

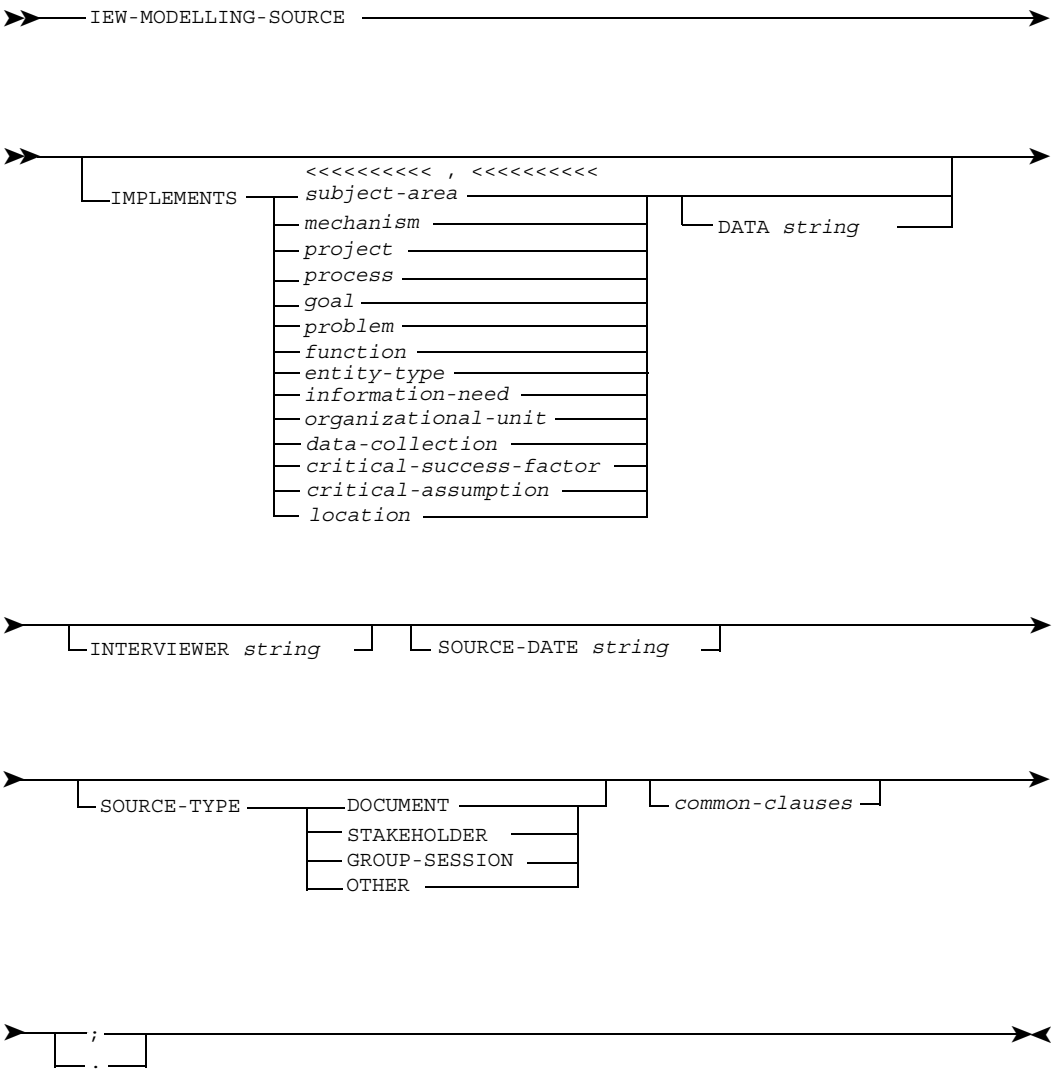
name is data to be imported from, or exported to, ADW/IEW.

string is data to be imported from, or exported to, ADW/IEW

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-MODELLING-SOURCE Syntax



where:

subject-area is an IEW-SUBJECT-AREA member name.

mechanism is an IEW-MECHANISM member name.

project is an IEW-PROJECT member name.

process is an IEW-PROCESS member name.

goal is an IEW-GOAL member name.

problem is an IEW-PROBLEM member name.

function is an IEW-FUNCTION member name.

entity-type is an IEW-ENTITY-TYPE member name.

information-need is an IEW-INFORMATION-NEED member name.

organizational-unit is an IEW-ORGANIZATIONAL-UNIT member name.

data-collection is an IEW-DATA-COLLECTION member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member.

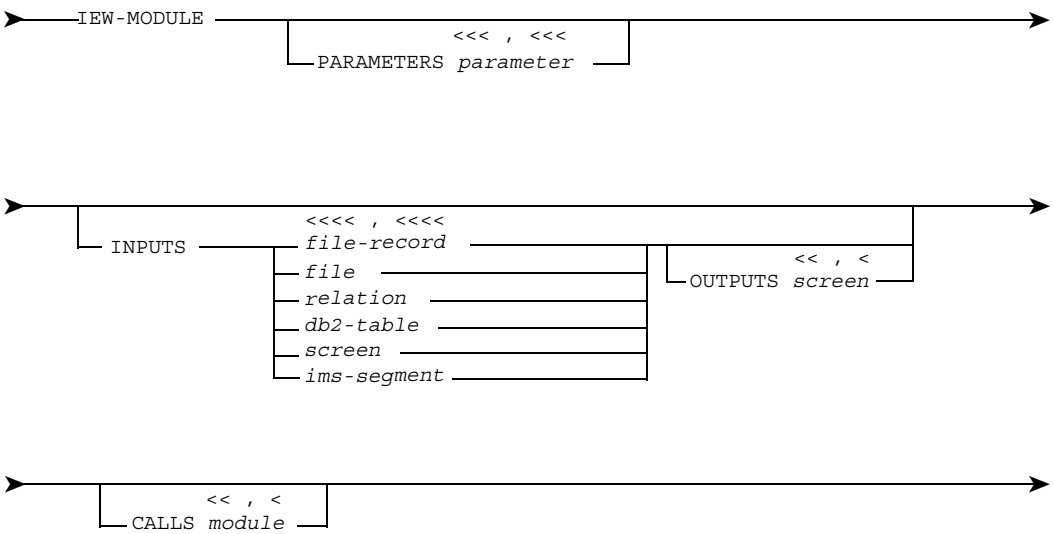
critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

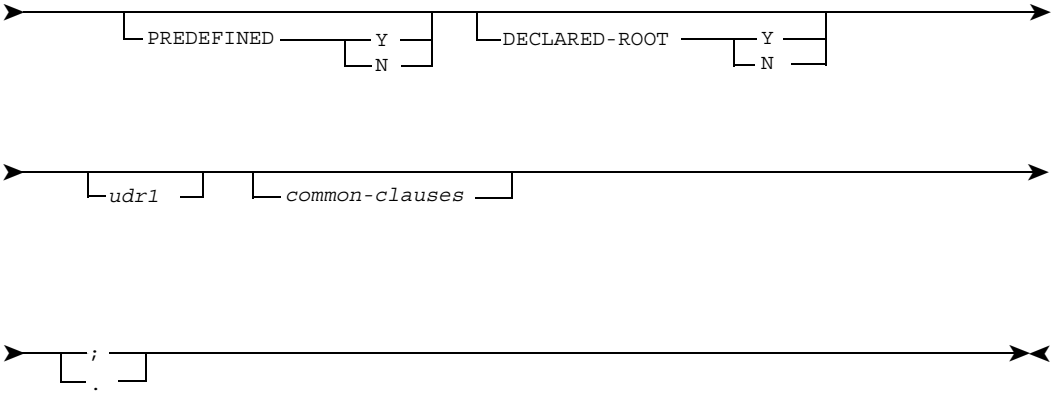
location is an IEW-LOCATION member name.

string is the data to be imported from, or exported to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-MODULE Syntax





where:

parameter is an IEW-PARAMETER member name.

file-record is an IEW-FILE-RECORD member name.

file is a FILE member name.

relation is an IEW-RELATION member name.

db2-table is a DB2-TABLE member name.

screen is an IEW-SCREEN member name.

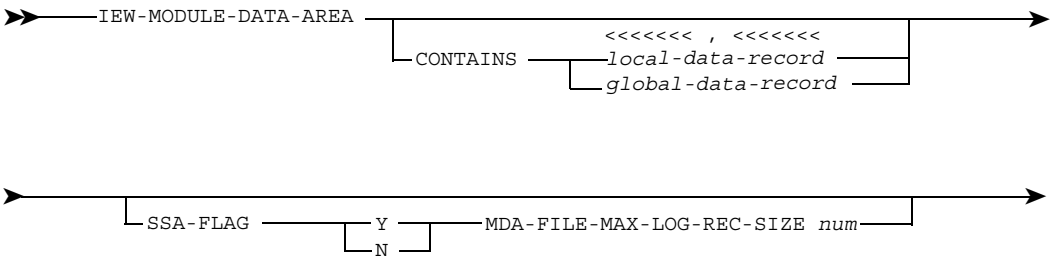
ims-segment is a SEGMENT member name.

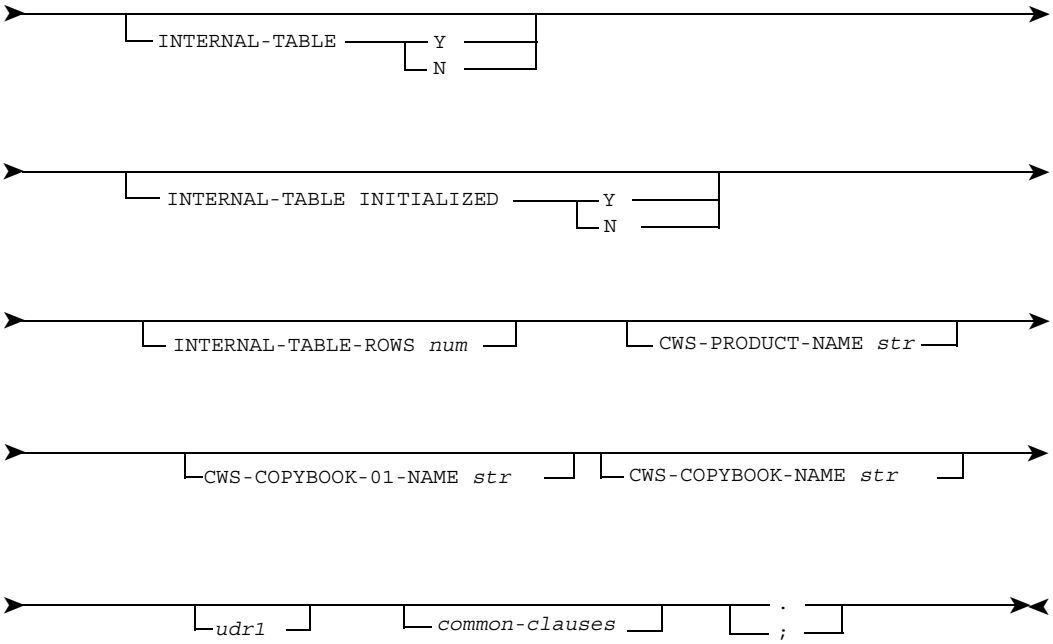
module is an IEW-MODULE member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-MODULE-DATA-AREA Syntax





where:

local-data-record is an IEW-LOCAL-DATA-RECORD member name.

global-data-record is an IEW-GLOBAL-DATA-RECORD member name.

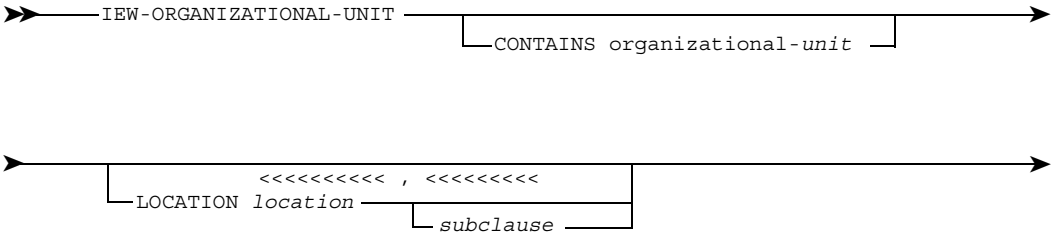
num is a positive integer.

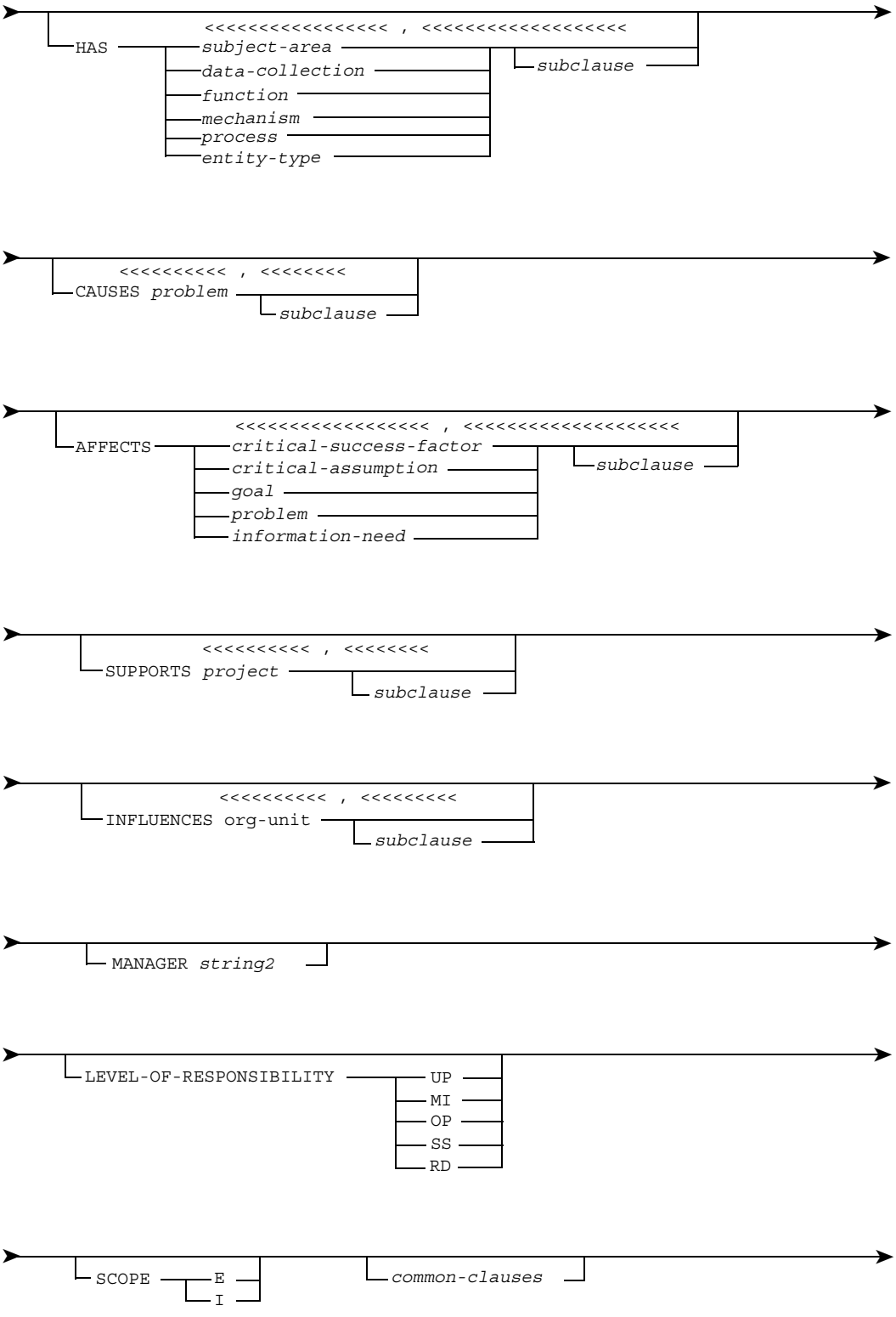
str is data to be imported from, or exported to, ADW/IEW.

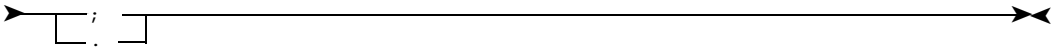
udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-ORGANIZATIONAL-UNIT Syntax




HAS
<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<<
subject-area
data-collection
function
mechanism
process
entity-type
subclause
CAUSES problem
<<<<<<<<<< , <<<<<<<<
subclause
AFFECTS
<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<<
critical-success-factor
critical-assumption
goal
problem
information-need
subclause
SUPPORTS project
<<<<<<<<<< , <<<<<<<<
subclause
INFLUENCES org-unit
<<<<<<<<<< , <<<<<<<<<
subclause
MANAGER string2
LEVEL-OF-RESPONSIBILITY
UP
MI
OP
SS
RD
SCOPE
E
I
common-clauses

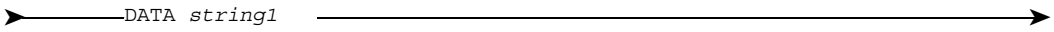


where:

organizational-unit is an IEW-ORGANIZATIONAL-UNIT member name.

location is an IEW-LOCATION member name.

subclause is:



where:

string1 is the data to be imported from, or exported to, ADW/IEW.

subject-area is an IEW-SUBJECT-AREA member name.

data-collection is an IEW-DATA-COLLECTION member name.

function is an IEW-FUNCTION member name.

mechanism is an IEW-MECHANISM member name.

process is an IEW-PROCESS member name.

entity-type is an IEW-ENTITY-TYPE member name.

problem is an IEW-PROBLEM member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

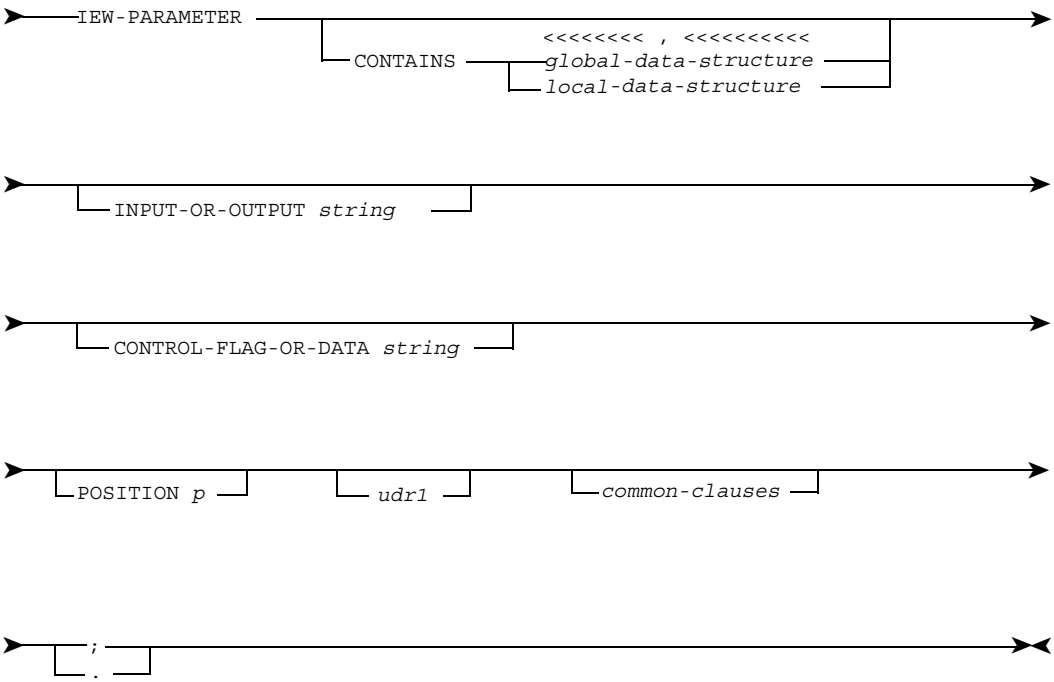
goal is an IEW-GOAL member name.

information-need is an IEW-INFORMATION-NEED member name.

string2 is a character string, with a maximum length of 32 characters.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PARAMETER Syntax



where:

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

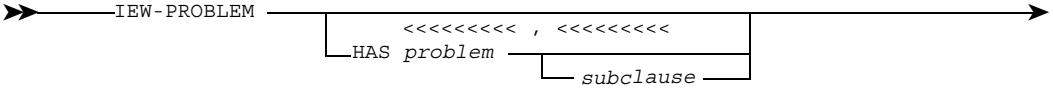
string is the data to be imported from, or exported to, ADW/IEW.

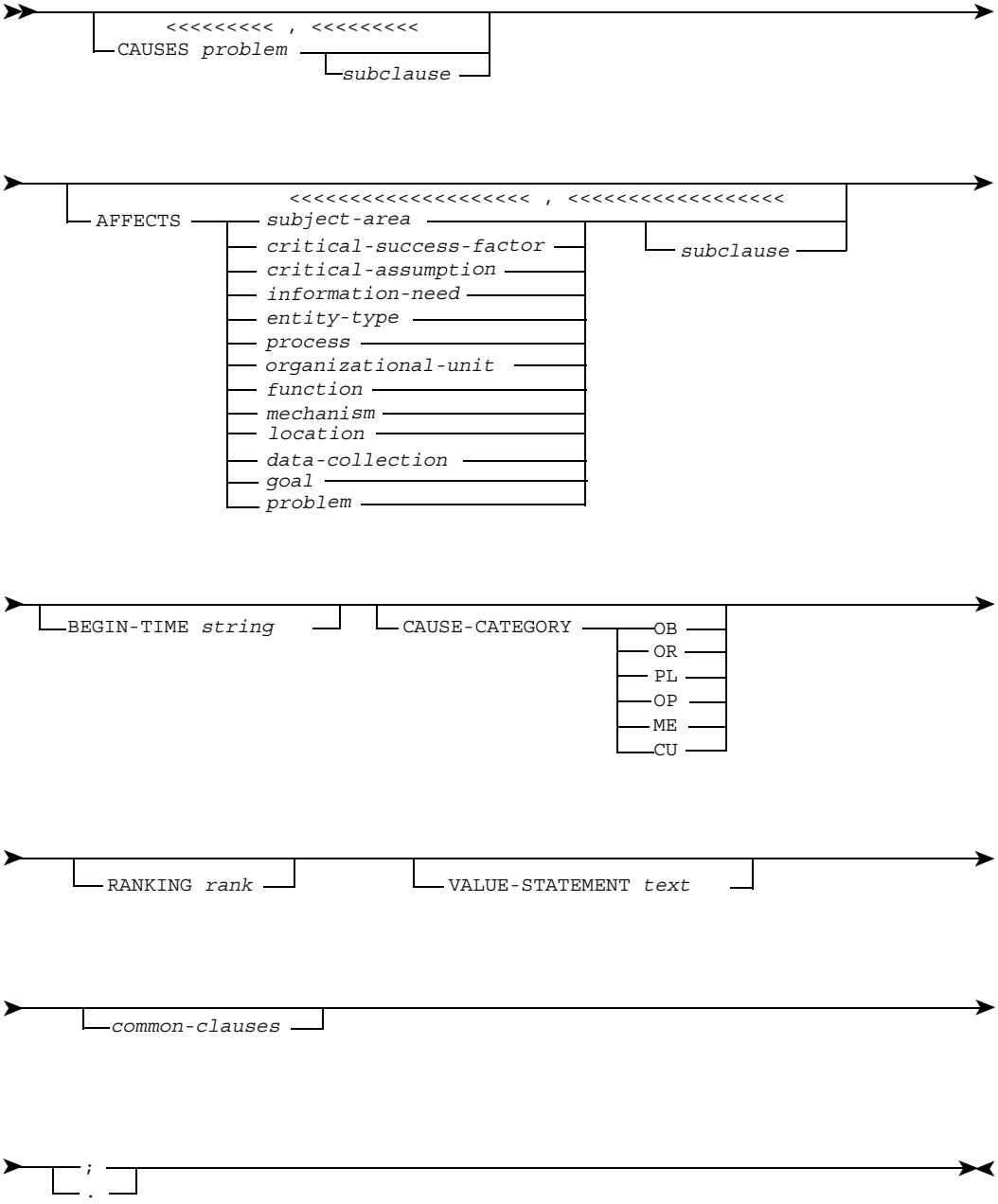
p is an integer.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PROBLEM Syntax

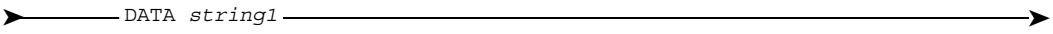



<<<<<<<<< , <<<<<<<<<
CAUSES problem
subclause
<<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<
AFFECTS
subject-area
critical-success-factor
critical-assumption
information-need
entity-type
process
organizational-unit
function
mechanism
location
data-collection
goal
problem
subclause
BEGIN-TIME string
CAUSE-CATEGORY
OB
OR
PL
OP
ME
CU
RANKING rank
VALUE-STATEMENT text
common-clauses
;
.

where:

problem is an IEW-PROBLEM member name.

subclause is:



where:

string is the data to be imported from, or exported to, ADW/IEW.

subject-area is an IEW-SUBJECT-AREA member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

information-need is an IEW-INFORMATION-NEED member name.

entity-type is an IEW-ENTITY-TYPE member name.

process is an IEW-PROCESS member name.

organizational-unit is an IEW-ORGANIZATIONAL-UNIT member name.

function is an IEW-FUNCTION member name.

mechanism is an IEW-MECHANISM member name.

location is an IEW-LOCATION member name.

data-collection is an IEW-DATA-COLLECTION member name.

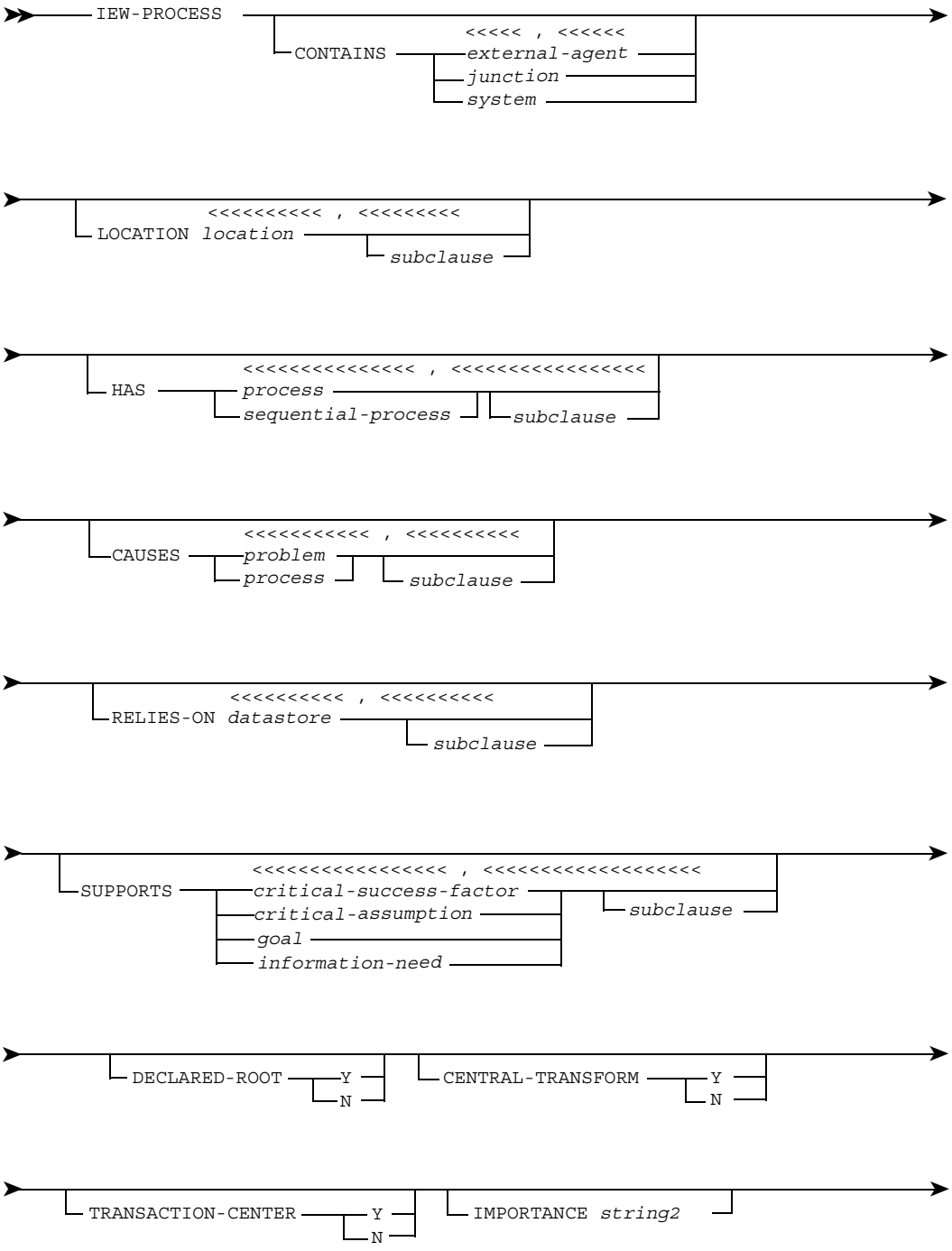
goal is an IEW-GOAL member name.

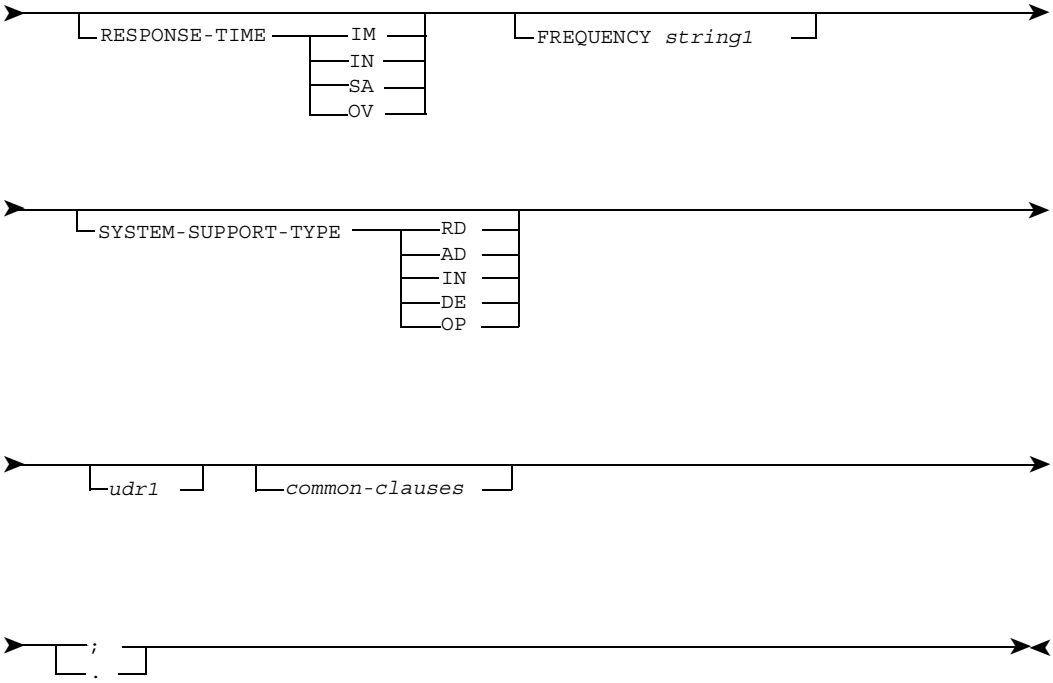
rank is an unsigned numeric value in the range 1 to 9,999.

text is up to 32,767 delimited strings, each string having a maximum length of 60 characters.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PROCESS Syntax

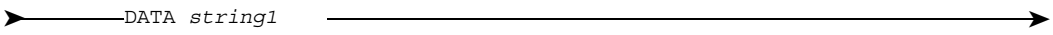




where:

external-agent is an IEW-EXTERNAL-AGENT member name.

subclause is:



string1 is the data for import from, or export to, ADW/IEW.

junction is an IEW-JUNCTION member name.

system is a SYSTEM member name.

location is an IEW-LOCATION member name.

process is an IEW-PROCESS member name.

sequential-process is an IEW-SEQUENTIAL-PROCESS member name.

problem is an IEW-PROBLEM member name.

datastore is an IEW-DATASTORE member name.

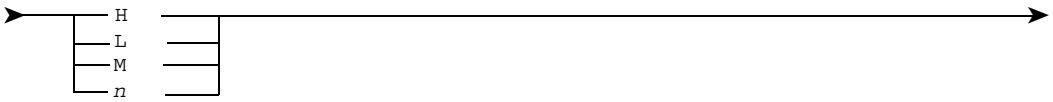
critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

goal is an IEW-GOAL member name.

information-need is an IEW-INFORMATION-NEED member name.

string2 is:

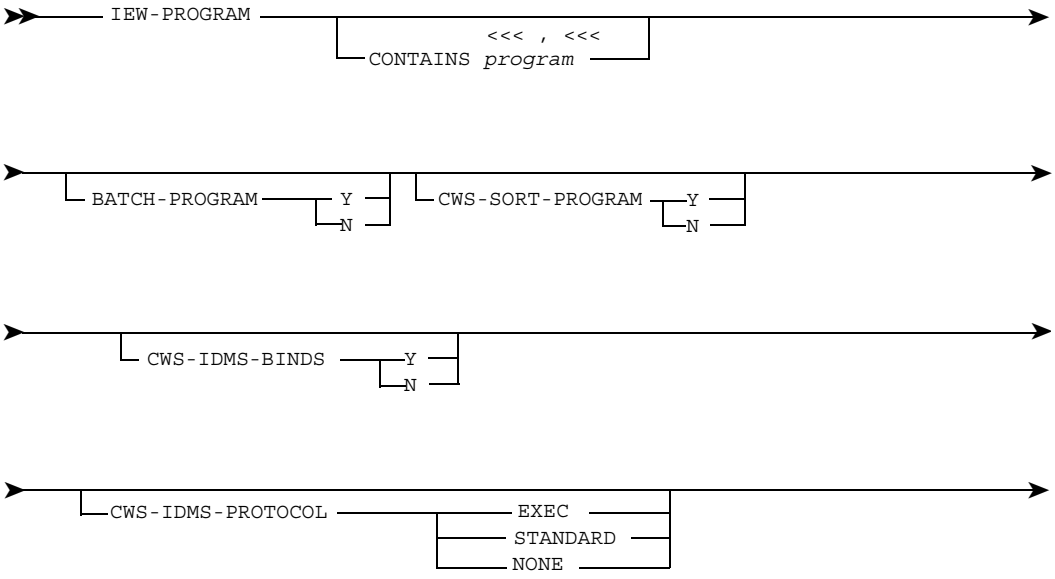


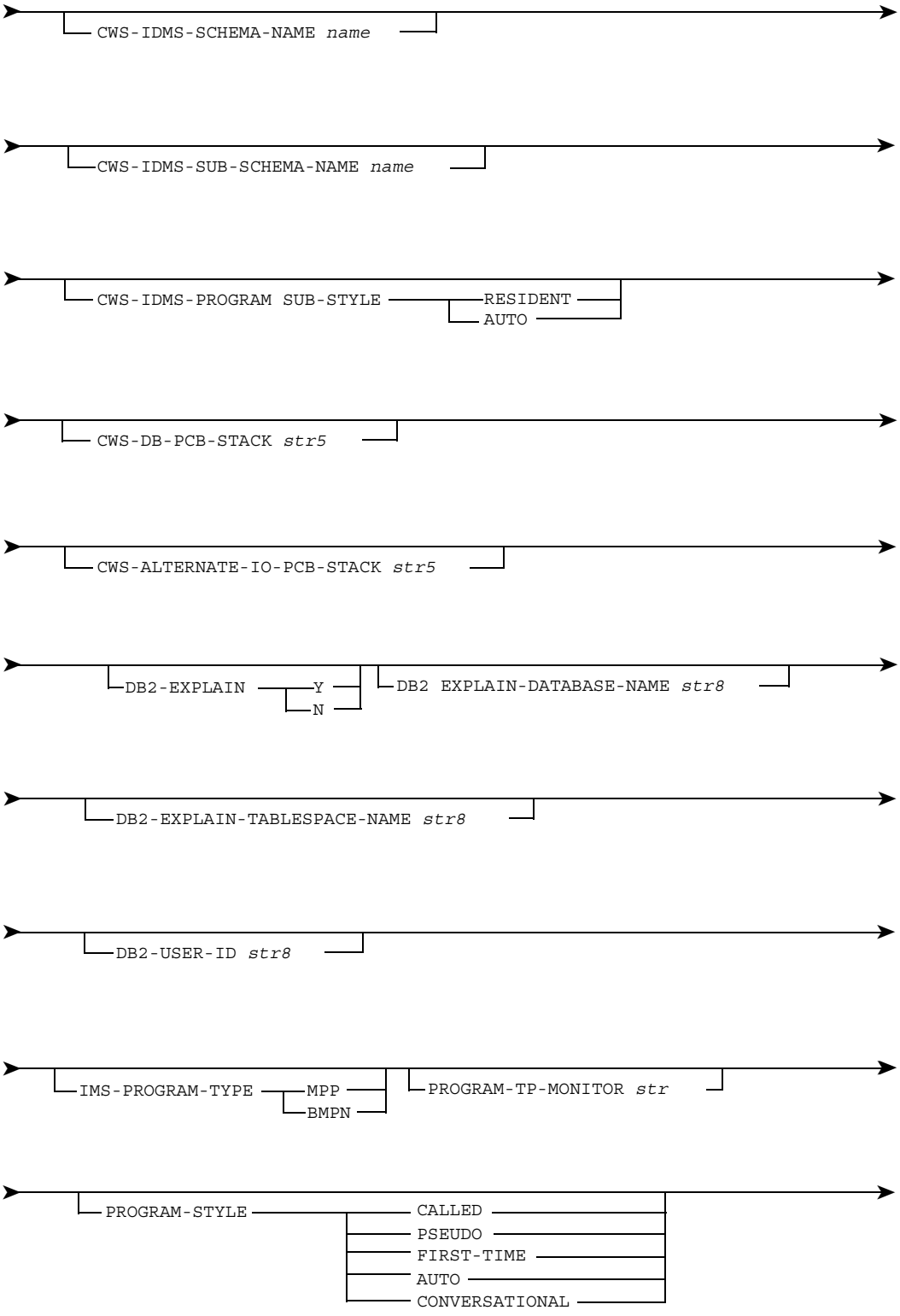
n is an integer in the range 1 to 999.

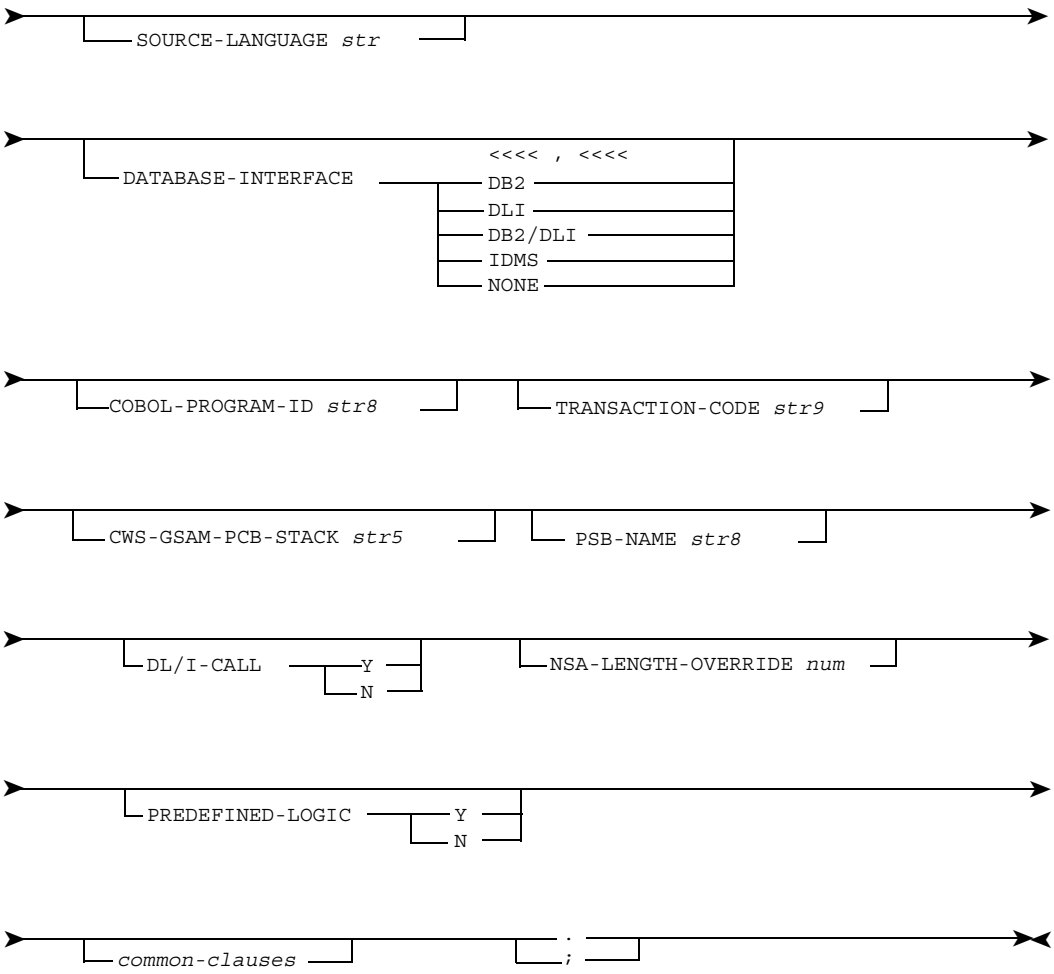
udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PROGRAM Syntax




CWS-IDMS-SCHEMA-NAME name
CWS-IDMS-SUB-SCHEMA-NAME name
CWS-IDMS-PROGRAM SUB-STYLE
RESIDENT
AUTO
CWS-DB-PCB-STACK str5
CWS-ALTERNATE-IO-PCB-STACK str5
DB2-EXPLAIN
Y
N
DB2 EXPLAIN-DATABASE-NAME str8
DB2-EXPLAIN-TABLESPACE-NAME str8
DB2-USER-ID str8
IMS-PROGRAM-TYPE
MPP
BMPN
PROGRAM-TP-MONITOR str
PROGRAM-STYLE
CALLED
PSEUDO
FIRST-TIME
AUTO
CONVERSATIONAL



where:

program is an IEW-PROJECT member name.

num is a positive integer.

name is data to be imported from, or exported to, ADW/IEW.

str is data to be imported from, or exported to, ADW/IEW.

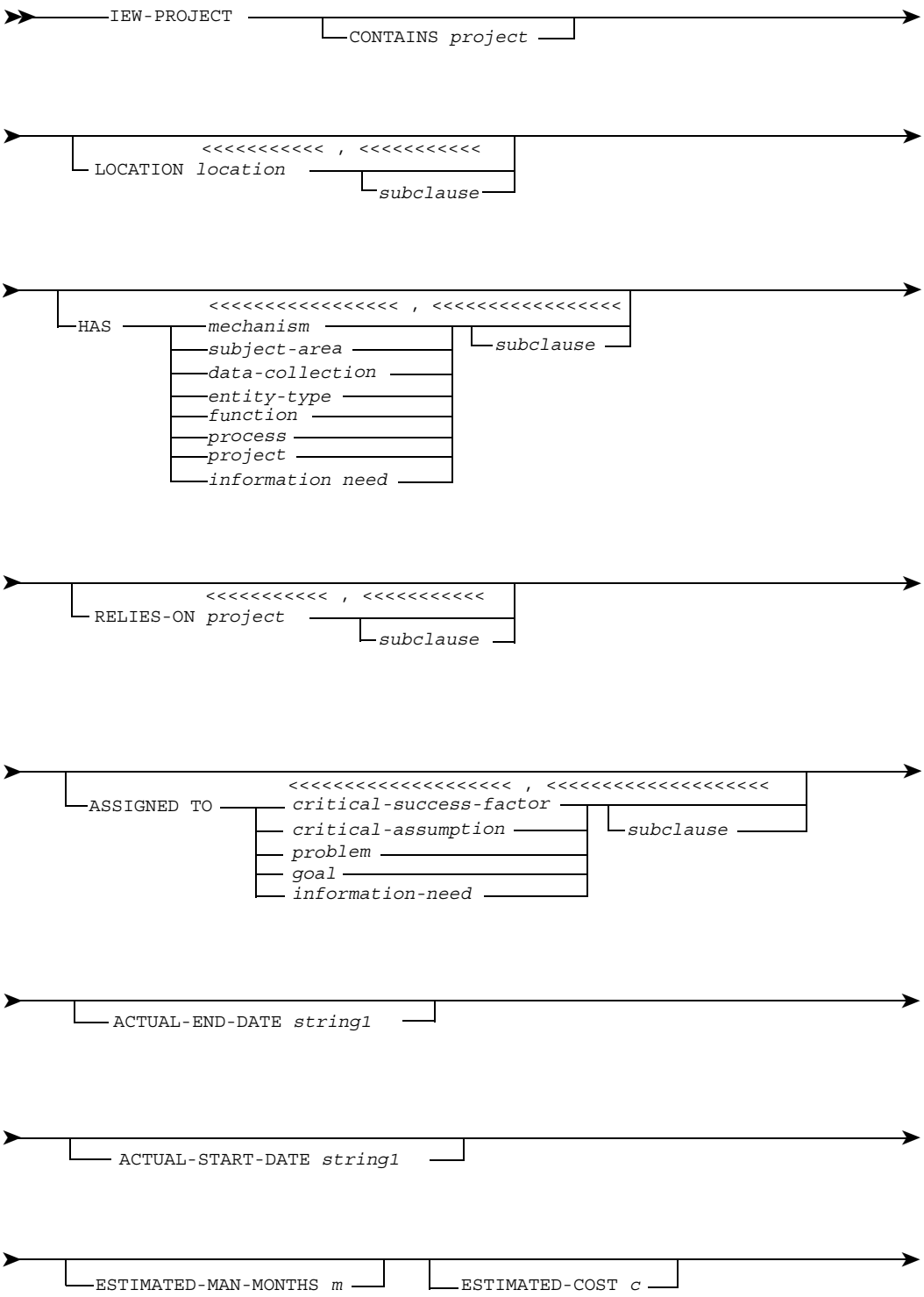
str5 is a 5-character string.

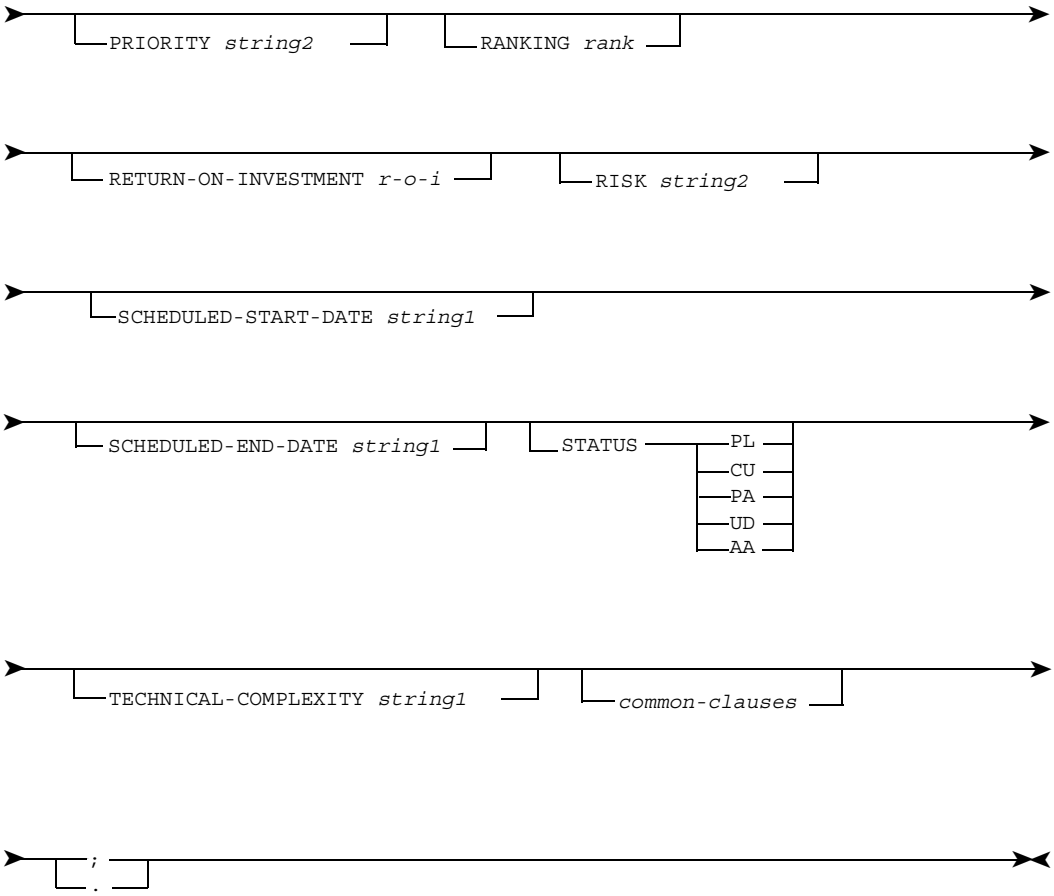
str8 is a 8-character string.

str9 is a 9-character string.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PROJECT Syntax



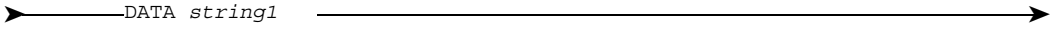


where:

project is an IEW/PROJECT member name.

location is an IEW-LOCATION member name.

subclause is:



string1 is the data for import from, or export to, ADW/IEW.

mechanism is an IEW-MECHANISM member name.

subject-area is an IEW-SUBJECT-AREA member name.

data-collection is an IEW-DATA-COLLECTION member name.

entity-type is an IEW-ENTITY-TYPE member name.

function is an IEW-FUNCTION member name.

process is an IEW-PROCESS member name.

project is an IEW-PROJECT member name.

problem is an IEW-PROBLEM member name.

information-need is an IEW-INFORMATION-NEED member name.

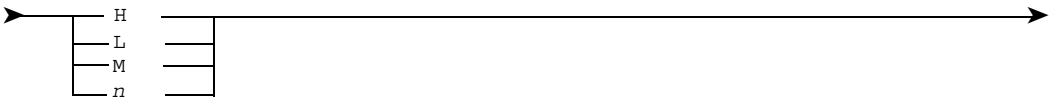
critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

goal is an IEW-GOAL member name.

m is an unsigned integer.

string2 is:



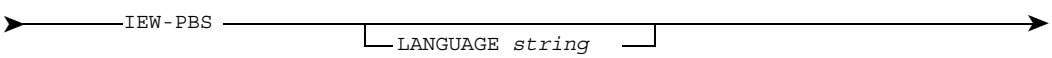
n is an integer in the range 1 to 999.

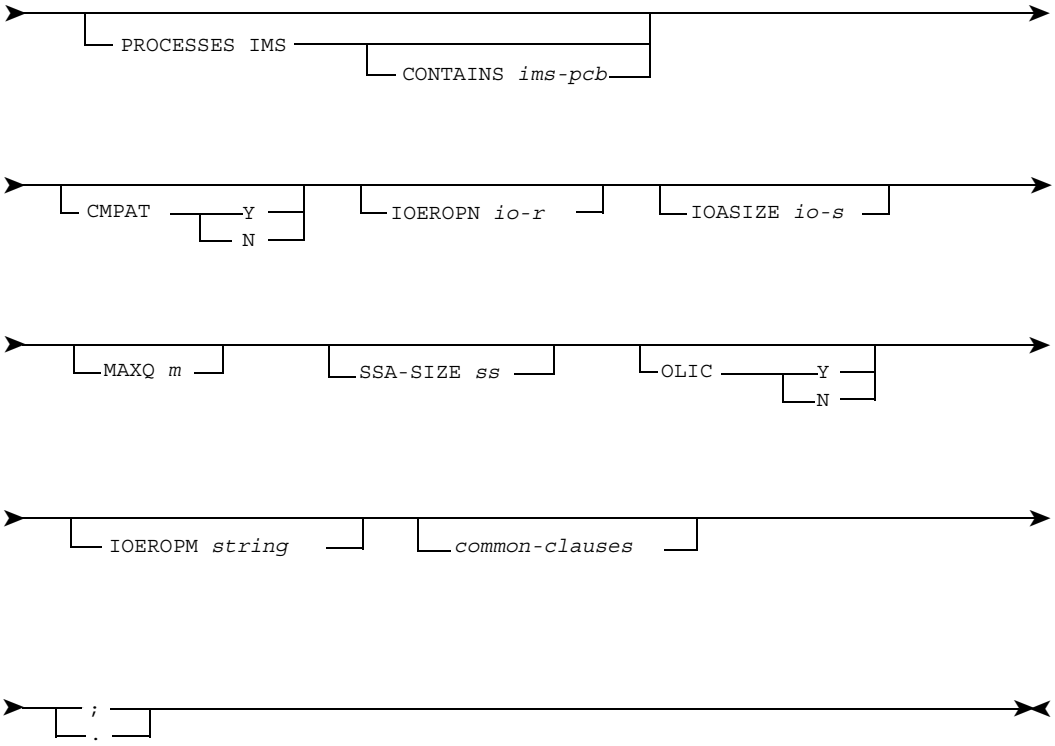
rank is an unsigned integer in the range 1 to 9,999.

r-o-i and c are integers in the range 1 to 99,999,999.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-PSB Syntax





where:

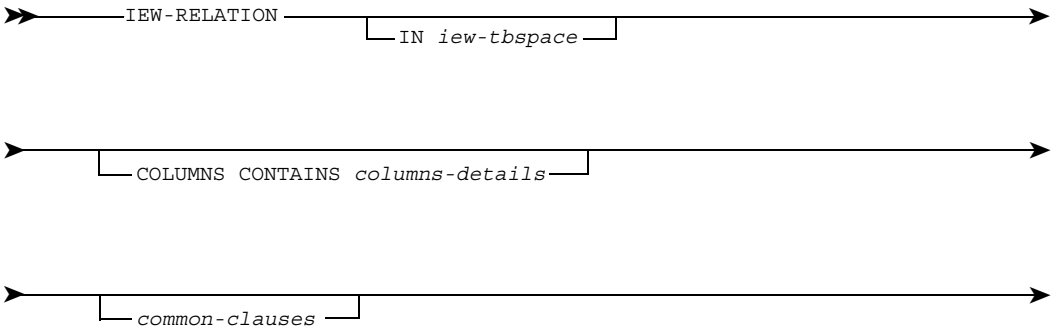
string is data for import from, or export to, ADW/IEW.

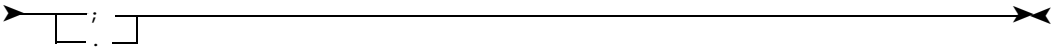
ims-pcb is a PCB member name.

io-r, *io-s*, *m*, and ss are unsigned integers.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-RELATION Syntax

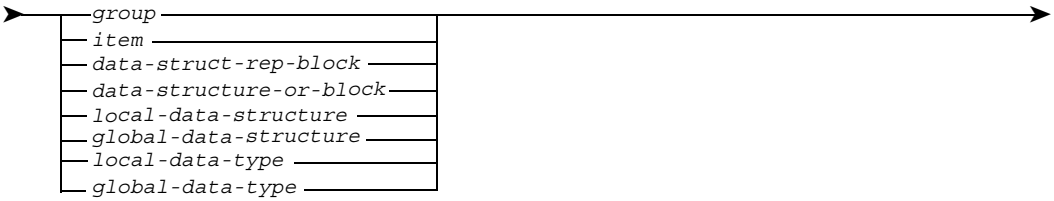


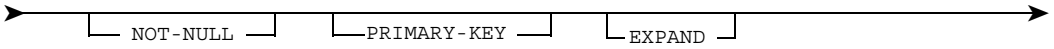


where:

iew-tbspace is an IEW-TBSPACE member name.

columns-details are:





where:

group is a GROUP member name.

item is an ITEM member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-STRUCTURE-OR-BLOCK member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

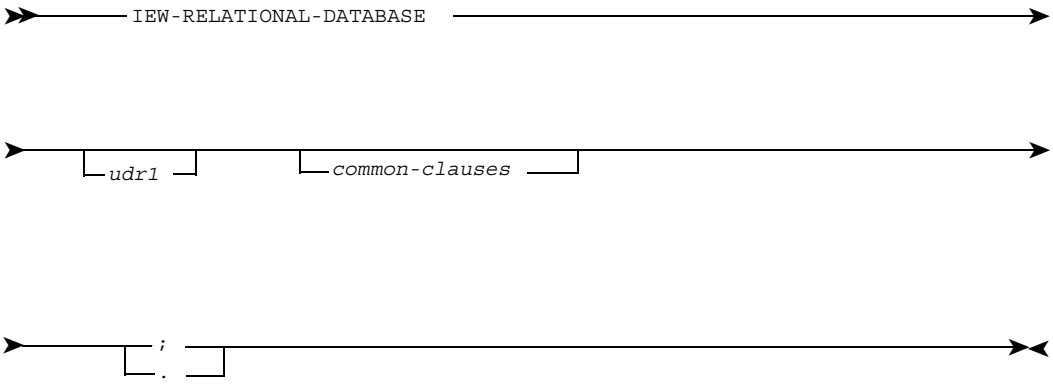
local-data-type is an IEW-LOCAL-DATA-TYPE member name.

global-data-type is an IEW-GLOBAL-DATA-TYPE member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

The EXPAND keyword is present by default but you can exclude it by tailoring Corporate Executive Routine MPDYIITAB0.

IEW-RELATIONAL-DATABASE Syntax

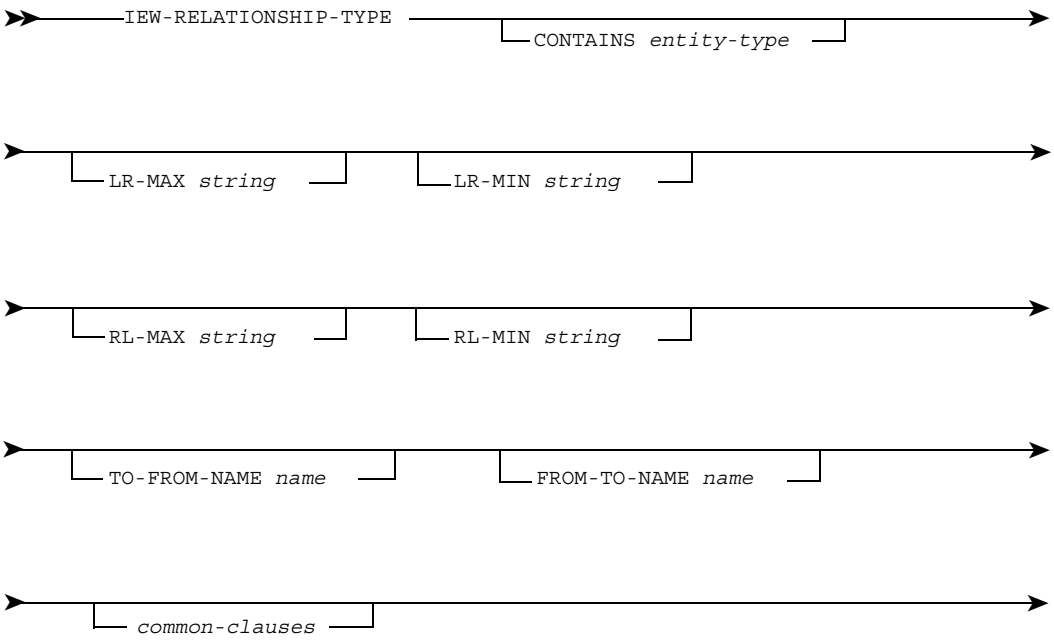


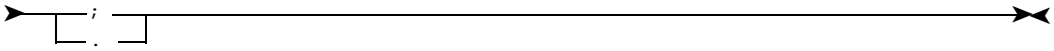
where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-RELATIONSHIP-TYPE Syntax





where:

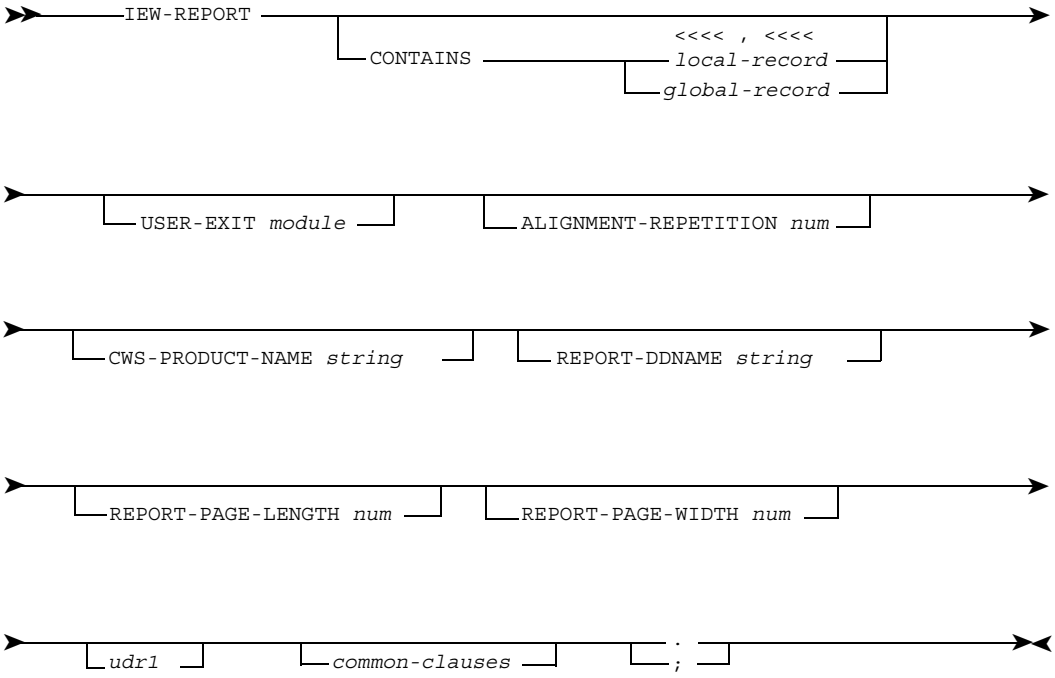
entity-type is an IEW-ENTITY-TYPE member name.

string is an integer or M (for many).

name is the data for import from, or export to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-REPORT Syntax



where:

local-record is an IEW-LOCAL-DATA-RECORD member name.

global-record is an IEW-GLOBAL-DATA-RECORD member name.

module is an IEW-MODULE member name.

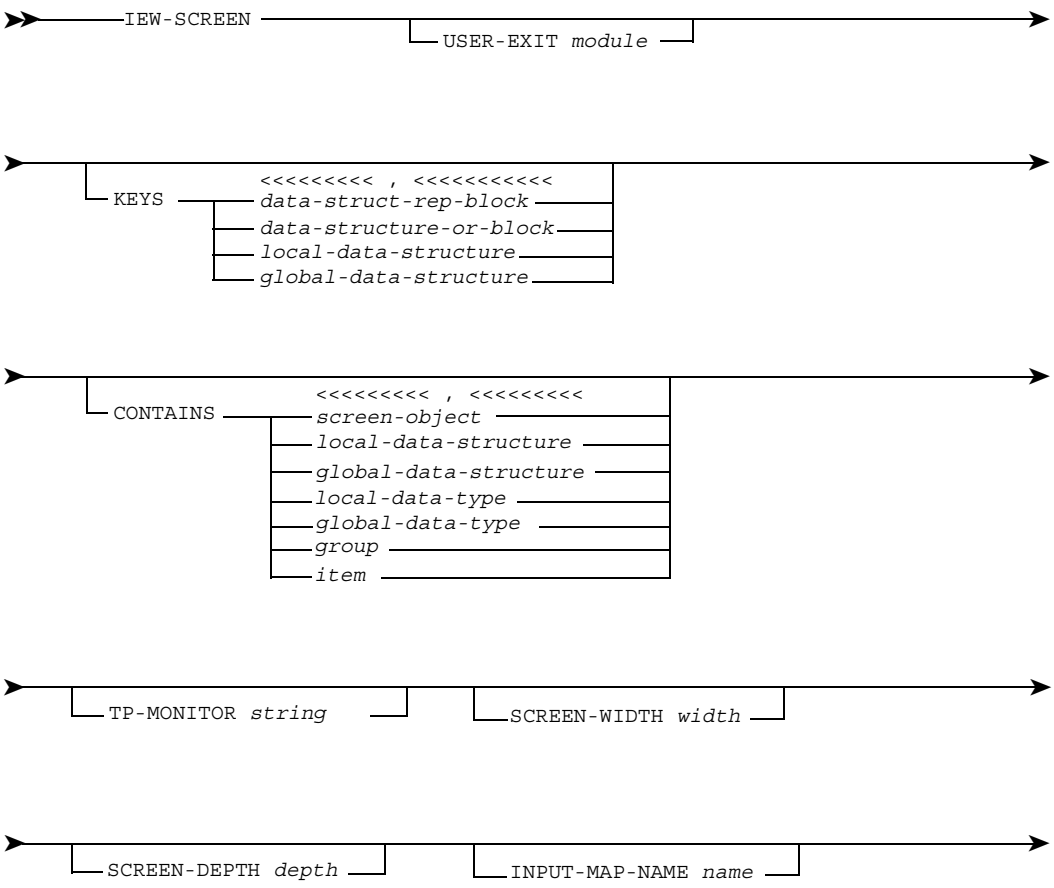
num is a positive integer.

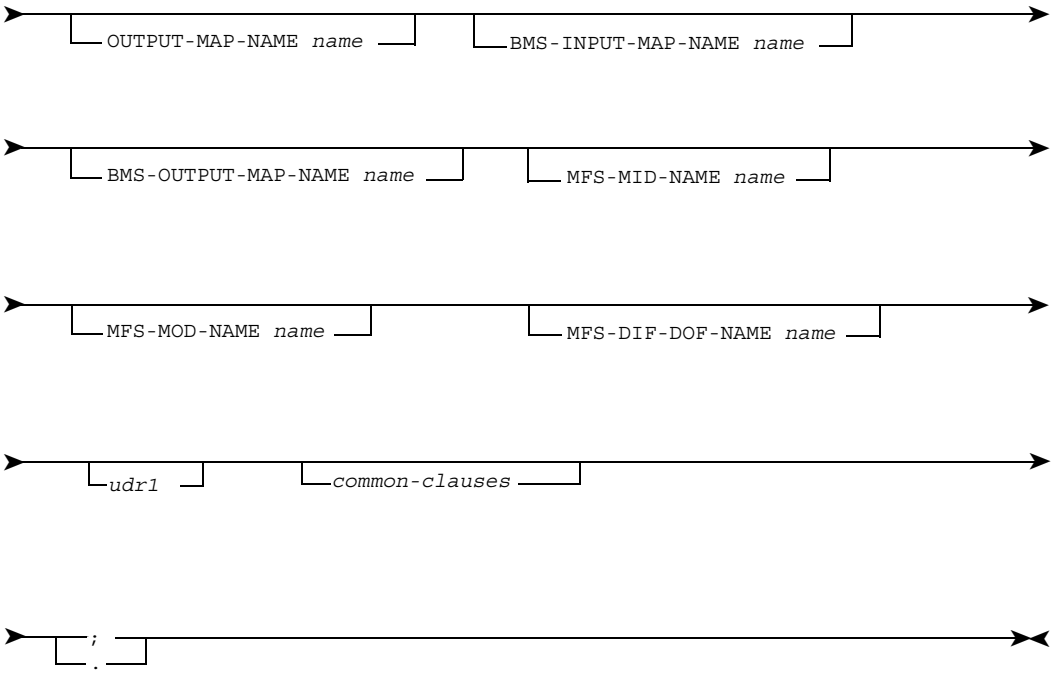
string is data to be imported from, or exported to, ADW/IEW.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SCREEN Syntax





where:

module is an IEW-MODULE member name.

global-data-structure is an IEW-GLOBAL-DATA-STRUCTURE member name.

local-data-structure is an IEW-LOCAL-DATA-STRUCTURE member name.

data-struct-rep-block is an IEW-DATA-STRUCT-REP-BLOCK member name.

data-structure-or-block is an IEW-DATA-STRUCTURE-OR-BLOCK member name.

screen-object is an IEW-SCREEN-OBJECT member name.

global-data-type is an IEW-GLOBAL-DATA-TYPE member name.

local-data-type is an IEW-LOCAL-DATA-TYPE member name.

group is a GROUP member name.

item is an ITEM member name.

string is either IMS/MFS or CICS/BMS.

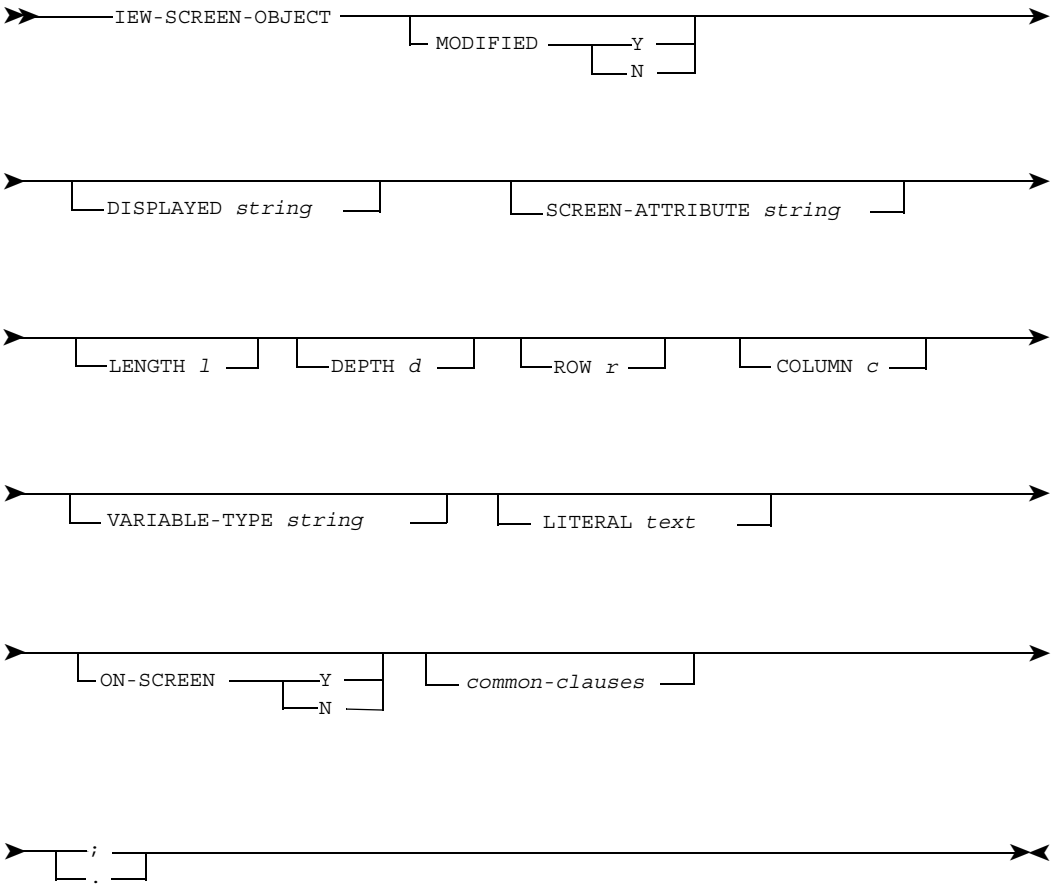
width and *depth* are unsigned integers.

name is the name of a repository member.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SCREEN-OBJECT Syntax



where:

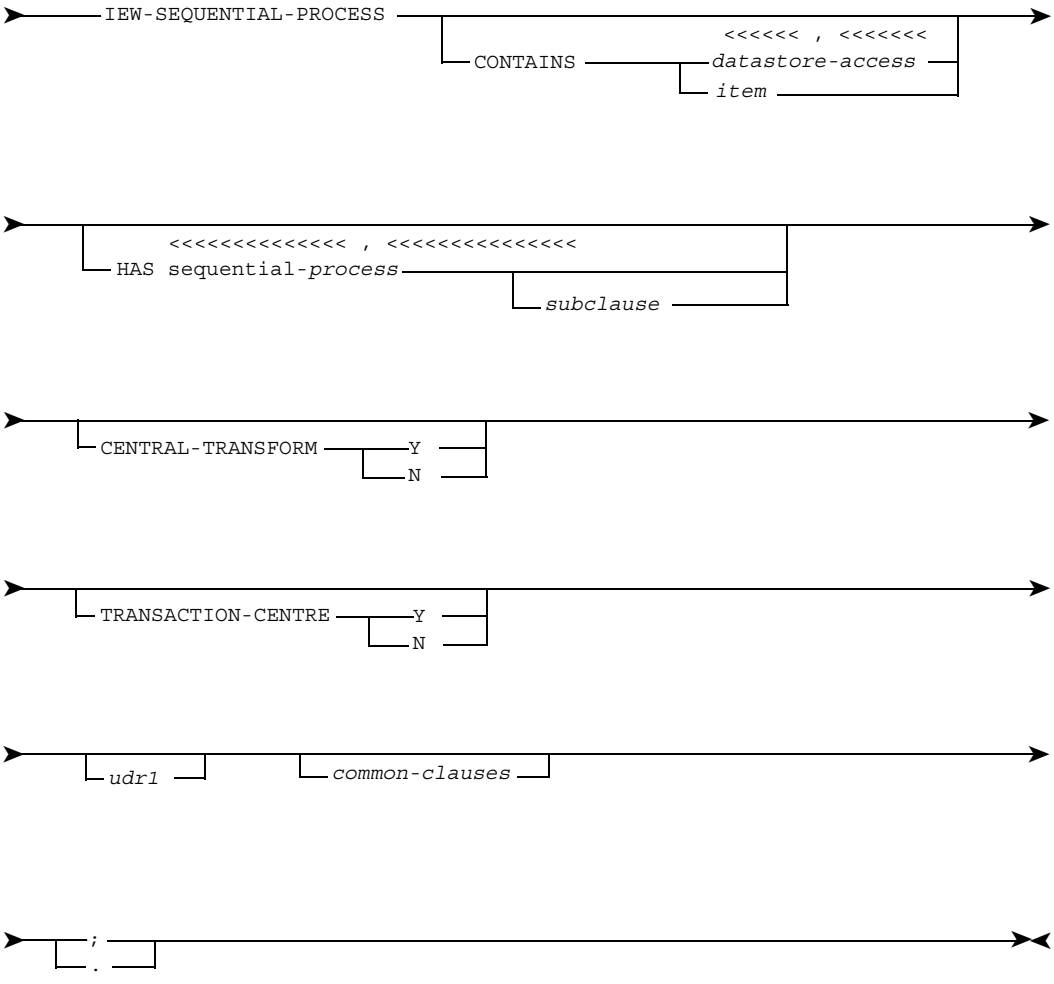
string is the data for import from, or export to, ADW/IEW.

d, *r*, *c*, *v*, and *l* are unsigned integers.

text is up to *d* delimited strings, each string having a maximum length of l (one) character.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SEQUENTIAL-PROCESS Syntax



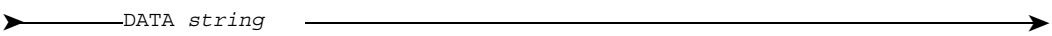
where:

datastore-access is an IEW-DATASTORE-ACCESS member name.

item is an ITEM member name.

sequential-process is an IEW-SEQUENTIAL-PROCESS member name.

subclause is:

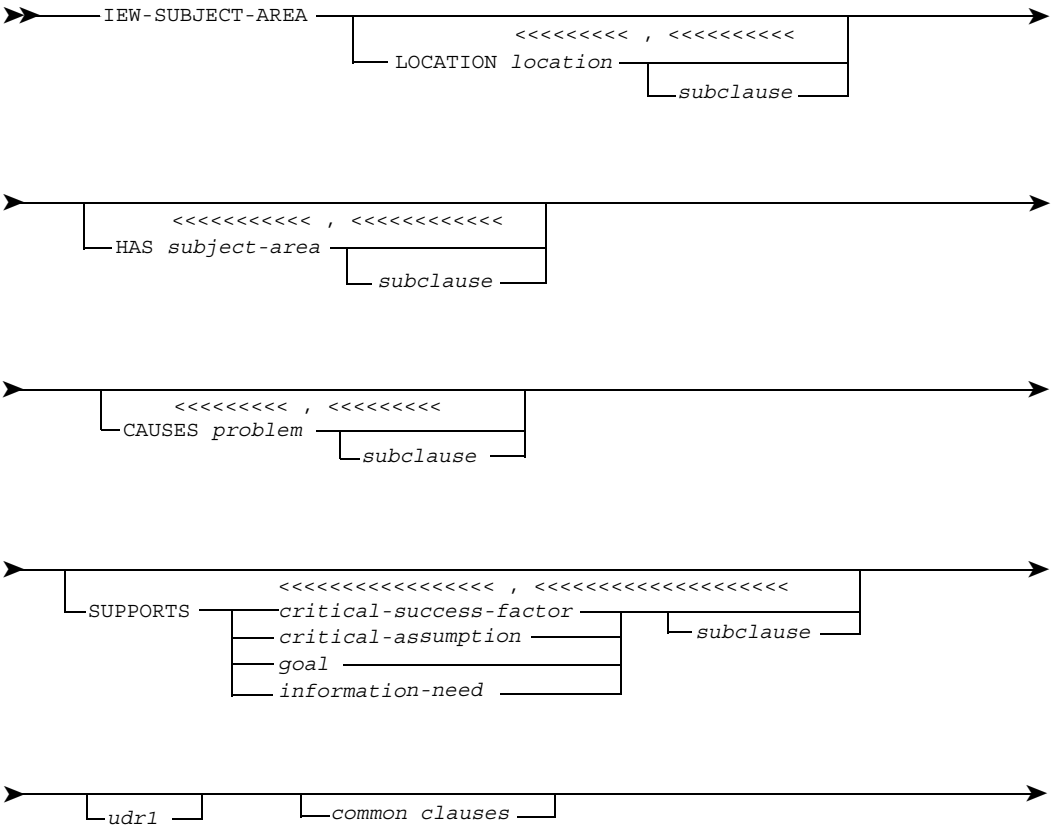


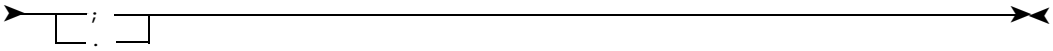
string is the data for import from, or export to, ADW/IEW.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SUBJECT-AREA Syntax

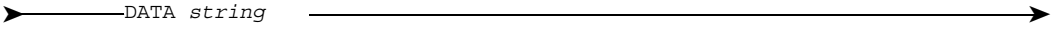




where:

location is an IEW-LOCATION member name.

subclause is:



string is the data to be imported from, or exported to, ADW/IEW.

subject-area is an IEW-SUBJECT-AREA member name.

problem is an IEW-PROBLEM member name.

critical-success-factor is an IEW-CRITICAL-SUCCESS-FACTOR member name.

critical-assumption is an IEW-CRITICAL-ASSUMPTION member name.

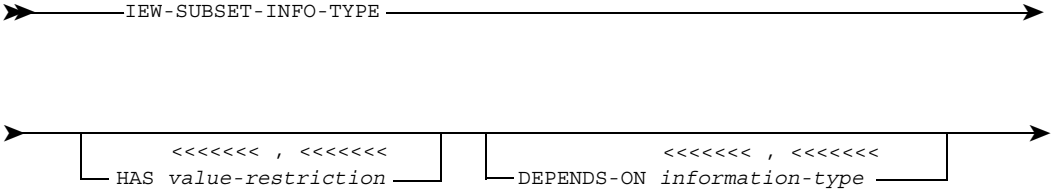
goal is an IEW-GOAL member name.

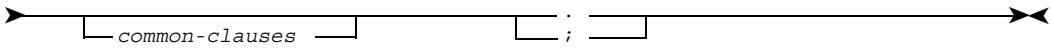
information-need is an IEW-INFORMATION-NEED member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SUBSET-INFO-TYPE Syntax





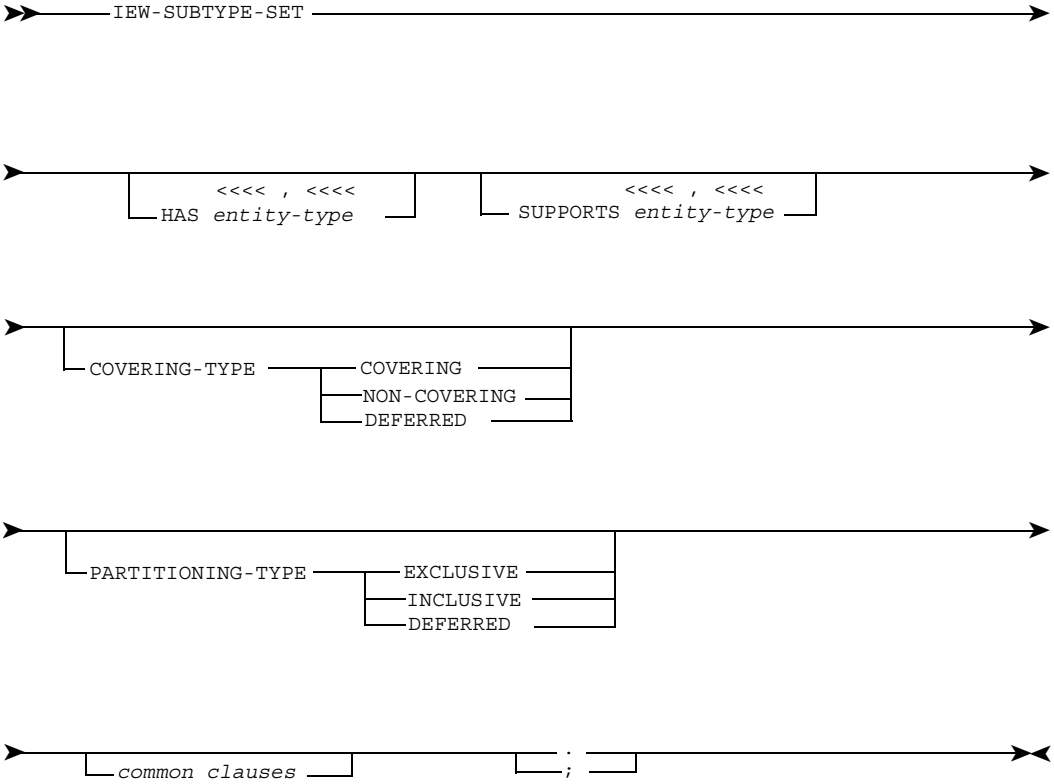
where:

information-type is an IEW-INFORMATION-TYPE member name.

value-restriction is an IEW-VALUE-RESTRICTION member name

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-SUBTYPE-SET Syntax

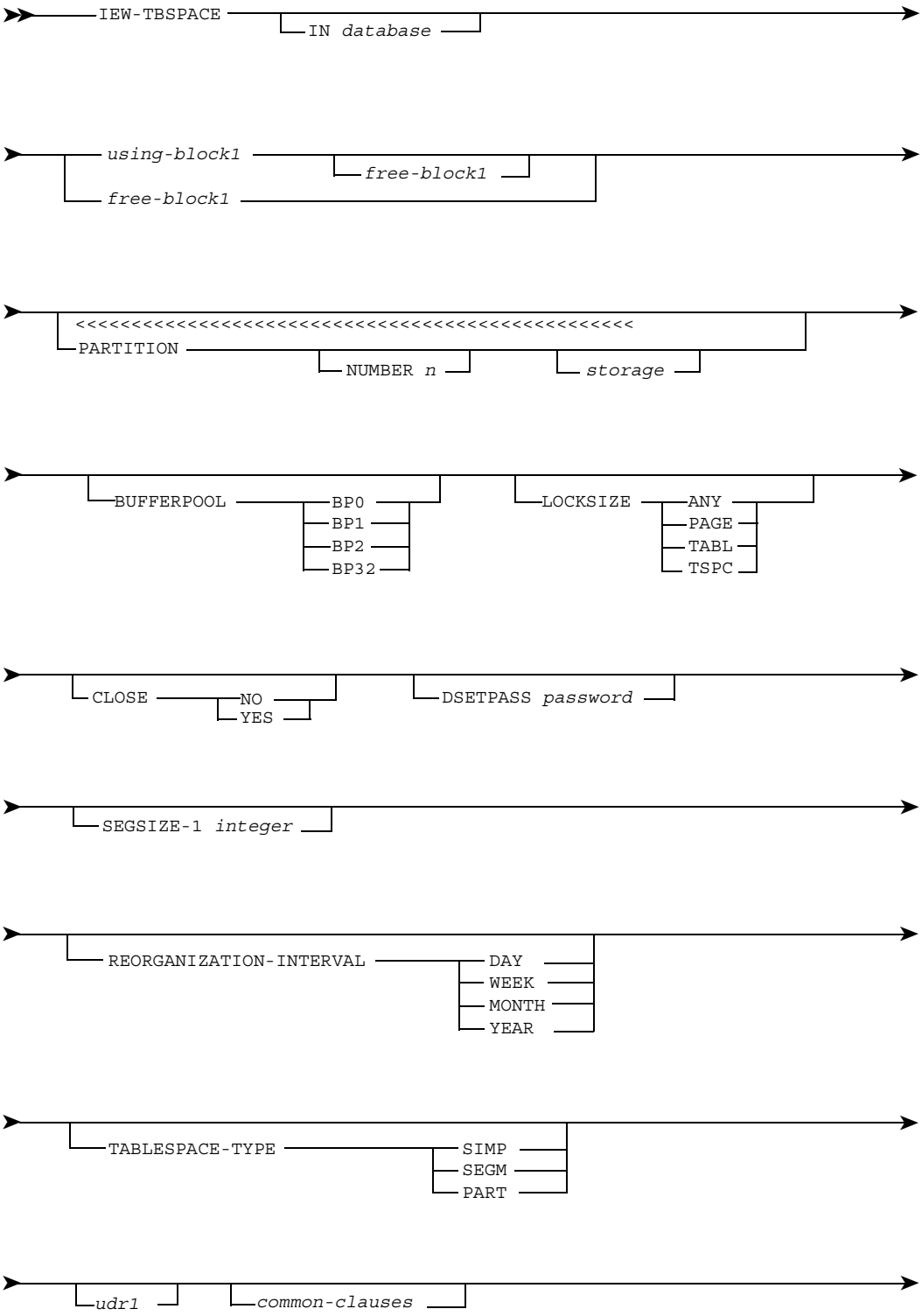


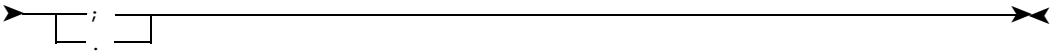
where:

entity-type is an IEW-ENTITY-TYPE member name.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-TBSPACE Syntax

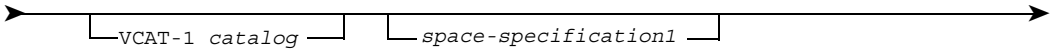




where:

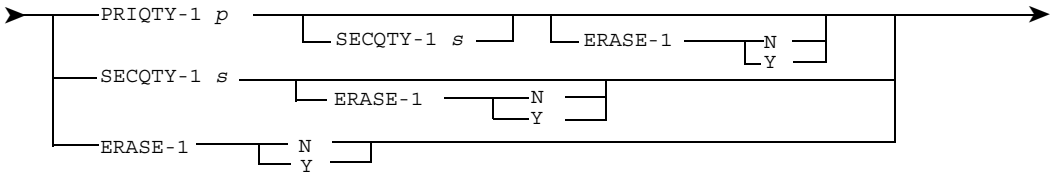
database is an IEW-RELATIONAL-DATABASE member name.

using-block1 is:



catalog is a VSAM catalog name of no more than 8 characters.

space-specification1 is:

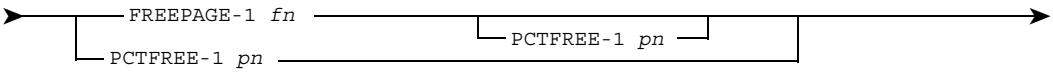


where:

p is an integer in the range 3 to 419,304.

s is an integer in the range 0 to 131,068.

free-block1 is:



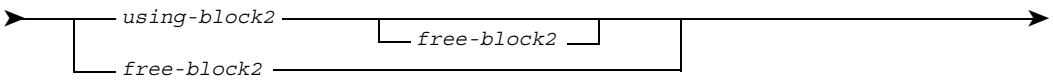
where:

fn is an integer in the range 0 to 255.

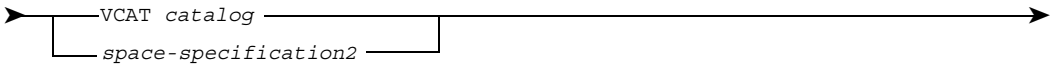
pn is an integer in the range 0 to 99.

n is an integer in the range 1 to 64.

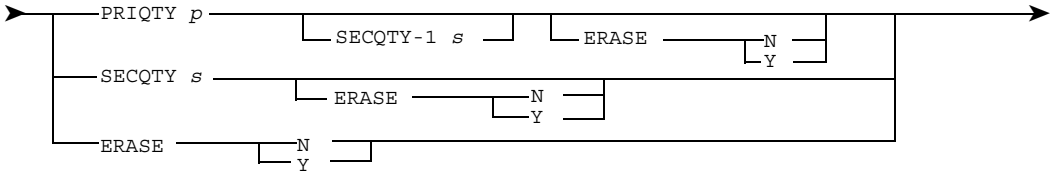
storage is:



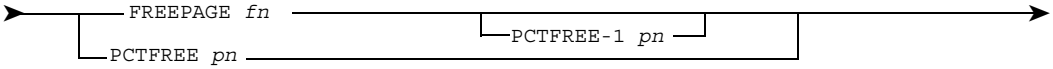
using-block2 is:



space-specification2 is:



free-block2 is:

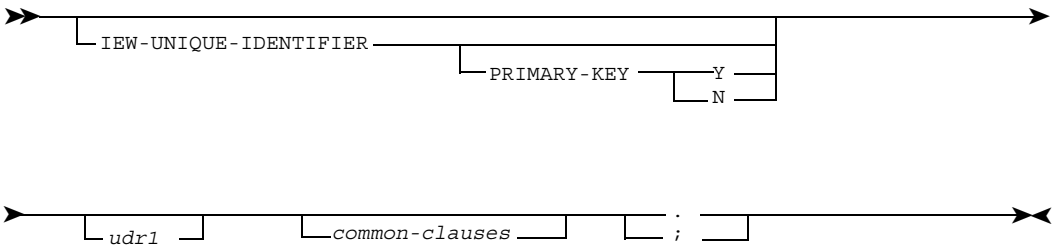


password is a VSAM data set password, of no more than 8 characters.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-UNIQUE-IDENTIFIER Syntax

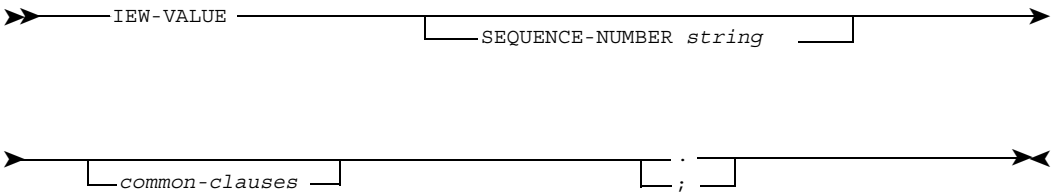


where:

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-VALUE Syntax

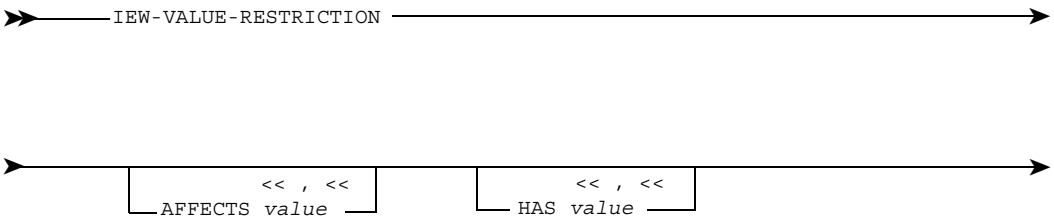


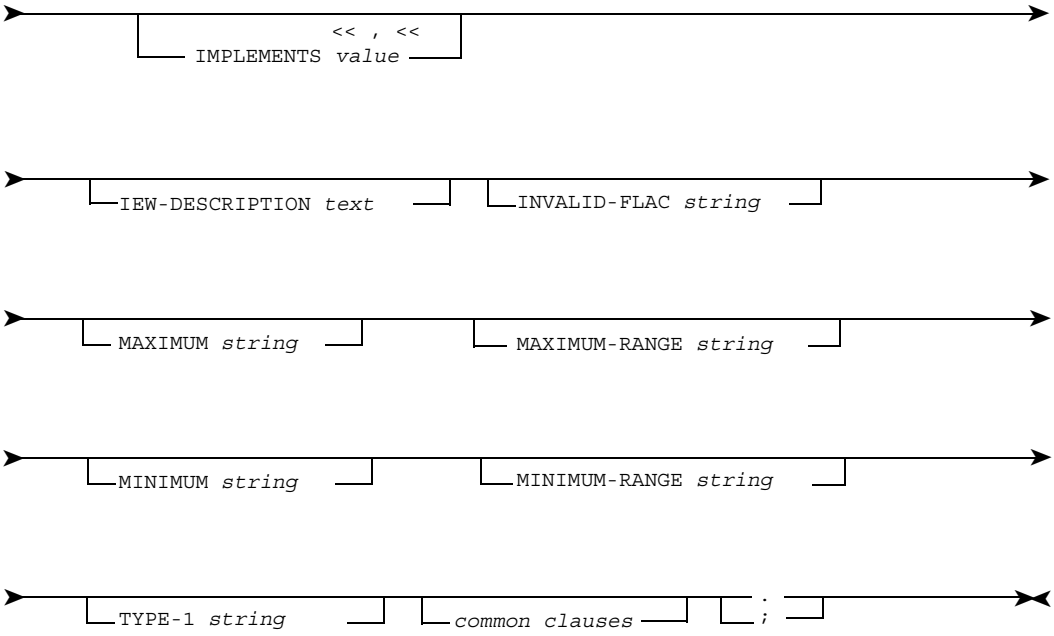
where:

string is data to be imported from, or exported to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-VALUE-RESTRICTION Syntax





where:

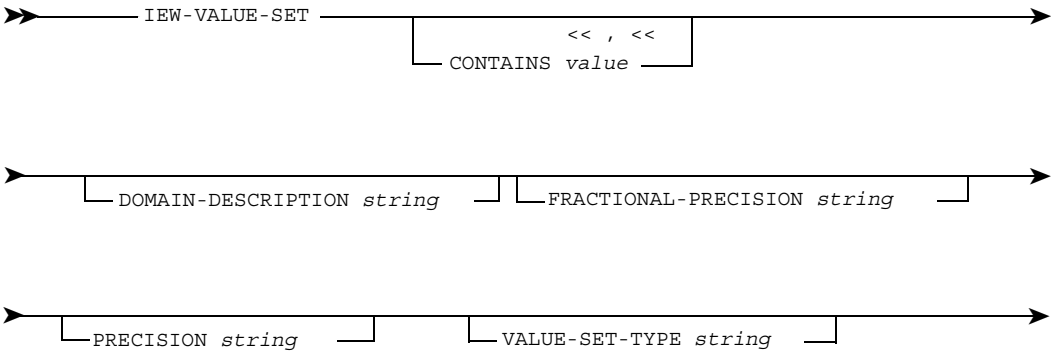
value is an IEW-VALUE member name.

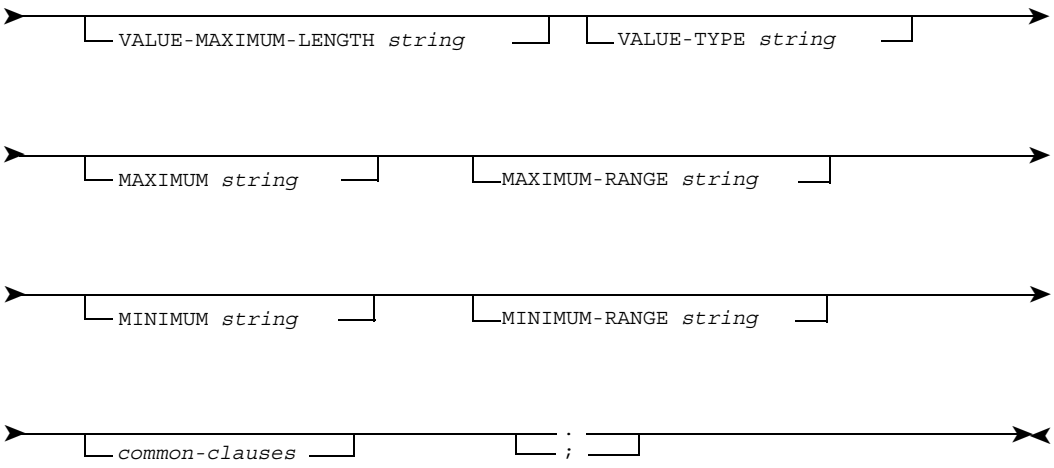
text is up to 32,767 delimited strings, each string having a maximum length of 60 characters.

string is data to be imported from, or exported to, ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IEW-VALUE-SET Syntax





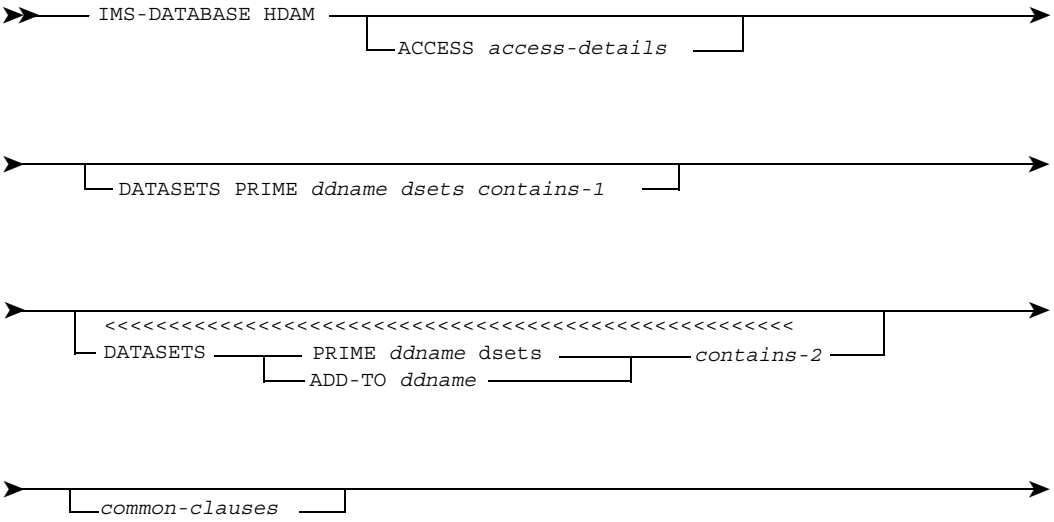
where:

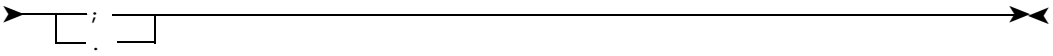
value is an IEW-VALUE member name.

string is data to be imported from, or exported to ADW/IEW.

common-clauses are defined in "Common Clauses Syntax" on page 152.

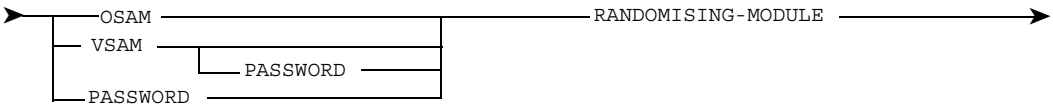
IMS-DATABASE HDAM Syntax

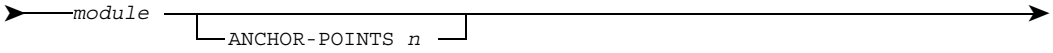


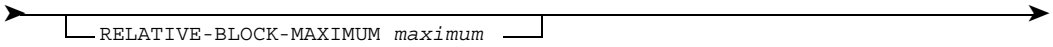


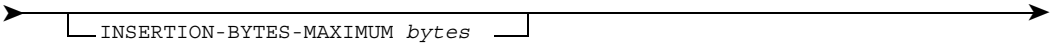
where:

access-details is:









module is a MODULE name.

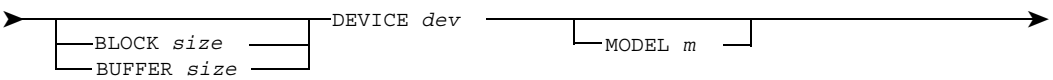
n is an unsigned integer in the range 1 to 255, being the number of root anchor points required in each control interval or block.

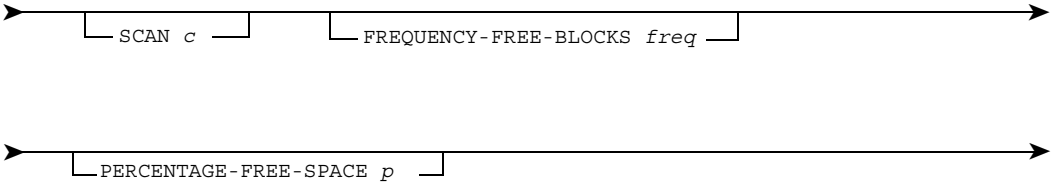
maximum is an unsigned integer in the range I to 16,777,215, being the maximum block number to be produced by the randomizing module.

bytes is an unsigned integer in the range 1 to 16,777,215, being the maximum number of bytes to be inserted into the root addressable area.

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the physical file.

dsets are:





where:

size is the number of bytes required per physical block or control interval.

dev is one of the numbers from this list:

2305	3330
2314	3340
2319	3350
	3375
	3380

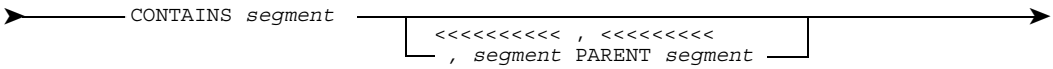
m is an integer: 1 or 2 if device is 2,305, or 1 or 11 if device is 3,330.

c is an unsigned integer in the range 0 to 255.

freq is either 0, or an unsigned integer in the range 2 to 100.

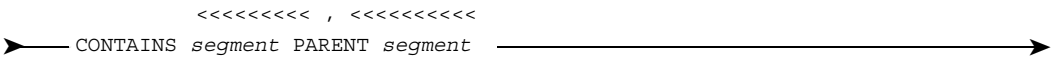
p is an unsigned integer in the range 0 to 99.

contains-1 is:



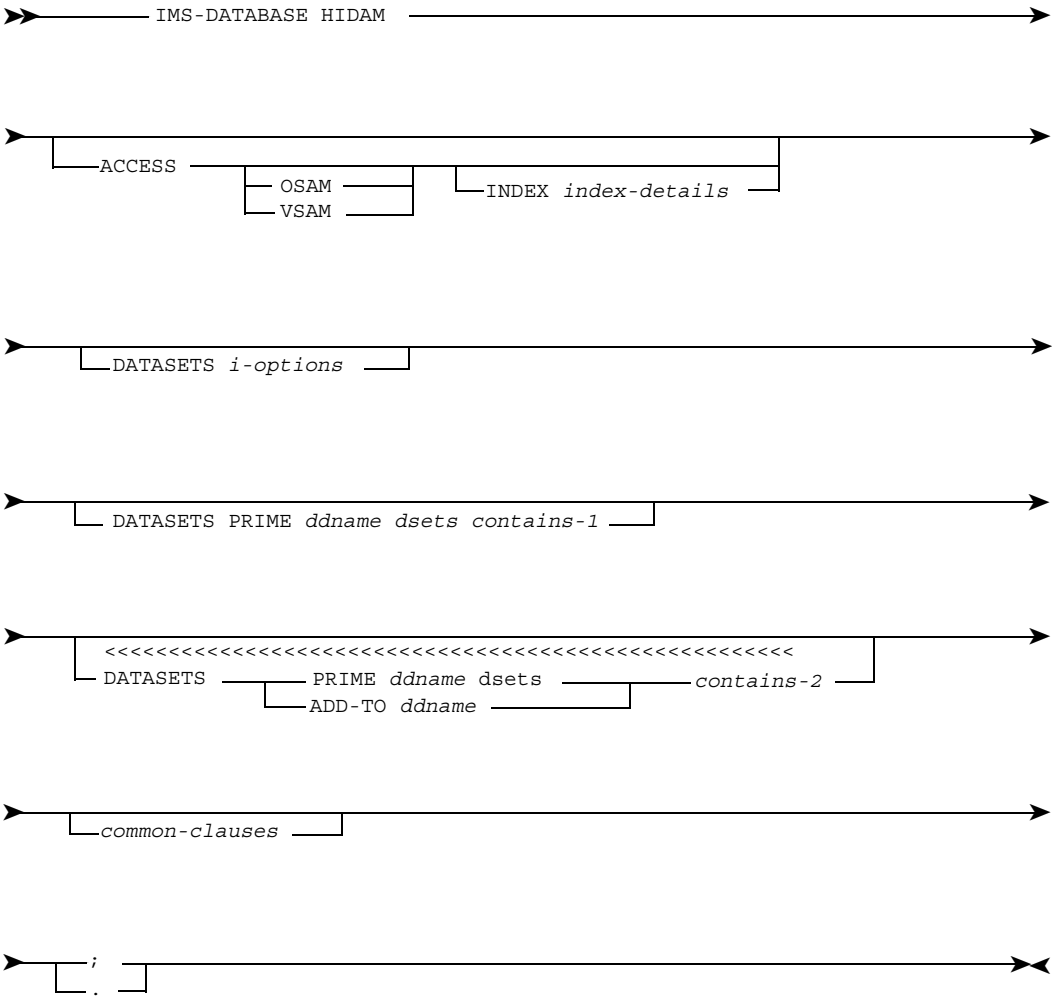
segment is the name of a PHYSICAL SEGMENT.

contains-2 is:



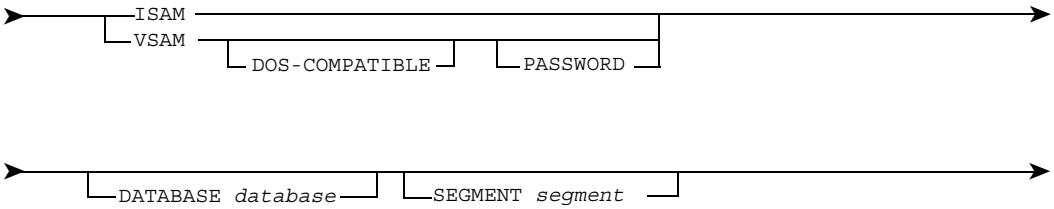
common-clauses are defined in "Common Clauses Syntax" on page 152.

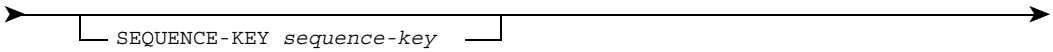
IMS-DATABASE HIDAM Syntax



where:

index-details are:





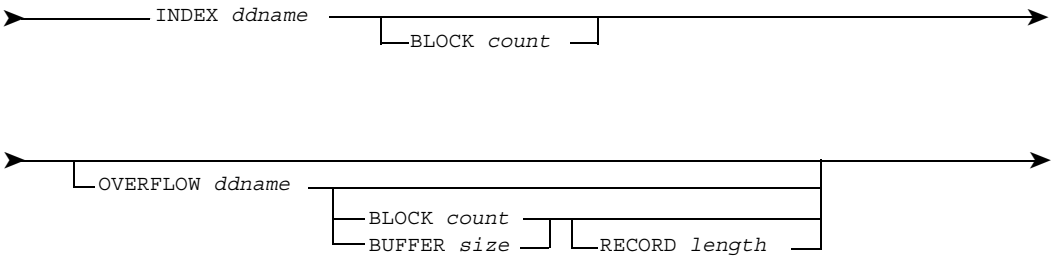
where:

database is the primary index database that is associated with this HIDAM database, consisting of 1 to 8 alphanumeric characters.

segment is the name of a PHYSICAL SEGMENT.

sequence-key is the sequence key name of the primary index database that is associated with this HIDAM database, consisting of 1 to 8 alphanumeric characters.

i-options is:



where:

ddname is the logical name used in the job control to identify the physical file, consisting of 1 to 8 alphanumeric characters.

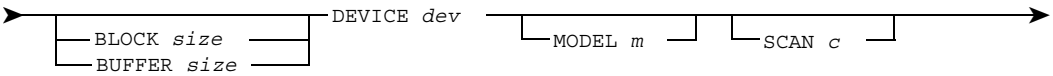
count is the number of logical records per physical block.

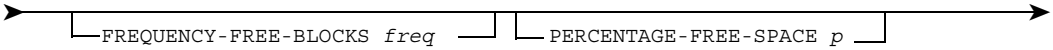
size is the number of bytes required per physical block or control interval.

length is the maximum length (in bytes) of a logical record. If VSAM is the operating system access method, length must be an even value.

ddname is 1 to 8 alphanumeric characters, being the logical name used in the job control to identify the physical file.

dsets are:





size is the number of bytes required per physical block or control interval.

dev is one of the numbers from this list:

2305	3330
2314	3340
2319	3350
	3375
	3380

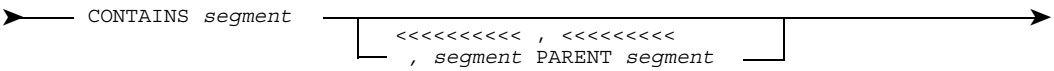
m is an integer: 1 or 2 if device is 2,305, or 1 or 11 if device is 3,330.

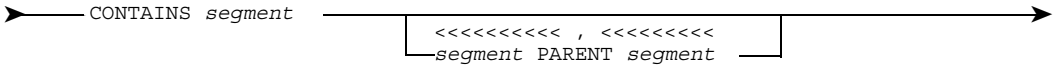
c is an unsigned integer in the range 0 to 255.

freq is either 0, or an unsigned integer in the range 2 to 100.

p is an unsigned integer in the range 0 to 99.

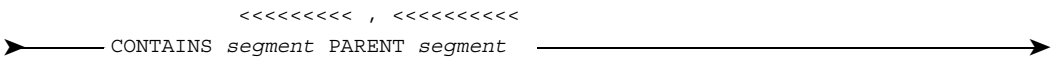
contains-1 is:





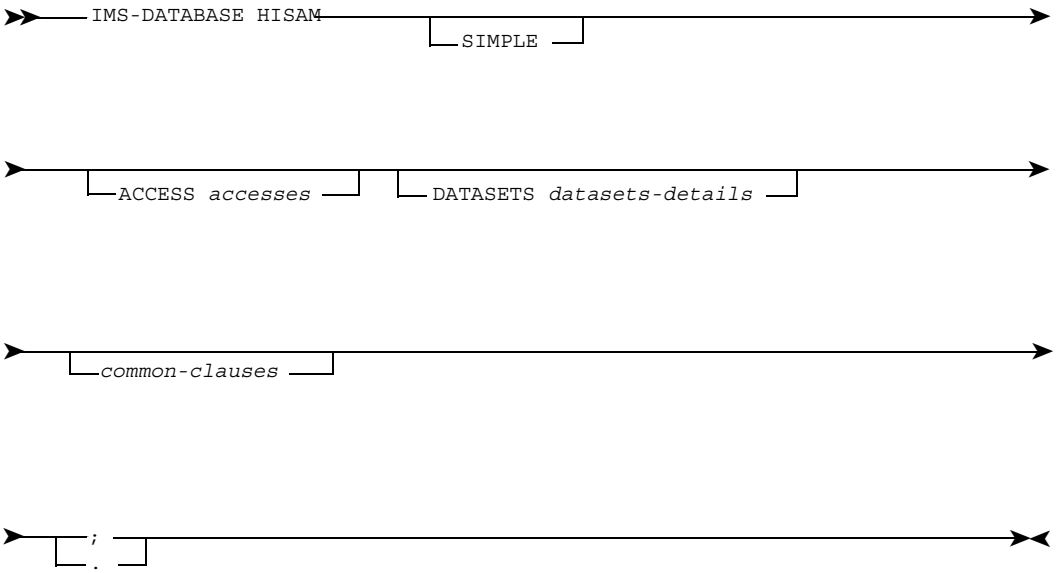
segment is the name of a PHYSICAL SEGMENT.

contains-2 is:



common-clauses are defined in "Common Clauses Syntax" on page 152.

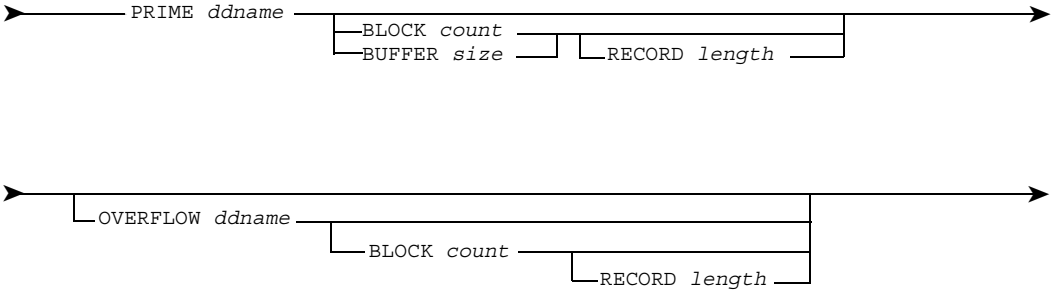
IMS-DATABASE HISAM Syntax

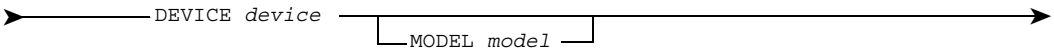


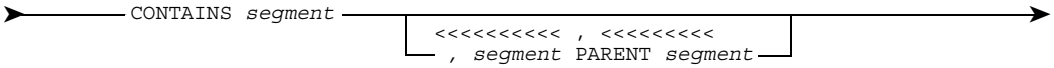
where:

accesses is:

datasets-details are:







where:

ddname is the logical name used in the job control to identify the physical file, consisting of 1 to 8 alphanumeric characters.

count is the number of logical records per physical block.

size is the number of bytes required per physical block or control interval.

length is the maximum length (in bytes) of a logical record. If VSAM is the operating system access method, length must be an even value.

device is one of the numbers from this list.

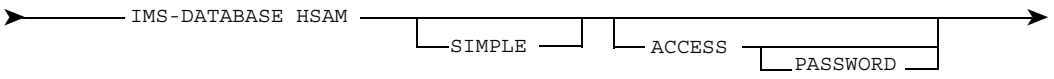
2305	3330
2314	3340
2319	3350
	3375
	3380

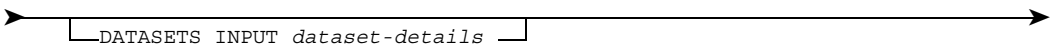
model is an integer: 1 or 2 if device is 2,305, or 1 or 11 if device is 3,330.

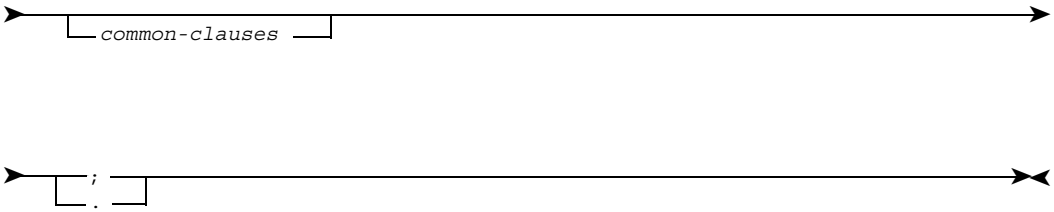
segment is the name of a PHYSICAL SEGMENT.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IMS-DATABASE HSAM Syntax

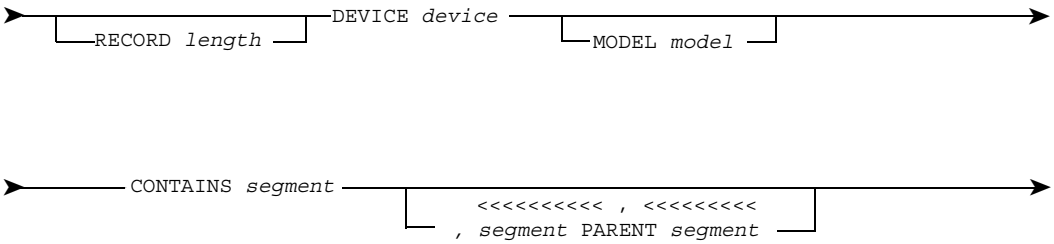






where:

datasets-details are:



where:

length is the maximum length (in bytes) of a logical record. If VSAM is the operating system access method, length must be an even value.

device is one of the numbers from this list:

2305	3330
2314	3340
2319	3350
	3375
	3380

model is an integer, 1 or 2 if device is 2,305, or 1 or 11 if device is 3,330.

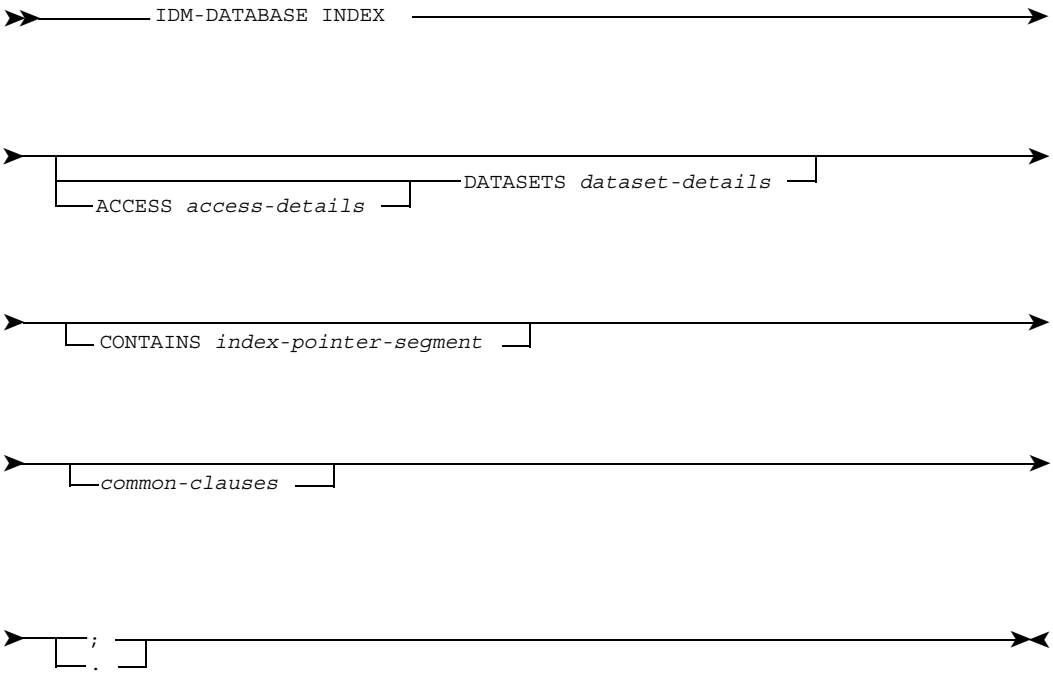
segment is the name of a PHYSICAL SEGMENT.

common-clauses are defined in "Common Clauses Syntax" on page 152.

Note:

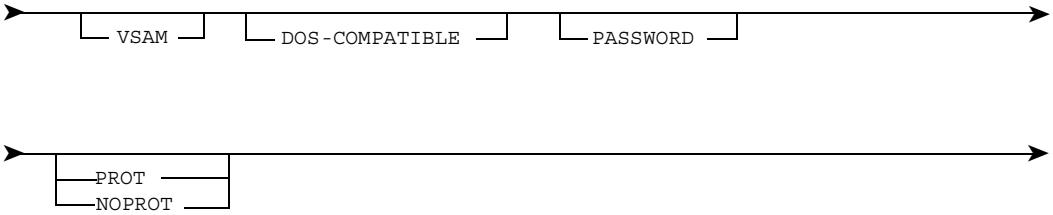
This IMS database access method type is not actually supported by ADW/IEW. However, it can be included in your repository if required.

IMS-DATABASE INDEX Syntax

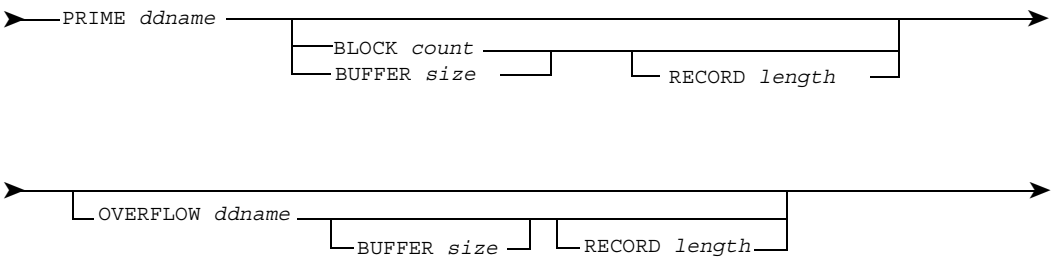


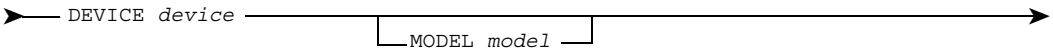
where:

access-details are:



dataset-details are:





where:

ddname is the logical name used in the job control to identify the physical file, consisting of 1 to 8 alphanumeric characters.

count is the number of logical records per physical block.

size is the number of bytes required per physical block or control interval.

length is the maximum length (in bytes) of a logical record. If VSAM is the operating system access method, length must be an even value.

device is one of the numbers from this list:

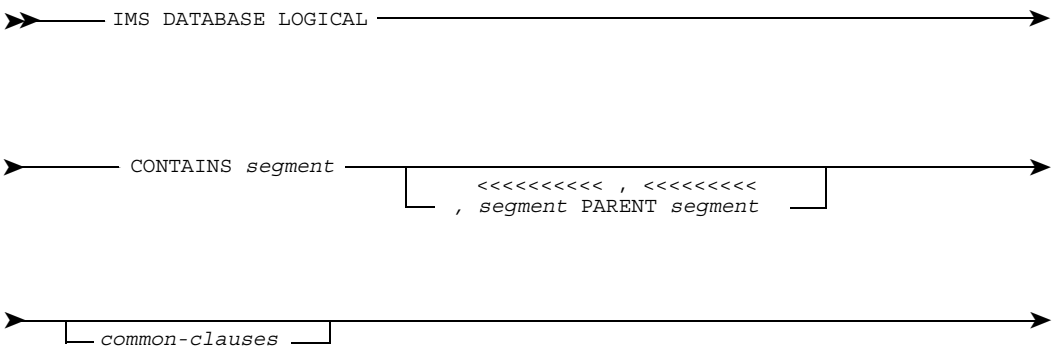
2305	3330
2314	3340
2319	3350
	3375
	3380

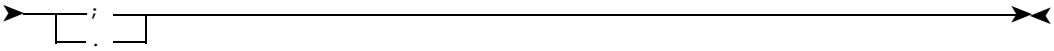
model is an integer: 1 or 2 if device is 2,305, or 1 or 11 if device is 3,330.

index-pointer-segment is the name of an INDEX-POINTER SEGMENT.

common-clauses are defined in "Common Clauses Syntax" on page 152.

IMS-DATABASE LOGICAL Syntax



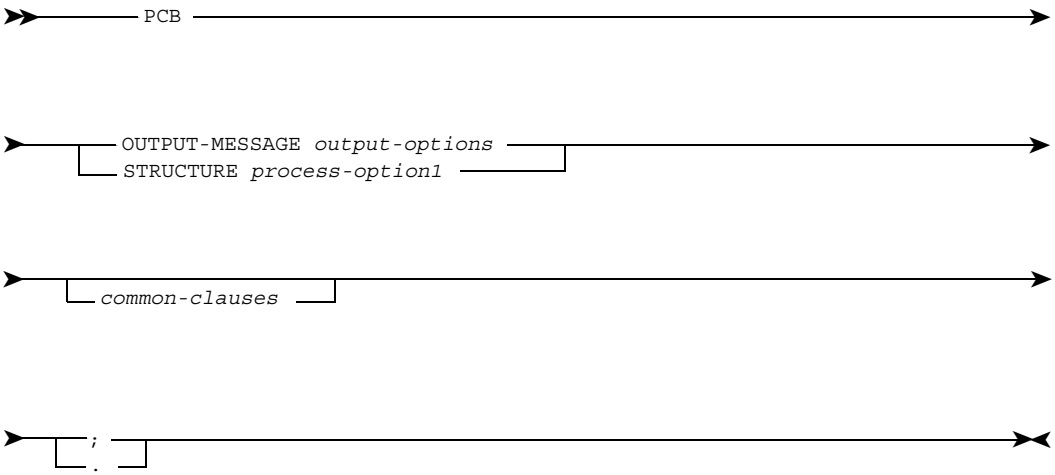


where:

segment is the name of a PHYSICAL SEGMENT.

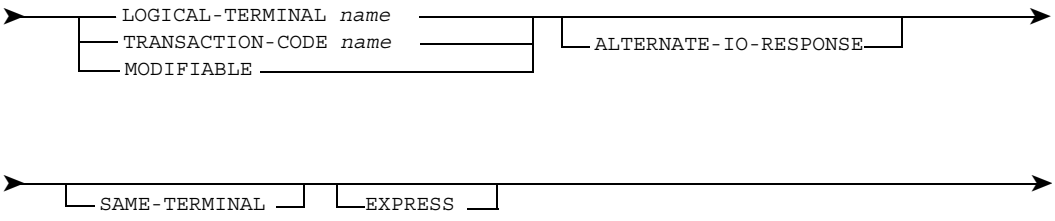
common-clauses are defined in "Common Clauses Syntax" on page 152.

PCB Syntax



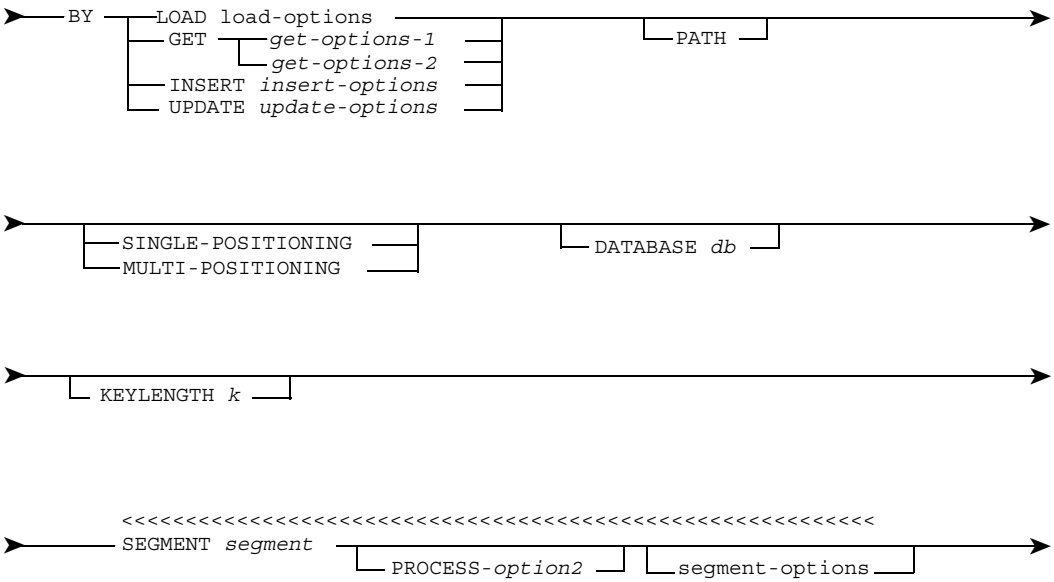
where:

output-options is:

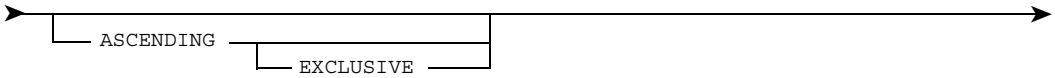


name is an alphanumeric name, 1 to 8 characters in length.

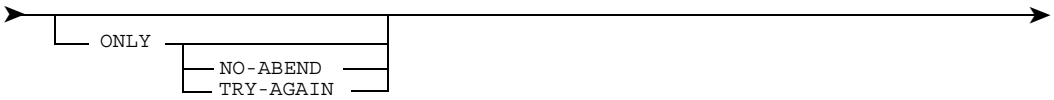
process-option1 is:



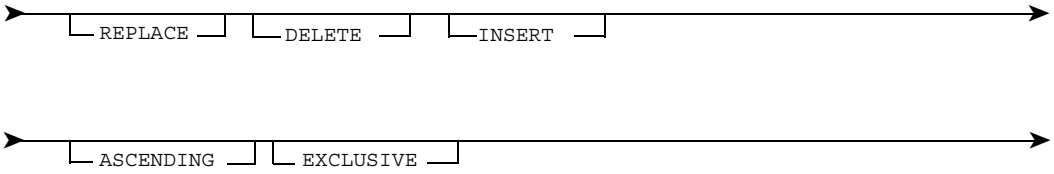
load-options are:



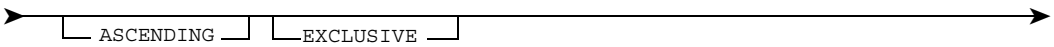
get options-1 are:



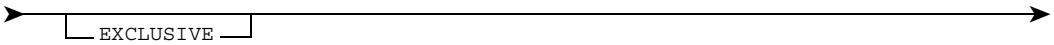
get-options-2 are:



insert-options are:



update-options are:



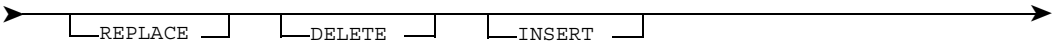
db is the name of a DATABASE member.

segment is the name of a SEGMENT member.

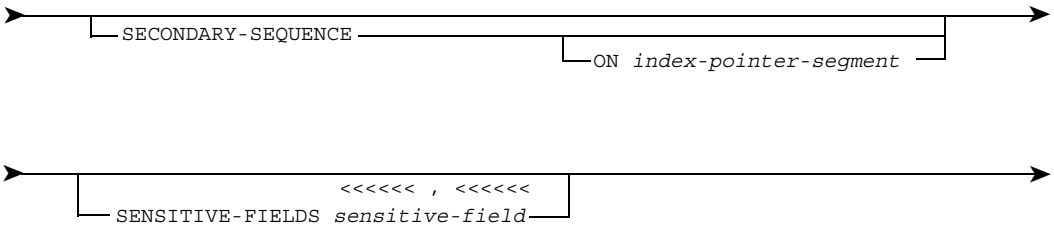
process-option2 are:

where:

g-options are:



segment-options are:

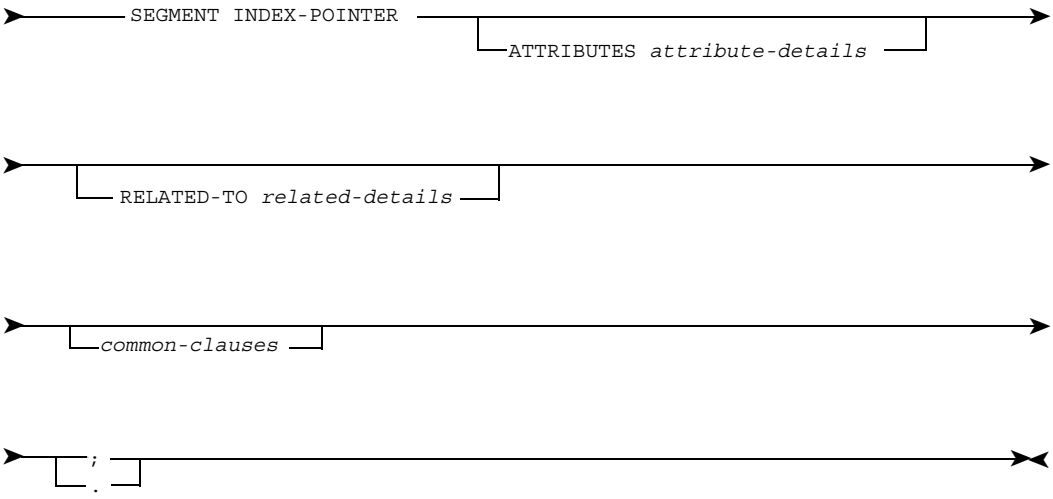


index-pointer-segment is the name of a SEGMENT member, that is a SEGMENT INDEX-POINTER.

sensitive-field is the name of a GROUP, ITEM or sequence key member or concatenated key member.

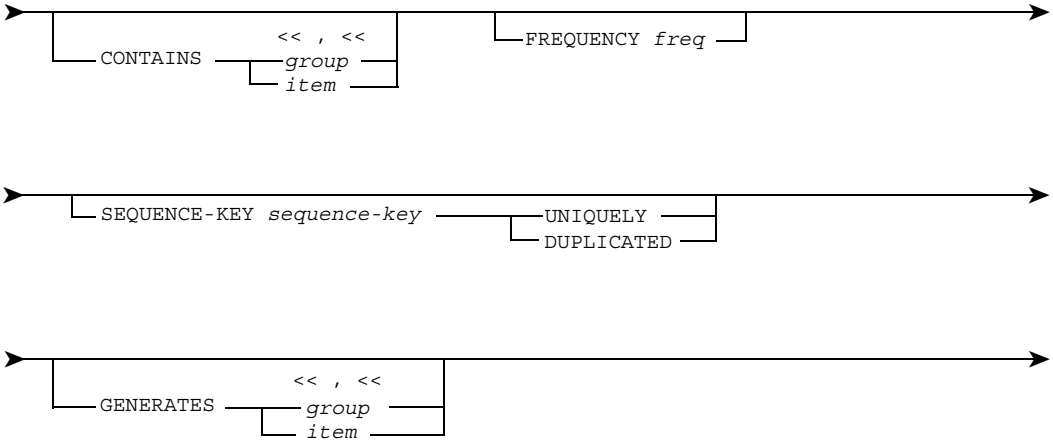
common-clauses are defined in "Common Clauses Syntax" on page 152.

SEGMENT INDEX-POINTER Syntax



where:

attribute-details are:



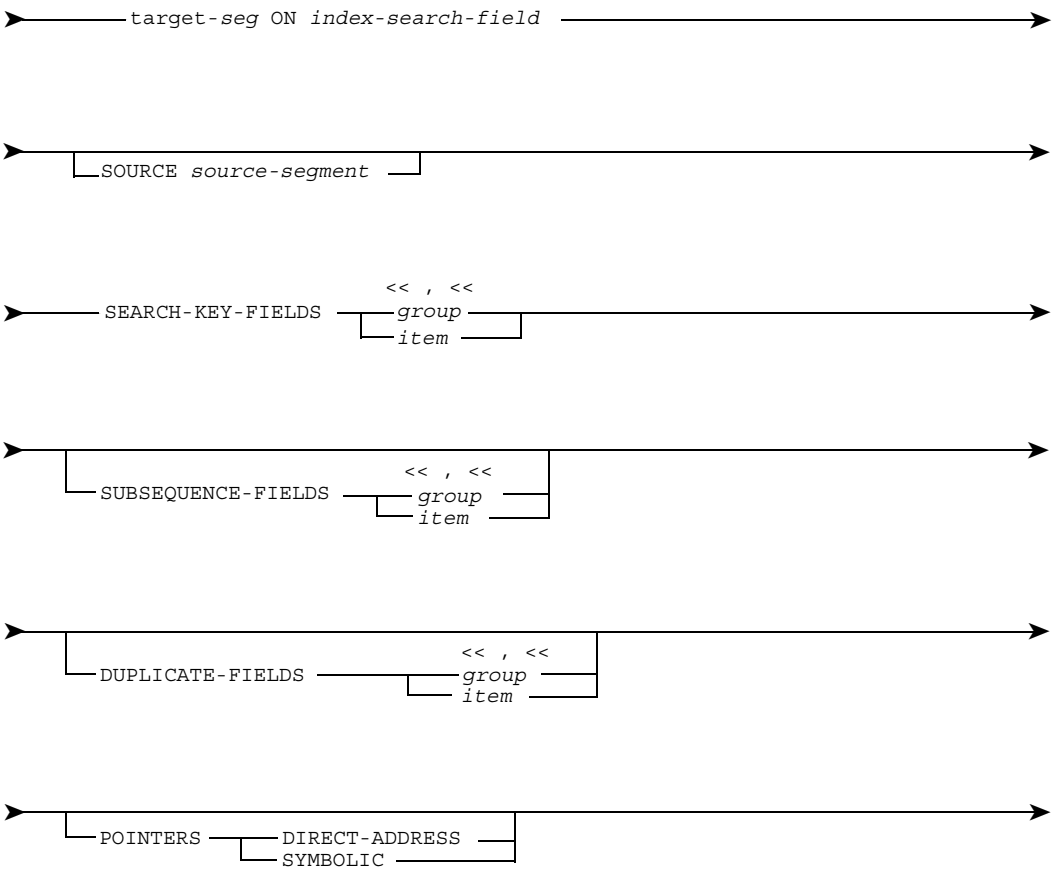
group is the name of a GROUP.

item is the name of an ITEM.

freq is an unsigned integer in the range 1 to 16,777,215.

sequence-key is a unique alphanumeric name of 1 to 8 characters.

related-details are:



where:

target-seg is the name of a SEGMENT that is a PHYSICAL TARGET-SEGMENT.

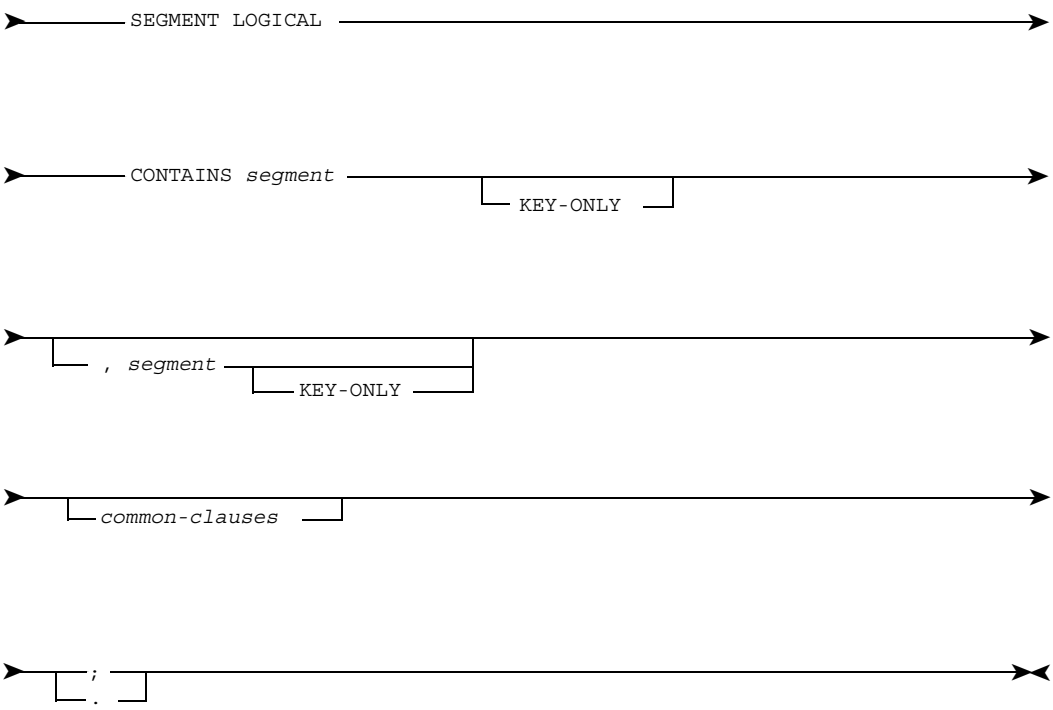
index-search-field is a 1 to 8 character unique alphanumeric name.

source-segment is the name of a SEGMENT that is a PHYSICAL SOURCE-SEGMENT.

module is the name of a MODULE.

common-clauses are defined in "Common Clauses Syntax" on page 152.

SEGMENT LOGICAL Syntax

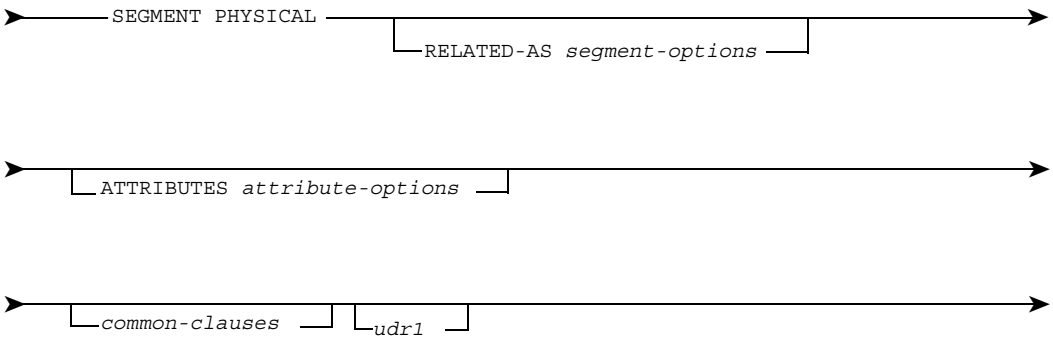


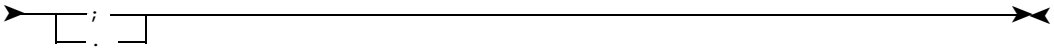
where:

segment is the name of a PHYSICAL SEGMENT.

common-clauses are defined in "Common Clauses Syntax" on page 152.

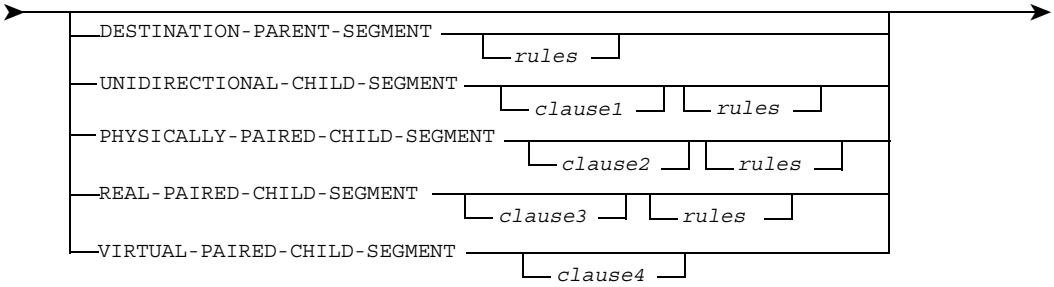
SEGMENT PHYSICAL Syntax

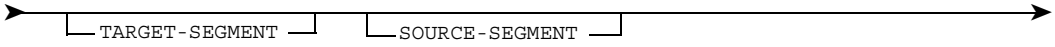




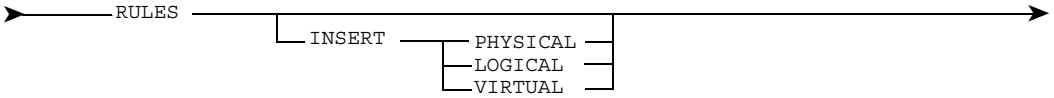
where:

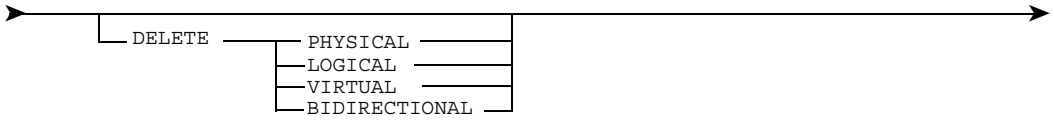
segment-options are:

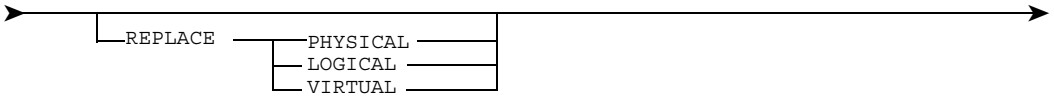




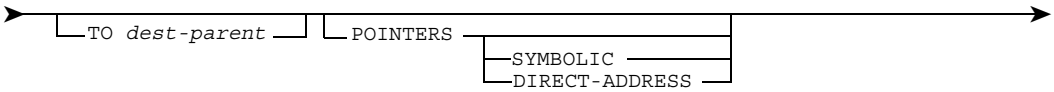
rules are:





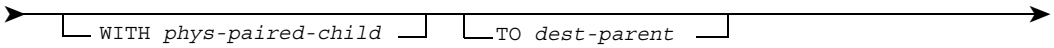


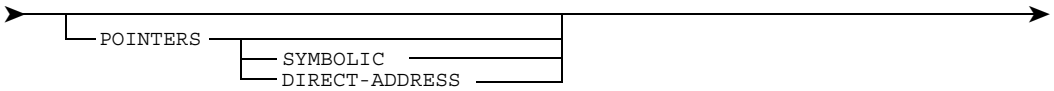
clause1 is:



dest-parent is the name of a PHYSICAL DESTINATION-PARENT-SEGMENT.

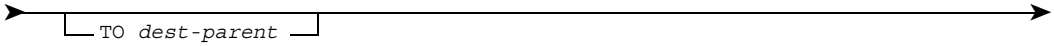
clause2 is:

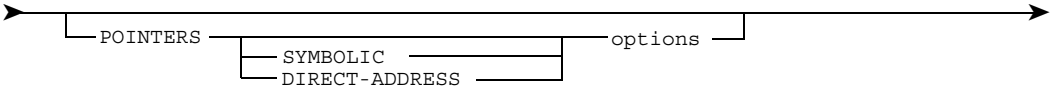




phys-paired-child is the name of a PHYSICALLY-PAIRED-CHILD-SEGMENT.

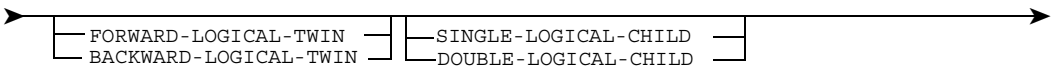
clause3 is:



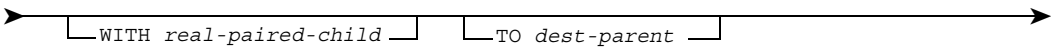


dest-parent is as defined above.

options are:



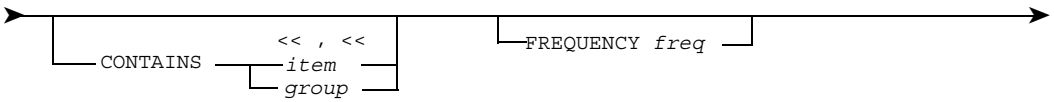
clause4 is:

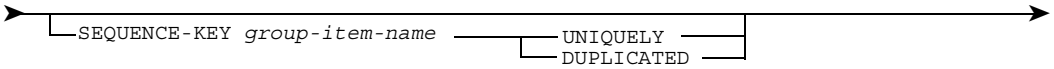


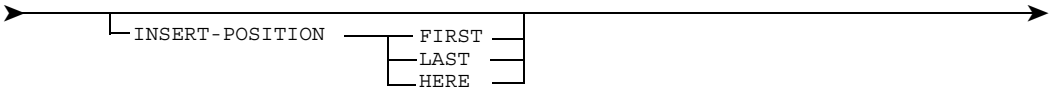
real-paired-child is the name of a PHYSICAL REAL-PAIRED-CHILD-SEGMENT.

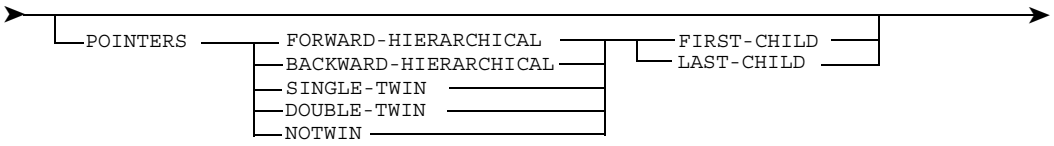
dest-parent is as defined above.

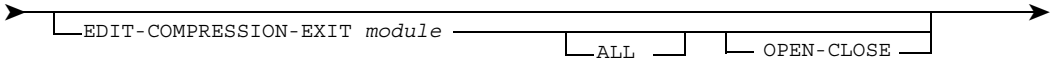
attribute-options are:

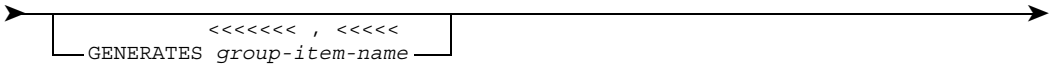












item is an ITEM member name.

group is a GROUP member name.

group-item-name is a GROUP or ITEM member name.

freq is an unsigned number in the range 1 to 16,777,215.

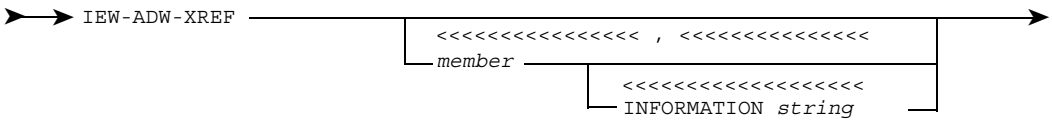
module is a MODULE member name.

udr1 is defined in "The Udr1 and Udr2 Clauses Syntax" on page 151.

common-clauses are defined in "Common Clauses Syntax" on page 152.

The Udr1 and Udr2 Clauses Syntax

The *udr1* clause is a user-defined EA relationship type. By default, the syntax is:

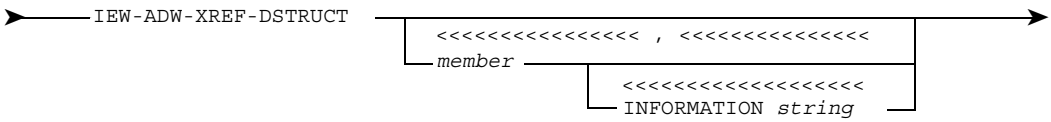


where:

member is the name of a repository member.

string is the data to be imported from, or exported to, ADW/IEW.

The *udr2* clause is a user defined EA relationship type. By default, this is:



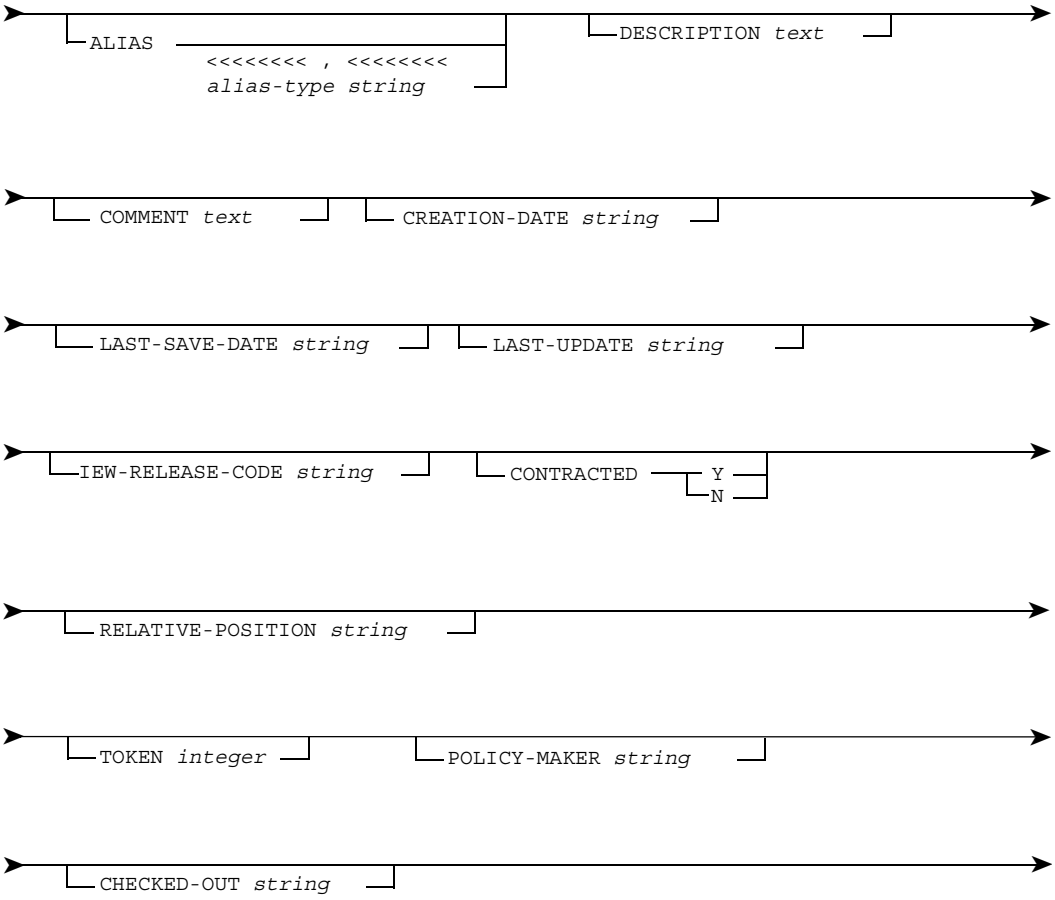
member is the name of a repository member.

string is the data to be imported from, or exported to, ADW/IEW.

Refer to "UDR Clauses" on page 4 for further details.

Common Clauses Syntax

The *common clauses* described in the syntax of the previous member-types are defined as:



where:

alias-type is one of IEW, LOCAL-NAME, FORMAL-NAME, GAMMA, or the other specified alias types (see "MPDYIITAB2" on page 30). Alias types can be tailored; for example, ADW users can the IEW alias type with an ADW alias type.

string is the data for import from, or export to, ADW/IEW.

integer is an 11 digit positive integer.

text is up to 32,767 delimited strings with each string having a maximum length of 60 characters.

Note:

The IEW-RELEASE-CODE clause is not currently used by either the import or export facilities. By specifying it you can manually document the ADW or IEW release levels in place when the object was imported or exported.

Appendix A

ADW/IEW Associations Mapped Against Manager Products Repository Relationship Clauses

`20000 Process "contains" Junction.`
Process CONTAINS Junction

`20001 Process "interacts with" External Agent.`
Process CONTAINS External Agent

`20006 Process "is scope of" Data Store.`
Process RELIES-ON Data Store.

`20007 Attribute Type "describes" Entity Type.`
Entity Type CONTAINS Attribute Type.

`20010 Data Flow "consists of" Data Flow.`
Dataflow CONTAINS Dataflow.

`20020 Data Flow Vector "carries" Data Flow.`
No direct mapping (used to connect 20040 to Dataflow Object).

`20023 Data Flow Vector "flows from" Junction.`
Dataflow IEW-ADW-XREF Junction INFORMATION "FLOWS FROM".

`20026 Data Flow Vector "flows to" Junction.`
Dataflow IEW-ADW-XREF Junction INFORMATION "FLOWS TO"

`20027 Organizational Unit "manages" Organizational Unit.`
Organizational Unit INFLUENCES Organizational Unit.

`20031 Subject Area "consists of" Subject Area.`
Subject Area HAS Subject Area.

`20034 <activity> "consists of" <activity>.`
Function HAS function
Function HAS Process
Process HAS Process
Process HAS Sequential Process
Sequential Process HAS Sequential Process

```
20040 Data Flow Vector.
```
Dataflow SOURCE Process
Dataflow SOURCE Sequential-Process
Dataflow DESTINATION Process
Dataflow DESTINATION Sequential-Process

```
20041 <view> "involves" Entity Type.
```
Data Collection IEW-ADW-XREF Entity Type
Dataflow IEW-ADW-XREF Entity Type
Datastore IEW-ADW-XREF Entity Type
Process IEW-ADW-XREF Entity Type
Sequential Process IEW-ADW-XREF Entity Type
Subject Area IEW-ADW-XREF Entity Type
External Agent IEW-ADW-XREF Entity Type

```
20042 <view> "involves" Attribute Type.
```
Data Collection IEW-ADW-XREF Attribute Type
Dataflow IEW-ADW-XREF Attribute Type
Datastore IEW-ADW-XREF Attribute Type
Process IEW-ADW-XREF Attribute Type
Sequential Process IEW-ADW-XREF Attribute Type
Subject Area IEW-ADW-XREF Attribute Type
External Agent IEW-ADW-XREF Attribute Type

```
20043 <view> "involves" Relationship Type
```
Data Collection IEW-ADW-XREF Relationship Type
Dataflow IEW-ADW-XREF Relationship Type
Datastore IEW-ADW-XREF Relationship Type
Process IEW-ADW-XREF Relationship Type
Sequential Process IEW-ADW-XREF Relationship Type
Subject Area IEW-ADW-XREF Relationship Type
External Agent IEW-ADW-XREF Relationship Type

```
20044 Relationship Type
```
Entity Type CONTAINS Relationship-Type CONTAINS Entity Type

```
20045 Sequential Process "contains" Data Store Access
```
Sequential Process CONTAINS Datastore Access

```
20046 Data Store Access "accesses" Entity Type.
```
Data Store Access IMPLEMENTS Entity Type

```
20047 Module "enacts" Sequential Process.
```
Module IEW-ADW-XREF Sequential Process

```
20049 Module "synchronously calls" Module.
```
Module CALLS Module

```
20050 Module "asynchronously calls" Module.
```
Module CALLS Module.

```
20051 Module "recursively calls" Module.
```
Module CALLS Module.

```
20052 Library "contains" Module.
```
Library CONTAINS Module.

```
20053 Module "has formal" Parameter.
```
Module PARAMETER Parameter.

```
20054 Information Need "consists of" Information Need.
```
Information Need HAS Information Need.

```
20055 Location "consists of" Location.
```
Location LOCATION Location.

```
20056 Project "consists of" Project.
```
Project HAS Project.

```
20057 Mechanism "consists of" Mechanism.
```
Mechanism HAS Mechanism.

```
20058 Data Collection "consists of" Data Collection.
```
Data Collection HAS Data Collection.

```
20059 Goal "consists of" Goal.
```
Goal HAS Goal.

```
20060 Problem "consists of" Problem.
```
Problem HAS Problem.

```
20061 DBD "has" Root Segment.
```
DBD <datasets-clause> CONTAINS Segment (1st entry).

```
20062 Segment 1"consists of" Segment 2.
```
IMS Database CONTAINS Segment 1 PARENT Segment 2

```
20063 Segment 1"has" Logical Child Segment 2.
```
Segment 1 RELATED-AS DESTINATION-PARENT.
Segment 2 RELATED-AS UNDIRECTIONAL-CHILD- SEGMENT TO Segment 1
Segment 1 is:
REAL-PAIRED-CHILD-SEGMENT if relationship 20083 present
PHYSICALLY-PAIRED CHILD SEGMENT if relationship 20150 present.

```
20064 Segment 1 "has" Secondary Target Segment. 2
```
Segment 2 <related-as-clause> TARGET-SEGMENT
Segment 1 <related-as-clause> Segment 2 ON <ixfld-name>

```
20065 Segment 1 "has" Secondary Index Source Segment 2.
```
Segment 2 <related-as-clause> SOURCE-SEGMENT Segment 1 <related-as-clause> Segment 2.

```
20068 <data object> "has" <data type>.
```
Attribute Type CONTAINS Global Data Type
Attribute Type CONTAINS Local Data Type
Global Data Structure CONTAINS Global Data Type
Global Data Structure CONTAINS Local Data Type
Local Data Structure CONTAINS Global Data Type
Local Data Structure CONTAINS Local Data Type.

```
20069 Data Record "references" Data Record.
```
Global Data Record IEW-ADW-XREF Global Data Record
Local Data Record IEW-ADW-XREF Local Data Record.

```
20071 <data structure> "implements" Entity Type.
```
Entity Type CONTAINS Global Data Structure.
Entity Type CONTAINS Local Data Structure.

```
20072 Tie "implements" <predicate type>.
```
Not mapped.

```
20073 <data area> "has composition" <data structure>.
```
File Record CONTAINS Global Data Structure
File Record CONTAINS Local Data Structure
Parameter CONTAINS Global Data Structure
Parameter CONTAINS Local Data Structure
Relation CONTAINS Global Data Structure <EXPAND>
Relation CONTAINS Local Data Structure <EXPAND>
Screen CONTAINS Global Data Structure
Screen CONTAINS Local Data Structure
Segment CONTAINS Global Data Structure
Segment CONTAINS Local Data Structure
DB2 Table CONTAINS Global Data Record
DB2 Table CONTAINS Local Data Record
DB2 View CONTAINS Global Data Record
DB2 View CONTAINS Local Data Record

```
20074 < data structure> "ties" <data structure>.
```
Global Data Structure IEW-ADW-XREF Global Data Structure
Global Data Structure IEW-ADW-XREF Local Data Structure
Local Data Structure IEW-ADW-XREF Global Data Structure
Local Data Structure IEW-ADW-XREF Local Data Structure
Data Structure or Block IEW-ADW-XREF Global Data Structure
Data Structure or Block IEW-ADW-XREF Local Data Structure
Data Structure Rep Block IEW-ADW-XREF Local Data Structure
Data Structure Rep Block IEW-ADW-XREF Global Data Structure

20075 <data structure> "contains" <data structure>.

Data Structure Or Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Data Structure Or Block

Data Structure Or Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Data Structure Repetition Block

Data Structure Or Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Global Data Structure

Data Structure Or Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Local Data Structure

Data Structure Repetition Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Data Structure Or Block

Data Structure Repetition Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Data Structure Repetition Block

Data Structure Repetition Block CONTAINS and
ADW-IEW-XREF-DSTRUCT Global Data Structure

Data Structure Repetition Block CONTAINS Local Data Structure

Global Data Structure CONTAINS Data Structure Or Block Structure

Global Data Structure CONTAINS Data Structure Repetition Block

Global Data Structure CONTAINS Global Data Structure

Global Data Structure CONTAINS Local Data Structure

Local Data Structure CONTAINS Data Structure Or Block

Local Data Structure CONTAINS Data Structure Repetition Block

Local Data Structure CONTAINS Global Data Structure

Local Data Structure CONTAINS Local Data Structure

Global Data Record CONTAINS Local Data Structure

Global Data Record CONTAINS Data Structure or Block

Global Data Record CONTAINS Data Structure Rep Block

Local Data Record CONTAINS Local Data Structure

Local Data Record CONTAINS Data Structure or Block

Local Data Record CONTAINS Data Structure Rep Block

20076 <data area> "has local description" <data structure>.

File Record IEW-ADW-XREF Global Data Structure
File Record IEW-ADW-XREF Local Data Structure
Relation IEW-ADW-XREF Global Data Structure
Relation IEW-ADW-XREF Local Data Structure
Screen IEW-ADW-XREF Global Data Structure
Screen IEW-ADW-XREF Local Data Structure

Segment IEW-ADW-XREF Global Data Structure
Segment IEW-ADW-XREF Local Data Structure
Parameter IEW-ADW-XREF Global Data Structure
Parameter IEW-ADW-XREF Local Data Structure
DB2 Table IEW-ADW-XREF Global Data Structure
DB2 Table IEW-ADW-XREF Local Data Structure
DB2 View IEW-ADW-XREF Local Data Structure
DB2 View IEW-ADW-XREF Data-Structure or Block
DB2 View IEW-ADW-XREF Data-Structure Rep Block

`20077 <data area> "local description has context" <data structure>.`
(File Record IEW-ADW-XREF Global Data Structure) INFORMATION 'string'
(File Record IEW-ADW-XREF Local Data Structure) INFORMATION 'string'
(Relation IEW-ADW-XREF Global Data Structure) INFORMATION 'string'
(Relation IEW-ADW-XREF Local Data Structure) INFORMATION 'string'
(Screen IEW-ADW-XREF Global Data Structure) INFORMATION 'string'
(Screen IEW-ADW-XREF Local Data Structure) INFORMATION 'string'
(Segment IEW-ADW-XREF Global Data Structure) INFORMATION 'string'
(Segment IEW-ADW-XREF Local Data Structure) INFORMATION 'string'

`20078 Module "includes" Module.`
Module CONTAINS Module

`20079 <system component> "contains" <data area>.`
File Database CONTAINS File Record
Relation IN tablespace IN relational-db

where: *`tablespace`* is generated from relation name concatenated with relational-db name.

`20080 <data area> "references" <data area>.`
File Record IEW-ADW-XREF File Record
Relation (CONSTRAINT) REFERENCES Relation
DB2 Table IEW-ADW-XREF DB2 Table
INFORMATION 'string'
DB2 Table IEW-ADW-XREF DB2 View
INFORMATION 'string'
DB2 View IEW-ADW-XREF DB2 Table
INFORMATION 'string'
DB2 View IEW-ADW-XREF DB2 View
INFORMATION 'string'

`20081 <system component> "contains" Data Area Reference.`
<File Record IEW-ADW-XREF File Record>
INFORMATION 'File-Database'
<Relation IEW-ADW-XREF Relation>
INFORMATION 'Relational-Database'

20082 Screen "initial cursor position is" <data structure> in Screen.
Screen KEYS Global Data Structure
Screen KEYS Local Data Structure
Screen KEYS Data-Structure or Block
Screen KEYS Data-Structure Rep Block

20083 segment-1 "is virtually paired with" Segment-2.
Segment-1 RELATED-AS REAL-PAIRED-CHILD- SEGMENT TO *parent*-2
Segment-2 RELATED-AS VIRTUALLY-PAIRED-CHILD-SEGMENT WITH Segment-1<TO *parent*-1>
where:
parent-1 is the physical parent of Segment-1 *parent*-2 is the physical parent of Segment-2

20084 <concatenation> "consists of" <predicate type>.
Attribute Type HAS Attribute Type
Attribute Type HAS Relationship Type

20085 Segment "has" Field.
Segment CONTAINS Field

20086 Modelling Source "is information source of" <planning object>.
Modelling Source IMPLEMENTS Critical Assumption
Modelling Source IMPLEMENTS Critical Success Factor
Modelling Source IMPLEMENTS Data Collection
Modelling Source IMPLEMENTS Entity Type
Modelling Source IMPLEMENTS Function
Modelling Source IMPLEMENTS Goal
Modelling Source IMPLEMENTS Information Need
Modelling Source IMPLEMENTS Location
Modelling Source IMPLEMENTS Mechanism
Modelling Source IMPLEMENTS Organizational Unit
Modelling Source IMPLEMENTS Problem
Modelling Source IMPLEMENTS Process
Modelling Source IMPLEMENTS Project
Modelling Source IMPLEMENTS Subject Area

20087 Project "includes" <planning object>.
Project HAS Subject Area
Project HAS Data Collection
Project HAS Entity Type
Project HAS Function
Project HAS Mechanism
Project HAS Process
Project HAS Information-Need

20088 Organizational Unit "is responsible for" <planning object>.
Organizational Unit HAS Data Collection

Organizational Unit HAS Entity Type
Organizational Unit HAS Function
Organizational Unit HAS Mechanism
Organizational Unit HAS Process
Organizational Unit HAS Subject Area

20089 <planning object> "supports" <planning object>.

Data Collection SUPPORTS Critical Assumption
Data Collection SUPPORTS Critical Success Factor
Data Collection SUPPORTS Goal
Data Collection SUPPORTS Information Need
Entity Type SUPPORTS Critical Assumption
Entity Type SUPPORTS Critical Success Factor
Entity Type SUPPORTS Goal
Entity Type SUPPORTS Information Need
Function SUPPORTS Critical Assumption
Function SUPPORTS Critical Success Factor
Function SUPPORTS Goal
Function SUPPORTS Information Need
Mechanism SUPPORTS Critical Assumption
Mechanism SUPPORTS Critical Success Factor
Mechanism SUPPORTS Goal
Mechanism SUPPORTS Information Need
Process SUPPORTS Critical Assumption
Process SUPPORTS Critical Success Factor
Process SUPPORTS Goal
Process SUPPORTS Information Need
Subject Area SUPPORTS Critical Assumption
Subject Area SUPPORTS Critical Success Factor
Subject Area SUPPORTS Goal
Subject Area SUPPORTS Information Need

20090 Problem "affects" <planning object>.

Problem AFFECTS Critical Assumption
Problem AFFECTS Critical Success Factor
Problem AFFECTS Data Collection
Problem AFFECTS Entity Type
Problem AFFECTS Function
Problem AFFECTS Goal
Problem AFFECTS Information Need
Problem AFFECTS Location
Problem AFFECTS Mechanism
Problem AFFECTS Organizational Unit
Problem AFFECTS Problem
Problem AFFECTS Process
Problem AFFECTS Subject Area

```
20091 <planning object> "causes" Problem.
```
Critical Assumption CAUSES Problem
Critical Success Factor CAUSES Problem
Data Collection CAUSES Problem
Entity Type CAUSES Problem
Function CAUSES Problem
Goal CAUSES Problem
Information Need CAUSES Problem
Location CAUSES Problem
Mechanism CAUSES Problem
Organizational Unit CAUSES Problem
Problem CAUSES Problem
Process CAUSES Problem
Subject Area CAUSES Problem

```
20092 <activity> "is performed at" Location.
```
Function LOCATION Location
Process LOCATION Location

```
20093 Function "involves" Subject Area.
```
Function INFLUENCES Subject Area

```
20094 <implementation> "implements" <planning object>.
```
Data Collection IMPLEMENTS Entity Type
Data Collection IMPLEMENTS Subject Area
Mechanism IMPLEMENTS Function
Mechanism IMPLEMENTS Process

```
20095 Process "precedes" Process.
```
Process CAUSES Process
Mechanism CAUSES Mechanism

```
20096 <data object> "is available at" Location.
```
Entity Type LOCATION Location
Subject Area LOCATION Location

```
20097 Organizational Unit "cites" <enterprise concern>.
```
Organizational Unit AFFECTS Critical Assumption
Organizational Unit AFFECTS Critical Success Factor
Organizational Unit AFFECTS Goal
Organizational Unit AFFECTS Information Need
Organizational Unit AFFECTS Problem

```
20098 Project "addresses" Goal.
```
Project ASSIGNED-TO Goal/Critical-Assumption/ Critical Success Factor/Information Need/Problem

```
20099 <enterprise concern> "impacts" <enterprise concern>.
```
Critical Assumption INFLUENCES Critical Success Factor
Critical Success Factor INFLUENCES Goal
Information Need INFLUENCES Critical Success Factor
Information Need INFLUENCES Goal

```
20100 <planning object> "requires" <planning object>.
```
Goal RELIES-ON Goal
Project RELIES-ON Project

```
20101 Goal "solves" Problem.
```
Goal AFFECTS Problem

```
20102 <implementation> "replace" <implementation>.
```
Data Collection AFFECTS Data Collection
Mechanism AFFECTS Mechanism

```
20103 <implementation> "is maintained at" location.
```
Data Collection LOCATION Location
Mechanism LOCATION Location

```
20104 Mechanism "accesses" Data Collection.
```
Mechanism CALLS Data Collection

```
20105 Organizational Unit "is based at" Location.
```
Organizational Unit LOCATION Location
Project LOCATION Location

```
20106 Organizational Unit "coordinates" Organizational Unit.
```
Organizational Unit CONTAINS Organizational Unit

```
20107 Organizational Unit "sponsors" Project.
```
Organizational Unit SUPPORTS Project

```
20108 Segment "is based on" Segment.
```
Logical Segment CONTAINS Physical Segment

```
20109 Segment "is based on" Logical Child Segment.
```
Logical Segment CONTAINS Physical Segment

```
20111 Module "gets" Screen.
```
Module INPUTS Screen

```
20112 Module "puts" Screen.
```
Module OUTPUTS Screen

```
20115 PSB "contains" <PCB>.
```
PSB CONTAINS DB PCB
PSB CONTAINS TP PCB

20116 DB PCB "uses processing sequence of" Secondary Index DBD.
STRUCTURE PCB SECONDARY-SEQUENCE ON *Segment*
where: *Segment* is the root segment of the secondary-index database

20117 DB PCB "enacts" DBD.
DB PCB DATABASE DBD

20119 DB PCB "has" Secondary Index DBD.
DB PCB DATABASE DBD

20120 DB PCB "is sensitive to" Segment.
DB PCB SEGMENT Segment

20121 DB PCB "is sensitive to" Field.
DB PCB SENSITIVE-FIELD Field

20123 Screen Variable "uses" Module.
Global Data Structure USER-EXIT Module
Local Data Structure USER-EXIT Module
Data Structure Or Block USER-EXIT Module
Data Structure Rep Block USER-EXIT Module

20124 Screen "has get before" Module.
Screen USER-EXIT Module

20125 Screen "has get after" Module.
Screen USER-EXIT Module

20126 Screen "has put before" Module.
Screen USER-EXIT Module

20127 Screen "has put after" Module.
Screen USER-EXIT Module

20128 Module Call "uses" Parameter.
(Module CALLS Module) PASSING Parameter

20129 Module "fans in to" Module.
Module IEW-ADW-XREF Module

20131 Module "accesses" Record.
Module INPUTS File Record

20132 Module "accesses" Relation.
Module INPUTS Relation

20133 Module "accesses" Segment.
Module INPUTS Segment

```
20134 Secondary Index Source Segment "has" First Duplicate Data
Field.
```
Segment DUPLICATE-DATA-FIELD Field (1st Entry)

```
20135 Secondary Index Source Segment "has" Second Duplicate Data
Field.
```
Segment DUPLICATE-DATA-FIELD Field (2nd Entry)

```
20136 Secondary Index Source Segment "has" Third Duplicate Data
Field.
```
Segment DUPLICATE-DATA-FIELD Field (3rd Entry)

```
20137 Secondary Index Source Segment "has" Fourth Duplicate Data
Field.
```
Segment DUPLICATE-DATA-FIELD Field (4th Entry)

```
20138 Secondary Index Source Segment "has" Fifth Duplicate Data
Field.
```
Segment DUPLICATE-DATA-FIELD Field (5th Entry)

```
20139 secondary Index Source Segment "has" First Subsequence Field.
```
Segment SUBSEQUENCE-FIELD field (1st entry)

```
20140 Secondary Index Source Segment "has" Second Subsequence Field.
```
Segment SUBSEQUENCE-FIELD field (2nd entry)

```
20141 Source Segment "has" Third Subsequence Field.
```
Segment SUBSEQUENCE-FIELD field (3rd entry)

```
20142 Secondary Index Source Segment "has" Fourth Subsequence Field.
```
Segment SUBSEQUENCE-FIELD field (4th entry)

```
20143 Secondary Index Source Segment "has" Fifth Subsequence Field.
```
Segment SUBSEQUENCE-FIELD field (5th entry)

```
20144 Secondary Index Source Segment "has" First Search Field.
```
Segment SEARCH-KEY-FIELD field (1st entry)

```
20145 Secondary Index Source Segment "has" Second Search Field.
```
Segment SEARCH-KEY-FIELD field (2nd entry)

```
20146 Secondary Index Source Segment "has" Third Search Field.
```
Segment SEARCH-KEY-FIELD field (3rd entry)

```
20147 Secondary Index Source Segment "has" Fourth Search Field.
```
Segment SEARCH-KEY-FIELD field (4th entry)

```
20148 Secondary Index Source Segment "has" Fifth Search Field.
```
Segment SEARCH-KEY-FIELD field (5th entry)

20149 Field "is implemented by" <data structure> in Segment.
NOT MAPPED-IEW-ADW-XREF notes on special processing within the import facility.

20150 Segment-1 "is physically paired with" Segment-2.
Segment1 RELATED-AS
PHYSICALLY-PAIRED-CHILD-SEGMENT WITH Segment2
<TO *parent2*> Segment2 RELATED-AS
PHYSICALLY-PAIRED-CHILD-SEGMENT WITH Segment1
<TO *parent1*>
where:
parent1 is the physical parent of Segment1
parent2 is the physical parent of Segment2

20151 Segment-1 "indexes" Segment-2.
<HIDAM INDEX DATABASE *database-name*> SEGMENT segment1
where:
database-name is the name proposed for the DBD that contains Segment 1.

20152 Primary Index "has" Indexing Field.
<HIDAM DB INDEX> SEQUENCE-KEY Indexing-Field

20155 Screen "contains" Screen Object.
Screen CONTAINS Screen Object

20156 Consolidation Command "acts on" <object> or
<relationship type>.
Not mapped

20157 Segment "is based on" Virtual Logical Child Segment.
Logical Segment CONTAINS Segment

20158 Consolidation Matrix Row Object.
Not mapped

20159 Consolidation Matrix Column Object.
Not mapped.

20160 Segment "is first in" Dataset.
<Database> DATASETS PRIME Dataset CONTAINS Segment

20173 Subtype Set "has supertype" Entity Type.
Subtype Set SUPPORTS Entity Type

20174 Subtype Set "has subtype" Entity Type.
Subtype Set HAS Entity Type

```
20186 Data Type Set "consists of" Data Type.
```
Data Type Set IEW-ADW-XREF Data Type
INFORMATION 'string'

```
20199 Data Type Set "describes" <data structure>.
```
Data Type Set CONTAINS Global Data Structure
Data Type Set CONTAINS Local Data Structure

```
20200 Data Schema "contains" <data structure>.
```
Data Schema CONTAINS Global Data Record

```
20201 Data Schema "displays " <reference>.
```
Data Schema IEW-ADW-XREF Global Data Record

```
20202 Information Type "consists of" Information Type.
```
Information Type CONTAINS Information Type

```
20205 Value Restriction "maximum value is" Value.
```
Value Restriction AFFECTS Value

```
20206 Value Restriction "Range has minimum value" Value.
```
Value Restriction IMPLEMENTS Value

```
20211 Information Type "has" Value Set.
```
Information Type HAS Value Set

```
20212 Value Set "contains" Value.
```
Value Set CONTAINS Value

```
20213 Value Restriction "has" Value.
```
Value Restriction HAS Value

```
20231 Database "display" Reference.
```
Relational Schema IEW-ADW-XREF Global Data Record INFORMATION 'string'
Relational Schema IEW-ADW-XREF Local Data Record INFORMATION 'string'

```
20232 Schema "contains" Data Area.
```
Relational Schema IEW-ADW-XREF DB2 Table
Relational Schema IBW-ADW-XREF DB2 View

```
20234 Unique Identifier component.
```
Unique Identifier IEW-ADW-XREF Local Data Structure

```
20235 Foreign Key "has component" <data structure>.
```
Foreign Key IEW-ADW-XREF Local data Structure

```
20237 Data Record "has" Unique Id.
```
Global Data Record IEW-ADW-XREF Unique Identifier
Local Data Record IEW-ADW-XREF Unique Identifier

```
20240 Foreign Key "provides" References.
```
Foreign Key IEW-ADW-XREF Global Data Record
Foreign Key IEW-ADW-XREF Local Data Record

```
20241 Subset Information Type "is subset of" Information Type.
```
Subset Information Type DEPENDS-ON Information Type

```
20242 Application "consists of" Module/Program.
```
Application CONTAINS Module
Application CONTAINS Program

```
20452 Subset Information Type "has" Value restriction.
```
Subset Information Type HAS Value Restriction

```
20453 Data Record "has" Reference.
```
Global Data Record IEW-ADW-XREF Foreign Key
Local Data Record IEW-ADW-XREF Foreign Key

```
20481 DB2 Subsystem "Contains" DB2 Database.
```
DB2 Subsystem IEW-ADW-XREF DB2 Database

```
20482 DB2 Subsystem "Contains" DB2 Storage Group.
```
DB2 Subsystem IEW-ADW-XREF DB2 Storage Group

```
20483 DB2 Subsystem "Contains" DB2 Tablespace.
```
DB2 Subsystem IEW-ADW-XREF DB2 tablespace

```
20486 DB2 Tablespace "contains" DB2 Table.
```
DB2 table IN DB2 Tablespace

```
20487 DB2 Component contains.
```
DB2 Tablespace IEW-ADW-XREF DB2 Tablespace Partition
Relational Index IEW-ADW-XREF DB2 Index Partition

```
20494 DB2 Database "has default" DB2 Storage Group.
```
DB2 Database STOGROUP DB2 Storage Group

```
20500 DB2 Index "contains" <column>.
```
Relational Index IEW-ADW-XREF Local Data Structure
INFORMATION 'string'
Relational Index IEW-ADW-XREF Global Data Structure INFORMATION 'string'

```
20501 Unique Identifier "supports" Index.
```
Unique Identifier IEW-ADW-XREF Relational Index

```
20553 DB2 View "is based on" DB2 View/Table.
```
DB2 view FROM DB2 View
DB2 View FROM DB2 Table

```
20604 Data Type Set "Has Display Default" composition.
```
Data Type Set IEW-ADW-XREF Data Type Set

```
20605 Data Type Set "Has Storage Default" composition.
```
Data Type Set IEW-ADW-XREF Data Type Set

Appendix B

ADW/IEW Object-Type Properties Mapped Against Manager Products Repository Attributes

Certain properties are common to many ADW/IEW objects and associations. These have therefore been defined as common attributes, though the interface will only generate or reference them when it is appropriate. Such properties are:

30025 PURPOSE
30075 LAST-UPDATE
30078 CONTRACTED
30102 LAST-SAVE-DATE
30110 CREATION-DATE
30028 RELATIVE-POSITION

A special case is also the object type code 10014 (IEW-RELEASE-CODE), which is reserved for ASG's future use.

10000	Process	IEW-PROCESS
30000	Declared Root	DECLARED-ROOT Y/N
30004	Central Transform	CENTRAL-TRANSFORM Y/N
30005	Transaction Centre	TRANSACTION-CENTRE Y/N
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE" "
31002	Frequency	FREQUENCY" "
31003	Importance	IMPORTANCE " "

10000	Process	IEW-PROCESS
31004	Response Time	RESPONSE-TIME IM/IN/SA/OV
31005	System Support Type	SYSTEM-SUPPORT-TYPE RD/AD/IN/DE/OP

10002	External Agent	IEW-EXTERNAL-AGENT
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10003	ATTRIBUTE-TYPE	IEW-ATTRIBUTE-TYPE
30007	Minimum per Subject	MINIMUM-VALUES-PER-SUBJECT " "
30008	Maximum per Subject	MAXIMUM-VALUES-PER-SUBJECT " "
30009	Maximum per Value	MAXIMUM-PER-VALUE " "
30011	Name	ALIAS LOCAL NAME " "
30054	Attribute Type in Entity Mode Entity Type Description Relative Position	Not mapped
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10004	Sequential Process	IEW-SEQUENTIAL-PROCESS
30004	Central Transform	CENTRAL-TRANSFORM Y/N
30005	Transaction Centre	TRANSACTION-CENTRE Y/N
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30102	Last Save Date	LAST-SAVE-DATE " "
30110	Creation Date	CREATION-DATE "

10005	Data Store Process	IEW-DATASTORE-ACCESS
30013	Data Store Access Action	ACCESS-ACTION READ/CREATE/UPDATE/DELETE
30075	Last Update	LAST-UPDATE " "
30110	Creation Date	CREATION-DATE " "

10007	Entity Type	IEW-ENTITY-TYPE
30025	Purpose	PURPOSE FUNDAMENTAL/ASSOCIATIVE/ ATTRIBUTAIVE/OTHER
30028	Entity Type in Entity Model Entity Diagram Position	RELATIVE-POSITION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "

10007	Entity Type	IEW-ENTITY-TYPE
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "

10008	Data Flow	IEW-DATAFLOW
10008	Data Flow	IEW-DATAFLOW
30017	Flow Expression	FLOW-EXPRESSION " "
30070	Output Junction Position	IEW-ADW-XREF
30071	Input Junction Position	IEW-ADW-XREF
30074	Dataflow Position	IEW-ADW-XREF
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10009	Junction	IEW-JUNCTION

10010	Organizational Unit	IEW-ORGANIZATIONAL-UNIT
31053	Manager	MANAGER " "
31054	Level of Responsibility	LEVEL-OF-RESPONSIBILITY UP/MI/OP/SS/RD
31055	Scope	SCOPE E/I
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "

10012	Data Store	IEW-DATASTORE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10013	Subject Area	IEW-SUBJECT-AREA
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10014	ADW/IEW Release Code	IEW-RELEASE-CODE " "
		(Reserve for future use)

10015	Module	IEW-MODULE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30102	Last Save Date	LAST-SAVE-DATE " "
30110	Creation Date	CREATION-DATE " "
30323	Predefined	PREDEFINED Y/N
30325	GAMMA Product Name	ALIAS GAMMA "name"

10016	Global Data Type	IEW-GLOBAL-DATA-TYPE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "

10016	Global Data Type	IEW-GLOBAL-DATA-TYPE
30105	Format	ENTERED-AS "picture"
30110	Creation Date	CREATION-DATE " "
30119	Internal Length	HELD-AS form-description length
30120	External Length	REPORTED-AS form-description length
30307	Data Type Type	form description
30332	SQL Data Type	SQL-DATA-TYPE
30333	SQL Data Type Max Length	SQL-DATA-TYPE-MAX-LENGTH integer
30334	SQL Data Type Precision	SQL-DATA- TYPE- PRECISION integer
30335	Data Type Scale	DATA TYPE SCALE integer

10017	Screen	IEW-SCREEN
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30102	Last Save Date	LAST-SAVE-DATE " "
30110	Creation Date	CREATION-DATE " "
30121	MFS DIF/DOF Name	MFS-DIF-DOF-NAME name
30123	TP Monitor	TP-MONITOR " "
30124	Screen Width (number of Rows)	SCREEN-WIDTH integer
30125	Screen Depth (number of columns)	SCREEN-DEPTH integer
30149	Input Map Set Name	INPUT-MAP-NAME name
30150	Output Map Set Name	OUTPUT-MAP-NAME name
30153	BMS Input Map Name	BMS-INPUT-MAP-NAME name
30154	BMS Output Map Name	BMS-OUTPUT-MAP-NAME name
30155	MFS MID Name	MFS-MID-NAME name
30156	MFS MOD Name	MFS-MOD-NAME name
30325	GAMMA Product Name	ALIAS GAMMA "name"

10018	Parameter	IEW-PARAMETER
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "

30078	Contracted	CONTRACTED Y/N
30079	Formal Name	ALIAS FORMAL-NAME "name"
30080	Input/Output	INPUT-OR-OUTPUT "INPUT"/"OUTPUT"
30081	Control Flag/Data	CONTROL-FLAG-OR-DATA "CONTROL-FLAG"/DATA"
30082	Position	POSITION integer
30110	Creation Date	CREATION-DATE " "

10019	**Library**	**IEW-LIBRARY**
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10020	**Literal**	**See 10063**

10021	**Variable**	**See 10063**

10022	**Graphic Box**	**See 10063**

10024	**Graphic Horizontal Line**	**See 10063**

10025	**Graphic Vertical Line**	**See 10063**

10026	Location	IEW-LOCATION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
30213	Address	ADDRESS " "
30151	Location Type	LOCATION-TYPE RO/FC/BU/SI

10027	Project	IEW-PROJECT
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31057	Actual Start Date	ACTUAL START DATE " "
31058	Actual End Date	ACTUAL END DATE " "
31059	Scheduled Start Date	SCHEDULED START DATE " "
31060	Scheduled End Date	SCHEDULED END DATE " "
31061	Estimated Man-months	ESTIMATED-MAN-MONTHS integer
31062	Estimated Resource (dollars)	ESTIMATED-COST integer
31063	Priority	PRIORITY "H"/"M"/"L"/"integer"

10027	Project	IEW-PROJECT
31064	Ranking	RANKING integer
31065	Return on Investment	RETURN-ON-INVESTMENT integer
31066	Risk	RISK "H"/"M"/"L"/"integer"
31067	Status	STATUS CU/PL/AA/PA/UD
31068	Technical Complexity	TECHNICAL-COMPLEXITY " "

10028	Mechanism	IEW-MECHANISM
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
30214	Disposition Rationale	DISPOSITION-RATIONALE " "
31039	Completeness Rating	COMPLETENESS-RATING "H"/"M"/"L" "*integer*"
31040	Date of Last Modification	DATE-LAST-MODIFIED " "
31041	Disposition	DISPOSITION KE/IN/RE
31042	Documentation Rating	DOCUMENTATION-RATING "H"/"M"/"L"/"*integer*"
31043	Efficiency Rating	EFFICIENCY-RATING "H"/"M"/"L"/"*integer*"
31044	Flexibility Rating	FLEXIBILITY-RATING "H"/"M"/"L"/"*integer*"
31045	Maintainability Rating	MAINTAINABILITY-RATING "H"/"M"/"L"/"*integer*"
31046	Reliability Rating	RELIABILITY-RATING "H"/"M"/"L"/"*integer*"
31047	Response Time	RESPONSE-TIME IM/IN/SA/OV
31048	Status	STATUS CU/PL/AA/PA/UD
31049	Technology	TECHNOLOGY HI/RE/NT/DB/FI/MF/PA/AU/MA/ BO
31050	Usability Rating	USABILITY-RATING "H"/"M"/"L"/"*integer*"

10029	Data Collection	IEW-DATA-COLLECTION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
30214	Disposition Rationale	DISPOSITION-RATIONALE " "
31027	Accessibility Rating	ACCESSIBILITY-RATING "H"/"M"/"L"/"*integer*"
31028	Date of Last Modification	DATE-LAST-MODIFIED " "
31029	Disposition	DISPOSITION KE/IN/RE
31030	Documentation Rating	DOCUMENTATION-RATING "H"/"M"/"L"/"*integer*"
31031	Efficiency Rating	EFFICIENCY-RATING "H"/"M"/"L"/"*integer*
31032	Integrity Rating	INTEGRITY-RATING "H"/"M"/"L"/"integer"
31033	Maintainability Rating	MAINTAINABILITY-RATING "H"/"M"/"L"/"*integer*"
31034	Redundancy Rating	REDUNDANCY-RATING "H"/"M"/"L"/"*integer*"
31035	Security Rating	SECURITY-RATING "H"/"M"/"L"/"*integer*"
31036	Status	STATUS CU/PL/AA/PA/UD
31037	Technology	TECHNOLOGY HI/RE/NT/DB/FI/MF/PA/AU/MA/ BO

10030	Goal	IEW-GOAL
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31016	Begin Time	BEGIN-TIME " "
31017	Planning Horizon	PLANNING-HORIZON PE/ST/TA
31018	Ranking	RANKING integer

10031	Problem	IEW-PROBLEM
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
30215	Value Statement	VALUE-STATEMENT " "
31023	Begin Time	BEGIN-TIME " "
31024	Cause Category	CAUSE-CATEGORY OB/OR/PL/OP/ME/CU
31025	Ranking	RANKING integer

10032	Segent	SEGMENT LOGICAL/PHYSICAL
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30100	Segment Length SEGM/BYTES = (MAXIMUM)	Not required
30101	Segment Frequency SEGM/FREQ =	SEGMENT PHYSICAL RELATED-AS type ATTRIBUTES FREQUENCY integer
30110	Creation Date	CREATION-DATE " "
30116	Name	ALIAS IMS "*name*"
30167	Physical Child Pointer Type: DEBD SEGM/PARENT = (SNGL/DBLE)	Not supported
30168	Segment Length: DEDB SEGM/BYTES = (MINIMUM)	Not supported
30169	Dependent Segment Type: SEGM/TYPE = (DIR/SEQ)	Not mapped
30170	Data Manipulation Rules: SEGM/RULES =	Not mapped
30171	Number of Subset Pointers: SEGM/SSPTR =	Not mapped
30195	Physical Child Pointer Type: SEGM/PARENT = (SNGL/DBLE)	SEGMENT PHYSICAL RELATED-AS REAL-PAIRED POINTERS SINGLE-LOGICAL/ DOUBLE-LOGICAL

10032	Segent	SEGMENT LOGICAL/PHYSICAL
30196	LCHILD FID or LPCK Indicator: SEGM/PARENT = (VIRTUAL/PHYSICAL) where: VIRTUAL = DIRECT PHYSICAL = SYMBOLIC	SEGMENT PHYSICAL RELATED-AS REAL-PAIRED POINTERS SYMBOLIC/DIRECT
30197	Segment Length: SEGM/BYTES = (MINIMUM)	Not required
30198	H/HB/T/TB/NT Pointer Indicator: SEGM/POINTER = (H/HB/T/TB/NT) where: H = FORWARD-HIERARCHICAL HB = BACKWARD-HIERARCHICAL NT = NOTWIN T = SINGLE-TWIN TB = DOUBLE-TWIN	SEGMENT PHYSICAL RELATED-AS type ATTRIBUTES FORWARD-HIERARCHICAL/ BACKWARD-HIERARCHICAL/ SINGLE-TWIN/DOUBLE-TWIN/ NOTWIN
30199	LT/LTB Pointer Indicator: SEGM/POINTER = (LT/LTB) where: LT = FORWARD-LOGICAL LTB = BACKWARD-LOGICAL	SEGMENT PHYSICAL RELATED-AS REAL-PAIRED POINTERS FORWARD-LOGICAL/ BACKWARD-LOGICAL
30200	LP Pointer Indicator: SEGM/POINTER = (LP)	SEGMENT PHYSICAL RELATED-AS REAL-PAIRED POINTERS SYMBOLIC DIRECT If POINTERS SYMBOLIC is specified add DIRECT.
30201	CTR Pointer Indicator SEGM/POINTER = (CTR)	Not required

10032	Segent	SEGMENT LOGICAL/PHYSICAL
30202	Paired Pointer Indicator: SEGM/POINTER = PAIRED	Not mapped
30203	Insert, Delete, Replace Rules: SEGM/RULES = type (path type)	SEGMENT PHYSICAL RELATED-AS type RULES INSERT/DELETE/ REPLACE
30204	Update Insert Rules: SEGM/RULES = insert rules	SEGMENT PHYSICAL RELATED-AS type ATTRIBUTES INSERT-POSITION FIRST/LAST/HERE
30205	Edit/Compression Exit Routine: SEGM/COMPRTN = (routine name)	SEGMENT PHYSICAL RELATED-AS type ATTRIBUTES EDIT-COMPRESSION-NAME NAME "*module-name*"
30206	Data/Key Modify Indicator: SEGM/COMPRTN = (KEY/DATA) where: DATA = "ALL" KEY = NULL	SEGMENT PHYSICAL RELATED-AS type ATTRIBUTES EDIT-COMPRESSION EXIT "*module-name*" "ALL"
30207	Processing Control Indicator: SEGM/COMPRTN = (INIT)	Not mapped
30249	Key/Data Indicator for Logical Segment Source: SEGM/SOURCE = (KEY/DATA) where: KEY-ONLY = "KEY-ONLY" DATA = NULL	SEGMENT LOGICAL CONTAINS physical-segment IN physical-db KEY-ONLY

10032	Segent	SEGMENT LOGICAL/PHYSICAL
30250	Key/Data Indicator for Destination Parent in a Concatenated Segment: SEGM/SOURCE = (KEY/DATA for destination parent) where: KEY-ONLY = "KEY-ONLY" DATA = NULL	SEGMENT LOGICAL CONTAINS physical-segment IN physical-db, destination-parent KEY-ONLY
30325	GAMMA Product name	ALIAS GAMMA "*name*"

10033	DBD	HDAM/HIDAM/HISAM/ HSAM/INDEX/LOGICAL/ SECONDARY-INDEX DATA-BASE
30075	Last Update	LAST-UPDATE "*date*"
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30116	Name	ALIAS IMS "*name*"
30152	DL/1 Access Method	IMS-DATABASE HSAM/HISAM/HDAM/HIDAM/ S-INDEX/LOGICAL
30158	DEDB Randomizing Module Name: DEDB DBD/RENAME = (MODULE NAME)	Not supported
30159	DEDB Data Set Group DD Name: AREA/DD1	Not supported
30160	DEDB Storage Device Type: (AREA/DEVICE=)	Not supported
30161	Device Model Number:	DATASETS PRIME ddname AREA/MODEL DEVICE "*device* MODEL "MODEL"

10033	DBD	HDAM/HIDAM/HISAM/ HSAM/INDEX/LOGICAL/ SECONDARY-INDEX DATA-BASE
30162	DEDB Control Interval Size: (AREA/SIZE=)	Not supported
30163	DEDB Control Intervals /UOW: AREA/UOW = (TOTAL_CNTL_INTERVAL)	Not supported
30164	DEDB Control Intervals/ UOW Overflow: AREA/UOW = (OVFLW_CNTL_INTERVALS)	Not supported
30165	DEDB Total Allocated UOWs: AREA/ROOT = (TOTAL UOWs)	Not supported
30166	DEDB Overflow UOWs: AREA/ROOT = (OVERFLOW_UOWS)	Not supported
30186	Password Indicator: (DBD/PASSWD=)	PASSWORD
30187	Operating System Access Method: DBD/ACCESS = OPERATING SYSTEM	HDAM/HIDAM/HISAM ACCESS OSAM/ISAM/VSAM
30188	Primary Data Set DD Name: First DATASET/DD1 =	HIDAM/HDAM/HISAM DATASETS PRIME "*ddname*"
30189	Storage Device Type: First DATASET/DEVICE =	HDAM/HIDAM/HSAM/ HISAM/S-INDEX DATASETS PRIME ddname DEVICE "*device*"
30190	DEDB Storage Device Model Number: DATASET/MODEL =	Not supported
30191	Control Interval Size: First DATASET/SIZE = (Primary)	DATASETS PRIME ddname BUFFER "*size*"

10033	DBD	HDAM/HIDAM/HISAM/ HSAM/INDEX/LOGICAL/ SECONDARY-INDEX DATA-BASE
30192	Number of Cylinders to Scan: First DATASET/SCAN=	HDAM/HIDAM DATASETS PRIME ddname DEVICE device SCAN "*cylinders*"
30193	Free Block Frequency Factor: First DATASET/FRSPC = (FREE BLK FREQUENCY)	HDAM/HIDAM DATASETS PRIME ddname DEVICE device FREQUENCY-FREE-BLOCKS "*frequency*"
30194	Free Space Percentage Factor: First DATASET/FRSPC = (FREE SPACE %)	HDAM/HIDAM DATASETS PRIME ddname DEVICE device PERCENTAGE-FREE-SPACE
30242	Protect Index Indicator: DBD/ACCESS = (PROT/NOPROT)	SECONDARY-INDEX ACCESS PROTECTED/NOPROTECTION
30243	Overflow Data Set DD Name: First DATASET/OVFLW =	HISAM DATASETS PRIME ddname
30244	Primary Data Set Logical Record Length: First DATASET/RECORD= (PRIMARY)	OVERELOW "*ddname*" HISAM DATASETS PRIME ddname BLOCK count RECORD "*length*"
30245	Overflow Logical Record Length: First DATASET/ RECORD = (OVERFLOW)	HISAM DATASETS PRIME ddname OVERFLOW ddname BLOCK count RECORD "*length*"
30246	Overflow Control Interval Size: First DATASET/SIZE = (OVERFLOW)	HISAM DATASETS PRIME ddname OVERFLOW ddname BUFFER "*size*"
30247	Blocking Factor for HDAM/HIDAM: First BLOCK "*count*" (DATASET/BLOCK =)	DATASETS PRIME ddname
30252	Randomising Module Name: DBD/RMNAME = (MODULE NAME)	HDAM RANDOMISING-MODULE "*module-name*"

10033	DBD	HDAM/HIDAM/HISAM/ HSAM/INDEX/LOGICAL/ SECONDARY-INDEX DATA-BASE
30253	Number of Root Anchor Points: DBD/RMNAME = (ROOT_ANCHOR POINTS	HDAM RANDOMISING-MODULE module-name ANCHOR-POINTS "*number*"
30254	Maximum Relative Block Number: DBD/RMNAME = (MAX REL BLOCK NUM)	HDAM RELATIVE-BLOCK-MAXIMUM "*relative-block"*
30255	Maximum Bytes of Record to Store: DBD/RENAME = (MAX BYTES TO STORE)	HDAM INSERTION-BYTES-MAXIMUM "*bytes*"
30257	HISAM Primary Blocking Factor: First DATASET/BLOCK = (HISAM PRIMARY)	HISAM DATASETS PRIME ddname BLOCK "*count*"
30258	HISAM Overflow Blocking Factor First DATASET/BLOCK = (HISAM OVERFLOW)	HISAM DATASETS PRIME ddname OVERFLOW ddname BLOCK COUNT
30259	Index Primary Blocking Factor: First DATASET/BLOCK = (INDEX PRIMARY)	HIDAM DATASETS INDEX ddname BLOCK "*count*"
30260	Index Overflow Blocking Factor: First DATASETE/BLOCK = (INDEX OVERFLOW)	HIDAM DATASETS INDEX ddname OVERFLOW ddname BLOCK "*count*"
30295	DL1/DOS Index Indicataor: DBD/ACCESS = (DOS COMP)	HIDAM INDEX DOS-COMP
30331	Dataset Label: First DATASET LABEL	Not supported

l

10034	PSB	IEW-PSB
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30116	Name	ALIAS IMS "*name*"
30261	Compiler Language: PSBGEN/LANG =	LANGUAGE " "
30262	Maximum Qx Database Calls: PSBGEN/MAXQ =	MAXQ integer
30263	BMP/MSG to DL/1 Compatibility: PSBGEN/CMPAT	CMPAT YES/NO
30264	Maximum I/O Area Size: PSBGEN/IOASIZE =	IOASIZE integer
30265	Maximum Total Length of SSAs: PSBGEN/SSASIZE	SSA-SIZE integer
30266	IMS/VS Condition code: PSBGEN/IOEROPN = (condition code)	IOEROPN integer
30267	Wait for Operator Response Flag: PSBGEN/IOEROPM = (WTOR) PSBGEN/OLIC =	IOEROPM " "
30268	OLIC Authorization Code:	OLIC YES/NO

10038	Modeling Source	IEW-MODELING-SOURCE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31070	Interviewer	INTERVIEWER " "
31071	Source Date	SOURCE-DATE " "
31072	Source Type	SOURCE-TYPE DOCUMENT/STAKEHOLDER GROUP-SESSION/OTHER

10039	Critical Assumption	IEW-CRITICAL-ASSUMPTION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31007	Begin Time	BEGIN-TIME " "
31008	Ranking	RANKING integer
31009	Stability	STABILITY "H/M/L/999"

10040	Critical Success Factor	IEW-CRITICAL-SUCCESS-FACTOR
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31011	Achievability Rating	ACHIEVABILITY-RATING "H"/"M"/"L"/"*integer*"
31012	Begin Time	BEGIN-TIME " "
31013	Controllability Rating	CONTROLLABILITY-RATING "H"/"M"/"L"/"*integer"*
31014	Ranking	RANKING integer

10041	Information Need	IEW-INFORMATION-NEED
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "
31020	Begin Time	BEGIN-TIME " "
31021	Ranking	RANKING integer

10048	Local Data Type	IEW-LOCAL-DATA-TYPE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30105	Format	ENTERED-AS "*picture*"
30110	Creation Date	CREATION-DATE " "
30119	Internal Length	HELD-AS form-description "*length*"
30120	External Length	REPORTED-AS form-description length
30307	Data Type Type	form-description
30332	SQL Data Type	SQL-DATA-TYPE " "
30333	SQL Data Type Max.Length	SQL-DATA-TYPE-MAX-LENGTH integer
30334	SQL Data Type Precision	SQL-DATA-TYPE-PRECISION integer
30335	SQL Data Type Scale	SQL-DATA-TYPE-SCALE integer

10049	Global Data Structure	IEW-GLOBAL-DATA-STRUCTURE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30115	Purpose	PURPOSE VALUE/POINTER/TOKEN

30116	Local Name	ALIAS LOCAL-NAME "*name*"
30117	Synchronized Indicator	SYNCHRONIZED-INDICATOR Y/N

10051	**Data Structure Or Block**	**IEW-DATA-STRUCTURE-OR-BLOCK**
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30116	Name	ALIAS "*name*"

10052	**Data Structure Repetition Block**	**IEW-DATA-STRUCTURE-REP-BLOCK**
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30112	Minimum	MINIMUM "*integer*"
30113	Maximum	MAXIMUM "*integer*"
30116	Name	ALIAS "*name*"

10053	**Relation**	**IEW-RELATION**
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10053	Relation	IEW-RELATION
30116	Name	ALIAS "*name*"
30137	Index Name	IEW-INDEX member type
30325	GAMMA Product Name	ALIAS GAMMA "*name*"
30336	Physical Name: SQL Table Name	ALIAS SQL "*name*"

10054	File Record	IEW-FILE-RECORD
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30116	Name	ALIAS "*name*"
30325	GAMMA Product Name	ALIAS GAMMA "*name*"

10055	Relational Schema	IEW-RELATIONAL-DATABASE
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "

10056	File Database	IEW-FILE-DATABASE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "

10056	File Database	IEW-FILE-DATABASE
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10058	Function	IEW-FUNCTION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Creation Date	CREATION-DATE " "

10059	Field	IEW-FIELD
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

10060	DB PCB	PCB STRUCTURE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30285	Processing Option: PCB/PROCOPT =	processing-clause

10060	DB PCB	PCB STRUCTURE
30286	Maximum Concatenated Key Length: PCB/KEYLEN =	KEYLENGTH "*keylength*"
30287	Positioning Indicator: PCB/POS=	SINGLE/MULTI-POSITIONING

10061	TP PCB	PCB OUTPUT-MESSAGE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "
30279	Message Destination Terminal: PCB/LTERM=	LOGICAL-TERMINAL "*name*"
30280	Message Destination Trans-code: PCB/NAME =	TRANSACTION-CODE "*name*"
30281	Alternate PCB Usage Indicator: PCB/ALTRESP =	ALTERNATE-IO-RESPONSE
30282	Same Terminal Indicator: PCB/SAMETRM=	SAME-TERMINAL
30283	Dynamic Modification Indicator: PCB/MODIFY =	MODIFIABLE
30284	Send or Back-out Message Indicator: PCB/EXPRESS=	EXPRESS

10062	Consolidation Matrix	Not Mapped
30306	Association Type	
30329	Reversed	

10063	Screen Object	IEW-SCREEN-OBJECT
30126	Literal	LITERAL " "
30128	Type where: Type is: -10020 Literal -10021 Variable -10022 Graphic Box -10024 Graphic Horizontal Line -10025 Graphic Vertical Line	VARIABLE-TYPE " "
30129	Modified	MODIFIED Y/N
30130	Displayed	DISPLAYED " "
30131	Content	SCREEN-ATTRIBUTE " "
30132	Length	LENGTH integer
30133	Depth	DEPTH integer
30134	Row	ROW integer
30135	Column	COLUMN integer
30292	On-Screen Indicator	ON-SCREEN Y/N

10064	Consolidation Command	Not mapped
30110	Creation Date	
30326	Recruitment List	
30327	Consolidation Command Type	
30328	Object Type List	

10065	DBD Dataset (2nd to nth Occurrence)	HDAM/HIDAM/HISAM/ HSAM/SECONDARY-INDEX/ LOGICAL DATABASES
30110	Creation Date	
30188	Primary Data Set DD Name: DATASET/DD1	HDAM/HIDAM/HISAM DATASET PRIME "*ddname*"
30189	Storage Device Type: DATASET/DEVICE =	HDAM/HIDAM/HISAM DATASETS PRIME ddname DEVICE "*device*"
30190	DEDB Storage Device Model Number: DATASET/MODEL =	Not supported
30191	DEDB Control Interval Size: DATASET/SIZE = (PRIMARY)	Not supported
30192	Number of Cylinders to Scan: DATASET/SCAN=	HDAM/HIDAM DATASETS PRIME ddname DEVICE device SCAN "*cylinders*"
30193	Free Block Frequency Factor: DATASET/FRSPC = (FREE_BLK_FREQ)	HIDAM DATASETS PRIME ddname DEVICE device FREQUENCY-FREE-BLOCKS "*frequency*"
30194	Free Space Percentage Factor: DATASET/FRSPC = (FREE.SPACE %)	HIDAM DATASETS PRIME ddname DEVICE device PERCENTAGE-FREE-SPACE "*percent*"

10065	DBD Dataset (2nd to nth Occurrence)	HDAM/HIDAM/HISAM/ HSAM/SECONDARY-INDEX/ LOGICAL DATABASES
30243	Overflow Data Set DD Name: DATASET/OVFLW =	HISAM DATASETS PRIME ddname OVERELOW "*ddname*"
30244	Primary Data Set Logical Record Length: DATASET/RECORD = (PRIMARY)	HISAM DATASETS PRIME ddname BLOCK count RECORD "*length*"
30245	Overflow Logical Record Length: DATASET/RECORD = (OVERFLOW)	HISAM DATASETS PRIME ddname OVERFLOW ddname BLOCK count RECORD "*length*"
30246	Overflow Control Interval Size: DATASET/SIZE (OVERFLOW)	HISAM DATASETS PRIME ddname OVERFLOW ddname BUFFER "*size*"
30247	Blocking Factor for HDAM/HIDAM: DATASET/BLOCK =	DATASETS PRIME ddname DEVICE device BLOCK "*count*" DATASETS PRIME ddname BLOCK "*count*"
30257	HISAM Primary Blocking Factor: DATASET/BLOCK= (HISAM PRIMARY)	DATASETS PRIME ddname BLOCK "*count*"
30258	HISAM Overflow Blocking Factor: DATASET/BLOCK = (HISAM OVERFLOW)	DATASETS PRIME ddname OVERFLOW ddname BLOCK "*count*"
30331	Dataset Label:	Not supported

10080	Subtype Set	IEW-SUBTYPE-SET
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30116	Local Name	ALIAS LOCAL-NAME "*name*"

10080	Subtype Set	IEW-SUBTYPE-SET
30384	Covering Type	COVERING-TYPE COVERING/ NON-COVERING/DEFERRED
30385	Partitioning Type	PARTITIONING-TYPE EXCLUSIVE/INCLUSIVE/DEFERRED
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "

10085	Unique Identifier	IEW-UNIQUE-IDENTIFIER
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30338	Primary Key	PRIMARY-KEY Y/N
30998	Policy Maker	POLICY-MAKER
30999	Checked out	CHECKED-OUT

10086	Data Schema	IEW-DATA-SCHEMA
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER
30999	Checked out	CHECKED-OUT

10089	Data Type	IEW-DATA-TYPE
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30399	Character Set	CHARACTER-SET " "
30400	Variable Length	VARIABLE-LENGTH Y/N
30404	Precision	PRECISION number
30405	Fractional Precision	FRACTIONAL-PRECISION number
30409	Synchronization Indicator	SYNCHRONIZATION-INDICATOR LEFT/RIGHT/SYNC
30410	Blank When Zero	BLANK-WHEN-ZERO- Y/N
30411	COBOL Picture	ENTERED-AS PICTURE "*picture*"
30545	Uppercase	UPPERCASE-IND Y/N
30624	Must Fill	MUSTFILL-IND Y/N
30829	Data Type Gamma Usage Flag	GAMMA-USAGE-FLAG Y/N
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "

10092	Data Type Set	IEW-DATA-TYPE-SET
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30495	Generation Date	GENERATION-DATE "*date*"

10092	Data Type Set	IEW-DATA-TYPE-SET
30706	Domain Definition	DOMAIN-DEFINITION " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHECKED-OUT " "

10094	Information Type	IEW-INFORMATION-TYPE
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30462	Type	TYPE-2 " "
30702	Additive	ADDITIVE YES/NO/NULL
30703	Boolean	BOOLEAN YES/NO/NULL
30704	Concatenated	CONCATENATED YES/NO/NULL
30705	Relative	RELATIVE YES/NO/NULL
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10095	Value Set	IEW-VALUE-SET
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30112	Minimum Cardinality	MINIMUM " "
30113	Maximum Cardinality	MAXIMUM " "
30458	Fractional Precision	FRACTIONAL-PRECISION " "
30459	Precision	PRECISION " "
30460	Value Set Type	VALUE-SET-TYPE " "

10095	Value Set	IEW-VALUE-SET
30463	Value Type	VALUE-TYPE " "
30474	Domain Description	DOMAIN-DESCRIPTION " "
30754	Minimum Range	MINIMUM-RANGE " "
30755	Maximum Range	MAXIMUM-RANGE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10096	Value Restriction	IEW-VALUE-RESTRICTION
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30112	Minimum Cardinality	MAXIMUM " "
30113	Maximum Cardinality	MAXIMUM " "
30461	Type	TYPE-1 " "
30472	Description	IEW-DESCRIPTION " "
30718	Invalid Flag	INVALID-FLAG " "
30754	Minimum Range	MINIMUM-RANGE " "
30755	Maximum Range	MAXIMUM-RANGE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10099	Value	IEW-VALUE
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30116	Local Name	ALIAL LOCAL-NAME "*name*"
30473	Sequence Number	SEQUENCE-NUMBER " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10118	Global Data Record	IEW-GLOBAL-DATA-RECORD
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10119	Local Data Record	IEW-LOCAL-DATA-RECORD
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30116	Local Name	ALIAS LOCAL-NAME "*name*"
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "

10121	Relational Table	IEW-DB2-TABLE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30116	Name	ALIAS LOCAL-NAME "*name*"
30517	CWS Product Name	CWS-PRODUCT-NAME " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "
31100	Edit Proc Name	EDITPROC process
31101	Validation Proc Name	VALIDPROC process
31102	Table Audit	AUDIT NONE/CHANGES/ALL
31119	Estimated Number of Rows	CARDINALITY integer

10122	Relational View	IEW-DB2-VIEW
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30116	Local Name	ALIAS LOCAL-NAME "*name*"
30517	CWS Product Name	CWS-PRODUCT-NAME " "
30725	Check Option	WITH -CHECK-OPTION
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "

10123	Foreign Key	IEW-FOREIGN-KEY
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30116	Name	ALIAS LOCAL-NAME "*name*"
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "

10124	Subset Information Type	IEW-SUBSET-INFO-TYPE
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "

10127	Application	IEW-APPLICATION
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "
30075	Last Update	LAST-UPDATE " "

10142	DB2 Database	IEW-DB2-DATABASE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Created	CREATION-DATE " "
30336	Physical Name	ALIAS SQL "*name*"
30562	DB2 Buffer Pool	BUFFERPOOL BPO/BP1/BP2/BP32K
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "
31122	Reorganization Interval	REORGANIZATION-INTERVAL DAY/WEEK/MONTH/YEAR

10143	Relational Index	IEW-INDEX
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30336	Physical Name	ALIAS SQL "*name*"
30562	DB2 Buffer Pool	BUFFERPOOL BPO/BP1/BP2/BP32K
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "
31088	Cluster	CLUSTER
31089	DB2 Close	CLOSE YES/NO
31090	Dataset Password	DSETPASS password
31091	DB2 Sub-Pages	SUBPAGES 1/2/4/8/16
31092	VCAT	VCAT-l " "
31093	Primary Qty in KBytes	PRIQTY-1 " "
31094	Secondary Qty in KBytes	SECQTY-l " "
31095	DB2 Erase	ERASE-l Y/N
31096	DB2 Freepage	FREEPAGE-l Y/N
31097	Percentage of Free Space	PCTFREE-l " "
31122	Reorganizational Interval	REORGANIZATION-INTERVAL DAY/WEEK/MONTH/YEAR
31129	DB2 Unique Index	UNIQUE

10144	DB2 Index Partition	IEW-DB2-INDEX-PARTITION
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked Out	CHEDKED-OUT " "

10145	DB2 Storage Group	IEW-DB2-STOGROUP
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Created	CREATION-DATE " "
30336	Physical Name	ALIAS SQL "*name*"
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "
31092	VCAT	VCAT " "
31098	Volume ID	VOLUMES vol-id
31099	ICF Password	PASSWORD " "

10146	DB2 Subsystem	IEW-DB2-SUBSYSTEM
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Created	CREATION-DATE " "
30116	Name	ALIAS LOCAL-NAME "*name*"
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "

10147	DB2 Tablespace	IEW-TBSPACE
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30078	Contracted	CONTRACTED Y/N
30110	Created	CREATION-DATE " "
30336	Physical Name	ALIAS SQL "*name*"
30558	Tablespace Type	TABLESPACE-TYPE SIMP/SEG/PART
30562	DB2 Buffer Pool	BUFFERPOOL BP0/BP1/BP2/BP32K
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHECKED-OUT " "
31089	DB2 Close	CLOSE YES/NO
31090	Dataset Password	DSETPASS password
31092	VCAT	VCAT-l " "

10147	DB2 Tablespace	IEW-TBSPACE
31093	Primary Qty in KBytes	PRIQTY-1 " "
31094	Secondary Qty in KBytes	SECQTY-1 " "
31095	DB2 Erase	ERASE-1 Y/N
31096	DB2 Freepage	FREEPAGE-1 Y/N
31097	Percentage of Free Space	PCTFREE-1 " "
31104	Tablespace Locksize	LOCKSIZE ANY/PAGE/TABLE/ TABLESPACE
31105	Pages per Segment	SEGSIZE-1 " "

10148	DB2 Tablespace Partition	IEW-DB2-TABLESPACE-PARTITION
30075	Last Update	LAST-UPDATE " "
30110	Created	CREATION-DATE " "
30998	Policy Maker	POLICY-MAKER " "
30999	Checked out	CHEDKED-OUT " "

20044	RELATIONSHIP-TYPE	IEW-RELATIONSHIP-TYPE
30034	From-To Name	FROM-TO-NAME " "
30035	Left-to-Right Minimum Cardinality	LR-MIN " "
30036	Left-to-Right Maximum Cardinality	LR-MAX " "
30037	To-From Name	TO-FROM-NAME " "
30038	Right-to-Left Minimum Cardinality	RL-MIN " "
30039	Right-to-Left Maximum Cardinality	RL-MAX " "

20044	RELATIONSHIP-TYPE	IEW-RELATIONSHIP-TYPE
30073	Relationship Type in Entity Model Entity Diagram Position	RELATIVE POSITION " "
30075	Last Update	LAST-UPDATE " "
30076	Definition	DESCRIPTION " "
30077	Comments	COMMENT " "
30110	Creation Date	CREATION-DATE " "

Appendix C

ADW/IEW Member Type Naming Prefixes

This table lists the member name prefixes used by each member type:

Member Type	Prefix
IEW-APPLICATION	AP-
IEW-ATTRIBUTE-TYPE	AT-
IEW-CRITICAL-ASSUMPTION	CA-
IEW-CRITICAL-SUCCESS-FACTOR	CF-
IEW-DATA-COLLECTION	DK-
IEW-DATA-SCHEMA	DSC-
IEW-DATA-STRUCT-REP-BLOCK	DR
IEW-DATA-STRUCTURE-OR-BLOCK	DO
IEW-DATA-TYPE	IDT
EW-DATA-TYPE-SET	DTS
IEW-DATAFLOW	IFN
EW-DATASTORE	DT
IEW-DATASTORE-ACCESS	DA
IEW-DB2-DATABASE	DDB
IEW-DB2-INDEX-PARTITION	DIP
IEW-DB2-STOGROUP	DSG

Member Type	Prefix
IEW-DB2-SUBSYSTEM	DSS
IEW-DB2-TABLE	RB
IEW-DB2-TABLESPACE-PARTITION	DTP
IEW-DB2-VIEW	RV
IEW-ENTITY-TYPE	NT
IEW-EXTERNAL-AGENT	EA
IEW-FIELD	FD
IEW-FILE-DATABASE	FF
IEW-FILE-RECORD	FR
IEW-FOREIGN-KEY	FK
IEW-FUNCTION	FU
IEW-GLOBAL-DATA-RECORD	GDR
IEW-GLOBAL-DATA-STRUCTURE	GD
IEW-GLOBAL-DATA-TYPE	GT
IEW-GOAL	GO
IEW-INDEX	I2
IEW-INFORMATION-NEED	IN
IEW-INFORMATION-TYPE	ITY
IEW-JUNCTION	JC
IEW-LIBRARY	LY
IEW-LOCAL-DATA-RECORD	LR
IEW-LOCAL-DATA-STRUCTURE	LS
IEW-LOCAL-DATA-TYPE	LI

Member Type	Prefix
IEW-LOCATION	LL
IEW-MECHANISM	ME
IEW-MFS-SCREEN	MC
IEW-MODELLING-SOURCE	MS
IEW-MODULE	IM
IEW-MODULE -DATA-AREA	MDA
IEW-ORGANIZATIONAL-UNIT	IO
IEW-PARAMETER	PI
IEW-PROBLEM	PM
IEW-PROCESS	PO
IEW-PROJECT	PE
IEW-PROGRAM	P2
IEW-PSB	PS
IEW-RELATION	RL
IEW-RELATIONAL-DATABASE	D2
IEW-RELATIONSHIP-TYPE	RT
IEW-REPORT	R2
IEW-SCREEN	SN
IEW-SCREEN-OBJECT	SO
IEW-SEQUENTIAL-PROCESS	SQ
IEW-SUBJECT-AREA	SJ
IEW-SUBSET-INFO-TYPE	SIT
IEW-SUBTYPE-SET	STS

Member Type	Prefix
IEW-TBSPACE	T2
IEW-UNIQUE-IDENTIFIER	UI
IEW-VALUE	VA
IEW-VALUE-RESTRICTION	VR
IEW-VALUE-SET	VT

Note:

The member name prefixes used by IMS member types are as supplied in the MDRIM (UDS-TABLE DU016).

Index

A

ADW EXPORT command 33
  syntax 36
ADW IMPORT command 18
  syntax 18
ADW IMPORT commands 8
AI.EXP associations file 8, 27
ALIAS clauses 4
analysis workstation member types 48
ASSOC file 10
associations file 8, 34

C

codes supported
  association codes 155
common clauses 152
CONTROL
  NEW-ALIASES command 5
  UDR command 4
CONTROL NEW-ALIASES command 5
conventions page vii

D

data types 22
database support
  IMS 23
dataflow processing 24
design workstation member types 48

E

export 7
extracting information onto the WBTA 10

I

IEW EXPORT command
  syntax 33
IEW IMPORT command 7, 18
  syntax 18
IEW TI long properties file 34
iew_ export_drop_ kdl variable 42
iew_ export_obj_token variable 43
iew_adw_currency variable 27
iew_attrib_name variable 30
iew_data_delim variable 27
iew_data_type(x) variables 28
iew_db2_name variable 32
iew_dmr_form_desc(x) variables 28
iew_export_ ass_token variable 43
iew_export_ print_option variable 38
iew_export_alias_supp variable 38
iew_export_alias_type variable 38
iew_export_crdate_supp variable 38
iew_export_data_delim variable 41
iew_export_data_type(x) variables 41
iew_export_ddname(n) variables 39
iew_export_dmr_form-desc(x) variable 41
iew_export_file_blksze(n) variables 40
iew_export_file_format(n) variables 40
iew_export_file_lrecl(n) variables 41
iew_export_file_supp variable 37
iew_export_gdt_suffix variable 42
iew_export_kdl variable 37
iew_export_lock_key variable 38
iew_export_lsdate_supp variable 38
iew_export_mess_supp variable 38
iew_export_sepa variable 42
iew_export_string_del variable 38
iew_export_update supp variable 38
iew_export_xref variable 42
iew_export_xref_dstruct variable 42
iew_ext_ddname(n) variables 27
iew_ignore_rel variable 31–32
iew_import_alias variable 27
iew_ims_mbr_type variable 33
iew_ims_prefix variable 33
iew_ims_rel variable 32
iew_ims_type variable 33
iew_index_type variable 42
iew_ix_col_clause variable 27
iew_ix_tab_expand variable 27
iew_make_alias variable 30
iew_mpr_currency variable 27
iew_object_prefix(x) variables 28
iew_object_type(x) variables 28

iew_pop_from_option variable 26
iew_pop_print_option variable 26
iew_pop_rollback_option variable 26
iew_pre_cc_preserve variable 26
iew_pre_delim variable 27
iew_pre_func variable 26
iew_pre_print_option variable 26
iew_pre_user_name variable 26
iew_pre_user_option variable 26
iew_rel_mem(n) variables 31
iew_rel_name variable 32
iew_rel_name(45) variable 31
iew_rel_prefix(45) variable 31
iew_tab_tab_expand variable 27
iew_tbspce_type variable 42
iew_xref variable 28
iew_xref_dstruct variable 28
iew_xtab_002 variable 44
iew_xtab_003 variable 45
iew_xtab_004 variable 43
IEW-ADW-XREF clause 151
IEW-ADW-XREF-DSTRUCT clause 151
IEWAI associations file 34
IEW-Application member type 51
IEW-ATTRIBUTE-TYPE member type 48
IEW-CRITICAL-ASSUMPTION member type 48
IEW-CRITICAL-SUCCESS-FACTOR member type 48
IEW-DATA-COLLECTION member type 48, 55
IEW-DATAFLOW member type 48
IEW-DATA-SCHEMA member type 48
IEW-DATASTORE member type 48
IEW-DATASTORE-ACCESS member type 48
IEW-DATA-STRUCT-REP-BLOCK member type 48
IEW-DATA-STRUCTURE-OR-BLOCK member type 48
IEW-DATA-TYPE member type 48
IEW-DATA-TYPE-SET member type 48
IEW-DB2-DATABASE member type 48
IEW-DB2-INDEX-PARTITION member type 48
IEW-DB2-STOGROUP member type 48
IEW-DB2-SUBSYSTEM member type 48
IEW-DB2-TABLE member type 48
IEW-DB2-TABLESPACE-PARTITION member type 48
IEW-DB2-VIEW member type 48
IEW-ENTITY-TYPE member type 48
IEW-EXTERNAL-AGENT member type 48
IEW-FIELD member type 48
IEW-FILE-DATABASE member type 48
IEW-FILE-RECORD member type 48
IEW-FOREIGN-KEY member type 49
IEW-FUNCTION member type 49
IEW-GLOBAL-DATA-RECORD member type 49
IEW-GLOBAL-DATA-STRUCTURE member type 49
IEW-GLOBAL-DATA-TYPE member type 49
IEW-GOAL member type 49
IEW-INDEX member type 49
IEW-INFORMATION-NEED member type 49
IEW-INFORMATION-TYPE member type 49
IEW-JUNCTION member type 49
IEW-LIBRARY member type 49
IEW-LOCAL-DATA-RECORD member type 49
IEW-LOCAL-DATA-STRUCTURE member type 49
IEW-LOCAL-DATA-TYPE member type 49
IEW-LOCATION member type 49
IEW-MECHANISM member type 49
IEW-MFS-SCREEN member type 49
IEW-MODELLING-SOURCE member type 49
IEW-MODULE member type 49
IEW-MODULE-DATA-AREA member type 49
IEWOI object file 34
IEW-ORGANISATIONAL-UNIT member type 49
IEW-PARAMETER member type 49
IEWPI short properties file 34
IEW-PROBLEM member type 49
IEW-PROCESS member type 49
IEW-PROGRAM member type 49
IEW-PROJECT member type 49
IEW-PSB member type 50
IEW-RELATIONAL-DATABASE member type 50
IEW-RELATIONSHIP-TYPE member type 50
IEW-REPORT member type 50
IEW-SCREEN member type 50
IEW-SCREEN-OBJECT member type 50
IEW-SEQUENTIAL-PROCESS member type 50
IEW-SUBJECT-AREA member type 50

IEW-SUBJECT-INFO-TYPE member type 50
IEW-SUBTYPE-SET member type 50
IEW-TBSPACE member type 50, 126
IEW-UNIQUE-IDENTIFIER member type 50, 129
IEW-VALUE member type 50, 130
IEW-VALUE-RESTRICTION member type 129
IEW-VALUE-SET member type 130
import 7
IMS data support 23
IMS DATABASE HSAM member type 138
IMS database support 23
IMS-DATABASE HDAM member type 50
IMS-DATABASE HIDAM member type 50
IMS-DATABASE HISAM member type 50
IMS-DATABASE HSAM member type 50
IMS-DATABASE INDEX member type 140
IMS-DATABASE LOGICAL member type 50
installation 3

L

Locking members 34
Logical unit of work 17
long properties file 8, 34
LUW 17

M

member types 47
MPDYIITAB0
EXECUTIVE-ROUTINE 26
MPDYIITAB1
EXECUTIVE-ROUTINE 33
MPDYIITAB2
EXECUTIVE-ROUTINE 30
MPDYIITAB3
EXECUTIVE-ROUTINE 31
MPDYIITAB4
EXECUTIVE-ROUTINE 33
MPDYIXTAB0
EXECUTIVE-ROUTINE 34
MPDYIXTAB1
EXECUTIVE-ROUTINE 43
MPDYIXTAB2
EXECUTIVE-ROUTINE 44
MPDYIXTAB3
EXECUTIVE-ROUTINE 45

N

Names
how exported member names are derived 35
how imported member names are derived 19

O

OBJECT file 10
object instances file 19
OI 19
OI.EXP object instance file 34

P

PCB member type 142
PI.EXP short properties file 8
planning workstation member types 48
previewing proposed members 11
PROP file 10
property codes supported 171

R

RADD command 12
reconciliation report 12
reconciling proposed members 9
relationship types 19
relative positions 25
resetting the WBTA 18
RIGN command 12
RREN command 12
RREP command 12
RUPD command 12

S

SEGMENT INDEX-POINTER member type 145
SEGMENT LOGICAL member type 147
SEGMENT PHYSICAL member type 147
short properties file 10, 34

T

tailoring
export 7
import 7
TEXT file 10
TI.EXP long properties file 34
tokens 43
ToolSet SERVICES 1

U

UDR clauses 4

W

WBTA 18
Workbench Translation Area 8