ASG Software Solutions

ASG-Manager Products™ Oracle Interface Reference Guide

Version 2.5 for Oracle Version 7.0
Publication Number: MPR0400-25-ORC
Publication Date: December 2000

The information contained herein is the confidential and proprietary information of Allen Systems Group, Inc. Unauthorized use of this information and disclosure to third parties is expressly prohibited. This technical publication may not be reproduced in whole or in part, by any means, without the express written consent of Allen Systems Group, Inc.

© 1998-2002 Allen Systems Group, Inc. All rights reserved.
All names and products contained herein are the trademarks or registered trademarks of their respective holders.

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:

© 2002 Allen Systems Group, Inc.
All names and products are trademarks or registered trademarks of their respective holders.

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 Installation and Administration 1

Installation Datasets 1
Enabling the ORACLE-RIM. 1
Handling of User Defined Relationships (UDRs). 2
Export Exit MPDY12PROF 5
Import Exits MPDY42DFLT and MPDYWTDFLT. 5

Installing the Oracle Components 6
Loading the Datasets in the ORACLE Environment 6
Creation of Oracle System Table Views. 7

Modifying the Repository Information Model (RIM) 8

Miscellaneous. 8
MethodManager Generation 8
Retained Procedures 8
Modifying Procedures. 8

2 Oracle Export 9

Oracle Tailoring 9
Oracle Generation Profile 9

Oracle Column Specification 9
Column Types 9
Additional Column Attributes. 10

The ORACLE CREATE Command 10
Generating SQL COMMENT ON and LABEL ON Statements 11
Taking User Exits 11
Specifying an Owner. 12
Specifying a Location 12
Expanding Nested Data Structures 12

Suppressing Referential Integrity 13
Creating Primary Key Indexes 13
Replacing Existing Objects 14
Creating Public Objects. 15
Creating Package Bodies. 15
Name Editing Options. 15
Output Generation Options 15
Syntax of the ORACLE CREATE Command 16

The ORACLE ALTER Command 18
Modifying Columns of a Table 19
Adding Columns to a Table 19
Adding or Dropping Referential Constraints on a Table 20
Adding or Dropping a Primary Key on a Table 21
Adding, Modifying or Dropping a Validation Routine on a Table. 22
Adding or Modifying the Auditing Option on a Table 22
Expanding Nested Data Structures 22
Specifying an Owner of a Table 23
Specifying a Location of a Table 23
Taking User Exits 23
Output Generation Options 24
Name Editing Options. 24
Syntax of the ORACLE ALTER Command. 24

The ORACLE DROP Command. 26
The Impact Analysis Report 27
Tailoring the Impact Analysis Report 27
Example of the structure of the Impact Analysis Report 28
Dropping Associated Objects 28
Taking User Exits 29
Output Generation Options 29
Syntax of the ORACLE DROP Command 30

The ORACLE GRANT/REVOKE Command 31
Specifying an Owner of an Object 31
Expanding Nested Data Structures 32
Changing Privileges on an Expanded Range of Users 32
Specifying the Grantor of a Privilege 33
Taking User Exits 33
Name Editing Options. 33
Output Generation Options 33
Syntax of the ORACLE GRANT/REVOKE Commands 34

The ORACLE SYNONYM Command 35
Public Synonyms. 36
Name Editing Options. 36
Taking User Exits 36
Specifying an Owner. 36

Output Generation Options 37
Generating the Last Synonyms in a ORACLE-USER Member 37
Syntax of the ORACLE SYNONYM Command 37

General Options 38
Output Generation Options 38
Name Editing Options 39

3 Oracle Import Interface 41

Performing the Import 42
Extracting Information from the Oracle System Tables 42
Transferring the Spool Files into the Host environment 46
Importing into the Manager Products Environment 46

4 Repository Member Types 51

ORACLE-CLUSTER 51
ORACLE-DATABASE 54
ORACLE-DBLINK 55
ORACLE-FILE 56
ORACLE-INDEX 56
ORACLE-PACKAGE 58
ORACLE-PRIVILEGE 59
ORACLE-PROFILE 63
ORACLE-ROLE 64
ORACLE-SCHEMA 65
ORACLE-SEQUENCE 66
ORACLE-SNAPLOG 66
ORACLE-SNAPSHOT 68
ORACLE-TABLE 71
ORACLE-TABLESPACE 74
ORACLE-TRIGGER 75
ORACLE-USER 77
ORACLE-VIEW 78

Appendix
Oracle Member Types 81

Overview of the Supported Common Clauses 81

Overview of the Supported Oracle Member Types 81
ORACLE-CLUSTER (Prefix ORCL-) 82
ORACLE-DATABASE (Prefix ORDB-) 82
ORACLE-DBLINK (Prefix ORDL-) 83
ORACLE-FILE (Prefix ORFI-) 83
ORACLE-INDEX (Prefix ORIX-) 84
ORACLE-PRIVILEGE (Prefix ORPV-) 84
ORACLE-PROFILE (Prefix ORPR-) 85
ORACLE-ROLE (Prefix ORRL-) 85
ORACLE-SEQUENCE (Prefix ORSQ-) 86
ORACLE-SNAPSHOT (Prefix ORSN-) 86
ORACLE-SNAPLOG (Prefix ORSL-) 87
ORACLE-TABLE (Prefix ORTA-) 88
ORACLE-TABLESPACE (Prefix ORTS-) 89
ORACLE-TRIGGER (Prefix ORTG-) 89
ORACLE-USER (Prefix ORUS-) 90
ORACLE-VIEW (Prefix ORVI-) 90
ITEM (Prefix IT-) 90

Index 93

Preface

This *ASG-Manager Products Oracle Interface Reference Guide* provides information for ASG-Manager Products (herein called Manager Products) users in Oracle environments. This publication is for both MethodManager (MMR) and Base-Line Technology (BLT) users.

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, "Installation and Administration," provides installation information and administrative functions for the ASG-Manager Products Oracle Interface.
- Chapter 2, "Oracle Export," provides information for the Oracle Export Interface and Oracle commands.
- Chapter 3, "Oracle Import Interface," provides information for the Oracle Import Interface.
- Chapter 4, "Repository Member Types," describes the syntax of the member types in which you document the objects, locations, and security system of an Oracle database.

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:

```
▶──COMMAND──────────────────────────────────────────────▶
```

A required keyword appears on the main path:

```
▶──COMMAND──KEYWORD──────────────────────────────────────▶
```

An optional keyword appears below the main path:

```
▶──COMMAND──┬──────────────────────────────────────────┬▶
            └──KEYWORD──┘
```

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

```
▶──COMMAND──┬─KEYWORD1─┬──────────────────────────────▶
            │ KEYWORD2 │
            └─KEYWORD3─┘
```

Or

```
▶──COMMAND──┬─KEYWORD1─┬──────────────────────────────▶
            │─KEYWORD2─│
            └─KEYWORD3─┘
```

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

```
▶──COMMAND──┬──────────────────────────────────────────┬▶
            │──KEYWORD1──┬──────────────────────────────┬
            └──KEYWORD2──┘
```

Convention	Represents

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:

```
                    <<<<<<<<
►──────── COMMAND ────variable ──────────────────────────────►
```

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:

```
                    <<< , <<
►──────── COMMAND ──── variable ──────────────────────────────►
```

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:

```
                                <<<<<<<<<<<<<<<<
►──────── COMMAND──────────┬──KEYWORD1──┬──────────────────────►
                           └──KEYWORD2──┘
```

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 maybe specified, but they are all optional:

```
                                <<<<<<<<<<<<<<<<
►──────── COMMAND──────────┬──KEYWORD1──┬──────────────────────►
                           └──KEYWORD2──┘
```

Allen Systems Group, Inc. uses these conventions in 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|<u>N</u>).

1 Installation and Administration

Installation Datasets

The Oracle Interface is delivered as part of the normal installation tape.

These datasets are referenced during installation:

Dataset	Description
MP.COM.UNLOAD	MPAID-UNLOAD of some Command-Members.
MP.CORP	Source of customizable EXECUTIVE routines. After your customization changes you CONSTRUCT all procedures to the MPAID.
MP.SOURCE **Files with name OR7*******	ORACLE SQL*Plus scripts which serve to exploit the Oracle system tables and to output ASCII report files containing information about Oracle objects.
BLT: **MP.UDS**	Status UDS2500 contains the member type group UG-ORACLE which includes all other needed definitions.
MMR: **MP.MMR.ADMIN**	Status MDRIM2500 contains the member type group UG-ORACLE which includes all other needed definitions.

Decide to use either the MMR- or the BLT-RIM datasets. EXPORT and IMPORT can be installed independently, but both use the same RIM-objects.

Enabling the ORACLE-RIM

It is necessary to add the RIM definition to the relevant hierarchy to work successfully with the Oracle interface. Therefore add the member type group UG-ORACLE to the HIERARCHY member.

Handling of User Defined Relationships (UDRs)

The Oracle interface uses three User Defined Relationships (UDRs) and the User Defined Sub-Relationship (UDRS). By default the UDRs 1, 2, 3 and the UDRS are used with these names:

UDR1 = LOGFILE

UDR2 = DATAFILE

UDR3 = CLUSTER-FILE

UDRS = CLUSTER-COLS

It is necessary to rename these UDRs with this CONTROL UDR command:

```
CONTROL UDR RELATIONSHIPS    LOGFILE
                             ,DATAFILE
                             ,CLUSTER-FILE
               SUB-RELATIONSHIP CLUSTER-COLS;
```

If UDR1, UDR2, UDR3, and UDRS are already in use you will not be able to rename these UDRs in this way.

Tailoring

This section describes how to use other UDRs for the Oracle interface.

During the loading of MP.CORP a small executive routine is created in status CORP2500 with dictionary name ORASETUDR and MPAID name SETUDR. This executive routine allows you to define the UDRs which can be used by the Oracle interface. If the default UDRs of the Oracle interface (UDR1, UDR2, UDR3, UDRS) are available, this executable must *not* be run.

After constructing this executive routine invoke it with this syntax:

```
SETUDR n n n [udrs];
```

where *n n n* specifies which UDRs will be used by the ORACLE interface. *n n n* must be integers between 1 and 9 (number of the UDR) in ascending order. These integers set the UDRs for LOGFILE, DATAFILE, and CLUSTER-COLS on other UDR numbers.

For example, if you want the Oracle interface to use the UDR4, UDR5, and UDR7 instead of UDR1, UDR2, and UDR3 execute this routine:

```
SETUDR 4 5 7;
```

The SETUDR routine changes some ALLOW/DISALLOW entries in the respective definitions of the Oracle member types. Therefore the routine must be performed before the CONSTRUCT UDS.

If the subrelationship UDRS is already in use and a renaming in CLUSTER-COLS is not possible, you can use a fourth parameter to specify the name for the subrelationship. This parameter needs to be defined for the Oracle interface. If there is a user defined name for the UDRS, this name must be specified as the fourth parameter. If the subrelationship has no specific name in the users environment specify the string UDRS as the fourth parameter.

For example, if you do not want the Oracle interface to use the user defined subrelationship under a user specific name, but under the default name UDRS instead, execute this routine:

```
SETUDR 4 5 7 UDRS;
```

Caution! The executive SETUDR can be performed only *once*. This executive changes the source code in a way that a second execution cannot operate correctly. ASG recommends using this executive *carefully*.

Note:

If the Oracle default name CLUSTER-COLS is not used for the subrelationship, in all members who use such subrelationships the own name for the subrelationship must be used instead of the keyword CLUSTER-COLS. This concerns members from types ORACLE-TABLE and ORACLE-SNAPSHOT.

The subrelationship default name of the Oracle interface can be replaced in this way. But the other UDRs of the Oracle interface must be named with the specific names of the Oracle interface. This must be performed with the command CONTROL UDR. If the default UDRs of the Oracle interface and their names are not changed (no execution of SETUDR), than the CONTROL UDR command must be executed in the way described above:

```
CONTROL UDR RELATIONSHIPS   LOGFILE
                            ,DATAFILE
                            ,CLUSTER-FILE
         SUB-RELATIONSHIP   CLUSTER-COLS;
```

The names LOGFILE, DATAFILE, and CLUSTER-FILE are assigned to the UDRs depending from the position in the command and must be specified in this order. If these names are not assigned to UDR1, UDR2, and UDR3, but later UDRs instead, it is necessary to list all previous UDRs and to state these names at the correct position. If you want to use UDR3, UDR4, and UDR5 for the Oracle interface and if you want to use the subrelationship under the name UDRS instead of the name CLUSTER-COLS, and if you have defined this by executing SETUDR, the CONTROL UDR command must be performed with these parameters:

```
CONTROL UDR RELATIONSHIPS  <name of UDR1>
                           ,<name of UDR2>
                           ,LOGFILE
                           ,DATAFILE
                           ,CLUSTER-FILE;
```

where *<name of UDR1>* and *<name of UDR2>* must specify the names under which these UDRs are used in their own dictionary environment. If no user defined names are defined for these UDRs, the strings UDR1 and UDR2 have to be specified at these positions. The specification of the user defined subrelationship can be omitted, if this subrelationship is applied under the default name UDRS and no user defined name for this subrelationship has to be defined.

To use the correct expression of the CONTROL UDR command during the execution of the SETUDR routine, a skeleton for the CONTROL UDR command is generated. This skeleton is stored as a user member with the name CONTROLUDR. This skeleton contains the Oracle specific UDR names at the correct positions.

For example, the routine SETUDR is executed with these parameters:

```
SETUDR 4 5 7 UDRS;
```

the following code of the CONTROL UDR command is generated:

```
CONTROL UDR RELATIONSHIPS
            UDR1
            ,UDR2
            ,UDR3
            ,LOGFILE
            ,DATAFILE
            ,UDR6
            ,CLUSTER-FILE
            ;
```

In the user member CONTROLUDR, which contains this generated statement, the entries UDR*n* must be overwritten by the user defined names under which these relationships are already in use. Before changing these name use the SHOW UDR command to clear up the names for the user defined relationships in your dictionary. In the example from above, no subrelationship clause is generated because the fourth parameter of the SETUDR routine has declared that no user defined name shall be used for the subrelationship.

Use the SHOW UDR command to check the correct setting of the UDRs.

After correcting the code of the generated CONTROL UDR command, the user member CONTROLUDR must be executed in the data dictionary which is provided for the ORACLE members.

For MMR-users the UDR clauses must be specified in the SEE clause of the HIERARCHY member.

Export Exit MPDY12PROF

The exit routine MPDY12PROF contains all the settings for the MPR supported relational systems. The options to modify and their allowed values are documented in the respective manuals.

All Exits available are described in the *ASG-Manager Products Relational Technology Support: DB2*:

Exit	Description
Output Exit	Used for manipulating the generated output.
Column Exit	Used when retrieving Repository members via COLUMN clauses.

Import Exits MPDY42DFLT and MPDYWTDFLT

Exit routines MPDY42DFLT and MPDYWTDFLT also control all importing for all supported relational systems.

Installing the Oracle Components

Loading the Datasets in the ORACLE Environment

Transfer all members OR7***** in the MP.SOURCE dataset into the PC-directory serving as search path for the Oracle program SQL*Plus. This table shows the relevant files

PL/SQL scripts of the ORACLE import facility:		
OR7CRVW	->	OR7CRVW.SQL
OR7DRVW	->	OR7DRVW.SQL
OR7START	->	OR7START.SQL
OR7XCLUS	->	CLUSTER.SQL
OR7XDBAS	->	DATABASE.SQL
OR7XDBLK	->	DBLINK.SQL
OR7XFILE	->	FILE.SQL
OR7XINDX	->	INDEX.SQL
OR7XPRIV	->	PRIVILEG.SQL
OR7XPROF	->	PROFILE.SQL
OR7XROLE	->	ROLE.SQL
OR7XSEQU	->	SEQUENCE.SQL
OR7XSLOG	->	SNAPLOG.SQL
OR7XSNAP	->	SNAPSHOT.SQL
OR7XTBSP	->	TBSPACE.SQL
OR7XTABL	->	TABLE.SQL
OR7XTRIG	->	TRIGGER.SQL
OR7XVIEW	->	VIEW.SQL
OR7XUSER	->	USER.SQL

Creation of Oracle System Table Views

The import facility does not exploit the Oracle system tables directly, but uses system table views tailored for this purpose.

As a preparatory step, create these views using the supplied Oracle script OR7CRVW.SQL before using the import facility. The created views are identified by the prefix MP_VIEW_. Creating the views requires privileged access rights of an Oracle system administrator.

To create the system table views

1. Start up the Oracle database system to be documented.
2. Start up Oracle SQL*Plus:

   ```
   SQL> CONNECT SYS/MANAGER
   ```

3. Execute the script OR7CRVW.SQI:

   ```
   SQL> @OR7CRVW
   ```

4. Check the protocol displayed during the creation of the views. Make sure that all views are created without error messages.

After creating the views you can quit SQL*Plus by using the command:

```
SQL> EXIT
```

Error messages may display if views with matching names already exist in the Oracle database. So to remove obsolete views from the database or to clear the database before creating new views use the script OR7DRVW.SQL by using the command:

```
SQL> @OR7DRVW
```

Modifying the Repository Information Model (RIM)

A complete RIM-definition is part of the Oracle interface. These definitions may be modified by the user in the normal way to meet the companies needs. Please note these restrictions:

- If you change the ENCODE-Keywords of the Oracle member types you must adapt the appropriate EXECs. The same action is required when you change the Attribute-Identifier.
- The mapping of used member type encode keywords is controlled in Import Profile MPDYWTDFLT for the importing of Oracle definitions. Changes for the Export use are discouraged.

Miscellaneous

MethodManager Generation

After changes to Oracle member types/attribute types in MMR you have to generate the appropriate panels for these repository objects. To activate the changes you should execute RULE050, Rule070, and RULE080 of the MethodManager Enabling Branch.

Retained Procedures

All frequently used Oracle procedures are coded with the RETAIN option, which keeps the procedure in core. This can speed up the processing time. The RETAIN option is only used when the EXECUTIVE-RETENTION is set to ON.

Modifying Procedures

Some procedures are shared between all supported relational systems such as DB2 or Oracle.

If Profile routines (MPDY12PROF, MPDY42DFLT, MPDYWTDFLT) are changed or central routines (i.e., MPDYMMLOCC) they should be checked that they are consistent for all systems.

Any change in the procedures require an reconstruct of the member, this must be done for EXECUTIVE-ROUTINEs with:

```
CONSTRUCT EXECUTIVE FROM procedure-name;
```

2 Oracle Export

Oracle Tailoring

Oracle Generation Profile

All tailoring is done in the general Export routine MPDY12PROF or in the routine that is pointed to by MPDY_GL_PROFILENAME.

The profile specified in this variable is equivalent to the DB2 profile definition.

Oracle Column Specification

Column Types

The column data types used in Oracle table definitions are very similar to those used in DB2. Therefore in most cases they can easily be transformed. For data types generated for DB2 see *ASG-Manager Products Relational Technology Support: DB2*.

The Oracle data types are transformed in the following manner

DB2-data type	Oracle
VARCHAR and maximum length ≤ 2000	VARCHAR2
VARCHAR and maximum length > 2000	LONG
GRAPHIC and maximum length ≤ 255	RAW
GRAPHIC and maximum length > 255	LONG RAW
DATE, TIME or TIMESTAMP	DATE
FLOAT	NUMBER
INTEGER or SMALLINT	NUMBER(38)

Additional Column Attributes

The full support of generation for Oracle tables requires additional column attributes. These are the DEFAULT values, the CHECK condition, and the exception table. These additional attributes are defined using the DB2 standard attribute COL-ATT1, COL-ATT2 and COL-REL1.

For example:

```
COLUMN CONTAINS
FE-COL1 WITH-DEFAULT COL-ATT1 "100" COL-ATT2 "col1< 999"
COL-REL1 TB-XYZ
```

results in:

```
col1 type DEFAULT 100 CHECK (col1 < 999) EXCEPTIONS INTO xyz.
```

The ORACLE CREATE Command

Use the ORACLE CREATE command to generate an SQL CREATE statement for a Oracle object defined by the relevant member type in your repository, by entering:

`ORACLE CREATE` *`member`* `;`

where *`member`* is the name of a repository member, of one of these types:

- ORACLE-DATABASE
- ORACLE-TABLESPACE
- ORACLE-DBLINK
- ORACLE-SCHEMA
- ORACLE-SEQUENZ
- ORACLE-CLUSTER
- ORACLE-SNAPSHOT
- ORACLE-SNAPLOG
- ORACLE-TRIGGER
- ORACLE-PACKAGE
- ORACLE-USER
- ORACLE-USER
- ORACLE-ROLE
- ORACLE-PROFILE
- ORACLE-INDEX
- ORACLE-TABLE
- ORACLE-VIEW

The Oracle data types of columns in tables or views is given in the ITEM and GROUP members named in the COLUMNS attribute of the relevant ORACLE-TABLE or ORACLE-VIEW member.

If you use distributed databases, you can specify a location as part of an object's name, to uniquely identify it across multiple sites.

Using similar ORACLE CREATE commands, you can easily generate similar SQL CREATE statements for objects with different owners. For example, you might want to create copies of a table, for different owners in a project team.

Generated output displays on the screen. You can tailor this output, file it on the MP-AID, and/or send it to an external dataset, using output generation options. You can also tailor output by calling Executive Routines (user exit routines) at set points (user exits) while generating output.

The systems administrator can also tailor output using the DB2 or ORACLE profile.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for details of using the DB2 (Oracle) profile and user exits

Refer to "Output Generation Options" on page 38 for details of output generation options

Refer to the DB2-TABLE member type in *ASG-Manager Products Relational Technology Support: DB2* for an example of a generated SQL CREATE TABLE statement.

Generating SQL COMMENT ON and LABEL ON Statements

The usage of WITH-COMMENTS or WITH-LABELS keywords to generate SQL COMMENT ON or LABEL ON and CREATE statements with the ORACLE CREATE command is not possible. For compatibility reasons the keywords are permitted, but do not affect the Oracle generation.

Taking User Exits

To take a user exit, enter:

```
ORACLE CREATE member USING exit-routine ;
```

where:

member is the name of a member of the member types listed above.

exit-routine is the name of an Executive Routine.

Note:

The systems administrator can alter the Oracle profile so that a default user exit is always taken when you use the ORACLE CREATE command. The USING keyword overrides any default user exits set this way.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for details of using the DB2 (Oracle) profile and user exits.

Specifying an Owner

To generate an SQL CREATE statement, specifying an owner (overriding any name given in the relevant repository member), enter:

```
ORACLE CREATE member SQLID owner ;
```

where:

member is the name of a member of the member types listed above.

owner is a delimited string of up to 8 characters, giving the ID of a specific user.

Specifying a Location

The usage of the LOCATION keyword, naming a location with the ORACLE CREATE command is not possible. For compatibility reasons the keyword is permitted but does not affect the Oracle generation.

Expanding Nested Data Structures

Use the EXPAND keyword to expand the contents of GROUP members (documenting columns) contained in a repository member (documenting a Oracle table, view or index). These GROUPs can contain other GROUPS, which are expanded in turn.

To expand the contents of a GROUP, enter:

```
ORACLE CREATE member EXPAND ;
```

where *member* is the name of a ORACLE-TABLE, ORACLE-VIEW, or ORACLE-INDEX member.

The NO-EXPAND keyword has the opposite effect, forcing all expansion off, overriding any EXPAND attribute in the member definition.

To force expansion off, enter:

```
ORACLE CREATE member NO-EXPAND ;
```

Note:

If the EXPAND/NO-EXPAND keyword is not used, expansion only occurs when any EXPAND attributes in the member are specified.

Refer to the DB2-TABLE member type in *ASG-Manager Products Relational Technology Support: DB2* for an example of a generated SQL CREATE TABLE statement.

Suppressing Referential Integrity

Use the SUPPRESS-RI keyword to suppress the generation of any referential integrity attributes (constraints, foreign keys) held in a ORACLE-TABLE member. The table, when created in your Oracle environment using the generated SQL CREATE TABLE statement, will have no referential integrity.

For example, you might use this keyword when you first create a table, to test its basic structure. To suppress the generation of referential integrity, enter:

```
ORACLE CREATE table SUPPRESS-RI ;
```

where *table* is the name of a ORACLE-TABLE member.

You can define a table with a self-referencing constraint. The constraint is not contained in the SQL CREATE TABLE statement generated, but is automatically generated in a subsequent (additional) ALTER TABLE statement, as required by Oracle.

Creating Primary Key Indexes

Use the NOINDEX keyword to suppress the creation of a primary key index simultaneously with the create of a Oracle table.

Note:

Any referential integrity attributes (constraints, foreign keys) held in a ORACLE-TABLE member Oracle Version 7 uses another method than Oracle Version 6 (and also another method as DB2) to establish primary key indexes for tables. In Oracle Version 6 a primary key index has to be applied with the command CREATE UNIQUE INDEX.

In Oracle Version 7 the primary key is created immediately with the table as a table constraint. There is a clause in the CREATE TABLE statement which allows the user to define the unique index for the primary key for the table immediately together with the table (USING INDEX ...). The command CREATE INDEX is in Oracle Version 7 provided for creating secondary (non unique) indexes.

The coupling of the creation of primary key indexes to the creation of tables in Oracle Version 7 is such persistently that a primary key index automatically is installed (with a system generated name) if the user who creates a table omits the USING INDEX clause and has no specific options for the primary key index.

This coupling of the creation of primary key indexes to the creation of tables is not as useful if all database objects are generated together out of a dictionary.

Tables, indexes, and so forth are from a dictionary managed as separate objects and their connections are documented via relationship attributes. If database objects shall be generated out of a dictionary it is advisable to analyze the relationships of the objects and to create a list of all objects which belong together (tables, indexes, snaplogs, etc.). Then a batch generation of all objects within this list can be started.

This homogeneous procedure is jeopardized due to the exceptional method of Oracle Version 7 to establish primary key indexes (that differs from the method to create secondary indexes). Because in Oracle Version 7 by default a primary key index immediately is created when a table is created (even if the USING INDEX clause is not specified) it is not possible to create the primary key index by a separate statement after the table is created.

To guarantee a homogeneous procedure to generate the database objects the Oracle interface does not support the specific method of Oracle Version 7 to create primary key indexes. As an alternative course the Oracle interface uses the method which was used in Oracle Version 6 and due to compatibility reasons is still available in Oracle Version 7.

The Oracle interface prefers to generate primary key indexes via a separate CREATE UNIQUE INDEX statement. But this procedure requires the suppression of the automatic generation of primary key indexes while creating Oracle tables. Oracle Version 7 offers a special clause of the CREATE TABLE statement which allows to suppress the automatic generation of primary key indexes ("DISABLE PRIMARY KEY").

This clause can additionally be generated if an ORACLE CREATE table statement is performed out of the dictionary. Use the parameter NOINDEX as table option to generate a CREATE TABLE statement in which the creation of a primary key is disabled. Only if this parameter is set, is it possible to process subsequently a CREATE UNIQUE INDEX for the primary key of that table. If Oracle has already automatically installed a primary key index for a table, a CREATE UNIQUE INDEX for the primary key is rejected.

Use the ORACLE CREATE index statement to generate primary key indexes as well as secondary key indexes from the repository. Primary key indexes must be marked as UNIQUE within the member definition.

Replacing Existing Objects

Use the REPLACE keyword to generate a ORACLE CREATE or REPLACE statement.

```
ORACLE CREATE REPLACE member ;
```

where *member* is the name of a ORACLE-TABLE, ORACLE-VIEW or ORACLE-INDEX member.

Note:

The REPLACE keyword can be entered for any CREATE command but it is only effective for the relevant member types.

Creating Public Objects

Use the PUBLIC keyword to generate a ORACLE CREATE PUBLIC statement.

```
ORACLE CREATE PUBLIC member ;
```

where *member* is the name of a ORACLE-TABLE, ORACLE-VIEW, or ORACLE-INDEX member.

Note:

The PUBLIC keyword can be entered for any CREATE command but it is only effective for the relevant member types.

Creating Package Bodies

To generate a package body enter:

```
ORACLE CREATE BODY member ;
```

where *member* is the name of a ORACLE-PACKAGE member.

Name Editing Options

Use the REPLACE/REPLACING, INSERTING, and DROPPING keywords to edit generated data names before they are output.

Refer to "Name Editing Options" on page 39 for details of name editing options

Output Generation Options

Use the ONTO keyword to direct your generated output to:

- A USER-MEMBER on the MP-AID (public or private)
- A sequential dataset
- A partitioned dataset

Refer to "Output Generation Options" on page 38 for further details of output generation.

Syntax of the ORACLE CREATE Command

```
ORACLE CREATE ──┬─┬─────────┬─┬─────────┬──────────────────────────────────►
                │ │REPLACE  │ │         │
                │ │PUBLIC   │ │         │
                │ │         │ │         │
                │ │         │ │         │
                │ │         
```

cluster-options are:

```
┌──────────────────────────────────────────────────────────────────────────────┐
└─SQLID owner ──────────────────────────────┬──────────────────────────────────┘
                                            ├─EXPAND ─────────────────────────┤
                                            └─NO-EXPAND ──────────────────────┘
```

owner is a delimited string of up to 8 characters, giving the ID of a specific user.

table-options are:

```
┌──────────────────────────────────────────────────────────────────────────────┐
└─SUPPRESS-RI ──┘  ┌─SQLID owner ──┐  ┌─EXPAND ──────┐
                   │               │  │              │
                   │               │  └─NO-EXPAND ───┘
```

```
┌──────────────────────────────────────────────────────────────────────────────┐
└─NO-INDEX ──────┘
```

where:

owner is as defined above.

view-options are:

```
┌──────────────────────────────────────────────────────────────────────────────┐
└─SQLID owner ──┘  ┌─EXPAND ──────┐
                   │              │
                   └─NO-EXPAND ───┘
```

where:

owner is as defined above.

index-options are:

```
┌──────────────────────────────────────────────────────────────────────────────┐
└─SQLID owner ──┘  ┌─EXPAND ──────┐
                   │              │
                   └─NO-EXPAND ───┘
```

where:

owner is as defined above.

trigger-options are:

```
──┬──────────────────────────────────────────────────────────────────►
  │ SQLID owner │
  └─────────────┘
```

For *name-editing-options* see "Name Editing Options" on page 39.

For *destination* see "Output Generation Options" on page 38.

The ORACLE ALTER Command

Use the ORACLE ALTER command to generate one or more SQL ALTER statements from the definition of an ORACLE-TABLE, ORACLE-CLUSTER, ORACLE-PACKAGE, ORACLE-SNAPSHOT, ORACLE-SEQUENCE, ORACLE-VIEW, ORACLE-INDEX, ORACLE-SNAPLOG, ORACLE-TBSPACE, ORACLE-USER, ORACLE-ROLE, ORACLE-TRIGGER, or ORACLE-PROFILE.

You can generate SQL ALTER statements from the repository member types listed above and then apply these SQL statements in your Oracle environment to alter specified Oracle objects. By accurately documenting the SQL objects with repository member definitions, you can maintain these objects with SQL statements generated from the definitions.

You can combine any of the options available for each member type, in any way you wish. This means that you can generate several SQL ALTER statements to make a combination of alterations to an object in your Oracle environment, using one ORACLE ALTER command and one member definition.

The ORACLE ALTER command does not change attributes in the relevant member. You should add or modify these attributes before entering the command. If the attributes are not present in the member definition, no SQL statements are generated. If they are present but have not been modified, the SQL statement(s) generated may be rejected when applied to your Oracle environment.

Note:

You may generate an SQL ALTER TABLE statement to drop a referential constraint, validation routine or primary key. If so, you should drop the attributes from which the SQL statement will be generated, after entering the ORACLE ALTER command. If the attributes have already been removed the SQL statement will not be generated.

Generated statements displays on the screen. You can tailor this output, file it on the MP-AID, or send it to an external dataset, using output generation options. You can also tailor output by calling Executive Routines (user exit routines) at set points (user exits) during output.

The systems administrator can tailor output by altering the DB2 or Oracle Profile.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for details of using the DB2 (Oracle) profile and user exits.

Refer to "Output Generation Options" on page 38 for details of output generation options.

Modifying Columns of a Table

To generate SQL ALTER TABLE statements to modify columns of a table, enter:

```
ORACLE ALTER member MODIFY COLUMNS n ;
```

where:

member is the name of a ORACLE-TABLE repository member.

n is a number from 1 to 299 specifying the columns to be modified. The columns to be modified are the last *n* columns derived from the COLUMNS attribute of the ORACLE-TABLE member from which the SQL ALTER TABLE statement is being generated. *n* must be less than the total number of columns derived from the COLUMNS attribute.

A separate SQL ALTER TABLE statement is generated for each column to be modified.

Adding Columns to a Table

To generate SQL ALTER TABLE statements to add columns to a table, enter:

```
ORACLE ALTER member ADD COLUMNS n ;
```

where:

member is the name of a ORACLE-TABLE repository member.

n is a number from 1 to 299 specifying the columns to be added to the table. The columns to be added to a table are the last *n* columns derived from the COLUMNS attribute of the ORACLE-TABLE member from which the SQL ALTER TABLE statement is being generated. *n* must be less than the total number of columns derived from the COLUMNS attribute.

A separate SQL ALTER TABLE statement is generated for each column to be added to a table.

An SQL statement to add a column to a table will be rejected when applied to your Oracle environment if the column already exists in the table.

Ensure that the ITEMs and GROUPs which define the new columns to be added to a table are specified in the COLUMNS attribute after the members which define the existing columns in a table. Existing columns should not be included in the *n* columns to be added to a table.

The Oracle data type of columns generated in SQL ALTER TABLE statements is derived from the definition of the ITEMs and GROUPs specified in the COLUMNS attribute.

Oracle requires that new columns added to a table must allow a null or default value. The ORACLE ALTER command displays a warning message if any of the columns are defined in the ORACLE-TABLE member definition as being NOT-NULL.

An SQL ALTER TABLE statement to add columns to a table cannot be generated from a ORACLE-TABLE member that contains an EDITPROC clause.

Refer to "The ORACLE CREATE Command" on page 10 for details of generating column data types.

Adding or Dropping Referential Constraints on a Table

To generate an SQL ALTER TABLE statement to add a referential constraint to a table, enter:

```
ORACLE  ALTER member
              ADD CONSTRAINT NAMED constraint-name ;
```

Or

```
ORACLE ALTER member ADD CONSTRAINT NUMBER n ;
```

where:

member is the name of a ORACLE-TABLE repository member.

constraint-name is a name specified in the NAMED attribute of the ORACLE-TABLE member which identifies the referential constraint to be added or dropped.

n is a number identifying which referential constraint is to be added or dropped by its sequence among other referential constraints in the ORACLE-TABLE member definition.

To generate an SQL ALTER TABLE statement to drop a referential constraint from a table, enter:

```
ORACLE  ALTER member
              DROP CONSTRAINT NAMED constraint-name ;
```

Or

```
ORACLE ALTER member DROP CONSTRAINT NUMBER n ;
```

Referential constraints are defined in the CONSTRAINT attributes of ORACLE-TABLE members. The SQL ALTER TABLE statement will be generated from a particular CONSTRAINT attribute. A ORACLE-TABLE member can have any number of CONSTRAINT attributes.

For example, to generate an SQL ALTER TABLE statement to add a referential constraint on a table from the third CONSTRAINT attribute in the member TB-DJB-CUST, enter:

```
ORACLE ALTER TB-DJB-CUST ADD CONSTRAINT NUMBER 3 ;
```

The SQL statement will be rejected when submitted to Oracle if you attempt to add a referential constraint that already exists in the table. The existing referential constraint will not be modified. If you want to replace it you should use the ORACLE ALTER command to first drop the existing referential constraint and then to add the new one.

Adding or Dropping a Primary Key on a Table

To generate an SQL ALTER TABLE statement to add a primary key, enter:

```
ORACLE ALTER member ADD PRIMARY-KEY ;
```

where *member* is the name of a ORACLE-TABLE repository member.

To generate an SQL ALTER TABLE statement to drop a primary key, enter:

```
ORACLE ALTER member DROP PRIMARY-KEY ;
```

Primary key columns are defined with the PRIMARY-KEY attribute in the ORACLE-TABLE member. All columns in the ORACLE-TABLE that have an associated PRIMARY-KEY keyword are generated as part of the primary key.

Columns which allow a null value cannot be part of the primary key. The ORACLE ALTER command therefore displays a warning message if any of the columns defined in the ORACLE-TABLE member as being part of the primary key do not have an associated NOT-NULL or NOT-NULL-WITH-DEFAULT keyword.

The SQL statement will be rejected when submitted to Oracle if you attempt to add a primary key to a table that already has one. The existing primary key will not be modified. If you wish to replace it you should first use the ORACLE ALTER command to drop the existing primary key and then to add the new one.

Adding, Modifying or Dropping a Validation Routine on a Table

The ORACLE ALTER command does not support the generation of validation routines from a ORACLE-TABLE. For compatibility reasons the VALIDPROC keyword is not rejected on an ORACLE ALTER.

Adding or Modifying the Auditing Option on a Table

The ORACLE ALTER command does not support the generation of auditing options from a ORACLE-TABLE. For compatibility reasons the AUDIT keyword is not rejected on an ORACLE ALTER.

Expanding Nested Data Structures

If you have used EXPAND or NO-EXPAND on the corresponding ORACLE CREATE command for this member, you should use the same keywords for this command, by entering:

```
ORACLE ALTER table EXPAND alterations ;
```

Or

```
ORACLE ALTER table NO-EXPAND alterations;
```

where:

table is the name of a ORACLE-TABLE member.

alterations specify the SQL ALTER TABLE statements to be generated.

EXPAND attributes in the ORACLE-TABLE member are used as the default.

Refer to the DB2-TABLE member type in *ASG-Manager Products Relational Technology Support: DB2* for further details about the EXPAND attribute.

Specifying an Owner of a Table

To generate an SQL ALTER TABLE statement for a table, with a specified owner (overriding any owner named in the relevant member), enter:

```
ORACLE ALTER table SQLID owner alterations ;
```

where:

table is the name of a ORACLE-TABLE member.

owner is the authorization ID of a specific user. This must be no more than 8 characters long, and delimited.

alterations specify the alterations to be generated.

Specifying a Location of a Table

The ORACLE ALTER command does not support the generation of a location from a ORACLE-TABLE. For compatibility reasons the LOCATION keyword is not rejected on an ORACLE ALTER.

Taking User Exits

To take a user exit, enter:

```
ORACLE ALTER member alterations USING exit ;
```

where:

member is the name of a ORACLE-TABLE member.

alterations are specifications of the alterations to be generated.

exit is the name of an Executive Routine.

Note:

The systems administrator can alter your Oracle profile so that a default user exit is always taken when you use the ORACLE ALTER command. The USING keyword overrides any default user exits set this way.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for further details of user exits.

Output Generation Options

Use the ONTO keyword to direct your generated output to:

- A USER-MEMBER on the MP-AID (public or private)
- A sequential dataset
- A partitioned dataset

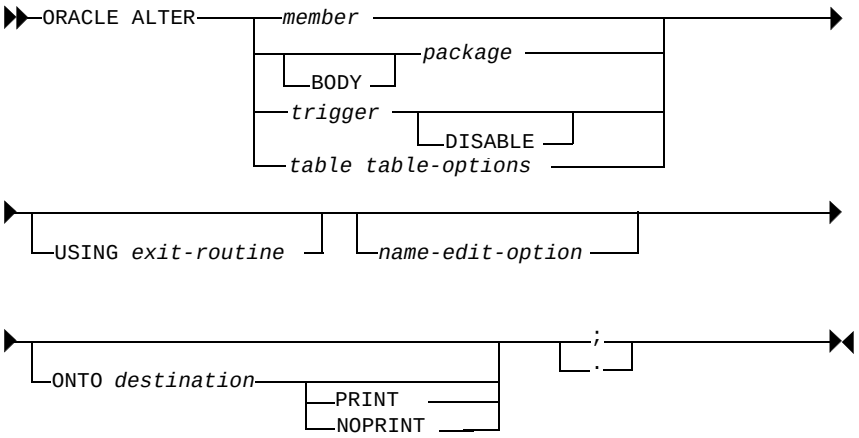
Refer to "Output Generation Options" on page 38 for details of output generation options.

Name Editing Options

Use the REPLACE/REPLACING, INSERTING, and DROPPING keywords to edit generated data names before they are output.

Refer to "Name Editing Options" on page 39 for further details of name editing options.

Syntax of the ORACLE ALTER Command



where:

member is the name of a member of one of the following types: ORACLE-CLUSTER, ORACLE-SNAPSHOT, ORACLE-SEQUENCE, ORACLE-VIEW, ORACLE-INDEX, ORACLE-SNAPLOG, ORACLE-TBSPACE, ORACLE-USER, ORACLE-ROLE, or ORACLE-PROFILE.

table is the name of ORACLE-TABLE members respectively.

trigger is the name of ORACLE-TRIGGER members respectively.

package is the name of ORACLE-PACKAGE members respectively.

user-exit is the name of a user-created executive routine.

table-options are:

```
>──┬──────────────┬──┬──────────────┬──────────────────────────────────────>
   ├──EXPAND──────┤  └──SQLID owner─┘
   └──NO-EXPAND───┘
>──┬──┬──┬──────────────┬──────────────┬──────────────────────────────────>
   │  │  ├──ADD─────────┬──COLUMNS n────┤                                  │
   │  │  │  ├──MODIFY───┘               │                                  │
   │  │  ├──ADD─────────┬──selection────┤                                  │
   │  │  │  └──DROP─────┘               │                                  │
   │  └──STORAGE─────────────────────────┴──────────────────────────────────┘
```

where:

owner is a delimited string of up to 8 characters, giving the authorization ID of a particular user.

selection is:

```
>──┬──PRIMARY-KEY──────────────────────────────────────────────────────────>
   └──CONSTRAINT──┬──NAMED constraint──┬──────────────────────────────────┤
                  └──NUMBER n──────────┘
```

where:

n is an integer.

constraint is a name specified in the NAMED attribute of a ORACLE-TABLE member.

name-editing-options as defined in "Name Editing Options" on page 39.

destination as defined in "Output Generation Options" on page 38.

The ORACLE DROP Command

Use the ORACLE DROP command to generate an SQL DROP statement and an optional impact analysis report.

To generate an SQL DROP statement, and an impact analysis report, for a Oracle object from its repository member definition, enter:

```
ORACLE DROP member ;
```

where *member* is the name of a repository member, of one of these types:

- ORACLE-TABLESPACE
- ORACLE-DBLINK
- ORACLE-SEQUENZ
- ORACLE-CLUSTER
- ORACLE-SNAPSHOT
- ORACLE-SNAPLOG
- ORACLE-TRIGGER
- ORACLE-PACKAGE
- ORACLE-USER
- ORACLE-ROLE
- ORACLE-PROFILE
- ORACLE-INDEX
- ORACLE-TABLE
- ORACLE-VIEW

To generate SQL DROP SYNONYM statements, use the ORACLE SYNONYM command.

If you execute an SQL DROP statement in your Oracle environment, the named Oracle object and all Oracle objects dependent upon it are dropped. For example, if you drop a table, all views and indexes dependent on that table are also dropped. So the ORACLE DROP command also generates an impact analysis report, showing the impact of dropping the specified member in your Oracle environment.

You can generate SQL DROP statements which Oracle may reject.

The generated output displays on the screen. You can tailor the SQL DROP statements, file them on the MP-AID, or send them to an external dataset, using output generation options. You can also tailor the SQL statements by calling Executive Routines (user exit routines) at set points (user exits) during output. This process is known as taking user exits. You cannot alter the impact analysis report using these methods; use the REPORT keyword for this purpose.

The systems administrator can tailor both the SQL statements and the impact analysis report, by altering the Oracle profile.

The Impact Analysis Report

By default, an impact analysis report is generated with the ORACLE DROP command, showing a hierarchy of the impacted repository members (and so the objects in the Oracle environment) that would be dropped or affected if that SQL DROP statement is applied to your Oracle environment. The first line in the report gives the member to be dropped. Each subsequent line shows an impacted member. Each line is numbered, in case of duplication. Entries that appear twice are only given in full once; the next appearances refer to the line number of the first appearance. For each line, the following is also shown, from left to right:

- The relationship between the impacted member and the member one level of indent above it
- The name of the member
- The owner for that member (if an owner exists).
- The type of member

Synonyms of members are also reported.

An impact analysis report is not generated when a ORACLE DROP command is applied to a ORACLE-INDEX member, as only the specified index would be dropped or affected by the SQL DROP INDEX statement.

To deliberately suppress the generation of the impact analysis report, enter:

```
ORACLE DROP member NO-IMPACT-REPORT ;
```

Members impacted by the ORACLE DROP command are reported but are not removed from the repository. If you drop the object from the Oracle environment, you should update the repository to reflect the changes, unless you intend to later recreate the object and those dependent on it. The impact analysis report will help you carry out these updates.

Tailoring the Impact Analysis Report

The impact analysis report is tailorable, with a set of rules defining how the repository is searched. The systems administrator can tailor the table which drives the search algorithm to allow extra search paths to be taken which may be used by the user in addition to the standard relationships.

You can also use specific keywords with the ORACLE DROP command to alter the report width (overriding the default of 80 characters) and the size of each indent in a nested structure (overriding the default of 5 characters). To specify the width of the report, enter:

```
ORACLE DROP member REPORT WIDTH w ;
```

where *w* is the new report width. This must be between 50 and 246 characters.

To specify the size of each indent in a nested structure, enter:

```
ORACLE DROP member REPORT INDENT i ;
```

where *i* is the width of each new indent. This must be between 5 and 20 characters.

You can use KEPT-DATA lists to help manage the impacted members. To store all members displayed in the report in an unnamed KEPT-DATA list, enter:

```
ORACLE DROP member REPORT KEEP ;
```

To store all reported members in a named KEPT-DATA list, enter:

```
ORACLE DROP member REPORT KEEP IN kept-name ;
```

where *kept-name* is the name of a KEPT-DATA list.

To add members to an existing KEPT-DATA list, enter either:

```
ORACLE DROP member REPORT ALSO ;
```

Or

```
ORACLE DROP member REPORT ALSO IN kept-name ;
```

Example of the structure of the Impact Analysis Report

For an example of the impact analysis report refer to *ASG-Manager Products Relational Technology Support: DB2*.

Dropping Associated Objects

It is possible not only to drop a cluster, table, or tablespace but to drop all object depending on the dropped object. To generate a drop-statement that does this enter one of the following commands.

To drop a Profile and un-assign it from any users or to drop a table and all constraints that refer to primary keys in that table enter:

```
ORACLE DROP member CASCADE ;
```

where *member* is a ORACLE-PROFILE or a ORACLE-TABLE.

To drop a tablespace and its contents enter:

```
ORACLE DROP tbspace CONTENTS ;
```

To drop a tablespace, its contents and the constraints form tables outside the tablespace referring to tables in the tablespace enter:

```
ORACLE DROP tbspace CONTENTS CASCADE;
```

To drop a cluster and all tables in that cluster enter:

```
ORACLE DROP cluster TABLES  ;
```

To drop a cluster and the tables that belong to the cluster together with constraints from tables outside the cluster enter:

```
ORACLE DROP cluster TABLES CASCADE ;
```

Taking User Exits

To take a user exit, enter:

```
ORACLE DROP member USING exit-routine ;
```

where:

member is the name of a ORACLE-INDEX, ORACLE-TABLE, ORACLE-VIEW member.

exit-routine is the name of an Executive Routine.

Note:

The systems administrator can alter the Oracle profile so that a default user exit is always taken when you use the ORACLE DROP command. The USING keyword overrides any default user exits set this way.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for details of using the DB2 (Oracle) profile and user exits.

Output Generation Options

Use the ONTO keyword to direct your generated output to:

- A USER-MEMBER on the MP-AID (public or private)
- A sequential dataset
- A partitioned dataset

Refer to "Output Generation Options" on page 38 for details of output generation options.

Syntax of the ORACLE DROP Command

```
ORACLE DROP ──┬──────────┬──────────────────────────────────────────────────────────
              │          │                                                          
              │          │                                                          
              │          │                                                          
```

The ORACLE GRANT/REVOKE Command

Use the ORACLE GRANT and ORACLE REVOKE commands to generate an SQL GRANT or REVOKE statement for a ORACLE object from its definition a ORACLE-PRIVILEGE repository member.

Use the ORACLE GRANT command to generate respectively SQL GRANT or REVOKE statements, by entering either:

```
ORACLE GRANT member ;
```

Or

```
ORACLE REVOKE member ;
```

where *member* is the name of a ORACLE-PRIVILEGE member.

The SQL GRANT statement will include the WITH GRANT OPTION keyword if the WITH-GRANT-OPTION attribute is present in the ORACLE-PRIVILEGE member.

You can generate SQL statements to change privileges for all ORACLE-USER members referred to directly or indirectly by the ORACLE-PRIVILEGE member. You can also generate SQL statements to change privileges for ORACLE-USER members representing single-user IDs only.

Generated output displays on the screen. You can tailor this output, file it on the MP-AID, or send it to an external dataset, using output generation options. You can also tailor output by calling Executive Routines (user exit routines) at set points (user exits) during output. This process is known as taking user exits.

The systems administrator can tailor output by altering the Oracle profile.

Specifying an Owner of an Object

To generate SQL statements to change privileges for a table or view, and override the current owner name, enter either:

```
ORACLE GRANT member SQLID owner ;
```

Or

```
ORACLE REVOKE member SQLID owner ;
```

member is the name of a ORACLE-PRIVILEGE or ORACLE-ROLE repository member.

owner is a delimited string of up to 8 characters, giving the ID of a specific user.

Expanding Nested Data Structures

If you have used EXPAND or NO-EXPAND on the corresponding ORACLE CREATE command for a ORACLE-PRIVILEGE member, you should use the same keywords for this command, by entering:

```
ORACLE GRANT member EXPAND ;
```

Or

```
ORACLE GRANT member NO-EXPAND ;
```

Or

```
ORACLE REVOKE member EXPAND ;
```

Or

```
ORACLE REVOKE member NO-EXPAND ;
```

where *member* is the name of a ORACLE-PRIVILEGE repository member.

EXPAND attributes in the ORACLE-TABLE member are used as the default.

Changing Privileges on an Expanded Range of Users

To generate SQL statements to change privileges for all ORACLE-USER members referred to directly or indirectly by a ORACLE-PRIVILEGE member, enter:

```
ORACLE GRANT member USER-EXPANSION FULL ;
```

Or

```
ORACLE REVOKE member USER-EXPANSION FULL ;
```

where *member* is the name of a ORACLE-PRIVILEGE member.

Alternatively, to generate SQL statements to change privileges for ORACLE-USER members representing single-user IDs only, enter either:

```
ORACLE GRANT member USER-EXPANSION SINGLE-IDS ;
```

Or

```
ORACLE REVOKE member USER-EXPANSION SINGLE-IDS ;
```

By default, only privileges on ORACLE-USER members referred to directly from the ORACLE-PRIVILEGE member are changed.

Specifying the Grantor of a Privilege

Use the BY-GRANTOR keyword to generate a BY keyword in an SQL REVOKE statement, by entering:

```
ORACLE REVOKE member BY-GRANTOR ;
```

where *member* is the name of a ORACLE-PRIVILEGE repository member.

Taking User Exits

To take a user exit, enter either:

```
ORACLE GRANT member USING exit-routine ;
```

Or

```
ORACLE REVOKE member USING exit-routine ;
```

where:

member is the name of a ORACLE-PRIVILEGE repository member.

exit-routine is the name of an Executive Routine.

Note:

The systems administrator can alter your Oracle Profile so that a default user exit is always taken when you use the ORACLE GRANT or ORACLE REVOKE commands. The USING keyword overrides any default user exits set this way.

Name Editing Options

Use the REPLACE/REPLACING, INSERTING, and DROPPING keywords to edit generated data names before they are output.

Output Generation Options

Use the ONTO keyword to direct your generated output to:

- A USER-MEMBER on the MP-AID (public or private)
- A sequential dataset
- A partitioned dataset

Syntax of the ORACLE GRANT/REVOKE Commands

```
>>-+-GRANT options-------------------------------------------------->
   |
   +-REVOKE options-+-----------------+
                    |                 |
                    +-BY-GRANTOR------+

>-+-----------------------------------+-+-----------------------------+-->
  |                                   | |                             |
  +-USING exit-routine----------------+ +-name-edit-options-----------+

>-+-------------------------------------------------------------+-+-;-+-><
  |                                                             | |   |
  +-ONTO destination-+-----------------+-------------------------+ +-.-+
                     |                 |
                     +-PRINT-----------+
                     |                 |
                     +-NOPRINT---------+
```

where:

options are:

```
>-+-privilege-+-----------------------+-------------------------->
  |           |                       |
  |           +-privilege-options-----+
```

where:

privilege-options are:

```
>-+-----------------+-+-EXPAND-----+------------------------------------>
  |                 | |            |
  +-SQLID owner-----+ +-NO-EXPAND---+

>-+-----------------------------------+------------------------------------>
  |                                   |
  +-USER-EXPANSION-+-----------------+
                   |                 |
                   +-FULL------------+
                   |                 |
                   +-SINGLE-IDS------+
```

where:

privilege is a the name of a ORACLE-PRIVILEGE member.

owner is a delimited string of up to 8 characters, giving the authorization ID of a particular user.

exit-routine is the name of an Executive Routine.

The ORACLE SYNONYM Command

Use the ORACLE SYNONYM command to generate SQL CREATE SYNONYM or DROP SYNONYM statements.

To generate SQL CREATE SYNONYM or DROP SYNONYM statements from a ORACLE-USER repository member, enter either:

```
ORACLE SYNONYM CREATE user ;
```

Or

```
ORACLE SYNONYM DROP user ;
```

where `user` is the name of a ORACLE-USER repository member.

To generate other SQL CREATE or DROP statements, use the ORACLE CREATE or ORACLE DROP command.

You can generate SQL statements for a table or a view. A separate SQL statement is generated for each ORACLE-TABLE or ORACLE-VIEW member named in the SYNONYMS attribute of the ORACLE-USER member. You can also generate the last synonyms in a ORACLE-USER member, for example if new synonyms have been added to an existing list, and you only need to generate the additions.

When you have applied an SQL DROP SYNONYM statement to your environment, you should remove or update the relevant ORACLE-USER member to reflect the changes.

Generated output is displayed on the screen. You can tailor this output, file it on the MP-AID, or send it to an external dataset, using output generation options, You can also tailor output by calling Executive Routines (user exit routines) at set points (user exits) during output. This process is known as taking user exits.

The systems administrator can tailor output using the Oracle profile.

Refer to *ASG-Manager Products Relational Technology Support: DB2* for details of the DB2 profile and user exits.

Refer to the DB2-USER member type in *ASG-Manager Products Relational Technology Support: DB2* for an example of an SQL.

CREATE SYNONYM statement generated by the ORACLE SYNONYM command.

Public Synonyms

To generate a public synonym statement from a ORACLE-USER repository member, enter:

```
ORACLE SYNONYM CREATE PUBLIC user ;
```

Or

```
ORACLE SYNONYM DROP PUBLIC user ;
```

where *user* is the name of a ORACLE-USER repository member.

Name Editing Options

Use the REPLACE/REPLACING, INSERTING, and DROPPING keywords to edit generated data names before they are output.

Taking User Exits

To take a user exit, enter either:

```
ORACLE SYNONYM CREATE user USING exit-routine ;
```

Or

```
ORACLE SYNONYM DROP user USING exit-routine ;
```

where *exit-routine* is the name of an Executive Routine.

Note:

The systems administrator can alter your Oracle Profile so that a default user exit is always taken with this command. The USING keyword overrides any default user exits set this way.

Specifying an Owner

To select a ORACLE-USER member with a specified owner, overriding any owner defined for that member, enter:

```
ORACLE SYNONYM CREATE user SQLID owner ;
```

Or

```
ORACLE SYNONYM DROP user SQLID owner ;
```

where *owner* is a delimited string of up to 8 characters, giving the authorization ID of a specific user.

Output Generation Options

Use the ONTO keyword to direct your generated output to:

- A USER-MEMBER on the MP-AID (public or private)
- A sequential dataset
- A partitioned dataset

Generating the Last Synonyms in a ORACLE-USER Member

To generate only the last synonyms listed in a ORACLE-USER member, enter either:

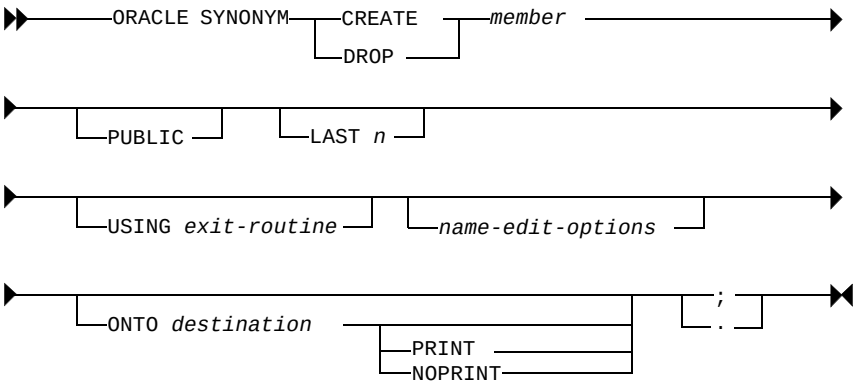
```
ORACLE SYNONYM CREATE user LAST n ;
```

Or

```
ORACLE SYNONYM DROP user LAST n ;
```

where *n* is a number, specifying how many synonyms (taken from the end of the list in the ORACLE-USER member) are to be selected.

Syntax of the ORACLE SYNONYM Command



where:

user is the name of a ORACLE-USER member.

owner is a delimited string of up to 8 characters, giving the authorization ID of a specific user.

n is an integer specifying how many synonyms are to be selected.

exit is the name of an Executive Routine.

General Options

Output Generation Options

Destination is:

```
>---+-PUBLIC-USER-MEMBER ---+---name ---+-------+---+-------+---+-------+---+
    |-PRIVATE-USER-MEMBER --|           |-NEW   |   |       |   |       |   |
    |-USER-MEMBER ----------|           |-APPEND|   |       |   |       |   |
                                        |-REPLACE---+       |   |       |   |
    |-SEQUENTIAL file ------------------------------------------+       |   |
    |                       |-sequential options -------------------------+   |
    |-PARTITIONED file MEMBER name ------------------------------------------+
                                        |-NEW   |
                                        |-REPLACE-+
```

where:

name is the name of a USER-MEMBER or the name of a member in a partitioned dataset.

file is the name of a sequential or partitioned dataset.

sequential-options are:

```
>---+-------+---+-------+---+-------+---+-------+---+
    |-FORMAT ---+-VARIABLE-+   |-RECORD-SIZE ---+-rsize-+   |
              |-FIXED -------+   |-RECORDSIZE ----+       |
>---+-BLOCK-SIZE ---+-blksize ---+
    |-BLOCKSIZE ----+
```

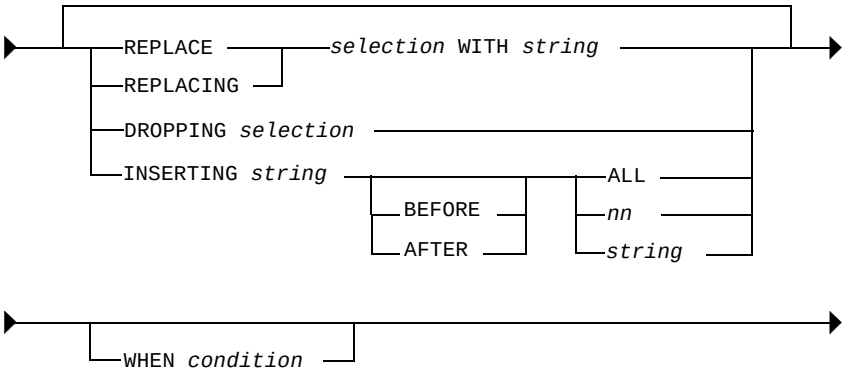
where:

rsize is the record length.

blksize is the block size.

Name Editing Options

Name-editing-options are:



where *selection* is:

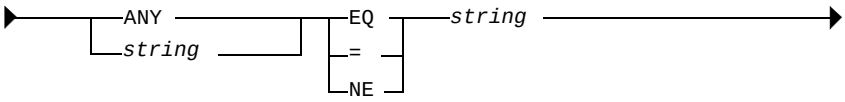
where

m and *p* are integers in the range 1 to 96.

string is a delimited string of not more than 32 printable characters.

nn is an unsigned integer in the range 1 to 96.

condition is:



where *selection* and *string* are as defined above.

3 Oracle Import Interface

The Oracle Import Interface imports information about Oracle objects from a PC-based Oracle database into the Manager Products environment. After documenting your Oracle environment in the repository you can use the functions provided by Manager Products to analyze, enhance, and maintain that environment. Refer to Table 1 for details of the objects you can import and the default member types and proposed names they are given in the repository.

Table 1 • Default repository member types and names of imported objects

ORACLE OBJECT	MEMBER TYPE	MEMBER-NAME
CLUSTER	ORACLE-CLUSTER	ORCL-name
DATABASE	ORACLE-DATABASE	ORDB-name
DATABASE LINK	ORACLE-DBLINK	ORDL-name
FILE	ORACLE-FILE	ORFI-name
INDEX	ORACLE-INDEX	ORIX-name
PRIVILEGE	ORACLE-PRIVILEGE	ORPV-name
PROFILE	ORACLE-PROFILE	ORPR-name
ROLE	ORACLE-ROLE	ORRL-name
SEQUENCE	ORACLE-SEQUENCE	ORSQ-name
SNAPSHOT	ORACLE-SNAPSHOT	ORSN-name
SNAPSHOT LOG	ORACLE-SNAPLOG	ORSL-name
TABLE	ORACLE-TABLE	ORTA-name
TABLESPACE	ORACLE-TABLESPACE	ORTS-name
TRIGGER	ORACLE-TRIGGER	ORTG-name
USER	ORACLE-USER	ORCR-name
VIEW	ORACLE-VIEW	ORVI-name
where *name* corresponds to the name of the object on the ORACLE database		

For further information about the default member types refer to Chapter 4, "Repository Member Types," on page 51.

Importing Information from Oracle in four stages:

1. Creating Oracle system table views in the Oracle database to be documented. Information about Oracle objects is stored in Oracle system tables. You use the views tailored for import purposes when you exploit the system tables.

2. Exploiting Oracle system tables outputs an ASCII report file for every selected Oracle object containing information about the object in the form of a report.

3. Transferring report files to the Host environment where they serve as input to the extract function.

4. Importing into Manager Products environment:

   a. **Extract.** Imports information from the report files into Procedures Language Global Variables in the Workbench Translation Area (WBTA).

   b. **Reconcile.** Generates proposed member names and types for the ORACLE objects, and compares these names and types with any members with matching names and types that exist on the repository.

   c. **Preview.** Generates complete proposed member definitions and allows the user to inspect them before updating the repository.

   d. **Populate.** Updates the repository with the proposed member definitions.

Performing the Import

Extracting Information from the Oracle System Tables

Information about Oracle objects can be extracted for these object types:

- CLUSTER
- DATABASE
- DATABASE-LINK
- FILE
- INDEX
- PRIVILEGE
- PROFILE
- ROLE
- SEQUENCE
- SNAPLOG
- SNAPSHOT
- TABLESPACE
- TABLE
- TRIGGER
- VIEW
- USER

In addition, information about related COLUMNs is extracted for the object types listed below:

- CLUSTER
- INDEX
- SNAPSHOT
- TABLE
- TRIGGER
- VIEW

ITEM repository members document the information about columns.

For the time being the import facility doesn't support the object types PACKAGE and SCHEMA. For detailed information about the supported object types refer to the appendix of this manual.

To extract the information you use a fully automated procedure comprising these steps:

- Starting up of the Oracle environment (SQL*Plus)
- Opening of a spool file
- Putting out retrieval results
- Closing the spool file
- Shutting down of the Oracle environment (SQL*Plus).

Syntax

```
>>─PLUS31[.EXE]  ──S[ILINT]──logon-id/password ──────────────>
>──@OR7START ──object-type  ──user ──object-name ────────────>
```

where:

`PLUS31[.EXE]` is the program name of SQL*Plus.

`-S[ILENT]` is a parameter suppressing SQL*Plus system messages.

`logon-id/password` is the logon ID and password of the privileged user being allowed access to the system table views created by the script CREATEVW.SQL.

object-type is a string of up to 8 characters giving the name of the object type to be retrieved. These character strings are allowed (in *uppercase*):

- CLUSTER
- DATABASE
- DBLINK *(short for Database-Link)*
- FILE
- INDEX
- PRIVILEG *(short for Privilege)*
- PROFILE
- ROLE
- SEQUENCE
- SNAPLOG
- SNAPSHOT
- TBSPACE *(short for Table space)*
- TABLE
- TRIGGER
- VIEW
- USER

user is the name of an Oracle database user.

```
If object-type is
         DATABASE or
         PROFILE or
         ROLE or
         USER
```

then only SYS is allowed as user name.

object-name is the name of an ORACLE object to be extracted. You can specify a generic name using ORACLE specific replacement characters, such as a percent sign (%).

Note:

All parameters are mandatory

For example entering: PLUS31 -S SYS/MANAGER @OR7START TABLE SCOTT% starts up SQL*Plus for the user SYS with password MANAGER. The extraction is performed for all tables belonging to the user SCOTT.

Spool file naming convention

```
<objekttype>.LST or
<objekttype>.LOG (depending on the ORACLE version)
```

Depending on the object type to be extracted one of the following files may be created:

CLUSTER.LST	PROFILE.LST	TABLE.LST
DATABASE.LST	ROLE.LST	TRIGGER.LST
DBLINK.LST	SEQUENCE.LST	VIEW.LST
FILE.LST	SNAPLOG.LST	USER.LST
INDEX.LST	SNAPSHOT.LST	
PRIVILEG.LST	TABLESPACE.LST	

Note:

Existing spool files are overwritten without warning.

The contents and the characteristics of the created spool files should not be altered in any way. Otherwise the correct import of the information into repository members is not ensured.

The Manager Products environment expects the spool files to be in ASCII format. If your Oracle environment produces another file format (e.g. ANSI) a format conversion to ASCII should be done beforehand.

Note:

The current use of scripts which are compatible to Oracle Version 6 causes some limitations for the reports content and for the data contained:

1. The report files are formatted for a maximum of 80 chars per line. No column-value may therefore be larger than 80 chars. This includes text descriptions like COLUMN COMMENTS etc.

2. The VIEW definition includes the describing SELECT statement. The analyzing tool expects valid DB2/ANSI-SQL statements. Oracle-specific enhancements are not supported as are Oracle-specific built in functions. Also the use of SQL-keywords where they are not keywords (by context) is not allowed.

Transferring the Spool Files into the Host environment

Use the file transfer function of your 3270 emulation program for transferring the spool files with name extension .LST (or .LOG respectively) from the Oracle environment into your Host environment. In the Host environment these files serve as input for the Manager Products components of the import facility.

Make sure that these file characteristics are selected for the file transfer:

File Options:	ASCII, CRLF
Record Format:	FIXED (BLOCKED)
Logical Record Format:	80

As to the Host environment data sets to be created observe these naming conventions:

1. The only convention to be observed for the prefix qualifiers is that all names must have the same ones.
2. It's only the suffix qualifier of the data set name that distinguishes the names. The suffix qualifier is identical with the corresponding PC based spool file name.

Example

```
MS.ORACLE.IMPORT.TABLE.LST
MS.ORACLE.IMPORT.TBSPACE.LST
...
```

Importing into the Manager Products Environment

Tailoring Import

In the initialization executive routine MPDY42DFLT one of the variables must contain the prefix qualifiers of the input data set names. The input data sets result from the file transfer.

To change the default prefix qualifiers enter: `UPD MPDY42DFLT` to open the executive routine MPDY42DFLT and select this section:

```
/*-------------------------------------------------------
/* prefix part of all oracle view-datasets for import
/*-------------------------------------------------------
global ts_oracle_prefix
       ts_oracle_prefix = :MS.ORACLE.IMPORT.:
global ts_oracle_suffix
       ts_oracle_suffix = :.LST:
```

Modify the assignments to prefix and/or suffix for your name choice. Make sure that the strings are delimited by literal delimiters (`:`).

Save the modified executive routine with `FILE` and enable by entering:

```
CONSTRUCT EXECUTIVE MPDY42DFLT;
```

Other Tailorable Executive Routines

You can tailor some more executive routines to customize imported information to suit your own purposes and environment, if needed.

MPDYWTDFLT	Sets up the default initialization executive routines and specifies ORACLE member types and member name prefixes.
MPDYWTDFLR	Controls the generation of common clauses among other things.

All executive routines of the form MPDYMM62*xx* control the layout of the generated repository members. *xx* is a string of two characters giving the identifier of the respective member type. These member type identifiers are specified in the executive routine MPDYWTDFLT mentioned above. Modifications of the default repository information model may imply modifications of these executive routines.

The Four steps of the Import

Four commands are provided that must be used in this sequence to support the four steps of the import:

1. EXTRACT ORACLE
2. RECONCILE
3. PREVIEW IMPORT
4. POPULATE

The commands and the procedure of the import correspond to the generic import facility of the Manager Products environment. For more detailed information refer to the generic import documentation.

Step 1—EXTRACT ORACLE

Imports information from the input data sets into Procedures Language Global Variables in the WorkBench Translation Area (WBTA).

Syntax

```
►─EXTRACT ──ORACLE ──objecttype ──object-name ──────────►
►─┬──────────────────┬──┬──;──┬──────────────────────────►◄
  └─CREATOR creator ─┘  └──.──┘
```

where:

object-type is a string of up to 8 characters, giving the name of the object type to be retrieved. The following character strings are allowed:

- CLUSTER
- DATABASE
- DBLINK *(short for Database-Link)*
- FILE
- INDEX
- PRIVILEG *(short for Privilege)*
- PROFILE
- ROLE
- SEQUENCE
- SNAPLOG
- SNAPSHOT
- TBSPACE *(short for Table space)*
- TABLE
- TRIGGER
- VIEW
- USER

object-name is the name of an Oracle object to be extracted. You can specify a generic name using the Oracle specific replacement character (%).

creator is the name of the user being the owner of the object to be imported.

Note:

The parameters *creator* and *object-name* are provided to limit the volume of the import. Version 2.5 of the import facility does not yet support these options. For now the entire file is imported.

Example

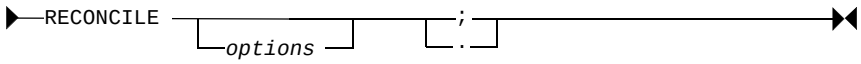
```
EXTRACT ORACLE TABLE % CREATOR SCOTT;
```

All tables belonging to user SCOTT are extracted

Step 2—RECONCILE

Generates proposed member names and types for the ORACLE objects and compares these names and types with any members with matching names and types that exist on the repository. The proposed names include a prefix identifying the type of object they document.

Syntax

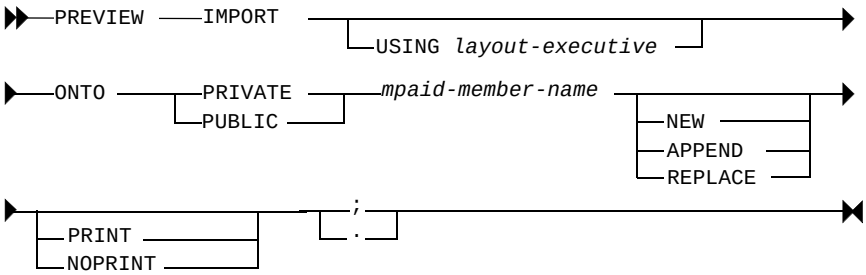


where *options* are additional clauses provided by the generic import facility. For details refer to the generic import documentation.

Step 3—PREVIEW IMPORT

Generates complete proposed member definitions and allows the user to inspect them. These members are of the member types provided by the Oracle Interface.

Syntax



where:

layout-executive is the name of an user specific executive routine used for additional modifications of the member layout.

mpaid-member-name is a string of up to 10 characters giving the valid name of a MPAID user-member containing the generated member definitions.

Note:

For detailed information about the PREVIEW IMPORT command refer to the documentation of the generic import facility.

All members are generated with an ALIAS of type ORACLE. The receiving repository should have this alias type defined to avoid encoding errors.

Step 4—POPULATE

Updates the repository with the proposed member definitions.

Syntax

```
>>─POPULATE ──┬──────────┬─FROM ──┬─BUFFER ──────────────────┬──>
              └─NOPRINT──┘        └─USER user-member-name ───┘

>──┬──────────┬──┬─;─┬──────────────────────────────────────>
  └─ROLLBACK ──┘  └─.─┘
```

where:

BUFFER indicates that the input is taken from the PREVIEW buffer.

user-member-name is the valid name of a MPAID USER-MEMBER containing the input for the repository update.

Note:

For detailed information about the POPULATE command refer to the documentation of the generic import facility.

4 Repository Member Types

Member Type Syntax Descriptions. This chapter describes the syntax of the member types in which you document the objects, locations, and security system of an Oracle database. The member types are documented in alphabetic order of member type name.

When you define a member it is checked to ensure that it follows the syntax.

ORACLE-CLUSTER

```
>>-ORACLE-CLUSTER ---+-------------------+-------------------------------->
                     +-AS cluster--------+

>--+-------------------------+--+-------------------------+--------------->
   +-CREATOR-OWNER user------+  +-IN tablespace-----------+

>--+-------------------+--+-------------------------------+--------------->
   +-column-spec-------+  +-SIZE 'size ---+-------+-'-----+
                                          +-K-----+
                                          +-M-----+

>--+-------------------+--+-------------------+--------------------------->
   +-limit-clause------+  +-storage-clause----+

>--+-------------------------------------------------------------------+-->
   +-INDEXED-----------------------------------------------------------+
   +-----------------------------------------+-HASHKEYS key------------+
   +-HASHCOL 'column-name'-------------------+

>--+-------------------------------+--+-------------------------+------->
   +-EXIT-SIZE 'size ---+-------+-'-+  +-INSTANCE integer--------+
                        +-K-----+
                        +-M-----+

>--+-------------------------------------------------------------------+-->
   +-DATAFILE ---oracle-file-name ---+-----------------------+-----------+
                                     +-INFO 'REUSE'----------+

>--+-------------------+--+-----+--><
   +-common clauses----+  +-;---+
                          +-.---+
```

where:

cluster is the name of an ORACLE-CLUSTER member.

user is the name of a ORACLE-USER member.

tablespace is the name of an ORACLE-TABLESPACE member.

column-spec is:

```
┌──────────────────────┬─CONTAINS contains-clause──────────────────────►
├─ENTERED-AS───────────┤
├─HELD-AS──────────────┤
├─REPORTED-AS──────────┤
└─DEFAULTED-AS─────────┘
```

where:

```
┌──────────────────────┬─single-column-clause──────────────────────────►
├─( number )───────────┤                                               │
└─HELD-AS──────────────┴─group EXPAND──────────────────────────────────┘
```

where:

number is the number of columns in a column set.

single-column-clause is:

```
┌─item──┬──────────┬──────────────────────────────────────────────────►
│       └─version──┘  └─KNOWN-AS local-name──┘
└─group─────────────┘
```

where:

item is the name of an ITEM member.

version is an integer in the range 1 to 15.

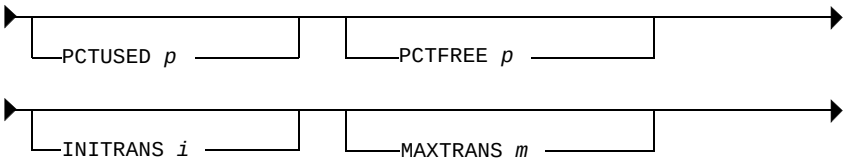
group is the name of a GROUP member.

local-name is the name of the column, consisting of no more than 18 characters.

group is as defined above.

size is an integer.

limit-clause is:



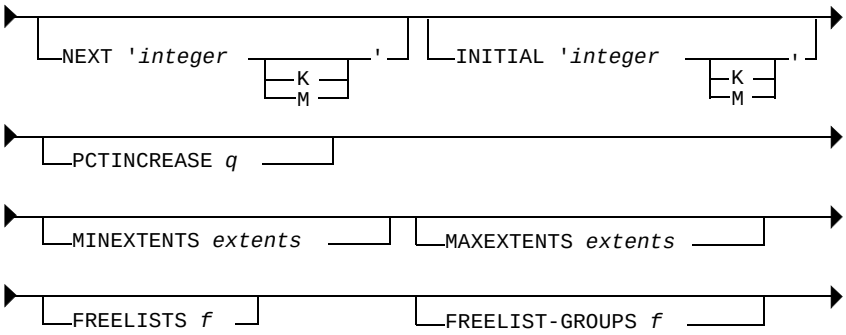
where:

p is an integer in the range of 1 to 99 (percentage).

i is an integer in the range of 2 to the value of the MAXTRANS parameter.

m is an integer in the range of the value of the INITRANS parameter to 255.

storage-clause is:



where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

column-name is the name of an ITEM member.

key is an integer with a minimum value of 2.

integer is as defined above.

size is as defined above.

oracle-file-name is the name of an ORACLE-FILE member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-DATABASE

```
>>-ORACLE-DATABASE -+-------------------+------------------------------------>
                    +-AS database       -+
>--+-------------------+--+-------------------+-------------------------------->
   +-CONTROLFILE       -+  +-EXCLUSIVE         -+
>--+-------------------+--+-------------------+-------------------------------->
   +-ARCHIVELOG        -+  +-MAXLOGFILES f     -+
>--+-------------------+--+-------------------+-------------------------------->
   +-MAXLOGMEMBERS m   -+  +-MAXLOGHISTORY h   -+
>--+-------------------+--+-------------------+-------------------------------->
   +-MAXLOGDATAFILES d -+  +-MAXINSTANCES i    -+
>--+-------------------------------------------------------------------------+
   |                <<<<<<<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<                |
   +-LOGFILE  -----oracle-file-name-----------------------------------------+
                                   +-group-block-+
>--+-------------------------------------------------------------------------+
   |                <<<<<<<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<                |
   +-DATAFILE -----oracle-file-name-----------------------------------------+
                                   +-INFO 'REUSE'-+
>--+-------------------+--+-----+-------------------------------------------><
   +-common clauses    -+  +-;-+
                           +-.-+
```

where:

database is the name of an ORACLE-DATABASE member.

f is an integer.

m is an integer with a minimum value of 1.

h is an integer with a minimum value of 0.

d is an integer with a minimum value of 1.

i is an integer with a minimum value of 1.

oracle-file-name is the name of an ORACLE-FILE member.

group-block is:

```
INFO ' ──┬─GROUP g ──────────┬─ ' ──
         ├─REUSE ────────────┤
         └─GROUP g, REUSE ───┘
```

where:

g is an integer in the range 1 to the value of the MAXLOGFILES parameter.

oracle-file-name is as defined above.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-DBLINK

```
ORACLE-DBLINK ──────AS dblink ──────────────────────────────
──┬──────────────────────────────────────────────────────┬──
  └─DOMAIN 'domain' ──┬──────────────────────────────┬──┘
                      │ <<<<<<<<<<<                  │
                      └─'domain' ────────────────────┘
──┬──────────────────────────────────────────────────────┬──
  └─CONNECT 'qualifier' ─┘
──┬──────────────────────────────────────────────────────┬──
  └─SEE user ─┘   └─USING 'connect-string' ─┘
──┬──────────────────────────────┬──┬──────────────────┬──┬──
  └─common clauses ─┘            │  │ ; ───────────────┤  │
                                 │  └─. ───────────────┘  │
                                 └────────────────────────┘
```

where:

dblink is the name of an ORACLE-DBLINK member.

domain is a remote database domain name.

qualifier is a connection qualifier.

connect-string is the specification of a remote database.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-FILE

```
>>-ORACLE-FILE ---------------------------------------------------------->
                 |                                                      |
                 |--AS oracle-file-name-------------------------------|

>--PATH 'path' ---------------------------------------------------------->
                 |                                                      |
                 |--FILESIZE 'size ------------------------------'-----|
                                     |-K-|
                                     |-M-|

>--------------------------------;-------------------------------------><
  |                              |
  |--common clauses--------------|-.-----------------------------------|
```

where:

oracle-file-name is the name of an ORACLE-FILE member.

path is the path specification of the ORACLE-FILE.

size is an integer.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-INDEX

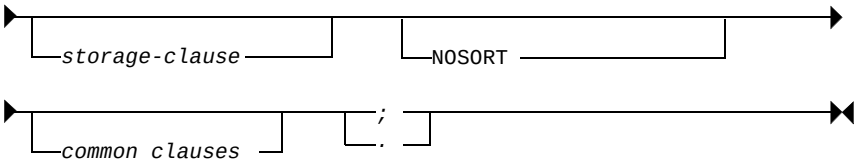
```
>>-ORACLE-INDEX --------------------------------------------------------->
                 |                                                      |
                 |--AS index------------------------------------------|

>-------------------------------------------------------------------------->
  |                                                                      |
  |--CREATOR-OWNER user----------------------------------SEE tablespace-|

>-------------------------------------------------------------------------->
  |                                                                      |
  |--UNIQUE--------------ON--cluster-------------------------------------|
                           |                  <<<<<< , <<<<<<
                           |--table CONTAINS contains-clause------------|

>-------------------------------------------------------------------------->
  |                                                                      |
  |--PCTFREE p-------------------------------------------------------|

>-------------------------------------------------------------------------->
  |                                                                      |
  |--INITRANS i------------------------------MAXTRANS m-------------|
```



where:

index is the name of an ORACLE-INDEX member.

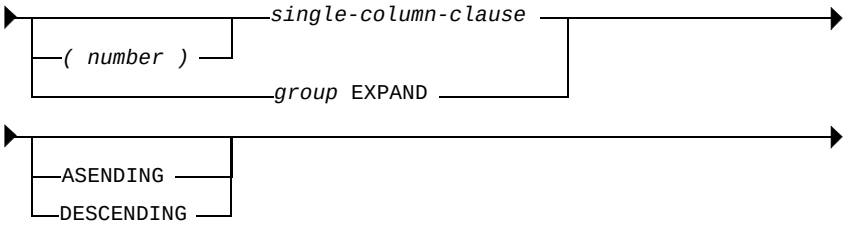
user is the name of a ORACLE-USER member.

tablespace is the name of an ORACLE-TABLESPACE member.

cluster is the name of an ORACLE-CLUSTER member

table is the name of an ORACLE-TABLE member

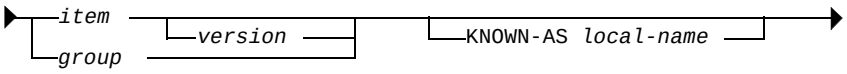
contains-clause is:



where:

number is the number of columns in a column set.

single-column-clause is:



where:

item is the name of an ITEM member.

version is an integer in the range 1 to 15.

group is the name of a GROUP member.

local-name is the name of the column, consisting of no more than 18 characters.

group is as defined above.

p is an integer in the range of 1 to 99 (percentage).

i is an integer in the range of 2 to 255.

m is an integer in the range of 1 to 255.

storage-clause is:

```
NEXT 'integer [K|M]'
INITIAL 'integer [K|M]'
PCTINCREASE q
MINEXTENTS extents
MAXEXTENTS extents
FREELISTS f
FREELIST-GROUPS f
```

where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-PACKAGE

```
ORACLE-PACKAGE [AS package]
[CREATOR-OWNER user] [SQL 'sql-spec']
[common clauses] [; | .]
```

where:

package is the name of an ORACLE-PACKAGE member.

user is the name of a ORACLE-USER member.

sql-spec is the package specification.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-PRIVILEGE

```
>>-ORACLE-PRIVILEGE ---------------------------------------------------->
                     |                                |
                     |-AS PRIVILEGE-------------------|
>------------------------------------------------------------------------>
  |-ALTER-PRIVILEGE   alter-privileges-------------------------------|
  |-CREATE-PRIVILEGE  create-privileges------------------------------|
  |-DROP-PRIVILEGE    drop-privileges--------------------------------|
  |-GENERAL-PRIVILEGE general-privileges-------------------------------|
                                                  <<,<<
  |-TABLE-PRIVILEGE   table-privileges  -ON-----table----------------|
  |-ALL ------------------------------------ON-----table----------------|
                                                  |-view-------------|
  |-SEE role-----------------------------------------------------------|
>------------------------------------------------------------------------>
  |-------------------------------------------------------------------|
  |-TO-----<<,<<-----------------------------------------------------|
         |-user-|                                                   |
         |-role-|   |-WITH-GRANT-OPTION-------------------------|
>-----------------------------------------------------------------------><
  |-------------------------------------------------------------------|
  |-common clauses-----|-------------;-----|
                       |-------------.-----|
```

where:

privilege is the name of an ORACLE-PRIVILEGE member.

alter privileges are:

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
|
|---ANY-CLUSTER ---------------|
|---ANY-INDEX -----------------|
|---ANY-PROCEDURE -------------|
|---ANY-ROLE ------------------|
|---ANY-SEQUENCE --------------|
|---ANY-SNAPSHOT --------------|
|---ANY-TABLE -----------------|
|---ANY-TRIGGER ---------------|
|---DATABASE ------------------|
|---PROFILE -------------------|
|---RESOURCE-COST -------------|
|---ROLLBACK-SEGMENT ----------|
|---SESSION -------------------|
|---SYSTEM --------------------|
|---TABLESPACE ----------------|
|---USER ----------------------|
```

create-privileges are:

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
|
|---ANY-CLUSTER ---------------|
|---ANY-INDEX -----------------|
|---ANY-PROCEDURE -------------|
|---ANY-ROLE ------------------|
|---ANY-SEQUENCE --------------|
|---ANY-SNAPSHOT --------------|
|---ANY-SYNONYM ---------------|
|---ANY-TABLE -----------------|
|---ANY-TRIGGER ---------------|
|---ANY-VIEW ------------------|
|---CLUSTER -------------------|
|---DATABASE-LINK -------------|
|---PROCEDURE -----------------|
|---PROFILE -------------------|
|---PUBLIC-DBLINK -------------|
|---PUBLIC-SYNONYM ------------|
|---ROLE ----------------------|
|---ROLLBACK-SEGMENT ----------|
|---SESSION -------------------|
|---SEQUENCE ------------------|
|---SNAPSHOT ------------------|
|---SYNONYM -------------------|
|---TABLE ---------------------|
|---TABLESPACE ----------------|
|---TRIGGER -------------------|
|---USER ----------------------|
|---VIEW ----------------------|
```

drop-privileges are:

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
|
|--ANY-CLUSTER ----------------|
|--ANY-INDEX ------------------|
|--ANY-PROCEDURE --------------|
|--ANY-ROLE -------------------|
|--ANY-SEQUENCE ---------------|
|--ANY-SNAPSHOT ---------------|
|--ANY-SYNONM -----------------|
|--ANY-TABLE ------------------|
|--ANY-TRIGGER ----------------|
|--ANY-VIEW -------------------|
|--PROFILE --------------------|
|--PUBLIC-DBLINK --------------|
|--PUBLIC-SYNONM --------------|
|--ROLLBACK-SEGMENT -----------|
|--TABLESPACE -----------------|
|--USER -----------------------|
```

general-privileges are:

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
|
|--ANALYZE-ANY ----------------|
|--AUDIT-ANY ------------------|
|--BECOME-USER ----------------|
|--BACKUP-ANY-TABLE -----------|
|--COMMENT-ANY-TABLE ----------|
|--DELETE-ANY-TABLE -----------|
|--EXECUTE-ANY-PROC -----------|
|--FORCE-ANY-TRANSACTION ------|
|--GRANT-ANY-PRIV -------------|
|--GRANT-ANY-ROLE -------------|
|--INSERT-ANY-TABLE -----------|
|--LOCK-ANY-TABLE -------------|
|--MANAGE-TABLESPACE ----------|
|--READUP ---------------------|
|--ROLE -----------------------|
|--RESTRICTED-SESSION ---------|
|--SELECT-ANY-SEQUENCE --------|
|--SELECT-ANY-TABLE -----------|
|--UNLIMITED-TABLESPACE -------|
|--UPDATE-ANY-TABLE -----------|
|--WRITEDOWN ------------------|
|--WRITEUP --------------------|
```

table-privileges are:

```
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<< ,<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
|--ALLTER ------------------------------------------------------------|
|--DELETE ------------------------------------------------------------|
|--INDEX -------------------------------------------------------------|
|--INSERT ------------------------------------------------------------|
|--SELECT ------------------------------------------------------------|
|--UPDATE ------------------------------------------------------------|
|--REFERENCES ---------------------------------------------------------|
|                 |                                                  |
|                 |--CONTAINS <<< columns <<< , ---------------------|
```

where *columns* are:

```
|--------------------------------------------------------------------|
|  |--(n)--|   |--item  --|   |--KNOWN-AS local-name --|
|  |       |   |--group --|   |                        |
|  |--group EXPAND ----------------------------------------|
```

where:

n is the number of columns in a column set.

item is the name of an ITEM member.

group is the name of a GROUP member.

local-name is a name of not more than 18 characters.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member

role is the name of an ORACLE-ROLE member.

Via SEE one single role can be referenced which consists of a set of privileges to be granted to users or other roles.

Via TO one or more roles (as well as users) can be referenced as grantees of the privilege.

user is the name of an ORACLE-USER member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-PROFILE

```
>>-ORACLE-PROFILE ---------------------------------------------------------->
                  |                                                        |
                  |-AS profile -----------------------------------------|
>------------------------------------------------------------------------------>
|-SESSIONS-P-U value -|  |-CPU-P-S cpu -|
>------------------------------------------------------------------------------>
|-CPU-P-U cpu ---------|  |-CONNECT-TIME cpu -|
>------------------------------------------------------------------------------>
|-IDLE-TIME time -------|  |-LOGICAL-READS-P-S value -|
>------------------------------------------------------------------------------>
|-LOGICAL-READS-P-C value -|  |-COMPOSITE-LIMIT value -|
>------------------------------------------------------------------------------>
|-PRIVATE-SGA space -|
>------------------------------------------------------------------------------>
|          <<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<|
|-SYNONYM synonym-name FOR -------|-table -|-|
                                  |-view -|
>-----------------------------------------------------------------------------><
|-------------------------|   |-; -|
|-common clauses -|   |-. -|
```

where:

profile is the name of an ORACLE-PROFILE member.

value is:

```
>-'-----|-integer -----|-'-------------------------------------------------->
        |-UNLIMITED -|
        |-DEFAULT -----|
```

where:

integer is an integer.

cpu is:

```
>-'-----|-time1 -------|-'-------------------------------------------------->
        |-UNLIMITED -|
        |-DEFAULT -----|
```

where:

time1 is an integer (time in 1/100 seconds).

time is:

```
>──'──┬─time2──────┬──'──────────────────────────────────>
      ├─UNLIMITED──┤
      └─DEFAULT────┘
```

where:

time2 is an integer (time in minutes).

value is as defined above.

space is:

```
>──'──┬─integer──┬─┬─K─┬─┬──'──────────────────────────>
      │          │ └─M─┘ │
      ├─UNLIMITED──────┤
      └─DEFAULT────────┘
```

where:

integer is as defined above.

synonym-name is a synonym, of no more than 18 characters.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-ROLE

```
>>─ORACLE-ROLE──┬──────────┬──────────────────────────>
                └─AS role──┘

>──┬──────────────────────────┬──────────────────────────>
  ├─PASSWORD 'password'──────┤
  └─EXTERN───────────────────┘

>──┬──────────────────────────────────────────────────┬──>
  └─SYNONYM synonym-name FOR──┬─table─┬──────────────┘
                              └─view──┘
```

```
►──┬──────────────────┬──┬──;──┬──────────────────────────────►◄
   └─common clauses───┘  └──.──┘
```

where:

role is the name of an ORACLE-ROLE member.

password is a password.

synonym-name is a synonym, of no more than 18 characters.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-SCHEMA

```
►►─ORACLE-SCHEMA ──┬──────────────────┬──────────────────────────►
                   └─AS schema────────┘

►─SEE ──┬─table──┬──────────────────────────────────────────────►
        ├─view───┤
        └─privilege─┘

►──┬──────────────────┬──┬──;──┬──────────────────────────────►◄
   └─common clauses───┘  └──.──┘
```

where:

schema is the name of an ORACLE-SCHEMA member.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member.

privilege is the name of an ORACLE-PRIVILEGE member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-SEQUENCE

```
>>-ORACLE-SEQUENCE ---------------------------------------------------------->
                   |  AS sequence  |
>--|  CREATOR-OWNER user  |--|  INCREMENT interval  |-------------------------->
>--|  START 'integer'  |--|  MAXVALUE 'integer'  |---------------------------->
>--|  MINVALUE 'integer'  |--|  CACHE cache  |------------------------------->
>--|  CYCLE  |--|  ORDER  |------------------------------------------------->
>--|  common clauses  |--|  ;  |--|  .  |--><
```

where:

sequence is the name of an ORACLE-SEQUENCE member.

user is the name of a ORACLE-USER member.

interval is an integer, but it cannot be 0.

integer is an integer.

cache is an integer with a minimum value of 2.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-SNAPLOG

```
>>-ORACLE-SNAPLOG ----------------------------------------------------------->
                  |  AS snaplog  |
>--ON table ---------------------------------------------------------------->
>--|  CREATOR-OWNER user  |--|  SEE tablespace  |------------------------------>
>--|  limit-clause  |--|  storage-clause  |------------------------------>
>--|  common clauses  |--|  ;  |--|  .  |--><
```

where:

snaplog is the name of an ORACLE-SNAPLOG member.

table is the name of an ORACLE-TABLE member.

user is the name of a ORACLE-USER member.

tablespace is the name of an ORACLE-TABLESPACE member.

limit-clause is:

```
──┬─PCTUSED p─┬──┬─PCTFREE p─┬──────────────────────────────────►
  │           │  │           │
  ├─INITRANS i─┤  ├─MAXTRANS m─┤
```

where:

p is an integer in the range of 1 to 100 (percentage).

i is an integer in the range of 2 to the value of the MAXTRANS parameter.

m is an integer in the range of the value of the INITRANS parameter to 255.

storage-clause is:

```
──┬─NEXT 'integer ──┬─K─┬─'─┬─INITIAL'integer ──┬─K─┬─'─┬──────►
  │                 │ M │   │                   │ M │   │
  │                 └───┘   │                   └───┘   │
  ├─PCTINCREASE q──────────┤
  ├─MINEXTENTS extents──────┤─MAXEXTENTS extents────────┤
  ├─FREELISTS f─────────────┤─FREELIST-GROUPS f─────────┤
```

where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-SNAPSHOT

```
>>-ORACLE-SNAPSHOT ---------------------------------------------------->
                    |  AS snapshot  |

>-|  CREATOR-OWNER user  |  |  column-spec  |  |-FROM table  -----|
                                               |      view       |
                                               |     snapshot    |

>-|-CLUSTER cluster CLUSTER-COLS item ----------------------------|
  |-limit-clause  |  |  storage-clause  |  |  SEE tablespace  |

>-|-REFRESH refresh-clause ---------------------------------------|

>-|-------------------|  |-;-|  |-----------------------------><<
  |  common clauses   |  |-.-|
```

where:

snapshot is the name of an ORACLE-SNAPSHOT member.

user is the name of a ORACLE-USER member.

column-spec is:

```
>-|-------------------|-CONTAINS contains-clause ---------->
  |-ENTERED AS -------|
  |-HELD-AS ----------|
  |-REPORTED-AS ------|
  |-DEFAULTED-AS -----|
```

where:

contains-clause is:

```
>-|-------------------|-single-column-clause ---------------|-->
  |-(number) ---------|                                     |
  |-----------------------group EXPAND ---------------------|

>-|-COLUMN-NAME name  |  |-EXPRESSION 'expression'-|-------->
```

where:

number is the number of columns in a column set.

single-column-clause is:

```
item  ──┬──────────┬──┬──────────────────────────────┬──
        │ version  │  │ KNOWN-AS local-name          │
group ──┴──────────┴──┴──────────────────────────────┴──
```

where:

item is the name of an ITEM member.

version is an integer in the range 1 to 15.

group is the name of a GROUP member.

local-name is the name of the column, consisting of no more than 18 characters.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member.

snapshot is the name of an ORACLE-SNAPSHOT member.

cluster is the name of an ORACLE-CLUSTER member.

limit-clause is:

```
──┬─PCTUSED p──┬──┬─PCTFREE p──┬──
  │            │  │            │
──┴─INITRANS i─┴──┴─MAXTRANS m─┴──
```

where:

p is an integer in the range of 1 to 100 (percentage).

i is an integer in the range of 2 to the value of the MAXTRANS parameter.

m is an integer in the range of the value of the INITRANS parameter to 255.

storage-clause is:

```
──┬─NEXT 'integer ──┬─K─┬──'──┬──┬─INITIAL'integer ──┬─K─┬──'──┬──
  │                 │ M │     │  │                   │ M │     │
  └─────────────────┴───┴─────┴──┴───────────────────┴───┴─────┴──
```

```
PCTINCREASE q

MINEXTENTS extents        MAXEXTENTS extents

FREELISTS f               FREELIST-GROUPS f
```

where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

tablespace is the name of an ORACLE-TABLESPACE member.

refresh-clause is:

```
FAST        START-DATA 'date'    NEXT-DATA 'date'
COMPLETE
FORCE
```

where *date* is a date expression

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-TABLE

```
>>-ORACLE-TABLE ------+------------------------------------------------>
                     |  AS table  |
>--+-------------------------------+--COLUMNS column-spec ------------->
   |  CREATOR-OWNER user           |
>--+-------------------------------------------------------------------->
   |  CLUSTER-OWNER user CLUSTER-COLS column-name ---------------------|
   |-------------------------------------------------------------------|
   |  limit-clause  |  storage-clause  |  IN tablespace  |
>--+-------------------------------------------------------------------->
   |  EXT-SIZE 'size ---------+-------'  |  INSTANCE integer  |
                             |  K  |
                             |  M  |
>--+-------------------------------------------------------------------->
   |  DATAFILE ----oracle-file-name ---------------------------------|
                                    |  INFO 'REUSE'  |
>--+-------------------------------+--;  |-------------------------------><
   |  common clauses  |            |  .  |
```

where:

`table` is the name of an ORACLE-TABLE member.

`user` is the name of a ORACLE-USER member.

`column-spec` is:

```
>--+-------------------------------+--CONTAINS contains-clause ---------->
   |  ENTERED AS  |
   |  HELD-AS     |
   |  REPORTED-AS |
   |  DEFAULTED-AS |
   <<<<<<<<<<<<<<<<<<<<<<<<
>--+--referential-constriant ------------------------------------------->
```

`contains-clause` is:

```
>--+-------------------------------+--single-column-clause ------------>
   |  (number)  |                  |
   |-------------------------------+--group EXPAND ------------------|
>--+-------------------------------------------------------------------->
   |  column-attributes  |
```

where:

`number` is the number of columns in a column set.

single-column-clause is:

```
item  ──┬──────────┬──┬──────────────────────────┬──
        │          │  │                          │
        └─version──┘  └─KNOWN-AS local-name──────┘
group ──┘
```

where:

item is the name of an ITEM member.

version is an integer in the range 1 to 15.

group is the name of a GROUP member.

local-name is the name of the column, consisting of no more than 18 characters.

group is as defined above.

column-attributes is:

```
──┬──────────┬──┬──────────────┬──
  │          │  │              │
  ├─NOT-NULL─┤  └─PRIMARY-KEY──┘
  └─WITH-DEFAULT─┘

──┬──────────────────┬──┬──────────────────┬──
  └─COL-REL1 member──┘  └─COL-ATT1 'string'─┘

──┬──────────────────┬──┬──────────────────┬──
  └─COL-REL2 member──┘  └─COL-ATT2 'string'─┘

──┬──────────────────┬──┬──────────────────┬──
  └─COL-REL3 member──┘  └─COL-ATT3 'string'─┘
```

where:

member is the name of a repository member.

string is a string of up to 254 characters.

referential-constraint is:

```
──CONSTRAINT──┬──────────────────────────┬──
              │                          │
              └─NAMED constraint-name────┘

──┬──────────────────────────────────┬──┬──────────────────┬──
  │                                  │  │                  │
  │                <<<<<   <<<<<     │  └─REFERENCE table──┘
  └─FOREIGN-KEY specification────────┘

──┬──────────────────────────────────
```

where:

constraint-name is the name of the constraint, consisting of no more than 8 characters.

specification is:

```
┌─single-column-clause──────────────────────────────────────►
└─group EXPAND──────────┘
```

where:

single-column-clause is as defined above.

group is as defined above.

table is as defined above.

cluster is the name of an ORACLE-CLUSTER member.

column-name is the name of an ITEM member.

limit-clause is:

```
┌──────────────────────────────────────────────────────────►
└─PCTUSED p──────────┘ └─PCTFREE p──────────┘

┌──────────────────────────────────────────────────────────►
└─INITRANS i─────────┘ └─MAXTRANS m─────────┘
```

where:

p is an integer in the range of 1 to 100 (percentage).

i is an integer in the range of 2 to the value of the MAXTRANS parameter.

m is an integer in the range of the value of the INITRANS parameter to 255.

storage-clause is:

```
┌──────────────────────────────────────────────────────────►
└─NEXT 'integer──┬──┬─K─┬──'─┘ └─INITIAL'integer──┬──┬─K─┬──'─┘
                 └──└─M─┘                         └──└─M─┘

┌──────────────────────────────────────────────────────────►
└─PCTINCREASE q──────┘

┌──────────────────────────────────────────────────────────►
└─MINEXTENTS extents─┘ └─MAXEXTENTS extents─┘

┌──────────────────────────────────────────────────────────►
└─FREELISTS f─┘        └─FREELIST-GROUPS f─┘
```

where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

tablespace is the name of an ORACLE-TABLESPACE member.

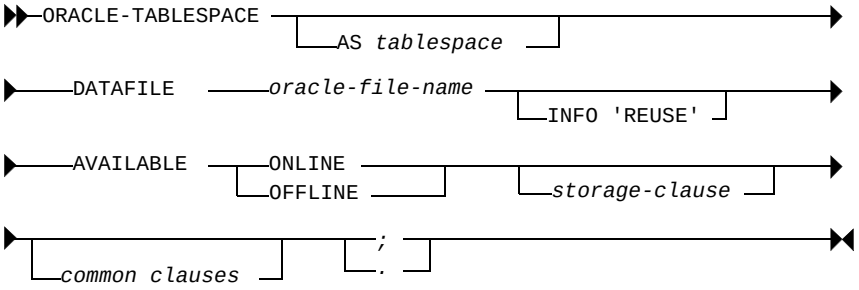
size is an integer.

integer is an integer.

oracle-file-name is the name of an ORACLE-FILE member.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-TABLESPACE



where:

tablespace is the name of an ORACLE-TABLESPACE member.

oracle-file-name is the name of an ORACLE-FILE member.

storage-clause is:

```
NEXT 'integer [K|M]'  |  INITIAL'integer [K|M]'
PCTINCREASE q
MINEXTENTS extents  |  MAXEXTENTS extents
FREELISTS f  |  FREELIST-GROUPS f
```

where:

integer is an integer.

q is an integer in the range of 0 to 100 (percentage).

extents is an integer with a minimum value of 1.

f is an integer with a minimum value of 1.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-TRIGGER

```
ORACLE-TRIGGER [AS trigger]
[CREATOR-OWNER user] ON table
OPTION [BEFORE|AFTER] ACTION action-clause SQL sql-block
[ROW-TRIGGER [NO|YES yes-clause]]
[common clauses] [;|.]
```

where:

trigger is the name of an ORACLE-TRIGGER member.

user is the name of a ORACLE-USER member.

table is the name of an ORACLE-TABLE member.

action-clause is:

```
▶─┬─DELETE ──┬──┬─,INSERT ──┬──┬─,update-clause ──┬──┬──────────────────────────────────────────▶
  │          │  │           │  │                  │  │
  │          │  │           │  └─,update-clause ──┘  │
  │          │  └─,update-clause ────────────────────┘  │
  │          │                                          │
  ├─INSERT ──┼──┬─,update-clause ──────────────────────┤
  │          │  │                                      │
  │          │  └──────────────────────────────────────┘
  │          │                                          │
  └─update-clause ──────────────────────────────────────┘
```

where:

update-clause is:

```
▶──┬─UPDATE ──┬──┬─CONTAINS contains-clause ──┬──┬──────────────────────────────────────────▶
   │          │  │                            │  │
   │          │  └────────────────────────────┘  │
   │          │                                  │
   └──────────┘                                  │
```

where:

contains-clause is:

```
▶──┬──┬──────────────────┬──┬─single-column-clause ──┬──┬──────────────────────────────────▶
   │  │                  │  │                        │  │
   │  ├─( number ) ──────┤  │                        │  │
   │  │                  │  │                        │  │
   │  └──────────────────┘  ├─group EXPAND ──────────┤  │
   │                        │                        │  │
   └────────────────────────┘                        │  │
                                                     │  │
```

```
▶──┬──┬─ASENDING ──┬──┬──────────────────────────────────────────────────────────────────▶
   │  │            │  │
   │  ├─DESCENDING ┤  │
   │  │            │  │
   └──┘            │  │
                   │  │
```

where:

number is the number of columns in a column set.

single-column-clause is:

```
▶──┬─item ──┬──┬──┬──┬─KNOWN-AS local-name ──┬──┬──────────────────────────────────────────▶
   │        │  │  │  │                       │  │
   │        │  │  │  │                       │  │
   │        │  │  │  │                       │  │
```

local-name is the name of the column, consisting of no more than 18 characters.

sql-block is the PL/SQL block that ORACLE executes to fire the trigger.

yes-clause is:

```
──┬──────────────────────────────┬──────────────────────────────────────►
  │ CONDITION 'condition'        │
  └──────────────────────────────┘

──┬──────────────────────────┬──┬──────────────────────────┬──────────►
  │ REF-OLD 'old'            │  │ REF-NEW 'new'            │
  └──────────────────────────┘  └──────────────────────────┘
```

where:

condition is a SQL condition to specify the trigger restriction.

old is a correlation name.

new is a correlation name.

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-USER

```
►►─ORACLE-USER ──────────────────────────────────────────────────────►
                 └─AS user ─┘

──┬──────────────────────────┬──┬──────────────────────────┬──────►
  │ PASSWORD 'password'      │  │ CONTAINS profile         │
  └──────────────────────────┘  └──────────────────────────┘

──┬──────────────────────────────────────────────────────────────────►
  │                 <<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<
  │ SYNONYMS synonym-name FOR ──┬─table─┬──────────────────────────┤
  │                             └─view──┘                          │
  └────────────────────────────────────────────────────────────────┘

──┬──────────────────────────────────────────────────────────────────►
  │      <<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<                           │
  │ SEE tablespace FOR ' for-clause'                                 │
  └──────────────────────────────────────────────────────────────────┘

──┬──────────────────────────┬──┬─────┬──────────────────────────►◄
  │ common clauses           │  │ ;   │
  └──────────────────────────┘  │ .   │
                                └─────┘
```

where:

user is the name of an ORACLE-USER member.

password is a password.

profile is the name of an ORACLE-PROFILE member.

synonym is a synonym of no more than 18 characters.

table is the name of an ORACLE-TABLE member.

view is the name of an ORACLE-VIEW member.

tablespace is the name of an ORACLE-TABLESPACE member.

for-clause is:

```
DEFAULT
TEMPORARY        QUOTA space
                 QUOTA space
```

where:

space is:

```
integer   K
          M
UNLIMITED
```

where *integer* is an integer

Refer to *ASG-Manager Products Dictionary/Repository User's Guide* for details of common clauses.

ORACLE-VIEW

```
ORACLE-VIEW   AS view

CREATOR-OWNER user   column-spec

WITH-CHECK-OPTION   SELECT ALL
                           DISTINCT

FROM <<<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<<<<
     table   CORRELATION-NAME correlation
     view

WHERE 'string'   HAVING 'string'
```

```
<<<<<<<<<<<<<<<<<<<<<<< , <<<<<<<<<<<<<<<<<<<<<<<
SEE joined-view ---------- FOR --- 'UNION'
                                  'UNION ALL'
                                  'INTERSECT'
                                  'MINUS'

WITH-CHECK-OPTION
                    CONSTRAINT 'constraint-name'

common clauses ---- ; ----
                    . ----
```

where:

view is the name of an ORACLE-VIEW member.

user is the name of a ORACLE-USER member.

column-spec is:

```
                <<<<< , <<<<<<
                CONSTRAINT 'contains-clause'
ENTERED-AS
HELD-AS
REPORTED-AS
DEFAULTED-AS
```

where:

contains-clause is:

```
                single-column-clause
( number )
                group EXPAND

COLUMN-NAME name        EXPRESSION 'expression'
```

where:

number is the number of columns in a column set.

single-column-clause is:

```
item
     version        KNOWN-AS local-name
group
```

where:

`item` is the name of an ITEM member.

`version` is an integer in the range 1 to 15.

`group` is the name of a GROUP member.

`local-name` is the name of the column, consisting of no more than 18 characters.

`group` is as defined above.

`name` is the name of the column in the view and consists of no more than 18 characters.

`expression` is an expression of no more than 255 characters, within delimiters.

`table` is the name of an ORACLE-TABLE member.

`view` is the name of an ORACLE VIEW member.

`string` is a valid SQL search condition.

`joined-view` is the name of an ORACLE-VIEW member.

If the ORACLE-VIEW is constructed via a set operation of two ore more subselect statements, the ORACLE-VIEWs which contain the subselect statements have to be referenced under SEE. The FOR clause specifies the type of set operation.

`constraint-name` is the name assigned to the check option constraint

Refer to *ASG-Manager Products Oracle Interface Reference Guide* for details of common clauses.

Appendix

Oracle Member Types

The following section provide an overview of the Oracle member types, relationships, attributes, and common clauses supported by the import facility.

Overview of the Supported Common Clauses

The generation of common clauses contents is controlled by the executive routine MPDYWTDFLR (see "Other Tailorable Executive Routines" on page 47). The import facility supports the following common clauses by default:

- NOTE
- ALIAS ORACLE

NOTE is updated with information about import date and time, and about the MANAGER Products user using the import facility.

ALIAS ORACLE is updated with the original ORACLE name of the object extracted from the ORACLE database. To enable this alias type use the command:

```
CONTROL NEW-ALIAS ADD ORACLE;
```

Overview of the Supported Oracle Member Types

This overview lists for each ORACLE member type:

- The names of the ORACLE system tables exploited by the import facility
- The attributes updated by the import facility
- The relationships to other ORACLE member types

ORACLE-CLUSTER (Prefix ORCL-)

- ORACLE system tables

  ```
  DBA_CLUSTERS
  DBA_DATA_FILES
  DBA_TAB_COLUMNS
  ```

- Attributes updated by the import facility

  ```
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  INITRANS
  MAXTRANS
  PCTFREE
  PCTUSED
  HASHKEYS
  ```

- Relationships to other ORACLE member types established by the import facility:

  ```
  CREATOR-OWNER        --> ORACLE-USER
  DATAFILE             --> ORACLE-FILE
  IN                   --> ORACLE-TABLESPACE
  COLUMNS CONTAINS     --> ITEM
  ```

ORACLE-DATABASE (Prefix ORDB-)

- ORACLE system tables

  ```
  V$DATABASE
  V$LOGFILE
  V$DATAFILE
  ```

- Attributes updated by the import facility:

  ```
  -/-
  ```

- Relationships to other ORACLE member types established by the import facility:

  ```
  DATAFILE       --> ORACLE-FILE
  LOGFILE        --> ORACLE-FILE
  ```

ORACLE-DBLINK (Prefix ORDL-)

- ORACLE system tables

  `DBA_DB_LINKS`

- Attributes updated by the import facility

  `USING`

- Relationships to other ORACLE member types established by the import facility:

  `-/-`

ORACLE-FILE (Prefix ORFI-)

Note:

The generated ORACLE-FILE member name is derived from the file and directory name extracted from the ORACLE system table.

- ORACLE system tables:

  `DBA_DATA-FILES`

- Attributes updated by the import facility:

  `SIZE`
  `PATH`

- Relationships to other ORACLE member types established by the import facility:

  `-/-`

ORACLE-INDEX (Prefix ORIX-)

- ORACLE system tables

  ```
  DBA_INDEXES
  DBA_IND_COLUMNS
  ```

- Attributes updated by the import facility

  ```
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  INITRANS
  MAXTRANS
  PCTFREE
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CREATOR-OWNER      --> ORACLE-USER
  SEE                --> ORACLE-TABLESPACE
  ON                 --> ORACLE-TABLE  (dummy reference)
  CONTAINS           --> ITEM          (dummy reference )
  ```

ORACLE-PRIVILEGE (Prefix ORPV-)

- ORACLE system tables

  ```
  DBA_SYS_PRIVS
  DBA_TAB_PRIVS
  ```

- Attributes updated by the import facility

  ```
  GRANTABLE
  SELECT
  ALTER-PRIVILEGE
  CREATE-PRIVILEGE
  DROP-PRIVILEGE
  GENERAL-PRIVILEGE
  ```

- Relationships to other Oracle member types established by the import facility:

  ```
  TO           --> ORACLE-USER
  ON           --> ORACLE-TABLE
  ```

Note:

The generated ORACLE-PRIVILEGE member name is derived from the user name and from the privilege type granted to the user.

ORACLE-PROFILE (Prefix ORPR-)

- ORACLE system tables

  `DBA_PROFILES`

- Attributes updated by the import facility

  ```
  SESSION_P_U
  CPU_P_S
  CPU_P_C
  CONNECT_TIME
  IDLE_TIME
  LOGICAL_READS_P_S
  LOGICAL_READS_P_C
  COMPOSITE_LIMIT
  PRIVATE_SGA
  ```

- Relationships to other ORACLE member types established by the import facility:

  `-/-`

ORACLE-ROLE (Prefix ORRL-)

- ORACLE system tables

  `DBA_ROLES`

- Attributes updated by the import facility

  `PASSWORD`

- Relationships to other ORACLE member types established by the import facility:

  `-/-`

ORACLE-SEQUENCE (Prefix ORSQ-)

- ORACLE system tables

  ```
  DBA_SEQUENCES
  ```

- Attributes updated by the import facility

  ```
  CYCLE
  ORDER
  START
  INCREMENT
  MINVALUE
  MAXVALUE
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CREATOR-OWNER  --> ORACLE-USER
  ```

ORACLE-SNAPSHOT (Prefix ORSN-)

- ORACLE system tables

  ```
  DBA_SNAPSHOTS
  DBA_TABLES
  ```

- Attributes updated by the import facility

  ```
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  INITRANS
  MAXTRANS
  PCTFREE
  PCTUSED
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CREATOR-OWNER    --> ORACLE-USER
  FROM             --> ORACLE-TABLE (dummy reference)
  CLUSTER          --> ORACLE-CLUSTER
  SEE              --> ORACLE-TABLESPACE
  CONTAINS         --> ITEM
  ```

ORACLE-SNAPLOG (Prefix ORSL-)

- ORACLE system tables

  ```
  DBA_SNAPSHOT_LOGS
  DBA_TABLES
  ```

- Attributes updated by the import facility:

  ```
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  INITRANS
  MAXTRANS
  PCTFREE
  PCTUSED
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CREATOR-OWNER --> ORACLE-USER)
  ON            --> ORACLE-TABLE      (dummy reference )
  SEE           --> ORACLE-TABLESPACE
  ```

ORACLE-TABLE (Prefix ORTA-)

- ORACLE system tables

  ```
  DBA_TABLES
  DBA_DATA_FILES
  DBA_TAB_COMMENTS
  OBJ$
  DBA_USERS
  TS$
  USER$
  DBA_CONSTRAINTS
  DBA_CONS_COLUMNS
  ```

- Attributes updated by the import facility:

  ```
  ROWS
  COMMENT
  OBID
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  INITRANS
  MAXTRANS
  PCTFREE
  PCTUSED
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CREATOR-OWNER      --> ORACLE-USER
  DATAFILE           --> ORACLE-FILE
  CLUSTER            --> ORACLE-CLUSTER
  IN                 --> ORACLE-TABLESPACE
  COLUMNS CONTAINS   --> ITEM
                       (with primary key clause and foreign key clause)
  ```

ORACLE-TABLESPACE (Prefix ORTS-)

- ORACLE system tables

  ```
  DBA_TABLESPACES
  DBA_DATA_FILES
  USER$
  TS$
  ```

- Attributes updated by the import facility

  ```
  INITIAL
  NEXT
  MINEXTENTS
  MAXEXTENTS
  PCTINCREASE
  ```

- Relationships to other ORACLE member types established by the import facility:

  ```
  DATAFILE        --> ORACLE-FILE
  ```

ORACLE-TRIGGER (Prefix ORTG-)

- ORACLE system tables

  ```
  DBA_TRIGGERS
  ```

- Attributes updated by the import facility

  ```
  OPTION
  ACTION
  CONDITION
  SQL
  ```

- Relationships to other Oracle member types established by the import facility

  ```
  CREATOR-OWNER       --> ORACLE-USER
  ON                  --> ORACLE-TABLE  (dummy reference )
  CONTAINS            --> ITEM          (dummy reference )
  ```

ORACLE-USER (Prefix ORUS-)

- Oracle system tables

  ```
  DBA_USERS
  ```

- Attributes updated by the import facility

  ```
  PASSWORD
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  CONTAINS     --> ORACLE-PROFILE
  SEE          --> ORACLE-TABLESPACE
  ```

ORACLE-VIEW (Prefix ORVI-)

- ORACLE system tables

  ```
  DBA_VIEWS
  DBA_DEPENDENCIES
  DBA_TAB_COMMENTS
  DBA_TAB_COLUMNS
  DBA_COL_COMMENTS
  DBA_SYNONYMS
  ```

- Attributes updated by the import facility:

  ```
  COMMENT

  SQL
  ```

- Relationships to other ORACLE member types established by the import facility:

  ```
  CREATOR-OWNER     --> ORACLE-USER
  FROM              --> ORACLE-TABLE(dummy reference )
  CONTAINS          --> ITEM
  ```

ITEM (Prefix IT-)

Note:

Information about related COLUMNs is extracted for these object types:

```
CLUSTER
INDEX
SNAPSHOT
TABLE
TRIGGER
VIEW
```

ITEM members document that information.

The import facility converts the ORACLE data types of the columns into corresponding DB2 data types. (The export facility works the other way round.)

Table 1 • Correlation between ORACLE and DB2 data types

ORACLE data type	DB2 data type
VARCHAR2	VARCHAR
LONG	VARCHAR
RAW	GRAPHIC
LONG RAW	GRAPHIC
DATE	DATE, TIME or TIMESTAMP
NUMBER	FLOAT
NUMBER(38)	INTEGER or SMALLINT

- ORACLE system tables

  ```
  DBA_TAB_COLUMNS
  DBA_COL_COMMENTS
  ```

- Attributes updated by the import facility

  ```
  COMMENT
  NULLS
  NUMBER
  TYPE
  LENGTH
  DEFAULT
  PRECISION
  ```

- Relationships to other ORACLE member types established by the import facility

  ```
  -/-
  ```

Index

C

column attributes 10
column types 9
commands
    ORACLE ALTER 18
        syntax 24
    ORACLE CREATE 10
        syntax 16
    ORACLE DROP 26, 30
    ORACLE GRANT 31
        syntax 34
    ORACLE REVOKE 31
        syntax 34
    ORACLE SYNONYM 35
        syntax 37
common clauses
    supported 81
conventions page vi

D

datasets
    installation 1

E

environment
    Manager Products
        importing 46
        importing information 42
export routine 9

G

generated output
    options 15

I

impact analysis report 27
    tailoring 27
import facility
    Oracle member types 81
import interface 41
importing information
    Manager Products environment 42
installation
    datasets 1
installing
    Oracle components 6

M

MPDY12PROF
    exporting the exit 5
MPDY42DFLT
    importing the exit 5
MPDYWTDFLT
    importing the exit 5

N

name editing
    options 39

O

ORACLE ALTER 18
    syntax 24
ORACLE CREATE 10
    syntax 16
ORACLE DROP 26
    syntax 30
ORACLE GRANT 31
    syntax 34
ORACLE REVOKE 31
    syntax 34
Oracle RIM
    enabling 1
ORACLE SYNONYM 35, 37
ORACLE-CLUSTER 51
ORACLE-DATABASE 54
ORACLE-DBLINK 55
ORACLE-FILE 56
ORACLE-INDEX 56
ORACLE-PACKAGE 58
ORACLE-PRIVILEGE 59
ORACLE-PROFILE 63
ORACLE-ROLE 64
ORACLE-SCHEMA 65
ORACLE-SEQUENCE 66

ORACLE-SNAPSHOT 68
ORACLE-TABLE 71
ORACLE-TABLESPACE 74
ORACLE-TRIGGER 75
ORACLE-USER 77
ORACLE-VIEW 78
output
    generation options 38

P

primary key indexes
    creating 13

R

Repository Information Model
    modifying 8
repository member types 51
    ORACLE-CLUSTER 51
    ORACLE-DATABASE 54
    ORACLE-DBLINK 55
    ORACLE-FILE 56
    ORACLE-INDEX 56
    ORACLE-PACKAGE 58
    ORACLE-PRIVILEGE 59
    ORACLE-PROFILE 63
    ORACLE-ROLE 64
    ORACLE-SCHEMA 65
    ORACLE-SEQUENCE 66
    ORACLE-SNAPLOG 66
    ORACLE-SNAPSHOT 68
    ORACLE-TABLE 71
    ORACLE-TABLSPACE 74
    ORACLE-TRIGGER 75
    ORACLE-USER 77
    ORACLE-VIEW 78

S

spool files
    transferring to host 46
system tables
    creating 7
    extracting information 42

U

User Defined Relationships
    handling 2