



HOME

User Manual

DELMIA Process Engineer[®]

ORACLE 11g Installation



Foreword

This manual provides an introduction to the basic operations and functions of the ORACLE 11g Installation.

While developing these functions we have made every effort to create a clearly organized, easy-to-understand program structure.

A user-friendly interface as well as a clear menu guide will enable you to quickly learn how to operate the program and to get familiar with its functions so that you can carry out your planning tasks in a quick and reliable way.

No Liability or Guarantee

Our programs and manuals have been compiled with great care and to the best of our knowledge. They have also been tested in a production setting. However, we assume no liability and provide no guarantee that the software and related descriptions are free of error or are suitable for special purposes.

DELMIA assumes no liability for any damage that may arise from the use of this software. By using this software, the user acknowledges this exclusion from liability and shall hold DELMIA exempt from all claims.

Copyright

The information in our documents may be copied and distributed for internal purposes provided it is done free of charge and the contents are not altered or distorted.

Any other form of usage, especially the sale on CD-ROM or in any other publication in whole or in part is only permitted after prior written consent by DELMIA.

Some parts of this software are owned by Unigraphics Solutions Inc. and are copyrighted © 2010. All rights reserved.

Some parts of this software are owned by combit® GmbH and are copyrighted. Report-/Print module List and Label® Version 8.0: Copyright combit® GmbH 1991-2010.

Modifications

Moreover, DELMIA retains the right to make modifications and improvements to the product described in this manual at any time without prior notification.

DELMIA and the 3DS logo are registered trademarks of Dassault Systèmes or its subsidiaries, in the United States or other countries.

© 2001-2010 Dassault Systèmes - All rights reserved

Table of Contents

1. Introduction	1
1.1 How to Use this Manual	1
1.2 Documentation Conventions and Symbols	2
1.3 New Functions in ORACLE 11g Installation	3
2. Installation Requirements	4
3. My First Database	5
4. Migrating an ORACLE 9i Database	6
5. Uninstalling ORACLE 9i from Windows	7
5.1 Stopping Running Services	7
5.2 Removing the ORACLE Entries from the Registration Database	7
5.3 Deleting the ORACLE Service Entries from the CURRENT_CONTROL_SET	7
5.4 Removing the ORACLE Files	8
5.5 Removing the Data Source	8
6. Installing the ORACLE Software	9
6.1 Client Settings	9
7. Creating a Database	10
7.1 Memory Areas	10
7.2 Undo Management	10
7.3 New Tablespaces	11
7.4 Schema	12
7.5 Log Parameter	13
7.6 SPFILE	13
7.7 Instance Settings	14
7.7.1 New ORACLE Parameters	14
7.8 Indices	17
8. Export and Import	19
8.1 Exporting Data from a Database	19
8.2 Importing Data into the Database	19
8.3 Importing the Process Engineer Database	20
8.3.1 User Schema	20
8.3.2 Access Rights	21

8.4	Tablespaces for Users	22
8.4.1	Creating Roles	22
8.4.2	Creating Blob Tables	23
9.	Datapump	24
10.	Process Engineer Database	25
11.	Upgrading a Database	27
12.	Using the DPE Database Assistant	28
13.	Maintenance of the Database	29
13.1	Performance Tuning	29
13.2	Maintenance Work	29
	List of Figures	37
	Index	38

1. Introduction

This manual explains how to use the Process Engineer ORACLE 11g Installation for your planning purposes.

It shows how to install and set up the ORACLE database so you can use it when working with DELMIA software.

This manual has been written mainly for system administrators and employees with administrator rights. The functions described in the following must be executed before the DELMIA Process Engineers or the ENOVIA Manufacturing Hubs are installed.

The information in this manual is necessary in order to successfully run the *ENOVIA Manufacturing Hub* on the ORACLE database. There are, however, further notes on the installation, performance, and other topics in the [Administration Manual](#).

1.1 How to Use this Manual

This manual enables you to get familiar with the operation and functions of the Process Engineer. This manual briefly describes:

- ORACLE 11g Installation

This manual is relevant for three types of user groups, and these groups should select the chapters important to them. The first user group includes those who have no previous knowledge of databases and want to install ORACLE on Windows. This group should read the following chapters:

- [Installation Requirements](#)
- [My First Database](#)
- [Process Engineer Database](#)

These chapters help you to install and configure a typical database for using DELMIA software in a few minutes. No special previous knowledge is necessary to do this. The advantage is that you can quickly start to work with DELMIA software.



Note

Standard parameters which suffice for average-sized databases is used here. An Oracle database administrator should check which requirements need to be met in individual cases, and if necessary, change the values accordingly. Further information can be found in the official ORACLE documentation.

The second user group comprises the ORACLE experts or those who want to use ORACLE with UNIX. This group should read the following chapters:

- [Installation Requirements](#)
- [Installing the ORACLE Software](#)
- [Creating a Database](#)

- [Importing Data into the Database](#)

The third user group includes ORACLE experts who have already installed a DELMIA database and only want to execute a migration to V5R17. This group should read the following chapters:

- [Migrating an ORACLE 9i Database](#)
- [Upgrading a Database](#)
- [Export and Import](#)



Note

When handling the ORACLE 11g Installation functions, please also refer to the general introduction to Process Engineer in the General Introduction Manual.



Click [General Introduction](#) to access the manual.

1.2 Documentation Conventions and Symbols

The symbols used in this manual are intended to provide you with keys to the contents in an immediately understandable manner.



This symbol is used to introduce key concepts that are covered in the sections immediately following this symbol. As a result, this symbol most frequently appears at the beginning of chapters or sections.



Note

*This symbol is used to mark notes, which provide you with additional information you need to have for further work. You will either find the Note sign at the beginning of a chapter or in a particular text passage in the chapter. Texts bearing this sign are additionally marked with **Note**. The text is always in italics.*



Caution

*This symbol indicates that the text that follows describes particular circumstances that you must avoid to avoid potential errors with the operation of the program or harm to data. You will either find the Caution sign at the beginning of a chapter or near a particular text passage in the chapter. Texts that are introduced by this sign are additionally marked with **Caution**. The text is always in italics.*

Example

This symbol marks examples which serve to illustrate a certain situation.

1)

This symbol marks the individual operational steps involved in a particular operating instruction. Operating instructions describe operational steps, for example, how to open a menu or execute a function.



This symbol marks listed subjects. The symbol for listed subjects can be either used to structure a continuous text or to list main subject keywords.

- This symbol marks list inside a bulleted or numbered list.



This symbol marks cross reference information that is available in another manual.

1.3 New Functions in ORACLE 11g Installation

No new functionality has been added for this release.

2. Installation Requirements

Make sure that the following requirements have been met before starting the installation.

Installations Requirements

- DVD with ORACLE 11g Database.
- Standard Edition or ORACLE 11g Database Enterprise Edition.
- Version 11.1.0.6 from ORACLE is a minimum requirement for the R20.

Hardware Requirements

- At least 2400 MHz CPU, at least 2GByte main memory.
- 10GB of free space on the hard disk.

Software Requirement

- Windows 2000 or Windows XP.

3. My First Database

- Please insert the **DVD** with ORACLE 11g Database Standard Edition in the DVD drive of the machine on which you have to install the database.
- Start the program **setup.exe** in the main directory of the DVD and follow the installation instructions on the screen.

4. Migrating an ORACLE 9i Database

As of version PE R17 **ORACLE 9i and ORACLE 11g** are supported. In order to do this, migrate the database to ORACLE 11g.

- In addition you have to migrate ORACLE 11g on every Oracle client (on all computers on which the PPR Server has been installed).

ORACLE 9i includes the "Locally Managed Tablespaces" aspect, in particular the "System Tablespaces".

However a Locally Managed SYSTEM - Tablespace is not achievable by migrating a database, but rather only by creating a new one with ORACLE 9i Release2 or Oracle 11g.

A migration is a very complex and often also a very time-consuming process. Before a migration it is necessary to read the current Readme files provided by Oracle's support. Please also consider relevant literature by Oracle.



Note

The following are the supported versions, which meet the minimum requirements and from which a change to ORACLE 11g can be performed directly - 8.1.7.4 / 9.2.0.4 / 9.2.0.5 / 9.2.0.6 / 9.2.0.7 / 9.2.0.8

We recommend migration with the help of the conventional Export/Import function. Even though more storage space might be needed and the migration might take a bit longer, it results in the following benefits – Reorganization of objects/If the exported Dump is not deleted explicitly, you can return the old version.

5. Uninstalling ORACLE 9i from Windows

You can uninstall the ORACLE 9i on the database and on the DELMIA PPR machines. If, however, you have no resource problems regarding hard disk space or the CPU, you can leave ORACLE 9i installed.



Note

You should make a backup of your database before deinstalling ORACLE. This task should be undertaken by an ORACLE Administrator. For a backup of your data, please refer to the [Export and Import](#).

5.1 Stopping Running Services

- 1) Start the dialog for showing the services on your operating system.
Start->Settings->Control Panel->Administrative Tools->Services.
- 2) Search for services with the prefix ORACLE that have the status 'Started' and stop them by selecting 'Stop' in the context menu.

5.2 Removing the ORACLE Entries from the Registration Database

You can find the entries added by ORACLE in the Windows registration database under the key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\Homex.
```

A database instance previously installed by DELMIA can be recognized by the entry ORACLE_SID if it contains the value 'EPE5' or 'PE8i'.

- In this case take note of the path that is entered under the key ORACLE_HOME. This is the uppermost directory under which the files of the ORACLE installation to be replaced are found, i.e. the files which are to be deleted.
- Delete the following key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\Homex.
```

5.3 Deleting the ORACLE Service Entries from the CURENT_CONTROL_SET

- 1) Delete the keys for which the path in which the instance to be deleted is entered in the ImagePath value. The keys are called:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Oracle ...
```

5.4 Removing the ORACLE Files

Delete the content and all subdirectories in the file explorer in the directory ORACLE_HOME, the value of which you have taken note of in the section [Removing the ORACLE Entries from the Registration Database](#).

5.5 Removing the Data Source

removing the data source/origin

- 1) Look for the following key in the registration database

HKEY_LOCAL_MACHINE->Software->ODBC->ODBCINST.INI

- 2) Search for a driver with the prefix ORACLE.

- Take note of the most current value of the driver field (e.g. D:\ORACLE\ORA92E\BIN\SQORA32.DLL).

- 3) Repeat this for the key with the prefix ORACLE.

Then delete all keys with the prefix ORACLE.

Also delete character strings under the key

HKEY_LOCAL_MACHINE->Software->ODBC->ODBCINST.INI->ODBC Drivers

which refer to keys with the prefix ORACLE.

- 4) Search for the following key in the registration database

HKEY_LOCAL_MACHINE->Software->ODBC->ODBC.INI

- 5) Delete the keys which have the noted value in the driver field and thus refer to an ORACLE driver.

- 6) Search for the following key in the registration database

HKEY_LOCAL_MACHINE->Software->ODBC->ODBC.INI->ODBC Data Sources

- Delete all character strings whose values refer to the ORACLE keys which you have deleted.

6. Installing the ORACLE Software



Note

You should clarify with your administrator which ORACLE settings are to be used before installing the DELMIA software and setting up the PPR hub.

The values given in this manual merely serve to make tried and proven settings available to all customers. Settings and requirements for correct operation of DELMIA software appear in *italics* and in *color*.

6.1 Client Settings

On each PPR server machine one ORACLE client runs. Set the time for expiry of sessions to five minutes for each ORACLE client.

To do so, the following value must be entered in the file **<ORACLE path>\network\admin\sqlnet.ora**:

```
SQLNET.EXPIRE_TIME=5
```

A dedicated server process is started for every ORACLE client.

The following are part of a dedicated server system:

- **tnsnames.ora**: Configuration file for converting the net service names into a server address
- **sqlnet.ora**: Configuration file on the client and server side (optional) listener:
- **Listener**: Server-side process that responds to connection requests.

7. Creating a Database

7.1 Memory Areas

Locally Managed Tablespaces and Automatic Segment-Space Management (ASSM).

- Locally Managed Tablespaces were introduced in ORACLE 8i.
- ASSM with ORACLE 9i.
- Only permanent Locally Managed Tablespaces can be created with ASSM.
- Automatic Segment Space Management determines how a free and assigned memory space can be managed.
- The extent management via Locally Managed Tablespaces with the AUTOALLOCATE option is ORACLE's attempt to make all storage parameters redundant.

The following options are available:

- MANUAL
- AUTO

In order to receive sufficient information on *Locally Managed Tablespaces* and ASSM, the knowledge of a **certified** ORACLE database administrator is necessary.



Note

At this point we would like to point out to you that errors may occur while working with Locally Managed Tablespaces and Automatic Segment Space Management in connection with lobs (Large Object Blocks). These errors are described in further detail in the ORACLE error database. The corresponding error numbers are 2784201, 3029292, and 3213101.

Lobs are used, for example, with

- Detailings (data which can be seen only in the V5 release)
- Data cards and scripting.

7.2 Undo Management

Tablespaces are required for working with the Undo Management. Names for the tablespaces can be selected individually by the database administrator (DBA) for the respective customers.

- In Version 9 is recommended to create an undo tablespace and to set the parameter UNDO_MANAGEMENT =AUTO in the initialization file.
- The database must be restarted in order to switch between the modes.
- It is not possible to use both methods within a database instance. If the automatic Undo Management is activated, the rollback segments are

managed by the system and the explicit creation of rollback segments does not apply.

- The first tablespace of the type UNDO found in the database is used as the undo tablespace. A specific undo tablespace can be set optionally by using the parameter UNDO_TABLESPACE in the initialization file.

The following ORACLE parameters are no longer used if you use the automatic Undo Management:

- ROLLBACK_SEGMENTS
- TRANSACTIONS
- TRANSACTIONS_PER:ROLLBACK_SEGMENT
- MAX_ROLLBACK_SEGMENTS

New ORACLE parameters, if an Undo Tablespace is used

UNDO_MANAGEMENT

If this parameter is set to AUTO, the Undo Management mode is used. If it set to MANUAL, the manual Undo Management mode is used (explicitly created rollback segments)

UNDO_TABLESPACE

This is a dynamic parameter which specifies the name of the Undo Tablespace used.

UNDO_RETENTION

Specifies the time in seconds in which the information for previously completed transactions are retained. This ensures reading consistency for transactions which take a long time.

UNDO_SUPPRESS_ERRORS

It is possible to hide error messages issued with SQL statements in manual Undo Management for the automatic Undo Management. (TRUE = hide, FALSE = show)

7.3 New Tablespaces

SYSAUX Tablespace

Besides The SYSTEM Tablespace ORACLE 11g introduces a new Duty Tablespace as an extension to SYSTEM Tablespace. It can neither be renamed nor be deleted.

By means of this tablespace:

- System and non-system data are separated
- System Tablespace are relieved
- A storage area is offered for components like the Enterprise Manager Repository, the LogMiner etc.

The SYSAUX Tablespace is

- Permanent
- Read Write

And the following storage areas apply:

- Extent Management Local
- Segment Space Management: AUTO

DEFAULT Table Space

Not only a Default Temporary Tablespace can be assigned to a database in ORACLE 11g, but also a Default Permanent Tablespace. As a rule, the previous default was "SYSTEM".

The commands CREATE DATABASE and ALTER DATABASE have been extended correspondingly. If the "create user" statement contains no Default Tablespace, the Default Permanent Tablespace is used.

Example

SQL> Create database O11g ... default tablespace ERGO_USR data file ...

For setting after creating the database:

SQL> Alter database O11g Default Tablespace "ERGO_USR";

The View DATABASE_PROPERTIES contains in the column PROPERTY_NAME the "default_permanent_tablespace" to which the value "ERGO_USR" (Column PROPERTY_VALUE) must now be assigned.

Bigfile Tablespace (BFT)

For the sake of completeness, it must be mentioned that ORACLE 11g introduces a new tablespace, which allows for management of much larger data volumes than before. The type so far known continues to exist under the name **Smallfile Tablespace**. The default for creation of a databases is Smallfile Tablespace. A Bigfile Tablespace should be used when the database has volume has exceeded the maximum size of 512 PetaBytes so far allowed.

Also new are the so called **Temporary Tablespace Groups** (TTG). With parallel sorting over several partitions and with a Single-SQL on a Bigfile Tablespace, working with a TTG is recommended.



Note

An increase in performance through the innovations is noticed only in extreme and specific application areas. In contrast the administrative extensions, like SYSAUX Tablespace, result in considerable relief through clear structuring.

7.4 Schema

The following names are recommended for ORACLE11g:

- Tablespace for UNDO **UNDOTBS1**
Size: Minimum 200MB
 Storage parameter:
 AUTOEXTEND ON NEXT 10M MAXSIZE 32767M
- Tablespace for SYSTEM **SYSTEM**
Size: Minimum 300 MB
 Storage parameter:
 AUTOEXTEND ON NEXT 10M MAXSIZE 32767M
- Tablespace for SYSTEM **SYS_AUX**
Size: Minimum 110MB
 Storage parameter:
 AUTOEXTEND ON NEXT 10M MAXSIZE 32767M EXTENT
 MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO

- Tablespace for USER **ERGO_USR**
Size: Minimum 500 MB
Storage parameter:
AUTOEXTEND ON NEXT 50M MAXSIZE 32767M EXTENT
MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
- Tablespace for INDEX **ERGO_IND**
Size: Minimum 500MB
Storage parameter:
AUTOEXTEND ON NEXT 50M MAXSIZE 32767M EXTENT
MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
- Tablespace for **TOOLS**
Size: Minimum 10MB
Storage parameter:
AUTOEXTEND ON NEXT 10M MAXSIZE 32767M EXTENT
MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO
- Tablespace for **TEMPORARY** (Default temporary tablespace)
Size: Minimum 500MB
Storage parameter:
AUTOEXTEND ON NEXT 10M MAXSIZE 32767M EXTENT
MANAGEMENT LOCAL UNIFORM SIZE 10M

7.5 Log Parameter

The following are recommended:

- Three redo log file groups with at least two files, minimum size of 30MB
- About 5 to 10 ORACLE log switches per hour should occur. The size of the log file and number of groups for log files should be set based on this value.
- You should check that no waiting time occurs for the archiving processes.

7.6 SPFILE

Please note, that ORACLE 11g Database has a binary server parameter file on the server. In contrast to the text-based PFILE, parameters can not be changed in a text editor. Oracle recommends using these SPFILES. If no SPFILE is found at startup, then the PFILE (init.ora) is used.

Any changes of parameters in SPFILES can be achieved by the ALTER SYSTEM command and written to the Spfile with "scope = spfile".

The advantages of these SPFILES include, for example:

- Persistent changes with ALTER SYSTEM (if requested)
- Oracle can write to the SPFILES (self-tuning parameters)

Where Oracle 11g is running under Windows, a PFILE is set up in the <ORACLE_HOME>/Database directory, which is linked to the SPFILE in the dbf directory.

7.7 Instance Settings

When creating an instance, the following settings should be made:

- The character set UTF8 is very important, since it cannot be changed once an instance has been created.

```
MAXLOGFILES 30
MAXLOGMEMBERS 3
MAXLOGHISTORY 800
MAXDATAFILES 256
MAXINSTANCES 1
CHARACTER SET UTF8
NATIONAL CHARACTER SET UTF8;
```

7.7.1 New ORACLE Parameters

OPTIMIZER_MODE

Choose and rules are not supported anymore, because support for the rule based optimizer has dropped. Unofficially though, these options can still be used.

STATISTICS_LEVEL

There are three new Statistic Levels. Basic, Typical, and All. The status of the current statistics can be found in the V\$STATISTICS_LEVEL view. Further information can be found in the chapter Maintenance Work.

- **Basic:** No statistics is created
- **Typical:** Generates statistics for the Shard Pool, SQL Execution Memory (PGA) and Recovery Time Estimation. Therefore the parameter DB_CACHE_ADVICE becomes invalid in 9.2.
- **All:** Like Typical + Row Source Execution Statistics + Timed OS Statistics
- PGA_AGGREGATE_TARGET =

Pga_aggregate_target must be set. If PGA_AGGREGATE_TARGET is not set in ORACLE 11g, 20% of the SGA is reserved for the PGA.

SHARED_POOL_SIZE

So far an overhead has been added automatically by Oracle to the parameter value stated by the user. As of Version 11g it must be included in the calculation by the administrator.

SGA_TARGET

SGA_TARGET is used in connection with Automatic Memory Management.

Specifies the total size the SGA is allowed to use. With this parameter the following parameters are set automatically:

- db_cache_size
- shared_pool_size
- SHARED_POOL_SIZE
- java_pool_size

- `streams_pool_size`

The self-tuning mechanism is only available for the following parameters:

- `log_buffer`
- `db_keep_cache_size`
- `db_recycle_cache_size`

RECYCLEBIN

Default value: ON

- ON: After deleting, tables are transferred to the recycle bin.
- OFF: Tables are deleted immediately upon DROP.



Note

Creating statistics with Analyze Schema can lead to the error “ORA-38301: DDL/DML can not be executed for objects in Recycle-Bin” in Oracle 11g. Latest then, the Scheme Statistics should be created with “DBMS_STATS.GATHER_SCHEMA_STATS”.



Note

The following parameters are recommendations. Some of the parameters depend on the computer (resources), for example, the resources of the physical main storage. For this reason, the values suggested here can only be considered guidelines. Similar to the buffer cache, the common pool can be optimized only through exact measurements relating to your environment.



Note

The parameter 'session_cached_cursors' can also be optimized only through exact measurements. The idea behind the session cached cursor is to keep cursors for commands which are continually opened by an application in the cache of the individual process. The parameter sets the number of the cursors which are available on the process level.



Note

Automatic Memory Management (AMM) is a new feature in Oracle 11g. With Oracle9i it was for the first time possible to change the SGA parameter dynamically, that is without a re-start of the database. However, the changes had to be made through the DBA. In ORACLE 11g the database automatically monitors the changing demands for the individual areas of the SGA and allocates the main storage based on current demand. “Automatic Shared Memory Management” is one and the same feature.

If you use AMM you have to specify only three initialization parameters:

- `SGA_TARGET` is a new parameter in ORACLE 11g. It reflects the total size of SGA for AMM.
- `SGA_MAX_SIZE` is the static upper limit of the SGA
- `PGA_AGGREGATE_TARGET` defines the reserved main storage for sorting and hash joins. `PGA_AGGREGATE_TARGET` is supposed to finish all discussions about the setting of the parameter `SORT_AREA_SIZE`,

HASH_AREA_SIZE etc. The system is given an upper limit for all PGAs of all sessions; the distribution of space is then dependent on the loads.

SGA_TARGET and SGA_MAX_SIZE normally have the same size. If you set the parameter SGA_MAX_SIZE higher, you reserve a float for operating peaks of the database. After the start of the instance, SGA_TARGET can dynamically be raised to the maximum size set with SGA_MAX_SIZE. In contrast, the SGA_MAX_SIZE can not be changed without re-start of the database.



Note

The following settings are recommended for the operation of the ORACLE instance in ORACLE 11g: If AMM is used, a part of this parameter becomes obsolete and so is not listed anymore (db_cache_size, shared_pool_size, java_pool_size). 80MB of the shared pool size are needed for the Enterprise Manager alone.

sga_max_size = This value may need to be increased if the values for the db_cache_size, shared_pool_size, log_buffer etc. can no longer be set because an insufficient amount of main storage has been reserved.

- **#Cache and I/O**

db_block_size = 8192

db_block_size * db_file_multiblock_read_count = 64kb

db_file_multiblock_read_count = 8

- **#Pools**

java_pool_size = 0 # (min. 24MB, whenever JVM is used)

- **#Optimizer**

optimizer_mode = FIRST_ROWS

- **#Processes and Sessions**

processes = 400

- **#Cursors**

open_cursors = 300

session_cached_cursors = 50

- **#SMALL**

- **#Pools**

- # log_buffer = 2097152 # 2MB main storage

- **#Sort, Hash Joins, Bitmap Indexes**

pga_aggregate_target = 104857600 # 100MB main storage

- **# LARGE**

- **#Pools**

- # log_buffer = 3145728 # 3MB main storage

- **# Sort, Hash Joins, Bitmap Indexes**

```

▪ # pga_aggregate_target = 157286400 # 150MB main storage

▪ # System Managed Undo and Rollback Segments

undo_management = AUTO
undo_retention = 14400

▪ #Cursors

cursor_sharing = SIMILAR

▪ # Miscellaneous

compatible = 10.2.0.1.0

```

7.8 Indices

When configured or used in a certain way, or when certain scripts are employed, (e.g. searching for a certain attribute using the DELMIA Process Engineer finder), the indices supplied by DELMIA may not be sufficient to prevent a linear correlation of run time and data volume.

In this case you need to define your own indices. These indices are deleted during upgrade to a new release. Please be sure to observe the following instructions.

- Stop PPR server
 - Set index
- 1) For an index of the ORACLE table XDOX and the column M_Y select a number N, and assign a new name to the index. The index must not begin with the Prefix "IDX".
 - 2) Execute this statement as the database user e5_database.
 - 3) You may have to adapt the name of the table range for the indices.

```

DROP INDEX I_C_<N>;

CREATE INDEX I_C_<N> ON XDOX ( M_Y ASC ) TABLESPACE
'ERGO_INDXX';

```

- Determine the class number

```

SELECT CLASS_ID FROM POET_CLASSES WHERE TABLE_NAME =
'XDOX';

```

- The following statement prompts you for the class number.

```

DELETE FROM POET_INDEXES WHERE SQL_INDEX_NAME =
'I_C_<N>';

INSERT INTO POET_INDEXES (CLASS_ID, POET_INDEX_NAME,
SQL_INDEX_NAME, TABLE_NAME);
VALUES (&CLASS_ID, 'I_C_<N>', 'I_C_<N>', 'XDOX');
COMMIT WORK;

```

- Example for table XDOERGOCOMPPRODUCTDEFAULTC242V0 and column M_STRATTRIBUTE_40178V0:

```

CREATE INDEX I_C_1 ON XDOERGOCOMPPRODUCTDEFAULTC242V0
(M_STRATTRIBUTE_40178V0 ASC ) TABLESPACE 'ERGO_INDXX';

```

```
INSERT INTO POET_INDEXES (CLASS_ID, POET_INDEX_NAME,  
SQL_INDEX_NAME, TABLE_NAME) VALUES (242, 'I_C_1', 'I_C_1',  
'XDOERGOCOMPPRODUCTDEFAULTC242V0');  
COMMIT WORK;
```

**Note**

By sticking to a certain name convention, deletion of indices by the upgrade process can be prevented. Indices with the Prefix 'CI_' are not deleted by the upgrade. The customer, however, needs to ensure that the index still has the desired effect. It is possible that an index is not necessary anymore or, in some cases, even has a negative effect

8. Export and Import

The data in the database can be exported and imported with ORACLE.

- Exporting is simply a supplement to physically saving data. Nevertheless exporting and importing data supports the migration between the versions ORACLE 9i and ORACLE 11g.

8.1 Exporting Data from a Database

- 1) Start the command window on your operating system:
Start->**Run**->'cmd'
- 2) Set the following character set for database user e5_database:
`SET NLS_LANG=AMERICAN_AMERICA.UTF8`
- 3) Execute the export for e5_database for the database user, and specify the location (in the example, c:\temp) for the dump and for the log file:
 - `exp e5_database/databaseora file=e5_database.dmp log=exp_e5_database.log`
- 4) Set the following character set for database user e5_ergotime:
`SET NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1`
- 5) Then execute the export and create the memory space for the dump file and the log file:
 - `exp e5_ergotime/ergomas file=c:\temp\e5_ergotime.dmp log=c:\temp\exp_e5_ergotime.log`



Note

The parameters 'file' and 'log' serve to specify the file name of the file to be generated and to specify a log file in which a log of the export is filed.



Note

In versions previous to release 12, the Users e5_rightsdata must also be saved by executing the following command: `exp e5_rightsdata/rightsora file=c:\temp\e5_rightsdata.dmp log=c:\temp\exp_e5_rightsdata.log`.



Note

For the versions before Release 15, the e5_base user additionally needs to run a back up by means of the following command: `exp e5_base/baseora file=e5_base.dmp log=exp_e5_base.log`.

8.2 Importing Data into the Database

The database with the corresponding physical and logical structures (database segments, database files) must be created before importing the data into the database. This is generally the responsibility of a database administrator.

However the **ORACLE 11g Setup CD**, which we have supplied allows the creation of an empty database. Suitable users must be created before data is imported into the database.

If the database has been created with the ORACLE 11g Setup CD that we supplied, then the *Tablespace temp* is automatically assigned to every created user as default tablespace. For this reason, a default *tablespace* must not be assigned to the *CREATE USER statement* in this case.

If you create your database in a different way, it is absolutely necessary to assign a *Tablespace* to the temporary data. As of ORACLE 11g the *Default Permanent Tablespace* exists. This is used for the entire database if the user has not assigned a Default Tablespace.

After creating a user it is necessary to set the content of the environment variable `NLS_LANG` correctly.

In the V5 R14 release, data types of type *long raw* have been converted into the datatype blob. This makes extensions of the import script necessary. The blob tables must therefore be created before the import. Further information can be found in the chapter [Creating Blob Tables](#).

The actual import is then executed with the `imp` command.

```
SET NLS_LANG=AMERICAN_AMERICA.UTF8
```

- `imp system/syselcaro fromuser=e5_database touser=e5_database file=e5_database.dmp log=import_e5_database.log`

```
SET NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
```

- `Imp system/syselcaro fromuser=e5_ergotime touser=e5_ergotime`
- `File=e5_ergotime.dmp log=import_e5_ergotime.log`

8.3 Importing the Process Engineer Database

8.3.1 User Schema

Two database users are required for the PPR hub. The names and passwords can be selected at will by the DB administrator. The standard names and passwords defined by DELMIA are:

- `e5_database/databaseora`
- `e5_ergotime/ergomas`

A user with the necessary rights must log on in order to be able to create new users in a database:

- User name: `sys`
- Password: `syselcaro` or `change_on_install`



Note

*If neither of the two passwords is accepted and the ORACLE installation was not carried out new, ask the database administrator for the passwords for the **User SYS** and **SYSTEM**, or have the administrator carry out the following steps.*

The following commands generate the e5_database-User, which contains the files of the DPE configuration.

First the old roles and users are deleted. As of release R12, commands regarding the user in connection with the rights-database e5_rightsdata can be ignored. As of release R15, commands regarding the user e5_base can be ignored.

Roles:

```
drop role e5_base_user;
drop role e5_database_user;
drop role e5_rightsdata_user;
```

User:

```
drop user e5_base cascade;
drop user e5_ergotime cascade;
drop user e5_ergotime cascade;
drop user e5_ergotime cascade;
CREATE USER e5_database IDENTIFIED BY databaseora
DEFAULT TABLESPACE ERGO_USR TEMPORARY TABLESPACE TEMP;
```

Before the last user e5_ergotime is created, the NLS_LANG parameter should be switched back again.

```
SET NLS_LANG=AMERICAN_AMERICA.WE8ISO8859P1
```

```
CREATE USER e5_ergotime IDENTIFIED BY ergomas
DEFAULT TABLESPACE ERGO_USR TEMPORARY TABLESPACE TEMP;
SET NLS_LANG=AMERICAN_AMERICA.UTF8
```

8.3.2 Access Rights

In the database, the database users need the following access rights:

- Grant alter session
- Grant create cluster
- Grant create database link
- Grant create procedure
- Grant create sequence
- Grant create session
- Grant create synonym
- Grant create table
- Grant create trigger
- Grant create view

In addition, the commands must be used to assign two select rights for the newly generated users. Only the administrator can assign these rights:

- grant select on v_\$process

- grant select on v_\$session

For the CREATE USER command you need to hold the CREATE USER privilege, which comes automatically with the DBA role. The CREATE USER privilege is very powerful. Be sure to assign this privilege to as few users as possible.

User privileges can be assigned directly or via roles. Roles can include privileges and other roles.

Alternatively, management of users and roles can be effected from the Enterprise Manager.



Note

In order to avoid problems in connection with Stored Procedures and Grants, “execute immediate” should be avoided in Stored Procedures.

8.4 Tablespaces for Users

Creating users also includes the definition of the tablespaces.

The database users should use the *standard table range* ERGO_USR and the temporary table range TEMP (depending on the names assigned in the chapter on “Table range”).

Additionally, the following [Quotas](#) (contingents) should be set. A newly created user has no inherent share in the storage space of the tablespaces. His shares are referred to as quotas. (The following is an example for the user “e5_database”):

```
alter user e5_base quota unlimited on ERGO_INDX;
alter user e5_base quota unlimited on ERGO_USR;
alter user e5_base quota 0K on system;
```

It is important here to note that the tablespaces *ergo_indx* and *ergo_usr* must exist. If the Oracle instance for the DPE user has not been created using a setup provided by DELMIA, certain changes may need to be made in naming.

For Temporary Tablespaces and Undo Tablespaces users do not need a quota. These segments can be created without any additional privileges and are managed by ORACLE.

8.4.1 Creating Roles

The CONNECT role includes the CREATE SESSION privilege. You can log in after he has received the CONNECT role. Now roles need to be generated for the various users. This is done by executing the following SQL command:

```
connect e5_database/databaseora
CREATE ROLE E5_DATABASE_USER NOT IDENTIFIED
```

After creating a user, it is necessary to set the content of the environment variable NLS_LANG correctly.

8.4.2 Creating Blob Tables

In the V5 R14 release, data types of type *long raw* have been *converted* into the data type blob. This makes extensions of the import script necessary.

The following blob tables must be created for the user "e5_database" before the import is executed:

```
"XBLOBVALUEC497V11", "XDOSCRIPC339V0", "XDOPRINTFORMC474V8" ,  
"XDOSCRIPTVARIABLEC473V8", "XDOVBAPROJECTC466V8", "XMURULE"
```

The DDL commands for the user "e5_database" are noted in the [Appendix](#).

If the tables have been created, import of the databases can be executed on the DOS level. It is important to note that the parameter *ignore* must be set to *yes* (ignore=y).

9. Datapump

There is a new tool generation for export and import. The control programs are expdp and impdp. To ensure compatibility, the old export (exp) and import (imp) tools are still supplied as well.

Advantages

- Server based tools for fast loading and unloading of data and metadata.
- Callable via PL/SQL
- Web based (EM DB control, EM grid control)

Because of the split up of exports and imports into different processes, they can run in parallel.

The parallel parameter starts two Worker processes. In order to fully exploit the performance, each Worker process is assigned its own directory to which the Dump files are written. In an ideal case two different controllers are employed. This prevents the IO System from becoming the bottle neck.



Note

Further discussion concerning the aforementioned options would exceed the scope of this documentation. However, Oracle's general recommendations apply here; they can be found in relevant documents under the heading "Oracle database administration".

10. Process Engineer Database

If you have never worked with the *ENOVIA Manufacturing Hub*, you can start with the demonstration database.

This database contains simple sample projects which you can immediately use. The demonstration database is imported in nine easy steps:

Importing an ORACLE (demonstration) Database:

- 1) Start the tool DBAssistant.exe in the directory DELMIA\PPRServer\program\bin.
- 2) Select the point 'Exchange – Import' and click **OK**.

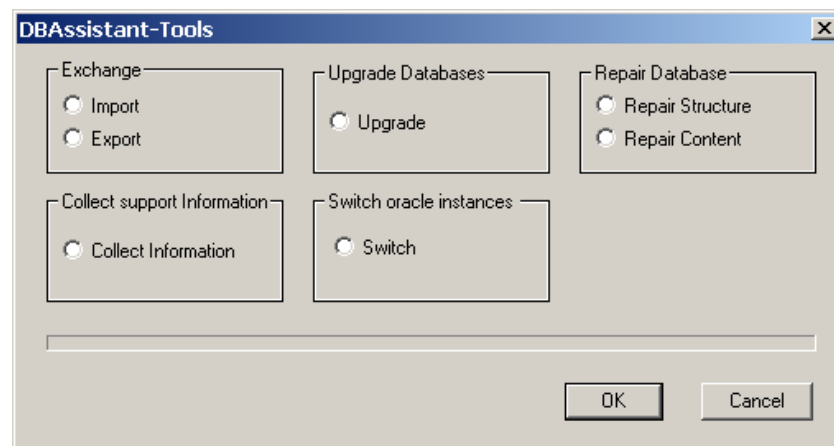


Figure 1: Import of the Process Engineer Database

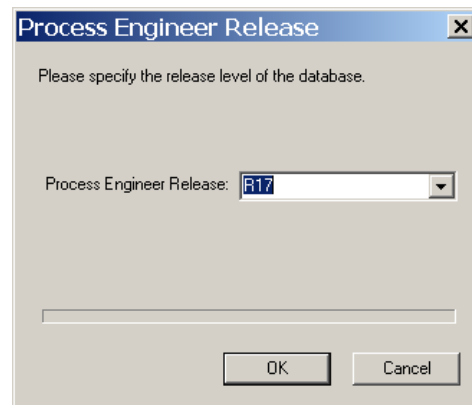


Figure 2: Process Engineer Release

- 1) Enter the connection parameters for the ORACLE database.
- 2) Also specify the users you want to create.
- 3) Select the "Oracle-Dumps" directory from the CD as the database to be imported. If you want to import the Process Engineer database, select the subdirectory "..\Oracle-Dumps\pe517".

Figure 3: Example of Import paRameters for the Process Engineer Database

- 4) If a name has been assigned to a Blob Dump during export and the standard tablespace name was not used for the import, the checkbox for use of standard Tablespace names must be deactivated.
- 5) Click **OK**.

Figure 4: TableSpaceAssistant

- 6) Another dialog appears where the corresponding tablespaces must be entered in the input fields.
- 7) Click **OK**.

11. Upgrading a Database

If you have previously used ORACLE 9i, please read both chapters [Installing the ORACLE Software](#) and [“Migrating an ORACLE 9i Database”](#).

Nothing in particular has to be taken into account with regard to the database when upgrading it; the setup of the PPR server makes all necessary changes to the schematics.

12. Using the DPE Database Assistant

Apart from the option to create users manually and to execute the database imports with command lines, a GUI-based tool can also be used to execute these tasks, as seen in the example of the demonstration database import.

The database assistant menu point can be called up in the start dialog of the DPE installation CD.

In a dialog it is then possible to select whether an import or export, a database upgrade, a collection of support-relevant information, or a switch-over of the database server is executed.

An example for the import of dumps can be found in the chapter [Process Engineer Database](#).

13. Maintenance of the Database

13.1 Performance Tuning

Optimization of the Database Instance

Some of the parameters from the chapter on instance settings are computer-dependent (resource-dependent), for example, that of the physical main storage. For this reason, the values suggested here can only be considered guidelines.

Similar to the buffer cache (*db_cache_size*), the common pool (*shared_pool_size*) can be optimized only through exact measurements regarding the environment. Both sizes can be adapted while running as of ORACLE 9i. Further relevant parameters for the load on the physical main storage are sort area (*sort_area_size* - ORACLE 8i) of the log buffer (*log_buffer*). If Automatic Memory Management (AMM) is used, please note the following:



Note

Automatic Memory Management (AMM) is a new feature in Oracle 11g. With Oracle 9i it was for the first time possible to change the SGA-Parameter dynamically, that is without a re-start of the database. However, the changes had to be made through the DBA. In ORACLE 11g the database controls the changing requirements for the individual areas of the SGA automatically and allocates the current requirements based on the main storage. "Automatic Shared Memory Management" is the same feature.

Dynamically configurable parameters can be changed by using ALTER SYSTEM SET <name>=<new value>.

The entire multiple of a so-called granule is always allocated. The new parameter SGA_MAXSIZE sets the maximum size of the SGA. This may not be exceeded by sum of the buffer or all other allocated areas in the SGA.

By setting the parameter *optimizer_mode* to the value *first_rows*, a combination of cost-based and heuristic analysis processes are used in order to optimize the running SQL commands in the database.

This parameter can be set either through an entry in the initially executed control table of the database (init/INSTANZNAME.ora) or through the SQL statement

```
ALTER SESSION SET OPTIMIZER_MODE=first_rows.
```

Further information on the function mode of the optimizer and the effects of this parameter can be found in the Oracle manual:

Oracle11g Database Performance Guide and Reference'

The scheme of the individual database users e5_database and e5_ergotime must be checked on a regular basis after this parameter has been set.

13.2 Maintenance Work

Simple Monitoring Activities

It is recommended at the very least to check the RAM and free disk space on the database server at regular intervals. Any shortages here could lead to severe performance impairments which in turn could lead to program crashes.

- The database is configured so that if more space is required, the size of the physical database files is increased.
- The physically required disk space on the file system for the most part depends on the volume of data processed, and the amount of RAM used depends largely on the number of simultaneously logged on users and the type of application that accesses the database. If during operation large changes come about, you must reconfigure the database instance.
- It is essential to create statistics at regular intervals for an optimum execution of the generated SQL statements. They can be created for tables, indices, and fields. *Please refer to the [Performance Tuning](#).*
- However, here are a few elementary things in which ORACLE 11g differs from previous versions. ORACLE 11g contains a lot of additional advisories for various purposes. They collect information, analyze statistics and on this basis make suggestions for improving performance or settling problems. The advisories use the Automatic Workload Repository usually refer to the current workload.

OPTIMIZER_MODE

Choose and rule are not supported anymore, because the rule based optimizer is no longer supported. Inofficially though, these options can still be used. Through ORACLE 11g analyzes exactly pin-pointing matters are possible. Statistics are automatically collected and saved in AWR (Automatic Workload Repository)

- Information about sessions is reported by the ASH (Active Session History). Historical information about the session history of an instance is provided.
- The ADDM (Automatic Database Monitor) evaluates everything and makes suggestions.

The focus is on the entire database. Special sessions or applications are not adjusted, however.

Requirements for the “Automatic SGA Tuning” are the following:

- The STATISTICS_LEVEL must be set to TYPICAL or ALL.

Then “Automatic SGA Tuning” is possible and every 60 seconds system statistics are collected.

To turn off this function selectively, turn off statistics interval or the ACTIVE SESSION HISTORY.

If the STATISTICS_LEVEL is set to Basic, then the:

- Table monitoring is turned off
- Automatic-SGA-Management is not working and
- SGA and PGA advisors do not work

**Note**

Further discussion concerning the aforementioned options would exceed the scope of this documentation. However, Oracle's general recommendations apply here; they can be found in relevant documents under the heading "Oracle database administration".

Appendix

Supplement to the Chapter 'Creating Blob Tables'

The DDL commands for creating the blob tables for the user "e5_database":

```
CREATE TABLE "XBLOBVALUEC497V11"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"M_DATA0V13" BLOB DEFAULT EMPTY_BLOB(),
PRIMARY KEY ( OID ) )
/
CREATE TABLE "XDOSCRIPC339V0"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"M_PCHANGEPROTOCOLOID0V0" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLCID1V0" NUMBER(38, 0) DEFAULT 0,
"M_NAME2V0" VARCHAR2(1000),
"M_NAMESHORT3V0" VARCHAR2(1000),
"M_NOTE4V0" VARCHAR2(4000),
"M_TYPENAME5V0" VARCHAR2(1000),
"M_TABLENAME6V0" VARCHAR2(1000),
"M_STREXTERNALID7V0" VARCHAR2(1000),
"M_CREATIONDATE8V0" DATE,
"M_MODIFICATIONDATE9V0" DATE,
"M_BWRITECHANGEPROTOCOL10V0" CHAR(1) DEFAULT chr(0),
"M_DBLSORTINDEX12V0" NUMBER DEFAULT 0,
"M_STRCREATOR13V0" VARCHAR2(1000),
"M_STRMODIFIER14V0" VARCHAR2(1000),
"M_PDODEFAULTTIMPLOID37V0" NUMBER(38, 0) DEFAULT 0,
"M_PDODEFAULTTIMPLCID38V0" NUMBER(38, 0) DEFAULT 0,
"M_TYPE39V0" VARCHAR2(1000),
"M_USEGRID41V0" CHAR(1) DEFAULT chr(0),
"M_STROBJECTUUUID8V3" VARCHAR2(1000),
"M_RUNASOWNER53V4" CHAR(1) DEFAULT chr(0),
"M_STRORIGINALUUUID9V8" VARCHAR2(1000),
"M_PUSEROWNEROID20V8" NUMBER(38, 0) DEFAULT 0,
"M_PUSEROWNERCID21V8" NUMBER(38, 0) DEFAULT 0,
"M_RUNOWNTRANSACTION74V8" CHAR(1) DEFAULT chr(0),
"M_STROEMOBJECTUUUID10V13" VARCHAR2(1000),
"M_SCRIPT24V13" BLOB DEFAULT EMPTY_BLOB(),
```

```

"STRPLMID" VARCHAR2(1000),
"BLOBNOTEID" NUMBER(38, 0) DEFAULT 0,
"BLOBNOTEID" NUMBER(38, 0) DEFAULT 0,
"BSIMPLESETEXIST" CHAR(1) DEFAULT chr(0),
PRIMARY KEY ( OID ) )
/

CREATE TABLE "XDOPRINTFORMC474V8"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"M_PCHANGEPROTOCOLOID0V0" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLCID1V0" NUMBER(38, 0) DEFAULT 0,
"M_NAME2V0" VARCHAR2(1000),
"M_NAMESHORT3V0" VARCHAR2(1000),
"M_NOTE4V0" VARCHAR2(4000),
"M_TYPENAME5V0" VARCHAR2(1000),
"M_TABLENAME6V0" VARCHAR2(1000),
"M_STREXTERNALID7V0" VARCHAR2(1000),
"M_STROBJECTUUID8V3" VARCHAR2(1000),
"M_STRORIGINALUUID9V8" VARCHAR2(1000),
"M_CREATIONDATE8V0" DATE,
"M_MODIFICATIONDATE9V0" DATE,
"M_BWRITECHANGEPROTOCOL10V0" CHAR(1) DEFAULT chr(0),
"M_DBLSORTINDEX12V0" NUMBER DEFAULT 0,
"M_STRCREATOR13V0" VARCHAR2(1000),
"M_STRMODIFIER14V0" VARCHAR2(1000),
"M_PUSEROWNERID20V8" NUMBER(38, 0) DEFAULT 0,
"M_PUSEROWNERCID21V8" NUMBER(38, 0) DEFAULT 0,
"M_PPLANTYPEOID22V8" NUMBER(38, 0) DEFAULT 0,
"M_PPLANTYPECID23V8" NUMBER(38, 0) DEFAULT 0,
"M_BDEFAULTFORM24V8" CHAR(1) DEFAULT chr(0),
"M_STROEMOBJECTUUID10V13" VARCHAR2(1000),
"M_FORMDATA23V13" BLOB DEFAULT EMPTY_BLOB(),
"STRPLMID" VARCHAR2(1000),
"BLOBNOTEID" NUMBER(38, 0) DEFAULT 0,
"BLOBNOTEID" NUMBER(38, 0) DEFAULT 0,
"BSIMPLESETEXIST" CHAR(1) DEFAULT chr(0),
PRIMARY KEY ( OID ) )
/

CREATE TABLE "XDSCRIPTVARIABLEC473V8"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,

```

```

"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"M_PCHANGEPROTOCOLOID0V0" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLCID1V0" NUMBER(38, 0) DEFAULT 0,
"M_NAME2V0" VARCHAR2(1000),
"M_NAMESHORT3V0" VARCHAR2(1000),
"M_NOTE4V0" VARCHAR2(4000),
"M_TYPENAME5V0" VARCHAR2(1000),
"M_TABLENAME6V0" VARCHAR2(1000),
"M_STREXTERNALID7V0" VARCHAR2(1000),
"M_STROBJECTUUID8V3" VARCHAR2(1000),
"M_STRORIGINALUUID9V8" VARCHAR2(1000),
"M_CREATIONDATE8V0" DATE,
"M_MODIFICATIONDATE9V0" DATE,
"M_BWRITECHANGEPROTOCOL10V0" CHAR(1) DEFAULT chr(0),
"M_DBLSORTINDEX12V0" NUMBER DEFAULT 0,
"M_STRCREATOR13V0" VARCHAR2(1000),
"M_STRMODIFIER14V0" VARCHAR2(1000),
"M_PUSEROWNEROID20V8" NUMBER(38, 0) DEFAULT 0,
"M_PUSEROWNERCID21V8" NUMBER(38, 0) DEFAULT 0,
"M_PPLANTYPEOID22V8" NUMBER(38, 0) DEFAULT 0,
"M_PPLANTYPECID23V8" NUMBER(38, 0) DEFAULT 0,
"M_LTYPE24V8" NUMBER(38, 0) DEFAULT 0,
"M_LDATATYPE26V8" NUMBER(38, 0) DEFAULT 0,
"M_STROEMOBJECTUUID10V13" VARCHAR2(1000),
"M_SCRIPT23V13" BLOB DEFAULT EMPTY_BLOB(),
"STRPLMID" VARCHAR2(1000),
"BLOBNOTEOID" NUMBER(38, 0) DEFAULT 0,
"BLOBNOTECID" NUMBER(38, 0) DEFAULT 0,
"BSIMPLESETEXIST" CHAR(1) DEFAULT chr(0),
"STRDSN" VARCHAR2(1000),
"STRUID" VARCHAR2(1000),
"BNOTVISIBLE" CHAR(1) DEFAULT chr(0),
"PScriptVARIABLEOID" NUMBER(38) DEFAULT 0,
"PScriptVARIABLECID" NUMBER(38) DEFAULT 0,
"PCONDITIONS" BLOB DEFAULT EMPTY_BLOB(),
PRIMARY KEY ( OID ) )
/

CREATE TABLE "XDOVBAPROJECTC466V8"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,

```

```

"M_PCHANGEPROTOCOLOID0V0" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLCID1V0" NUMBER(38, 0) DEFAULT 0,
"M_NAME2V0" VARCHAR2(1000),
"M_NAMESHORT3V0" VARCHAR2(1000),
"M_NOTE4V0" VARCHAR2(4000),
"M_TYPENAME5V0" VARCHAR2(1000),
"M_TABLENAME6V0" VARCHAR2(1000),
"M_STREXTERNALID7V0" VARCHAR2(1000),
"M_STROBJECTUUID8V3" VARCHAR2(1000),
"M_STRORIGINALUUID9V8" VARCHAR2(1000),
"M_CREATIONDATE8V0" DATE,
"M_MODIFICATIONDATE9V0" DATE,
"M_BWRITECHANGEPROTOCOL10V0" CHAR(1) DEFAULT chr(0),
"M_DBLSORTINDEX12V0" NUMBER DEFAULT 0,
"M_STRCREATOR13V0" VARCHAR2(1000),
"M_STRMODIFIER14V0" VARCHAR2(1000),
"M_PUSEROWNEROID20V8" NUMBER(38, 0) DEFAULT 0,
"M_PUSEROWNERCID21V8" NUMBER(38, 0) DEFAULT 0,
"M_PDODEFAULTTIMPLOID37V0" NUMBER(38, 0) DEFAULT 0,
"M_PDODEFAULTTIMPLCID38V0" NUMBER(38, 0) DEFAULT 0,
"M_STROEMOBJECTUUID10V13" VARCHAR2(1000),
"M_DATA23V13" BLOB DEFAULT EMPTY_BLOB(),
"STRPLMID" VARCHAR2(1000),
"BLOBNOTEOID" NUMBER(38, 0) DEFAULT 0,
"BLOBNOTECID" NUMBER(38, 0) DEFAULT 0,
"BSIMPLESETEXIST" CHAR(1) DEFAULT chr(0),
PRIMARY KEY ( OID ) )
/
CREATE TABLE "XMURULE"
("OID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"CID" NUMBER(38, 0) DEFAULT 0 NOT NULL ENABLE,
"BLOBFIRSTCONDITION" BLOB DEFAULT EMPTY_BLOB(),
"IFIRSTRESULT" NUMBER(38, 0) DEFAULT 0,
"BLOBSECONDCONDITION" BLOB DEFAULT EMPTY_BLOB(),
"ISECONDRESULT" NUMBER(38, 0) DEFAULT 0,
"ITHIRDRESULT" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLOID0V0" NUMBER(38, 0) DEFAULT 0,
"M_PCHANGEPROTOCOLCID1V0" NUMBER(38, 0) DEFAULT 0,
"M_NAME2V0" VARCHAR2(1000),
"M_NAMESHORT3V0" VARCHAR2(1000),
"M_NOTE4V0" VARCHAR2(4000),

```

```

"BLOBNOTEOID" NUMBER(38, 0) DEFAULT 0,
"BLOBNOTECID" NUMBER(38, 0) DEFAULT 0,
"M_TYPENAME5V0" VARCHAR2(1000),
"M_TABLENAME6V0" VARCHAR2(1000),
"M_STREXTERNALID7V0" VARCHAR2(1000),
"M_STROBJECTUUID8V3" VARCHAR2(1000),
"M_STRORIGINALUUID9V8" VARCHAR2(1000),
"M_STROEMOBJECTUUID10V13" VARCHAR2(1000),
"M_CREATIONDATE8V0" DATE,
"M_MODIFICATIONDATE9V0" DATE,
"M_BWRITECHANGEPROTOCOL10V0" CHAR(1) DEFAULT chr(0),
"M_DBLSORTINDEX12V0" NUMBER DEFAULT 0,
"M_STRCREATOR13V0" VARCHAR2(1000),
"M_STRMODIFIER14V0" VARCHAR2(1000),
"M_PUSEROWNEROID20V8" NUMBER(38, 0) DEFAULT 0,
"M_PUSEROWNERCID21V8" NUMBER(38, 0) DEFAULT 0,
"STRPLMID" VARCHAR2(1000),
"BSIMPLESETEXIST" CHAR(1) DEFAULT chr(0),
PRIMARY KEY ( OID ) )
/
.

```


List of Figures

Figure 1: Import of the Process Engineer Database.....	25
Figure 2: Process Engineer Release.....	25
Figure 3: Example of Import paRameters for the Process Engineer Database.....	26
Figure 4: TableSpaceAssistant	26

Index

N

Nonliabilityii