


Utilizing the PMDB

This section includes the following topics:

- [Prerequisites for configuring the PMDB in an SQL Server](#)
- [Prerequisites for configuring the PMDB in an Oracle Server](#)
- [Additional DBA settings](#)
- [How to disable the recycle bin in Oracle version 10g](#)
- [Enabling Automatic Undo Management \(AUM\)](#)
- [About backing up the database and creating archiving](#)
- [About maintenance operations](#)
- [About changing the size of tables and index extents](#)

Prerequisites for configuring the PMDB in an SQL Server

Install the Precise environment before configuring the SQL Server-based PMDB. Before applying the information in this section, refer to the *Precise Installation Guide*.

 SQL Server 2000 is not supported in v9.0, so an upgrade is not available for such installations.

For an SQL Server-based PMDB, use an MS-SQL dedicated instance.

To avoid using the MS-SQL fully automatic configuration function, change the following parameters:

- **Maximum server memory (MB).** Should be 80% of the server's RAM.
- **Minimum memory per query (KB).** Should be 512 instead of 1024.

An example to configure a parameter is the following command from the Query Analyzer:

```
EXEC sp_configure 'max server memory (MB)', [new value] RECONFIGURE WITH OVERRIDE
GO
```

Maintenance operations

Check the table and index status by running the following command:

```
dbcc showcontig ({table name}) with all_indexes
```

The results should be that the "Extent Scan Fragmentation" is less than 50 and the "Logical Scan Fragmentation" is low; otherwise, it will damage the index scans.

If one of the above conditions is confirmed, perform a reorganization on the indexes with the following command:

```
dbcc dbreindex ('{databasename.tableName}')
```

 Be aware that the index and the table will not be available during the rebuild.

Backing up the database and creating archiving

Use the Database Maintenance Plan function to create a backup plan for the database. Verify that the SQL Server agent is automatically started at database startup.

If you choose the simple recovery mode for the database (the transaction log can be deleted after a commit or checkpoint), the recovery will not be able to work to a point in time, but only to the backup time.

If you choose a full recovery mode for the database, verify that you also backup the transaction log and shrink the database periodically, to stop the transaction log from growing infinitely.

If you do not create a backup, information will be lost when you have a database crash and you will not be able to recover data.

Additional DBA settings for an SQL Server-based PMDB

To improve performance, the DBA on site must check that the settings in the following table contain the correct values.

Table 1 Additional DBA settings for the SQL Server-based PMDB

Setting	Description	Value
---------	-------------	-------

Size of the datafile	The default size when the PMDB is created.	2 GB
Size of the log file	The default size when the PMDB is created.	500 MB
Auto Extent size for datafiles and logs	The size of the new segment to be allocated.	30% of initial datafile (for example, if the datafile is initially 1 GB, the auto extent size should be 300 MB).
Tempdb location	Put the tempdb database on a fast I/O system.	

Prerequisites for configuring the PMDB in an Oracle Server

Install the Precise environment before configuring the Oracle-based PMDB. Before applying the information in this section, refer to the *Precise Installation Guide*. For an Oracle-based PMDB, use an Oracle dedicated instance.


Verify that you are creating the database with the proper NLS_CHARACTERSET. When the monitored Oracle instances use a multiple byte character set, use UTF8. AL32UTF8 is not supported by the PMDB.

The configuration that is recommended when the PMDB is based on an Oracle database can be found in the following paragraphs. For updated information, also see: my.precise.com/

If you install the PMDB on an Oracle legacy database, verify that the database block size is at least 8 KB. If not, try to install the PMDB on a different Oracle instance.

The following table specifies the database block size prerequisite.

Table 2 Oracle Server Database block size prerequisite

Parameter	Value	Improvement
db_block_size	Minimum 8 KB (16 KB recommended)	<p>This value defines the Oracle block size. The block size is set when creating the database and cannot be changed afterwards.</p> <p>A value of at least 8 KB ensures that the PMDB can perform smoothly. A lower value may cause serious performance problems.</p> <p>If you are installing on a legacy database, verify that the database block size is 8 KB or higher. If not, try to install the PMDB on a different Oracle instance.</p> <div>  The default block size on Windows NT is 2 KB, which is too small. To review this value, check the db_block_size parameter in the init.ora file. </div>

Required INIT.ORA changes

To improve performance, change some parameter values and verify that the values match those listed in the appropriate table.

The parameter values can be changed on a new and on an existing Oracle instance. The following tables describe the general and Oracle version specific parameters.

Table 3 General parameters for the Oracle-based PMDB

Parameter	Value	Improvement / Remarks
log_buffer	If the version <9.2.0.7, then the log_buffer = 3 MB, else the log_buffer= 16 MB.	
shared_pool_size	1/4 of the physical memory.	The shared pool should not be too small (< 250 MB) or too large (>2 GB).
log_checkpoint_timeout	900	
open_cursors	300	Verify that you have enough open cursors for the application to execute.
Processes	300	Verify that you have enough processes for the application to execute.

db_writer_processes	Should be set according to the number of I/O devices you have for the database.	Improves I/O performance in cases of large I/O. Each writer is writing to different disks (improves loads, for example).
---------------------	---	--

Table 4 Oracle 8i parameters

Parameter	Value	Improvement / Remarks
sort_area_size	(1/8th of the physical memory) / (concurrent users + 10), up to 6 MB.	For example: If you have 2 GB of Physical Memory and 90 users, then the calculation will be: (2 GB / 8) / 100 ≈ 2.5 MB
db_block_buffers	1/3rd of the physical memory divided by the size of the db_block_size.	The given values help to keep the I/O wait of the PMDB queries low.

Table 5 Oracle 9i parameters

Parameter	Value	Improvements / Remarks
db_cache_size	Replaces parameter db_block_buffers. Sets the final size of the buffer pool to 1/3rd of the physical memory.	
pga_aggregate_target	1/3rd of the physical memory.	
workarea_size_policy	Auto	This parameter should be set to auto when we use the pga_aggregate_target parameter.
session_cached_cursors	300	
DB-cache_advice	Verify that this parameter is set to off because the default value is on.	This parameter checks the work on the database and helps determine the recommended size of the Buffer pool. But this damages performance, so verify that it is on only for a short period.

Table 6 Oracle 10g parameter

Parameter	Value	Improvement / Remarks
sga_target	Automatic memory management. 1/4th of the physical memory.	Replaces the following parameters: <ul style="list-style-type: none"> db_cache_size shared_pool_size large_pool_size java_pool_size Verify that they are initialized with either 0 or the minimum value you need.

INIT.ORA example




The following is an example of the init.ora file for a Oracle 9.2 server with 2 GB RAM and 2 disks.

```
db_block_size=16384 # 16K
db_cache_size=268435456 # 256MB
log_buffer=5242880 # 5 MB
shared_pool_size=471859200 # 450MB
log_checkpoint_timeout=900 # 15 minutes
pga_aggregate_target=262144000 # 256MB
session_cached_cursors=300
open_cursors=300
processes=300
undo_management=AUTO
undo_tablespace=UNDOTBS1
undo_retention=900 # 15 minutes
```

Additional DBA settings

To improve performance, the DBA on site must modify the Rollback segment settings and the Redo log file settings. The following table describes those settings.

Table 7 Rollback segment and redo log file settings

Setting	Description	Value
Rollback segments (For Oracle 8i)	Minimum tablespace size	2 GB (4 GB in large installations)
Rollback segments (For Oracle 8i)	Minimum number of segments	20
Rollback segments (For Oracle 8i)	Minimum initial segment size	5 MB <div>  Each rollback segment should be able to grow to the size of the tablespace. </div>
Enable AUM (For Oracle 9i and later)	Automatic Undo Management	
Temporary tablespace	Minimum temporary tablespace size	2 GB <div>  The temporary tablespace is most likely to grow (according to the PMDB load). </div>
Redo log file	Minimum redo log file size	1 GB (4 GB in large installations) <div>  If the redo file is switching every 20 minutes, enlarge the file. </div>

Additional Rollback segment and Redo log file information

Each Rollback segment should contain a minimum of 5 extents. The size of each extent within a Rollback segment should be at least 1MB each. Their optimal size should be set to the summed size of those extents. For example, if creating a rollback segment with 5 extents of 1MB each, you should set the optimal size to 5MB. This is to prevent the changing size of the rollback segment to exhaust the tablespace's free space.

If possible, keep data tablespace indexes and temporary tablespace on different disk drives. Verify that the recycle bin in Oracle version 10 is disabled. See How to disable the recycle bin in Oracle version 10g . For more information refer to the Using Flashback Drop and Managing the Recycle Bin section in the *Oracle Administration Guide*.

Four Redo logs should be created, each with 1 mirror (2 members for each group). Remember that hardware mirroring is better than software mirroring.

In Oracle version 9i and higher, it is recommended to configure the Undo space management; otherwise, you must configure the rollback segments according to the rules in [Table 6-7](#).

If you are using an archive (`log_archive_start` set to true) and you enable parallel archiving writers, then the `log_archive_max_processes` should be set to more than 1 (same consideration as in the `db_writer_process`).

How to disable the recycle bin in Oracle version 10g

Our recommendation is to disable the recycle bin and this section describes the background and the action on how to disable the recycle bin.

The recycle bin is actually a data dictionary table containing information about dropped objects. Dropped tables and any associated objects such as indexes, constraints, nested tables, and the likes are not removed and still occupy space. They continue to count against user space quotas, until specifically purged from the recycle bin or the unlikely situation where they must be purged by the database because of tablespace space constraints.

The following information is taken from the *Oracle Database Administration Guide*. In version 10.0 and 10.1 use the following command:

```
alter system set "_recyclebin"=FALSE scope=BOTH;
```

In version 10.2, use the following command:

```
alter system set recyclebin=off scope=BOTH
```

Enabling Automatic Undo Management (AUM)

To enable Automatic Undo Management (AUM), you must create a new tablespace and set the `init.ora` parameters. For additional information on how to enable AUM, see the Oracle Database Administrator's Guide, "Managing Undo Space."

To enable Automatic Undo Management (AUM)

1. Create the undo tablespace by running the following command:

```
CREATE UNDO TABLESPACE tablespace_name DATAFILE 'datafile name' SIZE 32M REUSE AUTOEXTEND ON EXTENT MANAGEMENT LOCAL
```
2. Set the following `init.ora` parameters:
 - `undo_management = AUTO`
 - `undo_tablespace = tablespace_name`
 - `undo_retention = 900`

If the undo tablespace size is too large, you may decrease the number of seconds.

About backing up the database and creating archiving

The PMDB holds information for up to 3 years by default. If you want to backup this information, you should enable the Redo log archiving, and create a backup using `RMAN` or the Precise backup utility. For more information, refer to the official Oracle Backup and Recovery documentation.

If you do not create a backup, information will be lost when you have a database crash and you will not be able to recover data. If you create a backup without using the archive mode, you will be able to recover data to the time of the backup. All new operations (after the logs were archived) will be lost.

About maintenance operations

Every week the maintenance process analysis the PMDB, but to improve the database performance, the DBA must check and perform a rebuild to the indexes using the `alter index {name} rebuild` command.

About changing the size of tables and index extents

The PMDB uses the `products\dbms\files\tables_definition\ps_00_db_entities.xml` file to create the tablespaces for the tables and the indexes. To change the default sizes you can change the file before the PMDB installation or alter the tablespaces after the installation.

The following table shows the default extents sizes.

Table 8 Default extents sizes

Tablespace	Size
Large table	2 MB
Medium table	512 KB
Small table	80 KB
Large index	1 MB
Medium index	256 KB
Small index	80 KB