



**Hewlett Packard**  
Enterprise

# Operations Orchestration

Software Version: DB Maintenance Package 4.0

Windows and Linux Operating Systems

## **Database Maintenance**

Document Release Date: November 2016

Software Release Date: November 2016

## Legal Notices

### Warranty

The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

### Restricted Rights Legend

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

(missing or bad snippet)

### Copyright Notice

© 2005-2016 Hewlett Packard Enterprise Development LP

### Trademark Notices

(missing or bad snippet)(missing or bad snippet)

## Documentation Updates

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <https://softwaresupport.hpe.com/>.

This site requires that you register for an HP Passport and to sign in. To register for an HP Passport ID, click **Register** on the HPE Software Support site or click **Create an Account** on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HPE sales representative for details.

(missing or bad snippet)

## Contents

HPE OO Database Maintenance Procedures .....	4
Microsoft SQL Server Database Maintenance .....	5
Procedure for Maintaining Indexes and Statistics .....	5
Procedure for Purging Historical Data .....	7
Oracle Maintenance Procedures .....	9
Procedure for Maintaining Indexes and Statistics .....	9
Procedure for Purging Historical Data .....	10
MySQL Maintenance Procedures .....	12
PostgreSQL Maintenance Procedures .....	13

# HPE OO Database Maintenance Procedures

HPE OO database maintenance tasks include:

- Updating statistical information about indexes and tables
- Rebuilding indexes
- Purging historical data
- Reclaiming free space

Use the procedures and other tools described in the following chapters in order to keep your HPE OO database in good shape.

To update statistics and maintain indexes, you can also use other tools in accordance with your company policy. For data purging, it is highly recommended to use only HPE provided flow and procedures in order not to compromise database structure or data.

## Notes:

- Online index rebuilding may require an Enterprise grade database license. Make sure you have the correct license before attempting online index rebuild.
- Database maintenance activity usually consumes additional resources from the database. Schedule maintenance activity at the times when HPE OO is least active.

## Database Size

Note that HPE OO 10.x keeps all flow and step execution data in the database by default, enabling comprehensive debugging of previous flow runs. As a result, the database size grows in accordance to the system throughput and flow complexity. It is highly recommended to track your database size and make sure old, irrelevant information is periodically purged or removed.

As of version 10.22 and 10.50, the amount of data kept in the database can also be controlled by setting the system persistency level. See the *HPE OO Tuning Guide* for more details.

## Database Guide Document

It is highly recommended to read the *HPE OO Database Guide* for detailed instructions on how to configure your HPE OO database.

# Microsoft SQL Server Database Maintenance

This chapter provides information on how to install and use HPE OO Microsoft SQL Server database maintenance procedures. These procedures are applicable for Microsoft SQL Server 2008R2, 2012 and 2014 (SQL Server 2014 is supported only by HPE OO 10.60).

## Procedure for Maintaining Indexes and Statistics

Download the latest **HPE\_OO\_DB\_Maintenance.zip** from HPE Live Network and extract it.

To install and use the HPE OO maintenance stored procedures:

1. Log in to Microsoft SQL Server as “sa” or any member of the **sysadmin** role and run the following code in order to give the HPE OO user the ability to access **dm\_os\_performance\_counters** dynamic management view (DMV):

```
USE [master]
GO

GRANT VIEW SERVER STATE TO myOODBLogin
GO
```

Replace “myOODBLogin” with the actual login role created for HPE OO.

Alternatively, edit and use the **mssql\_grant\_view\_server\_state.sql** script that resides in the MS-SQL folder.

2. Edit the following T-SQL scripts and replace each “USE <your\_db\_name\_here>” in the file headers with your actual HPE OO database name. For example, if your database name is “OOPROD”, replace it with “USE OOPROD”.
  - **OO\_DB\_MAINTENANCE\_LOG.sql (optional)**
  - **OOIndexMaintenance.sql**

Do not skip this step; otherwise, the set of procedures will not be created in the correct database.

3. Log in to Microsoft SQL Server as the HPE OO user.
4. Run the following T-SQL scripts in the given order and verify that the new objects were created successfully:
  - **OO\_DB\_MAINTENANCE\_LOG.sql** (optional)
  - **OOIndexMaintenance.sql**
5. Tune your stored procedures according to the comments embedded in the script.

Edit and use the **mssql\_execute\_index\_maintenance.sql** script that resides in the MS-SQL folder In order to execute the index maintenance procedure.

Explanation for each of the stored procedure parameters:

- Replace “<your\_db\_name\_here>” with the actual name of your database. Note that there are two occurrences.
- The **@FragmentationXXX** parameters set the script’s fragmentation level sensitivity and course of action in each case. These threshold levels and subsequent actions are recommended by Microsoft’s documentation. Tune these values with caution.
- **@SortInTempdb** (once set to ‘Y’) lets you perform sorting operations during index reorganization/rebuild in **tempdb** rather than in memory, for better performance. If you choose to use this option, make sure you have sufficient free space in **tempdb**.
- **@Indexes** is a filter for including/excluding indexes in the maintenance operation. It is recommended to keep this filter as is, to analyze all indexes.
- **@TimeLimit** is the timeout in seconds for the maintenance operation to complete. Set it in accordance with your maintenance window boundaries, if applicable.
- **@LockTimeout** is the timeout in seconds to wait for object lock. Once expired, the specific operation fails and the procedure continues to the next object.
- **@LogToTable** determines whether maintenance operation results should be logged to a table. This lets you keep track of the maintenance operations and helps procedure debugging.

**Note:** This requires creating the table using the **OO\_DB\_MAINTENANCE\_LOG.sql** script.

- **@Execute** determines whether actual operations (such as index rebuild) are performed or not. If this parameter is set to ‘N’, the procedure performs a “dry run” and shows an analysis of the relevant objects.

## Procedure for Purging Historical Data

Download the latest **HPE\_OO\_DB\_Maintenance.zip** pack from HPE Live Network and extract it

To install and use the HPE OO history purging stored procedure:

1. Edit the **OOPurgeHistory.sql** script and replace “USE <your\_db\_name\_here>” at the beginning of the script with your actual HPE OO database name. For example, if your database name is “OOPROD”, replace it with “USE OOPROD”.

Do not skip this step; otherwise, the procedure will not be created in the correct database.

2. Log in to Microsoft SQL Server as the HPE OO user.
3. Run the `OOPurgeHistory.sql` script and verify that the new objects were created successfully.
4. Tune your stored procedures according to the comments embedded in the script.

You may edit and use the **mssql\_execute\_purge\_stored\_procedure.sql** script that resides in the MS-SQL folder in order to execute the purge stored procedure.

Explanation for each of the stored procedure parameters:

- Replace “<your\_db\_name\_here>” with the actual name of your database.
- The **PurgeExecutionsOlderThan** parameter determines how many days are kept, (protected) relative to the time the procedure starts running. By default, 90 days are kept. Older data is deleted, starting with the oldest records.
- **@PurgeExecutionsInBatchesOf** determines how many flows are handled together. Smaller values imply smaller, more frequent transactions, and higher values imply less frequent, larger transactions. 1,000 is recommended for most systems.
- **@ShouldPurgeExecutionSummary** determines if the **OO\_EXECUTION\_SUMMARY** table should be purged. The default value is “0” (do not purge this table). It is recommended to keep data in this table as it does not consume a lot of space. Use “1” only if you want to completely remove any reference to the relevant flows.
- **@verbose** determines verbosity level. “0” corresponds to “quiet” output, “1” corresponds to normal output, and “2” prints out detailed information.
- **@StopPurgingProcessAfter** is the timeout in hours for the operation to complete. Set it in accordance with your maintenance window boundaries if applicable.

- **@DeepClean** determines whether deep cleansing is performed. For example, searching for “orphan” records that may bloat the database unnecessarily. The default is “0” (off). Note that setting this flag to “1” prolongs the procedure run time, but the timeout limit is still imposed.
- **@DisableIndexes** determines whether certain database indexes are disabled for the duration of the purge operation. These indexes are rebuilt at the end of the purge operation. Using this feature speeds up the purge on the expense of flow drill-down (step-level debugging).

# Oracle Maintenance Procedures

This chapter provides information on how to install and use HP OO 10.x Oracle database maintenance procedures. These procedures are applicable for Oracle 11g R2 and 12c R1.

## Procedure for Maintaining Indexes and Statistics

Download the latest **HPE\_OO\_DB\_Maintenance.zip** pack from HPE Live Network and extract it.

Open the Oracle folder and select the folder that correlates with your HPE OO release.

To install and use the HPE OO maintenance stored procedures:

1. Log in to Oracle as “system” or any other user with a DBA role, and run the following commands. These system privileges are required in order to verify the stored procedure created in the following steps has the explicit (not role-based) privileges to execute the index analysis and rebuild:

```
GRANT CREATE TABLE TO OO;  
GRANT ANALYZE ANY TO OO;  
GRANT ALTER ANY INDEX TO OO;
```

Adapt the highlighted user name to match your environment.

Alternatively, edit and use the **oracle\_privs\_for\_index\_maintenance.sql** script that resides in Oracle folder.

2. Log in to Oracle as “OO” (the user created for HPE OO).
3. Run the **HPE\_OO\_DB\_MAINT.sql** script and verify that the new package and procedures were created successfully.
4. Tune your stored procedures according to the comments embedded in the script.

Explanation for each of the stored procedure parameters:

- **pMaxHeight (IN)** - The minimal index height threshold for index rebuilding. The Oracle documentation recommends 3. Smaller values may result in unnecessary rebuilding operations.

- **pMaxLeafsDeleted (IN)** - The minimal deleted leaves threshold for index rebuilding. The Oracle documentation recommends 15. Smaller values may result in unnecessary rebuilding operations.
- **pRebuild (IN)** - Should indexes be rebuilt (1) or only perform a dry-run (0). A dry-run will show only recommendations for index rebuilding.
- **pReturnValue (OUT)** - The number of rebuilt indexes

**Note:** ONLINE index rebuilding should only be performed when the enterprise edition is used. Otherwise, the index rebuilding operation may lock tables and indexes and may interfere with the operation of HPE OO.

## Procedure for Purging Historical Data

Download the latest **HPE\_OO\_DB\_Maintenance.zip** pack from HPE Live Network and extract it.

To install and use HPE OO history purging stored procedure:

1. Log in to Oracle as “OO” (the user created for HPE OO).
2. Run the **HPE\_OO\_DB\_MAINT.sql** script (only if you have not run it already) and verify that the new package and procedures were created successfully.
3. Tune your stored procedures according to the comments embedded in the script.

The following example shows how this procedure may be used. See the guidelines provided as comments in the procedure header for detailed explanations.

```
SET serveroutput ON SIZE 100000

DECLARE x integer:= 0;

BEGIN
  hpe_oo_db_maint.PurgeHistory(90,10000,0,1,4,0,0,x);
  DBMS_OUTPUT.put_line('A total of ' || TO_CHAR(x) || ' flows were handled.');
```

Explanation about the above code (parameters reference values - left to right):

- The **pPurgeExecutionsOlderThan** parameter determines how many days are kept (protected) relative to the time the procedure starts running. Older data is deleted, starting with the oldest records. This parameter has no default value and must be specified.

- **pPurgeExecutionsInBatchesOf** determines how many flows are handled together. Smaller values imply smaller, more frequent transactions, and higher values imply less frequent, larger transactions. 10,000 is recommended for most systems.
- **pShouldPurgeExecutionSummary** determines if the **OO\_EXECUTION\_SUMMARY** table should be purged. The default value is "0" (do not purge this table). It is recommended to keep data in this table as it does not consume a lot of space. Use "1" only if you want to completely remove any reference to the relevant flows.
- **pVerbose** determines verbosity level. "0" corresponds to "quiet" output, "1" corresponds to normal output, and "2" prints out detailed information.
- **pStopPurgingProcessAfter** is the timeout in hours for the operation to complete. Set it in accordance with your maintenance window boundaries if applicable.
- **pDeepClean** determines whether deep cleansing is performed. For example, searching for "orphan" records that may bloat the database unnecessarily. The default is "0" (off). Note that setting this flag to "1" prolongs the procedure run time, but the timeout limit is still imposed.
- **pDisableIndexes** determines whether certain database indexes are disabled for the duration of the purge operation.

These indexes are rebuilt at the end of the purge operation. Using this feature speeds up the purge at the expense of the flow drill-down (step-level debugging).

## MySQL Maintenance Procedures

This package contains procedures and utilities for Oracle and Microsoft SQL Server maintenance. It does not include MySQL maintenance procedures and utilities. These will be provided in future packages.

## PostgreSQL Maintenance Procedures

This package contains procedures and utilities for Oracle and Microsoft SQL Server maintenance. It does not include PostgreSQL maintenance procedures and utilities. These will be provided in future packages.

