

Aggregate Storage Outline Compaction Using ESSCMDQ

This document describes how to save a compacted outline for an aggregate storage database.

This document contains the following topics:

Installing the ESSCMDQ Utility	1
Starting the ESSCMDQ Shell	1
Sample Scripts	1
Command Reference	2
OpenOtl	2
WriteOtl	3
RestructOtl	3
CloseOtl	4

Installing the ESSCMDQ Utility

To install ESSCMDQ, download `ESSCMDQ.EXE` and copy it to the following directory.

For Release 11.1.1.x:

<HYPERION_HOME>\products\Essbase\EssbaseClient\bin

For Release 9.3.x or earlier:

<ARBORPATH>\bin

Starting the ESSCMDQ Shell

Before you start ESSCMDQ, make sure the Essbase Server is running.

To start ESSCMDQ, enter `ESSCMDQ` at the operating system command prompt, and log in as you would with ESSCMD.

Note: You may need to change the directory to the directory in which you installed the ESSCMDQ utility before you can execute the shell.

Sample Scripts

The following sample scripts perform outline compaction on an aggregate storage outline. As with ESSCMD, you can run the scripts in batch mode or enter the commands interactively.

To write a compacted outline to the file system:

```

Login "host" "user" "password" ;
Select "app" "db" ;
Openotl 3 1 "Null" "Null" "PathNameOpen" "y" "y" 0 ;
Writeotl 0 "3" 1 "Null" "Null" "PathNameWrite" ;
CloseOtl 0 ;
Logout ;

```

where *PathNameOpen* is the full path and filename of the outline to be opened, and *PathNameWrite* is the full path and filename to where the compacted outline should be written.

To write a compacted outline to an application:

```

Login "host" "user" "password" ;
Select "app" "db" ;
Openotl "2" 1 "app" "db" "db" "y" "y" 1 ;
Writeotl 1 "2" 1 "app" "db" "db" ;
Restructotl 1 ;
CloseOtl 1 ;
Unlockobj 1 "app" "db" "db" ;
Logout ;

```

Command Reference

The following ESSCMDQ commands referenced in the above scripts can be used for outline compaction. Any commands in the script that are equivalent to ESSCMD commands are not listed here. Refer to the *Oracle Essbase Technical Reference* for details about these commands.

OpenOtl

OpenOtl opens an outline and reads it into memory.

Syntax

```

OPENOTL location ObjType AppName DbName PathNameOpen fLockFile
fKeepTrans OtlhandleRet

```

Parameter	Description
<i>location</i>	Location of the outline file. 1 - Local/client. 2 - Remote/server.
<i>ObjType</i>	Type of object. Use 1 (outline object).
<i>AppName</i>	Name of the application.
<i>DbName</i>	Name of the database.
<i>PathNameOpen</i>	Name of the outline file. The filename must be the full path name, and must be in quotation marks.
<i>fLockFile</i>	Whether to lock the outline. Values: Y/N
<i>fKeepTrans</i>	Whether to keep transactions. Values: Y/N
<i>OtlhandleRet</i>	An integer value from 0 to 4 representing an outline handle. Recommended: 0

WriteOtl

WriteOtl saves outline information to disk.

Syntax

```
WRITEOTL Otlhandle location AppName DbName PathNameWrite
```

Parameter	Description
<i>Otlhandle</i>	An integer value from 0 to 4 representing an outline handle. Recommended: 0
<i>location</i>	Location of the outline file. 1 - Local/client. 2 - Remote/server.
<i>AppName</i>	Name of the application.
<i>DbName</i>	Name of the database.
<i>PathNameWrite</i>	Name of the outline file. The filename must be the full path name, and must be in quotation marks.

RestructOtl

RestructOtl restructures an outline on the server.

Syntax

```
RESTRUCTOTL RestructureType
```

Parameters	Description
<i>RestructureType</i>	Type of restructuring to do. This can be one of the following values: ESS_DOR_ALLDATA = 1 ESS_DOR_INDATA = 4 ESS_DOR_LOWDATA = 3 ESS_DOR_NODATA = 2

Example

```
RESTRUCTOTL 1
```

CloseOtl

CloseOtl frees all information associated with the outline.

Syntax

CLOSEOTL *Otlhandle*

Parameter	Description
<i>Otlhandle</i>	An integer value from 0 to 4 representing an outline handle. Recommended: 0

Example

CLOSEOTL 0