



ORACLE® ESSBASE

RELEASE 11.1.1

NEW FEATURES



This document describes the new features in Oracle Essbase Release 11.1.1. For more information about these features, see the *Oracle Essbase Database Administrator's Guide* and *Oracle Essbase Technical Reference*.

For information about the new features in Oracle Essbase Administration Services, see the *Essbase Administration Services New Features* booklet.

CONTENTS IN BRIEF

Installation and Configuration	3
Lifecycle Management	3
Typed Measures	4
Format Strings	4
Varying Attributes	4
Strategy for Backing Up and Restoring Block Storage Databases	5
Methods for Clearing Data from Specific Regions of Aggregate Storage Databases	6
Environment Variables Used in the Calculation Scripts and Formulas of Block Storage Applications	6
Optimization of the Replication of Aggregate Storage Databases	6
Grid Expansion During Queries of Transparent Partitions	7
Log Transaction Response Times from Transparent Partition Target	7
Grid Size Limits for Aggregate Storage Transparent Partition Target	7
Batch Insertion As Data Is Exported to Relational Databases	7
Unicode Support for Aggregate Storage	7
Common Logging	8
Implied Share Override	8
XOLAP	8
Performing Multiple SQL Data Loads in Parallel to Aggregate Storage Databases	8
Calculation Functions	8
MDX Functions	9
Report Writer Commands	10

Essbase.cfg Configuration Settings 10
MaxL Statements 11
API Functions and Structures 11

Installation and Configuration

Oracle Hyperion Enterprise Performance Management System Installer, Fusion Edition and Oracle's Hyperion Enterprise Performance Management System Configurator streamline and simplify the process of installing, upgrading, and configuring Essbase. Oracle's Hyperion Enterprise Performance Management System Configurator uses the Oracle's Hyperion Shared Services Registry, which enables configuration information to be stored and reused.

Oracle's Hyperion Enterprise Performance Management System Diagnostics confirms that product components are successfully installed, configured, and running.

The installation and configuration documentation set includes a reduced number of installation guides and enables you to find information quickly and easily. Individual product installation guides are replaced by the following guides:

- *Oracle Hyperion Enterprise Performance Management System Installation Start Here*
- *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide*
- *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*

You may also need to use one or more of the following guides:

- *Oracle Hyperion Enterprise Performance Management System Manual Deployment Guide*
- *Oracle Hyperion Enterprise Performance Management System High Availability Guide*
- *Oracle Hyperion Enterprise Performance Management System Installation and Configuration Troubleshooting Guide*
- *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*
- *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*
- *Oracle Hyperion Enterprise Performance Management System SSL Configuration Guide*

Lifecycle Management

Lifecycle Management (LCM) provides a consistent way for Hyperion products to migrate applications, cubes, repositories, or artifacts across product environments and operating systems. LCM interface is integrated with Oracle's Hyperion® Shared Services Console.

LCM features:

- Methods that enable you to browse applications and folders, search for artifacts, and compare applications and folders
- Application Migration Wizard:
 - Application to application
 - Application to local file system
 - Application to server file system
 - Exported application to application

- Package file, saved and loaded
- Migration status report
- Migration audit report
- Security—LCM Administrator role
- Command line utility
- API

Note:

LCM does not migrate data.

See the *Oracle Hyperion Enterprise Performance Management System Lifecycle Management Guide*.

Typed Measures

Typed measures extend the analytical capabilities of Essbase. In addition to numeric values, measures can also be associated with text- or date-typed values.

Text measures are tagged as “text” in whichever dimension measures are represented. They enable cell values to contain one of an enumerated list of text labels. These labels are defined, at the outline level, using a mapping artifact called a text list.

Date measures are tagged as “date” in the dimension where measures are represented. Date measures enable cell values in the form of a formatted date.

Format Strings

Using format strings, you can format the values (cell contents) of Essbase database members in numeric type measures so that they appear, for query purposes, as text, dates, or other types of predefined values. The resultant display value is the cell’s formatted value.

Format strings enable you to display more meaningful values in place of raw numeric values. For example, using a text based formatted value, you might display data cells as “High,” “Medium,” and “Low.”

Varying Attributes

Prior to this release, Essbase could store and calculate measures based on attributes. However, there are cases where attributes may change over a dimension such as time, or across multiple dimensions. For example, product packaging could vary over time as well as over geographical markets.

Varying attributes enable values in the base dimension to vary by one or more independent dimensions. With varying attributes, base members (such as products) can vary by size (an attribute) across time and scenario, or across markets.

Varying attributes add the capability of Essbase to store, and calculate measures for, attributes that vary over multiple dimensions. For example, a report showing the profit per package type can be calculated according to the changing attributes. A product that was packaged in 16 ounce containers in January, but in May changed to a package type of 20 ounces, may have its profits displayed separately according to the package group where it belonged at the time.

Essbase can also calculate varying attributes based on an "as was" scenario; for example, you can calculate the profit for the different package types as the assignments were at one given time. In this approach, you set the perspective, fixing on one time and ignoring the change in attributes over time. This can be useful for analysis of profits as if the changes to the attributes did not occur.

Additionally, Essbase can perform member selections based on a changing attribute. For example, you can query for all products that had a package type of 32_ounces in April.

Strategy for Backing Up and Restoring Block Storage Databases

Regular Essbase backups, which should be integrated into production server maintenance, are key to database maintenance. Backup frequency should be determined by the volatility of the database and server environment and the need for rapid database restoration (should a server interruption occur).

To back up and restore block storage databases, you can use either of the following methods:

- Automated database backup and restore and transaction logging and replay

Backup and restore provides the equivalent functionality of manually backing up and restoring a database. When a backed-up database is restored, transactions that occurred after the backup procedure are not recovered. However, with transaction logging and replay, post-backup transactions are captured and can be replayed. Thus, a backed-up database can be recovered to the most-recent state before the interruption occurred.

The use of the database backup and restore and transaction logging and replay features eliminates the need for various manual steps and, therefore, enables administrators to back up and recover databases more efficiently. Oracle recommends incorporating these features in your backup and recovery strategy.

- Manual backup and restore

Essbase customers who have designed a backup and restore strategy that uses manual procedures and who do not need the functionality of transaction logging and replay can continue using their manual strategy.

To back up and restore aggregate storage applications, you must use manual procedures.

See the *Oracle Hyperion Enterprise Performance Management System Backup and Recovery Guide*.

Methods for Clearing Data from Specific Regions of Aggregate Storage Databases

Within an aggregate storage database, you can clear data from a specific region and retain the data located in other regions. This feature is useful when you want to delete volatile data (such as data corresponding to the last month) but retain historical data. You must have Database Manager or Administrator permission to clear data.

Methods for clearing data from a region:

- Physical, in which the input cells in the region are physically removed from the database
- Logical, in which the input cells in the region are written to a new data slice and negative, compensating values are used, thus resulting in a value of zero for the cells that are cleared

To clear data from a specified region, you can use the **alter database** MaxL statement with the **clear data in region** grammar. The syntax for the statement is as follows:

```
alter database appname.dbname clear data in region {MDX set expression}
[physical];
```

Environment Variables Used in the Calculation Scripts and Formulas of Block Storage Applications

In block storage applications, in calculation scripts and outline member formulas, you can use system environment variables as placeholders for user-specific system settings. Because environment variables are defined at the operating system level, they are available to all calculation scripts and formulas on Essbase Server.

Note:

Environment variables cannot be used in MDX queries or in member formulas that are within aggregate storage outlines.

Optimization of the Replication of Aggregate Storage Databases

To optimize the replication of an aggregate storage database when the aggregate storage database is the target and a block storage database is the source and the two outlines are identical, use one of these methods:

- The **REPLICATIONASSUMEIDENTICALOUTLINE** configuration setting in `essbase.cfg`. The setting can be enabled at the server, application, or database level. The syntax for the setting is as follows:

```
REPLICATIONASSUMEIDENTICALOUTLINE [appname [dbname]] TRUE | FALSE
```
- The **alter database** MaxL statement with the **replication_assume_identical_outline** grammar. The statement can be enabled only at the database level. The syntax for the statement is as follows:

```
alter database appname.dbname enable | disable  
replication_assume_identical_outline;
```

Both optimization methods affect only the target aggregate storage application; the source block storage application is not affected. The methods do not apply to block storage replication.

Grid Expansion During Queries of Transparent Partitions

The FORCEGRIDEXPANSION configuration setting in the `essbase.cfg` file forces the expansion of the grid when transparent partitions are queried. Use of the setting ensures that correct results are retrieved, even if most data values are displayed as #MISSING (whether or not the cells that display #MISSING contain data).

The FORCEGRIDEXPANSION setting is used with the GRIDEXPANSION configuration setting.

Log Transaction Response Times from Transparent Partition Target

For requests sent from a data source to a transparent partition target (whether a block storage or aggregate storage database), you can log transaction response times using the ENABLE_DIAG_TRANSPARENT_PARTITION configuration setting in the `essbase.cfg` file. Logging these messages is helpful when troubleshooting response times that are too slow.

Grid Size Limits for Aggregate Storage Transparent Partition Target

When the transparent partition target is an aggregate storage database, you can specify the maximum size of the request grid and the response grid, using the MAX_REQUEST_GRID_SIZE and MAX_RESPONSE_GRID_SIZE configuration settings.

Batch Insertion As Data Is Exported to Relational Databases

The DATAEXPORTENABLEBATCHINSERT configuration setting in the `essbase.cfg` file specifies whether, when Essbase data is exported for direct insertion into a relational database, the batch-insert method, instead of the default row-insert method, is used. Rows of data can be inserted only if the relational database and the ODBC driver support the functionality.

Unicode Support for Aggregate Storage

Aggregate storage applications and databases can be Unicode-mode enabled.

Common Logging

Common logging enables users and administrators who are using multiple Hyperion products to find log messages in a common place, thus reducing time and effort for tracing the appropriate log file.

Accordingly, Essbase log files are located in *HYPERION_HOME/logs/essbase*.

Implied Share Override

Using Outline Editor in Administration Services Console, you can force implied share settings for aggregate storage and block storage outlines. For block storage databases, you can override the default Implied Share setting of active by using configuration file settings to control the Essbase Server default and application default settings. Two API functions are provided for programming convenience: `EssOtlGetImpliedShare` and `EssOtlSetImpliedShare`.

XOLAP

XOLAP (extended online analytic processing) is a variation on the role of OLAP in business intelligence. Specifically, XOLAP is an Essbase multidimensional database that stores only the outline metadata and retrieves data from a relational database at query time. XOLAP thus integrates a source relational database with an Essbase database, leveraging the scalability of the relational database with the more sophisticated analytic capabilities of a multidimensional database. Your business needs determine whether OLAP or XOLAP is best suited to your environment.

For information on how to designate models for XOLAP, see the Oracle Essbase Studio online help.

Performing Multiple SQL Data Loads in Parallel to Aggregate Storage Databases

When loading SQL data into aggregate storage databases, you can use up to eight rules files to load data in parallel. Each rules file must use the same authentication information (SQL user name and password).

Essbase initializes multiple temporary aggregate storage data load buffers (one for each rules file), where data values are sorted and accumulated. When the data is fully loaded into the data load buffers, Essbase commits the contents of all buffers into the database in one operation, which is faster than committing buffers individually.

Calculation Functions

The following calculation functions are new:

- @ILANCESTORS—Returns the specified members and either all ancestors of the members or the ancestors up to a specified distance
- @ILDESCENDANTS—Returns the specified members and either all descendants of the members or the descendants down to a specified distance
- @LANCESTORS—Returns all ancestors of the specified members or the ancestors up to a specified distance
- @LDESCENDANTS—Returns all descendants of the specified members or the descendants down to a specified distance
- @NEXTSIBLING—Returns the next sibling (the sibling to the immediate right) of the specified member
- @PREVSIBLING—Returns the previous sibling (the sibling to the immediate left) of the specified member
- @SHIFTSIBLING—Returns the specified member or the *n*th sibling of the specified member
- @ENUMVALUE—Returns the internal numeric value for a text value in a text list.
- @DATEDIFF—Returns the difference between two input dates.
- @DATEPART—Returns returns the Year/Quarter/Month/Week/Day/DayOfYear/Weekday as a number.
- @DATEROLL—To the given date, rolls (adds or subtracts) a number of specific time intervals.
- @FORMATDATE—Returns a formatted date string.
- @TODATEEX—Returns the numeric date value from input date-string according to the date-format specified.

MDX Functions

The following MDX functions are new:

- CellValue—Returns the numeric value of the current cell.
- EnumText—Returns the text value corresponding to a numeric value in a text list.
- EnumValue—Returns the internal numeric value for a text value in a text list.
- RealValue—Returns a value for the specified member or tuple without the inherited attribute dimension context.
- NumToStr—Converts a double-precision floating-point value into a decimal string.
- GetNextDay—To the given date and the week day, get the next date after input date that corresponds to the week day.
- GetFirstDay—For a given date_part, this function returns the first day of the time interval for the input date.
- GetLastDay—For a given date_part, this function returns the last day of the time interval for the input date.

- JulianDate—For the given UNIX date, get its Julian date.
- UnixDate—For the given Julian date, get its UNIX date.
- AttributeEx—This function is for varying attributes. Given the attribute member and the perspective setting, returns the associated base member list.
- WithAttrEx—This function is for varying attributes. Given the attribute dimension, condition, predicate, and perspective setting, returns the base member list satisfying the predicate.

The WITH PERSPECTIVE MDX clause enables you to qualify queries based on varying attribute perspective.

The MdxFormat directive enables the creation of format strings to change the appearance of member values.

Report Writer Commands

The following Report Writer commands are new:

- <ATTRIBUTEVA
- <WITHATTREX
- <PERSPECTIVE
- {MEANINGLESTEXT}
- {OUTFORMATTEDMISSING}
- {OUTFORMATTEDVALUES}
- {OUTMEANINGLESS}

Essbase.cfg Configuration Settings

The following essbase.cfg settings are new:

- DATAEXPORTENABLEBATCHINSERT
- ENABLE_DIAG_TRANSPARENT_PARTITION
- EXPORTFILESIZELIMIT
- FORCEGRIDEXPANSION
- HISLEVELDRILLTHROUGH
- MAXERRORMBRVERIFYREPORT
- MAX_REQUEST_GRID_SIZE
- MAX_RESPONSE_GRID_SIZE
- REPLICATIONASSUMEIDENTICALOUTLINE
- SPLITARCHIVEFILE
- TRANSACTIONLOGDATALOADARCHIVE

- TRANSACTIONLOGLOCATION

MaxL Statements

The following MaxL grammar was added:

- alter database...replay transactions...
- alter database...[force] archive...
- alter database...[force] restore...
- alter database...enable|disable replication_assume_identical_outline (aggregate storage only)
- alter database...clear aggregates|data in region (aggregate storage only)
- alter database...begin|end archive (new for aggregate storage)
- query database...list transactions... (block storage only)
- alter session set dml_output formatted_value on|off
- alter session set dml_output get_missing_cells on|off

The following MaxL and MaxL Shell statements were added:

- deploy MaxL Shell statement for deploying a cube from Oracle Essbase Studio
- query archive_file MaxL statement for getting information about the database backup archive file

API Functions and Structures

The following functionality had been added to the Oracle Essbase API:

Implied Share API Functions

The following API functions are new:

- EssOtlGetImpliedShare
- EssOtlSetImpliedShare

Incremental Dimbuild API Functions

The following API functions are new:

- EssIncrementalBuildDim
- EssBeginIncrementalBuildDim
- EssBeginStreamBuildDim
- EssEndIncrementalBuildDim

- EssEndStreamBuildDim

Typed Measures API Functions

The following new API functions are related to typed measures and the related text list objects:

- EssOtlCreateObject
- EssOtlPutSmartList
- EssOtlGetMemberSmartList
- EssOtlDeleteObject
- EssOtlGetSmartListInfo
- EssOtlFindObject
- EssOtlFreeObjectArray
- EssOtlFreeSmartListInfo
- EssOtlGetMemberSmartList
- EssOtlGetMemberType
- EssOtlGetObjectReferenceCount
- EssOtlGetObjectReferences
- EssOtlImportExportObject
- EssOtlListObjects
- EssOtlQueryObjects
- EssOtlSetMemberType
- EssOtlSetMemberTypeToSmartList

The following new MDX API functions are also related to typed measures:

- EssMdxGetCellInfo
- EssMdxGetFormattedValue
- EssMdxGetSmartlistforCell
- EssMdxGetFormatString
- EssMdxSetQueryCellProperties
- EssMdxGetQueryCellProperties
- EssMdxGetQueryOptions
- EssMdxSetQueryOptions

The following new Grid API functions are also related to typed measures:

- EssGGetFormattedValue
- EssGGetSmartlistforCell
- EssGGetGridPerspective

The following Grid API types are new:

- ESSG_CA_MISSINGCELL
- ESSG_CA_OUTOFRANGE

Varying Attributes API Functions

The following new C Outline API functions are related to varying attributes:

- EssOtlVaryingAssociateAttributeDimension
- EssOtlVaryingGetAttributeIndepDims
- EssOtlVaryingGetAssociatedAttributes
- EssOtlQueryVaryingAttributes

The following new Grid API function is related to varying attributes:

EssGGetGridPerspective

The following Grid API types are new:

- ESSG_PERSP_EXPLICIT
- ESSG_PERSP_REALITY

Database Backup and Transaction Logging API Functions

The following API functions are new:

- EssArchiveDatabase
- EssRestoreDatabase
- EssListTransactions
- EssReplayTransactions

The following API structures are new:

- ESS_DISKVOLUME_REPLACE_T
- ESS_LOG_DATALOAD_T
- ESS_SEQID_T
- ESS_TRANSACTION_ENTRY_T
- ESS_TRANSACTION_REPLAY_INP_T
- ESS_TRANSACTION_REQSPECIFIC_T

MDX and Grid API Functions

The following Grid API functions are new:

- EssGGetFormattedValue

- EssGGetSmartlistforCell
- EssGGetGridPerspective

The following Grid API types are new:

- ESSG_PERSP_EXPLICIT
- ESSG_PERSP_REALITY
- ESSG_CA_MISSINGCELL
- ESSG_CA_OUTOFRANGE

COPYRIGHT NOTICE

Essbase New Features, 11.1.1

Copyright © 2008, Oracle and/or its affiliates. All rights reserved.

Authors: EPM Information Development Team

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable: U.S. GOVERNMENT RIGHTS: Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.