



ORACLE® ESSBASE

Release 11.1.2.2.100

New Features



This document describes the new features in Oracle Essbase Release 11.1.2.2.100, 11.2.2.000, 11.1.2.1, and 11.1.2. For detailed information on these features, see the *Oracle Essbase Database Administrator's Guide*, *Oracle Essbase Technical Reference*, and *Oracle Essbase API Reference*.

For information about the new features in Oracle Essbase Administration Services, see the *Administration Services Readme*.

CONTENTS IN BRIEF

Essbase 11.1.2.2.100 New Features	2
Essbase 11.1.2.2.000 New Features	3
Essbase 11.1.2.1 New Features	9
Essbase 11.1.2 New Features	14

Essbase 11.1.2.2.100 New Features

Subtopics

- [UNC Path Support](#)
- [New Calculation Function](#)
- [New essbase.cfg Configuration Setting](#)

UNC Path Support

When working with Essbase clusters to achieve high availability of servers, it is customary to use network file systems for sharing the Essbase application files across systems.

On Windows, you can use Universal/Uniform Naming Convention (UNC) to describe the location of a network resource, such as a shared directory. Syntax:

```
\\ComputerName\SharedFolder\Resource
```

For Essbase running on Windows, UNC paths are supported to specify network shared paths for the following Essbase Server application artifacts:

- *ARBORPATH*
- Tablespace paths for aggregate storage applications
- Disk volume paths for block storage applications

New Calculation Function

@IRREX: Calculates the Internal Rate of Return on a cash flow that must contain at least one investment (negative) and one income (positive) value. Includes functionality to configure the initial guess and the number of iterations the algorithm can make.

The **@IRREX** calculation function is an extension of the **@IRR** calculation function, in which the initial guess of 0.07 cannot be changed.

New essbase.cfg Configuration Setting

SSBULKGRIDPROCESSING: Optimizes asymmetric Grid API queries for XOLAP.

Essbase 11.1.2.2.000 New Features

Subtopics

- [Block Storage Performance Enhancements](#)
- [Optimization of Distinct Count Calculations on Aggregate Storage Databases](#)
- [Distinct Way to Reference Shared Members in MDX](#)
- [Changing Essbase Server and Application Log Levels in Administration Services](#)
- [MDX Property Expressions](#)
- [Throttle Active Transactions](#)
- [Essbase API Changes](#)
- [Configuration Setting \(essbase.cfg\) Changes](#)
- [SET CALCPARALLEL Calculation Command Enhancement](#)
- [New MDX Functions](#)
- [MaxL Grammar Changes](#)

Block Storage Performance Enhancements

Subtopics

- [Block Storage Parallel Calculation](#)
- [Block Storage Parallel Restructuring](#)
- [Block Storage Parallel Data Export](#)
- [Block Storage Parallel Data Load](#)
- [Optimizing Block Storage Parallel Data Load](#)
- [Cache Sizes Larger than 4 GB](#)

Block Storage Parallel Calculation

Calculation tasks are generated along the last n sparse dimensions of an outline. These sparse dimensions used to identify tasks are called task dimensions. The number of task dimensions, n , is either selected dynamically by Essbase, or you can override the number by specifying a value for CALCTASKDIMS in the `essbase.cfg` file.

Block Storage Parallel Restructuring

By default, block storage restructuring is performed sequentially. Blocks are renumbered and reshaped from first to last, a time-intensive process. Parallel restructuring can reduce restructuring time by dividing block restructuring work across multiple concurrent threads to use available processor cores. Because calculation is performed separately from restructuring, each block can be restructured independently of other blocks.

The number of restructuring threads to use is defined in `essbase.cfg` using the `RESTRUCTURETHREADS` configuration setting.

Block Storage Parallel Data Export

Using the **export data** MaxL statement to export data from a block storage database in parallel, you specify a comma-separated list of export files, up to a maximum of 1024 file names. The number of file names determines the number of export threads. The number of available block-address ranges limits the number of export threads that Essbase actually uses. Essbase divides the number of actual data blocks by the specified number of file names (export threads). If there are fewer actual data blocks than the specified number of export threads, the number of export threads that are created is based on the number of actual data blocks. This approach results in a more even distribution of data blocks between export threads.

Note: In specifying the number of export files, it is important to consider the number of available CPU cores and I/O bandwidth on the computer on which Essbase Server runs. Specifying too large a number can result in poor performance.

The EXPORTTHREADS configuration setting has been enhanced to support block storage parallel data export. (EXPORTTHREADS is used only when executing the PAREXPORT ESSCMD command without specifying the -threads option.)

Block Storage Parallel Data Load

Parallel data load refers to the concurrent loading of multiple data files into an Essbase database. When working with large data sets (for example, a set of ten 2 GB files), loading the data sources concurrently enables you to fully utilize the CPU resources and I/O channels of modern servers with multiple processors and high-performance storage subsystems.

Parallel data load uses multiple parallel pipelines on the server side, and multiple threads on the client-side, to load multiple data files concurrently, thus enabling data loads to be truly optimized to the capabilities of modern servers.

Optimizing Block Storage Parallel Data Load

Parallel data load includes a “throttle” to limit the number of client threads or server pipelines that are used. A throttle control is important because if the data-file specification matches several hundred files, you should avoid spawning that many threads or pipelines. To control the number of threads or pipelines spawned by a data load request, set the limit using the **max_threads** grammar in the **import data** MaxL statement.

Cache Sizes Larger than 4 GB

The 64-bit Essbase Server has been expanded to accommodate larger index, data, and data file cache sizes without setting a scaling factor. The MEMSCALINGFACTOR configuration setting is deprecated. For full database functionality, you must upgrade to the latest client software.

For more information, see the Essbase Readme for this release.

Optimization of Distinct Count Calculations on Aggregate Storage Databases

On aggregate storage databases only, the NonEmptyCount MDX function is optimized so that the calculation of the distinct count for all cells can be performed by scanning the database only once. Without this optimization, the database is scanned as many times as the number of cells corresponding to the distinct count. The NonEmptyCount optimization is triggered when an outline member formula has the following syntax:

```
NONEMPTYCOUNT(set, measure, exclude_missing)
```

The `exclude_missing` parameter supports the NonEmptyCount optimization on aggregate databases by improving the performance of a query that queries metrics that perform a distinct count calculation.

Distinct Way to Reference Shared Members in MDX

In an outline that is enabled for duplicate member names, all member names, including shared members, are distinguishable by using a qualified member name syntax. However, in a unique member name outline (where duplicate members are *not* enabled), a member and its shared member are both identified by the same simple member name.

In previous releases, any reference in an MDX query to such a shared member was assumed to be a reference to the original member, not the shared member. With this release, you can get a unique name for shared members by using the MEMBER_UNIQUE_NAME property in an MDX query.

For example, in the Sample Basic database, the member [100-20] is an original member under parent [100], and has a shared member associated with it under parent [Diet]. The shared member [100-20] can be referred to explicitly, using the unique name [Diet].[100-20], as shown in the following query:

```
SELECT
  {Sales}
ON COLUMNS,
  {[Diet].[100-20]}} PROPERTIES MEMBER_UNIQUE_NAME
ON ROWS
FROM Sample.Basic;
```

For rules on the syntax and escape characters required to uniquely reference shared members in a unique member name outline, see the topic entitled “MDX Syntax for Specifying Duplicate Member Names and Aliases” in the *Oracle Essbase Technical Reference*.

Changing Essbase Server and Application Log Levels in Administration Services

You can change Essbase log levels at the server and application level in Administration Services Console. See “Changing Log Levels” in the *Oracle Essbase Administration Services Online Help*.

MDX Property Expressions

In addition to querying for intrinsic and custom properties of a particular member, you can also query for properties of related members, using the `PROPERTY_EXPR` function within the `DIMENSION PROPERTIES` section of the query axis.

See the MDX Property Expressions topic and the `CurrentAxisMember` topic in the *Oracle Essbase Technical Reference*

Throttle Active Transactions

Each allotted server thread for an application may create child threads for tasks such as parallel calculation, parallel data load or export, and parallel restructuring. However, if the total number of running threads is too high, threads may lose efficiency while contending for server resources. To control the issue of thread contention, you can use the `MAXTOTALACTIVETRANSACTIONS` and `MAXACTIVEUPDATETRANSACTIONS` `essbase.cfg` settings.

Essbase API Changes

Subtopics

- [Changed C Main API Structures](#)
- [Compiler Support](#)

Changed C Main API Structures

The data structures `ESS_DBINFO_T` and `ESS_DBSTATE_T` have expanded fields to accommodate 64-bit long typedefs. This is related to [“Cache Sizes Larger than 4 GB”](#) on page 4.

Compiler Support

The supported compilers list for building Essbase API programs has been updated. See the [“Supported Compilers”](#) topic in the *Oracle Essbase API Reference*.

Configuration Setting (essbase.cfg) Changes

Subtopics

- [New Configuration Settings](#)
- [Deprecated Configuration Setting](#)
- [CALCPARALLEL Configuration Setting Enhancement](#)

New Configuration Settings

- MAXACTIVEUPDATETRANSACTIONS. See [“Throttle Active Transactions”](#) on page 6.
- MAXTOTALACTIVETRANSACTIONS. See [“Throttle Active Transactions”](#) on page 6.
- NUMBLOCKSTOEXTEND: Determines the number of bytes by which data files in block storage databases are extended to accommodate block updates that require additional disk space.
- RESTRUCTURETHREADS: Specifies whether parallel restructuring is enabled for a database, and the number of threads to use.
- XOLAPMAXINCLAUSESIZE: Sets the maximum number of INLIST items allowed in SQL clauses (for XOLAP databases).

Deprecated Configuration Setting

The MEMSCALINGFACTOR configuration setting is deprecated. The 64-bit Essbase Server has been expanded to accommodate larger cache sizes without setting a scaling factor.

CALCPARALLEL Configuration Setting Enhancement

The maximum number of threads that can be made available for parallel calculation has been increased:

- Block storage databases:
 - Running on 32-bit platforms: 64
 - Running on 64-bit platforms: 128
- Aggregate storage databases (running on 32-bit or 64-bit platforms): 128

SET CALCPARALLEL Calculation Command Enhancement

The maximum number of threads that can be made available for parallel calculation has been increased:

- 32-bit platforms: 64
- 64-bit platforms: 128

New MDX Functions

- **PROPERTY_EXPR** function (MDX Property Expressions), for querying properties of ancestor members.
- **CurrentAxisMember**, for use with the **PROPERTY_EXPR** function.

MaxL Grammar Changes

- The **query database** MaxL statement for aggregate storage databases has a new **list aggregate_storage_slice_info** clause, to display information about data slices and views.
- The **deploy** MaxL statement for deploying cubes from Oracle Essbase Studio has the following new grammar:
 - If an Essbase model is enabled for XOLAP, you can, when redeploying the cube, use the **outline_in_background** grammar.
 - When the model being deployed contains single or multiple relational data sources, you can deploy in streaming mode using the **use streaming build** grammar if you want Essbase Studio to query the external data source directly (rather than using an ODBC connection).
- The **import data** MaxL statement has a new **using max_threads** clause, for specifying maximum number of threads to use for a parallel data load.
- The **alter system** and **alter application** MaxL statements have new **set ... message_level** clauses, to set the message logging level at the agent and application scopes.
- The **display system** and **display application** MaxL statements have new **message_level** clauses, to display the current message level settings.

Essbase 11.1.2.1 New Features

Subtopics

- [Essbase Security Backup File Enhancements](#)
- [Communicating with Essbase Using SSL](#)
- [Logging In to Essbase Using Logical Names](#)
- [Starting Essbase on Ports Qualified by Host Names](#)
- [MDX Query Execution Management](#)
- [Essbase API](#)
- [New MDX Functions](#)
- [New essbase.cfg Configuration Settings](#)
- [MaxL Statements and Grammar Changes](#)

Essbase Security Backup File Enhancements

Subtopics

- [Managing Essbase Security Backup Files](#)
- [Reconciling the Essbase Security File to the State of Essbase on an External Disk](#)

Managing Essbase Security Backup Files

You can manage the number of security backup files (*essbase_timestamp.bak*) that Essbase maintains; the interval at which the security backup files are created; and whether to switch to the latest, valid security backup file on startup if the *essbase.sec* file is invalid.

- **NUMBEROFSECFILEBACKUPS** configuration setting: Specifies the maximum number (up to 10) of security backup files that Essbase creates and maintains. By default, Essbase maintains two versions of *essbase_timestamp.bak*.
- **SECFILEBACKUPINTERVAL** configuration setting: Specifies the amount of time Essbase waits before creating a security backup file.
- **ENABLESWITCHTOBACKUPFILE** configuration setting: Specifies whether Essbase loads a valid security backup file at startup if the *essbase.sec* file is invalid.

- **Pre-upgrade Security File Backup**

When you upgrade to Essbase 11.1.2.1 from an earlier release, a backup of the security file for the earlier release is created before the security file is upgraded. The security file backup, *Essbase.Bak_preUpgrade*, is in *ARBORPATH/bin*. Unlike the *essbase_timestamp.bak* file, which regularly backs up the latest state of Essbase security, this pre-upgrade backup file is kept intact and is not subsequently updated by further operations.

Reconciling the Essbase Security File to the State of Essbase on an External Disk

When Essbase is started using the latest, valid security backup file (`essbase_timestamp.bak`) instead of `essbase.sec`, you can use the **alter system** MaxL statement to reconcile the security file to match the state of Essbase applications and databases on an external disk.

The **alter system reconcile** grammar logs messages in `essbase.log` when:

- An application or database folder is on the disk but not in the security file
- An application or database is in the security file but not on the disk. In this scenario, using the **alter system reconcile force** grammar removes the application or database from the security file.

Communicating with Essbase Using SSL

Essbase supports the Secure Socket Layer (SSL) protocol for all client-to-server and server-to-server communication. SSL allows Essbase to communicate unilaterally across a network with Essbase clients, including Oracle Hyperion Provider Services, Oracle Essbase Administration Services, and Essbase Studio, while preventing eavesdropping and tampering. End-point authentication and communications occur confidentiality over the Internet using CipherSuite cryptography.

These configuration settings supply parameters that enable SSL communication:

- AGENTSECUREPORT
- CLIENTPREFERREDMODE
- ENABLECLEARMODE
- ENABLESECUREMODE
- NETSSLHANDSHAKETIMEOUT
- SSLCIPHERSUITES
- WALLETPATH

For information on using SSL with Essbase, see the *Oracle Hyperion Enterprise Performance Management System Security Administration Guide*.

Logging In to Essbase Using Logical Names

Essbase cluster logical names can be used for login, in the form `<name>:<secure>`. The name is resolved by communication with the Provider Services servers specified in configuration files:

- APSRESOLVER in `essbase.cfg`—Server-to-server communication; for example, when defining Essbase servers for partitions or in the @XREF or @XWRITE calculation functions

- `aps.resolver.urls` in `essbase.properties`—Client-to-server communication; for example, when connecting to Essbase from Oracle Essbase Spreadsheet Add-in or Oracle Hyperion Smart View for Office

See *Oracle Hyperion Provider Services Administration Guide*.

Starting Essbase on Ports Qualified by Host Names

You can specify the host name to which Essbase binds by using the `ESSBASESERVERHOSTNAME` configuration setting in `essbase.cfg`. If no host name is specified, Essbase uses a system API to get the host name. Using host names enables you to partition network traffic on all the network interfaces of a computer. This is useful on computers with single network cards and those with multiple NIC cards.

MDX Query Execution Management

The `essbase.cfg` settings `QRYGOVEXECBLK` and `QRYGOVEXECTIME` can now also control long-running MDX queries.

Essbase API

Subtopics

- [New C Main API Functions](#)
- [New C Main API Structures](#)

New C Main API Functions

New user and group identity functions are added, enabling the specification of user directories and unique identity attributes during typical operations. The parameters of these functions identify users and groups that are hosted in a directory.

- `EssAddToGroupEx`
- `EssCreateExtGroup`
- `EssDeleteFromGroupEx`
- `EssDeleteGroupEx`
- `EssDeleteUserEx`
- `EssGetApplicationAccessEx`
- `EssGetDatabaseAccessEx`
- `EssGetGroupInfoEx`
- `EssGetGroupListEx`
- `EssGetUserInfoEx`

- EssKillRequestEx
- EssListConnectionsEx
- EssListGroupInfoEx
- EssListLocksEx
- EssListLoginsEx
- EssListRequestsEx
- EssListUsersInfoEx
- EssSetApplicationAccessEx
- EssSetCalcListEx
- EssSetDatabaseAccessEx
- EssSetFilterListEx
- EssSetGroupListEx

New C Main API Structures

The following new structures are related to the specification of user directories and unique identity attributes.

- ESS_CONNECTINFOEX_T
- ESS_LOCKINFOEX_T
- ESS_REQUESTINFOEX_T
- ESS_USERAPPEX_T
- ESS_USERDBEX_T
- ESS_USERINFOID_T

New MDX Functions

- DTS
- IsMatch

New essbase.cfg Configuration Settings

- AGENTSECUREPORT
- APSRESOLVER
- CLIENTPREFERREDMODE
- ENABLECLEARMODE
- ENABLESECUREMODE
- ENABLESWITCHTOBACKUPFILE

- ESSBASESERVERHOSTNAME
- NUMBEROFSECFILEBACKUPS
- SECFILEBACKUPINTERVAL
- SSLCIPHERSUITES
- SSOPTIMIZEDGRIDPROCESSING
- WALLETPATH

MaxL Statements and Grammar Changes

Subtopics

- [New MaxL Statements](#)
- [Changed MaxL Grammar](#)

New MaxL Statements

- The statement `display system` has new **configuration** grammar for displaying Essbase configured values set using the `essbase.cfg` file.

Changed MaxL Grammar

- Changes to `alter system` statement: The `reconcile` keyword displays discrepancies in application and database information between the security file and the external disk; the `reconcile force` keyword removes from the `essbase.sec` security file applications or databases that do not exist on the external disk but that are listed in the security file.
- Changes to `login` statements:
 - The `HOST-NAME` parameter can be an Essbase cluster logical name provided that you use the `APSRESOLVER` configuration setting to specify a Oracle Hyperion Provider Services server to resolve the logical name to the corresponding physical name.
 - If using SSL, append `:secure` to the `HOST-NAME` parameter; for example:

```
login essexer password on "localhost:secure";
```

Essbase 11.1.2 New Features

Subtopics

- [EPM System Security Enhancements](#)
- [Allocations on Aggregate Storage Databases](#)
- [Custom Calculations on Aggregate Storage Databases](#)
- [Dynamic Write-back to Blocks During Calculation](#)
- [Member Query Improvements](#)
- [Alias Table Enhancements](#)
- [Increased Length for User, Group, and Filter Names](#)
- [Conversion of Block Storage Outlines to Unicode-mode Aggregate Storage Outlines](#)
- [IPv6 Support](#)
- [OCI Support](#)
- [Monitoring Progress of Dimension Builds and Data Loads](#)
- [Dataload Error Debugging Improvements](#)
- [Increased Limit for MDX Query Size](#)
- [User Names with Leading Special Characters](#)
- [MaxL Shell Error Handling Improvements](#)
- [Exception Messages in Calculation Scripts](#)
- [Metadata Export to XML](#)
- [Essbase Failover Support with OPMN](#)
- [Specifying Shared Services Project Names](#)
- [MaxL Support in Java API](#)
- [Calculation with Varying Attributes](#)
- [Essbase API](#)
- [New Calculation Functions](#)
- [New Calculation Commands](#)
- [New essbase.cfg Configuration Settings](#)
- [MaxL Statements and Grammar Changes](#)

EPM System Security Enhancements

Subtopics

- [EPM System Security for Users and Groups](#)
- [User and Group Identities](#)

EPM System Security for Users and Groups

When running Essbase in Oracle Enterprise Performance Management System security mode, Essbase obtains user and group details (including user and group information and provisioning to Essbase applications) from Oracle Hyperion Shared Services. Essbase does not store all users and groups in the Essbase security file (`essbase.sec`); therefore, an Essbase Administrator does not need to explicitly synchronize security between Essbase and Shared Services.

When a user logs on to Essbase, Essbase queries Shared Services for that user's information. The privileges with which a user starts a session are preserved throughout the session, regardless of whether the user's privileges are changed in Shared Services during the session.

The Essbase Administrator can optionally delete a user or group from the Essbase security file. Any calculation or filter assignments are removed; however, the user still exists in Shared Services and is not deprovisioned.

Externally authenticated Essbase groups can be created. The specified group must exist in Shared Services.

Additionally, Essbase accepts the provider name along with the user or group name for user and group management and authentication. For example, users can be created and can log in with the format `userName@providerName`. This enables the coexistence of multiple Essbase users and groups with the same name, who are hosted on different provider domains. See [“User and Group Identities” on page 15](#).

User and Group Identities

Essbase, when in EPM System security mode, now enables user and group names to be non unique, if you specify the user or group's provider directory or unique identity attribute.

In MaxL, user and group names can be specified as `name@provider` or as a unique identity attribute.

The provider is the name of a user directory, such as LDAP or Active Directory, where the external user or group is hosted. The unique identity attribute, or "identity," is a unique string assigned to every user and group. The identity enables Essbase to distinguish between users and groups with the same name across providers.

Allocations on Aggregate Storage Databases

Allocations are used in the budgeting process to distribute revenues or costs.

The allocations feature allows you to allocate a given source amount to a target range of cells in an aggregate storage database. The source amount can be allocated to the target proportionately, based on a given basis, or the source amount can be spread evenly to the target.

You can perform aggregate storage allocations using the MaxL statement `execute allocation`. Using the Essbase API, use the `EssPerformAllocationAso` function and provide the information about the allocation in the `ESS_PERF_ALLOC_T` API structure.

Custom Calculations on Aggregate Storage Databases

Custom calculations extend the analytical capabilities of Essbase by enabling the execution of recurring calculations on aggregate storage databases.

You can write custom calculations for aggregate storage databases that update target level-0 cells. Custom calculation scripts are expressed in MDX.

Custom calculations on aggregate storage databases can be useful when the database is used for general ledger reporting, where double-entry accounting is in effect. Debit items, such as assets and expenses, must balance with credit items, such as equity and revenue.

You can perform custom calculations on an aggregate storage database using the MaxL statement `execute calculation` (aggregate storage version). Using the Essbase API, use `EssPerformCustomCalcASO` to perform or verify a custom calculation. Provide the information for the custom calculation in the `ESS_PERF_CUSTCALC_T` structure.

Dynamic Write-back to Blocks During Calculation

Using the `@XWRITE` calculation function, Essbase can write to blocks other than the ones being calculated. The write can occur in the same database or on a remote database. `@XWRITE`, like `@XREF`, uses a location alias.

Member Query Improvements

Applications developed to work with Essbase send numerous requests to Essbase upon startup or as a response to a user action. To improve response times when users query for member information, you can have the application send multiple similar queries as one, using these new API functions:

- `EssOtlGetMemberInfoArray` (an array version of `GetMemberInfo`)
- `EssOtlQueryMembersExArray` (an array version of `QueryMembersEx`)

Alias Table Enhancements

Subtopics

- [Increased Limit for Alias Tables](#)
- [Multiple Language Code Support for Alias Tables](#)

Increased Limit for Alias Tables

Essbase supports 32 alias tables for block storage and aggregate storage databases.

Multiple Language Code Support for Alias Tables

Using the Essbase API, you can specify multiple language codes for an alias table. When you create an alias table, a language code is not specified. You can get the set of language codes and clear the language codes associated with an alias table.

When clearing aliases from an alias table, language codes are removed from the alias table.

When copying an alias table, language codes are removed from the copied alias table.

When renaming an alias table, language codes are preserved in the renamed alias table.

Increased Length for User, Group, and Filter Names

Essbase supports these limits for user, group, and filter names:

- Non-Unicode application limit: 256 bytes
- Unicode-mode application limit: 256 characters

Conversion of Block Storage Outlines to Unicode-mode Aggregate Storage Outlines

In addition to supporting the conversion of non-Unicode block storage outlines to non-Unicode aggregate storage outlines, Essbase supports these conversion scenarios:

- Non-Unicode block storage outline to Unicode aggregate storage outline
- Unicode block storage outline to Unicode aggregate storage outline

IPv6 Support

Essbase supports the IPv6 internet protocol on Microsoft Windows 2008 and all UNIX platforms.

OCI Support

Essbase supports connecting to Oracle databases using the Oracle Call Interface (OCI). You can load data from Oracle databases using Data Prep Editor in Administration Services Console by specifying an OCI connect identifier. Additionally, you can specify an OCI connect identifier when deploying a cube using the MaxL **deploy** statement. For information on Oracle Essbase Studio support for OCI, see *Oracle Essbase Studio New Features*.

Monitoring Progress of Dimension Builds and Data Loads

Previously, data loads and dimension builds were always synchronized processes, meaning that Essbase has to wait for them to finish before making modifications.

If you use asynchronous data loads and dimension builds, during the process you can query for this information:

- State of dimension build/data load process: whether it is in progress, in the final stages, or completed
- Stage of the dimension build/data load process: whether opening the data source, reading the outline, building dimensions, verifying an outline, or writing an outline
- Number of data records processed and rejected so far
- The name and location of the error file
- Data records processed and rejected so far

These C Main API functions are added to handle asynchronous data loads and dimension builds:

- EssAsyncBuildDim issues an asynchronous dimension build request
- EssAsyncImport and EssAsyncImportASO issue asynchronous data load requests
- EssGetAsyncProcLog gets the error log for the asynchronous process
- EssGetAsyncProcState queries the state of the asynchronous process
- EssCancelAsyncProc cancels the asynchronous process
- EssCloseAsyncProc closes the connection for a finished or canceled asynchronous dimension build or data load

Dataload Error Debugging Improvements

Essbase offers improved dataload error messages for debugging purposes, and records causing errors are included in the log files.

These C Main API functions are added to improve debugging of dataload failures:

- EssUpdateEx—Sends an update specification to the active database as a single string.
- EssUpdateUtf8Ex—Sends an update specification to the active database as a single UTF-8-encoded string.
- EssUpdateFileEx—Sends an update specification to the active database from a file.
- EssUpdateFileUtf8Ex—Sends an update specification to the active database from a UTF-8-encoded file.
- EssUpdateFileASOEx—Sends an update specification to the active aggregate storage database from a file.
- EssUpdateFileASOUtf8Ex—Sends an update specification to the active aggregate storage database from a UTF-8-encoded file.

Increased Limit for MDX Query Size

Essbase supports MDX queries that exceed the 2^{32} query limit. Essbase sends 64-bit values representing cell offsets to the client. The Java MDX API now handles 64-bit offsets.

User Names with Leading Special Characters

User names can contain any characters defined within the code page referenced by the ESSLANG variable.

MaxL Shell Error Handling Improvements

The MaxL Shell **IfError** command can detect syntactical errors, nested script errors, and lack-of-permission errors. The MaxL Shell **Exit** command can accept a non-zero argument to return an exit status to the parent shell.

Exception Messages in Calculation Scripts

Essbase can exit calculations immediately under specified logical conditions. You can use the IF... ELSEIF calculation command block to specify the logical error conditions, and use the new @RETURN function to exit the calculation with customized error messages and levels.

Metadata Export to XML

You can extract metadata, either from the active database outline or an input outline file, to an XML file. Comparing versions of XML files enables you to view and track changes made to the outline over time.

The MaxL statement **export outline** is added to support exporting the metadata.

Essbase Failover Support with OPMN

Essbase supports clustering of Essbase servers to provide active-passive failover with write-back capability. Active-passive Essbase failover clusters use the service failover functionality of the Oracle Process Manager and Notification (OPMN) Server. Essbase agent and server processes use leases to ensure that at any given point-in-time there is one and only one owner for a given shared resource. These processes acquire a lease against a particular shared resource upon startup, continue to renew the lease during their lifetime, and surrender the lease upon termination.

These configuration settings supply tuning parameters that support Essbase failover clustering:

- **AGENTLEASEEXPIRATIONTIME**—Sets the maximum amount of time an agent can own a lease before the lease expires.
- **AGENTLEASEMAXRETRYCOUNT**—Specifies the number of times the agent attempts to acquire or renew a lease. If these attempts are unsuccessful, the agent terminates itself.
- **AGENTLEASERENEWALTIME**—Specifies the time interval in seconds after which the agent attempts to renew lease. This should always be less than **AGENTLEASEEXPIRYTIME**.
- **FAILOVERMODE**—Determines whether Essbase is deployed as a failover cluster.
- **SERVERLEASEEXPIRATIONTIME**—Sets the maximum amount of time Essbase Server can own a lease before the lease expires.
- **SERVERLEASEMAXRETRYCOUNT**—Specifies the number of times Essbase Server attempts to acquire or renew a lease. If these attempts are unsuccessful, Essbase Server terminates itself.
- **SERVERLEASERENEWALTIME**—Specifies the time interval in seconds after which Essbase Server attempts to renew lease. This should always be less than **SERVERLEASEEXPIRYTIME**.

Specifying Shared Services Project Names

If you migrate Essbase Server and any existing users and groups to Oracle Enterprise Performance Management System security mode, you can change the name of the global application and application project in Oracle Hyperion Shared Services by using the **alter system** MaxL statement with the **rename global registration name** grammar.

MaxL Support in Java API

Essbase clients developed in Java can execute MaxL statements to facilitate Oracle Essbase administrative tasks.

Calculation with Varying Attributes

These calculation functions and command are added to improve calculation of databases that use varying attributes:

- @ISMBRWITHATTR
- @ISATTRIBUTE
- SET SCAPERSPECTIVE

@WITHATTR is enhanced to allow varying attributes to be specified in the function argument.

Essbase API

Subtopics

- [New C Main API Functions](#)
- [New C Main API Structures](#)
- [New C Outline API Functions](#)
- [New Visual Basic Outline API Functions](#)

New C Main API Functions

- EssAsyncBuildDim
- EssAsyncImport
- EssAsyncImportASO
- EssCancelAsyncProc
- EssCloseAsyncProc
- EssGetAsyncProcLog
- EssGetAsyncProcState
- EssPerformAllocationAso

- EssPerformCustomCalcASO
- EssUpdateEx
- EssUpdateUtf8Ex
- EssUpdateFileEx
- EssUpdateFileUtf8Ex
- EssUpdateFileASOEx
- EssUpdateFileASOUtf8Ex

New C Main API Structures

- ESS_BLDDL_STATE_T
- ESS_PERF_ALLOC_T
- ESS_PERF_CUSTCALC_T

New C Outline API Functions

- EssOtlSetAliasTableLanguage
- EssOtlGetAliasTableLanguages
- EssOtlClearAliasTableLanguages
- EssOtlGetAltHierarchyEnabled
- EssOtlSetAltHierarchyEnabled
- EssOtlGetHierarchyType
- EssOtlSetHierarchyType
- EssOtlOpenOutlineEx
- EssOtlWriteOutlineEx
- EssOtlQueryMembersExArray
- EssOtlGetMemberInfoArray

New Visual Basic Outline API Functions

- EsbOtlSetAliasTableLanguage
- EsbOtlGetAliasTableLanguage
- EsbOtlClearAliasTableLanguages

New Calculation Functions

- @BETWEEN
- @EQUAL

- @EXPAND
- @ISATTRIBUTE
- @ISMBRWITHATTR
- @LIKE
- @MBRCOMPARE
- @MBRPARENT
- @NOTEQUAL
- @RETURN
- @XWRITE

@WITHATTR is enhanced to allow varying attributes to be specified in the function argument.

New Calculation Commands

SET SCAPERSPECTIVE

New `essbase.cfg` Configuration Settings

- AGENTLEASEEXPIRATIONTIME
- AGENTLEASEMAXRETRYCOUNT
- AGENTLEASERENEWALTIME
- APSRESOLVER
- ASOSAMPLESIZEPERCENT
- DIMBUILDERRORLIMIT
- FAILOVERMODE
- FILELOCKINGMODE
- NETSSLHANDSHAKETIMEOUT
- PERSISTUSERATLOGIN
- PRELOADUDANAMESPACE
- RTDEPCALCOPTIMIZE
- SERVERLEASEEXPIRATIONTIME
- SERVERLEASEMAXRETRYCOUNT
- SERVERLEASERENEWALTIME
- XOLAPSCHEMAVERIFICATION
- XOLAPMAXNUMCONNECTION
- XOLAPSQLIDLEPERIOD
- XOLAPENABLEHEURISTICS

MaxL Statements and Grammar Changes

Subtopics

- [New MaxL Statements](#)
- [Changed MaxL Grammar](#)

New MaxL Statements

- New statement **execute allocation** (aggregate storage only)
- New statement **execute calculation** (aggregate storage version)
- New statement **export outline**

Changed MaxL Grammar

- Changes to **alter user** and **alter group** statements to enable revoking filters.
- Changes to **create group** statement to enable creating external groups.
- Changes to **deploy** statement: the `odbc_dsn` keyword enables custom ODBC DSN names, and enables you to specify a Oracle Call Interface (OCI) connect identifier
- Changes to **drop user** and **drop group** statements to enable deletion from `essbase.sec` only.
- Changes to **USER-NAME** and **GROUP-NAME** terminals. All user and group names can be of the form *name*, *name@provider*, or **with identity** *STRING*.

COPYRIGHT NOTICE

Essbase New Features, 11.1.2.2.100

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Authors: EPM Information Development Team

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

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 is software or related documentation that 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 America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware 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 that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software or hardware 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.