What's New in Oracle Planning and Budgeting Cloud

July 2016 Release (16.07)

ORACLE

TABLE OF CONTENTS

ORACLE PLANNING AND BUDGETING CLOUD

This guide outlines what you need to know about new or improved functionality in Oracle Planning and Budgeting Cloud. Each section includes a brief description of the feature, the steps to take to either enable or begin using the feature, tips or considerations to keep in mind, and the resources available to help you.

SIMPLIFIED INTERFACE

Here is a summary of the key enhancements being introduced in the Simplified Interface.

DASHBOARDS

Dashboards are now much more powerful and easy to create.

Administrators design dashboards to provide an overview to key information that drives decisions. Then, using the dashboard, users can chart and even change key business data. For example, they can change a driver such as Volume in a form that's in a dashboard and immediately see its impact in other forms and charts:

In a dashboard you can include:

- Up to six forms
- Up to nine charts that dynamically depict data. Use chart types such as Area, Bar, Bubble, Column, Doughnut, Funnel, Gauge, Scatter, Radar, and so on.

- A new chart type called a *Tile* displays a specific value from the cube, for example, the June Sales target for a particular product.
- A new chart type called *Gauge* works much like a gas gauge: it shows whether a data value falls within an acceptable range or not. For example:
- A dynamic link to an external web site.
- Explanations of data called Commentary.

Dashboards and composite forms now support global POV bars, so that the local POVs that are common are combined in the global POV bar to avoid repeating them in each object. User variables are supported in both local and global POVs.

You have a great deal of flexibility in designing the dashboard. For example:

- You can choose to display objects vertically or horizontally.
- Each object can be a different size.
- You can set an object's size as Fixed or Flexible.
- The design can be asymmetric.

To create a dashboard:

1. Click or tap 

on the Home page, and then **Create**.
2. Name it.
3. From the design palette on the left, drag and drop objects onto the dashboard canvas.

As you design the dashboard, you can switch between using the design pallet and runtime mode, so you can see exactly what the dashboard user will see. You can also assign access to dashboards for users and groups.

When planners use a dashboard (referred to as *runtime*), they can change and save data, run rules, and so on. When they save data, it's saved in all the objects in a dashboard. They can set many aspects of the object, such as the type of chart displayed, the dashboard's title, and so on. A hover toolbar in the upper right corner is available for each object type, providing such options as Actions, Save, Refresh, Settings, and Maximize.

For more inform on designing dashboards, see "Designing Dashboards" in A*dministering Planning for Oracle Planning and Budgeting Cloud*. For information on using dashboards, see the chapter "Analyzing Data Using Dashboards" in *Working with Planning for Oracle Planning and Budgeting Cloud*.

HOME PAGE SHORTCUTS

The home page displays system announcements, lists recent activity, and provides shortcuts to open tasks and user favorites. Now, users can add recently used items and save their frequently used actions as favorites.

HIERARCHICAL VIEWS

With this release, forms, task lists, and reports can be viewed in either list view or hierarchical view.

RIGHT-CLICK MENUS

Within forms and input grids, cell level action menus such as comments and document attachments can now be accessed using a right-click menu.

NAVIGATION FLOWS

You can now create customizable navigation flows for groups of users. Within each navigation flow, you can create functional flows that can contain vertical and horizontal tabs with group specific dashboards, forms, tasks, reports, and approvals.

To work with navigation flows, in Settings, click or tap  .

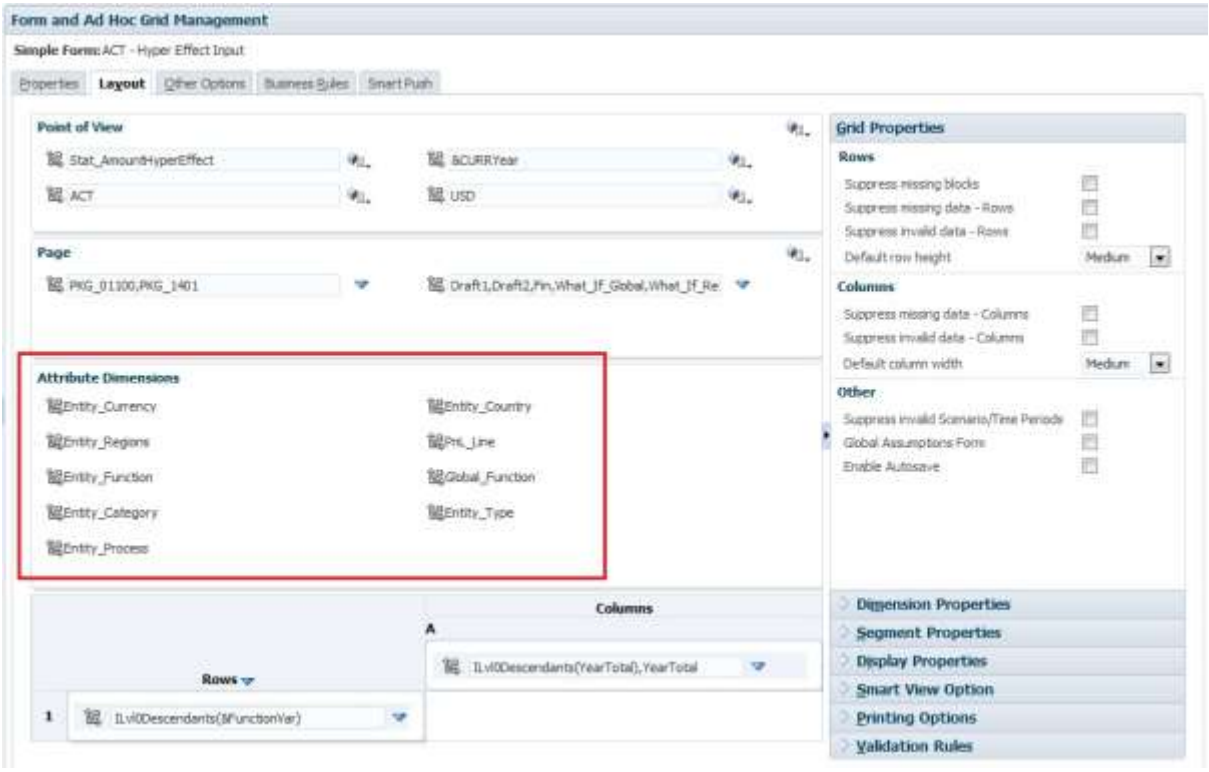
See “Designing Custom Navigation Flows” in *Administering Planning for Oracle Planning and Budgeting Cloud*.

SUPPORT FOR ATTRIBUTE DIMENSIONS

You can now use an attribute dimension as a dimension, as a filter in forms and reports, and within ad hoc grids. Using attribute dimensions enables administrators and end users to perform tasks such as:

- Filtering data using attribute members, such as by products with a certain color
- Performing cross-dimensional rollups across attribute members
- Reporting and analysis with attribute dimension members using Smart View, or financial reports
- Using attribute dimensions in dynamic user variables

Attribute dimensions are optional and are listed separately on the Layout tab of the Form Designer. Drag the Attribute dimension to a Point of View or to a row or column to add it to the form grid.



You can also use Attribute dimension members in Smart View where they can be used to filter data, to zoom, and to pivot.

See “Attribute Dimensions” in *Administering Planning for Oracle Planning and Budgeting Cloud*.

ARTIFACT LABELS

Administrators can use artifact labels to customize artifact names and descriptions based on the user's browser locale. For example, if you create a form, you can have the form name displayed in the language of the user.

The Artifact Strings page displays an Excel-style spreadsheet grid that is filtered by artifact and property type. Click Add Language to display a new language-specific column in which you can enter text for each artifact property (Name, Description, and so on). You can also export the artifact strings to edit them, and then import them.

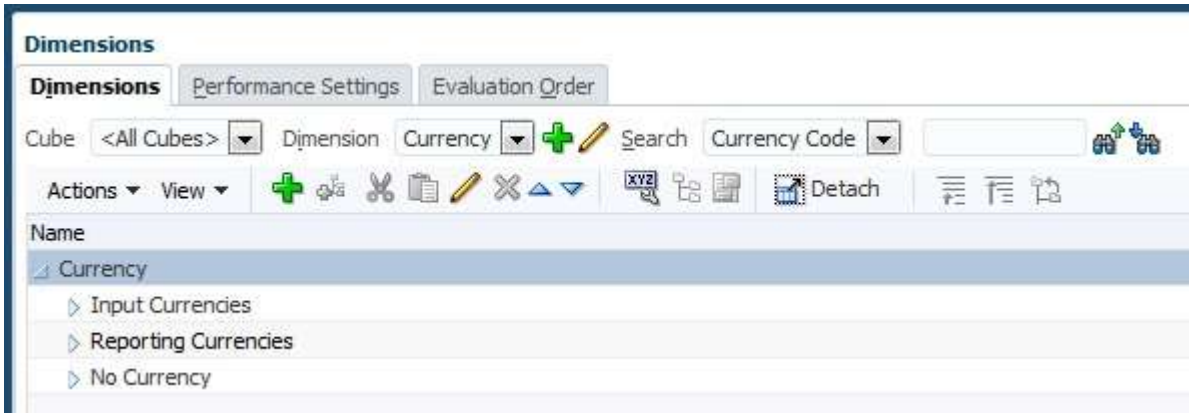
To work with artifact labels, in Settings, click or tap .

See “Customizing Artifact Strings by Language” in *Administering Planning for Oracle Planning and Budgeting Cloud*.

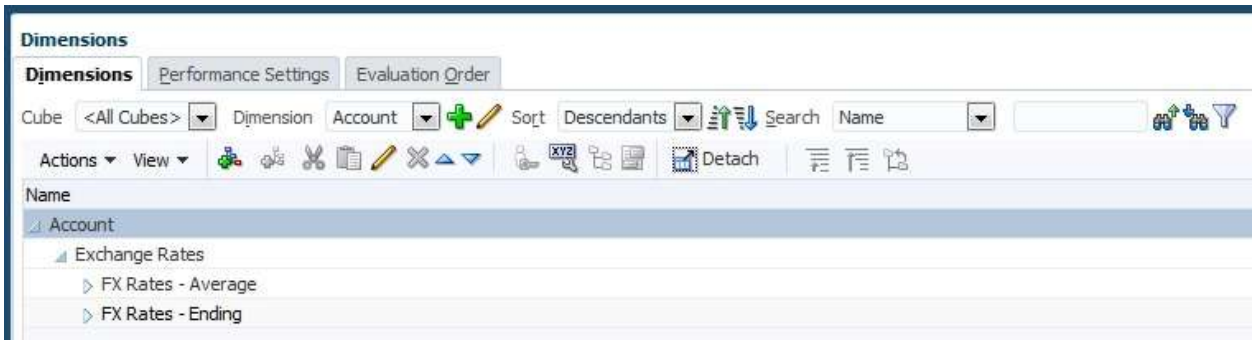
SIMPLIFIED CURRENCY

For new applications, administrators can optionally choose a simplified multicurrency option during application creation. Using simplified currency avoids the use of the Hsp_Rates dimension and adds a Currency dimension with exchange rates stored in the Account dimension.

The Currency dimension contains members for all your foreign currencies under Input Currencies and Reporting Currencies.



The Account dimension contains Exchange Rates that stores the ending and average rates used to convert foreign currencies into the base currency. These rates are in FX Rates-Average and FX Rates-Ending.

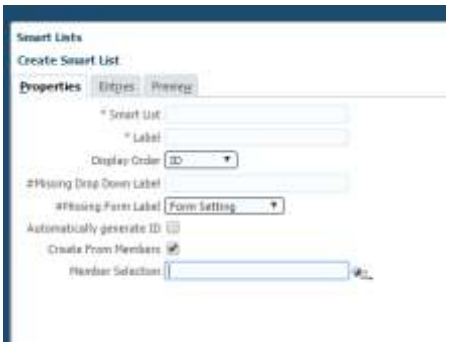


Exchange rate data is entered to an exchange rate form called Exchange Rates to Default Currency. Users with view permissions can view the rates for the currencies on this form.

See “About Simplified Multicurrency” in *Administering Planning for Oracle Planning and Budgeting Cloud*.

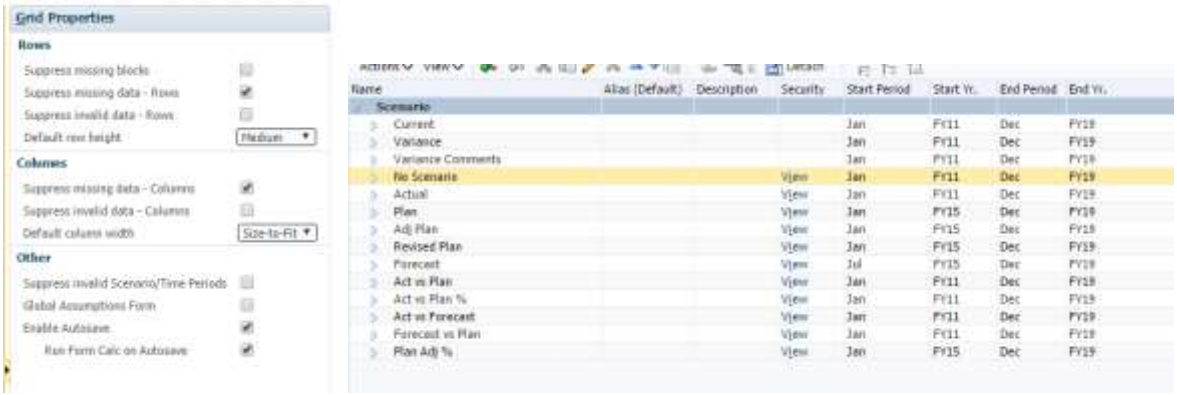
SMART LIST DYNAMIC SYNC WITH DIMENSION HIERARCHIES

You can now create smart lists based on dimension hierarchies. This dynamically updates smart list values based on member updates. These smart lists are tied to member names using member alias representation by smart list ID. This can be stored in separate alias table name, for example, SLAliases. User security for the dimension is honored for the smart lists created from that dimension’s hierarchy.



SYNC SCENARIO TIME RANGE WITH FORM DISPLAY

Form grid display can be tied to the start and end period for the respective scenarios on display. When the new grid level form property is checked, the time periods outside of the scenario time period range are suppressed.



CLEAR CUBE

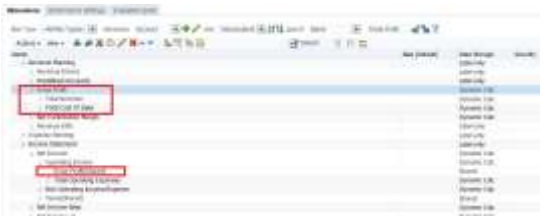
A new action menu in the Console allows customers to clear specific areas within both input and reporting cubes. For ASO reporting cubes, a partial clear allows the use of MDX expressions to identify areas of the cube to clear. Clear Cube jobs can be scheduled as jobs using the job scheduler.

DATA MAPS FROM ANY DENSE DIMENSION

When defining data maps, the designer can pick any of the dense dimensions with Smart List associated to be mapped against a dimension in the target cube.

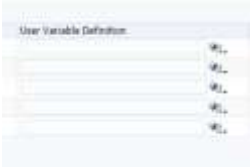
DRILL FROM SHARED MEMBERS ON A FORM

Users can now drill on shared members to navigate to the children of the base member. For example, if the member Gross Profit has children, and Gross Profit is also a shared member in another hierarchy, the user can drill into Gross Profit(shared) when used in a form to see children of Gross Profit.



LIMIT USER VARIABLES BY MEMBER FUNCTIONS

User Variables can now be bound to apply only for specific hierarchies within the dimension. The hierarchies can be selected using member selection in the user variable set up with Variables. Usage of user variables will then be restricted to be within the specified hierarchies. This is particularly useful when using dynamic user variables in data forms.



SPECIFY RANGES WITHIN SUBSTITUTION VARIABLES

You can now specify range values when defining substitution variables, for example, FY16:FY18

HIDE SAVE CONFIRMATION MESSAGES ON FORM SAVE

Form designers can now prevent the form save confirmation message from being displayed to users by specifying an option in form design.

DUPLICATE ALIASES

The usability and readability of forms and ad hoc grids is increased with support for duplicate aliases for members from different dimensions and for members within dimensions.

The guidelines are:

- Duplicate aliases are supported across dimensions and within dimensions.
- Aliases can have the same name as a member.
- When referencing an alias that resolves to multiple members in free-form mode, an error is returned noting that the alias cannot be resolved to a single member. You can do this either by using the Member Selector to select the correct alias name or by hand-typing the qualified name. Using the parent member name as a qualifier should be sufficient in most cases, given that only duplicate alias names are supported, not duplicate member names.

Note that member names are still required to be unique so that they can be used in rules and form designs to avoid name collisions.

INVALID INTERSECTIONS REPORTS

When using the valid intersections feature, users can input data only in specific areas of the application as defined in the valid intersections rules.

With Invalid Intersection Reports, administrators can review specific areas of the application where data exists that does not conform to the valid intersection rules and can be considered invalid. Data may reside in invalid intersections due to actions in business rules, data load, or data maps. Administrators can use this report to clear data from invalid intersections.

PLANNING STANDARD INTERFACE (WORKSPACE)

There are no new features being introduced specifically for the Standard (Workspace) Interface.

SMART VIEW FOR OFFICE

IMPORT METADATA FOR MICROSOFT WORD

In this release, Import Metadata functionality is extended to Microsoft Word.

The Import Metadata command gives you the ability to reuse Oracle Smart View for Office content in Word documents, content that has already been formatted and customized for Word. You can use the copy and paste commands in Word to retain the customized formatting of a Smart View object, such as a table or graph, and the use Import Metadata to bring the metadata into the copied object.

For example, suppose you have a Word document containing a highly-formatted table with Smart View data points, and you would like to reuse the table in a different Word document. Instead of recreating the table from scratch, you can reuse the work that is already available with the new Import Metadata command on the Smart View ribbon.

Using the Word copy and paste commands to copy the table from one Word document to another, only the data is copied; the metadata is not copied. After the copy and paste, use the Import Metadata command on the Smart View ribbon to import the metadata from the original document into the new document. You can copy and paste within the same Word document or to a different Word document.

See “Importing Metadata into Copied Word Documents” in the *Oracle Smart View for Office User's Guide*

INSERT ATTRIBUTES IN AD HOC GRIDS

You can now quickly add attribute dimensions to an ad hoc grid at any time during the ad hoc session.

To launch the Insert Attributes dialog box, where you can select the attribute members to add to the POV, click **Insert Attributes** on the Planning Ad Hoc ribbon.

When the attribute dimensions are in the POV, you can pivot or drag them to the grid as required. Once on the grid, you can use the Remove Only command to remove any unwanted attribute dimension or members. If the grid already contains some, but not all, attribute dimensions in the database, you can click Insert Attributes to select any of the remaining attributes to the POV.

Note that you can still use Member Selection to add specific attribute dimensions as required.

See “Inserting Attribute Dimensions on the Sheet” in the *Oracle Smart View for Office User's Guide*.

ATTRIBUTE AND TIME PERIOD DIMENSIONS IN PLANNING ADMIN EXTENSION

In the Planning Admin Extension, you can now work with attribute dimensions and the Time Period dimension. Just as with regular dimensions, you can use the Planning Admin Extension in the Smart View application to quickly import and edit attribute and time dimension application metadata.

See *Administering Planning for Oracle Planning and Budgeting Cloud*.

DATA MANAGEMENT

SERVICE INSTANCE INTEGRATIONS

Data Management can now be used as a primary gateway to move data from one service instance to another. A service instance is a self-contained unit often containing the web server and the database application (Oracle Planning and Budgeting Cloud application).

To use this feature, you open the parent service instance, and then use the target application option to register a child service instance. In this case, you use the "Cloud" option as the target application type. Once the child service instance is registered, it can be used as a source or target application.

NEW ALL DATA TYPES DATA LOAD METHOD

Data Management added a new All Data Types Data Load method which enables you to load numbers, text, Smartlists, and date.

See "All Data Types Data Loads" in the Administering Data Management for Oracle Enterprise Performance Management Cloud for detailed information.

CALCULATION MANAGER

REDESIGNED METADATA LOOP COMPONENT WITHIN A TEMPLATE

Metadata loop components enable you to assign a value to multiple members. Starting in this update, you can use either a function (Metadata) or a start and end value (Fixed). Previously, you could only use four functions in the metadata loop. In this release, you can use most member set functions.

To create a metadata loop component, drag its icon and drop it into the flow chart of a custom defined template.

See "Working With Metadata Loop Components" in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.*

CREATE A TEMPLATE USING A SCRIPT

When you create a template using a script, you can:

- Deploy the script template to Planning.
- Launch the template in Planning, where the steps will display allowing you to enter data at runtime.
- Use run-time functions and design-time prompts.

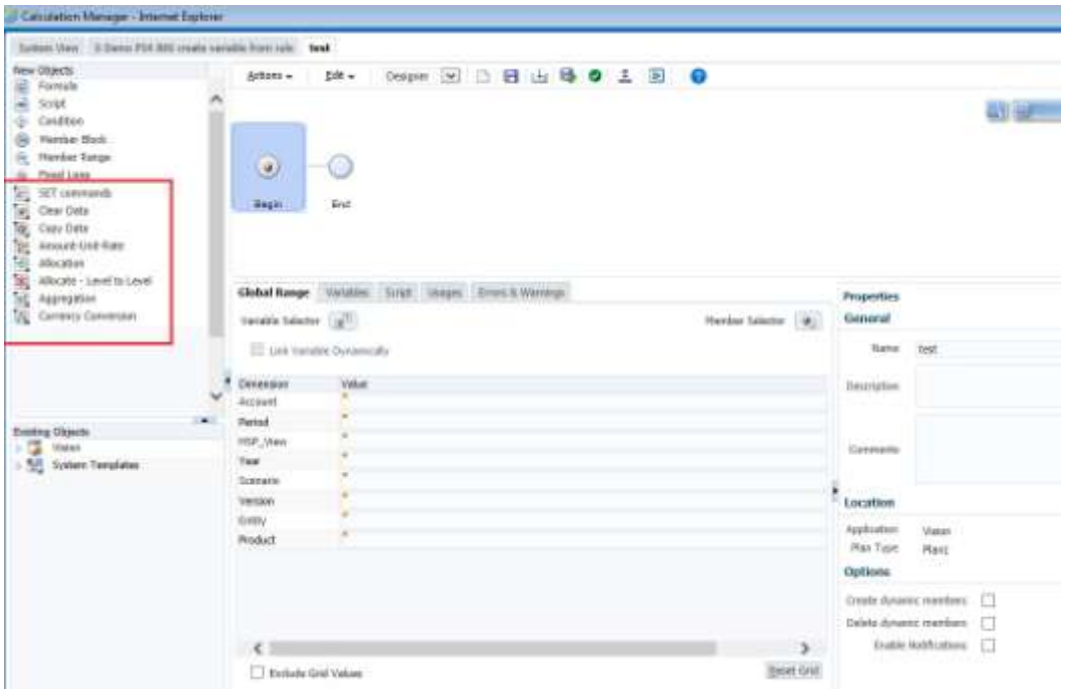
See "Creating a Template Using a Script" in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.*

NEW AND REVISED SYSTEM TEMPLATES

Revised System Templates include:

- Amount Unit Rates
- Allocations (This template replaces both Allocate Simple and Allocate Simple Exclude, and includes all of the functionality in both these templates.)
- Copy Data

System Templates are now displayed under New Objects:

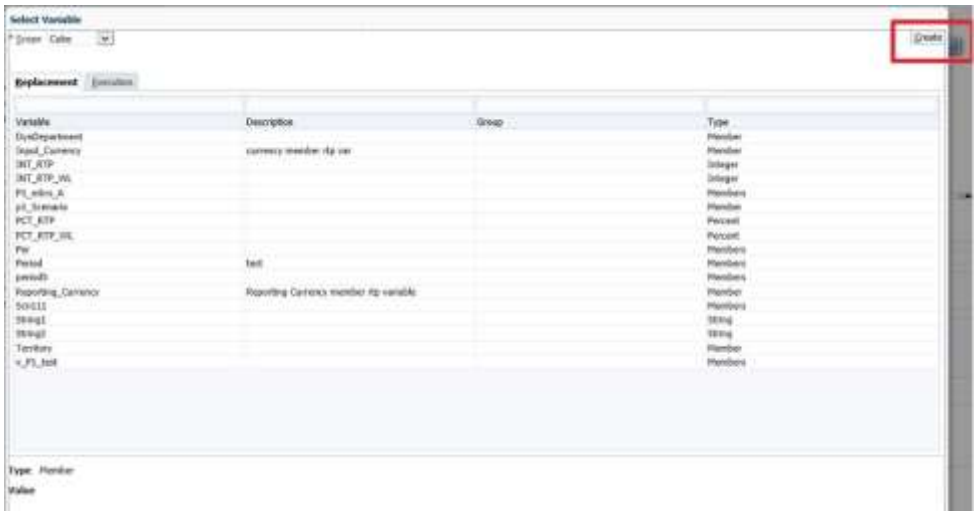


STEP VALIDATION ERRORS AND WARNINGS

You can now add a warning or an error to a step using validation conditions. Errors prevent the next step. Warnings allow the next step after you click OK on the warning message. You can use a design-time prompt or function on the validation condition. This allows you to use functions on design-time prompts without having to create non-promptable design-time prompts.

CREATE A NEW VARIABLE WHEREVER THE VARIABLE SELECTOR IS AVAILABLE

You can now create a variable anywhere that the Variable Selector is available. For example, if you are in a script rule, click the Variable icon, and then click **Create**.



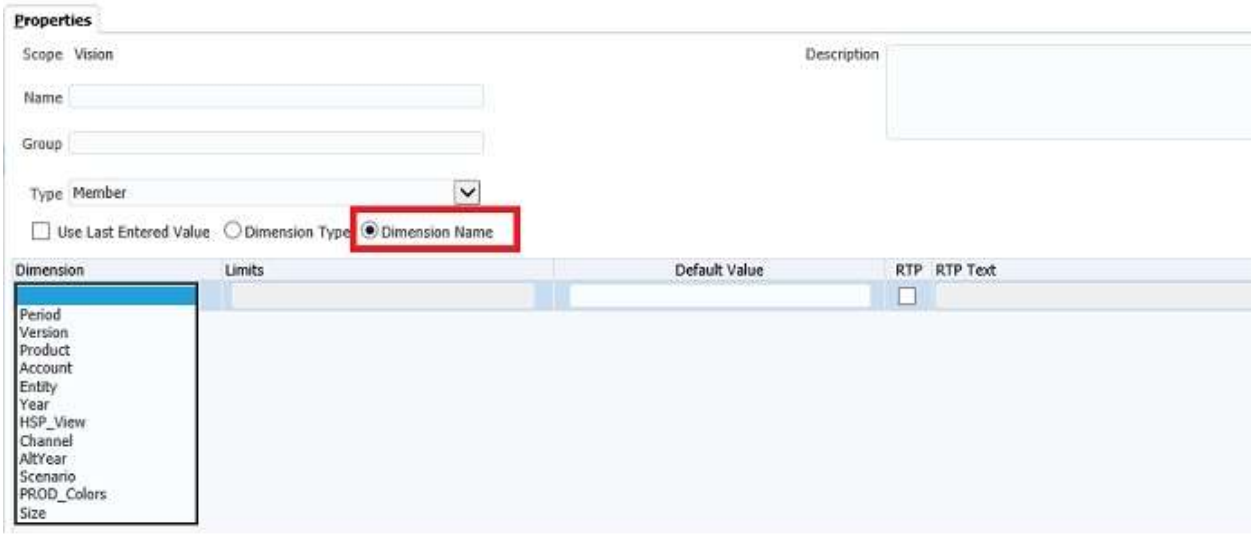
After you click **Create**, the **Create Variable** dialog box is displayed in the rule, and you can create the variable without having to leave the rule.

See “Working with Variables” in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.*

MEMBER TYPE VARIABLE INCLUDES ATTRIBUTE DIMENSION NAMES(S)

In the Variable Designer, for member type variables using “dimension name,” the list of dimensions now contains Attribute dimension names.

In the following example, “PROD_Colors” and “Size” are Attribute dimension names:



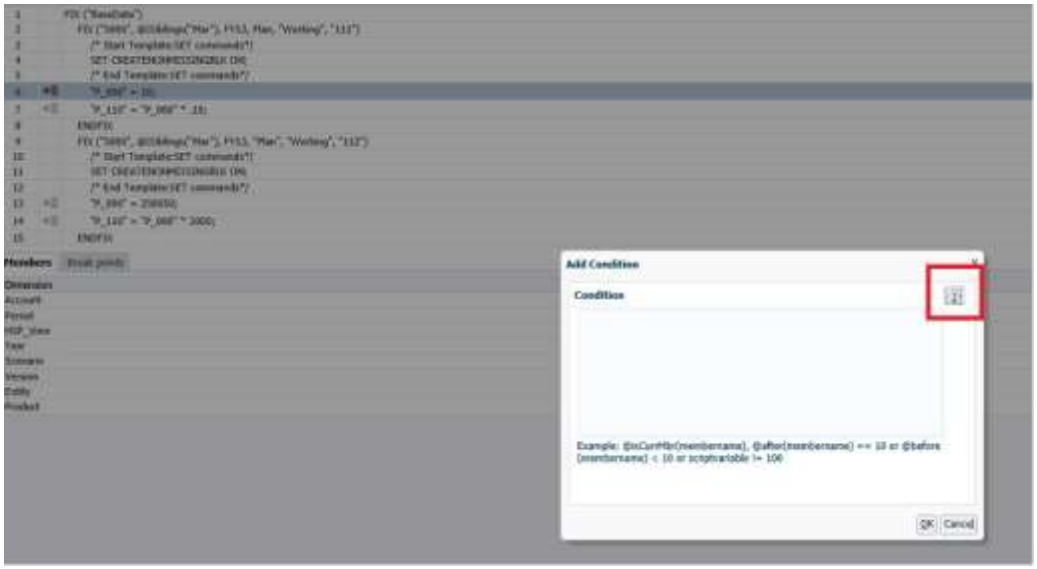
See “Working with Variables” in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.*

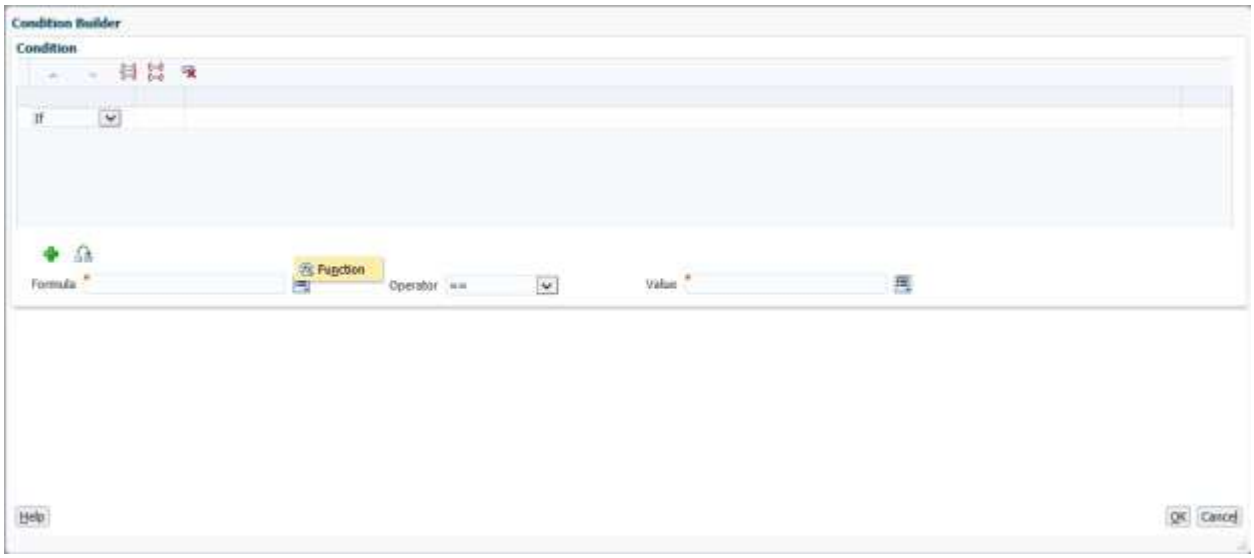
CONDITION BUILDER FOR DEBUGGING

When you are debugging business rules, a Condition Builder is now available to help you build conditions.

To access the Condition Builder:

1. Add a breakpoint to a line, then right-click, and then select **Add Condition**.
2. In the **Add Condition** dialog box, click the **Condition** icon.





See “Analyzing and Debugging Business Rules” in *Designing with Calculation Manager for Oracle Enterprise Performance Management Cloud.*

LOG MESSAGES AVAILABLE AFTER LOG OFF

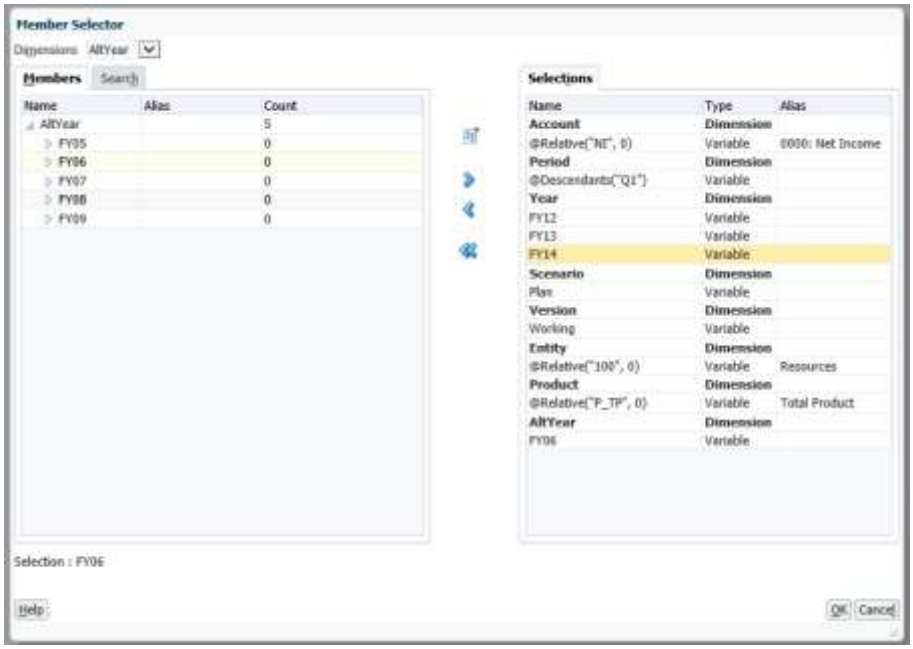
You can now see the Log Message tab even if you are logged off of Calculation Manager. For example, assume that you launch a rule in Calculation Manager, the rules takes a long time to run, and Calculation Manager times out and closes. When you open the rule again, the Log Message tab will still be displayed until you close the rule.

PARTIAL CLEAR OF DATABASE PROPERTIES

You can use the Member Selector dialog box to create MDX syntax and validation before running a partial clear.

To clear partial data using an MDX expression:

1. In Database Properties, expand an ASO application.
2. Right-click an ASO cube, then select **Clear**, and then **Partial Data**.
3. In the **Clear Partial Data dialog** box, click the **Member Selector** icon.
4. In the **Member Selector** dialog box, select a member or members , or use a function from each dimension, and then click **OK**.



The MDX expression is created. For example:

```
Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin(Crossjoin({[NI
].Levels(0).Members},{except(DESCENDANTS([Q1]),{[Q1]})}),{[FY12],[FY13],[F
Y14]}),{[Plan]}),{[Working]}),{[100].Levels(0).Members}),{[P_TP].Levels(0)
.Members}),{[FY06]}
```

Note: The MDX expression is validated when you click **OK** to clear data. If you modify the MDX expression directly, it will validate before running.

NEW DESIGN-TIME PROMPT FUNCTIONS

The following new design-time prompt functions are available:

@AVAILDIMCOUNT - Returns the number of available dimensions.

@DEPENDENCY - “Inclusive” returns member(s) from Input 1 for which Input 2 has member(s) specified from the same dimensions. “Exclusive” returns members from Input 1 for which Input 2 has no specified members in the same dimensions.

@DIMATTRIBUTE - Returns the attribute name if the specified attribute is associated with a dimension.

@DIMNAME - Returns the name of a dimension if it is valid for the database.

@DIMUDA - Returns the UDA name if the specified UDA is valid for the dimension.

@EVALUATE - Returns the result of an expression.

@FINDFIRST - Finds the first substring of a string that matches the given regular expression.

@FINDLAST - Finds the last substring of a string that matches the given regular expression.

@GETDATA - Returns the value of the slice.

@INTEGER - Returns an integer.

@ISDATAMISSING - Returns true if the value of the slice is missing.

@ISANDBOXED - Determines if the current application is sandboxed.

@ISVARIABLE - Determines if the argument is a variable.

@MATCHES - Returns “true” if the first substring of a string matches the given regular expression.

@MEMBERGENERATION - Returns the generation number of a member.

@MEMBERLEVEL - Returns the level number of a member.

@MSGFORMAT - Takes a set of objects, formats them, and then inserts the formatted strings into the pattern at the appropriate places.

@OPENDIMCOUNT - Returns the number of dimensions for which a member was not specified.

@VALUEDIMCOUNT - Returns the number of dimensions for which a member was specified.

@TOMDX - Returns an MDX expression.

NEW DESIGN-TIME PROMPT TYPES

The following new design-time prompt types are available:

- Percent
- Integer

- StringAsNumber
- DateAsNumber
- Smart List
- UDA

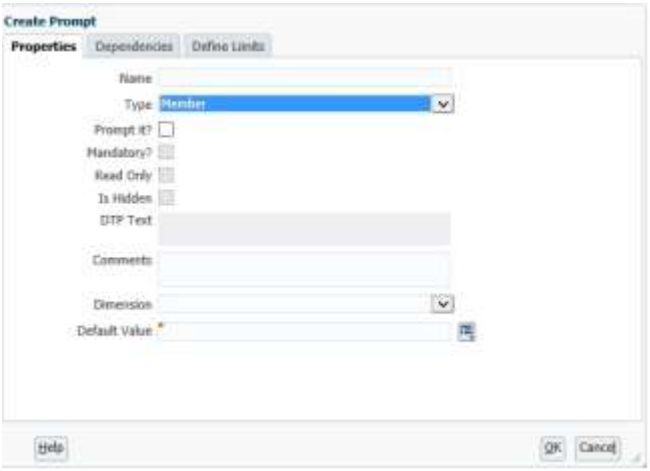
NEW DESIGN-TIME PROMPT GRID

A new grid is available when creating design-time prompts for custom-defined templates.

To create a design-time prompt:

1. Open an existing template or create a new template in the **Template Designer**.
2. In the **Design Time Prompt** tab, click  .
3. If you already have design-time prompts defined for the template, select where to insert the new design-time prompt.
4. In the **Create Prompt** dialog box, in the **Properties** tab, enter the information about the design-time prompt.

Note: The tabs that display in the **Create Prompt** dialog box are based on the type of design-time prompt, and the rows that are displayed are based on what is input.



NEW CUSTOM DEFINED FUNCTIONS

The following new custom defined functions are available:

New Bitwise Functions

@CalcMgrBitAnd - Performs a bitwise AND operation, which compares each bit of the first operand to the corresponding bit of the second operand. If both bits are 1, the corresponding result bit is set to 1; otherwise, the corresponding result bit is set to 0.

@CalcMgrBitOR - Performs a bitwise OR operation, which compares each bit of the first operand to the corresponding bit of the second operand. If either bit is 1, the corresponding result bit is set to 1; otherwise, the corresponding result bit is set to 0.
@CalcMgrBitExOR - Performs an exclusive bitwise OR operation, which compares each bit of the first operand to the corresponding bit of the second operand. If either bit is 1, the corresponding result bit is set to 1; otherwise, the corresponding result bit is set to 0.
@CalcMgrBitExBoolOR - Performs an exclusive boolean bitwise OR operation.
@CalcMgrBitCompliment - Performs a unary bitwise complement, which reverses each bit.
@CalcMgrBitShiftLeft - Performs a signed left shift.
@CalcMgrBitShiftRight - Performs a signed right shift.
@CalcMgrBitUnsignedShiftRight - Performs an unsigned right shift.

New Counter Functions

@CalcMgrCounterClearAll - Removes all keys and values from the counter
@CalcMgrCounterClearKey - Removes the value from the counter associated with the key
@CalcMgrCounterDecrement - Decrements the value in the counter based on the key. If the key is not found, a value of zero is set for the key
@CalcMgrCounterDecrementKey - Decrements the value in the counter based on the key. If the key is not found, a value of zero is set for the key
@CalcMgrCounterGetKeyNumber - Returns the text found in the counter based on the key. If the key is not found, missing value is returned.
@CalcMgrCounterGetKeyText - Returns the text found in the counter based on the key. If the key is not found, missing value is returned.
@CalcMgrCounterGetNumber - Returns the number from the counter specified by the key. If the key is not found or the value is not a number, missing value is returned.
@CalcMgrCounterGetText - Returns the text found in the counter based on the key. If the key is not found, missing value is returned.
@CalcMgrCounterIncrement - Increment the value in the counter specified by the key
@CalcMgrCounterIncrementKey - Increments the value in the counter based on the key. If the key is not found, a value of zero is set for the key.

New Date/Time Functions

@CalcMgrExcelToDate - Converts an Excel date to YYYYMMDD format.
@CalcMgrExcelToDateTime - Converts an Excel date to YYYYMMDDHHMMSS format.
@CalcMgrGetStringFormattedDateTime - Converts the date defined by format to date in the YYYYMMddHHmmss format. For example: `@CalcMgrGetFormattedDate(12302014, MMddyyyyHHmm)` returns `201412301430`.
@CalcMgrDateToExcel - Converts a date in YYYYMMDD format to an Excel date
@CalcMgrDateTimeToExcel - Converts a date in YYYYMMDDHHMMSS format to an Excel date
@CalcMgrRollDay - Roll the day up or down to the date which is in the YYYYMMDD format
@CalcMgrRollDate - Adds or subtracts (up or down) a single unit of time on the given date field without changing larger fields.For example,
`@CalcMgrRollDate(19960131,&quot;month&quot;,@_true)` results in the date of `19960229`, and `@CalcMgrRollDate(19960131, "day";,@_true)` results in the date of `19960101`. Possible values of date_part are: day, month, week and year.
@CalcMgrRollMonth - Roll the month up or down to the date which is in the YYYYMMDD format.

@CalcMgrRollYear - Roll the year up or down to the date which is in the YYYYMMDD format.

New Financial Functions

@CalcMgrExcelACCRINT - Returns the accrued interest for a security that pays periodic interest
@CalcMgrExcelACCRINTM - Returns the accrued interest for a security that pays interest at maturity
@CalcMgrExcelAMORDEGRC - Returns the depreciation for each accounting period by using a depreciation coefficient
@CalcMgrExcelAMORLINC - Returns the depreciation for each accounting period
@CalcMgrExcelCOUPDAYBS - Returns the number of days from the beginning of the coupon period to the settlement date
@CalcMgrExcelCOUPDAYS - Returns the number of days in the coupon period that contains the settlement date
@CalcMgrExcelCOUPDAYSNC - Returns the number of days from the settlement date to the next coupon date
@CalcMgrExcelCOUPNCD - Returns a number that represents the next coupon date after the settlement date
@CalcMgrExcelCOUPNUM - Returns the number of coupons payable between the settlement date and maturity date, rounded up to the nearest whole coupon
@CalcMgrExcelCOUPPCD - Returns a number that represents the previous coupon date before the settlement date
@CalcMgrExcelCUMIPMT - Returns the cumulative interest paid on a loan between start_period and end_period
@CalcMgrExcelCUMPRINC - Returns the cumulative principal paid on a loan between the start period and the end period
@CalcMgrExcelDB - Returns the depreciation of an asset for a specified period using the fixed-declining balance method
@CalcMgrExcelDDB - Returns the depreciation of an asset for a specified period using the double-declining balance method or some other method you specify
@CalcMgrExcelDISC - Returns the discount rate for a security
@CalcMgrExcelDOLLARDE - Converts a dollar price expressed as an integer part and a fraction part, such as 1.02, into a dollar price expressed as a decimal number. Fractional dollar numbers are sometimes used for security prices.
@CalcMgrExcelDOLLARFR - Converts a dollar price, expressed as a decimal number, into a dollar price, expressed as a fraction
@CalcMgrExcelDURATION - Returns the annual duration of a security with periodic interest payments
@CalcMgrExcelEFFECT - Returns the effective annual interest rate
@CalcMgrExcelFV - Returns the future value of an investment
@CalcMgrExcelFVSCHEDULE - Returns the future value of an initial principal after applying a series of compound interest rates
@CalcMgrExcelINTRATE - Returns the interest rate for a fully invested security
@CalcMgrExcelIPMT - Returns the interest payment for a given period for an investment based on periodic, constant payments and a constant interest rate
@CalcMgrExcelIRR - Returns the internal rate of return for a series of cash flows
@CalcMgrExcelISPMT - Calculates the interest paid during a specific period of an investment
@CalcMgrExcelMDURATION - Returns the Macauley modified duration for a security with an assumed par value of $100

@CalcMgrExcelMIRR - Returns the internal rate of return where positive and negative cash flows are financed at different rates
@CalcMgrExcelNOMINAL - Returns the annual nominal interest rate
@CalcMgrExcelNPER - Returns the number of periods for an investment
@CalcMgrExcelNPV - Returns the net present value of an investment based on a series of periodic cash flows and a discount rate
@CalcMgrExcelPMT - Returns the periodic payment for an annuity
@CalcMgrExcelPPMT - Returns the payment on the principal for a given period for an investment based on periodic, constant payments and a constant interest rate
@CalcMgrExcelPRICE - Returns the price per $100 face value of a security that pays periodic interest
@CalcMgrExcelPRICEDISC - Returns the price per $100 face value of a discounted security
@CalcMgrExcelPRICEMAT - Returns the price per $100 face value of a security that pays interest at maturity
@CalcMgrExcelPV - Returns the present value of an investment
@CalcMgrExcelRATE - Returns the interest rate per period of an annuity
@CalcMgrExcelRECEIVED - Returns the amount received at maturity for a fully invested security
@CalcMgrExcelSLN - Returns the straight-line depreciation of an asset for one period
@CalcMgrExcelSYD - Returns the sum-of-years' digits depreciation of an asset for a specified period
@CalcMgrExcelTBILLEQ - Returns the bond-equivalent yield for a Treasury bill
@CalcMgrExcelTBILLPRICE - Returns the price per $100 face value for a Treasury bill
@CalcMgrExcelTBILLYIELD - Returns the yield for a Treasury bill
@CalcMgrExcelXIRR - Returns the internal rate of return for a schedule of cash flows that is not necessarily periodic
@CalcMgrExcelXNPV - Returns the net present value for a schedule of cash flows that is not necessarily periodic
@CalcMgrExcelYIELD - Returns the yield on a security that pays periodic interest
@CalcMgrExcelYIELDDISC - Returns the annual yield for a discounted security; for example, a Treasury bill
@CalcMgrExcelYIELDMAT - Returns the annual yield of a security that pays interest at maturity

New Math Functions

@CalcMgrExcelCEILING - Rounds a number up (away from zero) to the nearest integer or to the nearest multiple of significance
@CalcMgrExcelCOMBIN - Returns the number of combinations for a given number of objects
@CalcMgrExcelEVEN - Rounds a number up to the nearest even integer
@CalcMgrExcelFACT - Returns the factorial of a number
@CalcMgrExcelFACTDOUBLE - Returns the double factorial of a number
@CalcMgrExcelFLOOR - Rounds a number down, toward zero
@CalcMgrExcelGCD - Returns the greatest common divisor
@CalcMgrExcelLCM - Returns the least common multiple
@CalcMgrExcelMROUND - Rounds a number to a specified number of digits
@CalcMgrExcelMULTINOMIAL - Returns the multi-nominal of a set of numbers
@CalcMgrExcelODD - Rounds a number up to the nearest odd integer
@CalcMgrExcelPOWER - Returns the result of a number raised to a power
@CalcMgrExcelPRODUCT - Multiplies its arguments
@CalcMgrExcelROUNDDOWN - Rounds a number down, towards zero
@CalcMgrExcelROUNDUP - Rounds a number up, away from zero

@CalcMgrExcelSQRT - Returns a positive square root
@CalcMgrExcelSQRTPI - Returns the square root of (number * pi)
@CalcMgrExcelSUMSQ - Returns the sum of the squares of the arguments
@CalcMgrExcelSUMPRODUCT - Returns the sum of the products of corresponding array components

New Statistical Functions

@CalcMgrExcelAVEDEV - Returns the average of the absolute deviations of data points from their mean
@CalcMgrExcelDEVSQ - Returns the sum of squares of deviations
@CalcMgrExcelLARGE - Returns the nth highest number
@CalcMgrExcelMEDIAN - Returns the median of the given numbers
@CalcMgrExcelSMALL - Returns the nth smallest number
@CalcMgrExcelSTDEV- Estimates standard deviation based on a sample
@CalcMgrExcelVAR - Estimates variance based on a sample
@CalcMgrExcelVARP - Estimates variance based on the entire population

New String Functions

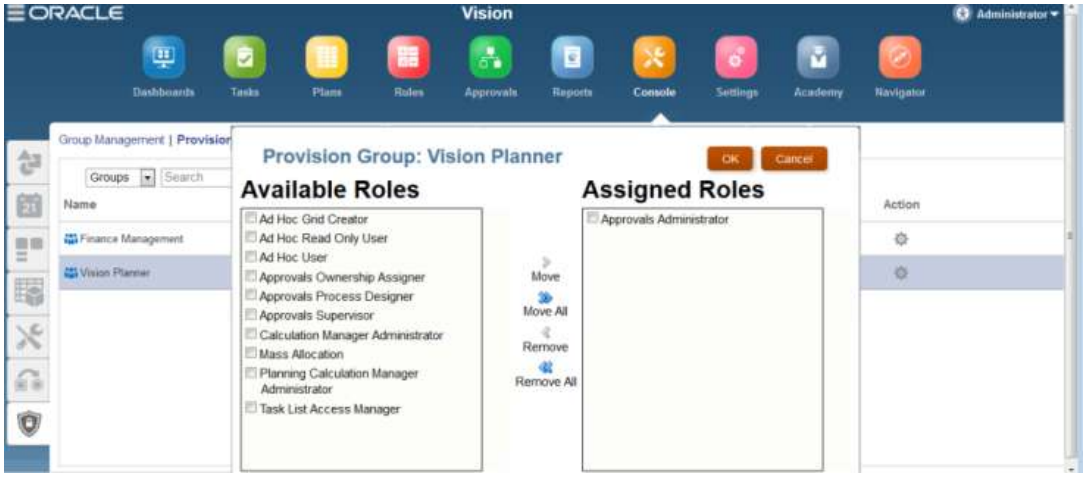
@CalcMgrFindFirst - Find the first substring of this string that matches the given regular expression.
@CalcMgrFindLast - Find the last substring of this string that matches the given regular expression.
@CalcMgrMatches - Returns true, if the first substring of this string that matches the given regular expression. For regular expression, see ”java.util.regex.Pattern” in the Java docs.
@CalcMgrMessageFormat - Creates a string with the given pattern and uses it to format the given arguments.
@CalcMgrStartsWith - Tests if this string starts with the specified prefix.

APPLICATION MANAGEMENT

APPLICATION ACCESS CONTROL

While the overall access rights granted to a user are controlled by the assigned identity domain role, Service Administrators can use the Access Control feature from the Console to assign additional application-level access by provisioning users and Native Directory groups with application-specific roles. For instance, a Planner in the service can now be assigned the Approvals Administrator role to enable the user to perform approvals-related activities.

To assign application access control, in Console, click or tap  .



See “Managing Application-Level Provisioning” in *Administering Access Control for Oracle Enterprise Performance Management Cloud*.

MIGRATION FUNCTIONALITY MOVED TO CONSOLE CARD

All migration functionality has been moved to the Console card, accessed from the Home page.

To perform lifecycle management activities, from the Console, click or tap  .

WEB SERVICES

NEW ORACLE CLOUD REST APIS FOR ORACLE PLANNING AND BUDGETING CLOUD

You can now use Planning REST APIs to import a data slice, export a data slice, and clear a data slice.

METADATA UPDATE AND CUBE REFRESH CONSIDERATIONS

After modifying application metadata, you must complete a cube refresh before:

- importing artifacts from snapshots
- the next scheduled maintenance window starts

The time required to complete a cube refresh operation depends on the metadata that was changed and the impact it has on the cube. For example, a refresh after updating a sparse block storage cube member may not take much time while a cube refresh after updating a dense block storage cube member or an aggregate storage cube member could take a long time. The quantity of Essbase data also impacts the cube refresh time. Oracle recommends that you evaluate these factors and schedule cube refresh to ensure that it completes before the application is backed up during the next maintenance window.

DISCLAIMER

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle Corporation.