Oracle® Hyperion Workforce Planning, Fusion Edition

Administrator's Guide

RELEASE 11.1.2.1
## Contents

**Documentation Accessibility** ........................................................... 7

**Chapter 1. Setting Up Workforce Planning** ............................................. 9

  - About Workforce Planning ................................................................. 9
  - Prerequisites .......................................................................................... 10
  - Business Model .................................................................................... 10
  - Creating and Initializing a Workforce Planning Application .................. 11
  - Using Performance Management Architect Application Administration ... 12
  - Using Classic Application Administration ........................................... 13
  - Loading Information into Workforce Planning ....................................... 14
  - About Loading Information Using DIM Adapter for Planning ................. 15
  - Logging On and Accessing Workforce Planning .................................... 15

**Chapter 2. Working with Workforce Planning** ........................................... 17

  - Working with Employees ....................................................................... 17
    - Transferring Employees ..................................................................... 17
    - Reconciling Transfers ...................................................................... 18
    - Increasing the Maximum Number of TBHs ........................................ 19
    - Converting TBHs to Hires ................................................................. 19
    - Changing Employee Status .............................................................. 20
    - Terminating Employees .................................................................... 20
  - Planning Department Expenses ............................................................ 20
  - Working with Accounts ........................................................................ 20
    - Spreading Data Using System Members ............................................ 20
    - About Salary Accounts ...................................................................... 21
  - Working with Data Forms ...................................................................... 22
  - Working with Smart Lists and Menus .................................................... 22
  - Working with Member Formulas ............................................................ 23
  - Working with Business Rules ............................................................... 23
  - Using Business Rules to Roll Up Higher Level Entities ......................... 24
  - Working with UDAs ........................................................................... 24
  - Managing Access Permissions .............................................................. 25
What's Next? ........................................................  25

**Chapter 3. Customizing Workforce Planning** ...............................................  27
About Customizing Your Application .........................................................  27
Adding a Performance Rating .................................................................  27
  With Performance Management Architect Applications ..........................  28
  With Classic Applications ....................................................................  29
Changing the Pay Rate for Maternity Status ..............................................  30
  With Performance Management Architect Applications ..........................  31
  With Classic Applications ....................................................................  31
Changing Bonus Frequency .......................................................................  32
  With Performance Management Architect Applications ..........................  32
  With Classic Applications ....................................................................  33
Adding a Salary Adjustment Type .............................................................  34
  With Performance Management Architect Applications ..........................  34
  With Classic Applications ....................................................................  40

**Appendix A. Workforce Planning Structure** .................................................  45
Predefined Data Forms ............................................................................  45
Predefined Accounts ................................................................................  47
Predefined Member Formulas ...................................................................  53
  Adjusted Annual Salary .......................................................................  53
  Adjusted Annual Salary (Prior) ..........................................................  53
  Annual Overtime ................................................................................  54
  Annual Salary ...................................................................................  54
  Bonus ..................................................................................................  54
  Bonus % ............................................................................................  55
  Bonus Basis .......................................................................................  55
  Cal TP-Index .....................................................................................  55
  Contractor FTE ..................................................................................  56
  Contractor Headcount .........................................................................  56
  CYTD Sign On Bonus ..........................................................................  57
  CYTD Total Salary .............................................................................  57
  CYTD Total Salary (Prior) ...................................................................  57
  Departed Headcount .........................................................................  58
  Differential % ....................................................................................  58
  Fiscal TP-Index ..................................................................................  58
  FT/PT ..................................................................................................  59
  FUTA ..................................................................................................  59
  FUTA Cap ...........................................................................................  60
<table>
<thead>
<tr>
<th>Item</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUTA Rate</td>
<td>60</td>
</tr>
<tr>
<td>FYTD Total Salary</td>
<td>61</td>
</tr>
<tr>
<td>FYTD Total Salary (Prior)</td>
<td>61</td>
</tr>
<tr>
<td>Health Care Costs</td>
<td>61</td>
</tr>
<tr>
<td>LOA Headcount</td>
<td>62</td>
</tr>
<tr>
<td>Maternity Headcount</td>
<td>62</td>
</tr>
<tr>
<td>Medicare</td>
<td>63</td>
</tr>
<tr>
<td>Medicare Rate</td>
<td>63</td>
</tr>
<tr>
<td>Merit</td>
<td>63</td>
</tr>
<tr>
<td>Merit Adjustment %</td>
<td>64</td>
</tr>
<tr>
<td>Merit Adjustment % Cum</td>
<td>64</td>
</tr>
<tr>
<td>NumPeriods</td>
<td>65</td>
</tr>
<tr>
<td>On Sabbatical Headcount</td>
<td>65</td>
</tr>
<tr>
<td>Other FTE</td>
<td>66</td>
</tr>
<tr>
<td>Other Headcount</td>
<td>66</td>
</tr>
<tr>
<td>Overage</td>
<td>67</td>
</tr>
<tr>
<td>Rec. Bonus %</td>
<td>68</td>
</tr>
<tr>
<td>Rec. Merit %</td>
<td>69</td>
</tr>
<tr>
<td>Regular FTE</td>
<td>70</td>
</tr>
<tr>
<td>Regular Headcount</td>
<td>70</td>
</tr>
<tr>
<td>Salary</td>
<td>70</td>
</tr>
<tr>
<td>Social Security Tax</td>
<td>72</td>
</tr>
<tr>
<td>Spread_445</td>
<td>72</td>
</tr>
<tr>
<td>Spread_454</td>
<td>73</td>
</tr>
<tr>
<td>Spread_544</td>
<td>73</td>
</tr>
<tr>
<td>Spread_Actual_365</td>
<td>73</td>
</tr>
<tr>
<td>Spread_Actual_Actual</td>
<td>73</td>
</tr>
<tr>
<td>Spread_Average</td>
<td>74</td>
</tr>
<tr>
<td>Spread_Balance</td>
<td>74</td>
</tr>
<tr>
<td>Spread_First</td>
<td>74</td>
</tr>
<tr>
<td>Spread_Flow</td>
<td>74</td>
</tr>
<tr>
<td>SSTax Cap</td>
<td>75</td>
</tr>
<tr>
<td>SSTax Rate1</td>
<td>75</td>
</tr>
<tr>
<td>SSTax Rate2</td>
<td>75</td>
</tr>
<tr>
<td>Status</td>
<td>75</td>
</tr>
<tr>
<td>SUI</td>
<td>76</td>
</tr>
<tr>
<td>SUI Cap</td>
<td>77</td>
</tr>
<tr>
<td>SUI Rate</td>
<td>77</td>
</tr>
<tr>
<td>Target Rate</td>
<td>77</td>
</tr>
</tbody>
</table>
Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

**Accessibility of Code Examples in Documentation**

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

**Accessibility of Links to External Web Sites in Documentation**

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

**Access to Oracle Support for Hearing-Impaired Customers**

Oracle customers have access to electronic support through My Oracle Support or by calling Oracle Support at 1.800.223.1711. Hearing-impaired customers in the U.S. who wish to speak to an Oracle Support representative may use a telecommunications relay service (TRS). Information about the TRS is available at http://www.fcc.gov/cgb/consumerfacts/trs.html, and a list of telephone numbers is available at http://www.fcc.gov/cgb/dro/trsphonebk.html. International hearing-impaired customers should use the TRS at +1.605.224.1837. An Oracle Support engineer will respond to technical issues according to the standard service request process.
Setting Up Workforce Planning

About Workforce Planning

Oracle Hyperion Workforce Planning, Fusion Edition is a Web-based workforce planning and budgeting solution. It gives corporations a tool to model future headcount and related expenses, and provides a reliable source of up-to-date information about workforce expenses. By planning salary expense in direct correlation to headcount, planners can effectively manage one of their largest variable expenses.

With Workforce Planning, all decision-makers and front-line managers can communicate which course of action to take and ensure budget holders collaborate, which optimizes and streamlines the headcount process. When a material event occurs that causes a change in direction, planners have the flexibility to adapt rapidly, ensuring that plans are relevant and useful.

More specifically, Workforce Planning:

- Calculates workforce-related expenses, including headcount, payroll, salary, taxes, and health care benefits
- Includes employee transfer functionality, facilitating headcount management across dynamic organizations
- Provides a framework for customizing planning, meeting the needs of global enterprises
- Provides drill-through from summary values to underlying detailed data
- Includes event-based activities, such as new hires, that trigger expenses for space allocation, equipment, and hiring bonuses
- Integrates with Oracle Hyperion Planning, Fusion Edition data for reconciliation, forecasting, and reporting
• Integrates with other systems to load information: with flat files for Oracle Hyperion EPM Architect, Fusion Edition applications, and typically with Oracle's Hyperion® Data Integration Management Adapter for Planning for Classic applications
• Supports working with Workforce Planning data forms using Oracle Hyperion Smart View for Office, Fusion Edition, both online and offline

Prerequisites

This section describes prerequisites for setting up and managing Workforce Planning. You should understand:

• Planning functionality (see the *Oracle Hyperion Planning Administrator’s Guide*, the *Oracle Hyperion Planning User’s Guide*, and their online help)
• The Workforce Planning business model (see “Business Model” on page 10)
• If you use Performance Management Architect, see the *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide* and online help.

Business Model

All companies create plans to help prepare for the future, aligning their limited corporate resources—people and dollars—against the strategies that they believe best leverage their competitive market advantage. Through collaborative planning, departments coordinate and allocate the company's finite resources. Companies that can best detect market opportunities and quickly realign their resources gain a competitive advantage. A company's workforce represents its most critical resource.

Using Workforce Planning, companies determine the employee resources needed to achieve their targets, assign existing employees to various positions, and plan for adding new employees. Companies must also gauge the various direct and indirect costs incurred by employees, such as health care, equipment, and taxes. Workforce Planning calculates these expenses—both simple and complex—based on certain drivers. Driver-based planning runs key business assumptions through models, providing the insight to proactively manage the volatility of future financial performance. For example, performance drives bonus and merit increases, primary factors in determining total compensation.

Workforce Planning allows actions such as transferring employees to another department, planning for their departure, and placing them on maternity leave or leave of absence.

Corporate planners, operational managers, or department managers prepare the workforce plans, sometimes including multiple scenarios. They submit them to senior financial and Human Resource managers for review and approval. A corporate planner typically consolidates the plan and prepares reports about the workforce. Companies can iterate plans, when necessary, to respond to changing conditions.
Creating and Initializing a Workforce Planning Application

This section describes how to create and initialize a new Workforce Planning application, using either Performance Management Architect application administration (see “Using Performance Management Architect Application Administration” on page 12) or Classic application administration (see “Using Classic Application Administration” on page 13).

Note: Oracle Hyperion Public Sector Planning and Budgeting, Fusion Edition does not support Workforce Planning, so if you use both modules, set them up as separate applications.

Initializing Workforce Planning loads predefined:

- Workforce Planning Account, Entity, Employee, Scenario, and Version members
- Data forms
- Smart Lists
- Member formulas
- Business rules
- Menus
- UDAs

Before you initialize Workforce Planning:

- Study Appendix A, “Workforce Planning Structure.” Review the predefined elements, identifying which elements you can use and which elements you need to customize. The better you understand the model and plan your application, the easier it is for planners to use Workforce Planning.
- Update your dimension outlines to resolve differences between Workforce Planning member names and names in other applications.
- Plan the Entity dimension structure if you will use both Workforce Planning and Oracle Hyperion Capital Asset Planning, Fusion Edition.

Initializing Workforce Planning produces one No Entity member; initializing Oracle Hyperion Capital Asset Planning, Fusion Edition produces a few Entity members. You can change the Entity members after you initialize the applications.

Note: The Workforce Planning model is based on a 12-month calendar, and the default calculations in Workforce Planning support a single year. However, you can modify the calculations to support multiple years. Also, some member formulas, such as spreading, rely on a monthly calendar. If your application uses custom time periods, you must modify these formulas to support those time periods.
Using Performance Management Architect Application Administration

Follow the steps in this section to use Performance Management Architect application administration.

Creating a Workforce Planning Application

To add the Workforce Planning plan type to an existing Planning application, skip to the next section.

➢ To create a Workforce Planning application:

1. For instructions, see “Building Applications” in the Oracle Hyperion Enterprise Performance Management Architect Edition Administrator’s Guide.

2. For Plan Type, select Wrkforce or Wrkforce and Capex if the application will use both modules.

Adding the Workforce Planning Plan Type to an Existing Planning Application

If you have an existing Planning application, follow these steps to add the Workforce Planning plan type to an existing Planning application.

➢ To add the Workforce Planning plan type to a Planning application:

1. In Performance Management Architect, select Navigate, then Administer, and then Application Library.

2. In the Application Library, right-click the application to which you want to add the Workforce Planning plan type, select Initialize, then select Wrkforce, or Wrkforce and Capex if the application will use both modules.

3. Click Yes to confirm.

Deploying Workforce Planning Applications to Planning

After you create a Workforce Planning application, it is validated and deployed to Planning immediately. The deployment process creates a Planning application automatically.

If the application does not initialize successfully, validation fails and error messages display. Correct any errors and redeploy the application. Your Workforce Planning application is initialized when you successfully deploy an application with Wrkforce selected as the plan type.

If you need to redeploy a Workforce Planning application to Planning, for instructions, see “Validating and Deploying” in the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide.

After you deploy the application from Performance Management Architect to Planning, communicate the URL for logging on, the logon instructions, and information about the planning process.
Tip: You can put the URL on your company’s intranet.

**Using Classic Application Administration**

To create a new Workforce Planning application using Classic application administration, see the *Oracle Hyperion Planning Administrator’s Guide*. When using the Classic Application Wizard to create a Classic application, select the **Wrkforce** plan type.

After creating a Workforce Planning Classic application, you must initialize it.

**Note:** When you create a Classic Planning application, if you select Hyperion Calculation Manager instead of Oracle’s Hyperion® Business Rules as the calculation module, you must manually load the predefined business rules if you select the Capex or Wrkforce plan type. To load the predefined business rules into your application, use the instructions in “Loading Predefined Business Rules for Workforce Planning and Capital Asset Planning” in the *Hyperion Calculation Manager Designer’s Guide*.

**Note:** If you upgrade a Classic Planning application to Performance Management Architect, and then add the Capex or Wrkforce plan type to the application, the predefined business rules are automatically loaded.

**Note:** After you select Calculation Manager as the calculation module for a Classic Planning application, or migrate the application’s business rules to Hyperion Calculation Manager, you cannot use Oracle’s Hyperion® Business Rules with that application.

To initialize a Workforce Planning application:

1. **Start and log on to Planning.**
2. **In Planning, select Administration, then Initialization, and then Workforce.**
   A message confirms successful initialization, and the menu no longer displays **Initialize Workforce**.
3. **Optional:** If the application also uses the Capex plan type, select Administration, then Initialization, and then **Capital Asset**.
4. **Ensure that the Workforce Planning model suits your company’s needs.**
5. **Refresh the application.**
   See “Loading Information into Workforce Planning ” on page 14, and Chapter 2, “Working with Workforce Planning.”

**Caution!** To prevent overwriting your Actual data, tag your Actual scenarios with the UDA named **ACTUAL** before running any calculations.
Loading Information into Workforce Planning

If your Workforce Planning application is new, you may want to load information, such as the existing employee structure, employee properties, and payroll data, from enterprise systems:

- If you use Performance Management Architect application administration, load information using a flat file (see the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide).
- If you use Classic application administration, you can use DIM Adapter for Planning (see “About Loading Information Using DIM Adapter for Planning” on page 15).

You can also:

- Manually enter information into Workforce Planning.
- Load metadata and data that generates data and rules files for loading metadata and data through Oracle Essbase Administration Services. See the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide.

To load workforce information—data and metadata—using DIM Adapter for Planning into a Classic application:

1. Load metadata (for example, employees and departments) into Workforce Planning.

   Load Account, Entity, Employee, and user-defined dimensions from any flat file or ODBC-compliant database. Load members, shared members, and attribute values into dimensions. Oracle recommends that you first load a small sample of employees. Verify the results, make any needed changes, then load the entire dimension.

   See Appendix A, “Workforce Planning Structure,” for a list of Workforce Planning accounts and their properties.

2. Refresh the application to update the Oracle Essbase outline.

3. Set up the data load dimension and dimension driver members.

   The data load dimension is the dimension to which you are loading data, and corresponds to the target table in the DIM Adapter for Planning. The driver dimension is the dimension to which you are loading data in an Essbase database. See “Loading Data” in the Oracle Hyperion Planning Administrator’s Guide.

4. Load data and employee properties into the Essbase database for the Workforce Planning application.

   The outlines must match; you can specify only the members and parent member names.

5. Refresh the application to update the data.

   Each time you modify the application structure, reload the data.

   **Note:** Load fixed accounts only into the BegBalance member and load varying accounts into all time periods. You set the effective POV for each record using the POV port.
About Loading Information Using DIM Adapter for Planning

After you install and configure DIM Adapter for Planning, you must configure an application connection in Workflow Manager before you can extract data from sources or write data into targets. See the Oracle Hyperion Planning Administrator’s Guide, “Loading Members and Data” chapter. For specifics on loading workforce-related information into Planning, see DIM Adapter for Planning Online Help.

Note: You can load information using Oracle’s Hyperion® Data Integration Management Adapter for Planning only into Classic applications.

Logging On and Accessing Workforce Planning


To log on to EPM Workspace and access Performance Management Architect (for Performance Management Architect application administration), Planning, and Workforce Planning:

1. Ensure that the Web server is started and the Web application server is running in the Services panel.
2. In the Web browser, enter the URL for the EPM Workspace Log On page.
3. Enter your system user name.
4. Enter your system password.
5. Click Log On.
6. For Performance Management Architect applications: To access Performance Management Architect, select an option from the Navigate menu.
   For example, select Navigate, then Administer, and then select Dimension Library or Application Library. For information about Performance Management Architect, see the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide or online help.
7. Select Navigate, then Applications, then Planning, and than select an application.
Working with Workforce Planning

In This Chapter

- Working with Employees ................................................................. 17
- Planning Department Expenses ....................................................... 20
- Working with Accounts ................................................................... 20
- Working with Data Forms ................................................................. 22
- Working with Smart Lists and Menus ............................................... 22
- Working with Member Formulas ..................................................... 23
- Working with Business Rules .......................................................... 23
- Using Business Rules to Roll Up Higher Level Entities ..................... 24
- Working with UDAs ....................................................................... 24
- Managing Access Permissions ......................................................... 25
- What’s Next? ................................................................................... 25

Working with Employees

This section explains Workforce Planning employee tasks. To change employee status, planners click cells in data forms and select items from Smart Lists or menus. They can also run business rules to modify employee records (for example, to transfer an employee to a different department).

Employee properties are account members such as Position, Grade, or Salary. Employee members can have associated Smart Lists, and they may depend on another employee property. For example, the value in the FT/PT (full time/part time) Account member depends on the value entered in the FTE (full-time equivalent) account. You can customize the employee properties as described in Chapter 3, “Customizing Workforce Planning.” For example, you can change the employee type, grade, FTE, status, and performance that drive calculations.

Transferring Employees

Planners transfer employees using business rules. Depending on your business needs, employees can either be transferred in two steps, using the Transfer Out and Transfer In business rules, or in one step, using the Transfer business rule.

The two-step transfer process provides security; it ensures that a manager in Department A cannot see member data for Department B, without the appropriate access permissions. The
owner of Department A should transfer out an employee during the same month that the owner of Department B transfers in the employee.

To use the two-step employee-transfer process:

1. Run the Transfer Out business rule to transfer the employee out of the current department.
   The Transfer Out business rule clears the Action account after the transfer out month.

2. Run the Transfer In business rule to transfer the employee into the new department.
   The Transfer In business rule clears the Action account before the transfer in month.

To undo the action for Transfer Out or Transfer In, planners change the Action value in the data form back to the desired value. In most cases, the desired value is #missing.

Planners can use a one-step transfer process if security is not an issue. The Transfer business rule transfers employees out of one department and into another.

To undo the Transfer business rule, planners can launch the rule again, using the same settings but reversing the order of the departments for transfer.

After an employee is transferred out, employee data stays in the old department, but salary for that employee is not calculated within that department. When the employee is transferred to the new department, the status is set to Active for that department and the employee’s salary is calculated in the new department.

While no other information is changed or cleared, other accounts are based on the Action account, so transfers can cause a ripple effect when other accounts are calculated. For example, Action drives Status, which drives Salary, which drives Total Salary, which drives Taxes and Total Compensation. Total Compensation + Taxes drives Expenses, and so on.

**Note:** Planners can transfer an employee into a department even if the employee is already in the department. This is useful when planners want to move the transfer-in date.

**Tip:** Oracle recommends that planners review the Reconcile Transfer Report before approving a plan. See “Reconciling Transfers” on page 18.

You can create task lists to add workflow to the transfer process to suit your business needs. For example, tasks can send e-mail reminders to department managers in the old and new departments. For information about managing task lists, see Oracle Hyperion Planning Administrator’s Online Help.

### Reconciling Transfers

Employees are typically transferred out of one department and into another in the same month. To ensure that the timing of transfers correctly reflects your intentions, Workforce Planning includes two reports that compare employee Transfer Out and In months, enabling you to make corrections, if necessary.
Before approving plans, open the Reconcile Transfer Report data form and launch the Unreconciled Transfers business rule, which validates that transferred employees’ Transfer In and Transfer Out months match. When Transfer Out and In months are not identical, discrepancies display for employees to which you have access. (If there are no discrepancies, no rows display). Review displayed rows:

<table>
<thead>
<tr>
<th>Displayed Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#missing</td>
<td>No errors (Transfers In equal Transfers Out) for the month</td>
</tr>
<tr>
<td>-1</td>
<td>Transfers Out exceed Transfers In for the month</td>
</tr>
<tr>
<td>0</td>
<td>Too many transfers in the month (Transfers In equal Transfers Out but the number of transfers is not 1)</td>
</tr>
<tr>
<td>1</td>
<td>Transfers In exceed Transfers Out in the month</td>
</tr>
</tbody>
</table>

**Note:** You cannot transfer an employee into and out of the same department in the same month.

For example, if two employees are transferred out of Department A in January, but are not transferred to any department until March, their row’s January cells display -1 and their March cells display 1.

**Tip:** Create a Smart List to display meaningful text instead of numerical values. See *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help*.

To make corrections, right-click on an individual’s name, and select the menu item to launch the Reconcile Employee Transfer report, where you can change transfer months.

### Increasing the Maximum Number of TBHs

Workforce Planning includes 30 TBH (To be Hired) employee members for each department, so departments can add 30 new hires during the current plan year, 30 more for the next year, and so on for all subsequent years. You can determine the number of TBHs to allow for each department. To add TBHs, add them on the **Dimensions** tab.

### Converting TBHs to Hires

After TBHs are hired, planners convert the TBH members to the names of newly-hired employees. They create a member for the new employee’s name and run the Reconcile TBH business rule. This reconciles the data for the TBH member for the department to the actual new hire.
Changing Employee Status

To change an employee’s status, planners use a data form to select a new value for Status, such as Maternity Leave or Departed. They can also use the Change Employee Status business rule.

Terminating Employees

To terminate an employee, planners can:

- Run the Plan Departure business rule
- Change the employee’s Status in the data form to Departed or Resigned

Note: If necessary, bonus payments can be made after employees are terminated.

To correct an employee termination error, planners click the data form cell containing the Departed value, then press Delete to remove the value. If an employee was on a non-active status such as Sabbatical when they were terminated, planners must manually change the status after undoing a termination.

Planning Department Expenses

You can plan expenses by individuals or departments. Department expenses are not associated with specific employees, and include items such as printers, telephone, and furniture. You can use the Department General employee member to plan for expenses charged to departments. If you use this member, pay special attention to taxes.

Tip: You can plan turnover with the Turnover data form.

Working with Accounts

You can add new accounts that roll up into aggregates. You can also associate accounts with calculations by adding member formulas or business rules for them. See “Predefined Accounts” on page 47.

Spreading Data Using System Members

You can use System Members account members to spread data for calculations in Workforce Planning. You can spread driver members across time (including annual accounts such as yearly salary) or member values based on another member.

To spread data, you can use:

- A predefined spread type, such as Spread_Average, Spread_Balance, Spread_First, Spread_Flow, Spread_445, Spread_454, Spread_544, Spread_Actual_365, and
Spread_Actual_Actual. If you delete the spread type members, calculations based on them do not work.

Example: Salary = “Annual Salary” * Spread_445

- The time balance spread formula expression, [TimeBalanceSpread]. This formula expression returns the spread type member based on the account’s Time Balance spreading property. It relies on the predefined System Members, which are only available to Workforce Planning applications.

Example: Salary = “Annual Salary” * [TimeBalanceSpread]

Note: Oracle recommends using the time balance spread formula expression so that changes to your outline do not affect Workforce Planning calculations.

About Salary Accounts

Salary accounts include members that indicate how much employees make, how much they are paid, and modifying factors such as Status (Departed, for example), Salary Basis (Hourly, for example), or Pay Type (Exempt, for example).

Annual Salary is calculated:

Table 2 Annual Salary Calculation

<table>
<thead>
<tr>
<th>Salary Basis</th>
<th>Pay Type</th>
<th>Salary Rate</th>
<th>Annual Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hourly</td>
<td>Exempt</td>
<td>Amount the employee makes per hour</td>
<td>Multiply Salary Rate by the Number of hours worked per week, then multiply the result by 52</td>
</tr>
<tr>
<td>Hourly</td>
<td>NonExempt</td>
<td>Amount the employee makes per hour</td>
<td>Multiply Salary Rate by the lesser of Number of hours worked per week or 40 hours, then multiply the result by 52 weeks</td>
</tr>
<tr>
<td>Annual</td>
<td>NA (Pay Type is considered only for Hourly employees)</td>
<td>Amount the employee makes per year</td>
<td>Multiply Salary Rate by FTE</td>
</tr>
</tbody>
</table>

Conceptually, the Employee Expenses member is calculated thus, using hypothetical rounded numbers:

Table 3 Model of Salary Calculations

<table>
<thead>
<tr>
<th>Type of Value</th>
<th>Member Name</th>
<th>Example Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>Salary Rate</td>
<td>100,000</td>
</tr>
<tr>
<td>Input</td>
<td>FTE</td>
<td>0.75</td>
</tr>
<tr>
<td>Calculated</td>
<td>Annual Salary</td>
<td>75,000</td>
</tr>
</tbody>
</table>

Note: See Table 2, “Annual Salary Calculation,” on page 21.

Input | Status | Maternity (65% = 48,000) |
<table>
<thead>
<tr>
<th>Type of Value</th>
<th>Member Name</th>
<th>Example Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculated</td>
<td>Salary</td>
<td>4,000 (48,000 / 12)</td>
</tr>
<tr>
<td>Input or Calculated</td>
<td>Merit Adjustment %</td>
<td>5%</td>
</tr>
<tr>
<td>Calculated</td>
<td>Merit</td>
<td>200 (4,000 * 5%)</td>
</tr>
<tr>
<td>Calculated</td>
<td>Adjusted Salary</td>
<td>4,200 (4,000 + 200)</td>
</tr>
<tr>
<td>Input or Calculated</td>
<td>Bonus %</td>
<td>10%</td>
</tr>
<tr>
<td>Calculated</td>
<td>Bonus (annual total, based on two 6-month bonuses)</td>
<td>4,800</td>
</tr>
<tr>
<td>Calculated</td>
<td>Total Salary</td>
<td>9,000 (4,200 + 4,800)</td>
</tr>
<tr>
<td>Input</td>
<td>Health Plan</td>
<td>Family Plan</td>
</tr>
<tr>
<td>Calculated</td>
<td>Health Care Cost</td>
<td>300</td>
</tr>
<tr>
<td>Calculated</td>
<td>Total Compensation</td>
<td>9,300 (9,000 + 300)</td>
</tr>
<tr>
<td>Calculated</td>
<td>Taxes</td>
<td>2,000</td>
</tr>
<tr>
<td><strong>Calculated</strong></td>
<td><strong>Employee Expenses</strong></td>
<td><strong>11,300</strong></td>
</tr>
</tbody>
</table>

Planners input some of the values, and others Workforce Planning calculates. For values listed as Input or Calculated in the table, planners can accept the default value or override it. For example, planners can override the calculated Bonus % to reflect company policy.

Essbase calculates members in the order of the outline. If you move accounts in the outline, you must be careful about dependencies. Likewise, if you rewrite the logic for the Salary calculations, you may need to reorder the outline. For a list of expense members and the outline hierarchy, see “Predefined Accounts” on page 47.

**Working with Data Forms**

When you make changes to data forms in your model, be sure that changes are synchronized with business logic such as business rules, member formulas, and outline structure. Whenever you modify your business logic, you should check your data forms.

**Working with Smart Lists and Menus**

For Smart Lists and menus, Workforce Planning provides predefined labels that are set as resource strings in the HspCustomMsgs template file. For a list of predefined Smart Lists and menus, see “Predefined Smart Lists” on page 80 and “Predefined Menus” on page 83.

If you add Smart Lists and menus to your application, you should add corresponding labels to the HspCustomMsgs file. You should also update the file whenever you modify the predefined Smart Lists or menus.

22 Working with Workforce Planning
You can localize the labels in your application by updating the localized versions of the HspCustomMsgs file. For instructions on modifying labels or preparing files for localization, see “Customizing Planning Web Client” in Oracle Hyperion Planning Administrator’s Online Help.

**Note:** Modifying the HspCustomMsgs file overrides default values installed with Workforce Planning or creates new values.

After you enter data based on Smart List values, any calculations based on the Smart Lists may not be correct if the lists change. You may need to adjust data in the database to reflect the new values. To understand the implications of changing Smart Lists, see the examples in Chapter 3, “Customizing Workforce Planning.” However, you can delete predefined menus without impacting predefined calculations.

**Working with Member Formulas**

Workforce Planning member formulas include formula expressions, such as [TimeBalanceSpread]. Oracle recommends that you include formula expressions when you write new formulas. Making formulas independent of the outline, enables calculations to work if you change the outline. See “Predefined Member Formulas” on page 53.

See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help or help system.

**Working with Business Rules**

Planners use business rules to perform calculations on employee data, such as:

- Adding new employees to departments. For example, planners use the Add TBH Hourly or Add TBH Salary business rule to add new hires to departments.
- Changing employee status. For example, when employees take a leave of absence, planners use the Change Employee Status business rule to change the employee’s status from Active to Leave of Absence.
- Transferring employees in and out of departments. For example, when employees change departments, planners use the Transfer business rule to move employees out of the old departments and into the new departments.

See “Predefined Business Rules” on page 84.

Administrators work with business rules in the Administration Services Console (Administration Console). You can use the Administration Console to create, validate, launch, and manage business rules. You can also view the content of predefined Workforce Planning business rules (see Hyperion Business Rules Administrator’s Online Help). In Planning, you associate business rules with menus and data forms, and set properties for business rules (see Oracle Hyperion Planning Administrator’s Online Help). Planners and other users launch business
rules to set and calculate employee data on data forms (see Oracle Hyperion Planning User’s Online Help).

**Using Business Rules to Roll Up Higher Level Entities**

To roll up upper-level entities (for example, Division on the page drop-down of the data form 8. Headcount and FTE), create a new business rule similar to the predefined Rollup business rule and associate it with the data form. In the new business rule, include an additional rollup of the Entity dimension. If your application is multicurrency, create a new rule in Oracle Essbase Administration Services, copy and paste the following text, then save the new business rule:

```
SET AGGMISSG ON;
FIX ([Scenario], [Version], [Year], "Local", "HSP_InputValue")
CALC DIM ("Account", "Period", "Employee", "Entity");
ENDFIX
```

**Note:** If your application uses a single currency, remove “Local” from the FIX statement. If you renamed dimensions, the CALC DIM statement should list those dimension names instead.

Run the Rollup business rule instead of the Calc Data on Form calc script on Workforce Planning data forms that contain accounts with member formulas (such as 7. Employee Expenses Summary form, or 8. Headcount and FTE form). These data forms already have the Rollup business rule associated with them. For customized data forms, you may need to associate this rule with the data form during data form design.

**Working with UDAs**

You can use user-defined attributes (UDAs)—words or phrases describing a particular characteristic of members—within calc scripts and reports so they return lists of members associated with the specified UDA. Workforce Planning includes one predefined UDA called ACTUAL. This UDA flags the Actual scenario and ignores member formulas. This prevents actual data from being overwritten.

**Note:** Member formulas do not calculate data in any scenario that has the ACTUAL UDA assigned. Before or after initializing Workforce Planning, assign the ACTUAL UDA to the scenario that is used to store actuals data.

See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help and Oracle Essbase Database Administrator’s Online Help.
Managing Access Permissions

You control access to Workforce Planning—who can view, add, delete, and change information, and to which departments they have access. See how to:

- Set up users and groups in the Oracle Hyperion Enterprise Performance Management System User and Role Security Guide
- Assign access in Oracle Hyperion Planning Administrator’s Online Help

Before planners can add and modify information for a department, they must have access to the department and the Department General entity member.

If you give planners access to a member, they also have access to that member’s data. Salary information is sensitive. You can hide it from viewing and reporting by denying users or groups access to members or to their parents (assign the access type None). You can also screen information by withholding access to certain data forms.

What’s Next?

After setting up your application and loading employee data, you can:

- Set up task lists for planners. To help guide users in the planning process, administrators can set up task lists. The task list shows the activities for which users are responsible and task due dates. It might also include instructions for completing the tasks (see Oracle Hyperion Planning Administrator’s Online Help).
- Assess rolling forecasts by loading prior payroll actual data (typically from the General Ledger and HR system) into the history time periods of the Forecast scenario. Historical information such as prior month actuals remains intact, enabling managers to compare actual results to the same month’s forecast, the same month’s budget, and the prior month’s actuals.
- Customize your Workforce Planning application to accommodate any of your company’s special requirements. See Chapter 3, “Customizing Workforce Planning.”
In This Chapter

About Customizing Your Application ................................................................. 27
Adding a Performance Rating ......................................................................... 27
Changing the Pay Rate for Maternity Status ..................................................... 30
Changing Bonus Frequency ............................................................................ 32
Adding a Salary Adjustment Type ................................................................. 34

About Customizing Your Application

This chapter provides examples for customizing Workforce Planning applications. Before customizing a Workforce Planning application, evaluate the predefined Workforce Planning model—described in Appendix A, “Workforce Planning Structure”—to identify which elements you can use and which you need to customize. Study the logic, formulas, and interrelationships of the predefined members.

Whenever you change the Workforce Planning model, validate the model to test modifications and refresh the application.

Adding a Performance Rating

This example:

- Adds a new Performance rating called Star Performer to the Performance Smart List.
- Updates its associated member formulas to calculate a merit increase and bonus of 20% each.

You can customize other Smart Lists and member formulas similarly. This example does not affect any business rules.

Tip: Smart Lists are used in predefined member formulas, and drive the calculations of the Workforce Planning model. Review how Smart List entries affect member formulas so that you understand the effects of changing or adding a Smart List.
With Performance Management Architect Applications

To add Star Performer as a member of the Performance Smart List:

1 Select Navigate, then Administer, and then Dimension Library.
2 In Shared Library, right-click the Performance Smart List, then select Create Member, and then As Child.
3 Enter the name Star Performer, and then click OK.
   Star Performer is displayed as a Performance Smart List member.
4 In Star Performer properties, in the Value column, enter Star Performer as the Label. Leave all other default values.
   See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

Tip: Labels are displayed in the user interface. Labels can be text or they can reference a resource variable by name. For example, to set a label to Star Performer, either set it to Star Performer directly or set it to the name of a resource—such as LABEL_STAR_PERFORMER—which may be localized. For information about localizing labels, see Oracle Hyperion Planning Administrator’s Online Help.

5 Click Save.
   The Performance Smart List dimension now includes the new performance rating member for use in other calculations.
6 Redeploy the application after this change.
   See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

Add a 20% merit increase calculation and 20% bonus calculation based on Star Performer performance ratings by extending existing member formulas:

1 Select Navigate, then Administer, and then Dimension Library.
2 In Shared Library, select WFP-Accounts, and then click until you see Compensation Assumptions.
3 Select Rec. Merit % to display its properties.
4 In Property, double-click Member Formula, and then click the ellipsis to display the Memo Editor dialog box.
5 In the Memo Editor dialog box, add the following calculation to the member formula to assign a 20% merit increase to employees with a Star Performer rating:

   ELSEIF("Performance" == [Performance.StarPerformer]) 0.20;

   The revised member formula is:

   [OpenInputValueBlock]
   IF (@ISMBR("BegBalance"))
IF ("Performance" == [Performance.FailsToMeetExpectations]) 0.00;
ELSEIF ("Performance" == [Performance.NeedsImprovement]) 0.02;
ELSEIF ("Performance" == [Performance.MeetsExpectations]) 0.04;
ELSEIF ("Performance" == [Performance.ExceedsExpectations]) 0.06;
ELSEIF ("Performance" == [Performance.FarExceedsExpectations]) 0.10;
ELSEIF ("Performance" == [Performance.StarPerformer]) 0.20;
ENDIF
ELSE
  #MISSING;
ENDIF
[CloseInputValueBlock]

6 Validate the change.

See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

Note: You can only validate a member formula after you deploy the Workforce Planning application to Planning.

7 Click OK.

8 Make similar changes to the Rec. Bonus % member formula.

9 Click OK.

10 Redeploy the application after this change. See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

With Classic Applications

To add Star Performer as a member of the Performance Smart List:

1 Select Administration, then Manage, and then Smart Lists.

2 Select the Performance Smart List, and then click Edit.

3 Select Entries, and then click Add.

4 Enter the following:
   In the Name column, enter Star Performer
   In the Label column, enter Star Performer

   Tip: Labels are displayed in the user interface. Labels can be text or they can reference a resource variable by name. For example, to set a label to Star Performer, either set it to Star Performer directly or set it to the name of a resource—such as LABEL_STAR_PERFORMER—which may be localized. For information about localizing labels, see Oracle Hyperion Planning Administrator’s Online Help.

5 Click Save.

The Performance Smart List dimension now includes the new performance rating member for use in other calculations.
Now add a 20% merit increase calculation and 20% bonus calculation based on Star Performer performance ratings by extending existing member formulas:

1. Select Administration, then Manage, and then Dimensions.
2. Select the Account dimension, and then, under Compensation Assumptions, select Rec. Merit %.
3. Select Rec. Merit %, click Edit, and then select Member Formula.
4. In Enter Member Formula, add the following calculation to the member formula to assign a 20% merit increase to employees with a Star Performer rating:

   ELSEIF (*Performance* == [Performance.StarPerformer]) 0.20;

   The revised member formula is:

   [OpenInputValueBlock]  
   IF (@ISMBR("BegBalance"))  
   IF (*Performance* == [Performance.FailsToMeetExpectations]) 0.00;  
   ELSEIF (*Performance* == [Performance.NeedsImprovement]) 0.02;  
   ELSEIF (*Performance* == [Performance.MeetsExpectations]) 0.04;  
   ELSEIF (*Performance* == [Performance.ExceedsExpectations]) 0.06;  
   ELSEIF (*Performance* == [Performance.FarExceedsExpectations]) 0.10;  
   ELSEIF (*Performance* == [Performance.StarPerformer]) 0.20;  
   ENDIF  
   ELSE  
   #MISSING;  
   ENDIF  
   [CloseInputValueBlock]

5. Click Validate Member Formula to validate the change.

   Note: You can validate a member formula only after you deploy the Workforce Planning application to Planning (see Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help).

6. Click Save.
7. Make similar changes to the Rec. Bonus % member formula.
8. Click Save.

### Changing the Pay Rate for Maternity Status

This example changes the salary percentage that employees on Maternity status receive. The current model pays 65%; the example changes it to 80%. To do this, you modify the member formula for the Salary account member.
With Performance Management Architect Applications

To change the Maternity status pay rate from 65% to 80% in Performance Management Architect applications:

1. In Performance Management Architect, select Navigate, then Administer, and then Dimension Library.
2. In Shared Library, select WFP-Accounts, and then click \( \text{until you see Adjusted Salary.} \)
3. Select Salary from Adjusted Salary, then in Property, double-click Member Formula, and then click the ellipsis button to display and edit its value.
4. In Memo Editor, change \(.65 \) to \(.80 \) in this statement:
   \[
   \text{ELSEIF } \("\text{Status}\) == [\text{Status.Maternity}]\)
   \[0.80 \times \"Annual Salary\) \times [\text{TimeBalanceSpread}]\];
5. Click Validate to validate the member formula.

**Note:** You can validate a member formula only after you deploy the Workforce Planning application to Planning.

6. Click OK.

Now employees whose Status is Maternity are paid 80% of their salary each month, until their status changes. You must calculate employee data before the data reflects this change.

7. Redeploy the application after this change.

See Oracle Hyperion Enterprise Performance Management Architect Administrator's Online Help.

**Note:** The Redeploy option is available only after an application is successfully deployed.

With Classic Applications

To change the Maternity status pay rate from 65% to 80% in Classic applications:

1. Select Administration, then Manage, then Dimensions, and then Account.
2. Select the member Salary.
3. Click Edit and then select Member Formula.
4. Change \(.65 \) to \(.80 \) in this code:
   \[
   \text{ELSEIF } \("\text{Status}\) == [\text{Status.Maternity}]\)
   \[0.80 \times \"Annual Salary\) \times [\text{TimeBalanceSpread}]\];
5. Click Save.

Now employees whose Status is Maternity are paid 80% of their salary each month, until their status changes.
6 Refresh the application to update the outline in Essbase.

Note: You must calculate employee data before the data reflects this change.

**Changing Bonus Frequency**

This example changes bonuses from twice a year (March and September) to quarterly (March, June, September, and December). To make the change:

- Add the June and December members to the Compensation Adjustment data form.
- Extend the member formula for the Bonus Basis member to include June and December.

Change the Compensation Adjustments data form to reflect the new bonus months:

1. In Planning, select Administration, then Manage, then Data Forms, and then select the Compensation Adjustments data form.
2. Click Edit and then select Layout.
3. To the right of Column Definition 2, click for the Period dimension.
4. Expand YearTotal, Q2 and Q4, and then select Jun and Dec.
5. Reorder the members so they display in this order: March, June, September, December.
6. Click Submit and then Finish.

The Compensation Adjustments data form now reflects the new bonus months.

Next you change the member formula for the Bonus Basis member to include calculations for the new bonus months.

**With Performance Management Architect Applications**

To change the member formula for the Bonus Basis member in Performance Management Architect applications:

1. In EPM Workspace, select Navigate, then Administer, and then Dimension Library.
2. In Shared Library, expand WFP-Accounts, WFP-Assumptions, Compensation Assumptions, and Bonus Basis to select the member Bonus Basis.
3. In the Value column of the Bonus Basis properties, double-click Member Formula, and then click the ellipsis button to display and edit its value.

The member formula currently allows first and third Quarter bonuses for employees whose type is Regular (set using the Smart List Employee Type):

4. In the Memo Editor dialog box, extend the member formula to review bonuses every three months instead of six, and then save it:
IF ("Employee Type" == [EmployeeType.Regular] AND (@ISMBR ("Mar") OR @ISMBR("Jun") OR @ISMBR("Sep") OR @ISMBR("Dec")))
3;
ELSE
#MISSING;
ENDIF
[CloseInputValueBlock]
Notice that the 6 is now 3, indicating that the formula looks to the last three months instead of six.

5 Click Validate to validate the member formula.

Note: You can validate a member formula only after you deploy the Workforce Planning application to Planning (see Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help).

6 Click OK.

Planners can now input bonus values into the Compensation Adjustments data form quarterly instead of semiannually. You must calculate employee data before the data reflects this change.

7 Redeploy the application after this change. See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

Note: The Redeploy option is available only after an application is successfully deployed.

With Classic Applications

➢ To change the member formula for the Bonus Basis member in Classic applications:

1 In Planning, select Administration, then Manage, then Dimensions and select the Account dimension.

2 Select the Bonus Basis member, click Edit, and then select Member Formula.

3 The member formula currently allows first and third Quarter bonuses for employees whose type is Regular (set using the Smart List Employee Type).

4 Extend the member formula to review bonuses every three months instead of six, and then save it:

[OpenInputValueBlock]
IF ("Employee Type" == [EmployeeType.Regular] AND (@ISMBR ("Mar") OR @ISMBR("Jun") OR @ISMBR("Sep") OR @ISMBR("Dec")))
3;
ELSE
#MISSING;

[CloseInputValueBlock]
ENDIF

[CloseInputValueBlock]

Notice that the 6 is now 3, indicating that the formula looks to the last three months instead of six.

5  Refresh the application to update the outline in Essbase.

Planners can now input bonus values into the Compensation Adjustments data form quarterly instead of semiannually.

Note:  You must calculate employee data before the data reflects this change.

**Adding a Salary Adjustment Type**

This example adds a new salary adjustment type: an equity adjustment, which enables managers to adjust employee salaries by a specified percent to correct their being significantly underpaid. To make the change:

- Create input members for equity month and adjustment percent.
- Add the input members to the Compensation Adjustments data form, in which planners select the effective month (using a Smart List) then enter the equity percent change (usually an increase).
- Add members that calculate the adjusted salary as of the effective month.

**With Performance Management Architect Applications**

To add the Equity Month and Equity Adjustment % input members to Performance Management Architect applications:

1  In Oracle Hyperion EPM Architect, Fusion Edition, select Navigate, then Administer, and then Dimension Library.

2  In Shared Library, expand WFP-Accounts, WFP-Assumptions, and Employee Properties to select the member Merit Month.

3  Right-click Merit Month, then select Create Member, and then select As Sibling.

4  Enter the name, Equity Month, for the new member, and click OK.

   Equity Month is inserted below Merit Month in the Master View. Note that we are associating this member with the Smart List named Month.

   Note:  Whenever you create members, consider assigning access permissions to them based on functions or department.

5  Under the Value column, set these Equity Month properties:
Table 4  Equity Month Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Exchange Rate Type</td>
<td>None</td>
</tr>
<tr>
<td>Data Type</td>
<td>Smart List</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Store</td>
</tr>
<tr>
<td>Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Source Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>Month</td>
</tr>
</tbody>
</table>

6 In Shared Library, expand WFP-Accounts, WFP-Assumptions, and Compensation Assumptions to select the member Merit Adjustment %.

7 Right-click Merit Adjustment %, then select Create Member, and then select As Sibling.

8 Enter the name, Equity Adjustment %, for the new member, and then click OK. Equity Adjustment % is inserted below Merit Adjustment % in the Master View.

9 Under the Value column, set the Equity Adjustment % properties:

Table 5  Equity Adjustment % Properties and Values

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Exchange Rate Type</td>
<td>None</td>
</tr>
<tr>
<td>Data Type</td>
<td>Percentage</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Store</td>
</tr>
<tr>
<td>Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Source Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>None</td>
</tr>
</tbody>
</table>

10 Click Save.
11 Redeploy the application after this change.

See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.

Note: The Redeploy option is available only after an application is successfully deployed.

To add the new input members—Equity Month and Equity Adjustment %—to the Compensation Adjustments data form:

1 In Planning, select Administration, then Manage, then Data Forms, and then select the Compensation Adjustments data form.

2 In Layout, under Column, to the right of the Account members, select.

3 Select the Equity Month and Equity Adjustment % members, and then add them to the end of the list:
   - Annual Salary
   - Pay Type
   - Salary Basis
   - Hours per week
   - Differential %
   - Performance
   - Merit Month
   - Rec. Merit %
   - Override Merit %
   - Merit Adjustment %
   - Equity Month
   - Equity Adjustment %

4 Save the data form.

Add the calculated members Equity Adjustment % Cum and Equity:

1 In Oracle Enterprise Performance Management Workspace, Fusion Edition, select Navigate, then Administer, and then Dimension Library.

2 In Shared Library, expand WFP-Accounts, WFP-Assumptions, and Compensation Assumptions to select the member Equity Adjustment %.

3 Right-click Equity Adjustment %, then select Create Member, and then As Sibling.

4 Enter the name, Equity Adjustment % Cum, for the new member, and click OK.

   Equity Adjustment % Cum is inserted below Equity Adjustment %.

5 Set Equity Adjustment % Cum properties:
Table 6  Equity Adjustment % Cum Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Exchange Rate Type</td>
<td>None</td>
</tr>
<tr>
<td>Data Type</td>
<td>Percentage</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Dynamic Calc</td>
</tr>
<tr>
<td>Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Source Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>None</td>
</tr>
</tbody>
</table>

6 Save the properties.
7 Double-click Member Formula, and then click the ellipsis button to display and edit its value.
8 In the Memo Editor, define and save the formula for Equity Adjustment % Cum:

```
[OpenInputValueBlock]
IF ("Cal TP-Index" == 1 AND "BegBalance"->"Equity Month"<= [Month.Jan])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 2 AND "BegBalance"->"Equity Month"<= [Month.Feb])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 3 AND "BegBalance"->"Equity Month"<= [Month.Mar])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 4 AND "BegBalance"->"Equity Month"<= [Month.Apr])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 5 AND "BegBalance"->"Equity Month"<= [Month.May])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 6 AND "BegBalance"->"Equity Month"<= [Month.Jun])
"BegBalance"->"Equity Adjustment %";
```

Adding a Salary Adjustment Type 37
ELSEIF ("Cal TP-Index" == 7 AND "BegBalance"->"Equity Month"<= [Month.Jul])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 8 AND "BegBalance"->"Equity Month"<= [Month.Aug])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 9 AND "BegBalance"->"Equity Month"<= [Month.Sep])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 10 AND "BegBalance"->"Equity Month"<= [Month.Oct])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 11 AND "BegBalance"->"Equity Month"<= [Month.Nov])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 12 AND "BegBalance"->"Equity Month"<= [Month.Dec])
"BegBalance"->"Equity Adjustment %";
ELSE
#MISSING;
ENDIF
[CloseInputValueBlock]

9 Click Validate to validate the member formula.

Note: You can validate a member formula only after you deploy the Workforce Planning application to Planning (see the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide).

10 Click OK.

11 In Shared Library, expand WFP-Accounts, Employee Expenses, Total Compensation, Total Salary, and Adjusted Salary to select the member Merit.

12 Right-click Merit, then select Create Member, and then select As Sibling.

13 Enter the name, Equity, for the new member, and then click OK.

   Equity is inserted below Merit in the Shared Library.

14 Set Equity properties:
Table 7  Equity Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Expense</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Flow</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Exchange Rate Type</td>
<td>Average</td>
</tr>
<tr>
<td>Data Type</td>
<td>Currency</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Store</td>
</tr>
<tr>
<td>Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Source Plan Type</td>
<td>Wrkforce</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>None</td>
</tr>
</tbody>
</table>

15 Double-click **Member Formula**, and then click the ellipsis button to display and edit its value.

16 **In the Memo Editor**, enter and save the following formula for **Equity**:

```
[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))
"Salary" * "Equity Adjustment % Cum";
ENDIF
[CloseInputValueBlock]
```

**Note:** If you add an equity adjustment account, you must modify the Salary member formula to reflect the equity adjustment.

17 **Click Validate** to validate the member formula.

**Note:** You can validate a member formula only after you deploy the Workforce Planning application to Planning.

18 **Click OK.**

19 **Redeploy the application after this change.**

See *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.*

**Note:** The Redeploy option is available only after an application is successfully deployed.

Now planners can input salary adjustments into the Equity Adjustment % member on the Compensation Assumptions data form, and select the effective month for the adjustment from
a Smart List. The calculated equity value is added to employee’s adjusted salary, starting from the effective month.

### With Classic Applications

To add the Equity Month and Equity Adjustment % input members in Classic applications:

1. In Planning, select **Administration**, then **Manage**, then **Dimensions** and then select the **Account** dimension.

2. **After Merit Month**, add the **Equity Month** input member. See *Oracle Hyperion Planning Administrator’s Online Help*.

3. Set properties for **Equity Month**:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Data Type</td>
<td>Non-Currency</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Store</td>
</tr>
<tr>
<td>Plan Type: Wrkforce</td>
<td>Ignore</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>Month</td>
</tr>
</tbody>
</table>

Note that we are associating this member with the Smart List called Month.

#### Note:
Whenever you create members, consider assigning access permissions to them based on functions or department.

4. **Now add the Equity Adjustment % input member after Merit Adjustment %** and then set its properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
</tbody>
</table>
To add the new input members—Equity Month and Equity Adjustment %—to the Compensation Adjustments data form:

1. In Oracle Hyperion Planning, Fusion Edition, select Administration, then Manage, then Data Forms, and then select the Compensation Adjustments data form.

2. Click Edit, and then Layout.

3. To the right of Account, select , then select the Equity Month and Equity Adjustment % members, and then order them like this:
   - Annual Salary
   - Pay Type
   - Salary Basis
   - Hours per week
   - Differential %
   - Performance
   - Merit Month
   - Rec. Merit %
   - Override Merit %
   - Merit Adjustment %
   - Equity Month
   - Equity Adjustment %

4. Save the data form.

To add the calculated members Equity Adjustment % Cum and Equity:

1. After the Equity Adjustment % member, create the Equity Adjustment % Cum member and set its properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Saved Assumption</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Non-Expense</td>
</tr>
</tbody>
</table>

Adding a Salary Adjustment Type
<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Balance</td>
<td>Balance</td>
</tr>
<tr>
<td>Skip</td>
<td>None</td>
</tr>
<tr>
<td>Data Type</td>
<td>Percentage</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Dynamic Calc</td>
</tr>
<tr>
<td>Plan Type Wrkforce</td>
<td>Ignore</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>None</td>
</tr>
</tbody>
</table>

2. Save the properties and, in **Member Formula**, define and save the formula for **Equity Adjustment % Cum**:

```plaintext
[OpenInputValueBlock]

IF ("Cal TP-Index" == 1 AND "BegBalance"->"Equity Month"<= [Month.Jan])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 2 AND "BegBalance"->"Equity Month"<= [Month.Feb])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 3 AND "BegBalance"->"Equity Month"<= [Month.Mar])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 4 AND "BegBalance"->"Equity Month"<= [Month.Apr])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 5 AND "BegBalance"->"Equity Month"<= [Month.May])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 6 AND "BegBalance"->"Equity Month"<= [Month.Jun])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 7 AND "BegBalance"->"Equity Month"<= [Month.Jul])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 8 AND "BegBalance"->"Equity Month"<= [Month.Aug])
"BegBalance"->"Equity Adjustment %";
```

42 Customizing Workforce Planning
ELSEIF ("Cal TP-Index" == 9 AND "BegBalance"->"Equity Month"<= [Month.Sep])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 10 AND "BegBalance"->"Equity Month"<= [Month.Oct])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 11 AND "BegBalance"->"Equity Month"<= [Month.Nov])
"BegBalance"->"Equity Adjustment %";
ELSEIF ("Cal TP-Index" == 12 AND "BegBalance"->"Equity Month"<= [Month.Dec])
"BegBalance"->"Equity Adjustment %";
ELSE
#MISSING;
ENDIF

3 Add the Equity member after the Merit member, and then set and save its properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Type</td>
<td>Expense</td>
</tr>
<tr>
<td>Variance Reporting</td>
<td>Expense</td>
</tr>
<tr>
<td>Time Balance</td>
<td>Flow</td>
</tr>
<tr>
<td>Exchange Rate Type</td>
<td>Average</td>
</tr>
<tr>
<td>Data Type</td>
<td>Currency</td>
</tr>
<tr>
<td>Data Storage</td>
<td>Store</td>
</tr>
<tr>
<td>Plan Type: Wrkforce</td>
<td>Addition</td>
</tr>
<tr>
<td>Smart Lists</td>
<td>None</td>
</tr>
</tbody>
</table>

4 In Member Formula, enter and save the following formula for Equity:

[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))
"Salary" * "Equity Adjustment % Cum";
ENDIF
[CloseInputValueBlock]
Note: If you add an equity adjustment account, to reflect the equity adjustment, you must modify the Salary member formula.

5 Refresh the application to update the outline in Oracle Essbase.

Now planners can input salary adjustments into the Equity Adjustment % member on the Compensation Assumptions data form and select the effective month for the adjustment from a Smart List. The calculated equity value is added to the employee's adjusted salary, starting from the effective month.
This appendix lists the predefined Workforce Planning elements, helping you determine how elements affect each other.

**Predefined Data Forms**

Planners use data forms to work with employee information.

**Table 12  Predefined Data Forms**

<table>
<thead>
<tr>
<th>Data Form Name</th>
<th>Axis Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Manage Existing Employees Status</td>
<td>Row: Employee, Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Scenario, Version, Year</td>
</tr>
<tr>
<td>2. Manage Existing Employees</td>
<td>Row: Employee</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Account</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Scenario, Version, Year</td>
</tr>
<tr>
<td>3. Existing Employee Related Expenses</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Employee</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Scenario, Version, Year</td>
</tr>
<tr>
<td><strong>Data Form Name</strong></td>
<td><strong>Axis Definitions</strong></td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| 4. New Hires        | Row: Employee  
                          Column: Account  
                          Page: Entity  
                          POV: Currency, Period, Scenario, Version, Year |
| 5. New Hire Expenses| Row: Account  
                          Column: Period  
                          Page: Entity, Employee  
                          POV: Currency, Scenario, Version, Year |
| 6. Compensation Adjustments | Row: Employee  
                          Column: Period, Account  
                          Page: Entity  
                          POV: Currency, Scenario, Version, Year |
| 7. Employee Expenses Summary | Row: Account  
                          Column: Period  
                          Page: Entity, Employee  
                          POV: Currency, Scenario, Version, Year |
| 8. Headcount and FTE | Row: Account  
                          Column: Period  
                          Page: Entity, Employee  
                          POV: Currency, Scenario, Version, Year |
| 9. Employee Information | Row: Employee  
                          Column: Account  
                          Page: Entity  
                          POV: Currency, Period, Scenario, Version, Year |
| Begin vs End        | Row: Employee  
                          Column: Account, Period  
                          Page: Entity  
                          POV: Currency, Scenario, Version, Year |
| Budget vs Target    | Row: Account  
                          Column: Version  
                          Page: Entity  
                          POV: Currency, Employee, Period, Scenario, Year |
<table>
<thead>
<tr>
<th>Data Form Name</th>
<th>Axis Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department Level Expenses</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Employee, Scenario, Version, Year</td>
</tr>
<tr>
<td>Department Level Headcount</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Employee, Scenario, Version, Year</td>
</tr>
<tr>
<td>Reconcile Employee Transfer</td>
<td>Row: Entity</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Employee</td>
</tr>
<tr>
<td></td>
<td>POV: Account, Currency, Scenario, Version, Year</td>
</tr>
<tr>
<td>Reconcile Transfer Report</td>
<td>Row: Employee</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: N/A</td>
</tr>
<tr>
<td></td>
<td>POV: Account, Currency, Entity, Period, Scenario, Version, Year</td>
</tr>
<tr>
<td>Tax Rates</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Year</td>
</tr>
<tr>
<td></td>
<td>Page: N/A</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Employee, Entity, Period, Scenario, Version, Year</td>
</tr>
<tr>
<td>Turnover</td>
<td>Row: Employee, Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity</td>
</tr>
<tr>
<td></td>
<td>POV: Currency, Scenario, Version, Year</td>
</tr>
</tbody>
</table>

### Predefined Accounts

Keep in mind the fixed and varying nature of accounts when working with member formulas and loading data. Some accounts store data that may change during the year (for example, an employee’s status, position, and salary), whereas other account data does not usually change during the year (for example, an employee’s health care plan). Varying accounts store their data in the descendants of Year Total. Fixed accounts store their data in the BegBalance member.

When a formula for a fixed member references a varying member, or when a formula for a varying member references a fixed member, you may need to use the CrossDim operator (->, a dash followed by a right-angle bracket with no spaces in between) with the appropriate time period, based on where its data is stored. This calculation provides an example of a varying
component (Status) that references a fixed component (Start Month) for determining Status using the CrossDim operator:

```
[OpenInputValueBlock]
IF (@ISDESC("New Employees"))
IF ("BegBalance"->"Start Month" != #MISSING AND "Cal TP-Index" >= "BegBalance"->"Start Month") "Status" = [Status.Active];
ENDIF
ELSE
IF ("Action" == [Status.TransferIn]) "Status" = [Status.Active];
ELSE
"Status" = "Action";
ENDIF
IF("Status" == #MISSING) "Status" = @PRIORS(SKIPMISSING, "Status");
ENDIF
ENDIF
[CloseInputValueBlock]
```

For information about the CrossDim operator, see the *Oracle Essbase Database Administrator’s Guide*.

The table shows how Workforce Planning sets up the predefined account members. These accounts are fixed: Merit Month, Start Month, Health Plan, Performance, and Merit Adjustment %. All the other accounts are varying.

<table>
<thead>
<tr>
<th>Table 13</th>
<th>Predefined Accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Account</strong></td>
<td></td>
</tr>
<tr>
<td>System Members</td>
<td></td>
</tr>
<tr>
<td>Cal TP-Index</td>
<td></td>
</tr>
<tr>
<td>Fiscal TP-Index</td>
<td></td>
</tr>
<tr>
<td>NumPeriods</td>
<td></td>
</tr>
<tr>
<td>Spread_Average</td>
<td></td>
</tr>
<tr>
<td>Spread_Balance</td>
<td></td>
</tr>
<tr>
<td>Spread_First</td>
<td></td>
</tr>
<tr>
<td>Spread_Flow</td>
<td></td>
</tr>
<tr>
<td>Spread_445</td>
<td></td>
</tr>
<tr>
<td>Spread_454</td>
<td></td>
</tr>
<tr>
<td>Spread_544</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>Spread_Actual_365</td>
<td></td>
</tr>
<tr>
<td>Spread_Actual_Actual</td>
<td></td>
</tr>
<tr>
<td>Unreconciled Transfers</td>
<td></td>
</tr>
<tr>
<td>TransferInCount</td>
<td></td>
</tr>
<tr>
<td>TransferOutCount</td>
<td></td>
</tr>
<tr>
<td>WFP-Accounts</td>
<td></td>
</tr>
<tr>
<td>Assumption Input</td>
<td></td>
</tr>
<tr>
<td>SSTax Rate1 Input</td>
<td></td>
</tr>
<tr>
<td>SSTax Rate2 Input</td>
<td></td>
</tr>
<tr>
<td>SSTax Cap Input</td>
<td></td>
</tr>
<tr>
<td>SUI Rate Input</td>
<td></td>
</tr>
<tr>
<td>SUI Cap Input</td>
<td></td>
</tr>
<tr>
<td>FUTA Rate Input</td>
<td></td>
</tr>
<tr>
<td>FUTA Cap Input</td>
<td></td>
</tr>
<tr>
<td>Medicare Rate Input</td>
<td></td>
</tr>
<tr>
<td>WFP-Assumptions</td>
<td></td>
</tr>
<tr>
<td>Employee Properties</td>
<td></td>
</tr>
<tr>
<td>Employee Type</td>
<td></td>
</tr>
<tr>
<td>Pay Type</td>
<td></td>
</tr>
<tr>
<td>FT/PT</td>
<td></td>
</tr>
<tr>
<td>FTE</td>
<td></td>
</tr>
<tr>
<td>Merit Month</td>
<td></td>
</tr>
<tr>
<td>Start Month</td>
<td></td>
</tr>
<tr>
<td>Position</td>
<td></td>
</tr>
<tr>
<td>Grade</td>
<td></td>
</tr>
<tr>
<td>Tax Region</td>
<td></td>
</tr>
<tr>
<td>Health Plan</td>
<td></td>
</tr>
<tr>
<td>Action</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td></td>
</tr>
<tr>
<td>------------------</td>
<td></td>
</tr>
<tr>
<td>Status</td>
<td></td>
</tr>
<tr>
<td>Performance</td>
<td></td>
</tr>
<tr>
<td>Salary Basis</td>
<td></td>
</tr>
<tr>
<td>Compensation Assumptions</td>
<td></td>
</tr>
<tr>
<td>Target Rate</td>
<td></td>
</tr>
<tr>
<td>Market Adjustment</td>
<td></td>
</tr>
<tr>
<td>Salary Rate</td>
<td></td>
</tr>
<tr>
<td>Hours per week</td>
<td></td>
</tr>
<tr>
<td>Annual Salary</td>
<td></td>
</tr>
<tr>
<td>Annual Overtime</td>
<td></td>
</tr>
<tr>
<td>Differential %</td>
<td></td>
</tr>
<tr>
<td>Merit Target %</td>
<td></td>
</tr>
<tr>
<td>Rec. Merit %</td>
<td></td>
</tr>
<tr>
<td>Override Merit %</td>
<td></td>
</tr>
<tr>
<td>Bonus Basis Input</td>
<td></td>
</tr>
<tr>
<td>Bonus Basis</td>
<td></td>
</tr>
<tr>
<td>Rec. Bonus %</td>
<td></td>
</tr>
<tr>
<td>Override Bonus %</td>
<td></td>
</tr>
<tr>
<td>Bonus %</td>
<td></td>
</tr>
<tr>
<td>Bonus Adjustment $</td>
<td></td>
</tr>
<tr>
<td>Merit Adjustment %</td>
<td></td>
</tr>
<tr>
<td>Merit Adjustment % Cum</td>
<td></td>
</tr>
<tr>
<td>Adjusted Annual Salary</td>
<td></td>
</tr>
<tr>
<td>Adjusted Annual Salary (Prior)</td>
<td></td>
</tr>
<tr>
<td>Turnover %</td>
<td></td>
</tr>
<tr>
<td>Tax Rate Assumptions</td>
<td></td>
</tr>
<tr>
<td>SSTax Rate1</td>
<td></td>
</tr>
<tr>
<td>SSTax Rate2</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>---</td>
</tr>
<tr>
<td>SSTax Cap</td>
<td></td>
</tr>
<tr>
<td>SUI Rate</td>
<td></td>
</tr>
<tr>
<td>SUI Cap</td>
<td></td>
</tr>
<tr>
<td>FUTA Rate</td>
<td></td>
</tr>
<tr>
<td>FUTA Cap</td>
<td></td>
</tr>
<tr>
<td>Medicare Rate</td>
<td></td>
</tr>
<tr>
<td>Total Headcount</td>
<td></td>
</tr>
<tr>
<td>Department Headcount</td>
<td></td>
</tr>
<tr>
<td>Total Regular Headcount</td>
<td></td>
</tr>
<tr>
<td>Regular Headcount</td>
<td></td>
</tr>
<tr>
<td>Departed Headcount</td>
<td></td>
</tr>
<tr>
<td>LOA Headcount</td>
<td></td>
</tr>
<tr>
<td>Maternity Headcount</td>
<td></td>
</tr>
<tr>
<td>On Sabbatical Headcount</td>
<td></td>
</tr>
<tr>
<td>Turnover Headcount Adjustment</td>
<td></td>
</tr>
<tr>
<td>Contractor Headcount</td>
<td></td>
</tr>
<tr>
<td>Temporary Headcount</td>
<td></td>
</tr>
<tr>
<td>Other Headcount</td>
<td></td>
</tr>
<tr>
<td>Total FTE</td>
<td></td>
</tr>
<tr>
<td>Regular FTE</td>
<td></td>
</tr>
<tr>
<td>Contractor FTE</td>
<td></td>
</tr>
<tr>
<td>Temporary FTE</td>
<td></td>
</tr>
<tr>
<td>Other FTE</td>
<td></td>
</tr>
<tr>
<td>Employee Expenses</td>
<td></td>
</tr>
<tr>
<td>Total Compensation</td>
<td></td>
</tr>
<tr>
<td>Total Salary</td>
<td></td>
</tr>
<tr>
<td>Adjusted Salary</td>
<td></td>
</tr>
<tr>
<td>Salary</td>
<td></td>
</tr>
</tbody>
</table>
Account
Merit
Overtime
Bonus
Sign On Bonus
Commissions
Health Care Costs
Severance
Other Compensation
Turnover Adjustment
Taxes
Social Security Tax
Medicare
SUI
FUTA
Employee Related Expenses
Training
Travel
Telephone
Office Supplies
Departure Costs
Miscellaneous
Total Employee Capital Purchases
Hardware
Software
Furniture
New Hire Related Expenses
Recruiting
Relocation
Predefined Member Formulas

Member formulas are used for employee calculations.

Adjusted Annual Salary

Formula

[OpenInputValueBlock]"Annual Salary" * (1 + "Merit Adjustment % Cum");

[CloseInputValueBlock]

Description

Calculates the adjusted annual salary for the period.

Adjusted Annual Salary (Prior)

Formula

[OpenInputValueBlock]

IF(@ISLEV([Dimension("DIM_NAME_PERIOD")],0))

@PRIOR("Adjusted Annual Salary");

ELSE"Adjusted Annual Salary";

ENDIF

[CloseInputValueBlock]

Description

Calculates the adjusted annual salary for the prior period.
Annual Overtime

Formula

[OpenInputValueBlock]
IF ("Salary Basis" == [SalaryBasis.Annual] OR "Salary Basis" == #MISSING OR "Hours per week" <= 40)
#MISSING;
ELSEIF ("Salary Basis" == [SalaryBasis.Hourly])
   IF ("Pay Type" == [PayType.Exempt] OR "Pay Type" == #MISSING)
      #MISSING;
   ELSEIF ("Pay Type" == [PayType.NonExempt] AND "Hours per week" > 40)
      ("Hours per week" - 40) * "Salary Rate" * 52;
   ENDIF
ENDIF
[CloseInputValueBlock]

Description
Calculates the annual overtime for hourly employees.

Annual Salary

Formula

[OpenInputValueBlock]
IF ("Salary Basis" == [SalaryBasis.Annual] OR "Salary Basis" == #MISSING)
"Salary Rate" * "FTE";
ELSEIF ("Salary Basis" == [SalaryBasis.Hourly])
IF ("Pay Type" == [PayType.Exempt] OR "Pay Type" == #MISSING)
"Hours per week" * "Salary Rate" * 52;
ELSEIF ("Pay Type" == [PayType.NonExempt])
@MIN("Hours per week", 40) * "Salary Rate" * 52;
ENDIF
ENDIF
[CloseInputValueBlock]

Description
Calculates the annual salary. Considers FTE factor.

Bonus

Formula

[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO"), "ACTUAL"])) "Bonus %" * "Adjusted Salary" * "Bonus Basis";
ENDIF

Description
Calculates bonuses based on a bonus percentage and basis.

**Bonus %**

**Formula**

IF("Override Bonus %" == #MISSING) "Rec. Bonus ";
ELSE "Override Bonus %";
ENDIF

Description
Calculates the bonus percentage based on a recommended percentage. Allows bonus percentage overrides.

**Bonus Basis**

**Formula**

IF ("Employee Type" == [EmployeeType-Regular]) AND (@ISMBR([Period("FIRST_QTR_PERIOD")]) OR @ISMBR([Period("THIRD_QTR_PERIOD")]))
6;
ELSE
#MISSING;
ENDIF

Description
Calculates the bonus basis used to determine bonuses.

**Cal TP-Index**

**Formula**
Description
System member that returns the time period index based on a calendar year, regardless of the fiscal year start month. Example: Fiscal year starting in July: Jan=1, Feb=2, Mar=3, and so on.

**Contractor FTE**

**Formula**

```plaintext
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
ELSE
#MISSING;
ENDIF
ENDIF
```

**Description**
Calculates Full Time Equivalent (FTE) for Contractor employee types.

**Contractor Headcount**

**Formula**

```plaintext
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
1;
ELSE
#MISSING;
ENDIF
```
CYTD Sign On Bonus

Formula

[OpenInputValueBlock]
[CYTD("Sign On Bonus")]
[CloseInputValueBlock]

Description
Calculates the calendar year-to-date total for Sign On Bonus.

CYTD Total Salary

Formula

[OpenInputValueBlock]
[CYTD("Total Salary")]
[CloseInputValueBlock]

Description
Calculates the calendar year-to-date for Total Salary.

CYTD Total Salary (Prior)

Formula

[OpenInputValueBlock]
IF ("CYTD Total Salary" <= "Total Salary")
#MISSING;
ELSE "CYTD Total Salary" - "Total Salary";
ENDIF
[CloseInputValueBlock]

Description
Calculates the calendar year-to-date total for prior period’s Total Salary.
**Departed Headcount**

**Formula**

```
[OpenInputValueBlock]

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))

IF("Employee Type" == [EmployeeType.Regular] AND ("Status" == [Status.Departed] OR "Status" == [Status.Resigned]))

1;
ELSE

#MISSING;
ENDIF
ENDIF

[CloseInputValueBlock]
```

**Description**

Calculates Headcount for Regular employee type with a Status of Departed.

**Differential %**

**Formula**

```
[OpenInputValueBlock]

IF (@ISLEV("Employee", 0))

IF ("Salary Basis" == [SalaryBasis.Annual])

("Annual Salary" - "Target Rate") / "Target Rate";
ELSE

#MISSING;
ENDIF
ENDIF

[CloseInputValueBlock]
```

**Description**

Calculates the differential percentage between Target Base Annual salary and the actual Annual salary.

**Fiscal TP-Index**

**Formula**
Description
System member that returns the time period index based on a fiscal year. Example: Fiscal year starting in July: Jul=1, Aug=2, Sep=3, and so on.

**FT/PT**

**Formula**

```
IF ("Salary Basis" == [SalaryBasis.Annual])
    IF ("FTE" == #MISSING)
        #MISSING;
    ELSEIF ("FTE"< 1)
        [FT_PT.PT];
    ELSE
        [FT_PT.FT];
    ENDIF
ELSEIF ("Salary Basis" == [SalaryBasis.Hourly])
    IF ("Hours per week"< 40)
        [FT_PT.PT];
    ELSE
        [FT_PT.FT];
    ENDIF
ELSE
    #MISSING;
ENDIF
```

**Description**
Defines whether an employee is a full-time or part-time employee.

**FUTA**

**Formula**
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))
    IF ("Tax Region" == [TaxRegion.USA])
        IF ("CYTD Total Salary" <= "FUTA Cap")
            "FUTA Rate" * "Total Salary";
        ELSEIF ("CYTD Total Salary (Prior)" <= "FUTA Cap")
            "FUTA Rate" * ("FUTA Cap" - "CYTD Total Salary (Prior)");
        ELSE
            #MISSING;
        ENDIF
    ENDIF
ENDIF

[CloseInputValueBlock]

Description
Calculates simple Federal Unemployment Taxes based on a tax region, rate and cap.

**FUTA Cap**

**Formula**

[OpenInputValueBlock]

[CrossRef("FUTA Cap Input")]

[CloseInputValueBlock]

**Description**

Returns a reference to Federal Unemployment Tax (FUTA) cap at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**FUTA Rate**

**Formula**

[OpenInputValueBlock]

[CrossRef("FUTA Rate Input")]

[CloseInputValueBlock]

**Description**

Returns a reference to Federal Unemployment Tax (FUTA) rate at a predefined location. This member shields other calculations from knowing where the value is actually stored.
**FYTD Total Salary**

Formula

[OpenInputValueBlock]

@SUMRANGE("Total Salary", @CURRMBRRANGE([Dimension("DIM_NAME_PERIOD")], LEV, 0, ,0));

[CloseInputValueBlock]

Description

Calculates the fiscal year-to-date for Total Salary.

**FYTD Total Salary (Prior)**

Formula

[OpenInputValueBlock]

@SUMRANGE("Total Salary", @CURRMBRRANGE([Dimension("DIM_NAME_PERIOD")], LEV, 0, ,-1));

[CloseInputValueBlock]

Description

Calculates the fiscal year-to-date for prior period’s Total Salary.

**Health Care Costs**

Formula

[OpenInputValueBlock]

IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))

IF ("Salary"<> #MISSING) AND ("Employee Type" == [EmployeeType.Regular])

IF("BegBalance"->"Health Plan" == [HealthPlan.IndividualPlan]) 100;
ELSEIF("BegBalance"->"Health Plan" == [HealthPlan.IndividualPlus1]) 175;
ELSEIF("BegBalance"->"Health Plan" == [HealthPlan.FamilyPlan]) 300;
ELSE #MISSING;
ENDIF
ELSE
#MISSING;
ENDIF
ENDIF
[CloseInputValueBlock]

**Description**
Calculates health care costs based on selected health plan.

**LOA Headcount**

**Formula**
[OpenInputValueBlock]
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
IF("Employee Type" == [EmployeeType.Regular] AND "Status" == [Status.LeaveOfAbsence])
1;
ELSE
#MISSING;
ENDIF
ENDIF
[CloseInputValueBlock]

**Description**
Calculates Headcount for Regular employee type with a Status of Leave Of Absence.

**Maternity Headcount**

**Formula**
[OpenInputValueBlock]
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
IF("Employee Type" == [EmployeeType.Regular] AND "Status" == [Status.Maternity])
1;
ELSE #MISSING;
ENDIF
**Medicare**

**Formula**

```plaintext
[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")), "ACTUAL"))
IF ("Tax Region" == [TaxRegion.USA])"Total Salary" * "Medicare Rate";
ENDIF
ENDIF
[CloseInputValueBlock]
```

**Description**
Calculates simple Medicare costs based on a tax region, rate and cap.

**Medicare Rate**

**Formula**

```plaintext
[OpenInputValueBlock]
[CrossRef("Medicare Rate Input")]
[CloseInputValueBlock]
```

**Description**
Returns a reference to Medicare rate at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**Merit**

**Formula**

```plaintext
[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")), "ACTUAL"))"Salary" * "Merit Adjustment % Cum";
ENDIF
[CloseInputValueBlock]
```

**Description**
Calculates Headcount for Regular employee type with a Status of Maternity.
Calculates merit increases based on a merit percentage.

**Merit Adjustment %**

**Formula**

[OpenInputValueBlock]

IF (@ISMBR("BegBalance"))

IF("Override Merit %" == #MISSING)"Rec. Merit %";
ELSE"Override Merit %";
ENDIF
ELSE
#MISSING;
ENDIF
[CloseInputValueBlock]

**Description**

Calculates the merit adjustment percentage based on a recommended percentage. Allows merit percentage overrides.

**Merit Adjustment % Cum**

**Formula**

[OpenInputValueBlock]

IF ("Cal TP-Index" == 1 AND "BegBalance"->"Merit Month"<= [Month.Jan]) "BegBalance"->"Merit Adjustment %";

ELSEIF ("Cal TP-Index" == 2 AND "BegBalance"->"Merit Month"<= [Month.Feb]) "BegBalance"->"Merit Adjustment %";
ELSEIF ("Cal TP-Index" == 3 AND "BegBalance"->"Merit Month"<= [Month.Mar]) "BegBalance"->"Merit Adjustment %";
ELSEIF ("Cal TP-Index" == 4 AND "BegBalance"->"Merit Month"<= [Month.Apr]) "BegBalance"->"Merit Adjustment %";
ELSEIF ("Cal TP-Index" == 5 AND "BegBalance"->"Merit Month"<= [Month.May]) "BegBalance"->"Merit Adjustment %";
ELSEIF ("Cal TP-Index" == 6 AND "BegBalance"->"Merit Month"<= [Month.Jun]) "BegBalance"->"Merit Adjustment %";
ELSEIF ("Cal TP-Index" == 7 AND "BegBalance"->"Merit Month"<= [Month.Jul]) "BegBalance"->"Merit Adjustment %";

[CloseInputValueBlock]
ELSEIF ("Cal TP-Index" == 8 AND "BegBalance"->"Merit Month"<= [Month.Aug]) "BegBalance"->"Merit Adjustment %"
ELSEIF ("Cal TP-Index" == 9 AND "BegBalance"->"Merit Month"<= [Month.Sep]) "BegBalance"->"Merit Adjustment %"
ELSEIF ("Cal TP-Index" == 10 AND "BegBalance"->"Merit Month"<= [Month.Oct]) "BegBalance"->"Merit Adjustment %"
ELSEIF ("Cal TP-Index" == 11 AND "BegBalance"->"Merit Month"<= [Month.Nov]) "BegBalance"->"Merit Adjustment %"
ELSEIF ("Cal TP-Index" == 12 AND "BegBalance"->"Merit Month"<= [Month.Dec]) "BegBalance"->"Merit Adjustment %"
ENDIF

Description
Calculates the cumulative merit adjustment percentages. Use for calculating merit.

NumPeriods

Formula
[OpenInputValueBlock][NumberOfPeriodsInYear];
[CloseInputValueBlock]

Description
System member that returns the number of periods in a year.

On Sabbatical Headcount

Formula
[OpenInputValueBlock]
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
IF("Employee Type" == [EmployeeType.Regular] AND "Status" == [Status.OnSabbatical])
1;
ELSE
#MISSING;
ENDIF
ENDIF

Predefined Member Formulas
Description
Calculates Headcount for Regular employee type with a Status of On Sabbatical.

Other FTE
Formula

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
IF(@ISLEV("Employee",0) AND "Employee Type"> [EmployeeType.Temporary]
"Status" != [Status.TransferOut] AND "Status" != #MISSING)"FTE";
ELSE
#MISSING;
ENDIF
ENDIF

Description
Calculates Full Time Equivalent (FTE) for Other employee types.

Other Headcount
Formula

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")), "ACTUAL") OR @ISMBR("Department General")))
IF(@ISLEV("Employee",0) AND "Employee Type"> [EmployeeType.Temporary]
"Status" != [Status.TransferOut] AND "Status" != #MISSING)
1;
ELSE #MISSING;
ENDIF
ENDIF

Description
Calculates Headcount for Other employee types.

**Overtime**

**Formula**

```plaintext
[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")), "ACTUAL"))
IF (@ISDESC("New Employees"))
IF ("BegBalance"->"Start Month" != #MISSING)
IF ([FiscalOffset] == 0)
IF ("Cal TP-Index">="BegBalance"->"Start Month")"Annual Overtime" *
[TimeBalanceSpread];
ELSE
#MISSING;
ENDIF
ELSEIF ("BegBalance"->"Start Month" - [FiscalOffset]) > 0)
IF ("Fiscal TP-Index">= ("BegBalance"->"Start Month" -
[FiscalOffset]))"Annual Overtime" * [TimeBalanceSpread];
ELSE
#MISSING;
ENDIF
ELSEIF ("Fiscal TP-Index">= ("BegBalance"->"Start Month" +
[NumberOfPeriodsInYear] - [FiscalOffset]))"Annual Overtime" *
[TimeBalanceSpread];
ELSE
#MISSING;
ENDIF;
ELSE
#MISSING;
ENDIF;
ELSE
#MISSING;
ENDIF;
ELSE
IF (@ISMBR("BegBalance"))
#MISSING;
ELSE
```

Predefined Member Formulas 67
IF ("Annual Overtime" == #MISSING) OR ("Status" == [Status.Departed])
OR ("Status" == [Status.Resigned]) OR ("Status" ==
[Status.TransferOut]) OR ("Status" == #MISSING)
#MISSING;
ELSEIF ("Status" == [Status.LeaveOfAbsence])
0;
ELSEIF ("Status" == [Status.Maternity])
0.65 * "Annual Overtime" * [TimeBalanceSpread];
ELSE"Annual Overtime" * [TimeBalanceSpread];
ENDIF
ENDIF;
ENDIF;
ENDIF;
[CloseInputValueBlock]
Description
Calculates overtime for new and existing employees. Existing employees’ overtime takes
employee status into account.

Rec. Bonus %

Formula

[OpenInputValueBlock]
IF ("Grade"<= 2)"Rec. Bonus %" = 0.10;
ELSEIF ("Grade"<= 4)"Rec. Bonus %" = 0.15;
ELSE"Rec. Bonus %" = 0.20;
ENDIF
IF ("Employee Type" == [EmployeeType.Regular])
IF ("BegBalance"->"Performance" ==
[Performance.FailsToMeetExpectations])
0;
ELSEIF ("BegBalance"->"Performance" ==
[Performance.NeedsImprovement])"Rec. Bonus %" * 0.50;
ELSEIF ("BegBalance"->"Performance" ==
ELSEIF ("BegBalance"->"Performance" ==
[Performance.ExceedsExpectations])"Rec. Bonus %" * 1.1;
ELSE
#MISSING;
ENDIF
ELSE
#MISSING;
ENDIF
[CloseInputValueBlock]

Description
Calculates a predefined bonus percentage based on performance.

Rec. Merit %

Formula
[OpenInputValueBlock]
IF (@ISMBR("BegBalance"))
IF (*Performance" == [Performance.FailsToMeetExpectations])
0.00;
ELSEIF (*Performance" == [Performance.NeedsImprovement])
0.02;
ELSEIF (*Performance" == [Performance.MeetsExpectations])
0.04;
ELSEIF (*Performance" == [Performance.ExceedsExpectations])
0.06;
ELSEIF (*Performance" == [Performance.FarExceedsExpectations])
0.10;
ELSE
#MISSING;
ENDIF
ENDIF
[CloseInputValueBlock]

Description
Calculates a predefined merit percentage based on performance.
**Regular FTE**

Formula

[OpenInputValueBlock]

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))


ELSE

#MISSING;

ENDIF

ENDIF

[CloseInputValueBlock]

Description

Calculates Full Time Equivalent (FTE) for Regular employee types.

**Regular Headcount**

Formula

[OpenInputValueBlock]

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))

IF("Employee Type" == [EmployeeType.Regular] AND "Status" == [Status.Active]) 1;

ELSE

#MISSING;

ENDIF

ENDIF

[CloseInputValueBlock]

Description

Calculates Headcount for Regular employee type with a Status of Active.

**Salary**

Formula
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))
IF (@ISDESC("New Employees"))
IF ("BegBalance"->"Start Month" != #MISSING)
IF ([FiscalOffset] == 0)
IF ("Cal TP-Index" >= "BegBalance"->"Start Month") "Annual Salary" * [TimeBalanceSpread];
ELSE
#MISSING;
ENDIF
ELSEIF ("BegBalance"->"Start Month" - [FiscalOffset] > 0)
IF ("Fiscal TP-Index" >= ("BegBalance"->"Start Month" - [FiscalOffset])) "Annual Salary" * [TimeBalanceSpread];
ELSE
#MISSING;
ENDIF
ELSEIF ("Fiscal TP-Index" >= ("BegBalance"->"Start Month" + [NumberOfPeriodsInYear] - [FiscalOffset])) "Annual Salary" * [TimeBalanceSpread];
ELSE
#MISSING;
ENDIF
ELSE
#MISSING;
ENDIF
ELSE
IF (@ISMBR("BegBalance"))
#MISSING;
ELSE
IF ("Annual Salary" == #MISSING) OR ("Status" == [Status.Departed]) OR ("Status" == [Status.Resigned]) OR ("Status" == [Status.TransferOut]) OR ("Status" == #MISSING)
#MISSING;
ELSEIF ("Status" == [Status.LeaveOfAbsence])
0;
}
ELSEIF ("Status" == [Status.Maternity])

0.65 * "Annual Salary" * [TimeBalanceSpread];

ELSE

"Annual Salary" * [TimeBalanceSpread];

ENDIF

ENDIF

ENDIF

ENDIF

[CloseInputValueBlock]

Description

Calculates salary for new and existing employees. Existing employees’ salary takes employee status into account.

Social Security Tax

Formula

[OpenInputValueBlock]

IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")), "ACTUAL"))

IF ("Tax Region" == [TaxRegion.USA])

IF ("CYTD Total Salary" <= "SSTax Cap")

"SSTax Rate1" * "Total Salary";

ELSEIF ("CYTD Total Salary (Prior)" <= "SSTax Cap")

"SSTax Rate1" * ("SSTax Cap" - "CYTD Total Salary (Prior)") + "SSTax Rate2" * ("CYTD Total Salary" - "SSTax Cap");

ELSE

"SSTax Rate2" * "Total Salary";

ENDIF

ENDIF

ENDIF

[CloseInputValueBlock]

Description

Calculates simple Social Security Taxes (FICA) based on a tax region, rate and cap.

Spread_445

Formula

[OpenInputValueBlock]

[TimeBalanceFormula("Spread_445")]

[CloseInputValueBlock]
Description
System member that returns a spread factor following 4 week, 4 week, 5 week quarter.

**Spread_454**

**Description**
System member that returns a spread factor following 4 week, 5 week, 4 week quarter.

**Spread_544**

**Description**
System member that returns a spread factor following 5 week, 4 week, 4 week quarter.

**Spread_Actual_365**

**Description**
System member that returns a spread factor following a 365-day calendar year.

**Spread_Actual_Actual**

**Description**
System member that returns a spread factor.
Description
System member that returns a spread factor following the actual days in a calendar year.

**Spread_Average**

Formula

```
[TimeBalanceFormula("Spread_Average")]
```

Description
System member that returns a spread factor following the time balance average pattern.

**Spread_Balance**

Formula

```
[TimeBalanceFormula("Spread_Balance")]
```

Description
System member that returns a spread factor following the time balance last pattern.

**Spread_First**

Formula

```
[TimeBalanceFormula("Spread_First")]
```

Description
System member that returns a spread factor following the time balance first pattern.

**Spread_Flow**

Formula

```
[TimeBalanceFormula("Spread_Flow")]
```
Description
System member that returns a spread factor following the normal flow pattern.

**SSTax Cap**

**Formula**

[OpenInputValueBlock]
[CrossRef("SSTax Cap Input")]
[CloseInputValueBlock]

**Description**
Returns a reference to Social Security Tax (FICA) cap at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**SSTax Rate1**

**Formula**

[OpenInputValueBlock]
[CrossRef("SSTax Rate1 Input")]
[CloseInputValueBlock]

**Description**
Returns a reference to Social Security Tax (FICA) primary rate at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**SSTax Rate2**

**Formula**

[OpenInputValueBlock]
[CrossRef("SSTax Rate2 Input")]
[CloseInputValueBlock]

**Description**
Returns a reference to Social Security Tax (FICA) secondary rate at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**Status**

**Formula**
Description
Calculates the employee’s status based on last action applied. Status determines whether an employee is paid.

SUI

Formula
[OpenInputValueBlock]
IF (NOT @ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL"))
IF ("Tax Region" == [TaxRegion.USA])
IF ("CYTD Total Salary" <= "SUI Cap")"SUI Rate" * "Total Salary";
ELSEIF("CYTD Total Salary (Prior)" <= "SUI Cap")"SUI Rate" * ("SUI Cap" - "CYTD Total Salary (Prior)");
ELSE
#MISSING;
ENDIF
ENDIF
ENDIF
[CloseInputValueBlock]

Description
Calculates simple State Unemployment Insurance (SUI) based on a tax region, rate and cap.

**SUI Cap**

*Formula*

[OpenInputValueBlock]
[CrossRef("SUI Cap Input")]
[CloseInputValueBlock]

*Description*
Returns a reference to State Unemployment Insurance (SUI) cap at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**SUI Rate**

*Formula*

[OpenInputValueBlock]
[CrossRef("SUI Rate Input")]
[CloseInputValueBlock]

*Description*
Returns a reference to State Unemployment Insurance (SUI) rate at a predefined location. This member shields other calculations from knowing where the value is actually stored.

**Target Rate**

*Formula*

[OpenInputValueBlock]
IF (@ISLEV("Employee", 0))
IF ("Salary Basis" == [SalaryBasis.Annual])
IF("Grade" == 1) "Target Rate" = 80000;
ELSEIF("Grade" == 2)"Target Rate" = 90000;
ELSEIF("Grade" == 3)"Target Rate" = 100000;
ELSEIF("Grade" == 4)"Target Rate" = 110000;
ELSEIF("Grade" == 5)"Target Rate" = 120000;
ELSEIF("Grade" == 6)"Target Rate" = 130000;
ELSEIF("Grade" == 7)"Target Rate" = 140000;
ELSEIF("Grade" == 8)"Target Rate" = 150000;
ELSEIF("Grade" == 9)"Target Rate" = 160000;
ELSEIF("Grade" == 10)"Target Rate" = 170000;
ELSEIF("Grade" == 11)"Target Rate" = 180000;
ELSEIF("Grade" == 12)"Target Rate" = 190000;
ELSEIF("Grade" == 13)"Target Rate" = 200000;
ELSE "Target Rate" = #MISSING;
ENDIF
ELSEIF ("Salary Basis" == [SalaryBasis.Hourly])"Target Rate";
ENDIF
ENDIF
[CloseInputValueBlock]

Description
Calculates the predefined Target Rate based on employee grade.

**Temporary FTE**

**Formula**

[OpenInputValueBlock]

IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO")], "ACTUAL") OR @ISMBR("Department General")))
ELSE
#MISSING;
ENDIF
ENDIF
[CloseInputValueBlock]
Description
Calculates Full Time Equivalent (FTE) for Temporary employee types.

**Temporary Headcount**

**Formula**

```plaintext
[OpenInputValueBlock]
IF (NOT (@ISUDA([Dimension("DIM_NAME_SCENARIO"), "ACTUAL") OR @ISMBR("Department General")))
1;
ELSE
#MISSING;
ENDIF
ENDIF
[CloseInputValueBlock]
```

Description
Calculates Headcount for Temporary employee types.

**Turnover Adjustment**

**Formula**

```plaintext
[OpenInputValueBlock]
IF (@ISMBR("Department General") ) "Turnover Adjustment" = ("Total Employees"->"Total Compensation" + "Turnover Adjustment") * "Turnover %";
ELSE
#MISSING;
ENDIF
[CloseInputValueBlock]
```

Description
Calculates a turnover adjustment.
**Turnover Headcount Adjustment**

**Formula**

[OpenInputValueBlock]

IF (@ISMBR("Department General"))"Turnover Headcount Adjustment" = @ROUND({"Total Employees"->"Total Headcount" + "Turnover Headcount Adjustment") * "Turnover %", 0);

ELSE

#MISSING;

ENDIF

[CloseInputValueBlock]

**Description**

Calculates a turnover headcount adjustment for employees.

---

**Predefined Smart Lists**

Planners use data form Smart Lists to work with employee data.

**Table 14  Predefined Smart Lists**

<table>
<thead>
<tr>
<th>Smart List</th>
<th>Associated Account Member</th>
<th>Used in Member Formulas</th>
</tr>
</thead>
<tbody>
<tr>
<td>EmployeeType</td>
<td>Employee Type</td>
<td>Bonus Basis, Rec. Bonus %, Regular FTE, Regular Headcount, Departed Headcount, Contractor FTE, Health Care Costs, Contractor Headcount, LOA Headcount, Temporary FTE, Other FTE, Maternity Headcount, Temporary Headcount, On Sabbatical Headcount, Other Headcount</td>
</tr>
<tr>
<td>FT_PT</td>
<td>FT/PT</td>
<td>FT/PT</td>
</tr>
<tr>
<td>HealthPlan</td>
<td>Health Plan</td>
<td>Health Care Costs</td>
</tr>
<tr>
<td>Month</td>
<td>Start Month</td>
<td>Merit Adjustment % Cum</td>
</tr>
<tr>
<td>PayType</td>
<td>Pay Type</td>
<td>Annual Overtime, Annual Salary</td>
</tr>
<tr>
<td>Performance</td>
<td>Performance</td>
<td>Rec. Merit %, Rec. Bonus %</td>
</tr>
<tr>
<td>Position</td>
<td>Position</td>
<td>Assistant, Manager, Supervisor, Director, VP</td>
</tr>
<tr>
<td>SalaryBasis</td>
<td>Salary Basis</td>
<td>Differential %, Annual Overtime, Annual Salary, Target Rate, FT/PT</td>
</tr>
<tr>
<td>Status</td>
<td>Status</td>
<td>Salary, Regular FTE, Regular Headcount, Departed Headcount, Contractor FTE, Contractor Headcount, LOA Headcount, Temporary FTE, Other FTE, Maternity Headcount, Temporary Headcount, On Sabbatical Headcount, Other Headcount, Status</td>
</tr>
<tr>
<td>TaxRegion</td>
<td>Tax Region</td>
<td>Social Security Tax, Medicare, SUI, FUTA</td>
</tr>
</tbody>
</table>
Predefined Smart List Entries

Review the predefined Smart Lists to determine if you need to change any or add new ones. Smart Lists are used in predefined member formulas and drive the calculations of the Oracle Hyperion Workforce Planning, Fusion Edition mode. Before changing or adding Smart Lists, review how Smart List entries affect member formulas.

Oracle recommends adding new entries to the predefined Smart Lists instead of replacing them. If you change Smart Lists when there is data already in the system, you must update the data to the new values. Consider potential future changes when designing your Smart Lists.

You may want to customize these predefined Smart Lists entries:

- Tax Region—your company’s tax regions, depending on the level of granularity of tax calculations in your application. To support new tax types, add a new Tax Region and add the appropriate member with the business logic under Taxes.
- Performance—the employee performance categories for your company, such as Meets Expectations.
- Position—position titles such as Software Engineer and Director (not used in predefined calculations, so you can easily change the Performance Smart List entries).
- Status—employee status, such as Active, Departed, and Maternity.

### Table 15 Predefined Smart List Entries

<table>
<thead>
<tr>
<th>Smart List</th>
<th>Entries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee Type</td>
<td>Regular, Contractor, Temporary</td>
</tr>
<tr>
<td>FT_PT</td>
<td>PT, FT</td>
</tr>
<tr>
<td>HealthPlan</td>
<td>IndividualPlan, IndividualPlus1, FamilyPlan</td>
</tr>
<tr>
<td>Smart List</td>
<td>Entries</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Month</td>
<td>Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec</td>
</tr>
<tr>
<td>PayType</td>
<td>Exempt, NonExempt</td>
</tr>
<tr>
<td>Performance</td>
<td>FailsToMeetExpectation, NeedsImprovement, MeetsExpectations, ExceedsExpectations, FarExceedsExpectation</td>
</tr>
<tr>
<td>Position</td>
<td>Assistant, Manager, Supervisor, Director, VP</td>
</tr>
<tr>
<td>SalaryBasis</td>
<td>Annual, Hourly</td>
</tr>
<tr>
<td>Status</td>
<td>Active, Departed, Disability, LeaveOfAbsence, Maternity, OnSabbatical, TransferOut, TransferIn, Resigned</td>
</tr>
<tr>
<td>TaxRegion</td>
<td>NoRegion, USA</td>
</tr>
</tbody>
</table>
Predefined Menus

Planners use menus to work with employee data in data forms. The information listed in the Label Value column displays when planners click a row member.

### Table 16  Predefined Menu: WFPMenu

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Employee Status</td>
<td>Runs the predefined business rule to change the status of selected employees</td>
</tr>
<tr>
<td>Transfer In</td>
<td>Runs the predefined business rule to transfer selected employees into a department</td>
</tr>
<tr>
<td>Transfer Out</td>
<td>Runs the predefined business rule to transfer selected employees out of a department</td>
</tr>
<tr>
<td>Plan Departure</td>
<td>Runs the predefined business rule to plan for employee departures</td>
</tr>
<tr>
<td>Edit Related Expenses</td>
<td>Lets planners edit related expenses</td>
</tr>
</tbody>
</table>

### Table 17  Predefined Menu: WFPMenuNewHires

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add TBH</td>
<td>Menu header for Add Salary TBH and Add Hourly TBH menu items</td>
</tr>
<tr>
<td>Add Salary TBH</td>
<td>Adds annual salaried TBH to the selected department</td>
</tr>
<tr>
<td>Add Hourly TBH</td>
<td>Adds hourly paid TBH to the selected department</td>
</tr>
<tr>
<td>Remove TBH</td>
<td>Runs the predefined business rule to remove new hires</td>
</tr>
<tr>
<td>Reconcile TBH</td>
<td>Runs the predefined business rule to reconcile TBHs with actual new hires</td>
</tr>
</tbody>
</table>

### Table 18  Predefined Menu: WFPMenuRecXfer

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calculate Transfer Report</td>
<td>Runs the predefined business rule to calculate the transferred employees report</td>
</tr>
<tr>
<td>Reconcile Employee Transfer</td>
<td>Goes to the Reconcile Employee Transfer Data form</td>
</tr>
</tbody>
</table>

### Table 19  Predefined Menu: WFPMenuToMEE

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>To Manage Existing Employees Status</td>
<td>Goes to the Manage Existing Employee Status data form</td>
</tr>
<tr>
<td>To Manage Existing Employees</td>
<td>Goes to the Manage Existing Employee data form</td>
</tr>
</tbody>
</table>

### Table 20  Predefined Menu: WFPMenuToNewHires

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>To New Hires</td>
<td>Goes to the New Hires data form</td>
</tr>
</tbody>
</table>
Table 21  Predefined Menu: WFPMenuToRecXfer

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Go To Reconcile Transfer Report</td>
<td>Goes to the Reconcile Transfer Report data form</td>
</tr>
</tbody>
</table>

This table lists the predefined data forms that have associated menus.

Table 22  Predefined Menus in Data Forms

<table>
<thead>
<tr>
<th>Data Form</th>
<th>Associated Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Manage Existing Employee Status</td>
<td>WFPMenu</td>
</tr>
<tr>
<td>2. Manage Existing Employees</td>
<td>WFPMenu</td>
</tr>
<tr>
<td>3. Existing Employee Related Expenses</td>
<td>WFPMenuToMEE</td>
</tr>
<tr>
<td>4. New Hires</td>
<td>WFPMenuNewHires</td>
</tr>
<tr>
<td>5. New Hire Expenses</td>
<td>WFPMenuToNewHires</td>
</tr>
<tr>
<td>Reconcile Employee Transfer</td>
<td>WFPMenuToRecXfer</td>
</tr>
<tr>
<td>Reconcile Transfer Report</td>
<td>WFPMenuRecXfer</td>
</tr>
</tbody>
</table>

Predefined Business Rules

Planners use predefined business rules to calculate employee data.

Add TBH Hourly

Description

Adds hourly-paid employees to the selected department.

Formula

```
SET CREATE NONMISSING BLK ON;
VAR numTBH = [NumTBH];
VAR validTBH = 0;
FIX(@CHILDREN("New Employees"), [Scenario], [Version], [Department], [Year], "Local", "HSP_InputValue")
   FIX(@LEVMBRS("Period", 0))
"Start Month"
   (                   
      IF (@ISMBR("BegBalance"))
         validTBH = 0;
      IF (numTBH > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Employee Properties")) == #MISSING )
         "Start Month" = [StartMonth] -> "Cal TP-Index";
      "Health Plan" = [HealthPlan];
      numTBH = numTBH - 1;
      validTBH = 1;
   )
```
Add TBH Salary

Description

Adds salaried employees to the selected department.

Formula

SET CREATENONMISSINGBLK ON;
VAR numTBH = [NumTBH];
VAR validTBH = 0;
FIX(@CHILDREN("New Employees"), [Scenario], [Version], [Department], [Year], "Local", "HSP_InputValue")
   FIX(@LEVMBRS("Period", 0))
   "Start Month"
   (       IF (@ISMBR("BegBalance"))
           validTBH = 0;
           IF (numTBH > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Employee Properties")) == #MISSING )
                   "Start Month" = [StartMonth]->"Cal TP-Index";
                   "Health Plan" = [HealthPlan];
                   numTBH = numTBH - 1;
                   validTBH = 1;
               ENDIF
           ENDIF
       IF (validTBH == 1)
           "Grade" = [Grade];
           IF ([MarketAdjustment] != 0)
               "Market Adjustment" = [MarketAdjustment];
           ENDIF
           "FTE" = [FTE];
           "Position" = [Position];
           "Tax Region" = [TaxRegion];
           "Employee Type" = [EmployeeType];
           "Pay Type" = [PayTypeNonExempt];
           "Salary Basis" = 2;
           "Hours per week" = [HoursPerWeek];
           "Target Rate" = [SalaryRate];
           "Salary Rate" = [SalaryRate] + [MarketAdjustment];
       ENDIF
)
"Salary Basis" = 1;
"Pay Type" = [PayTypeExempt];
ENDIF
)
ENDFIX
CALC DIM ("Account");
FIX(@LEVMBRS("Period", 0))
"Salary Rate" = "Target Rate" + "Market Adjustment";
ENDFIX
CALC DIM ("Account");
ENDFIX

Change Employee Status

Description
Changes employee status.

Formula

SET CREATENONMISSINGBLK ON;
VAR startIndex;
VAR endIndex;
VAR existingAction = #MISSING;
FIX ([Scenario], [Version], [Employees], [Department], [Month], [Year], "Local",
"HSP_InputValue")
"Action"
{
    startIndex = [Month]->"Fiscal TP-Index";
    endIndex = startIndex + [Duration];
    IF (endIndex < startIndex)
        endIndex = startIndex;
    ENDIF
};
ENDFIX
FIX ([Scenario], [Version], [Employees], [Department], [Year], "Local",
"HSP_InputValue")
"Action"
{
    IF ("Fiscal TP-Index" == startIndex)
        "Action" = [Action];
    ELSEIF ("Fiscal TP-Index" > startIndex AND "Fiscal TP-Index" < endIndex)
        existingAction = "Action";
        "Action" = #MISSING;
    ELSEIF ("Fiscal TP-Index" == endIndex AND "Action" == #MISSING)
        IF (existingAction == #MISSING)
            "Action" = 1;
        ELSE
            "Action" = existingAction;
        ENDIF
    ENDIF
};
CALC DIM ("Account");
ENDFIX
**CopyProps**

**Description**
Copies employee properties from BegBalance into all time periods.

**Formula**

```plaintext
FIX ([Scenario], [Version], [Department], [Year], @CHILDREN("New Employees"), @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Start Month" (  
  "Grade" = "BegBalance"->"Grade";
  "FTE" = "BegBalance"->"FTE";
  "Position" = "BegBalance"->"Position";
  "Tax Region" = "BegBalance"->"Tax Region";
  "Employee Type" = "BegBalance"->"Employee Type";
  "Pay Type" = "BegBalance"->"Pay Type";
  "Salary Basis" = "BegBalance"->"Salary Basis";
  "Salary Rate" = "BegBalance"->"Target Rate" + "BegBalance"->"Market Adjustment";
  "Target Rate" = "BegBalance"->"Target Rate";
  "Market Adjustment" = "BegBalance"->"Market Adjustment";
  IF ("Salary Basis" == 2)
    "Hours per week" = "BegBalance"->"Hours per week";
ENDIF
);
CALC DIM (*Account*);
ENDFIX
```

**Plan Departure**

**Description**
Lets users plan for an employee's departure.

**Formula**

```plaintext
SET AGGMISSG ON;
FIX ([Scenario], [Version], [Employees], [Department], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Action" (  
  IF ("Fiscal TP-Index" > [Month]->"Fiscal TP-Index")
    "Action" = #MISSING;
  ELSEIF ("Fiscal TP-Index" == [Month]->"Fiscal TP-Index")
    "Action" = [Action];
ENDIF
);
CALC DIM (*Account*);
ENDFIX
```

**Reconcile TBH**

**Description**
Reconciles TBHs with actual new hires.
**Formula**

```plaintext
FIX ([Scenario], [Version], [Department])
  DATACOPY [SrcEmployee] TO [DestEmployee];
ENDIF
FIX ([Scenario], [Version], [Department], [DestEmployee], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Start Month" (
  IF ("Cal TP-Index" == "BegBalance"->"Start Month")
    "Action" = 1;
    "BegBalance"->"Start Month" = #MISSING;
  ENDIF
  "Grade" = "BegBalance"->"Grade";
  "FTE" = "BegBalance"->"FTE";
  "Position" = "BegBalance"->"Position";
  "Tax Region" = "BegBalance"->"Tax Region";
  "Employee Type" = "BegBalance"->"Employee Type";
  "Pay Type" = "BegBalance"->"Pay Type";
  "Salary Basis" = "BegBalance"->"Salary Basis";
  "Salary Rate" = "BegBalance"->"Target Rate" + "BegBalance"->"Market Adjustment";
  IF ("Salary Basis" == 2)
    "Hours per week" = "BegBalance"->"Hours per week";
  ENDIF
);
CALC DIM ("Account");
ENDIF
FIX ([Scenario], [Version], [Department])
  CLEARDATA [SrcEmployee];
  CALC DIM ("Account");
ENDIF
```

**Remove TBH**

**Description**
Removes a TBH.

**Formula**

```plaintext
FIX ([Scenario], [Version], [Department])
  CLEARDATA [LocalEmployee];
  CALC DIM ("Account");
ENDIF
```

**Rollup**

**Description**
Rolls up certain dimensions.

**Formula**

```plaintext
SET AGGMISSG ON;
FIX ([Scenario], [Version], [Department], [Year], "Local", "HSP_InputValue")
```
**Transfer**

Description

Performs a one-step employee transfer.

Formula

```plaintext
SET CREATENONMISSINGBLK ON;
VAR savedAction = 0;
FIX ([Scenario], [Version], [Employees], [Year])
    DATACOPY [FromDepartment] TO [ToDepartment];
ENDFIX
FIX ([Scenario], [Version], [Employees], [FromDepartment], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Action"(
    IF ("Fiscal TP-Index" == [Month]->"Fiscal TP-Index")
        "Action" = 7;
    ELSEIF ("Fiscal TP-Index" > [Month]->"Fiscal TP-Index")
        "Action" = #MISSING;
    ENDIF
);  
CALC DIM ("Account");
ENDFIX
FIX ([Scenario], [Version], [Employees], [ToDepartment], [Year], [Month], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Action"(
    IF ("Fiscal TP-Index" == [Month]->"Fiscal TP-Index")
        "Action" = 8;
    ELSEIF ("Fiscal TP-Index" < [Month]->"Fiscal TP-Index")
        "Action" = #MISSING;
    ELSE
        IF (savedAction <> 7 AND "Action" == 8)
            "Action" = #MISSING;
        ELSEIF ("Action" == 7)
            savedAction = 7;
        ENDIF
    ENDIF
);  
CALC DIM ("Account");
ENDFIX
```

**Transfer In**

Description

Transfers an employee into a department.

Formula

```plaintext
SET CREATENONMISSINGBLK ON;
VAR savedAction = 0;
```
FIX ([Scenario], [Version], [Employees], [Department], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")

"Action"(
    IF ("Fiscal TP-Index" == [Month]->"Fiscal TP-Index")
        "Action" = 8;
    ELSEIF ("Fiscal TP-Index" < [Month]->"Fiscal TP-Index")
        "Action" = #MISSING;
    ELSE
        IF (savedAction <> 7 AND "Action" == 8)
            "Action" = #MISSING;
        ELSEIF ("Action" == 7)
            savedAction = 7;
        ENDIF
    ENDIF
)
ENDFIX

FIX ([Scenario], [Version], [Employees], [Department], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")

"Action"
{
    IF ([FTE] != #MISSING)
        "FTE" = [FTE];
    ENDIF;
    IF ([Position] != #MISSING)
        "Position" = [Position];
    ENDIF;
    IF ([Grade] != #MISSING)
        "Grade" = [Grade];
    ENDIF;
    IF ([TaxRegion] != #MISSING)
        "Tax Region" = [TaxRegion];
    ENDIF;
    IF ([EmployeeType] != #MISSING)
        "Employee Type" = [EmployeeType];
    ENDIF;
    IF ([HealthPlan] != #MISSING)
        "Health Plan" = [HealthPlan];
    ENDIF;
    IF ([MeritMonth] != #MISSING)
        "Merit Month" = [MeritMonth];
    ENDIF;
    IF ([SalaryBasis] != #MISSING)
        "Salary Basis" = [SalaryBasis];
        IF ("Salary Basis" == 2)
            "Hours per week" = [HoursPerWeek];
        ENDIF
    ENDIF;
    IF ([SalaryRate] != #MISSING)
        "Salary Rate" = [SalaryRate];
    ENDIF;
    IF ([PayType] != #MISSING)
        "Pay Type" = [PayType];
    ENDIF;
)
CALC DIM ("Account");
ENDFIX
Transfer Out

Description

Transfers an employee out of a department.

Formula

FIX ([Scenario], [Version], [Employees], [Department], [Year], @LEVMBRS("Period", 0), "Local", "HSP_InputValue")
"Action"

(IF  ("Fiscal TP-Index" == [Month]->"Fiscal TP-Index")
 "Action" = 7;
 ELSEIF  ("Fiscal TP-Index" > [Month]->"Fiscal TP-Index")
 "Action" = #MISSING;
ENDIF
);

CALC DIM ("Account");
ENDFIX

Unreconciled Transfers

Description

Compares transfers out against transfers in to produce a reconciliation report.

Formula

SET CREATEONMISSINGBLK ON;

FIX ([Scenario], [Version], [Year], @REMOVE(@LEVMBRS("Employee",0), @LIST(@CHILDREN("New Employees"), "Department General", "No Employee")), @LEVMBRS("Period", 0), "No Entity", "Local", "HSP_InputValue")

"TransferOutCount" = #MISSING;

"TransferInCount" = #MISSING;
ENDFIX

FIX ([Scenario], [Version], [Year], @REMOVE(@LEVMBRS("Employee",0), @LIST(@CHILDREN("New Employees"), "Department General", "No Employee")), @LEVMBRS("Period", 0), "Local", "HSP_InputValue")

FIX(@REMOVE(@LEVMBRS("Entity",0), @LIST("No Entity")))

"Action" (

IF  ("Action" == 7)
 "No Entity"->"TransferOutCount" = "No Entity"->"TransferOutCount" + 1;
 ELSEIF  ("Action" == 8)
 "No Entity"->"TransferInCount" = "No Entity"->"TransferInCount" + 1;
ENDIF
);
Fix {[Scenario], [Version], [Year], @REMOVE(LEVMBRS("Employee", 0), LIST(CHILDREN("New Employees"), "Department General", "No Employee")), LEVMBRS("Period", 0), "No Entity", "Local", "HSP_InputValue")

"Unreconciled Transfers"(
  IF ("TransferInCount" == "TransferOutCount")
    IF ("TransferInCount" == 1 OR "TransferInCount" == #MISSING)
      #MISSING;
    ELSE
      0;
    ENDIF
  ELSEIF ("TransferInCount" > "TransferOutCount")
    1;
  ELSE
    -1;
  ENDIF
);
ENDFIX
Index

A
access permissions, setting up, 25
accessing
Performance Management Architect, 15
Planning, 15
Workforce Planning, 15
Workspace, 15
accounts
adding, 20
fixed and varying, 48
fixed and varying, loading data into, 14
predefined, 47
salary, 21
ACTUAL UDA, 13, 24
Add TBH Hourly business rule, 84
defined, 84
using, 23
Add TBH Salary business rule
defined, 85
using, 23
annual salary calculation, 21
Application Library, working with, 15
applications
adding in Performance Management Architect, 12
creating in Performance Management Architect, 12
creating, overview, 11
deploying in Performance Management Architect, 12
initializing Classic applications, 13
localizing, 23
selecting, 15
business model, 10
business rules
predefined, 84
working with, 23
C
calc scripts, predefined, 84
calculations, performing, 23
calendar, 11
Change Employee Status business rule, 20, 23
Classic applications
adding a salary adjustment type, example, 40
adding a Smart List member, 29
changing a pay rate, example, 31
changing bonus frequency, example, 33
creating a Workforce Planning application, 13
initializing, 13
loading information into, 14
creating
Classic applications, 13
Performance Management Architect applications, 12
CrossDim operator, 48
customizing
before you start, 27
business rules and calc scripts, 84
examples, 27
menus, 83
Smart List entries, 81
Smart Lists, 80
understanding effects of, 45
D
data
avoid losing, 13
loading using Application Link Adapter for Hyperion Planning, 15
BegBalance member, 47
benefits of Workforce Planning, 9
bonuses, changing the frequency, 32
loading, overview, 14
spreading, 20
data forms
   predefined, 45
   viewing and managing, 15
   working with, 22
department expenses, planning, 20
Department General member, 20
departments, maximum number of TBHs, 19
deploying Performance Management Architect
   applications, 12
Dimension Library, working with, 15

E
   elements of Workforce Planning, 45
Employee Type Smart List, 81
employees
   changing status, 20
   converting TBHs to hires, 19
   terminating, 20
   transferring, 17
   working with, 17
equity adjustment, example of adding, 34
   examples of customizing Workforce Planning, 27

F
   fixed accounts
      described, 48
      loading data into, 14
   formula expressions
      time balance, 21
      working with, 23
FT_PT Smart List, 81

H
   HealthPlan Smart List, 81
Hourly
      effect on Annual Salary calculation, 21
      Smart List entry, 82
      HspCustomMsgs template file, 22

I
   importing, 14
   initializing Workforce Planning applications, 11
      with Classic application administration, 13
   introduction, 9

L
   labels
      about localizing, 28, 29
      adding or modifying, 23
   loading data
      overview, 14
      using Application Link Adapter for Hyperion Planning, 14
      using DIM Adapter for Planning, 14
   localizing applications, 23
   logging on
      to Workforce Planning, 15
      to Workspace, 15

M
   maternity status, changing pay rate, 30
   member formulas
      predefined, 53
      working with, 23
   menus
      predefined, 83
      working with, 22
   metadata
      loading using DIM Adapter for Planning, 15
      loading, overview, 14

N
   Navigate menu, working with, 15

O
   overview, 9

P
   password, 15
   PayType Smart List, 82
   Performance Management Architect
      accessing, 15
      adding a salary adjustment type, example, 34
      adding a Smart List member, 28
      changing a pay rate, example, 31
      changing bonus frequency, example, 32
      creating an application, 12
      documentation for, 15
loading information from enterprise systems, 14
performance rating, example of adding a new, 27
Performance Smart List, 81
Position Smart List, 81
prerequisites, 10

Reconcile Employee Transfer data form
defined, 47
using, 19
Reconcile Transfer Report data form
defined, 47
using, 19

salary
account members described, 21
adding an adjustment type, 34
calculating annual salary, 21
example of changing pay rate, 30
SalaryBasis Smart List, 82
scenario, Actual, 24
security. See access permissions
setting up Workforce Planning, 9
Smart Lists
entries, 81
example of changing, 27
predefined, 80
working with, 22
spreading data
and custom time periods, 11
using system members, 20
Status Smart List, 81
status, changing for employees, 20
structure of Workforce Planning, 45
system members, spreading data with, 20

task lists, using for process management, 25
Tax Region Smart List, 81
TBHs
adding, 19
converting to hires, 19
increasing the maximum number, 19
terminating employees, 20
time balance spread formula expressions, 21
time periods, custom, 11
TimeBalanceSpread, 21, 23
To be Hired members. See TBHs
Transfer business rule
defined, 89
using, 18
Transfer In business rule
defined, 89
using, 18
Transfer Out business rule
defined, 91
using, 18
transferring employees
described, 17
reconciling transfers, 18

UDAs
ACTUAL UDA, 13, 24
working with, 24
Unreconciled Transfers business rule
defined, 91
using, 19
URL for Workspace, 15
user name, 15

varying accounts
described, 48
loading data into, 14

working with
Application Library, 15
Dimension Library, 15
Workspace, logging on, 15

Year Total, 47
years, supporting multiple, 11