## Contents

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

**Chapter 1. Overview and Initialization of Capital Asset Planning** ................................. 9
  - About Capital Asset Planning .......................................................... 9
  - Capital Asset Planning Capabilities ........................................... 9
  - Prerequisites .............................................................................. 10
  - Business Model ........................................................................ 10
  - Creating and Initializing Capital Asset Planning Applications ................. 10
    - Using Performance Management Architect Application Administration .. 11
    - Using Classic Application Administration ..................................... 12
  - Loading Information into Capital Asset Planning .................................. 13
  - Loading Information Using DIM Adapter for Planning .......................... 14
  - Logging On and Accessing Capital Asset Planning ............................... 15

**Chapter 2. Working with Capital Asset Planning** ............................................. 17
  - Working with Tangible and Intangible Assets ....................................... 17
    - Establishing Global Assumptions ................................................... 17
    - Managing Tangible Assets .......................................................... 18
    - Managing Intangible Assets ....................................................... 22
    - Adjusting Costs and In-Service Dates .......................................... 24
    - Reviewing Financial Impact Information ....................................... 25
  - Managing Access Permissions ....................................................... 28
  - Considerations for Working with Capital Asset Planning ....................... 28

**Chapter 3. Customizing Capital Asset Planning** .............................................. 31
  - About Customizing Applications .................................................... 31
  - Adding Asset Classes ................................................................. 31
    - With Performance Management Architect Applications .................. 32
    - With Classic Applications ......................................................... 32
  - Adding Custom Fields .................................................................. 33
  - Changing Cash Flow Staggered Allocations ....................................... 34
  - Adding Data Forms ..................................................................... 34
  - Adding Retirement Options for New Assets ...................................... 35
Documentation Accessibility

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.
Overview and Initialization of Capital Asset Planning

In This Chapter

- About Capital Asset Planning ................................................................. 9
- Creating and Initializing Capital Asset Planning Applications ................. 10
- Loading Information into Capital Asset Planning .................................... 13
- Loading Information Using DIM Adapter for Planning ............................ 14
- Logging On and Accessing Capital Asset Planning ................................. 15

About Capital Asset Planning

Oracle Hyperion Capital Asset Planning, Fusion Edition is a Web-based solution that helps you manage, prioritize, and plan for capital expenses.

Using Capital Asset Planning, you create a model of your organization's capital expenses, providing an effective way for decision makers and front-line managers to communicate throughout the request, justification, review, and approval process.

The Capital Asset Planning model is based on a 12-month calendar, and the default Capital Asset Planning calculations support multiple years. If your application uses custom time periods, you must modify time-related formulas to support the different time periods.

Capital Asset Planning Capabilities

- Assists with creating capital expense plans and submitting them for approval
- Allows timing and cost adjustment to capital expenses
- Establishes global assumptions for each asset class and sets calculation drivers
- Provides communication and notification to ensure a smooth and efficient request and approval process
- Includes scenario simulation, enabling accurate prediction of the impact of capital expense plans on cash flow, profit, and loss
- Includes asset transfers, facilitating effective and efficient asset utilization across departments
- Integrates with Oracle Hyperion Planning, Fusion Edition data for reconciliation, forecasting, and reporting
- Enables customizing the planning process to meet the needs of global enterprises
Integrates with other systems to load information: with flat files for Oracle Hyperion EPM Architect, Fusion Edition applications, and direct data loads from source systems using data integration for Classic applications

Prerequisites
Before you set up and manage Capital Asset Planning, you should understand:

- Planning (see the Oracle Hyperion Planning Administrator’s Guide or online help)
- Performance Management Architect functionality if you are using Performance Management Architect application administration (see the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide or online help)
- The Capital Asset Planning business model and structure (see “Business Model” on page 10) and Appendix A, “Capital Asset Planning Structure”)

Business Model
All companies create forward-looking plans to prepare for the future, aligning corporate resources—people and dollars—against strategies that leverage competitive market advantage. Through collaborative planning, departments coordinate and allocate the capital expenses required to augment the organization’s capacity.

Capital Asset Planning helps manage and prioritize capital expenses during budgeting and planning. With Capital Asset Planning, you can manage existing assets and plan for capital expenses such as asset purchases. You can also:

- Perform driver-based calculations to assess the impact of changes and additions on profit, cash flow, and funding
- Request and approve capital expense plans with appropriate justification

Capital Asset Planning assists with tasks such as planning for transfers and related expenses, improvements, impairments, retirements, replacements, and financial impact. It also helps corporate planners consolidate plans, prepare reports on capital expenses, and iterate plans to respond to changing conditions.

Creating and Initializing Capital Asset Planning Applications
This section describes how to create and initialize a new Capital Asset Planning application, using either Performance Management Architect application administration or Classic application administration.

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

- Dimensions and members
- Data forms
- Smart Lists
- Member formulas
- Business rules
- Menus

Before you initialize Capital Asset Planning:

- Study Appendix A, “Capital Asset 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 will be for planners to use Capital Asset Planning.
- Update dimension outlines to resolve differences between Capital Asset Planning member names and member names of other applications.
- Plan the structure of the Entity dimension if you intend to use both Capital Asset Planning and Oracle Hyperion Workforce Planning, Fusion Edition. Initializing Workforce Planning produces one No Entity member, whereas initializing Capital Asset Planning produces a few Entity members. You can change the Entity members after you initialize the applications.

**Using Performance Management Architect Application Administration**

Follow the steps in this section to use Performance Management Architect application administration to create a Capital Asset Planning application.

**Creating a Capital Asset Planning Application**

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

1. For instructions, see “Building Applications” in the *Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide*.
2. For Plan Type, select Capex, or, if the application will use both plan types, select Capex and Wrkforce.
Adding the Capital Asset Planning Plan Type to an Existing Planning Application

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

► To add the Capital Asset 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 Capital Asset Planning plan type, select Initialize, and then select Capital Asset or Workforce and Capital Asset if the application will use both plan types.

3. To confirm the initialization, click Yes.

Deploying a Capital Asset Planning Application to Planning

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

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

If you need to redeploy a Capital Asset 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 log on 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 Capital Asset Planning application and use Classic application administration, see the Oracle Hyperion Planning Administrator’s Guide. When using the Classic Application Wizard to create a Classic application, select the Capital Asset Planning plan type.

After creating a Capital Asset Planning Classic application or adding Capex as a plan type to an existing Planning application, you must initialize it.
Note: When you create a Classic Planning application for either the Capex or Wrkforce plan type, 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. 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 either 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 business rules to Hyperion Calculation Manager, you cannot use Oracle's Hyperion® Business Rules with that application.

To initialize Capital Asset Planning for Classic Planning:
1. Start and log on to Planning.
2. In Planning, select Administration, then Initialization, and then Capital Asset Planning. A message confirms when initialization is successful; Initialize Capital Asset Planning no longer displays on the Initialization menu.
3. Optional: If the application also uses the Workforce plan type, select Administration, then Initialization, and then Workforce.
5. Validate the model if you change your Capital Asset Planning application.
6. Refresh the application.

Loading Information into Capital Asset Planning

If your Capital Asset Planning application is new, you may want to load information, such as the existing account structure and properties, from enterprise systems:

- If you are using Performance Management Architect application administration, load information using a flat file (see the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide).
- If you are using Classic application administration, you can use Oracle's Hyperion® Data Integration Management Adapter for Planning (see “Loading Information Using DIM Adapter for Planning” on page 14).

You can also:
● Manually enter information into Capital Asset 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 Planning Administrator's Guide or online help.

To load information—data and metadata—using DIM Adapter for Planning:

1 Load metadata (for example, entities and accounts) into Capital Asset Planning.

Load Account 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 accounts. Verify the results, make any needed changes, and then load the entire dimension.

See Appendix A, “Capital Asset Planning Structure” for a list of Capital Asset 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 member properties into the Essbase database for the Capital Asset 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.

Loading Information Using DIM Adapter for Planning

After you install and configure DIM Adapter for Planning, you can install and configure adapters that enable you to retrieve and write data for other Oracle Hyperion Enterprise Performance Management System products. After you configure an adapter, you must configure an application connection in Workflow Manager before you can extract data from sources or write data into targets. See Hyperion Data Integration Management Adapter for Planning Online Help.

Note: You can load information using Oracle's Hyperion® Data Integration Management Adapter for Planning only for Classic applications.
Logging On and Accessing Capital Asset Planning


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

1. Ensure that the Web server is started and the Web application server is running in the Services panel.
2. In your browser, enter the URL for the EPM Workspace Log On page.
3. Enter your system user name and password.
4. Click Log On.
5. For Performance Management Architect application administration: to access Performance Management Architect, select an option from the Navigate menu.
   For example, select Navigate, then Administer, then Dimension Library or Application Library.
   See the Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide or online help.
6. For Classic Planning applications, select Navigate, then Applications, then Planning, and then select an application.
7. To open a data form: Select Administration, then Manage, and then Data Forms. From Data Form Folders, select Capital, then select a data form from Data Form.
8. To edit a data form: Open a data form (see the previous step), and then click Edit.
Working with Tierable and Intangible Assets

You can plan for new tangible or intangible expenditures and check the impact on profit and loss, cash flow, and balance sheets. You can also review expenditures and adjust the timing and cost of capital spending. In addition, you can perform actions on existing assets such as transfers, retirements, improvements, and impairments.

Predefined tasks include:

- Establishing global assumptions (see “Establishing Global Assumptions” on page 17)
- Managing tangible assets (see “Managing Tangible Assets” on page 18)
- Managing intangible assets (see “Managing Intangible Assets” on page 22)
- Reviewing financial impact information (see “Reviewing Financial Impact Information” on page 25)

Establishing Global Assumptions

Capital Asset Planning is a driver-based planning tool. You can set drivers by establishing global assumptions for each asset class (for example, buildings or machinery) or for all tangible or intangible assets.

Assumptions you can set:

- Useful life of assets
- Depreciation methods
- Depreciation conventions
- Amortization methods
- Insurance, repair, and maintenance expenses
- Depreciation rates for declining balance methods
Cash flow incidence, which determines cash flow impacts and allocations for asset purchases

Funding options for asset requests

➢ To establish global assumptions:


2. Set assumptions for the drivers.
   For permissible values, see “Predefined Smart Lists” on page 65.

Managing Tangible Assets

Managing tangible assets includes planning for new assets, making timing and cost adjustments, and performing actions on existing assets, such as asset improvements, transfers, and retirements.

Requesting Asset Purchases

Given the financial implications, asset purchases require major management decisions. Each request must be justified based on need, cost, cost benefit, and financial impact (profit and loss and cash flow).

➢ To request asset purchases:


2. Right-click an asset class, and select Add New Asset.
   If no assets exist, this message displays:
   There are no valid rows of data for this data form.
   If this occurs:
   a. Right-click the message.
   b. From the menu, select Add New Asset.

3. Enter asset details, such as asset class, description, number of requests (to plan for multiple asset requests of the same value), Asset CAR #, priority, justification, asset units, purchase date, in-service date, asset rate, salvage value, and physical location.

   Note: If you leave the in-service date blank, the in-service date is assumed to match the purchase date. The in-service date cannot precede the purchase date.

4. Click Finish.

5. View the impact of depreciation at the top of the data form.

6. Optional: You can
Review profit and loss impact
Review balance sheet impact
Review cash flow impact
Perform timing adjustments
Perform cost adjustments
Remove assets

Managing Existing Specified Assets

Planners and cost-center managers plan actions such as transfers, retirements, and improvements for tangible assets.

Adding Existing Assets

You can add existing assets:

- Manually using Capital Asset Planning
- Manually using Oracle Hyperion Smart View for Office, Fusion Edition (see Oracle Hyperion Smart View for Office User’s Online Help)
- Using Performance Management Architect with Performance Management Architect applications (see Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help)
- Loading information (see “Loading Information Using DIM Adapter for Planning” on page 14)

To add existing tangible assets to entities:

1. Open data form 03. Manage Existing Specified (see “Loading Information Using DIM Adapter for Planning” on page 14).
2. Select the entity Operations.
3. Right-click an asset class, and select Add Existing Asset.
   
   If no assets exist, this message displays:
   
   There are no valid rows of data for this data form.
   
   If this occurs:
   
   a. Right-click the message.
   
   b. Select Add Existing Asset.
   
4. Enter details such as asset class, description, number of requests, in-service date, asset units, asset rate, salvage value, purchase date, and physical location.

   Note: If you leave the in-service date blank, the in-service date defaults to the purchase date. The in-service date cannot precede the purchase date.
5 Click Finish.

The asset is added with the prefix Base SP, indicating that the asset is base-specified and not an improvement.

6 To view calculated details of assets:
   a. Right-click the line item and select Calculated Details.
   b. Run the calculated data business rule to view the aggregated numbers for depreciation, gross asset balances, accumulated depreciation, and asset-related expenses set in global assumptions.

7 Optional: Review cash flow impact.

Transferring Assets

To ensure optimum use of assets, facilities managers and cost-center managers can transfer fixed asset resources across departments.

To transfer assets:
1 Open data form 03. Manage Existing Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2 Right-click an asset and select Transfer Assets.
3 Enter details, such as asset class, line item, transfer from, transfer to, and transfer date.
4 Click Finish.

The asset is transferred and impacts expenses. To view the impact of asset transfer in the source and destination entry, right-click a line item, and select Calculated details.

Retiring Assets

When assets are retired, asset balances are terminated as of the retirement date, and losses or gains on sales or write-offs are calculated.

To retire assets:
1 Open data form 03. Manage Existing Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2 Right-click the asset, and select Retire Assets.
3 Enter details, such as asset class, line item, retire date, retire option, sale value or write-off, and retire cost.
4 Click Finish.

To view the impact of asset transfer in the source and destination entry, right-click a line item, and select Calculated details.
Removing Assets

When assets are removed, asset balances are terminated as of the removal date, and losses or gains on sales or write-offs are calculated.

To remove assets:
1. Open data form 03. Manage Existing Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2. Right-click the asset, and select Remove Assets.
3. Enter details, such as line item and asset class.
4. Click Finish.

Improving Assets

Planners and cost-center managers plan for improving assets (upgrading equipment, adding floor space, and so on). Improvements that increase asset cost become a part of the original asset cost. Each asset improvement adds a line item to the base line item. You can add up to three improvements to each base line item.

To improve assets:
1. Open data form 03. Manage Existing Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2. Right-click an asset (base line item), and select Improve Assets.
3. Enter details, such as description, asset units, asset rate, physical location, salvage value, physical value, purchase date, and in-service date.
4. Click Finish.
   A line item with the prefix IM is added below the original asset.

To view calculated details and aggregated values for improved assets:
1. Right-click an improved line item, and select Calculated Details.
2. Select the year from which the improvement is planned, and Click GO.
   The improvement details from the month of the In Service date display.
3. Select Base Line Item page view, and click GO.
   The aggregated value for the base line item and its improvement display.

Managing Unspecified Assets

Unspecified assets are not added at each asset level; the asset information is for reporting purposes only.
To manage unspecified assets:

1. Open data form 04. Manage Existing Unspecified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2. Select Entity Operations.
3. Select Property, Plant and Equipment Gross, then click GO.
4. In the data form, for each asset class and month, enter the unspecified asset values.
5. Select Accumulated Depreciation, and click GO. Click Finish to save the data.
6. In the data form, for each asset class and month, enter the accumulated depreciation and depreciation values for three years.
   a. Select values for department, scenario, and version.
   b. Click Launch. The account balances are rolled up.
8. Select Property, Plant and Equipment Gross, and click GO to view the rolled up net balance.

### Managing Intangible Assets

Financial managers can perform these actions on intangible assets and evaluate their impact on business performance:

- “Requesting Intangible Asset Purchases” on page 22
- “Adding Intangible Assets” on page 23
- “Impairing Intangible Assets” on page 24
- “Reviewing Financial Impact Information” on page 25

### Requesting Intangible Asset Purchases

To request intangible assets:

1. Open data form 05. New Intangible Asset Requests (see “Logging On and Accessing Capital Asset Planning” on page 15). A composite data form displays for creating intangible asset requests (bottom) and amortization and other related expenses (top).
2. Right-click an asset class, and select Add New asset.
   - If no assets exist, this message displays: There are no valid rows of data for this data form.
   - If this occurs:
a. Right-click the message.
b. Select Add New Asset.

3. Enter details such as asset class, description, number of requests (for multiple requests of same value), justification, acquisition cost, additional charge, purchase date, in-service date, and salvage value.

4. Click Finish.

5. To view amortization calculations:
   a. Right-click an asset line item, and select Profit and Loss Impact.
   b. In the amortization form, select the year, and click GO.

6. Optional: You can
   - Review profit and loss impact
   - Review balance sheet impact
   - Review cash flow impact
   - Calculate intangible
   - Remove assets

---

**Adding Intangible Assets**

To add intangible assets:

1. Open data form 06. Manage Existing Intangibles Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).

2. Select Operations for the entity.

3. Right-click an asset class, and select Add Existing Asset.

   If no assets exist, this message displays:
   
   There are no valid rows of data for this data form.

   If this occurs:
   a. Right-click the message.
   b. Select Add Existing Asset.

4. Enter details such as description, asset class, number of requests, acquisition cost, additional charge, in-service date, and salvage value.

5. Click Finish.

6. Optional: To view calculated details of the assets, right-click a line item, such as Base SP1, and select Calculated details.
Impairing Intangible Assets

To impair intangible assets:
1. Open data form 06. Manage Existing Intangibles Specified (see “Logging On and Accessing Capital Asset Planning” on page 15).
2. Right-click a line item, and select Impair Asset.
3. Enter details such as asset class, line item, impair date, fair value, impair option, and capitalize%.
4. Click Finish.
5. **Optional:** To view the impact of impairment, right-click a line item, and select Calculated details.

**Note:** If you select the capitalize option, the impairment value is posted to capital reserve. If you select the partial capitalize option, the impairment value is apportioned to the capital reserve, based on the capitalize%. Also, amortization is reduced from the month of impairment.

Adjusting Costs and In-Service Dates

After adding assets, you can adjust costs and in-service dates by moving capital expense spending plans to different periods—useful if you change the priority for assets or adjust costs or units based on better estimates. The basic cost and depreciation are recalculated when users right-click the line item and select Calculate Asset.

To adjust costs and in-service dates:
1. **Add an asset** (see “Working with Tangible and Intangible Assets” on page 17).
   a. Open the 02. New Asset Requests data form (see “Logging On and Accessing Capital Asset Planning” on page 15).
   b. Right-click an asset class, such as Machinery and Equipment, and select Add New Asset.
   c. Enter information for Asset Class, Number of Requests, Description, CAR #, Priority, Justification, Asset Units, Asset Rate, Salvage Value, Purchase Date, and In Service Date. Then click Finish.

   This adds the asset request, calculates and updates the basic cost, and calculates depreciation across the useful life of the asset. New asset requests are added as line items in the data form.
2. In the 02. New Asset Requests data form, update the number of units, for example, from 3 to 4.
3. Update the In Service Date, for example, from 4/1/07 to 6/1/07.
4. Click Save.

   Select the line item, then select Calculate Asset to recalculate the basic cost based on the new value. Depreciation and other balances are moved to the new period as appropriate.
5. **Optional:** To review changes, open data form 08. Capital Expenditure Summary (see “Logging On and Accessing Capital Asset Planning” on page 15). To view details of each asset, right-click an asset class.
On the line item, click **Profit and Loss Impact** to see the depreciation impact. Click **Cash Flow Impact** to see the cash flow impact.

## Reviewing Financial Impact Information

Financial impact reports show the effect of capital expenses and actions related to capital expenses on profit and loss, balance sheet, and cash flow. You can review reports by department or across departments, for the same asset class, all asset classes, or one asset.

Available report summaries:

- Capital expenditure summary
- Intangible expenditure summary
- Profit and loss impact
- Balance sheet impact
- Cash flow impact
- Depreciation summary
- Amortization summary
- Depreciation what ifs

### Capital Expenditure Summaries

To view capital expenditure summaries:

1. Log on to EPM Workspace and open the Capital data forms folder (see “Logging On and Accessing Capital Asset Planning” on page 15).
2. Select **Current Fiscal Year**, and then click **GO**.
3. Click **08. Capital Expenditure Summary**.

   For each asset class, asset request summaries display.

4. **Optional:** To view details of each asset, right-click an asset class, then select **Asset Details**.

   A drill-down data form displays with details of each asset.

5. **Optional:** To roll up assets, right-click an asset class, then select **Rollup Assets**.

### Intangible Expenditure Summaries

Intangible expenditure summaries detail the asset requests for each asset class and asset.

To view intangible expenditure summaries:

2. Select **Current Fiscal Year**, and then click **GO**.
In this view, only months that have data display.

3 **Optional:** To view details of each asset, right-click an asset class, and select **Intangible Details**. A drill-down data form displays with asset details.

### Reviewing Profit and Loss Impact Reports

For tangible assets, you can review the expected profit and loss impact of capital expenses.

To review profit and loss impact reports:

   
   A summary of profit and loss statements displays.

2. Select the asset class **Tangible Asset**, the entity **Operations**, and then either the line item **Total New** or **Total Existing**.
   
   - To review existing assets, select the line item **Total Existing**.
   
   - To review new assets select the line item **Total New**.

3. Select **Current Fiscal Year** for the year, and then click **GO**.

### Reviewing Balance Sheet Impact Reports

For tangible assets, you can review the anticipated balance sheets.

To review balance sheet impact reports:

   
   A summary of balance sheet statements displays.

2. Select the asset class **Tangible Asset**, the entity **Operations**, and then the line item **Total New** or **Total Existing**.
   
   - To review existing assets, select the line item **Total Existing**.
   
   - To review new assets select the line item **Total New**.
   
   - To review the consolidated balance sheet of tangible and intangible assets, select the asset class **Total Fixed Assets** and then the line item **Total Existing**.

3. Select **Current Fiscal Year** as the year, and then click **GO**.

### Reviewing Cash Flow Impact

For tangible assets, you can review the anticipated cash flow statements.
To review cash flow impact statements:

   
   A summary of cash flow statements displays.

2. Select **Tangible Asset** for asset class, **Operations** for entity, and either **Total New** or **Total Existing** for the line item.

   Cash flow impacts are affected only if you sell or improve existing assets.

3. Select **Total Fixed Assets** for asset class, and **Total Existing** for line item to review a consolidated balance sheet of tangible and intangible assets.

4. Select **Current Fiscal Year** for year, and click **GO**.

**Viewing Depreciation Summaries**

Depreciation summaries detail the depreciation of various asset classes.

To view depreciation summaries:


   A depreciation summary of various asset classes displays.

2. To view details of each asset, right-click an asset class, then select **Depreciation Details**.

   A drill-down data form with the depreciation details for each asset displays. In this view, only the months having data display.

**Viewing Amortization Summaries**

Amortization summaries detail asset class amortization.

To view amortization summaries:


   An amortization summary of various asset classes displays.

2. To view asset details, right-click an asset class, then select **Amortization Details**.

   A drill-down data form with the amortization details for each asset displays. In this view, only months that have data display.
Working with Depreciation What Ifs

To view depreciation summary data and perform what-if analysis:


   A composite data form displays with depreciation summary information (top) and global assumptions for depreciation (bottom).

2. Select Total New for asset class and Current Fiscal Year for year.

3. In the bottom data form, change an asset class depreciation method, and then save the data.

   For example, for Computers, change the depreciation method from SYD to SLN.

4. In the depreciation form (top), right-click an asset class, and then select Calculate Asset.

   For example, right-click Computers.

5. In Calculate Existing Asset, Click Finish.

   Similarly, in the global assumptions for depreciation data form (bottom), you can change the values for Useful Life and Depreciation Convention, and then recalculate.

Managing Access Permissions

You can control access to Capital Asset Planning—who can view, add, delete, or change information and access which department.

If planners can access a member, they can access its data. You can hide asset information by denying users or groups access to members or parents (assign the access type None). You can also screen information by withholding access to certain data forms.

- To set up users and groups, see Oracle Hyperion Enterprise Performance Management System User and Role Security Guide.

- To assign access permissions, see Oracle Hyperion Planning Administrator’s Guide.

Considerations for Working with Capital Asset Planning

When working with Capital Asset Planning, consider:

- Depreciation calculations for existing assets before the application period range are supported only for the SLN and SYD depreciation methods, not for the DB Year or DB Period depreciation method. For example, if the period range for the Capital Asset Planning application is Jan 2004 to Dec 2015, and the existing asset in-service date is 1/1/2000, depreciation calculations are supported only for the SLN and SYD methods.

- If the salvage value is set to 0 (zero), the DB Year or DB Period depreciation method may not produce the desired results. To produce correct depreciation calculations when using the DB Year depreciation method, Oracle recommends that the salvage value be set to at least 1% of the basic cost.
Timing adjustment does not work when it is based on a change of purchase date and staggered cash flow. This is because users can modify the staggered cash flow allocation percentage. To use timing adjustment in this case, calculate allocated cash flow percentages manually. Timing adjustment works correctly for other cash flow assumptions.

Users can make up to three improvements for each existing asset. To add additional improvements, the administrator must add the appropriate IM members.

The Capital Asset Planning model is based on a 12–month calendar. It is not a weekly model.

When planning transfers, ensure that users have appropriate access permissions to the source and destination entities.

For multicurrency applications, depreciation calculations use the base currency for the entity member calculated. If the currency override option is in effect, depreciation calculations use the currency of the entered value.

Capital Asset Planning does not restrict depreciation calculations for intangible assets or amortization calculations for tangible assets. Business administrators managing global assumptions should define drivers appropriately.
This chapter provides examples to help you understand how to customize Capital Asset Planning applications.

**About Customizing Applications**

Before customizing a Capital Asset Planning application, evaluate the predefined Capital Asset Planning model—described in Appendix A, “Capital Asset Planning Structure”—to identify which elements you can use and which elements you need to customize. Study the logic, formulas, and interrelationships of the predefined members. Whenever you modify the Capital Asset Planning model, validate the model to test modifications and refresh the application.

**Adding Asset Classes**

You may want to add an asset class to:

- Add an asset class that is not predefined (add tangible asset members under the Tangible Assets parent and add intangible assets members under the Intangible Assets parent)
Change the useful life of an asset set
Depreciate assets by using a different convention or method

This example adds a new member to the Asset Class dimension. Depending on whether you have a Performance Management Architect application or a Classic application, see “With Performance Management Architect Applications” on page 32 or “With Classic Applications” on page 32.

With Performance Management Architect Applications

To add asset classes to Performance Management Architect applications:
2. Select Navigate, then Administer, and then Dimension Library.
3. In Shared Library, select Asset Class.
4. Select Total Fixed Assets, and then select Tangible Assets.
5. Right-click Tangible Assets, then select Create Member, and then As Child.
6. Enter the member name, such as Heavy Machinery, in the New Member dialog box, and then click OK.
   The member is created as a child of Tangible Assets.
7. Select the new member name to view the member's properties, and then select Planning as the Category.
8. Set the Data Type to Currency, and then save the properties.
9. Select Navigate, then Administer, and then Application Library.
10. Right-click the Capital Asset Planning application you are running, such as CapExplan, and then select Edit.
11. Drag members from the Dimension Library to the application, and then click Save.
   The new asset class is created and is available in the Capital Asset Planning application.

With Classic Applications

To add asset classes in Classic applications:
1. In Planning, select Administration, then Manage, then Dimensions, and then select the Asset Class dimension.
2. In the Search box, enter Tangible Assets and then click either or .
3. Select Tangible Assets, and then click Add Child.
4  In Member Properties, name the new asset class member, set its Data Type to Currency, and then click Save.

After creating members, assign access permissions to them.

5  Select Administration, and then Data Source to refresh the database.

Adding Custom Fields

You may want to add custom fields. For example, to add a drop-down list for tracking purchase justifications, you can add a field called Purchase Reason to the New Asset request menu and data form.

This example:
- Modifies the Add Asset business rule to include a runtime prompt for the purchase reason.
- Adds a member—Purchase Reason Code—to data form 02A. New Asset Requests — Addition.
- Verifies the results.

To add a custom field to the launched Add Asset business rule:

1  Start Essbase Administration Services Console, expand Business Rules, Repository View, and Rules, and then select Add Asset business rule.

2  Select Local variable, and then select Add Task to add the variable.

3  Select Apply task.

4  In Save as, type PurchReason.

5  Select Runtime prompt, and then place PurchReason next to the PhyLocation variable.

6  In Source, after line “UOM” = 1;,, add this statement:

   "Purchase Reason Code" = [PurchReason]

   This section of the code now reads:

   "Purchase Date" = [PurchaseDate];
   "In Service Date" = [InServiceDate];
   "UOM" = 1; /*Default to Nos*/
   "Purchase Reason Code" = [PurchReason];

7  Click Save.

To add the Purchase Reason Code member to the New Assets Requests data form:

1  Start Planning, then select Administration, then Manage, then Data Forms and then select data form 02A. New Asset Requests — Addition from the data form folder Capital.

2  Select Edit.

3  Select Layout.

4  Select Column Definition 3, click Edit.
In Members, add Purchase Reason Code, and then click Submit.

Click Save.

To verify that Purchase Reason is added:
1. Open data form 02A. New Asset Requests — Addition.
2. Right-click an asset class, select Add New Asset, and then click Finish.

   Notice the Purchase Reason field.

### Changing Cash Flow Staggered Allocations

By default, cash flow staggered allocation is set to four months from the period of purchase. In this example, the staggered allocation is spread to six months by modifying the Add Asset business rule.

To change a cash flow staggered allocation:
1. Log on to Essbase Administration Services Console, expand Business Rules, Repository View, Rules, and then select the Add Asset business rule.
2. Change the variable, cashStaggeredPers to 6 from 4 at the following statement “IF (CashFlowIncidence== -1) |cashStaggeredPers = 6;”
3. Save the business rule.

To verify the change:
1. Open data form 02. New Asset Requests.
2. Right-click an asset class that has staggered cash flow incidence, such as Machinery and Equipment, and then select Add New Asset.
3. Enter new asset request details with an Asset Rate of 180000 (and other details of your choice), and then click Finish.
4. Right-click the new asset line item, and then select Cash Flow Impact.
5. In the Cash Flow Impact data form, select FY07, and then click GO.

   Verify that the basic cost (180000) has been allocated across six months instead of four, and each month’s cash flow is 30000.

### Adding Data Forms

This example shows how to add data forms that request new assets for each asset class. These data forms allow multiple requests for each asset class, and save and calculate in one step.

To add data forms:
1. In Planning, select Administration, then Manage, and then Data Forms.
Adding Retirement Options for New Assets

This example adds retirement options for new assets. It:

- Adds a menu item—RetireNewAsset—to the NewAssetMenu that launches a Retire New Asset business rule.
- Adds retirement-related members to the New Asset Requests — Addition data form.
- Verifies the results.

To add the new menu item, RetireNewAsset:

1. In Planning, select Administration, then Manage, and then Menus.
2. Select New Asset Menu, and then click Edit.
3. From NewAssetMenu, select 6.0 CalcDepr, and then click Add Sibling.
4. In Add Menu Item, set these values:
   - Menu item: RetireNewAsset
   - Label: Retire Asset
· Type: Business Rule
· Required Dimension: Line Item
· Plan Type: Capex
· Business Rules: Retire Asset
· View Type: Streamline View
· Window Title: Retire New Asset
· OK Button Label: OK
· Cancel Button Label: Cancel
· Select Launch in a separate window

5 Click Save and then close the dialog box.

➤ To update the New Asset Request — Addition data form to include the new retirement-related accounts:

1 Select data form 02A. New Asset Request – Addition and then click Edit.
2 Select Layout, and then click Edit Column Task.
3 Select Column Definition 3, and then click Edit.
4 In Member Selection Account, add these accounts to the selection:
   · Delayed start date
   · Reason delayed
   · Premature end date
   · Reason ended
5 Click Save and then Finish.

➤ To verify:

1 In Planning, select Administration, then Manage, then in Data Forms expand Forms, and then select Capital.
2 Select 02. New Asset Requests.
3 Right-click a line item, such as Honda Car, then select Retire Asset, and then enter details, such as asset class, line item, retire date, retire option, sale value for write-off, and retire costs.
4 Click Finish.
5 Verify that the data form has the premature end date and the reason for retirement.
6 Right-click the retired line item, and then select Profit and Loss Impact.
7 Select FY09 and then click GO. Verify that the depreciation is terminated in the period April FY09 and the Loss/(Gain) on Sale of Property, Plant, and Equipment is calculated.
Changing Labels

You may want to change field labels. This example renames the field Asset CAR # (Capital Acquisition Request number) to Asset AFE No (Approval for Expense number).

To change the runtime prompt:

1. Log on to Essbase Administration Services Console.
2. Expand Business Rules, Repository View and Global Variables.
3. Select the global variable AssetCAR.
4. Change the Prompt String value, for example, to Asset AFE No. (Approval for Expense).

Depending on whether you have a Performance Management Architect or Classic application, to rename the Cap No. member, see “With Performance Management Architect Applications” on page 32 or “With Classic Applications” on page 32.

With Performance Management Architect Applications

To rename the CAR No. member to AFE No. in Performance Management Architect Applications:

1. In Performance Management Architect, select Navigate, then Administer, and then Dimension Library.
2. Right-click Account, and then select Find Members.
3. In the Find Members dialog, select Name in Select By, enter CAR No in Value, and then click OK.
4. Right-click CAR No, and select Rename Member.
5. In Rename Member, enter AFE No., and then click OK.
6. Drag members from the Dimension Library to the application, and then click Save.
7. Redeploy the application. See Oracle Hyperion Enterprise Performance Management Architect Administrator’s Online Help.
   
   The renamed member is now available in the Capital Asset Planning application.
8. Ensure that all business rules and data forms that reference CAR No, reference AFE No instead.
9. Open 02. New Asset Requests, and then verify the field name change to AFE No. See “Logging On and Accessing Capital Asset Planning” on page 15

With Classic Applications

To rename the CAR No. member to AFE No. in Classic applications:

1. In Planning, select Administration, then Dimensions and then select the Account dimension.
In the Search box, enter **CAR No.** and then click either 

or .

3 Click **Edit.**

4 In **Member Properties**, change the Name to **AFE No.**, and then click **Save.**

5 Select **Administration**, then **Manage Database**, and then select **Database** under **Refresh Database Options.**

6 Ensure that all business rules that reference **CAR No.**, reference **AFE No** instead.

7 Select **Administration**, then **Manage** then **Data Forms**, and then expand **Data Forms** and select **Capital.**

8 Select **02. New Asset Requests**, and then verify the field name change to **AFE No.**

### Showing Hidden Fields

Based on business needs, you can show or hide data form fields. This example makes visible the hidden fields defined for the Add Asset business rule:

- Installation
- Freight
- Taxes %
- Additional charges
- Retirement obligation

To show fields:

1 Log on to Oracle Essbase Administration Services Console.
2 Expand **Business Rules, Repository View, and Rules.**
3 Select the **Add Asset** business rule.
4 Select **Variables.**
5 Select **Run-Time Prompts.**
6 Clear **Hide for these fields:**
   - Installation
   - Freight
   - Taxes %
   - Additional charges
   - Retirement obligation
7 Save the business rule.
8 In Planning, select **Administration**, then **Manage**, then **Data Forms**, and then expand **Data Forms** and select **Capital.**
9 Select **02. New Asset Requests.**
10 Right-click an asset class, such as **Machinery and Equipment**, in the bottom of the data form, and then select **Add New Asset**.

11 In **Add New Asset**, verify the changed fields are available.

**Enabling Assumptions at the Entity Level**

When using multiple entities, you must update data forms and business rules to ensure that business rules apply to different departments. See “**Updating Data Forms to Use Multiple Entities**” on page 39 and “**Updating Business Rules to Use Multiple Entities**” on page 39.

**Updating Data Forms to Use Multiple Entities**

To update data forms to use multiple entities:

1 In **Planning**, edit these two data forms to update the Point of View dimensions: 01. Global Capital Assumptions and 15A. Global Capital Assumptions - Depreciation only.
   a. Select **Administration**, then **Manage**, and then **Data Forms**.
   b. Select the data form to edit, and then click **Edit**.
   c. Select **Layout**.
   d. In **Page** select the **Entity** dimension, then **Descendants(Entity)** and then select a member using a relationship such as **Descendants(Entity)**.
   e. Click **Save**.

2 Update business rules (see “**Updating Business Rules to Use Multiple Entities**” on page 39).

**Updating Business Rules to Use Multiple Entities**

When using multiple entities, update the Capital Asset Planning business rules as described here. To ensure that global assumptions apply to different departments, global assumptions must be stored in multiple members of the Entity dimension.

For instructions about working with business rules, see **Hyperion Business Rules Administrator’s Web Help** and **Hyperion Business Rules Web Launcher User’s Web Help**.

**Note:** When planning for transfers, the calculated details for depreciation and other related expenses are based on assumptions set for the source entity. If assumptions for the destination entity are different, you must calculate assets in the destination entity to see the impact.

To update business rules to use multiple entities:

1 Update data forms as described in “**Updating Data Forms to Use Multiple Entities**” on page 39.
Using Administration Services, update the Capital Asset Planning business rules to remove the “No Entity” reference. Doing so enables business rules to vary by Entity dimension.


Appendix A, “Capital Asset Planning Structure” shows each business rule formula.

Example: Updating “No Entity” in the Add Asset Business Rule

Before:

"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";

After:

"Useful Life (in Years)" = "No Scenario"->"No Version"->"Global"->"Useful Life (in Years)";

Example: Updating “No Entity” in the AddExistAsset Business Rule

Before:

deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
deprConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";

After:

deprMethod = "No Year"->"No Scenario"->"No Version"->"Global"->"Depreciation Method";
deprConvention = "No Year"->"No Scenario"->"No Version"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"Global"->"Repairs %";
Adding a Project Dimension
You can add a Project dimension with information that fits your business needs. See:

- “Creating the Project Dimension” on page 41
- “Updating Data Forms for the Project Dimension” on page 42
- “Updating Business Rules for the Project Dimension” on page 42

Creating the Project Dimension
Depending on whether you have a Performance Management Architect application or a Classic application, see “With Performance Management Architect Applications” on page 41 or “With Classic Applications” on page 42.

With Performance Management Architect Applications

➤ To create a Project dimension in Performance Management Architect Applications:

1. In Performance Management Architect Shared Library, create a generic **Project** dimension and a Dimension Alias with these properties:
   - **Dimension Name**: Project
   - **Dimension Class**: Generic
   - **Dimension Alias**: Project

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

2. Under the **Project** dimension, create a member called **No Project** to store global assumptions. See Oracle Hyperion Enterprise Performance Management Architect Administrator's Online Help.

3. Add other members necessary to the **Project** dimension. See Oracle Hyperion Enterprise Performance Management Architect Administrator's Online Help.

4. Drag the **Project** dimension into the Performance Management Architect application you want to deploy.

5. Deploy the Capital Asset Planning application (see Oracle Hyperion Enterprise Performance Management Architect Administrator's Online Help).

   You can also load the Project dimension and its members from an external flat file and refresh the Capital Asset Planning application.

6. Update data forms and business rules (see “Updating Data Forms for the Project Dimension” on page 42 and “Updating Business Rules for the Project Dimension” on page 42).
With Classic Applications

To create a Project dimension for Classic applications:

1. In Planning, select Administration, then Manage, and then Dimensions.
2. Click Add Dimension.
3. Name the dimension Project, then select Apply Security (so you can assign access permissions to this dimension), and then accept the other defaults.
4. Click Save.
5. Update data forms and business rules (see “Updating Data Forms for the Project Dimension” on page 42 and “Updating Business Rules for the Project Dimension” on page 42).

Updating Data Forms for the Project Dimension

After creating a Project dimension, update data forms.

To update data forms to include the Project dimension:

1. In Planning, edit the Point of View dimensions for these data forms: 01. Global Capital Assumptions, and 15A. Global Capital Assumptions - Depreciation only.
   a. Select Administration, then Manage, and then Data Forms.
   b. Select the data form, and then click Edit.
   c. Select Layout.
   d. In Point of View, select No Project as the Project member.
   e. Click Save and then Finish.
2. Edit the other Capital Asset Planning data forms to select Descendant members for the Project dimension:
   a. Select Administration, then Manage, and then Data Forms.
   b. Select the data form, and then click Edit.
   c. Select Layout.
   d. In Page, for the Project member, select members by relationship for Project, such as Descendants(Project).
   e. Click Save, and then Finish.
3. Update business rules to enable storing global assumptions in multiple entities (see “Updating Business Rules for the Project Dimension” on page 42).

Updating Business Rules for the Project Dimension

After creating the Project dimension and updating data forms, update the Capital Asset Planning business rules.
To update business rules for the Project dimension:

1 Using Administration Services, create a Global Variable called **Project** with these properties:
   - **Name**: Project
   - **Type**: Member
   - **Dimension**: Project
   - **Limits**: `@Descendants("Dimension")`, where "Dimension" is the limit you want to set, for example, `@Descendants("Project")`
   - **Usage Type**: Runtime prompt
   - **Prompt String**: Project

2 Update the Capital Asset Planning business rules to include the Project member:
   - Add the [Project] member to every place the FIX statement appears in the business rules.
   - Update references to global assumptions in the business rules by adding No Project to them.

3 You must update these business rules:


   Appendix A, “Capital Asset Planning Structure” lists the formulas for the business rules.

---

**Example: Updating the Add Asset Business Rule**

**Example of Including the Project Dimension**

```
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"-> "No Project"->"Global"->"Useful Life (in Years)";
"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"-> "No Project"->"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"-> "No Project"->"Global"->"Funding Incidence";
cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Cash Flow Incidence";
fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Funding Incidence";
deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Depreciation Method";
deprConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Depreciation Convention";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Repairs %";
fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Funding %";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Insurance %";
```
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"No Project"->"Global"->"Maintenance %";

Example of Adding Project When it is Referenced for Global Assumptions

FIX(@CHILDREN("Total New"), [Hidden_Scenario], [Hidden_Version], [Project], [Department], [AssetClass], "Local", "HSP_InputValue")  
FIX([Hidden_Scenario], [Hidden_Version],[Project], [Department], "Local", "HSP_InputValue")  
    CALC DIM ("Period");

Example: Updating the Remove Asset Business Rule

Example of Adding Project when it is Referenced for Global Assumptions

SET UPDATECALC OFF;
SET AGGMISSG ON;

FIX ([Hidden_Scenario], [Hidden_Version],[Project], [Department], [AllAssetClass]) 
    CLEARDATA [LocalLineItem];
ENDFIX

FIX ([Hidden_Scenario], [Hidden_Version],[Project], [Department])  
    @ANCESTORS([LocalLineItem]);
    @ANCESTORS([AllAssetClass]);
ENDFIX

Data Forms

When you modify data forms, ensure that the changes are synchronized with business logic (such as business rules, member formulas, and outline structure). Similarly, when you modify business logic, review your data forms.

Smart Lists and Menus

Calculations based on modified Smart Lists may be incorrect. Therefore, you may need to adjust data in the database to reflect the modified values. You can delete predefined menus without impacting predefined calculations.

For Smart Lists and menus, Capital Asset Planning provides predefined labels that are set as resource strings in the HspCustomMsgs template file. If you add or change Smart Lists and menus to your application, you should add the labels to the HspCustomMsgs file. You can localize the labels in your application by updating the localized versions of the HspCustomMsgs file. Modifying the HspCustomMsgs file replaces the default values installed with Capital Asset Planning. For instructions about modifying labels or preparing files for localization, see Oracle Hyperion Planning Administrator’s Guide.

To review the implications of changing Smart Lists, and for a list of predefined Smart Lists and menus, see Appendix A, “Capital Asset Planning Structure”
Member Formulas

Capital Asset Planning member formulas include formula expressions, such as \[\text{Formula Name}\]. Oracle recommends that you include formula expressions in formulas. For Performance Management Architect applications, see Oracle Hyperion Enterprise Performance Management Architect Administrator’s Guide and for Classic applications, see Oracle Hyperion Planning Administrator’s Guide.

Business Rules

Planners use business rules to perform calculations on asset data when:

- Adding assets to departments. For example, when assets are purchased, planners use the Create Asset business rule to add the assets to a department.
- Transferring assets in and out of departments. For example, planners use the Transfer business rule to transfer assets between departments.

See Appendix A, “Capital Asset Planning Structure” for a list of business rules and their formulas.

You can use Administration Console in Oracle Essbase Administration Services to create, validate, launch, and manage business rules, including the predefined Capital Asset Planning business rules. (See Hyperion Business Rules Administrator’s Guide.)

This appendix lists the predefined elements of Capital Asset Planning, and helps you determine how elements affect each other.

### Predefined Data Forms

Planners use data forms to work with asset information.

**Table 1  Predefined Data Forms**

<table>
<thead>
<tr>
<th>Data Form Name</th>
<th>Axis Definitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>01. Global Capital Assumptions</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Account</td>
</tr>
<tr>
<td></td>
<td>Page: N/A</td>
</tr>
<tr>
<td></td>
<td>POV: Entity, Line Item, Period, Scenario, Version, Year, Currency</td>
</tr>
<tr>
<td>02A. New Asset Requests - Addition</td>
<td>Row: Asset Class, Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Period, Year, Currency</td>
</tr>
<tr>
<td>Data Form Name</td>
<td>Axis Definitions</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>02B. Total Expense Impact</td>
<td>Row: Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Account, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Asset Class, Currency</td>
</tr>
<tr>
<td>02C. Cash Flow Impact - Line Item Details</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>03. Manage Existing Specified</td>
<td>Row: Asset Class, Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Period, Year, Currency</td>
</tr>
<tr>
<td>03A. Existing Specified Drill Down</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>03B. Existing Specified Expenses</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>04. Manage Existing Unspecified</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Account, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Line Item, Currency</td>
</tr>
<tr>
<td>05A. New Intangible Requests - Additions</td>
<td>Row: Asset Class, Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Period, Year, Currency</td>
</tr>
<tr>
<td>Data Form Name</td>
<td>Axis Definitions</td>
</tr>
<tr>
<td>----------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>06. Manage Existing Intangibles Specified</td>
<td>Row: Asset Class, Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Period, Year, Currency</td>
</tr>
<tr>
<td>06A. Existing Intangible Specified Drill Down</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>06B. Existing Specified Expenses - Intangibles</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>07. Manage Existing Intangibles Unspecified</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Account, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Line Item, Currency</td>
</tr>
<tr>
<td>08. Capital Expenditure Summary</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Account, Currency</td>
</tr>
<tr>
<td>08A. Capital Expenditure Summary - Line Item details</td>
<td>Row: Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Year, Period, Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>09. Intangible Expenditure Summary</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Account, Currency</td>
</tr>
<tr>
<td>Data Form Name</td>
<td>Axis Definitions</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>09A. Intangible Account Balance</td>
<td>Row: Account, Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>09B. Intangible Asset Balances</td>
<td>Row: Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Year, Period, Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>10. Profit and Loss Impact</td>
<td>Row: Account</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
<tr>
<td>13. Depreciation Summary</td>
<td>Row: Asset Class</td>
</tr>
<tr>
<td></td>
<td>Column: Period</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Scenario, Version, Line Item, Year</td>
</tr>
<tr>
<td></td>
<td>POV: Account, Currency</td>
</tr>
<tr>
<td>13A. Depreciation Summary - Line Item</td>
<td>Row: Line Item</td>
</tr>
<tr>
<td></td>
<td>Column: Year, Period, Account</td>
</tr>
<tr>
<td></td>
<td>Page: Entity, Asset Class, Scenario, Version</td>
</tr>
<tr>
<td></td>
<td>POV: Currency</td>
</tr>
</tbody>
</table>
### Predefined Composite Data Forms

Planners use composite data forms to see information from multiple views.

#### Table 2  Predefined Composite Data Forms

<table>
<thead>
<tr>
<th>Composite Data Form Name</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>02. New Asset Requests</td>
<td>• 02B. Total Expense Impact</td>
</tr>
<tr>
<td></td>
<td>• 02A. New Asset Requests - Addition</td>
</tr>
<tr>
<td>05. New Intangible Asset Requests</td>
<td>• 02B. Total Expense Impact</td>
</tr>
<tr>
<td></td>
<td>• 05A. New Intangible Requests - Additions</td>
</tr>
<tr>
<td>15. Depreciation What ifs</td>
<td>• 13. Depreciation Summary</td>
</tr>
<tr>
<td></td>
<td>• 15A. Global Capital Assumptions - Depreciation only</td>
</tr>
</tbody>
</table>

### Predefined Accounts

For Oracle Hyperion EPM Architect, Fusion Edition applications, you can view the properties and hierarchy for predefined account members by opening an application and selecting dimensions and members in the Dimension Library. Properties are displayed in the Property Grid. For Classic applications, in Oracle Hyperion Planning, Fusion Edition, select Administration, then Dimensions, then select the Account dimension.
<table>
<thead>
<tr>
<th>Account</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Accounts (Capital Expense)</td>
<td></td>
</tr>
<tr>
<td>Assets</td>
<td></td>
</tr>
<tr>
<td>Property, Plant and Equipment Net</td>
<td></td>
</tr>
<tr>
<td>Property, Plant and Equipment Gross</td>
<td></td>
</tr>
<tr>
<td>Accumulated Depreciation</td>
<td></td>
</tr>
<tr>
<td>Intangible Assets, Net</td>
<td></td>
</tr>
<tr>
<td>Intangible Assets - Indefinite</td>
<td></td>
</tr>
<tr>
<td>Intangible Assets Finite, Net</td>
<td></td>
</tr>
<tr>
<td>Intangible Assets Finite, Gross</td>
<td></td>
</tr>
<tr>
<td>Accumulated Amortization</td>
<td></td>
</tr>
<tr>
<td>Liabilities</td>
<td></td>
</tr>
<tr>
<td>Long Term Debt</td>
<td></td>
</tr>
<tr>
<td>Asset Retirement Liability</td>
<td></td>
</tr>
<tr>
<td>Capital Reserve</td>
<td></td>
</tr>
<tr>
<td>Cash Flows</td>
<td></td>
</tr>
<tr>
<td>Net Cash Flows</td>
<td></td>
</tr>
<tr>
<td>Cash Outflow from Capital Additions</td>
<td></td>
</tr>
<tr>
<td>Cash Inflow from Funding</td>
<td></td>
</tr>
<tr>
<td>Proceeds from Sale of Property, Plant and Equipment</td>
<td></td>
</tr>
<tr>
<td>Proceeds from Sale of Intangibles</td>
<td></td>
</tr>
<tr>
<td>Expenses (Capital)</td>
<td></td>
</tr>
<tr>
<td>Fixed Assets Related Expenses</td>
<td></td>
</tr>
<tr>
<td>Insurance</td>
<td></td>
</tr>
<tr>
<td>Maintenance</td>
<td></td>
</tr>
<tr>
<td>Repairs</td>
<td></td>
</tr>
<tr>
<td>Other Operational Costs</td>
<td></td>
</tr>
<tr>
<td>Depreciation</td>
<td></td>
</tr>
<tr>
<td>Amortization</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>---</td>
</tr>
<tr>
<td>Loss/(Gain) on Sale of Property, Plant and Equipment</td>
<td></td>
</tr>
<tr>
<td>Loss/(Gain) on Sale of Intangibles</td>
<td></td>
</tr>
<tr>
<td>Impairment of Assets</td>
<td></td>
</tr>
<tr>
<td>Retirement Expenses</td>
<td></td>
</tr>
<tr>
<td>Capital Assumptions</td>
<td></td>
</tr>
<tr>
<td>Capital Driver Assumptions</td>
<td></td>
</tr>
<tr>
<td>Useful Life (in Years)</td>
<td></td>
</tr>
<tr>
<td>Depreciation Method</td>
<td></td>
</tr>
<tr>
<td>Depreciation Convention</td>
<td></td>
</tr>
<tr>
<td>Cash Flow Incidence</td>
<td></td>
</tr>
<tr>
<td>Funding %</td>
<td></td>
</tr>
<tr>
<td>Funding Incidence</td>
<td></td>
</tr>
<tr>
<td>Insurance %</td>
<td></td>
</tr>
<tr>
<td>Repairs %</td>
<td></td>
</tr>
<tr>
<td>Maintenance %</td>
<td></td>
</tr>
<tr>
<td>Amortization Method</td>
<td></td>
</tr>
<tr>
<td>Depreciation Rate</td>
<td></td>
</tr>
<tr>
<td>Asset Properties</td>
<td></td>
</tr>
<tr>
<td>Asset ID</td>
<td></td>
</tr>
<tr>
<td>Asset Description</td>
<td></td>
</tr>
<tr>
<td>CAR No.</td>
<td></td>
</tr>
<tr>
<td>UOM</td>
<td></td>
</tr>
<tr>
<td>Purchase Reason Code</td>
<td></td>
</tr>
<tr>
<td>Justification</td>
<td></td>
</tr>
<tr>
<td>Priority</td>
<td></td>
</tr>
<tr>
<td>Physical Location</td>
<td></td>
</tr>
<tr>
<td>Asset Control Properties</td>
<td></td>
</tr>
<tr>
<td>Asset Status</td>
<td></td>
</tr>
<tr>
<td>Account</td>
<td>Purchase Date</td>
</tr>
</tbody>
</table>
|--------------------------|---------------|-----------------|--------------------|-----------------|-----------------|-------------------|-----------------|-------------------|-------------------|---------------------|-------------------|---------------------|---------|--------------|--------|---------------|-----------|-------------------|----------------------|-------------------|---------------------|--------------------|---------------------|--------|-----------------|------------------|---------------------|----------
Predefined Asset Classes

Table 4  Predefined Asset Classes

<table>
<thead>
<tr>
<th>Asset Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Asset</td>
</tr>
<tr>
<td>Total Fixed Assets</td>
</tr>
<tr>
<td><strong>Asset Class</strong></td>
</tr>
<tr>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Tangible Assets</td>
</tr>
<tr>
<td>Buildings</td>
</tr>
<tr>
<td>Land</td>
</tr>
<tr>
<td>Land Improvements</td>
</tr>
<tr>
<td>Machinery and Equipment</td>
</tr>
<tr>
<td>Furniture and Fixtures</td>
</tr>
<tr>
<td>Office Equipment</td>
</tr>
<tr>
<td>Vehicles</td>
</tr>
<tr>
<td>Capitalized Interests</td>
</tr>
<tr>
<td>Computers</td>
</tr>
<tr>
<td>Computer Software Costs</td>
</tr>
<tr>
<td>Internal Use Computer Software</td>
</tr>
<tr>
<td>Capital Leased Assets</td>
</tr>
<tr>
<td>Construction In Progress</td>
</tr>
<tr>
<td>Leasehold Improvements</td>
</tr>
<tr>
<td>Other Property, Plant and Equipment</td>
</tr>
<tr>
<td>Intangible Assets</td>
</tr>
<tr>
<td>Brand Valuation</td>
</tr>
<tr>
<td>Copyrights</td>
</tr>
<tr>
<td>Computer Software</td>
</tr>
<tr>
<td>Customer Lists</td>
</tr>
<tr>
<td>Customer Relationships</td>
</tr>
<tr>
<td>Contractual Rights</td>
</tr>
<tr>
<td>Distribution Rights</td>
</tr>
<tr>
<td>Franchise Rights</td>
</tr>
<tr>
<td>Goodwill</td>
</tr>
<tr>
<td>Licenses</td>
</tr>
<tr>
<td>License Agreements</td>
</tr>
</tbody>
</table>
### Asset Class
- Non Compete Agreements
- Media Content
- Product Development & Improvements
- Patents
- Royalty
- Trademarks
- Trade Names
- Trade Secrets
- Other Intangible Assets
- Global Assumption Assets
- All Assets
- All Tangibles
- All Intangibles
- Capital Work In Process

## Predefined Line Items

### Table 5  Predefined Line Items

<table>
<thead>
<tr>
<th>Line Item</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Global</td>
<td></td>
</tr>
<tr>
<td>No Line Item</td>
<td></td>
</tr>
<tr>
<td>Total All</td>
<td></td>
</tr>
<tr>
<td>Total New</td>
<td></td>
</tr>
<tr>
<td>Line Item 1</td>
<td></td>
</tr>
<tr>
<td>Line Item 2</td>
<td></td>
</tr>
<tr>
<td>Line Item 3</td>
<td></td>
</tr>
<tr>
<td>Line Item 4</td>
<td></td>
</tr>
<tr>
<td>Line Item 5</td>
<td></td>
</tr>
<tr>
<td>Line Item 6</td>
<td></td>
</tr>
</tbody>
</table>
Line Item

Line Item 7
Line Item 8
Line Item 9
Line Item 10
Line Item 11
Line Item 12
Line Item 13
Line Item 14
Line Item 15
Line Item 16
Line Item 17
Line Item 18
Line Item 19
Line Item 20
Line Item 21
Line Item 22
Line Item 23
Line Item 24
Line Item 25
Line Item 26
Line Item 27
Line Item 28
Line Item 29
Line Item 30
Total Existing
Total Specified
SP1
Base SP1
<table>
<thead>
<tr>
<th>Line Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>IM1 SP1</td>
</tr>
<tr>
<td>IM2 SP1</td>
</tr>
<tr>
<td>IM3 SP1</td>
</tr>
<tr>
<td>SP2</td>
</tr>
<tr>
<td>Base SP2</td>
</tr>
<tr>
<td>IM1 SP2</td>
</tr>
<tr>
<td>IM2 SP2</td>
</tr>
<tr>
<td>IM3 SP2</td>
</tr>
<tr>
<td>SP3</td>
</tr>
<tr>
<td>Base SP3</td>
</tr>
<tr>
<td>IM1 SP3</td>
</tr>
<tr>
<td>IM2 SP3</td>
</tr>
<tr>
<td>IM3 SP3</td>
</tr>
<tr>
<td>SP4</td>
</tr>
<tr>
<td>Base SP4</td>
</tr>
<tr>
<td>IM1 SP4</td>
</tr>
<tr>
<td>IM2 SP4</td>
</tr>
<tr>
<td>IM3 SP4</td>
</tr>
<tr>
<td>SP5</td>
</tr>
<tr>
<td>Base SP5</td>
</tr>
<tr>
<td>IM1 SP5</td>
</tr>
<tr>
<td>IM2 SP5</td>
</tr>
<tr>
<td>IM3 SP5</td>
</tr>
<tr>
<td>SP6</td>
</tr>
<tr>
<td>Base SP6</td>
</tr>
<tr>
<td>IM1 SP6</td>
</tr>
<tr>
<td>IM2 SP6</td>
</tr>
<tr>
<td>IM3 SP6</td>
</tr>
<tr>
<td>Line Item</td>
</tr>
<tr>
<td>------------</td>
</tr>
<tr>
<td>SP7</td>
</tr>
<tr>
<td>Base SP7</td>
</tr>
<tr>
<td>IM1 SP7</td>
</tr>
<tr>
<td>IM2 SP7</td>
</tr>
<tr>
<td>IM3 SP7</td>
</tr>
<tr>
<td>SP8</td>
</tr>
<tr>
<td>Base SP8</td>
</tr>
<tr>
<td>IM1 SP8</td>
</tr>
<tr>
<td>IM2 SP8</td>
</tr>
<tr>
<td>IM3 SP8</td>
</tr>
<tr>
<td>SP9</td>
</tr>
<tr>
<td>Base SP9</td>
</tr>
<tr>
<td>IM1 SP9</td>
</tr>
<tr>
<td>IM2 SP9</td>
</tr>
<tr>
<td>IM3 SP9</td>
</tr>
<tr>
<td>SP10</td>
</tr>
<tr>
<td>Base SP10</td>
</tr>
<tr>
<td>IM1 SP10</td>
</tr>
<tr>
<td>IM2 SP10</td>
</tr>
<tr>
<td>IM3 SP10</td>
</tr>
<tr>
<td>Total Unspecified</td>
</tr>
<tr>
<td>Unspecified</td>
</tr>
<tr>
<td>Adj SP1</td>
</tr>
<tr>
<td>Adj SP2</td>
</tr>
<tr>
<td>Adj SP3</td>
</tr>
<tr>
<td>Adj SP4</td>
</tr>
<tr>
<td>Adj SP5</td>
</tr>
<tr>
<td>Adj SP6</td>
</tr>
</tbody>
</table>
**Predefined Member Formulas**

Member formulas are used for capital expense calculations.

**Cal TP-Index**

**Formula**

\[\text{[OpenInputValueBlock]} \, \text{[CalendarTPIndex]} \, \text{[CloseInputValueBlock]}\]

**Description**

System member that returns the time period index based on a calendar year. Example: Fiscal year starting in July: Jan=1, Feb=2, Mar=3, and so on.

**Capital Driver Assumptions**

**Formula**

`#MISSING;`
Description
Label account to group assumptions.

**First Date**

**Formula**

```
[OpenInputValueBlock] [FirstDate] [CloseInputValueBlock]
```

**Description**

System member that returns a value that represents the first date for the application.

**Fiscal TP-Index**

**Formula**

```
[OpenInputValueBlock] [FiscalTPIndex] [CloseInputValueBlock]
```

**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.

**NumPeriods**

**Formula**

```
[OpenInputValueBlock] [NumberOfPeriodsInYear]; [CloseInputValueBlock]
```

**Description**

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

**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**

**Formula**

```
[OpenInputValueBlock] [TimeBalanceFormula("Spread_454")]
[CloseInputValueBlock]
```
Description
System member that returns a spread factor following 4 week, 5 week, 4 week quarter.

**Spread_544**

Formula

```
[OpenInputValueBlock] [TimeBalanceFormula("Spread_544")]
[CloseInputValueBlock]
```

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

**Spread_Actual_365**

Formula

```
[OpenInputValueBlock] [TimeBalanceFormula("Spread_Actual_365")]
[CloseInputValueBlock]
```

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

**Spread_Actual_Actual**

Formula

```
[OpenInputValueBlock] [TimeBalanceFormula("Spread_Actual_Actual")]
[CloseInputValueBlock]
```

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

**Spread_Average**

Formula

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

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

**Spread_Balance**

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

**Spread_First**

Formula

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

**Spread_Flow**

Formula

Description
System member that returns a spread factor following the normal flow pattern.

**TP-Date**

Formula

Description
System member that returns a value that represents the date corresponding to the time period.

**YR-Index**

Formula

Description
System member that returns a value that represents the year offset from the first year in the application.
**Predefined Smart Lists**

Planners use Smart Lists in data forms to work with asset data. Associated account members are listed where applicable.

<table>
<thead>
<tr>
<th>Smart List</th>
<th>Associated Account Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>DepMethod</td>
<td>Depreciation Method</td>
</tr>
<tr>
<td>PhysicalLocation</td>
<td>Physical Location</td>
</tr>
<tr>
<td>AssetStatus</td>
<td>Asset Status</td>
</tr>
<tr>
<td>DepConv</td>
<td>Depreciation Convention</td>
</tr>
<tr>
<td>TangibleFlag</td>
<td>None</td>
</tr>
<tr>
<td>CashFlowIncidence</td>
<td>Cash Flow Incidence, Funding Incidence</td>
</tr>
<tr>
<td>RetireOptions</td>
<td>Retirement Options</td>
</tr>
<tr>
<td>AmortizationMethod</td>
<td>Amortization Method</td>
</tr>
<tr>
<td>DelayReasons</td>
<td>Reason delayed</td>
</tr>
<tr>
<td>EndReasons</td>
<td>Reason ended</td>
</tr>
<tr>
<td>AssetPriority</td>
<td>Priority</td>
</tr>
<tr>
<td>AssetUOM</td>
<td>UOM</td>
</tr>
<tr>
<td>ImpairmentOptions</td>
<td>Impairment Option</td>
</tr>
<tr>
<td>PurchReason</td>
<td>Purchase Reason Code</td>
</tr>
</tbody>
</table>

**Predefined Smart List Entries**

Review the predefined Smart Lists to determine if you need to change any of them or add new ones. Smart Lists are used in predefined member formulas and drive the calculations of the Oracle Hyperion Capital Asset Planning, Fusion Edition model; 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.
<table>
<thead>
<tr>
<th>Smart List</th>
<th>Entries</th>
</tr>
</thead>
</table>
| DeprMethod        | ● NoDepr  
|                   | ● SLN  
|                   | ● SYD  
|                   | ● DBYear  
|                   | ● DBPeriod  |
| PhysicalLocation  | ● Location1  
|                   | ● Location2  
|                   | ● Location3  |
| AssetStatus       | ● Delete  
|                   | ● New  
|                   | ● Active  
|                   | ● Retired  
|                   | ● Sold  
|                   | ● Transferred  |
| DeprConvention    | ● ProrateBegPer  
|                   | ● ProrateActDate  
|                   | ● MidPeriod  
|                   | ● MidYear  |
| TangibleFlag      | ● Tangible  
|                   | ● Intangible  |
| CashFlowIncidence | ● MonthsPrior2  
|                   | ● MonthsPrior1  
|                   | ● Staggered  
|                   | ● SameMonth  
|                   | ● MonthsCredit1  
|                   | ● MonthsCredit2  
|                   | ● MonthsCredit3  
|                   | ● MonthsCredit4  |
| RetireOptions     | ● Sale  
|                   | ● WriteOff  |
| AmortizationMethod| ● FiniteLivedEven  
|                   | ● IndefiniteLived  |
| DelayReasons      | ● TransferIn  |
### Predefined Menus

Planners use menus to work with asset data in data forms. The tables in this section describe these predefined menus. The information listed in the Label Value column is displayed when planners click a row member.

**Table 8 Predefined Menus: Capital Asset Planning**

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amortization Summary Menu</td>
<td>Runs the predefined business rule and summarizes the amortized assets</td>
</tr>
<tr>
<td>Capital Expense Summary Menu</td>
<td>Runs the predefined business rule and summarizes the planned capital expenses</td>
</tr>
<tr>
<td>Depreciation Summary Menu</td>
<td>Runs the predefined business rule and summarizes the depreciated assets</td>
</tr>
<tr>
<td>Existing Intangible Menu</td>
<td>Runs the predefined business rule to add existing intangible assets</td>
</tr>
<tr>
<td>Existing Specified Menu</td>
<td>Runs the predefined business rule to add existing tangible assets</td>
</tr>
<tr>
<td>Intangible Expense Summary Menu</td>
<td>Runs the predefined business rule and summarizes the expenses for intangibles</td>
</tr>
<tr>
<td>New Asset Menu</td>
<td>Runs the predefined business rule to create assets</td>
</tr>
<tr>
<td>Label Value</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>New Intangible Menu</td>
<td>Runs the predefined business rule to create intangible assets</td>
</tr>
</tbody>
</table>

Table 9  Predefined Menu: New Intangible Menu

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add New Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Profit Loss Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Balance Sheet Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Cash Flow Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Remove Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Amortization</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

Table 10  Predefined Menu: New Asset Menu

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add New Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Profit Loss Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Balance Sheet Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Cash Flow Impact</td>
<td>Data form</td>
</tr>
<tr>
<td>Remove Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calc Depreciation</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

Table 11  Predefined Menu: Existing Specified Menu

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transfer</td>
<td>Business rule</td>
</tr>
<tr>
<td>Retire</td>
<td>Business rule</td>
</tr>
<tr>
<td>Improve</td>
<td>Business rule</td>
</tr>
<tr>
<td>Asset Expenses</td>
<td>Data form</td>
</tr>
<tr>
<td>Add Existing</td>
<td>Business rule</td>
</tr>
<tr>
<td>Remove Existing</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td></td>
<td>(Not applicable)</td>
</tr>
<tr>
<td>Calculate Details</td>
<td>Data form</td>
</tr>
<tr>
<td>Label Value</td>
<td>Type</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Cash Flow Impact</td>
<td>Data form</td>
</tr>
</tbody>
</table>

**Table 12  Predefined Menu: Existing Intangible Menu**

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Existing Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Impair</td>
<td>Business rule</td>
</tr>
<tr>
<td>Transfer</td>
<td>Business rule</td>
</tr>
<tr>
<td>Retire</td>
<td>Business rule</td>
</tr>
<tr>
<td>Enter Expense</td>
<td>Data form</td>
</tr>
<tr>
<td>Remove Existing Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Line</td>
<td>(Not applicable)</td>
</tr>
<tr>
<td>Calculate Details</td>
<td>Business rule</td>
</tr>
<tr>
<td>Cashflow Impact</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

**Table 13  Predefined Menu: Capital Asset Summary Menu**

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cap Sum Asset Balances</td>
<td>Data form</td>
</tr>
<tr>
<td>Roll up Asset</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

**Table 14  Predefined Menu: Intangible Expense Summary Menu**

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intangible LI Details</td>
<td>Data form</td>
</tr>
</tbody>
</table>

**Table 15  Predefined Menu: Depreciation Summary Menu**

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Depreciation Details</td>
<td>Data form</td>
</tr>
<tr>
<td>Roll up Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Depreciation</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Existing Depreciation</td>
<td>Business rule</td>
</tr>
</tbody>
</table>
Table 16  Predefined Menu: Amortization Summary Menu

<table>
<thead>
<tr>
<th>Label Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amortization Details</td>
<td>Data form</td>
</tr>
<tr>
<td>Roll up Asset</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Amortization</td>
<td>Business rule</td>
</tr>
<tr>
<td>Calculate Existing Amortization</td>
<td>Business rule</td>
</tr>
</tbody>
</table>

This table lists the predefined data forms that have associated menus. The other data forms do not have associated predefined menus.

Table 17  Predefined Data Form Menus

<table>
<thead>
<tr>
<th>Data Form</th>
<th>Associated Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>02A. New Asset Requests - Addition</td>
<td>NewAssetMenu</td>
</tr>
<tr>
<td>03. Manage Existing Specified</td>
<td>ExistingSpecifiedMenu</td>
</tr>
<tr>
<td>03A. Existing Specified Drill Down</td>
<td>ExistingSpecifiedMenu</td>
</tr>
<tr>
<td>03B. Existing Specified Expenses</td>
<td>ExistingSpecifiedMenu</td>
</tr>
<tr>
<td>05A. New Intangible Requests - Additions</td>
<td>NewIntangibleMenu</td>
</tr>
<tr>
<td>06. Manage Existing Intangibles Specified</td>
<td>ExistingIntangibleMenu</td>
</tr>
<tr>
<td>08. Capital Expenditure Summary</td>
<td>CapitalExpenseSummaryMenu</td>
</tr>
<tr>
<td>08A. Capital Expenditure Summary - Line Item details</td>
<td>CapitalExpenseSummaryMenu</td>
</tr>
<tr>
<td>09. Intangible Expenditure Summary</td>
<td>IntangibleExpSummMenu</td>
</tr>
<tr>
<td>13. Depreciation Summary</td>
<td>DeprSummaryMenu</td>
</tr>
<tr>
<td>13A. Depreciation Summary - Line Item details</td>
<td>CapitalExpenseSummaryMenu</td>
</tr>
<tr>
<td>14. Amortization Summary</td>
<td>AmortSummaryMenu</td>
</tr>
<tr>
<td>14A. Amortization Summary - Line Item details</td>
<td>CapitalExpenseSummaryMenu</td>
</tr>
</tbody>
</table>

Predefined Business Rules

Planners use predefined business rules to calculate capital expense data.

Add Asset

Description

Adds a new asset.
Formula

SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR paramsSet = 0;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR numAsset = [NumAsset];
VAR preExistingPers = 0;
VAR setStartDepr = 0;

FIX(@CHILDREN("Total New"), [Hidden_Scenario], [Hidden_Version], [Department], [AssetClass])

/*@ Setup properties for new Asset */
FIX("No Year", "BegBalance")
"Asset Description" (
IF (numAsset > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING )
numAsset = numAsset - 1;
"Asset Status" = 0; /* Indicate that we have a new asset */
"Asset Description" = [AssetDesc];
"CAR No." = [AssetCAR];
"Justification" = [Justification];
"Priority" = [Priority];
"Asset Units" = [AssetUnits];
"Asset Rate" = [AssetRate];
"Installation" = [Installation];
"Freight" = [Freight];
"Salvage Input" = [SalvageValue];
"Salvage" = [SalvageValue] * "Asset Units";
"Taxes %" = [TaxesPct];
"Additional Charges" = [AdditionalCharges];
"Retirement Obligation" = [RetirementObs];
"Physical Location" = [PhyLocation];
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"-
>"Useful Life (in Years)";
"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"-
>"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"-
>"Global"->"Funding Incidence";
"Purchase Date" = [PurchaseDate];
"In Service Date" = [InserviceDate];
"UOM" = 1; /*Default to Nos */
IF ("In Service Date" < "Purchase Date")
    "In Service Date" = "Purchase Date";
ENDIF
ENDFIX

/* Calculate depreciation section */
FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"(
/* Only process newly added asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" == 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
    paramsSet = 1;
/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;
cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"-
Cash Flow Incidence

IF (cashFlowIncidence == -1)
cashStaggeredPers = 4;
ELSE
/* Extract year and month from date to adjust based in incidence values in global assumptions */
yearVal = @INT(cashOutflowDate / 10000) * 10000;
monthVal = cashOutflowDate - yearVal;
monthVal = monthVal + cashFlowIncidence;

IF (monthVal <= 0)
  yearVal = yearVal - 10000;
  monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
  yearVal = yearVal + 10000;
  monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence == -1)
fundingStaggeredPers = 4;
ELSE
yearVal = @INT(fundingDate / 10000) * 10000;
monthVal = fundingDate - yearVal;
monthVal = monthVal + fundingIncidence;

IF (monthVal <= 0)
  yearVal = yearVal - 10000;
  monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
  yearVal = yearVal + 10000;
  monthVal = monthVal - 1200;
ENDIF

fundingDate = yearVal + monthVal;
fundingStaggeredPers = 1;
ENDIF

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
deprConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance Cost";
"Insurance %";
    maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
    repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";
    fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

    /* check if we have a pre-existing asset */
    IF (inServiceDate < "First Date")
        yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
        monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
        IF (monthVal < 0)
            IF (yearVal > 0)
                yearVal = yearVal - 1;
            ENDIF
            monthVal = 1200 + monthVal;
        ENDIF
        preExistingPers = yearVal * 12 + monthVal / 100;
        setStartDepr = 1;
    ELSE
        preExistingPers = 0;
        setStartDepr = 0;
    ENDIF

    /* setup for depreciation */
    numDeprPeriods = 0;
    periodOffset = 0;

    IF (deprMethod > 0)
        priorAccumDepr = 0;
        periodicPriorAccumDepr = 0;
        deprAmt = 0;

        /* Add a periodic vs annual property instead of additional deprMethods */
        /* Annual Methods SumYearDigits = 2, DecliningBalance Year = 3*/
        IF (deprMethod == 2 OR deprMethod == 3)
            isAnnual = 1;
            life = "No Year"->"Useful Life (in Years)";
            persInSection = "NumPeriods";
        ELSE /* Periodic Methods */
            isAnnual = 0;
            life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
            persInSection = 1;
        ENDIF

        setCashImpact = 0;
        setFundingImpact = 0;
        yearVal = @INT(inServiceDate / 10000) * 10000;

        /* DecliningBalance methods */
        IF (deprMethod == 3 OR deprMethod == 4)
            /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);

ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
    deprMethod = 1;
    deprConvention = 1;
ENDIF
ENDIF

ENDIF

/* Set Asset Status to active when we hit the end */
IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF
ENDIF

/* Start depreciation calculations */
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Depreciation" = #MISSING;
    "Accumulated Depreciation" = #MISSING;
    "Property, Plant and Equipment Gross" = #MISSING;
    "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
    "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;

    "Cash Flow Allocator" = #MISSING;
    "Funding Allocator" = #MISSING;
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Repairs" = #MISSING;
    "Retirement Expenses" = #MISSING;
    "Capital Expenditure" = #MISSING;

    IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
    ENDIF

    /* Check if we should start setting cash flow impact values */
    IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setCashImpact = cashStaggeredPers;
        cashAllocPct = 1 / cashStaggeredPers;
    ENDIF

    /* Check if we should start setting funding values */
    IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setFundingImpact = fundingStaggeredPers;
    ENDIF
fundingAllocPct = 1 / fundingStaggeredPers;

ENDIF

IF (setCashImpact > 0)
    "Cash Flow Allocator" = cashAllocPct;
    setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
    "Funding Allocator" = fundingAllocPct;
    setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/* Check if we should start depreciation */
IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1)
OR 
    (deprConvention == 2) /* Prorate Actual Date */
    IF (dayVal == 1)  /* Treat same as Begin Period */
        split1stAmt = 0;
    ELSE
        numDeprPeriods = numDeprPeriods + 1;
        split1stAmt = 1;
    ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
    numDeprPeriods = numDeprPeriods + 1;
    split1stAmt = 1;
ELSEIF (deprConvention == 4)  /* MidYear */
    split1stAmt = 0;
    delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
    split1stAmt = 0;
ENDIF

/* Declining balance methods */
IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / NumPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numDeprPeriods = numDeprPeriods - preExistingPers;
    IF (numDeprPeriods > 0)
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
        periodicPriorAccumDepr = deprAmt * preExistingPers;
    ENDIF
ENDIF

/* Calculate Depreciation amounts */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)

    lifeIndex = @INT(periodOffset / persInSection);

    IF (deprMethod == 1) /* Straight Line */
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
    ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
        IF (lifeIndex == 0)
            /* cost * rate * month / 12; for 1st period */
            deprAmt = basicCost * deprRate * persIn1stYear / NumPeriods / persInSection;
        ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
            deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
        ENDIF
    ELSEIF (deprMethod == 2) /* Sum of Years Digits */
        deprRate = (life - lifeIndex) * 2 / (life * (life + 1));
        deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
    ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
    deprRate = (life - lifeIndex) * 2 / (life * (life + 1));
    deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
    IF (deprConvention == 2)
        /* assume that there are 30 days in each month */
        deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
    ELSEIF (deprConvention == 4)
        ;
    ELSE
        deprAmt = deprAmt - deprAmt / 2;
    ENDIF
    split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
    deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF
IF (isAnnual == 1)
    IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
        priorAccumDepr = periodicPriorAccumDepr;
    ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
    "Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumDepr + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue - saleValue + deprAmt;
    ELSE
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue + deprAmt;
    ENDIF
    "Retirement Expenses" = retirementCosts - retirementObs;
    "Proceeds from Sale of Property, Plant and Equipment" = saleValue - retirementCosts;
ENDIF
periodOffset = periodOffset +1;
ENDIF
 delayStart = delayStart -1;
ENDIF
ENDIF
);
ENDFIX
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
    CALC DIM ("Period");
    @ANCESTORS("Line Item 1");
    @ANCESTORS([AssetClass]);
ENDFIX
AddExistAsset

Description

Adds an existing asset.

Formula

```
SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR paramsSet = 0;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR numAsset = [NumAsset];
VAR preExistingPers = 0;
VAR setStartDepr = 0;
```
/* Setup properties for new Asset */
FIX("No Year", "BegBalance")
"Asset Description" ( 
IF (numAsset > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING ) 
umAsset = numAsset - 1;
"Asset Status" = 0;    /* Indicate that we have a new asset */
"Asset Description" = [AssetDesc];
"Asset Units" = [AssetUnits];
"Asset Rate" = [AssetRate];
"Installation" = [Installation];
"Freight" = [Freight];
"Salvage Input" = [SalvageValue];
"Salvage" = [SalvageValue] * "Asset Units";
"Taxes %" = [TaxesPct];
"Additional Charges" = [AdditionalCharges];
"Retirement Obligation" = [RetirementObs];
"Physical Location" = [PhyLocation];
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Purchase Date" = [PurchaseDate];
"In Service Date" = [InserviceDate];
IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF
ENDFIX

END

/* Calculate depreciation section */
FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"(
/* Only process newly added asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" == 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
  paramsSet = 1;
/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

depMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
depConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
  yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
  monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
  IF (monthVal < 0)
    IF (yearVal > 0)
      yearVal = yearVal - 1;
    ENDIF
    monthVal = 1200 + monthVal;
  ENDIF
  preExistingPers = yearVal * 12 + monthVal / 100;
  setStartDepr = 1;
ELSE
    preExistingPers = 0;
    setStartDepr = 0;
ENDIF

/* setup for depreciation */
numDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
  priorAccumDepr = 0;
  periodicPriorAccumDepr = 0;
  deprAmt = 0;

  /* Add a periodic vs annual property instead of additional deprMethods */
  /* Annual Methods SumYearDigits = 2, DecliningBalance Year = 3*/
  IF (deprMethod == 2 OR deprMethod == 3)
    isAnnual = 1;
    life = "No Year"->"Useful Life (in Years)";
    persInSection = "NumPeriods";
  ELSE /* Periodic Methods */
    isAnnual = 0;
    life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
    persInSection = 1;
ENDIF
yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
   /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
   deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
   deprMethod = 1;
   deprConvention = 1;
ENDIF

ENDIF

/* Set Asset Status to active when we hit the end */
IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
   paramsSet = 0;
   IF ("Asset Status" == 0)
      "Asset Status" = 1;
   ENDIF
ENDIF

/* Start depreciation calculations */
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
   /* clear out any previously calculated values */
   "Depreciation" = #MISSING;
   "Accumulated Depreciation" = #MISSING;
   "Property, Plant and Equipment Gross" = #MISSING;
   "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
   "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;
   "Cash Flow Allocator" = #MISSING;
   "Funding Allocator" = #MISSING;
   "Cash Outflow from Capital Additions" = #MISSING;
   "Cash Inflow from Funding" = #MISSING;
   "Long Term Debt" = #MISSING;
   "Insurance" = #MISSING;
   "Maintenance" = #MISSING;
   "Repairs" = #MISSING;
   "Retirement Expenses" = #MISSING;
   "Capital Expenditure" = #MISSING;

   /* Commented out for now need to verify
   IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
      "Capital Expenditure" = basicCost - retirementObs;
   ENDIF
   */
   IF ((numDeprPeriods == 0) AND (("TP-Date" == inServiceDate) OR (setStartDepr == 1)) OR
(deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1) AND "TP-Date" > yearVal))

setStartDepr = 0;
periodOffset = 0;
persIn1stYear = "NumPeriods"; /* not adjusted - "Cal TP-Index" + 1; */
numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
totDeprPeriods = numDeprPeriods;

dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);

delayStart = 0;
IF (deprConvention == 2) /* Prorate Actual Date */
   IF (dayVal == 1) /* Treat same as Begin Period */
      split1stAmt = 0;
   ELSE
      numDeprPeriods = numDeprPeriods + 1;
      split1stAmt = 1;
   ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
   numDeprPeriods = numDeprPeriods + 1;
   split1stAmt = 1;
ELSEIF (deprConvention == 4) /* MidYear */
   split1stAmt = 0;
   delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
   split1stAmt = 0;
ENDIF

/* Declining balance methods */
IF (deprMethod == 3 OR deprMethod == 4)
   /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
   deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
   numDeprPeriods = numDeprPeriods - preExistingPers;
   IF (numDeprPeriods > 0)
      deprAmt = (basicCost - salvageVal) / totDeprPeriods;
      periodicPriorAccumDepr = deprAmt * preExistingPers;
   ENDIF
ENDIF

ENDIF

/* Calculate Depreciation amounts */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
   lifeIndex = @INT(periodOffset / persInSection);
ENDIF
IF (deprMethod == 1) /* Straight Line */
    deprAmt = (basicCost - salvageVal) / totDeprPeriods;
ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
    IF (lifeIndex == 0)
        /* cost * rate * month / 12; for 1st period */
        deprAmt = basicCost * deprRate * persIn1stYear / "NumPeriods" / persInSection;
    ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
        deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
    ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
    deprRate = (life - lifeIndex) * 2 / (life * (life +1));
    deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
    IF (deprConvention == 2)
        /* assume that there are 30 days in each month */
        deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
    ELSEIF (deprConvention == 4);
    ELSE
        deprAmt = deprAmt - deprAmt / 2;
    ENDIF
    split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
    deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
    IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
        priorAccumDepr = periodicPriorAccumDepr;
    ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
"Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
  IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
      "No Year"->"BegBalance"->"Reason Ended" == 3)
    netValue = basicCost - priorAccumDepr + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
      "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue -
      saleValue + deprAmt;
    ELSE
      "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue +
      deprAmt;
    ENDIF
    "Retirement Expenses" = retirementCosts - retirementObs;
    "Proceeds from Sale of Property, Plant and Equipment" = saleValue -
    retirementCosts;
  ENDIF
ENDIF
  periodOffset = periodOffset +1;
ENDIF
delayStart = delayStart -1;
ENDIF
ENDIF
);
ENDFIX
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
  CALC DIM ("Period");
  @IANCESTORS("Line Item 1");
  @ANCESTORS([AssetClass]);
ENDFIX

AddExistIntangible

Description
Adds an existing intangible asset.

Formula

SET CREATENONMISSINGBLK ON;

VAR numAsset = [NumAsset];
VAR periodOffset = 0;
VAR amortMethod = 0;
VAR amortRate = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR purchaseDate = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX(@CHILDREN("Base SPs"), [Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])
/* Find a new asset line item */
FIX("No Year", "BegBalance")
"Asset Description" ( 
IF (numAsset > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING )
    numAsset = numAsset - 1;
"Asset Status" = 0; /* Indicate that we have a new asset */
"Asset Description" = [AssetDesc];
"Acquisition Costs" = [AcquisitionCost];
"Additional Charges" = [AdditionalCharges];
"Asset Units" = 1;
"Salvage Input" = [SalvageValue];
"Salvage" = [SalvageValue] * "Asset Units";
"Asset Rate" = "Acquisition Costs" + "Additional Charges";
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->
"Useful Life (in Years)";
    "Purchase Date" = [PurchaseDate];
    "In Service Date" = [InserviceDate];
    IF ("In Service Date" < "Purchase Date")
        "In Service Date" = "Purchase Date";
    ENDIF
ENDIF 
)
ENDFIX 

/* Amortization section */
FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"(
/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)

IF (@ISMBR("BegBalance") AND paramsSet == 0)
/* Initialized all the variables required to calculate depreciation */
paramsSet = 1;

/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
"Amortization Method";

insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Maintenance %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
   impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
   impairmentDate1 = #MISSING;
ENDIF
IF ("No Year"->"Impairment Date2" <> #MISSING)
   impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
   impairmentDate2 = #MISSING;
ENDIF

impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
   yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
   monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate -
   @INT(inServiceDate / 10000) * 10000);
   IF (monthVal < 0)
      IF (yearVal > 0)
         yearVal = yearVal - 1;
      ENDIF
   monthVal = 1200 + monthVal;
ENDIF

Predefined Business Rules 87
preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
setStartAmort = 1;
ELSE
preExistingPers = 0;
setStartAmort = 0;
ENDIF
numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
priorAccumAmort = 0;
amortAmt = 0;
yearVal = @INT(inServiceDate / 10000) * 10000;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF
ENDIF

IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
paramsSet = 0;
IF ("Asset Status" == 0)
"Asset Status" = 1;
ENDIF
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)

/* clear out any previously calculated values */
"Amortization" = #MISSING;
"Accumulated Amortization" = #MISSING;
"Intangible Assets Finite, Gross" = #MISSING;
"Loss/(Gain) on Sale of Intangibles" = #MISSING;
"Proceeds from Sale of Intangibles" = #MISSING;
"Cash Flow Allocator" = #MISSING;
"Funding Allocator" = #MISSING;
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Impairment of Assets" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Reserve" = #MISSING;
"Capital Expenditure" = #MISSING;

/* Commented out for now need to verify
IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
"Capital Expenditure" = basicCost - retirementObs;
ENDIF
*/

/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate OR (setStartAmort == 1))
setStartAmort = 0;
periodOffset = 0;
numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in
Years)";
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
intangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numAmortPeriods = numAmortPeriods - preExistingPers;
    IF (numAmortPeriods > 0)
        priorAccumAmort = amortAmt * preExistingPers;
    ENDIF
ENDIF
numRemainingPers = numAmortPeriods;
ENDIF

/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)

    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate1)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
            "Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
        ENDIF
        basicCost = basicCost - (intangibleNet - impairmentFairVal1);
        amortAmt = impairmentFairVal1 / numRemainingPers;
    ELSEIF ("TP-Date" == impairmentDate2)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
            "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
        ENDIF
        basicCost = basicCost - (intangibleNet - impairmentFairVal2);
        amortAmt = impairmentFairVal2 / numRemainingPers;
    ENDIF
ENDIF
IF (periodOffset + 1 == numAmortPeriods)
    amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF
priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;
/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))

"Intangible Assets Finite, Gross" = basicCost;
"Amortization" = amortAmt;
"Accumulated Amortization" = priorAccumAmort;
"Insurance" = insuranceCost;
"Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
  "No Year"->"BegBalance"->"Reason Ended" == 3)

netValue = basicCost - priorAccumAmort + retirementCosts;
IF ("No Year"->"BegBalance"->"Reason Ended" == 3)

"Loss/(Gain) on Sale of Intangibles" = netValue - saleValue +
amortAmt;
ELSE

"Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
ENDIF
"Retirement Expenses" = retirementCosts - retirementObs;
"Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF

ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF

FIX([Hidden_Scenario], [Hidden_Version], [Department])

@IANCESTORS("Line Item 1");
@ANCESTORS([AssetClassInt]);
ENDFIX

---

**Add Intangible**

**Description**

Adds a new intangible asset.

**Formula**

```
SET CREATENONMISSINGBLK ON;

VAR numAsset = [NumAsset];
VAR periodOffset = 0;
VAR amortMethod = 0;
```
VAR amortRate = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX(@CHILDREN("Total New"), [Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])

/* Find a new asset line item */
FIX("No Year", "BegBalance")
"Asset Description" (  
IF (numAsset > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING )  
numAsset = numAsset - 1;
"Asset Status" = 0;  /* Indicate that we have a new asset */
"Asset Description" = [AssetDesc];
"CAR No." = [AssetCAR];
"Justification" = [Justification];
"Acquisition Costs" = [AcquisitionCost];
"Additional Charges" = [AdditionalCharges];
"Asset Units" = 1;
"Salvage Input" = [SalvageValue];
"Salvage" = [SalvageValue] * "Asset Units";
"Asset Rate" = "Acquisition Costs" + "Additional Charges";
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->
>"Useful Life (in Years)";
"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->
>"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
>"Funding Incidence";
"Purchase Date" = [PurchaseDate];
"In Service Date" = [InserviceDate];

IF ("In Service Date" < "Purchase Date")
 "In Service Date" = "Purchase Date";
ENDIF
ENDIF
ENDFIX

/* Amortization section */
FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"(

/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)

IF (@ISMBR("BegBalance") AND paramsSet == 0)
 /* Initialized all the variables required to calculate depreciation */
 paramsSet = 1;

 /* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;

cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->
>"Cash Flow Incidence";

IF (cashFlowIncidence == -1)
cashStaggeredPers = 4;
ELSE
 /* Extract year and month from date to adjust based in incidence values in
 global assumptions */
 yearVal = @INT(cashOutflowDate / 10000) * 10000;
 monthVal = cashOutflowDate - yearVal;
 monthVal = monthVal + cashFlowIncidence;

 IF (monthVal <= 0)
 yearVal = yearVal - 10000;
monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence == -1)
    fundingStaggeredPers = 4;
ELSE
    yearVal = @INT(fundingDate / 10000) * 10000;
    monthVal = fundingDate - yearVal;
    monthVal = monthVal + fundingIncidence;
    IF (monthVal <= 0)
        yearVal = yearVal - 10000;
        monthVal = 1200 + monthVal;
    ELSEIF (monthVal > 1200)
        yearVal = yearVal + 10000;
        monthVal = monthVal - 1200;
    ENDIF

    fundingDate = yearVal + monthVal;
    fundingStaggeredPers = 1;
ENDIF

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization Method";

insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
    impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
    impairmentDate1 = #MISSING;
ENDIF
IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF
impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/**********************************
 /* check if we have a pre-existing asset */
 IF (inServiceDate < "First Date")
     yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
     monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
     IF (monthVal < 0)
         IF (yearVal > 0)
             yearVal = yearVal - 1;
         ENDIF
         monthVal = 1200 + monthVal;
     ENDIF
     preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
 ELSE
     preExistingPers = 0;
 ENDIF
numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    setCashImpact = 0;
    setFundingImpact = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF
ENDIF
IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Amortization" = #MISSING;
    "Accumulated Amortization" = #MISSING;
    "Intangible Assets Finite, Gross" = #MISSING;
    "Loss/(Gain) on Sale of Intangibles" = #MISSING;
    "Proceeds from Sale of Intangibles" = #MISSING;
ENDIF
"Cash Flow Allocator" = #MISSING;
"Funding Allocator" = #MISSING;
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Impairment of Assets" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Reserve" = #MISSING;
"Capital Expenditure" = #MISSING;

IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    "Capital Expenditure" = basicCost - retirementObs;
ENDIF

IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    setCashImpact = cashStaggeredPers;
    cashAllocPct = 1 / cashStaggeredPers;
ENDIF

IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    setFundingImpact = fundingStaggeredPers;
    fundingAllocPct = 1 / fundingStaggeredPers;
ENDIF

IF (setCashImpact > 0)
    "Cash Flow Allocator" = cashAllocPct;
    "Cash Outflow from Capital Additions" = basicCost * cashAllocPct;
    setCashImpact = setCashImpact - 1;
ENDIF

IF (setFundingImpact > 0)
    "Funding Allocator" = fundingAllocPct;
    "Cash Inflow from Funding" = fundingAmt * fundingAllocPct;
    "Long Term Debt" = fundingAmt * fundingAllocPct;
    setFundingImpact = setFundingImpact - 1;
ENDIF

/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate OR (setStartAmort == 1))
    setStartAmort = 0;
    periodOffset = 0;
    numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
    insuranceCost = insuranceCost / "NumPeriods";
    maintenanceCost = maintenanceCost / "NumPeriods";
    intangibleNet = basicCost;
    amortAmt = (basicCost - salvageVal) / numAmortPeriods;

    /* Adjust for pre existing assets */
    IF (preExistingPers > 0)
        numAmortPeriods = numAmortPeriods - preExistingPers;
    ENDIF
IF (numAmortPeriods > 0)
    priorAccumAmort = amortAmt * preExistingPers;
ENDIF

numRemainingPers = numAmortPeriods;

/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)
    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate1)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
        ENDIF
        "Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
    ELSEIF ("TP-Date" == impairmentDate2)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
        ENDIF
        "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal1); amortAmt = impairmentFairVal1 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)
    IF (impairmentOpt == 1)
        "Impairment of Assets" = intangibleNet - impairmentFairVal2;
    ELSEIF (impairmentOpt == 2)
        "Capital Reserve" = intangibleNet - impairmentFairVal2;
    ELSEIF (impairmentOpt == 3)
        "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
        "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal2); amortAmt = impairmentFairVal2 / numRemainingPers;
ENDIF

IF (periodOffset + 1 == numAmortPeriods)
    amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF

priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Intangible Assets Finite, Gross" = basicCost;
    "Amortization" = amortAmt;
    "Accumulated Amortization" = priorAccumAmort;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
ENDIF
/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
  IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
      "No Year"->"BegBalance"->"Reason Ended" == 3)
    netValue = basicCost - priorAccumAmort + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
      "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue +
    ELSE
      "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
    ENDIF
    "Retirement Expenses" = retirementCosts - retirementObs;
    "Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
  ENDIF
  ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF
ENDIF

CalcAmort

Description
Calculates amortization.

Formula

SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX([Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])

FIX ("BegBalance", "No Year")
"Basic Cost" {  
  IF ("Asset Status" > 0)  
    IF ("In Service Date" < "Purchase Date")  
      "In Service Date" = "Purchase Date";  
      ENDIF  
    "Basic Cost" = "Acquisition Costs" + "Additional Charges";  
    "Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"-  
    >"Useful Life (in Years)";  
    "Salvage" = "Salvage Input" * "Asset Units";  
    ENDIF  
  }  
ENDFIX

FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"{  
/* Only calc valid asset line items */  
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)  
  IF (@ISMBR("BegBalance") AND paramsSet == 0)  
    /* Initialized all the variables required to calculate depreciation */  
    paramsSet = 1;  
  ENDIF

98 Capital Asset Planning Structure
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;

cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";

IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
cashflowChanged = 1;
ELSE
cashflowChanged = 0;
ENDIF

IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
cashStaggeredPers = 4;
ELSE
/* Extract year and month from date to adjust based in incidence values in
global assumptions */
yearVal = @INT(cashOutflowDate / 10000) * 10000;
monthVal = cashOutflowDate - yearVal;
monthVal = monthVal + cashFlowIncidence;

IF (monthVal <= 0)
yearVal = yearVal - 10000;
monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
yearVal = yearVal + 10000;
monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence <> "No Year"->"Funding Incidence")
fundingChanged = 1;
ELSE
fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
fundingStaggeredPers = 4;
ELSE
yearVal = @INT(fundingDate / 10000) * 10000;
monthVal = fundingDate - yearVal;
monthVal = monthVal + fundingIncidence;

IF (monthVal <= 0)
yearVal = yearVal - 10000;
monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
ENDIF

fundingDate = yearVal + monthVal;
fundingStaggeredPers = 1;
ENDIF

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization Method";

insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
    impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
    impairmentDate1 = #MISSING;
ENDIF

IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF

impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
```
setStartAmort = 1;
ELSE
    preExistingPers = 0;
    setStartAmort = 0;
ENDIF

numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    setCashImpact = 0;
    setFundingImpact = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF

ENDIF

IF (@ISMBR("No Year"))
    paramsSet = 0;
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    // clear out any previously calculated values */
    "Amortization" = #MISSING;
    "Accumulated Amortization" = #MISSING;
    "Intangible Assets Finite, Gross" = #MISSING;
    "Loss/(Gain) on Sale of Intangibles" = #MISSING;
    "Proceeds from Sale of Intangibles" = #MISSING;
    IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
        "Cash Flow Allocator" = #MISSING;
    ENDIF
    IF (fundingIncidence <> -1 OR fundingChanged == 1)
        "Funding Allocator" = #MISSING;
    ENDIF
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Impairment of Assets" = #MISSING;
    "Retirement Expenses" = #MISSING;
    "Capital Reserve" = #MISSING;
    "Capital Expenditure" = #MISSING;
    "Retirement Expenses" = #MISSING;
    IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
    ENDIF
    IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
```
setCashImpact = cashStaggeredPers;
cashAllocPct = 1 / cashStaggeredPers;
ENDIF
IF ("TP-Date" == fundingDate AND (delayedStart == #MISSING OR (delayedStart > #MISSING AND "TP-Date" > delayedStart)))
    setFundingImpact = fundingStaggeredPers;
fundingAllocPct = 1 / fundingStaggeredPers;
ENDIF
IF (setCashImpact > 0)
    "Cash Flow Allocator" = cashAllocPct;
    setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";
IF (setFundingImpact > 0)
    "Funding Allocator" = fundingAllocPct;
    setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";
/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate OR (setStartAmort == 1))
    setStartAmort = 0;
    periodOffset = 0;
    numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
    insuranceCost = insuranceCost / "NumPeriods";
    maintenanceCost = maintenanceCost / "NumPeriods";
    intangibleNet = basicCost;
    amortAmt = (basicCost - salvageVal) / numAmortPeriods;
/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numAmortPeriods = numAmortPeriods - preExistingPers;
    IF (numAmortPeriods > 0)
        priorAccumAmort = amortAmt * preExistingPers;
    ENDIF
ENDIF
numRemainingPers = numAmortPeriods;
ENDIF
/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)
    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal) * capitalizePct;
"Capital Reserve" = intangibleNet - impairmentFairVal;
ENDIF
"Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
ENDIF

basicCost = basicCost - (intangibleNet - impairmentFairVal1);
amortAmt = impairmentFairVal1 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)
  IF (impairmentOpt == 1)
    "Impairment of Assets" = intangibleNet - impairmentFairVal2;
  ELSEIF (impairmentOpt == 2)
    "Capital Reserve" = intangibleNet - impairmentFairVal2;
  ELSEIF (impairmentOpt == 3)
    "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
    "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
  ENDIF
basicCost = basicCost - (intangibleNet - impairmentFairVal2);
amortAmt = impairmentFairVal2 / numRemainingPers;
ENDIF

IF (periodOffset + 1 == numAmortPeriods)
  amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF

priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
  "Intangible Assets Finite, Gross" = basicCost;
  "Amortization" = amortAmt;
  "Accumulated Amortization" = priorAccumAmort;
  "Insurance" = insuranceCost;
  "Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
  IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)
    netValue = basicCost - priorAccumAmort + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
      "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue + amortAmt;
      ELSE
        "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
      ENDIF
  ELSE
    "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
  ENDIF
  "Retirement Expenses" = retirementCosts - retirementObs;
  "Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF
);
ENDFIX

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" (  
    IF (cashflowChanged == 1 AND "Asset Status" > 0)  
        "Cash Flow Incidence" = cashFlowIncidence;  
    ENDIF  
    IF (fundingChanged == 1 AND "Asset Status" > 0)  
        "Funding Incidence" = fundingIncidence;  
    ENDIF  
)
ENDFIX

ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
@IANCESTORS([LineItem]);
@ANCESTORS([AssetClassInt]);
ENDFIX

CalcDepr

Description
Calculates depreciation.

Formula

SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persInstYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR deprAmtLast = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR retirementObs = 0;
VAR retirementCosts = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR retireOption = 0;
VAR saleValue = 0;
VAR netValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart,
VAR split1stAmt;
VAR paramsSet = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartDepr = 0;

FIX([AssetClass], [Hidden_Scenario], [Hidden_Version], [Department])

FIX ("BegBalance", "No Year")
"Basic Cost" (  
  IF ("Asset Status" > 0)  
    IF ("In Service Date" < "Purchase Date")  
      "In Service Date" = "Purchase Date";
    ENDIF  
    "Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
    "Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
    "Salvage" = "Salvage Input" * "Asset Units";
  ENDIF
)  
ENDFIX

FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"  
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)  
  IF (@ISMBR("BegBalance") AND paramsSet == 0)  
    paramsSet = 1;
    /* Initialized all the variables required to calculate depreciation */  
    /* eliminate days from date */

Predefined Business Rules 105
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;

cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";

IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
cashflowChanged = 1;
ELSE
  cashflowChanged = 0;
ENDIF

IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
cashStaggeredPers = 4;
ELSE
  /* Extract year and month from date to adjust based in incidence values in global assumptions */
  yearVal = @INT(cashOutflowDate / 10000) * 10000;
  monthVal = cashOutflowDate - yearVal;
  monthVal = monthVal + cashFlowIncidence;
  IF (monthVal <= 0)
    yearVal = yearVal - 10000;
    monthVal = 1200 + monthVal;
  ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
 ENDIF

  cashOutflowDate = yearVal + monthVal;
  cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence <> "No Year"->"Funding Incidence")
fundingChanged = 1;
ELSE
  fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
fundingStaggeredPers = 4;
ELSE
  yearVal = @INT(fundingDate / 10000) * 10000;
  monthVal = fundingDate - yearVal;
  monthVal = monthVal + fundingIncidence;
  IF (monthVal <= 0)
    yearVal = yearVal - 10000;
    monthVal = 1200 + monthVal;
  ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
  ENDIF

  fundingDate = yearVal + monthVal;
  fundingStaggeredPers = 1;
ENDIF
yearVal = yearVal + 10000;
monthVal = monthVal - 1200;
ENDIF

fundingDate = yearVal + monthVal;
fundingStaggeredPers = 1;
ENDIF

/* Initialize for depreciation calc in next section */

basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

depMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
"Depreciation Method";
depConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Repairs %";

defundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Funding %";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
    setStartDepr = 1;
ELSE
    preExistingPers = 0;
    setStartDepr = 0;
ENDIF

/* setup for depreciation */
umDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
persIn1stYear = "NumPeriods"; /* Not adjusted - "Cal TP-Index" + 1; */
priorAccumDepr = 0;
periodicPriorAccumDepr = 0;
deprAmt = 0;
/* Add a periodic vs annual property instead of additional deprMethods */
/* Annual Methods SumYearDigits =2 DecliningBalance Year = 3*/
IF (deprMethod == 2 OR deprMethod == 3)
    isAnnual = 1;
    life = "No Year"->"Useful Life (in Years)";
    persInSection = "NumPeriods";
ELSE /* Periodic Methods */
    isAnnual = 0;
    life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
    persInSection = 1;
ENDIF

setCashImpact = 0;
setFundingImpact = 0;
yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
    deprMethod = 1;
    deprConvention = 1;
ENDIF

ENDIF

IF (@ISMBR("No Year"))
    paramsSet = 0;
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Depreciation" = #MISSING;
    "Accumulated Depreciation" = #MISSING;
    "Property, Plant and Equipment Gross" = #MISSING;
    "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
    "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;
    IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
        "Cash Flow Allocator" = #MISSING;
    ENDIF
    IF (fundingIncidence <> -1 OR fundingChanged == 1)
        "Funding Allocator" = #MISSING;
    ENDIF
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
"Repairs" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Expenditure" = #MISSING;

IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
"Capital Expenditure" = basicCost - retirementObs;
ENDIF

/* Check if we should start setting cash flow impact values */
IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
setCashImpact = cashStaggeredPers;
cashAllocPct = 1 / cashStaggeredPers;
ENDIF

/* Check if we should start setting funding values */
IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
setFundingImpact = fundingStaggeredPers;
fundingAllocPct = 1 / fundingStaggeredPers;
ENDIF

IF (setCashImpact > 0)
"Cash Flow Allocator" = cashAllocPct;
setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
"Funding Allocator" = fundingAllocPct;
setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/* Check if we should start depreciation */
IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1) OR (deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1) AND "TP-Date" > yearVal))
setStartDepr = 0;
periodOffset = 0;
persIn1stYear = "NumPeriods"; /* Not adjusted - "Cal TP-Index" + 1; */
numDeprPeriods = "NumPeriods" - "No Year"->"BegBalance"->"Useful Life (in Years)";
totDeprPeriods = numDeprPeriods;
dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000) * 10000;
delayStart = 0;
IF (deprConvention == 2) /* Prorate Actual Date */
IF (dayVal == 1) /* Treat same as Begin Period */
    split1stAmt = 0;
ELSE

Predefined Business Rules 109
numDeprPeriods = numDeprPeriods + 1;
split1stAmt = 1;
ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
numDeprPeriods = numDeprPeriods + 1;
split1stAmt = 1;
ELSEIF (deprConvention == 4) /* MidYear */
split1stAmt = 0;
delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
split1stAmt = 0;
ENDIF

IF (deprMethod == 3 OR deprMethod == 4)
/* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ELSEIF (deprMethod == 1)
deprRate = (basicCost - salvageVal) / totDeprPeriods;
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
numDeprPeriods = numDeprPeriods - preExistingPers;
IF (numDeprPeriods > 0)
deprAmt = (basicCost - salvageVal) / totDeprPeriods;
periodicPriorAccumDepr = deprAmt * preExistingPers;
ENDIF
ENDIF

ELSE /* Depreciation calculation section */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)

lifeIndex = @INT(periodOffset / persInSection);

IF (deprMethod == 1) /* Straight Line */
deprAmt = (basicCost - salvageVal) / totDeprPeriods;
ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
IF (lifeIndex == 0)
/* cost * rate * month / 12; for 1st period */
deprAmt = basicCost * deprRate * persIn1stYear / "NumPeriods" / persInSection;
ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
deprRate = (life - lifeIndex) * 2 / (life * (life +1));
deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

ENDIF

IF (split1stAmt == 1)
IF (deprConvention == 2)
    /* assume that there are 30 days in each month */
    deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
ELSEIF (deprConvention == 4)
ENDIF
ELSE
    deprAmt = deprAmt - deprAmt / 2;
ENDIF
split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
    deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
    IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
        priorAccumDepr = periodicPriorAccumDepr;
    ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
    "Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
        "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumDepr + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue - saleValue + deprAmt;
    ELSE
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue + deprAmt;
   ENDIF
ENDIF
"Retirement Expenses" = retirementCosts - retirementObs;
"Proceeds from Sale of Property, Plant and Equipment" = saleValue -
CalcExistAmort

Description
Calculates amortization for an existing asset.

Formula

```
SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR purchaseDate = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
```
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX([Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])

FIX ("BegBalance", "No Year")
"Basic Cost" (  
    IF ("Asset Status" > 0)  
        IF ("In Service Date" < "Purchase Date")  
            "In Service Date" = "Purchase Date";
        ENDFI
        "Basic Cost" = "Acquisition Costs" + "Additional Charges";
        "Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
        "Salvage" = "Salvage Input" * "Asset Units";
    ENDFI
)
ENDFIX

FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"(
/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
/* Initialized all the variables required to calculate depreciation */
paramsSet = 1;

/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"-
Amortization Method:

insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
    impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
    impairmentDate1 = #MISSING;
ENDIF
IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF

impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
ELSE
    preExistingPers = 0;
    setStartAmort = 0;
ENDIF

numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF

ENDIF
IF (@ISMBR("No Year"))
    paramsSet = 0;
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
/* clear out any previously calculated values */
"Amortization" = #MISSING;
"Accumulated Amortization" = #MISSING;
"Intangible Assets Finite, Gross" = #MISSING;
"Loss/(Gain) on Sale of Intangibles" = #MISSING;
"Proceeds from Sale of Intangibles" = #MISSING;
"Cash Flow Allocator" = #MISSING;
"Funding Allocator" = #MISSING;
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Impairment of Assets" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Reserve" = #MISSING;
"Capital Expenditure" = #MISSING;
"Retirement Expenses" = #MISSING;

/* Commented out for now need to verify */
IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
"Capital Expenditure" = basicCost - retirementObs;
ENDIF

/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate OR (setStartAmort == 1))
setStartAmort = 0;
periodOffset = 0;
umAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
intangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
numAmortPeriods = numAmortPeriods - preExistingPers;
IF (numAmortPeriods > 0)
priorAccumAmort = amortAmt * preExistingPers;
ENDIF
ENDIF
numRemainingPers = numAmortPeriods;
ENDIF

/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)
/* If capitalized book in Capital Reserve instead of Impairment */
IF ("TP-Date" == impairmentDate1)
IF (impairmentOpt == 1)
"Impairment of Assets" = intangibleNet - impairmentFairVal1;
ENDIF
ENDIF
Predefined Business Rules 115
ELSEIF (impairmentOpt == 2)
"Capital Reserve" = intangibleNet - impairmentFairVal1;
ELSEIF (impairmentOpt == 3)
"Capital Reserve" = (intangibleNet - impairmentFairVal1) *
capitalizePct;
"Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
ENDIF

basicCost = basicCost - (intangibleNet - impairmentFairVal1);
amortAmt = impairmentFairVal1 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)
IF (impairmentOpt == 1)
"Impairment of Assets" = intangibleNet - impairmentFairVal2;
ELSEIF (impairmentOpt == 2)
"Capital Reserve" = intangibleNet - impairmentFairVal2;
ELSEIF (impairmentOpt == 3)
"Capital Reserve" = (intangibleNet - impairmentFairVal2) *
capitalizePct;
"Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
ENDIF
basicCost = basicCost - (intangibleNet - impairmentFairVal2);
amortAmt = impairmentFairVal2 / numRemainingPers;
ENDIF

IF (periodOffset + 1 == numAmortPeriods)
amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF

priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
 "Intangible Assets Finite, Gross" = basicCost;
 "Amortization" = amortAmt;
 "Accumulated Amortization" = priorAccumAmort;
 "Insurance" = insuranceCost;
 "Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
 "No Year"->"BegBalance"->"Reason Ended" == 3)
netValue = basicCost - priorAccumAmort + retirementCosts;
IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
 "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue + amortAmt;
ELSE
 "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
ENDIF
"Retirement Expenses" = retirementCosts - retirementObs;
"Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
ENDIF
periodOffset = periodOffset + 1;
numRemainingPers = numRemainingPers - 1;

ENDIF
ENDIF
ENDIF

); ENDFIX

ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
    @ANCESTORS([[LineItem]]);
    @ANCESTORS([AssetClassInt]);
ENDFIX

CalcExistDepr

Description
Calculates depreciation for an existing asset.

Formula

SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR deprAmtLast = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR purchaseDate = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR retirementObs = 0;
VAR retirementCosts = 0;
VAR retireOption = 0;
VAR saleValue = 0;
VAR netValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR paramsSet = 0;
VAR preExistingPers = 0;
VAR setStartDepr = 0;

FIX([AssetClass], [Hidden_Scenario], [Hidden_Version], [Department])

FIX ("BegBalance", "No Year")
"Basic Cost" {
IF ("Asset Status" > 0)
IF ("In Service Date" < "Purchase Date")
"In Service Date" = "Purchase Date";
ENDIF
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
ENDIF
ENDFIX

FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
paramsSet = 1;
/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

depMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
depConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
  yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
  monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
  IF (monthVal < 0)
    IF (yearVal > 0)
      yearVal = yearVal - 1;
    ENDIF
  monthVal = 1200 + monthVal;
ENDIF
preExistingPers = yearVal * 12 + monthVal / 100;
setStartDepr = 1;
ELSE
preExistingPers = 0;
setStartDepr = 0;
ENDIF

/* setup for depreciation */
numDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
priorAccumDepr = 0;
periodicPriorAccumDepr = 0;
dprAmt = 0;
/* Add a periodic vs annual property instead of additional deprMethods */
/* Annual Methods SumYearDigits =2 DecliningBalance Year = 3*/
IF (deprMethod == 2 OR deprMethod == 3)
isAnnual = 1;
  life = "No Year"->"Useful Life (in Years)";
persInSection = "NumPeriods";
ELSE /* Periodic Methods */
isAnnual = 0;
  life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
persInSection = 1;
ENDIF

yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
  /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
dprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
dprMethod = 1;
dprConvention = 1;
ENDIF

ENDIF

ENDIF

IF (@ISMBR("No Year"))
    paramsSet = 0;
    ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Depreciation" = #MISSING;
    "Accumulated Depreciation" = #MISSING;
    "Property, Plant and Equipment Gross" = #MISSING;
    "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
    "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;

    "Cash Flow Allocator" = #MISSING;
    "Funding Allocator" = #MISSING;
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Repairs" = #MISSING;
    "Retirement Expenses" = #MISSING;
    "Capital Expenditure" = #MISSING;

    /* Commented out for now need to verify
    IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
    ENDIF
    */
    /* Check if we should start depreciation */
    IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1)
        OR
        (deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1) AND "TP-Date" > yearVal)
        setStartDepr = 0;
        periodOffset = 0;
        persIn1stYear = "NumPeriods";
        numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
        totDeprPeriods = numDeprPeriods;
        dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
        monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
        delayStart = 0;
        IF (deprConvention == 2) /* Prorate Actual Date */
            IF (dayVal == 1) /* Treat same as Begin Period */
                split1stAmt = 0;
                ELSE
                ENDIF
            ELSE
                ENDIF
        ENDIF
    ENDIF

    "Capital Expenditure" = basicCost - retirementObs;
ENDIF
/* Commented out for now need to verify */
IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    "Capital Expenditure" = basicCost - retirementObs;
ENDIF
*/
numDeprPeriods = numDeprPeriods + 1;
split1stAmt = 1;
ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
    numDeprPeriods = numDeprPeriods + 1;
split1stAmt = 1;
ELSEIF (deprConvention == 4) /* MidYear */
    split1stAmt = 0;
delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
    split1stAmt = 0;
ENDIF

IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numDeprPeriods = numDeprPeriods - preExistingPers;
    IF (numDeprPeriods > 0)
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
        periodicPriorAccumDepr = deprAmt * preExistingPers;
    ENDIF
ENDIF

ENDIF

/* Depreciation calcuation section */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
    lifeIndex = @INT(periodOffset / persInSection);
    IF (deprMethod == 1) /* Straight Line */
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
    ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
        IF (lifeIndex == 0)
            /* cost * rate * month / 12; for 1st period */
            deprAmt = basicCost * deprRate * persIn1stYear / "NumPeriods" / persInSection;
        ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
            deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
       ENDIF
    ELSEIF (deprMethod == 2) /* Sum of Years Digits */
        deprRate = ((life - lifeIndex) * 2 / (life * (life +1)));
        deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
    ENDIF
ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
    deprRate = ((life - lifeIndex) * 2 / (life * (life +1)));
    deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
    IF (deprConvention == 2)
        /* assume that there are 30 days in each month */
deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
ELSEIF (deprConvention == 4)

ELSE
deprAmt = deprAmt - deprAmt / 2;
ENDIF
split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
priorAccumDepr = periodicPriorAccumDepr;
ENDIF
ELSE
priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND ( prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))

"Property, Plant and Equipment Gross" = basicCost;
"Depreciation" = deprAmt;
"Accumulated Depreciation" = periodicPriorAccumDepr;
"Insurance" = insuranceCost;
"Maintenance" = maintenanceCost;
"Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
 "No Year"->"BegBalance"->"Reason Ended" == 3)
netValue = basicCost - priorAccumDepr + retirementCosts;
IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
"Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue -
saleValue + deprAmt;
ELSE
"Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue +
deprAmt;
ENDIF
"Retirement Expenses" = retirementCosts - retirementObs;
"Proceeds from Sale of Property, Plant and Equipment" = saleValue -
retirementCosts;
ENDIF
ImpairIntangible

Description
Impairs an intangible asset.

Formula

SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR newValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR isNew = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX([LineItem], [Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])

FIX("No Year", "BegBalance")
"Asset Description" ( 
IF ("Asset Description" <> #MISSING)
IF ("Impairment Date1" == #MISSING)
  "Impairment Date1" = [ImpairmentDate];
  "Impairment Fair Value1" = [FairValue];
ELSE
  "Impairment Date2" = [ImpairmentDate];
  "Impairment Fair Value2" = [FairValue];
ENDIF
IF ("Impairment Date2" <> #MISSING AND "Impairment Date2" <= "Impairment Date1")
  "Impairment Date1" = [ImpairmentDate];
  "Impairment Fair Value1" = [FairValue];
  "Impairment Date2" = #MISSING;
  "Impairment Fair Value2" = #MISSING;
ENDIF
IF ("Impairment Date1" <> #MISSING)
  impairmentDate1 = @INT("Impairment Date1" / 100) * 100;
ELSE
  impairmentDate1 = #MISSING;
ENDIF
IF ("Impairment Date2" <> #MISSING)
  impairmentDate2 = @INT("Impairment Date2" / 100) * 100;
ELSE
  impairmentDate2 = #MISSING;
ENDIF
"Impairment Option" = [ImpairmentOpt];
"Partial Capitalize %" = [CapitalizePct];
impairmentFairVal1 = "Impairment Fair Value1";
impairmentFairVal2 = "Impairment Fair Value2";
IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
ENDIF
);
ENDFIX

/* Amortization Section */
FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"("No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization")

/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
   /* Initialized all the variables required to calculate depreciation */
   paramsSet = 1;
   /* Check if we are a descendant of a new Asset */
   IF (@ISDESC("Total New"))
      isNew = 1;
   ELSE
      isNew = 0;
   ENDIF

   /* eliminate days from date */
   delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
   prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
   purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
   inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
   cashOutflowDate = purchaseDate;
   fundingDate = purchaseDate;

   cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
   IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
      cashflowChanged = 1;
   ELSE
      cashflowChanged = 0;
   ENDIF

   IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
      cashStaggeredPers = 4;
   ELSE
      /* Extract year and month from date to adjust based in incidence values in global assumptions */
      yearVal = @INT(cashOutflowDate / 10000) * 10000;
      monthVal = cashOutflowDate - yearVal;
      monthVal = monthVal + cashFlowIncidence;
      IF (monthVal <= 0)
         yearVal = yearVal - 10000;
         monthVal = 1200 + monthVal;
      ELSEIF (monthVal > 1200)
yearVal = yearVal + 10000;
monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence <> "No Year"->"Funding Incidence")
fundingChanged = 1;
ELSE
  fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
fundingStaggeredPers = 4;
ELSE
  yearVal = @INT(fundingDate / 10000) * 10000;
  monthVal = fundingDate - yearVal;
  monthVal = monthVal + fundingIncidence;
  IF (monthVal <= 0)
    yearVal = yearVal - 10000;
    monthVal = 1200 + monthVal;
  ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
  ENDIF
  fundingDate = yearVal + monthVal;
  fundingStaggeredPers = 1;
ENDIF

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization Method";

insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
    impairmentDate1 = #MISSING;
ENDIF
IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF
impairmentFairVal1 =  "No Year"->"Impairment Fair Value1";
impairmentFairVal2 =  "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
    setStartAmort = 1;
ELSE
    preExistingPers = 0;
    setStartAmort = 0;
ENDIF
numAmortPeriods = 0;
periodOffset = 0;
IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    setCashImpact = 0;
    setFundingImpact = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (@ISMBR("No Year"))
        paramsSet = 0;
    ENDIF
    IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
        /* clear out any previously calculated values */
        "Amortization" = #MISSING;
        "Accumulated Amortization" = #MISSING;
        "Intangible Assets Finite, Gross" = #MISSING;
        "Loss/(Gain) on Sale of Intangibles" = #MISSING;
    ENDIF
ENDIF
"Proceeds from Sale of Intangibles" = #MISSING;

IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
    "Cash Flow Allocator" = #MISSING;
ENDIF
IF (fundingIncidence <> -1 OR fundingChanged == 1)
    "Funding Allocator" = #MISSING;
ENDIF
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Impairment of Assets" = #MISSING;
"Capital Reserve" = #MISSING;
"Capital Expenditure" = #MISSING;
"Retirement Expenses" = #MISSING;

/*/ Only calculate cash impact and funding for new assets */
IF (isNew == 1)
    IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
    ENDIF
    IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setCashImpact = cashStaggeredPers;
        cashAllocPct = 1 / cashStaggeredPers;
    ENDIF
    IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setFundingImpact = fundingStaggeredPers;
        fundingAllocPct = 1 / fundingStaggeredPers;
    ENDIF
    IF (setCashImpact > 0)
        "Cash Flow Allocator" = cashAllocPct;
        setCashImpact = setCashImpact - 1;
    ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
    "Funding Allocator" = fundingAllocPct;
    setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/*/ Check if we should start amortization */
IF ("TP-Date" == inServiceDate) OR (setStartAmort == 1))
    setStartAmort = 0;

Capital Asset Planning Structure
periodOffset = 0;
numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
intangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numAmortPeriods = numAmortPeriods - preExistingPers;
    IF (numAmortPeriods > 0)
        priorAccumAmort = amortAmt * preExistingPers;
    ENDIF
ENDIF

numRemainingPers = numAmortPeriods;
ENDIF

/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)

    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate1)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
        "Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal1);
    amortAmt = impairmentFairVal1 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)
    IF (impairmentOpt == 1)
        "Impairment of Assets" = intangibleNet - impairmentFairVal2;
    ELSEIF (impairmentOpt == 2)
        "Capital Reserve" = intangibleNet - impairmentFairVal2;
    ELSEIF (impairmentOpt == 3)
        "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
    "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal2);
    amortAmt = impairmentFairVal2 / numRemainingPers;
ENDIF

IF (periodOffset + 1 == numAmortPeriods)
    amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF

priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;
/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEnd)

"Intangible Assets Finite, Gross" = basicCost;
"Amortization" = amortAmt;
"Accumulated Amortization" = priorAccumAmort;
"Insurance" = insuranceCost;
"Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEnd)

IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)

netValue = basicCost - priorAccumAmort + retirementCosts;
IF ("No Year"->"BegBalance"->"Reason Ended" == 3)

"Loss/(Gain) on Sale of Intangibles" = netValue - saleValue + amortAmt;
ELSE

"Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
ENDIF
"Retirement Expenses" = retirementCosts - retirementObs;
"Proceeds from Sale of Intangibles" = saleValue - retirementCosts;

ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF

ENDIF

ENDIF

ENDIF

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" ( IF (cashflowChanged == 1 AND "Asset Status" > 0)

"Cash Flow Incidence" = cashFlowIncidence;
ENDIF
IF (fundingChanged == 1 AND "Asset Status" > 0)

"Funding Incidence" = fundingIncidence;
ENDIF
)
ENDFIX

ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
@IANCESTORS([LineItem]);
@ANCESTORS([AssetClassInt]);
ENDFIX
**Improve Asset**

**Description**

Adds an improvement to an asset.

**Formula**

```
SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR paramsSet = 0;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR numAsset = 1;
VAR preExistingPers = 0;
VAR setStartDepr = 0;
```
FIX(@SIBLINGS([LineItem]), [Hidden_Scenario], [Hidden_Version], [Department], [AssetClass])

/* Setup properties for new Asset */
FIX("No Year", "BegBalance")
"Asset Description" (
IF (numAsset > 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING )
numAsset = numAsset - 1;
"Asset Status" = 0; /* Indicate that we have a new asset */
"Asset Description" = [AssetDesc];
"Asset Units" = [AssetUnits];
"Asset Rate" = [AssetRate];
"Installation" = [Installation];
"Freight" = [Freight];
"Salvage Input" = [SalvageValue];
"Salvage" = [SalvageValue] * "Asset Units";
"Taxes %" = [TaxesPct];
"Additional Charges" = [AdditionalCharges];
"Retirement Obligation" = [RetirementObs];
"Physical Location" = [PhyLocation];
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Purchase Date" = [PurchaseDate];
"In Service Date" = [InserviceDate];
IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF
ENDIF
);
ENDFIX

/* Calculate depreciation section */
FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"(
/* Only process newly added asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" == 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
paramsSet = 1;
/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;

cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
IF (cashFlowIncidence == -1)
cashStaggeredPers = 4;
ELSE
    /* Extract year and month from date to adjust based in incidence values in
global assumptions */
    yearVal = @INT(cashOutflowDate / 10000) * 10000;
    monthVal = cashOutflowDate - yearVal;
    monthVal = monthVal + cashFlowIncidence;
    IF (monthVal <= 0)
        yearVal = yearVal - 10000;
        monthVal = 1200 + monthVal;
    ELSEIF (monthVal > 1200)
        yearVal = yearVal + 10000;
        monthVal = monthVal - 1200;
    ENDIF
    cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";
IF (fundingIncidence == -1)
fundingStaggeredPers = 4;
ELSE
    yearVal = @INT(fundingDate / 10000) * 10000;
    monthVal = fundingDate - yearVal;
    monthVal = monthVal + fundingIncidence;
    IF (monthVal <= 0)
        yearVal = yearVal - 10000;
        monthVal = 1200 + monthVal;
    ELSEIF (monthVal > 1200)
        yearVal = yearVal + 10000;
        monthVal = monthVal - 1200;
    ENDIF
    fundingDate = yearVal + monthVal;
fundingStaggeredPers = 1;
ENDIF

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";
deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
deprConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";
fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
  yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
  monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
  IF (monthVal < 0)
    IF (yearVal > 0)
      yearVal = yearVal - 1;
    ENDIF
    monthVal = 1200 + monthVal;
  ENDIF
  preExistingPers = yearVal * 12 + monthVal / 100;
  setStartDepr = 1;
ELSE
  preExistingPers = 0;
  setStartDepr = 0;
ENDIF

/* setup for depreciation */
umDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
  priorAccumDepr = 0;
  periodicPriorAccumDepr = 0;
  deprAmt = 0;
  /* Add a periodic vs annual property instead of additional deprMethods */
  /* Annual Methods SumYearDigits = 2, DecliningBalance Year = 3*/
  IF (deprMethod == 2 OR deprMethod == 3)
    isAnnual = 1;
    life = "No Year"->"Useful Life (in Years)";
    persInSection = "NumPeriods";
  ELSE /* Periodic Methods */
    isAnnual = 0;
    life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
    persInSection = 1;
  ENDIF
  yearVal = @INT(inServiceDate / 10000) * 10000;
  /* DecliningBalance methods */
  IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
  ENDIF
  /* Reset depr method to SLN convention to prorate 1st period for preExisting...
assets */
    IF (preExistingPers > 0)
        deprMethod = 1;
        deprConvention = 1;
    ENDIF

    ENDIF

ENDIF

/* Set Asset Status to active when we hit the end */
IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF

/* Start depreciation calculations */
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Depreciation" = #MISSING;
    "Accumulated Depreciation" = #MISSING;
    "Property, Plant and Equipment Gross" = #MISSING;
    "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
    "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;
    "Cash Flow Allocator" = #MISSING;
    "Funding Allocator" = #MISSING;
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Repairs" = #MISSING;
    "Retirement Expenses" = #MISSING;
    "Capital Expenditure" = #MISSING;

    IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
    ENDIF

    /* Check if we should start setting cash flow impact values */
    IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setCashImpact = cashStaggeredPers;
        cashAllocPct = 1 / cashStaggeredPers;
    ENDIF

    /* Check if we should start setting funding values */
    IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setFundingImpact = fundingStaggeredPers;
        fundingAllocPct = 1 / fundingStaggeredPers;
    ENDIF
IF (setCashImpact > 0)
    "Cash Flow Allocator" = cashAllocPct;
    setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
    "Funding Allocator" = fundingAllocPct;
    setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/* Check if we should start depreciation */
IF (numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1)
OR
    (deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1) AND "TP-Date" > yearVal)
setStartDepr = 0;
periodOffset = 0;
persIn1stYear = "NumPeriods"; /* not adjusted - "Cal TP-Index" + 1; */
numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
totDeprPeriods = numDeprPeriods;
dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
delayStart = 0;
IF (deprConvention == 2) /* Prorate Actual Date */
    IF (dayVal == 1)  /* Treat same as Begin Period */
        split1stAmt = 0;
    ELSE
        numDeprPeriods = numDeprPeriods + 1;
        split1stAmt = 1;
    ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
    numDeprPeriods = numDeprPeriods + 1;
    split1stAmt = 1;
ELSEIF (deprConvention == 4)  /* MidYear */
    split1stAmt = 0;
    delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
    split1stAmt = 0;
ENDIF

/* Declining balance methods */
IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

136  Capital Asset Planning Structure
/** Adjust for pre existing assets */
IF (preExistingPers > 0)
    numDeprPeriods = numDeprPeriods - preExistingPers;
    IF (numDeprPeriods > 0)
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
        periodicPriorAccumDepr = deprAmt * preExistingPers;
    ENDIF
ENDIF
ENDIF

/* Calculate Depreciation amounts */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
    lifeIndex = @INT(periodOffset / persInSection);
    IF (deprMethod == 1) /* Straight Line */
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
    ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
        IF (lifeIndex == 0) /* cost * rate * month / 12; for 1st period */
            deprAmt = basicCost * deprRate * persIn1stYear / "NumPeriods" / persInSection;
        ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
            deprAmt = (basicCost - periodicPriorAccumDepr) * deprRate / persInSection;
        ENDIF
    ELSEIF (deprMethod == 2) /* Sum of Years Digits */
        deprRate = (life - lifeIndex) * 2 / (life * (life +1));
        deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
    ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
    deprRate = (life - lifeIndex) * 2 / (life * (life +1));
    deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
    IF (deprConvention == 2)
        /* assume that there are 30 days in each month */
        deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
    ELSEIF (deprConvention == 4)
        ;
    ELSE
        deprAmt = deprAmt - deprAmt / 2;
    ENDIF
    split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
    deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
    "Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumDepr + retirementCosts;
        IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue - saleValue + deprAmt;
        ELSE
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue + deprAmt;
        ENDIF
    "Retirement Expenses" = retirementCosts - retirementObs;
    "Proceeds from Sale of Property, Plant and Equipment" = saleValue - retirementCosts;
    ENDIF
    periodOffset = periodOffset +1;
ENDIF
    delayStart = delayStart -1;
ENDIF
ENDIF

); ENDFIX

ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
CALC DIM ("Period"); @ANCESTORS([LineItem]); @ANCESTORS([AssetClass]);
ENDFIX
**Remove Asset**

**Description**
Removes an asset.

**Formula**

```sql
SET UPDATECALC OFF;
SET AGGMISSG ON;

FIX ([Hidden_Scenario], [Hidden_Version], [Department], [AllAssetClass])
    CLEARDATA [LocalLineItem];
ENDFIX

FIX ([Hidden_Scenario], [Hidden_Version], [Department])
    @ANCESTORS([LocalLineItem]);
    @ANCESTORS([AllAssetClass]);
ENDFIX
```

**Retire Asset**

**Description**
Retires an asset.

**Formula**

```sql
SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR deprAmtLast = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR retirementObs = 0;
VAR retirementCosts = 0;
```
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR retireOption = 0;
VAR saleValue = 0;
VAR netValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR paramsSet = 0;
VAR isNew = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartDepr = 0;

FIX([LineItem], [AssetClass], [Hidden_Scenario], [Hidden_Version], [Department])

FIX("No Year", "BegBalance")
"Asset Description" (
  IF ("Asset Status" >= 0)
    "Premature End Date" = [RetireDate];
    "Retirement Options" = [RetireOption];
  IF ("Retirement Options" == 1)
    "Reason ended" = 3;
  ELSE
    "Reason ended" = 2;
  ENDIF
  "Sale Value" = [SaleValue];
  "Retirement Costs" = [RetirementCosts];
  "Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
  "Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
  "Salvage" = "Salvage Input" * "Asset Units";
  IF ("In Service Date" < "Purchase Date")
    "In Service Date" = "Purchase Date";
  ENDIF
ENDIF
);
ENDFIX

/* Depreciation section */
FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"

IF ("No Year"->"BegBalance"->"Asset Status" >= 0)

IF (@ISMBR("BegBalance") AND paramsSet == 0)
    paramsSet = 1;

    /* Check if we are a descendant of a new Asset */
    IF (@ISDESC("Total New"))
        isNew = 1;
    ELSE
        isNew = 0;
    ENDIF

    /* Initialized all the variables required to calculate depreciation */
    /* eliminate days from date */
    delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
    prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
    purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
    inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
    cashOutflowDate = purchaseDate;
    fundingDate = purchaseDate;

    cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";

    IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
        cashflowChanged = 1;
    ELSE
        cashflowChanged = 0;
    ENDIF

    IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
        cashStaggeredPers = 4;
    ELSE
        /* Extract year and month from date to adjust based in incidence values in
global assumptions */
        yearVal = @INT(cashOutflowDate / 10000) * 10000;
        monthVal = cashOutflowDate - yearVal;
        monthVal = monthVal + cashFlowIncidence;

        IF (monthVal <= 0)
            yearVal = yearVal - 10000;
            monthVal = 1200 + monthVal;
        ELSEIF (monthVal > 1200)
            yearVal = yearVal + 10000;
            monthVal = monthVal - 1200;
        ENDIF

        cashOutflowDate = yearVal + monthVal;
        cashStaggeredPers = 1;
    ENDIF

    fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

    IF (fundingIncidence <> "No Year"->"Funding Incidence")
fundingChanged = 1;
ELSE
  fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
  fundingStaggeredPers = 4;
ELSE
  yearVal = @INT(fundingDate / 10000) * 10000;
  monthVal = fundingDate - yearVal;
  monthVal = monthVal + fundingIncidence;
  IF (monthVal <= 0)
    yearVal = yearVal - 10000;
    monthVal = 1200 + monthVal;
  ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
  ENDIF
  fundingDate = yearVal + monthVal;
  fundingStaggeredPers = 1;
ENDIF

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

  deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
  deprConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
  insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
  maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
  repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";
  fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

  /* check if we have a pre-existing asset */
  IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
      IF (yearVal > 0)
        yearVal = yearVal - 1;
      ENDFIL
    monthVal = 1200 + monthVal;
ENDIF
preExistingPers = yearVal * 12 + monthVal / 100;
setStartDepr = 1;
ELSE
preExistingPers = 0;
setStartDepr = 0;
ENDIF

/* setup for depreciation */
numDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
priorAccumDepr = 0;
periodicPriorAccumDepr = 0;
deprAmt = 0;
/* Add a periodic vs annual property instead of additional deprMethods */
/* Annual Methods SumYearDigits =2 DecliningBalance Year = 3*/
IF (deprMethod == 2 OR deprMethod == 3)
isAnnual = 1;
life = "No Year"->"Useful Life (in Years)";
persInSection = "NumPeriods";
ELSE /* Periodic Methods */
isAnnual = 0;
life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
persInSection = 1;
ENDIF

setCashImpact = 0;
setFundingImpact = 0;
yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
/* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */

depRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
depMethod = 1;
depConvention = 1;
ENDIF

ENDIF

ENDIF

IF (@ISMBR("No Year"))
paramsSet = 0;
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
/* clear out any previously calculated values */
"Depreciation" = #MISSING;
"Accumulated Depreciation" = #MISSING;
"Property, Plant and Equipment Gross" = #MISSING;
"Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
"Proceeds from Sale of Property, Plant and Equipment" = #MISSING;

IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
  "Cash Flow Allocator" = #MISSING;
ENDIF
IF (fundingIncidence <> -1 OR fundingChanged == 1)
  "Funding Allocator" = #MISSING;
ENDIF
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Repairs" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Expenditure" = #MISSING;

/* Check if we should start setting cash flow impact values */
IF (isNew == 1)
  IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    "Capital Expenditure" = basicCost - retirementObs;
  ENDIF
  IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
    setCashImpact = cashStaggeredPers;
    cashAllocPct = 1 / cashStaggeredPers;
  ENDIF
/* Check if we should start setting funding values */
IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
  setFundingImpact = fundingStaggeredPers;
  fundingAllocPct = 1 / fundingStaggeredPers;
ENDIF
IF (setCashImpact > 0)
  "Cash Flow Allocator" = cashAllocPct;
  setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";
IF (setFundingImpact > 0)
  "Funding Allocator" = fundingAllocPct;
  setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

ENDIF
/* Check if we should start depreciation */
IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1)
OR
(deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1 AND "TP-Date" > yearVal))

setStartDepr = 0;
periodOffset = 0;
persIn1stYear = "NumPeriods";
numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years");
totDeprPeriods = numDeprPeriods;

dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);

delayStart = 0;
IF (deprConvention == 2) /* Prorate Actual Date */
   IF (dayVal == 1) /* Treat same as Begin Period */
      split1stAmt = 0;
   ELSE
      numDeprPeriods = numDeprPeriods + 1;
      split1stAmt = 1;
   ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
   numDeprPeriods = numDeprPeriods + 1;
   split1stAmt = 1;
ELSEIF (deprConvention == 4) /* MidYear */
   split1stAmt = 0;
   delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
   split1stAmt = 0;
ENDIF

IF (deprMethod == 3 OR deprMethod == 4)
   /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
   deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
   numDeprPeriods = numDeprPeriods - preExistingPers;
   IF (numDeprPeriods > 0)
      deprAmt = (basicCost - salvageVal) / totDeprPeriods;
      periodicPriorAccumDepr = deprAmt * preExistingPers;
   ENDIF
ENDIF

ENDIF

/* Depreciation calculation section */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
   lifeIndex = @INT(periodOffset / persInSection);
ENDIF
IF (deprMethod == 1) /* Straight Line */
    deprAmt = (basicCost - salvageVal) / totDeprPeriods;
ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
    IF (lifeIndex == 0)
        /* cost * rate * month / 12; for 1st period */
        deprAmt = basicCost * deprRate * persIn1stYear / "NumPeriods" / persInSection;
    ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
        deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
    ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
    deprRate = (life - lifeIndex) * 2 / (life * (life +1));
    deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
    IF (deprConvention == 2)
        /* assume that there are 30 days in each month */
        deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
    ELSEIF (deprConvention == 4);
    ELSE
        deprAmt = deprAmt - deprAmt / 2;
    ENDIF
    split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
    deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
    IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
        priorAccumDepr = periodicPriorAccumDepr;
    ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
ENDIF
"Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
        "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumDepr + retirementCosts;
        IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue -
            saleValue + deprAmt;
        ELSE
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue +
            deprAmt;
        ENDIF
        "Retirement Expenses" = retirementCosts - retirementObs;
        "Proceeds from Sale of Property, Plant and Equipment" = saleValue -
        retirementCosts;
    ENDIF
    periodOffset = periodOffset +1;
ENDIF
delayStart = delayStart -1;
ENDIF
ENDIF

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" (  
    IF (cashflowChanged == 1 AND "Asset Status" > 0)
        "Cash Flow Incidence" = cashFlowIncidence;
    ENDIF
    IF (fundingChanged == 1 AND "Asset Status" > 0)
        "Funding Incidence" = fundingIncidence;
    ENDIF
)
ENDFIX

ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
CALC DIM ("Period");
@ANCESTORS([LineItem]);
@ANCESTORS([AssetClass]);
ENDFIX

Retire Intangible

Description
Retires an intangible asset.

Formula
SET CREATENONMISSINGBLK ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR isNew = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

FIX([LineItem], [Hidden_Scenario], [Hidden_Version], [Department], [AssetClassInt])

FIX("No Year", "BegBalance")
"Asset Description" (  
IF ("Asset Status" >= 0)  
"Premature End Date" = [RetireDate];  
"Retirement Options" = [RetireOption];  
IF ("Retirement Options" == 1)  
"Reason ended" = 3;
ELSE
    "Reason ended" = 2;
ENDIF
"Sale Value" = [SaleValue];
"Retirement Costs" = [RetirementCosts];
IF ("In Service Date" < "Purchase Date")
    "In Service Date" = "Purchase Date";
ENDIF
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
ENDIF
); ENDIF
ENDFIX

/* Amortization section */
FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"(

/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
    /* Initialized all the variables required to calculate depreciation */
    paramsSet = 1;
    /* Check if we are a descendant of a new Asset */
    IF (@ISDESC("Total New"))
        isNew = 1;
    ELSE
        isNew = 0;
    ENDIF
    /* eliminate days from date */
    delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
    prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
    purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
    inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
    cashOutflowDate = purchaseDate;
    fundingDate = purchaseDate;
    cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
    IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
        cashflowChanged = 1;
    ELSE
        cashflowChanged = 0;
    ENDIF
    IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
        cashStaggeredPers = 4;
    ELSE
        /* Extract year and month from date to adjust based in incidence values in global assumptions */
        yearVal = @INT(cashOutflowDate / 10000) * 10000;

Predefined Business Rules 149
monthVal = cashOutflowDate - yearVal;
monthVal = monthVal + cashFlowIncidence;

IF (monthVal <= 0)
  yearVal = yearVal - 10000;
  monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
  yearVal = yearVal + 10000;
  monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence <> "No Year"->"Funding Incidence")
fundingChanged = 1;
ELSE
  fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
fundingStaggeredPers = 4;
ELSE
  yearVal = @INT(fundingDate / 10000) * 10000;
  monthVal = fundingDate - yearVal;
  monthVal = monthVal + fundingIncidence;

  IF (monthVal <= 0)
    yearVal = yearVal - 10000;
    monthVal = 1200 + monthVal;
  ELSEIF (monthVal > 1200)
    yearVal = yearVal + 10000;
    monthVal = monthVal - 1200;
  ENDIF

  fundingDate = yearVal + monthVal;
  fundingStaggeredPers = 1;
ENDIF

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

  amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization Method";

  insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
    impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
ELSE
    impairmentDate1 = #MISSING;
ENDIF

IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF

impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
ELSE
    preExistingPers = 0;
    setStartAmort = 0;
ENDIF

numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    setCashImpact = 0;
    setFundingImpact = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF

ENDIF

IF (@ISMBR("No Year"))
    paramsSet = 0;
ENDIF
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)

    /* clear out any previously calculated values */
    "Amortization" = #MISSING;
    "Accumulated Amortization" = #MISSING;
    "Intangible Assets Finite, Gross" = #MISSING;
    "Loss/(Gain) on Sale of Intangibles" = #MISSING;
    "Proceeds from Sale of Intangibles" = #MISSING;

    IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
        "Cash Flow Allocator" = #MISSING;
    ENDIF

    IF (fundingIncidence <> -1 OR fundingChanged == 1)
        "Funding Allocator" = #MISSING;
    ENDIF

    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Impairment of Assets" = #MISSING;
    "Capital Reserve" = #MISSING;
    "Capital Expenditure" = #MISSING;
    "Retirement Expenses" = #MISSING;

    IF (isNew == 1)
      IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        "Capital Expenditure" = basicCost - retirementObs;
      ENDIF

      IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setCashImpact = cashStaggeredPers;
        cashAllocPct = 1 / cashStaggeredPers;
      ENDIF

      IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
        setFundingImpact = fundingStaggeredPers;
        fundingAllocPct = 1 / fundingStaggeredPers;
      ENDIF

      IF (setCashImpact > 0)
        "Cash Flow Allocator" = cashAllocPct;
        setCashImpact = setCashImpact - 1;
      ENDIF

      "Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

      IF (setFundingImpact > 0)
        "Funding Allocator" = fundingAllocPct;
        setFundingImpact = setFundingImpact - 1;
      ENDIF

      "Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
      "Long Term Debt" = fundingAmt * "Funding Allocator";
ENDIF

/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate) OR (setStartAmort == 1)
setStartAmort = 0;
periodOffset = 0;
numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
tangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

ENDIF /* Adjust for pre existing assets */
IF (preExistingPers > 0)
numAmortPeriods = numAmortPeriods - preExistingPers;
IF (numAmortPeriods > 0)
priorAccumAmort = amortAmt * preExistingPers;
ENDIF
numRemainingPers = numAmortPeriods;

ENDIF /* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)

/* If capitalized book in Capital Reserve instead of Impairment */
IF ("TP-Date" == impairmentDate1)
IF (impairmentOpt == 1)
"Impairment of Assets" = intangibleNet - impairmentFairVal1;
ELSEIF (impairmentOpt == 2)
"Capital Reserve" = intangibleNet - impairmentFairVal1;
ELSEIF (impairmentOpt == 3)
"Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
"Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve"
ENDIF
basicCost = basicCost - (intangibleNet - impairmentFairVal1);
amortAmt = impairmentFairVal1 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)
IF (impairmentOpt == 1)
"Impairment of Assets" = intangibleNet - impairmentFairVal2;
ELSEIF (impairmentOpt == 2)
"Capital Reserve" = intangibleNet - impairmentFairVal2;
ELSEIF (impairmentOpt == 3)
"Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
"Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve"
ENDIF
basicCost = basicCost - (intangibleNet - impairmentFairVal2);
amortAmt = impairmentFairVal2 / numRemainingPers;
ENDIF
IF (periodOffset + 1 == numAmortPeriods)
    amortAmt = (basicCost - salvageVal) - priorAccumAmort;
ENDIF

priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR
    preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Intangible Assets Finite, Gross" = basicCost;
    "Amortization" = amortAmt;
    "Accumulated Amortization" = priorAccumAmort;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
        "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumAmort + retirementCosts;
        IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
            "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue +
                amortAmt;
        ELSE
            "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
        ENDIF
        "Retirement Expenses" = retirementCosts - retirementObs;
        "Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
    ENDIF
ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF
ENDIF
)
ENDFIX

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" (
    IF (cashflowChanged == 1 AND "Asset Status" > 0)
        "Cash Flow Incidence" = cashFlowIncidence;
    ENDIF
    IF (fundingChanged == 1 AND "Asset Status" > 0)
        "Funding Incidence" = fundingIncidence;
    ENDIF
)
ENDFIX
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [Department])
RollupAssetEntities

Description
Rolls up assets by Entities.

Formula

SET AGGMISSG ON;

FIX ([Scenario], [Version])
   CALC DIM ("Entity");
ENDFIX

RollupAssets

Description
Rolls up assets.

Formula

SET AGGMISSG ON;

FIX ([Scenario], [Version], [Department])
   CALC DIM ("Account", "Period", "Asset Class", "Line Item");
ENDFIX

Transfer Asset

Description
Transfers a new asset.

Formula

SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR deprAmtLast = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR retirementObs = 0;
VAR retirementCosts = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR retireOption = 0;
VAR saleValue = 0;
VAR xferAsset = 0;
VAR paramsSet = 0;
VAR netValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartDepr = 0;

/* Transfer section: Locate either an asset with same Asset ID or a new one for transfer */
FIX(@CHILDREN("Total New"), [Hidden_Scenario], [Hidden_Version], [DepartmentTo],
   [AssetClass], "No Year", "BegBalance")
"Asset Description" (}
IF (NOT @ISMBR([DepartmentFrom]))
  IF ((@MAXS(SKIPMISSING, @CHILDREN("Asset Properties"))) == #MISSING) OR (*Asset ID* == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING))
  IF (*Asset ID* == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)
/* Clear out any duplicate Assets based on Asset ID if already transferred */
IF (xferAsset == 1)
"Asset Status" = -1; /* mark line item for deletion */
ELSE
xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";
"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";
"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";
IF ("In Service Date" < "Purchase Date")
"In Service Date" = "Purchase Date";
ENDIF
"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ELSEIF (xferAsset == 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING)

xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retention Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retention Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * 
"Taxes %") + "Freight" + "Additional Charges" + "Retention Obligation" + 
"Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->
"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";

"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->
"Global"->"Funding Incidence";

"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF

"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ENDIF
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [AssetClass],
[SrcLineItem], "No Year", "BegBalance")
"Premature End Date" (
  IF (xferAsset == 1)
    "Asset Status" = 0; /* set status to new to trigger recalc below */
    "Premature End Date" = [TransferDate];
    "Reason Ended" = 1;
    "Sale Value" = #MISSING;
  ENDIF
);
ENDFIX
/* Calculate depreciation after transfer */

FIX(@DESCENDANTS("Total New"), [AssetClass], [Hidden_Scenario], [Hidden_Version],
[DepartmentFrom], [DepartmentTo])

FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"(
IF ("No Year"->"BegBalance"->"Asset Status" == 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
paramsSet = 1;
IF ("No Year"->"In Service Date" < "No Year"->"Purchase Date")
 "No Year"->"In Service Date" = "No Year"->"Purchase Date"; ENDIF

/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
cashOutflowDate = purchaseDate;
fundingDate = purchaseDate;

cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
"Cash Flow Incidence";

IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
cashflowChanged = 1;
ELSE
 cashflowChanged = 0;
ENDIF

IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
cashStaggeredPers = 4;
ELSE
 /* Extract year and month from date to adjust based in incidence values in
global assumptions */
 yearVal = @INT(cashOutflowDate / 10000) * 10000;
 monthVal = cashOutflowDate - yearVal;
 monthVal = monthVal + cashFlowIncidence;

IF (monthVal <= 0)
 yearVal = yearVal - 10000;
 monthVal = 1200 + monthVal;
ELSEIF (monthVal > 1200)
 yearVal = yearVal + 10000;
 monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
"Funding Incidence";
IF (fundingIncidence <> "No Year"->"Funding Incidence")
    fundingChanged = 1;
ELSE
    fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
    fundingStaggeredPers = 4;
ELSE
    yearVal = @INT(fundingDate / 10000) * 10000;
    monthVal = fundingDate - yearVal;
    monthVal = monthVal + fundingIncidence;
    IF (monthVal <= 0)
        yearVal = yearVal - 10000;
        monthVal = 1200 + monthVal;
    ELSEIF (monthVal > 1200)
        yearVal = yearVal + 10000;
        monthVal = monthVal - 1200;
    ENDIF
    fundingDate = yearVal + monthVal;
    fundingStaggeredPers = 1;
ENDIF

/*/ Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

deprrMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
deprrConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";

fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

/*/ check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000); 
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF

160 Capital Asset Planning Structure
ENDIF
preExistingPers = yearVal * 12 + monthVal / 100;
setStartDepr = 1;
ELSE
preExistingPers = 0;
setStartDepr = 0;
ENDIF

/* setup for depreciation */
umDeprPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
priorAccumDepr = 0;
periodicPriorAccumDepr = 0;
deprAmt = 0;
/* Add a periodic vs annual property instead of additional deprMethods */
/* Annual Methods SumYearDigits =2 DecliningBalance Year = 3*/
IF (deprMethod == 2 OR deprMethod == 3)
isAnnual = 1;
life = "No Year"->"Useful Life (in Years)";
persInSection = "NumPeriods";
ELSE /* Periodic Methods */
isAnnual = 0;
life = "No Year"->"Useful Life (in Years)" * "NumPeriods";
persInSection = 1;
ENDIF

setCashImpact = 0;
setFundingImpact = 0;
yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
/* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
depMethod = 1;
deprConvention = 1;
ENDIF

ENDIF

ENDIF

IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
paramsSet = 0;
IF ("Asset Status" == 0)
"Asset Status" = 1;
ENDIF
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
/* clear out any previously calculated values */
"Depreciation" = #MISSING;
"Accumulated Depreciation" = #MISSING;
"Property, Plant and Equipment Gross" = #MISSING;
"Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
"Proceeds from Sale of Property, Plant and Equipment" = #MISSING;

IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
   "Cash Flow Allocator" = #MISSING;
ENDIF
IF (fundingIncidence <> -1 OR fundingChanged == 1)
   "Funding Allocator" = #MISSING;
ENDIF
"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Repairs" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Expenditure" = #MISSING;

IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
   "Capital Expenditure" = basicCost - retirementObs;
ENDIF

/* Check if we should start setting cash flow impact values */
IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
   setCashImpact = cashStaggeredPers;
   cashAllocPct = 1 / cashStaggeredPers;
ENDIF

/* Check if we should start setting funding values */
IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
   setFundingImpact = fundingStaggeredPers;
   fundingAllocPct = 1 / fundingStaggeredPers;
ENDIF

IF (setCashImpact > 0)
   "Cash Flow Allocator" = cashAllocPct;
   setCashImpact = setCashImpact - 1;
ENDIF
"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
   "Funding Allocator" = fundingAllocPct;
   setFundingImpact = setFundingImpact - 1;
ENDIF
"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/* Check if we should start depreciation */
IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1)
OR
(deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1 AND "TP-Date" > yearVal))

setStartDepr = 0;
periodOffset = 0;
persIn1stYear = "NumPeriods";
numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
totDeprPeriods = numDeprPeriods;

dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
delayStart = 0;
IF (deprConvention == 2) /* Prorate Actual Date */
  IF (dayVal == 1) /* Treat same as Begin Period */
    split1stAmt = 0;
  ELSE
    numDeprPeriods = numDeprPeriods + 1;
    split1stAmt = 1;
  ENDIF
ELSEIF (deprConvention == 3) /* Mid Period */
  numDeprPeriods = numDeprPeriods + 1;
  split1stAmt = 1;
ELSEIF (deprConvention == 4) /* MidYear */
  split1stAmt = 0;
  delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
ELSE /* Prorate Begin Period */
  split1stAmt = 0;
ENDIF

IF (deprMethod == 3 OR deprMethod == 4)
  /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
  deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
  numDeprPeriods = numDeprPeriods - preExistingPers;
  IF (numDeprPeriods > 0)
    deprAmt = (basicCost - salvageVal) / totDeprPeriods;
    periodicPriorAccumDepr = deprAmt * preExistingPers;
  ENDIF
ENDIF

ENDIF

/* Depreciation calcuation section */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
  lifeIndex = @INT(periodOffset / persInSection);
ENDIF
IF (deprMethod == 1) /* Straight Line */
  deprAmt = (basicCost - salvageVal) / totDeprPeriods;
ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
  IF (lifeIndex == 0)
    /* cost * rate * month / 12; for 1st period */
    deprAmt = basicCost * deprRate * persIn1stYear / 
               "NumPeriods" / persInSection;
  ELSE /* (cost - total depreciation from prior periods) * rate; for all 
        middle periods */
    deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
  ENDIF
ELSEIF (deprMethod == 2) /* Sum of Years Digits */
  deprRate = (life - lifeIndex) * 2 / (life * (life +1));
  deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
ENDIF

IF (split1stAmt == 1)
  IF (deprConvention == 2)
    /* assume that there are 30 days in each month */
    deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
  ELSEIF (deprConvention == 4)
  ELSE
    deprAmt = deprAmt - deprAmt / 2;
  ENDIF
  split1stAmt = 0;
ENDIF

/* Adjust for rounding errors */
IF (periodOffset + 1 == numDeprPeriods)
  deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
ENDIF

periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
  deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
  IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
    priorAccumDepr = periodicPriorAccumDepr;
  ENDIF
ELSE
  priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" == inServiceDate OR 
preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
  "Property, Plant and Equipment Gross" = basicCost;
  "Depreciation" = deprAmt;
  "Accumulated Depreciation" = periodicPriorAccumDepr;
  "Insurance" = insuranceCost;
  "Maintenance" = maintenanceCost;
/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
  IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
      "No Year"->"BegBalance"->"Reason Ended" == 3)
    netValue = basicCost - priorAccumDepr + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
      "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue -
      saleValue + deprAmt;
      ELSE
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue +
        deprAmt;
      ENDIF
    "Retirement Expenses" = retirementCosts - retirementObs;
    "Proceeds from Sale of Property, Plant and Equipment" = saleValue -
    retirementCosts;
  ENDIF
  periodOffset = periodOffset +1;
ENDIF
delayStart = delayStart -1;
ENDIF
ENDIF
ENDFIX

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" (  
  IF (cashflowChanged == 1 AND "Asset Status" > 0)
    "Cash Flow Incidence" = cashFlowIncidence;
  ENDIF
  IF (fundingChanged == 1 AND "Asset Status" > 0)
    "Funding Incidence" = fundingIncidence;
  ENDIF
)
ENDFIX
ENDFIX

/* Clear out any asset line items marked for delete */
FIX(@CHILDREN("Total New"), [Hidden_Version], [DepartmentTo], [AssetClass])
[Hidden_Scenario](  
  IF ("No Year"->"BegBalance"->"Asset Status" == -1)
    @CURRMBR("Line Item") = #MISSING;
  ENDIF
  );
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo])
  CALC DIM ("Period");
  @ANCESTORS([SrcLineItem]);
  @ANCESTORS([AssetClass]);
ENDFIX
TransferExistAsset

Description

Transfers an existing asset.

Formula

```
SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR deprMethod = 0;
VAR deprRate = 0;
VAR deprConvention = 0;
VAR numDeprPeriods = 0;
VAR totDeprPeriods = 0;
VAR persIn1stYear = 12;
VAR priorAccumDepr = 0;
VAR periodicPriorAccumDepr = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR deprAmt = 0;
VAR deprAmtLast = 0;
VAR persInSection = 0;
VAR life = 0;
VAR lifeIndex = 0;
VAR isAnnual = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR repairsCost = 0;
VAR retirementObs = 0;
VAR retirementCosts = 0;
VAR purchaseDate = 0;
VAR retireOption = 0;
VAR saleValue = 0;
VAR xferAsset = 0;
VAR paramsSet = 0;
VAR netValue;
VAR yearVal;
VAR monthVal;
VAR dayVal;
VAR delayStart;
VAR split1stAmt;
VAR preExistingPers = 0;
VAR setStartDepr = 0;

/* Transfer section: Locate either an asset with same Asset ID or a new one for transfer */
FIX(@CHILDREN("Base SPs"), [Hidden_Scenario], [Hidden_Version], [DepartmentTo],
[AssetClass], "No Year", "BegBalance")
"Asset Description" (}

166 Capital Asset Planning Structure
IF (NOT @ISMBR([DepartmentFrom]))

IF (@MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING) OR ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING))

IF ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)
/* Clear out any duplicate Assets based on Asset ID if already transferred */
IF (xferAsset == 1)
"Asset Status" = -1; /* mark line item for deletion */
ELSE
xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";
"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";
IF ("In Service Date" < "Purchase Date")
"In Service Date" = "Purchase Date";
ENDIF
"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ELSEIF (xferAsset == 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING)

Predefined Business Rules 167
xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Asset Units" * "Asset Rate" + ("Asset Units" * "Asset Rate" * "Taxes %") + "Freight" + "Additional Charges" + "Retirement Obligation" + "Installation";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";
"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF

"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [AssetClass], [SrcLineItem], "No Year", "BegBalance")
"Premature End Date" (IF (xferAsset == 1)
  "Asset Status" = 0; /* set status to new to trigger recalc below */
  "Premature End Date" = [TransferDate];
  "Reason Ended" = 1;
  "Sale Value" = #MISSING;
ENDIF
);
ENDFIX

/* Calculate depreciation after transfer */
FIX(@DESCENDANTS("Base SPs"), [AssetClass], [Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo])

FIX (@LEVMBRS("Period", 0), @LEVMBRS("Year", 0))
"Depreciation"
IF ("No Year"->"BegBalance"->"Asset Status" == 0)
IF (@ISMBR("BegBalance") AND paramsSet == 0)
paramsSet = 1;

/* Initialized all the variables required to calculate depreciation */
/* eliminate days from date */
delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

/* Initialize for depreciation calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

deprMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Method";
depConvention = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Depreciation Convention";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
repairsCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Repairs %";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
IF (monthVal < 0)
    yearVal = yearVal - 1;
ENDIF
monthVal = 1200 + monthVal;
ENDIF
preExistingPers = yearVal * 12 + monthVal / 100;
setStartDepr = 1;
ELSE
preExistingPers = 0;
setStartDepr = 0;
ENDIF

/* setup for depreciation */
umDeprPrPeriods = 0;
periodOffset = 0;

IF (deprMethod > 0)
    persIn1stYear = "NumPeriods" - "Cal TP-Index" + 1;
    priorAccumDepr = 0;
    periodicPriorAccumDepr = 0;
    deprAmt = 0;
    /* Add a periodic vs annual property instead of additional deprMethods */
    /* Annual Methods SumYearDigits =2 DecliningBalance Year = 3*/
    IF (deprMethod == 2 OR deprMethod == 3)
        isAnnual = 1;
        life = "No Year" - "Useful Life (in Years)";
        persInSection = "NumPeriods";
    ELSE /* Periodic Methods */
        isAnnual = 0;
        life = "No Year" - "Useful Life (in Years)" * "NumPeriods";
        persInSection = 1;
    ENDIF
ENDIF

yearVal = @INT(inServiceDate / 10000) * 10000;

/* DecliningBalance methods */
IF (deprMethod == 3 OR deprMethod == 4)
    /* depreciation rate = 1 - (salvage / cost) ^ (1 / life) */
    deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

/* Reset depr method to SLN convention to prorate 1st period for preExisting assets */
IF (preExistingPers > 0)
    deprMethod = 1;
    deprConvention = 1;
ENDIF
ENDIF
ENDIF

IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Depreciation" = #MISSING;
    "Accumulated Depreciation" = #MISSING;
    "Property, Plant and Equipment Gross" = #MISSING;
    "Loss/(Gain) on Sale of Property, Plant and Equipment" = #MISSING;
    "Proceeds from Sale of Property, Plant and Equipment" = #MISSING;
    "Cash Flow Allocator" = #MISSING;
    "Funding Allocator" = #MISSING;
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Repairs" = #MISSING;
"Retirement Expenses" = #MISSING;
"Capital Expenditure" = #MISSING;

/*
IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
  "Capital Expenditure" = basicCost - retirementObs;
ENDIF */

/* Check if we should start depreciation */
IF ((numDeprPeriods == 0) AND ("TP-Date" == inServiceDate) OR (setStartDepr == 1) OR
  (deprConvention == 4 AND "Fiscal TP-Index" == @INT("NumPeriods" / 2)+1) AND "TP-Date" > yearVal))
  setStartDepr = 0;
  periodOffset = 0;
  persIn1stYear = "NumPeriods" - "Cal TP-Index" + 1;
  numDeprPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
  totDeprPeriods = numDeprPeriods;
  dayVal = "No Year"->"BegBalance"->"In Service Date" - inServiceDate;
  monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
  delayStart = 0;
  IF (deprConvention == 2) /* Prorate Actual Date */
    IF (dayVal == 1) /* Treat same as Begin Period */
      split1stAmt = 0;
    ELSE
      numDeprPeriods = numDeprPeriods + 1;
      split1stAmt = 1;
    ENDIF
  ELSEIF (deprConvention == 3) /* Mid Period */
    numDeprPeriods = numDeprPeriods + 1;
    split1stAmt = 1;
  ELSEIF (deprConvention == 4) /* MidYear */
    split1stAmt = 0;
    delayStart = @INT("NumPeriods" / 2) - "Fiscal TP-Index" +1;
  ELSE /* Prorate Begin Period */
    split1stAmt = 0;
  ENDIF
IF (deprMethod == 3 OR deprMethod == 4)
  /* depreciation rate = 1 - ((salvage / cost) ^ (1 / life)) */
  deprRate = 1 - @POWER(salvageVal / basicCost, 1 / life);
ENDIF

insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
repairsCost = repairsCost / "NumPeriods";
/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numDeprPeriods = numDeprPeriods - preExistingPers;
    IF (numDeprPeriods > 0)
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
        periodicPriorAccumDepr = deprAmt * preExistingPers;
    ENDIF
ENDIF
ENDIF

/* Depreciation calculation section */
IF (deprMethod > 0 AND periodOffset < numDeprPeriods AND delayStart < 1)
    lifeIndex = @INT(periodOffset / persInSection);
    IF (deprMethod == 1) /* Straight Line */
        deprAmt = (basicCost - salvageVal) / totDeprPeriods;
    ELSEIF (deprMethod == 3 or deprMethod == 4) /* Declining Balance */
        IF (lifeIndex == 0)
            /* cost * rate * month / 12; for 1st period */
            deprAmt = basicCost * deprRate * persIn1stYear /
                    "NumPeriods" / persInSection;
        ELSE /* (cost - total depreciation from prior periods) * rate; for all middle periods */
            deprAmt = (basicCost - priorAccumDepr) * deprRate / persInSection;
        ENDIF
    ELSEIF (deprMethod == 2) /* Sum of Years Digits */
        deprRate = (life - lifeIndex) * 2 / (life * (life +1));
        deprAmt = (basicCost - salvageVal) * deprRate / persInSection;
    ENDIF
    IF (split1stAmt == 1)
        IF (deprConvention == 2)
            /* assume that there are 30 days in each month */
            deprAmt = deprAmt * @MAX(30 - dayVal, 1) / 30;
        ELSEIF (deprConvention == 4)  
            
        ELSE
            deprAmt = deprAmt - deprAmt / 2;
        ENDIF
        split1stAmt = 0;
    ENDIF
    /* Adjust for rounding errors */
    IF (periodOffset + 1 == numDeprPeriods)
        deprAmt = (basicCost - salvageVal) - periodicPriorAccumDepr;
    ENDIF
    periodicPriorAccumDepr = periodicPriorAccumDepr + deprAmt;
ENDIF

/* Use accumulated depr as first value for Mid-Year conversion */
IF (deprConvention == 4 AND "TP-Date" == inServiceDate AND monthVal > 600)
    deprAmt = periodicPriorAccumDepr;
ENDIF

IF (isAnnual == 1)
IF (@INT((periodOffset + 1) / persInSection) == (lifeIndex + 1))
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF
ELSE
    priorAccumDepr = periodicPriorAccumDepr;
ENDIF

/* Only assign to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
    "Property, Plant and Equipment Gross" = basicCost;
    "Depreciation" = deprAmt;
    "Accumulated Depreciation" = periodicPriorAccumDepr;
    "Insurance" = insuranceCost;
    "Maintenance" = maintenanceCost;
    "Repairs" = repairsCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
    IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)
        netValue = basicCost - priorAccumDepr + retirementCosts;
        IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue - saleValue + deprAmt;
        ELSE
            "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue + deprAmt;
        ENDIF
    ELSE
        "Loss/(Gain) on Sale of Property, Plant and Equipment" = netValue - retirementCosts;
    ENDIF
ENDIF
periodOffset = periodOffset +1;
ENDIF
delayStart = delayStart -1;
ENDIF
ENDFIX

/* Clear out any asset line items marked for delete */
FIX(@CHILDREN("Base Sps"), [Hidden_Version], [DepartmentTo], [AssetClass])
    IF ("No Year"->"BegBalance"->"Asset Status" == -1)
        @CURRMBR("Line Item") = #MISSING;
    ENDIF
); ENDIF

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo])
### TransferExistIntangible

**Description**

Transfers an existing intangible asset.

**Formula**

```plaintext
SET CREATENONMISSINGBLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retireOption = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR purchaseDate = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR paramsSet = 0;
VAR xferAsset = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;

/* Transfer section: Locate either an asset with same Asset ID or a new one for transfer */
FIX(@CHILDREN("Base SPs"), [Hidden_Scenario], [Hidden_Version], [DepartmentTo], [AssetClassInt], "No Year", "BegBalance")
"Asset Description" ( Capital Asset Planning Structure

174  Capital Asset Planning Structure
IF (NOT @ISMBR([DepartmentFrom]))

IF (@MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING) OR ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)

IF ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)

/* Clear out any duplicate Assets based on Asset ID if already transferred */
IF (xferAsset == 1)
"Asset Status" = -1; /* mark line item for deletion */
ELSE

xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";

"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

IF ("In Service Date" < "Purchase Date")
"In Service Date" = "Purchase Date";
ENDIF

"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ELSEIF (xferAsset == 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING)

xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";

"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

IF ("In Service Date" < "Purchase Date")
  "In Service Date" = "Purchase Date";
ENDIF

"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [AssetClassInt], [SrcLineItem], "No Year", "BegBalance")
"Premature End Date"(
  IF (xferAsset = 1)
    "Asset Status" = 0; /* set status to new to trigger recalc below */
    "Premature End Date" = [TransferDate];
    "Reason Ended" = 1;
    "Sale Value" = #MISSING;
  ENDIF)
);
ENDFIX

FIX(DESCENDANTS("Base SPs"), [Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo], [AssetClassInt])

FIX (LEVMBR("Period", 0) LEVMBR("Year", 0))
"Amortization"

/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)

IF (@ISMBR("BegBalance") AND paramsSet == 0)
   /* Initialized all the variables required to calculate depreciation */
   paramsSet = 1;

   /* eliminate days from date */
   delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
   prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
   purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
   inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;

   /* Initialize for amortization calc in next section */
   basicCost = "No Year"->"Basic Cost";
   salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
   capitalizePct = "No Year"->"Partial Capitalize %";
   saleValue = "No Year"->"Sale Value";
   retirementCosts = "No Year"->"Retirement Costs";
   retirementObs = "No Year"->"Retirement Obligation";

   amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->
   "Amortization Method";

   insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
   "Global"->"Insurance %";
   maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->
   "Global"->"Maintenance %";

   IF ("No Year"->"Impairment Date1" <> #MISSING)
      impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
   ELSE
      impairmentDate1 = #MISSING;
   ENDIF

   IF ("No Year"->"Impairment Date2" <> #MISSING)
      impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
   ELSE
      impairmentDate2 = #MISSING;
   ENDIF

   impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
   impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

   /* check if we have a pre-existing asset */
   IF (inServiceDate < "First Date")
      yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
      monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate
      - @INT(inServiceDate / 10000) * 10000);
   IF (monthVal < 0)
      IF (yearVal > 0)
         yearVal = yearVal - 1;
   ENDIF

Predefined Business Rules  177
monthVal = 1200 + monthVal;
ENDIF
preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
setStartAmort = 1;
ELSE
preExistingPers = 0;
setStartAmort = 0;
ENDIF

numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF
ENDIF

IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF

IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Amortization" = #MISSING;
    "Accumulated Amortization" = #MISSING;
    "Intangible Assets Finite, Gross" = #MISSING;
    "Loss/(Gain) on Sale of Intangibles" = #MISSING;
    "Proceeds from Sale of Intangibles" = #MISSING;
    "Cash Flow Allocator" = #MISSING;
    "Funding Allocator" = #MISSING;
    "Cash Outflow from Capital Additions" = #MISSING;
    "Cash Inflow from Funding" = #MISSING;
    "Long Term Debt" = #MISSING;
    "Insurance" = #MISSING;
    "Maintenance" = #MISSING;
    "Impairment of Assets" = #MISSING;
    "Capital Reserve" = #MISSING;
    "Capital Expenditure" = #MISSING;
    "Retirement Expenses" = #MISSING;

    /*
     * Check if we should start amortization */

     IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
         "Capital Expenditure" = basicCost - retirementObs;
     ENDIF
*/
IF (("TP-Date" == inServiceDate) OR (setStartAmort == 1))
setStartAmort = 0;
periodOffset = 0;
numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in Years)";
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
intangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numAmortPeriods = numAmortPeriods - preExistingPers;
    IF (numAmortPeriods > 0)
        priorAccumAmort = amortAmt * preExistingPers;
    ENDIF
ENDIF

numRemainingPers = numAmortPeriods;
ENDIF

;/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)

    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate1)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
        ENDIF
        "Impairment of Assets" = intangibleNet - impairmentFairVal1 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal1);
    amortAmt = impairmentFairVal1 / numRemainingPers;
    ELSEIF ("TP-Date" == impairmentDate2)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
        ENDIF
        "Impairment of Assets" = intangibleNet - impairmentFairVal2 - "Capital Reserve";
    ENDIF
    basicCost = basicCost - (intangibleNet - impairmentFairVal2);
    amortAmt = impairmentFairVal2 / numRemainingPers;
ELSEIF ("TP-Date" == impairmentDate2)

    IF (periodOffset + 1 == numAmortPeriods)
        amortAmt = (basicCost - salvageVal) - priorAccumAmort;
    ENDIF
priorAccumAmort = priorAccumAmort + amortAmt;
intangibleNet = basicCost - priorAccumAmort;

/* Only assign values to member if we are in range */
IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))
  "Intangible Assets Finite, Gross" = basicCost;
  "Amortization" = amortAmt;
  "Accumulated Amortization" = priorAccumAmort;
  "Insurance" = insuranceCost;
  "Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
  IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR "No Year"->"BegBalance"->"Reason Ended" == 3)
    netValue = basicCost - priorAccumAmort + retirementCosts;
    IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
      "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue + amortAmt;
    ELSE
      "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
    ENDIF
  ELSE
    "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
  ENDIF
  "Retirement Expenses" = retirementCosts - retirementObs;
  "Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF ENDIF ENDIF

ENDFIX

/* Clear out any asset line items marked for delete */
FIX(@CHILDREN("Base Sps"), [Hidden_Version], [DepartmentTo], [AssetClassInt])
[Hidden_Scenario](
  IF ("No Year"->"BegBalance"->"Asset Status" == -1)
    @CURRMBR("Line Item") = #MISSING;
  ENDIF
);
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo])
CALC DIM ("Period");
@ANCESTORS([SrcLineItem]);
@ANCESTORS([AssetClassInt]);
Transfer Intangible

Description

Transfers an intangible asset.

Formula

```
SET CREATE NONMISSING BLK ON;
SET UPDATECALC OFF;
SET AGGMISSG ON;

VAR periodOffset = 0;
VAR amortMethod = 0;
VAR numAmortPeriods = 0;
VAR priorAccumAmort = 0;
VAR basicCost = 0;
VAR salvageVal = 0;
VAR amortAmt = 0;
VAR retirementCosts = 0;
VAR retirementObs = 0;
VAR impairmentDate1 = 0;
VAR impairmentDate2 = 0;
VAR impairmentFairVal1 = 0;
VAR impairmentFairVal2 = 0;
VAR inServiceDate = 0;
VAR delayedStartDate = 0;
VAR prematureEndDate = 0;
VAR maintenanceCost = 0;
VAR insuranceCost = 0;
VAR cashOutflowDate = 0;
VAR fundingDate = 0;
VAR purchaseDate = 0;
VAR cashFlowIncidence = 0;
VAR fundingIncidence = 0;
VAR cashStaggeredPers = 1;
VAR fundingStaggeredPers = 1;
VAR cashAllocPct = 1;
VAR fundingAllocPct = 1;
VAR fundingAmt = 0;
VAR setCashImpact = 0;
VAR setFundingImpact = 0;
VAR capitalizePct = 0;
VAR impairmentOpt = 1;
VAR numRemainingPers;
VAR intangibleNet;
VAR netValue;
VAR saleValue;
VAR yearVal;
VAR monthVal;
VAR xferAsset = 0;
VAR paramsSet = 0;
VAR cashflowChanged = 0;
VAR fundingChanged = 0;
VAR preExistingPers = 0;
VAR setStartAmort = 0;
```
/* Transfer section: Locate either an asset with same Asset ID or a new one for transfer */
FIX(@CHILDREN("Total New"), [Hidden_Scenario], [Hidden_Version], [DepartmentTo], [AssetClassInt], "No Year", "BegBalance")
"Asset Description" {

IF (NOT @ISMBR([DepartmentFrom]))

IF (@MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING) OR ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)

IF ("Asset ID" == [DepartmentFrom]->[SrcLineItem]->"Asset ID" AND [DepartmentFrom]->[SrcLineItem]->"Asset ID" <> #MISSING)
/* Clear out any duplicate Assets based on Asset ID if already transferred */
IF (xferAsset == 1)
"Asset Status" = -1; /* mark line item for deletion */
ELSE
xferAsset = 1;
"Asset Status" = 0;
"Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
"Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
"CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
"Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
"Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
"UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
"Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

"Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
"Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
"Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
"Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
"Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
"Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
"Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
"Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
"Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
"Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
"Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
"Basic Cost" = "Acquisition Costs" + "Additional Charges";
"Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
"Salvage" = "Salvage Input" * "Asset Units";

"Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
"Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";
"Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
"In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

IF ("In Service Date" < "Purchase Date")
"In Service Date" = "Purchase Date";
ENDIF
"Delayed Start Date" = [TransferDate];
"Reason Delayed" = 1;
ENDIF

ELSEIF (xferAsset == 0 AND @MAXS(SKIPMISSING, @CHILDREN("Asset Properties")) == #MISSING)

    xferAsset = 1;
    "Asset Status" = 0;
    "Asset ID" = [DepartmentFrom]->[SrcLineItem]->"Asset ID";
    "Asset Description" = [DepartmentFrom]->[SrcLineItem]->"Asset Description";
    "CAR No." = [DepartmentFrom]->[SrcLineItem]->"CAR No.";
    "Justification" = [DepartmentFrom]->[SrcLineItem]->"Justification";
    "Physical Location" = [DepartmentFrom]->[SrcLineItem]->"Physical Location";
    "UOM" = [DepartmentFrom]->[SrcLineItem]->"UOM";
    "Priority" = [DepartmentFrom]->[SrcLineItem]->"Priority";

    "Asset Units" = [DepartmentFrom]->[SrcLineItem]->"Asset Units";
    "Asset Rate" = [DepartmentFrom]->[SrcLineItem]->"Asset Rate";
    "Installation" = [DepartmentFrom]->[SrcLineItem]->"Installation";
    "Freight" = [DepartmentFrom]->[SrcLineItem]->"Freight";
    "Salvage Input" = [DepartmentFrom]->[SrcLineItem]->"Salvage Input";
    "Taxes %" = [DepartmentFrom]->[SrcLineItem]->"Taxes %";
    "Acquisition Costs" = [DepartmentFrom]->[SrcLineItem]->"Acquisition Costs";
    "Additional Charges" = [DepartmentFrom]->[SrcLineItem]->"Additional Charges";
    "Retirement Obligation" = [DepartmentFrom]->[SrcLineItem]->"Retirement Obligation";
    "Retirement Options" = [DepartmentFrom]->[SrcLineItem]->"Retirement Options";
    "Partial Capitalize %" = [DepartmentFrom]->[SrcLineItem]->"Partial Capitalize %";
    "Basic Cost" = "Acquisition Costs" + "Additional Charges";
    "Useful Life (in Years)" = "No Scenario"->"No Version"->"No Entity"->"Global"->"Useful Life (in Years)";
    "Salvage" = "Salvage Input" * "Asset Units";

    "Cash Flow Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
    "Funding Incidence" = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

    "Purchase Date" = [DepartmentFrom]->[SrcLineItem]->"Purchase Date";
    "In Service Date" = [DepartmentFrom]->[SrcLineItem]->"In Service Date";

    IF ("In Service Date" < "Purchase Date")
        "In Service Date" = "Purchase Date";
    ENDIF

    "Delayed Start Date" = [TransferDate];
    "Reason Delayed" = 1;
ENDIF
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [AssetClassInt],
[SrcLineItem, "No Year", "BegBalance")
"Premature End Date"(
    IF (xferAsset == 1)
        "Asset Status" = 0;    /* set status to new to trigger recalc below */
        "Premature End Date" = [TransferDate];
        "Reason Ended" = 1;
        "Sale Value" = #MISSING;
    ENDIF
);  
ENDFIX

/* Calculate amortization after transfer */
FIX(@DESCENDANTS("Total New"); [Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo], [AssetClassInt])
FIX (@LEVMBRS("Period", 0) @LEVMBRS("Year", 0))
"Amortization"(

/* Only calc valid asset line items */
IF ("No Year"->"BegBalance"->"Asset Status" >= 0)
    IF (@ISMBR("BegBalance") AND paramsSet == 0)
        /* Initialized all the variables required to calculate depreciation */
        paramsSet = 1;

        /* eliminate days from date */
        delayedStartDate = @INT("No Year"->"Delayed Start Date" / 100) * 100;
        prematureEndDate = @INT("No Year"->"Premature End Date" / 100) * 100;
        purchaseDate = @INT("No Year"->"Purchase Date" / 100) * 100;
        inServiceDate = @INT("No Year"->"In Service Date" / 100) * 100;
        cashOutflowDate = purchaseDate;
        fundingDate = purchaseDate;
        cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";

        IF (cashFlowIncidence <> "No Year"->"Cash Flow Incidence")
            cashflowChanged = 1;
        ELSE
            cashflowChanged = 0;
        ENDIF

        IF (cashFlowIncidence == -1 AND cashflowChanged == 1)
            cashStaggeredPers = 4;
        ELSE
            /* Extract year and month from date to adjust based in incidence values in global assumptions */
            yearVal = @INT(cashOutflowDate / 10000) * 10000;
            monthVal = cashOutflowDate - yearVal;
            monthVal = monthVal + cashFlowIncidence;

            IF (monthVal <= 0)
                yearVal = yearVal - 10000;
                monthVal = 1200 + monthVal;
            ELSEIF (monthVal > 1200)
                yearVal = yearVal + 10000;

            cashOutflowDate = yearVal * 10000 + monthVal;
            inServiceDate = cashOutflowDate + purchaseDate;
            cashFlowIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Cash Flow Incidence";
        ENDIF
    ENDIF
ENDFIX
monthVal = monthVal - 1200;
ENDIF

cashOutflowDate = yearVal + monthVal;
cashStaggeredPers = 1;
ENDIF

fundingIncidence = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding Incidence";

IF (fundingIncidence <> "No Year"->"Funding Incidence")
   fundingChanged = 1;
ELSE
   fundingChanged = 0;
ENDIF

IF (fundingIncidence == -1 AND fundingChanged == 1)
   fundingStaggeredPers = 4;
ELSE
   yearVal = @INT(fundingDate / 10000) * 10000;
   monthVal = fundingDate - yearVal;
   monthVal = monthVal + fundingIncidence;
   IF (monthVal <= 0)
      yearVal = yearVal - 10000;
      monthVal = 1200 + monthVal;
   ELSEIF (monthVal > 1200)
      yearVal = yearVal + 10000;
      monthVal = monthVal - 1200;
   ENDIF
   fundingDate = yearVal + monthVal;
   fundingStaggeredPers = 1;
ENDIF

/* Initialize for amortization calc in next section */
basicCost = "No Year"->"Basic Cost";
salvageVal = "No Year"->"Salvage Input" * "No Year"->"Asset Units";
impairmentOpt = "No Year"->"Impairment Option";
capitalizePct = "No Year"->"Partial Capitalize %";
saleValue = "No Year"->"Sale Value";
retirementCosts = "No Year"->"Retirement Costs";
retirementObs = "No Year"->"Retirement Obligation";

amortMethod = "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Amortization Method";
insuranceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Insurance %";
maintenanceCost = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Maintenance %";
fundingAmt = basicCost * "No Year"->"No Scenario"->"No Version"->"No Entity"->"Global"->"Funding %";

IF ("No Year"->"Impairment Date1" <> #MISSING)
   impairmentDate1 = @INT("No Year"->"Impairment Date1" / 100) * 100;
Predefined Business Rules
ELSE
    impairmentDate1 = #MISSING;
ENDIF
IF ("No Year"->"Impairment Date2" <> #MISSING)
    impairmentDate2 = @INT("No Year"->"Impairment Date2" / 100) * 100;
ELSE
    impairmentDate2 = #MISSING;
ENDIF

impairmentFairVal1 = "No Year"->"Impairment Fair Value1";
impairmentFairVal2 = "No Year"->"Impairment Fair Value2";

/* check if we have a pre-existing asset */
IF (inServiceDate < "First Date")
    yearVal = (@INT("First Date" / 10000) * 10000 - @INT(inServiceDate / 10000) * 10000) / 10000;
    monthVal = ("First Date" - @INT("First Date" / 10000) * 10000) - (inServiceDate - @INT(inServiceDate / 10000) * 10000);
    IF (monthVal < 0)
        IF (yearVal > 0)
            yearVal = yearVal - 1;
        ENDIF
        monthVal = 1200 + monthVal;
    ENDIF
    preExistingPers = @ROUND(yearVal * 12 + monthVal / 100, 0);
    setStartAmort = 1;
ELSE
    preExistingPers = 0;
    setStartAmort = 0;
ENDIF

numAmortPeriods = 0;
periodOffset = 0;

IF (amortMethod == 1)
    priorAccumAmort = 0;
    amortAmt = 0;
    setCashImpact = 0;
    setFundingImpact = 0;
    yearVal = @INT(inServiceDate / 10000) * 10000;
    monthVal = (inServiceDate - @INT(inServiceDate / 10000) * 10000);
ENDIF
ENDIF
IF (@ISMBR("No Year") AND @ISMBR("BegBalance"))
    paramsSet = 0;
    IF ("Asset Status" == 0)
        "Asset Status" = 1;
    ENDIF
ENDIF
IF (NOT @ISMBR("BegBalance") AND paramsSet == 1)
    /* clear out any previously calculated values */
    "Amortization" = #MISSING;
    "Accumulated Amortization" = #MISSING;
"Intangible Assets Finite, Gross" = #MISSING;
"Loss/(Gain) on Sale of Intangibles" = #MISSING;
"Proceeds from Sale of Intangibles" = #MISSING;

IF (cashFlowIncidence <> -1 OR cashflowChanged == 1)
   "Cash Flow Allocator" = #MISSING;
ENDIF

IF (fundingIncidence <> -1 OR fundingChanged == 1)
   "Funding Allocator" = #MISSING;
ENDIF

"Cash Outflow from Capital Additions" = #MISSING;
"Cash Inflow from Funding" = #MISSING;
"Long Term Debt" = #MISSING;
"Insurance" = #MISSING;
"Maintenance" = #MISSING;
"Impairment of Assets" = #MISSING;
"Capital Reserve" = #MISSING;
"Capital Expenditure" = #MISSING;
"Retirement Expenses" = #MISSING;

IF ("TP-Date" == purchaseDate AND (delayedStartDate == #MISSING OR (delayedStartDate
> #MISSING AND "TP-Date" > delayedStartDate)))
   "Capital Expenditure" = basicCost - retirementObs;
ENDIF

IF ("TP-Date" == cashOutflowDate AND (delayedStartDate == #MISSING OR
(delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
   setCashImpact = cashStaggeredPers;
   cashAllocPct = 1 / cashStaggeredPers;
ENDIF

IF ("TP-Date" == fundingDate AND (delayedStartDate == #MISSING OR
(delayedStartDate > #MISSING AND "TP-Date" > delayedStartDate)))
   setFundingImpact = fundingStaggeredPers;
   funding AllocPct = 1 / fundingStaggeredPers;
ENDIF

IF (setCashImpact > 0)
   "Cash Flow Allocator" = cashAllocPct;
   setCashImpact = setCashImpact - 1;
ENDIF

"Cash Outflow from Capital Additions" = basicCost * "Cash Flow Allocator";

IF (setFundingImpact > 0)
   "Funding Allocator" = fundingAllocPct;
   setFundingImpact = setFundingImpact - 1;
ENDIF

"Cash Inflow from Funding" = fundingAmt * "Funding Allocator";
"Long Term Debt" = fundingAmt * "Funding Allocator";

/* Check if we should start amortization */
IF ("TP-Date" == inServiceDate OR (setStartAmort == 1))
   setStartAmort = 0;
   periodOffset = 0;
   numAmortPeriods = "NumPeriods" * "No Year"->"BegBalance"->"Useful Life (in
Years)";

Predefined Business Rules  187
insuranceCost = insuranceCost / "NumPeriods";
maintenanceCost = maintenanceCost / "NumPeriods";
intangibleNet = basicCost;
amortAmt = (basicCost - salvageVal) / numAmortPeriods;

/* Adjust for pre existing assets */
IF (preExistingPers > 0)
    numAmortPeriods = numAmortPeriods - preExistingPers;
    IF (numAmortPeriods > 0)
        priorAccumAmort = amortAmt * preExistingPers;
    ENDIF
ENDIF
numRemainingPers = numAmortPeriods;
ENDIF

/* Amortization calculation section */
IF (amortMethod == 1 AND periodOffset < numAmortPeriods)

    /* If capitalized book in Capital Reserve instead of Impairment */
    IF ("TP-Date" == impairmentDate1)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal1;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal1) * capitalizePct;
        ENDIF
        basicCost = basicCost - (intangibleNet - impairmentFairVal1);
        amortAmt = impairmentFairVal1 / numRemainingPers;
    ELSEIF ("TP-Date" == impairmentDate2)
        IF (impairmentOpt == 1)
            "Impairment of Assets" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 2)
            "Capital Reserve" = intangibleNet - impairmentFairVal2;
        ELSEIF (impairmentOpt == 3)
            "Capital Reserve" = (intangibleNet - impairmentFairVal2) * capitalizePct;
        ENDIF
        basicCost = basicCost - (intangibleNet - impairmentFairVal2);
        amortAmt = impairmentFairVal2 / numRemainingPers;
    ENDIF

    IF (periodOffset + 1 == numAmortPeriods)
        amortAmt = (basicCost - salvageVal) - priorAccumAmort;
    ENDIF

    priorAccumAmort = priorAccumAmort + amortAmt;
    intangibleNet = basicCost - priorAccumAmort;

    /* Only assign values to member if we are in range */
    IF ("TP-Date" >= delayedStartDate AND ("TP-Date" >= inServiceDate OR...
preExistingPers > 0) AND (prematureEndDate == #MISSING OR "TP-Date" < prematureEndDate))

"Intangible Assets Finite, Gross" = basicCost;
"Amortization" = amortAmt;
"Accumulated Amortization" = priorAccumAmort;
"Insurance" = insuranceCost;
"Maintenance" = maintenanceCost;
ENDIF

/* Set the loss/gain if we had a sale or writeoff */
IF ("TP-Date" == prematureEndDate)
IF ("No Year"->"BegBalance"->"Reason Ended" == 2 OR
 "No Year"->"BegBalance"->"Reason Ended" == 3)
netValue = basicCost - priorAccumAmort + retirementCosts;
IF ("No Year"->"BegBalance"->"Reason Ended" == 3)
 "Loss/(Gain) on Sale of Intangibles" = netValue - saleValue + amortAmt;
ELSE
 "Loss/(Gain) on Sale of Intangibles" = netValue + amortAmt;
ENDIF
 "Retirement Expenses" = retirementCosts - retirementObs;
 "Proceeds from Sale of Intangibles" = saleValue - retirementCosts;
ENDIF
ENDIF

periodOffset = periodOffset +1;
numRemainingPers = numRemainingPers -1;
ENDIF
ENDIF
ENDIF 
ENDFIX

FIX ("BegBalance", "No Year")
"Cash Flow Incidence" ( 
 IF (cashflowChanged == 1 AND "Asset Status" > 0)
 "Cash Flow Incidence" = cashFlowIncidence;
ENDIF
 IF (fundingChanged == 1 AND "Asset Status" > 0)
 "Funding Incidence" = fundingIncidence;
ENDIF
)
ENDFIX

ENDFIX

/* Clear out any asset line items marked for delete */
FIX(@CHILDREN("Total New"), [Hidden_Version], [DepartmentTo], [AssetClassInt])
[Hidden_Scenario] ( 
 IF ("No Year"->"BegBalance"->"Asset Status" == -1)
 @CURRMBR("Line Item") = #MISSING;
ENDIF
);
ENDFIX

FIX([Hidden_Scenario], [Hidden_Version], [DepartmentFrom], [DepartmentTo])
CALC DIM (*Period*);
ANCESTORS([SrcLineItem]);
ANCESTORS([AssetClassInt]);
ENDFIX
Index

Numbers
01. Global Capital Assumptions data form, 18, 47
02. New Asset Requests data form, 18, 51
02A. New Asset Requests - Addition data form, 33, 47
02B. Total Expense Impact data form, 48
02C. Cash Flow Impact - Line Item Details data form, 48
03. Manage Existing Specified data form, 19, 20, 21, 48
03A. Existing Specified Drill Down data form, 48
03B. Existing Specified Expenses data form, 48
04. Manage Existing Unspecified data form, 22, 48
05. New Intangible Asset Requests data form, 22, 51
05A. New Intangible Requests - Additions data form, 48
06. Manage Existing Intangibles Specified data form, 23, 49
06A. Existing Intangible Specified Drill Down data form, 49
06B. Existing Specified Expenses - Intangibles data form, 49
07. Manage Existing Intangibles Unspecified data form, 49
08. Capital Expenditure Summary data form, 49
08A. Capital Expenditure Summary - Line Item details data form, 49
09. Intangible Expenditure Summary data form, 25, 49
09A. Intangible Account Balance data form, 50
09B. Intangible Asset Balances data form, 50
10. Profit and Loss Impact data form, 26, 50
11. Balance Sheet Impact data form, 26, 50
12. Cash Flow Impact data form, 27, 50
13. Depreciation Summary data form, 27, 50
13A. Depreciation Summary - Line Item details data form, 50
14. Amortization Summary data form, 27, 51
14A. Amortization Summary - Line Item details data form, 51
15. Depreciation What ifs data form, 28, 51
15A. Global Capital Assumptions - Depreciation only data form, 51

A
about Capital Asset Planning, 9
access permissions
setting up, 28
transfers, 29
accessing
Performance Management Architect, 15
Planning, 15
Workspace, 15
accounts
adding, 36
fixed and varying, loading data into, 14
predefined, 51
adding
accounts, 36
asset classes, 31
assets, 18
custom fields, 33
data forms, 34
eexisting assets, 19
intangible assets, 23, 31
options for retiring new assets, 35
Project dimension, 41
tangible assets, 31
allocated cash flow percentages, calculating, 29
allocations, staggered cash flow, 34
amortization
summary, 27
viewing, 23, 27
Amortization Summary - Line Item details data form, 51
Amortization Summary data form, 27, 51
Application Library, working with, 32
applications
    creating Classic applications, 12
    creating Performance Management Architect applications, 11
    creating, overview, 10
    localizing, 44
    selecting, 15
    working with, 17
Asset CAR #, entering, 18
asset classes
    adding, 31
    predefined, 55
asset requests, 18
assets
    adding, 18, 34
    adding existing, 19
    adding intangible, 23
    adding retirement options, 35
    balance sheet impact, 26
    calculated details, 20
    cash flow impact, 26
    classes, adding, 31
    considerations for working with, 29
    depreciation, 27, 28
    entering CAR #, 18
    impairing intangible, 24
    improving, 21
    in-service date, 18
    managing existing, 19
    managing intangible, 22
    managing tangible, 18
    managing unspecified, 21
    modifying CAR #, 37
    prefix for, 20
    profit and loss impact, 26
    removing, 21
    requesting intangible, 22
    retiring, 20
    showing hidden fields, 38
    transferring, 20
    working with, 17
assumptions, enabling for entities, 39

B
Balance Sheet Impact data form, 26
Base Line Item page view, 21
Base SP, described, 20
business model, 10
business rules
    customizing, 45
    loading, 11
    predefined, 70
    updating for multiple entities, 39
C
calculated details, viewing, 20
calendar
    considerations for, 29
    described, 9
Capital Acquisition Request #, modifying name, 37
Capital Expenditure Summary data form, 49
capital reserve, 24
capitalize%, 24
CAR #
    entering, 18
    modifying, 37
cash flow
    staggered allocations, changing, 34
    viewing, 26
Cash Flow Impact - Line Item Details data form, 48
Cash Flow Impact data form, 27
Classic applications
    creating and initializing, 12
    loading information into, 13
Create Asset business rule, using, 45
creating
    Classic applications, 12
    Performance Management Architect applications, 11
custom fields, adding, 33
customizing Capital Asset Planning, 31

D
data
    accessing, 28
    loading using DLM Adapter for Planning, 14
    loading, overview, 13
data forms
    01. Global Capital Assumptions, 18, 47
    02. New Asset Requests, 18, 51
    02A. New Asset Requests - Addition, 33, 47
02B. Total Expense Impact, 48
02C. Cash Flow Impact - Line Item Details, 48
03. Manage Existing Specified, 19, 20, 21, 48
03A. Existing Specified Drill Down, 48
03B. Existing Specified Expenses, 48
04. Manage Existing Unspecified, 22, 48
05. New Intangible Asset Requests, 22, 51
05A. New Intangible Requests - Additions, 48
06. Manage Existing Intangibles Specified, 23, 24, 49
06A. Existing Intangible Specified Drill Down, 49
06B. Existing Specified Expenses - Intangibles, 49
07. Manage Existing Intangibles Unspecified, 49
08. Capital Expenditure Summary, 49
08A. Capital Expenditure Summary - Line Item details, 49
09. Intangible Expenditure Summary, 25, 49
09A. Intangible Account Balance, 50
09B. Intangible Asset Balances, 50
10. Profit and Loss Impact, 26, 50
11. Balance Sheet Impact, 26, 50
12. Cash Flow Impact, 27, 50
13. Depreciation Summary, 27, 50
13A. Depreciation Summary - Line Item details, 50
14. Amortization Summary, 27, 51
14A. Amortization Summary - Line Item details, 51
15. Depreciation What ifs, 28, 51
15A. Global Capital Assumptions - Depreciation only, 51
access permissions for, 28
adding, 34
and managing, 15
considerations when changing, 44
loading predefined, 11
predefined, 47
predefined menus in, 70
updating for projects, 42
updating Point of View dimensions, 39
using business rules, 45
Data Type, changing, 32
departments, enabling assumptions for, 39
deploying Performance Management Architect applications, 12
depreciation analyzing, 28
calculations, 29
changing the method, 28
considerations for, 28
summary, 27
viewing, 18, 22
Depreciation Summary data form, 27
Depreciation What ifs data form, 28, 51
dimensions
adding projects, 41
loading predefined Capital Asset Planning dimensions, 11
E
etities
enabling assumptions for, 39
updating business rules for, 39
Existing Intangible Specified Drill Down data form, 49
Existing Specified Drill Down data form, 49
Existing Specified Expenses - Intangibles data form, 49
Existing Specified Expenses data form, 48
expenditures
for intangibles, 25
summaries, 25
F
field labels, modifying, 37
fields
adding, 33
showing hidden, 38
financial impact review, 25
fixed accounts, loading data into, 14
formula expressions, customizing, 45
freight, 38
G
global assumptions
described, 17
establishing, 17
managing, 29
Global Capital Assumptions - Depreciation only data form, 51
Global Capital Assumptions data form, 18, 47

Index 193
H
hidden fields, showing, 38
HspCustomMsgs template file, 44

I
IM prefix, described, 21
impairing intangible assets, 24
impairment value, 24
improving assets, 21
in-service date, defining, 18
initializing Capital Asset Planning applications, 10
with Classic application administration, 12
with Performance Management Architect
application administration, 11
intangible assets
adding, 23
impairing, 24
managing, 22
requesting, 22
working with, 17
Intangible Expenditure Summary data form, 25
intangible expenditures, viewing, 25

L
labels
modifying, 37
predefined, 44
line items
predefined, 57
prefix for, 21
loading data
overview, 13
using Application Link Adapter for Hyperion Planning, 13
using Data Integration Management Adapter for Planning, 13
localizing applications, 44
logging on
to Capital Asset Planning, 15
to Workspace, Performance Management Architect, and Planning, 15

M
Manage Existing Intangibles Specified data form, 23, 49
Manage Existing Intangibles Unspecified data form, 49
Manage Existing Specified data form, 19, 20, 21, 48
Manage Existing Unspecified data form, 22, 48
managing
access permissions, 28
existing specified assets, 19
intangible assets, 22
tangible assets, 18
unspecified assets, 21
member formulas
customizing, 45
loading predefined, 11
predefined, 61
members, loading predefined, 11
menus
customizing, 44
in data forms, 70
loading predefined, 11
predefined, 67
metadata
loading using Data Integration Management, 14
loading, overview, 13
method of depreciation, changing, 28
modifying Capital Asset Planning, 31
multicurrency applications, 29

N
New Asset Requests - Addition data form, 33, 47
New Asset Requests data form, 18, 51
new assets, adding, 18
New Intangible Asset Requests data form, 22, 51
New Intangible Requests - Additions data form, 48
No Entity, removing, 40

P
Performance Management Architect
accessing, 15
creating an application, 11
POV dimensions, updating, 39
predefined accounts, 51
predefined business rules, 70
predefined data forms, 47
predefined line items, 57
predefined member formulas, 61
predefined menus, 67
predefined Smart List entries, 65
predefined Smart Lists, 65
prerequisites, 10
profit and loss impact
  reports, 26
  specifying, 23
Profit and Loss Impact data form, 26
Project dimension, adding, 41
prompt string, modifying, 37
purchases, 18

R
removing assets, 20, 21
Retirement Obligation field, 38
retirement options, adding, 35
retiring assets, 20
runtime prompts, adding, 33

S
salvage value
  considerations for, 28
  entering, 21
showing hidden fields, 38
Smart Lists
  customizing, 44
  loading predefined, 11
  predefined, 65
  predefined entries, 65
spreading data with custom time periods, 9

T
tangible assets
  balance sheets, 26
  cash flow impact, 26
  improving, 21
  managing, 18
  profit and loss impact, 26
  removing, 21
  retiring, 20
  specifying existing, 19
  transferring, 20
  working with, 17
taxes, 38
time periods
  custom, 9
  described, 9
timing adjustments, considerations for, 29
Total Expense Impact data form, 48
transferring
  considerations for, 29
  described, 20
  impact of, 20
  with multiple entities, 39

U
unspecified assets, managing, 21
URL for Workspace, 15
useful life, changing, 28
users and groups, setting up, 28

V
varying accounts, loading data into, 14
viewing
  amortization summaries, 27
  balance sheet impact, 26
  capital expenditure summary, 25
  cash flow impact, 26
  depreciation, 18, 27
  financial impact, 25
  improvements, 21
  intangible expense summary, 25
  profit and Loss impact, 26

W
Workspace, logging on, 15
write-offs, 20

Y
years, supporting multiple, 9