Oracle® Hyperion Financial Reporting Studio

User's Guide
Release 11.1.2.4.700
Updated: February 2016
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentation Accessibility</td>
<td>13</td>
</tr>
<tr>
<td>Documentation Feedback</td>
<td>15</td>
</tr>
<tr>
<td><strong>Chapter 1. Getting Started with Reporting Studio</strong></td>
<td></td>
</tr>
<tr>
<td>About Financial Reporting</td>
<td>17</td>
</tr>
<tr>
<td>Using Reporting Studio to Design Reports</td>
<td>17</td>
</tr>
<tr>
<td>Using EPM Workspace to Work With Books and Batches</td>
<td>19</td>
</tr>
<tr>
<td>Basic Reporting Studio Tasks</td>
<td>20</td>
</tr>
<tr>
<td>Installing Reporting Studio</td>
<td>20</td>
</tr>
<tr>
<td>Starting Reporting Studio</td>
<td>20</td>
</tr>
<tr>
<td>Opening Reports</td>
<td>21</td>
</tr>
<tr>
<td>Saving Objects</td>
<td>21</td>
</tr>
<tr>
<td>Viewing Report Server Information</td>
<td>21</td>
</tr>
<tr>
<td>Designing Reports</td>
<td>21</td>
</tr>
<tr>
<td>Before Designing a Report</td>
<td>21</td>
</tr>
<tr>
<td>Using the Report Designer to Design a Report</td>
<td>22</td>
</tr>
<tr>
<td>Design Considerations</td>
<td>23</td>
</tr>
<tr>
<td><strong>Chapter 2. Defining Members</strong></td>
<td></td>
</tr>
<tr>
<td>About Defining Members</td>
<td>27</td>
</tr>
<tr>
<td>Assigning Members</td>
<td>28</td>
</tr>
<tr>
<td>Assigning Members to Data Rows or Columns</td>
<td>28</td>
</tr>
<tr>
<td>Assigning Members to the Page Axis</td>
<td>29</td>
</tr>
<tr>
<td>Assigning Special Members</td>
<td>30</td>
</tr>
<tr>
<td>Assigning Members Using Functions</td>
<td>32</td>
</tr>
<tr>
<td>Functions Available Through Member Selection and the Formula Bar</td>
<td>33</td>
</tr>
<tr>
<td>Functions Available Only Through Member Selection</td>
<td>34</td>
</tr>
<tr>
<td>Searching for Members</td>
<td>38</td>
</tr>
<tr>
<td>Sorting Members</td>
<td>39</td>
</tr>
<tr>
<td>Previewing Selected Members</td>
<td>40</td>
</tr>
<tr>
<td>Copying Members Between Applications</td>
<td>40</td>
</tr>
<tr>
<td>Removing or Adding Members</td>
<td>41</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
</tr>
<tr>
<td>------</td>
<td>---------</td>
</tr>
<tr>
<td>63</td>
<td>Local Objects in Financial Reporting</td>
</tr>
<tr>
<td>64</td>
<td>Adding Text Objects</td>
</tr>
<tr>
<td>65</td>
<td>Modifying Text Objects</td>
</tr>
<tr>
<td>65</td>
<td>Saving Text Objects</td>
</tr>
<tr>
<td>66</td>
<td>Adding Grids</td>
</tr>
<tr>
<td>66</td>
<td>About Grids</td>
</tr>
<tr>
<td>67</td>
<td>Defining the Dimension Layout for a Grid</td>
</tr>
<tr>
<td>68</td>
<td>Adding a New Grid</td>
</tr>
<tr>
<td>68</td>
<td>Inserting an Existing Grid</td>
</tr>
<tr>
<td>68</td>
<td>Selecting a Grid</td>
</tr>
<tr>
<td>69</td>
<td>Modifying the Properties of Grid Objects</td>
</tr>
<tr>
<td>69</td>
<td>Saving Grid Objects</td>
</tr>
<tr>
<td>70</td>
<td>Inserting Images</td>
</tr>
<tr>
<td>71</td>
<td>Saving Image Objects</td>
</tr>
<tr>
<td>71</td>
<td>Inserting Charts</td>
</tr>
<tr>
<td>72</td>
<td>Saving Chart Objects</td>
</tr>
<tr>
<td>72</td>
<td>Inserting Saved Report Objects</td>
</tr>
<tr>
<td>73</td>
<td>Creating Headers and Footers</td>
</tr>
<tr>
<td>74</td>
<td>Positioning Report Objects</td>
</tr>
<tr>
<td>75</td>
<td>Resizing Report Objects</td>
</tr>
<tr>
<td>76</td>
<td>Saving Report Objects</td>
</tr>
<tr>
<td>76</td>
<td>Removing Report Objects from a Report</td>
</tr>
<tr>
<td>76</td>
<td>Designing Headers and Footers</td>
</tr>
<tr>
<td>77</td>
<td>Creating a Title Page</td>
</tr>
<tr>
<td>77</td>
<td>Inserting Page Breaks Before Report Objects</td>
</tr>
<tr>
<td>78</td>
<td>Viewing Report Summaries</td>
</tr>
<tr>
<td>78</td>
<td>Viewing Server Information</td>
</tr>
</tbody>
</table>

**Chapter 5. Customizing Grids**

<table>
<thead>
<tr>
<th>Page</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>Inserting Rows or Columns in Grids</td>
</tr>
<tr>
<td>83</td>
<td>Cutting, Copying, and Pasting in Grids</td>
</tr>
<tr>
<td>83</td>
<td>Specifying Row and Column Headings for Grids</td>
</tr>
<tr>
<td>85</td>
<td>Overriding Row and Column Headings for Grids</td>
</tr>
<tr>
<td>86</td>
<td>Using Row and Column Templates</td>
</tr>
<tr>
<td>86</td>
<td>About Row and Column Templates</td>
</tr>
<tr>
<td>87</td>
<td>How Inheritance Works with Row and Column Templates</td>
</tr>
<tr>
<td>88</td>
<td>Property Values in Rows and Column Templates</td>
</tr>
<tr>
<td>89</td>
<td>Creating Row and Column Templates</td>
</tr>
<tr>
<td>92</td>
<td>Inserting a Row and Column Template</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Unlinking a Row and Column Template</td>
<td>92</td>
</tr>
<tr>
<td>Modifying a Row and Column Template</td>
<td>93</td>
</tr>
<tr>
<td>Saving a Row and Column Template</td>
<td>93</td>
</tr>
<tr>
<td>Setting Data Query Optimization Options</td>
<td>94</td>
</tr>
<tr>
<td>Displaying Line Item Detail for Financial Management</td>
<td>95</td>
</tr>
<tr>
<td>Showing Supporting Detail for Planning</td>
<td>95</td>
</tr>
<tr>
<td>Defining the Grid Point of View</td>
<td>97</td>
</tr>
<tr>
<td>Showing or Hiding the Grid POV Bar</td>
<td>97</td>
</tr>
<tr>
<td>Selecting Members for the Grid Point of View</td>
<td>98</td>
</tr>
<tr>
<td>Setting Up a Grid Point of View</td>
<td>98</td>
</tr>
<tr>
<td>Inserting Page Breaks</td>
<td>99</td>
</tr>
<tr>
<td>Defining Page Axis Headings</td>
<td>99</td>
</tr>
<tr>
<td>Changing the Database Connection of Grids</td>
<td>101</td>
</tr>
<tr>
<td>Assigning Multiple Database Connections Grids</td>
<td>101</td>
</tr>
<tr>
<td>Adding Auto Calculations to Data Rows and Columns</td>
<td>102</td>
</tr>
<tr>
<td>Applying Formatting to Auto Calculated Data Rows and Columns</td>
<td>105</td>
</tr>
<tr>
<td>Defining Formula Rows or Columns</td>
<td>106</td>
</tr>
<tr>
<td>Defining Headings for Formula Rows and Columns</td>
<td>108</td>
</tr>
<tr>
<td>Expanding Rows or Columns</td>
<td>108</td>
</tr>
<tr>
<td>Multiple Calculation Iterations</td>
<td>112</td>
</tr>
<tr>
<td>Using the Formula Bar</td>
<td>113</td>
</tr>
<tr>
<td>Example Report with Formulas</td>
<td>113</td>
</tr>
<tr>
<td>Creating Formulas Using the Formula Bar</td>
<td>115</td>
</tr>
<tr>
<td>Editing Formulas Using the Formula Bar</td>
<td>116</td>
</tr>
<tr>
<td>Deleting Formulas</td>
<td>116</td>
</tr>
<tr>
<td>Modifying Member Selection Using the Formula Bar</td>
<td>117</td>
</tr>
<tr>
<td>Accessing Property Sheet Features</td>
<td>117</td>
</tr>
<tr>
<td>Calculating at the Cell Level</td>
<td>118</td>
</tr>
<tr>
<td>Using Text Functions to Display Information</td>
<td>119</td>
</tr>
<tr>
<td>Freezing Grid Headers</td>
<td>120</td>
</tr>
<tr>
<td>Chapter 6. Defining the User POV</td>
<td>121</td>
</tr>
<tr>
<td>Setting Up the User POV from the Report Designer</td>
<td>121</td>
</tr>
<tr>
<td>Selecting Members for the User POV from the Report Designer</td>
<td>122</td>
</tr>
<tr>
<td>Chapter 7. Providing for Detailed Data and Documents in Reports</td>
<td>123</td>
</tr>
<tr>
<td>Setting Up Expansions to Access Detail Data Within Reports</td>
<td>124</td>
</tr>
<tr>
<td>Positioning Expansion Rows and Columns</td>
<td>124</td>
</tr>
<tr>
<td>Attaching Cell Documents in a Report</td>
<td>124</td>
</tr>
<tr>
<td>Supported Microsoft Office File Types</td>
<td>125</td>
</tr>
</tbody>
</table>
Sqrt ...........................................................  191
Sum ...........................................................  191
Truncate / Trunc .................................................  192
Text Functions ......................................................  194
  About Text Functions ..............................................  194
  Annotation ......................................................  200
  CalcStatus ......................................................  202
  CellText ........................................................  203
  Date ...........................................................  206
  DataSource ......................................................  209
  DPDescription ...................................................  210
  DPJustification ...................................................  211
  DPNotesAndAttachments ...........................................  212
  Footnote .......................................................  213
  GetCell .........................................................  215
  GetHeading .....................................................  216
  GridDimension ..................................................  217
  HFMCurrency ...................................................  218
  ListOfCellDocuments ..............................................  219
  MemberAlias ....................................................  220
  MemberDescription ...............................................  221
  MemberName ...................................................  223
  MemberProperty .................................................  224
  MemberQualifiedName ............................................  225
  Page ...........................................................  226
  PageCount ......................................................  227
  PlanningAnnotations ..............................................  227
  Specifying Demarcations Between Annotations .........................  229
  ProcessManagementStatus ..........................................  230
  ReportAuthor ....................................................  230
  ReportCreated ...................................................  231
  ReportDesc .....................................................  231
  ReportFolder ....................................................  232
  ReportModified ..................................................  232
  ReportModifiedBy ................................................  233
  ReportName .....................................................  233
  ReportRunBy ....................................................  233
  RetrieveValue ....................................................  234
Conditional Functions ................................................  234
Chapter 11. Managing the Explore Repository ............................................. 245
Creating Folders ........................................................................ 245
Duplicating Objects in the Repository ........................................ 246
Filtering Report Objects ................................................................ 246
Cutting, Copying, and Pasting in the Repository ................................ 247
Managing Database Connections ............................................... 247
Exporting Repository Objects and Report Objects ......................... 247
Importing Report Objects into the Repository ............................... 247
Sending Links to Reports Through Email ..................................... 247
Deleting Items from the Repository ............................................. 248
Changing the Database Connection for Reports and Report Objects in the Repository ........................................................................ 248
Changing a Report Database Connection ..................................... 248
Managing Security for Objects in the Repository ......................... 249

Chapter 12. Using Batch Processing .................................................... 251

Chapter 13. Working with Books and Snapshot Books .......................... 253

Support for Non ASCII User Accounts ......................................... 255
POV Management ...................................................................... 255
Using the User POV Utility ...................................................... 256
Financial Reporting Initial User Preferences Definition ..................... 259
Report Server Tasks ............................................................... 260
  Specifying the Maximum Number of Calculation Iterations ........... 260
  Assigning Financial Reporting TCP Ports for Firewall Environments or Port Conflict Resolution .................................................. 261
  Accessing Server Components Through a Device that Performs NAT ........................................................................ 262
Essbase Ports ................................................................. 263
  Checking the Current Essbase Connections .................................. 265
  Changing Settings in Essbase Configuration Files ......................... 265
  Calculating the Formula for the Maximum Number of Essbase Ports ........................................................................ 265
Scheduler Command Line Interface ............................................. 266
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating Batch Input Files</td>
<td>266</td>
</tr>
<tr>
<td>Launching Batches from a Command Line</td>
<td>267</td>
</tr>
<tr>
<td>Scheduling Batches Using an External Scheduler</td>
<td>267</td>
</tr>
<tr>
<td>Encoding Passwords</td>
<td>267</td>
</tr>
<tr>
<td>Modifying Attributes</td>
<td>268</td>
</tr>
<tr>
<td>Batch Input File XML Tag Reference</td>
<td>270</td>
</tr>
<tr>
<td>BATCH_JOB_OBJECT - Node Tag</td>
<td>270</td>
</tr>
<tr>
<td>RUN_OPTIONS - Child Node Tag</td>
<td>270</td>
</tr>
<tr>
<td>NOTIFICATION / EMAIL - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>JOB_STATUS - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>JOB_OBJECT - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>DATA_SOURCE_USER_CREDENTIALS - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>HR_USER_CREDENTIALS - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>OUTPUT_OPTIONS - Child Node Tag</td>
<td>271</td>
</tr>
<tr>
<td>CHILD NODE - HTML</td>
<td>272</td>
</tr>
<tr>
<td>CHILD NODE - PDF</td>
<td>272</td>
</tr>
<tr>
<td>CHILD NODE - SAVE_AS_SNAPSHOT</td>
<td>272</td>
</tr>
<tr>
<td>CHILD NODE - PRINT</td>
<td>272</td>
</tr>
<tr>
<td>USER_POV - Child Node</td>
<td>272</td>
</tr>
<tr>
<td>Increasing the ADM Maximum Property Value Length</td>
<td>273</td>
</tr>
<tr>
<td>Microsoft Word Template</td>
<td>273</td>
</tr>
<tr>
<td>Showing Cell Text, Planning Unit Annotations and Financial Reporting Documents in Annotations</td>
<td>275</td>
</tr>
<tr>
<td>Updating Fonts in Financial Reporting in Linux/UNIX Server Environments</td>
<td>275</td>
</tr>
</tbody>
</table>

**Appendix A. Frequently Asked Questions**

- Why does changing dimensions in the User POV have no affect on the grid? .......................... 277
- Why are the Grid POVs not displayed in the Report Designer? ........................................ 277
- How will a user running Workspace know that the dimension is set on the Grid POV? ............. 278
- How can you design a report so that changes to the User POV changes the members selected from a row, column, or page axis on a grid? ........................................ 279
- Why do you need a Book POV? ................................................................................................ 279
- What are Report Member Selections? ....................................................................................... 279
- Why not call Report Member Selections the Report POV? ..................................................... 279
- Why do you need a Batch POV? ................................................................................................ 279

**Appendix B. Currency Table**

- ................................................................................................................................. 281

**Appendix C. Using Oracle User Productivity Kit**

- ................................................................................................................................. 283
Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.
Documentation Feedback

Send feedback on this documentation to: epmdoc_ww@oracle.com

Follow EPM Information Development on these social media sites:
LinkedIn - http://www.linkedin.com/groups?gid=3127051&goback=.gmp_3127051
Twitter - http://twitter.com/hyperionepminfo
Facebook - http://www.facebook.com/pages/Hyperion-EPM-Info/102682103112642
Google+ - https://plus.google.com/106915048672979407731/#106915048672979407731/posts
YouTube - https://www.youtube.com/user/EvolvingBI
About Financial Reporting

Subtopics

- Using Reporting Studio to Design Reports
- Using EPM Workspace to Work With Books and Batches

Oracle Hyperion Financial Reporting is a powerful tool for designing and presenting analytic data graphically. You can design traditional financial report formats such as cash management reports, profit and loss statements, and balance sheets. You can also design nontraditional formats for financial or analytic data that include text and graphics.

Financial Reporting uses an object-oriented graphical report layout. Whether you have little or advanced experience using an object-oriented graphical program, it is easy to design reports. You use report objects, such as text boxes, grids, images, and charts, to design your reports. You can use the same report objects in multiple reports.

Financial Reporting tasks can be preformed in Oracle Hyperion Financial Reporting Studio, Oracle Hyperion Enterprise Performance Management Workspace, or both.

Using Reporting Studio to Design Reports

Subtopics

- Studio Explore
- Report Designer
- Property Sheets

Reporting Studio is where you design your reports. After you log in to Reporting Studio, you can create a new report, open an existing report, and access the repository. When designing a
report, you can select menu commands, use shortcuts from the toolbar, or right-click to select shortcut menu items.

You can design and view reports that retrieve and format data from Oracle Hyperion Financial Management, Oracle Hyperion Planning, and Oracle Essbase.

**Note:** You must be a Report Designer or Global Administrator to design report in the Report Designer. See the Oracle Enterprise Performance Management System Security Configuration Guide for information on rights and security.

**Studio Explore**

After you log in to Reporting Studio, the Studio Explore page is displayed. To navigate in Studio Explore, use the menu commands at the top of the page, use the toolbar buttons, or right-click to access the shortcut menus.

You can perform the following tasks in Studio Explore:

- Create a new report
- Open an existing report
- Access the Reporting Studio repository
- Display report server information

**Report Designer**

After you select to create a new report or open an existing report in Reporting Studio, the Report Designer is displayed. In the Report Designer, you select menu commands or use toolbar buttons to add report objects to the report. You can then customize each report object using its property sheet. For example, you can select a grid, and use the Grid Properties property sheet to specify dimensions, select members, and specify heading options. You can also select a report object and right-click to display a shortcut menu.

You can use the following components as you design reports in Report Designer:

- Headers
- Text Boxes
- Grids
  - Grids consist of the following:
    - **Rows**—Rows can contain data, text, or a formula. You can format individual rows in grids.
    - **Columns**—Columns can contain data, text, or a formula. You can format individual columns in grids.
    - **Page Axes**—The third intersecting reference on a grid report object. A page repeats the data on the row and column over multiple members of additional dimensions. You can define members on a page to show different views of data without rerunning the report.
Property Sheets

Property sheets define the options for the currently-selected report component. When you set a property option, it is automatically applied to the selected report component.

Property sheets change based on the category selected. There are several categories available, depending on the type of report component selected. Each category shows another set of characteristics you can set for a selected report component. By default, property sheets are displayed on the right side of the Report Designer.

To show or hide property sheets, select View, and then Property Sheet.

Using EPM Workspace to Work With Books and Batches

In addition to the functionality available in Reporting Studio, EPM Workspace provides expanded functionality when designing reports. In EPM Workspace, you can open, copy, delete, move, import, export, and rename repository objects such as reports, books, and graphics; you can create, open delete, import export, and rename folders; and you can preview reports and books in HTML or PDF format.

Components in EPM Workspace include:

- **Explore**—Open stored files, create folders, export and import files, and set up email links to folders, reports, and snapshot books.
- **Book Editor**—Create a book of reports, which can be saved as a book or a snapshot book.
- **Batch Editor**—Add objects to a batch job, remove objects from a batch job, schedule a batch, and indicate whether you want the batch deleted after it runs.
- **Scheduler**—Schedule a batch and view the status of batches that are scheduled for processing.

**Note:** If you have a Viewer role, you can view reports and export documents to Microsoft Word, Microsoft Excel, and Microsoft PowerPoint.

Basic Reporting Studio Tasks

Subtopics

- Installing Reporting Studio
- Starting Reporting Studio
- Opening Reports
- Saving Objects
- Viewing Report Server Information

Note: This section discusses the Financial Reporting tasks performed in Reporting Studio. For information on the Financial Reporting task performed in EPM Workspace, see the Oracle Hyperion Enterprise Performance Management Workspace User's Guide.

Installing Reporting Studio

➢ To install Reporting Studio:

1. From EPM Workspace, select Tools, and then Install, and then Launch Financial Reporting Studio.
2. After the download is complete, close all browsers and run the Reporting Studio installer (FinancialReportingStudio.exe) as an administrator.

Starting Reporting Studio

➢ To start Reporting Studio:

1. Select Start, then All Programs, then Oracle, then Financial Reporting Studio 11.1.2.4.000, and then Financial Reporting Studio 11.1.2.4.000.
2. In the Login dialog box, enter your User Name, Password, and Server URL.

➢ To start Reporting Studio from a web browser:

1. Ensure that you have Reporting Studio installed.
2. Highlight a report in the repository, then select File, then Open In, and then Studio Preview.

You can also right-click a report and select Open In, and then Studio Preview.

The selected report is displayed in Reporting Studio.

Tip: To start Reporting Studio without opening a report, logon to EPM Workspace, select Tools and then, select Launch Financial Reporting Studio.

Note: The version of Reporting Studio must match the version of the Financial Reporting server. To find the version of Reporting Studio, select Help/About, and then select Show Details.
Opening Reports
You can open an existing report to modify the report design or to display the report with current data. Data is refreshed when the report is opened or viewed. Static data can be saved with a report when you save it as a snapshot report.

To open a report, select File, and then Open, or double-click the desired report.

Saving Objects
To save an object in Reporting Studio, from the Report Designer, select File, and then Save Object.

Viewing Report Server Information
In Reporting Studio, you can review report server information including:
- Report Activity
- JVM Memory
- Threads

To view report server information, select File, and then Report Server Info.

Designing Reports
Subtopics
- Before Designing a Report
- Using the Report Designer to Design a Report
- Design Considerations

Before Designing a Report
Before designing reports, consider the following points:
- Has the data source application been designed to support the required hierarchies and calculations for all reports?
- What is the goal of the report?
- Who are the end users of the report?
- How frequently is the report generated and by how many users?
- What is an acceptable retrieval/generation time?
- How many cells does the report query before suppression?
Using the Report Designer to Design a Report

You can design reports in the Report Designer using a blank report layout where you can specify the contents of the report, including grids, charts, images, text boxes, headers, and footers.

To design a report in the Report Designer:

1. In Reporting Studio, select File, then New, and then Report.
2. Select Insert, then Grid, and then select an area for the grid.
3. In Select a Database Connection, select an existing database connection, or click New Database Connection to create a new database connection.
4. Enter your User Name and Password, and then click OK.

If you create a new database connection, consider the following points:

- If you are designing a report with a Planning data source, select Planning Details as the data source type only if your grid contains Supporting Detail or Planning Unit Annotations; otherwise, select Essbase or Financial Management as the data source type.
- For an Essbase or Financial Management data source, enter the database server name in the Server text box.
  
  You can access an Essbase server in embedded mode or 3 Tier APS mode. For the 3 Tier APS mode, Financial Reporting defines the APS server machine name in the JConsole using the EssbaseJAPIServer property. All API calls are sent through that server.
  
  - In both modes, you specify the Essbase server directly; for example, EssServer1.
    
    With an Essbase cluster name, for both modes, you can either:
    
    - Specify the Essbase cluster name directly. For example, EssCluster1. Financial Reporting internally resolves the cluster name to the server name at runtime.
    - Enter the APS URL with an Essbase cluster name. For example: http://host:port/aps/Essbase?ClusterName=EssCluster1, where host:port is the APS Server to resolve the Essbase cluster name. Failover is supported across the Essbase servers in the Essbase cluster.
  
  - In 3 Tier mode only, you can enter the APS cluster name. For example, APSCluster1.
    
    The APS cluster name and members are defined in the Essbase Admin Utility. This method supports failover and load balancing across the Essbase servers in the APS cluster.

- Click to select the Application/Catalog and Database.
  
  The Database/Cube Lookup is combined into the Application/Catalog Lookup button. The Application/Catalog Lookup button displays a tree view of the applications and catalogs and corresponding database and cubes; The applications are listed as parents and the databases are listed as children. For data sources that are not associated with a database, only applications are listed.
Tip: To view guidelines when designing a report, select View, and then Guidelines, or right-click and select Show Guidelines. To display row and column headers when previewing the report on the web, select View, and then Row & Column Headers in Web Preview.

Design Considerations

Subtopics

- Report Performance
- Planning as a Database Connection
- Support for Workforce Planning Smart Lists
- Database Connection for Essbase and Planning
- Data Queries - Expanded Versus Single Data Segments
- Planning Considerations for Designing Reports Using Suppress Missing Blocks
- Considerations for Changes to Dimension and Member Names
- Recommendations for Designing Dynamic Reports

Report Performance

Report performance can vary for each client. What is acceptable for one client may not be acceptable for another. For example, if reports are run in batches during off-peak hours, they may have to be completed before the next business day. When data is updated hourly, report performance may only be acceptable if report output returns in seconds.

Planning as a Database Connection

Once you install the Planning ADM driver, you can select Planning as a database connection. The Planning Details ADM driver is optimized as a data source to provide features such as supporting details, planning unit annotations and metadata filtering. If your report grid does not use these Planning features, for optimal performance, choose Essbase as the database connection for your report. For more information, see “Showing Supporting Detail for Planning” on page 95 and “PlanningAnnotations” on page 227.

Additional Considerations

- Attribute dimensions are not supported as dimensions in Planning. (Use the MemberProperty text function to display attribute values in your report.)
- Hsp_Rates is not available as a dimension when using the Planning data source.
- Member selections using any level but level 0 are not supported.

Support for Workforce Planning Smart Lists

Smart Lists are attribute-like values assigned to members, where the actual assignment is stored as data in the Essbase cube for the member (for example, 0, 1, 2, 3, 4) and the Planning Repository
translates those numbers to textual strings (for example, 0="Straight Line Depreciation", 1="Double Declining Balance").

Examples of Financial Reporting Support of Smart Lists:

- Grids which reference cells in a Planning database that contain Smart Lists return a string value instead of a number. String values are always shown in the results instead of numbers.
- The Text function `GetCell()` returns the string value instead of a number when it references a cell containing an Smart List.
- Using numeric comparison operators and string comparison operators (for example, equals and startswith), Conditional Formatting and Suppression allows selection of any cell (Data, Formula, Text) with the "Cell Value" operand. For example, Conditional Suppression can refer to a data cell which contains a Smart List string and apply Conditional Suppression such as "suppress row if cell value B6 starts with 'ABC'".
- You can apply conditional formatting and suppression to Workforce Planning or Smart List textual data.
- When a Smart List string is retrieved from Planning, the string value (for example “Meets”) and numeric values are stored and are available, enabling the following capabilities:
  - Conditionally suppress and conditionally format based on either the Smart List string value or the underlying numeric value.
  - Sort rows or columns that have Smart List values mixed with numeric data-cell values. When this occurs, the Smart List numeric is used in the comparison with the other numeric cell values.
  - Access the Smart List numeric value through a formula reference to the cell. All formula references to Smart-List cells operate on the Smart List numeric value.

Database Connection for Essbase and Planning

When you change a report from an Oracle Hyperion Interactive Reporting database connection to a Planning database connection or vice versa, there are several restrictions because the data source reference is the only change. For more information, see “Changing a Report Database Connection” on page 248.

Data Queries - Expanded Versus Single Data Segments

A data segment is a row or column that retrieves data from a database. An expanded data segment is a row or column that can expand, so that the resulting grid expands to two or more rows or columns when viewed. Often, expanded data segments use functions such as `Children Of` or `Descendants Of`. A single data segment is a row or column that remains a single row or column when shown in the viewer.

While it is generally valid to use expanded and single data segments in the same grid, when designing a grid with large amounts of data, consider using expanded data segments rather than single data segments. Expanded data segments provide some performance advantages over single
data segments; however, to produce detailed formatting on different data rows or columns, use single data segments.

Planning Considerations for Designing Reports Using Suppress Missing Blocks

You can use Suppress Missing Blocks to improve performance when rows or columns contain sparse dimensions. Suppress Missing Blocks allows placement of large sparse dimensions in rows, while providing good response time if the density of the query is low. Only blocks with data are retrieved. For example, you can place an employee dimension consisting of thousands of members in rows, and place the entity in the page or POV. Then, only employees of the selected entity are retrieved.

Suppress Missing Blocks aids the performance of Suppress Missing Data when suppressing a large number of rows, such as 90% or more. The Suppress Missing Blocks setting can degrade performance if few or no rows are suppressed. Also, certain suppressed blocks may ignore Dynamic Calc members.

Considerations for Changes to Dimension and Member Names

If the dimensions or members in the data source are renamed, you must manually update each report in Reporting Studio to reflect the changes.

Recommendations for Designing Dynamic Reports

- Use expanded data segments for optimal performance:
  - Use functions on expanded data segments that are not placed on separate rows or columns.
  - Use multiple member selections on expanded data segments that are not placed in separate rows or columns.
  - Use single data segments only when required for formatting or calculations.

- Write efficient formulas:
  - Use row or column formulas rather than cell formulas, when possible.
  - Use reference properties.
  - Use a cell reference instead of cross-axis references
  - Remove unnecessary parenthesis from formulas.

- Do not limit your report to the Grid object:
  - Add functions in text boxes that highlight specific areas.
  - Hide a grid of data in a report to highlight just the graphics.

- Create report templates:
  - Leverage the power of POVs and prompting functions.
  - Design once using the CurrentPOV and prompting functions.
● Learn the power of Grid, User, Book, and Batch POVs.
● Create reusable objects and row and column templates. Text, Image, Grid, Chart objects, and row and column templates can be reused in multiple reports.
About Defining Members

When you specify data for a grid, you set criteria that identifies the data to retrieve. You set the criteria by assigning members to the rows, columns, and page axes in the grid, as well as to the grid point of view (POV) and user POV bars. (See Chapter 6, “Defining the User POV”.)

You can create lists of reusable members, and use functions to retrieve members dynamically. Before you define members, you must insert a grid in the report and define its dimension layout. (See “Adding Grids” on page 66.)

Note: If you use Planning Details as the database connection to go against a Planning cube, all available dimensions are listed; however, you have access privileges to members only in certain dimensions. This affects the output of your report. (See “Access Privileges for Members in the Planning Database Source” on page 45.)
Assigning Members

Subtopics
- Assigning Members to Data Rows or Columns
- Assigning Members to the Page Axis
- Assigning Special Members
- Assigning Members Using Functions
- Functions Available Through Member Selection and the Formula Bar
- Functions Available Only Through Member Selection

Assigning Members to Data Rows or Columns

You assign members and member lists to retrieve data for your reports. You can select members to be displayed in your report, or you can use functions to select dimension members dynamically. You can insert a separate row or column in a grid for each selected member, or you can place all of the selected members in one row or column.

To assign members to data rows or columns:

1. Open a report.
2. In the Select Members dialog box, select a dimension.
   - You can either double-click a dimension in the grid, or select a dimension in the grid and click the button in the formula bar with the dimension name.
   - If you know the member names for a dimension, enter the member names, separated by commas, into the formula bar text area. Case sensitivity is enabled for member names, however it is not enabled for function names. For example, for the Measures dimension you might enter children of (Profit), then click the check mark to verify the member names.
3. Assign members, member list, or functions.
   - To assign individual members, select the Members tab, then highlight the members to add to the data row or column.
   - To assign member lists, select the Lists tab, then highlight the member lists to add to the data row or column.
   - To assign functions, select the Functions tab, then highlight the functions to add to the data row or column.
4. Add the highlighted items to the Selected area (click ), or add a member function (children, descendents, and so on) to a selected member, click .
   - The Add Relationship button is available only on the Members tab.
5. Optional: In the Selected area, review the selected members by clicking View, and then selecting Preview.
   - For more information, see “Previewing Selected Members” on page 40.
Optional: To insert a row or column for each selected member, select Place selections into separate rows (or Columns).

This option is available only in the Select Members dialog box.

Click OK to return to the Report Designer.

Assigning Members to the Page Axis

You can assign members to the page axis of a grid. When you select members for the page axis and run the report, the system generates a grid for each page member. If a chart references a grid with multiple page members, a new chart is generated for each page member.

For example, suppose you assign the following members of the Scenario and Entity dimensions to the page axis:

- Target 1 and Target 2 of the Scenario dimension
- Radio Promotions and Web Promotions to the Entity dimension

When you view the report, it is formatted and populated with data from all combinations of members on the page axis. The report contains four grids, one for each page member combination.

You can select any combination of the members:

- Target 1: Radio Promotions
- Target 1: Web Promotions
- Target 2: Radio Promotions
- Target 2: Web Promotions

To add or change dimensions that are assigned to the page axis, see “Adding Grids” on page 66.

Note: In Financial Reporting, the members (and corresponding data) in the page axis are all retrieved when the report is run. Reports with a large number of members on the page axis, may impact performance.

To assign members to the page axis:

1. Open a report.
2. On the Pages bar, select a dimension .
3. On the formula bar, select the button with the dimension name.

Note: If you know the member names for a dimension, enter the member name into the formula bar text area. For example, for the Measures dimension enter Profit, Sales. Click the check mark, to verify the member selection.

4. In the Member Selection dialog box, select the members to use on the page axis, and then click OK.
Assigning Special Members

You can use the Select Members dialog box to select the special members Prompt, Same As, Current Point of View, and User Point of View.

- Prompt acts like a variable that asks the user to select members when a report is run. You determine the dimension and defaults at design time, and the user selects the members when generating the report. Prompt also enables the report writer to filter the dimension members from which the user can select when running the report.
- Same As creates a column or row member selection with the same member selection setting as another column or row.
- Current Point of View acts as a variable that employs the POV to specify the member of a dimension when the report is run.
- User Point of View can be selected as a member on the book POV. The parameters can be modified by the end user.

Setting Up Report Prompts for Members

A prompt is a way of enabling the viewer to assign members to a dimension. The prompt is set up during design time. As a report designer, you can limit the type of data to assign to a dimension. You can also specify whether the Respond to Prompts dialog box displays member names, descriptions (for a Financial Management database connection) or aliases (for Essbase or Planning database connections), or both member names and descriptions or aliases.

For example, you might create a report that enables the user to select from lines of products. The report designer selects Prompt as a member of Product, then selects specific lines of product. The end user who runs the report selects from those product lines.

➢ To set up a report prompt for members:
  1. Open a report.
  2. Double-click a dimension cell for which you want to assign a prompt.
  3. In the Available area of the Members tab, select Prompt for (dimension name) and move it the Selected area.
  4. Click OK.

The Define Prompts dialog box is displayed:

5. **Optional:** To limit the prompt selection to specific default members, or a list of members, or both, take an action:
   - If the member name is known, enter the member name in Default Member or click the Default Member search button. If multiple member are provided for the prompt, members must be separated by commas. For information on selecting members, see “Assigning Members to Data Rows or Columns” on page 28.
Note: If you leave Default Member blank, the Respond to Prompts dialog box does not display a default member when a user runs the report or creates a batch. The user must specify a member rather than accept a blank default. If a member is not selected, the report is not run.

- If the list of members is known, click the Choices List button to select a list of members. If multiple member lists are provided for the prompt, member lists must be separated by commas. For more information, see “Defining Member Lists” on page 43.

6 From Member Labels in Prompt, select the labels to display in the Respond to Prompts dialog box when you run the report:

- If the database connection is Essbase or Planning, select Member Name, Alias, or Member Name and Alias.
- For Planning, select Member Name, Alias, or Member Name and Alias
- If the database connection is Financial Management, select Member Name, Description, or Member Name and Description.
- For an Essbase connection, select Member Name, Alias, or Member Name and Alias.

7 Click OK.

Setting Up Same As References
Same As enables the designer to obtain a member selection from another row or column. The following characteristics apply to Same As in Financial Reporting:

- In a report, members referred to by Same As are always from the same dimension, same type of database connection, and same grid.

- Members referred to by Same As are only available for member selections on rows and columns in a grid. Same As cannot be used as member selection in:
  - Page Axis
  - Grid POV
  - User POV
  - Book POV
  - Batches POV
  - Default member selection for a prompt
  - A response to a prompt
  - Member function member parameter, such as range
  - Member selection in the book editor

- Same As can reference a row or column that gets its member selection from a prompt or Current POV.

- Same As cannot reference another Same As reference.

- Same As is not supported within row and column templates.

- Same As must be used alone and cannot be combined with any other member selections.
To set up Same As:

1. Open a report and double-click a member selection for which you want to set up Same As.
2. On the Members tab, expand Same As Member Selections, then select Same As for the row or column for the dimension, then click Add, and then the to add the Same As for the dimension to the selected area.
3. Click OK.

Setting Up a Current Point of View

A current POV enables the designer to obtain a member for a row, column, or page from the user POV.

To set up a current POV:

1. Open a report and double-click a member selection for which you want to set up a current POV.
2. On the Members tab, select Current Point of View for the dimension, then click the Add button, to add the Current Point of View for the dimension to the Selected area.
3. Click OK.

Assigning Members Using Functions

You can use functions to select members dynamically. After you select a function, you can edit its parameters. For more information on the dynamic use of functions, see “Using the Formula Bar” on page 113. For more information on editing function parameters, see “Including a Member in a Function” on page 44.

Note: Except where noted, all functions listed are supported for Essbase and Plannings. Supported functions for Financial Management are noted in separate columns.

To assign members using functions:

1. Open a report.
2. Double-click a dimension cell.
3. In Select Members, select the Functions tab.
4. Highlight one or more functions.
5. Click the Add button, to open the Edit (name) Function dialog box.
   The Edit (name) Function dialog box is not displayed for functions that do not require parameter values, such as BottomofHierarchy, TopofHierarchy, AllMembers, and SuppressSharedMembers.
6. Double-click the Value column inside row 1, then edit its value by clicking the Lookup button.
7. Repeat step 2 through step 6 to edit the remaining rows.
Functions Available Through Member Selection and the Formula Bar

Table 1 Functions Available Through Member Selection and the Formula Bar

<table>
<thead>
<tr>
<th>Function</th>
<th>Syntax</th>
<th>Financial Management</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Children</td>
<td>Children of member (Inclusive)</td>
<td>x</td>
<td>The members one level below the specified parent</td>
</tr>
<tr>
<td>Descendants</td>
<td>Descendants of member (Inclusive)</td>
<td>x</td>
<td>The members at all levels below the specified parent</td>
</tr>
<tr>
<td>BottomOfHierarchy</td>
<td>Bottom of Hierarchy dimension</td>
<td>x</td>
<td>All Level 0 members</td>
</tr>
<tr>
<td>Siblings</td>
<td>Siblings of member (Inclusive)</td>
<td>x</td>
<td>The members with the same parent as the specified member</td>
</tr>
<tr>
<td>OnSameLevelAs</td>
<td>Same level as member</td>
<td></td>
<td>The members from the same dimension and on the same level as the specified member</td>
</tr>
<tr>
<td>SameLevelAs</td>
<td></td>
<td></td>
<td>All members of the same level.</td>
</tr>
<tr>
<td>OfSameGeneration</td>
<td>Same generation as member</td>
<td></td>
<td>The members of the same dimension and generation as the specified member</td>
</tr>
<tr>
<td>Parent</td>
<td>Parent of member (Inclusive)</td>
<td>x</td>
<td>The member one level above the specified member</td>
</tr>
<tr>
<td>TopOfHierarchy</td>
<td>Top of Hierarchy dimension</td>
<td>x</td>
<td>The top-level member of hierarchy</td>
</tr>
<tr>
<td>SystemMemberList</td>
<td>System-defined Member List Lev(n),dimension</td>
<td>x</td>
<td>The members in a specified system-defined member list</td>
</tr>
<tr>
<td>Ancestors</td>
<td>Ancestors of member (Inclusive)</td>
<td>x</td>
<td>The members at all levels above the specified member</td>
</tr>
<tr>
<td>AllMembers</td>
<td>All Members in Hierarchy dimension</td>
<td>x</td>
<td>The members of the specified dimension</td>
</tr>
<tr>
<td>Member</td>
<td>member</td>
<td>x</td>
<td>The specified member</td>
</tr>
<tr>
<td>Members</td>
<td></td>
<td></td>
<td>All members of the current dimension</td>
</tr>
<tr>
<td>Base</td>
<td>Base members of Hierarchy dimension</td>
<td>x</td>
<td>The base or level 0 members of a hierarchy. Base Member function is only available for Financial Management, and is not supported for Essbase and Planning. For Essbase and Planning, use BottomOfHierarchy AND Descendants functions.</td>
</tr>
<tr>
<td>UserDefined1</td>
<td>All members where user defined is selected attribute</td>
<td>x</td>
<td>Financial Management only. All entities and custom dimensions can be used with the following exceptions: View, ICP, Year, Period, Scenario. The members of selected attribute</td>
</tr>
<tr>
<td>Function</td>
<td>Syntax</td>
<td>Financial Management</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------</td>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UserDefined2</td>
<td>All members where user defined is selected</td>
<td>x</td>
<td>Financial Management only. All entities and custom dimensions can be used with the following exceptions: View, ICP, Year, Period, Scenario. The members of selected attribute</td>
</tr>
<tr>
<td></td>
<td>attribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UserDefined3</td>
<td>All members where user defined is selected</td>
<td>x</td>
<td>Financial Management only. All entities and custom dimensions can be used with the following exceptions: View, ICP, Year, Period, Scenario. The members of selected attribute</td>
</tr>
<tr>
<td></td>
<td>attribute</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Functions Available Only Through Member Selection

The member functions listed in the following table can be selected from the Functions tab through member selection. They cannot be entered dynamically in the formula bar.

#### Table 2  Functions Available Only Through Member Selection

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Property</td>
<td>Used with Essbase and Planning only. Extracts the members with the specified property value.</td>
</tr>
<tr>
<td>UserMemberList</td>
<td>Used with Essbase only. The members in a specified user-defined member list</td>
</tr>
</tbody>
</table>
| Match          | Used with Financial Management, Essbase and Planning. The members that match a specified pattern or set of characters.  
**Note:** When using the asterisk (*) wildcard with Planning, the pattern may only contain one asterisk and it may only appear as the last character. |
| MatchEx        | Used with Essbase only. Performs wildcard member selection. Essbase searches for member names that match the pattern you specify, and returns the member names it finds.  
**Note:** See “MATCHEX” in the Oracle Essbase Technical Reference for detailed information on this function. Financial Reporting only allows matching on the AL/T|MBR|BOTH option. |
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CommonChildren</td>
<td>Used with Financial Management only. This function is valid for the common members in the Entity dimension only. For example, with the following structure:</td>
</tr>
<tr>
<td></td>
<td>Parent1</td>
</tr>
<tr>
<td></td>
<td>ChildA</td>
</tr>
<tr>
<td></td>
<td>ChildB</td>
</tr>
<tr>
<td></td>
<td>ChildC</td>
</tr>
<tr>
<td></td>
<td>Parent2</td>
</tr>
<tr>
<td></td>
<td>ChildA</td>
</tr>
<tr>
<td></td>
<td>ChildB</td>
</tr>
<tr>
<td></td>
<td>ChildD</td>
</tr>
<tr>
<td></td>
<td>The common children function replaces the Member with the Parent. In this example, the Member is Parent1 and the Parent is Parent2. Therefore, the result of using the CommonChildren function in this example would be the following output:</td>
</tr>
<tr>
<td></td>
<td>Parent2.ChildA</td>
</tr>
<tr>
<td></td>
<td>Parent2.ChildB</td>
</tr>
<tr>
<td></td>
<td>Parent2.ChildC</td>
</tr>
<tr>
<td></td>
<td>This is the children of Parent1 with Parent2 now substituted as the parent. In this example, since ChildA and ChildB are children of Parent2 the result for these two would be valid. However, the result for Parent2.ChildC is not really valid and would return nothing.</td>
</tr>
<tr>
<td>PeriodOffset</td>
<td>Used with a Financial Management database connection only. This function enables you to perform arithmetic offset on only the Period dimension.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Arithmetic offset is the distance to move forward or backward from a specified level.</td>
</tr>
<tr>
<td></td>
<td>For example, to list this data:</td>
</tr>
<tr>
<td></td>
<td><strong>Current Qtr Next Month</strong></td>
</tr>
<tr>
<td></td>
<td>you set the parameters as follows:</td>
</tr>
<tr>
<td></td>
<td>Member = Current POV</td>
</tr>
<tr>
<td></td>
<td>Offset = +1</td>
</tr>
<tr>
<td></td>
<td>Hierarchy = Dim</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> PeriodOffset always spans the period and year combination. Therefore, if you select Jan offset -2 (assuming the year is 2002), the answer (member shown) would be Nov 2001.</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RelativeMember</td>
<td>Used for all dimensions with Essbase and Planning database connections. This function is also available for the Year dimension with a Financial Management database connection. This function enables you to perform arithmetic offset.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>Arithmetic offset is the distance to move forward or backward on a specified level.</td>
</tr>
<tr>
<td></td>
<td>For example, with an Essbase database connection, to list this data:</td>
</tr>
<tr>
<td><strong>Current Qtr Next Month</strong></td>
<td>you set the parameters as follows:</td>
</tr>
<tr>
<td></td>
<td>Member = Current Point Of View</td>
</tr>
<tr>
<td></td>
<td>Offset = 1</td>
</tr>
<tr>
<td></td>
<td>Hierarchy = Year</td>
</tr>
<tr>
<td></td>
<td>RelativeMemberList = Lev0, Year</td>
</tr>
<tr>
<td></td>
<td>UseFirstDescendant = checked</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>In this example, UseFirstDescendant = checked causes the system to start from the first descendant of Qtr1, which is Jan. The offset is +1, which makes the Next Month = Feb.</td>
</tr>
<tr>
<td></td>
<td>If the current POV = Qtr1, the result is as follows:</td>
</tr>
<tr>
<td><strong>Current Qtr Next Month</strong></td>
<td>Qtr1 Feb</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>When using a time dimension, keep the offset in the same calendar year as the specified level. For example, if you start with June, you can offset back to January or forward to December.</td>
</tr>
<tr>
<td></td>
<td>For a Financial Management database connection, to list this data:</td>
</tr>
<tr>
<td><strong>Year Next Year</strong></td>
<td>you set the parameters as follows:</td>
</tr>
<tr>
<td></td>
<td>Year = Current Point Of View for Year</td>
</tr>
<tr>
<td></td>
<td>Offset = 1</td>
</tr>
<tr>
<td></td>
<td>Hierarchy = Year</td>
</tr>
<tr>
<td></td>
<td>If the current Point of View = Year, the result is as follows:</td>
</tr>
<tr>
<td><strong>Current Year Next Year</strong></td>
<td>2003 2004</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| DynamicMemberList     | Used with Financial Management database connection only. This function enables you to select a member list that is defined to support dynamic POV on the Entity dimension. DynamicMemberList is based on a POV, and takes Entity and Parent (optional) as parameters, If the Parent parameter is specified, the Parent name is used for all entities returned from the enumeration of the list. If the Parent parameter is not specified, then no substitution is done.  
**Note:** The enhanced Financial Management DynamicMemberList functionality is not available in Financial Reporting.  
Entity is based on the Scenario, Year, and Period defined at the POV level. Therefore, the system will not use Scenario, Year, and Period on the grid. If you place Scenario, Year, or Period on the grid, the following message is displayed: “5200: Error executing query: Scenario, year, and period must be specified on the point of view for a dynamic member list.”  
**Note:** The DynamicMemberList can be executed from the Lists tab or Functions tab. On the Lists tab, you begin by selecting a (dynamic) list, then adding the Entity and Parent parameters. On the Functions tab, you begin by selecting the DynamicMemberList function, then selecting a dynamic list (DynamicNamedGroup), Entity, and Parent. |
| Dynamic Time Series Members | The following Dynamic Time Series members can be defined within Essbase. Only Dynamic Time Series members that are defined in the Essbase database are displayed in the Functions tab of the Select Members dialog box in Financial Reporting.  
- H-T-D History-to-date  
- Y-T-D Year-to-date  
- S-T-D Season-to-date  
- P-T-D Period-to-date  
- Q-T-D Quarter-to-date  
- M-T-D Month-to-date  
- W-T-D Week-to-date  
- D-T-D Day-to-date  
These members provide up to eight levels of period-to-date reporting. How many members you use, and which members, depends on your data and database outline.  
**Note:** These members are used only with time-based dimensions and the lowest-level data. You can view H-T-D, Q-T-D, and M-T-D Dynamic Time Series members on the Functions tab if you select the Year dimension for the Sample application and Basic database.  
For more information on Dynamic Time Series members, see the *Oracle Essbase Database Administrator's Guide*, Volume 1. |
| Substitution Variables | Stored in a directory for the selected dimension in the Select Members dialog box, substitution variables serve as placeholders for information that changes regularly. A variable value can be changed at any time by the Essbase administrator. These are useful for reports that depend on reporting periods. With a substitution variable, such as CurMnth, set on the server, you can change the assigned value each month to the time period.  
**Note:** The list of available Essbase substitution variables displayed in the Member Selection dialog is obtained when you start the Financial Reporting Report Client. If you add a new substitution variable using the Essbase Application Manager, you must restart the Report Client to see the addition. Anytime you run a report, the latest values for all Essbase substitution variables are used. You can view a CurMonth substitution variable member on the Members tab if you select the Year dimension for the Sample application and Basic database. A substitution variable is preceded by a green square.  
For more information on substitution variables, see the *Oracle Essbase Database Administrator's Guide*, Volume 1. |
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SuppressSharedMember</td>
<td>For Essbase and Planning only, suppress the display of duplicate members. This function is only valid with the following functions: DIMBOTTOM, OFSAMEGEN, or ONSAMELEVELAS.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Use UNION and not AND to work correctly.</td>
</tr>
<tr>
<td>Range</td>
<td>For Essbase, and Planning database connections, you can specify a range of members by selecting a start member and an end member of the range. For example, you can use the Year dimension to specify all months in the first quarter by selecting January as the start member of the range and March as the end member. For Financial Management, the Range function applies only to the Period dimension. For example, to list this data: Beginning Period Ending Period for a range of members in the current quarter, you set the parameters as follows: Start Member = July End Member = September Years to Iterate Over = 0 The number of iterations you specify determines the number of times you pass through the Period dimension. For example, if you specify 1 for the Years to Iterate Over parameter, the report spans July to September of the following year. <strong>Note:</strong> Do not use a Prompt and a Range function in the same grid when the Range function also includes Prompts.</td>
</tr>
<tr>
<td>OrderBy</td>
<td>For Planning database connection only, you can specify the order of members that come back from member selection by a member property. The OrderBy function comprises two required parameters: Property—Provides the Member Name option and a list of all properties available for members. Name is the default. Direction—Provides the options for directions: Ascending, Descending, Hierarchy Ascending, Hierarchy Descending. The OrderBy function applies to all member selections in the Selected list.</td>
</tr>
<tr>
<td>SortHierarchy</td>
<td>For Web Analysis only, sort members in outline order, by hierarchy. It is translated into Essbase report script command &lt;sortHierarchy. See Essbase documentation for details.</td>
</tr>
</tbody>
</table>

**Searching for Members**

You can perform searches for members to edit. You can search using a text string or by property (that is, name and description). You can also use a blank space as a separator to perform simultaneous searches. If you enclose a string of characters in quotation marks, the system searches for an exact match, including the blank spaces between the characters.

The system saves the previous 25 search criteria strings. If you are searching for a list or function you must activate this function by using the Find Next or Find Previous button.
Note: For Financial Management, when you search for an entity, you must search on parent.entity (for example. D62VIELO.D62475LO). When you search just the entity, you must precede the entity with a wildcard, (for example, *D62475LO).

**Table 3** Find Options

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find text box</td>
<td>Used to enter the characters for performing a search, or select a saved search criteria string from the list. The search string can be a partial search string, a word, or multiple words.</td>
</tr>
<tr>
<td>Use Wildcards</td>
<td>Use the Find text with wildcard symbols (* and ?) to search for the specified members. By default, Use Wildcards is enabled.</td>
</tr>
<tr>
<td></td>
<td>- Use ? to represent a single character search.</td>
</tr>
<tr>
<td></td>
<td>- Use * to represent a multiple character search.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If the search criteria has a space in it, enclose the search phrase in a double quote. For example, <em>IC Offset</em> should be &quot;<em>IC Offset</em>&quot;. If the location of the &quot;<em>&quot; wildcard is at the end only, for example IC Offset</em>, then you need not enclose the phrase in quotes.</td>
</tr>
<tr>
<td>Parameters...</td>
<td>Click to set search parameters for the Find text box. Choose from the following parameter options:</td>
</tr>
<tr>
<td></td>
<td>- Search—Select a member property by which to search.</td>
</tr>
<tr>
<td></td>
<td>- Wildcard Search (default)—Insert the text and wildcard symbols (* and ?) in the Find text box to search for the specified member. The * represents multiple characters and the ? represents a single character.</td>
</tr>
<tr>
<td></td>
<td>- Case-sensitive Search—Allows case-sensitive text in the Find text box to search for the specified member.</td>
</tr>
<tr>
<td>(Find Previous)</td>
<td>Use this to search for a list or function. Highlights the previous item that matches the search criteria.</td>
</tr>
<tr>
<td>(Find Next)</td>
<td>Use this to search for a list or function. Highlights the next item that matches the search criteria.</td>
</tr>
</tbody>
</table>

To search for members:

1. Open a report, and double-click a dimension cell.

2. In the Available dialog box, enter text or select a saved criteria string.

   The wildcard option is automatically enabled during member selection.

3. Perform an action:

   - To search for a member, click Find.
   - To search for a list or a function, click Find Next, or Find Previous.

4. Click OK to return to your report.

**Sorting Members**

You can sort members to facilitate viewing them within member selection. By default, the members are displayed in the sort order of the database connection.
You can sort selectable members in a preferred order, and you can sort the members that are selected to match the row or column order of data in your report. The order in which selected items are displayed in member selection is the order in which they are evaluated and displayed on the report.

You sort the members by property and display them in ascending or descending order. By default, the system sorts the list numerically, then alphabetically.

You can enable the system to sort the list whenever the view changes, or you can sort the list manually.

➤ To sort members manually:
1. Open a report, and double-click a dimension cell.
2. In the Selected Members list, click View, and then select Sort.
3. In Sort (dimension) Items, select the property for the primary sort, then click.
4. Click OK twice.

Previewing Selected Members

You can preview the members or member lists selected for your report before you run it. The members that you view result from an evaluated member list or function.

➤ To preview members:
1. Open a report, and double-click a cell.
2. In Select Members, click the Selected area.
3. Click View, and then select Preview.

Copying Members Between Applications

You can take any valid members from an application, such as an Excel spreadsheet, and paste them as new members in the corresponding dimension of another application. For example, suppose you have the following members in Excel for the Year dimension: Jan, Feb, Mar, Apr, May, Jun, July, Aug, Sep, Oct, Nov, Dec. You can copy those members to a grid that contains the Year dimension. If you paste members into a row or column that contains a secondary database connection, the type of database connection is carried over with the members.

➤ To copy members between applications:
1. Open an application containing the members you want copied to a report.
2. Copy the members to use with Financial Reporting.
4. Select File, and then Open.
5 Open a report.
6 Select a cell in a grid that contains the dimension to which you want to copy members.
7 In the formula bar, clear the content, right-click, and then select **Paste**.
8 Edit the text in the formula bar by adding commas between members.
9 Click ✔️ to verify the member names.

### Removing or Adding Members

You can remove or add members from a dimension. Removing members changes the output of your report, giving you more control over its content.

➤ To remove or add members that are contained in one cell:
1 Click the cell containing the members. The dimension name is displayed on the formula bar button; members are displayed in the text area of the formula bar.
2 In the formula bar, take an action:
   - Click the dimension name that is displayed on the command button to display the Select Members dialog box.
     - To remove one or more items from the **Selected** area, select the items, then click the **Remove** button.
     - To remove all items, click the **Remove All** button.
   - Edit the text area of the formula bar by removing or adding members. Separate members with commas. Then click ✔️ to verify the member names.

➤ To remove members that are in separate rows or columns:
1 Select the row or column.
2 Right-click the row or column, then select **Delete**.

### Selecting and Filtering Members Based on Criteria

**Subtopics**
- Selecting Multiple Members Based on Criteria
- Filtering the View of Members Based on Criteria

### Selecting Multiple Members Based on Criteria

You can select members dynamically based on user-defined criteria. You define criteria by creating expressions of members, Boolean operations, and commands.
Boolean operators enable you to specify precise member combinations for the report, which is useful for dealing with large volumes of data. Use the AND, OR, UNION, and NOT Boolean operators, combined with expression commands, to refine your member selections.

To select multiple members based on criteria:
1. Open a report.
2. Double-click a dimension cell for which you want to assign members.
3. In the Select Members dialog box, right-click inside the Selected area, then click Advanced.

**Note:** At least two members must be displayed in the Selected area before you can create criteria.

4. Build expressions by using one or more of the following operators and symbols:
   - Select NOT to add the Boolean operator NOT to the expression. NOT is the inverse of the selected condition.
   - Enter a left parenthesis, (, to add an opening character to the expression.
   - Enter a right parenthesis, ), to add a closing character to the expression.
   - Click in the OPERATOR column, then select AND, OR, or UNION.
     - When all conditions must be met, use the AND operator.
     - When one condition of several must be met, use the OR operator.
     - To combine the data, use the UNION operator.

**Note:** You must use two pairs of parentheses if you are using three members. For example, if you selected descendants of Market and you want to exclude East, West, and South, your advanced member selection query should be as follows: Descendants of Market AND NOT ( East AND NOT ( West AND NOT SOUTH ))

5. Click OK to return to your report.

**Filtering the View of Members Based on Criteria**

You define filters to view only members that meet your criteria. You can then select from the filtered list for your report. For example, you can display Regions in the United States division only. Filters are useful for reducing the number of members that are displayed.

To define filters:
1. Open a report.
2. Double-click a dimension cell for which you want to edit a member.
   - The Select members dialog box is displayed.
3. Right-click inside the Available area and make one of the following selections:
Defining and Editing Member Lists

Subtopics

- Defining Member Lists
- Editing Member Lists

Defining Member Lists

Member lists are user-defined saved queries or system-defined lists of members. System-defined lists are typically created by the database administrator. Member lists can include members, other member lists, and functions.

**Note:** User-defined lists can be created only with Essbase as a database connection. The size of a member list cannot exceed 4K.

To define a member list:

1. Open a report.
2. Double-click a dimension cell for which you want to assign a member.
3. In the Select Members dialog box, take an action:
   - In the Members tab, highlight the members to include in the member list, right-click inside the Available area, then click Save Highlighted.
   - In the Lists tab, right-click inside the Available area, then click Create.
4. From the Available area of the Create (dimension) List dialog box, select the members, member lists, and functions to include in the member list and move them to the Selected area using the Add button, or Add Relationship button.
5. Optional: To edit a function, use the procedure described in “Including a Member in a Function” on page 44.
6. Click OK.
7. Enter a name and description for the member list, then click OK.
**Note:** Do not use quotation marks (""") when entering a member list. If you do, Financial Reporting saves the name with the quotation marks, but you cannot select the name later.

8 Click **OK** to return to your report.

### Editing Member Lists

You can edit user-defined member lists. For example, you can add or remove members or functions from a member list.

➢ To edit a member list:

1 Open a report, and double-click a dimension cell.

2 In **Select Members**, select the **Lists** tab.

3 Double-click a member list.

4 In the **Edit...List** dialog box, do one of the following:
   - To add members to the **Selected** list, highlight the items in the **Members**, **Lists**, or **Functions** tabs, then click **Add**, **Add Relationship**, or **Add**.
   - To remove members from the list, highlight the members in the **Selected** area, then click **Remove**.
   - To remove all items, click **Remove All**.

5 Click **OK**.

6 Click **OK** to return to your report.

### Including a Member in a Function

You can edit a function by changing its parameter values. For example, you can specify that a function include or exclude a specified member, or you can change the hierarchy value of a function.

➢ To edit a function:

1 Open a report and select the data row or column or a data row or column heading cell in which you want to edit a function.

2 In the **Selected** area of the **Select Members** dialog box, double-click the function to edit.

3 Double-click the **Value** column inside row 1, then edit its value by clicking the **Lookup** button.

4 Repeat step 1 through step 3 to edit the remaining rows.
Access Privileges for Members in the Planning Database Source

If your grid uses Planning as a database connection, Financial Reporting gives you access privileges on members in certain dimensions. The dimensions include Accounts, Entities, Scenarios, and Versions. In the Select Member dialog box, all members for the Planning database connection are listed regardless of access privileges. Therefore, you can place members on a grid that are not returned when the report is run. The members you choose affect the output of the report. For example, if you select a member to which the user cannot access in the Point of View or Page Axis, an error message is returned.

For information on setting access privileges security for an end user, see Oracle Hyperion Enterprise Performance Management Workspace Administrator’s Guide.

The following four scenarios describe which values are returned on a grid that contains valid and invalid dimensions or members.

For the following scenarios, the user is reporting against a Planning database connection and has access privileges to the following dimensions and members:

- Accounts: Sales, Profit
- Versions: 1st Draft, 3rd Draft, Final Version
- Entities: North, Canada
- Scenarios: Budget

Scenario 1

The user runs a report for the following grid:

Page: Budget

<table>
<thead>
<tr>
<th></th>
<th>1st Draft</th>
<th>2nd Draft</th>
<th>3rd Draft</th>
</tr>
</thead>
<tbody>
<tr>
<td>North</td>
<td>#.#.#</td>
<td>#.#.#</td>
<td>#.#.#</td>
</tr>
<tr>
<td>Central</td>
<td>#.#.#</td>
<td>#.#.#</td>
<td>#.#.#</td>
</tr>
<tr>
<td>South</td>
<td>#.#.#</td>
<td>#.#.#</td>
<td>#.#.#</td>
</tr>
</tbody>
</table>

Result: Because the user does not have access to Central, South, and 2nd Draft, the report returns the following results:

Page: Budget

<table>
<thead>
<tr>
<th></th>
<th>1st Draft</th>
<th>3rd Draft</th>
</tr>
</thead>
<tbody>
<tr>
<td>North</td>
<td>23.89</td>
<td>12.90</td>
</tr>
</tbody>
</table>

Scenario 2

The user runs a report for the following grid:
Page: Actual

<table>
<thead>
<tr>
<th></th>
<th>1st Draft</th>
<th>2nd Draft</th>
<th>3rd Draft</th>
</tr>
</thead>
<tbody>
<tr>
<td>North</td>
<td>##.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
<tr>
<td>Central</td>
<td>#.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
<tr>
<td>South</td>
<td>#.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
</tbody>
</table>

**Result:** Because the user does not have access to Scenarios: Actual on the Page, no page is displayed and an error message describing your access privileges is returned.

**Scenario 3**

The user runs a report for the following grid:

Page: Budget

<table>
<thead>
<tr>
<th></th>
<th>1st Draft</th>
<th>2nd Draft</th>
<th>3rd Draft</th>
</tr>
</thead>
<tbody>
<tr>
<td>London</td>
<td>#.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
<tr>
<td>Paris</td>
<td>#.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
<tr>
<td>New York</td>
<td>#.##</td>
<td>#.##</td>
<td>#.##</td>
</tr>
</tbody>
</table>

**Result:** Because the user does not have access to the Entities on the rows, no page is displayed and an error message describing your access privileges is returned.

**Scenario 4**

The user runs a report for the following grid:

Page: Budget

<table>
<thead>
<tr>
<th>Descendants of Versions</th>
<th>1st Draft</th>
<th>2nd Draft</th>
<th>3rd Draft</th>
</tr>
</thead>
<tbody>
<tr>
<td>Descendents of Entities</td>
<td>#.##</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Result:** The report returns the following data:

Page: Budget

<table>
<thead>
<tr>
<th></th>
<th>1st Draft</th>
<th>3rd Draft</th>
<th>Final Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>North</td>
<td>23.89</td>
<td>12.90</td>
<td>67.12</td>
</tr>
<tr>
<td>Canada</td>
<td>2.67</td>
<td>8.90</td>
<td>54.78</td>
</tr>
</tbody>
</table>
This chapter describes the kinds of charts you can create and the ways in which you can format them.

**Note:** Reporting Studio has an updated charting engine for an overall improved charting presentation and user experience. To review the differences between the original charting engine and the updated charting engine, see the document titled *Oracle Hyperion Financial Reporting Charting*.

### About Charts

**Subtopics**

- Bar Charts
- Line Charts
- Combo Charts
- Pie Charts

A chart is a graphical representation of report data from a grid on a report. The chart retrieves data dynamically from the grid, so if data in the grid changes, the chart is updated automatically.

- Bar Charts
- Line Charts
- Pie Charts
- Combo Charts

You can define and format the following components of a chart:

- Chart type and data content
- Chart attributes such as title, title angle, font angle, and background formatting
Legend and axis information such as labels, font, background color, and orientation

Formatting options specific to a chart type, such as bar width for a bar chart

Horizontal and vertical positioning of a chart on a report

**Bar Charts**

A bar chart displays data sets, with any number of values assigned to each data set. Properties for the following items are supported for the bar chart:

- Appearance
- Legend
- Axes
- Element Style
- Bar

You can display data sets by grouping corresponding bars or stacking them, or you can display them in separate rows. You can show bars with or without a 3D effect, applying one of the following bar shapes: bar, triangle bar, diamond bar, or cylinder.

**Line Charts**

A line chart displays data sets, with automatic, uniform spacing along the X axis. Properties for the following items are supported for the line chart:

- Appearance
- Legend
- Axes
- Element Style
- Line

**Combo Charts**

A combo chart is a combination bar chart and line chart. Combo charts are typically used for comparison charting. The combo chart accepts a group of bar data sets and line data sets. It uses all of the display parameters found in bar charts and line charts. Two data set selections are required: one for the bar rendering and one for the line rendering. Properties for the following items are supported for the combo chart:

- Appearance
- Legend
- Axes
- Element Style
Pie Charts

A pie chart shows one data set segmented in slices of a pie. Properties for the following items are supported for the pie chart:

- Appearance
- Legend
- Element Style
- Pie

Formatting Charts

Subtopics

- Defining Chart Data
- Using the Format Chart Dialog Box
- Formatting Fonts in Charts
- Formatting Chart Border and Background Options
- Formatting Bars in Bar and Combo Charts
- Formatting Lines in Line and Combo Charts
- Defining Combo Chart Options
- Formatting Pie Charts
- Setting Background Options for a Chart
- Positioning Charts on Reports
- Defining Gridlines
- Defining Styles for Data Sets
- Resizing and Moving Charts

You format charts to customize their appearance and layout in your reports. You can also format charts to portray data using different chart types, including bar charts or combo charts. You make formatting changes to charts in the Format Chart dialog box. You access this dialog box from the Chart Properties property sheet.

Defining Chart Data

You define chart data using the Chart Properties property sheet. You can identify, from the controlling grid, the rows and columns that contain the data to display. You can reference data or formula rows and columns from the controlling grid by selecting contiguous or non contiguous rows and columns for the data range. You can also include or exclude auto calculations from the data set.
If you are designing a line, bar, or combo chart, you can reference multiple rows or columns from the grid. If you are designing a pie chart, you can reference any single row or column from the grid. When you define a combo chart, you specify the data rows and columns for the bar and line.

Charts display aggregate rows or aggregate columns that expand. For example, if you specify row 1 and row 1 contains a function that retrieves 10 child members, the chart displays 10 data sets, or one data set for each child member.

To define chart data:

1. Open a report and create a grid and a chart.
2. From the Grid drop-down list, select the name of the grid for this chart to reference. The grid must exist on the current report.
3. For Legend Items from Grid, select Rows to reference row data or Columns to reference column data.
4. Specify the range of data to display in the chart for all chart types (bar, line, combo, or pie). In the Data Range list boxes, select the rows to include from the Rows list of check boxes and the columns to include from the Columns list of check boxes.

   **Note:** You can specify contiguous or non contiguous rows and columns when selecting a data range.

5. Optional: To include auto calculations from the data set, select the Include Auto Calculation check box.
6. Optional: To insert a page break before the chart, select the Page Break Before check box.
7. Click the Format Chart button to apply formatting to the chart. See “Using the Format Chart Dialog Box” on page 50.
8. Optional: Modify the horizontal and vertical positions for the object using the drop-down list boxes for Position Object on Report. For more information, see “Positioning Charts on Reports” on page 55.
9. To add related content, select the Add Related Content check box, then click Setup. See “Applying Related Content” on page 128.
10. In the Name field, enter a name for the chart.
11. To view the chart with data, select File, and then Print Preview or Web Preview.

### Using the Format Chart Dialog Box

You make formatting changes to charts by clicking the Format Chart button in the Chart Properties sheet. The tabs that are displayed on Format Chart depends on the type of chart you select from the Chart Properties property sheet. For example, if you select a pie chart, the dialog box includes a Pie Options tab but not a Bar Options tab.

**Note:** Depending on the type of chart you are using, certain controls on the Format Chart dialog box may be disabled.
<table>
<thead>
<tr>
<th>Tab</th>
<th>Description</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appearance</td>
<td>You use the Appearance tab to specify a title for the chart, a chart border, font, gridlines, gridline style, gridline color, and grid background color for all chart types. You can indicate a depth for the chart grid. The Appearance tab also provides a preview of your chart.</td>
<td></td>
</tr>
<tr>
<td>Legend</td>
<td>You use the Legend tab to specify legend properties for all chart types. You can specify the location of the legend on the chart and define the legend title. You can set the font, font style, font size, and text effects for the legend title. The default setting for suppress repeating labels is off, but you can enable the option to inherit settings from the grid. For more information, see “Defining Chart Legend Titles” on page 58.</td>
<td></td>
</tr>
<tr>
<td>Axes</td>
<td>You use the Axes tab to specify axis properties, including metadata, primary, and secondary axis titles and the axis labels. You can indicate primary and secondary axis labels and specify font options for both. You can also specify data labels for the primary and secondary axis using a default or custom range of values. You can specify font options, such as font, font style, font size, and text effects. <strong>Note:</strong> You can provide a secondary axis title label when working with combo charts. These titles and labels display a value between zero and 100, representing a percentage. If you are working with combo charts, you can show data labels for the secondary axis using a default or custom range of values. You can show metadata labels and specify a font angle for display of the labels. You can also modify other font options for metadata labels. For more information, see “Defining Chart Axis Titles” on page 59.</td>
<td></td>
</tr>
<tr>
<td>Element Style</td>
<td>You use the Element Style tab to specify attributes such as bar and pie fill, line style, line marker style, marker size, and line width for data sets, depending on the chart type. For example, you specify pie fill only for a pie chart. You select each data set and specify a pattern, style, color, and shape to apply to it.</td>
<td></td>
</tr>
<tr>
<td>Line Options</td>
<td>When formatting a line chart, you use the options on the Line Options tab to specify the type of line (rows, percent, or stacked-label display). You can also show point values to display the values in your chart and set font options for these values. The Line Options tab provides a preview of your line chart as you modify the formatting.</td>
<td></td>
</tr>
<tr>
<td>Bar Options</td>
<td>When formatting a bar chart, you use the Bar Options tab to specify the type of bar (group or stack) and the orientation (vertical or horizontal) of the bar chart. You can specify a bar width (zero to 100) for the bars in the chart. You can select a bar border style and width (zero to 20). You can show point values to display the data values in your chart and set font options for these values. The Bar Options tab provides a preview of your bar chart as you modify the formatting.</td>
<td></td>
</tr>
<tr>
<td>Line Options and Bar Options</td>
<td>When formatting a combo chart, the Format Chart dialog box includes a Line Options tab and a Bar Options tab. Consult the Line Options tab and Bar Options tab descriptions in this table for information about the tabs. <strong>Note:</strong> When you are formatting a combo chart, the Chart Properties property sheet displays range selection options for data on the line and bar charts.</td>
<td></td>
</tr>
<tr>
<td>Pie Options</td>
<td>You use the Pie Options tab to specify formatting options for a pie chart. You can modify the angle of the pie chart, specify a separation distance (which applies to the first slice of the pie chart only), and select slice values (percent, data, or label) to display in the pie chart. You can also format the font options for these values. You can indicate the position (exterior or radial) for the slice values. You can specify a solid slice border style and a border width (zero to 20) for the slice. The Pie Options tab provides a preview of your pie chart as you modify the formatting options.</td>
<td></td>
</tr>
</tbody>
</table>

### Formatting Fonts in Charts

When working with charts, you can format fonts to customize their appearance. Font formatting options include font selection, font size, color, and font angle (zero to 315 degrees, in increments of 45 degrees).
You can format fonts by clicking the Format button, which is displayed throughout the tabs located on the Format Chart dialog box.

The Format dialog box enables you to preview your font option selections before returning to the chart. For more information on formatting fonts, see “Modifying Font Options” on page 148.

### Formatting Chart Border and Background Options

You can format the chart border and background using the Borders and Background dialog box. You access this dialog box by clicking the Borders and Background icon, which is displayed throughout the Format Chart dialog box.

Borders and background format options include border type, border color, background color, and border width.

### Formatting Bars in Bar and Combo Charts

You can display bars in a bar or combo chart by grouping data sets and specifying their orientation on the chart. You can also display point values in your bar or combo chart. You can modify the widths of bars and bar borders.

Use the following guidelines to specify how bars are displayed in your charts:

- Vertical orientation starts the bars on the horizontal axis and displays the bars vertically.
- Horizontal orientation starts the bars on the vertical axis and displays the bars horizontally.
- The bar width represents the relative width of the bar as a percentage of available space. A smaller number results in more space between bars.
- You can apply shading (pattern and color) by data set using the Element Style tab of the Format Chart dialog box. For more information, see Table 4 on page 51.

To format bars in bar and combo charts:

1. Open a report and create or select a bar or combo chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Select the Bar Options tab.
4. Select a Type option:
   - To display bars side-by-side, select Group.
   - To display bars stacked vertically bottom-to-top, select Stack.
5. Select an Orientation option:
   - To display the bars vertically on the horizontal axis, select Vertical.
   - To display the bars horizontally on the vertical axis, select Horizontal.
6. To show the numerical value or data that the bar represents, select the Show Point Values check box. The value is displayed within the bar.
Click the Font button to set font options for the bar values.

To specify a style to be applied to the border of the bars in the chart, select Solid, Dotted, or Dashed from the Bar Border Style drop-down list.

To alter the width of the bars, use the slide tool in the Bar Width box.

To increase or decrease the width of the bar borders, use the slide tool in the Bar Border Width box.

Click OK.

Note: You can define styles for the bars representing the current data set. For more information, see “Defining Styles for Data Sets” on page 57.

Formatting Lines in Line and Combo Charts

You can specify display options for lines in line and combo charts. For example, you can display lines in separate rows, stacked as a percentage of the total of all values, or stacked as absolute values.

To format lines in line and combo charts:

1 Open a report and create or select a line or combo chart.

2 On the Chart Properties property sheet, click the Format Chart button.

3 Select the Line Options tab.

4 Select one of the following line types:
   - To display the line sets in separate rows, select Rows. For example, if you define the 3D depth as nonzero, the lines are displayed in separate rows, from front to back.
   - To display the line sets stacked on top of each other, normalized to 100%, select Percent. As each line set is drawn, its values are added to previous values displayed and are shown as percentages of the total of all values.
   - To display the line sets stacked, select Stack. As each line set is drawn, its values are added to previous values. Select one of the following from the drop-down list:
     - To show pop-ups with accumulated values, select Total Value.
     - To show pop-ups as individual item values, select Segment Value.

5 To show the numerical value or data that the line represents, select the Show Point Values check box. The value is displayed on top of the line.

6 To set font options for the point values of the lines, click the Font button.

7 Click OK.

Note: You can define styles for the lines representing the current data set. For more information, see “Defining Styles for Data Sets” on page 57.
Defining Combo Chart Options

A combo chart combines a line chart and a bar chart. Typically used to contrast data sets, combo charts employ bar and line options and are set with the same variables as bar charts and line charts.

When you are formatting a combo chart, the Format Chart dialog box includes the Line Options tab and Bar Options tab. For more information, see Table 4 on page 51. For information about defining bar options for a combo chart, see “Formatting Bars in Bar and Combo Charts” on page 52. For information about defining line options for a combo chart, see “Formatting Lines in Line and Combo Charts” on page 53.

Formatting Pie Charts

When creating a pie chart, you can specify the starting angle for the first pie slice, the level of separation between pie slices, the position and type of pie slice labels, and the slice border style and width. You can also set font options including font, font size, effects, and angle for the textual display of the slice values.

To format a pie chart:

1. Open a report and create or select a pie chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Click the Pie Options tab.
4. To specify the angle of the first pie slice, use the slide tool for Pie Angle.
5. To indicate the distance between the pie slices, use the slide tool for Separation. Moving the slide tool to the left makes the space between slices smaller; moving the slide tool to the right increases the space between slices.
6. Select a Show Slice Values as option:
   - To show data values as a percentage, select Percent.
   - To show the absolute value of the pie chart data, select Data.
   - To display the data value as a label, select Label.
7. To set font options for the slice values shown in the chart, click the Font button.
8. Select a Position option:
   - To show the slice values outside of the pie chart, select Exterior.
   - To show the slice values inside of the pie chart, select Radial.
9. From the Slice Border Style drop down list, select a solid to apply a solid line to the border of the slices in your pie chart.
10. To indicate the width of the borders for the pie chart slices, use the slide tool for Slice Border Width. The maximum width is 100.
11. Click OK.
**Setting Background Options for a Chart**

You can set background options for a chart, such as a border, gridlines, gridline color, gridline style, and grid background. You can also set the grid depth.

▶ To set background options for a chart:

1. Open a report and select a chart.
2. From the Chart Properties property sheet, click the **Format Chart** button.
3. Select the **Appearance** tab.
4. Click the **Borders and Background** icon to format the border and background of the chart.
5. To modify the gridlines, gridline color, style, or background, select an option from the drop-down list boxes:
   - To specify which gridlines to consider for the current format (both, vertical, horizontal or none), select an option from the **Gridlines** drop-down list.
   - To define a style (solid, dotted, dashed, dotdash, or bar) for the gridlines, select an option from the **Gridline Style** drop-down list.
   - To define a color for the gridlines, select an option from the **Gridline Color** drop-down color palette.
   - To define a background color to display behind the gridlines, select an option from the **Grid Background** drop-down color palette.

   **Note:** When setting background options for a pie chart, these options are not available.

6. Click **OK**.

**Positioning Charts on Reports**

You can specify a chart’s position on the report and how it is displayed when printed. You can specify a position relative to the grid, a position relative to other report objects, or no position at all. If you do not specify a position, the chart remains in the location where it was originally created on the report.

▶ To position a chart on a report:

1. Open a report and select a chart.
2. On the Chart Properties property sheet, modify the horizontal and vertical positions for the object using **Position Object on Report**:
   a. From the **Horizontal** drop-down list, select one of the following horizontal positions:
• Select None to anchor this chart at its current position in the report. The chart is printed in this position.
• Select Relative for the chart to maintain a distance relative to other report objects located to the left of it. This option assures that the chart is not printed on top of another report object.
• Select Left to position the chart against the left margin of the report.
• Select Center to position the chart in the center of the report.
• Select Right to position the chart against the right margin of the report.

b. In the Vertical drop-down list, select one of the following vertical positions:
• Select None to anchor this chart at its current position in the report. The chart is printed in this position.
• Select Relative for the chart to maintain a distance relative to other report objects located above it. This option assures that the chart is not printed on top of another report object.
• Select Top to position the chart at the top of the report.
• Select Middle to position the chart in the middle of the report.
• Select Bottom to position the chart at the bottom of the report.

**Defining Gridlines**

You can display gridlines behind a bar, line, or combo chart. You can show or hide gridlines, and you can specify gridline color, style, and background color.

➢ To define gridlines:

1. Open a report and select a chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Select the Appearance tab.
4. From the Gridlines drop-down list, select one of the following options:
   • Horizontal to show horizontal gridlines
   • Vertical to show vertical gridlines
   • Both to display horizontal and vertical gridlines
   • None to hide gridlines
5. Optional: To specify a style (solid, dotted, dashed, dotdash, or bar) for the gridlines, click Gridline Style and select an option from the drop-down list.
6. Optional: To specify a color for the gridlines, click Gridline Color and select a color from the drop-down color palette.
7. Optional: To specify a background color to display behind the gridlines, click Grid Background and select a color from the drop-down color palette.
Defining Styles for Data Sets

You can define styles for data sets in charts. You can define a pattern and color for each data set in your report. The specified shading and patterns are applied to each respective data set to differentiate values in the chart.

To define styles for a data set:
1. Open a report and select a chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Select the Element Style tab.
4. Select a data set.
5. Select a fill pattern for the selected data set.
6. From the Color 1 and Color 2 drop-down list boxes, select two colors for contrasting the pattern.
7. Select a line style, line marker style, marker size, and line width to apply to the data set.
8. Click OK.

Resizing and Moving Charts

You can change the size of a chart and move a chart to another location in a report.

To resize a chart:
1. Select a chart by clicking on that chart object.
2. Click and drag a handle, located on the perimeter of the chart, to change the size.

To move a chart:
1. Click inside the chart, selecting the chart as an object.
2. Hold down the left mouse button and drag the chart to another location in the report.

Formatting Chart Titles and Labels

Subtopics

- Defining Chart Legend Titles
- Defining Chart Titles
- Defining Chart Axis Titles
- Defining Axis Labels
- Viewing Charts

You can specify and format chart legend titles, chart titles, chart axes titles, and axis labels in your chart. You can also format the font and borders and background of the titles and axes you indicate. You specify chart titles and labels in the Format Chart dialog box.
Defining Chart Legend Titles

You define a legend to identify and describe the data sets used in your chart. You can specify font and background options and border color. You can also specify the placement location for the legend within the chart boundaries.

You can also suppress repeating labels, through which you can inherit settings from the referenced grid. Repeating labels are axis labels that repeat in a chart if you do not suppress them.

**Note:** Chart legend options are available for all chart types.

To define chart legend options:
1. Open a report and select a chart.
2. On the Chart Properties property sheet, click the **Format Chart** button.
3. Select the **Legend** tab to display the formatting and placement options for legends.
4. Select the **Place Legend** check box to select the positioning of the chart legend relative to the chart boundaries. You can select: Top, Bottom, Left or Right.
5. Click the **Borders and Background** icon to format the border and background of the legend.
6. In the **Legend Title** text box, enter a title for the legend.
7. Click the **Font** button to set font options, including font, font size, effects, and angle, for the legend title and its labels, then click **OK**.
8. To suppress X-axis labels that would otherwise repeat, select from the following options for **Suppress Repeating Labels**: Inherit Settings from Grid, On, or Off.
9. Click **OK**.

Defining Chart Titles

When defining chart appearance, you can specify a title and then select font and background options for the title.

To define a chart title:
1. Open a report and select a chart.
2. From the Chart Properties property sheet, click the **Format Chart** button.
3. Select the **Appearance** tab.
4. In the **Chart Title** text box, enter a title for the chart.

**Tip:** When defining legend, axes or chart titles, you can print the title on multiple lines using a `\n` in the title. You can use a `\' s` to indicate a possessive apostrophe in the legend, axes or chart title. You can combine these commands to use apostrophes with titles that span multiple lines.
Defining Chart Axis Titles

You can specify a metadata title and primary and secondary axis titles, to identify the data that is displayed on the axes of your chart. You can also specify font options, including font size, color, and font angle. You can label your metadata using the same formatting options.

You can also specify font options, including font size, color, and font angle. You can label your metadata using the same formatting options.

You can specify color, border, and background options for the title of each axis.

Note: Chart axis options are available for bar, line, and combo charts only.

To define a chart axis title:

1. Open a report and select a chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Select the Axes tab to display the formatting options for axes, then perform one or more of the following actions:
   - In the Metadata Title text box, enter the title for the horizontal axis.
   - In the Primary Axis Title text box, enter the title for the primary vertical axis. This is displayed to the left of the chart.
   - In the Secondary Axis Title text box, enter the title for the secondary vertical axis. This is displayed to the right of the chart.
4. Click the Font button to set font options and then click OK.
5. Click the Borders and Background icon then specify border and background options for the axis title and click OK.
6. Click OK.

Defining Axis Labels

You can define and format labels for the primary and secondary axis in a chart. For a combo chart, you can specify and format a secondary axis label. You can also display and format metadata labels.
To define axis labels:

1. Open a report and select a chart.
2. On the Chart Properties property sheet, click the Format Chart button.
3. Select the Axes tab and perform one or more of the following actions:
   - To use the default range of values on the primary axis, select the check box for Primary Axis Label.
   - To enter a range of values, including minimum, maximum, and step, on the primary axis, select the check box for Primary Axis Label and the Custom Range check box.
     
     **Note:** You can use the custom range to change the starting and ending numbers of your axis and the numeric intervals. For example, you can display values 40 through 60 at three step intervals by entering 40, 60, and 3 in the Min, Max, and Step check boxes, consecutively.
   - To use the default range of values on the secondary axis of a combo chart, select the check box for Secondary Axis Label.
     
     **Note:** Secondary Axis only applies to combo chart types.
   - To display the percentage of the default range of values on the secondary axis in a bar or line chart, select the check box for Secondary Axis Label.
   - To enter a range of values, including minimum, maximum, and step, on the secondary axis in a bar or line chart, select the check box for Secondary Axis Label and the Custom Range check box.
4. Click the Font button to set font options for the data labels, and then click OK.
5. Click the Borders and Background icon, then specify border and background options for the data labels, and then click OK.
6. To show the metadata labels with a standard display, select the Show Metadata Labels check box.
   
   **Note:** Financial Reporting implements staggered labels as a default to ensure the even distribution of data on report axes.
7. Click the Font button to set font options for the metadata labels, then click OK.

**Viewing Charts**

You can print and print-preview the chart rendering in Reporting Studio and Financial Reporting.
To build reports, you create a report design by adding report objects to the design and defining the report object properties. Report objects include text boxes, grids, images, and charts. When you create a report, you start with a blank design area. You can modify the blank report as necessary.

### Adding Report Objects

Report objects are defined as text boxes, grids, charts, or images defined in the design area of a report. Adding report objects to your report involves specifying the report layout and object properties, and selecting the data content. You can also insert report objects previously stored in the repository. Report objects can be used in multiple reports.
After you add a report object, you can assign properties to it such as format options. Format options include font, alignment, spacing, shading, and positioning.

To add a report object:

1. Select File, then New, and then Report.
2. Add report objects to your report.
3. Save your report by selecting File, and then Save.

Using Linked and Local Objects

Subtopics

- Linked Objects
- Local Objects in Financial Reporting

Report objects can be saved to the repository and serve as a resource for other reports. They can be inserted into other reports as linked or local (unlinked) objects and used “as is” or customized.

Linking objects to a source object can save time and effort. By updating one source object, you update all reports that are linked to this source object.

The option to link or not to link is specified while performing the following functions:

- Saving a report object to the repository
- Inserting an object from the repository into your report. When you insert a saved object from the repository as a linked object, you can clear the link after it is copied into the report. However, you cannot reverse this action.

Linked Objects

Linked objects enable you to save an object in a report and in the repository. You can link the object in the report to the source object in the repository so that modifications to the source object are reflected in the object that exists in the report.

When you insert a source object into a report, modify it, then link it to the source object when saving the object, the changes are reflected in the source object and all objects that are linked to it in other reports. For an example of creating a linked report object, see “Saving Text Objects” on page 65.

For example, suppose you insert into your report a text object, Company_Name, from the repository. You then modify the text in Company_Name and save the object, specifying a link to the source object. The revised Company_Name text object overwrites the source object in the repository, and any new report or existing report that is also linked to the Company_Name source object reflects the modified text.
The following table outlines the objects that can be linked to the repository and included in your reports, and provides the location of procedural information for each object type in Financial Reporting.

**Table 5 Using Linked Objects**

<table>
<thead>
<tr>
<th>Object</th>
<th>Link to Source Object Procedural Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grid</td>
<td>You can link to source object via the Save Object dialog box. For more information, see “Saving Grid Objects” on page 69.</td>
</tr>
<tr>
<td>Row and column template</td>
<td>You can link to source object via the Save Object dialog box. Once linked the Link to Source Object is displayed on the Row and Column Template Properties property sheet.</td>
</tr>
<tr>
<td>Chart</td>
<td>You can link to source object via the Save Object dialog box. For more information, see “Inserting Charts” on page 71.</td>
</tr>
<tr>
<td>Text box</td>
<td>You can link to source object via the Save Object dialog box. For more information, see “Saving Text Objects” on page 65.</td>
</tr>
<tr>
<td>Image</td>
<td>You can link to source object via the Save Object dialog box. For more information, see “Saving Image Objects” on page 71.</td>
</tr>
</tbody>
</table>

**Local Objects in Financial Reporting**

When you save a repository object to a report without linking it to the source object, the local object (the object on the current report) is not linked to the saved object in the repository. Changes to the local object do not affect the object saved to the repository, and changes to the repository object do not affect the local object in the report.

This option enables you to customize your report’s copy of the report object to suit your individual requirements.

The following figure illustrates the behavior of linked and local objects in a report.
In Report A, Report Object 1 is copied from the repository as a local object. Changes made to the local report object are not reflected in the source object. Report Object 3 is linked to its source object in the repository. Changes made to Report Object 3, when saved, are reflected in the source object and other reports that are linked to that source object, including Report Object 3 in Report B.

### Adding Text Objects

Text objects are report objects that can contain text or text functions. For example, you can enter a label, description, or a function that retrieves the current date in a text box. For more information about text functions, see “Text Functions” on page 194. You can enter multiple paragraphs of text. Like other report objects, you can move, resize, or reposition the area containing the text.

You can create a new text object or insert an existing text object into the report. New text objects are empty areas where you can enter text, text functions, or rich text from an editor or word processor. Text objects are stored in the repository and can be inserted into numerous reports. You might want to add the same text object, for example, in all your report designs.

➢ To add a text object:

1. Select File, and then Open.
2. Open a report and select Insert, and then Text.
3. Point to where you want to position the upper left corner of the text box, click and drag to the desired size, then release the mouse button.
4. Click inside the text object, and enter the text directly in the text box.
To insert a text box from the repository:

1. Select **File**, and then **Open**.
2. Open a report and select **Insert**, and then **Saved Object**.
3. In the **Type** drop-down list, select **Text**.
4. Select the directory where you want to search for the text object, then select the desired text object.
5. Perform an action:
   - To use the text object as a linked object, select **Link to Source Object**.
   - To use the text object as an unlinked object, deselect **Link to Source Object**. This is the default option.
6. Click **Insert**.

### Modifying Text Objects

You can modify the properties of a text object. This includes inserting a function in the text object, modifying the vertical or horizontal position of the text box, and selecting the autosize function, which enables the display of all of the text contained in the text object. You can also select the Page Break Before check box, which displays the text box on the top of the next page when you preview or print the report. You can also link the current text object to a source object in the repository.

**Note:** The Link to Source Object check box is enabled after the text object is saved to the repository and the Link to Source Object check box is selected.

To modify a text object:

1. Select the text object.
2. Using the **Text Properties** sheet, modify the properties of the object. To modify the text, select **Format**, and then **Font**.

   **Tip:** To resize the text object, drag a handle. To move the text object, drag inside the text object border.

### Saving Text Objects

When you save a text object, you can create a link between that text object and the source object in the repository. Text objects can be saved to the repository for reuse when building reports. Once they are saved to the repository, they can be inserted into a report as linked or unlinked objects.

**Note:** Changes to a linked object are not saved when you save and close your design.
To save the text object to the repository:

1. Click the text object, then select File, and then Save Object.
2. Select a folder or create a folder.
3. Perform the following actions:
   a. In the Name text box, enter the report text object name. You can use upper and lowercase letters, spaces and numbers. The program does not accept special characters such as \ / %, ?, +, >, |, “,” *.
   b. In the Description text box, enter the description of the text object.
4. Perform an action:
   - To create a link from the text object in the report to the source object in the repository, select Link to Source Object.
   - To save without creating a link from the text object in the report to the saved object in the repository, clear Link to Source Object.

   **Note:** For a description of linked objects, see “Using Linked and Local Objects” on page 62.
5. Click Save to save the text object to the repository. When you save a text object, only the specified object is saved, not the entire report design.

### Adding Grids

**Subtopics**
- About Grids
- Defining the Dimension Layout for a Grid
- Adding a New Grid
- Inserting an Existing Grid
- Selecting a Grid
- Modifying the Properties of Grid Objects
- Saving Grid Objects

### About Grids

Grids are tables that can contain data from external database connections. You can add a grid to your report, define its dimension layout, select members, and then format the grid. You can also insert an existing grid to reuse a pre-formatted grid design. You can use text, dimensions, members, and formulas to define the grid content. You can also hide a grid. For example, you might want to display a chart in a report, but not the grid associated with the chart.

**Note:** In Microsoft Excel, reports containing two grids that are side by side do not display the grids side by side; instead, the grids are displayed one above the other.
Defining the Dimension Layout for a Grid

You define the dimension layout for the grid by specifying a database connection, then dragging the available dimensions from the Point of View (POV) frame into the Rows, Columns, and Pages frames. The dimensions that are not placed in the Rows, Columns, or Pages frames are displayed in the user POV bar for that grid. You can layer multiple dimensions on a row, column, or page. For example, you might want to place the Scenario and Time dimensions on the same axis to show actual and budget data over a period of time.

**Note:** If the selected database connection for the grid contains attribute dimensions, they are displayed above the Point of View frame. Like the dimensions, you can drag attribute dimensions into the Point of View frame or the Rows, Columns, or Pages frames.

### Table 6  Dimension Layout Options

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grid</td>
<td>Grid name. This option is available only if the grid is not linked to a source grid in the repository.</td>
</tr>
<tr>
<td>Attribute Dimensions</td>
<td>Type of dimension that enables analysis and is based on the attributes or qualities of the members of its base dimension. You can drag attribute dimensions from this frame into the Point of View, Pages, Rows, or Columns frames. Note: Attribute dimensions are supported only for Essbase.</td>
</tr>
<tr>
<td>Point of View</td>
<td>Dimensions that define the grid’s point of view. The point of view identifies the data to retrieve in the grid. You can drag dimensions from this frame to the Pages, Rows, or Columns frames.</td>
</tr>
<tr>
<td>Pages</td>
<td>Third axis on a grid. You can drag into this frame the dimensions to use on the Page axis from the Attribute Dimension and Point of View frames.</td>
</tr>
<tr>
<td>Columns</td>
<td>Vertical display of information in a grid or table. You can drag into this frame the dimensions to use on the column.</td>
</tr>
<tr>
<td>Rows</td>
<td>Horizontal display of information in a grid or table. You can drag into this frame the dimensions to use on the row axis from the Attribute Dimension and Point of View frames.</td>
</tr>
</tbody>
</table>

After you define the dimension layout, you can select the members to use for each dimension. See “Assigning Members to Data Rows or Columns” on page 28 and “Assigning Members to the Page Axis” on page 29.
Adding a New Grid

To add a new grid to your report:

1. Open a report and select Insert, and then Grid.
2. Point to where you want to place the upper left corner of the grid, and then drag to the desired size.
3. In Select a Database Connection, select a database connection, and then click OK.
4. In the Dimension Layout window, drag the dimension icons from the Attribute Dimensions frames, if they exist, and from Point of View frames into the Pages, Rows, or Columns frames, and then click OK.

Note: You can place multiple dimensions into the Rows, Columns, or Pages frames.

When you add a new grid, the system assigns a default name to the grid. You can rename the grid in the Dimension Layout dialog box or when you save it. The default name assigned is Grid$n$, where $n$ is a system-assigned identification number. For example, if you save a grid that is the sixth system-named grid saved in the report, the default name is Grid6.

The system uses the grid name when a function or chart references the grid. For example, if you design a chart to graphically display the data from a grid, the chart properties must reference the grid by its name.

Inserting an Existing Grid

To insert an existing grid in your report:

1. Open a report and select Insert, and then Saved Object.
2. In Type, select Grid.
3. Select the directory where you want to search for the grid object, then select the desired grid object.
   - To use the grid object as a linked object, select Link to Source Object.
   - To use the grid object as an unlinked object, clear Link to Source Object.

   see “Using Linked and Local Objects” on page 62.
4. Click Insert.

Selecting a Grid

To select a grid, click the top left gray cell to select the row or column inside the grid object.

When you select a grid, you can right-click to access the following shortcut menu options:

- **Dimension Layout**—Change the dimensions for the pages, rows, and columns of a grid.
- **Grid Point of View**—Display or hide the Grid Point of View bar.
- **Grid Point of View Setup**—Select members for the Grid Point of View.
Data Query Optimization Settings—Suppress the rows on the database connection server or the report server, include or exclude Essbase sparse command, and set the number of asymmetric columns to consolidate. (See “Setting Data Query Optimization Options” on page 94.)

Modifying the Properties of Grid Objects

You can customize grid objects using the Grid Properties sheet, which contains three primary categories for modifying a selected grid: General, Suppression, and Position. You can use General to define heading preferences and grid options. Suppression is used to hide a grid, or values in a grid’s cells and Position is used to modify the object’s vertical or horizontal positioning on the grid. The following procedure describes how to modify the properties of a grid object.

To modify the properties of a grid object:

1. Open a report and select the object to be modified.
2. Modify the object using the Grid Properties sheet.
3. Optional: Click the Dimension Layout button in the Grid Properties sheet to change the dimensions on the grid’s axes. If this is a linked grid, the Grid field displays the name of the grid and its path in the repository.

Tip: To resize the grid object, drag a handle. To move the grid object, drag the grid’s title bar.

Saving Grid Objects

When you save a grid object, you can create a link between that grid object and the source object in the repository. When you save the object in the repository, it can be reused by linking to another report.

Note: You must be assigned a Global Administrator or Report Designer role to perform this function.

To save the grid object to the repository:

1. Click the grid, and do one of the following:
   - Select File, and then Save Object
   - Right-click, and then select Save Object.
2. Select a folder in which to store the grid, or create a new folder.
3. In the Name text box, enter the grid name. You can use upper and lowercase letters, spaces and numbers. The program does not accept special characters such as \ / %, ?, + < > | ' “ *.
4. In the Description text box, enter the description of the grid.
5. Perform an action:
To create a link from the grid object in the report to the source object in the repository, select **Link to Source Object**.

To save a grid object without creating a link from the grid object in the report to the saved source object in the repository, clear **Link to Source Object**.

**Note:** For a description of linked objects, see “Using Linked and Local Objects” on page 62.

6 Click **Save** to save the grid object to the repository. When you save a grid object, only the specified object is saved, not the entire report design.

## Inserting Images

Image objects are graphics or pictures that can be used in your report. The following image formats can be used:

- Bitmaps: .bmp and .dib
- Graphics Interchange Format (GIF): .gif
- Joint Photographic Experts Group (JPEG): .jpg
- Icon: .ico and .cur

You can add images to the header, footer, or body of a report. After you add an image frame with an image file to the report, you can format the image object. For more information on image formatting, see “Formatting Images” on page 170.

You can insert an image file that is stored in a file system outside Financial Reporting into the image frame, or you can insert an entire image object that is saved in Studio Explore.

**Note:** Image objects are saved with the report. As a result, the speed of the application may be affected based on the size of your image object.

To insert a new image object located outside of Studio Explore:

1 Open a report and select **Insert**, and then **Image**.

2 Point to where you want to place the upper left corner of the image frame, click and drag to the desired size, then release the mouse button.

3 In the **Files of type** drop-down list, select an image format.

4 Select an image file from the default Financial Reporting bin directory, or click the **Look in** drop-down list to search for an image file located on your file system.

5 Click **Open** to select the image file.

6 Use the Image Properties sheet to modify the image.

**Note:** To add the image to the repository, or to save it as a linked object, select File then, Save Object. For information, see Saving Image Objects.
To insert an image object from the repository:

1. Open a report and select **Insert**, and then **Saved Object**.
2. In the **Type** drop-down list, select **Image**.
3. Select the desired image object.
4. Perform an action:
   - To use the image object as a linked object, select **Link to Source Object**.
   - To use the image object as an unlinked object, clear **Link to Source Object**. This is the default.

   **Note:** For a description of linked objects, see “Using Linked and Local Objects” on page 62.
5. Click **Insert**.

**Saving Image Objects**

When you save an image object, you can create a link between that image object and the source object in the repository. Linking report objects to a source object saves time and effort. By updating one source object, you update all reports that are linked to this source object.

To save the image object to the repository:

1. Select the image, then select **File**, and then **Save Object**.
2. Specify a name and a location to where you want to save the image.
3. Perform an action:
   - To create a link from the image object in the report design to the source object in the repository, select **Link to Source Object**.
   - To save without creating a link from the image object in the report design to the saved object in the repository, clear **Link to Source Object**.
4. Click **Save** to save the image object to the repository. When you save an image object in a report, only the specified object is saved, not the entire report design.

**Inserting Charts**

Charts enable you to display data from a grid on the report. When you insert a chart in a report, the chart retrieves data from the specified grid on the current report. You can insert bar, pie, line, and combo (bar and line) charts.

For descriptions of each type of chart, see “About Charts” on page 47. For information on designing charts, see Chapter 3, “Customizing Charts.”
To insert a new chart object:

1. Open a report and select Insert, and then Chart.
2. Point to where you want to place the upper left corner of the chart, click and drag to the desired size, then release the mouse button.
3. Modify the chart using the Chart Properties sheet.

To insert a chart object from the repository:

1. Open a report and select Insert, and then Saved Object.
2. In the Type drop-down list, select Chart.
3. Select the desired chart object.
4. Perform an action:
   - To use the chart object as a linked object, select Link to Source Object.
   - To use the chart object as an unlinked object, clear Link to Source Object. This is the default.

   Note: For a description of linked objects, see “Using Linked and Local Objects” on page 62.
5. Click Insert.

Saving Chart Objects

When you save a chart object, you can create a link between that chart object and the source object in the repository. Linking report objects to a source object saves time and effort. By updating one source object, you update all reports that are linked to this source object.

To save the chart object to the repository:

1. Select the chart, then select File, and then Save Object.
2. Specify a name and a location to where you want to save the chart.
3. Perform an action:
   - To create a link from the chart object in the report design to the source object in the repository, select Link to Source Object.
   - To save without creating a link from the chart object in the report design to the saved object in the repository, clear Link to Source Object.
4. Click Save to save the chart object to the repository. When you save a chart object objects in a report, only the specified object is saved, not the entire report design.

Inserting Saved Report Objects

You can insert saved grids, charts, images, and text objects from the repository into a report.
To insert a saved report object:

1. From the Report Designer, do one of the following.
   - Select **Insert**, and then **Saved Object**
   - Right-click and select **Insert Object**, and then **Saved Object**.

2. In the **Insert Saved Object** dialog box, specify the object to insert.
   - **Name**—Name of the object
   - **Description**—Description of the object
   - **Type**—Grid, Text, Image, or Chart

3. Optional: Select **Link to Source Object** to link the saved object to the source object.

4. Click **Insert**.

### Creating Headers and Footers

You can design a custom header and footer in your report using inches or centimeters as your unit of measure for the height. The header is located immediately below the top margin, and the footer is located immediately above the bottom margin. Headers and footers can be used to add a description, add graphics, or insert text functions such as GetCell, Date, and Page. For more information, see “Designing Headers and Footers” on page 76.

To create headers and footers:


2. In the **Header Height** text box, perform an action:
   - To include a header in your report, enter the height of the header.
   - When you specify a height for the header and footer, the system takes into account the top and bottom margins and page size specified in Page Setup. For more information, see “Changing the Page Setup” on page 171.
   - To exclude a header from your report, enter zero (0).

3. In the **Footer Height** text box, perform an action:
   - To include a footer in your report, enter the height of the footer.
   - When you specify a height for the header and footer, the system takes into account the top and bottom margins and page size specified in Page Setup.
   - To exclude a footer from your report, enter zero (0).

4. In **Show as Annotations—Cell Text**: For Financial Management and Planning, select to show cell text in annotations.

5. In **Show as Annotations—Planning Unit Annotations**: for Planning, select to show Planning Unit Annotations in annotations.

6. In **Show as Annotations—Document Attachments**: For Financial Management only, select to show document attachments in annotations.
For information about specifying margins, paper size or viewing size, see “Changing the Page Setup” on page 171.

Positioning Report Objects

Positioning report objects involves placing objects in the desired positions in the Report Designer workspace. For example, you can position a text box on the top left of the report, and position a grid in the center of the report. How you position objects affects the online report and the printed page. You position report objects by specifying position property values or using rulers to position objects manually.

It is important for the appearance of your report to be able to position objects because report objects can change size depending on the amount of data they contain. You can set the position of a report object relative to other report objects. For example, if report objects above the given report object increase or decrease in size, the given report object moves as necessary to maintain the same distance between it and the other report objects.

If you do not position a report object, the report object is printed at its current location regardless of whether report objects above it or to its left increase or decrease in size.

You can also overlap objects, which enables you to organize them however you choose when laying out a report. Objects that can be overlapped include text boxes, grids, charts, and images. Overlapping objects, you can send a report object to the back (behind another object) or bring it to the front (ahead of another object).

Note: The ability to overlap objects is a design convenience only. When printing reports containing overlapping may not print as displayed in the designer, requiring you to rearrange the objects in your report.

When working with multiple overlapped objects in the Report Designer, you can print the report regardless of your object layout. For example, a portion of a text box overlaps a chart, or layers of charts are placed on top of one another. Regardless of how you place the objects in your report, you can print the report.

To position a report object:

1 Open a report.
2 Position the report objects by performing one or more of the following actions:
   - To position a grid, select a grid, then select the Position category in the Grid Properties sheet. In the Horizontal and Vertical drop-down lists, select the grid’s horizontal and vertical position on the report.
   - To position a chart, select a chart, then in the Chart Properties sheet’s Horizontal and Vertical drop-down lists, select the chart’s horizontal and vertical position on the report.
   - To position a text box, select a text object, then in the Text Properties sheet’s Horizontal and Vertical drop-down lists, select the text box’s horizontal and vertical position on the report.
To position an image, select an image, then in the Image Properties sheet’s **Horizontal** and **Vertical** drop-down lists, select the horizontal and vertical position on the report.

To place the selected object ahead of other objects that it overlaps, select an object, right-mouse click, and select **Bring to front**.

To place the selected object behind other objects that it overlaps, select an object, right-mouse click, and select **Send to back**.

**Tip:** To position grid and chart objects manually, click inside the header area and drag to the desired position. To position text and image objects manually, click the object and drag to the desired position.

**Note:** When clicking on a text object, it may move slightly, which may cause some position settings, such as Center, to change. To correct this, reapply the text object position.

### Resizing Report Objects

You can resize report objects so they are displayed fully expanded or at a specified height and width when the report is previewed or printed. You can also enable the system to automatically increase the size of a text box and grid as more area is needed to include all resulting data. When you resize report objects, the contents remain the same size, but the space around the contents changes. For example, you can drag a handle of a grid border to reserve an amount of space to the left of the grid. When you view the grid, the area to the left of the grid remains blank.

**Note:** If you reduce the size of a text box or grid so that there is more content than can fit in it, scroll bars are displayed on the border of an online report.

1. **Open a report.**
2. **Resize the report objects as follows:**
   - To resize a text box, select the text object, then in the Text Properties sheet, select **Autosize**. The text area fully expands when you preview or print the report.

   **Note:** You are limited to 27 lines of text. To see all lines of text, deselect **Autosize**.

   - To enable the system to horizontally and vertically expand a grid, select the grid, then select the **General** category on the Grid Properties sheet. Select **Autosize**.

   **Tip:** To manually adjust the size of a report object, drag a report object’s handle. Use the guidelines and the ruler on the Report Designer workspace to assist in sizing report objects.
Saving Report Objects

When you save a report object, you can create a link between that source object and the source object in the repository. Saving report objects in the repository enables you to reuse them on a consistent basis. This saves time and enables you to build a library of objects to access and use when creating reports.

To save report objects:
1. Click the report object, then select File, and then Save Object.
2. Select a folder or create a new folder.
3. In the Name text box, enter the report object name. You can use upper and lowercase letters, spaces and numbers. The program does not accept special characters such as \ / %, ?, +, <, >, |, ‘, “, *.
4. In the Description text box, enter the description of the report object.
5. Perform an action to save the local report object to the repository:
   - To create a link from the local object in the report to the source object in the repository, select Link to source object.
   - To save without creating a link from the local object in the report design to the saved object in the repository, clear Link to source object.
6. Click Save to save the report object to the repository. When you save report objects in a report, only the specified object is saved, not the entire report design.

Removing Report Objects from a Report

When you remove a linked report object from a report, it is removed from only the report and not from the repository. You can still use the source report object in other reports. For information on removing report objects from the server, see Chapter 11, “Managing the Explore Repository.”

To remove a report object from a report, select a report object, then click Edit, then Delete.

Designing Headers and Footers

Headers and footers are displayed at the top and bottom of each page in a report and can be aligned to the right, left, or center. Headers and footers can include text, images, and formulas that retrieve text, such as the report name. You can also include the page number in a header or footer.

To design headers and footers:
1. Open a report.
2. Click the space outside of a report object to display the Report Properties sheet.
3 Remove any report objects that are located in the header or footer area, then perform one or both of the following actions:
   - To specify a header height, enter a height in inches or centimeters in the **Header Height** text box.
   - To specify a footer height, enter a height in the **Footer Height** text box.

   **Tip:** If you do not want a header or footer, specify zero (0) for the header or footer height, respectively. The height of the header and footer defaults to zero (0) units in a blank report layout.

4 Perform either of the following actions:
   - To insert text in a header or footer:
     a. Select **Insert**, and then **Text**.
     b. In the header or footer area, click where you want to place the upper left corner of the text box and drag to the desired size, then release the mouse button.
     c. Click inside the text box. Then, enter the exact text in the text box, or specify a formula in the text box to dynamically retrieve header or footer information. To specify a formula, add functions such as page, date, and report description, click the **Insert Functions** button in the Text Properties sheet.
   - To insert an image in a header or footer:
     a. Select **Insert**, and then **Image**.
     b. In the header or footer area, point to where you want to place the upper left corner of the image and drag to the desired size, then release the mouse button.

5 Set properties using the **Text** or **Image Properties** sheet.

**Creating a Title Page**
A title page creates a new page at the beginning of your report with the user-defined text.

To create a title page:
1 Open a report to which you want to add a title page and add a text box at the top of the Report Designer workspace.
2 Select a report object below the text box, then from that report object's property sheet, select **Page Break Before**.
3 Design the text for the title page using the options in the **Text Properties** sheet.
4 Select **File**, and then **Print Preview** to view the title page.

**Inserting Page Breaks Before Report Objects**
You insert a page break before a report object to move the report object to the next page.
To insert a page break before a report object:

1. Open a report and select a report object above which you want to insert a page break.
2. From the report object’s property sheet, select the **Page Break Before** check box.

### Viewing Report Summaries

A Report summary shows the name, the description, and the creator of a report. In addition, a report summary shows the modifier, date, and time the report was last modified.

Summary information is saved when you save the report. The following table describes the options you can select to be displayed in your report summary:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the report</td>
</tr>
<tr>
<td>Description</td>
<td>The report description, which you can modify</td>
</tr>
<tr>
<td>Author</td>
<td>The user who created the report</td>
</tr>
<tr>
<td>Created On</td>
<td>The date and time the report was created</td>
</tr>
<tr>
<td>Last Modified By</td>
<td>The user who last modified the report</td>
</tr>
<tr>
<td>Last Modified On</td>
<td>The date and time the report was last modified</td>
</tr>
<tr>
<td>Object type</td>
<td>The type of object you are viewing</td>
</tr>
<tr>
<td>Location</td>
<td>The location to which this object is saved</td>
</tr>
<tr>
<td>Database Connections</td>
<td>Identifies the database connections used for this report object</td>
</tr>
</tbody>
</table>

To view report summaries:

1. Open a report.
2. Select **File**, and then **Summary**.

### Viewing Server Information

You view server information to monitor the activity on your server. Server information consists of various statistics of your report server. The following table describes the report server statistics:

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reports Currently Running</td>
<td>The total number of reports currently running on the report server.</td>
</tr>
<tr>
<td>Statistic</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Total Requested This</td>
<td>The total number of reports that were requested to run since the report</td>
</tr>
<tr>
<td>Session</td>
<td>server was started.</td>
</tr>
<tr>
<td>Total Memory (Bytes)</td>
<td>The amount of memory used on the report server’s virtual computer.</td>
</tr>
<tr>
<td>Free Memory (Bytes)</td>
<td>The amount of remaining memory available on the report server’s virtual</td>
</tr>
<tr>
<td></td>
<td>computer. Memory is usually dynamically allocated to the virtual computer.</td>
</tr>
<tr>
<td></td>
<td>When the memory gets too low, the system allocates more memory, which</td>
</tr>
<tr>
<td></td>
<td>decreases the amount of free memory available on the server.</td>
</tr>
<tr>
<td>Threads—For Running</td>
<td>The number can be changed in JConsole. For information on JConsole, see the</td>
</tr>
<tr>
<td>Reports</td>
<td>“Property Information” topic in the Oracle Hyperion Financial Reporting</td>
</tr>
<tr>
<td></td>
<td>Administrator’s Guide.</td>
</tr>
</tbody>
</table>

To view server information:

1. Open a report.
2. Select File, and then Report Server Info.
3. Click Close.
You can customize grids by calculating data rows and columns. You can use functions to refine your report data or to add dynamic data, such as the date the report was created. You can also cut, copy, or paste rows or columns within a grid or across multiple grids.

Before you begin to customize grids, you must insert at least one grid and define its dimension layout. To customize a report, you must insert a text box or a grid in your report. For more information, see “Adding Report Objects” on page 61.

Inserting Rows or Columns in Grids

You insert rows or columns into a grid to add data to the grid.
Table 9  Row and Column Types

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
</table>
| Data | A data row or column contains members of dimensions that are retrieved from a database connection. The member in a data row or column defaults to the top-level member of the dimension.  
*Note:* A # symbol is used to indicate data rows, columns, or cells in grids. |
| Formula | A formula row or column displays values based on specified formulas. A formula row or column is commonly used to calculate totals, averages, and variances on data rows or columns.  
*Note:* An = # symbol is used to indicate formula rows, columns, or cells in grids. |
| Text | A text row or column primarily contains text typed into the cells, or added dynamically through a text function.  
*Note:* Text rows, columns, or cells in grids do not contain symbols when added to a grid. They are initially blank until populated with text. |

By default, when you add a grid to a report, it contains one of each of the following components: a data row, an anchor row, a data column, and an anchor column. An anchor row or column is a blank row or column located at the end of a row or column. You can add one or more rows or columns within a grid, or you can use the anchor row and column to append new rows and columns.

**Tip:** Use the formatting information provided in “Formatting Grids” on page 144 to format the grid you added to your report.

The grid boundary remains the same size as you add rows and columns. For information on resizing the grid, see “Resizing Report Objects” on page 75.

➢ To insert rows or columns in grids:

1. Select a cell or a row or column, then select Insert, then Row/Column, then Data, and then Formula, or Text. The new row is inserted above the selected row; the new column is inserted to the left of the selected column.

   **Tip:** To insert multiple rows/columns, select a number of rows or column on the grid equal to the number of rows or columns you want to insert.

2. To append a row or column at the end of a grid, select the grid or an anchor row or column, then perform an action:

   - For a row, select Insert, then Row, then Data, and then Formula, or Text.
   - For a column, select Insert, then Column, then Data, and then Formula, or Text.

3. To populate the row or column with data, formula, or text, perform an action:

   - For a data row or column, double-click the row or column heading and select dimension members to place on the row or column. For more information, see “Assigning Members to Data Rows or Columns” on page 28.
   - For a formula row or column, select the row or column heading and build a formula in the formula bar.

82
For a text row or column, enter text directly into the row or column. For more information on formatting text rows or columns, see “Formatting Text Boxes” on page 143.

Note: To delete a row or column, select the desired row or column, then select Edit, and then Delete.

Cutting, Copying, and Pasting in Grids

You can copy, and paste rows and columns and formula rows and columns within the same grid or between two different grids in a report. You can only cut and paste rows and columns and formula rows and columns within the same grid.

Note: For formula rows and columns, you must cut the entire row or column.

To cut, copy, and paste rows or columns in grids:

1. Open Reporting Studio.
2. Select File, then New, and then Report to open a blank report, or select an existing report.
3. Create a grid.
4. Select a row or column and perform an action:
   - To cut a row or column and place the information on the Windows clipboard, select Edit, then Cut.
   - To copy a row or column and place the information on the Windows clipboard, select Edit, and then Copy.

   Tip: To select multiple consecutive rows or columns, press the SHIFT key and click multiple rows or columns.
5. Click the destination row or column.
6. To paste the selection in the destination row or column, select Edit, and then Paste.

   Note: Columns are pasted to the left of the selected destination column. Rows are pasted above the selected destination row. Pasting inserts the column or row and does not overwrite the data in the destination row or column.
7. Select File, and then Save to save the report.

Specifying Row and Column Headings for Grids

Defining row and column headings for the grid enables you to specify the information you want displayed in the row and column headings for the entire grid. You can specify one of the following member labels for row and column headings:
The member name, which displays the member name as the heading.

The alias or description, which displays the alias or description as the heading.

The member name and alias or description, which displays the member name and alias or description as the heading.

**Note:** Whether you specify an alias or a description depends on your type of database connection. If you are reporting against a Essbase or Planning database connection, select Alias. If you are reporting against a Financial Management database connection, select Description.

When specifying row and column headings, you must define where you want row headings to be positioned in relation to the columns. For example, you might want your row headings to be positioned before column A.

**Note:** You cannot place row headings before columns that are sorted.

You can also suppress headings for repeating members in a segment. When you suppress headings, only the first instance of the heading for a member that is repeated is displayed. Headings for the additional repeating members are hidden.

If you are using the Entity dimension through a Financial Management database connection, you can select the check box for Short Name for Entities for the row, column, or grid using the relevant properties sheet. This option enables you to display the short name of the entity in your report, removing the parent portion of the entity that typically displays.

The standard display for an entity using Financial Management is “parent.child.” If you select Short Name for Entities on the relevant properties sheet, the entity displays as “child,” removing the “parent” from the display. For example, an entity that typically displays as “UnitedStates.Connecticut” displays as “Connecticut” when the Short Name for Entities check box is selected.

You may also want to select the alias table you want to use. You can select alias tables in your Essbase or Planning database connection. Although you may not specify aliases on the grid level, if you display aliases for a row or column, the system displays the alias table as selected on the grid level.

The heading properties you specify on the grid apply to all rows and columns unless you specify otherwise.

**To define row and column headings for the grid:**

1. **Open a report and select the grid.**
2. **Select one of the following member label options:**
   - **Member Name**, to show the member name as the heading.
   - **Alias:Default**, to show the alias for a member from an Essbase or Planning database connection OR **Description**, to show a description for a member from a Financial Management database connection.
Both, to show the member name and alias or description as the heading.

**Note:** The option that is displayed on the Properties sheet changes depending on your type of database connection. For example, if you are using Financial Management as your type of database connection, the Description option is displayed. In this case, the Alias:Default option is not displayed unless you are using a Essbase or Planning database connection.

3 Select a column. The row heading is positioned to the left of the selected column. Columns are represented alphabetically as Column A, Column B, Column C, and so on.

4 **Optional:** Select the Short Name for Entities check box to display the short name for entities in your Financial Management grid.

5 **Optional:** Select the Suppress Repeats check box to hide headings for repeating members in a segment.

6 **Optional:** Select the alias table to use when showing aliases. By default, this option is set to the default alias table for the type of database connection.

**Note:** Once in the grid, text can be aligned horizontally or vertically. For more information, see “Aligning Text in Text Boxes” on page 143.

### Overriding Row and Column Headings for Grids

Defining row and column headings enables you to override the heading properties you set for the grid. You can specify one of the following types of headings for rows or columns:

- **Member Labels**
  - The member name, which displays the member name as the heading.
  - The alias or description, which displays the alias or description as the heading.
  - The member name and alias or description, which displays the member name and alias or description as the heading.

- **Custom Headings**
  - Customized text, which displays the user-defined text as the heading.
  - Text functions, which uses a text function to get data to display as the heading.

You can also suppress headings for repeating members in a segment. When you suppress headings, only the first instance of the heading for a member that is repeated is displayed. Headings for the additional repeating members are hidden.

- To override row and column headings for the grid:
  1. Open a report and a formula row or column in the grid.
  2. Do one of the following tasks:
     - Select **Member Labels** and then select one of the following member labels options:
       - **Member Name**, to display the member name as the heading.
– **Alias: Default**, to display an alias for a member from an Essbase or Planning database connection OR **Description**, to display a description for a member from a Financial Management database connection.
– **Both**, to display the member name and alias or description as the heading.

**Note:** The option that is displayed on the Properties sheet changes depending on your database connection. For example, if you are using Financial Management as your database connection, the **Description** option is displayed. In this case, the **Alias: Default** option is not displayed.

3. Select **Custom Heading** and then do one or both of the following tasks:

– Enter a user-defined text heading to be displayed as the heading.

  **Tip:** You can enter keyboard returns to create text headings for columns with multiple lines.

– Enter a text function, or click the **Insert Function** button, and select the text function you want to use for displaying data on the headings.

**Note:** You must select the entire row or column to display the Suppress Repeats option in the Row and Column property sheets.

3. **Optional:** Select the **Short Name for Entities** check box to display the short name for entities in your Financial Management grid.

### Using Row and Column Templates

Subtopics

– About Row and Column Templates
– How Inheritance Works with Row and Column Templates
– Property Values in Rows and Column Templates
– Creating Row and Column Templates
– Inserting a Row and Column Template
– Unlinking a Row and Column Template
– Modifying a Row and Column Template
– Saving a Row and Column Template

### About Row and Column Templates

Row and column templates enable you to select a contiguous set of rows or columns and save them as an object in the repository for reuse in one or more reports.

A row and column template comprises one or more row or column definitions. These row or column definitions may contain the following components:

– Database connection/dimension/member selections
When you create a row and column template, the following items are discarded:

- Cell formulas
- Secondary database connection specifications
- Text from the data cell area (text in the heading cells is retained)
- Conditional formatting
- Related Content
- Conditional suppression

After you create a row and column template, you can save it to the repository, with the following limitations:

- Formulas must refer to rows and columns within the selected rows or columns that comprise the template.
- Formulas outside the row and column template cannot refer to selected rows and columns that comprise or are part of a row and column template.

**Note:** Cell format inheritance must be set before creating the row and column template. You can format cells using the Format Cells dialog box. For more information, see “Inheriting Cell Formatting” on page 151. When inheriting formatting within a row and column template, you must use the Inherit Formatting from option on the Row and Column Template properties sheet, which indicates the options that are available for inheritance. For example, suppression, formatting, and row height or column width.

**Note:** When a report contains two or more templates and one of the templates cannot be retrieved because it is moved, renamed, deleted from the repository, or the data source for the template has changed, the link to the existing template will remains unbroken. No alerts for broken links are given. However, orphan links can be checked by viewing the logs (FRWebApp.log, FRReportServer.log or FRClient.log).

## How Inheritance Works with Row and Column Templates

When inheritance is used with linked row and column templates, certain formatting options are dynamic and vary from report to report. When a linked row or column template is used in a grid, the format inheritance is automatically reset to the first available non-template row above
the row template or to the left of the column template. To keep the format of a row/column template consistent, or if you are experiencing formatting inconsistencies with inheritance and row or column templates, you can perform an action:

- Insert a hidden text row/column containing the formatting you wish to show in the row/column template and set the Inherit from on the Property sheet to reference the hidden row or column.
- Do not use inheritance, but rather set the row or column template with formats through the Format Dialog Box.

### Property Values in Rows and Column Templates

The following tables describe the property values that are saved, inherited, or discarded in a row and column template.

**Table 10  Saved or Inherited Property Values (User option)**

<table>
<thead>
<tr>
<th>Property Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row and Column Template database connection</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Heading—Suppress Repeats</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Display Line Item Details</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Headings</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Allow Expansions</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Calculations</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Hide Always</td>
<td>Saved by row or column</td>
</tr>
<tr>
<td>Positive Number Formatting</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Negative Number Formatting</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Decimal Places</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Scale Value</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Top Border</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Right Border</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Bottom Border</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Left Border</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Shading</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Horizontal Alignment</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Vertical Alignment</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
</tbody>
</table>
### Property Value

<table>
<thead>
<tr>
<th>Property Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indent</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Font</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Font Style</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Font Size</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Font Effects</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Replace Value</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
<tr>
<td>Inherit Formatting From</td>
<td>Saved by cell or inherited by cell—user option</td>
</tr>
</tbody>
</table>

**Note:** If the reference is internal to the row and column template, then it is applied when the row and column template is inserted. If the reference is external, then it is discarded and the “Inherit Formatting From” setting on the row and column template properties property sheet is applied. For more information, see “Using Row and Column Templates” on page 86.

<table>
<thead>
<tr>
<th>Property Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Height/Column Width</td>
<td>Saved by row or column or inherited from another row or column and applied to all rows or columns in the row and column template—User option</td>
</tr>
<tr>
<td>Suppression</td>
<td>Saved by row or column or inherited from another row or column and applied to all rows or columns in the row and column template—User option</td>
</tr>
<tr>
<td>Page Break Before</td>
<td>Saved by row or column discarded</td>
</tr>
</tbody>
</table>

#### Table 11  Discarded Property Values

<table>
<thead>
<tr>
<th>Property Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text from text cells</td>
<td>Discarded—text in the heading cells is retained</td>
</tr>
<tr>
<td>Cell Formulas</td>
<td>Discarded</td>
</tr>
<tr>
<td>Investigation</td>
<td>Discarded</td>
</tr>
<tr>
<td>Conditional Formatting</td>
<td>Discarded</td>
</tr>
<tr>
<td>Text in non-heading cells</td>
<td>Discarded</td>
</tr>
<tr>
<td>Calculation Order</td>
<td>Discarded; irrelevant outside of a grid</td>
</tr>
<tr>
<td>Secondary database connection</td>
<td><strong>Error.</strong> See <em>step 9.b on page 91</em> for more information.</td>
</tr>
<tr>
<td>External References in Formulas (text or numeric)</td>
<td><strong>Error</strong></td>
</tr>
</tbody>
</table>

### Creating Row and Column Templates

You create row and column templates based on rows and columns in a grid. When considering rows and columns for inclusion in a template, select those that contain the member selections, formulas, and formatting suitable for future use. The row and column template can then be imported for use in other grids.
Note: When templates are updated, grids containing a link to the template are updated.

Tip: Row and column templates must contain at least one data row or column. For example, your template cannot consist solely of text and formula rows or columns.

➢ To create a row and column template:

1. Open a report that contains a grid, or create a new grid with supported rows or columns.
2. Select a set of rows or columns to be included in the template.

   Note: The entire row(s) or column(s) must be selected.

3. Select File, then Save Object to save the object to the repository.

   The Save Object dialog box is displayed, Or, right-click on the selected row(s) or column(s) and select Save as Row and Column Template.

4. In the Name text box, enter a unique name for the row and column template (for example, Sales Template).

5. Optional: In the Description text box, provide a brief description. The description is displayed in the repository under Description.

6. For the Type drop-down list box, Row and Column Template is selected by default.

7. Optional: To save the row and column template as a linked object in the current grid, select Link to Source Object. To save the row and column template in the grid without linking it to the current grid, deselect Link to Source Object.

   Note: Link to Source Object is disabled if all rows and columns in a grid are selected for inclusion in a row and column template, as you cannot link this object in the grid. The following error message is displayed: Cannot create a linked row and column template based on the selected rows. The grid needs to have at least one data row that is not part of a row and column template.

8. Click Save to return to the grid. Do one of the following to save the row and column template object as linked:
   - insert a blank data row or column into the grid, omitting it from the row and column template
   - deselect one row or column from the template

   Note: A grid must contain at least one row or column that is not committed to a linked row and column template.

9. Click Save. The system validates the row and column template and checks for properties that are not supported. Depending on the content of your row and column template, you may encounter one or more of the following dialog boxes:
a. If your row and column template contains formulas with external references (for example, to cells outside of the selected row and column template), you are prompted to modify those formulas before saving the row or column template.

**Note:** You can save a row and column template that contains cell formulas, as these can be discarded. For more information, see step 9.c on page 91.

b. If a secondary database connection was specified within the row and column template, a dialog box displays prompting you to continue saving the template. If you save the template, the secondary database connection you specified in the row and column template is discarded. The primary database connection is then used for the row and column template.

c. If unsupported properties are found, an Information dialog box presents a list of properties to discard before saving the template. For example, if the row and column template contains a cell formula, you can choose to save the template without the formula or not save the template.

10 **Review the properties to be discarded, then perform one of the following tasks:**

- Click **Yes** to save the template with the unsupported properties, and proceed to the next step.

  **Note:** If you select Yes, the grid is permanently changed. Items listed in the message box are discarded from the grid.

- Click **No** to prevent changes to the rows and columns (the template is not saved) and return to the grid.

11 **Select the options to save, inherit, or discard settings from rows using the Suppression, Row Height, and Page Breaks dialog box. The following definitions apply to these selections:**

- **Suppression**—You can save or inherit any application of suppression (that is, if zero, missing, or error) within the rows and columns selected.

- **Row Height**—If you selected a row for your template, you can save or inherit any application of row height, which can be changed manually or numerically from the Row Height text box on the Row Properties property sheet, within the selected rows and columns.

- **Column Width**—If you selected a column for your template, you can save or inherit any application of column width, which can be changed manually or numerically from the Column Width text box on the Column Properties property sheet, within the selected rows and columns.

- **Page Breaks**—You can save or discard page breaks specified within the rows and columns selected.

  **Note:** The options that display on this screen vary according to your selection in the grid.

12 **Click OK.**
Inserting a Row and Column Template

After saving the row and column template to the repository as an object, you can insert it into an unlimited number of compatible grids. When a row or column template is inserted into a report, formulas outside of the template are adjusted accordingly. Only formulas within a grid are updated; not external formulas, which refer to the grid with the template, from text objects or another grid.

To insert a row and column template:

1. Open a report and select a row or column adjacent to which you want to insert the row and column template.

   **Note:** Row templates are inserted below the selected row. Column templates are inserted to the left of the selected cell.

2. Select Insert, then Saved Object. The Insert Saved Object dialog box is displayed with a view of the Explore repository.

3. Select the row and column template to insert.

4. To link or unlink a row and column template to or from the repository as an object, perform an action:
   - To create a link between the report and the template in the repository, select Link to Source Object. You can link the row and column template to the current grid so that the grid maintains a dynamic connection to the row and column template preserved in the repository.

      **Note:** A template that is linked within a grid can only be modified through format inheritance. For more information, see “Inheriting Cell Formatting” on page 151.

   - To use the template in the report without maintaining a dynamic link to the source object in the Explore repository, clear Link to Source Object. For more information, see “Unlinking a Row and Column Template” on page 92.

5. Click Insert.

   **Note:** In a grid on a report, a row and column template is marked with yellow heading cells. Dynamic linking is displayed on the Row and Column Template Properties property sheet.

Unlinking a Row and Column Template

After inserting a row and column template into a grid, you can unlink it. This enables you to use the row and column template as a structural template rather than a dynamically connected replica of the row and column template in the repository.

When you unlink a row and column template, the rows and columns in the template are converted into standard, unlinked rows and columns.
To unlink a row and column template:
1. Open a report that contains a grid with a row and column template inserted.
2. Select the row and column template to unlink.
3. On the Row and Column Template Properties property sheet, clear the Link to Source Object check box.
4. When an Information dialog box asks whether to unlink this object, click Yes.

Modifying a Row and Column Template
You can modify a row and column template using one of the following methods:

- Create a new row and column template and save it using the same name as a row and column template located in the repository. For more information, see “Creating Row and Column Templates” on page 89.
- Unlink a row and column template from a grid, then modify and save it to the repository using the same name and folder of the row and column template in the repository.

**Note:** Either methods replace the original row and column template in the repository.

To modify a row and column template by unlinking a row and column template:
1. Open a report and create a grid.
2. Select and unlink a row and column template from the repository, inserting the rows and columns into your grid. For more information, see “Unlinking a Row and Column Template” on page 92.

**Note:** Unlinking the template from the repository dissolves the template and inserts the rows and columns that comprised the template into your grid.
3. Modify the rows and columns as needed.
4. Save the row and column template to the repository using the same name and folder as the row and column template you want to replace.
5. Click OK to confirm the replacement of the original row and column template.

Saving a Row and Column Template

To save a row or column as an object in the repository:
1. In the Report Designer, select the desired rows or columns.
2. Right-click and select Save as Row and Column Template.
Setting Data Query Optimization Options

When working with grids, you can set data query optimization options to enable you to optimize the efficiency of your specified data queries.

Because data queries extract one or more values from a multidimensional database connection, it is beneficial to speed up this process using the data query optimization settings.

For example, you can specify suppression options on your servers and include or exclude the Essbase sparse command, which is used to address the low probability that data exists for every combination of dimension members. For example, a dimension is sparse when intersections of data are empty.

**Note:** When applying optimization options, you may use MDX (Multidimensional expression) to query Essbase data sources on a grid basis or report script.

➢ To set data query optimization options:

1. Open or create a report that contains a grid.
2. Click Task, and then Data Query Optimization Settings.

   **Note:** The options that display in the Data Query Optimization Setting dialog box, depend on the type of database connection you are using. For example, the Essbase Sparse Command option is only displayed when Essbase is the database connection.

   **Note:** The ability to suppress rows can be applied to any data source (Financial Management, Planning, and Essbase). All other options on this dialog box apply to Essbase only.

3. To indicate if the server on which you want to suppress rows, select one of the following options from the Suppress the Rows on the grid:

4. You can specify whether to query a report using MDX (Multidimensional Expression) or report script. For the check box, Essbase Queries Use MDX, perform one of the following:
   a. Clear the check box to use the report script queries. This will disable the “Show Drill Through” setting on the grid.
   b. Select the check box to use MDX for querying the Essbase data source (this is the default), then select a method for running the query:
      i. **Combine queries using with members** (this is the default). This method combines the MDX queries into a single query using the MDX with members syntax. By minimizing the number of queries, performance should be improved.
      ii. **User separate queries.** This method issues separate MDX queries for each row which has a member selection function (for example, “Descendants of” or “Children of”).

5. To indicate if the Essbase sparse command is included or excluded, select one of the following options from the Sparse Command Is frame:
• Included
• Excluded

**Note:** The Sparse Command option is only displayed when the database connection uses Essbase a data source.

6 **Optional:** Click **Set to Default** to use the default data query optimization settings.
7 Click **OK** to accept the settings you indicated.
8 **Optional:** Click **Cancel** to exit and return to the grid in the Report Designer.

# Displaying Line Item Detail for Financial Management

For Financial Management database connections, a Financial Management user can select if they want to display line item detail for rows only on Account dimensions. A user can also select the location of the line item detail; before or after the parent member of the line item detail. To display line item detail, all dimensions must be set to a Level 0 member.

➢ To display line item detail for Financial Management:

1 Open a report and select a row in the grid.
2 In the Row Properties Sheet, select Display Line Item Details.
3 **Optional:** On the Grid Properties sheet, select the **Position** category and do one of the following in the **Position of Line Item Detail** box:
   - To display line item detail before the parent member, click **Before Members**.
   - To display line item detail after the parent member, click **After Members**. This is the default.

# Showing Supporting Detail for Planning

When using the Planning database connection, and selecting a row, you can display supporting detail for a member. If supporting detail exists for one or more cells in the selected row, additional rows are inserted to show the supporting detail. In addition, you can control whether a page break is allowed within the rows of supporting detail.

➢ To show supporting detail for the Planning database connection:

1 Open a report and select a row in the grid that contains a member with supporting detail.
2 In the **Row Properties** sheet, select **Display Supporting Details**.
3 **Specify whether to allow page breaks with the rows of supporting detail:**
   - To allow a page break within the rows of supporting detail, select the **Allow Page Break Within** check box.
To keep the parent member and the supporting detail on the same page, deselect the Allow Page Break Within check box. If the supporting detail does not fit on the current page, the parent member and supporting detail is moved to the next page.

**Note:** If the supporting detail spans multiple pages, this option defaults to Allow Page Break Within and page breaks are used.

4 Optional: To position the supporting detail before or after the parent member, select the Grid Properties sheet, select the Position category and select an option in the Position of Supporting Detail box:

- To display supporting detail before the parent member, click Before Members.
- To display supporting detail after the parent member, click After Members. This is the default.

5 Optional: To indent each generation within the supporting detail:

a. Select the heading cell of the member that is going to display the supporting detail.

b. Select Format, then Cells, and then Alignment tab.

c. In the Indent Increases for Each Generation by box, select a positive number by which you want to indent (to the right) each generation in the supporting detail list, or select a negative number to indent in reverse (to the left) each generation in the supporting detail list.

**Note:** This property applies only to heading cells on the rows. It is enabled for Planning database connection only when Display Supporting Detail is selected. For more information on formatting, see Chapter 9, “Formatting Reports.”

6 Optional: To apply additional formatting to supporting detail, use conditional formatting. The following two options are useful for supporting detail:

- Supporting Detail. If, at runtime, supporting detail results from a cell, you can format the supporting detail.

- Relative Generation. You can format the different generations of supporting detail independently. For example, if the resulting hierarchy of a member includes four generations of supporting detail, you can specify the third generation to format. The conditions would resemble the following example:

  **Note:** In order for all types of database connections to use relative generation, you must have selected a member relationship that returns members on multiple levels; for example, Children of...(Inclusive), Descendants of..., Descendants of...(Inclusive), Parents of... (Inclusive), Ancestors of, Ancestors of...(Inclusive), and AllMembers. For Planning, you may use relative generation with a member or dynamic member if supporting detail is turned on.

  Condition 1: If
  Supporting Detail Is True And
  Relative Generation (of) Market = 3
Defining the Grid Point of View

The grid point of view (POV) enables designers to specify the members for a dimension on a grid without placing the dimension on the row, column, or page intersection. After a dimension is specified on the grid POV, the designer determines the ways that the viewer can select members, as follows:

- Any member. The viewer can select any member for the dimension.
- A member from list. The viewer can select from a limited list of members that the designer previously defines.
- Nothing, Lock Member Selection. The viewer cannot select any member for the dimension.

Note: To improve report performance, do not use dimensions with only one member selected (for the entire grid) into a row, column, or page axis. Although, it does not increase the potential cell count, it adds overhead to the entire query. Dimensions that are variable or set to one member should be left in the POV. For variable members, include the user POV and for fixed members, include the grid POV. The rules for each database connection for a grid are as follows: - If the dimension is on a grid axis (row, column, or page), use the member selected for that dimension. - If the dimension is set on the grid POV, use the member selected for the grid POV dimension. - Use the member selected for the user POV dimension.

Showing or Hiding the Grid POV Bar

As a report designer, you can hide or display the grid POV bar. Hiding the grid POV prohibits you from changing the grid POV when viewing a report. By default, the grid POV bar is hidden. However, even when a POV bar is hidden, its values are still operative when running the report.

Note: The grid POV only displays in a HTML view, and not in a PDF view. In HTML view, the report is interactive, so you can change the Grid POV members, whereas a PDF view is a static document, and you cannot change the Grid POV.

To show or hide a grid POV bar:

1. Open a report.
2. Select the grid for which you want to show or hide its grid POV bar.
3. Do one of the following tasks in the Grid Properties sheet:
   - To show the grid POV, select the Grid Point of View check box.
   - To hide the grid POV, deselect the Grid Point of View check box.
Tip: As an alternative, you can also select a grid, right-click inside the grid, and select Grid Point of View.

Selecting Members for the Grid Point of View

On the grid point of view (POV) bar you can select members for dimensions that are not used in rows, columns, or the page axis. The members defined on the POV bar complete the criteria used to retrieve data for the grid. Each POV dimension can contain only one member.

Note: For Essbase and Planning database connections, Oracle recommends excluding dimensions with security on it in the Grid POV since it may cause unexpected results for end users.

➢ To select a POV member for the grid POV:
  1. Open a report.
  2. To display the grid POVs, select Grid Point of View from the Grid Properties sheet.
  3. On the grid POV, click on a dimension to display the Select Members dialog box.
  4. In the Select Members dialog box, each dimension is displayed on a separate tab. Select the dimension listed on the tab, then select an associated member. Repeat for all dimension on the tabs.
  5. Select Members Changed During Related Content Link to change the Grid POV members during a Related Content link to a report which has dimensions on the Grid POV.
  6. Click OK. The members are displayed on the grid POV.

Setting Up a Grid Point of View

A designer can set up a grid point of view (POV), to control or limit the selection of members available to end users who view or print the report.

For more information, see Chapter 2, “Defining Members.”

➢ To set up a grid POV:
  1. Open a report with members selected for dimensions on the grid POV.
  2. Right-click a grid and select Grid Point of View Setup.
  3. For each dimension presented in the dialog box, you can specify the availability of member selection to the end user:
     - Any Member—All members are available.
     - A Member from List—Only the members you specify are available. Click in the Choices List column. The Select Members dialog box is displayed.
     - Nothing, Lock Member Selection for a specific dimension—No members are available.
When the Report Viewers May Select column is set to A Member from List, Select a member from the Select Members dialog box, then click OK. For more information on selecting members, see Chapter 2, “Defining Members.”

Select the Show Dimension Name and Show Member Name check box to display the dimension name with the member name.

Select the Show Alias check box to display alias names.

Click OK.

**Inserting Page Breaks**

You can insert a page break before a row or column in a grid from the Row Properties and Column Properties sheets. This enables you to break your report data at a specified location within a grid while keeping required line items together. Page breaks are indicated with a heavy, dashed line.

To insert a page break:

1. Open a report, then select a grid.
2. Select or highlight the row or column in which you want to insert a page break.

**Note:** You cannot insert a page break before the first column in a report or above the first row in a report.

3. In Row or Column Properties sheet, select the Page Break Before option.

**Note:** If Fit Length to Page is selected in Page Setup, you cannot apply page breaks to rows. If Fit Width to Page is selected in Page Setup, you cannot apply page breaks to columns. For more information on Page Set Up, see Chapter 13, “Working with Books and Snapshot Books.”

4. Perform an action:

   - Select Position at Top when selecting a row if you want the remaining part of your grid to be placed at the top of the new page or select Position at Left when selecting a column if you want the remaining part of your grid to be placed at the left margin of the new page.
   - Select Same Position to place the grid in the same position as it is currently placed. For example, if a grid is placed halfway down the page and you insert a page break with Same Position selected, the remaining part of the grid is displayed halfway down on the second page.

**Defining Page Axis Headings**

A page axis heading prints the member name every time the page member changes. Members on the page axis can have their own custom heading.
For more information on printing the page dimension heading, see “Printing Page Dimension Headings” on page 139.

Table 12 Options for Page Dimension Headings

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New page, Position at Top</td>
<td>Prints a new page at the top of the next page. Use this option when chart is linked to a grid and you want to ensure the chart and grid print next to one another whenever the page member changes.</td>
</tr>
<tr>
<td>New page, Position Same</td>
<td>Prints a page at the specified grid location. Use this option in the following situations:</td>
</tr>
<tr>
<td></td>
<td>- The grid has horizontal and/or vertical alignment properties set and you want the grid to print in the same position on each page. For example, if you set the Grid position to Center/Middle, it is printed in the center of each page.</td>
</tr>
<tr>
<td></td>
<td>- When a chart is linked to a grid and you want to ensure the chart and grid print next to one another whenever the page member changes.</td>
</tr>
<tr>
<td>Same Page</td>
<td>Prints the new page immediately after the data from the previous page.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> When a chart linked to a grid, the chart is printed once. The chart is not printed every time the page member changes.</td>
</tr>
</tbody>
</table>

To define a page dimension heading:

1. Open a report and select the Pages list box on the grid.
2. On the Page Properties sheet, in the Headings area, select a dimension.
3. Do one of the following tasks:
   - Select Member Labels and then select one of the following member labels options:
     - Member Name, to display the member name as the heading.
     - Alias:Default, to display an alias for a member from an Essbase or Planning database connection OR Description, to display a description for a member from a Financial Management database connection.
     - Both, to display the member name and alias or description as the heading.

   **Note:** The option that is displayed on the Properties sheet changes depending on your database connection. For example, if you are using Financial Management as your database connection, the Description option is displayed. In this case, the Alias:Default option is not displayed.

   - Select Custom Heading and then do one or both of the following tasks:
     - Enter a user-defined text heading to be displayed as the heading.

       **Tip:** You can enter keyboard returns to create text headings for columns with multiple lines.

     - Enter a text function, or click the Insert Function button, and specify a text function to use a text function to get data to be displayed on the headings.
**Note:** For more information on adding a text function to a row or column heading, see “Using Text Functions to Display Information” on page 119.

4. **Optional:** To set formatting options, click the **Format** button.

5. **Specify a height for the header.**

6. **In the Page Printing Positioning Option area, select one of the following options:**
   - New Page—Top
   - New Page—Same
   - Same Page

   **Note:** For definitions of the Page Printing Positioning options, see Table 12 on page 100.

### Changing the Database Connection of Grids

Database connections of grids is done in Workspace. When you change the database connection of a grid, you display data from another source. Because the dimension names for the new database may differ from the default database, the administrator needs to prepare a mapping schema to maintain the integrity of the data structure. If the structure of the specified database connection differs from that of the current database connection, the report cannot repopulate.


### Assigning Multiple Database Connections Grids

You can retrieve data from a different database connections and define the database connection for each data row or data column.

You can preserve the members selected in the grid for the database connection on a grid row or column. You must map all dimensions that are assigned to rows or columns in a grid. Dimensions on the point of view (POV) need not to be mapped.

**Note:** Database connections must be of the same type (for example, both must be Essbase database connections or both must be Financial Management database connections). Multiple data sources in one grid that are of different types (Financial Management, Planning, Essbase) are not supported. To retrieve data from a different database type, add a second grid to the report. The second grid retrieves data from a different database type connection. In the first grid, reference the data from the second grid with formulas. Use text functions to display the correct member names for the formulas and hide the display of the second grid in the report.
To assign multiple database connections of the same type to a grid:

1. Open a report.

2. In a grid, select one or more data rows or columns for which you want to change the database connection.

3. In the Row or Column Properties sheet, click the Select Database Connection button.

4. In the Select a Database Connection dialog box, select a database connection from the Database Connection drop-down list, then click OK.

**Note:** To create another database connection of the same type, select another data row or column, then click Database Connection button again.

5. In the Map Dimensions Between... dialog box, map dimensions between the default database connection and the selected database connection. You do this by selecting a dimension from the pull-down list in the right column for each dimension in the left column.

6. Optional: Select an alias table for the new database connection.

7. Click OK.

**Adding Auto Calculations to Data Rows and Columns**

You can append auto calculations to the data rows and columns of your grids. For example, you might want to calculate the total of rows or columns that expand and contain multiple dimensions.

When you define an auto calculation for a data row or column, a row or column is inserted at runtime that contains the calculated value. This calculation goes against all members in a row or column and is displayed when you print or preview the report.

You can apply several formatting options. For each auto calculation, you can specify a heading for the calculated row or column and a group heading. You can specify the location of the calculation row or column in relation to the member, and you can insert blank rows before and after calculation rows and group headings. You can also specify page breaks within rows of calculation and after a calculation row.

You can apply several of the formatting options provided with auto calculation but exclude the auto calculation function for a specified cell, row, or column.

You can define different types of calculation for each dimension or member in the row or column using one of the following functions:

- **Average**—The average value of values in the segment in this dimension at runtime (ignores non-number values)
- **Average All**—The average value of values in the segment in this dimension at runtime (assigns 0 to non-number values)
- **Count**—The number of values in the segment in this dimension at runtime (ignores non-number values)
- **Count All**—The number of values in the segment in this dimension at runtime (assigns 0 to non-number values)
- Maximum—The largest value of the values in the segment in this dimension at runtime
- Minimum—The smallest value of the values in the segment in this dimension at runtime
- Product—This function multiplies all numbers or references in the segment in the dimension and returns the product at runtime,
- Total—The sum of the values in the segment in this dimension at runtime
- No Calculation—Calculation is not performed on the selected cell, row, or column. However, formatting provided with auto calculation is retained. You can apply No Calculation to an auto calculation row, column, or cell or non-calculation row, column, or cell.

**Note:** If errors or missing data exist for members, the previous operations include the member value in the subtotal, but the member value is treated as a zero value.

**Tip:** You can use the formatting options without having calculation done. This enables you to retain the member grouping without calculations such as totals.

To add auto calculations to data rows or columns:

1. Open a report.
2. Select the heading cell of a row or column that contains multiple members.
3. In the **Heading Row or Column Properties** sheet, select **Auto Calculation**, then click **Setup**.
4. In the **Calculation** area, select a function for the type of calculation you want to perform. Select **No Calculation** to only include auto calculation formatting.
5. **Optional:** To add headings to the calculation or group rows or columns, take any of the following actions in the **Headings** area.
   - Select the Group Heading check box, then enter a custom heading to describe the row or column. To insert formulas in the headings, click the **Functions** button. For more information, see “Using Text Functions to Display Information” on page 119.

   **Note:** When using the MemberName function, you must enter “current” for the row/column/page parameters.

   **Note:** The header is suppressed if the surrounding data is suppressed.

   **Tip:** You can suppress repeats on multiline headers by selecting the Grid Properties sheet, General category, and selecting the Suppress Repeats check box.

   - In the **Auto Calculation Heading** text box, enter a custom heading for the calculation row or column. To insert formulas in the headings, click the **Functions** button. For more information, see “Using Text Functions to Display Information” on page 119.
6 Optional: To insert blank rows or columns before or after calculation rows or columns and group headings, select any check boxes in the Insert Blank Row / Column area.

7 To change the height or width of the inserted row or column, enter the unit in the Row Height or Column Width text box.

Tip: You can specify whether to insert the auto calculation row or column before or after the expanded members of the row or column. Do this by displaying the Grid Properties sheet, then select the Position category.

8 Select Allow Page Breaks Within to allow a page break within the rows of auto calculation. When deselected, the heading cell and the row are kept together. If the row is too long to fit on a page, the heading cell and the row is moved to the next page. If the auto calculation row is longer than one page, the option defaults to Allow Page Break Within.

9 Select Allow Page Breaks After Auto Calculation to allow a page break after the calculation row.

10 Click OK.

11 To view the report, select File, and then Print Preview. The example report resembles Figure 2.

Note: See Figure 3 to view the same report with applied conditional formatting.

Figure 2  Example Report with Auto Calculation

Sales

Q1 1 Sales

East  20,621
West  31,674

SubTotal  52,295

Q1 2 Sales

East  22,440
West  33,572

SubTotal  56,021

GrandTotal  108,316
Applying Formatting to Auto Calculated Data Rows and Columns

You can use the conditional formatting feature to apply additional formatting to the auto calculation rows and columns, such as underline, bold, and indent. The conditional formatting feature contains two conditions specifically designed for auto calculation:

- Auto Calculation—Used to apply formatting to the auto calculated cells and auto calculation heading.
- Auto Calculation Group Heading—Used to apply formatting to the section headings.

For more information on conditional formatting, see “Applying Conditional Formatting to Grids” on page 158.

Figure 3, displayed below, is an example report that uses conditional formatting to specify font, alignment, and border and shading settings. The following conditions were applied to the example report by selecting the East, West Heading Cell and Calculated Data Cell:

- Format 1: IF (Auto Calculation Year is True, then Format Cells (double border top, Font = Ariel, Bold, Font size = 12)
- Format 2: IF (Auto Calculation Market is True, then Format Cells (single border top)
- Format 3: IF (Auto Calculation Group Heading Market is True, then Format Cells (Alignment = center, Font Style = italic)
- Format 4: IF (Auto Calculation Group Heading Year is True, then Format Cells (Bold, Font size = 12, Color = blue)

Figure 3  Print Preview of Example Report with Applied Conditional Formatting

<table>
<thead>
<tr>
<th>Sales</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qtr1 Sales</td>
</tr>
<tr>
<td>East</td>
</tr>
<tr>
<td>West</td>
</tr>
<tr>
<td>SubTotal</td>
</tr>
<tr>
<td>Qtr2 Sales</td>
</tr>
<tr>
<td>East</td>
</tr>
<tr>
<td>West</td>
</tr>
<tr>
<td>SubTotal</td>
</tr>
<tr>
<td>Grand Total</td>
</tr>
</tbody>
</table>
To apply formatting to calculated data rows and columns:

1. Open a report.
2. Select a cell or group of cells that is set up with auto calculation.
3. Select **Format**, and then **Conditional Format**.
4. Select **Auto Calculation** or **Auto Calculation Group Heading** from the Property drop-down list located below **Condition 1: If**, then enter the conditions.

Figure 4  Conditional Formatting Applied to Specify an Auto Calculated Cell

5. Click the **Format Cells** button to apply formatting to the calculated cell.
6. Click **OK** to accept the formatting than you specified for the calculated cell. For more information on formatting, see “Formatting Grids” on page 144.

**Note:** You can use conditional formatting to establish formatting for calculated cells. For more information, see “Applying Conditional Formatting to Grids” on page 158.

7. To view the report, select **File**, and then **Print Preview**. The example report resembles **Figure 3**.

**Defining Formula Rows or Columns**

A formula consists of a combination of grid references, mathematical functions, and arithmetic operators. A grid reference uses values from the current grid or another grid in a calculation. You can define arithmetic formulas on formula rows, columns, cells, or a range of formula cells in a grid. For example, you might want to build a formula that adds one or more rows. Or, you might want to multiply two rows, add a third row, then display the result. For a list of mathematical functions you can use in formulas, see “Mathematical Functions” on page 174.

When defining formula rows or columns, keep in mind the following points:

- Formula results are calculated based on the underlying data values in one or more specified cells. If cells used in a formula are formatted to use scaling, the formula results may differ from the displayed values for the cells, since the underlying, unformatted data values are used in the formula.
- By default, cells that contain missing data (#MISSING) are not treated as zero (0) in formulas, unless specified differently using the IfNonNumber property. In a formula that uses division, an ERROR is returned.

- When using the Financial Management database connection, the results vary, based on the settings of the MissingValuesAreZeroInFormulas (formerly MissingValuesAreZeroInFormulasInHFM) option in the JConsole.exe file. For information on JConsole, see the “Property Information” topic in the Oracle Hyperion Financial Reporting Administrator’s Guide. If the property is set to one (1), the #Missing value is treated as a zero. However, if the property is set to zero (0), the #Missing value is treated as missing which results in #missing or #error values. The default setting is one (1).

- To specify mathematical properties, you can use dot notation. Dot notation is a syntax that specifies properties for a grid or another property. You specify a property using a period (.) followed by the property name.

Syntax:

```
gridname.row\column[x].property
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gridname</td>
<td>The name of the grid. Grid names cannot include spaces. For more information on grid references, see “Mathematical Functions” on page 174.</td>
</tr>
<tr>
<td>x</td>
<td>The row or column, number, and property.</td>
</tr>
</tbody>
</table>

For example, mygrid1.row[3].sum adds all elements in row 3 of mygrid1.

**Note:** When specifying the full gridname notation, you must also indicate the row, column, or cell.

This example returns the sum of the nth column:

```
mygrid2.column[C].sum
```

You can omit references to the current grid, row, column, or cell. The following example returns the absolute value of the cell located at the intersection of row 5 and column B in the current grid:

```
ABS([5,B])
```

- To define formula rows or columns:
  1. Open a report.
  2. Select a row or column in a grid.
  3. Select Insert, then Row, and then Formula or Insert, then Column, and then Formula.
  4. Select the formula row or column, or a range of formula cells.
  5. In the drop-down menu from the formula bar, select a function. For a list of mathematical functions, see “Mathematical Functions” on page 174.
6. Insert your cursor in the formula text box and enter the rest of the formula.
7. Repeat steps 4 through 6 as necessary to build your formula.
8. Select File, and then Print Preview to see the results of your formula.

Defining Headings for Formula Rows and Columns

You can define a heading for a row or column. By default, the formula is shown as the heading for the row or column. You may also create a custom heading for the row or column.

➢ To define headings for formula rows and columns:
1. Select the heading cell.
2. From the Headings area of the Row or Column Properties sheet, do one of the following
   - Select the Show Formula option. This is the default.
   - Click the Custom Headings button. Enter text in the Custom Heading text box area.
     You can also insert a text function in your custom heading.

Expanding Rows or Columns

In the designer, you can use the design-time row, column, and cell to reference the calculated rows, columns, or cells.

The syntax for row or column is:

\[ \text{AXIS}[\text{Segment} (\text{Range})].\text{Property} \]

The syntax for a cell is:

\[ \text{CELL}[\text{Row Segment} (\text{Range}), \text{Column Segment} (\text{Range})].\text{Property} \]

<table>
<thead>
<tr>
<th>Reference</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AXIS</strong></td>
<td>A row or column containing the expanded cells identified by the given row and column templates. The default is row. For example, ([3]) refers to row 3 and ([C]) refers to column C. The following reference: ([3].\text{sum}) in a column formula is calculated as the sum of row 3. Optional.</td>
</tr>
<tr>
<td><strong>CELL</strong></td>
<td>The intersection of the expanded cells, given the row and column template coordinates, respectively. For example, the following reference: ([2,C]) is the single cell where row 2 intersects column C. Optional.</td>
</tr>
</tbody>
</table>
| **Segment** | A value that represents a row or column ID in the designer. Referenced segments refer to design-time rows or columns, which might expand in the viewer. View-time rows and columns are the expanded rows and columns. Use numbers for row IDs and letters for column IDs. 

**Note:** If you are using an expanded formula row, the default segment is row. If you are using an expanded formula column, the default segment is column. For example, row \([5]\) refers to expanded row 5. For more information on using segments, see “Expressing the Segment Argument” on page 109. Optional. |
**Reference** | **Definition**  
---|---  
**Range** | The expanded cells used in the calculation of the function.  
Range refers to view-time expanded rows and columns. Calculations are performed after suppression. Sorting is performed after calculations. For more information on sorting and suppression, see "Conditional Suppression" on page 164 and "Sorting Rows, Columns, and Pages" on page 153.  
Use colons to refer to a range of expanded rows or columns. For example, (3:6) addresses rows 3, 4, 5, and 6. For more information on using ranges, see “Expressing the Range Argument” on page 110.  
Optional.  
**Property** | Optional. A function applied to aggregate rows, columns, or cells. For more information on using properties, See “Aggregate Property Arguments” on page 176.  

---

**Expressing the Segment Argument**

The Segment argument is a value that represents a row or column ID in the designer. Referenced segments refer to design-time rows or columns, which might expand in the viewer. View-time rows and columns are the expanded rows and columns. You can use numbers for row IDs and letters for column IDs. You can refer to one or more segments in any standard formula. The following table explains the types of notation used for multiple segments in standard formulas.

<table>
<thead>
<tr>
<th>Type</th>
<th>Notation</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single segment</td>
<td>[ ]</td>
<td>Single (design-time) segment</td>
<td>This expression points to segment 2: row[2]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong> Previous releases support this notation.</td>
</tr>
<tr>
<td>Nonconsecutive segment</td>
<td>[:]</td>
<td>Refers to multiple single rows or column segments.</td>
<td>This expression points to segment rows 2, 5, 7, 8 and 10: row[2;5;7;8;10]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Note:</strong> Use a semicolon rather than a comma to delineate row or column references.</td>
</tr>
<tr>
<td>Segment Range</td>
<td>[:]</td>
<td>Refers to a range of rows or column segments.</td>
<td>This expression points to segment rows 2, 3, 4 and 5: row[2:5]</td>
</tr>
</tbody>
</table>

You can use any combination of the above.

**Example:**

Point to segment rows 1, 3, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15 and 17:

```
row[1; 3; 5:10; 12:15; 17]
```

**Example:**

If the following expression references row segment 20, but the grid contains only 15 row segments, the formula works only on segment 4:

```
row[4; 20]
```

If you enter a larger starting segment than ending segment, the system internally switches them around.
Example:
Show how the segments are internally switched:

\[
\text{row}[6:3] \text{ becomes } \text{row}[3:6]
\]

Multiple references to the same segments are valid. This causes the same segments to be included twice in an expression.

Example:
Sum row 5 twice:

\[
\text{row}[2;3;5;7;5].\text{sum}
\]

Example:
Reference row segments 2, 3, 4, 5, 6, 7, and segment 5 again. It includes segment 5 twice in the calculation of the average:

\[
\text{row}[2;7;5].\text{avg}
\]

Text segments are ignored.

Example:
Ignore row segment 3 if it is a text row:

\[
\text{row}[1:5]
\]

**Expressing the Range Argument**

The Range argument refers to view-time expanded rows and columns, on which calculations are performed after suppression. Sorting is performed after calculation is completed.

<table>
<thead>
<tr>
<th>Table 14</th>
<th>Range Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
<td><strong>Notation</strong></td>
</tr>
<tr>
<td>None</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Single Range</td>
<td>[segment (range)]</td>
</tr>
<tr>
<td>Consecutive Range Reference</td>
<td>[:]</td>
</tr>
<tr>
<td>Nonconsecutive Range references</td>
<td>[..]</td>
</tr>
</tbody>
</table>

If a range is used, you can use any combination of the above range notations.

Example:
Point to expanded rows 5 through 10 in design-time row 2:
row[2(5:10)]

**Note:** If row segment 2 expands to 15 rows, the function operates on only expanded rows 5 through 10.

**Example:**
Identify columns C through E in design-time column A:
column[A(C:E)]

**Example:**
Point to expanded rows 1, 3, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15 and 17 in design-time row 2:
row[2(1,3, 5:10, 12:15, 17)]

**Note:** When the grid contains suppression properties, expanded rows and columns may be suppressed. For information on suppressing data, see “Conditional Suppression” on page 164.

If the Range argument refers to a number of expanded rows that is greater than the maximum number of rows expanded in the segment, the extra rows are ignored.

**Example:**
In the following example, expanded rows 6 through 10 are identified, but segment 1 contains only 3 expanded rows. In this situation, the function works only on expanded rows 6 through 8:
row[1(6:10)]

If none of the expanded rows in the Range exist, the entire axis reference is ignored.

**Example:**
Ignore the reference to 23 if segment 3 only contains 10 expanded rows:
row[3(23)]

**Example:**
If you enter a larger starting range than ending range, the system internally switches them around.

**Example:**
Show how the range is reversed:
row[1(7:5)] becomes row[1(5:7)]

Multiple references to the same expanded rows or columns are valid. However, this may cause the same rows or columns to be included twice in an expression.

**Example:**
Sum expanded row 5 twice:
row[2(3,5,7,5)].sum

Example:
Specify expanded rows 3, 4, 5, 6, 7, and row 5. It includes row 5 twice in the calculation of the average:
row[2(3:7,5)].ave

**Identifying Multiple Segments with Ranges**

The syntax for a range in a segment range is:

\[AXIS[Segment Start : Segment End (Range Start : Range End)]\]

where Segment Start and Segment End refers to multiple segments.

Example:
Points to expanded rows 5, 6, and 7 in row segments 1, 2, 3, and 4:
row[1:4(5:7)]

Example:
Point to expanded rows 5, 7, and 9 in row segments 1, 2, 3, and 4:
row[1:4(5,7,9)]

If the segment argument refers to segments that do not exist in the axis, the segments are ignored. This includes suppressed rows or columns.

Example:
Point to expanded rows 4, 6, 7, 8, 9, and 10 in row segment 1 and 4, 5, and 6 in row segment 3 and all expanded rows in row segment 7:
row[1(4,6:10); 3(4:6); 7]

**Multiple Calculation Iterations**

You can specify the maximum number of calculation iterations (MaximumCalculationIterations) for all grids and cells to resolve dependencies within references in formulas in the JConsole.exe file. During the calculation process of a grid, it may be necessary to evaluate a cell multiple times due to reference precedence. This usually occurs in grids with references to other grids. The maximum iteration property indicates the number of times a formula cell can be evaluated before it is marked as unresolved. Setting the maximum iteration property avoids the possibility of cells, with circular referencing, being evaluated an infinite amount of times. Circular referencing occurs when one cell refers to another cell, which then refers to the original cell.

**Note:** For information on JConsole, see the “Property Information” topic in the *Oracle Hyperion Financial Reporting Administrator’s Guide.*
If there are no circular references and calculation cells are returning #Error, you can increase the maximum iteration property value.

For more information, see Oracle Hyperion Enterprise Performance Management Workspace Administrator's Guide.

**Using the Formula Bar**

The formula bar provides a central location for the administration of formulas and selection of members when setting up grids in Financial Reporting.

**Note:** You can type selected functions to pick members dynamically. See Table 2 for a list of functions that can be used in the Formula Bar and Table 2 on page 34 for a list of functions that cannot be used in the Formula Bar.

The formula bar contains a text box for the entry of formulas and a drop-down list box of functions to choose from when building formulas. If you select a member in a grid, member selection is activated on the formula bar.

The formula bar is displayed below the toolbar so you can easily access its functionality. It is visible but disabled when you are not using grids.

The Financial Reporting formula bar is similar in functionality and appearance to the formula bar in Excel. You can use the formula bar to perform the following tasks on grids:

- Define, edit, and delete a formula in a cell, column, or row
- Modify member selection for a column or row

**Example Report with Formulas**

You can use the formula bar to create formulas that total data, average data, or perform other functions of your choosing. When creating formulas for rows that summarize data for the year, for example, you use the formula bar to select the functions and dimensions that return your result. The following example illustrates this information:
- Monthly total figures for the Product dimension
- A total for all months, calculated using the Sum function
- An average amount per month, calculated using the Avg function

**Note:** Use custom headings to create titles for the Total and Average columns. This is described in Chapter 4, “Building Reports.”

To create this example, you would create a grid in the designer with Product on the rows and Year on the columns. Double-click Year to bring up member selection. Place all twelve months in the selected pane and remove Year. Click OK.

After you select the members, insert two formula columns: one for the annual total and another for the average monthly amount.

In our example, the first formula column that falls outside the months of the Year member contains a Sum function. This function adds the figures for each month. Since all months are defined in one cell only, the reference is to that cell location. The formula is built in the formula bar:

```
Sum(Cell [A,1])
```

The second formula column calculates the average of the months for the Year member. Because there are 12 months in a year, the Avg function adds all monthly totals and divides the total by 12. The formula is:
In general, there are two ways to specify a data formula in a grid, a row/column formula, or a cell formula:

- **Row/Column**—Apply one formula to the entire row or column, relative to each cell in that row or column. You define a formula by clicking on the formula row or column header to highlight the entire row or column, then entering the formula in the formula bar. There are some performance benefits when applying row/column formulas.

- **Cell**—Apply a formula only to cells in a formula row or column. You define a formula by clicking on the cell, selecting Use a Custom Formula from the properties sheet, and entering the formula in the formula bar. Also, if the cell intersects a formula row and formula column, you can select to use the row formula or the column formula as the cell formula. For more information, see “Calculating at the Cell Level” on page 118.

Therefore, if a formula repeats for each cell in a row or column, use a row/column formula. If different formulas are performed on each cell, use a cell formula.

➤ To access the formula bar:

1. Open or create a report.
2. Select a dimension heading or formula row/column in the grid to enable the formula bar.

### Creating Formulas Using the Formula Bar

When you select a cell, column, or row containing a formula specification, you can apply a unique formula to that cell.

For example, the custom formula depicted above adds the value of the cell located at A,2 to the value of the cell located at B,2.

**Tip:** See the following list to optimize formulas:

- Use row/column formulas as opposed to cell formulas whenever possible.
- Use reference properties instead of functions when possible. Rows, columns, or cell references can have a property associated with it. For example, the row reference: \([1:5].\text{sum}\) produces the same result as using the Sum function; \(\text{Sum}([1:5])\). However, the first formula executes faster since it is being used as a reference property.
- Avoid using cross-axis references and consider using a cell reference if possible.
- Use parentheses only when necessary. Improper use of parentheses can result in unnecessary evaluation iterations within the evaluation routine of the calculation engine.

➤ To create a formula using the formula bar:

1. Open or create a report.
2. Insert a grid.
3 Insert a formula row or column and select the formula cell. Alternatively, you can apply a formula to the entire column or row by selecting it.

Note: When you use a cell formula within a row or column formula, the cell formula should reference the cells exactly (intersection) and not just the row and column.

4 Create a formula for the cell using one of the following methods:
   - Type the formula directly into the formula bar.
   - Use the drop-down menu on the formula bar, shown below, to select the function to be used in the formula. For guidance on creating formulas, see “Defining Formula Rows or Columns” on page 106. For a description of available mathematical functions, see “Mathematical Functions” on page 174.

   Tip: When using the formula bar, you can access Function help by clicking the Help button.

5 Validate the formula using the check mark, ✅, on the formula bar.

Editing Formulas Using the Formula Bar

You can use the formula bar to modify a formula within a cell in a grid. The ability to edit a formula with ease is essential, as formula syntax is likely to change for reporting purposes.

To edit a formula using the formula bar:
1 Open a report that contains a grid.
2 Select a formula cell, formula row, or formula column that contains or uses a formula. The associated formula is displayed in the formula bar.
3 Modify the formula in the formula bar. You can use the drop-down list of functions, to select a formula.
4 Validate the formula by clicking the check mark, ✅, in the formula bar.

   The validated formula is displayed in the formula bar.

Deleting Formulas

You can delete a formula that you no longer want to use.

To delete a formula:
1 Select the formula row or column.
2 From the menu bar, select Edit, and then Delete.

Note: The Delete button, ❌, does not delete the formula, but rather cancels changes when editing the formula.
Modifying Member Selection Using the Formula Bar

When working with members in a grid, you can use the formula bar to modify the selected members.

The following example shows a textual modification of member selection using the formula bar.

**Note:** The entry of member names in the Formula Bar is not case-sensitive.

First, you select the heading cell in the grid. The dimension is displayed in the button on the Formula Bar. Next, you can use the Formula Bar to dynamically enter the member function for the selected dimension.

**Note:** A member function can be entered dynamically on data heading cells. For more information on member functions and syntax, see “Assigning Members Using Functions” on page 32.

To modify member selection using the formula bar:

1. Open or create a report.
2. Select a heading cell of a data row or column in the grid.
3. On the formula bar, click the dimension button, then use the **Select Members** dialog box to select or remove members.
   
The new members are displayed in the grid. For information on using member selections, see Chapter 2, “Defining Members.”

   **Note:** When you select a member in the grid, the dimension button on the formula bar is activated. You must indicate a member that exists within the current dimension.

   - Modify the member name(s) in the text box on the formula bar.
   - Dynamically enter a member function for a dimension using the formula bar.

   **Note:** You cannot combine the use of numerical functions with member functions; that is, `Sum(Children of (Product) (Inclusive))`

4. Click to accept the member entry.
   
The grid is set to obtain data for the new member/member function.

### Accessing Property Sheet Features

When you select a report component, a corresponding property sheet is displayed on the right side of the design area. When you select another component, the category and the property sheet...
features change. For example, when you select a cell, the Cell Properties sheet is displayed. You can use the property sheet to make modifications to the selected component.

In general, the property sheet enables you to set various options for the currently selected report component. For more information, see Chapter 1, “Getting Started with Reporting Studio.”

Calculating at the Cell Level

A cell in a formula row or column can contain a formula. You can create a custom cell-level formula on cells in a formula row or column. When you select a formula cell, the Cell Properties sheet allows you to choose a calculation preferences:

- If the cell is an intersection of a formula row and formula column, you can choose whether calculation is based on the row formula, the column formula, or a new custom formula.
- If a cell is located in a formula row, you can choose whether calculation is based on a row formula or a new custom formula.
- If a cell is located in a formula column, you can choose whether calculation is based on a column formula or a new custom formula.

The following table shows an example of a cell that is calculated to display the % variance. Cell D4 displays the % variance between the sum of Jan and the sum of Feb, specifically, \(((356-350)/350) \times 100\) = 1.71%. A cell level formula gives the intended result by placing a custom formula in the cell D4. The formula needed to get the correct result is \((([B]-[A])/[A]) \times 100\).

<table>
<thead>
<tr>
<th>Table 15</th>
<th>Reason for Using Cell Level Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Total Corp</td>
<td>Jan</td>
</tr>
<tr>
<td>1 Cola</td>
<td>100</td>
</tr>
<tr>
<td>2 Root Beer</td>
<td>200</td>
</tr>
<tr>
<td>3 Grape</td>
<td>50</td>
</tr>
<tr>
<td>4 [1].sum</td>
<td>350</td>
</tr>
</tbody>
</table>

To calculate at the cell level:

1. Open a report.
2. Select a formula cell or cells in which you want to enter a formula.
3. In the Calculation order area of the Cell Properties sheet, select one of the following options:
   - Select Use the row formula to apply the the row calculation to the cells.
   - Select Use the column formula to apply the column calculation to the cells.
   - Select Use a custom formula to add a new formula, perform the following actions:
     a. Go to the formula bar and select a function from the function drop-down menu.
b. With your cursor in the formula text box, complete the formula.
c. Select File, and then Print Preview to see the results of your formula.

**Note:** You can select multiple contiguous formula cells in a formula row or column to change the calculation order.

### Using Text Functions to Display Information

You use text functions to display report information. You can display information about a report, such as its description, information about a database that is associated with a grid, and a piece of data from your database connection.

You can use text functions in text boxes, data rows or columns, formula rows or columns, or Pages headings. Enclose the formula in double angle brackets (<<) and (>>) to distinguish different text functions. The font for the text function can be formatted like regular text. The text functions and their syntax are defined in “Text Functions” on page 194.

**Note:** You must format the entire text function, including the angle brackets, with the same font properties in the text box.

➤ To use text functions to display information:

1. Open a report, then select one of the following items:
   - Select a text box.
   - Select a heading cell, then Custom Heading from the property sheet.
   - Select the Pages drop-down list in a grid, then Custom Heading from the property sheet.

2. Click the Insert Function button. The Insert Function dialog box is displayed.

3. To select the desired function, perform an action:
   - Double-click the function in the Available Functions list to add it to the Selected Function list.
   - Select a function from the Available Functions list and click the Add button, \( \text{Add} \), to add it to the Selected Function list.

4. In the Select Function area, make your changes to the function’s parameters, then click OK.

**Note:** A report designer has the option to enable or disable error messages for text functions that are displayed in the grid or text object. The default is to display error messages. To disable this feature, from the Grid Properties sheet or the Text Properties sheet, deselect the Show Text Function Errors option.

5. Select File, and then Print Preview to run the report and display the data produced by the function.
Freezing Grid Headers

You can select a grid for which you want the row and column heading labels to display when scrolling an HTML report in Workspace.

To freeze the row and column headings of a grid:

1. Select the grid for which you want to freeze the row and column headings.
3. To unlock rows and columns, select the grid and clear Freeze Grid Headers.

Note: To view the report within the entire area, you must expand the grid size in Reporting Studio.

Browser Behavior When Freezing Grid Headers

Note the behaviors specific to using Internet Explorer and Firefox:

- **Scrolling.** In Internet Explorer, the row and column headings are locked; in Firefox, only the column headings are locked.

- **Grid height and width.** The height and width set by the designer is visible in HTML preview for both Internet Explorer and Firefox. Scrollbars are only shown when all rows or columns do not display. When Freeze Grid Headers is turned on, the autosize property is ignored.

- **Horizontal scrolling.** When the width of the grid exceeds the width of the design, the horizontal scrollbar is shown. However, the behavior of the horizontal scrollbar is different in the two browsers. In Internet Explorer, the row headings are adjusted so that they are always shown; in Firefox, the horizontal scrollbar will scroll all the columns, including the headings.

Note: When using both “Freeze Grid Headers” and “Row Headings Before Column X”, if X is any column greater than A, the horizontal scroll bar is not displayed.

- **Vertical scrollbar position.** In Internet Explorer, the vertical scrollbar is shown at the rightmost position in the table. In Firefox, the scrollbar appears underneath the final column heading, cutting into the final column. To accommodate this behavior, an additional column is added to the grid so that the last column is not covered up by the scrollbar.

- **Scrolling speed.** Scrolling speed is faster in Firefox because the scrolling is built in and does not require JavaScript as Internet Explorer does.
Every data value in a report is derived from the intersection of a member from each dimension in an Essbase, Planning, or Financial Management database connection. Financial Reporting enables a designer to place these dimensions on the report grid or user point of view (POV). Report viewers can change the member selected for dimensions on the user POV. This enables report viewers to customize the reports to fit their needs. The user POV can also be used in books. For more information on selecting the user POV as a member in the book POV, see Chapter 13, “Working with Books and Snapshot Books.”

In a report, the member specified for a dimension in the user POV is used for all grids with that dimension on the user POV. For example, if you select the Budget member for the Scenario dimension, all grids in the report with scenario in the user POV use Budget when you retrieve data. The user POV is not displayed if all of the dimensions are placed on the grid.

The following criteria are used to show dimensions on the user POV bar:

- Dimensions not currently defined on a row, a column, or a page
- Dimensions on a row, a column, or a page that are flagged for the current POV
- Dimensions without a member selected on a grid POV
- Attribute dimensions dragged to the POV area in the Dimension Layout dialog box

### Setting Up the User POV from the Report Designer

You can set up the user POV for a database connection while designing reports. This limits the members available on the user POV for that type of database connection. The user POV setup specified remains in force for all grids that uses that type of database connection. For more information on member selection, see Chapter 2, “Defining Members.”

For information on setting up the user POV in the Report Designer workspace, see *Oracle Hyperion Enterprise Performance Management Workspace Administrator’s Guide*.

To set up the user POV in the Report Designer:

1. Open a report that contains a grid for which you want to set up the POV.
2 Select Task, and then Set Up User POV.

3 Select Any Member or A Member from List for a dimension in the Report Viewers May Select column.

4 When the Report Viewers May Select column is set to A Member from List, click 📊. The Select Members dialog box is displayed.

5 Select a member from the Select Members dialog box, and then click OK.

Note: For more information on selecting members, see Chapter 2, “Defining Members.”

6 Select the Show Dimension Name and Show Member Name check boxes to display the dimension name with the member name.

7 Select the Show Alias check box to display alias names, then select an Alias Table from the From Alias Table drop-down list.

Note: If Financial Management is the database connection, Show Descriptions is displayed instead of Show Alias.

8 Click OK.

### Selecting Members for the User POV from the Report Designer

The user and grid POV enables report viewers to customize reports by selecting members for dimensions. The following rules apply:

- Only one member can be selected for a dimension on the POV.
- The default member for a dimension is the top member.
- For an attribute dimension, you can specify None. In this case the attribute dimension is not used in the report.

To select a member for the user POV:

1 Open a report with dimensions displayed in the user POV.

2 On the user POV, click the dimension for which you want to select a member. The Select Members dialog box is displayed.

3 Select the member to run for this report, then click OK. The member is displayed on the user POV.

4 Optional: If you do not want the attribute dimension to be used in the report select None.
Providing Expansions and Related Content includes the following topics and tasks:

*Expansions* and *Related Content* are features that enable the designer to specify detailed levels of data in a report. You can design your report to allow end users to view detail data residing in the same report using Expansions or in a linked report using Related Content.

For example, you might want to provide the detail of a Market dimension by displaying the areas that comprise a market, such as East, West, North, and South.

- **Using Expansion.** You can set up viewing of same-report detail data by enabling expansion of summary rows and columns. A report set up with expansion enables end users to select a summary-level row or column and view its related detail in the same report.

- **Using Related Content.** You can select a cell, row or column labels, the entire grid, charts, images and text boxes, and then set up links to objects in Financial Reporting, Web Analysis, and URLs. A report set up with Related Content enables users to move between documents in different applications.

You can view a report that has expansion and Related Content enabled during the design phase by selecting File, then Web Preview.

**Note:** The report designer must save the report as a Report type to view Related Content in the Report Designer workspace. Reports saved as snapshots lose this capability.
Setting Up Expansions to Access Detail Data Within Reports

You can design a report to display detail rows and columns of a grid at the request of the end user. You do this by enabling the expansion of a dimension that has parent-child relationships among its members. For example, if you enable the Time dimension to expand, users can expand a row that displays quarters into months.

Users access detail rows and columns by selecting summary rows and columns from reports that are displayed.

To set up access to detail data within a report:

1. Open a report.
2. Select the row or column headings for which you want to enable expansion.
3. Select the Allow Expansion check box in the Heading Row or Heading Column Properties sheet.

Note: When Hybrid Data is available for a base-level member in Essbase, an expansion arrow is displayed next to the base-level member and you can expand into Hybrid Data. Expansions are not allowed on rows with line-item-detail or supporting detail, or on columns containing a row with line-item-detail.

4. Double-click the dimension to select members to expand. For more information on selecting members, see Chapter 2, “Defining Members.”

Positioning Expansion Rows and Columns

You can specify whether expanded items are displayed before or after the dimension’s summary-level row or column.

To position expansion rows and columns:

1. Open a report.
2. Click the grid for which you want to specify expansion positions.
3. On the Grid Properties sheet, select the Position category.
4. Do one of the following in the Position of Expansions box:
   - To display expansions after the summary-level row or column, click After Parent. This is the default.
   - To display expansions before the summary-level row or column, click Before Parent.

Attaching Cell Documents in a Report

Cell documents that are attached at the data source can be accessed and printed to reports. The attached documents, (Microsoft Word, Excel, or PowerPoint documents, PDF or .txt
documents) print at the end of a PDF or snapshot report. The documents is accessed in the HTML client when drilling into a cell through related content. In addition, footnotes can be inserted into a text object or text cell of the report that lists information about the attached cell documents by selecting the ListOfCellDocuments option from the property sheet.

To include cell documents in a grid’s report, they must exist in Essbase (Linked Reporting Objects) or Financial Management (cell attachments). The data cell must be selected with the RetrieveCellDocument option. All cell documents have a POV associated with the data cell to which they are attached.

**Note:** In Essbase, Linked Reporting Objects are attached through the Excel Add-in. In Financial Management, cell documents are attached through the Financial Management web client. For more information, refer to the respective product’s documentation.

A data cell can contain multiple cell documents. If a data cell contains cell documents in the Financial Management or Essbase data source, a designer can extract those documents, based on the given cell’s POV. Cell documents can be attached to reports contained in books and snapshot books.

**Supported Microsoft Office File Types**


- Microsoft Word files (*.doc, *.docx, *.rtf)
- Microsoft Excel files (*.xls, *.xlsx)
- Microsoft PowerPoint files (*.ppt, *.pptx)
- PDF files (*.pdf)
- Text files (*.txt)

**Supported Print and Print Preview Options**

The following table describes the support matrix of cell document output for Reporting Studio and Financial Reporting.

<table>
<thead>
<tr>
<th>Report Output Type</th>
<th>Support</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reporting Studio Print and Print Preview</td>
<td>None</td>
<td>- The Print or Print Preview commands do not print cell documents.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- The ListOfCellDocuments function returns the FileName and Description parameters, but not the PageNumber parameter.</td>
</tr>
<tr>
<td>Report Output Type</td>
<td>Support</td>
<td>Comments</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------</td>
<td>----------</td>
</tr>
</tbody>
</table>
| Financial Reporting Export to PDF — in Workspace | Microsoft Office file types, PDF and .txt files | ● A PDF file is generated. Only supported file types are included. Non-supported cell documents are not included in the PDF.  
● Ignored documents are logged in the FRPrintSrv.log file.  
● The ListOfCellDocuments function returns the FileName and Description parameters for all cell documents, including non-supported documents. The PageNumber parameter is only displayed for the supported Microsoft Office documents. |
| Financial Reporting PDF Preview — in Workspace | Microsoft Office file types, PDF and .txt files. | ● The PDF file is displayed in EPM Workspace.  
**Note:** When using Microsoft Windows 8, ensure that Adobe Reader is the default choice for displaying PDF files.  
● A PDF file is generated. Only supported file types are included. Non-supported cell documents are not included in the PDF.  
● Ignored documents are logged in the FRPrintSrv.log file  
● The ListOfCellDocuments function returns the FileName and Description parameters for all cell documents, including non-supported documents. The PageNumber parameter is only displayed for the supported Microsoft Office documents. |
| Financial Reporting HTML Preview — in Workspace | All file types | ● All cell documents in HTML preview are hyperlinks. When you click a cell document hyperlink, the Related Content dialog is displayed. All cell documents associated with the corresponding data cell are displayed.  
● When you click a document in the Related Content dialog box, the browser’s File Download dialog is displayed where you can choose to open or save the cell document. When you choose Open, the cell document displays in a new browser. The associated file type for the cell document must be installed on the user’s computer.  
● Depending on the user settings on the browser, the cell document may be opened automatically in a new browser window.  
● The ListOfCellDocuments function returns the FileName and Description parameters, but not the PageNumber parameter. |

In Reporting Studio, you must first save the reports as a snapshot if you wish to print-preview the contents of cell documents.

Reports containing cell documents can be saved through Reporting Studio as Snapshot output. You can use the Save option to save the report as a snapshot in the repository. The cell documents are attached to the end of the report.

In Workspace, you can view reports containing cell documents in PDF or HTML output:

- PDF output. You use the Export option to save the report in a PDF file outside the repository. The cell documents are attached to the end of the report.

- HTML output. You use the Save or Save As option to save the report into the repository. The cell documents are stored with hyperlinks to the report.

> To attach cell documents:

1. **In Reporting Studio,** create or select a report that uses a Financial Management or Essbase data source in the database connection.
2 Select a data type cell in a grid. On the **Cell Properties** sheet, select the **Retrieve Cell Documents** check box.

**Note:** For optimum retrieval performance, select Retrieve Cell Documents only for cells with attachments to include in the report.

3 If cell documents exist for the selected cell, they are appended to the report when printed.

**Considerations**

Please make note of the following items when working with Cell Attachment:

- Include the correct supported extensions for files.
- Files which are corrupt or improperly uploaded to the ADM provider (through Financial Management client or Essbase Add-in for Excel) will not print.
- We do not support Microsoft files containing macros or that trigger UI components requiring user input. Some Microsoft files with macros perform operations when the file is opened which include the launching of dialogs requiring user input. Files that display dialogs or UI components requiring user input may not be available to print. Additionally, this may also hold up the completion of the PDF generation process. The server’s administrator may need to kill the application through Task Manage to resume printing activities. Because the application is launched under the SYSTEM account, it may not be visible since the process was not initiated by the login account.
- Documents containing file extensions that are not properly registered to print will fail.

**Printing Cell Documents**

You can use the Page Setup dialog box - Page Tab, to print cell documents with consecutive page numbers. The starting page of the first cell document is determined by the length of the report. Each document prints on a new page. If you do not select to print consecutive numbers on your cell documents, the page numbering reverts to the setting of each document.

For information on setting your print option for cell documents, see “Changing the Page Setup” on page 171.

**Note:** Text files are inserted into a Word template before the actual printing occurs. For information on customizing the template, see the Chapter 14, “Administrative Information for Financial Reporting” chapter.

**Note:** In Financial Management, when you upload files for attachments to cell, the file name truncates to 20 characters. You can change the document’s name. Make sure to retain the file name extension (.doc, .xls).
Printing a List of Attached Cell Documents

You can use the ListOfCellDocuments text function to display a list of all cell documents and attributes such as description and start page number that are retrieved in the report. The function can be entered into a text object, text cell, or text cell heading.

Setting Up Links to Related Content

Related Content allows a designer to set up links to Financial Reporting documents, as well as Oracle Hyperion Interactive Reporting and Oracle Hyperion Web Analysis documents, and custom URL links. A user, when viewing a report in HTML in Workspace, can drill into a related content link and execute the linked report or URL. Related Content is supports in dynamic and snapshot reports and books.

Applying Related Content

You can apply Related Content to these areas:

- Steps for applying Related Content to Oracle Hyperion Financial Data Quality Management:
  - You have to create a Planning data source where the data in those data sources was loaded via ERP (Integration Adapter for Oracle Applications) from the FDM source data. Then in the Financial Reporting report, you create a related content link on a data cell/row/column that drills to the source data in FDM.
  - When you run the report in Workspace, the HTML Viewer displays the report and the Related Content cells are displayed with an underline, if you click on the link, then you will drill down to the FDM Landing page where you can view the source data.

- Steps for applying Related Content to FDM: You have to create a Financial Reporting report that connects to an Financial Management, Essbase, or Planning data source where the data in those data sources was loaded via ERP (Integration Adapter for Oracle Applications) from the FDM source data. Then in the Financial Reporting report, you create a related content link on a data cell/row/column that drills to the source data in FDM. When you run the report in Workspace, the HTML Viewer displays the report and the Related Content cells are displayed with an underline, if you click on the link, then you will drill down to the Oracle Hyperion Financial Data Quality Management Landing page where you can view the source data.

  Note: Related content drill-down works the same using FDM and FDMEE

- Related Content can be specified for any cell within a grid object (data cells, text cells, formula cells, heading cells), text object, image object or chart object. When related content is any cell except a text cell, the point of view (POV) included in the related content link is limited to the unambiguous members for that cell. For examples of related content results in a grid, see “Examples of Related Content Results when Specified in Grid Cells” on page 131.

- Related content can be specified conditionally through the Conditional Format dialog on any grid object cell. On the Replace tab of the Format Cells dialog, you can specify Related
Content links. For example you can enter a condition: “if Market member name is “East”, link to ReportABC; if Market member name is “West”, link to ReportXYZ. See “Applying Conditional Formatting to Grids” on page 158.

- Related Content can be specified on Image, Chart and Text objects. The link is applied to the entire object. Chart and Image objects with Related Content can be mouse-clicked to point to the Related Content link. For Text objects with related content, the entire textual content is marked as a hyperlink. When clicked, the Related Content link is followed.
- When a data cell has a related content link because of a cell document, if that cell is referenced through a <<GetCell()>> or the <<CellText()>> function, the <<GetCell()>> or the <<CellText()>> value will also have the related content link. The behavior of a related content link in a cell document is the same as a related content link designed in a Financial Reporting report. To view more information about <<CellText()>>, see “CellText” on page 203.
- URL strings in Text objects and Grid object text cells are replaced with a “[Link]” hyperlink to that URL. URLs must begin with “http”, “https, or “ftp” in order for a URL to be converted. For example, if you type “This is a link to Google http://www.google.com” the result is “This is a link to Google [Link]”.

**Note:** Text object and Grid object text cells are evaluated after any <<CellText()>> and <<Annotation()>> functions are evaluated. So if cell text or annotation functions contain URLs, they are processed into hyperlinks as appropriate.

- When you select related content for a grid, you can also specify dimension descriptions for the related content link dimensions. The textual information is displayed in the book Table of Contents and navigation path (breadcrumb). The description is defined using the <<Value()>> function in the Add Related Content dialog box. See “Modifying the Properties for Related Content” on page 132.

You can design a report that provides access to another report or a URL through a hyperlink. You can create numerous links through a data cell, text cell, formula cell or heading cell in the report design. Links can be created to reports through Reporting and Analysis servers, if they are registered through Shared Services or on the RelatedContentURLS property that you can set in the JConsole.exe file. For information on JConsole, see the “Property Information” topic in the Oracle Hyperion Financial Reporting Administrator's Guide. End users can click a cell in HTML preview of a Financial Reporting report in Workspace with Related Content links, then select a linked report from a Related Content menu.

When using Related Content on a report, the system sets the point of view (POV) of the report to that of the selected cell. For example, if the Gross Margin row of the Canada column is selected, the POV for the account is set to Gross Margin and the POV for the entity is set to Canada before the linked report is displayed.

**Note:** The POV context is not passed when you link to books through Related Content.
You can define Related Content for data cells within a row or column. You can specify the same report for a group of cells by selecting the desired cells in the grid and then defining the Related Content. Also, based on the type of Related Content selected, you can modify properties, such as their label and URL and default display options.

Object-level security is not enforced when you select Related Content, which enables you to select any report among the entire contents of each repository. However, object-level security is enforced when the report is requested for viewing.

To create a link to Related Content:

1. Select a report for which you want to specify Related Content.
2. Select the rows, columns, cell object, grid, chart, image or text object for which you want to provide a link to Related Content.
3. From the Property sheet, select the Add Related Content check box, then click the Setup button. The Add Related Content dialog box is displayed.

Each Reporting and Analysis server repository is listed as a node in a tree view. The remote servers are obtained from the RelatedContentURLS property that you can set in the JConsole.exe file.

Note: For information on JConsole, see the “Property Information” topic in the Oracle Hyperion Financial Reporting Administrator's Guide, or Application Management.

4. You can link a folder for Related Content, which enables an end user to select from the complete contents of the folder, or expand the contents of a folder and select a report. To make a selection, perform any of the following actions:

   • To select a folder, highlight the folder in the Available list, then click the arrow button to move the folder to the Selected Related Content area.
   • To select a report object, expand the folder by clicking the node (+), highlight a report object, then click the Add button, to move the report to the Selected Related Content area.

5. You can modify the properties of the selected report object while you add the link or after you add the link:

   • To modify properties while selecting a report object, select the Show Properties dialog when adding content from Hyperion sources check box. As you select a report object to link, the Related Content Properties dialog box is displayed.
   • To modify properties after selecting report objects, highlight the report object in the Selected Related Content list, then click the Properties button.

   For information on changing properties, see the “Modifying the Properties for Related Content” on page 132.

   • To see all repository object types in the Financial Reporting Related Content dialog, select Show all object types. When this option is not selected, only Financial Reporting objects are shown.
Note: This option simplifies book functionality

6 Select as many additional links as needed.

Note: For information about changing the properties of the links, see “Modifying the Properties for Related Content” on page 132.

7 To change the position of a report object within the Selected Related Content list, select the object, then click the Up arrow or Down arrow to move the report object up or down, respectively.

8 Click OK.

Examples of Related Content Results when Specified in Grid Cells

The following report image has related content on all cells. Measure and Market are on the point of view (POV), Product aliases are on the rows, Year and Scenario are on column 1 and 2. Column 3 is a formula column and column 4 is a text column.

<table>
<thead>
<tr>
<th>Measures</th>
<th>Market</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profit</td>
<td>East</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>First Qtr</th>
<th>Second Qtr</th>
<th>Formula B - A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Colas</td>
<td>2.747</td>
<td>3.352</td>
<td>605</td>
</tr>
<tr>
<td>Root Beer</td>
<td>562</td>
<td>610</td>
<td>48</td>
</tr>
<tr>
<td>Cream Soda</td>
<td>591</td>
<td>922</td>
<td>331</td>
</tr>
<tr>
<td>Fruit Soda</td>
<td>1.480</td>
<td>1.615</td>
<td>135</td>
</tr>
</tbody>
</table>

- If you execute related content on the row headings (Colas, Root Beer, Cream Soda, and Fruit Soda), the POV included in the related content link includes all the dimensions except Year and Scenario. You cannot specify a Year or Scenario member when clicking on the row heading since it is ambiguous which member you want.
- If you execute related content from the formula cells, Product, Measures and Market are included; Year and Scenario are not included.
- If you execute related content from the formula heading cell (Formula B-A), only Measure and Market from the POV are included.
- If you execute related content from the “Link to rc900” text cell, only Measure and Market from the POV are included.
- If you execute related content from the “Click to Show PDF” text cell, Product, Measures, and Market are included; Year and Scenario are not included.
Changing the Related Content Server for Multiple Reports or Grids in the Repository

After moving content from another server, you can use Workspace to change the server for selected reports, grids, or both in the repository by specifying a new URL or a substitution variable. You may want to change the related content servers; for example, you might do this when moving your reports and grids from a development server to a production server.

For instructions on changing the related content server in the repository, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

Removing Access from Related Content

You can remove a link from your report to deny Related Content to end users.

To remove access to a linked report:
1. Select a report that contains links to Related Content.
2. Select the rows, columns, or data cells from which you want to remove a link for Related Content.
3. From the property sheet, select the Add Related Content check box, then click the Setup button. The Add Related Content dialog box is displayed.
4. Highlight the report object to remove in the Selected Related Content list and click the Remove arrow button.
5. Click OK.

Modifying the Properties for Related Content

You can modify properties of a selected report object. Based on the type of report object selected, you can modify such properties such as its label or URL, and you can set default display options.

To modify properties for linked objects:
1. Create a link to a related content in the Add Related Content dialog box. For information on creating a link, see “Applying Related Content” on page 128.
2. In the Add Related Content dialog box, select a report object in the Selected Related Content list, then click the Properties button to display the Related Content Properties dialog box.

Related Content Properties Dialog box

The items in the Related Content Properties dialog box that are displayed for editing depend on the report type you selected. For example, for Financial Reporting, all display options are provided for a report, snapshot, book, and snapshot book. However, for a custom link, only the label and a URL fields are available.
The following items are available for editing:

- **Label**—Enables you to change the name of the report object link. You can apply the <<Value()>> text function to specify descriptions of the dimension values for the related content link. See “Specifying Related Content Link Descriptions” on page 133.

- **URL tabs**—Displays the URL of the linked object which may be modified if needed.

- **Display Options**:
  - In the Provide list. For Financial Reporting, select the formats that can be viewed by the end user (HTML and PDF). For Web Analysis, select whether the report can go to the Java Client, Workspace, or both.
  - In the Default list. Select the format to link with the report name as the primary selection. The format not selected is listed as the secondary selection.

**Tip:** Click Save As Default to update your current selections.

### Specifying Related Content Link Descriptions

In HTML output of a book, you may want to see the description of the related content results in the table of contents and navigation path (breadcrumb). In the Label field of the Related Content Properties dialog, you can include a text function, <<Value()>>, to specify the related content result you want shown. Using the image below as an example, where you have Measure and Market on the POV, Product on the rows (showing aliases), Year (with custom heading text) on the first column, and Scenario on the second column of a report result, if the description of the related content link is 
rc900<<Value(Product)>><<Value(Year)>><<Value(Market)>> the following will show in related content results of the book’s table of contents and navigation path:

<table>
<thead>
<tr>
<th>Measures/Market:</th>
<th>Profit</th>
<th>East</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fruit Soda</td>
<td>1,490</td>
<td>1,615</td>
</tr>
<tr>
<td>Cream Soda</td>
<td>591</td>
<td>922</td>
</tr>
<tr>
<td>Fruit Soda</td>
<td>562</td>
<td>610</td>
</tr>
</tbody>
</table>

- If you click on cell 3,B (value of 922), the text displayed is: rc900 CreamSoda, Second Qtr, East.
- If you click on cell 1,A (value of 2,747), the text displayed is: rc900 Colas, First Qtr, East.
- If you click on the “Fruit Soda” heading cell, the text displayed is: rc900 Fruit Soda, East (Year is not included since it is ambiguous).
Using Smart View for Office

Smart View provides a common Microsoft Office interface for Financial Reporting.

The centralized interface enables simultaneous use of multiple Oracle products, and improves integration with Microsoft Office. The Smart View implementation provides the following Workspace functionality:

- Exports the current page of the current data object to Excel, Word, or PowerPoint
- Exposes Financial Management and Essbase functions in Excel, Word, and PowerPoint content
- Exposes Essbase functions in Excel, Word, and PowerPoint content.
- Notifies you when you can upgrade to new releases of Hyperion Smart View.
- For more information on using the capability, see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide*. 

Note: The heading value of the appropriate dimension is shown, not necessarily the member name. For example, “Cola” is shown rather than “100”, and “First Qtr” is shown rather than “Qtr1”.

Note: If the dimension specified in the <<Value()>> function is not included in the POV of the related content link, the <<Value()>> text is replaced with an empty string (" ").
You can preview reports in Reporting Studio or from a web browser through EPM Workspace. You can also print reports from EPM Workspace. As you preview or print a report, the data is refreshed with new data, calculations, and formatting.

Snapshot reports contain data retrieved at the time the report is saved as a snapshot. The data in a snapshot report remains the same; changes in the database connection since the snapshot was last saved are not reflected when it is previewed or printed.

Note: For viewing and printing options provided through the Workspace. See Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

**Responding to Prompts When Previewing**

If a report is designed with prompts, those prompts are displayed when you preview the report. You must respond to the prompt and provide the requested information by selecting members from the prompt list. You can also edit the prompts manually. If the prompts contain alias names, you can edit the alias names by converting to member names. For more information on defining prompts, see Chapter 2, “Defining Members.”

To respond to a prompt when previewing:

1. From Studio Explore, open a report that has prompts.
2. Under the Selection column in the Respond to Prompts dialog box, perform an action:
   - If the member is known, enter the member name in the text box for the respective prompt. If the text box is disabled, the prompt contains alias names. To edit the text...
box, select the Edit Member Names check box. If multiple members are provided for the prompt, members must be separated by commas.

**Note:** Selecting the Edit Member Names check box displays the member names in the text box, not the alias names. Edit the member name associated with that alias.

- Click the Go to Member Selection button. The Select Members dialog box opens. The default member is listed in the right panel - selected area. Click the Add to Selected arrow and Remove from Selected arrow located in the center panel of the dialog to move members between the panels.

For more information on defining members, see Chapter 2, “Defining Members.”

3 Select OK.

4 Optional: To undo changes made to prompts, click Reset from the Respond to Prompts dialog box.

5 Click Run. The report or book is displayed.

### Previewing the Current User POV Settings

In Workspace, a user can preview the user POV before running a report and make changes to the user POV from. For example, it may save time for users to verify that the members on the user POV are appropriate before running the report instead of after the output is displayed.

For information on how a user can preview the user POV settings for a report or book from Workspace, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

### Setting Preview Preferences

Using Workspace, you can set preview preferences. You can preset how to view a report or book. The default settings for opening a report or book are PDF Preview or HTML Preview.

For detailed information on this option, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

### Previewing Reports in a Web Browser

You can use the Web Preview feature in Studio Explore or the Designer to preview reports in a web browser through Workspace.

You can use the Web Preview feature in Studio Explore or the Designer to preview reports in a web browser through EPM Workspace.

1 Open any type of report

2 Click the Web Preview button on the toolbar.

**Note:** EPM Workspace provides preview and printing options.
For information on viewing reports and books using Workspace, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

**Previewing Reports in Reporting Studio**

You can preview a report to view the contents before saving changes or printing. You can zoom in or zoom out, scroll through the report, or print the report. When you preview a report, the data in the report is refreshed. Data in snapshot reports are not refreshed.

Books are previewed using a web browser through Workspace. For information on previewing books and reports, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

➢ To preview a report:

1. From Studio Explore, highlight a report or snapshot.
2. Select File, and then PDF Preview.
3. Use Adobe Reader to navigate the PDF document, and then close Adobe Reader when finished.

**Previewing Snapshot Reports**

You view snapshot reports to display and print static reports. You can open a snapshot report from the Reporting Studio repository.

➢ To preview a snapshot report:

1. Select one or more snapshot reports from the repository
2. Select File, and then Print Preview

**Printing Tips for Report Designers**

Subtopics

- Positioning and Alignment
- Printing Text Boxes
- Printing Grids
- Printing Page Dimension Headings
- Printing Page Members
- Printing Headers and Footers

You can place report objects anywhere in a report. However, the printed results are determined by the report properties and data.
Note: When printing reports containing overlapping objects, objects may not print as displayed in the designer, requiring you to rearrange the objects in your report. For more information, see Chapter 4, “Building Reports.”

Positioning and Alignment
You can set relative positioning on autosized report objects. This is an advanced positioning mechanism that gives you precise control over how report objects are positioned relative to one another. If none of the report objects in your report is autosized, then the relative positioning is ignored.

If you set the position of a report object to relative, all report objects within the design shift so that the distance between the autosized report object and fixed report objects remain the same.

For example, GridA is autosized and has horizontal positioning set to Center, therefore, GridA is printed in the center of the page. ChartA is linked to GridA, and has its vertical position set to Relative and is placed one inch from the bottom edge of GridA. ChartA is printed one inch from the bottom of Grid A.

If the alignment for ChartA is not set to relative, it is printed on the absolute coordinates in the layout of the grid, so it might overwrite Grid A.

Note: Vertical and horizontal positioning are not maintained after changes are made to the report.

Printing Text Boxes
You can set the autosize property for a text box. When you print a report, the height of a text box increases or decreases to accommodate all text that is entered in the text box. The width of the text box does not change.

Note: When you autosize, the maximum number of lines for viewing is 27. To view all lines of text, deselect Autosize.

If the autosize property is not set when you print a report, the report is affected in these ways:

- The text box is printed at the height and width of the grid.
- Text that does not fit in the text box is truncated.

Printing Grids
You can set the autosize property for a grid. When the autosize property is set and you print the grid, the height and width of the grid increase or decrease to accommodate all data in the grid.
When the autosize property is not set, the grid is printed at the height and width specified. If a row or column does not fit on a page, it is printed on subsequent pages. The columns are printed first using as many pages as needed, followed by the rows.

You use the positioning properties of the grid to align the grid on the page. Manually positioning the grid to the right or bottom of the page does not ensure that the grid will print on the right or bottom of the report.

When a grid is set to autosize, and does not fit on one page, it is printed on subsequent pages. The columns are printed first, starting on the left margin of the next page, followed by the rows, which are printed starting at the top of the next page.

**Printing Page Dimension Headings**

You can specify a page dimension heading for a grid using the Page Properties sheet. When you print a report, the page dimension heading is printed every time the page member changes.

The following list describes how the page dimension heading is printed:

- The text is left justified at the left edge of the grid.
- The text is autosized vertically, and the width is set to the width of the grid.
- The heading is printed above the first row in the grid each time the page member changes.

**Printing Page Members**

When you print a report, a new page is generated every time a page member changes. Using the Page Properties sheet, you can specify how to print the page using one of the following options:

- **New Page, Top**—The new page is printed at the top of the page. Use this option when a chart is linked to a grid and you want to ensure the chart and grid print next to one another every time the page member changes.

- **New Page, Same**—The new page is printed at the grid location specified. This option is used for the following situations:
  - The grid has horizontal and/or vertical alignment properties set and you want the grid to print in the same position on each page. For example, if you set the grid position to Center/Middle, it is printed in the center of each page.
  - When a chart is linked to a grid and you want to ensure the chart and grid print next to one another every time the page member changes.

- **Same Page**—The new page is printed immediately after the data from the previous page.

**Note:** If a chart is linked to a grid, the chart is printed once; not every time the page member changes.
Printing Headers and Footers

You can place an autosized text box in the header. However, if the text box height expands beyond the header, it prints beyond the header boundary.

You can place an autosized text box in the footer. However, if the text box height expands beyond the footer, the text is truncated.

Note: If an object is placed near the top or bottom of the page, it might overlap the header or the footer.
Formatting reports enables you to specify how report components, such as text boxes, grids, images, and charts, appear in your reports. For example, you can specify fonts and add borders. When you format a report, you define the properties of the components. The formatting you apply affects the printed and online report presentation.

Note: Each object has default formatting properties. You can use the default format properties if you do not want to format your report.

For general information about report objects, including overlapping objects, see Chapter 4, “Building Reports.” For information about formatting charts, see Chapter 3, “Customizing Charts.”

**Setting Preferences**

The following preferences can be set for all reports:

- “Default Formatting Preferences” on page 141
- “User Preferences and Formatting Options” on page 142

**Default Formatting Preferences**

Default Formatting preferences specify default data formatting for all subsequently created reports. Options are organized by their ability to amend, format, or replace data returned from the data source.
### Table 16  Default Formatting Preferences

<table>
<thead>
<tr>
<th>Formatting Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Leading and Trailing Formatting</strong></td>
<td></td>
</tr>
<tr>
<td>Currency Symbol</td>
<td>Inserts the following currency formatting symbols into the Positive Prefix and Negative Prefix text boxes: Dollar ($), Cents (¢), Pound (£), Euro (€), Deutschmark (DM), Franc (F), and Yen (¥).</td>
</tr>
<tr>
<td>Positive Prefix</td>
<td>Character to precede positive numeric values</td>
</tr>
<tr>
<td>Positive Suffix</td>
<td>Character to follow positive numeric values</td>
</tr>
<tr>
<td>Negative Prefix</td>
<td>Character to precede negative numeric values. <strong>Warning:</strong> The minus sign (−) is the default prefix. Deleting the default prefix without replacing it causes negative values to display positively.</td>
</tr>
<tr>
<td>Negative Suffix</td>
<td>Character to follow negative numeric values</td>
</tr>
<tr>
<td><strong>Numeric Formatting</strong></td>
<td></td>
</tr>
<tr>
<td>Grouped Thousands Check Box</td>
<td>Numeric digits grouped by thousands</td>
</tr>
<tr>
<td>Minimum Decimals</td>
<td>Minimum number of decimal places to display</td>
</tr>
<tr>
<td>Maximum Decimals</td>
<td>Maximum number of decimal places to display</td>
</tr>
<tr>
<td>Scale</td>
<td>Abbreviated values by tens, hundreds, thousands, ten-thousands, hundred-thousands, millions, and billions.</td>
</tr>
<tr>
<td>Use Negative Color Check Box</td>
<td>Negative numbers are signified by a selected color</td>
</tr>
<tr>
<td>Select Negative Color</td>
<td>Color representing negative values</td>
</tr>
<tr>
<td><strong>Samples</strong></td>
<td></td>
</tr>
<tr>
<td>Update Samples</td>
<td>Updates the samples panel based on the most recent formatting selections</td>
</tr>
<tr>
<td>Replace Missing With</td>
<td>Replaces missing values with either a text string or zero</td>
</tr>
</tbody>
</table>

### User Preferences and Formatting Options

User preferences are global settings applied to new documents. User Preferences can be overridden by database connection formatting and document-based formatting.

Order of formatting precedence:

1. Options saved with documents
2. Options saved with the database connection
3. Options specified in the User Preferences dialog box

Spreadsheet user preferences and chart user preferences are identical to spreadsheet options and chart properties. They are applied only to subsequently created documents.
Formatting Text Boxes

Subtopics

- Specifying Fonts for Text Boxes
- Aligning Text in Text Boxes
- Adding Borders and Shading to Text Boxes
- Formatting Grids
- Expanding the View of a Grid
- Formatting Best Practices
- Formatting Numbers in Grids
- Hiding Rows, Columns, or Grids
- Controlling the Evaluation of Suppression in Hidden Rows or Columns
- Sorting Rows, Columns, and Pages

Specifying Fonts for Text Boxes

To specify a font:

1. Open a report and select text within a text box.
2. Select Format, and then Font.
3. Specify the font, font style, size, and color.
4. To place a line through the selected text, select Strikeout.
5. To place a line beneath the selected text, select Underline.
6. Click OK.

Aligning Text in Text Boxes

You can align text in text boxes horizontally. You can also increase or decrease the margin of text in text boxes using the indent option.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alignment</td>
<td>Select Left, Center, or Right to specify the horizontal alignment for text in text boxes. For example, if you select Center, the text is aligned in the center of the text box.</td>
</tr>
<tr>
<td>Indent</td>
<td>Select the number of spaces to indent the text to the right for left-aligned text.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>Indent functionality is limited to left-aligned text.</td>
</tr>
</tbody>
</table>

To align text in a text box:

1. Open a report and create a text box or select an existing text box.
2. In the Text Properties property sheet, select one of the following from the Position Object on Report - Horizontal drop-down list.
• None—Positions the printed text at its current location regardless of whether report objects above it or to its left change in size.
• Relative—Positions the printed text relative to report objects to its left.
• Left—Positions the text box on the left side of the report.
• Center—Positions the text box in the center of the report.
• Right—Positions the text box on the right side of the report.

3 Select one of the following from the **Position Object on Report - Vertical** drop-down list.
- None—Positions the printed text at its current location regardless of whether report objects above it or to its left change in size.
- Relative—Positions the printed text relative to report objects above it.
- Top—Positions the text box at the top of the report.
- Middle—Positions the text box in the middle of the report.
- Bottom—Positions the text box on the bottom of the report.

### Adding Borders and Shading to Text Boxes

- To add borders, shading, or both to a text box:
  1. Open a report and select a text box.
  2. In the **Text Properties** sheet:
     • To place a border around the text box, select the **Show Border** check box.
     • To add shading to the text box, select a shade from the **Shading** drop-down list.

### Formatting Grids

Reporting Studio enables you to perform a variety of formatting options when working with grids.

### Expanding the View of a Grid

You can expand the view of a grid to the size of the Report Designer workspace. This is especially helpful when your grid contains many rows and columns. The expanded view enables you to locate cells more easily.

- To expand the view of a grid:
  1. Right-click the title bar of the grid.
  2. Select **Zoom**.
  3. To return to the original view, right-click the title bar of the grid and deselect **Zoom**.
Formatting Best Practices

Cell formatting has precedence over row/column formatting and grid formatting. Cell formatting allows you to make exceptions to row/column formatting and grid formatting. Row/column formatting has precedence over grid formatting. Row/column formatting allows you to make exceptions to grid formatting. Therefore, the correct sequence in formatting a report is to first format the grid, then apply different formatting to rows/columns, then apply specific cell formatting.

Note: Conditional formatting takes precedence over all other formatting and replaces formatting previously set at the grid, row/column, or cell level. For example, if you use conditional formatting to change cell colors, yet want to retain a specific number of decimal places applied with regular formatting, you must specify both formats using conditional formatting.

Formatting Numbers in Grids

Subtopics

- Using Special Currency Formats
- Aligning Text in Grids
- Modifying Font Options
- Creating Custom Colors
- Adding Borders or Shading
- Replacing Text
- Inheriting Cell Formatting
- Copying Cell Formatting

You can format how numbers are displayed in a grid. You can specify the following options for an entire grid or for individual cells, rows, or columns:

- Positive number symbols, including prefixes, suffixes and currency symbols
- Negative number symbols, including prefixes, suffixes and currency symbols, and an option to display negative numbers in red
- Grouping and Decimal Symbols (using user preferences explicit for grid cell definitions)

To format numbers in a grid:

1. **Create a grid in a report in the Report Designer, or open an existing report containing a grid.**
2. **Select a cell.**
3. **Select Format, and then Cells.**
4. **Select the Number tab.**

You can view the results of your number format option selections in the Sample section at the bottom of the Number tab.
Optional: Specify a prefix or suffix for positive numbers by selecting a symbol from the Prefix or Suffix drop-down list, or type a user-defined symbol into the Prefix or Suffix fields. A prefix symbol is placed to the left of each positive value. A suffix symbol is placed to the right of each positive value.

Note: You can use a prefix or suffix to reverse the sign of an account member without inserting another row or column. When using conditional formatting, you specify an account row and set the condition. Then, in the Positive Number Symbols Prefix fields, enter the negative (-) symbol and click OK twice. The negative (-) symbol is displayed in front of the numbers for that row.

Optional: Specify a prefix or suffix for negative numbers by selecting a symbol from the Prefix or Suffix drop-down list box or type a user-defined symbol into the Prefix or Suffix fields. A prefix symbol is placed to the left of each negative value. A suffix symbol is placed to the right of each positive value.

Optional: To display negative values in red, select the Display in Red option. By default, this option is deselected.

Select the number of decimal places to apply to a number from the Decimal Places drop-down list box. The default ranges are from 0 to 9.

Select one of the following for Grouping and Decimal Symbols:

- Defined by Preferences to use the symbols specified in the user Preferences dialog box, enabling users to determine which symbols are used.
- Defined in Report to specify thousands and decimal separators for the current report:
  - Select a character for separating thousands in values from the Thousands Separator drop-down list. For example, select comma (,) to display the value (1,000), or select period (.) to display the value (1.000). The options are: comma (,), period (.), underscore (_), and Space. Alternatively, you can specify another symbol in the drop-down by highlighting the field and typing another symbol.
  - Select a character to represent decimal points (for example, 900.06) from the Decimal Separator drop-down list box. The options are: comma (,), period (.), underscore (_), and Space. Alternatively, you can specify another symbol in the drop-down by highlighting the field and typing that symbol.

Select or enter a number from the Scale Value By drop-down list box by which to multiply values for scaling purposes. The default scaling option is 1, which does not scale values. If you select .001 instead, the value 2,000 is displayed as 2. The options are: percentages, tens, hundreds, thousands, ten thousands, hundred thousands, millions, or billions. Alternatively, you can specify an amount in the drop-down box.

Click Apply to apply the number formatting you specified for the selected cells, and then Click OK to save the number formatting and close the dialog box.

Using Special Currency Formats

Some currencies use specialized formats for their currency symbols. This topic provides the information necessary to insert these special characters in reports.
To add a special currency symbol, for example the Turkish lira symbol, to reports, the font used for that symbol must be installed on your local machine and the server to which you connect. See "Oracle Enterprise Performance Management System Installation and Configuration Guide" for font installation instructions.

To add a special currency symbol to reports:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select a cell.
3. Select Format, and then Cells.
4. Select the Font tab, and then select the font for the currency symbol:
   - AbakuTLSymSans for the Turkish lira symbol
   - Rupee for the Indian rupee symbol
5. Click Apply to apply the font selection.
6. Select the Number tab.
7. In Prefix or Suffix, enter the character or keystroke sequence for the desired currency symbol:
   - Ctrl+Alt+T to add the Turkish lira symbol as the prefix or suffix for numeric values
   - The accent grave character (´) to add the Indian rupee symbol as the prefix or suffix for numeric values
8. Click Apply, and then click OK to apply the currency symbol as the prefix or suffix for the selected cells, save the font and symbol formatting and close the dialog.

Aligning Text in Grids

You can specify options for aligning and indenting text in selected cells. You can select options for horizontal and vertical alignment. You can also specify the number of character spaces to indent text from the margins of selected cells.

To align text in grids:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select the cells you want to format.
3. Select Format, and then Cells to access the Format Cells dialog box.
4. Select the Alignment tab to specify alignment for the selected cells.
5. From the Horizontal drop-down list, select an alignment option. The options are Left, Center, and Right.
6. From the Vertical drop-down list, select an alignment option. The options are Top, Middle, and Bottom.
7. To indent the heading and the associated text, data, or formula cells in the row, from the Indent value box, select the number of spaces to indent the values from the left cell margin.
Note: This control is enabled only if you selected Left from the Horizontal drop-down list.

8 To indent the row headings of each generation of a selected member:
   a. Confirm that you selected a row with the desired heading cell from the Indent Increases for Each Generation By value box.

   Note: You can specify a reverse indentation (from right to left) by using a negative number in the format property.

   b. Select the number of spaces to indent the text. This number is added to the number in the Indent value box if it contains a value.

   Note: For Planning, select Show Supporting Detail to view detail items. Also, for all database connections including Planning, select a member relationship that returns members on multiple levels; for example: Children of ...(Inclusive), Descendants of..., Descendants of ...(Inclusive), Parents of...(Inclusive), Ancestors of, Ancestors of...(Inclusive), and AllMembers. This control is enabled only if you selected Left from the Horizontal drop-down list.

9 Click Apply to apply the text alignment options to the cells.

10 Click OK to confirm the text alignment options you specified and close the dialog box.

Modifying Font Options

You can specify the font, font style, font size, font color, strikeout lines, and underlining for text in selected cells. A preview of the text displays in the dialog box as you modify the font options.

To modify font options:

1 Create a grid in a report in the Report Designer, or open an existing report containing a grid.

2 Select the cells you want to format.

3 Select Format, and then Cells to access the Format Cells dialog box.

4 Select the Font tab and perform one or more of the following actions:
   - Select a font from the Font list box.
   - Select a font style from the Font Style list box. The options are: Regular, Bold, Italic, or Bold Italic.
   - Select a font size from the Size list box.
   - Select the Strikeout check box in the Effects section to apply strikeout lines. For example, if you apply the strikeout option to a cell that contains the word “SubTotal,” the word is displayed with a line through the middle.
   - Select the Underline check box in the Effects section to underline text. For example, if you apply the underline option to a cell that contains the word “SubTotal,” the word is displayed with a line beneath it.
   - Click the Select button in the Effects section to select a color to apply to the selected cell.
You can select a color from either the Basic colors palette or the Custom colors palette.
To define a custom color, see “Creating Custom Colors” on page 149.

5. Click **Apply** to apply the font options you selected.

6. Click **OK** to confirm the font options you specified and close the dialog box.

**Creating Custom Colors**

Financial Reporting provides a palette of basic colors for text in reports and other artifacts. Using Financial Reporting, you can add to that palette by defining custom colors.

**Note:** Custom colors are not saved between Studio sessions.

To define a custom color:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.

2. Select the cells that you want to format.

3. Select **Format** and then **Cells** to access the Format Cells dialog box.

4. Select the **Font** tab, and then click **Select** in the **Effects** section to display the Color dialog box.

5. Select **Define Custom Colors**, and then specify Hue, Sat (saturation), Lum (luminescence), and percentage of red, green, and blue for the color.

As you specify the color properties, the Color/Solid box displays the current state of the color.

6. When you are done, select **Add to Custom Colors**, then click **OK** in the Color dialog box, and then **OK** again in the Format Cells dialog box to save the color definition.

To discard the color definition, click **Cancel** in the Color and the Format Cells dialog boxes.

**Adding Borders or Shading**

You can apply borders and shading to add emphasis to selected cells, rows, or columns in grids in your reports. You can specify a border style and then apply it to any sides of the selected area. You can also select a shading color to apply shading to your selection.

The following table describes the buttons for indicating border placement options for cells:

<table>
<thead>
<tr>
<th>Table 18</th>
<th>Border Placement Options</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Button</strong></td>
<td><strong>Task</strong></td>
</tr>
<tr>
<td><img src="image" alt="Border" /></td>
<td>Click to remove the borders from each selected grid, cell, row, or column.</td>
</tr>
<tr>
<td><img src="image" alt="Border" /></td>
<td>Click to place a border around the outside edge of each selected grid, cell, row, or column.</td>
</tr>
<tr>
<td>Button</td>
<td>Task</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td><img src="button1.png" alt="Border Button" /></td>
<td>Click to place a border along the top of each selected grid, cell, row, or column.</td>
</tr>
<tr>
<td><img src="button2.png" alt="Border Button" /></td>
<td>Click to place a border along the bottom of each selected grid, cell, row, or column.</td>
</tr>
<tr>
<td><img src="button3.png" alt="Border Button" /></td>
<td>Click to place a border along the left side of each selected grid, cell, row, or column.</td>
</tr>
<tr>
<td><img src="button4.png" alt="Border Button" /></td>
<td>Click to place a border along the right side of each selected grid, cell, row, or column.</td>
</tr>
</tbody>
</table>

To add borders or shading to cells in a grid:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select the items you want to format: a grid, or one or more cells, rows, or columns in a grid.
3. Select Format, and then Cells to access the Format Cells dialog box.
4. Select the Borders & Shading tab.
5. To specify border options, perform the following steps:
   a. In the Style box, click the preferred line style for the borders.
   b. Click one or more of the border placement buttons. For a description of the border placement buttons, see Table 18 on page 149.
6. To add shading, click the Select button and choose a color to apply shading.
7. Click Apply to apply the border and shading options you selected.
8. Click OK to confirm the border and shading options you selected and close dialog box.

**Replacing Text**

When working with cells in a grid, you can replace values in a cell with different text. This replaces all text in all selected cells with the replacement text you specify. You can also specify to include related content links that will display if certain conditions are met.

To replace a value in a grid selection:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select the cells you want to format.
3. Select Format, and then Cells to access the Format Cells dialog box.
4. Select the Replace tab.
5. Select the Replace value with check box to enable the text box, then enter the replacement text in the text box.
Note: If the check box is selected and no text is entered in the text box, the selected cells are displayed as blank in the report.

6 Click Apply to apply the text replacement to your selected cells.

7 Click OK to confirm the replacement of the specified text string.

Inheriting Cell Formatting

When formatting cells in a grid, you can inherit formatting from a cell and apply it to another cell. The following formatting options can be inherited in a grid: number, alignment, font, borders and shading, and replace.

Note: If you are working with a row and column template in the current grid, you must establish your format inheritance options before saving a row and column template. For more information, see “Creating Row and Column Templates” on page 89.

To inherit cell formatting:

1 Open a report that contains a grid, or create a new grid with formatted rows or columns.

2 Select the cells to which you want to apply inherited formatting.

3 Select Format, and then Cells.

4 Click the Inherit Formatting tab.

5 In the Inherit These Formatting Options area, expand the plus (+) symbols and click the box to select the formatting options to inherit, including number, alignment, font, borders and shading, and replace text.

Note: To select all formatting options, click Select All. To deselect all formatting options, click Unselect All.

Tip: To specify the row or column from which you want to inherit formatting, you must first specify the options you want to inherit (number, alignment, font, borders and shading, and replace).

6 In the Inherit formatting from area, select the row or column from which to inherit formatting, using the up and down arrows. Choose from the following inheritance options:

● Select the current row and another column

   Tip: Use the up and down arrow keys to make your selection.

● Select the current column and another row

● Select another row and another column
Note: You can only inherit from a cell that does not have previously defined inheritance. For example, if you format column A and then inherit formatting into column B, you cannot then inherit formatting from column B into column C. You can, however, format column A, select columns B and C, and inherit formatting for column B and C from column A.

7 Click Apply.
8 Click OK.

Note: If you have inherited formatting and then modify the format of the cell to which you inherited formatting, the inheritance you established no longer applies. For example, if you inherit the font name property in cell A4 from cell B4 and you subsequently modify the font name in cell A4, the format inheritance no longer exists. This breaks the formatting link that was previously established between the two cells.

### Copying Cell Formatting

You can copy cell formatting from a cell or range of cells and apply it to a destination cell or range of cells in a grid using Format Painter.

Note: Format Painter does not copy inherited formatting or conditional formatting.

➢ To copy cell formatting:

1 Create a grid in a report, or open an existing report containing a grid.
2 Format a cell or range of cells using grid formatting. For more information, see “Formatting Grids” on page 144.
3 Select the cell or range of cells containing the formatting to copy.
4 Click .
5 Select the destination cell or range of cells to apply the copied formatting. The formatting is changed when you release the mouse.

### Hiding Rows, Columns, or Grids

You can hide rows, columns, or an entire grid so that their values are not displayed when you print or view a report. You might want to hide a grid, for example, if you include a chart based on the grid data and want to display only the chart on the report. When the grid is hidden, the grid’s title bar changes from blue to gray.

The data and calculations in rows or columns are evaluated, regardless of the formatting that is applied to them.

➢ To hide a grid:

1 Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2 Select a grid.
3 In the Grid Properties property sheet, select Suppression category.
4 Select the Hide Grid check box.

➤ To hide a row or column:
1 Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2 Select a row or column.
3 Select the Hide Always check box.

Note: You can specify whether the data values of hidden row or column are taken into account during suppression. For information on this option, see Controlling the Evaluation of Suppression in Hidden Rows or Columns.

Controlling the Evaluation of Suppression in Hidden Rows or Columns

When evaluating suppression, you can include or ignore the values and calculations in a hidden row or column. This option is enabled only when the Hide Always check box is selected for the row or column.

➤ To specify if hidden rows or columns are taken into account during suppression:
1 Select a hidden row or column.
2 Perform an action:
   - To include values and calculations, deselect the Suppression Ignores Row or Column check box. This is the default setting.
   - To ignore values and calculations, select the Suppression Ignores Row or Column check box.

Note: Hidden cells are ignored when suppression is evaluated unless they are specifically referenced using conditional suppression. For more information, see “Conditional Suppression” on page 164.

Sorting Rows, Columns, and Pages

You can sort a range of consecutive rows or columns in a grid based on the values in a specified column or row. You can also sort columns, rows, and pages based on their headings. If you select a range of columns, the system sorts the columns based on the values in a specified row. If you select a range of rows, the system sorts the rows based on the values in a specified column. If the column or row by which you sort contains multiple members, the sort is based only on the values of the first member.
You can sort columns or rows in ascending or descending order. When you sort in ascending order the lowest number, the beginning of the alphabet, or the earliest date appears first in the sorted list. When you sort in descending order, the highest number, the end of the alphabet, or the latest date appears first in the sorted list.

You can sort only multiple ranges of columns or rows. Suppressed rows or columns are not sorted.

To sort a range of rows and columns:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.

2. Set up sorting for the grid:
   a. Select a grid to display the Grid Properties sheet.
   b. Select the General category.
   c. Select Sort, and then click Setup to display the Sort Dialog box.

3. From Apply Sorting to the, select Rows, Columns, or Pages.

   **Note:** For the Pages option, you can only select headings from the Sort by: drop-down menu. All other options are disabled.

4. For Rows or Columns, use From and To to select the range of columns or rows to sort.

   **Note:** When auto calculations is enabled, the outermost layers contains the grand total of all rows in the calculation. Sorting on the outermost layer is ignored. Also, when you select sort with auto calculation set on multiple layers, you must sort on all heading layers which have auto calculations.

5. In the Sort by area:
   a. From the first drop-down list, select the headings or row or column on which to base the sort, then select Ascending or Descending sort order.
   b. Optional: In the Sort by area, set up one or two additional sorts:
      i. For each additional sort, select the Then by check box.
      ii. From the second or third drop-down list or both, select the headings or row or column on which to base the sort.
      iii. For each additional sort, select the Ascending or Descending sort order.

6. Optional: To set up additional sorts:
   a. Click Add Sort. A tab is added to the Sort dialog box.
   b. In the tab, repeat steps 3 through 5 for the new sort.

7. Optional: To delete a sort:
   a. Click Delete Sort.
   b. From the drop-down list, select the sort to delete.

8. Click OK.
About Conditional Formatting

You can use conditional formatting to apply specified formatting to cells in a grid. For example, suppose you define the criterion “current cell value = 0” and then define a format to change the color of the cell font to blue if the criterion is met. If the data returned for this cell has a value of 0, then the font color is changed to blue.

**Note:** Conditional formatting takes precedence over all other formatting and replaces formatting previously set at the grid, row or column, or cell level. For example, if you use conditional formatting to change cell colors, yet want to retain a specific number of decimal places applied with regular formatting, you must specify both formats using conditional formatting.

**Note:** When you select multiple cells, the default formatting is derived from the upper left cell of the selected group of cells.

After you define the conditions and add the formats, you return to the report.

The current grid is displayed. Each cell with conditional formatting contains a colored dash (see cell D2 below) as an indication that a conditional format is applied to this cell.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>#</td>
<td>= #</td>
<td>= #</td>
<td>= #</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>#</td>
<td>= #</td>
<td>= #</td>
<td>= #</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>- #</td>
<td>= #</td>
<td>= #</td>
<td>= #</td>
</tr>
<tr>
<td>4</td>
<td>Product</td>
<td>#</td>
<td>= #</td>
<td>= #</td>
<td>= #</td>
</tr>
</tbody>
</table>

You can apply conditional formatting to enhance your reports. For example, you can underline any values greater than 1000. You can also set up more complex conditional formatting. For example, if the account type equals expense and the current cell value is greater than 1000, apply bold to the cell. In addition, if the value of a product cell is less than 1000, then set the background color of the cell to lime green. You can accomplish this type of formatting by specifying additional conditions and formats, as in the following example:

You can specify conditional formatting for related content on any grid object cell. For example, if the Market member name is “East” link to “ReportA”, if the Market member is “West”, link to “ReportB”. In the Format Cells dialog, you can specify the related content links. See “Replacing Text” on page 150.

From the Conditional Format dialog box, select Account Type from the Condition 1: If dropdown list box.
Then, select Expense from the Comparison Options drop-down list box. Click Format Cells to apply bold formatting to the selected cell, leaving all other font properties unchanged.

Next, add a second condition to the selected cell. Click the Add Condition 2 button, setting the Condition 2: If statement to Current Cell Value. Set the operator value to > (greater than), leaving the comparison value set to Number. Set the numeric value to 1000, as illustrated below:
Note: Click Delete Condition to delete a condition.

Next, add another Format by clicking the Add Format 2 button. Establish three conditions as follows to set the background color to lime green when a Cola cell has a value < 1000:

![Conditional Format](image)

Note: Use AND and OR to combine or differentiate the conditions.

After you establish the conditions, you can select the check box for Allow Parentheses in the Options box.

The following scenario performs the following operation:

Condition 1 and Condition 3
or
Condition 2 and Condition 3

For this example, if a cell has the label Cola associated with it using the Alias:Default and the cell value < 1000, the background is set to lime green. If a Member Name contains Cola and the cell value is < 1000, the background is set to lime green.

Note: The manner in which you use conditional formatting can impact performance depending on the size of the report. Performance is also contingent on the criteria used and the frequency of use (every cell). Each or all of these factors combined can affect performance. Data value comparisons are the fastest. Criteria such as data value, Member Name, and Member Alias/Description are faster because they are part of the metadata or data query. Avoid criteria such as Generation, Level, Account Type, and Attribute Value whenever possible, as performance is slower because those criteria are not part of the regular metadata or data query.
Applying Conditional Formatting to Grids

You can apply conditional formatting when working with grids in the Report Designer. First, you establish a condition for the selected cells. Then you specify formatting for values that meet that condition.

You can specify conditionally formatting for related content on any grid object cell. For example, if the Market member name is “East” link to “ReportA”, if the Market member is “West”, link to “ReportB”. On the Format Cells dialog, you can specify the related content links. See “Replacing Text” on page 150.

To apply conditional formatting:

1. Create a grid in a report, or open an existing report containing a grid.
2. Select the cells to which you want to apply conditional formatting.
3. Select Format, and then Conditional Format.
   You can also right-click and select Conditional Format from the shortcut menu.
4. From Select Property, select a property.

Note: Your property selection determines the options that are available in the drop-down list boxes.

The following table lists each conditional formatting property with the applicable data sources.

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Source Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cell Value</td>
<td>Essbase Planning Planning Financial Management</td>
</tr>
<tr>
<td>Row Value</td>
<td>Essbase Planning Planning Financial Management</td>
</tr>
<tr>
<td>Column Value</td>
<td>Essbase Planning Planning Financial Management</td>
</tr>
<tr>
<td>Current Cell Value</td>
<td>Essbase Planning Planning Financial Management</td>
</tr>
<tr>
<td>Property</td>
<td>Data Source Usage</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>Member Name</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td>Position Within</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
</tbody>
</table>
|                                | The Position Within property allows designers to format the first or last row or column of a data segment with multiple members or a member function. For example, to set the condition of the top row in a multiple member row or column, the conditions would resemble the following example:  
|                                | Condition 1: If                       |
|                                | Position Within Row Is Top Row, then  |
|                                | Format Cells = $.                      |
| Description                    | Financial Management                  |
| Alias                          | Essbase                               |
|                                | Planning                              |
| Generation                     | Essbase                               |
|                                | Planning                              |
| Relative Generation            | Essbase                               |
|                                | Planning                              |
| Level                          | Essbase                               |
|                                | Planning                              |
| Account Type                   | Essbase                               |
|                                | Planning                              |
|                                | Financial Management                  |
| Auto Calculation               | Essbase                               |
|                                | Planning                              |
|                                | Financial Management                  |
| Auto Calculation Group Heading | Essbase                               |
|                                | Planning                              |
|                                | Financial Management                  |
| Attribute Value                | Essbase                               |
| Line Item Detail               | Financial Management                  |
| Supporting Detail              | Planning                              |
5 From the **Reference Value** drop-down list box, select a value corresponding to the property value you selected from the **Select Property** drop-down list box.

6 **Select an operator:**
   - `=` (equal to)
   - `< >` (less than or greater than)
   - `>` (greater than)
   - `<` (less than)
   - `>=` (greater than or equal to)
   - `<=` (less than or equal to)
   - equals (is equal to)
   - not equals (is not equal to)
   - starts with
   - ends with
   - contains
   - Is

   **Note:** The full list of operators may not be displayed at the same time, as they rely on previous formatting selections.

7 **Select one of the following comparison options:**
   - **Number**—Enables the assignment of a certain value to a cell.
   - **Cell Value**—Returns the cell location; for example, A, 3.
   - **Row Value**—Returns the number of the row.
   - **Column Value**—Returns the letter of the column.
   - **Zero**—Assigns zero value for condition.
   - **No Data**—Indicates no data value for selection.
   - **Error**—Assigns the condition as an error.
   - **String**—Enables the definition of a string for the condition.
   - **0, 1, 2**—Indicates the number of levels or generations in the selected dimension.

   **Note:** The options in the drop-down list depend on your property selection.

8 To add another condition, click the **Add Format 2** button. Repeat step 4 through step 7 to continue adding conditions.
Note: The number on this button changes to as many as 7 as you add conditions.

9 Select And to combine this condition with the next condition, or select Or to distinguish this condition from the next condition.

10 Click Format Cells to assign a format to the condition. For more information, see “Formatting Grids” on page 144.

11 Use the Allow Not and Allow Parentheses check boxes, located in the Options section of the Conditional Format dialog box, to establish logic for conditions:
   - Allow Not enables you to negate a condition.
   - Allow Parentheses enables you to define precedence, or order of consideration for conditions, and specify the set of conditions as intended.

12 When you finish adding conditions for Format 1, perform an action:
   - To return to the grid in your report, click OK.
   - To add another format to your selection, click Add Format 2. For more information, see “Adding Conditional Formats” on page 161.

Deleting Conditions

You can delete a condition if it is no longer applicable to the grid, cell, row, or column. This functionality enables you to delete a single condition rather than an entire format, which can hold as many as seven conditions. For information on deleting formats, see “Deleting Conditional Formats” on page 163.

To delete conditions:

1 Create a grid in a report in the Report Designer, or open an existing report containing a grid with conditional formatting applied.

2 Select a grid or one or more cells, rows, or columns in a grid to which conditional formatting is applied.

3 Select Format, and then Conditional Format.

4 In the Conditional Format dialog box, click the Delete Format button. Use the drop-down list to select the condition to delete.

5 Click OK.

Adding Conditional Formats

You can specify as many as seven formats when applying conditional formatting. For each format, you can create as many as seven conditions.

Additional formats are added to enable the application of multiple formats when different conditions are met. After a condition, or conditions, is formatted, you can add another format that is tied to the cell.
For example, you can create one format that contains row and column numerical conditions to be applied to the selected cell. Then, you can create another format that specifies text color, shading, or alignment for the selected cell.

To add a format:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select the cells you want to format.
3. Select Format, and then Conditional Format to access the Conditional Format dialog box.
4. Add a condition to the cell, as described in “Applying Conditional Formatting to Grids” on page 158.
5. Click the Add Format #, where # is the number of the next format to be added.
6. Specify conditions to be included in the format.
7. To add more conditions and formats, repeat step 4 through step 6 as needed.
8. When completed, click OK.

Changing the Order of Conditional Formats

After you add multiple formats, you can change the order in which the conditions are evaluated and formats are applied to the selected cells. For example, suppose you specified a blue format for cells with a value of zero. In the second format, you specified a column value equal to the cell value and a row value equal to a specific number. Then, you specified a third format for the selected cell.

Using the arrows in the upper right corner of Conditional Format dialog box, you can change the order in which the formats are applied to a selected cell. The order of the formats is significant, because formats are absolute. The conditions created in the first format take precedence over the other formats and conditions.

To change the order of the conditional formats:

1. Create a grid in a report in the Report Designer or open an existing report containing a grid.
2. Select the cells whose format order you want to modify.
3. Select Format, and then Conditional Format.
4. Add three formats, as described in “Adding Conditional Formats” on page 161.
5. Select the format tab whose order you want to change.
6. Do either of the following:
   - To move a formatted tab to the left, click the Move Left button,
   - To move a formatted tab to the right, click the Move Format Tab Right button.
7. Click OK to return to the grid.
**Deleting Conditional Formats**

You can delete a format when the conditions it contains are no longer applicable. For example, if a format contains six conditions and you delete the format, you delete all six conditions contained in that one format.

To delete a format:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select a cell containing a numerical value with conditional formatting.

*Note:* Cells with conditional formatting are indicated with a colored bar.

3. Click **Format**, and then **Conditional Format**.
4. Click **Delete Format**. From the drop-down list, select the format number you want to delete.
5. Click **OK** to return to the grid.

**Row Height and Column Width Changing**

You can change row heights and column widths in grids. You can apply these changes to the entire grid or to specific rows or columns.

To change row height and column width:

1. Create a grid in a report in the Report Designer, or open an existing report containing a grid.
2. Select the rows or columns to format.
3. Perform an action:
   - To specify the row height, select a row and enter a value in the **Row Height** field on the **Row Properties** property sheet.
   - You can also use the arrows to increase or decrease the value.
   - To specify the column width, select a column and enter a value in the **Column Width** field on the **Column Properties** property sheet.
   - You can also drag the handle of a row or column in the grid to the desired size. If you make a row or column too small to see, when the pointer becomes a split line, drag the row or column handle until the row or column is visible.

**Merging Cells into One Cell**

You can combine two or more cells into one cell in a grid. For example, you can combine several cells to create one data cell, text cell, or formula cell. When you merge cells, the value and format of the upper-left cell is placed into the resulting merged cell. When the merged cell is separated into individual cells, all resulting cells inherit the formatting of the previously merged cell.
Note: Text automatically wraps within a grid’s cell. When print previewing PDF reports in Workspace, text cells with long text are automatically merged into the next cell to the right, if that cell is empty. When previewing HTML reports, text cells are not automatically merged. You may want to use the Merge feature, along with increasing the column width, to expand the area for text to display (the rows height automatically adjusts to display all text).

➢ To merge cells:

1. Open a report that contains a grid.
2. Select the cells to merge.
3. Right-click the selected cells, then select Merge.

Tip: To change the formatting of the merged cell, such as changing the alignment and font selection, select Format, and then Cells.

➢ To separate a merged cell into individual cells:

1. Open a report that contains a grid.
2. Select the merged cell.
3. Right-click the selected cell, then select Merge. The resulting cells inherit the formatting of the previously merged cell.

Conditional Suppression

Subtopics

- Conditional Suppression Behavior
- Using the Basic Option in Conditional Suppression
- Using the Advanced Option in Conditional Suppression
- Deleting Conditional Suppression

You can suppress the display of rows, columns, or grids using one or both of the following methods:

- Use the Basic option to suppress rows, columns, or grids based upon If Zero, If Missing, and If Error. For procedures, see “Using the Basic Option in Conditional Suppression” on page 166.

- Use the Advanced option to suppress rows, columns, or grids based on a variety of attributes. For example, you can specify that if a row contains cells with values below 100, the entire row is suppressed. For more information on using this type of conditional suppression, see “Using the Advanced Option in Conditional Suppression” on page 167.
Conditional Suppression Behavior

Conditional suppression behaves as follows:

- If an entire row or column meets the criteria specified for conditional suppression, the entire row or column is hidden; if only some of the cells in a row or column meet the criteria specified for conditional suppression, the row or column is not hidden.

- Suppressed data is not included in calculations. For example, if you suppress rows 23 and 24 and then calculate a sum for rows 10 through 30, the sum does not include the values in rows 23 and 24.

**Note:** When evaluating suppression, you can include or ignore the values and calculations in hidden rows and columns. This option is enabled only when the Hide Always check box is selected for the row or column. For instructions on this option, see “Controlling the Evaluation of Suppression in Hidden Rows or Columns” on page 153. Regardless of the setting you select, you can evaluate suppression based on hidden cells when you specifically reference the cells by using the Advanced Options method for conditional suppression.

- Hidden cells are ignored when evaluating suppression of a row or column unless they are specifically referenced. The output of the report is based on the condition used. The following example is a report with hidden columns.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>Hidden B Column</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Cola</td>
<td>Missing</td>
<td>Missing</td>
</tr>
<tr>
<td>2</td>
<td>Root Beer</td>
<td>Missing</td>
<td>Missing</td>
</tr>
<tr>
<td>3</td>
<td>Diet</td>
<td>61</td>
<td>Missing</td>
</tr>
</tbody>
</table>

The above report design is used to describe three conditional suppression scenarios.

**Note:** In the following 3 examples, column B is hidden and rows 1, 2, and 3 are selected for conditional suppression.

The first example describes the results when specifying suppression based on missing data. Note that because the condition does not reference the hidden column or a specific cell in the hidden column, the hidden column is ignored:

Suppress Row If: Data Values in Current Row = No Data

<table>
<thead>
<tr>
<th></th>
<th>East</th>
<th>South</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diet</td>
<td>61</td>
<td>Missing</td>
</tr>
</tbody>
</table>

The following example describes the results when specifying suppression based on the data cells in the hidden column B. Note that since the condition specifies the hidden column B, Root Beer is not suppressed because cell B2 contains data:

Suppress Row If: Data Values in Column B = No Data
When you use conditional suppression based on attributes, you can suppress formula rows or columns based on the suppression status of other data rows or columns. You can suppress text rows or columns based on the suppression status of other rows and columns.

**Note:** You cannot suppress data rows or columns based on the values of formula rows or columns.

The following table describes whether suppression is valid when referencing another suppressed data, formula, or text row or column.

<table>
<thead>
<tr>
<th>If the Row / Column Type you are suppressing =</th>
<th>And if suppression is based on Row / Column of Type =</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Formula</td>
</tr>
<tr>
<td>Data</td>
<td>Available</td>
</tr>
<tr>
<td>Formula</td>
<td>Available</td>
</tr>
<tr>
<td>Text</td>
<td>Available</td>
</tr>
</tbody>
</table>

**Using the Basic Option in Conditional Suppression**

Use basic conditional suppression to suppress data in rows or columns based on If Zero, If Missing, Is Error. Basic conditional suppression can be combined with advanced suppression options. See “Using the Advanced Option in Conditional Suppression” on page 167.

To suppress data:

1. **Create a grid in a report in the Report Designer, or open an existing report containing a grid.**
2. **Select the grid, rows, or columns to which you want to apply suppression.**
3. **In the Row or Column Properties sheet, click the Basic Options button.**

**Note:** If you selected a grid, first select the Suppression category in the Grid Properties property sheet.

4. **Select one or more of the following options:**
   - Select **If Zero, Suppress** to hide the rows or columns where all cell values equal zero.
   - Select **If Missing, Suppress** to hide the rows or columns where all cells are empty.
   - Select **If Error, Suppress** to hide the rows or columns where data cannot be retrieved or calculated because of errors.

5. **Optional:** To replace #ZERO, #MISSING, or #ERROR values in a row or column that cannot be hidden, take one or more of the following actions:
Note: You must have a grid selected and select the Suppression category in the Grid Properties property sheet.

- To replace zero (#ZERO) values in cells, type a value in the Zero Values text box.
- To replace no data (#MISSING) in cells, type a value in the No Data text box.
- To replace erroneous data (#ERROR) in cells, type a value in the Error text box.

6 To select all options and specify customized suppression criteria, select Advanced Options.

Using the Advanced Option in Conditional Suppression

The Advanced Option is used to suppress rows or columns in a grid based on specified attributes or values. Conditional suppression can be simple or complex, and can contain a maximum of seven conditions. Advanced conditional suppression can be combined with basic suppression options. See “Using the Basic Option in Conditional Suppression” on page 166.

When you define conditions, you can include the Not operator to suppress rows or columns that do not meet the condition (that result in false), and parenthesis to enclose and separate conditions from other conditions. Conditional suppression may be based on data or formula values.

To apply conditional suppression:

1 Create a report, or open an existing report containing a grid.
2 Select the rows, columns, or grid to which you want to apply suppression.
3 In the property sheet, click Advanced Options, then click the Setup button to display the Conditional Suppression dialog box.

Note: If you selected a grid, first select the Suppression category in the Grid Properties property sheet.

Conditional Suppression displays a Basic Options label above the suppression options. Basic options use OR operators:

- Basic Options:None—Basic options not selected with Advanced options.
- If Zero, Suppress OR—If Zero selected with Advanced options.
- If Zero OR Missing, Suppress OR—If Zero and If Missing selected with Advanced options.
- If Zero OR Error, Suppress OR—If Zero and If Error selected with Advanced options.
- If Zero OR Missing OR Error, Suppress OR—If Zero, If Missing and If Error selected with Advanced options.
- If Missing, Suppress OR—If Missing selected with Advanced options.
If Missing OR Error, Suppress OR—If Missing and If Error selected with Advanced options.

If Error, Suppress OR—If Error selected with Advanced options.

From Suppress Row or Column If, select a property.

The property selected determines which options are available in the drop-down lists for your expression. The following table lists each property with the default reference, operator, and comparison values. The default values are displayed automatically and can be changed.

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Source Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Values in Row</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Data Values in Column</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Data Values in Current Row</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Member Name</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Description</td>
<td>Financial Management</td>
</tr>
<tr>
<td>Alias - Default</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Generation</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Level</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Account Type</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Auto Calculation</td>
<td>Essbase</td>
</tr>
<tr>
<td></td>
<td>Financial Management</td>
</tr>
<tr>
<td></td>
<td>Planning</td>
</tr>
<tr>
<td>Line Item Detail</td>
<td>Financial Management</td>
</tr>
<tr>
<td>Property</td>
<td>Data Source Usage</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>Attribute Value</td>
<td>Essbase</td>
</tr>
<tr>
<td>Supporting Detail</td>
<td>Planning</td>
</tr>
</tbody>
</table>

5 From the Reference Value drop-down list box, select a value corresponding to the property value you selected from the Suppress Row or Column If drop-down list box.

6 Select one of the following operators, if applicable:

**Note:** The available operators depend on your property selection.

- < > (less than or greater than)
- > (greater than)
- >= (greater than or equal to)
- < (less than)
- <= (less than or equal to)
- not equals (is not equal to)
- starts with
- ends with
- contains
- Is

7 Select one of the following comparison options, if applicable:

**Note:** The available comparison options depend on your property selection.

- **Number**—Enables the assignment of a certain value to a cell.
- **Cell Value**—Returns the cell location; for example: A, 3.
- **Row Value**—Returns the number of the row.
- **Column Value**—Returns the letter of the column.
- **Zero**—Assigns zero value for condition.
- **No Data**—Indicates no data value for selection.
- **Error**—Assigns the condition as an error.
- **string**—Enables the definition of a string for the condition.
- **1, 2, 3,...**—Indicates the number of levels or generations in the selected dimension.
- **Non Expense**—Assigns the condition as a non-expense.
- **Expense**—Assigns the condition as an expense.
- **True**—Assigns a True value to the condition.
To add another condition, click the Add Condition 2 button.

**Note:** The number on this button is increased to as many as 7 as you add conditions.

To combine the current condition with the following condition, select one of the following operators:
- Select **And** if both combined conditions must be met.
- Select **Or** if only one of the combined conditions must be met.

Optional: In the Options area, add one of the following logic options to your condition:
- Select **Allow Not** to allow suppression only if the condition is not met.
- Select **Allow Parentheses** to define precedence, or order of consideration for the conditions, then specify the set of conditions as intended.

To add another condition, click the Add Condition 3 button.

Click **OK** to return to the grid in your report.

### Deleting Conditional Suppression

You can delete conditional suppression if it is no longer applicable. This functionality enables you to delete one condition at a time, which can hold as many as seven conditions.

To delete conditional suppression:

1. Open a report that contains a grid with conditional suppression.
2. Select the rows, columns, or grid to which conditional suppression is applied. The Advanced Options button in the property sheet is selected.
3. In the property sheet, click **Setup** to display the Conditional Suppression dialog box.

**Note:** If you selected a grid, first select the Suppression category in the Grid Properties property sheet.

4. In the Conditional Suppression dialog box, click **Delete Condition**. In the drop-down list, click the condition to delete. Repeat to delete additional conditions.
5. Click **OK**.

### Formatting Images

You can format the display of images in your report. For example, once you place an image, you can stretch the image, insert a page break before the image, and specify horizontal and vertical positions for the image.

To format images in a report:

1. Open a report and select an image to format.
2 To expand or shrink the image to fill the size of the image boundary on the Report Designer workspace, select Stretch on the Image Properties property sheet.

3 Optional: To set the horizontal positioning of a printed image, do one of the following tasks in the Horizontal drop-down list box:
   - To position the printed image relative to an object to its left, select Relative.
   - To position the printed image to the left side of the report, select Left.
   - To position the printed image in the middle of the report, select Center.
   - To position the printed image to the right of the report, select Right.
   - To position the printed image at its current location regardless of whether report objects to its left change in size, select None.

4 Optional: To set the vertical positioning of a printed image, do one of the following tasks in the Vertical drop-down list box:
   - To position the printed image relative to an object above it, select Relative.
   - To position the printed image on the top of the report, select Top.
   - To position the printed image to the middle of the report, select Middle.
   - To position the printed image to the bottom of the report, select Bottom.
   - To position the printed image at its current location regardless of whether report objects above it change in size, select None.

5 To start the image on a new page when it is printed, select the Page Break Before check box.

6 To move the image, click the image and, with all borders activated, drag it to another location.

   Note: Do not click and drag the borders of the image, because this stretches the image vertically or horizontally. Be sure to click the entire image.

Changing the Page Setup

You can alter page settings that control how your reports are printed or viewed. You can change the paper size, page orientation, heading display settings, default width and length for the report, and create a custom size for the work area. You can also specify whether cell documents print with consecutive page numbers or print the page settings on each document.

To change the page setup:

1 Open a report and select File, and then Page Setup.

2 Select the Page tab.

3 Select Letter, Legal, Ledger, A4 or A3 to specify the paper size.

4 When you print a report with Cell Document attachments enabled, the file attachment page numbers can be renumbered to reflect the page numbers of the report. For example, a nine page report with a five page Word document attached can renumber the Word document output to reflect the page number.
10 to 15. To print consecutive page numbers on cell documents, select Consecutive Page Numbers. To print the page numbers as specified on each cell document, clear Consecutive Page Numbers.

To learn more about cell documents, see “Attaching Cell Documents in a Report” on page 124.

5 Select Portrait or Landscape to set the orientation.

6 Set the display of row and column headings on your report:
   a. To display the row heading on every page of the report, select the Row Headings on Each Page check box. Clear this check box to display the row heading only once, on the first page of the report.
   b. To display the column heading on every page of the report, select the Column Headings on Each Page check box. Clear this setting to display the column heading only once, on the first page of the report.

   Note: The default selections include row headings and column headings on each page.

7 Optional: Enable the following options to fit the length or width of your report on one page.
   ● To scale the width of the report so that it fits on one page, select the Fit Width to Page check box. Deselect this setting to retain the original dimensions specified in the Design mode.
   ● To scale the length of the report so that it fits on one page, select the Fit Length to Page check box. Deselect this setting to retain the original dimensions specified in the Design mode.

   Note: If you select Fit Width to Page, you cannot insert a column page break. If you select Fit Length to Page, you cannot insert a row page break.

8 Select the Margins tab and enter values in the margin text boxes. You can specify a value for the top, bottom, left, and right margins. To adjust the margins, use one of the following methods:
   ● Use the up and down scroll arrows to adjust the margins of your report by increments of 100th of an inch.
   ● Type the margin settings for your report, in inches.

   Note: The Margins tab shows a preview of your report using the margins you specify.

9 Optional: Select the Workspace Size tab and then select the Use a Custom Workspace Size check box to specify dimensions for the workspace that are different than the paper size specified on the Page tab. To specify a workspace size, use one of the following methods:
   ● Use the up and down scroll arrows to adjust the report workspace size by increments of 100th of an inch.
   ● Type a width and height for your workspace, in inches.

   Note: Neither the height nor the width can exceed 20 inches.

10 When you are finished modifying the Page Setup options, click OK.
Using Functions

In This Chapter

Mathematical Functions ................................................................. 174
Text Functions .............................................................................. 194
Conditional Functions ................................................................. 234
Financial Functions ...................................................................... 237
Mathematical Functions

Subtopics

- About Mathematical Functions
- Abs
- Average
- AverageA
- CountA
- Count
- Difference
- Eval
- Max
- Min
- Mod
- PercentOfTotal
- Pi
- Product
- Random
- Round
-.Sqrt
- Sum
- Truncate / Trunc

About Mathematical Functions

Subtopics

- Arguments in Mathematical Functions
- Mathematical Operators in Expressions
- Mathematical Operands in Expressions
- Natural Precedence

Mathematical functions perform calculations on numeric values or data from a grid. Mathematical functions, formulas, and their syntax are not case-sensitive.

The syntax for a mathematical function is:

\[ \text{FunctionName}(\text{arguments}) \]

where:

\text{FunctionName} is the name of the mathematical function.

\text{arguments} are a numeric value, a row, column, or cell reference, or an embedded function.
Arguments in Mathematical Functions

Subtopics

- Numeric Arguments
- Row, Column, or Cell Reference Arguments
- Aggregate Property Arguments
- Reference Property Arguments
- Embedded Functions as Arguments

Numeric Arguments

The syntax for a numeric argument is:

\[(\text{numeral}_1, \text{numeral}_2, \ldots \text{numeral}_n)\]

where numerals 1 through n are any numbers including decimals and negative values. For example, the expression `Average(10, 20, 30)` returns the value 20.

Row, Column, or Cell Reference Arguments

The syntax for a row, column, or cell argument is:

\[\text{FunctionName}(\text{GridName}.\text{GridElement}[\text{segment}(\text{range})].\text{Property})\]

Table 19  Argument Components

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>(Optional) Name of a grid. For example, <code>Difference(grid1.row[5], grid2.row[5])</code> returns the difference of two rows on grid1 and grid2. If <code>GridName</code> is not specified, the default is the current grid where the formula is entered.</td>
</tr>
<tr>
<td>GridElement</td>
<td>(Optional) One of the following keywords: row, col, column, or cell. For example, <code>Max(row[1], row[2], row[3])</code> returns the maximum value of three rows. (The keyword row and column or col is optional.) You must specify row and column segment IDs. For example, <code>cell[2, A]</code> refers to the cell that is the intersection between row 2 and column A. The keyword cell is optional. For example, <code>[5, B]</code> refers to the cell that is the intersection between row 5 and column B. Cell references can use [row, col] syntax or [col, row] syntax. If <code>GridElement</code> is specified, letters represent columns and numbers represent rows; for example, <code>Max([1,A], [2,A], [3,A])</code></td>
</tr>
<tr>
<td>segment</td>
<td>(Required) Row, column, or cell reference number of a grid. For an expanded row or column, you must specify the segment. For example, <code>row[2]</code> addresses row segment 2. Segments are enclosed in square brackets [ ].</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>range</td>
<td>(Optional) Rows, columns, or cells that are expanded from the specified segment. If range is specified, the formula is calculated using only the specified range. For example, <code>row[2][3:5]</code> uses only the 3rd through 5th rows of expanded segment 2. When range is not provided, all expanded cells are used. <strong>Note:</strong> If a segment expands to only one row or column, do not use the range argument.</td>
</tr>
<tr>
<td>property</td>
<td>(Optional) One of the following keywords: average, averageA, count, countA, max, min, product, or sum. Property is used to aggregate the expanded rows, columns, or cells specified. Do not specify property when a reference is an argument. By not specifying property, the function calculates the reference in the most appropriate way. For example, the following expression returns the average of the cells within rows 1 and 2: <code>Average(row[1], row[2])</code> Whereas the following example first calculates the average of row[1], then the average of row[2], adds these two results, then divides by 2: <code>Average(row[1].average, row[2].average)</code> The default property for a row, column, or cell reference that is not used as a function argument is the sum property. For example, the following expression's default property is sum: <code>row[2]</code>. See &quot;Aggregate Property Arguments&quot; on page 176 and &quot;Reference Property Arguments&quot; on page 177.</td>
</tr>
</tbody>
</table>

Because segment is the only required part of a reference, the following references are the same:

```
Grid1.row[1].sum
```

`AverageA` and `CountA` include `#missing` and `#error` cells in the calculation. For example, if row 1 is a segment row that expands to Qtr1 = 100, Qtr2 = 200, Qtr3 = #missing, and Qtr4 = 400, the following function returns the value four (4):

```
row[1].CountA
```

All other functions exclude `#missing` data or `#error` cells. For example, the previous example of row 1 that expands to Qtr1 = 100, Qtr2 = 200, Qtr3 = #missing, and Qtr4 = 400, returns three in this example:

```
row[1].Count
```

### Aggregate Property Arguments

An aggregate row, column, or cell contains multiple rows, columns, or cells, respectively. The aggregate property argument is the last argument in the following mathematical function syntax:

```
FunctionName(GridName.Gridelement[segment(range)].property)
```

You apply aggregate properties to a row, column, or cell reference. Aggregate properties include:

- Average
- AverageA
- Count
- CountA
- Max
- Min
- Product
- Sum

When used as a mathematical function argument, the default for property is the same as the function. In the following example the default property is `Average`:

```
Average(row[2])
```

When not used as a mathematical function argument, the default for property is `sum`. In the following example the default property is the sum of an aggregate row:

```
row[2]
```

**Reference Property Arguments**

A reference property argument specifies how to treat formula reference results and is used in conjunction with the other properties.

There is one reference property argument: `IfNonNumber/IFFN`.

`IfNonNumber` specifies a replacement for `#Missing` and `#Error` values with a specific numeric value.

The syntax is:

```
AXIS[segment(range)].IfNonNumber(arg).AggregateProperty
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>AXIS</code></td>
<td>(Optional) A row, column, or cell keyword.</td>
</tr>
<tr>
<td><code>Segment(range)</code></td>
<td>Any axis reference, such as a row number or column letter.</td>
</tr>
<tr>
<td><code>IfNonNumber</code></td>
<td>How to treat missing or error data within the <code>AxisRef</code>.</td>
</tr>
<tr>
<td><code>(arg)</code></td>
<td>What number to use if missing or error data is encountered within the <code>AxisRef</code>.</td>
</tr>
<tr>
<td><code>AggregateProperty</code></td>
<td>(Optional) The aggregate function used for aggregate segments. See &quot;Aggregate Property Arguments&quot; on page 176.</td>
</tr>
</tbody>
</table>

For example:

If `cell[1,A] = 3` and `cell[1,B] = #Missing`,

The following expression returns `#Error`:

```
cell[1,A] / cell[1,B]
```

The following expression replaces `cell[1,B]` with 1 and returns 3:

```
cell[1,A] / cell[1,B].IfNonNumber(1)
```
Note: If you use suppression for #Missing or #Error in a grid, and the grid contains a formula row or column that uses the IfNonNumber property, #Missing and #Error remain suppressed.

Note: When using a Financial Management database connection, and the MissingValuesAreZeroInFormulas (formerly MissingValuesAreZeroInFormulasInHFM) option in JConsole is set to one (1), the #Missing value is zero regardless of the setting of the IfNonNumber property. However, if the file is set to zero, the IfNonNumber property behaves as described. (For information on JConsole, see “Property Information” in the Oracle Hyperion Financial Reporting Administrator’s Guide.

Embedded Functions as Arguments

You can embed functions as arguments within a function. In the following example, the function Average is embedded in the function Sum:

\[ \text{sum(row[3:5], avg(row[4:6], 40, 50), row[7; 9], 70, 80)} \]

- Row segments 3, 4 and 5
- The average of row segments 4, 5 and 6, with the numbers 40 and 50
- Row segments 7 and 9
- The numbers 70 and 80

Mathematical Operators in Expressions

Operators are symbols that perform arithmetical tasks or comparisons, or refer to ranges of columns, rows, or cells. Use operators in formulas that perform calculations on data rows or columns.

Table 21 Operators in Standard Formulas

<table>
<thead>
<tr>
<th>Type</th>
<th>Operator</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
</table>
| Arithmetical  | +        | Add         | Add the values in rows 4 and 5.  
\[ [4] + [5] \]  
A space is needed after +.  
- Incorrect: 5+4  
- Correct: 5+4  
- Correct: 5+ -4 |
<table>
<thead>
<tr>
<th>Type</th>
<th>Operator</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arithmetical</td>
<td>-</td>
<td>Subtract</td>
<td>Subtract 3 from the values in row 4.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A space is needed after - .</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Incorrect: 5-4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Correct: 5 - 4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Correct: 5 - -4</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>*</td>
<td>Multiply</td>
<td>Multiply the absolute values in row 4 by 150.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>abs([4]) * 150</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>/</td>
<td>Divide</td>
<td>Divide the values in row 4 by the values in row 5.</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>%</td>
<td>Percentage</td>
<td>Add the values in row 4 and multiply it by 10.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>row[4].sum * 10%</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>^</td>
<td>Exponential</td>
<td>Return the sixth exponent of row 4.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[4]^6</td>
</tr>
<tr>
<td>Arithmetical</td>
<td>Decimal</td>
<td>Decimal number</td>
<td>Multiply the values in row 4 by 5.67.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[4] * 5.67</td>
</tr>
<tr>
<td>Reference</td>
<td>[]</td>
<td>Specifies a row, column, or cell in a grid. Use numbers for rows and letters for columns.</td>
<td>Add the values in rows 4 and 12.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Sum ([4], [12])</td>
</tr>
<tr>
<td>Unary minus</td>
<td>-</td>
<td>Changes the sign of a value</td>
<td>Divide the values in column 4 by 12 and change the sign of the result.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>([4] / 12)</td>
</tr>
</tbody>
</table>

**Tip:** Use the `Eval` function to perform arithmetic operations as part of a function parameter. See “Eval” on page 185.

**Tip:** Natural precedence determines the order to carry out operations in expressions with multiple operators. See “Natural Precedence” on page 180.

### Mathematical Operands in Expressions

Operands specify the values that an operator uses to produce a result.

**Table 22  Operands Supported in Financial Reporting**

<table>
<thead>
<tr>
<th>Operand</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Literal</td>
<td>3, 0.0, 27.5, 65.334, -841</td>
</tr>
<tr>
<td>Row or column reference number</td>
<td>[1], [4], [8], [A], [C:D]</td>
</tr>
<tr>
<td>Operand</td>
<td>Examples</td>
</tr>
<tr>
<td>------------------------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>Cell reference number</td>
<td>[2, E], [E, 2]</td>
</tr>
<tr>
<td>Note: This cell reference addresses row 2 column E.</td>
<td></td>
</tr>
<tr>
<td>Functions</td>
<td>Average, Min, Max</td>
</tr>
<tr>
<td>Scoping</td>
<td>Grid1.row[3]</td>
</tr>
</tbody>
</table>

In the following row formula, which multiplies the values in row 3 by 100, the row reference [3] and 100 are both operands:

\[
[3] \times 100
\]

**Natural Precedence**

If a standard formula has two or more operators, the system performs the operations in the order of the operators' natural precedence.

The order of natural precedence to perform operations in an expression that has multiple operators is:

1. Unary minus (-# where # is any number)
   
   **Note:** Do not add a space after the unary minus operator.
2. Multiplication (*) and division (/)
3. Subtraction (-#) and addition (+# where # is any number)
   
   **Note:** Add a space after the subtraction and addition operator.

For example, if row 10 has a value of 8, the following expression produces a value of 20 for that column by calculating \(3 \times 4 + 8 = 20\), following the order of natural precedence:

\[
[10] + 3 \times 4
\]

**Abs**

Abs is a mathematical function that returns the absolute value of a numeric value, row, column, or cell. The absolute value of a number is that number without a negative sign. A negative number becomes positive, while a positive number remains positive.

**Syntax:**

\[
\text{Abs} (\text{argument})
\]

where **argument** is one of the following values:
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, $\text{Abs}(-20)$ returns the value 20.</td>
</tr>
<tr>
<td></td>
<td>Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid.</td>
</tr>
<tr>
<td></td>
<td>References can be specified in several ways. The reference syntax is: $\text{GridName}.\text{GridElement[segment(range)].Property}$. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

Examples:

Return the value 30:

$\text{Abs}(-30)$

Return the absolute value of the value in row 1:

$\text{Abs(row[1])}$

Calculate the absolute value of the sum of column E:

$\text{Abs(column[E].sum)}$

Point to expanded rows 1 through 3 within design segment 3 of Grid1:

$\text{Abs(Grid1.row[3(1:3)])}$

### Average

Average is a mathematical function that returns the average of a group of numeric values, rows, columns, or cells. Average excludes #missing and #error cells when obtaining the average.

**Note:** The calculation does not include missing values regardless of whether they are suppressed.

**Syntax:**

$\text{Average}(\text{arguments})$

or

$\text{Avg}(\text{arguments})$

where $\text{arguments}$ is one or more of the following values:

<p>| Argument | Description                                                                 |</p>
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. For example <code>Avg(Grid1.row[4(3:5)])</code> returns the average of grid1, row segment 4, range 3 through 5. References can be specified in several ways. The reference syntax is: <code>GridName.GridElement[segment(range)].Property</code>. See “Row, Column, or Cell Reference Arguments” on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

Examples:

Return the value 20:

Avg(10, 30, 20)

Return the average of all numbers that are part of three aggregate rows:

`Average(row[1],row[6],row[8])`

Calculate the average of three aggregate columns; E, G, and I. (The calculation produces three numbers, then calculates the average of the three numbers.)

`Avg(column[E]avg,column[G]avg,column[I].avg)`

Calculate the average of two columns; E on the current grid and E on grid2. (The calculation includes all expanded columns.)

`Avg(column[E],grid2column[E])`

Calculate the average of aggregate row 3 and divide the average by 100:

```plaintext
Avg(row[3])/100
```

**AverageA**

`AverageA` is a mathematical function that returns the average of a group of numeric values, rows, columns, or cells. `AverageA` includes #missing and #error cells, which are treated as zero values when obtaining the average.

**Note:** #missing and #error are included only for rows or columns that are not suppressed.

Syntax:

```plaintext
AverageA(arguments)
```

or

```plaintext
AvgA(arguments)
```

where `arguments` is one or more of the following values:
### Argument Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, ( \text{AverageA}(10, 20, 30) ) returns the value 20. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: ( \text{GridName. GridElement[segment(range)].property} ). See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175. For example, ( \text{AvgA(Grid1.row}[4(3:5)] ) returns the average of grid1, row segment 4, range 3 through 5.</td>
</tr>
</tbody>
</table>

#### Example:

If a grid has four rows with the values 10, 20, 30, and #error, the following formula in the 5th row returns the value 15:

\[ \text{AverageA}([1:4]) \]

### CountA

**CountA** is a mathematical function that returns the number of values in a group of numeric values, rows, columns, or cells. **CountA** includes #missing and #error cells when obtaining the count only for rows or columns that are not suppressed.

**Syntax:**

\[ \text{CountA}(\text{arguments}) \]

where **arguments** is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, ( \text{CountA}(10, 20, 30, 50) ) returns the value 4. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: ( \text{CountA(GridName. GridElement[segment(range)].property} ). See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
</tbody>
</table>

**Examples:**

If a grid has four rows with the values 10, 20, 30, and #error, the following formula in the 5th row returns the count of four rows:

\[ \text{CountA}([1:4]) \]

Return the count of four rows:
CountA(row[1], row[6], row[8] row[where data yields #error])

**Count**

Count is a mathematical function that returns the number of values in a group of numeric values, rows, columns, or cells. Count excludes #missing and #error when obtaining the count.

**Syntax:**

Count(*arguments*)

where *arguments* is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Count(10, 20, 30) returns the value 3. Numeric values can include decimals and negative values. See “Numeric Arguments” on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName.GridElement[segment(range)].Property. See “Row, Column, or Cell Reference Arguments” on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function. For a list of mathematical functions. See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

**Examples:**

Return the count of three rows: 1, 6, and 8:

Count(row[1], row[6], row[8])

Return the count of 3 columns:

Count( column[E], column[G], column[I])

Calculate the count of two columns located on different grids:

Count( grid1.column[E], grid2.column[E])

Calculate the count of the cell located at row 4, column D:

Count(cell[D,4])

Calculate the count of aggregate row 3 in grid 5:

Count(grid5.row[3])

**Difference**

Difference is a mathematical function that returns the absolute value of the difference of a numeric value, row, or column subtracted from another numeric value, row, or column. This is also known as the variance.
Syntax:
Difference(arg1, arg2)

where arg2 is subtracted from arg1 and is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Difference(3, 5) returns the absolute value 2.</td>
</tr>
<tr>
<td></td>
<td>Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or</td>
<td>A pointer to a row, column, or cell within a grid.</td>
</tr>
<tr>
<td>reference</td>
<td>References can be specified in several ways. The reference syntax is: GridName. GridElement[segment(range)].Property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See &quot;Mathematical Functions&quot; on page 174.</td>
</tr>
</tbody>
</table>

**Note:** Difference returns the absolute value of arg2 subtracted from arg1, whereas the minus sign in subtraction negates a number.

**Examples:**

Return the absolute value of 8:

```
Difference(3, -5)
```

Calculate the difference of two aggregate columns:

```
Difference(column[E], column[G])
```

Calculate the difference of two columns that are located on different grids, grid1 and grid2:

```
Difference(grid1.column[E], grid2.column[E])
```

**Note:** You can type the text label “Difference” or “Variance”.

**Eval**

Eval is a mathematical function that evaluates an expression. Use Eval as an embedded function argument to consolidate multiple expressions into one expression.

**Syntax:**

```
Eval(expression)
```

where expression is one or more of the following values:
### Argument Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName.GridElement[segment(range)].Property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>operators</td>
<td>Any of the supported arithmetic operators (+, -, *, /, ^, %).</td>
</tr>
</tbody>
</table>

**Example:**

Divide row 1 by row 2, and then round the data to four places:

\[
\text{Round(Eval([1]/[2]),4)}
\]

### Max

Max is a mathematical function that returns the maximum value in a group of numeric values, rows, columns, or cells.

**Syntax:**

\[
\text{Max(}\text{arguments}\text{)}
\]

where arguments is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, (\text{Max}(10,20,30)) returns the value 30. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName.GridElement[segment(range)].Property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
</tbody>
</table>

**Examples:**

Return the maximum value in rows 1, 6, and 8:

\[
\text{Max(row[1], row[6], row[8])}
\]

Calculate the maximum of the sums of aggregate rows:

\[
\text{Max(row[1].sum, row[2].sum, row[3].sum)}
\]
**Min**

Min is a mathematical function that returns the minimum value in a group of numeric values, rows, columns, or cells.

**Syntax:**

Min(*arguments*)

where *arguments* is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Min(10,20,30) returns the value 10. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName.GridElement[segment(range)].Property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function. See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

**Examples:**

Return the minimum value in rows 1, 6, and 8:

Min (row[1], row[6], row[8])

Calculate the minimum of the sums of aggregate rows:

Min(row[1].sum, row[2].sum, row[3].sum)

**Mod**

Mod is a mathematical function that returns the remainder, or modulus, from a division.

**Syntax:**

Mod(*arg1*, *arg2*)

where *arg2* is the divisor, and *arg1* and *arg2* are one of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Mod(6,4) returns the value 2. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName.GridElement[segment(range)].Property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>

Example:
Divide 10 by 5 and return the remainder of 0:
\[ \text{Mod}(10, 5) = 0 \]

**PercentOfTotal**

PercentOfTotal is a mathematical function that returns the result of a numeric value, row, column, or cell divided by another numeric value, row, column, or cell which is multiplied by 100.

Syntax:
PercentOfTotal \((arg1, arg2)\)

where:
- \(arg1\) is a component of the running total \((arg2)\), usually a row or column reference.
- \(arg2\) is the running total relative to \(arg1\), usually a cell reference containing the grand total.
- \(arg1\) is divided by \(arg2\) with the result multiplied by 100. \(arg1\) and \(arg2\) are one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, (\text{PercentOfTotal}(100, 20)) returns the value 500. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName. GridElement[segment(range)].Property. See “Row, Column, or Cell Reference Arguments” on page 175.</td>
</tr>
</tbody>
</table>

Examples:
Return the value of 5 percent.
\(\text{PercentOfTotal}(20, 400)\)

Divide the value of each cell in column A by the total market value in cell A5, multiply the result by 100, and display the resulting PercentOfTotal in column B.
\(\text{PercentOfTotal} ([A],[A,5])\)
Using the above example, the following table shows the PercentOfTotal results in column B:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sales</td>
<td>Percent of Total</td>
</tr>
<tr>
<td>2</td>
<td>Mkt1</td>
<td>60</td>
</tr>
<tr>
<td>3</td>
<td>Mkt2</td>
<td>120</td>
</tr>
<tr>
<td>4</td>
<td>Mkt3</td>
<td>120</td>
</tr>
<tr>
<td>5</td>
<td>Total Mkt</td>
<td>300</td>
</tr>
</tbody>
</table>

**Tip:** You enter the formula by clicking on the header for Column B and using the formula bar.

**Pi**

Pi is a mathematical function that returns the number 3.14159265358979, the mathematical constant, accurate to 15 digits. Pi is the ratio of the circumference of a circle to its diameter.

**Syntax:**

\[ \text{PI()} \]

**Example:**

Return row 3 divided by the product of Pi and 2:

\[ \text{row[3]} / (\text{PI()} \times 2) \]

**Product**

Product is a mathematical function that multiplies all numbers or references and returns the product.

**Syntax:**

\[ \text{Product(arguments)} \]

where arguments is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Product(2,20) returns the value 40. Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName. GridElement[segment(range)].property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

**Example:**

Return 40:

\[\text{Product}(2, 20)\]

---

**Random**

Random is a mathematical function that returns a random number between 0.0 and 1.0.

**Syntax:**

\[\text{Random}()\]

**Example:**

Return a random number between 0.0 and 1.0 and multiply it by 1000:

\[\text{Random}() \times 1000\]

---

**Round**

Round is a mathematical function that rounds a number up or down by the specified digits.

**Syntax:**

\[\text{Round}(\text{arg1}, \text{integer})\]

where \(\text{arg1}\) is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, Round(81.3987, 3) returns the value 81.399. Numeric values can include decimals and negative values. See “Numeric Arguments” on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName. GridElement[segment(range)].property. See “Row, Column, or Cell Reference Arguments” on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See “Mathematical Functions” on page 174.</td>
</tr>
</tbody>
</table>

\(\text{Integer}\) specifies the number of digits to round the number:

- If \(\text{integer}\) is greater than zero, the number is rounded to the specified number of decimal places.
• If \textit{integer} is zero, the number is rounded to the nearest integer.
• If \textit{integer} is less than zero, the number is rounded to the left of the decimal point.

Examples:
Round to 3 decimals:
Round(3594.5567,3)=3594.557

Round to the nearest integer:
Round(3594.5567,0)=3595

Round to the thousand, (also known as scaling):
Round(3594.5567,-3)=4000

\textbf{Sqrt}

\textit{Sqrt} is a mathematical function that returns the square root of a numeric value, row, column, or cell.

Syntax:
\texttt{Sqrt(\textit{argument})}

where \textit{argument} is one of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, \texttt{Sqrt(100)} returns the value 10.</td>
</tr>
<tr>
<td></td>
<td>Numeric values can include decimals and negative values. See &quot;Numeric Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: \texttt{GridName}. \texttt{GridElement[segment(range)].property}. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
<tr>
<td>function</td>
<td>An embedded function.</td>
</tr>
<tr>
<td></td>
<td>See &quot;Mathematical Functions&quot; on page 174.</td>
</tr>
</tbody>
</table>

Example:
Return the value of 4:
\texttt{Sqrt(16)}

\textbf{Sum}

\textit{Sum} is a mathematical function that returns the summation of a group of numeric values, rows, columns, or cells.
Syntax:

\[ \text{Sum}(\text{arguments}) \]

where \textit{arguments} is one or more of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, \text{Sum}(10,20,30) returns the value 60.</td>
</tr>
<tr>
<td></td>
<td>Numeric values can include decimals and negative values. See “Numeric Arguments” on page 175.</td>
</tr>
<tr>
<td>row, column, or cell reference</td>
<td>A pointer to a row, column, or cell within a grid.</td>
</tr>
</tbody>
</table>
|                        | References can be specified in several ways. The reference syntax is: GridName.  
                        | GridElement[segment(range)].property. See “Row, Column, or Cell Reference Arguments” on page 175. |
| function               | An embedded function.                                                      |
|                        | See “Mathematical Functions” on page 174.                                  |

Examples:

Return the value 30:

\text{sum}(10,20)

Return the sum of three rows:

\text{sum}(\text{row}[1], \text{row}[6], \text{row}[8])

Calculate the sum of three aggregate columns:

\text{sum}(\text{column}[E], \text{column}[G], \text{column}[I])

Calculate the sum of two columns which are located on different grids:

\text{sum}(\text{grid1.col}[E], \text{grid2.colm}[E])

**Truncate / Trunc**

Truncate is a mathematical function that removes the specified number of digits from numeric values.

Syntax:

\text{Trunc}(\text{arg1}, \text{integer})

where \textit{arg1} is one of the following values:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>numeric</td>
<td>A numeric value. For example, 234.567.</td>
</tr>
<tr>
<td></td>
<td>See “Numeric Arguments” on page 175.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>row, column, or cell</td>
<td>A pointer to a row, column, or cell within a grid. References can be specified in several ways. The reference syntax is: GridName. GridElement[segment(range)].property. See &quot;Row, Column, or Cell Reference Arguments&quot; on page 175.</td>
</tr>
</tbody>
</table>

**Integer** specifies the number of digits to remove:

- A positive integer determines the number of digits to the right of the decimal point.
- A zero (0) integer returns the integer located to the left of the decimal point.
- A negative integer indicates the number of digits to the left of the decimal point.

**Examples:**

The following statement uses a positive integer of 2. The first two digits to the right of the decimal point remain, and the following digit is removed:

\[
\text{Trunc}(234.567, 2) = 234.56
\]

The following statement uses a zero (0) integer. All digits to the right of the decimal point are removed:

\[
\text{Trunc}(234.567, 0) = 234
\]

The following statement uses a negative integer of -2. All digits to the right of the decimal point are removed and the last 2 digits of the integer are truncated.

\[
\text{Trunc}(234.567, -2) = 200
\]

**Note:** Any formatting previously applied to a cell, column, or row is maintained when you use the Trunc function. The following example shows the results of a Trunc function where the cell value was previously formatted to display three decimal places: \[
\text{Trunc}(234.567, 0) = 234.000
\]
Text Functions

Subtopics

- About Text Functions
- Annotation
- CalcStatus
- CellText
- Date
- DataSource
- DPDescription
- DPJustification
- DPNotesAndAttachments
- Footnote
- GetCell
- GetHeading
- GridDimension
- HFMCurrency
- ListOfCellDocuments
- MemberAlias
- MemberDescription
- MemberName
- MemberProperty
- MemberQualifiedName
- Page
- PageCount
- PlanningAnnotations
- Specifying Demarcations Between Annotations
- ProcessManagementStatus
- ReportAuthor
- ReportCreated
- ReportDesc
- ReportFolder
- ReportModified
- ReportModifiedBy
- ReportName
- ReportRunBy
- RetrieveValue

About Text Functions

Subtopics

- Text Function Considerations
- Using the Current/Cur Keyword in a Text Function
- Using Text Functions to Retrieve Decision Package Information

Text functions return report information, such as the report name or the report description.
You use text functions in grids or in text objects.

**Text Function Considerations**

- Enclose text functions in braces:
  
  `<<TextFunction (arguments)>>`
  
- You can use spaces between arguments.
- All arguments are enclosed in quotes, except for numeric arguments.
  
  For example, the parameter “NYC, New York” in the function
  `<<MemberName("Grid1", NYC, New York)>>` should be restated as
  `<<MemberName("Grid1", "NYC, New York")>>`.
- Arguments related to date formatting are case-sensitive.
- To apply text functions to a grid, create a text row or column, or a heading cell, and then insert the text function.

**Using the Current/Cur Keyword in a Text Function**

You can use the `current` keyword with certain text functions to make the formulas more dynamic, more useful, and less redundant. The `current` keyword can replace required parameters in text functions. Hard-coded parameters produce the same results, whereas the `current` keyword can produce dynamic values. For example, when you use `current` for the Page parameter in a multiple page grid, it produces the value for each page.

You can use any form of the `current` keyword; `current`, or `cur`. Current keywords are not case-sensitive.

**Note:** The `current` keyword is supported in a text object only when using the following text functions: `MemberName`, `MemberAlias`, and `MemberDescription`, where `current` specifies the Page dimension parameter. However, the Grid name, Row, and Column must be defined, where applicable, and cannot use the `current` keyword.

**Example 1:**

In the `GetCell` function, many of the parameters are or may be relative to the current cell in which the function is entered. Therefore, the `current` keyword can be applied to this function. The `GetCell` function uses the following syntax:

`<<GetCell ("GridName", Row, Column, Page)>>`

In the following example, the hard-coded parameters for `GetCell` return the same cell value of Grid 1, cell 1A, page 1 across multiple pages:

`<<GetCell ("Grid1", 1, A, 1)>>`

The next example uses the `current` keyword with the `GetCell` function. The cell where the function is entered displays different values for each page. Notice that the `current` keyword is also applied to the grid name. If the grid name is changed, the text formula is still correct.
Example 2

When you use the `current` keyword in a cell that expands, the results are relative to the expanded cell.

The following example uses `GetCell` functions in an expanded cell (Column A expands to columns Qtr1, Qtr2, Qtr3 and row 1 expands to rows East and West). Text Row 1 uses the `current` keyword for the Column parameter; Text Row 2 uses a hard-coded value for the Column parameter.

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qtr1, Qtr2, Qtr3</td>
<td>East, West</td>
</tr>
<tr>
<td>Text Row 1</td>
<td>Text Row 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Quarter 1</th>
<th>Quarter 2</th>
<th>Quarter 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>East</td>
<td>5,120</td>
<td>4,502</td>
<td>7,304</td>
</tr>
<tr>
<td>West</td>
<td>3,405</td>
<td>2,300</td>
<td>4,462</td>
</tr>
<tr>
<td>Text Row 1</td>
<td>5,120</td>
<td>4,502</td>
<td>7,304</td>
</tr>
<tr>
<td>Text Row 2</td>
<td>5,120</td>
<td>5,120</td>
<td>5,120</td>
</tr>
</tbody>
</table>

- Text Row 1 uses the `current` keyword for the Column parameter, which results in the update of the current column of each expanded cell (Qtr1, Qtr2, Qtr3). Also, because the `current` keyword is used for the Page parameter, the values in Text Row 1 update each page relative to the current page of the grid.
- Text Row 2 uses a hard-coded value for the Column parameter (A), which results in the same value of 5,120 for all quarters. A hard coded reference to a Column parameter returns the same top left value of the column and row.

<table>
<thead>
<tr>
<th>Text Function</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;&lt;DataSource(&quot;GridName.Axis[ID]&quot;,InfoType)&gt;&gt;</code></td>
<td><code>&lt;&lt;DataSource(current, &quot;App&quot;)&gt;&gt;</code></td>
</tr>
</tbody>
</table>

Parameters where applicable:
- `GridName`: supported in all cells
- `InfoType`: not applicable

**Note:** The current grid name is used.
<table>
<thead>
<tr>
<th>Text Function</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;&lt;GetCell(&quot;GridName&quot;,Row,Col,Page)&gt;&gt;</code></td>
<td>`&lt;&lt;GetCell(cur, 1, cur, current)&gt;&gt;</td>
</tr>
<tr>
<td><strong>Parameters where applicable:</strong></td>
<td><code>&lt;&lt;GetCell(cur, 1(3), cur, current)&gt;&gt;</code></td>
</tr>
<tr>
<td>- GridName: supported in all cells</td>
<td><strong>Note:</strong> The value from the third (expanded) row of row 1 is shown.</td>
</tr>
<tr>
<td>- Row: supported in all cells</td>
<td><code>&lt;&lt;GetCell(cur, cur, A, current)&gt;&gt;</code></td>
</tr>
<tr>
<td>- Col: supported in all cells</td>
<td><strong>Note:</strong> The value from the second (expanded) column of column A would be shown.</td>
</tr>
<tr>
<td>- Page: supported in all cells</td>
<td><code>&lt;&lt;GetCell(cur, cur, A(B), current)&gt;&gt;</code></td>
</tr>
</tbody>
</table>

Use the following in a data row or column heading:

`<<MemberName("DimName")>>`

Use the following in a non-data row or column heading:

`<<MemberName("GridName",Row/Col/Page,"DimName")>>`

Use the following parameter where applicable:

**DimName**: Heading cells only

Use the following in a data row or column heading:

For Essbase:

`<<MemberAlias("DimName")>>`

For Financial Management:

**Note:** The following syntax cannot be used with the Oracle Fusion General Ledger product.

`<<MemberDescription("DimName")>>`

`<<MemberDesc("DimName")>>`

Use the following in a non-data row or column heading:

`<<MemberAlias("GridName",Row/Col/Page,"DimName")>>`

Use the following parameters where applicable:

**DimName**: Heading cells only

`<<CalcStatus("GridName", Row, Col, Page)>>`

**Note:** The parameters allowing the current keyword are highlighted in bold.

`<<CellText("GridName", Row, Column, Page)>>`

`<<CellText("GridName", Row, Column, Page, MemberOverride)>>`
<table>
<thead>
<tr>
<th>Text Function</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;&lt;GridDimension(&quot;GridName&quot;, &quot;Axis&quot;, index)&gt;&gt;</code></td>
<td><code>&lt;&lt;MemberAlias(&quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td><code>&lt;&lt;HFMCurrency(&quot;GridName&quot;, Row, Col, Page)&gt;&gt;</code></td>
<td><code>&lt;&lt;MemberAlias(&quot;GridName&quot;, &quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td><strong>Note:</strong> The parameters allowing the current keyword are highlighted in</td>
<td>`&lt;&lt;MemberAlias(&quot;GridName&quot;, Row</td>
</tr>
<tr>
<td>bold.</td>
<td><code>&lt;&lt;MemberDesc(&quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberDesc(&quot;GridName&quot;, &quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td>`&lt;&lt;MemberDesc(&quot;GridName&quot;, Row</td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberName(&quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberName(&quot;GridName&quot;, &quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td>`&lt;&lt;MemberName(&quot;GridName&quot;, Row</td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberQualifiedName(&quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberQualifiedName(&quot;GridName&quot;, &quot;DimName&quot;)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td>`&lt;&lt;MemberQualifiedName(&quot;GridName&quot;, Row</td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberProperty(DimensionName, Property)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td><code>&lt;&lt;MemberProperty(&quot;GridName&quot;, DimensionName, Property)&gt;&gt;</code></td>
</tr>
<tr>
<td></td>
<td>`&lt;&lt;MemberProperty(&quot;GridName&quot;, Row</td>
</tr>
<tr>
<td><strong>Note:</strong> The following are not available for Oracle Fusion General Ledger.</td>
<td></td>
</tr>
<tr>
<td><code>&lt;&lt;PlanningAnnotations(&quot;GridName&quot;, Row, Column, Page, Attributes)&gt;&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;&lt;PlanningAnnotations(&quot;GridName&quot;, Row, Column, Page, Attributes, Range)&gt;&gt;</code></td>
<td></td>
</tr>
<tr>
<td><strong>Note:</strong> The parameters allowing the current keyword are highlighted in</td>
<td></td>
</tr>
<tr>
<td>bold.</td>
<td></td>
</tr>
<tr>
<td><code>&lt;&lt;ProcessManagementStatus(&quot;GridName&quot;, Row, Column, Page)&gt;&gt;</code></td>
<td></td>
</tr>
<tr>
<td><strong>Note:</strong> The parameters allowing the current keyword are highlighted in</td>
<td></td>
</tr>
<tr>
<td>bold.</td>
<td></td>
</tr>
<tr>
<td><code>&lt;&lt;ListOfCellDocuments(&quot;GridName&quot;, Information)&gt;&gt;</code></td>
<td></td>
</tr>
<tr>
<td><strong>Note:</strong> The parameters allowing the current keyword are highlighted in</td>
<td></td>
</tr>
<tr>
<td>bold.</td>
<td></td>
</tr>
</tbody>
</table>
Using Text Functions to Retrieve Decision Package Information

Oracle Hyperion Public Sector Planning and Budgeting decision packages consolidate financial information and supporting justification for a specific action or proposed policy. Decision packages present requests for funding along with narrative justification and supporting information that may be submitted through the review and approval process. For example, Public Safety Improvements could be a municipal decision package that contains budget requests defined by the Fire Department and the Police Department that present the line item costs and total amount involved to fulfill the decision package.

Decision packages are created at the application level in Planning for a planning unit hierarchy (a combination of scenario, entity, and version member). Each decision package can have multiple budget requests, and you can retrieve the description, justification, and notes and attachments at the decision package level or at the budget request level.

You can use the following text functions in Financial Reporting Studio to retrieve decision package information such as descriptions, justifications, and notes and attachments to use on
financial reports. Where applicable, you can insert these functions into a text box; a grid’s text row, column, or cell; or a custom heading.

- **DPDescription**
- **DPJustification**
- **DPNotesAndAttachments**

These functions can be used to retrieve decision package information from either a BSO cube or an ASO cube in Planning. Unlike BSO cubes, decision package items are available as dimensions in ASO cubes. When you design reports in Financial Reporting Studio using ASO cubes, the decision package items are available as dimensions to use in a grid row, a column, a page, or a user POV.

**Note:** For information about adding decision packages to Financial Reporting books, see “Adding Decision Package Reports to Financial Reporting Books” in the *Oracle Hyperion Financial Reporting User’s Guide*. For detailed information about decision packages, see the *Oracle Hyperion Public Sector Planning and Budgeting User’s Guide*.

### Annotation

**Annotation** returns the requested information about an annotation within a grid object. You can define what information to return.

**Syntax:**

```plaintext
<<Annotation("GridName", Row, Col, Page, Attributes, Reference, Attachments)>>

<<Annotation("GridName", Row, Col, Page)>>

<<Annotation("GridName", Row, Col, Page, Attributes, Reference)>>

<<Annotation("GridName", Row, Col, Page, Attributes, Attachments)>>

<Annotation("GridName", AllRows, AllCols, Page, Attributes, Attachments)>>
```

**where:**

- **GridName** is the name of the grid containing the annotations. The “current” keyword is supported when Attribute is used in a grid’s text cell or custom heading.
- **Row** is the row number on the grid. The “current” keyword is supported in when Attribute is used in a grid’s text cell or custom heading.

**Note:** To select all rows, **row** can be replaced with the keyword **AllRows**.

- **Col** is the column reference (letter) on the grid. The “current” keyword is supported when Attribute is used in a grid’s text cell or custom heading.

**Note:** To select all Columns, **Col** can be replaced with the keyword **AllCols**.
- **Page** is a numeric value representing the index of member combinations on all page dimensions of the grid. The first page dimension combination in a grid has an index of 1, the second page dimension combination has an index of 2, and so on.

- **Attributes** identify the information to be returned from the annotation. They may include the keyword **All** (default value) or a combination of one or more of the following keywords: **Title & Description & Author & Date & Category**. Attribute keywords can be arranged in any order of output. Each attribute must be separated with an ampersand (&).

- **Reference** is a keyword noting which annotations will be returned, where:
  - **All** (default) returns the initial annotation and all replies.
  - **Top <n>** returns the initial annotation and first <n> replies.
  - **Bottom <n>** returns the initial annotation and the last <n> replies. The initial annotation is always shown and the number <n> does not include the initial annotation.
  - **BottomOnly <n>** returns the last <n> replies. The initial annotation shown for 4 or less <n> replies; the initial annotation with 5 or more <n> replies is not shown.
  - **First** returns the initial annotation only.
  - **Replies** returns all replies. The initial annotation is not shown.

- **Attachments** is a boolean (true/false) value indicating whether attachments associated with the annotation should print with the report where:
  - **True** means that attachments will print at the end of the report.
  - **False** means that no attachments will print.

**Examples:**

```
<<Annotation("Grid1", cur, A, cur, All, All, true)>>
<<Annotation("Grid1", cur, A, cur, Title & Description & Author & Date & Category, Top 10, true)>>
<<Annotation("Grid10", 284, AB, cur, Title & Description & Date & Category, Bottom 10, false)>>
<<Annotation("Grid1", cur, A, cur, Title & Description & Author, BottomOnly 4, false)>>
<<Annotation("Grid Name", 1, A, 1, All)>>
<<Annotation("Grid Name", "AllRows", "AllCols", 1, All)>>
<<Annotation("Grid Name", 1, A, 1, All, First)>>
<<Annotation("Grid Name", 1, cur, cur, All, Replies)>>
<<Annotation("Grid Name", 1, cur, cur)>>
<<Annotation("Grid Name",1, cur, cur, Desc, Replies)>>
<<Annotation(cur, cur, A, cur, Desc)>>
<<Annotation(cur, 3, ABC, cur, title & Desc, true)>>
```
CalcStatus

CalcStatus is a text function that returns the calculation status of a cell for a Financial Management database connection. You can use CalcStatus in report headers, rows, and columns.

Syntax:

<<CalcStatus("GridName", Row, Col, Page)>>

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a Financial Management grid. This value must be enclosed in quotation marks</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on.</td>
</tr>
<tr>
<td>Col</td>
<td>Letter representing the column ID of the grid. The first column ID is A, the second column ID is B, and so on.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on.</td>
</tr>
</tbody>
</table>

Oracle Fusion Forecast Management can return these calculation status values:

- OK
- OK - No data
- OK - System Changed
- Needs Subcalculate
- Needs Translation
- Needs Consolidation
- Needs Consolidation No Data
- Locked

Example:

Display the calculation status of the cell that is in row 21, column B on page 1 of Grid1:

<<CalcStatus("Grid1",21,B,1)>>

Note: CalcStatus supports the current keyword. See “Using the Current/Cur Keyword in a Text Function” on page 195.
CellText

Subtopics
- CellText Using AllRows/AllCols
- CellText Using MemberOverride
- LabeledCellText

CellText is a text function that retrieves the textual content from the following database connections:
- A Financial Management cell
- Planning cell annotations
- A Linked Reporting Object (LRO) from an Essbase cell

CellText applies only to a Cell note LRO. See the Oracle Essbase Database Administrator’s Guide.

The row, column, and page arguments apply to the intersection in a grid that contains the linked text or note.

Syntax:
```celltext
<<CellText("GridName", Row, Column, Page)>>
```

CellText can also provide an override of the grid’s “natural” point of view (POV) to a member selection when a cell does not exist in a grid.

Syntax:
```celltext
<<CellText("GridName", Row, Column, Page, MembeOverride)>>
```

See “CellText Using MemberOverride” on page 204 for details.

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a grid. This value must be enclosed in quotation marks</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on.</td>
</tr>
<tr>
<td>Column</td>
<td>Letter that represents the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on.</td>
</tr>
<tr>
<td>MemberOverride</td>
<td>String of dimensions and corresponding member selections that override the CellText POV. See “CellText Using MemberOverride” on page 204.</td>
</tr>
<tr>
<td>AllRows/AllCols</td>
<td>Keywords that allow for an all row or column reference. See “CellText Using AllRows/AllCols” on page 204.</td>
</tr>
</tbody>
</table>
**Note:** If the row or column parameters refer to a segment that expands, the top-left expanded cell is used. You can refer to any expanded cell using the “range” notation.

**Example:**
Create a report with linked text in the cell that is in row 21, column B of a grid and that is on the first page dimension.

```
<<CellText("mygrid",21,B,1)>>
```

**Tip:** To view result-time rows and columns, select View, then Row & Column Headers in Web Preview, and then preview the report by clicking Web Preview.

**Note:** CellText only supports the Current keyword when used in a grid cell; not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

### CellText Using AllRows/AllCols

**Table 24**  
**CellText Examples Using AllRows/AllCols**

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;,3,AllCols,1)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any column of row 3, page 1.</td>
</tr>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;,Curr,Allcols,1)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any column of the current row (&quot;Curr&quot;), page 1.</td>
</tr>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;,Curr,Allcols,Curr)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any column of the current row (&quot;Curr&quot;), current page.</td>
</tr>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;,AllRows,C,1)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any row of column C, page 1.</td>
</tr>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;,AllRows,Allcols,2)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any cell on page 2. “Any cell” meaning any row/column since AllRows/AllCols is specified for both row and column.</td>
</tr>
<tr>
<td><code>&lt;&lt;CellText(&quot;Grid1&quot;, AllRows, Allcols, Curr)&gt;&gt;</code></td>
<td>Includes cell text that occurs on any cell on the current page.</td>
</tr>
</tbody>
</table>

### CellText Using MemberOverride

**MemberOverride** can consist of one or more dimensions. It uses the following format:

```
DimensionName = MemberName, DimensionName = MemberName
```

where:

- **DimensionName** is the name of an override dimension.
- **MemberName** is the name of the override member for the dimension.

Follow these guidelines when you specify member overrides:
If you specify two or more member overrides, separate them with a comma (,) or a semicolon (;) and enclose each dimension and member name in quotation marks (" "), For example:

"Entity"="Acme","Period"="Q1"

If you specify only one member override, do not include the separator character.

Enclose dimension names and member names in quotation marks (" ") if they contain any of the following characters: ; , = ( ) < >.

For a single member override, the parameter may be enclosed in quotation marks (" "). For example, CellText is properly evaluated if you use the following member override statement:

"Scenario=Budget"

Table 25  CellText Examples Using MemberOverride

<table>
<thead>
<tr>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;&lt;CellText(cur, cur, A, cur)&gt;&gt;</td>
<td>Existing syntax with no dimension override</td>
</tr>
<tr>
<td>&lt;&lt;CellText(cur, cur, A, cur, Scenario=Budget)&gt;&gt;</td>
<td>New syntax with one dimension override</td>
</tr>
<tr>
<td>&lt;&lt;CellText(&quot;Grid Name&quot;, 1, A, current, Value = &quot;Entity Currency&quot;)&gt;&gt;</td>
<td>One dimension override using quotation marks in the member name</td>
</tr>
<tr>
<td>&lt;&lt;CellText(&quot;Grid Name&quot;, 1, A, cur, Value = Entity Currency)&gt;&gt;</td>
<td>No quotation marks in the member name that contains a space</td>
</tr>
<tr>
<td>&lt;&lt;CellText(&quot;Grid Name&quot;, 1, A, cur, &quot;Value=$USD&quot; = Entity Currency)&gt;&gt;</td>
<td>Quotation marks surrounding dimension name that contains an equal sign</td>
</tr>
<tr>
<td>&lt;&lt;CellText(Grid1, 1, cur, cur, Value = Entity Currency, Scenario=Actual)&gt;&gt;</td>
<td>Two-dimension override that uses a comma separator</td>
</tr>
<tr>
<td>&lt;&lt;CellText(Grid1, 1, cur, cur, Value = Entity Currency; Scenario=Actual)&gt;&gt;</td>
<td>Two-dimension override that uses a semicolon separator</td>
</tr>
<tr>
<td>&lt;&lt;CellText(&quot;Current&quot;, 34, BB, cur, &quot;Value&quot;= Entity Currency, Scenario=Actual; Period = Qtr3)&gt;&gt;</td>
<td>Three-dimension override that uses comma and semicolon separators</td>
</tr>
<tr>
<td>&lt;&lt;CellText(cur, 1(3), A(B), cur, 300 = ABC , &quot;Americas, Value=(in $USD);&quot; = &quot;Entity Currency (USD)* , Scenario=Actual)&gt;&gt;</td>
<td>Four-dimension override</td>
</tr>
</tbody>
</table>

LabeledCellText

LabeledCellText retrieves the textual content from a Financial Management database connection through their ADM driver.

Syntax:

LabeledCellText("GridName", Row, Column, Page, "Label", [Attachments], [MemberOverride])
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a grid. This value must be enclosed in quotation marks.</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on. The ALLROWS and CUR keywords are supported.</td>
</tr>
<tr>
<td>Column or Col</td>
<td>Letter representing the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on. The ALLCOLS and CUR keywords are supported.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on.</td>
</tr>
<tr>
<td>Label</td>
<td>Identifies the Financial Management cell label to which the desired cell texts (and optionally attachments) correspond. This value must be enclosed in quotation marks.</td>
</tr>
<tr>
<td>Attachments</td>
<td>(Optional) A true or false value that indicates whether attachments associated with the cell text should be retrieved.</td>
</tr>
<tr>
<td>MemberOverride</td>
<td>(Optional) String of comma separated dimensions and corresponding member selections that override the active POV. See “CellText Using MemberOverride” on page 204.</td>
</tr>
</tbody>
</table>

**Examples:**

LabeledCellText("Grid1", CUR, A, 1, "MATURITYDATE")

LabeledCellText("Grid1", 3, CUR, 1, "RATING", TRUE)

LabeledCellText("Grid1", 3, CUR, 1, "RATING", TRUE, Year=Quarter1)

**Date**

Date is a text function that returns the date and time that an online report is populated with data, or the date and time that a snapshot report is saved. The date and time is retrieved from the report server and corresponds to the country in which the report server is located. You can use Date in a text box, text cell, or a row or column heading.

**Syntax:**

```
<<Date("format","TimeZoneId")>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Defaults to the user preference for date/time. Valid values are those accepted by Java SimpleDateFormat, or “user” or empty (“user”). Date and time format characters are case-sensitive.</td>
</tr>
<tr>
<td>TimeZoneId</td>
<td>Defaults to user preferences for the time zone. Valid values are those accepted by Java TimeZone, getTimeZone(), or the server time zone. For example, for TimeZone getTimeZone() to specify the east coast, enter: &quot;America/New_York&quot;. When using TimeZoneId, enclose the format parameter within quotation marks. This applies only when TimeZoneId is indicated. You do not need to update reports prior to the TimeZoneId enhancement.</td>
</tr>
<tr>
<td>Format Characters</td>
<td>Meaning</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>G</td>
<td>era designator</td>
</tr>
<tr>
<td>M</td>
<td>month in year</td>
</tr>
<tr>
<td>d</td>
<td>day in month</td>
</tr>
<tr>
<td>h</td>
<td>hour in am/pm (1-12)</td>
</tr>
<tr>
<td>H</td>
<td>hour in day (0-23)</td>
</tr>
<tr>
<td>m</td>
<td>minute in hour</td>
</tr>
<tr>
<td>s</td>
<td>second in minute</td>
</tr>
<tr>
<td>S</td>
<td>millisecond</td>
</tr>
<tr>
<td>E</td>
<td>day in week</td>
</tr>
<tr>
<td>D</td>
<td>day in year</td>
</tr>
<tr>
<td>F</td>
<td>day of week in month</td>
</tr>
<tr>
<td>w</td>
<td>week in year</td>
</tr>
<tr>
<td>W</td>
<td>week in month</td>
</tr>
<tr>
<td>a</td>
<td>am/pm marker</td>
</tr>
<tr>
<td>k</td>
<td>hour in day (1-24)</td>
</tr>
<tr>
<td>K</td>
<td>hour in am/pm (0-11)</td>
</tr>
<tr>
<td>z</td>
<td>time zone</td>
</tr>
<tr>
<td>y</td>
<td>year</td>
</tr>
<tr>
<td>'</td>
<td>literal text</td>
</tr>
<tr>
<td>&quot;&quot;</td>
<td>two single quotes to produce one single quote</td>
</tr>
</tbody>
</table>

**Note:** Use 2 single quotation marks to produce a single quote or an apostrophe as in O’Clock (O”Clock).

Considerations when using *format* characters:

- The number or count of *format* characters determines the format:
  - For text, less than four pattern letters means use a short or abbreviated form, if it exists.
  - Four or more pattern letters means use full form.
M or MM means use the month as a number.

MMM means use the three-letter abbreviation for the month.

MMMM means use the full name of the month.

For example, if the date is April 24:

- `<Date("dd-M-yy")>` displays 24-4-02
- `<Date("dd-MM-yy")>` displays 24-04-02
- `<Date("dd-MMMM-yy")>` displays 24-Apr-02
- `<Date("dd-MMMMM-yy")>` displays 24-April-02

- For numbers, the number of pattern letters is the minimum number of digits. Shorter numbers are padded with zeros. Year is a special case. If you use “yy”, the last two digits of the year are displayed, but if you use “yyyy,” the four-digit year is displayed.

For example, if the month is February:

- `<Date("MM-yyyy")>` displays 02-2002

- All upper and lower case alphabet characters are considered patterns, regardless of whether they are designated format characters.

- Enclose literal text within single quotation marks. To use a single quotation mark in literal text, enclose the text within single quotes and use two single quotes for the apostrophe.

For example, the format to print 4 o’clock is:

- `<Date("hh 'o''clock' ")>`

Examples:

<table>
<thead>
<tr>
<th>Date Format</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;Date(&quot;d/M/yy&quot;)&gt;</code></td>
<td>23/3/02</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;d-MM-yy&quot;)&gt;</code></td>
<td>23-Mar-02</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;EEEE, MMMM dd, yyyy&quot;)&gt;</code></td>
<td>Tuesday, March 23, 2002</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;h:mm:ss a&quot;)&gt;</code></td>
<td>12:52:05 PM</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;h:mm:ss a zzzz&quot;)&gt;</code></td>
<td>12:52:05 PM Eastern Standard Time</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;EEEE, MMMM dd, yyyy G 'time:' h:mm:ss a zzzz&quot;)&gt;</code></td>
<td>Tuesday, March 23, 2002 AD time: 12:52:05 PM Eastern Standard Time</td>
</tr>
<tr>
<td><code>&lt;Date(&quot;hh 'o''clock' a, zzzz&quot;)&gt;</code></td>
<td>12:00 PM, Eastern Standard Time</td>
</tr>
</tbody>
</table>

**Note:** You must use two single quotation marks to produce one single quotation mark in the text of your formatted result.
**DataSource**

DataSource is a text function that returns the application name, database, or alias table name of a grid. Use DataSource in a text box, text cell, or a row or column heading.

Syntax:

```
<<DataSource("GridName.Axis[ID]", InfoType)>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GridName</code></td>
<td>(Required) Any grid in a report, enclosed in quotes.</td>
</tr>
<tr>
<td><code>Axis[]</code></td>
<td>(Optional) A pointer to a design segment of the grid. Axis can be one of the following keywords: row, col, or column, and it is not case-sensitive. For example, <code>&lt;&lt;DataSource(&quot;Grid1.Row[1]&quot;,DB)&gt;&gt;</code> When Axis is not provided, the grid's default information is returned. Use Axis when a grid has multiple database connections.</td>
</tr>
<tr>
<td><code>ID</code></td>
<td>(Required if Axis is used) Indicates the design-time, row number, or column letter from which the database connection information is retrieved. If ID points to an invalid or nonexistent row or column ID, the grid default settings are used. In addition, if a grid does not have a secondary database connection, the grid default settings are used. For example, <code>&lt;&lt;DataSource(&quot;Grid1.Col[A]&quot;,DB)&gt;&gt;</code></td>
</tr>
</tbody>
</table>
| `InfoType` | (Required) One of the following keywords:  
  - **App** returns the application name.  
  - **DB** returns the database name.  
  - **Alias** returns the alias table name. To return the alias, the grid must have the alias table enabled. By default, the alias table is disabled.  
  - **Name** returns the database connection name associated with the specified grid.  
  
  **Note:** “Server” is not supported as an argument for InfoType. |

**Note:** All arguments are case insensitive.

**Example 1:**

Insert data source information in your report:

```
<<DataSource("Grid1", App)>>
```

**Example 2:**

Show the database name of the data source associated with the heading where the function was entered:

```
<<DataSource(cur, DB)>>
```
Note: DataSource supports the current keyword. See “Using the Current/Cur Keyword in a Text Function” on page 195.

DPDescription

DPDescription is a text function that retrieves the description of a decision package member.

Note: See “Using Text Functions to Retrieve Decision Package Information” on page 199.

Syntax:

\[
\text{DPDescription}("GridName", \text{Row}, \text{Column}, \text{Page}, \text{DPLevel/BRLevel})
\]

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a grid. This value must be enclosed in quotation marks.</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>Column of Col</td>
<td>Letter that represents the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>DPLevel/BRLevel</td>
<td>\text{DPLevel} retrieves information about the description of a decision package. \text{BRLevel} retrieves information about the budget requests for a decision package.</td>
</tr>
</tbody>
</table>

Examples:

Retrieve a budget request description.

\[
<<\text{DPDescription}("Grid1", 1, B, 2,\text{BRLevel})>>
\]

Retrieve a decision package description for the current page.

\[
<<\text{DPDescription}("Grid1", 1, B, \text{Curr},\text{DPLevel})>>
\]

Retrieve a decision package description for the current row and column.

\[
<<\text{DPDescription}("Grid1", \text{Curr}, \text{Curr}, 2,\text{DPLevel})>>
\]

Create a decision package for a planning unit hierarchy that is a combination of Entity: “California”, Scenario: “Budget”, and Version: “Stage 4”.

\[
<<\text{DPDescription}("Grid1", 1, B, 2,\text{DPLevel})>>
\]

If your grid has “California” in row 1, “FY13” in column B, and “Stage 4” on page 2, then the above syntax would retrieve decision package information in a text box or a grid’s cell text only if you select “Budget” in the User POV and “Stage 4” in the pages.
**DPJustification**

DPJustification is a text function that retrieves decision package justification.

**Note:** See “Using Text Functions to Retrieve Decision Package Information” on page 199.

**Syntax:**

```plaintext
DPJustification("GridName", Row, Column, Page, DLevel/BRLLevel, QuestionOption, CategoryName, QuestionName)
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GridName</code></td>
<td>Name of a grid. This value must be enclosed in quotation marks.</td>
</tr>
<tr>
<td><code>Row</code></td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td><code>Column/Col</code></td>
<td>Letter that represents the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td><code>Page</code></td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td><code>DLevel/BRLlevel</code></td>
<td><code>DLevel</code> retrieves information about the description of a decision package. <code>BRLLevel</code> retrieves information about the budget requests for a decision package.</td>
</tr>
<tr>
<td><code>QuestionOption</code></td>
<td>One of the following values:</td>
</tr>
<tr>
<td></td>
<td>1—Retrieves only mandatory questions</td>
</tr>
<tr>
<td></td>
<td>2—Retrieves only answered questions</td>
</tr>
<tr>
<td></td>
<td>3—Retrieves only a specific category</td>
</tr>
<tr>
<td><code>CategoryName</code></td>
<td>(Optional) Category name details. Use only if <code>QuestionOption</code> is 2 or 3.</td>
</tr>
<tr>
<td><code>QuestionName</code></td>
<td>(Optional) Question name details. Use only if <code>QuestionOption</code> is 2 or 3.</td>
</tr>
</tbody>
</table>

**Examples:**

Retrieve budget request justifications for all mandatory questions for Row 1, Column A, Page 1.

```plaintext
<<DPJustification("Grid1", 1, A, 1, BRLLevel, 1)>>
```
Retrieve decision package justifications for all answered questions where the category name and question name match the given text for the “Current Row”, “Column B”, and “Page 1”.

<<DPJustification("Grid2", CURR, B, 1,DPLevel, 2,"Monorail construction","Justify the need for monorail on BG road")>>

Retrieve decision package justifications for all answered questions for all category names and question names for the “Current Row”, “Column B”, and “Page 1”.

<<DPJustification("Grid2", CURR, B, 1,DPLevel, 2,"" ,"" )>>

Retrieve decision package justifications for all answered questions where the category name and question name match the given text.

<<DPJustification("Grid2", All, B, 1,DPLevel, 2,"Monorail construction","Justify the need for monorail on BG road")>>

DPNotesAndAttachments

DPNotesAndAttachments is a text function that retrieves the notes and attachments associated with a decision package.

Note: See “Using Text Functions to Retrieve Decision Package Information” on page 199.

Syntax:

DPNotesAndAttachments("GridName", Row, Column, Page, DPLevel/BRLevel Attributes, Range, Notes/Attachments)

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a grid. This value must be enclosed in quotation marks.</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>Column or Col</td>
<td>Letter that represents the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on. The CUR keyword is supported.</td>
</tr>
<tr>
<td>DPLevel/BRLevel</td>
<td>DPLevel retrieves information about the description of a decision package. BRLevel retrieves information about the budget requests for a decision package.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **Attributes** | One of the following values:  
  - All  
  - Title  
  - Author  
  - Date  
  - Text  

For example, you can use attributes to filter the notes and attachments submitted by an author or with a specific title.  
**Note:** Arrange attributes in the preferred order of output, and separate each attribute with an ampersand (&).

| **Range**      | Use the keywords All, Top, or Bottom to select a specific number of notes and attachments from the top or bottom of an output, or all notes and attachments.  

For example:  
- All returns all annotations  
- Top 5 returns the first five annotations  
- Bottom 10 returns the last ten annotations  
- Bottom 1 returns the last annotation |

| **Notes/Attachments** | One of the following values:  
  - 1—Notes  
  - 2—Attachments |

**Example:**

Retrieve the top three decision packages notes for Row 1, Column B, and the current page.

```
<<DPNotesAndAttachments("Grid1", 1,B,CURR,DPLevel,All,Top 3, 1)>>
```

Retrieve all budget request notes for the current row, current column, and current page.

```
<<DPNotesAndAttachments("Grid1", Curr,Curr,Curr,BRLevel,All,All,1)>>
```

Retrieve all decision package attachments for the current row, current column and current page.

```
<<DPNotesAndAttachments("Grid1", Curr,Curr,Curr,DPLevel,All,All,2)>>
```

**Footnote**

The Footnote text function, used in Text Objects only, returns the requested information about footnotes within a given report object. Footnote prints information about footnotes that occur in the object specified, or in all the objects in the report if the first parameter is set to "all". When a grid is referenced, all the footnotes on the grid are printed. The partial POV footnotes are printed first, followed by any full POV footnotes. The full POV footnotes are printed in order based on the location of the cell from top left to bottom right, page by page. More than one syntax is supported.
Syntax:

Footnote("ObjectName", Attributes, Reference, Attachments)
Footnote("ObjectName")
Footnote("ObjectName", Attributes)
Footnote("ObjectName", Attributes, Reference)
Footnote("ObjectName", Attributes, Attachments)

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ObjectName</td>
<td>Name of the report object containing the footnotes. If &quot;all&quot;, the function returns all available footnotes from all objects in the report. The ObjectName parameter is updated when a grid object is renamed.</td>
</tr>
</tbody>
</table>
| Attributes  | Information returned from the footnote. This may include the following keywords:
  - All (default value)
  - A combination of one or more of these keywords; Title, Description (Desc), Author, Date, Category

The Attributes keywords may be arranged in the preferred order of output. Separate each attribute with an ampersand (&). |
| Reference    | Keyword denoting which annotations will be returned where:
  - All (default value)—The initial annotation and all replies
  - Top <n>—The initial annotation and first replies. (The initial annotation is always shown, and the number does not include the initial annotation.)
  - Bottom—The initial annotation and the last number of replies specified in Bottom <n>. (The initial annotation is always shown, and the number does not include the initial annotation.)
  - BottomOnly <n>—The last replies. The initial annotation is not shown unless there are less replies than the BottomOnly <n> number. For example, BottomOnly 5 for an annotation with five replies does not show the initial annotation, but for an annotation with four replies, the initial annotation is shown.
  - First—The initial annotation only.
  - Replies—All the replies except the initial annotation. |
| Attachments  | Boolean (true/false) value indicating whether any attachments should be printed with the footnote.
  - True—Attachments print along with the report, at the end.
  - False (default value)—Attachments do not print. |

Examples:

<<Footnote("Grid1", All, All, true)>>
<<Footnote(All, All, All, true)>>
<<Footnote(All, All, Bottom 2, true)>>
<<Footnote(All, All, BottomOnly 20, true)>>
<<Footnote(All, All, First, true)>>
<<Footnote(All, All, Replies, true)>>

214
GetCell is a text function that returns a data value from a grid.

**Note:** When referencing a cell containing Planning or Financial Management data, GetCell returns a string value instead of a number.

**Syntax:**

```
<<GetCell("GridName", Row, Column, Page)>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GridName</code></td>
<td>Name of a grid or the Current keyword.</td>
</tr>
<tr>
<td><code>Row</code></td>
<td>Numeric value representing the row number of the grid. The first row in a grid has an index of one, the second row has an index of two, and so on. The Current keyword can also be used.</td>
</tr>
</tbody>
</table>
Considerations when using `GetCell`:

- If the row or column parameters refer to a segment that expands, the top-left expanded cell is used. You can refer to any expanded cell using the “range” notation.
- `GetCell` only supports the `Current` keyword when used in a grid cell; not in a text box object, header, or footer. The `Current` keyword can be used in any of the parameters to make the function more dynamic. See “Using the Current/Cur Keyword in a Text Function” on page 195.

Example:

Set up a grid to show the net income for several quarters, and display this value in a text box that contains an executive summary of the report.

The grand total for the period was `<<GetCell("mygrid",21,B,1)>>`

The report is created with the value in the cell that is in row 21, column B of a grid, and is on the first page dimension.

## `GetHeading`

`GetHeading` is a text function that obtains text headings from a specified grid.

### Syntax:

```
<<GetHeading("GridName", Page, Reference, Offset)>>
<<GetHeading("GridName", Page, Reference,"Delimiter")>>
```

### Table 1: `GetHeading` arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>GridName</code></td>
<td>Name of a grid containing the heading to extract. The <code>Current</code> keyword can also be used.</td>
</tr>
<tr>
<td><code>Reference</code></td>
<td>View time reference of the heading to obtain. The <code>Current</code> keyword can also be used. See “Row, Column, or Cell Reference Arguments” on page 175.</td>
</tr>
<tr>
<td><code>Offset</code></td>
<td>Number that represents the dimension in a row or column heading with more than one dimension. A numeric offset returns a single dimension heading. For example, in a row heading with three dimensions, you can specify (starting from left to right) “1” to return the first dimension, “2” to return the second dimension, and “3” to return the third dimension.</td>
</tr>
<tr>
<td>&quot;Delimiter&quot;</td>
<td>Quoted string that separates all headings from the row or column. If you provide a delimiter, you obtain all headings in the reference, separated by the provided delimiter.</td>
</tr>
</tbody>
</table>
Examples:
Use GetHeading to return the following column and row headings.

<table>
<thead>
<tr>
<th></th>
<th>A(A)</th>
<th>A(B)</th>
<th>A(C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jan</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feb</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1(1)</td>
<td>Georgia</td>
<td>112</td>
<td>67</td>
</tr>
<tr>
<td>1(2)</td>
<td>East</td>
<td>5,864</td>
<td>3,322</td>
</tr>
<tr>
<td>1(3)</td>
<td>Market</td>
<td>15,904</td>
<td>9,277</td>
</tr>
</tbody>
</table>

Use the syntax below to return the data in the previous sample grid:

<table>
<thead>
<tr>
<th>GetHeading Example</th>
<th>Gets or Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,A,2)</td>
<td>Jan</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,A(B),2)</td>
<td>Feb</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,A(C),&quot;-&quot;)</td>
<td>Actual - Mar</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,1,1)</td>
<td>Georgia</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,1(2),1)</td>
<td>East</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,1(2),2)</td>
<td>&lt;error&gt; (it is an invalid reference)</td>
</tr>
<tr>
<td>GetHeading(&quot;Grid1&quot;,1,1(3),&quot;-&quot;)</td>
<td>Market</td>
</tr>
</tbody>
</table>

Note: GetHeading only supports the Current keyword when used in a grid cell; not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

**GridDimension**

GridDimension is a text function that returns the name of a dimension on a grid’s page, column, or row axis.

Syntax:

```
<<GridDimension("GridName", "Axis", index)>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a grid in the report enclosed in quotation marks. The Current keyword can also be used.</td>
</tr>
</tbody>
</table>
**Argument** | **Description**  
---|---
*Axis* | Grid axis where the dimension is placed: page, row, or column. Keywords must be used. The *Current* keyword can also be used.
*index* | Numeric value representing the offset or index of the dimension on the named axis. The first dimension on an axis has an index of one, the second dimension has an index of two, and so on.  
You can use the *Current* keyword on a heading cell only (page, row, or column).

**Example:**

A shared grid, named *mygrid*, has several dimensions on each row, column, and page axes, and displays the name of the first dimension on the row axis in a text box. Because the dimension may change in the future, you use the `GridDimension` function in the text box. Type the following in the text box:

This report is based on the `<<GridDimension("mygrid","Row",1)>>` dimension.

**Note:** See “Using the Current/Cur Keyword in a Text Function” on page 195.

**HFMCurrency**

`HFMCurrency` is a text function that returns the ID of the currency property for a cell in a Financial Management database connection.

**Syntax:**

`<<HFMCurrency("GridName", Row, Col, Page)>>`

**Argument** | **Description**  
---|---
*GridName* | Name of a Financial Management grid. This value must be enclosed in quotation marks.
*Row* | Numeric value representing the row index of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on.
*Col* | Letter representing the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on.
*Page* | Numeric value representing the index of member combinations on all page dimensions of the grid. The first page dimension combination in a grid has an index of 1, the second page dimension combination has an index of 2, and so on.

A designer can map the string returned from Financial Management to another string by way of a properties file change. The mappings are defined in the JConsole file. For example, a designer might add these lines to HFM Entity Currency section:

- `EntityCurrency_USD=$`
- `EntityCurrency_EUR=_`
Note: For information on JConsole, see “Property Information” in the Oracle Hyperion

If USD is returned from Financial Management, the U.S. dollar symbol ( $ ) is displayed; if EUR
is returned from Financial Management, the European euro symbol ( _ ) is displayed. The
properties file includes these and other examples as comments.

Example:
Display the currency property for the cell that is in row 21, column B on page 1 of Grid1:

<<HFMCurrency("Grid1",21,B,1)>>

Note: HFMCurrency only supports the Current keyword when used in a grid cell; not in a text
box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function”
on page 195.

ListOfCellDocuments

Note: ListOfCellDocuments is only available with Financial Management or Essbase as the
database connection.

ListOfCellDocuments is a text function that returns a list of all cell documents and their file
attributes that a designer has selected to be retrieved from cells within a report. You can use this
function in a text box, text cell or text cell heading.

In Financial Management or Essbase, documents are inserted into the data source cells of a
report. The Financial Reporting designer selects the cells and chooses Retrieve Cell Documents
on the property sheet. If those documents exist, they are appended to the report when printed
or saved as a snapshot or PDF. to view the list of cell documents that exist, the designer adds the
ListOfCellDocuments to the report in a text box, text cell or text cell heading.

Syntax:

<<ListOfCellDocuments ("GridName", FileName&Description&PageNumber)>>

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>The name of a grid in the report.</td>
</tr>
<tr>
<td>FileName</td>
<td>Returns the name of the cell document.</td>
</tr>
<tr>
<td>Description</td>
<td>Returns the description of each cell document.</td>
</tr>
</tbody>
</table>

Note: Documents extracted from the Financial Management database connection, are truncated to 20 characters; 16
characters for the document name, and 4 characters for the file name extension.
<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
</table>
| `PageNumber` | Returns the consecutive start page number of each cell document within the report. This is not necessarily the page number specified on the document.  
**Note:** Page numbers only display when selecting PDF format for print or web preview. |

**Example:**

In the following example, two grids in the report are selected to print cell document information. In Grid1, the file name, description, and the consecutive start page number of the extracted cell documents are requested, in Grid2 only the file name and start page numbers are requested.

```
<<ListOfCellDocuments ("Grid1", FileName&Description&PageNumber)>>
<<ListOfCellDocuments ("Grid2", FileName&PageNumber)>>
```

The resulting list might look similar to the following, where Grid1 has two attached documents and Grid2 has one attached document:

**Attached Documents:**

- April Variance High Volume Page3
- April Variance Low Volume Page9
- May Variance Page 18

**Note:** `ListOfCellDocuments` supports the `Current` keyword. See “Using the Current/Cur Keyword in a Text Function” on page 195.

### MemberAlias

**Note:** This function is available only with Essbase or Planning details as the database connection.

`MemberAlias` is a text function. Based on the syntax used, it returns the alias of the member assigned to a dimension in the row, column, page, or Point of View (POV).

- The following syntax returns the alias of a row heading, column heading, or page. The syntax can be deployed from any heading cell, text cell, or text box object and can reference any grid in a current report.

  ```
  <<MemberAlias ("GridName", Row/Col/Page, "DimName")>>
  ```

- The following syntax returns the alias in a row heading, column heading, or page. The syntax can be deployed from any row, column, or page heading in a grid that is designated as a custom heading. To designate a custom heading, click the page, row heading, or column heading, and then select Custom Heading on the property sheet. You can add the function in the space provided below the custom heading.

  ```
  <<MemberAlias("DimName")>>
  ```
The following syntax returns the alias of a grid or user POV. The syntax can be deployed from any text cell or text box object and can reference any grid and corresponding POV in a current report.

```
<<MemberAlias("GridName", "POV DimName")>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of the grid</td>
</tr>
<tr>
<td>Row/Col/Page</td>
<td>Numeric value representing the index of member combinations on all page dimensions of the grid. The first page dimension combination in a grid has an index of 1, the second page dimension combination has an index of 2, and so on.</td>
</tr>
<tr>
<td>DimName</td>
<td>Name of a dimension in the grid row, column, or page.</td>
</tr>
<tr>
<td>POV DimName</td>
<td>Name of the dimension on the grid or user POV.</td>
</tr>
</tbody>
</table>

Examples:

Yield the alias Diet Root Beer, which is the alias that is assigned to the Product dimension in Grid1, column A:

```
<<MemberAlias("Grid1", A, "Product")>>
<<MemberAlias(current, A, "Product")>>
```

Yield the alias for the Year dimension. The syntax is placed on the grid’s custom heading:

```
<<MemberAlias("Year")>>
<<MemberAlias(current)>>
```

Yield the alias for the Scenario dimension for the POV associated with Grid1:

```
<<MemberAlias("Grid1", "Scenario")>>
<<MemberAlias(current, "Scenario")>>
```

Note: MemberAlias only supports the Current keyword when used in a grid cell or in a text box object when specifying the Page dimension parameter. Other parameters (Grid, Row, Column) are not supported in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

**MemberDescription**

Note: MemberDescription is available only with Financial Management as the database connection

MemberDescription is a text function. Based on the syntax used, it returns the description of the member assigned to a dimension in the row, column, page, or Point of View (POV).
The following syntax returns the description of a row heading, column heading, or page. The syntax can be deployed from any heading cell, text cell or text box object and can reference any grid in a current report.

```
<<MemberDescription("GridName", Row/Col/Page, "DimName")>>
```

The following syntax returns the description in a row heading, column heading, or page. The syntax can be deployed from any row, column, or page heading in a grid that is designated as a custom heading. To designate a custom heading, click the page, row heading or column heading, and then select Custom Heading on the property sheet. You can add the function in the space provided below the custom heading.

```
<<MemberDescription("DimName")>>
```

The following syntax returns the description of a grid or user POV. The syntax can be deployed from any text cell or text box object and can reference any grid and corresponding POV in a current report.

```
<<MemberDescription("GridName", "POV DimName")>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of the grid.</td>
</tr>
<tr>
<td>Row/Col/Page</td>
<td>Numeric identifier of the row, the alphabetical identifier of the column, or the numeric identifier of the page.</td>
</tr>
<tr>
<td>DimName</td>
<td>Name of a dimension in the grid row, column, or page.</td>
</tr>
<tr>
<td>POV DimName</td>
<td>Name of the dimension on the grid or user POV.</td>
</tr>
</tbody>
</table>

Examples:

Return the description assigned to the Product dimension in Grid1, column A:

```
<<MemberDescription("Grid1", A, "Product")>>
<<MemberDescription(current, A, "Product")>>
```

Return the description for the Year dimension. The syntax is placed on the grid’s custom heading:

```
<<MemberDescription("Year")>>
<<MemberDescription(Current)>>
```

Return the description for the Scenario dimension for the POV associated with Grid1:

```
<<MemberDescription("Grid1","Scenario")>>
<<MemberDescription(current,"Scenario")>>
```

**Note:** MemberDescription only supports the Current keyword when used in a grid cell or in a text box object when specifying the Page dimension parameter. Other parameters (Grid, Row, Column) are not supported in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

222
**MemberName**

*MemberName* is a text function. Based on the syntax used, it returns the name of the member assigned to a dimension in the row, column, page, or Point of View (POV).

- The following syntax returns the name of a row heading, column heading, or page. The syntax can be deployed from any heading cell, text cell or text box object and can reference any grid in a current report.

  \[
  \text{<<MemberName("GridName", Row/Col/Page, "DimName")>>}
  \]

- The following syntax returns the name in a row heading, column heading, or page. The syntax can be deployed from any row, column, or page heading in a grid that is designated as a custom heading. To designate a custom heading, click the page, row heading, or column heading, and then select Custom Heading on the property sheet. You can add the function in the space provided below the custom heading.

  \[
  \text{<<MemberName("DimName")>>}
  \]

- The following syntax returns the name of a grid or user POV. The syntax can be deployed from any text cell or text box object and can reference any grid and corresponding POV in a current report.

  \[
  \text{<<MemberName("GridName", "POV DimName")>>}
  \]

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of the grid.</td>
</tr>
<tr>
<td>Row/Col/Page</td>
<td>Numeric identifier of the row, the alphabetical identifier of the column, or the numeric identifier of the page.</td>
</tr>
<tr>
<td>DimName</td>
<td>Name of a dimension in the grid row, column, or page.</td>
</tr>
<tr>
<td>POV DimName</td>
<td>Name of the dimension on the grid or user POV.</td>
</tr>
</tbody>
</table>

**Examples:**

Return the name assigned to the Product dimension in Grid1, column A:

\[
\text{<<MemberName("Grid1", A, "Product")>>}
\]
\[
\text{<<MemberName(current, A, "Product")>>}
\]

Return the name for the Year dimension. The syntax is placed on the grid’s custom heading:

\[
\text{<<MemberName("Year")>>}
\]

Return the name for the Scenario dimension for the POV associated with Grid1:

\[
\text{<<MemberName("Grid1", "Scenario")>>}
\]
\[
\text{<<MemberName(current, "Scenario")>>}
\]
**Note:** MemberName only supports the **Current** keyword when used in a grid cell or in a text box object when specifying the Page dimension parameter. Other parameters (Grid, Row, Column) are not supported in a text box object, header, or footer. See "Using the Current/Cur Keyword in a Text Function" on page 195.

### MemberProperty

**Note:** MemberProperty is applicable to Planning details and Essbase database connections.

**MemberProperty** is a text function. Based on the syntax used, it returns the member property value of a dimension in the row, column, page, or Point of View (POV). You can use this function to display a member property in a text row or column.

- The following syntax returns the member property of a row heading, column heading, or page. The syntax can be deployed from any heading cell, text cell or text box object and can reference any grid in a current report.

  `<<MemberProperty("GridName", Row/Col/Page, DimName, Property)>>

- The following syntax returns the member property in a row heading, column heading, or page. The syntax can be deployed from any row, column, or page heading in a grid that is designated as a custom heading. To designate a custom heading, click the page, row heading, or column heading, then select Custom Heading on the property sheet. You can add the function in the space provided below the custom heading.

  `<<MemberProperty("DimName", Property)>>`

- The following syntax returns the member property of a grid or user POV. The syntax can be deployed from any text cell or text box object and can reference any grid and corresponding POV in a current report.

  `<<MemberProperty("GridName", POV DimName, Property)>>`

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of the grid.</td>
</tr>
<tr>
<td>Row/Col/Page</td>
<td>Numeric identifier of the row, the alphabetical identifier of the column, or the numeric identifier of the page.</td>
</tr>
<tr>
<td>POV DimName</td>
<td>Name of the dimension on the grid or user POV.</td>
</tr>
<tr>
<td>Property</td>
<td>Any custom property (attribute dimensions).</td>
</tr>
</tbody>
</table>

**Examples:**

Return the member property assigned to the Product dimension in Grid1, column A:

`<<MemberProperty("Grid1", A, Product, Pkg Type)>>`

`<<MemberProperty(current, A, Product, Pkg Type)>>`

Return the member property for the Market dimension. The syntax is placed on the grid’s custom heading:
<<MemberProperty("Market", Population)>>
<<MemberProperty(current, Population)>>

Return the member property for the Market dimension for the POV associated with Grid1:
<<MemberProperty("Grid1", Market, CurrencyCategory)>>
<<MemberProperty(current, Market, CurrencyCategory)>>

Retrieve the Member Alias and Ounces property of a product in a custom heading. (To do this, select the heading, then on the Heading Row Properties sheet, select the Custom Heading option, and then click Functions).
<<MemberAlias(current, current, Product)>>:
<<MemberProperty(current, current, Product, Ounces)>>

The report output would resemble the following:

<table>
<thead>
<tr>
<th>Qtr 1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Code: Ounces_12</td>
<td>5.096</td>
</tr>
<tr>
<td>Diet Code: Ounces_12</td>
<td>1.399</td>
</tr>
<tr>
<td>Caffeine Free Code: Ounces_18</td>
<td>593</td>
</tr>
</tbody>
</table>

Note: MemberProperty only supports the Current keyword when used in a grid cell, not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

MemberQualifiedName

MemberQualifiedName is a text function used with Essbase data sources. Based on the syntax used, it returns the qualified name of a non-unique member assigned to a dimension in the row, column, page, or Point of View (POV). A non-unique member is a child of multiple parents.

For example, if a member named 100 that is a child of the Product1 member and the Product 2 member, then 100 is a non unique member name. MemberQualifiedName shows the qualified names in reports. The qualified member names in this example are [Product1].[100] and [Product2].[100].

- The following syntax returns the name of a row heading, column heading, or page. The syntax can be deployed from any heading cell, text cell or text box object and can reference any grid in a current report.
  <<MemberQualifiedName("GridName", Row/Col/Page, "DimName")>>
- The following syntax returns the qualified name in a row heading, column heading, or page. The syntax can be deployed from any row, column or page heading in a grid that is designated as a custom heading. To designate a custom heading, click the page, row heading, or column heading, and then select Custom Heading on the property sheet. You can add the function in the space provided below the custom heading.
  <<MemberQualifiedName("DimName")>>
The following syntax returns the name of a grid or user POV. The syntax can be deployed from any text cell or text box object and can reference any grid and corresponding POV in a current report.

\[
<<\text{MemberQualified\text{Name}("GridName","POV\ DimName")}}>>
\]

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of the grid.</td>
</tr>
<tr>
<td>Row/Col/Page</td>
<td>Numeric identifier of the row, the alphabetical identifier of the column, or the numeric identifier of the page.</td>
</tr>
<tr>
<td>DimName</td>
<td>Name of a dimension in the grid row, column, or page.</td>
</tr>
<tr>
<td>POV DimName</td>
<td>Name of the dimension on the grid or user POV.</td>
</tr>
</tbody>
</table>

Examples:

Return the qualified non-unique name assigned to a the Product dimension in Grid1, column A:

\[
<<\text{MemberQualified\text{Name}("Grid1", A, "Product")}}>>
\]

\[
<<\text{MemberQualified\text{Name}(current, A, "Product")}}>>
\]

Return the name for the Year dimension. The syntax is placed on the grid’s custom heading:

\[
<<\text{MemberQualified\text{Name}("Year")}}>>
\]

Return the qualified non-unique name for the Scenario dimension for the POV associated with Grid1:

\[
<<\text{MemberQualified\text{Name}("Grid1", "Scenario")}}>>
\]

\[
<<\text{MemberQualified\text{Name}(current, "Scenario")}}>>
\]

Note: \text{MemberQualified\text{Name} only supports the} Current \text{keyword when used in a grid cell; not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.}

**Page**

Page is a text function that returns the current page number of a printed report. Use this function in a text object.

Syntax:

\[
<<\text{Page()}}>>
\]

Example:

If the current page is eight, yield: Page 8.

Page\text{<<PAGE()}}>>
Note: Page works for printed reports only. Online reports display [[PageCount()] for the page number. The page axis on a grid is different from the page number in a printed report.

**PageCount**

PageCount is a text function that returns the total number of pages in a printed report. Use this function in a text object.

Syntax:

`<<PageCount()>>`

Example:

If the current page count is 6, yields: “The total number of pages is: 6”.

The total number of pages is: `<<PageCount()>>`

Note: PageCount works for printed reports only. Online reports display `[[PageCount()]]` for the page number.

**PlanningAnnotations**

PlanningAnnotations is a text function that retrieves the critical notes or comments associated with a cell’s Planning Unit from the Planning database connection. Planning Units are a combination of Scenario, Version, and Entity and are derived in Financial Reporting through cell references in a grid.

Syntax:

`<<PlanningAnnotations("GridName", Row, Column, Page, Attributes, Range)>>`

`<<PlanningAnnotations("GridName", Row, Column, Page, Attributes)>>`

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>(Required) Name of a grid.</td>
</tr>
<tr>
<td>Row</td>
<td>(Required) Numeric value representing the row number of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on.</td>
</tr>
<tr>
<td>Column</td>
<td>(Required) Letter that represents the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on.</td>
</tr>
<tr>
<td>Page</td>
<td>(Required) Numeric value representing the index of member combinations on all page dimensions of the grid. The first page dimension combination in a grid has an index of 1, the second page dimension combination has an index of 2, and so on.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Attributes</td>
<td>Any of the following values: All, Title, Author, Date, Text, PlanningUnit. (PlanningUnit is the combination of scenario, version, and entity and is derived from the GridName, Row, Column, and Page.) Arrange attributes in the preferred order of output, and separate each attribute with an ampersand (&amp;).</td>
</tr>
<tr>
<td>Range</td>
<td>Use the keywords All, Top, or Bottom. to select the number of annotations from the top or bottom of an output, or all annotations.</td>
</tr>
<tr>
<td></td>
<td>• All returns all annotations</td>
</tr>
<tr>
<td></td>
<td>• Top 5 returns the first five annotations</td>
</tr>
<tr>
<td></td>
<td>• Bottom 10 returns the last ten annotations</td>
</tr>
<tr>
<td></td>
<td>• Bottom 1 returns the last annotation</td>
</tr>
</tbody>
</table>

Example 1:

“All” is assumed for the Range parameter.

```plaintext
<<PlanningAnnotations("Grid Name", 1,a,1,All, All)>>
<<PlanningAnnotations("Grid Name", 100,AB,10,All, All)>>
<<PlanningAnnotations(cur, cur,a,cur,Text & Title & Author, Top 5)>>
<<PlanningAnnotations(cur, 315, AB, 255, Text&Title&Author, Top 5)>>
<<PlanningAnnotations(cur, cur A, Cur, PlanningUnit, Top 5)>>
<<PlanningAnnotations("cur, 123, ABC, 101, PlanningUnit, Top5)>>
<<PlanningAnnotations("Gridl, 1, current, cur, Title & Text, Bottom 10)>>
<<PlanningAnnotations("Gridl, 105, ABC, cur, Title & Text, Bottom 10)>>
<<PlanningAnnotations("Current", 34, BB, cur, "All", Top 40)>>
<<PlanningAnnotations(cur, cur, A cur, "Text & Title & Author", Top 5)>>
<<PlanningAnnotations(cur, cur, A cur, "Text&Title&Author, "bottom15")>>
<<PlanningAnnotations(cur, 1(3), A(B), cur, "Title&Author&Date&Text, "All")>>
<<PlanningAnnotations(cur, 1(3), A(B), cur, "Title&Author&Date&Text, "All")>>
```

Example 2:

Retrieve annotation text in row 1, column A, of a grid on the current page. Display the bottom three annotations and all attributes associated with the annotation (title, author, date, text, and planning unit).

Use this syntax in a text cell:

```plaintext
<<PlanningAnnotations(Current, 1, A, Current, All, Bottom 3)
```

Annotations are returned chronologically in descending order by date, with the most recent annotations on top and the oldest annotation on the bottom. The resulting annotation text resembles the following:

**Title:** Status - Under Review  
**Author:** John Smith  
**Date:** Mar 25, 2003 10:32:49 AM  
**Planning Unit:** Budget, 1st Draft, East  
**Text:** Please review and approve

______________
Title: Status - Not Signed Off
Author: Mary Brown
Date: Mar 21, 2003 2:59:11 PM
Planning Unit: Budget, 1st Draft, West
Text: Sorry, Try Again

Title: Status - Under Review
Author: Admin
Date: Mar 21, 2003 2:54:16 PM
Planning Unit: Budget, 1st Draft, South
Text: Please review and approve budget for 1st draft

Note: PlanningAnnotations only supports the Current keyword when used in a grid cell; not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

Specifying Demarcations Between Annotations

The default demarcation symbols that separate Planning Details unit annotations are a series of em dash characters (—).

You can specify another demarcation symbol by configuring the PUASeparator in the JConsole file. For information on JConsole, see “Property Information” in the Oracle Hyperion Financial Reporting Administrator’s Guide.

Note: The following procedure affects all reports that retrieve Planning Details unit annotations.

➢ To enter new demarcation symbols:

1 Log off Financial Reporting.
2 Open the JConsole file for editing.
3 Locate PUASeparator and repeat the number of demarcation symbols to use; for example:

   PUASeparator= — — — — — — — — — — — — — — — — — —

4 Save and close the file.
5 Restart the Report Server service, and then log on to Financial Reporting.
**ProcessManagementStatus**

*ProcessManagementStatus* is a text function that returns the status of a cell in a Financial Management database connection. Possible process levels are: Not Started, First Pass, Submitted, Approved, Published, Not Supported, Review Levels 1 through 10. Use *ProcessManagementStatus* in a text box, a text cell, or a text row or column heading. *ProcessManagementStatus* is available only with Financial Management as the database connection.

**Syntax:**

\[ \textless \text{ProcessManagementStatus("GridName", Row, Col, Page\textgreater)} \]

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridName</td>
<td>Name of a Financial Management grid. <em>GridName</em> must be enclosed in quotation marks.</td>
</tr>
<tr>
<td>Row</td>
<td>Numeric value representing the row index of the grid. The first row in a grid has an index of 1, the second row has an index of 2, and so on.</td>
</tr>
<tr>
<td>Col</td>
<td>Letter representing the column of the grid. The first column in a grid has an index of A, the second column has an index of B, and so on.</td>
</tr>
<tr>
<td>Page</td>
<td>Numeric value representing the page dimension index of the grid. The first page dimension in a grid has an index of 1, the second page dimension has an index of 2, and so on.</td>
</tr>
</tbody>
</table>

**Example 1:**

Status for the cell in row 21, column B on page 1 of Grid1:

\[ \textless \text{ProcessManagementStatus("Grid1", 21,B,1\textgreater)} \]

\[ \textless \text{ProcessManagementStatus(Current,21,B,1\textgreater)} \]

**Example 2:**

Display the status for all cells in column B, on page 1 of the current grid:

\[ \textless \text{ProcessManagementStatus(Current,Current,B,1\textgreater)} \]

**Note:** *ProcessManagementStatus* only supports the *Current* keyword when used in a grid cell; not in a text box object, header, or footer. See “Using the Current/Cur Keyword in a Text Function” on page 195.

**ReportAuthor**

*ReportAuthor* is a text function that returns the user name of the person who created the report. Use this function in a text box, text cell, row, or column heading.

**Syntax:**

\[ \textless \text{ReportAuthor()} \]

230
Example:

Return the name of the report author:

<<ReportAuthor()>>

**ReportCreated**

ReportCreated is a text function that returns the date a report was created. Use this function in a text box, text cell, row, or column heading.

Syntax:

<<ReportCreated("format","TimeZoneId")>>

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Characters, enclosed in quotation marks, that define the format of the date and time. Format defaults to the user preference for date /time. Valid values are those accepted by Java SimpleDateFormat, “user”, or empty (“user”). For a detailed description of date and time formats, see Table 26 on page 207.</td>
</tr>
<tr>
<td>TimeZoneId</td>
<td>Defaults to user preferences for the time zone. Valid values are those accepted by Java TimeZone.getTimeZone(), or the server time zone. For example, for TimeZone.getTimeZone() to specify the east coast, enter America/New_York.</td>
</tr>
</tbody>
</table>

**Note:** The date and time format is case-sensitive. ReportCreated returns a value only after the report is saved.

Example:

Insert the report creation date in the body of a report:

<<ReportName()>> - Created on <<ReportCreated("d-MM-yy")>>

**ReportDesc**

ReportDesc is a text function that returns the description of the current report. Use this function in a text box, text cell, row, or column heading.

Syntax:

<<ReportDesc()>>

Example:

Insert a report description in the body of a report:

<<ReportDesc()>>
Note: You can define a report description when you save the report. Afterward, you can change the description in the repository or when you save the report again with the Save As command.

ReportFolder

ReportFolder is a text function that returns the path of the folder where the report is located. Use this function in a text box, text cell, row, or column heading.

Syntax:

```
<<Reportfolder()>>
```

Example:

Insert the path of the folder where the report is located:

```
<<Reportfolder()>>
```

Considerations when using ReportFolder:

- The report must be located in a folder other than the root folder. The root folder is not considered a report folder and returns an empty string.
- The report must be saved to a folder in order to return the correct string. An unsaved report returns an empty string.
- If you open a report containing a ReportFolder function, save it to a new folder. When you use Print Preview to preview the report, the previous location of the folder is displayed. This is by design for performance reasons. To reevaluate the report, modify a grid or report object, then use Print Preview to see the new folder location.

ReportModified

ReportModified is a text function that returns the date the current report was last modified. Use this function in a text box, text cell, row, or column heading.

Syntax:

```
<<ReportModified("format","TimeZoneId")>>
```

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>format</td>
<td>Characters, enclosed in quotation marks, that define the format of the date and time. Format defaults to the user preference for date/time. Valid values are those accepted by Java SimpleDateFormat, &quot;user&quot;, or empty (&quot;user&quot;). For a detailed description of date and time formats, see Table 26 on page 207.</td>
</tr>
<tr>
<td>TimeZoneId</td>
<td>Defaults to user preferences for the time zone. Valid values are those accepted by Java TimeZone.getTimeZone(), or the server time zone. For example, for TimeZone.getTimeZone() to specify the east coast, enter: America/New_York.</td>
</tr>
</tbody>
</table>

232
Note: The date and time format is case-sensitive. ReportModified returns a value only after the report is saved.

Example:
Insert the date Jan 19, 2013. (This is the date that the report was last modified.)
Report Modified: <<ReportModified("MMM dd, yyyy")>>

ReportModifiedBy

ReportModifiedBy is a text function that returns the user name of the last user to save the report. Use this function in a text box, text cell, row, or column heading.

Syntax:
<<ReportModifiedBy()>>

Example:
Insert the name of the last user to save the report:
<<ReportModifiedBy()>>

ReportName

ReportName is a text function that returns the name of the current report. Use this function in a text box, text cell, row, or column heading.

Syntax:
<<ReportName()>>

Example:
Insert the report name:
<<ReportName( )>>

ReportRunBy

ReportRunBy is a text function that returns the user name of the user who is running the report. Use this function in a text box, text cell, row, or column heading.

Syntax:
<<ReportRunBy()>>

Example:
Insert the name of the user who is running the report:
RetrieveValue

RetrieveValue functions in existing reports are no longer supported and must be manually updated to use the GetCell function.

Conditional Functions

Subtopics
- IfThen, If
- Conditional Operators
- Complex Conditions

IfThen, If

IfThen is a conditional function that returns a value when the condition equals True, and another value when the condition equals False.

Syntax:
IfThen(Condition, TrueParameter, FalseParameter)

- Condition is a logical expression that evaluates to true or false. Full conditional logic can be used as well as complex Boolean operators (And, Not, and Or). A condition can also test for #missing and #error values.
- TrueParameter and FalseParameter are expressions that are evaluated based on the outcome of the condition.

Conditional Operators

When using the conditional operators in Table 27 consider the following:
- Expression can be any valid formula expression. The expression can be any combination of a constant (integer or real number), a reference, or another function.
- Reference can be any valid reference; thus the IFNN reference property can be utilized as part of the reference.
- Condition can be any valid condition applied to the complex conditions And, Not, and Or. These operators can have embedded conditions. (And, Not, and Or operators require surrounding parentheses.)
- When any expression within the condition returns an #error or #missing value, the If function returns #missing or #error. This does not apply when you use the IsMissing, IsError, or IsNonNumeric conditions.
<table>
<thead>
<tr>
<th>Conditional Operator</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
</table>
| Equal To            | \( \text{expression} = \text{expression} \) | Tests if the left expression is equal to the right expression.  
  Note: If rounding is required, use the Round function.  
  Example:  
  \[
  1 = 4
  \]  
  Returns false |
| Greater Than        | \( \text{expression} > \text{expression} \) | Tests if the left expression is greater than the right expression.  
  Example:  
  \[
  1 > 4
  \]  
  Returns false |
| Greater Than or     | \( \text{expression} \geq \text{expression} \) | Tests if the left expression is greater than or equal to the right expression.  
  Note: The correct syntax is “\(\geq\)”. The syntax “\(\Rightarrow\)” is not supported.  
  Example:  
  \[
  1 \geq 4
  \]  
  Returns false |
| Equal To            | \( \text{expression} \leq \text{expression} \) | Tests if the left expression is less than or equal to the right expression.  
  Note: The correct syntax is “\(\leq\)”. The syntax “\(<\)" is not supported.  
  Example:  
  \[
  1 \leq 4
  \]  
  Returns true |
| Less Than           | \( \text{expression} < \text{expression} \) | Tests if the left expression is less than the right expression.  
  Example:  
  \[
  1 < 4
  \]  
  Returns true |
| Less Than or        | \( \text{expression} \leq \text{expression} \) | Tests if the left expression is less than or equal to the right expression.  
  Note: The correct syntax is “\(\leq\)”. The syntax “\(\leq\)" is not supported.  
  Example:  
  \[
  1 \leq 4
  \]  
  Returns true |
| Not Equal To        | \( \text{expression} \neq \text{expression} \) | Tests if the left expression is not equal to the right expression.  
  Note: If rounding is required, use the Round function.  
  Example:  
  \[
  1 \neq 4
  \]  
  Returns true |
<table>
<thead>
<tr>
<th>Conditional Operator</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>IsMissing</td>
<td>IsMissing (reference)</td>
<td>Tests if the reference contains a #missing result.</td>
</tr>
<tr>
<td></td>
<td>IsMiss (reference)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> If the reference is an expanded row or column, then all resulting cells must be #missing in order for the condition to be true.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IsMissing([1])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 1 has a #missing value.</td>
</tr>
<tr>
<td>IsError</td>
<td>IsError (reference)</td>
<td>Tests if the reference contains an #error result.</td>
</tr>
<tr>
<td></td>
<td>IsErr (reference)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> If the reference is an expanded row or column, all resulting cells must be #error in order for the condition to be true. Only formula rows and columns can result in #error.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IsError([2])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 2 has a #error value.</td>
</tr>
<tr>
<td>IsNonNumeric</td>
<td>IsNN (reference)</td>
<td>Tests if the reference contains a #missing or #error results.</td>
</tr>
<tr>
<td></td>
<td>IsNonNumber (reference)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IfNN (reference)</td>
<td><strong>Note:</strong> If the reference is an expanded row or column, all resulting cells must be #missing and/or #error in order for the condition to be true.</td>
</tr>
<tr>
<td></td>
<td>IfNonNumber (reference)</td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IsNN([3])</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true if row 3 has a #missing or #error value.</td>
</tr>
<tr>
<td>Parenthesis</td>
<td>(condition)</td>
<td>Groups a condition.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1 &gt; 4)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns false</td>
</tr>
</tbody>
</table>

**Complex Conditions**

<table>
<thead>
<tr>
<th>Complex Conditions</th>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>And</td>
<td>(condition AND condition)</td>
<td>Compares two conditions. Returns true if all conditions are true.</td>
</tr>
<tr>
<td></td>
<td>(condition &amp; condition)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1 &gt; 4 AND 5 &gt; 2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns false</td>
</tr>
<tr>
<td>Not</td>
<td>NOT (condition)</td>
<td>Negates the result by reversing the result of the condition.</td>
</tr>
<tr>
<td></td>
<td>! (condition)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Example:</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Not (1 &gt; 4)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Returns true</td>
</tr>
</tbody>
</table>
Complex Conditions

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>(condition OR condition)</td>
<td>Compares two conditions. Returns true if any of the conditions are true.</td>
</tr>
<tr>
<td>(condition</td>
<td>condition)</td>
</tr>
<tr>
<td>(condition</td>
<td>condition)</td>
</tr>
</tbody>
</table>

Example:
(1 > 4 OR 5 > 2)  
Returns true

Complex conditions And, Or, and Not are fully supported. However, they must be surrounded by parentheses.

Valid example:
If ( ([A] > [B] and [A] > 1000), [A], [B])

Invalid example:
If ( [A] > [B] and [A] > 1000, [A], [B])

Financial Functions

Subtopics

- Rank
- Variance / Var
- VariancePercent/VarPer

Rank

Rank is a financial function that provides a rank value for a value in a specified range. Rank is processed by Financial Reporting and does not depend on the database connection.

Syntax:

**Rank** ([Reference], Order)

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>The range of cells, rows, or columns to rank, with letters identifying columns and numbers identifying rows. For example, specify [A,1:5] to rank the values for rows 1 through 5 in column A. You can use the .ifNN property with a range of cells to assign numbers to any cells with nonnumeric values so that those cells can be ranked. For example, you can use .ifNN(-1) to assign the value -1 to any cell with a missing value.</td>
</tr>
<tr>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| **Order** | Indicates the order by which the values are ranked. The lowest value ranked in ascending order receives a rank result of 1. The largest value ranked in descending order receives a rank result of 1. The order can be indicated by any of the following keywords or values:  
  - Ascending  
  - Descending  
  - Asc  
  - Des  
  - Desc  
  - 1 (the number 1 is the same as "ascending")  
  - 0 (zero is the same as "descending")  
  The keywords are not case-sensitive.  
**Note:** Do not enclose the number or keyword indicating order in quotation marks. |
| **Unique** | (Optional) A Boolean keyword indicating how to treat equal values in the Reference parameter where:  
  - **false** (or omitted) — equal values receive the same ranking; ranked results may be duplicated  
  - **true** — equal values receive a unique ranking; there are no duplicate rankings. Values in the Reference parameter are ranked on a first come, first ranked basis. For example, if values in rows 2 and 5 are equal, the value in row 2 is ranked before that of row 5. |

**Examples:**

This formula in column B ranks the values in rows 1 through 5 in column A in descending order:

```
Rank([A;1:5], descending)
```

The result might be as follows:

<table>
<thead>
<tr>
<th>East</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cola</td>
<td>16</td>
</tr>
<tr>
<td>Fruit Drinks</td>
<td>23</td>
</tr>
<tr>
<td>Beer</td>
<td>16</td>
</tr>
<tr>
<td>Diet</td>
<td>missing</td>
</tr>
<tr>
<td>Root Beer</td>
<td>0</td>
</tr>
</tbody>
</table>

When two values are equal, they receive the same rank value. In the example above, Cola and Beer have the same value and therefore the same rank.

This formula in column B assigns the value of -1 to any nonnumeric value so it can be ranked:

```
Rank([A;1:5].ifNN(-1), descending)
```

In the following result, the missing value now has a rank of 5:
<table>
<thead>
<tr>
<th></th>
<th>East</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cola</td>
<td>16</td>
<td>2</td>
</tr>
<tr>
<td>Fruit Drinks</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>Beer</td>
<td>16</td>
<td>2</td>
</tr>
<tr>
<td>Diet</td>
<td>missing</td>
<td>5</td>
</tr>
<tr>
<td>Root Beer</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

Example:
The following example builds on the previous example explaining how the new “unique” parameter affects the results:

This formula in column B assigns the value of -1 to any nonnumeric value so it can be ranked, and also indicates that each ranking should be unique:

\[
\text{Rank}([\text{A,1:5}].\text{ifNN}(-1), \text{descending}, \text{true})
\]

In the following result, the missing value now has a rank of 5, and Beer has a value of 3 (even though it has the same data value as Cola):

<table>
<thead>
<tr>
<th></th>
<th>East</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cola</td>
<td>16</td>
<td>2</td>
</tr>
<tr>
<td>Fruit Drinks</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>Beer</td>
<td>16</td>
<td>3</td>
</tr>
<tr>
<td>Diet</td>
<td>missing</td>
<td>5</td>
</tr>
<tr>
<td>Root Beer</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

**Variance / Var**

Variance is a financial function that evaluates the difference between the specified values based on account type for the current account. For example, for an Income, Flow, Asset, or Balance account, a positive result represents an increase, so the result appears as a positive number. For Expense or Liability accounts, a positive result represents a decrease, so the result appears as a negative number.

This function is available for standard Essbase, Planning details, and Financial Management database connections.

- For Essbase, you can use members of an Accounts type dimension that are tagged as Expense or Non Expense.
- You can use the following UDA (User Defined Attributes) Accounts types that are also defined as Asset, Liability, Equity, Revenue, and Expense.
For Financial Management, you can use the following Accounts types: Asset, Liability, Income, Expense, Flow, Balance, and Balance Recurring.

Syntax:

Var(reference1, reference2)

where reference1 and reference2 are references to a row, column, or cell that correspond to members of the same Account dimension whose variance results are to be calculated.

### Expected Results Using Essbase

<table>
<thead>
<tr>
<th>Column A</th>
<th>Column B</th>
<th>Var ([A] , [B])=0</th>
<th>Var ([A] , [B])&gt;0</th>
<th>Var ([A] , [B])&lt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non Expense</td>
<td>Non Expense</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>

### Expected Results Using Planning

<table>
<thead>
<tr>
<th>Column A</th>
<th>Column B</th>
<th>Var ([A] , [B])=0</th>
<th>Var ([A] , [B])&gt;0</th>
<th>Var ([A] , [B])&lt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Equity</td>
<td>Equity</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Revenue</td>
<td>Revenue</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>

### Expected Results Using Financial Management

<table>
<thead>
<tr>
<th>Column A</th>
<th>Column B</th>
<th>Var ([A] , [B])=0</th>
<th>Var ([A] , [B])&gt;0</th>
<th>Var ([A] , [B])&lt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Income</td>
<td>Income</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Flow</td>
<td>Flow</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Column A</td>
<td>Column B</td>
<td>Var ([A], [B]) (\geq 0)</td>
<td>Var ([A], [B]) (&gt; 0)</td>
<td>Var ([A], [B]) (&lt; 0)</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>----------------------------</td>
<td>-------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>Balance</td>
<td>Balance</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Balance Recurring</td>
<td>Balance Recurring</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
</tbody>
</table>

**Variance Behavior**

Variance expects comparison of the same account type. When you compare two different account types, like Sales & Expense, Variance performs the math without applying the logic of the account type. For example:

<table>
<thead>
<tr>
<th>Sales</th>
<th>Expense</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>-400</td>
<td>100</td>
<td>-500</td>
</tr>
</tbody>
</table>

- In Essbase and Planning, when the Variance function is applied to a dimension that is not tagged as an Accounts type, an #error results at runtime.
- In Essbase, when the Variance function is applied to a dimension that is not tagged as an Accounts type, an #error results at runtime.
- #missing is treated as zero (0), unless specified differently using the `ifnonnumber` property.
- #error results in #error, unless specified differently using the `ifnonnumber` property.

**Examples**

Variance accepts cell, column, or row references only. For more information, see “Row, Column, or Cell Reference Arguments” on page 175.

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample syntax referencing a column:</td>
<td>Var ([A], [B])</td>
</tr>
<tr>
<td>Sample syntax referencing a row:</td>
<td>Var ([3], [4])</td>
</tr>
<tr>
<td>Sample syntax referencing a cell:</td>
<td>Var (Cell [3,A], [3,B])</td>
</tr>
</tbody>
</table>

In this example, the variance between column A (Actual) and column B (Budget) is calculated as:

\[
\text{Var([A],[B])}
\]

This example produces the following report:

<table>
<thead>
<tr>
<th></th>
<th>Year</th>
<th>Product</th>
<th>Market</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sales (Income)</td>
<td>400,855</td>
<td>373,080</td>
<td>27,775</td>
<td></td>
</tr>
<tr>
<td>COGS (Expense)</td>
<td>179,336</td>
<td>158,940</td>
<td>-20,396</td>
<td></td>
</tr>
</tbody>
</table>
**VariancePercent/VarPer**

VariancePercent is a financial function that evaluates the difference, in percent, between the specified values based on account type for the current account. For example, for an Income, Flow, Asset, or Balance account, a positive result represents an increase, so the result appears as a positive number. For Expense or Liability accounts, a positive result represents a decrease, so the result appears as a negative number.

This function is available for standard Essbase, Planning, and Financial Management database connections.

- In Essbase, you can use members of an Accounts type dimension that are tagged as "Expense" or "Non Expense".
- In Planning, you can use Essbase along with the following UDA (User Defined Attributes) Accounts types that are also defined as Asset, Liability, Equity, Revenue, and Expense.
- In Financial Management, you can use the following Accounts types: Asset, Liability Income, Expense, Flow, Balance, Balance Recurring.

**Syntax:**

\[ \text{VarPer}(\text{reference1}, \text{reference2}) \]

where \( \text{reference1} \) and \( \text{reference2} \) are references to a row, column, or cell that correspond to members of the same Account dimension whose VariancePercent results are calculated.

**Expected Results Using Essbase**

<table>
<thead>
<tr>
<th>Col A</th>
<th>Col B</th>
<th>( \text{VarPer}([A], [B])=0 )</th>
<th>( \text{VarPer}([A], [B])&gt;0 )</th>
<th>( \text{VarPer}([A], [B])&lt;0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non Expense</td>
<td>Non Expense</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>

**Expected Results Using Planning**

<table>
<thead>
<tr>
<th>Col A</th>
<th>Col B</th>
<th>( \text{VarPer}([A], [B])=0 )</th>
<th>( \text{VarPer}([A], [B])&gt;0 )</th>
<th>( \text{VarPer}([A], [B])&lt;0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Equity</td>
<td>Equity</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Revenue</td>
<td>Revenue</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
</tbody>
</table>
# Expected Results Using Financial Management

<table>
<thead>
<tr>
<th>Col A</th>
<th>Col B</th>
<th>VarPer ([A], [B])=0</th>
<th>VarPer ([A], [B])&gt;0</th>
<th>VarPer ([A], [B])&lt;0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset</td>
<td>Asset</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Liability</td>
<td>Liability</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Income</td>
<td>Income</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Expense</td>
<td>Expense</td>
<td>0</td>
<td>Returns a negative value</td>
<td>Returns a positive value</td>
</tr>
<tr>
<td>Flow</td>
<td>Flow</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Balance</td>
<td>Balance</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
<tr>
<td>Balance Recurring</td>
<td>Balance Recurring</td>
<td>0</td>
<td>Returns a positive value</td>
<td>Returns a negative value</td>
</tr>
</tbody>
</table>

## VariancePercent Behavior

VariancePercent expects comparison of the same account type. When you compare two different account types, like Sales & Expense, the VariancePercent function performs the straight math without applying the logic of the account type. For example:

<table>
<thead>
<tr>
<th>Sales</th>
<th>Expense</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>-400</td>
<td>100</td>
<td>-5.</td>
</tr>
</tbody>
</table>

- In Essbase and Planning, when VariancePercent is applied to a dimension that is not “Accounts”, an #error results at runtime.
- #missing is treated as zero (0), unless specified differently using ifnonnumber property.
- #error is results in #error, unless specified differently using ifnonnumber property.

## Examples

VariancePercent accepts, cell, column, or row references only. See “Row, Column, or Cell Reference Arguments” on page 175.

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample syntax referencing a column:</td>
<td>VarPer ([A], [B])</td>
</tr>
<tr>
<td>Sample syntax referencing a row:</td>
<td>VarPer ([3], [4])</td>
</tr>
<tr>
<td>Sample syntax referencing a cell:</td>
<td>VarPer (Cell [3,A], [3,B])</td>
</tr>
</tbody>
</table>

In this example, the VariancePercent between column A (Actual) and column B (Budget) is calculated as follows:

VarPer([A],[B])
This example produces the following report:

<table>
<thead>
<tr>
<th>Year</th>
<th>Product</th>
<th>Market</th>
<th>Actual</th>
<th>Budget</th>
<th>Variance</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sales(Income)</td>
<td>400,855</td>
<td>373,080</td>
<td>7%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COGS (Expense)</td>
<td>179,336</td>
<td>158,940</td>
<td>-13%</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Explore Repository stores Financial Reporting objects (reports, snapshots, grids, charts, image objects, text object, row and row and column templates). You can also print and print preview reports and sort items by clicking a column field header.

Items saved to the repository can be used multiple times. For example, you can use a report as a basis for designing another report, or you can use a report object, such as an image object, in several reports.

To access the all available functionality in Workspace, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

This chapter contains the following topics:

### Creating Folders

You create and use folders to help organize items in the repository. You can add folders at multiple levels while in the repository or while saving reports and report objects. When a user adds an object in the repository, by default only that user and an administrator can access the object.
To allow other users to see an object, the originator or a Report Designer must assign access to other users or groups. You must have Report Designer or Global Administrator roles to create a folder.

➢ To create folders in the repository:

   
   The repository is displayed.

2. Click the location within the repository where you want the folder to reside.

3. Select File, then New, and then Folder. The default folder name is titled New Folder.

4. Enter a folder name in the Enter New Folder Name text box. You can use uppercase and lowercase letters, numbers, and spaces.

5. Press OK to activate the folder.

### Duplicating Objects in the Repository

Use Workspace to duplicate objects in the repository.

For detailed information, see the Oracle Hyperion Enterprise Performance Management Workspace User's Guide.

### Filtering Report Objects

You can filter repository objects including reports, snapshots, and saved report objects to help you easily locate an object. You filter report objects by the type of object you seek. The system searches all folders in the repository for matching object types and displays them, by folder, in a tree hierarchy. A variety of filter combinations is provided for filtering.

➢ To sort by object types:

1. Select View, and then Display Items of Type.

2. Select an object type (All Items, Report, Snapshot, or Saved Report Object) to display in the repository.

3. Optional: Repeat steps 1 and 2 to select additional object types.

To redisplay all object types in the repository, select View, then Display Items of Type, and then All Items.

You can also sort objects by clicking the column headings. For example, you can display the oldest or newest reports first by clicking the Modified heading, or display report objects alphabetically by clicking the Name heading.
Cutting, Copying, and Pasting in the Repository

In Workspace, you can use the cut-and-paste feature to move report objects between different folders in the repository. Your ability to cut and paste depends on your access privileges.

For detailed information, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

Managing Database Connections

The Database Connection Manager is in Workspace. It provides a means to manage reports, books, snapshots, snapshot books, saved reports objects (row/column templates, grids), batches, and scheduled batches. Reports and report objects can be moved from a development environment to a production environment, or between production servers.

Use Workspace to manage the database connections. For detailed information on adding, editing, and deleting database connections, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

Note: The Financial Reporting database manager does not manage Web Analysis databases.

Exporting Repository Objects and Report Objects

In Workspace Explore, you can export saved objects, report objects and folders to a specific directory on your computer.

During export, you can specify the format of the objects you are exporting. For information on exporting objects and report objects, see the Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

Importing Report Objects into the Repository

In Workspace Explore, you can import objects, reports, folders and zipped files previously exported to a file system.

For detailed information on importing objects, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.

Sending Links to Reports Through Email

Using Workspace Explore, you can use the email Link feature to send a URL link of objects or folders through email. The link can be opened in Workspace by a Financial Reporting user. While sending an email, you can also add and store a list of frequently used email addresses.

For detailed information on sending links to reports through email, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.
Deleting Items from the Repository

You delete items from the repository when no longer needed. You can permanently delete any items to which you have access privileges.

To delete items from the repository:

1. From the repository list, select the items to delete.

   **Tip:** You can filter the repository by selecting View, and then Display Items of Type. Select any of the following types: Report, Snapshot, or Saved Report Object.

   **Tip:** Press Shift to select multiple, consecutive items in the repository. Press Ctrl to select multiple, nonconsecutive items in the repository.

2. Select Edit, and then Delete.

3. Click Yes to confirm the deletion.

   **Note:** Before deleting a folder, you must delete all objects contained inside. You cannot delete a folder containing hidden objects due to insufficient security privileges.

Changing the Database Connection for Reports and Report Objects in the Repository

Using Workspace, you can change the database connection for reports and books in the repository. You may want to change the database connection when moving from a development environment server to a production environment server.

For information on changing database connections, see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide*.

Changing a Report Database Connection

When changing a report from an Essbase database connection to a Planning database connection or vice versa, there are several restrictions because the data source reference is the only change. There is no conversion.

- For a list of considerations to follow when changing the database connection from Essbase to Planning for a report, see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide*.

- For a list of considerations to follow when changing the database connection from Planning to , see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide*.
Managing Security for Objects in the Repository

Using Workspace, you can apply security to objects (reports, snapshot reports, books, snapshot books, and batches), folders, and report objects (chart objects, grid objects, text objects, grids, image objects, and graphics) in the repository. Objects, folders, and report objects are secured by restricting or allowing access privileges for certain users and groups. As a result, only certain designers or viewers can view objects in the repository from Financial Reporting. The roles and permissions that an administrator assigns to a user or group determine if access privileges are restricted in the repository.

For more information on assigning roles and access privileges, see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.*
Using Workspace, you can create, edit, and save batches if you have Administrator or Designer rights. Batches can also be scheduled, with full Batch Scheduler capability. You use batches to group and process sets of reports or books. You can send the output generated by running a batch to a printer or to the repository, and you can export both as HTML and PDF files. You can also email users the exported output.

You can define and save batches in the Batch Editor and copy and delete batches in the repository. Deleting a batch from the repository requires access rights to the batch. By default, only the scheduling user is assigned access rights to snapshots and snapshot books that are saved to the repository. Having access rights to a batch, you can run every report in the batch when scheduled. The scheduling user can assign access to other users and groups at the time of scheduling.

Batches have a batch POV, which you can modify while scheduling. You can also specify prompt information for the batch. The specified POV and prompt information provides values for the POV and prompts specified for each book and report contained in the batch. You can schedule batches for processing immediately or at a later time.

For complete information on batch processing, see Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.
Using Workspace, you can assemble reports and snapshots into a book, which enables you to generate output for those reports in one session. You can also configure your book to generate several versions of a report that are run for different member selections. By creating a book containing those reports, you can run all reports at once. The reports contained in the book can be printed or viewed individually or as an entire book.

There are two types of books that you can define and save to the Explore repository:

- **Books.** Reports, and snapshots can be added to a book. When you run a book, the data within the reports is dynamically retrieved from the database; the data within the snapshots remains static.

- **Snapshot books.** A book can be saved as a snapshot book. All reports are converted to snapshots containing static data for that period of time. The data contained in the reports, and a table of contents, is created when you save a book as a snapshot book. Any time a snapshot book is viewed, data-level security does not apply. Data-level security is applied when the snapshot book is created and is based upon the data level security that is set up for the user that saved the snapshot book.

Using Workspace, you can perform the following additional tasks with books and snapshot books from Financial Reporting:

- Use batches to group and process sets of books.
- Email links to the books or snapshot books from the repository.
- Change the data source for a book in the repository.
- Make an exact copy of books or snapshot books in the repository.
- Permanently delete books or snapshot books.
- Import books and snapshot books from a file system outside of the repository. When you import the book, it is saved within the Export repository. You can then perform any tasks on the book.
- Export books and snapshot books.

For complete information on the above topics, see *Oracle Hyperion Enterprise Performance Management Workspace User’s Guide.*
Support for Non ASCII User Accounts

It is a requirement that the Financial Reporting Studio logon page will look to the default Windows font to render the characters in this page. So, for example, to see Korean characters rendered properly in a Japanese operating system, you need to change the default system font for your Japanese machine to Korean. You will see some common system dialogs rendered using the default language. These are Windows dialogs where the font comes from the default language and not the Financial Reporting user’s language. So the best practice is to use a foreign language operating system that suits the user’s preferred language or to set the default Windows language to the preferred language.

POV Management

The Manage User POV utility is a command line utility where the administrator creates, views, updates, or deletes the Financial Reporting POV for one or more users. The utility is installed with Reporting Studio to the EPM_ORACLE_HOME\Products\financialreporting\bin directory as ManageUserPov.cmd and comes with a sample configuration file called ManageUserPOV.properties.
To run the utility, open a command prompt in the \EPM_ORACLE_HOME\Products\financialreporting\bin directory and run the ManageUserPov.cmd:

ManageUserPOV <Mode> <ConfigFileName>

where  Mode = Import, Export, or Delete (case insensitive)

where ConfigFileName = the name of the configuration file (the default name is ManageUserPov.properties located in the \EPM_ORACLE_HOME\Products\financialreporting\bin directory).

Example: ManageUserPOV Export ManageUserPov.properties

Two files must be predefined before executing the POV utility; the configuration file, such as ManageUserPov.properties, and, when importing, an import file such as ManageUserPovImport.xml. Sample files are provided in the <financialreporting> \bin directory:

- ManageUserPov.properties file. Use this file to define the report server, assign administrator and data source credentials, specify the import and export file names, and set the various options such as user and data source filters and whether or not to perform a dry run, or confirm deletion of User POVs. See the sample ManageUserPov.properties file for details.

- The ManageUserPovImport.xml file. An ImportFileName, such as ManageUserPovImport.xml. The XML file created during an export can also be edited and used for importing.

Import Behavior Guidelines

- Database connections which do not yet exist in the repository are created if the TYPE, SERVER, APPNAME, and DBNAME attributes are provided. Valid data source TYPEs are: Essbase, Financial Management, and Planning.

- All dimensions and members are validated against the data source to ensure that only valid User POVs are created or updated.

- Individual user access to a member is not validated; administrators should assign User POVs to only those users with access rights to view them.

- User POVs are only created for provisioned users.

- Partial User POVs are valid; only dimensions specified in the import file are updated.

Using the User POV Utility

The following instances describe how the User POV Utility is used.

Exporting User POVs

To export all existing User POVs:

1. Edit the ManageUserPov.properties file and specify the ReportServer.
Open a command prompt in the `financialreporting\bin` directory and type:
```
ManageUserPov.cmd Export ManageUserPov.Properties
```

When prompted, supply the Admin User and Password.
A `ManageUserPovExport.xml` file is created which contains all the current User POVs in the system. A sample console interaction:
```
EPM_ORACLE_HOME\Products\financialreporting\binManageUserPov.cmd Export ManageUserPov.properties
Please enter AdminUser : AdminUser
Please enter AdminPassword :*********
INFO : Starting Hyperion Financial Reporting ManageUserPOV Utility
INFO : Ending Hyperion Financial Reporting ManageUserPOV Utility
```

**Importing User POVs**

To update an existing User POV:
1. Follow the steps to export all existing User POVs.
2. Open the `ManageUserPovExport.xml` file and update the desired User POV in the file. For example, specify a Member Name="East" for the Market Dimension, for the User LoginID="TestUser", for the DATASOURCE_NAME="Essbase Sample Basic on TestServer"
3. Edit the `ManageUserPov.properties` file and specify these filters to perform the update for a single user and data source:
```
Users = TestUser
Datasources = Essbase Sample Basic on TestServer
```

Note: If the Datasource credentials are different from the Admin credentials, you should specify an asterisk (*) to prompt for them separately, such as:
```
DatasourceUser=*, DatasourcePassword=*
```
4. Save the file as `ManageUserPovImport.xml`.
5. Open a command prompt in the `financialreporting\bin` directory and type:
```
ManageUserPov.cmd Import ManageUserPov.properties
```
6. When prompted, supply the Admin User and Password and optionally the Datasource User and Password.
The User POV is updated with the new value for Market. A sample console interaction:
```
C:\Hyperion\financialreporting\bin>ManageUserPov.cmd Import
ManageUserPov.properties
Please enter AdminUser : AdminUser
Please enter AdminPassword :*********
Please enter DataSourceUser : TestUser
```
INFO : Starting Hyperion Financial Reporting ManageUserPOV Utility
INFO : Setting POV for user: TestUser
INFO : Processing DataSource: Essbase Sample Basic on TestServer
INFO : Processing Year = Year
INFO : Processing Measures = Measures
INFO : Processing Product = Product
INFO : Processing Market = East
INFO : Processing Scenario = Scenario
INFO : Processing Caffeinated = Caffeinated
INFO : Processing Ounces = Ounces
INFO : Processing Pkg Type = Pkg Type
INFO : Processing Population = Population
INFO : Processing Intro Date = Intro Date
INFO : Processing Attribute Calculations = Attribute Calculations
INFO : Ending Hyperion Financial Reporting ManageUserPOV Utility

Deleting User POVs

➤ To delete an existing User POV:

1. Edit the ManageUserPov.properties file and specify the ReportServer and desired filters. For example, to delete a single user “TestUser” POV for the data source “Essbase Sample Basic on TestServer”, specify:

   Users = TestUser
   Datasources = Essbase Sample Basic on TestServer

2. Open a command prompt in the EPM_ORACLE_HOME\Products\financialreporting\bin directory and enter: ManageUserPov.cmd Delete ManageUserPov.properties.

3. When prompted, supply the Admin User and Password.

   The User POV is deleted. A sample console interaction:

   EPM_ORACLE_HOME\Products\financialreporting\bin\ManageUserPov.cmd Delete ManageUserPov.properties
   PR_Studio_Installation_Directory\Products\financialreporting\bin\ManageUserPov.cmd Delete ManageUserPov.properties

   Please enter AdminUser : AdminUser
   Please enter AdminPassword :*********
INFO : Starting Hyperion Financial Reporting ManageUserPOV Utility

This action will delete 1 POV’s

Are you sure you want to proceed? Press 1 to Proceed and 0 to Cancel : 1

INFO : Ending Hyperion Financial Reporting ManageUserPOV Utility

Financial Reporting Initial User Preferences Definition

Initial User Preferences for Financial Reporting can be defined for all users through the JConsole.exe file. These settings will appear for an end user when logging into EPM Workspace for the first time. The settings can then be changed, if desired, in the User Preferences dialog box.

For information about JConsole, see the “Property Information” topic in the Oracle Hyperion Financial Reporting Administrator’s Guide.

The settings that can be defined are as follows:

# Specify a default user preference value for ruler unit.
# Value can be either Inches or Centimeters.
#com.hyperion.reporting.HRPrefs.Units=

# Specify a default user preference value for grid line color.
# Value is in the format of #RRGGBB where RR is hex value for red, GG is hex value for green and BB is blue as specified by HTML specifications.
#com.hyperion.reporting.HRPrefs.glcolor=

# Specify a default user preference value for grid line style.
# Value can be either Line or Dot.
#com.hyperion.reporting.HRPrefs.glstyle=

# Specify a default user preference value for grid line snapping.
# Value can be either True or False.
#com.hyperion.reporting.HRPrefs.glsnap=

# Specify a default user preference value for MS Office version.
#com.hyperion.reporting.HRPrefs.office_format=

# Specify a default user preference value for location of User POV bar.
# Value can be either above or viewpane.
#com.hyperion.reporting.HRPrefs.locationuserpov=

# Specify a default user preference value for user’s language.
# Value can be any ISO language code for supported languages.
Report Server Tasks

Subtopics

- Specifying the Maximum Number of Calculation Iterations
- Assigning Financial Reporting TCP Ports for Firewall Environments or Port Conflict Resolution
- Accessing Server Components Through a Device that Performs NAT

Financial Reporting Server configuration is performed as part of the installation process. The following topics describe additional tasks that can be performed.

Specifying the Maximum Number of Calculation Iterations

You can specify the maximum number of calculation iterations for all grids and cells on the MaximumCalculationIterations property in JConsole.exe file to resolve dependencies within references in formulas. During the calculation process of a grid, it may be necessary to evaluate a cell multiple times due to reference precedence. This usually occurs in grids with references to other grids. The maximum iteration property indicates the number of times a formula cell can be evaluated before it is marked as unresolved. Setting the maximum iteration property avoids the possibility of cells, with circular referencing, being evaluated an infinite
amount of times. Circular referencing occurs when one cell refers to another cell which then refers to the original cell.

If there are no circular references and calculation cells are returning #error, you can increase the maximum iteration property value. The default value for the maximum number of calculation iterations is 5. This file also contains comments to guide you in modifying this value, if necessary.

**Note:** For information on JConsole, see the “Property Information” topic in the *Oracle Hyperion Financial Reporting Administrator’s Guide.*

### Assigning Financial Reporting TCP Ports for Firewall Environments or Port Conflict Resolution

By default, Financial Reporting components communicate with each other through Remote Method Invocation (RMI) on dynamically assigned Transmission Control Protocol (TCP) ports. To communicate through a firewall, you must specify the port of each Financial Reporting component separated by the firewall in the JConsole.exe file, and then open the necessary ports in your firewall.

In addition, you may need to open ports for the Reports Server RDBMs, for data sources that you report against, and for LDAP/NTMLM for external authentication.

**Note:** Ports should be opened in the firewall only for Financial Reporting components that must communicate across the firewall. If the Financial Reporting components are not separated by a firewall, they can use the default dynamic port setting.

The Scheduler server defaults to 8299, this can be changed by modifying the SchedulerServer property and restarting. The Print Server defaults to 8297, this can be changed by modifying the PrintServers property and restarting. Once connected, all RMI Services can use anonymous ports by default for communication. Alternatively a range of ports can be configured for communication by setting RMI Port Range Lower and RMI Port Range Upper within the Financial Reporting configuration. The Financial Reporting properties are now stored in the HIT registry and can be modified using JConsole or Enterprise Manager. You can change the port assignments to use in a firewall environment for servers in the JConsole.exe file using RMI Port Range Upper and RMI Port Range Lower.

**Note:** For information on JConsole, see the “Property Information” topic in the *Oracle Hyperion Financial Reporting Administrator’s Guide.*

**Note:** If you are unable to connect to a Network Address Translated server from Financial Reporting Studio install outside the firewall, the workaround is to add the `DuseHostname=true` JVM parameter on the foundation server, and add the following property in the default-domain.cfg:

```properties
policies:iiop:server_address_mode_policy:publish_hostname="true".
```
Accessing Server Components Through a Device that Performs NAT

Subtopics

- Network Address Translation (NAT)
- Remote Method Invocation (RMI)
- Issue Using RMI Through Devices that Perform NAT
- Adding Required Java Arguments on UNIX Systems

The following topics discuss how to access server components through Network Address Translation (NAT).

Network Address Translation (NAT)

Network Address Translation (NAT) makes possible the use of a device, such as a router, to act as an agent between two networks whereby only one unique IP address is required to represent an entire group of computers.

Remote Method Invocation (RMI)

Communication between the client and server in Financial Reporting is achieved via Java’s Remote Method Invocation (RMI) protocol. By default, an RMI server program communicates with clients using the IP address of the computer on which it is running.

Issue Using RMI Through Devices that Perform NAT

The combination of Java RMI and NAT has an inherent issue, in that Java attempts to route client requests to the IP of the computer on which the RMI server is running, rather than to the masqueraded address supplied by NAT.

To work around this issue, the Java Virtual Machine (JVM) accepts two arguments that allow RMI server programs to communicate with clients using the IP of another computer or device, such as a router that performs NAT.

If your Report Client computers access Reports Server components though a device that masks outgoing packets through NAT (Network Address Translation), then you must follow the procedures listed below.

Adding Required Java Arguments on UNIX Systems

When using a firewall, the following procedures explain how Java arguments are added to UNIX systems.
For the WebLogic Web server:

1. Open `/…/Oracle/Middleware/user_projects/epmsystem1/bin/deploymentScripts/setCustomParamsFinancialReporting.sh` in a text editor, and add `-Djava.rmi.server.hostname=<IP or hostname of NAT device> -Djava.rmi.server.useLocalHostname=false` to the `JAVA_OPTIONS` variable declaration.

2. Start/restart the components.

## Essbase Ports

### Subtopics

- Checking the Current Essbase Connections
- Changing Settings in Essbase Configuration Files
- Calculating the Formula for the Maximum Number of Essbase Ports

This section describes the differences between Essbase ports and connections when running this release of Financial Reporting.

Important considerations:

- You are licensed by Essbase ports.
- A 100 concurrent user license for Essbase means 100 Essbase ports are licensed.
- An unlimited number of connections is allowed on each of those ports.
- The number of connections you open to Essbase is not relevant for licensing purposes. What matters is the number of Essbase ports.

When a user runs a report in Financial Reporting, connections are opened to Essbase. For performance optimization purposes, these connections are cached. When the connections become idle, a process is run periodically to close them.

The system administrator can modify the length of time before a connection is considered inactive (`MinimumConnectionInactiveTime`, default of 5 minutes) and the length of time before inactive connections are closed (`CleanUpThreadDelay`, default of 5 minutes) in the `JConsole.exe` file.

**Note:** for information on JConsole, see the “Property Information” topic in the *Oracle Hyperion Financial Reporting Administrator’s Guide*.

The number of ports used by Financial Reporting varies, depending on the configuration, as follows:

- If a Report Client such as the Windows UI runs a report, two Essbase connections are made; one for the Report Client, and one for the Report Server.
If the Report Client and Report Server are on the same computer, two Essbase connections using one Essbase port are made.

- The Report Client keeps the Essbase connection until the window with the report displayed is closed.
- The Report Server keeps this Essbase connection until the process is run to close idling open connections.
- When both connections are closed, the port is released.

If the Report Client and Report Server are on two different machines, two Essbase connections using two Essbase ports are made.

- The Report Client keeps the Essbase connection until the window with the report displayed is closed.
- The Report Server keeps this Essbase connection until the process is run to close idle open connections.
- When the Report Client connection is closed, the corresponding port for that connection is released.
- When the Financial Reporting connection is closed, the corresponding port for that connection is released.

When a Reporting Studio user such as the Browser UI runs a report, two Essbase connections are made: one for the web server and one for the Report Server.

- If the web server and Report Server are on the same computer, two Essbase connections using one Essbase port are made.
  - The web server keeps the Essbase connection until the process is run to close idle open connections.
  - The Report Server keeps this Essbase connection until the process is run to close idle open connections.
  - When both connections are closed, the port is released.

- If the web server and Report Server are on two different computers, two Essbase connections using two Essbase ports are made.
  - The web server keeps the Essbase connection until the process is run to close idling open connections.
  - The Report Server keeps this Essbase connection until the process is run to close idling open connections.
  - When the web server connection is closed, the corresponding port for that connection is released.
  - When the Report Server connection is closed, the corresponding port for that connection is released.

The recommended configuration is as follows:

- The Report Server and web server are installed on the same computer.
The Report Client is installed on several other computers. In this case, you must take two Essbase ports only for users working with the Report Client. All users connecting to view reports in EPM Workspace take a single Essbase port for each Essbase user, because the web server and Report Server are on the same computer.

**Checking the Current Essbase Connections**

To check for current connections in the Essbase server command window, type USERS in the command window. The list displays the current connections and ports currently used.

**Changing Settings in Essbase Configuration Files**

When adding or increasing the following Essbase client settings in the `essbase.cfg` file, Essbase Connection Time Outs are reduced:

- NETDELAY 1000
- NETRETRYCOUNT 1000

**Calculating the Formula for the Maximum Number of Essbase Ports**

The basic formulas for calculating the maximum number of Essbase ports you need for Financial Reporting are as follows:

- If EPM Workspace and Report Server are on the same computer:
  
  Number of Essbase ports = 2 X the number of Report Clients
  + the number of EPM Workspace users

- If EPM Workspace and Report Server are on different computers:
  
  Number of Essbase ports = 2 X the number of Report Clients
  + 2 X the number of EPM Workspaces

  **Note:** This formula is for Financial Reporting and does not consider other ways users might be connecting to Essbase; for example, the Application Manager, Oracle Hyperion Web Analysis, or the Excel Add-in. You must consider those potential port-takers separately. If they are used on the same computer as one of the Financial Reporting components, no extra ports are taken as long as the same Essbase user ID is being used.

Data source considerations are as follows:

- If you run a report with two data sources, your number of connections doubles, but the number of ports remains the same as described previously.
If you run a report with three data sources, your number of connections triples, but the number of ports remains the same as described previously.

If, after closing the report with two data sources, you run a report with a 3rd data source, your connections increases again but the number of ports does not change.

A user’s connection is open for at least five minutes and remains open for up to 10 minutes, assuming no new activity occurs during that time. If you have a limited number of Essbaseports, and many users are accessing Financial Reporting, you may want to lower both values to 30 seconds (30000).

**Scheduler Command Line Interface**

**Subtopics**

- Creating Batch Input Files
- Launching Batches from a Command Line
- Scheduling Batches Using an External Scheduler
- Encoding Passwords
- Modifying Attributes

Scheduler Command Line Interface is the process of launching a Financial Reporting batch input file from a command line. You can automate the process of launching batch input files using an external scheduler or launching batch input files after some external event occurs, such as the completion of a consolidation.

**Note:** For the Financial Reporting Scheduler to successfully send email notifications, a valid SMTP mail server must be specified in the JConsole.exe file.

**Creating Batch Input Files**

The batch input file specifies the options for the scheduled batch such as the name of the batch to be scheduled, output destinations, email notification information, POV settings, and others.

To create a batch input file:

1. Right-click a previously scheduled batch in the Batch Scheduler dialog box and choose Export for Command Line Scheduling.
2. Open the `mybatch.xml` where `mybatch` is the name of your batch input file.
3. Modify this file as needed by editing the values in the tags, see “Modifying Attributes” on page 268 for the commonly used attributes.
Launching Batches from a Command Line

You can use the ScheduleBatch.cmd command file provided in the financialreporting\bin directory to launch the batch specified in the batch input file against a Financial Reporting scheduler server.

To launch a batch from a command line prompt in the financialreporting\bin directory, enter the command by specifying the fully qualified name of the batch input file and the computer name or IP address of the Scheduler Server on which to schedule the batch, for example:

```
ScheduleBatch c:\DailyReports\mybatch.xml MySchedulerServer
```

where MyBatch.xml is the name of your batch input file and MySchedulerServer is the name or IP address of your scheduler server which is typically located on the same computer as the report server.

This launches a batch to run immediately against the scheduler server specified.

Scheduling Batches Using an External Scheduler

You can launch a batch on a periodic basis from an external scheduler. To do this, you set up your own command files and call them from the external scheduler.

For example, you might have a NightlyBatch.cmd file containing these lines:

```
call ScheduleBatch MgtSummaryBatch.xml hr_Server
call ScheduleBatch MgtDetailBatch.xml hr_Server
```

Encoding Passwords

User names and passwords are not present when you export the batch input file. To specify user ID or data source ID in the batch input file, use the following file to produce an encoded password for use in the batch input file:

- Windows—EncodePassword.cmd
- UNIX—EncodePassword

**Note:** This procedure is required.

To encode passwords:

1. Open the batch input file to modify the data source and user ID passwords.
2. From the command line, run the EncodePassword.cmd file.
3. Type `EncodePassword Password`, where Password is the new password you want to use.
4. Place the encoded password produced in the batch input file.
## Modifying Attributes

In a typical batch input file, there are very few attributes to modify. Most attributes are already set properly based on the originally scheduled batch. The following table lists attributes that you are most likely to modify for the associated XML tags.

### Table 35  Commonly Used Attributes

<table>
<thead>
<tr>
<th>Category</th>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>AUTHOR</td>
<td>Displays in the batch scheduler's User ID column and is a useful place to show a comment or the name of the XML file that generated the batch.</td>
</tr>
<tr>
<td>Email</td>
<td>ATTACH_RESULTS</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to attach PDF or HTML files generated to the email.</td>
</tr>
<tr>
<td></td>
<td>FAILURE_RECIPIENTS</td>
<td>Email to recipients if schedule batch failed</td>
</tr>
<tr>
<td></td>
<td>FAILURE_SUBJECT</td>
<td>Text if scheduled batch fails</td>
</tr>
<tr>
<td></td>
<td>RECIPIENTS</td>
<td>A comma-separated list of recipient’s email addresses.</td>
</tr>
<tr>
<td></td>
<td>SENDER</td>
<td>The sender’s email address.</td>
</tr>
<tr>
<td></td>
<td>SUBJECT</td>
<td>The subject of the email.</td>
</tr>
<tr>
<td>Credentials</td>
<td>DS_PASSWD</td>
<td>The encrypted data source password from an existing batch or that you generate using the command line utility.</td>
</tr>
<tr>
<td></td>
<td>DS_USER_NAME</td>
<td>The data source user whose credentials are used for running the reports/books in the batch.</td>
</tr>
<tr>
<td></td>
<td>HR_PASSWD</td>
<td>The encrypted Financial Reporting user password from an existing batch or that you generate using the command line utility.</td>
</tr>
<tr>
<td></td>
<td>HR_USER_NAME</td>
<td>The Financial Reporting user whose credentials are used for running the reports/books in the batch.</td>
</tr>
<tr>
<td>HTML and PDF output</td>
<td>HTML VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate HTML output for the batch.</td>
</tr>
<tr>
<td></td>
<td>PDF VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate PDF output for the batch.</td>
</tr>
<tr>
<td></td>
<td>HTML_EXPORT_HTML_FOLDER_LABEL</td>
<td>If exporting as HTML (Value=Yes), The path and folder to external directory.</td>
</tr>
<tr>
<td></td>
<td>PDF_EXPORT_HTML_FOLDER_LABEL</td>
<td>If exporting as PDF (Value=Yes), the path and folder to external directory.</td>
</tr>
<tr>
<td>Category</td>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Snapshot Output</td>
<td>SAVE_AS_SNAPSHOT VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to save the snapshot output in the repository.</td>
</tr>
<tr>
<td>SAVE_NAME</td>
<td>The Folder Name where the Snapshots are to be stored. This must be specified in ReportStore:\ format. If SAVE_NAME = “”, the snapshot output is saved to the same folder as the original object.</td>
<td></td>
</tr>
<tr>
<td>USER_NAMES</td>
<td>Comma-separated Financial Reporting user names who are granted access to the snapshot output.</td>
<td></td>
</tr>
<tr>
<td>GROUP_NAMES</td>
<td>Comma-separated Financial Reporting group names which are granted access to the snapshot output. A special system-defined group, called Everyone, includes all Financial Reporting users and can be used to ensure that all users have access to a snapshot output.</td>
<td></td>
</tr>
<tr>
<td>Printed Output</td>
<td>PRINT NAME</td>
<td>The printer name, if the PRINT VALUE attribute is set to “Yes”.</td>
</tr>
<tr>
<td>PRINT VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate printed output for the batch.</td>
<td></td>
</tr>
</tbody>
</table>

Note: In the USER_POV section of the XML file, HIDDEN="0" indicates a dimension which is on the POV and therefore is a candidate or value to be set in the XML file. The value to be changed is “_” in this example.
Batch Input File XML Tag Reference

Subtopics

- BATCH_JOB_OBJECT - Node Tag
- RUN_OPTIONS - Child Node Tag
- NOTIFICATION / EMAIL - Child Node Tag
- JOB_STATUS - Child Node Tag
- JOB_OBJECT - Child Node Tag
- DATA_SOURCE_USER_CREDENTIALS - Child Node Tag
- HR_USER_CREDENTIALS - Child Node Tag
- OUTPUT_OPTIONS - Child Node Tag
- CHILD NODE - HTML
- CHILD NODE - PDF
- CHILD NODE - SAVE_AS_SNAPSHOT
- CHILD NODE - PRINT
- USER_POV - Child Node

The following topics provide a complete listing of tags and values for the associated attributes. The structure of an XML file is similar to a tree level or directory structure. There is basically one parent-level node tag, and the tags that follow are child node tags.

**BATCH_JOB_OBJECT - Node Tag**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHOR</td>
<td>Displays in the batch scheduler's User ID column and is a useful place to show a comment or the name of the XML file that generated the batch.</td>
</tr>
<tr>
<td>BATCH_JOB_ID</td>
<td>A random number assigned to the batch</td>
</tr>
<tr>
<td>BATCH_NAME</td>
<td>The name of batch. For example, &quot;ReportStore:\SchdApi\Batches\TestBatch3&quot;.</td>
</tr>
<tr>
<td>REPORT_SERVER_NAME</td>
<td>The name of the report server where the batch is located</td>
</tr>
<tr>
<td>UNSAVED BATCH</td>
<td>The value of this attribute must be set to &quot;No&quot;.</td>
</tr>
</tbody>
</table>

**RUN_OPTIONS - Child Node Tag**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FREQUENCY</td>
<td>The value of this attribute should be “1”.</td>
</tr>
<tr>
<td>RUN_IMMEDIATELY</td>
<td>The value of this attribute should be &quot;Yes&quot;.</td>
</tr>
</tbody>
</table>

270
NOTIFICATION / EMAIL - Child Node Tag

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATTACH_RESULTS</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to attach PDF or HTML files generated to the email.</td>
</tr>
<tr>
<td>RECIPIENTS</td>
<td>A comma-separated list of recipients email addresses.</td>
</tr>
<tr>
<td>SENDER</td>
<td>The sender’s email address</td>
</tr>
<tr>
<td>SUBJECT</td>
<td>The subject of the email.</td>
</tr>
</tbody>
</table>

JOB_STATUS - Child Node Tag

This must be copied as it is shown in the following example, JOB_STATUS
CURRENT_STATUS="Pending"

JOB_OBJECT - Child Node Tag

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJECT_ID</td>
<td>Leave this attribute blank.</td>
</tr>
</tbody>
</table>

DATA_SOURCE_USER_CREDENTIALS - Child Node Tag

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS_PASSWD</td>
<td>The encrypted data source password from an existing batch or that you generate using the command line utility.</td>
</tr>
<tr>
<td>DS_USER_NAME</td>
<td>The data source user whose credentials are used for running the reports/books in the batch.</td>
</tr>
</tbody>
</table>

HR_USER_CREDENTIALS - Child Node Tag

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HR_PASSWD</td>
<td>The encrypted Financial Reporting user password from an existing batch or that you generate using the command line utility.</td>
</tr>
<tr>
<td>HR_USER_NAME</td>
<td>The Financial Reporting user whose credentials are used for running the reports/books in the batch.</td>
</tr>
</tbody>
</table>

OUTPUT_OPTIONS - Child Node Tag

This XML tag enables you to select the format of the batch output.
**CHILD NODE - HTML**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTML VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate HTML output for the batch.</td>
</tr>
</tbody>
</table>

**CHILD NODE - PDF**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDF VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate PDF output for the batch.</td>
</tr>
</tbody>
</table>

**CHILD NODE - SAVE_AS_SNAPSHOT**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAVE_AS_SNAPSHOT VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to save the snapshot output in the repository.</td>
</tr>
<tr>
<td>SAVE_NAME</td>
<td>The Folder Name where the Snapshots are to be stored. This must be specified in ReportStore:\ \ format. If SAVE_NAME = “”, the snapshot output is saved to the same folder as the original object.</td>
</tr>
<tr>
<td>USER_NAMES</td>
<td>Comma-separated Financial Reporting user names who are granted access to the snapshot output.</td>
</tr>
<tr>
<td>GROUP_NAMES</td>
<td>Comma-separated Financial Reporting group names which are granted access to the snapshot output. A special system-defined group, called Everyone, includes all Financial Reporting users and can be used to ensure that all users have access to a snapshot output.</td>
</tr>
<tr>
<td>SUBJECT_TOKENS</td>
<td>This attribute can be left blank or removed from the text file.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>This attribute is ignored if USER_NAMES or GROUP_NAMES is used.</td>
</tr>
</tbody>
</table>

**CHILD NODE - PRINT**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRINT NAME</td>
<td>The printer name, if the PRINT VALUE attribute is set to “Yes”.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>You must make sure that this printer is available to the scheduler server.</td>
</tr>
<tr>
<td>PRINT VALUE</td>
<td>Enter a “Yes” or “No” value, depending on whether you want to generate printed output for the batch.</td>
</tr>
</tbody>
</table>

**USER_POV - Child Node**

This node is optional. If the User POV is not specified here, the USER POV of the data source user specified in the text file is used instead.
Caution! This should be modified only by Global Administrators. Specifying a partial USER POV does not work.

Note: In the USER_POV section of the XML file, HIDDEN="0' indicates a dimension which is on the POV and therefore is a candidate or value to be set in the XML file. The value to be changed is “_” in this example.

Increasing the ADM Maximum Property Value Length

In rare cases where Financial Reporting retrieves member information from the data source, the returned information may be too large to retrieve. This may cause error messages and/or a Financial Reporting server to shut down. Example(s) of large retrievals:

- Financial Management Cell Text
- Essbaseoutline member formulas

The error message that is received is the following, where “xxxx” is the size in kilobytes of the information that is being retrieved.

“Property value is too long “xxxx” > 4,096, please consider increasing MAX_PROPERTY_VALUE_LENGTH in ADM.Properties” error message.

To remedy this situation, the ADM.properties file needs to be adjusted to accommodate the larger information retrievals.

To adjust the ADM.properties file

1. Open the ADM.properties file in a text editor. The file is typically located in:
   `<HYPERION HOME>\common\ADM\<BI+ VERSION>\lib\ADM.properties`
2. Locate the line MAX_PROPERTY_VALUE_LENGTH.
3. Increase the default setting of 4 (as in 4K). For example, if the error message is as follows: “Property value is too long 9503 > 4096, increasing the value to 10 resolved the issue.

   The default value of 4K and the above error message indicating the property size of 9503 is greater than the default value of 4096, by increasing to 10K, the new property value size is large enough to accommodate the formula.

Microsoft Word Template

Microsoft Word template provides the ability for Text (.txt) files to be printed. When selecting Print, the existing routine that prints Word documents is used to print the text file. After the print job has finished, the file is closed without saving, preserving the template for the next text file. Text file printouts will have a consistent look and feel when shown in PDF preview.
The Word template, FR_TextTemplate.doc, is located in the Financial Reporting print server in the ${home.reports}/bin/ folder. The template contains some pre-formatted content which can be customized.

Template Settings

By default, the Word template has the following settings:

- The "<<FR_content>>" string is the only content in the body of the document.
- The text is formatted to Courier New, 10pt font.
- Print option is Portrait with 1 inch margins.
- The footer is set to “print date and time” and current page number. These settings are updated and shown in all text files printed through the Financial Reporting print server.

Elements of the template that you can customize:

- You can customize the default formatting such as font, tabs, margins.
- You can control the page setup of the template. For example, always print in landscape mode or set the paper size to A4.
- You can customize the headers and footers. For example, show the page numbers in the footer, which is automatically updated when generating for PDF preview in books and reports.
- You can customize the preexisting content of the template. For example, insert a standard confidentiality clause at the end of every printout (see Customizing the Template).
- You can add images or backgrounds to the template. For example, a company logo.

Customizing the Template

To protect against the Word document becoming corrupt, make a backup before customizing the template. The template can be opened through Microsoft Word. Any saved changes are applied the next time a text file is printed by the Financial Reporting print server. When a text file is selected to print, the Financial Reporting print server opens the FR_TextTemplate.doc file and looks for the string, <<FR_content>>. If found, the context of the text file is inserted in that location, replacing the string. If the string is not found, the contents of the text file is inserted at the end of the Word document. If there is preexisting content in the template, it is shown before the data in the text file.

We recommend that you always include the <<FR_content>> string in the template. This allows you to control the font and paragraph settings of the content in the inserted text file. The content inherits the formatting that is currently in the <<FR_content>> string.

**Note:** You do not need to restart the Financial Reporting print server when a change is made.

To Customize the template:

1. Open Microsoft Word.
2 Select File, and then Open and navigate to the FR_TextTemplate.doc document located in the (default) ${home.reports}/bin/ folder, and click Open.

3 To change formatting, highlight <<FR_content>> and make your formatting changes.

4 Click Save.

Note: If you save the template to a different location than the default location, you must indicate the new location for TextPrintTemplate in the JConsole.exe file.

For information on JConsole, see the “Property Information” topic in the Oracle Hyperion Financial Reporting Administrator’s Guide.

**Showing Cell Text, Planning Unit Annotations and Financial Reporting Documents in Annotations**

Note: Planning Unit Annotations is currently not used with Oracle Fusion General Ledger.

A designer can globally control whether a report will automatically query for, and return Cell Text, Planning Unit Annotations and/or document attachments as Annotations in a given Financial Reporting report. Based on the .properties settings, the Report Property Sheet displays a set of controls than can turn this functionality on or off on a report by report basis. The global settings in fr_global.properties are:

- If AllowCellTextAsAnnotation=true, the Cell Text check box on the Report Property sheet is shown (for Financial Management and Planning, not Oracle Essbase yet). The default setting will be unchecked (turned off).
- If AllowPlanningUnitAnnotationsAsAnnotations=true, the Planning Unit Annotations check box on the Report Property sheet is shown for Planning. The default setting will be unchecked (turned off).
- If the AllowDocAttachmentsAsAnnotations=true, the Document Attachments check box on the Report Property Sheet is shown for Financial Management. The default setting will be unchecked (turned off).

**Updating Fonts in Financial Reporting in Linux/UNIX Server Environments**

When deploying Financial Reporting to a Linux or a UNIX server, some of the Microsoft system fonts that are available in Financial Reporting may not be available when printing, since these fonts are not included in Linux or UNIX. For example, the font “Microsoft Sans Serif” in bold or italics will initially display as a regular font without bold or italics applied.
In order to deploy these fonts on a Financial Reporting server on Linux or UNIX, the fonts must be downloaded and installed.

**Note:** Financial Reporting only supports true-type fonts. These fonts can be either licensed from Microsoft, or you can download a free alternative from the web.

After the fonts have been installed on the server, you must update the `xdo.cfg` file to include the new fonts. The `xdo.cfg` file maps all the fonts in the report to font files on the server. This file also contains other print server settings.

By default, the `xdo.cfg` file is located in:

```
EPM_ORACLE_INSTANCE/FinancialReporting/lib/FOProcessor
```

To update the `xdo.cfg` file, run the `FRPrintServerUtility` on the Financial Reporting server. The `FRPrintServerUtility` scans the system fonts and revises the `xdo.cfg` file to recognize the new fonts.

By default, the `FRPrintServerUtility` is located in:

```
EPM_ORACLE_HOME/products/financialreporting/bin
```
Why does changing dimensions in the User POV have no affect on the grid?

Multiple grids can be included in a report, where one or more of the grids is affected by changes to the dimensions in the user POV.

In Figure 5, the two grids look identical, but have different values for Measures and Year.

If you go back to the Report client and turn on the grid POV, you see that the members for Product, Market, and Scenario have been set for Grid 2. Thus, changing these dimensions in the Report Designer has no effect on Grid 2.

It is useful for the designer of the report to indicate the use of page dimensions in a text box or in row/column headers.
Why are the Grid POVs not displayed in the Report Designer?

If you disable the Grid Point of View, it is not displayed in the Report Designer.

How will a user running Workspace know that the dimension is set on the Grid POV?

The Report designer should design the grid with a text row footnote that the grid POV for a dimension is set to a particular member in the grid. Another option for the Report designer is to put the dimension value into the row or column headers (see the following figure for an example.) If the designer turned off the grid POV, the column description, using a custom heading would correctly denote that the grid is displaying Actual for the Scenario dimension.
How can you design a report so that changes to the User POV changes the members selected from a row, column, or page axis on a grid?

Use the CurrentPOV member for the selected axis or choose a member selection function with CurrentPOV as the member parameter. When the report runs, the axis value is taken from the user POV. You can design a grid so that all axis values are taken from the user POV, making the report completely dynamic on a per user basis.

Why do you need a Book POV?

It allows anyone running the book to use the same default values for all grids in all reports in the book simultaneously each time the book is run. It is specific to a database connection for a book, but not specific to any report or grid object in a report in that book. As the book POV values are saved with the book itself, changes to the book POV do not affect the book designer's user POV.

What are Report Member Selections?

The Report Member Selections define the values for any dimensions not on a grid's row, column, or page axis or set in the grid POV for all Grids in a specific report in the book.

Why not call Report Member Selections the Report POV?

All POVs allow a single member to be set for a dimension. Report Member Selections allow multiple members to be set for a dimension. If multiple members or member selection functions resulting in multiple members are set for a dimension, the report in the book is run for each member. If multiple members or member selection functions resulting in multiple members are set for multiple dimensions, the report in the book is run for all combinations of those members. Just like a grid is generated for each combination of the members on the page axis in a report, a report is generated for each combination of the members set using Report Member Selections in a book.

Why do you need a Batch POV?

It allows anyone scheduling the batch to use the same default values for all grids in all reports in the batch simultaneously when the batch is run. It is specific to a database connection for a batch, but not specific to any report or grid object in a report in that batch. Since the batch POV values are only saved when the batch is scheduled, you can set different values for the same batch.
Changes to the batch POV do not affect the batch designer's user POV. Also, the changes to the batch designer's user POV do not affect the batch POV.
This appendix lists and describes the currency symbols that come prepopulated with Oracle Hyperion Financial Reporting.

**Note**: Finland, France, Germany, Greece, Italy, Luxembourg, the Netherlands, Spain, Belgium, Austria, Cyprus, Slovenia, Malta, Ireland and Portugal use the European euro currency.

### Table 36  Currency Table

<table>
<thead>
<tr>
<th>Country</th>
<th>Currency Name</th>
<th>Currency Symbol</th>
<th>ISO Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Argentina</td>
<td>peso</td>
<td>$</td>
<td>ARS</td>
</tr>
<tr>
<td>Australia</td>
<td>dollar</td>
<td>A$</td>
<td>AUD</td>
</tr>
<tr>
<td>Austria</td>
<td>schilling</td>
<td>S</td>
<td>ATS</td>
</tr>
<tr>
<td>Belgium</td>
<td>franc</td>
<td>BEF</td>
<td>BEF</td>
</tr>
<tr>
<td>Brazil</td>
<td>real</td>
<td>R$</td>
<td>BRL</td>
</tr>
<tr>
<td>Canada</td>
<td>dollar</td>
<td>Can$</td>
<td>CAD</td>
</tr>
<tr>
<td>Chile</td>
<td>peso</td>
<td>Ch$</td>
<td>CLP</td>
</tr>
<tr>
<td>China</td>
<td>yuan</td>
<td>Y</td>
<td>CNY</td>
</tr>
<tr>
<td>Denmark</td>
<td>krone</td>
<td>Dkr</td>
<td>DKK</td>
</tr>
<tr>
<td>Egypt</td>
<td>pound</td>
<td>£E</td>
<td>EGP</td>
</tr>
<tr>
<td>European Union</td>
<td>European Currency Unit</td>
<td>€</td>
<td>XEU</td>
</tr>
<tr>
<td>Finland</td>
<td>markka</td>
<td>mk</td>
<td>FIM</td>
</tr>
<tr>
<td>France</td>
<td>franc</td>
<td>F</td>
<td>FRF</td>
</tr>
<tr>
<td>Germany</td>
<td>deutsche mark</td>
<td>DM</td>
<td>DEM</td>
</tr>
<tr>
<td>Greece</td>
<td>drachma</td>
<td>Dr</td>
<td>GRD</td>
</tr>
<tr>
<td>Hong Kong</td>
<td>dollar</td>
<td>HK$</td>
<td>HKD</td>
</tr>
<tr>
<td>India</td>
<td>rupee</td>
<td>Rs</td>
<td>INR</td>
</tr>
<tr>
<td>Country</td>
<td>Currency Name</td>
<td>Currency Symbol</td>
<td>ISO Code</td>
</tr>
<tr>
<td>-------------------------</td>
<td>----------------</td>
<td>-----------------</td>
<td>----------</td>
</tr>
<tr>
<td>Indonesia</td>
<td>rupiah</td>
<td>Rp</td>
<td>IDR</td>
</tr>
<tr>
<td>Ireland</td>
<td>punt</td>
<td>IR£</td>
<td>IEP</td>
</tr>
<tr>
<td>Israel</td>
<td>new shekel</td>
<td>NIS</td>
<td>ILS</td>
</tr>
<tr>
<td>Italy</td>
<td>lira</td>
<td>Lit</td>
<td>ITL</td>
</tr>
<tr>
<td>Japan</td>
<td>yen</td>
<td>¥</td>
<td>JPY</td>
</tr>
<tr>
<td>Korea (South)</td>
<td>won</td>
<td>W</td>
<td>KRW</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>franc</td>
<td>LuxF</td>
<td>LUF</td>
</tr>
<tr>
<td>Malaysia</td>
<td>ringgit</td>
<td>RM</td>
<td>MYR</td>
</tr>
<tr>
<td>Mexico</td>
<td>peso</td>
<td>Mex$</td>
<td>MXP</td>
</tr>
<tr>
<td>Netherlands</td>
<td>guilder</td>
<td>fl</td>
<td>NLG</td>
</tr>
<tr>
<td>New Zealand</td>
<td>dollar</td>
<td>NZ$</td>
<td>NZD</td>
</tr>
<tr>
<td>Norway</td>
<td>krone</td>
<td>NKr</td>
<td>NOK</td>
</tr>
<tr>
<td>Philippines</td>
<td>peso</td>
<td>Php</td>
<td>PHP</td>
</tr>
<tr>
<td>Poland</td>
<td>zloty</td>
<td>Zl</td>
<td>PLN</td>
</tr>
<tr>
<td>Portugal</td>
<td>escudo</td>
<td>Esc</td>
<td>PTE</td>
</tr>
<tr>
<td>Russia</td>
<td>ruble</td>
<td>R</td>
<td>RUR</td>
</tr>
<tr>
<td>Singapore</td>
<td>dollar</td>
<td>S$</td>
<td>SGD</td>
</tr>
<tr>
<td>South Africa</td>
<td>rand</td>
<td>R</td>
<td>ZAR</td>
</tr>
<tr>
<td>Spain</td>
<td>peseta</td>
<td>Ptas</td>
<td>ESP</td>
</tr>
<tr>
<td>Sweden</td>
<td>krona</td>
<td>SEK</td>
<td>SEK</td>
</tr>
<tr>
<td>Switzerland</td>
<td>franc</td>
<td>SwF</td>
<td>CHF</td>
</tr>
<tr>
<td>Taiwan</td>
<td>new dollar</td>
<td>NT$</td>
<td>TWD</td>
</tr>
<tr>
<td>Thailand</td>
<td>baht</td>
<td>Bht</td>
<td>THB</td>
</tr>
<tr>
<td>Turkey</td>
<td>lira</td>
<td>TL</td>
<td>TRL</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>pound</td>
<td>£</td>
<td>GBP</td>
</tr>
<tr>
<td>United States of America</td>
<td>dollar</td>
<td>$</td>
<td>USD</td>
</tr>
</tbody>
</table>
Using Oracle User Productivity Kit

If the Oracle User Productivity Kit (UPK) is deployed and Oracle Hyperion Enterprise Performance Management Workspace is configured by an Administrator with a valid URL for the UPK Player package, users can access UPK content for Oracle Enterprise Performance Management System. For more information on configuring UPK, see the “Workspace Server Settings” section in the Oracle Hyperion Enterprise Performance Management Workspace Administrator’s Guide and the “Oracle User Productivity Kit” section in the Application Support Guide.

Note: There are pre built UPK content modules available. See the data sheets that include UPK for Oracle Enterprise Performance Management System available on Oracle.com, http://www.oracle.com/us/products/applications/tutor-upk/064788.html. Financial Management and Planning modules include appropriate content for Oracle Smart View for Office and Reporting Studio. Oracle Hyperion Financial Management and Oracle Hyperion Planning support invoking UPK content in a context sensitive manner. UPK content launched from Smart View or Reporting Studio launches the full player package outline unfiltered for context. Reporting Studio and Smart View users can utilize a roles filter to see only the Oracle Smart View for Office or Oracle Hyperion Financial Reporting Studio content.

To open UPK Help:

1 Take one action:
   - Select the Help menu, and then select Oracle User Productivity Kit.
   - From the Help tool bar, click UPK.
   - From a dialog box, click Help, then from the Help toolbar click UPK.

2 Optional: If you opened a dialog box, close the dialog box when done.