

Oracle® Fusion Middleware

Business Intelligence Mobile App Designer User's Guide

11g Release 1 (11.1.1)

E54309-05

September 2015

Explains how to use the Oracle Business Intelligence Mobile App Designer to create purposeful mobile apps to run on mobile phones and tablets.

Oracle Fusion Middleware Business Intelligence Mobile App Designer User's Guide, 11g Release 1 (11.1.1)
E54309-05

Copyright © 2013, 2015, Oracle and/or its affiliates. All rights reserved.

Primary Author: Leslie Grumbach Studdard

Contributor: Oracle Business Intelligence product management, development, and quality assurance teams.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	ix
Audience	ix
Documentation Accessibility	ix
Related Documents	ix
Conventions	x
What's New in This Guide	xi
Updates in Release 11.1.1.9	xi
Updates in Release 11.1.1.7.10-BIMAD-PS1	xiii
1 Introduction	
1.1 What Is Oracle BI Mobile App Designer?	1-1
1.2 What Data Can I Use?	1-1
1.3 How Do I Access the Oracle BI Mobile App Designer?	1-2
1.4 What is the Apps Library?	1-2
1.5 How Do I Access the Apps Library?	1-3
2 Creating Apps	
2.1 Launch the Create App Wizard	2-1
2.2 Choose Your Device Type	2-2
2.3 Choose Your Data Source	2-2
2.4 Save Your App	2-4
3 Designing Apps	
3.1 The Oracle BI Mobile App Designer Workspace	3-2
3.1.1 Setting Properties	3-3
3.2 Setting App Properties	3-3
3.2.1 Setting the App Color Theme	3-4
3.2.2 Adding a Cover Image	3-5
3.2.3 Limiting Data Rows to Return for Design and Preview	3-6
3.3 Adding and Editing Pages	3-6
3.3.1 Selecting a Page Template	3-7
3.3.2 Changing the Page Name	3-8
3.3.3 Changing Between Main Page and Subpage	3-9

3.3.4	Moving Pages.....	3-9
3.3.5	Deleting Pages.....	3-9
3.4	Editing the Cover Page.....	3-10
3.4.1	Updating the Cover Page Background Image.....	3-10
3.5	Inserting Components and Plugins to Your Pages.....	3-11
3.6	Adding Exploration Pages.....	3-12
3.6.1	Customizing Text Filters on the Exploration Page.....	3-16
3.6.1.1	Specify Default Values for Text Filters.....	3-16
3.6.1.2	Enable or Disable Search for Text Filters.....	3-17
3.6.1.3	Customize Selected Background Color and Font Weight.....	3-17
3.6.2	Setting the Operator and Values for Numeric Filters on the Exploration Page.....	3-17
3.7	Adding Tile Pages.....	3-18
3.7.1	Creating a Tile Page.....	3-20
3.7.2	Adding a Subpage to a Tile Page.....	3-22
3.7.3	Customizing Tiles.....	3-24
3.7.3.1	Specify Number of Columns.....	3-24
3.7.3.2	Apply a Background Color.....	3-24
3.7.3.3	Apply Filters.....	3-25
3.7.3.4	Sort.....	3-25
3.7.3.5	Add Stoplight Formatting.....	3-26
3.7.3.6	Resize Tile Margins.....	3-31
3.8	Inserting Images.....	3-32
3.9	Inserting Frames.....	3-34
3.9.1	Adding a Border or Background Color.....	3-35
3.9.2	Inserting Additional Rows and Columns.....	3-36
3.9.3	Joining and Unjoining Cells.....	3-37
3.10	Adding Navigation Pages.....	3-38
3.10.1	Navigation Page for Tablet.....	3-38
3.10.1.1	Adding a Search Filter.....	3-41
3.10.2	Navigation Page for Phone.....	3-42
3.10.2.1	Adding a Search Filter.....	3-48
3.10.3	Stoplight Formatting for Navigation and Accordion Pages.....	3-49
3.10.3.1	Apply Formatting Based on Static Values.....	3-50
3.10.3.2	Apply Formatting Based on Percent Achievement.....	3-51
3.10.3.3	Apply Formatting Based on Percent Growth.....	3-53
3.10.3.4	Customizing the Colors.....	3-55
3.11	Adding Accordion Pages.....	3-56
3.11.1	Adding Stoplight Formatting to the Accordion Page.....	3-58
3.12	Inserting Tables.....	3-59
3.12.1	Customizing Tables.....	3-60
3.13	Inserting Pivot Tables.....	3-60
3.13.1	Inserting a Pivot Table.....	3-61
3.14	Inserting Charts.....	3-62
3.14.1	Inserting a Chart.....	3-63
3.14.2	Changing the Formula Applied to a Chart Measure Field.....	3-64
3.14.3	Changing the Series Color in a Bar or Line Chart.....	3-65
3.14.4	Changing the Width and Style of the Lines in a Line Chart.....	3-66

3.14.5	Setting the Treatment for Null Values.....	3-67
3.15	Inserting Maps.....	3-68
3.15.1	Inserting a Map	3-68
3.15.2	Setting Map Properties.....	3-72
3.15.2.1	Setting the Initial Zoom Level and Map Center.....	3-72
3.15.2.2	Customizing the Base Map Display.....	3-73
3.15.2.3	Setting the Color Distribution of Circle or Area Markers.....	3-74
3.15.2.4	Setting the Size Distribution of Circle Markers.....	3-74
3.16	Inserting Lists	3-75
3.16.1	Inserting a List.....	3-75
3.16.2	Customizing Lists	3-77
3.16.2.1	Specify Default Values for the List.....	3-77
3.16.2.2	Change List Orientation	3-78
3.16.2.3	Customize Font and Background Styles	3-78
3.16.2.4	Specify List Sort Order.....	3-79
3.17	Inserting Data Fields.....	3-80
3.17.1	Insert Data Fields	3-80
3.17.2	Display Data Fields Side-by-Side	3-80
3.18	Inserting Text.....	3-82
3.18.1	Display Text Items Side by Side	3-82
3.19	Calculated Fields.....	3-84
3.19.1	Adding Calculated Fields.....	3-84
3.19.2	Adding Calculated Fields to App Components.....	3-86
3.19.3	Editing Calculated Fields.....	3-87
3.20	Global Filters.....	3-87
3.20.1	Adding Global Filters.....	3-87
3.20.2	Applying Global Filters	3-89
3.20.3	Editing Global Filters	3-90
3.21	Customizing Background Images	3-91
3.21.1	Sizing the Background Image	3-92
3.22	Adding Sidebars.....	3-93
3.22.1	Inserting a Sidebar	3-94
3.22.2	Adjusting the Label Field Width	3-96
3.22.3	Customizing the Fields in the Sidebar.....	3-96
3.22.3.1	Group By a Data Field	3-96
3.22.3.2	Apply a Formula	3-97
3.22.3.3	Customize Data Formatting.....	3-98
3.22.3.4	Customize Labels, Fonts, and Background	3-98

4 Previewing, Sharing, and Publishing Apps

4.1	Previewing Apps.....	4-1
4.1.1	Preview on Your Desktop	4-1
4.1.2	Preview on a Mobile Device.....	4-2
4.2	Sharing Apps	4-3
4.3	Embedding Apps	4-4
4.4	Publishing Apps to the Apps Library.....	4-5

5 Component Property Features

5.1	Working with Tables	5-1
5.1.1	Customizing Alternating Row Colors	5-1
5.1.2	Setting Table Options	5-2
5.1.2.1	Set the Number of Rows to Display	5-2
5.1.2.2	Define Filters for Data Displayed in the Table	5-3
5.1.2.3	Show or Hide the Total Row	5-3
5.1.3	Customizing Column Headers	5-3
5.1.4	Customizing Table Data Display	5-3
5.1.4.1	Applying Conditional Formats to a Table Column	5-5
5.1.4.2	Managing Formats	5-6
5.1.4.3	Setting Table Data Formatting Options	5-6
5.1.4.4	Formatting Numeric Data Columns	5-6
5.1.4.5	Formatting Date Type Data Columns	5-7
5.1.4.6	Custom Data Formatting	5-7
5.1.4.7	About the Sort Option	5-8
5.1.4.8	Removing a Sort Order	5-9
5.1.4.9	About Grouping	5-9
5.1.4.10	Apply Group Left	5-9
5.1.4.11	Applying Subtotals	5-10
5.1.5	Customizing Table Totals	5-11
5.1.5.1	Formatting Total Cell Data	5-12
5.1.5.2	Applying a Formula	5-12
5.1.6	Inserting Links to Table Column Data	5-13
5.2	Working with Pivot Tables	5-13
5.2.1	Customizing the Pivot Table	5-13
5.2.1.1	Applying Filters	5-13
5.2.1.2	Customizing the Display of Totals	5-13
5.2.2	Customizing Pivot Table Headers	5-14
5.2.3	Customizing Pivot Table Data	5-14
5.3	Adding Filters to Tiles, Charts, Tables, and Pivot Tables	5-14
5.4	Adding a Search Filter to Navigation and Accordion Lists	5-14
5.5	Adding Static or Dynamic Links	5-16
5.6	Features of Metric Fields	5-19
5.6.1	Apply Data Formatting	5-19
5.6.2	Apply Custom Data Formatting	5-20
5.6.3	Update the Formula	5-21
5.6.3.1	Predefined Formulas Available from the Menu	5-21
5.6.3.2	Applying Custom Formulas	5-21
5.6.4	Add Spotlight Formatting	5-21
5.7	Setting Borders	5-21
5.8	Setting Margins	5-23
5.9	Using the Color Picker	5-23
5.10	Custom Number and Date Formats	5-24
5.10.1	Supported Number Format Mask Definitions	5-24
5.10.2	Supported Date Format Mask Definitions	5-26

6 Configuring App Settings

6.1	Updating or Refreshing the Data Source.....	6-1
6.1.1	Add Multiple Subject Areas to a Data Source	6-1
6.1.2	Refresh Excel Data	6-2
6.1.3	Update a BI Publisher Data Model.....	6-3
6.2	Accessing App Settings.....	6-3
6.3	Configuring Parameters.....	6-4
6.3.1	Configure Parameters for a BI Subject Area Data Source	6-4
6.3.2	Configure Parameters for a BI Publisher Data Model Data Source	6-6
6.4	Setting App Caching Properties	6-7
6.4.1	Enable Data Caching	6-7
6.5	Adding Translations	6-8
6.5.1	What Is App Translation?.....	6-9
6.5.2	Downloading a Translation File	6-9
6.5.3	Translating the XLIFF File	6-10
6.5.4	Uploading the Translated File	6-10

7 Adding Custom Plugins

7.1	What Are BI Mobile App Designer Plug-in Extensions	7-1
7.2	Exploring the Sample Plugins	7-2
7.3	Setting Up Your Development Environment	7-2
7.3.1	Download and Install the Prerequisites	7-2
7.3.2	Download the SDK with Sample Data	7-3
7.3.3	Files Included in the SDK	7-4
7.4	Coding the Custom Plug-in.....	7-5
7.4.1	Plug-in Structure	7-5
7.4.2	Property Support	7-7
7.4.2.1	Predefined Properties	7-8
7.4.3	Fields Support	7-8
7.4.3.1	Defining the Data Fields	7-9
7.4.4	JavaScript APIs That Can Be Used in Custom Plug-ins.....	7-10
7.4.4.1	handleClickEvent Method.....	7-10
7.4.4.2	getPixelValue Method.....	7-10
7.5	Accessing Data at Run Time.....	7-11
7.5.1	Accessing Data Stored in rows Array	7-11
7.5.2	Accessing Data Through a JavaScript Object	7-11
7.6	Testing Your Plug-in	7-12
7.7	Preparing Your Plug-in for Upload.....	7-12
7.8	Uploading to the Plug-in Gallery	7-12
7.8.1	Accessing the Plug-in Gallery	7-13
7.8.2	Uploading Plug-ins.....	7-13
7.8.3	Managing Plug-ins	7-14

A Reference for Administrators

A.1	About the Apps Library	A-1
A.1.1	What Is the Apps Library?	A-1

A.1.2	What Happens When a User Subscribes to an App?	A-2
A.1.3	About Local and Remote Apps Libraries	A-3
A.2	Setting Up the Apps Library Locations	A-3
A.2.1	Setting Up a Local Apps Library	A-4
A.2.1.1	Create the Apps Library Folder in the Catalog	A-4
A.2.1.2	Set Up the Configuration File	A-4
A.2.2	Enabling Remote Connection to an Apps Library on Another Instance	A-4
A.3	Sharing Access to the Apps Library with App Consumers	A-5
A.4	Securing Apps in the Apps Library	A-6
A.5	Required Permissions to Use Oracle BI Mobile App Designer and Run Apps	A-6
A.6	Moving Apps Between Environments	A-7
A.7	Adding Custom Maps	A-9
A.8	Activating Metadata Startup Cache to Improve Performance	A-11
A.8.1	Configuring Metadata Cache on Startup	A-11
A.8.2	Clearing the Subject Area Metadata Cache	A-12
A.9	Configuring Single Sign-On	A-14
A.9.1	Setup Procedure for Oracle Access Manager	A-14
A.10	Enabling Oracle BI EE 10g-Style Initialization Block Security	A-16
A.11	Protecting Against Out of Memory Errors Using Memory Guard	A-17
A.11.1	What are the Memory Guard Settings?	A-17
A.11.1.1	Restricting Maximum Data Sizes for App Processing	A-17
A.11.1.2	Configuring Free Memory Threshold	A-18
A.11.2	Setting Memory Guard Properties	A-19

Preface

Oracle Business Intelligence Mobile App Designer is a tool for creating platform-independent mobile apps that can draw on data from a variety of data sources. BI Mobile App Designer empowers business users to create stunning and interactive analytical apps for any mobile device. Users can create purpose-built apps for any line of business that help effectively showcase insights and provide summary information at a glance, with detailed information only a touch or swipe away.

Audience

This document is intended for anyone who plans to create mobile apps using Oracle Business Intelligence Mobile App Designer.

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.

Related Documents

See the Oracle Business Intelligence documentation library for a list of related Oracle Business Intelligence documents.

In addition:

- See the Oracle BI Mobile App Designer page on the Oracle Technology Network.
- See the Oracle Learning Library for Oracle Business Intelligence-related training resources.
- See the Product Information Center support note (Article ID 1267009.1) on My Oracle Support at <https://support.oracle.com>.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in This Guide

This preface describes changes to Oracle BI Mobile App Designer User's Guide since the last release.

This preface contains the following topics:

- [Updates in Release 11.1.1.9](#)
- [Updates in Release 11.1.1.7.10-BIMAD-PS1](#)

Updates in Release 11.1.1.9

Updates in Oracle BI Mobile App Designer 11g Release 1 (11.1.1.9) include:

- [Exploration Page Updates](#)
- [Changes to the Map Component](#)
- [Sample Plugins](#)
- [Performance Enhancement for Subject Area-Based Apps](#)
- [Protect Against Out of Memory Failures with Memory Guard](#)

Exploration Page Updates

You can now use calculated fields as filters in your Exploration pages. After you create the calculated field, simply drag it to the Exploration page filter list as you do any other data field.

See [Adding Exploration Pages](#).

Changes to the Map Component

Several changes and new features for the map component are introduced in this release, including:

- [Support for "Current Location" as Map Center Point](#)
- [Initial View and Zoom Level Settings Updated](#)
- [Default Map Now Uses the AlbersUSA Projection](#)
- [Changes to the Map Configuration File](#)

Support for "Current Location" as Map Center Point

For maps based on latitude and longitude values, you can now enable **Use Current Location** as the **Initial View**. When enabled, the app detects the user's current location from the browser and uses it as the center point of the map. Users accessing the app

from different locations see the map centered appropriately for their current geographic position.

See [Use Current Location as Initial View](#).

Initial View and Zoom Level Settings Updated

When setting the Initial View properties of a map, Latitude and Longitude settings are no longer available to specify the initial center point of the map. Instead, when you pan and zoom in the map editor, the initial view is automatically saved. Also the number of zoom levels has been increased to 18 to give you a greater range of zoom options.

See [Setting the Initial Zoom Level and Map Center](#).

Default Map Now Uses the AlbersUSA Projection

The US maps included with Mobile App Designer now use the AlbersUSA projection. Previously the Mercator projection was used. If you are upgrading from an earlier version of Mobile App Designer you will notice some differences in the representations of the States.

Changes to the Map Configuration File

In this release the following attributes of the <geojson> element are deprecated:

Deprecated Attribute	Description	Now Use
default-zoom-level	The default zoom level of the map when the map displays in the app. If not set, defaults to 0 (entire world). This property can be reset within the app designer.	Set the default zoom level using the Initial View properties of the map in the app.
center-location	The default center point of the map when it first displays in the app. If not set, defaults to 0,0 (longitude and latitude). This property can be reset within the app designer.	Set the default center point by centering your map as desired in the designer and then enabling the Initial View property. This freezes the current setting as the start-up view of your map.
label	The name that displays in the selection list in the app designer. Example: "World Regions"	display-name "display-name" is required and provides the name that displays in the selection list in the app designer.

The following are new attributes of the <geojson> element added for this release.

New Attribute	Description
display-name	Replaces "label" attribute from the previous release. Required. The name that displays in the selection list in the app designer. Example: "World Regions"
outline-file	Optional. For US maps, you can provide an additional outline file to enhance the way borders are drawn in the map. In this release, an outline-file is provided for the USA Counties Map.
projection	Optional. For US maps only. Specify the AlbersUSA projection or the Mercator projection.

See [Section A.7, "Adding Custom Maps"](#) for the full description of the map configuration file.

More Maps Included

The Mobile App Designer now includes many more country and region maps. The full list is available at

<http://www.oracle.com/technetwork/middleware/bi-foundation/default-2397753.html>

Sample Plugins

The plugins tab of the Insert menu includes a set of sample plugins. Explore these samples to inspire you to design your own. See [Inserting Components and Plugins to Your Pages](#).

Performance Enhancement for Subject Area-Based Apps

Mobile App Designer now supports caching of the Subject Area business views. This feature reduces rendering time by enabling apps to access the Subject Area metadata from a cache rather than loading it afresh from the Oracle BI Server every time. This feature must be explicitly turned on. See [Activating Metadata Startup Cache to Improve Performance](#) for more information.

Protect Against Out of Memory Failures with Memory Guard

This release provides a set of properties to protect against out-of-memory errors by blocking app requests that generate excessive amounts of data. The properties set limits on data size and free memory availability. When an app exceeds the set limits, processing of the app terminates. You can configure the limits to suit your system needs. For more information see [Protecting Against Out of Memory Errors Using Memory Guard](#).

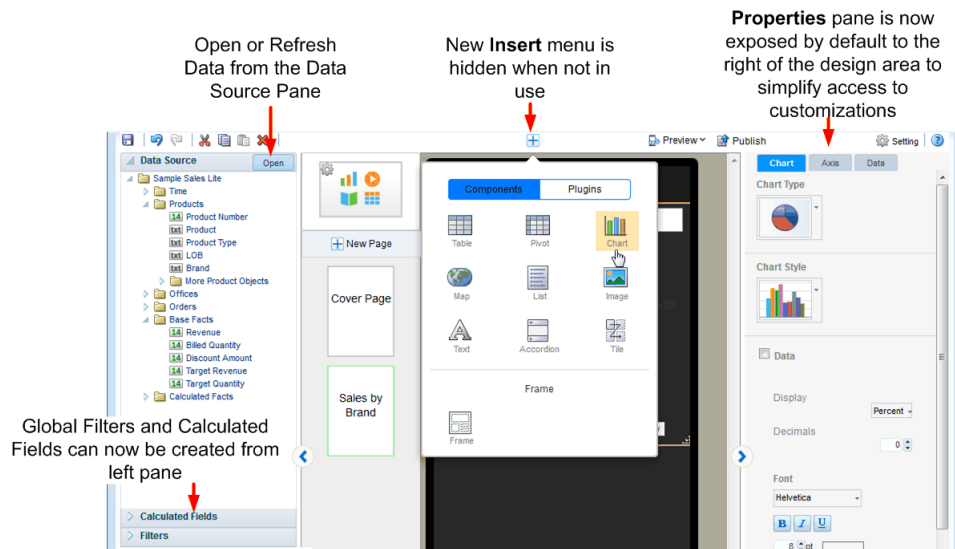
Updates in Release 11.1.1.7.10-BIMAD-PS1

Updates in Oracle BI Mobile App Designer Release 2 (11.1.1.7.10-BIMAD-PS1) include:

- [User Interface Redesign](#)
- [Map Component](#)
- [Exploration Page](#)
- [Global Filters](#)
- [Calculated Fields](#)
- [Filter Component Renamed "List"; Defaults and Multiselect Now Supported](#)
- [Software Development Kit \(SDK\) Available to Simplify Adding Plugins](#)
- [Plugin Gallery Simplifies Uploading and Managing Plugins](#)

User Interface Redesign

Several key functions of the user interface have been redesigned to enable easier interaction with and customization of app components.



The redesigned functions include:

- Ribbon toolbars no longer display at the top of the design area.
- The Insert component menu is now activated only when you need to insert a component.
- The options available from the contextual toolbars are now displayed on the right hand Properties pane.
- The Open or Refresh data source option is now located on the Data Source pane.
- Global Filters and Calculated Fields can now be created and managed from the left accordion pane.

Map Component

This version of BI Mobile App Designer includes a new Map component to enable you to create location-intelligent visualizations. For more information see [Section 3.15, "Inserting Maps."](#)

Exploration Page

A new page type enables you to provide your app consumers the most flexibility in choosing combinations of data to visualize. Exploration pages provide multiple filters to enable easy exploration of the data presented on the page. Any text or numeric data fields can be added as filters enabling your users to mix and match the fields that interest them to provide new and personal views of the data. For more information, see [Section 3.6, "Adding Exploration Pages."](#)

Global Filters

You can now create global filters that can be applied to components throughout your app. For more information, see [Section 3.20, "Global Filters."](#)

Calculated Fields

The Calculated Fields feature enables you to create calculated fields that do not exist in your data, and then use these fields throughout your app. See [Section 3.19, "Calculated Fields."](#)

Filter Component Renamed "List"; Defaults and Multiselect Now Supported

The first release of BI Mobile App Designer included a component called "Filter." In this release the Filter component has been renamed "List." The List component has been enhanced to enable you to specify default selections for the list when the app page containing the list is first opened. In addition, multiple selections are also now supported. See [Section 3.16, "Inserting Lists."](#)

Software Development Kit (SDK) Available to Simplify Adding Plugins

BI Mobile App Designer supports adding JavaScript plug-in extensions to the app editor to add custom components to your apps. In this release you can download a Software Development Kit (SDK) from the app designer to facilitate the coding and testing of plugins. See [Chapter 7, "Adding Custom Plugins."](#)

Plugin Gallery Simplifies Uploading and Managing Plugins

To simplify uploading and managing the custom plugins you create with the SDK, you can now use the Plugin Gallery. See [Section 7.8, "Uploading to the Plug-in Gallery."](#)

Introduction

Oracle Business Intelligence Mobile App Designer is a tool for designing purposeful apps for mobile phones and tablets. This chapter provides an overview of the BI Mobile App Designer. It includes the following sections:

- [What Is Oracle BI Mobile App Designer?](#)
- [What Data Can I Use?](#)
- [How Do I Access the Oracle BI Mobile App Designer?](#)
- [What is the Apps Library?](#)
- [How Do I Access the Apps Library?](#)

1.1 What Is Oracle BI Mobile App Designer?

Oracle BI Mobile App Designer is a powerful design-time studio and run-time engine for creating purposeful apps for any line of business. Create visually engaging apps to showcase insights and provide summary information at a glance, with detailed information only a touch or swipe away.

The apps you create with BI Mobile App Designer are platform and device independent. Apps run in HTML5 on any modern browser on your mobile device. No client install is required.

The responsive Web engine of the BI Mobile App Designer detects your device screen size and automatically adjusts the app display, so you can open the same app on any size device, including your PC.

The intuitive design studio user interface empowers business users to create stunning and interactive analytical apps.

1.2 What Data Can I Use?

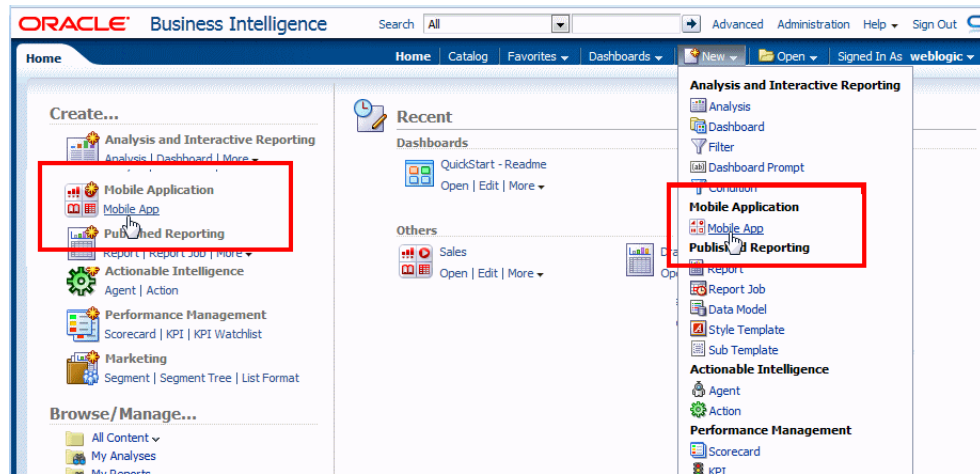
Because BI Mobile App Designer is integrated with Oracle Business Intelligence Enterprise Edition you can leverage the data models you have already created for your organization's data. Use any of the following data sources:

- BI Subject Area
- BI Publisher Data Model
- Excel spreadsheet

1.3 How Do I Access the Oracle BI Mobile App Designer?

After installation, the Oracle BI Mobile App Designer is available to users with the BI Author role. Begin designing your first app by selecting **Mobile App** from the Home page **Create** menu or from the **New** menu on the global header, as shown in Figure 1-1.

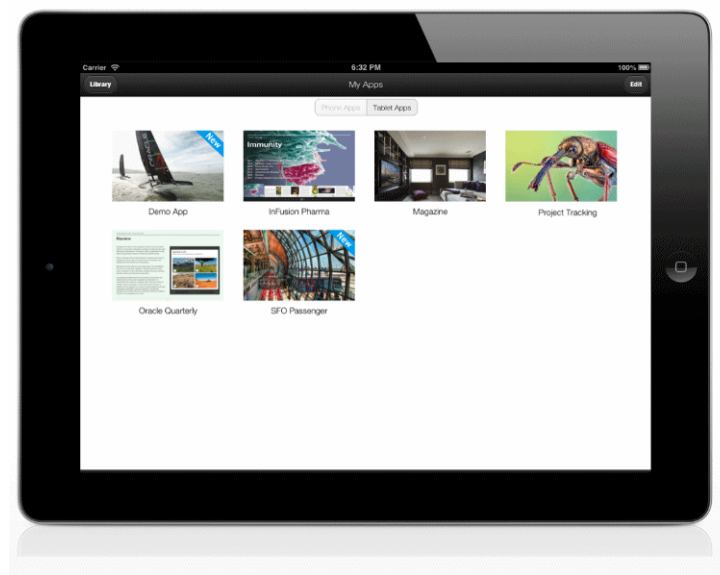
Figure 1-1 Accessing the Oracle BI Mobile App Designer



1.4 What is the Apps Library?

The Apps Library is where you manage your apps and subscribe to new apps. When you open the Apps Library from your mobile device you can choose to subscribe to the apps you have permission to access.

Figure 1-2 My Apps Library



When you subscribe to an app, it becomes available in your My Apps Library viewer. You can switch between your My Apps Library and the shared Apps Library to subscribe to more apps.

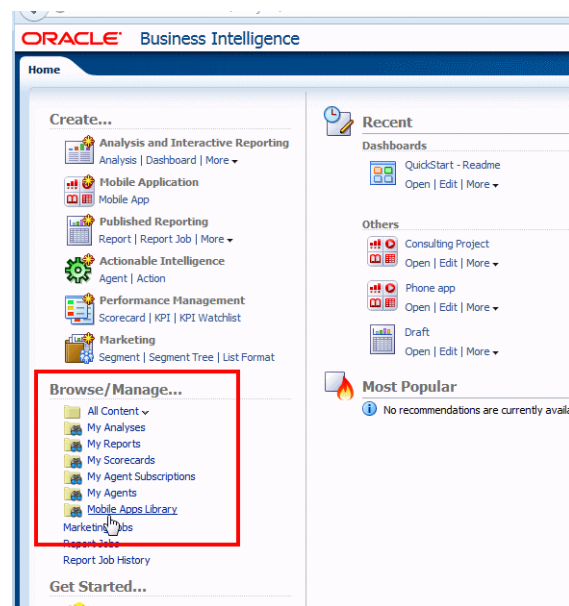
When you create a new app you have the option of publishing it to the library. Once published it is immediately available to other users with access to the library and permissions to view the app. Users subscribe to an app to add it to their personal My Apps library. When you update the app, users are notified that a new version is available.

The Apps Library must be set up by an administrator. See [Section A.2, "Setting Up the Apps Library Locations."](#)

1.5 How Do I Access the Apps Library?

The Apps Library URL is available from the Oracle Business Intelligence home page under the **Browse/Manage** region.

Figure 1–3 Apps Library Link on Oracle BI Home Page



Add this URL to your mobile device home screen for easy access to all your BI mobile apps.

Creating Apps

You can create an app in four easy steps:

- [Launch the Create App Wizard](#)
- [Choose Your Device Type](#)
- [Choose Your Data Source](#)
- [Save Your App](#)

More Learning Sources for Getting Started:

Video: [Starting and Exploring BI Mobile App Designer](#)

Tutorial: [Creating Mobile Apps with Oracle BI Mobile App Designer Version 2](#)

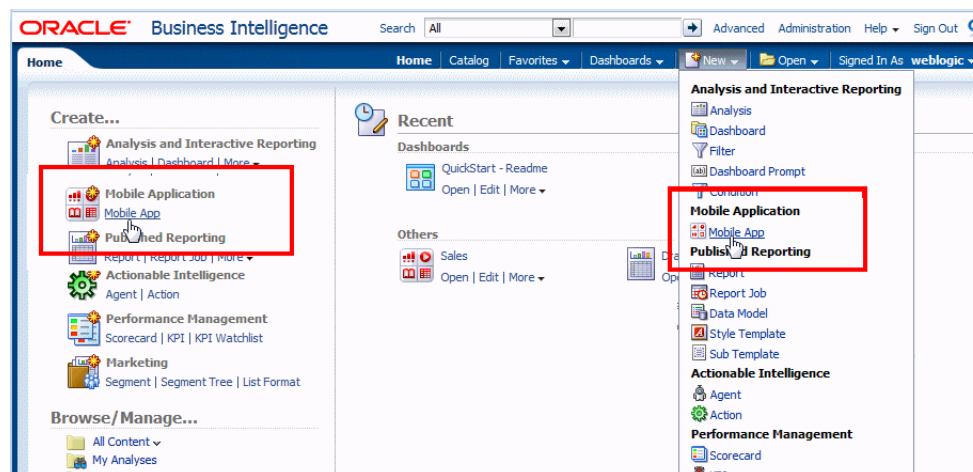
2.1 Launch the Create App Wizard

Launch the Create App Wizard in one of the following ways:

- From the **Home** page, under the **Create** region, select **Mobile App**.
- On the global header, click **New** and then select **Mobile App**.

Both methods are shown in [Figure 2–1](#).

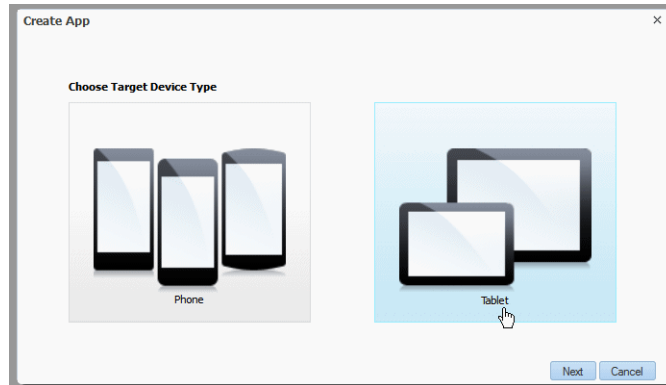
Figure 2–1 Launch the BI Mobile App Creation Wizard



2.2 Choose Your Device Type

In the App Creation wizard, choose Phone or Tablet as the target device type for your app.

Figure 2–2 *Choosing a Device Type*



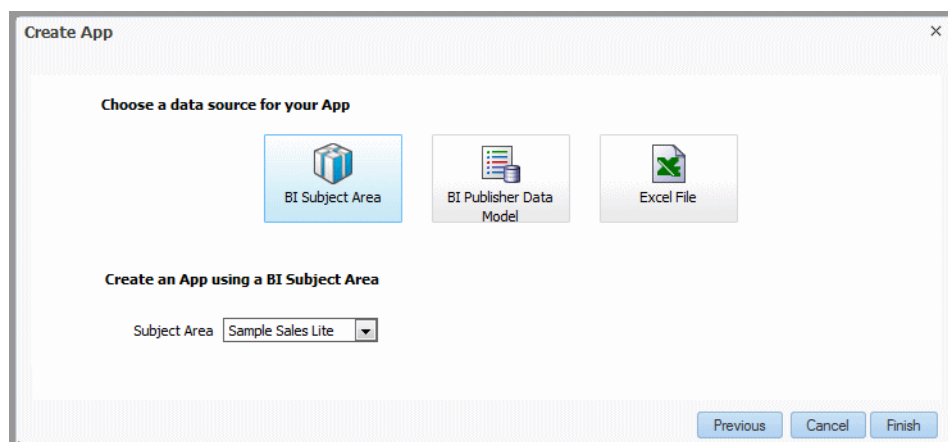
The Mobile App Designer presents a workspace optimized for the device type chosen here.

Tip: Although both types can be opened on any device, Oracle recommends that you consider carefully the use case for your app and design for one target device. If your app is for retrieving specific information quickly, a phone app is appropriate. If you intend for your users to spend more time interacting with your app, choose tablet.

2.3 Choose Your Data Source

Select the data source type to use for your app.

Figure 2–3 *Choosing a Data Source Type*



The Mobile App Designer supports the following data sources:

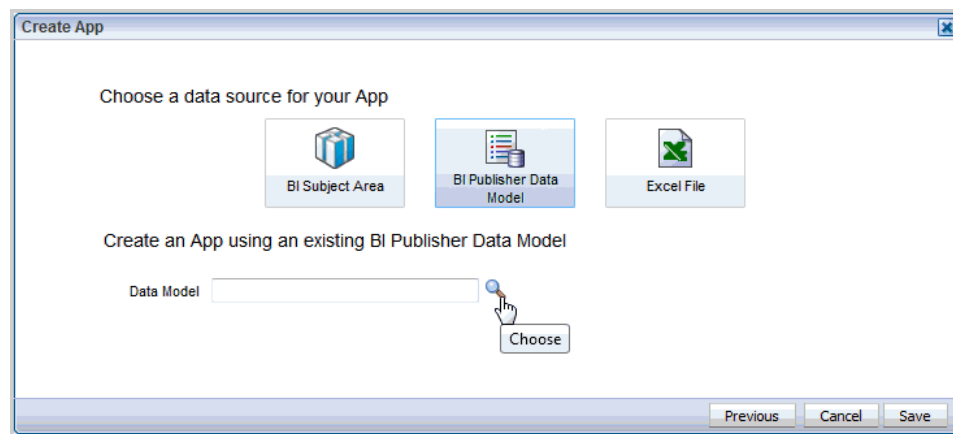
- BI Subject Area
Click **BI Subject Area** and select the Subject Area from the list.

In this step you can select one BI Subject Area. To use multiple subject areas in your data source see [Section 6.1.1, "Add Multiple Subject Areas to a Data Source."](#)

- **BI Publisher Data Model**

Click **BI Publisher Data Model** and choose the data model from the Catalog.

Figure 2–4 Choosing a BI Publisher Data Model as the data source

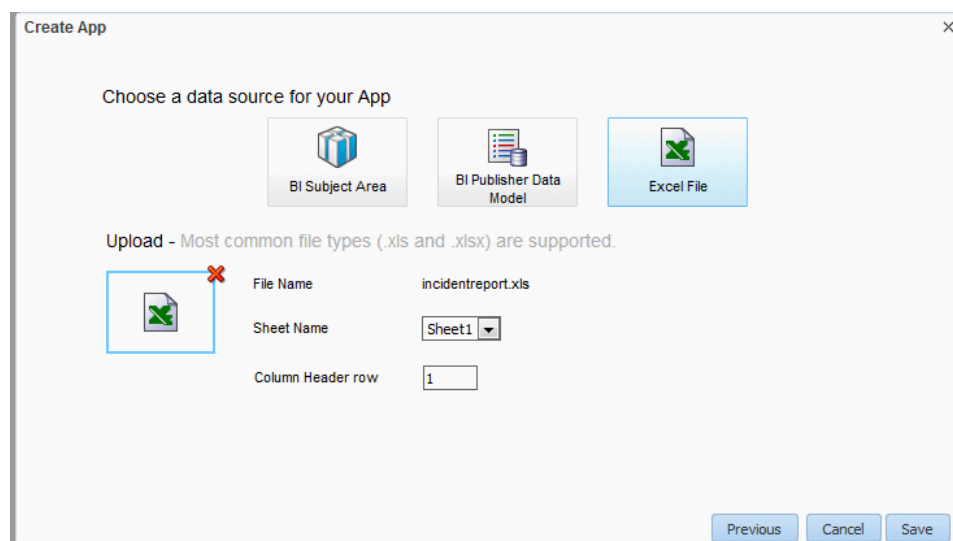


BI Publisher data models must include sample data. For information about BI Publisher data models, see *Oracle Fusion Middleware Data Modeling Guide for Oracle Business Intelligence Publisher*.

- **Excel spreadsheet**

Click **Excel File** and then either drag the file from your desktop to the wizard, or click Browse to select the file from your file system for upload. Mobile App Designer supports .xls and .xlsx file types. For additional details about Excel file support see "Creating a Data Set Using a Microsoft Excel File" in *Oracle Fusion Middleware Data Modeling Guide for Oracle Business Intelligence Publisher*.

Figure 2–5 Uploading an Excel file as the data source



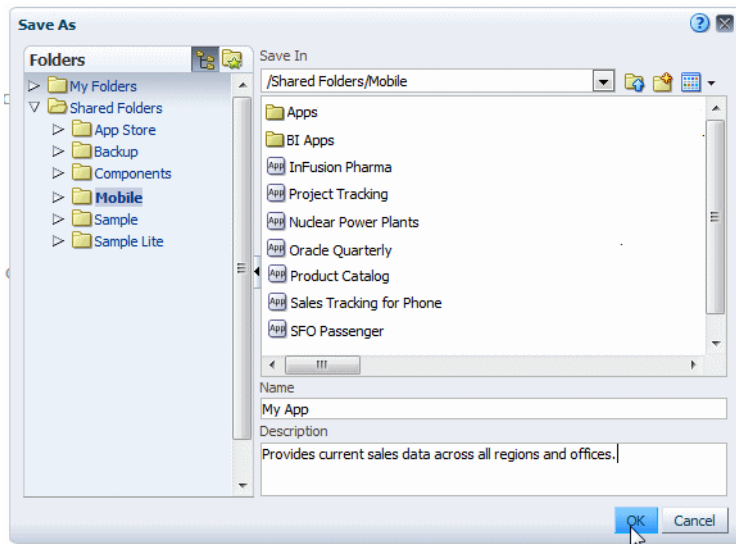
Specify the following for your Excel data source:

- **Sheet Name** - if the Excel Workbook includes multiple sheets, select the sheet to use. The data must be on a single sheet.
- **Column Header row** - specify the row that contains your column headers. Defaults to 1.

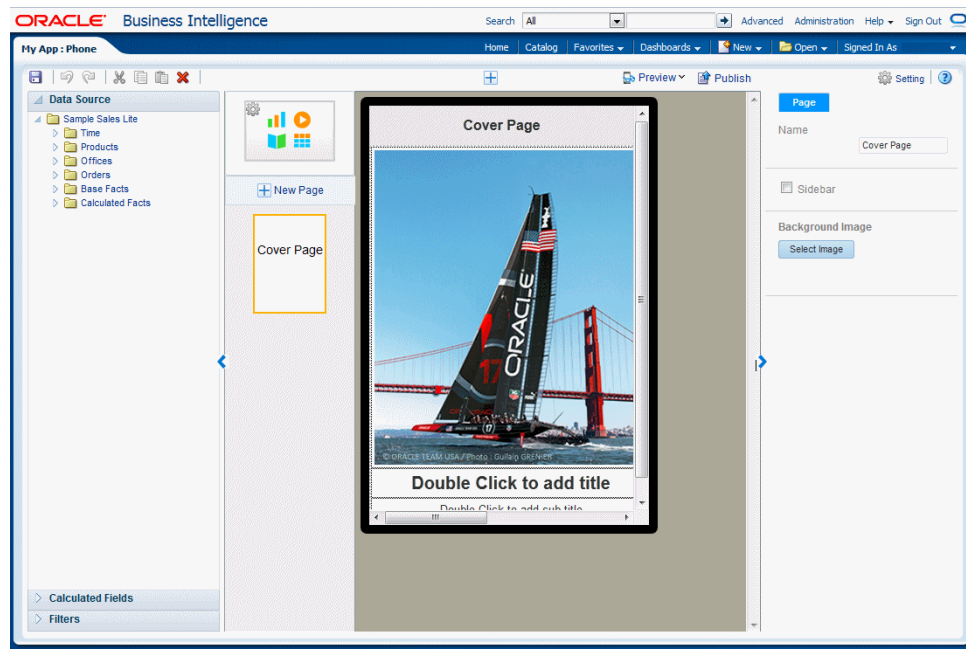
2.4 Save Your App

Click **Finish** to save your app. Select a folder location, then enter a name and click **OK**.

Figure 2–6 *Saving the App*



The app designer opens to display the default cover page. The design area displays as a tablet or phone as appropriate for your device type choice. See [Chapter 3, "Designing Apps"](#) to begin designing your app.

Figure 2–7 BI Mobile App Designer

Designing Apps

This chapter describes how to insert and customize the components available in BI Mobile App Designer.

- [The Oracle BI Mobile App Designer Workspace](#)
- [Setting App Properties](#)
- [Adding and Editing Pages](#)
- [Editing the Cover Page](#)
- [Inserting Components and Plugins to Your Pages](#)
- [Adding Exploration Pages](#)
- [Adding Tile Pages](#)
- [Inserting Images](#)
- [Inserting Frames](#)
- [Adding Navigation Pages](#)
- [Adding Accordion Pages](#)
- [Inserting Tables](#)
- [Inserting Pivot Tables](#)
- [Inserting Charts](#)
- [Inserting Maps](#)
- [Inserting Lists](#)
- [Inserting Data Fields](#)
- [Inserting Text](#)
- [Calculated Fields](#)
- [Global Filters](#)
- [Customizing Background Images](#)
- [Adding Sidebars](#)

More Learning Sources:

Video: [Starting and Exploring BI Mobile App Designer](#)

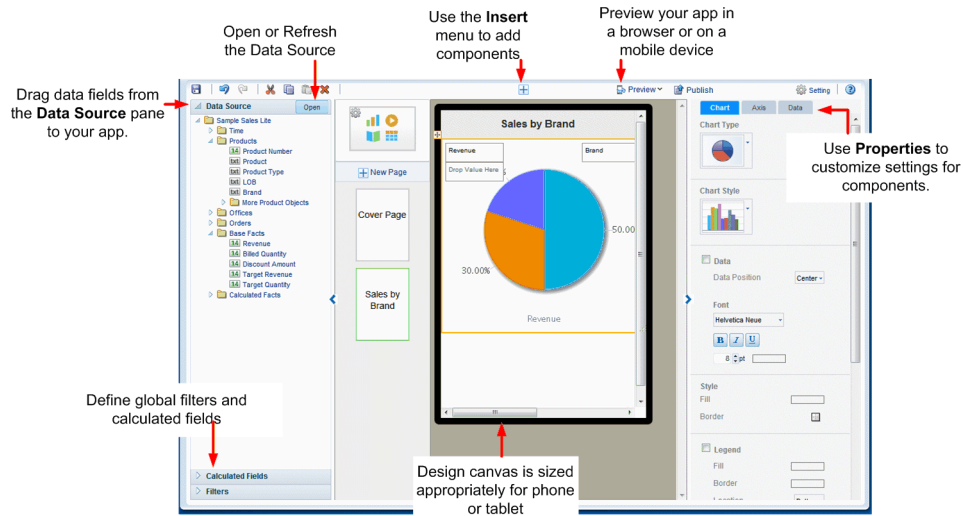
Tutorial: [Creating Mobile Apps with Oracle BI Mobile App Designer Version 2](#)

Tutorial: [Creating an Advanced Mobile App with Oracle BI Mobile App Designer](#)

3.1 The Oracle BI Mobile App Designer Workspace

Figure 3–1 shows the Oracle BI Mobile App Designer workspace.

Figure 3–1 Oracle BI Mobile App Designer



Note the following features of the app designer:

- The workspace displays the canvas area as a phone or a tablet-sized screen depending on the target device type you chose. The tablet area is 1024 x 768 pixels. The phone area is 320 x 480 pixels. This design area size does not constrain the run time display. Mobile App Designer's responsive display engine detects the device screen size and adjusts the viewing area appropriately for the device an app is being viewed with.
- Click **Insert** to add new components to your app.
- The right hand pane is the **Properties** palette to modify and fine-tune settings for the component selected in the design area. You can collapse the **Properties** pane using the handle on the left edge.
- The left hand pane is an accordion pane that includes the **Data Source** pane, the **Filters** pane and the **Calculated Fields** pane. You can collapse the pane using the handle on the right edge.
 - Use the **Data Source** pane to select and drag data fields to the visualizations you insert to your app.
 - Use **Filters** to define filters that you can apply to multiple components throughout your app. For more information, see [Global Filters](#).
 - Use **Calculated Fields** to construct calculated fields that you can reuse throughout your app. For more information, see [Calculated Fields](#).
- **Preview** your app at anytime during the design phase in the mobile device simulator, and even share the preview with others. See [Chapter 4, "Previewing, Sharing, and Publishing Apps."](#)
- When your app is complete, click **Publish** to publish your app to your site's Apps Library to make it available to users. See [Section 4.4, "Publishing Apps to the Apps Library"](#) for more information.

3.1.1 Setting Properties

The **Properties** pane or palette displays the properties for the component currently selected in the workspace. For example, when you select a table in the workspace, the Table properties display.

When not using the Properties palette you can hide it using the arrow handler on the left edge of the pane.

Components with many available property settings are organized on tabs within the Properties palette, for example, when you select a chart, the properties are organized by Row, Axis, and Data tabs.

The most commonly used settings are described in the sections for each component and in [Chapter 5, "Component Property Features."](#)

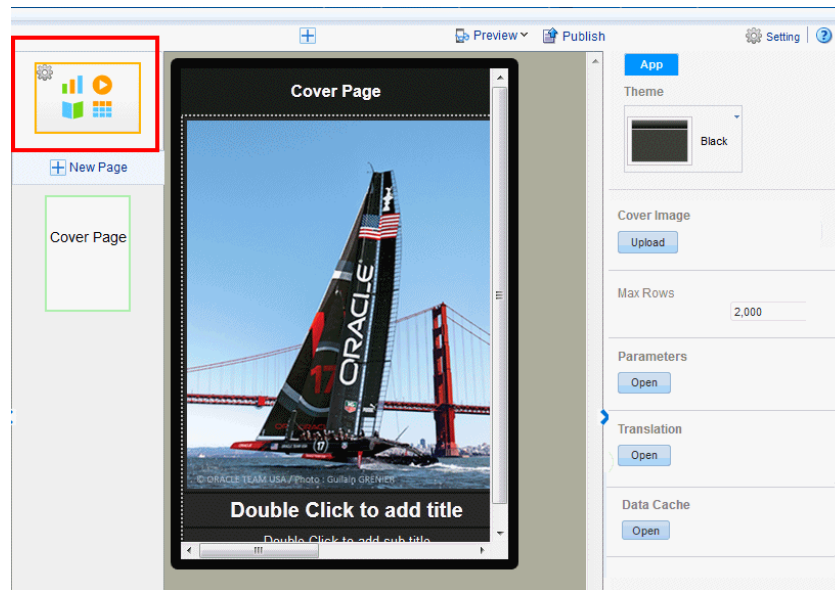
[Figure 3–2](#) shows properties available for a Chart.

Figure 3–2 Sample Properties Pane for a Chart



3.2 Setting App Properties

To set app-level properties, click the app icon at the top of the page list as shown in [Figure 3–3](#).

Figure 3–3 Select the App icon to display the App properties

You can set the following for your app:

- [Setting the App Color Theme](#)
- [Adding a Cover Image](#)
- [Limiting Data Rows to Return for Design and Preview](#)

For information about the Parameters, Translation, and Data Cache settings, see [Chapter 6, "Configuring App Settings."](#)

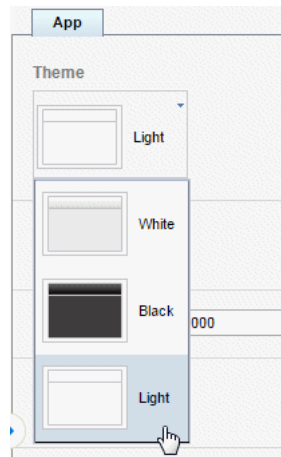
3.2.1 Setting the App Color Theme

You can set your app color theme to be black, white, or light. App components (charts, tables, filters, and so on) display complementary default coloring depending on your chosen theme.

Tip: Ensure to choose your app color theme before you begin inserting components. Components take on the color theme at the time you insert them. If you change the color theme after inserting components, you may need to reinsert them to reset the coloring appropriately.

To set the App theme:

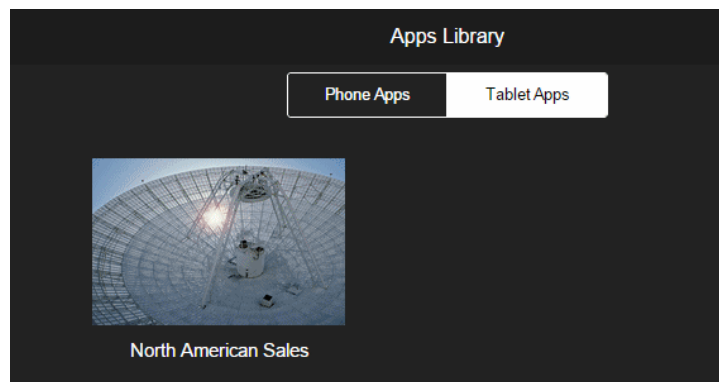
1. Select the **App** icon to display the app properties.
2. From the **Theme** menu, select Black, White, or Light. (Note that the Light theme currently only governs the display of the Navigation page.)



3.2.2 Adding a Cover Image

The cover image is the image that displays for your app in the apps library as shown in [Figure 3-4](#).

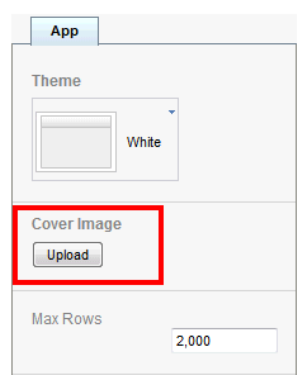
Figure 3-4 Cover Image Shown in Apps Library



To change the background image on the cover page of your app, see [Customizing Background Images](#).

To add the cover image to display in the Apps Library:

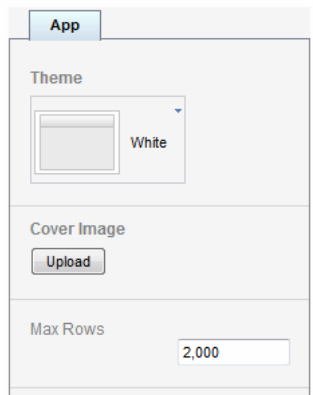
1. Select the **App** icon to display the app properties.
2. Under the **Cover Image** property, click **Upload**.



3. Select the image (jpeg, gif, png, or bmp) file and click **Upload**.

3.2.3 Limiting Data Rows to Return for Design and Preview

The Mobile App Designer populates the components of your app with data from your data source during design and preview. If your data source is large, you can enhance design-time interaction by limiting the number of rows that are used for design and preview. By default, the **Max rows** is set to 2000.

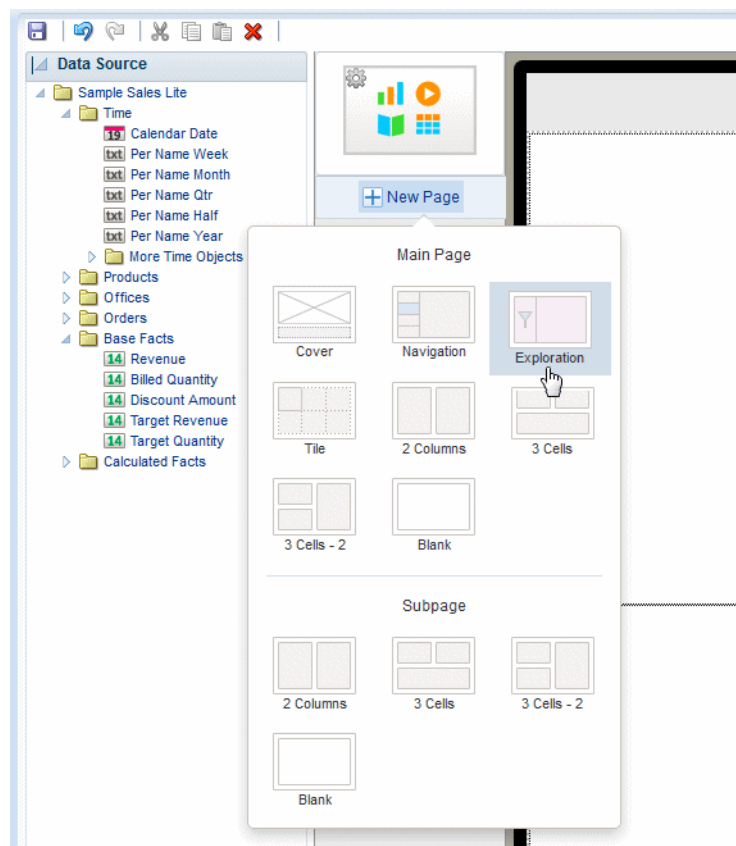


To edit this number, enter the number of rows to return in the app property setting.

3.3 Adding and Editing Pages

Video: Adding Pages to Your App

The **New Page** menu provides several preformatted page templates. When you insert a page, first choose a **Main Page** or a **Subpage**.

Figure 3–5 Inserting a New Page for a Tablet App

Main Pages and Subpages

A main page is a page at the top level of your app. A main page displays in the app menu. You can navigate through the main pages of the app by swiping through the app sequentially, or you can navigate directly to a specific page using the page menu.

A subpage presents detail information of its parent page. A subpage does not display in the app page menu. Typically you navigate to a subpage by tapping a data item on its main page (such as a tile or a chart value). The data presented on the subpage is automatically filtered based on the item you tapped on the main page. When you access a subpage by swiping the previous page, the subpage displays all data received from the previous page.

When you create a subpage beneath another subpage, each subpage treats the subpage before it as its parent page; that is, the data presented on the subpage is automatically filtered based on the item you tapped on the previous subpage.

Video: Adding Detail Information Using Sidebars and Subpages

3.3.1 Selecting a Page Template

The Mobile App Designer provides several preformatted page templates to help you get started with your design:

- **Cover**

The default cover page includes an image and headings. Use a cover page to introduce your app. See [Editing the Cover Page](#) to customize your cover page.

- **Navigation**

A navigation page provides a master-detail interaction between a set of hierarchical filters that users select to control the display of the detail region of the page. The detail region contains visualizations you define that automatically update based on the selection in the navigation region. See [Adding Navigation Pages](#).

- **Exploration** (available for tablet apps only)

An exploration page provides dynamic filtering based on fields from your data source that you choose. This design provides your app consumers the most flexibility for exploring combinations of dimensions and measures. Users can easily mix and match filters in new and different ways to gain insights through exploration. See [Adding Exploration Pages](#).

- **Tile**

A tile page provides a set of dynamic, scrollable tiles containing visualizations that enable easy comparison of key measures across your data set. A tile is generated for each occurrence of a specific item in your data. See [Adding Tile Pages](#).

- **Accordion** (available for phone apps only)

The accordion page provides an expandable display of a specific data dimension and a key measure (such as Product and Revenue). Add visualizations to the expansion area that you can expand and collapse easily to accommodate the mobile device viewing area. See [Adding Accordion Pages](#).

- **Columns or cells**

Choose from several column-cell design patterns to arrange your app components.

- **Blank**

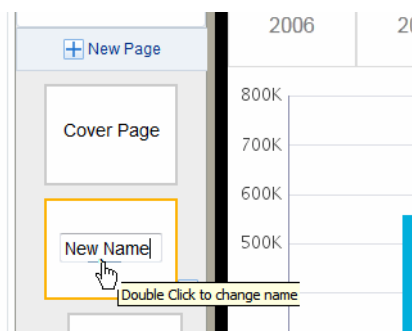
To design a custom page layout, choose **Blank**. You can stack components by simply dragging and dropping them to the blank page or, insert **Frames** to create your own column and cell arrangement for side-by-side and more precise placement of components. See [Inserting Frames](#).

3.3.2 Changing the Page Name

The page name displays at the top of the page and in the app menu. To edit the page name:

- Double-click the page name text on the page icon in the left column as shown in [Figure 3–6](#) to enable the text for editing. Enter the new name.

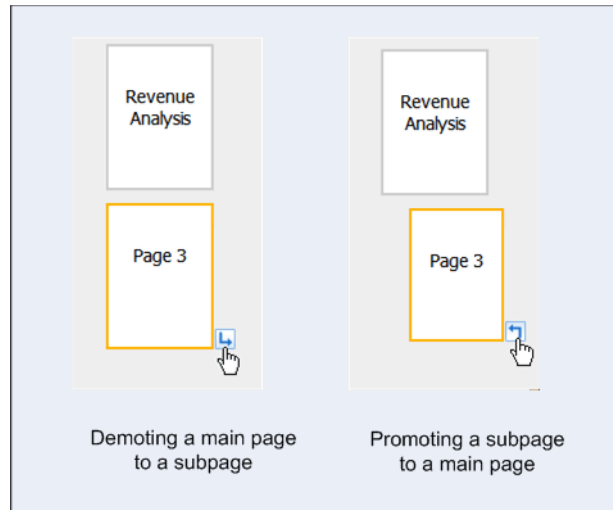
Figure 3–6 *Updating the Page Name*



3.3.3 Changing Between Main Page and Subpage

A main page can be demoted to a subpage by clicking the right-arrow icon. A subpage can be promoted to a main page by clicking the left-arrow icon.

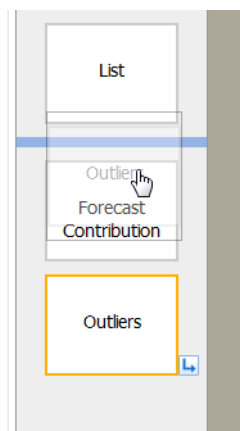
Figure 3–7 *Changing the Page Hierarchy*



3.3.4 Moving Pages

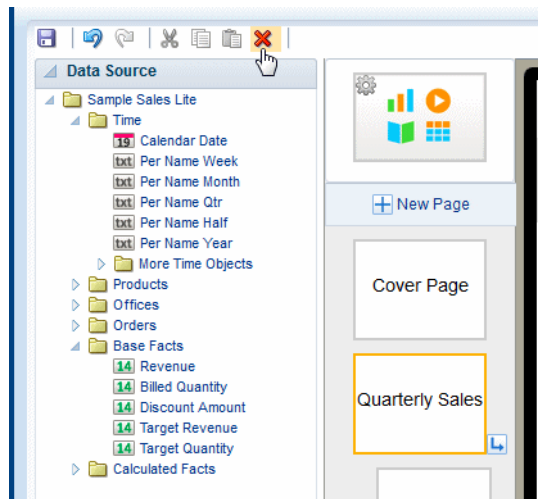
To move a page, select and drag the page to the new position.

Figure 3–8 *Moving a Page*



3.3.5 Deleting Pages

To delete a page, select the page and then click the delete icon in the toolbar.

Figure 3–9 Deleting the Quarterly Sales Page

3.4 Editing the Cover Page

A cover page with a default image is included when you create an app. You can update the cover page image and title text; or, delete these default components and insert any other app component to the cover page.

Video: Changing Your App Cover Page

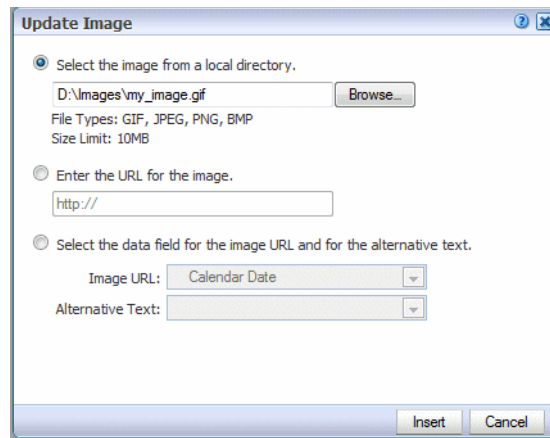
3.4.1 Updating the Cover Page Background Image

To update the cover page background image:

1. Double-click the image on the page.
2. In the **Update Image** dialog, specify one of the following sources for the image:
 - **Select the image from a local directory:** Click **Browse** to specify the file name and directory of the image on a local directory to upload the image.
 - **Enter the URL for the image:** Enter the URL where the image is stored.
 - **Select the data field for the image URL and for the alternative text:**

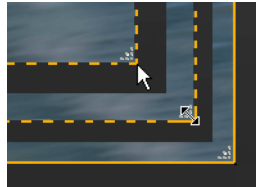
Image URL: Select the field from the data that contains a URL to an image.

Alternative Text: If your data includes a field that contains alternative text for the image, you can select that field to define the alternative text that is displayed for the image.

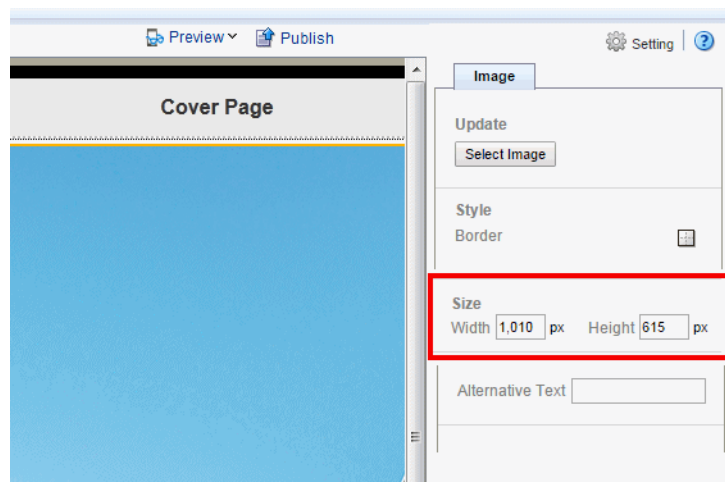
Figure 3–10 Update Image

3. Optionally resize the image in one of these ways:

- Drag the right bottom corner of the image. To preserve the aspect ratio when resizing an image, press and hold the Shift key before starting to drag the corner.

Figure 3–11 Resizing an Image

- Modify the width and height in the **Properties** pane. With the image selected, you can enter precise height and width values in pixels.

Figure 3–12 Setting Image Dimensions from the Properties pane

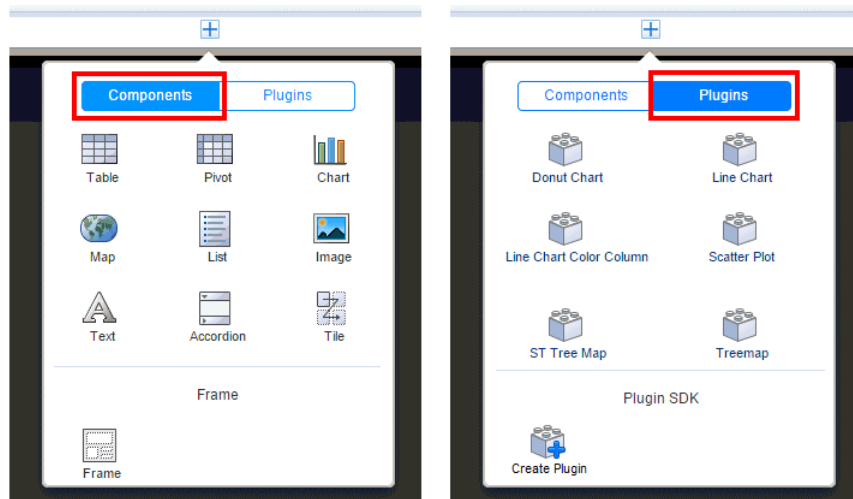
3.5 Inserting Components and Plugins to Your Pages

After adding a page you can start inserting visual components to the page. Mobile App Designer not only provides a full range of components such as charts, tables, and

maps, but also provides a facility for you to extend the visualization capabilities by creating your own plugins.

To insert a component or plugin:

Click the **Insert** icon. Choose from the **Components** or **Plugins**.



The components are documented in this chapter.

The plugins delivered with this release are samples to give you some ideas for designing your own. For additional descriptions see:

- <http://www.oracleappdesigner.com/2015/07/introduction-to-line-bar-chart-plugin.html>
- <http://www.oracleappdesigner.com/2015/07/plugin-donuts-chart.html>

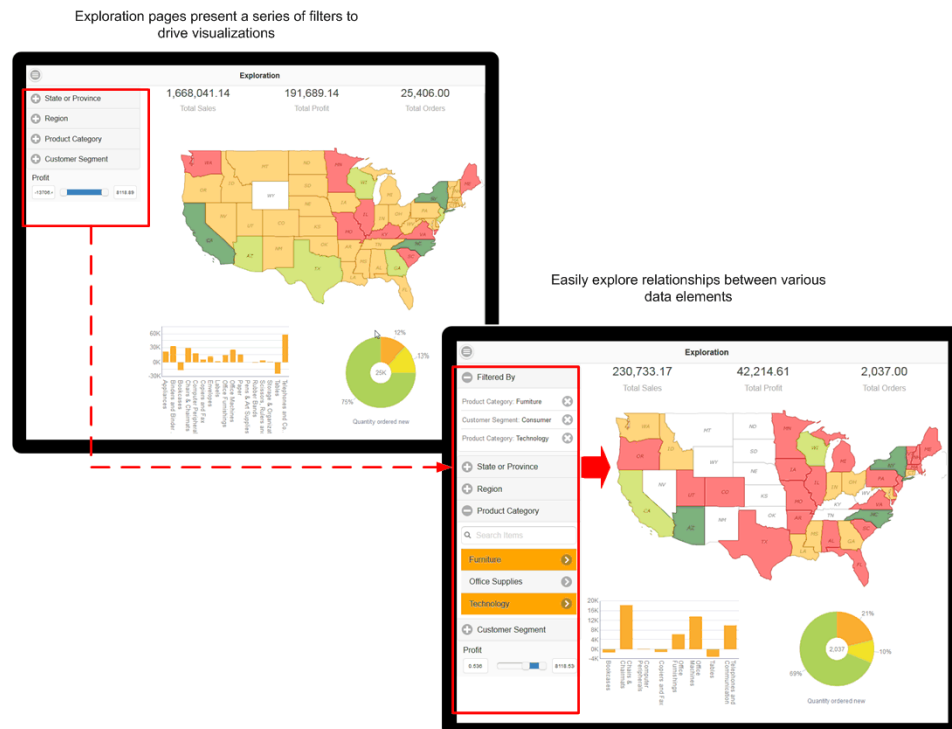
For information about creating your own custom plugins see [Adding Custom Plugins](#).

3.6 Adding Exploration Pages

Use an Exploration page to provide your tablet app consumers the most flexibility in choosing combinations of data to visualize.

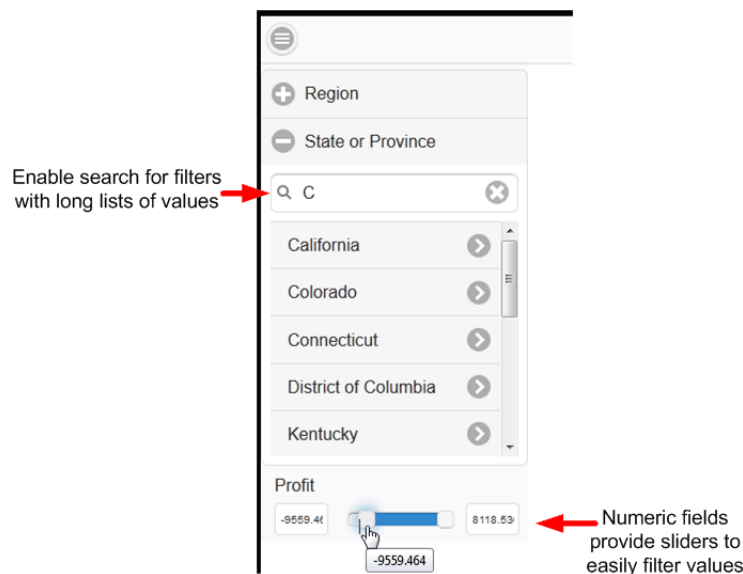
Video: Adding Exploration Pages to Filter Data

Exploration pages provide multiple filters to enable easy exploration of the data presented on the page. Any text or numeric fields (including calculated fields) can be added as filters enabling your users to mix and match the data that interest them to provide new and personal views of the data.

Figure 3–13 Sample Exploration Page

Numeric fields (such as Sales, Salary, Units Sold) are presented as sliders. You can customize the operator that defines the slider.

For text filters with long lists of values, you can enable a search option. The search option prompts you to start typing text to narrow the choices.

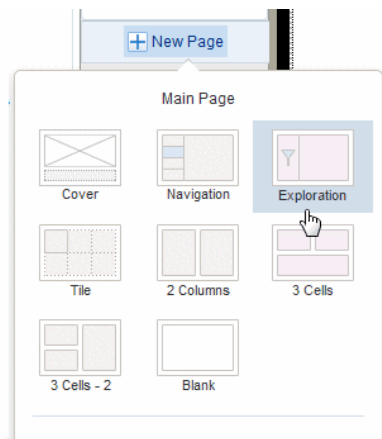
Figure 3–14 Filtering Options for Text and Numeric Fields

At run time you can select multiple values from each filter list. You can also define default values for filters so that when the page launches, it is initially filtered by one or more values.

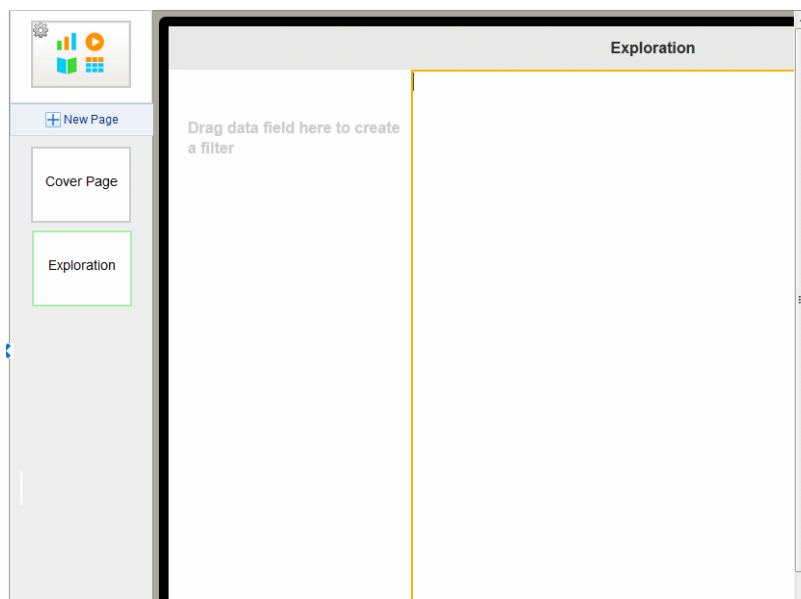
Note that date fields are not supported.

To add an Exploration page:

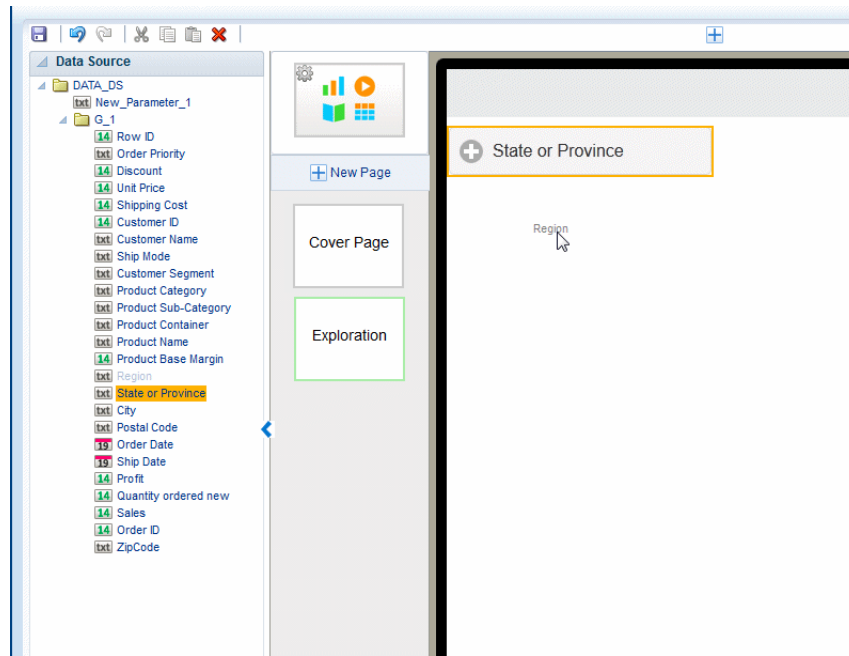
1. Click **New Page** and then click **Exploration**.



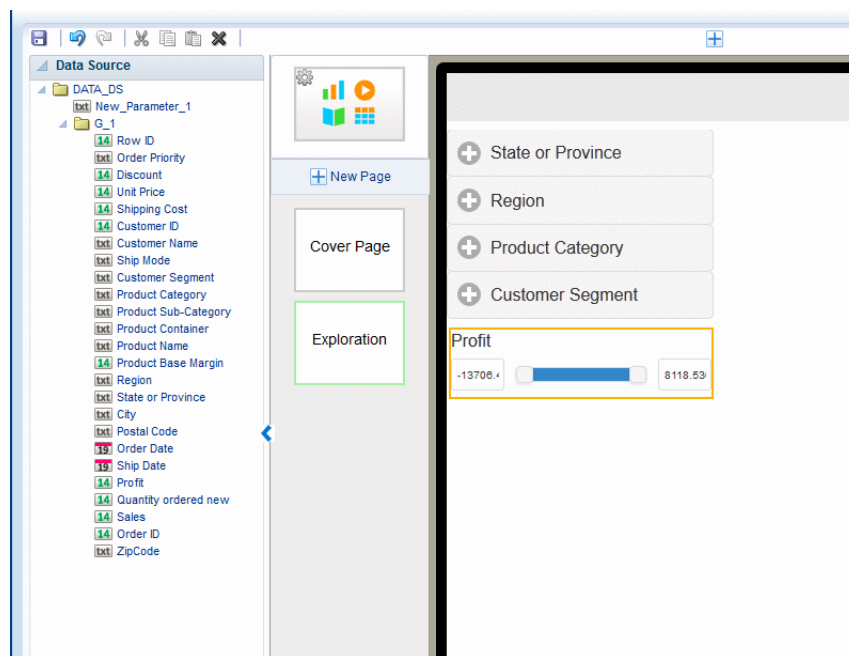
The inserted page is divided into two sections. The left section is where you drag data fields to create filters. The wider, right hand section is where you insert the visualizations that will be driven by the filters.



2. Drag text or numeric data fields from the Data Source pane or from the Calculated Fields pane and drop them on the left side of the page to create as many filters as desired for your app page.



The field labels from the data source are displayed by default. You can update the filter label on the Properties pane. Numeric fields display as a slider-style filter.



3. Add the visualizations (charts, maps, tables) to the right side of the page. All the visualizations are filtered when you make selections on the left side.
4. Optionally customize the filters:
 - [Customizing Text Filters on the Exploration Page](#)
 - [Setting the Operator and Values for Numeric Filters on the Exploration Page](#)

3.6.1 Customizing Text Filters on the Exploration Page

Use the Properties pane to apply the following customizations to text filters:

- [Specify Default Values for Text Filters](#)
- [Enable or Disable Search for Text Filters](#)
- [Customize Selected Background Color and Font Weight](#)

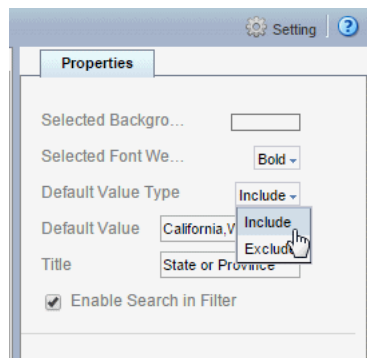
3.6.1.1 Specify Default Values for Text Filters

You can set default values for filters so that when users open the page specific values are already applied. You can remove the defaults as you explore the data, choosing different values. You can define values to include by default, or values to exclude by default.

To specify default values:

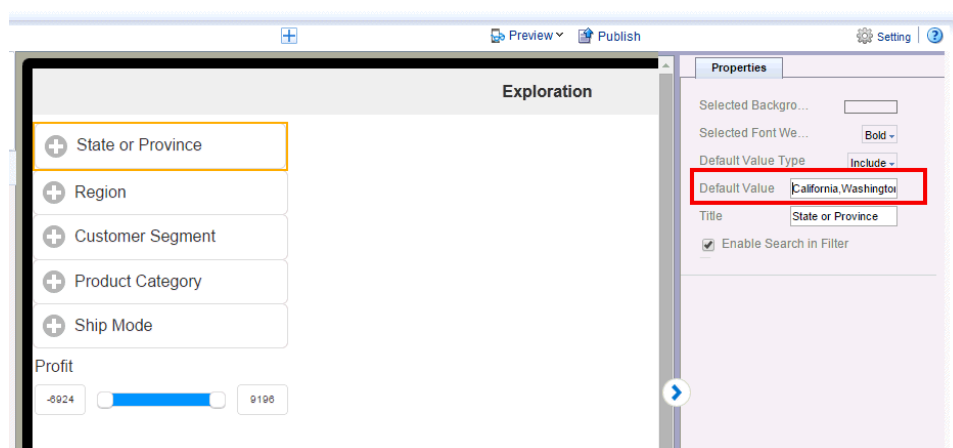
1. Select the Filter to display its Properties.
2. In the **Default Value Type** property, choose whether to Include or Exclude the specified values.

Figure 3–15 Choosing the Default Value Type



3. In the **Default Value** property, enter the values. Enter multiple values separated by a comma. (**Important:** Do not enter spaces between the values.)

Figure 3–16 Entering Default Values for the State or Province Filter



3.6.1.2 Enable or Disable Search for Text Filters

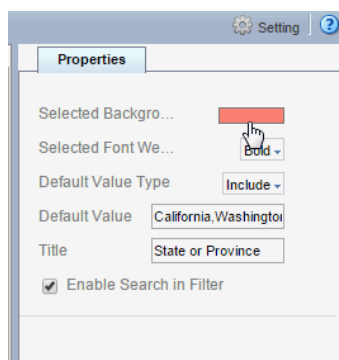
By default text filters include the Search option (see [Figure 3–14](#)) to enable you to more easily navigate long lists of values. For shorter lists you can disable this feature by deselecting the **Enable Search in Filter** option.

3.6.1.3 Customize Selected Background Color and Font Weight

The selected background color and font weight define how the filters that are selected display in the list. Set these properties to highlight the active filters.

To change the default background color: On the Properties pane, click the **Selected Background** color bar, then choose a new color from the color picker.

Figure 3–17 Customizing the Selected Background Color



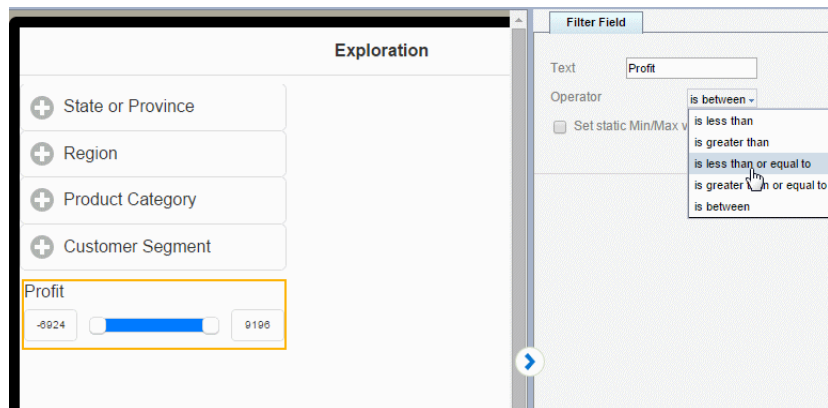
To change the default **Selected Font Weight**: On the Properties pane, select Bold, Light, or Normal from the list.

3.6.2 Setting the Operator and Values for Numeric Filters on the Exploration Page

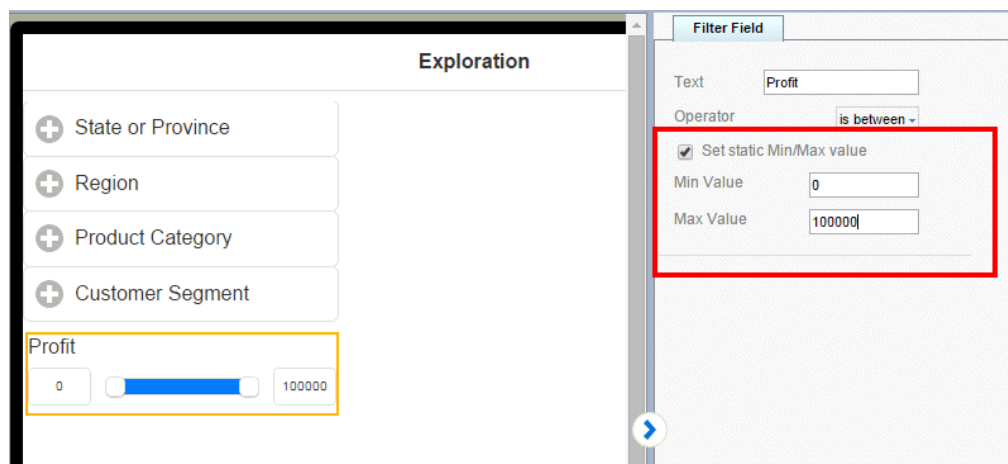
By default, the Operator "is between" is applied with the Minimum and Maximum values applied dynamically from the lowest and highest values present in the data. You can select a different operator as well as configure filters to display static minimum and maximum values.

To specify values and operator for numeric fields:

1. Select the numeric Filter to display its Properties.
2. To change the Operator, select from the list. Supported operators are:
 - is less than
 - is greater than
 - is less than or equal to
 - is greater than or equal to
 - is between (default)

Figure 3–18 Changing the Numeric Filter Operator

3. To set absolute minimum and maximum values for the slider, rather than have the values set dynamically from the minimum and maximum values present in the data, select **Set static Min/Max Value**, and then enter the absolute minimum and maximum values.

Figure 3–19 Setting Values for the Slider Numeric Filter

3.7 Adding Tile Pages

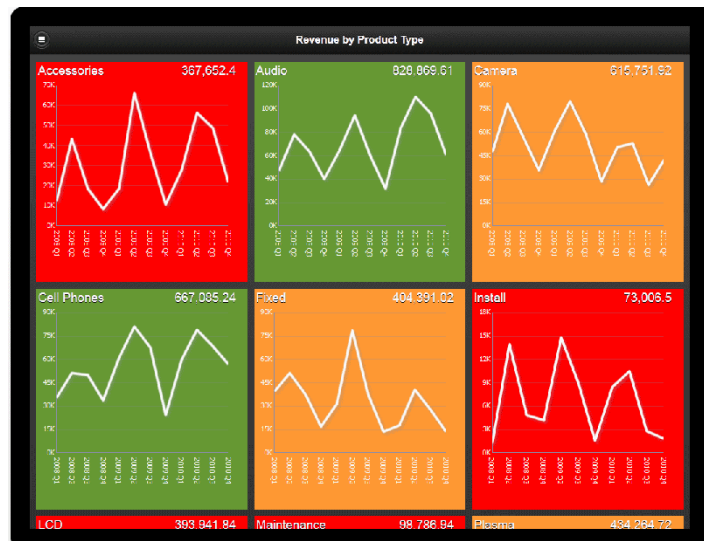
Tile pages provide a scrollable set of tiles each containing the same components repeated for a specific field in your data. Use a tile page to provide an instant view of a key measure across a specific dimension. This view not only provides a complete picture within each tile, but also enables an at-a-glance comparison across your data set.

Video: Adding Tile Pages to Apps

The example tiles shown in [Figure 3–20](#) provide an overview of revenue for a set of products. Each tile displays the overall revenue and a chart showing revenue by quarter. When you design a tile page you can include any visualization in each tile.

Figure 3–20 Tiles Displayed on a Tablet

To enhance the visual impact of the values represented in each tile, you can add stoplight formatting to highlight tiles with results that fall outside the range of specific threshold values.

Figure 3–21 Tiles with Stoplight Formatting

When a tile page is paired with a subpage, each tile is a touch point to open the subpage filtered by the tile value. This combination enables your users to quickly assess areas that require attention and drill down for deeper analysis.

Figure 3-22 Tap Tile to Display Detail Subpage

Features of tile pages include:

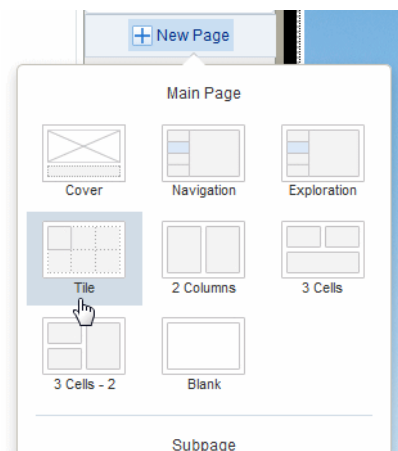
- Each tile is a touch point. When you associate a subpage with a tile page, tapping a tile opens the detail subpage.
- Tiles load dynamically as you scroll down.
- Tile width is based on the number of columns specified and your device type display area. You can define the number of columns up to four across.
- You can customize the tiles to include any component.

3.7.1 Creating a Tile Page

When you insert a new page, **Tile** is available as a page template selection. You can also insert a tile into a specific page area using the option on the **Add Component** menu.

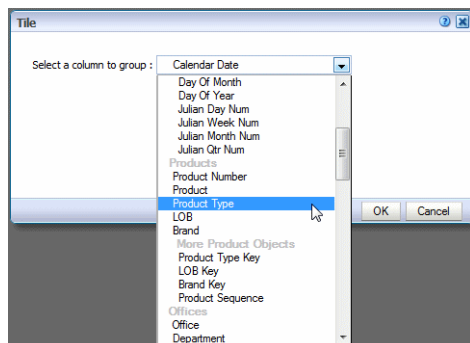
To insert a tile page:

1. Click **New Page** and then click **Tile**.



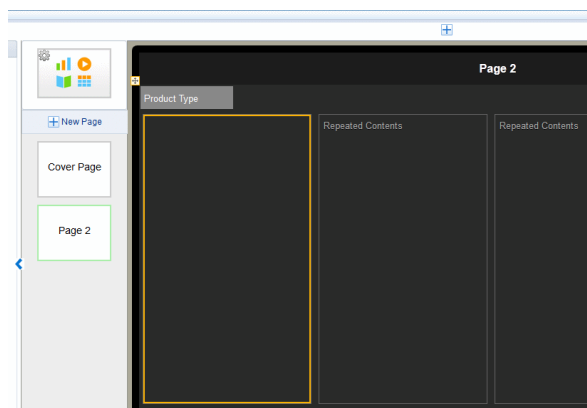
2. In the **Tile** dialog, choose the data field that you want grouped for each tile. The drop-down list includes all the fields from your data. The example tile page in [Figure 3–23](#) will display one tile for each Product Type.

Figure 3–23 *Selecting Data Field Column to Define Tiles*



When inserted, the tiles display as shown in [Figure 3–24](#). Note that in the app designer, only the first tile displays the contents. Use **Preview** when you want to see content across all tiles.

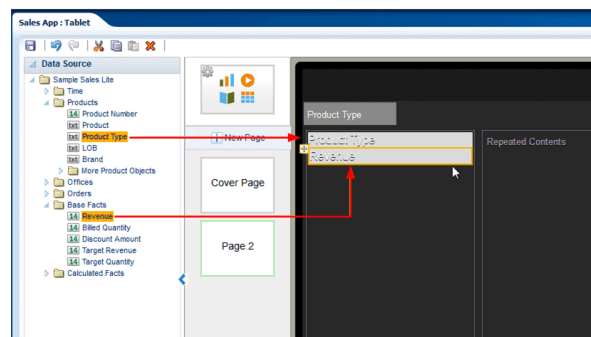
Figure 3–24 *Inserted Tile Page*



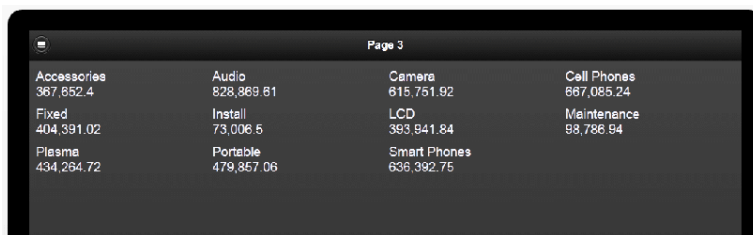
3. To add contents to the tiles, select the first tile, and drag the components from the **Insert** menu to the first tile.

Tip: For precise placement of components within a tile, use Frames to position your contents. See [Inserting Frames](#).

A common use case is to add the data field label to the tile with a measure. For example, drag **Product Type** from the **Data Source** pane to the first tile and then drag **Revenue** from the **Data Source** pane to the first tile.

Figure 3–25 Adding Field Label and Measure to Tile

4. Preview your tile page to view the contents so far. Click **Preview**.



To finish your tile page, add the components to the first tile. For example, to add a chart, simply drag the chart component from the **Insert** menu and customize the chart as desired. An example is shown in [Figure 3–26](#). For information about more customizations you can apply to your tile page, see [Customizing Tiles](#).

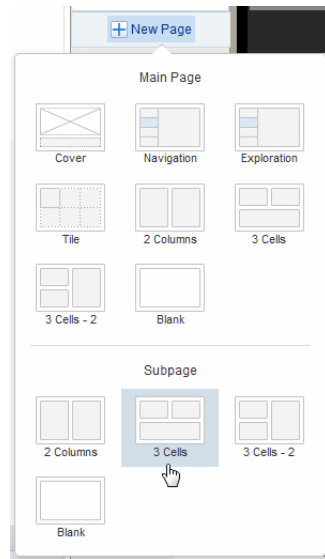
Figure 3–26 Sample Tile Page

3.7.2 Adding a Subpage to a Tile Page

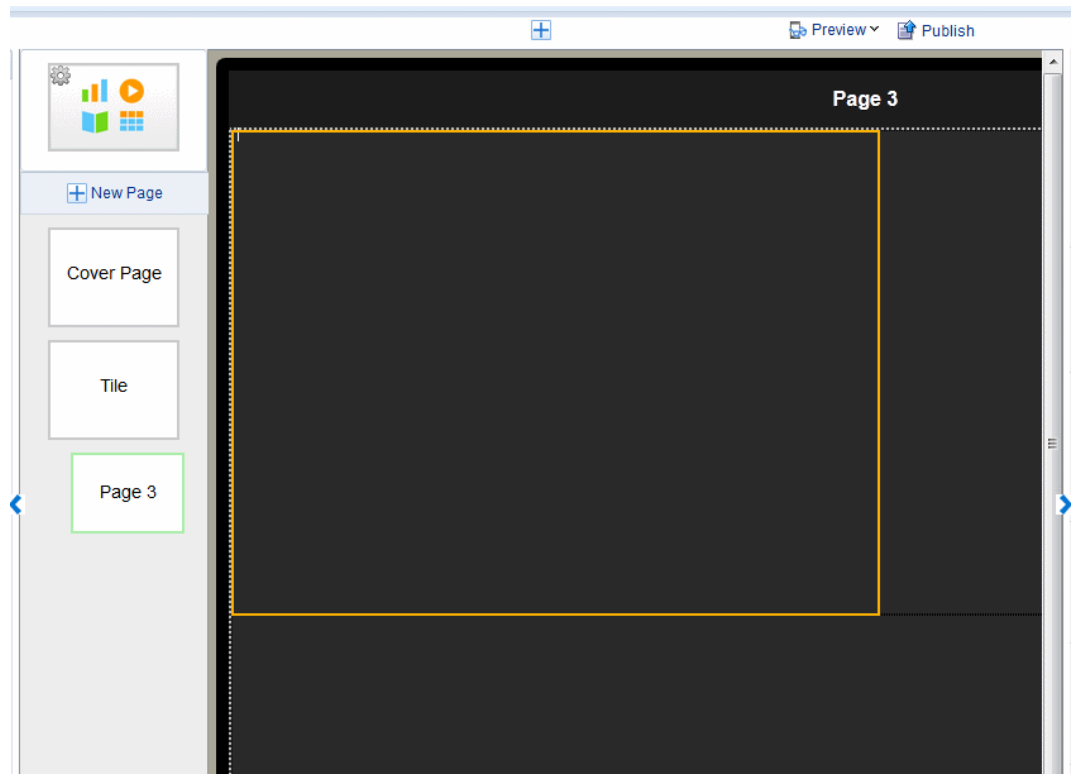
A common use case is to add a subpage to a tile page to provide more detailed information about the data provided in a specific tile. The detail subpage opens when you tap a tile.

To add a subpage:

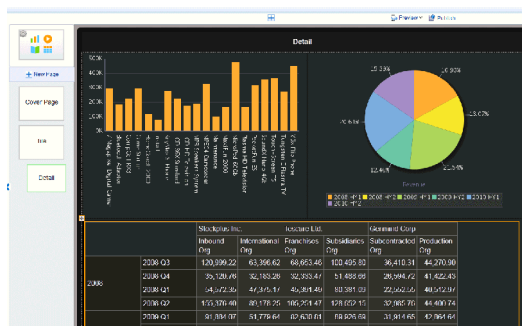
1. Select the Tile page.
2. Click **New Page** and then select one of the Subpage options from the menu.



3. The empty subpage is inserted for you to add content.



4. Add components to your subpage. The example in [Figure 3-27](#) shows a subpage with two charts and a pivot table. At run time, when you tap a tile on the tile page, the components on this page show detail for the tapped item only.

Figure 3–27 Example Subpage

3.7.3 Customizing Tiles

You can insert any combination of components to the tiles. To accommodate the components you choose you can apply the following customizations:

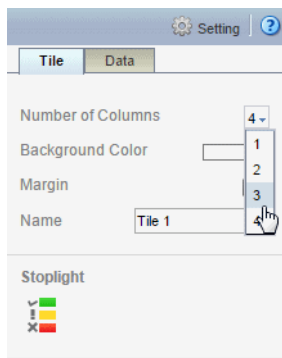
- [Specify Number of Columns](#)
- [Apply a Background Color](#)
- [Apply Filters](#)
- [Sort](#)
- [Add Stoplight Formatting](#)
- [Resize Tile Margins](#)

Select the first tile to display the **Tile Properties** pane.

3.7.3.1 Specify Number of Columns

To change the number of columns that display across the page, click the **Columns** option and select the number of columns. Supported values are one through four.

The column width automatically adjusts depending on the size and type of mobile device used to view the app.

Figure 3–28 Changing the Number of Columns Displayed

3.7.3.2 Apply a Background Color

To change the default background color: On the Properties pane, click the **Background Color** bar, then choose a new color from the color picker.

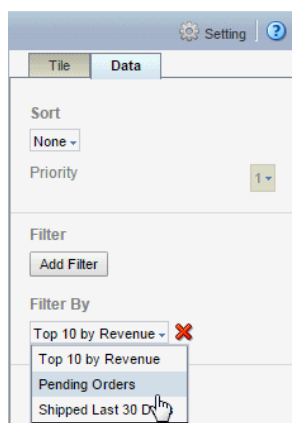
3.7.3.3 Apply Filters

Apply a filter to refine the items displayed in the tile page. For example, you can apply a filter to limit the display to only:

- The top 10 salaries
- The bottom 25 store sales
- Employees in the IT department
- Sales that are between \$10,000 and \$20,000 and in the Southern region

You can add multiple filters and manage the order in which they are applied to the tile page.

The **Apply Filter** option is available from the **Data** tab of the properties pane.



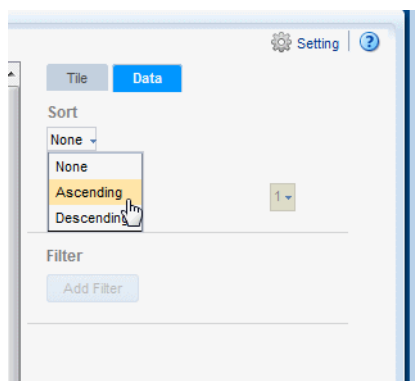
See [Adding Global Filters](#) for information about creating filters.

3.7.3.4 Sort

The **Sort** option sorts the tiles by the field you selected as the tile grouping field. For example, if you chose Product Type as the tile grouping field, and then select Sort ascending, the tiles will be sorted from A - Z by the Product Type name.

To sort the tiles:

1. Select the first tile to view the **Tile** properties.
2. In the Tile properties pane, click the **Data** tab.
3. From the **Sort** menu choose **None** to apply no sort, **Ascending**, or **Descending**.



3.7.3.5 Add Stoplight Formatting

Stoplight formatting applies one of three distinct background colors to each tile depending on the value of a chosen aggregated field in your data. You can customize the colors or use the default colors red, amber, and green. Use stoplight formatting to quickly assess an indicator as unacceptable, acceptable, or desirable.

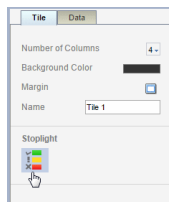
The Stoplight feature enables you to highlight values conditionally based on static threshold values, a comparison to a target value, or a comparison to another field.

- [Apply Formatting Based on Static Values](#)
- [Apply Formatting Based on Percent Achievement](#)
- [Apply Formatting Based on Percent Growth](#)

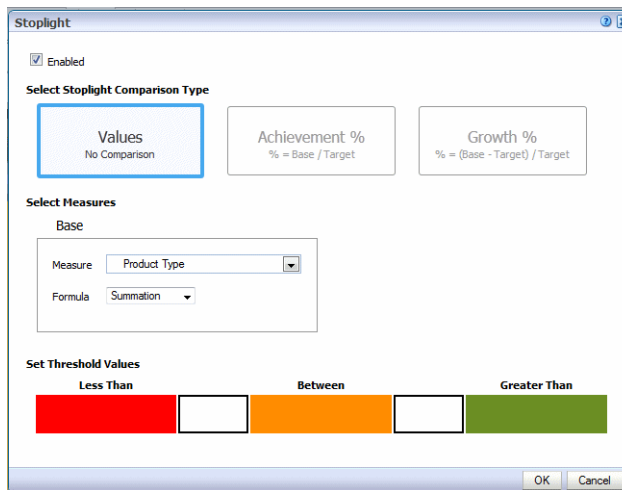
3.7.3.5.1 Apply Formatting Based on Static Values Use this option when the unacceptable, acceptable, and desired values are the same for the aggregated field for all items. You can customize the colors for each range.

To apply conditional formatting based on static values:

1. Select the first tile to view the Tile properties. On the Tile properties pane, click **Stoplight**.

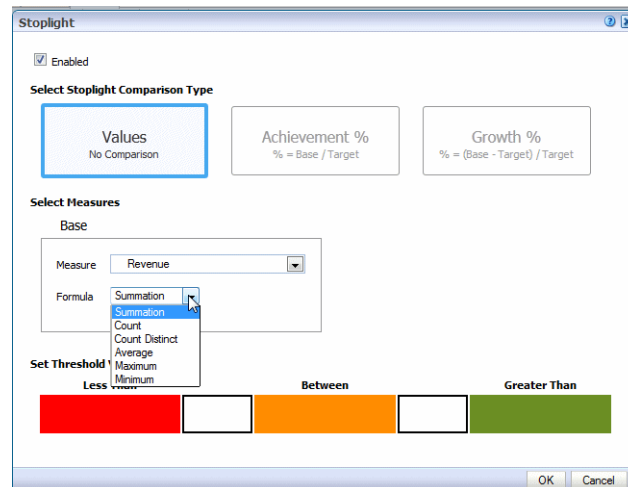


2. In the **Stoplight** dialog select **Values**.

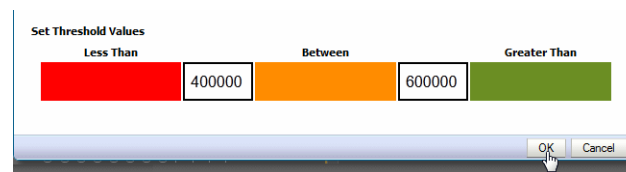


3. Select the **Measure** field on which to base the formatting and then select the formula to apply to the Measure.

For example, assume you have grouped tiles by Product Type. You want to apply formatting to each tile based on the value of Revenue for each product type. Select Revenue as the **Measure** and Summation as the **Formula**.



4. Enter the threshold values for the background colors. Values less than the left entry will display the **Less Than** background color. Values greater than the right entry will display the **Greater Than** background color. Values between display the **Between** color.



To change the default colors, see [Customizing the Colors](#).

5. Click OK.

Note that for Tile pages the stoplight formatting does not display in the design workspace.

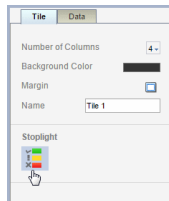
6. Click **Preview** to view how the formatting will display at run time.



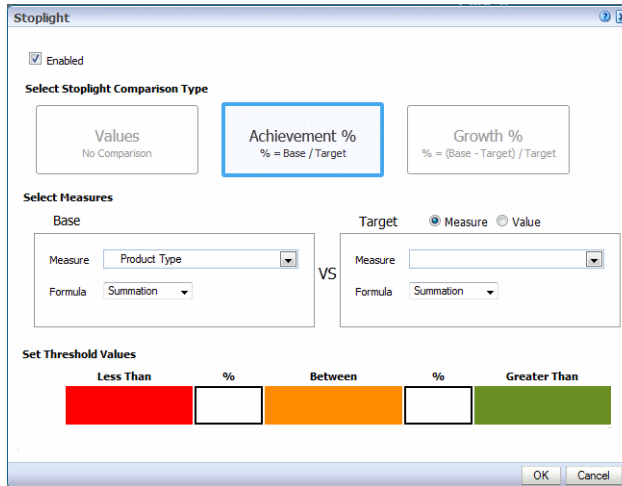
3.7.3.5.2 Apply Formatting Based on Percent Achievement This option enables you to conditionally highlight the aggregate value based on the percentage of a target value. For example, you want to highlight sales that are less than 50 percent of target to display as red, sales that are 50-75 percent of the target as amber and sales that are greater than 75 percent of the target to display as green.

To apply conditional formatting based on the percent achievement of a target:

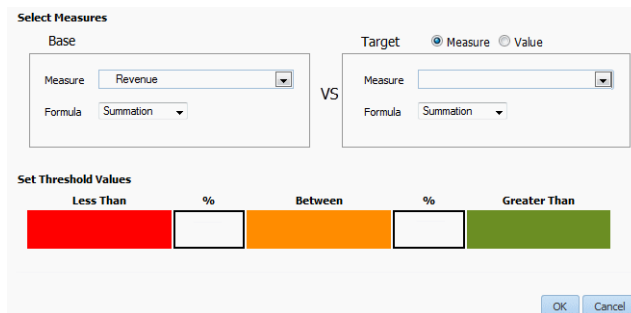
1. Select the first tile to view the Tile properties. On the Tile properties pane, click **Stoplight**.



2. In the Stoplight dialog select **Achievement %**.



3. Select the **Base Measure** from the data field list and the **Formula** to apply to the measure field. In this example, Revenue will be summed for each tile to establish the base measure.



4. Choose the **Target Measure** or enter a **Value**.
 - **Measure** - select a field from the data to supply the target value for comparison.
 - **Value** - enter a static value to supply the target value for comparison.

The app calculates what percentage the calculated Base measure value is of the Target value. In the example below, the Revenue column (Base) is compared to the Target Revenue column (Target). The calculation performed is:

$$\text{Revenue} / \text{Target Revenue} \times 100\%$$

Therefore if your Revenue is \$8,000 and your Target Revenue is \$10,000, the percent achievement is:

$$8,000/10,000 \times 100\% = 80\%$$

5. Enter the **Less Than** and **Greater Than** percentage values for the background colors.

If the Base value percentage of the Target value is less than the percentage you enter on the left, the **Less Than** color displays. If the Base value percentage of the Target value is greater than the value you enter on the right, the **Greater Than** color displays.

6. To change the default colors, see [Customizing the Colors](#).

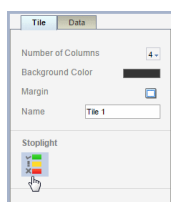
3.7.3.5.3 Apply Formatting Based on Percent Growth This option enables you to conditionally highlight the aggregate value based on the percent difference of the base value from the target value. The calculation applied is:

$$(\text{Base} - \text{Target}) / \text{Target} \times 100\%$$

For example, you want to compare sales from the current and previous quarters and you want to see when sales in the current quarter showed less than 10% growth from the previous quarter. When sales showed 10% growth you want to display the value in green, when 0-10% growth display amber, and when less than 0% growth display red.

To apply conditional formatting based on percent growth:

1. Select the first tile to view the Tile properties. On the Tile properties pane, click **Stoplight**.



2. In the **Stoplight** dialog select **Growth %**.

3. Select the **Base Measure** from the data field list and the **Formula** to apply to the measure field. In this example, Revenue will be summed for each tile to establish the base measure.

4. Choose **Target Measure** or **Value**.
 - **Measure** - select a field from the data to supply the target value for comparison.
 - **Value** - enter a static value to supply the target value for comparison.


The app calculates the percent difference that the **Base** column value is from the **Target** value. In the example below, the Revenue column (Base) is compared to the Quarter Ago Revenue column (Target). The calculation performed is:

$$((\text{Revenue} - \text{Quarter Ago Revenue}) / \text{Quarter Ago Revenue}) \times 100\%$$

Therefore if your Revenue is \$11,000 and your Quarter Ago Revenue is \$10,000, the percent growth is:

$$((11,000 - 10,000) / 10,000) \times 100\% = 10\%$$

5. Enter the **Less Than** and **Greater Than** percentage values for the background colors.



The **Select Measures** dialog box is used to compare a Base measure with a Target measure. It features two main sections: **Base** and **Target**, separated by a **VS** label. The **Base** section has a **Measure** dropdown set to **Revenue** and a **Formula** dropdown set to **Summation**. The **Target** section has a **Measure** dropdown set to **Quarter Ago Revenue** and a **Formula** dropdown set to **Summation**. Below these is the **Set Threshold Values** section, which includes three color-coded boxes: **Less Than** (red), **Between** (orange), and **Greater Than** (green). The **Less Than** box contains a percentage sign and the value **0**. The **Between** box contains a percentage sign and the value **10**. The **Greater Than** box is empty. At the bottom right are **OK** and **Cancel** buttons.

If the Base value percentage difference from Target value is less than the percentage you enter on the left, the **Less Than** color displays. If the Base value percentage difference from the Target value is greater than the value you enter on the right, the **Greater Than** color displays.

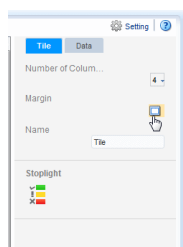
6. To change the default colors, see [Customizing the Colors](#).

3.7.3.6 Resize Tile Margins

You can adjust the space around each tile to make the tiles closer together or further apart.

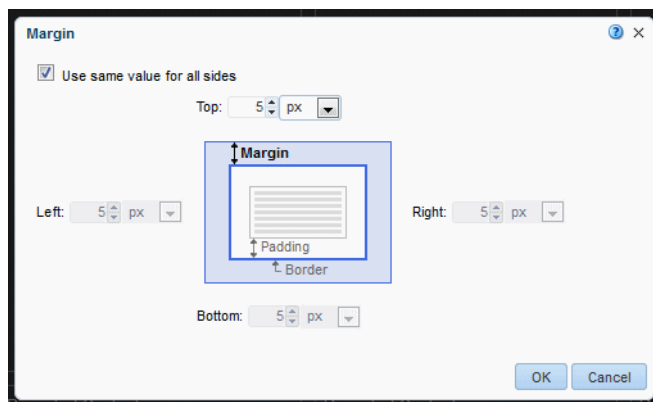
To customize the margin area between tiles:

1. Select the first tile to display the tile properties.
2. Click the **Margin** tool.



3. Enter the new margin values and select the unit of measurement.

Figure 3–29 Customizing Tile Margins



4. Click **OK**.

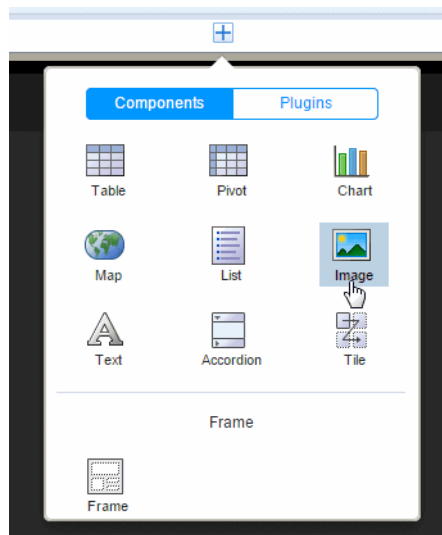
3.8 Inserting Images

The Mobile App Designer supports the following methods for including images in a page:

- **Static image:** Upload a static image that is saved in the app. An uploaded image file must be in one of the following graphic file formats: GIF, JPEG, PNG, or BMP. The image file cannot be larger than 10 MB.
- **Static URL:** Specify a static link to a URL where an image is stored. At run time the image is retrieved from the stored location.
- **Dynamic URL:** When the image URL is provided in a data field, specify the field that contains the URL. The value of the data field is evaluated at runtime enabling dynamic insertion of an image.

To insert an image:

1. Click **Insert**, and then drag the **Image** component to the page.

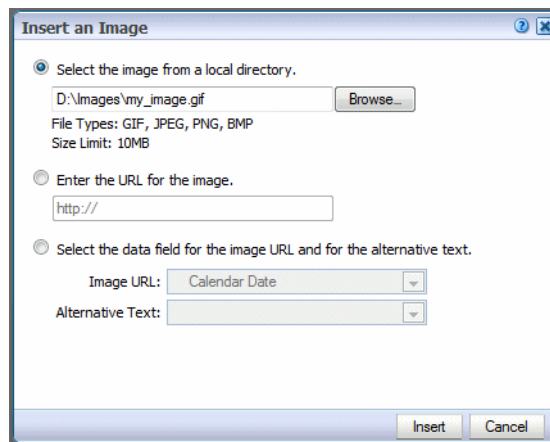


2. In the **Insert an Image** dialog, specify one of the following sources for the image:

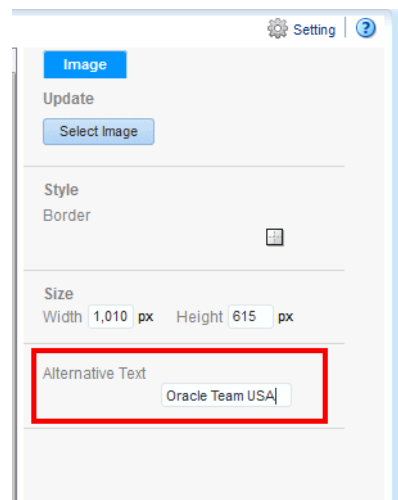
- **Select the image from a local directory:** Click **Browse** to specify the file name and directory of the image on a local or mapped drive to upload the image.
- **Enter the URL for the image:** Enter the URL where the image is stored.
- **Select the data field for the image and for the alternative text:**

Image URL: Select the field from the data that contains a URL of an image.

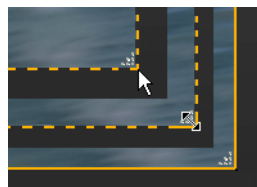
Alternative Text: If the data includes a field that contains alternative text for the image, then select that field to display alternative text.

Figure 3–30 Insert Image Dialog

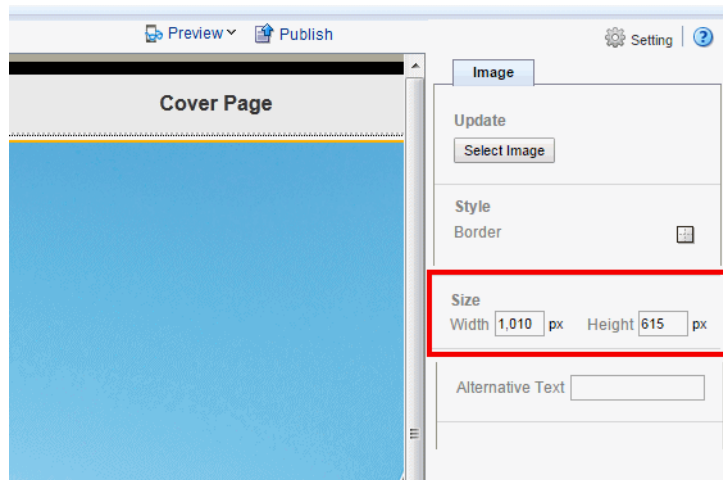
3. Add alternative text for the image. Click the **Properties** pane and enter text in the **Alternative Text** field as shown in [Figure 3–31](#).

Figure 3–31 Entering Alternative Text for an Image in the Properties Pane

4. Optionally resize the image in one of these ways:
 - Drag the right bottom corner of the image. To preserve the aspect ratio when resizing an image, press and hold the Shift key before starting to drag the corner.

Figure 3–32 Resizing an Image

- Modify the width and height in the Properties pane. The Properties pane enables you to enter precise height and width values in pixels, centimeters, inches, or points.

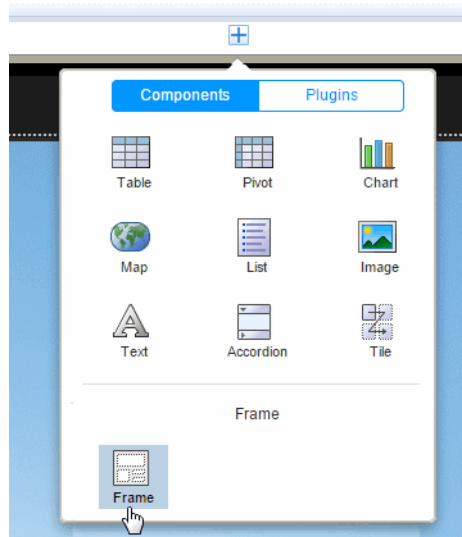
Figure 3–33 *Setting Image Dimensions from the Properties pane*

3.9 Inserting Frames

Use frames to divide your app page into sections for the precise positioning of components. When you start from a Blank page, typically you insert a frame before you begin inserting components. When you choose a preformatted template you can edit a default frame by selecting the frame and using the **Frame** properties. You can insert a frame inside another frame.

To insert a frame:

1. Select the area of the page where you want to insert the frame. On the **Insert Component** menu click **Frame**.



2. Enter the number of rows and columns for the frame and click **OK**.

Figure 3–34 shows the **Insert Frame** dialog.

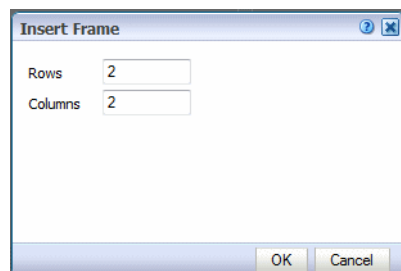
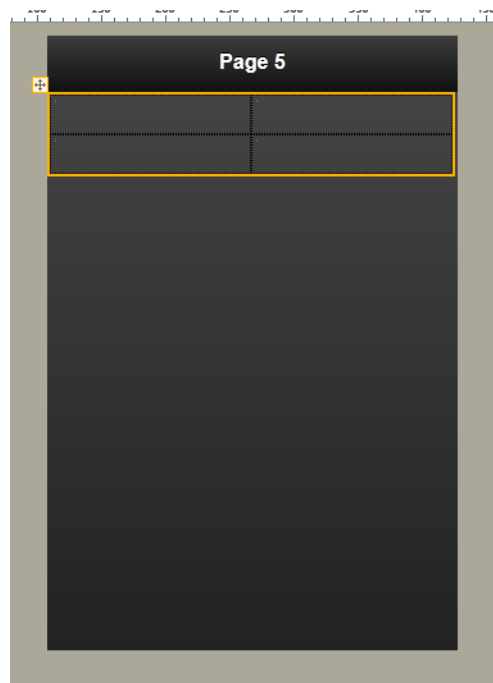
Figure 3–34 Insert Frame Dialog

Figure 3–35 shows the inserted frame.

Figure 3–35 Example of a Frame Inserted in the Design Area

Features of frames include:

- By default, frame columns are sized equally across the insertion area and frame row height defaults to the height of one row of text.
When you insert a component to a frame, the frame automatically resizes to accommodate the component.
- You can adjust the column width and height by either positioning the mouse pointer over the border and dragging the blue resize bar, or by changing the frame column properties in the **Properties** pane.
- You can insert a frame inside another frame.

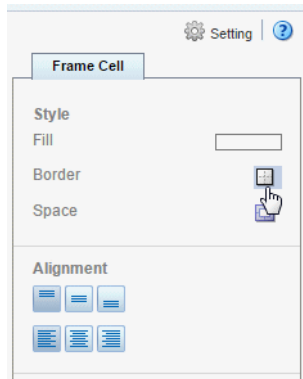
When you select a frame cell, the **Frame Cell** properties display to enable additional customization of fonts, alignment, borders, and background colors.

3.9.1 Adding a Border or Background Color

By default, the gridlines are displayed in the design area only and are not shown during run time.

To display the gridlines in your app:

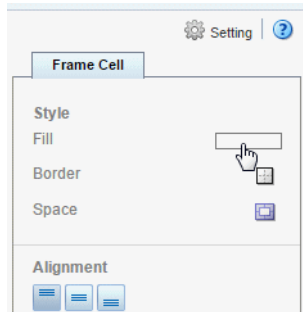
1. Select the frame to display the Frame Cell properties.
Click the **Border** icon.



2. Choose the border style from the **Border** dialog. See [Section 5.7, "Setting Borders"](#) for more information about the **Border** dialog.

To add a background color to a frame cell:

1. Click the **Fill** color bar to launch the Color Picker.

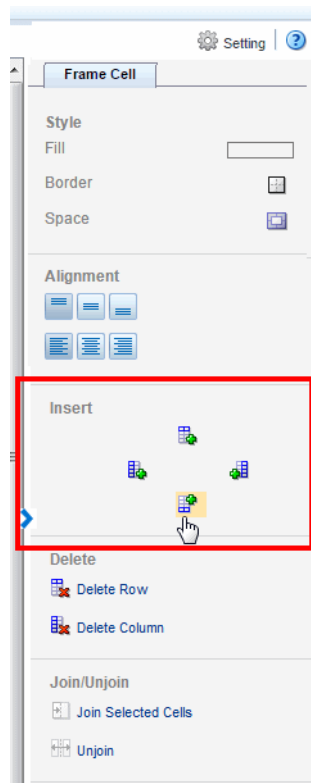


2. Select a color and click **OK**.
See also: [Using the Color Picker](#).

3.9.2 Inserting Additional Rows and Columns

To insert additional rows or columns to a frame:

1. Select the frame cell that is the focal point to display its properties.
2. Click the appropriate command button under **Insert**:

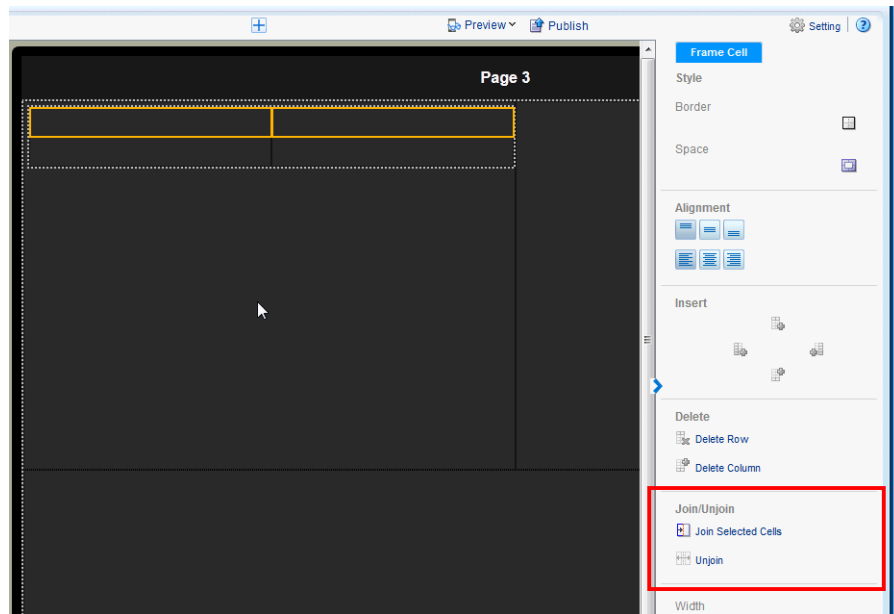


- Add a Row above
- Add a Column to the right
- Add a Row below
- Add a Column to the left

3.9.3 Joining and Unjoining Cells

To join frame cells horizontally or vertically:

1. Select multiple adjacent cells by holding down the Ctrl key and clicking each frame cell.
2. Click the **Join** command button.



To unjoin cells that have been joined:

1. Select the joined cell and click the **Unjoin** button.

3.10 Adding Navigation Pages

The creation process for the navigation page is different for tablets and phones. See the appropriate section for your app:

- [Navigation Page for Tablet](#)
- [Navigation Page for Phone](#)

See also:

Video: Adding Navigation Lists for Investigating Data

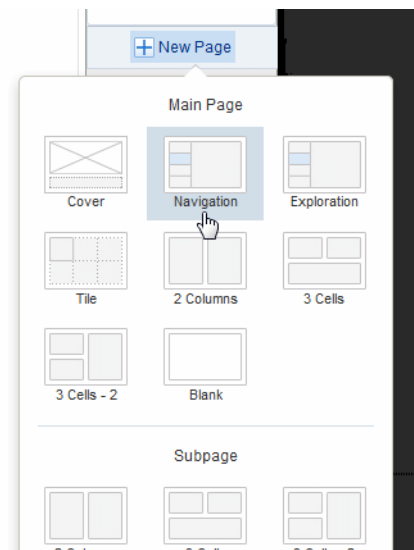
3.10.1 Navigation Page for Tablet

The Navigation page defines a master-detail relationship between a navigable set of filters and the visualizations displayed on the page. The navigation area of the page contains a hierarchy of filters that you tap through to drive the display on the detail side of the page. As you tap, you can see successively more refined sets of data, or you can stop within a level to see data just for its members, or a particular member. You can navigate up and down the hierarchical filters to see just the subset of data that interests you.

Figure 3–36 Navigation List Selection Drives the Detail Display

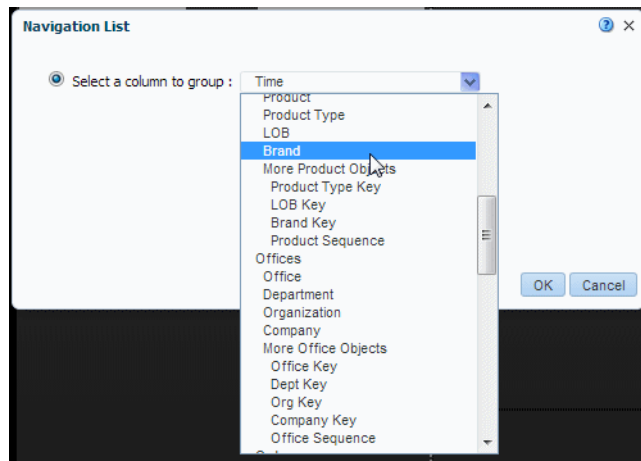
To insert a Navigation page:

1. Click **New Page** and then select **Navigation**.



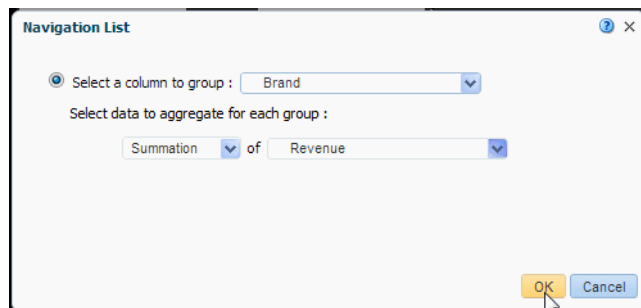
2. Select the data field from the list to define the top level of the navigation menu. In the example, the first level of the navigation list is grouped by Brand.

Figure 3–37 Selecting the First Level of the Navigation List

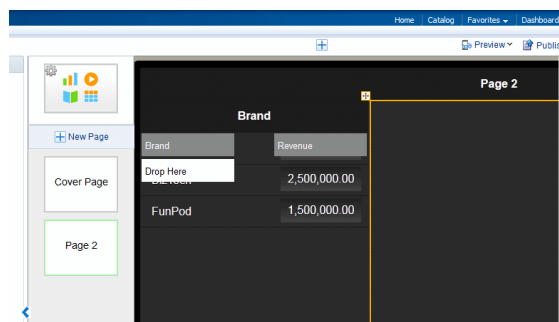


- Next choose a data field to aggregate for the group element and then choose an aggregation function. In this example, the Revenue for each Brand will be summed.

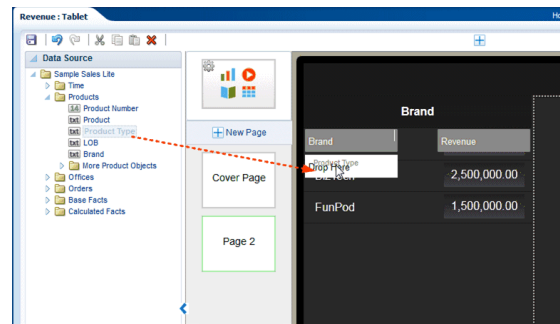
Figure 3–38 Choosing the Aggregation Function and Element



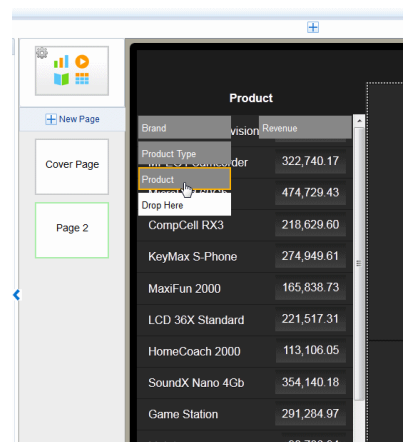
When you click **OK** the members of the group by column you selected display as the navigation list along with the aggregated value for each member of the list.



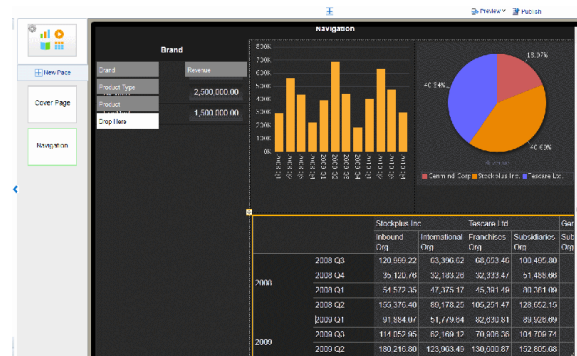
- To add a second level to the navigation list, drag the element from the Data Source pane to the **Drop Here** box on the page.

Figure 3–39 Adding a Second Level to the Navigation Page

- Continue adding levels as needed by dragging the fields from the Data Source pane to the **Drop Here** box. To see the members of a specific level, click that level.

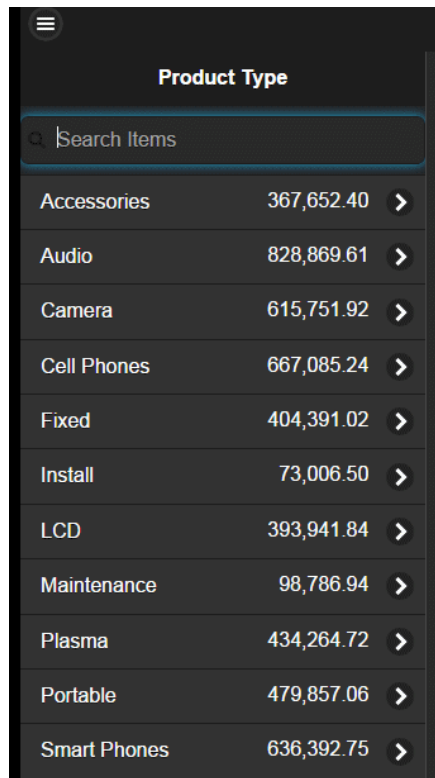
Figure 3–40 Viewing the Members of the Product Level

- To add stoplight formatting to your navigation list, see [Stoptlight Formatting for Navigation and Accordion Pages](#).
- Add the components to the detail region of the page. At run time, these components refresh as items on the navigation list are tapped. The example shows two charts and a pivot table.



3.10.1.1 Adding a Search Filter

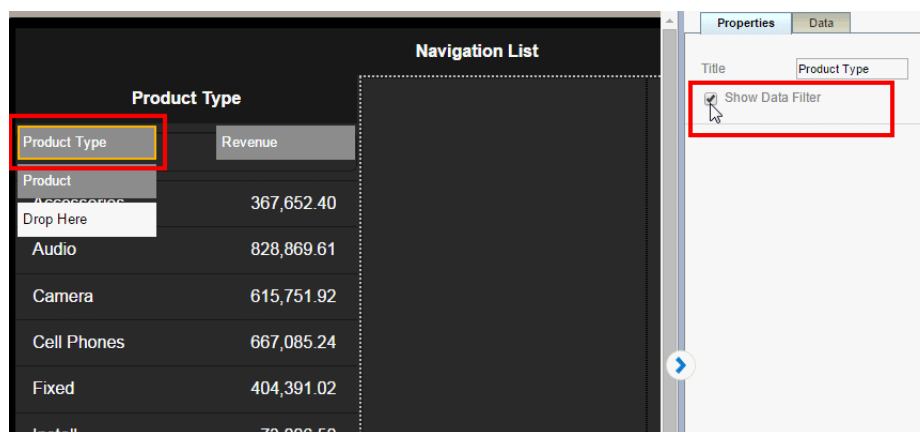
If a level in your navigation list includes many members (for example, States or Products) you can add a search filter to enable your app users to more easily find a specific item.

Figure 3–41 Navigation List with Search Filter Enabled

You can enable the search filter independently for each level.

To add a search filter to a navigation list level:

1. Select the navigation item for which you want to add a search filter.
2. In the Properties pane, enable the **Show Data Filter** property.

Figure 3–42 Adding a Search Filter to the Product Type Level in the Navigation List

3.10.2 Navigation Page for Phone

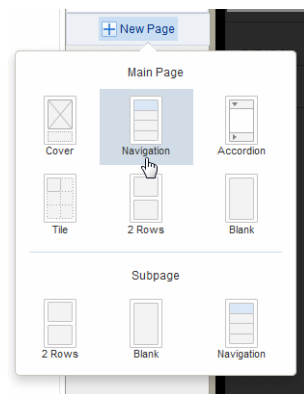
The navigation page for phone enables you to create a hierarchical list of filters that you can navigate to see a detail display of the specific items that interest you.

Tutorial: Creating Navigation Pages for Phone Apps

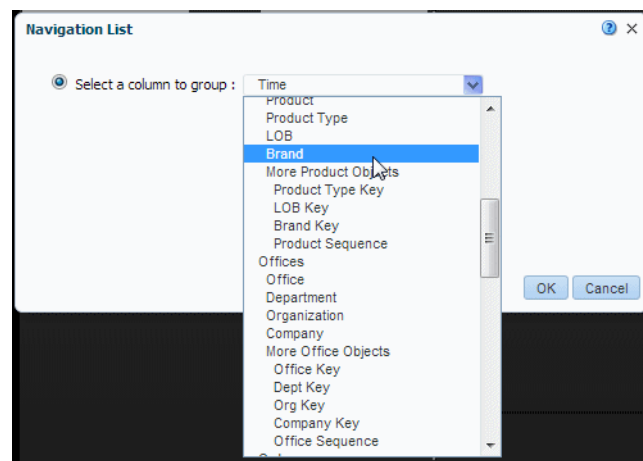
Figure 3–43 Navigation List Series on Phone

To create a navigation page:

1. Click **New Page** and then select **Navigation**.

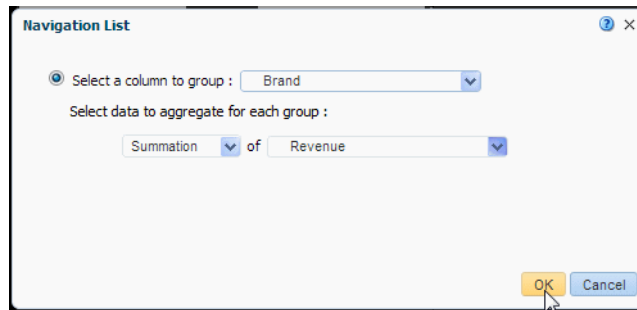


2. Select the data field from the list to define the top level of the navigation menu. In the example, the first level of the navigation list is grouped by **Brand**.

Figure 3–44 Selecting the First Level Hierarchy of the Navigation List

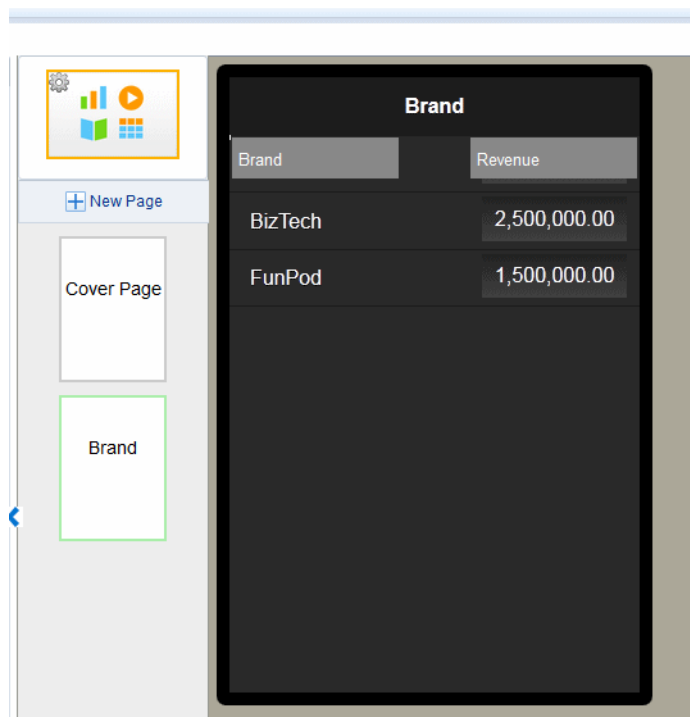
- Next choose the measure field to aggregate for the group element. In this example, Revenue is summed for each Brand (shown in [Figure 3–45](#)).

Figure 3–45 Choosing the Measure Field and Aggregation Function

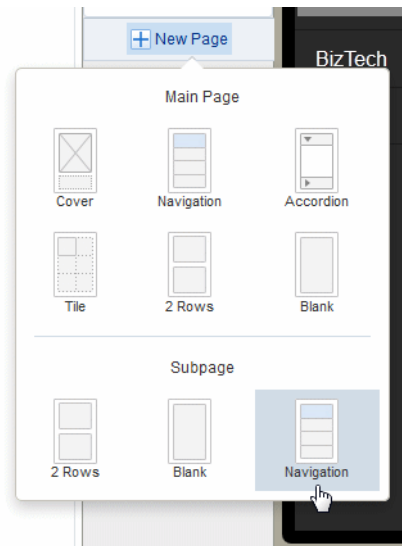


When you click **OK** the data element you selected displays as a list with the aggregated measure as shown in [Figure 3–46](#).

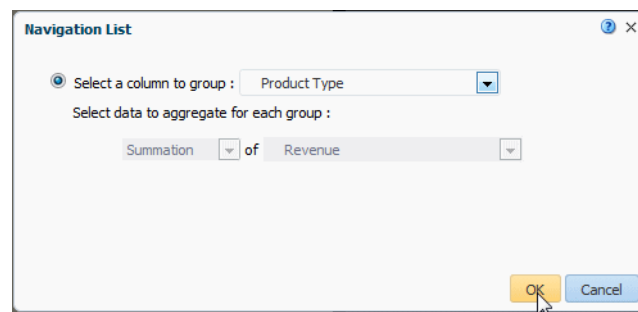
Figure 3–46 First Level Navigation List



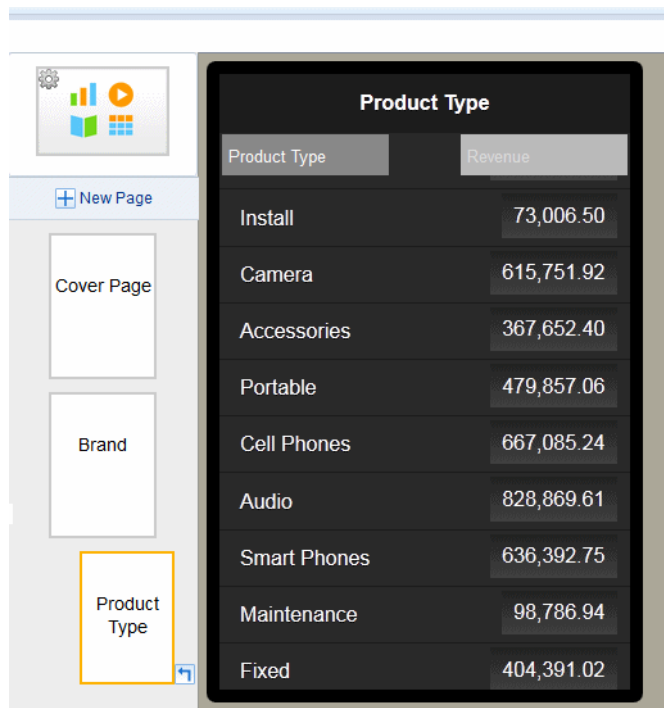
- To add a second level to the navigation list, create a subpage under the first navigation list page as shown in [Figure 3–47](#).

Figure 3–47 Inserting Navigation List Subpage

5. Select the data field to group for the second level of the navigation list. In this example (Figure 3–48), Product Type is the second level. Every child level uses the same aggregation selected for the first level. In the child levels the aggregation selection is for display only and cannot be updated.

Figure 3–48 Creating the Second-Level Navigation List

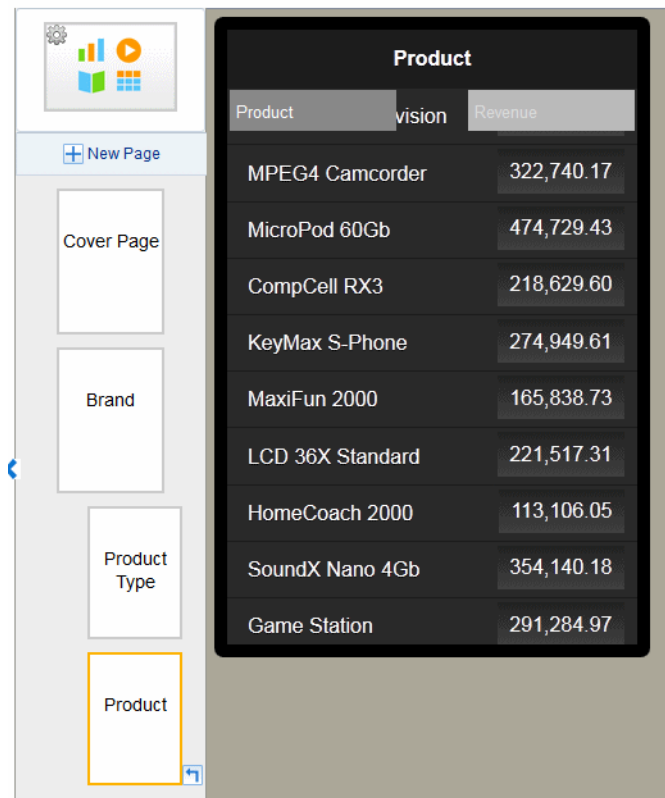
When you click OK, your subpage displays the members of the group you selected as shown in Figure 3–49.

Figure 3–49 Second Level of Navigation Page

The screenshot shows a software interface with a sidebar on the left and a main content area on the right. The sidebar contains a 'New Page' button and three subpage options: 'Cover Page', 'Brand', and 'Product Type'. The 'Product Type' subpage is highlighted with a yellow border. The main content area displays a table titled 'Product Type' with two columns: 'Product Type' and 'Revenue'.

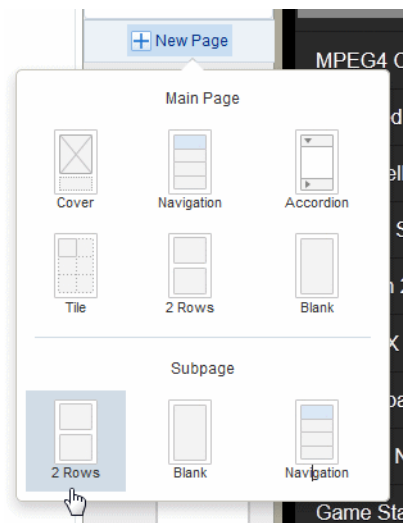
Product Type	Revenue
Install	73,006.50
Camera	615,751.92
Accessories	367,652.40
Portable	479,857.06
Cell Phones	667,085.24
Audio	828,869.61
Smart Phones	636,392.75
Maintenance	98,786.94
Fixed	404,391.02

6. To create another level, add a Navigation type subpage under the subpage you just created. Select the group element for this page as described in the previous step. [Figure 3–50](#) shows the navigation page created for the Product field.

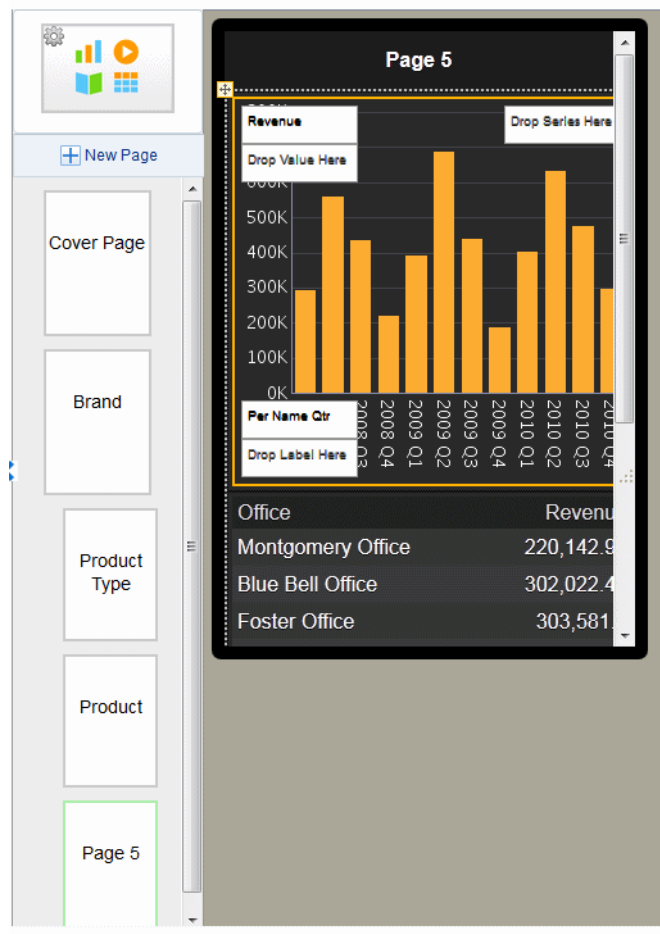
Figure 3–50 *Third level navigation page*

You can continue adding subpages to create as many levels as your app requires.

7. To add a detail page with results driven by the selections on the previous pages, add a new subpage beneath the final navigation page.

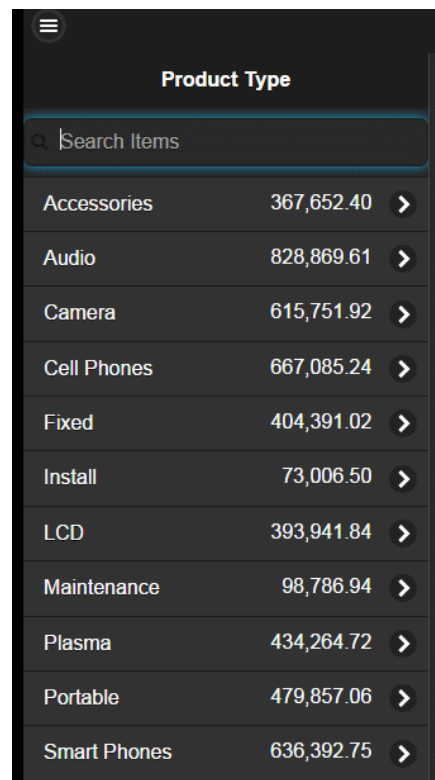
Figure 3–51 *Adding a Detail Page to Navigation Series*

8. Add components to the detail page. The following example shows a detail page with a chart and table.

Figure 3–52 Final Detail Page in Navigation Series

3.10.2.1 Adding a Search Filter

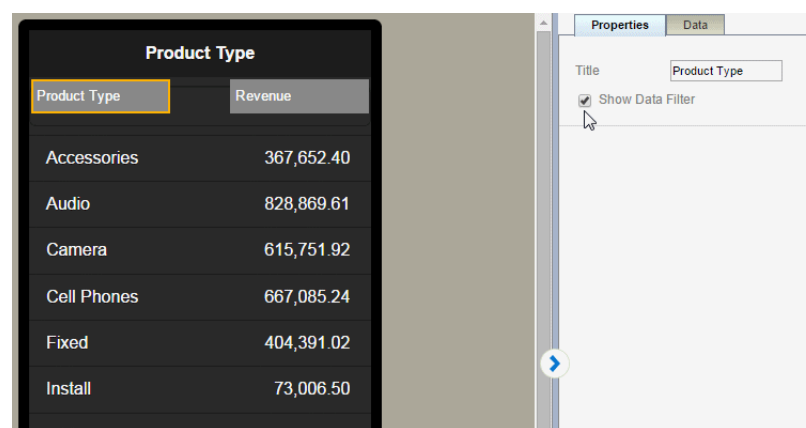
If a level in your navigation list includes many members (for example, States or Products) you can add a search filter to enable your app users to more easily find a specific item.

Figure 3–53 Navigation List with Search Filter Enabled

You can enable the search filter independently for each level.

To add a search filter to a navigation list level:

1. Select the navigation item for which you want to add a search filter.
2. In the Properties pane, enable the **Show Data Filter** property.

Figure 3–54 Adding a Search Filter to the Product Type Level in the Navigation List

3.10.3 Stoplight Formatting for Navigation and Accordion Pages

The Stoplight feature enables you to highlight values conditionally based on static threshold values, a comparison to a target value, or a comparison to another field.

- [Apply Formatting Based on Static Values](#)

- [Apply Formatting Based on Percent Achievement](#)
- [Apply Formatting Based on Percent Growth](#)

See also:

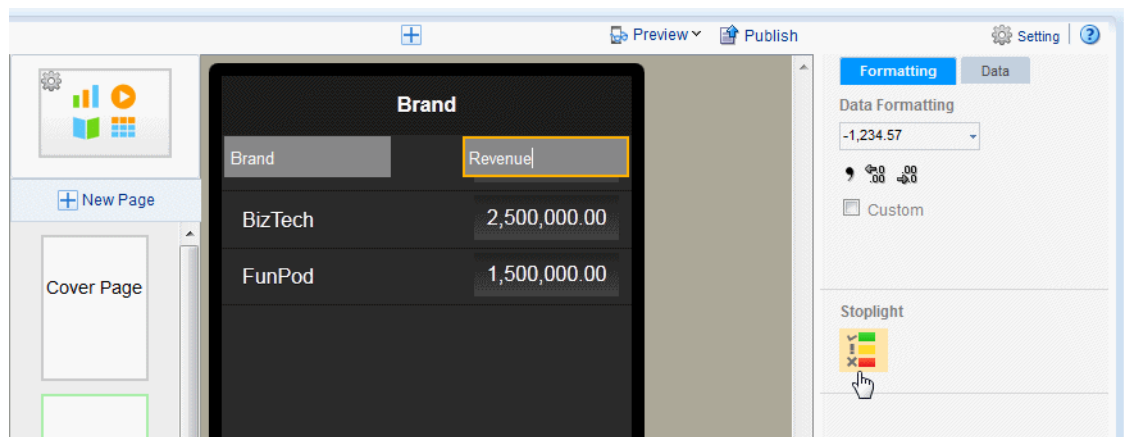
Video: Formatting Navigation Lists

3.10.3.1 Apply Formatting Based on Static Values

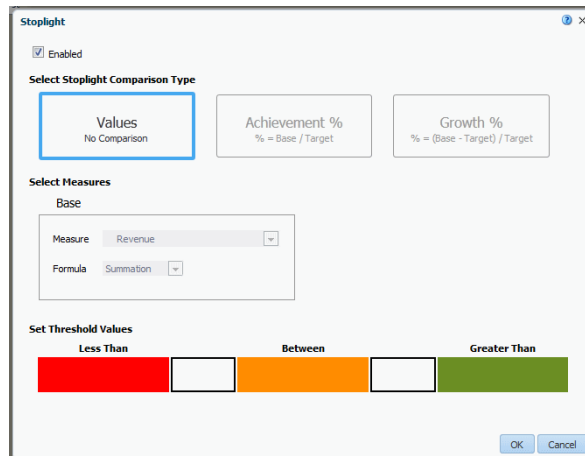
Use this option when the unacceptable, acceptable, and desired values are the same for the aggregated field at each level of the navigation. You can customize the colors for each range.

To apply conditional formatting based on static values:

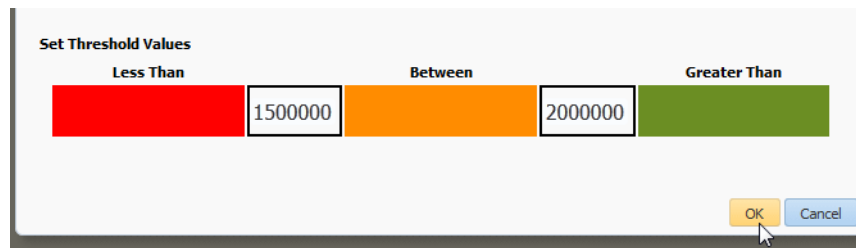
1. On the navigation or accordion page, select the measure field and click **Stoplight**. In this example the measure field is Revenue.



2. In the Stoplight dialog select **Values**. The Base Measure and Formula that you chose when you defined the navigation or accordion page are displayed, but are not editable.



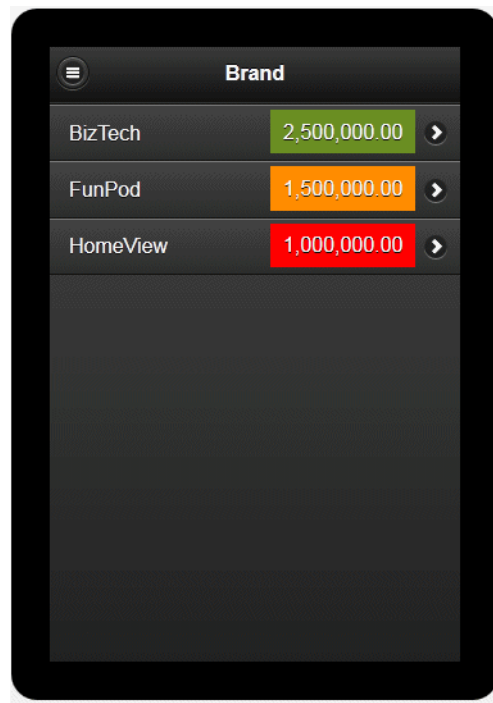
3. Enter the threshold values for the background colors. Values less than the left entry will display the **Less Than** background color. Values greater than the right entry will display the **Greater Than** background color. Values between display the **Between** color.



The dialog box titled "Set Threshold Values" contains three sections: "Less Than" with a red background, "Between" with an orange background, and "Greater Than" with a green background. The "Less Than" section has a text input field containing "1500000". The "Between" section has a text input field containing "2000000". At the bottom right, there are "OK" and "Cancel" buttons. A mouse cursor is pointing at the "OK" button.

To change the default colors, see [Customizing the Colors](#).

4. Preview your app.

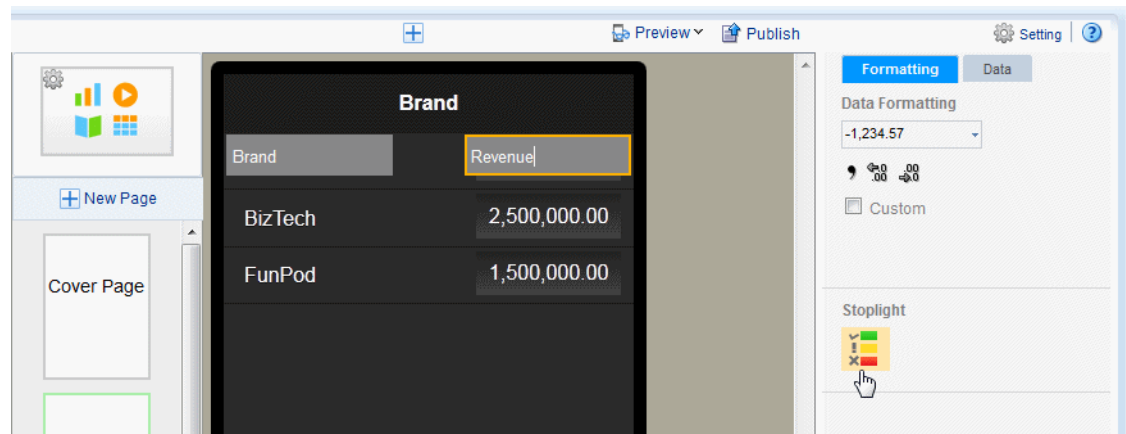


3.10.3.2 Apply Formatting Based on Percent Achievement

This option enables you to conditionally highlight the aggregate value based on the percentage of a target value. For example, you want to highlight sales that are less than 50 percent of target to display as red, sales that are 50-75 percent of the target as amber and sales that are greater than 75 percent of the target to display as green.

To apply conditional formatting based on the percent achievement of a target:

1. On the navigation or accordion page, select the aggregation field and click **Stoplight**. In this example the aggregation field is Revenue.



2. In the Stoplight dialog select **Achievement %**. The Base Measure and Formula that you chose when you defined the navigation or accordion page are displayed, but are not editable.

3. Choose **Target Measure** or **Value**.
 - **Measure** - select a field from the data to supply the target value for comparison.
 - **Value** - enter a static value to supply the target value for comparison.

The app calculates what percentage the Base column value is of the Target value. In the example above, the Revenue column (Base) is compared to the Target Revenue column (Target). The calculation performed is:

$$\text{Revenue} / \text{Target Revenue} \times 100\%$$

Therefore if your Revenue is \$8,000 and your Target Revenue is \$10,000, the percent achievement is:

$$8,000 / 10,000 \times 100\% = 80\%$$

4. Enter the **Less Than** and **Greater Than** values percentage values for the background colors.

If the Base value percentage of the Target value is less than the percentage you enter on the left, the **Less Than** color displays. If the Base value percentage of the Target value is greater than the value you enter on the right, the **Greater Than** color displays.

5. To change the default colors, see [Customizing the Colors](#).

3.10.3.3 Apply Formatting Based on Percent Growth

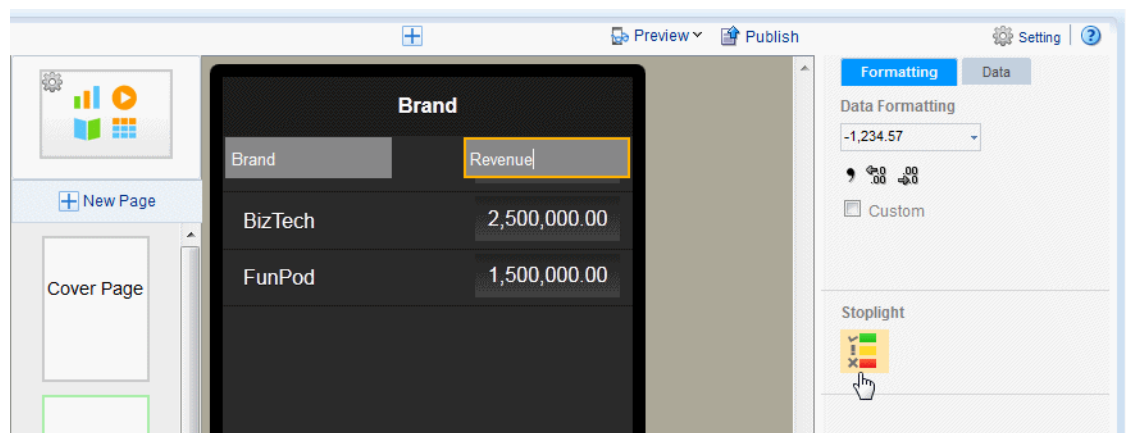
This option enables you to conditionally highlight the aggregate value based on the percent difference that the base value is from the target value. The calculation applied is:

$$(\text{Base} - \text{Target}) / \text{Target} \times 100\%$$

For example, you want to compare sales from the current and previous quarters and you want to see when sales in the current quarter showed less than 10% growth from the previous quarter. When sales showed greater than 10% growth you want to display the value in green, when 0-10% growth in amber, and when less than 0% growth in red.

To apply conditional formatting based on percent growth:

1. On the navigation or accordion page, select the measure field and click **Stoplight**. In this example the measure field is Revenue.



- In the Stoplight dialog select **Growth %**. The Base Measure and Formula that you chose when you defined the navigation or accordion page are displayed, but are not editable.

Stoplight

☒ Enabled

Select Stoplight Comparison Type

Values
No Comparison

Achievement %
% = Base / Target

Growth %
% = (Base - Target) / Target

Select Measures

Base

Measure: Revenue

Formula: Summation

VS

Target ☒ Measure ☐ Value

Measure: Quarter Ago Revenue

Formula: Summation

Less Than % Between % Greater Than

OK Cancel

- Choose **Target Measure** or **Value**.
 - Measure** - select a field from the data to supply the target value for comparison.
 - Value** - enter a static value to supply the target value for comparison.

The app calculates the percent difference that the Base column value is from the Target value. In the example above, the Revenue column (Base) is compared to the Quarter Ago Revenue column (Target). The calculation performed is:

$$((\text{Revenue} - \text{Quarter Ago Revenue}) / \text{Quarter Ago Revenue}) \times 100\%$$

Therefore if your Revenue is \$11,000 and your Quarter Ago Revenue is \$10,000, the percent growth is:

$$((11,000 - 10,000) / 10,000) \times 100\% = 10\%$$

- Enter the Less Than and Greater Than values percentage values for the background colors.

Select Measures

Base

Measure: Revenue

Formula: Summation

VS

Target: ☒ Measure ☐ Value

Measure: Quarter Ago Revenue

Formula: Summation

Set Threshold Values

Less Than: 0% (Red bar)

Between: 10% (Orange bar)

Greater Than: (Green bar)

OK Cancel

If the Base value percentage difference from Target value is less than the percentage you enter on the left, the **Less Than** color displays. If the Base value percentage difference from the Target value is greater than the value you enter on the right, the **Greater Than** color displays.

5. To change the default colors, see [Customizing the Colors](#).

3.10.3.4 Customizing the Colors

When the default colors do not suit the needs of your app, you can customize them to any color you require. For example, in some apps, the Less Than values should display as green and the Greater Than values as red; or, you may choose to display a different color scheme altogether.

To customize a color:

- Click the color bar to open the color picker and choose the color desired.

Figure 3–55 Changing the Default Color

Stoplight

☒ Enabled

Select Stoplight Comparison Type

Values: No Comparison

Growth %: $\% = (\text{Base} - \text{Target}) / \text{Target}$

Select Measures

Base

Measure: Revenue

Formula: Summation

Set Threshold Values

Less Than: 1500 (Red bar)

Greater Than: 000 (Green bar)

Color Picker

Choose a color: ff8c00

☒ Last Used Color

☐ Default Color

Custom Color...

OK Cancel

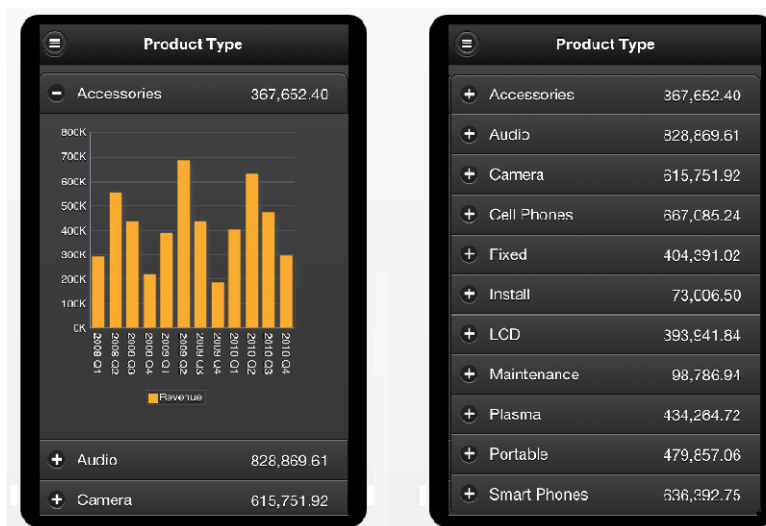
3.11 Adding Accordion Pages

The accordion page provides an expandable display of a specific data dimension and a key measure (such as Product and Revenue). Add visualizations to the expansion area that you can view and interact with easily from a mobile phone.

The Accordion page is available for phone apps.

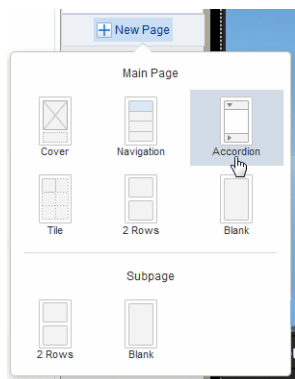
The accordion component is available for insertion to a tablet page.

Figure 3–56 Sample Accordion Page: Expanded and Collapsed

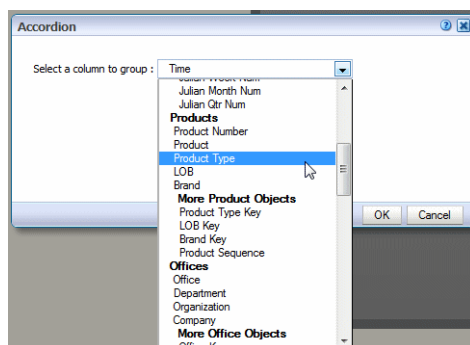


To create an accordion page:

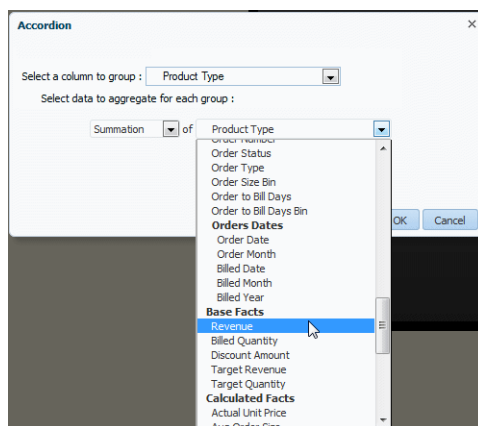
1. Click **New Page** and then click **Accordion**.



2. Select the data column to define each section of the accordion. In this example, an accordion section is created for each Product Type.

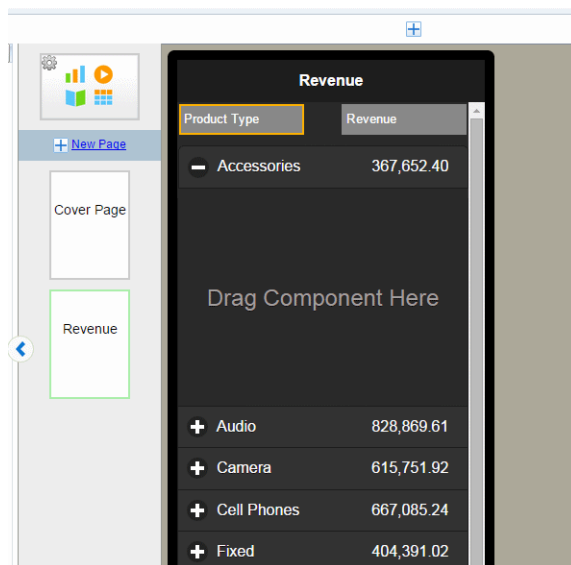


- Next select the measure field to aggregate for each section and select the aggregation type: Summation, Count, or Count Distinct. In this example, Revenue is summed.



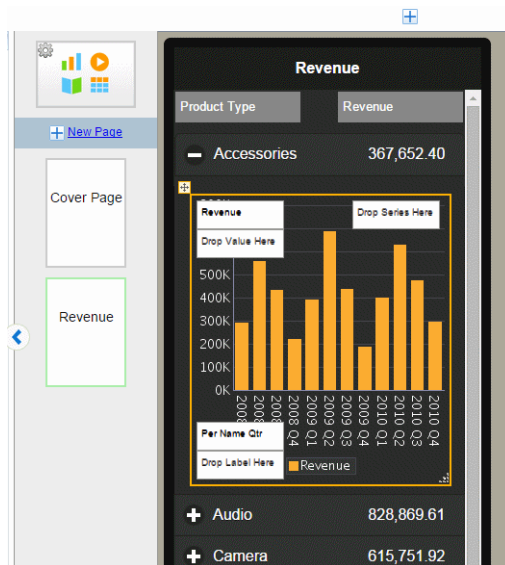
More formula types are supported for the measure field, see [Section 5.6, "Features of Metric Fields."](#)

- Click OK to insert the accordion.



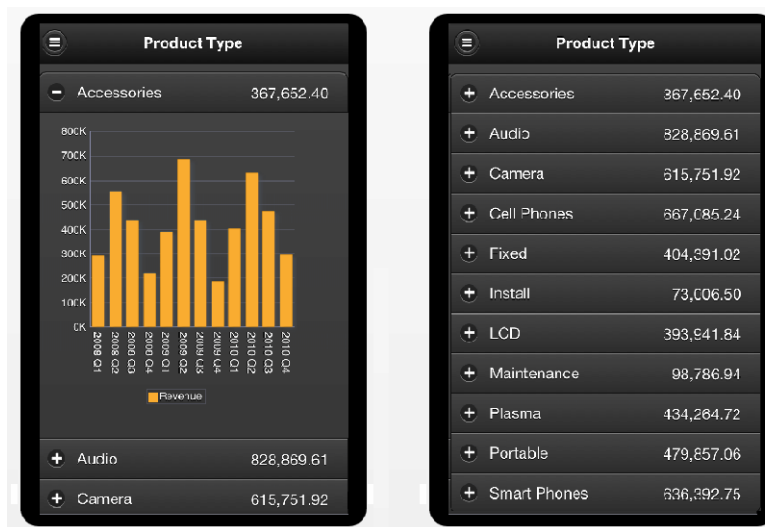
- Now you can insert the components you want to display when each section is expanded. Simply drag the component to the **Drag Component Here** area of the first expanded accordion section and follow the procedures for inserting the

specific component. To design your accordion page you only insert components in the first accordion section. In this example a chart showing revenue by quarter is inserted.



- To preview your accordion page, click Preview. The example displays in the Preview page as shown in the figure. Expand and collapse the accordion sections to see the data for each section.

Figure 3–57 Accordion Preview



3.11.1 Adding Stoplight Formatting to the Accordion Page

See [Stoplight Formatting for Navigation and Accordion Pages](#) for steps on adding stoplight formatting to your accordion page.

3.12 Inserting Tables

A default table includes a header, data columns, and a total row. The table supports "group left" functionality (outlines) that merges fields with the same values as well as subtotals, grand totals, and custom calculations.

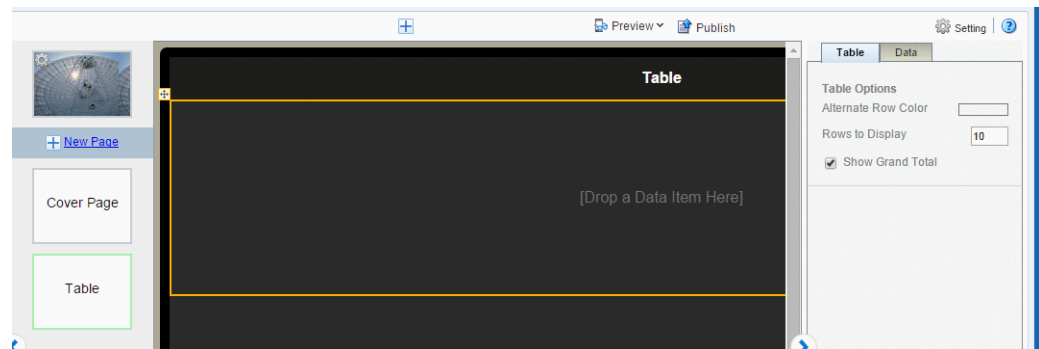
For detailed information about table options see [Section 5.1, "Working with Tables."](#)

To insert a table:

1. Click **Insert**, and then drag the **Table** component to the page.

[Figure 3–58](#) shows an inserted, empty data table. Notice that the **Table** properties now display.

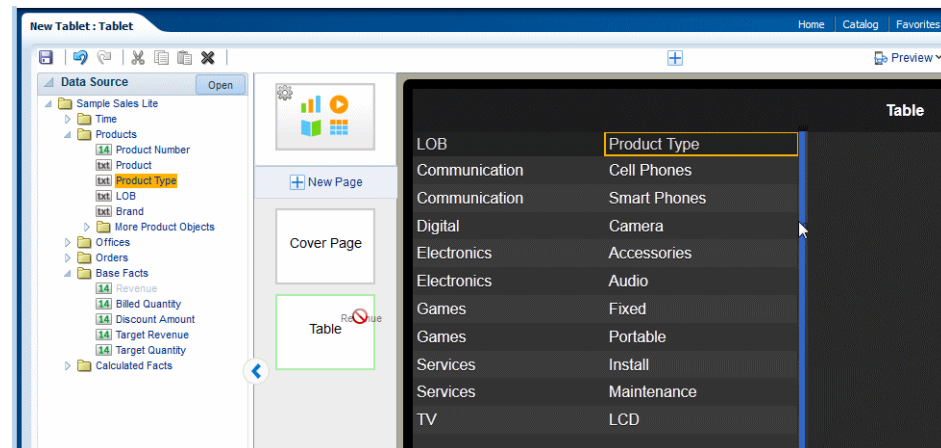
Figure 3–58 Example of an Inserted, Empty Data Table



2. To add data columns to the table, select a field from the **Data Source** pane and drag it to the table.

[Figure 3–59](#) shows adding columns to the table. Notice that when you drop a field on the table the sample data immediately displays.

Figure 3–59 Example of Columns Added to a Table

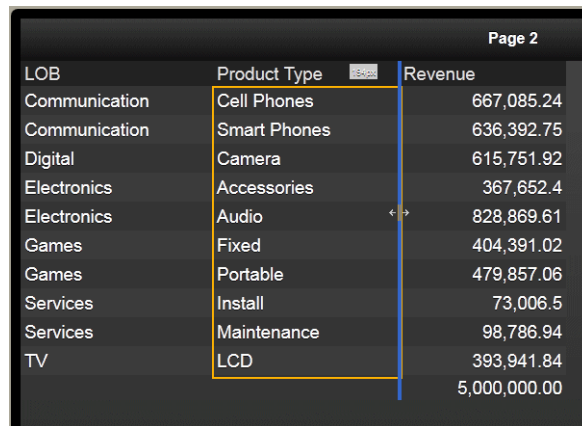


3. Continue to drag the fields from the **Data Source** pane to form the columns of the table. If you must reposition a column that you have already added, select it and drag it to the correct position.

To resize columns, position the cursor over the column border until the cursor switches to a handler, then drag the column border to the desired width. Notice

that as you drag the column edge the width in pixels displays to enable precise sizing.

Figure 3–60 Resizing Table Columns



LOB	Product Type	Revenue
Communication	Cell Phones	667,085.24
Communication	Smart Phones	636,392.75
Digital	Camera	615,751.92
Electronics	Accessories	367,652.4
Electronics	Audio	828,869.61
Games	Fixed	404,391.02
Games	Portable	479,857.06
Services	Install	73,006.5
Services	Maintenance	98,786.94
TV	LCD	393,941.84
		5,000,000.00

Some formatting is applied to the table, specifically:

- The inserted table includes a total row that displays the sum of the items in numeric columns. You can remove this row or edit the display and calculation applied. See [Section 5.1.5, "Customizing Table Totals."](#)
- Numbers and dates display default formatting and alignment. To change the formatting, see [Section 5.1.4.3, "Setting Table Data Formatting Options."](#)

3.12.1 Customizing Tables

Edit the table properties using the **Properties** pane. As you select different areas of the table, notice that the Properties pane changes to display specific for properties for the area selected. The following palettes are available to customize the display of your table. For details see [Section 5.1, "Working with Tables."](#)

- Table
- Table Column Header
- Column
- Total Cell

3.13 Inserting Pivot Tables

A pivot table provides views of multidimensional data in tabular form. It supports multiple measures and dimensions and subtotals at all levels. [Figure 3–61](#) shows a pivot table.

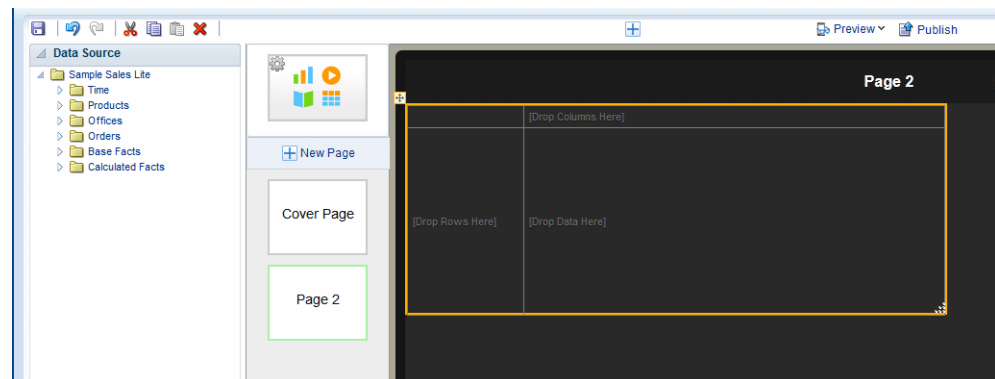
Figure 3–61 A Pivot Table

		BizTech	FunPod	HomeView	Services	TV	Total
2008		658,691.92	542,913.29	268,994.79	46,336.82	262,358.87	1,500,000.00
	2008 Q3	187,744.11	152,889.64	63,592.56	18,732.66	74,859.80	434,226.31
	2008 Q4	103,089.63	65,169.09	30,965.58	4,201.59	26,663.99	219,143.30
	2008 Q1	125,295.34	114,556.80	50,933.48	3,689.29	47,244.19	290,785.62
	2008 Q2	242,582.84	189,978.76	123,303.17	19,712.28	103,590.89	565,844.77
2009		821,826.00	556,965.90	321,508.40	61,056.79	260,448.81	1,700,000.00
2010		1,019,482.08	400,721.11	379,796.81	64,397.83	315,398.88	1,800,000.00
Total		2,500,000.00	1,500,000.00	1,000,000.00	171,793.44	828,206.56	5,000,000.00

3.13.1 Inserting a Pivot Table

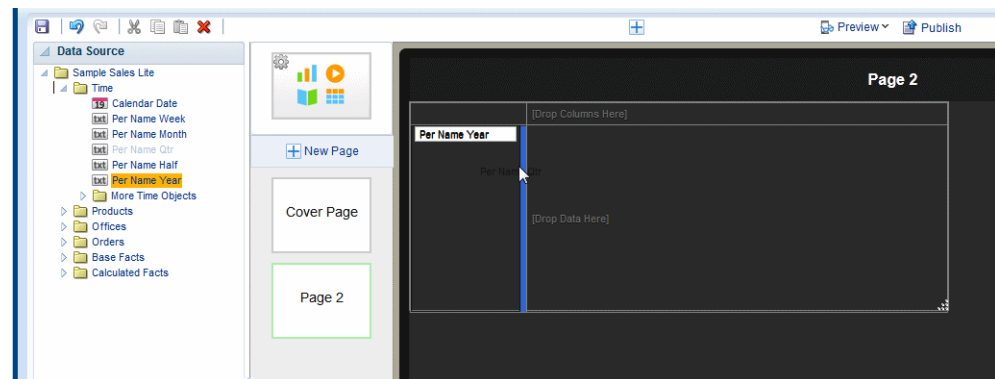
To insert a pivot table:

1. On the **Insert** tab, drag the **Pivot** component to the page. [Figure 3–62](#) shows the empty pivot table structure.

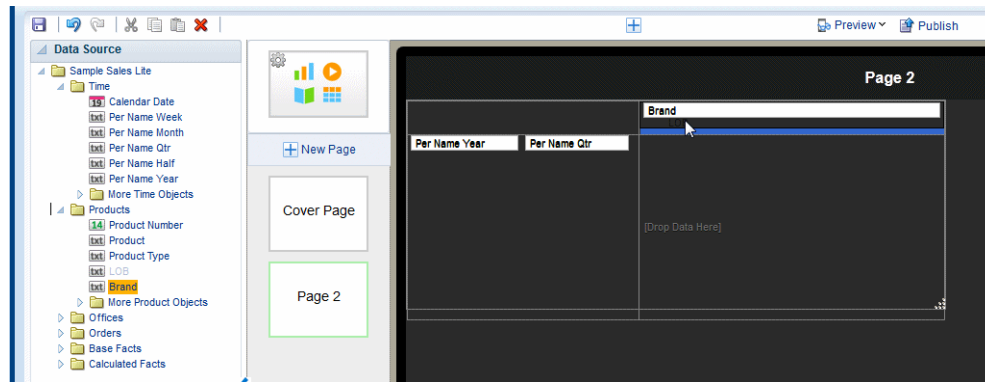
Figure 3–62 Inserted Pivot Table Structure

2. Drag and drop data fields from the Data Source pane to the row, column, and data positions.

To create multiple dimensions, precisely drop each level to its position in the pivot table structure as shown in [Figure 3–63](#).

Figure 3–63 Dragging Data Fields to the Pivot Table

You can stack multiple dimensions in both rows and columns.



3. Optionally resize the pivot table by clicking and dragging the handler in the lower right corner of the pivot table, as shown in [Figure 3–64](#).

Figure 3–64 Resizing Pivot Table

139.16	81,604.95
754.69	48,334.94
314.61	58,980.73
300.08	121,262.76
806.92	382,019.08

For more details about pivot table options, see [Section 5.2, "Working with Pivot Tables."](#)

3.14 Inserting Charts

The Mobile App Designer supports a variety of chart types and styles to present rich visualizations of your data.

After you insert a chart, you can edit the chart properties using the Properties pane. The Properties pane enables you to enter very specific custom settings for the following:

- Chart Effect
- Chart Legend
- Chart Plot Area
- Chart Title
- Chart Label
- Chart Values

Select the appropriate component in the map to see the available properties.

Topics:

- [Inserting a Chart](#)
- [Changing the Formula Applied to a Chart Measure Field](#)
- [Changing the Series Color in a Bar or Line Chart](#)

- Changing the Width and Style of the Lines in a Line Chart
- Setting the Treatment for Null Values

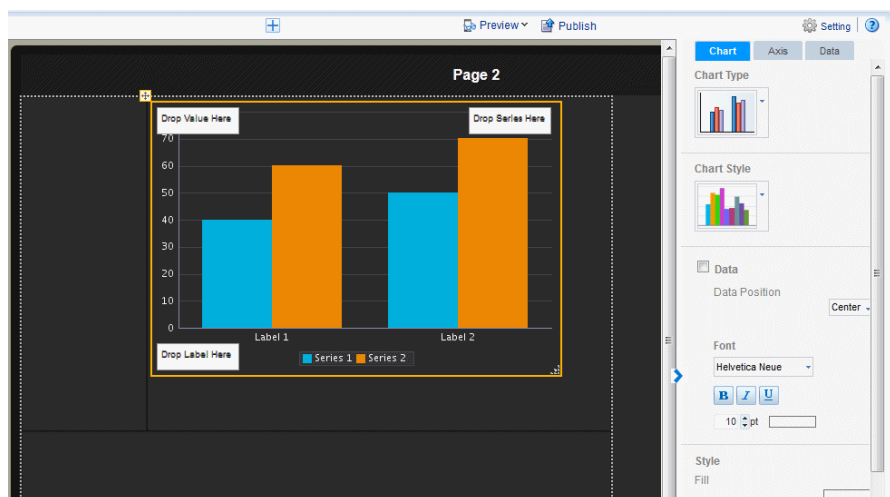
3.14.1 Inserting a Chart

To insert a chart:

1. From the **Insert** menu, drag the Chart component to the page.

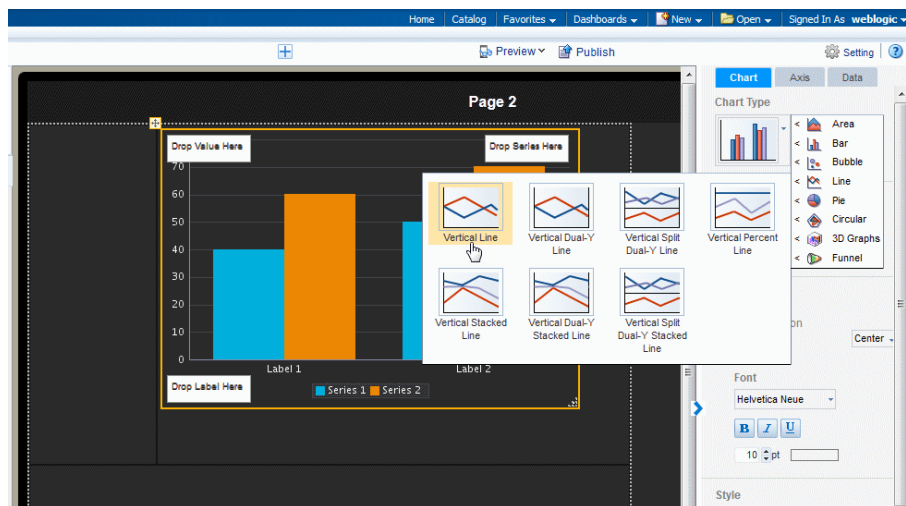
By default a vertical bar chart is inserted and the **Chart** properties are displayed, as shown in [Figure 3–65](#).

Figure 3–65 Default Chart and Chart Properties



2. To change the chart type, click the **Chart Type** list to select the chart type required for your app. [Figure 3–66](#) shows changing the chart type to Pie.

Figure 3–66 Changing the Chart Type



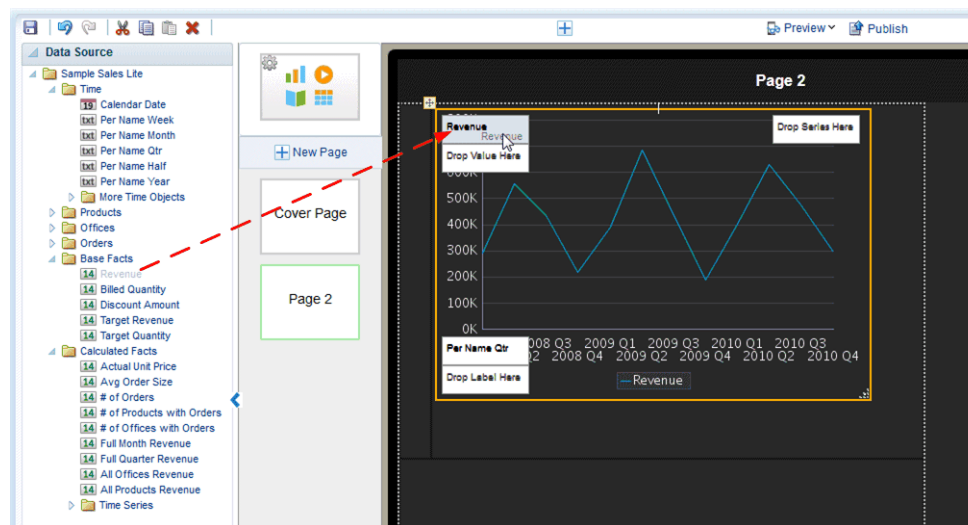
3. Drag the data fields from the Data Source to the appropriate areas in the chart. Where you drag a data element depends on the chart type and on the information you want to display.

For example, a vertical bar chart includes the following options:

- Label
- Value
- Series

The chart immediately updates with the preview data, as shown in [Figure 3–67](#)

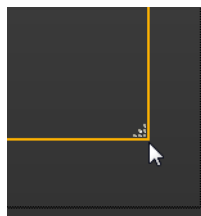
Figure 3–67 Dragging Data Fields to a Chart



4. To resize the chart, drag and drop the resize handler on the lower right corner of the chart, as shown in [Figure 3–68](#).

To preserve the aspect ratio when resizing a chart, press and hold the Shift key before dragging the handler.

Figure 3–68 Chart Resize Handler



3.14.2 Changing the Formula Applied to a Chart Measure Field

By default, the chart displays a sum of the values of the chart measure. You can change the formula applied to a chart measure field using the **Measure** properties.

To change the chart measure field formula:

1. Select the measure field in the chart. This displays the **Measure** properties as shown in [Figure 3–69](#).

Figure 3–69 Changing the Chart Measure Field Formula

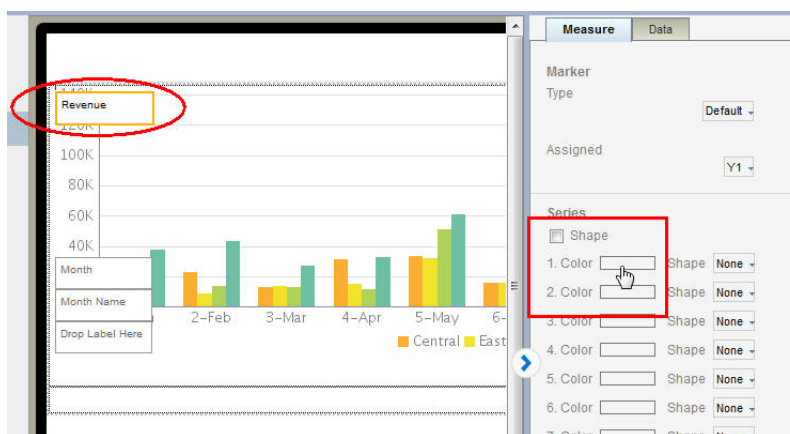
2. Click the **Data** tab and select a formula from the **Formula** list. To include a more complex formula, use a Calculated Field. See [Calculated Fields](#).

3.14.3 Changing the Series Color in a Bar or Line Chart

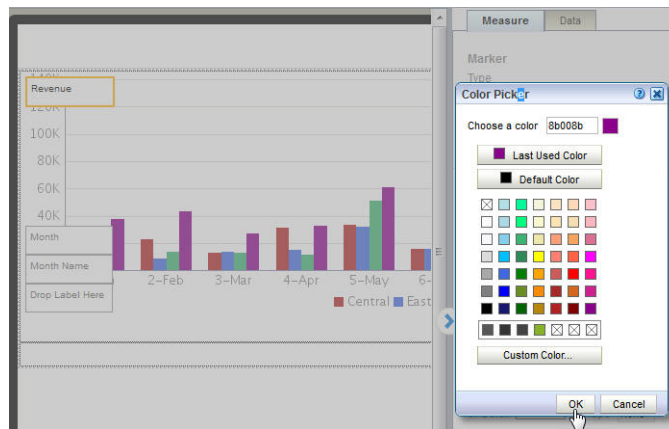
You can customize the appearance of your line or bar chart by updating the colors displayed.

To change the color of the lines or bars in a chart:

1. Select the measure field in the chart. This displays the **Measure** properties as shown in [Figure 3–70](#).

Figure 3–70 Changing the Color in a Bar Chart

2. Click the **Color** box for the series member that you want to customize.
3. Use the **Color Picker** to select the color you want displayed in the chart. Repeat for each series member.

Figure 3-71 Selecting New Bar Colors Using the Color Picker

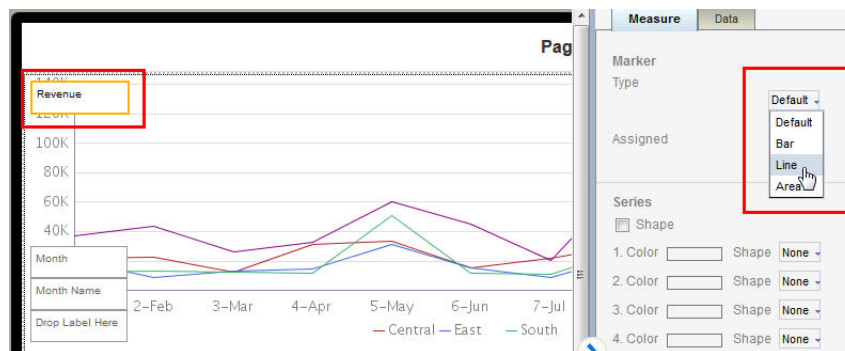
3.14.4 Changing the Width and Style of the Lines in a Line Chart

You can customize the appearance of your line chart by customizing the width and style of the lines.

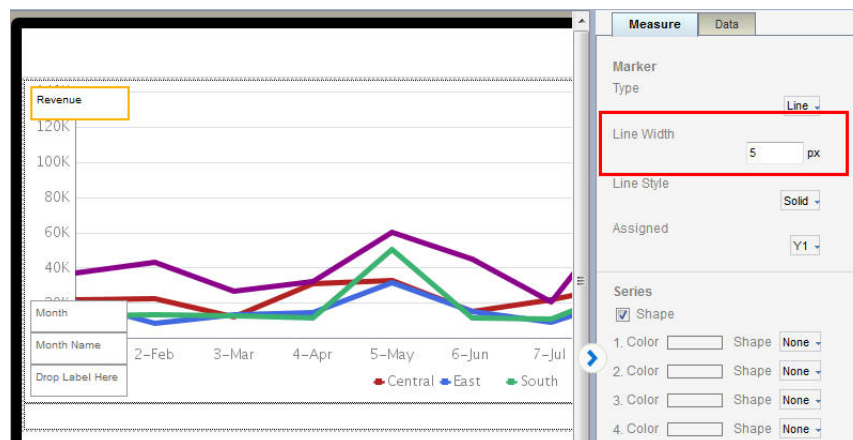
To change the width of the lines in a line chart:

1. Select the measure field in the chart. This displays the **Measure** properties.

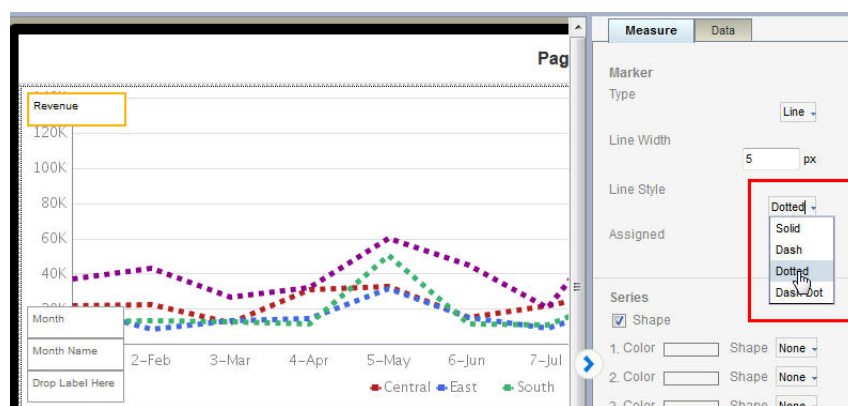
From the Type list, select **Line** as shown in [Figure 3-72](#).

Figure 3-72 Selecting Line Type Chart

2. To customize the line width, update the pixel setting as shown in [Figure 3-73](#).

Figure 3–73 Setting the Line Width to 5 Pixels

3. To customize the Line Style, choose from Solid (default), Dash, Dotted, or Dot-Dash.

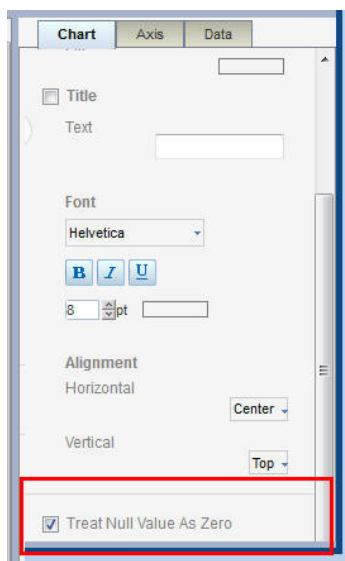
Figure 3–74 Selecting New Bar Colors Using the Color Picker

3.14.5 Setting the Treatment for Null Values

By default, null values are treated as a zero value. Depending on the nature of your data you may want to turn off this setting.

To turn off Treat Null Values as Zero:

1. Click within the chart to display the **Chart** properties tab.
2. Scroll to bottom of the Chart properties tab and deselect the **Treat Null Values as Zero**.



3.15 Inserting Maps

Use the map component to show how data is distributed across geographic locations.

Each map includes a preset region identifier that specifies how data is mapped to a region. For example, the region identifiers for the US States map are:

- the full state name (for example, "Florida")
- the two-letter state abbreviation (for example, "FL")

So if your data source identifies the state as "Florida" or "FL" the mapping engine correctly aggregates the data for Florida.

If your data does not include the region identifiers shown for the provided maps, you can edit the map files to meet your requirements.

The **Locations** map type uses latitude and longitude values to aggregate data for a location. To use the Locations map type, your data source must include latitude and longitude data values.

Mobile App Designer offers over fifty country and region maps that you can install and use. For the complete list and install instructions see the BI Mobile App Designer Maps Resource page at:

<http://www.oracle.com/technetwork/middleware/bi-foundation/map-code-2408077.html>

To use custom maps with custom identifiers, see [Section A.7, "Adding Custom Maps."](#)

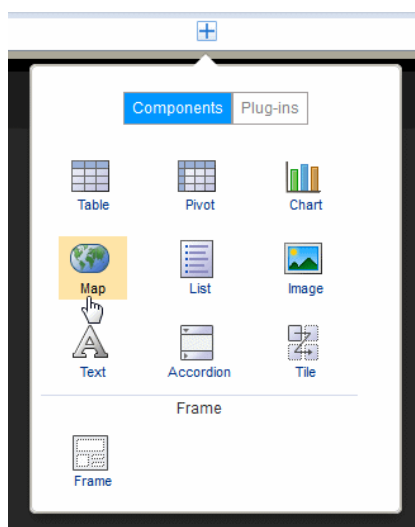
To help you get started using maps, watch the following videos:

- Video: Creating App Pages with Maps
- Video: Setting Up Locations Maps and Map Subpages

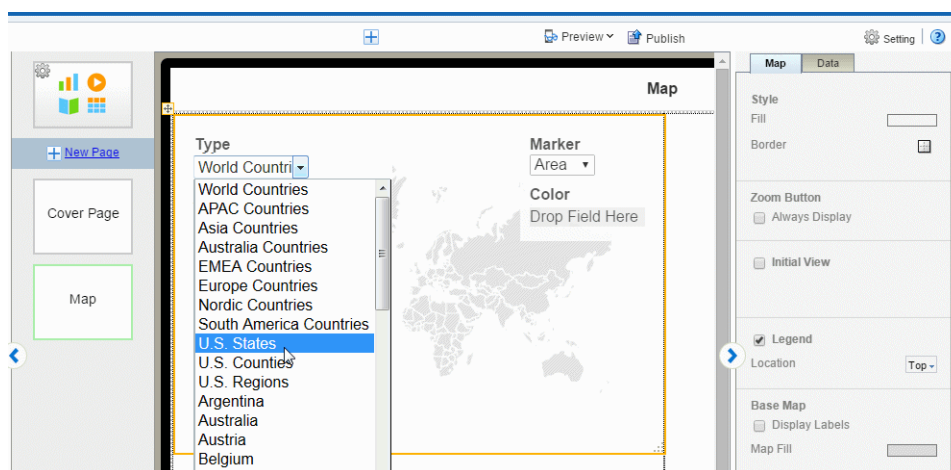
3.15.1 Inserting a Map

To insert a map on your app page:

1. Click **Insert Component**, and then drag **Map** to the area of the page you want to insert it.



2. Select the **Map Type**.

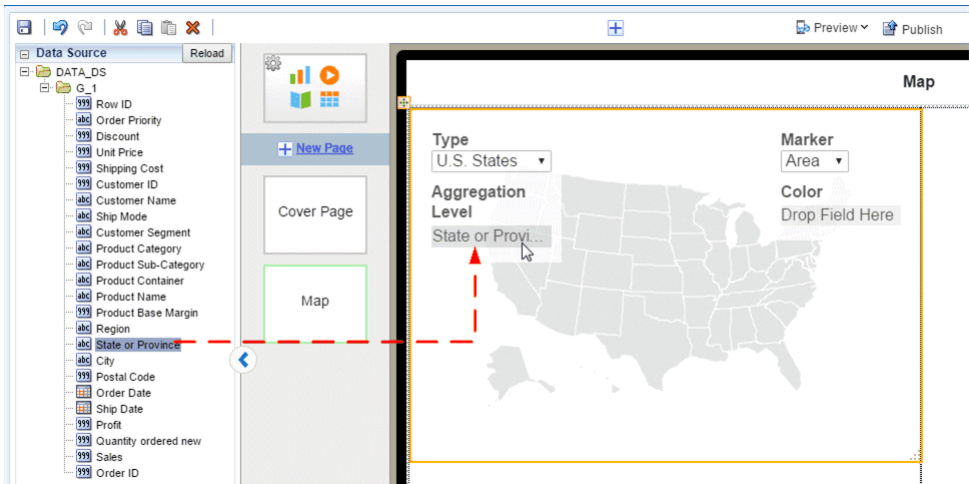


Mobile App Designer provides a set of maps. You may see more maps listed if your site has added custom maps.

3. Drag the data source field for which the data is to be aggregated to the **Aggregation Level**.

The **Aggregation Level** is the geographical entity for which data is aggregated. For example, in a United States map you would aggregate the data by state.

Figure 3–75 Assigning a Data Field to the Aggregation Level



The values in your data for this field must match a supported region identifier for the map. For example, if you choose the US States map, the data field you assign to Aggregation Level must contain either the full state names (such as "Wyoming") or the two-letter state abbreviations (such as "WY").

Some examples are shown in the following table:

Map Type	Aggregation Level	Identifiers for the Aggregation Level
World Countries	Country	Country Name, for example: "France" ISO3 three-letter country code, for example: "FRA" ISO2 two-letter country code, for example: "FR"
US States	State	Full state name, example: "Wyoming" Two-letter state abbreviation, example: "WY"
US Counties	County	Two-letter state abbreviation _county name, for example: "IL_COOK" and "CA_ORANGE"
Locations	N/A	Latitude and Longitude values, for example: "47N" and "122W"
All others	Varies	See http://www.oracle.com/technetwork/middleware/bi-foundation/map-code-2408077.html

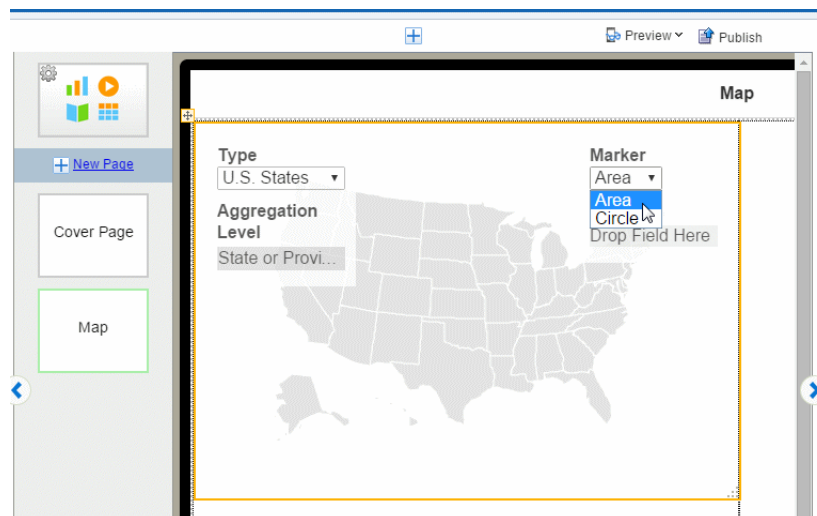
If your data does not include a supported identifier, you can edit the map files. See [Section A.7, "Adding Custom Maps."](#)

Longitude and Latitude (Locations Type Only)

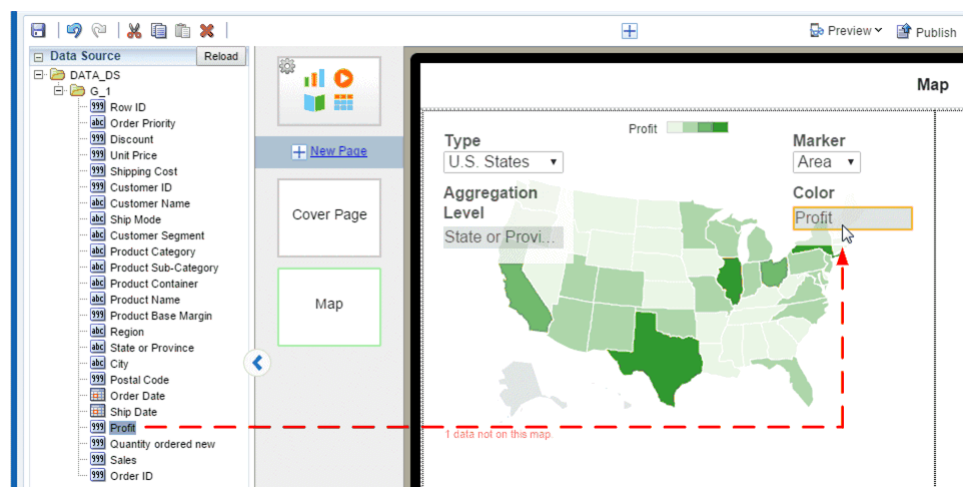
If you selected the Locations map Type then your data source must include longitude and latitude values. Drag the appropriate data fields to the Longitude and Latitude positions in the map component.

Selecting a Marker Type

The Marker specifies how the aggregated data displays on your map. You can choose either Area or Circle (the Locations Map Type supports the Circle marker only).

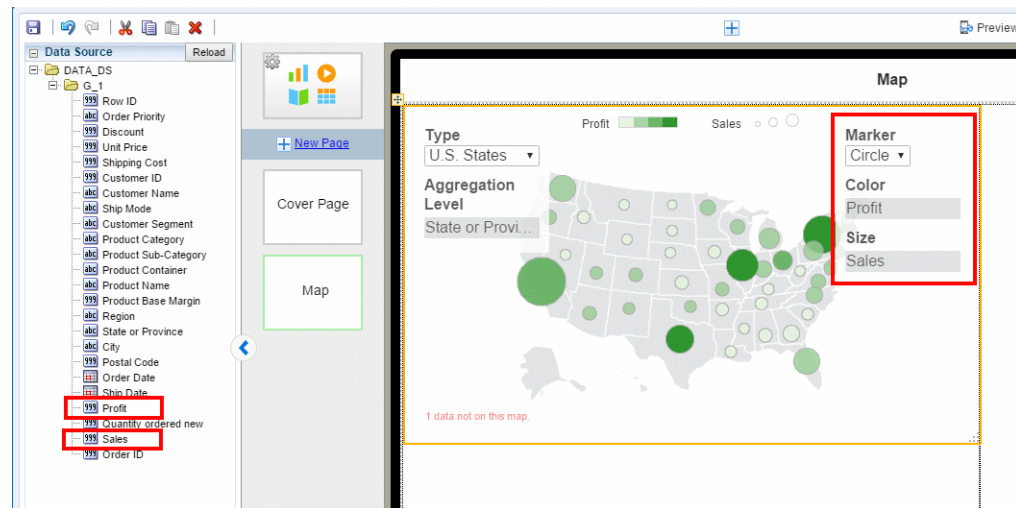
Figure 3–76 Selecting a Marker Type

Area - In Area maps, the color of the region signifies the aggregated value. Drop the data value to be aggregated in the **Color** field. In the following example, the color of each State signifies the aggregated Profit.

Figure 3–77 Assigning a Data Field to Area Marker Color

Circle - A circle displays at each location in the data that corresponds to the Aggregation Level. Circle markers can represent two dimensions by color and size.

In the following example, the color of the circle represents Profit while the size of the circle represents Sales.

Figure 3–78 Adding Values to the Color and Size Attributes for the Circle Marker Map

3.15.2 Setting Map Properties

Properties you can set for maps include:

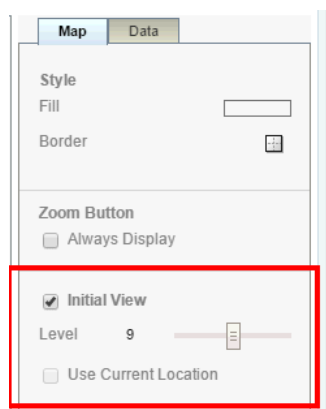
- [Setting the Initial Zoom Level and Map Center](#)
- [Customizing the Base Map Display](#)
- [Setting the Color Distribution of Circle or Area Markers](#)
- [Setting the Size Distribution of Circle Markers](#)

3.15.2.1 Setting the Initial Zoom Level and Map Center

You can configure how the map initially displays to users when the app opens by setting the zoom level and center location.

To configure the initial view settings:

1. Select the map to display the map properties.



2. Select **Initial View** then set the **Zoom** level on the slider.
3. To set the center location, drag the map within the designer display area until the map is centered as desired. This setting is saved.

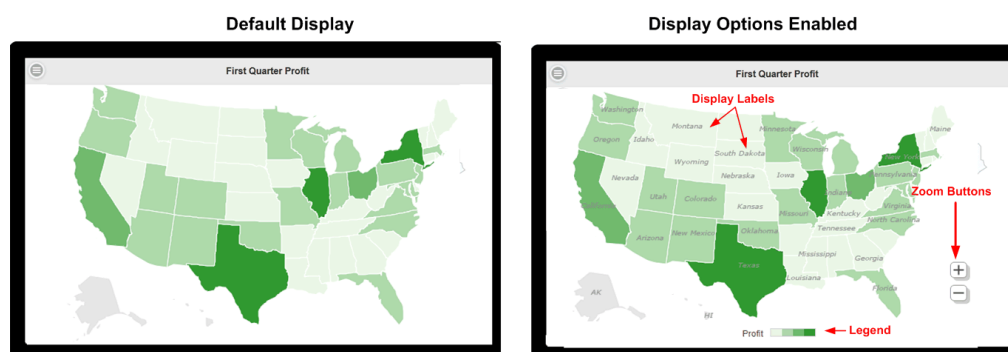
Use Current Location as Initial View

This option is enabled for Locations type maps only (maps that use latitude and longitude values). When you select **Use Current Location** as the **Initial View**, the app detects the user's current location from the browser and uses it as the center point of the map. For example, if you design a Sales app using latitude and longitude data, then when your app user gets off a plane in San Francisco and views the app, the map is centered in San Francisco. A different user of the same app getting off a plane in Chicago will see his map centered in Chicago.

3.15.2.2 Customizing the Base Map Display

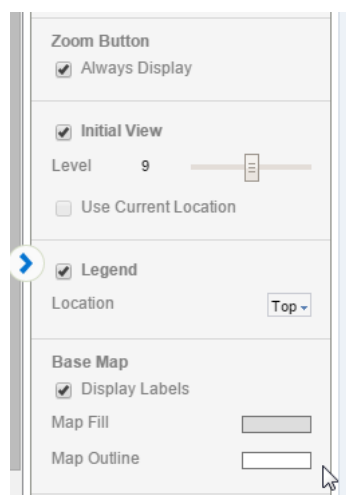
You can customize your map to display a legend, the zoom buttons, and geographical labels.

Figure 3–79 Some Map Display Options



To set display options:

1. Select the map to display the map properties.
2. To display the **Zoom Button**, select **Always Display**.



- **Legend** - displays the legend to show the value of the markers. When Area marker is used, the color scale displays. When Circle marker is used, circle size scale displays as well.

You can set the Location of the legend to be the Top or the Bottom of the map,

- **Display Labels** - displays the label (for example, Country Name or State Abbreviation) for each geographic element. The label used can be configured

in the map configuration file (see [Section A.7, "Adding Custom Maps"](#)).

3.15.2.3 Setting the Color Distribution of Circle or Area Markers

Use the color distribution properties to specify how the marker color is distributed across the data points.

Cut Point

The Cut Point sets the way the color distribution is calculated based on either an equal range of values or equal number of members.

Scale

By default the scale is Linear.

If your data points are mostly distributed near the maximum value, choose **Power** to enhance the color distinction between the higher numbers.

If your data points are mostly distributed near the minimum value, choose **Square Root** to enhance the color distinction between the lower members.

Scale cannot be set when the Cut Point is set to equal number of members.

Set 0 as Center Value

Sets 0 as the center point of the color distribution.

of Steps

Number of color steps between the lowest and highest values.

Color Type

When set to Sequential, choose the Start and End colors to display. The lowest value displays the Start color and the highest values displays the End color. Values in between display color between the two according to your selections for Cut Point, Scale, and Number of Steps.

When set to Diverging, choose Start, Center, and End colors. The Center color is the mid point of the data values. The higher values diverge from the Center toward the End color, the lower values diverge from the Center color toward the Start color.

Opacity

Set the slider to configure the opacity level.

3.15.2.4 Setting the Size Distribution of Circle Markers

You can configure how the circle size represents your data.

Scale

By default the scale is Linear.

If your data points are mostly distributed near the maximum value, choose **Power** to enhance the size distinction between the higher numbers.

If your data points are mostly distributed near the minimum value, choose **Square Root** to enhance the size distinction between the lower members.

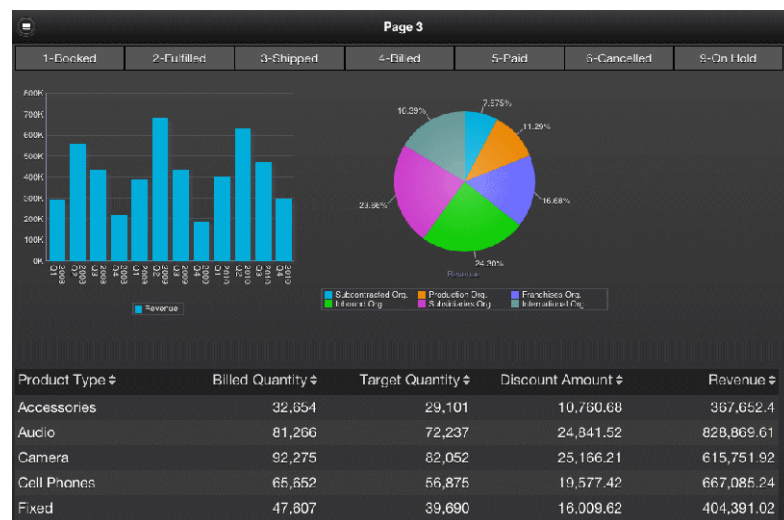
Min Size / Max Size

Manually set the start (lowest value) and end (highest value) size of the circles in pixels.

3.16 Inserting Lists

The List component displays all values of a data field in a vertical or horizontal interactive list that behaves as a filter for the other components on the page. Tap a list item to update the results in all other tables, charts, or other visualizations on the page. [Figure 3–80](#) shows a page that displays two charts and a table. The list component across the top of the page displays order status. Tapping an order status updates the other components to show only results for the item tapped.

Figure 3–80 Example of Horizontal List

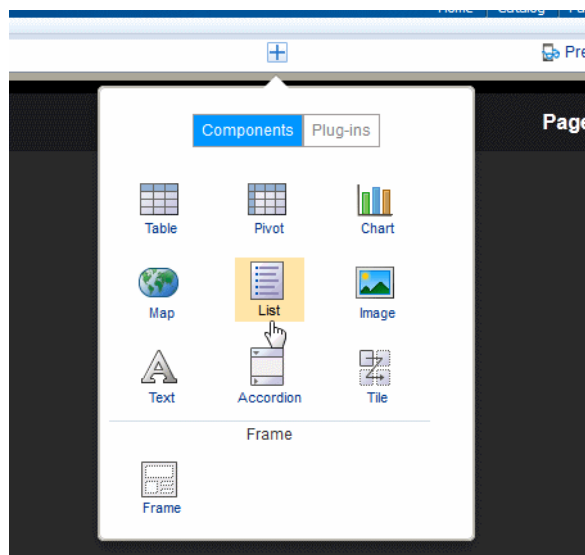


You can define default values for the list to be applied when a user first opens the page. This feature is useful when the unfiltered data set is large.

3.16.1 Inserting a List

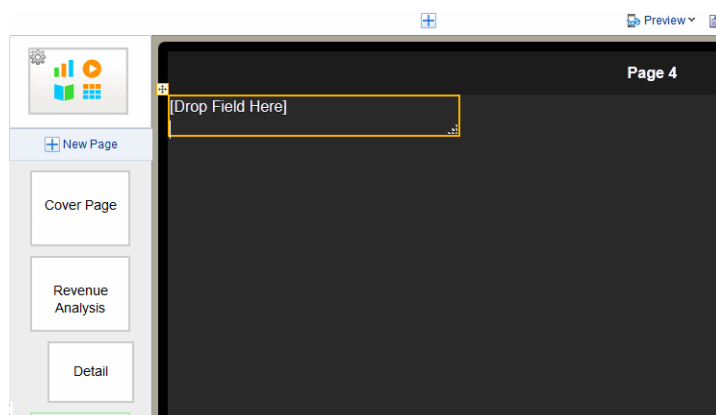
To insert a list:

1. Click **Insert Component** and then select **List**.



This inserts an empty list to the design area.

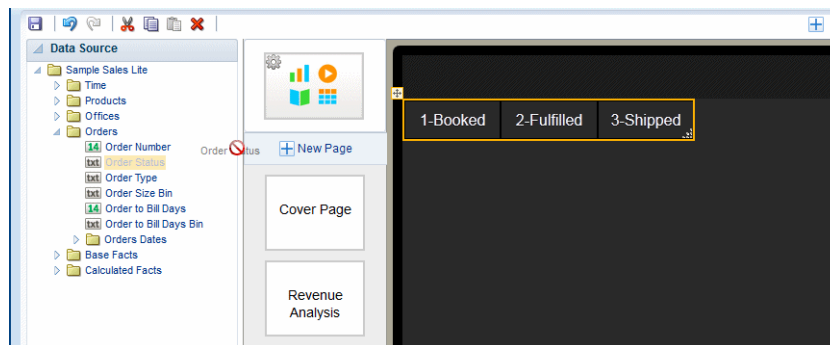
Figure 3–81 An Inserted List



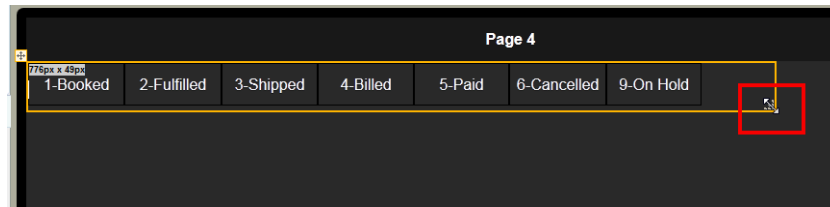
2. Drag an element from the **Data Source** pane and drop it on top of the empty list in the layout.

Figure 3–82 shows a list of Order Status values.

Figure 3–82 A List Showing Order Status



3. To change the size of the list in the page, click and drag the handler in the bottom right corner.



4. (Optional) Customize the appearance of the list and set defaults. See [Customizing Lists](#).

3.16.2 Customizing Lists

Use the List Properties set to:

- [Specify Default Values for the List](#)
- [Change List Orientation](#)
- [Customize Font and Background Styles](#)
- [Specify List Sort Order](#)

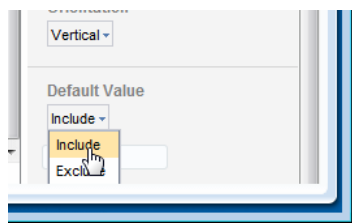
3.16.2.1 Specify Default Values for the List

You can configure your list to display only specific values when users first open the app page.

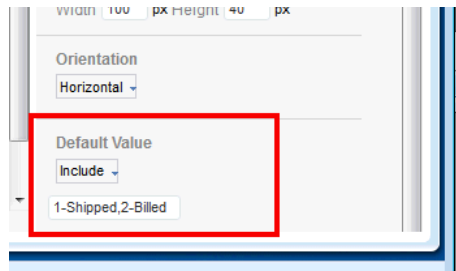
To specify default values:

1. Select the List to display its Properties.
2. In the **Default Value** property, choose whether to Include or Exclude the specified values.

Figure 3–83 *Choosing the Default Value Type*



3. Enter the default values. Enter multiple values separated by a comma. (**Important:** Do not enter spaces between the values.)

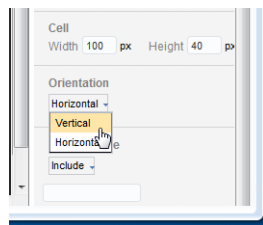
Figure 3–84 *Entering Default Values for a List*

3.16.2.2 Change List Orientation

The list can display either horizontally across the page or frame or vertically down the page. To change the orientation of the list:

1. Select the List component on the page to display its Properties.
2. Under **Orientation**, select Horizontal or Vertical.

[Figure 3–85](#) shows the Orientation property.

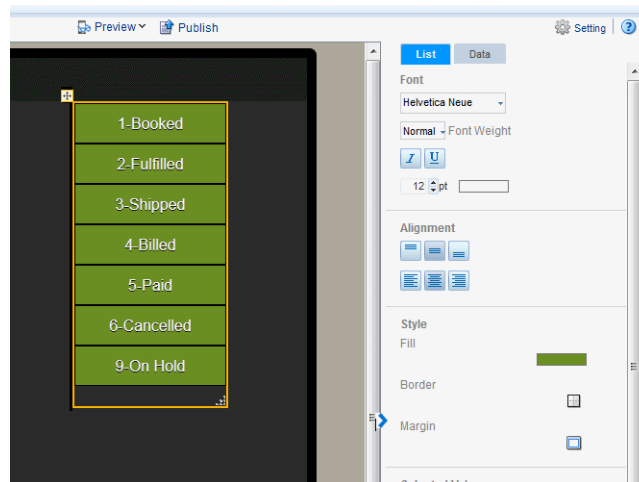
Figure 3–85 *Changing Filter Orientation to Vertical*

3.16.2.3 Customize Font and Background Styles

You can customize the list font and background styles for selected and nonselected modes.

To customize the font and background style when no list items are selected:

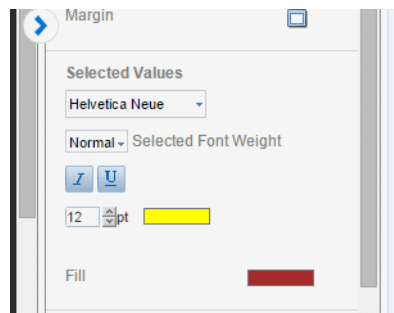
1. Select the list component in the design area to display its Properties.
2. Set the following properties in the Font region:
 - Font type and Font weight
 - Font Size
 - Border
 - Background Color
 - Font Color

Figure 3–86 Customized List Background Color

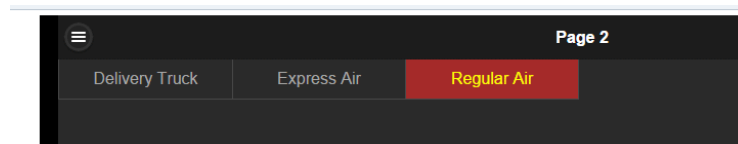
The **Selected Values** properties control the appearance of the item in a list when it is selected. By default, the selected item displays as amber.

To customize the font and background style of the selected item:

1. Select the list component in the design area to display its Properties.

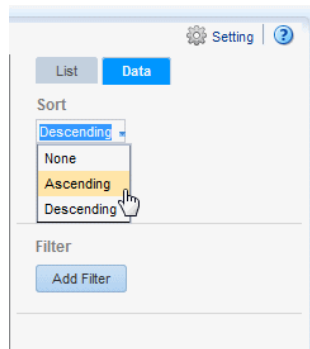


2. Set the following properties in the **Selected Values** region:
 - Font type, Weight, and Style
 - Font size
 - Font Color
 - Fill color when selected

Figure 3–87 Customized Selected Value Fill and Font Color

3.16.2.4 Specify List Sort Order

To specify the list sort order, click the **Data** tab on the **Properties** pane and choose Ascending, Descending, or None.

Figure 3–88 *Setting the List Sort Order*

3.17 Inserting Data Fields

You can insert a data field to display in your app by simply dragging it from the Data Source pane to the app page where you want it to display. To position data fields more precisely, or to insert fields side-by-side, insert a frame first. See [Inserting Frames](#).

3.17.1 Insert Data Fields

To insert a data field:

1. Select the area in the page where you want to insert the field.
2. Drag the field from the **Data Source** pane to the position in the app page.

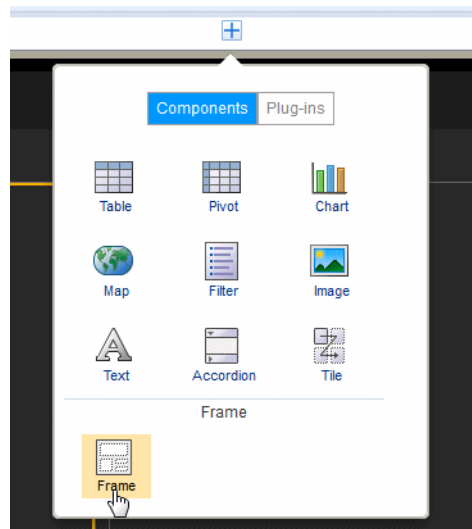
3.17.2 Display Data Fields Side-by-Side

To display data fields side-by-side as shown in [Figure 3–89](#), insert a Frame first to position the data fields.

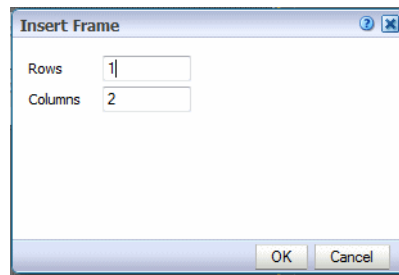
Figure 3–89 *Data Fields Displayed Side-by-Side*

Accessories	367,652.40	Audio	828,869.61
Fixed	404,391.02	Install	73,006.50
Plasma	434,264.72	Portable	479,857.06

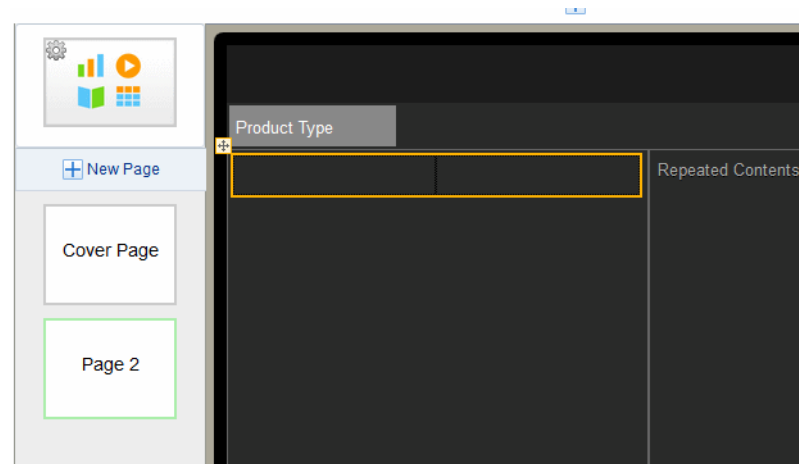
1. Select the area of the page to display the data fields. In this example, the area is a Tile.
2. On the **Add Component** menu, drag the **Frame** component to the desired position on the page (in this example, to the first tile).



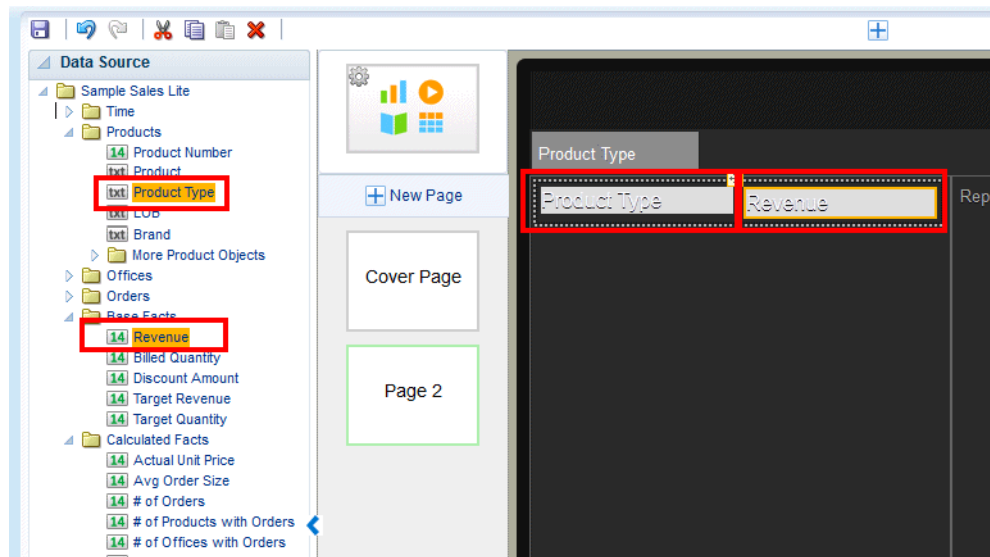
3. In the **Frame** dialog, enter 1 Row and 2 Columns.



The Frame displays in the page.



4. Drag the data fields from the Data Source pane to the frame cells where you want the data to display.



3.18 Inserting Text

Use the text component to enter free-form text in the layout.

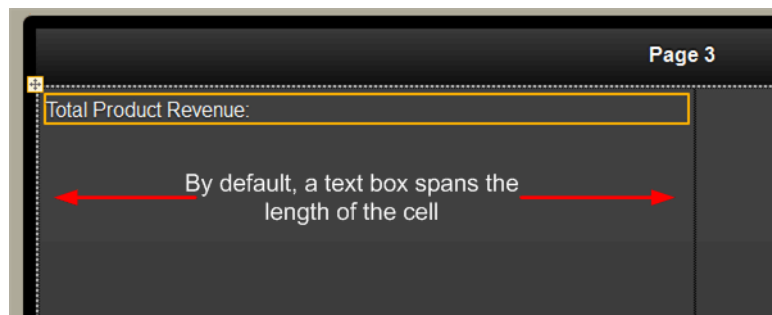
To insert text:

1. From the **Add Component** menu, select **Insert Text**.
2. Double-click the default inserted text to enter your text.
3. Use the **Properties** pane to customize the text display.

3.18.1 Display Text Items Side by Side

By default, when you insert a text item to a frame cell, the text box spans the length of the frame cell.

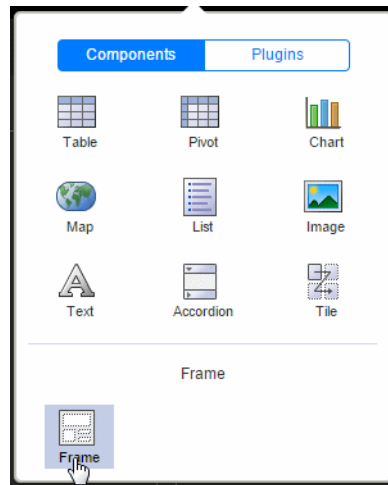
Figure 3–90 *Default Text Field Span*



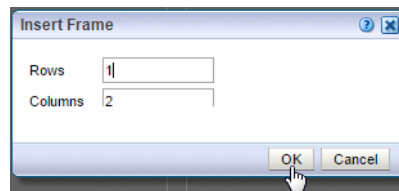
To arrange text fields side by side with data fields or other text fields, use a frame to align the elements. You can insert multiple frames inside outer frames.

Example: To display a data field in line with the text item:

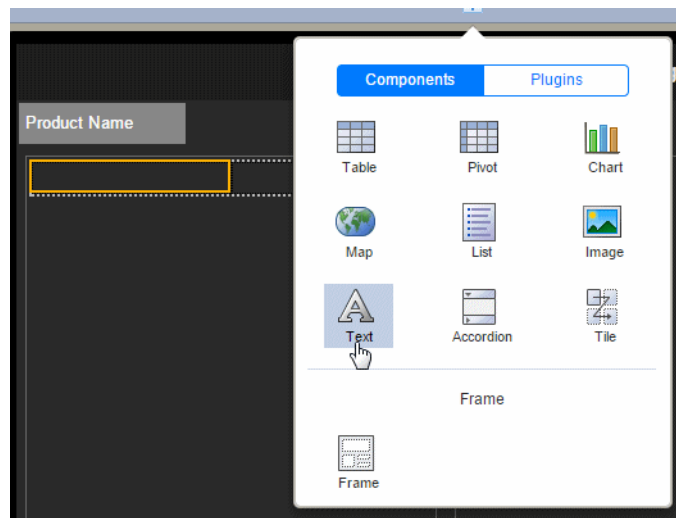
1. Insert a frame.



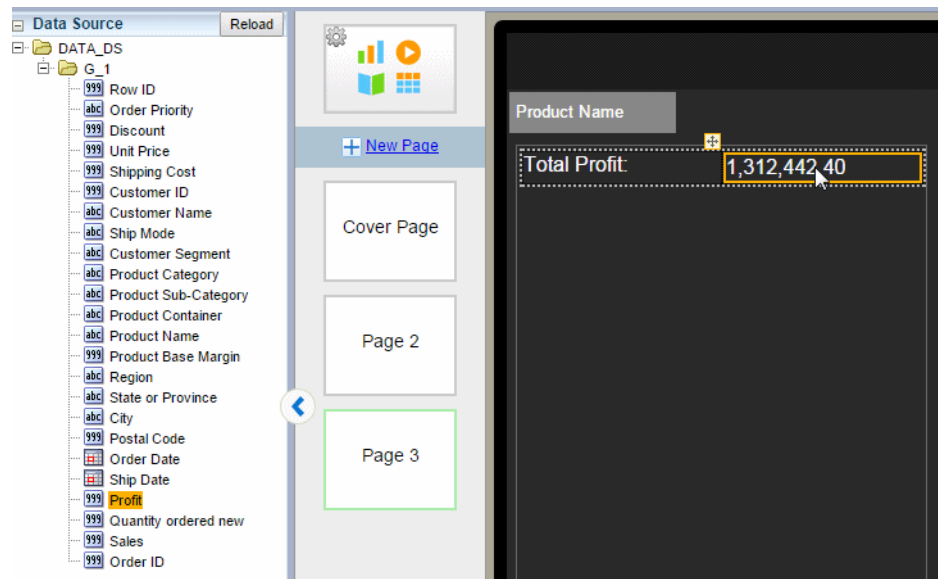
2. Specify the columns and rows. This example uses a single row frame with two columns.



3. Select the frame cell and insert a text component.



4. Enter your text string in the Text field, for example: Total Profit.
5. Drop a data field inline with the text as shown in [Figure 3-91](#).

Figure 3–91 Data Item Inline with Text Item

3.19 Calculated Fields

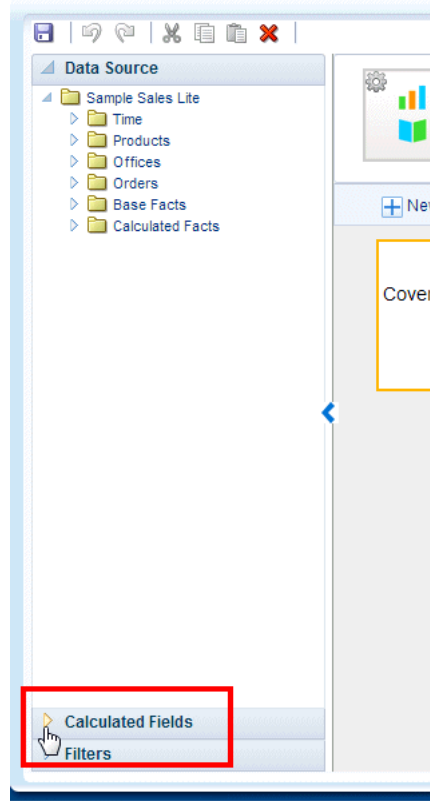
The calculated fields feature enables you to create a calculated field using common expressions and then use it throughout your app. You define calculated fields by applying formulas or functions to the existing data fields in your app. You then drag this calculated field to the component in your app (charts, tables, tiles, sidebars) just as you would any other field in your data source.

Video: Adding Calculated Fields and Filters

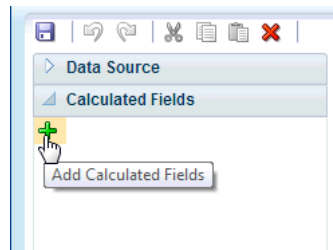
3.19.1 Adding Calculated Fields

To create a calculated field:

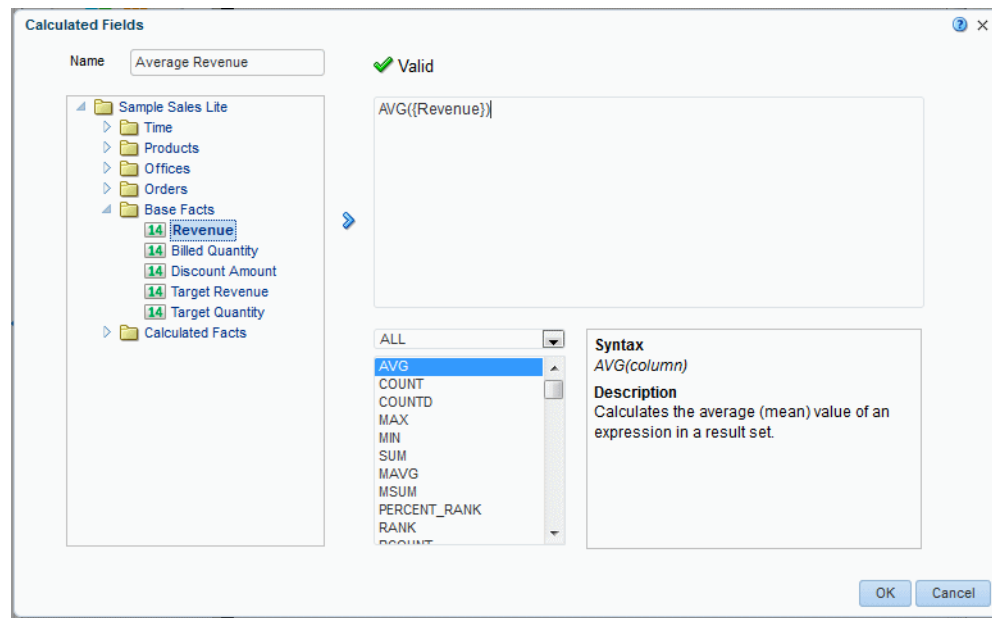
1. On the left-side accordion pane, click **Calculated Fields**.

Figure 3–92 Opening the Calculated Fields Pane

2. Click Add Calculated Fields.

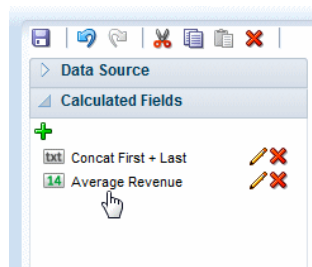


3. In the **Calculated Fields** dialog, enter a **Name** for your calculated field.
4. Select the functions and data fields to construct the calculated field:
 Double-click the function name to add it to the calculation box. To add data fields, select the field and then use the shuttle button to position it in your formula.
 When you select a function, the Syntax, Description, and an Example display on the right.



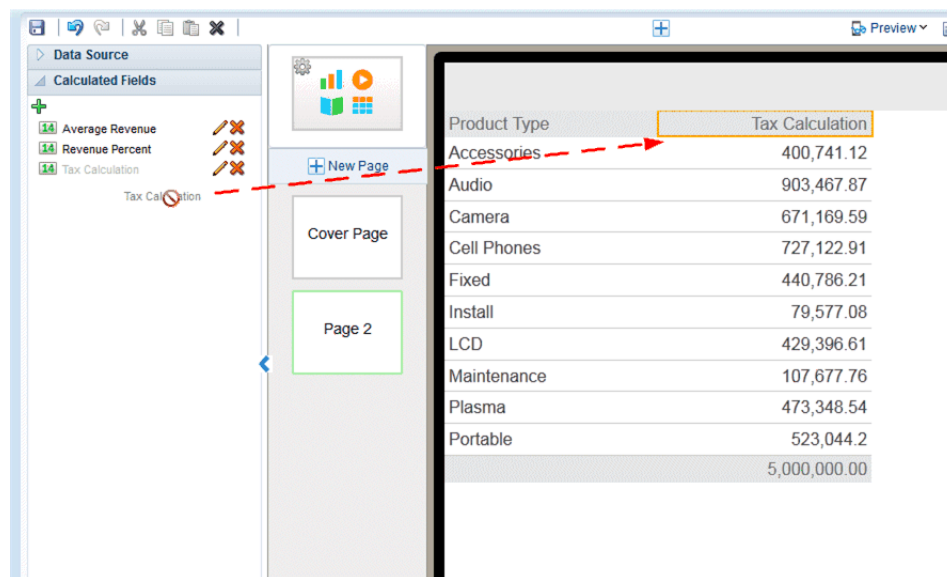
The expression is validated as you enter it.

- When you click OK the expression displays in the Calculated Fields list.



3.19.2 Adding Calculated Fields to App Components

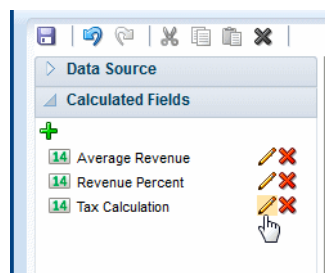
To add a calculated field to a component, simply drag and drop it from the Calculated Fields pane the same way you would a data field from the data source pane. Add calculated fields to tables, charts, lists, maps, pivot tables, sidebars, exploration page filters, and tile pages.

Figure 3–93 Dragging a Calculated Field to a Table

3.19.3 Editing Calculated Fields

To edit a calculated field:

1. Open the **Calculated Fields** pane.
2. Click the **Edit** icon.



3. In the **Calculated Fields** dialog, edit the calculation and click OK. See [Adding Calculated Fields](#) for descriptions of the dialog fields.

3.20 Global Filters

You can define named, global filters and then apply these filters to charts, tables, and tiles throughout your app. Using global filters enables you to define a filter once and use it multiple times.

Video: Adding Calculated Fields and Filters

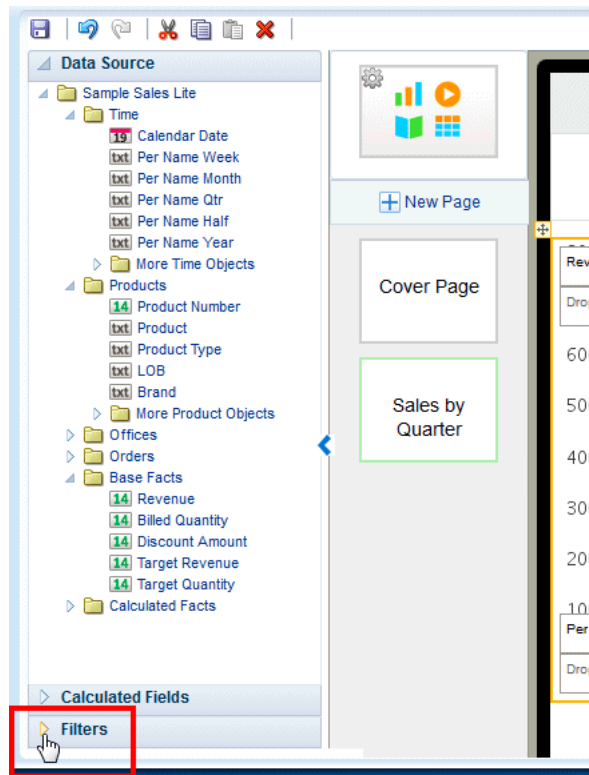
- [Adding Global Filters](#)
- [Applying Global Filters](#)
- [Editing Global Filters](#)

3.20.1 Adding Global Filters

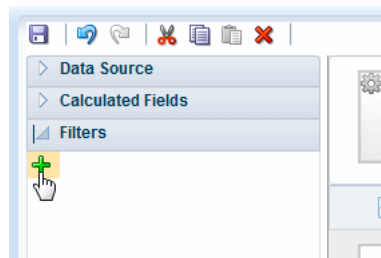
To add a global filter:

1. On the left-side accordion pane, click **Filters** to open the **Filters** pane.

Figure 3–94 Opening the Filters Pane



2. Click **Add Filter**.



3. In the **Filter** dialog, enter the fields to define the filter.

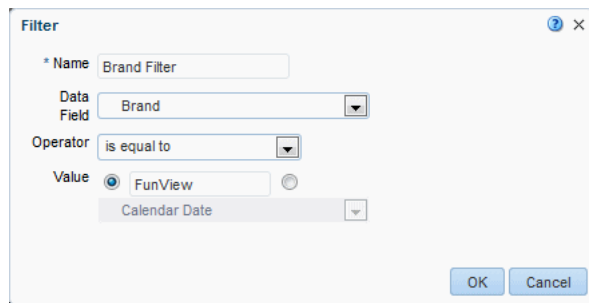


Table 3–1 Fields to Define Filters

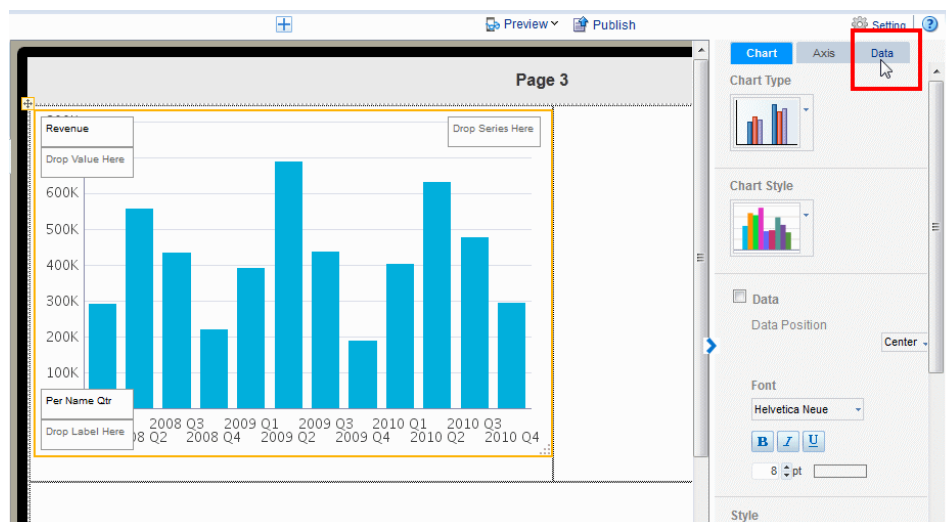
Field	Description
Name	Enter a name for this filter.
Data Field	Select the data field to define the filter. All fields from your data source are available.
Operator	<p>The available operators depend on the data type of the Data Field selected (numeric, text, or date). Not all operators listed display for each data type. Select the operator to define the qualifying criteria for your filter:</p> <ul style="list-style-type: none"> is equal to is not equal to is less than is greater than is less than or equal to is greater than or equal to is between is in top is in bottom is null is not null is in is not in contains not contain starts with ends with is in last N days is in next N days
Value	In the Value field, specify one or more values to use when applying the filter or condition. You can enter a static value or select a data field.

3.20.2 Applying Global Filters

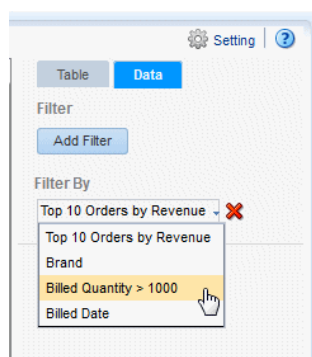
You can apply global filters to charts, tables, tiles, and maps.

To apply a global filter to a chart, table, tile, or map:

1. Select the chart, tile, table, or map in the work area to display the properties for the component.



2. On the Properties palette, select the **Data** tab.
3. On the Data tab, click **Add Filter**.
4. Choose the filter to apply from the list.

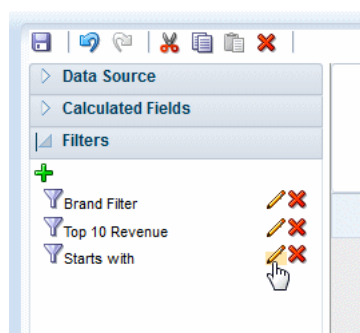


5. To apply more filters to the component, repeat steps 3-5. Filters are applied in the order you arrange them

3.20.3 Editing Global Filters

To edit a global filter:

1. Open the **Filters** pane.
2. Click the **Edit** icon.



3. In the **Filters** dialog, edit the criteria and click OK. See [Adding Global Filters](#) for descriptions of the dialog fields.

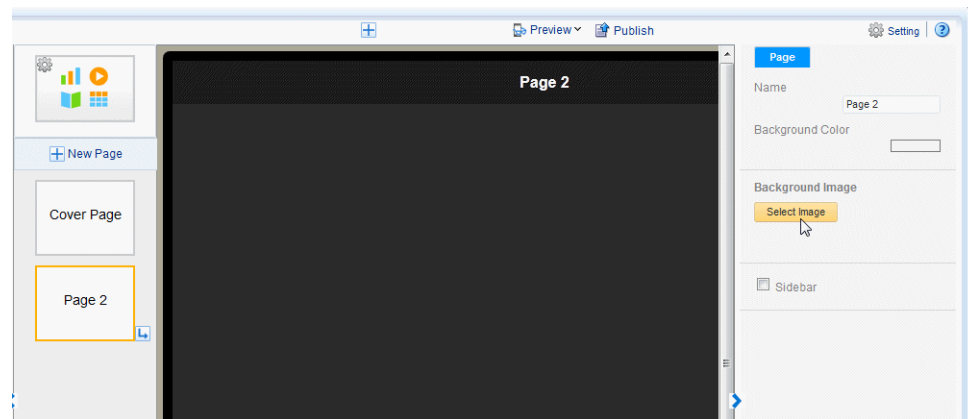
3.21 Customizing Background Images

You can add an image to an app page background and configure the display characteristics of the image. A background image enables you to add other app components on top of it.

To add a background image to a page:

1. Select the page to enable the **Page** properties. The example in [Figure 3–93](#) shows the Page properties.

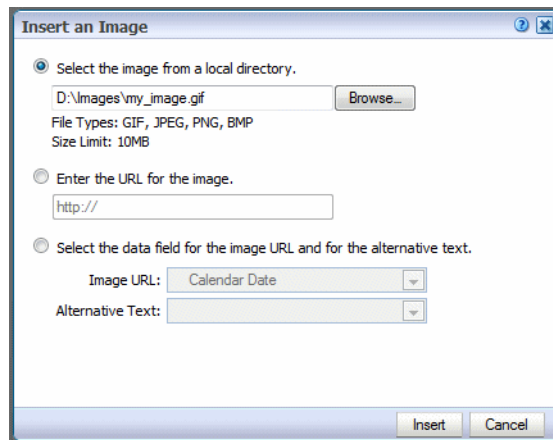
Figure 3–95 The Page Layout Properties



2. Under **Background Image**, click **Select Image** to choose an image.
3. In the **Insert an Image** dialog, specify the source for the image:
 - **Select the image from a local directory:** Click **Browse** to specify the file name and directory of the image on a local or mapped drive to upload the image.
 - **Enter the URL for the image:** Enter the URL where the image is stored.
 - **Select the data field for the image URL and for the alternative text:**

Image URL: Select the field from the data that contains a URL to an image.

Alternative Text: If the data includes a field that contains alternative text for the image, then select that field to display alternative text.

Figure 3–96 Insert an Image Dialog

4. Click **Insert** to insert the image to the app page. By default, the image is scaled to fit the page.

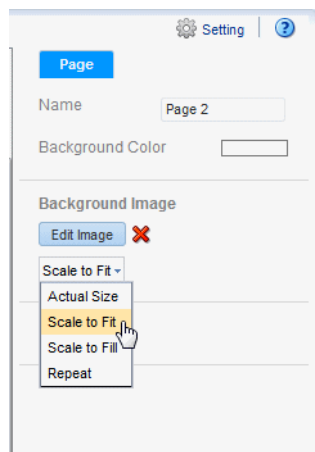


You can now add app components on top of the background.

3.21.1 Sizing the Background Image

To configure the size of the background image:

1. Select the page to enable the **Page** properties
2. Select a fit option from the menu.



- **Actual Size** - displays the image according to the actual pixel dimensions.

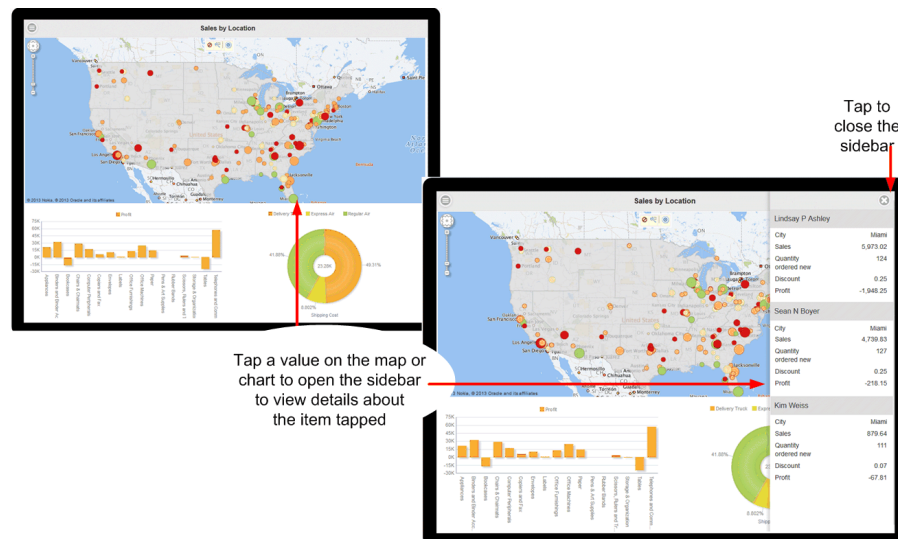
- **Scale to Fit** - expands the image maintaining the scale until one dimension reaches the edge. If one dimension reaches the edge of the canvas and the other dimension has not, then that area of the canvas is not covered.
- **Scale to Fill** - expands the image maintaining the scale to cover the entire canvas. If one dimension extends past the canvas for the other dimension to reach the edge, the image dimension that extends past the canvas is cropped.
- **Repeat** - repeats the image at actual size to fill the canvas. Use this option to use a small image to create a pattern or texture in the background.

3.22 Adding Sidebars

A sidebar provides pop-up detail information when you tap a chart, a map, or tile on a page. The following figure shows a page with sidebar enabled. Tapping a value on the page, such as the city of Miami on the map, launches the sidebar on the right side of the page showing detail information for the selection.

Video: Adding Detail Information Using Sidebars and Subpages

Figure 3–97 Sidebar Displaying Details



To close the sidebar, click the close icon in the upper right corner.

Use the sidebar when you want to provide a set of detail information about the values shown on your page and still maintain the context by staying on the app page. If you want to provide a full set of drill-down visualizations, consider using a subpage instead (see [Main Pages and Subpages](#)).

Design Note: A page can support either a sidebar or a subpage, but not both. When designing your page, consider the amount and type of information you want to display from each touch point. If you want to supply a simple table-style list of additional data, then a sidebar is most likely appropriate. If you want to supply additional visualizations, such as charts and tables driven by the touch point, then a subpage would be more appropriate.

Note the following about sidebars:

- A sidebar is enabled at the page level. If you want every page of your app to display a sidebar, you must set up the sidebar for each page.
- The sidebar is activated at run time when you tap a value on a chart, a map, or a tile.
- You can define one sidebar per page; therefore, all items that can activate the sidebar display the same set of fields. In the example in [Figure 3–97](#) the fields Customer, City, Sales, and so on display when you tap a value on either the charts or the map.

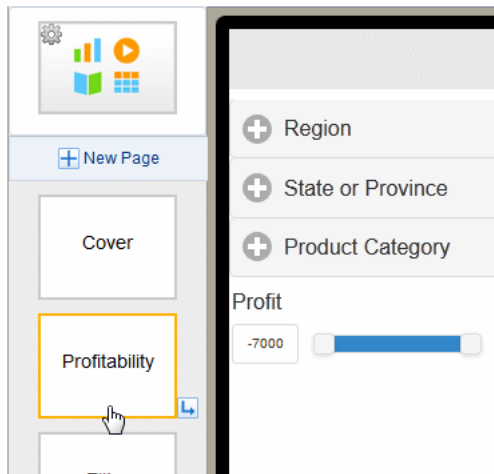
To add and customize a sidebar to your app page see the following sections:

- [Inserting a Sidebar](#)
- [Adjusting the Label Field Width](#)
- [Customizing the Fields in the Sidebar](#)

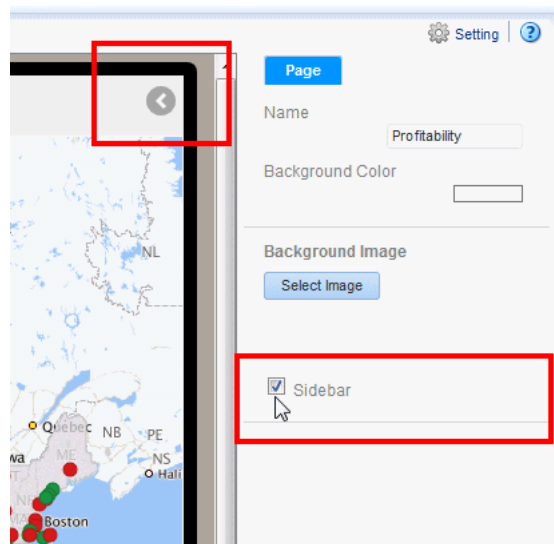
3.22.1 Inserting a Sidebar

To add a sidebar to an app page:

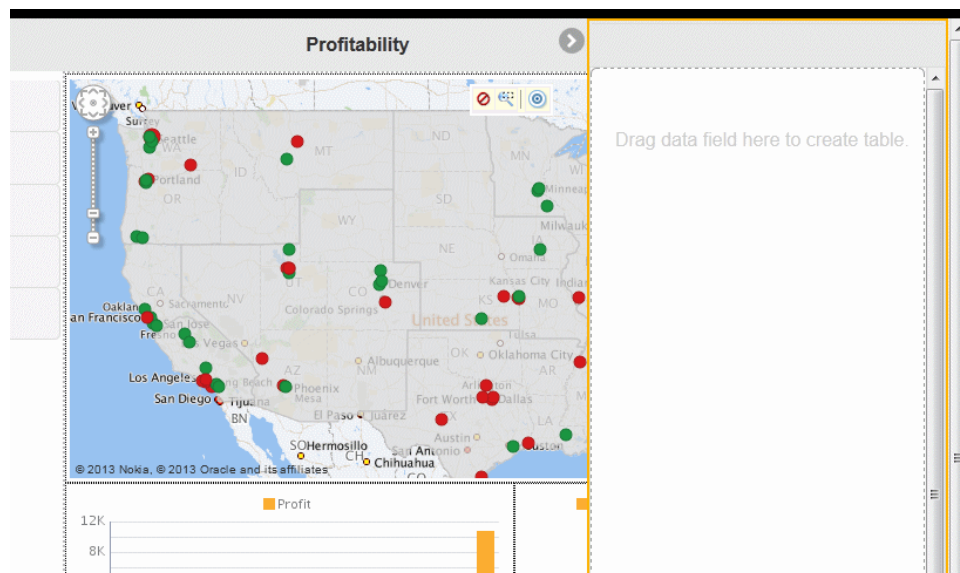
1. Select the app page.



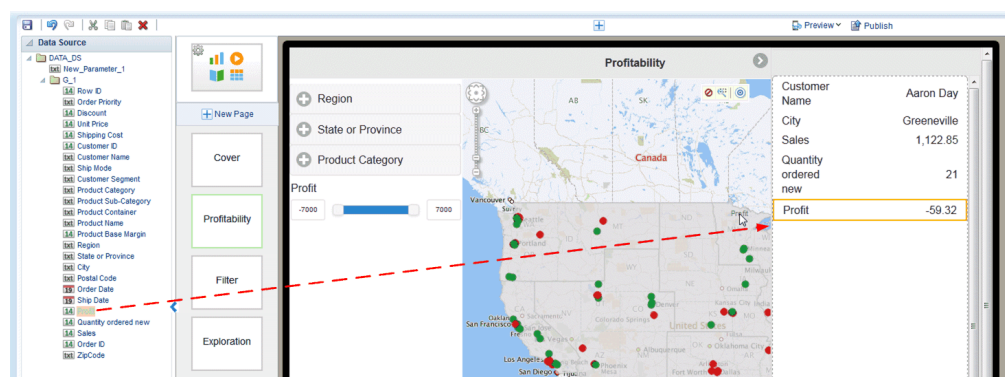
2. On the **Properties** pane, set the Sidebar property to True. After you set the property, the Sidebar indicator displays in the upper right corner of the page, as shown in the following figure.



- Click the Sidebar indicator to open the sidebar panel for editing.



- Drag the data elements that you want to provide detail for from the Data Source pane to the Sidebar panel.



The sidebar presents the data as a table. You can customize how the data displays and also apply grouping.

5. (Optional) Adjust the label field size for the sidebar label column. To adjust the space allowed for the label field

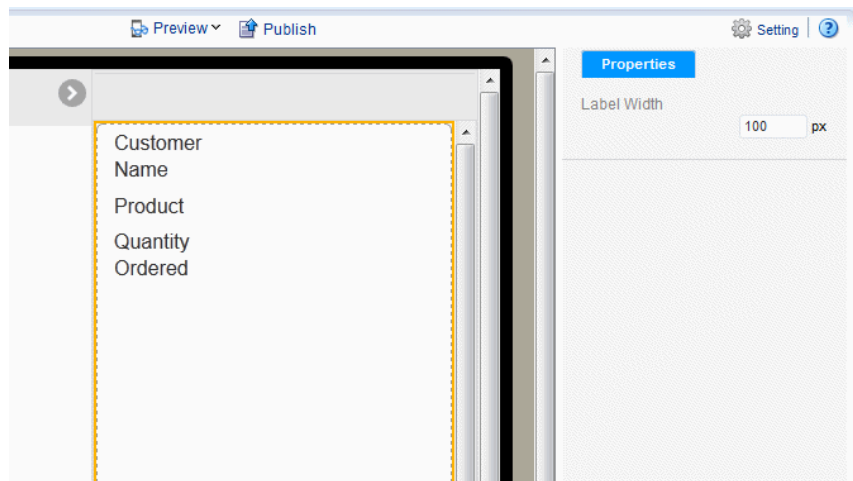
3.22.2 Adjusting the Label Field Width

By default the label field width is 100 pixels. The label field width is set for the entire sidebar and cannot be adjusted individually for data fields.

To adjust the label column width for the sidebar:

1. Click the edge of the sidebar component to select the entire component.
2. In the Properties pane, enter the width in pixels for the label field.

Figure 3–98 *Setting the Sidebar Label Width Property*



3.22.3 Customizing the Fields in the Sidebar

You customize each field in the sidebar individually. When you select a field, the Field properties display. Use the Field **Properties** set to group data fields and customize the display. You can perform the following for a data field in the sidebar:

- [Group By a Data Field](#)
- [Apply a Formula](#)
- [Customize Data Formatting](#)
- [Customize Labels, Fonts, and Background](#)

3.22.3.1 Group By a Data Field

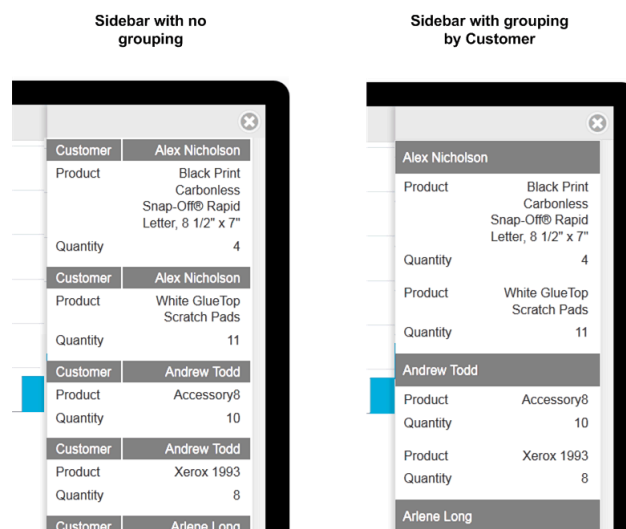
The Group By feature groups occurrences of the specified data field together, aggregating the values of the other fields. For example, suppose in your sidebar you place the data fields:

- Customer
- Product
- Quantity Ordered

If you Group By Customer, the sidebar shows one entry per customer with all the customer products grouped below the single entry. See the example in [Figure 3–99](#).

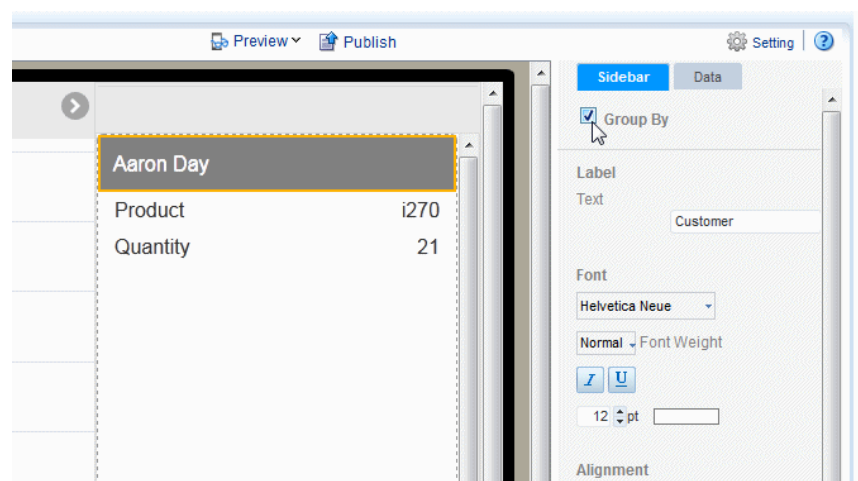
The ungrouped sidebar shows multiple entries for a single customer, the grouped entry creates one entry per customer.

Figure 3–99 Examples showing ungrouped and grouped data



To group the fields in your sidebar:

1. Select the data field in the sidebar to display its Properties.
2. Select **Group By**.

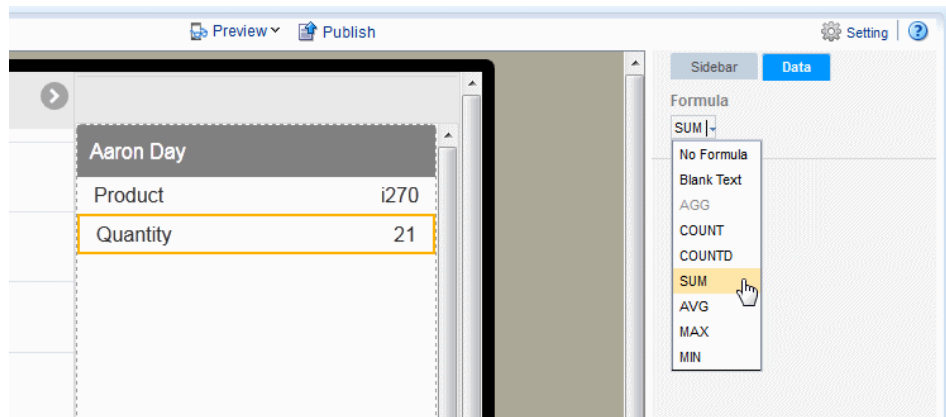


3.22.3.2 Apply a Formula

By default, numeric fields are summed. You can change the formula in the Properties pane to a basic predefined formula.

To change the formula:

1. Select the data field in the sidebar to display its Properties.
2. Click the Data tab.
3. Choose from the list of formulas. The following formulas are supported: No Formula, Blank Text, Count Distinct, Sum, Average, Maximum, and Minimum.



To apply a more complex formula to a data field, use the Calculated Fields feature to create the field then add it to the sidebar. See [Calculated Fields](#).

3.22.3.3 Customize Data Formatting

Customize the formatting applied to a number, date, or currency field. See [Section 5.6.1, "Apply Data Formatting."](#)

3.22.3.4 Customize Labels, Fonts, and Background

Use the **Properties** for the sidebar Field for customizations such as:

- Edit the label shown. The default label comes from the data, for example "Quantity ordered new". Use the Label property to edit the display name.
- Edit font, font color, font background for both the data and the label.

Previewing, Sharing, and Publishing Apps

Mobile App Designer makes it easy to preview and share your apps while still in design mode. Once your app is finished, with just a few clicks you can publish it to your company's Apps Library where it can immediately be consumed by authorized users.

This chapter includes the following sections:

- [Previewing Apps](#)
- [Sharing Apps](#)
- [Embedding Apps](#)
- [Publishing Apps to the Apps Library](#)

4.1 Previewing Apps

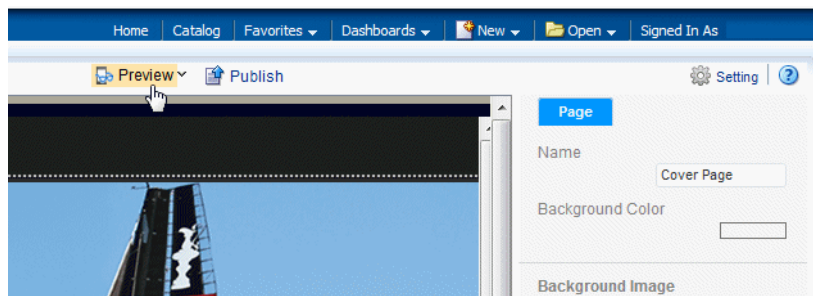
While still in design mode you can preview your app at any time. You can preview on your desktop in the mobile device simulator with a single click, or you can preview on a mobile device.

4.1.1 Preview on Your Desktop

To preview an app on your desktop:

1. In the Mobile App Designer, click **Preview**.

Figure 4–1 *Selecting Preview Option*



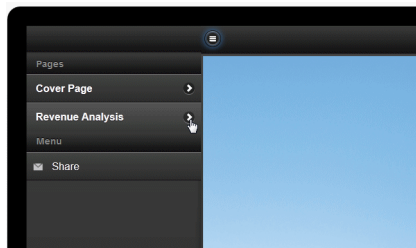
Your app is displayed in the tablet or mobile phone simulator.

Figure 4–2 Tablet Simulator in Preview Mode



To view the other pages of your app, click the page menu icon in the upper left corner, then click the page title.

Figure 4–3 Using the Page Menu to Navigate the App



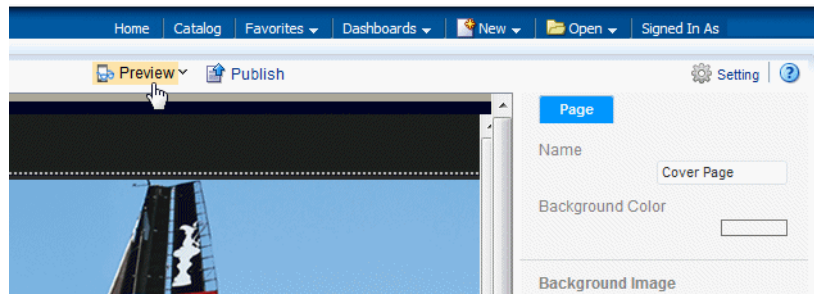
2. Click **Edit** to return to the designer.

4.1.2 Preview on a Mobile Device

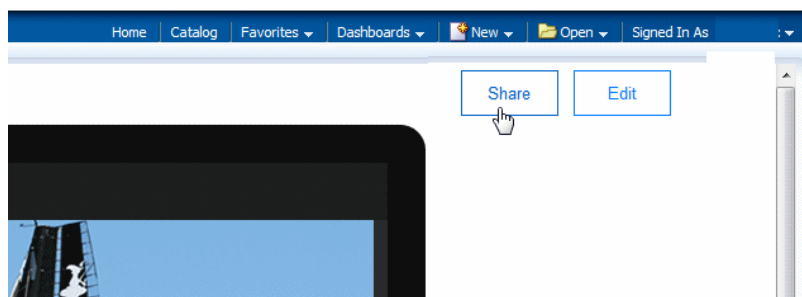
To preview an app on a mobile device:

1. In the Mobile App Designer, click **Preview**.

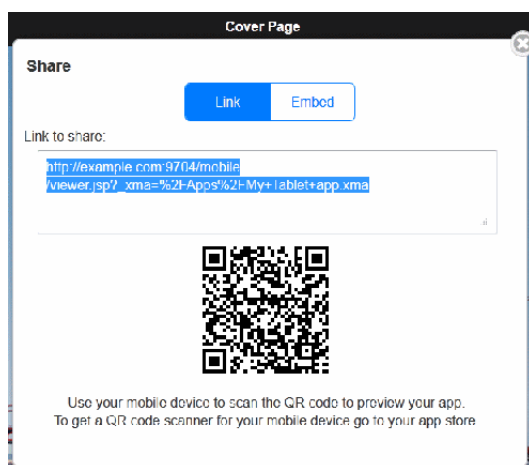
Figure 4–4 Selecting Preview Option



2. On the Preview page, click **Share**.



3. To view in your mobile device, scan the QR code; or, copy the **Link to share** and e-mail it to yourself or others.

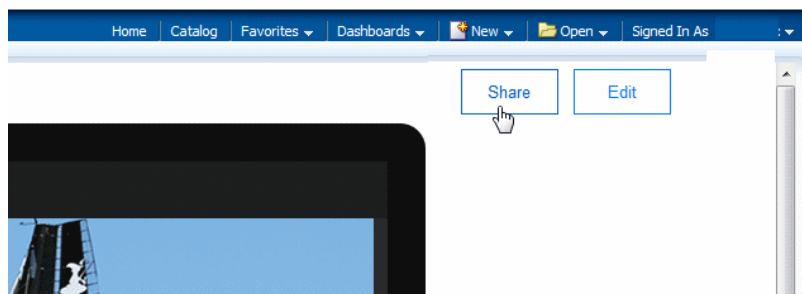


4.2 Sharing Apps

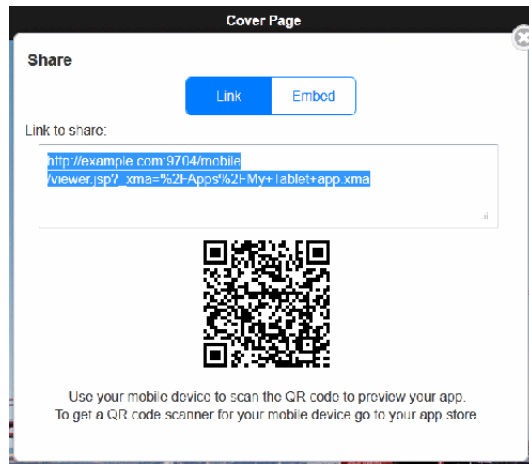
The **Share** option of the mobile app viewer generates a link to your app that you can share with others.

To generate the link to share:

1. Open your app in the mobile app viewer, either by clicking **Preview** in the designer or by clicking the **Open** link in the catalog.
2. In the mobile app viewer, click **Share**.



3. Copy the **Link to share** and send it to yourself or others.

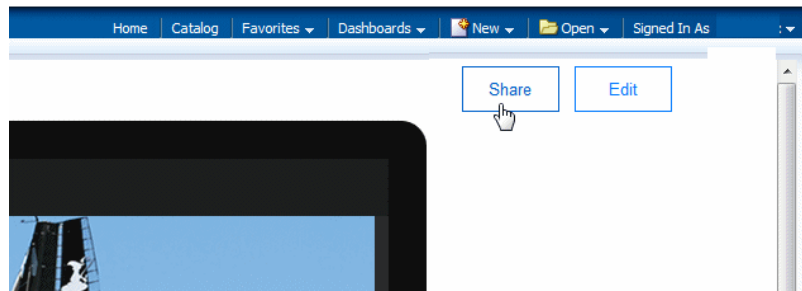


4.3 Embedding Apps

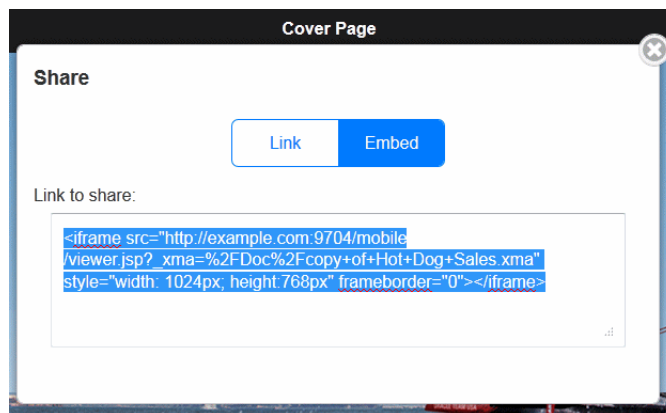
The mobile app viewer also generates a code snippet for you to easily embed your app into other Web pages.

To generate the code snippet for embedding:

1. Open your app in the mobile app viewer, either by clicking **Preview** in the designer or by clicking the **Open** link in the catalog.
2. In the mobile app viewer, click **Share**.



3. In the **Share** dialog, click **Embed**.



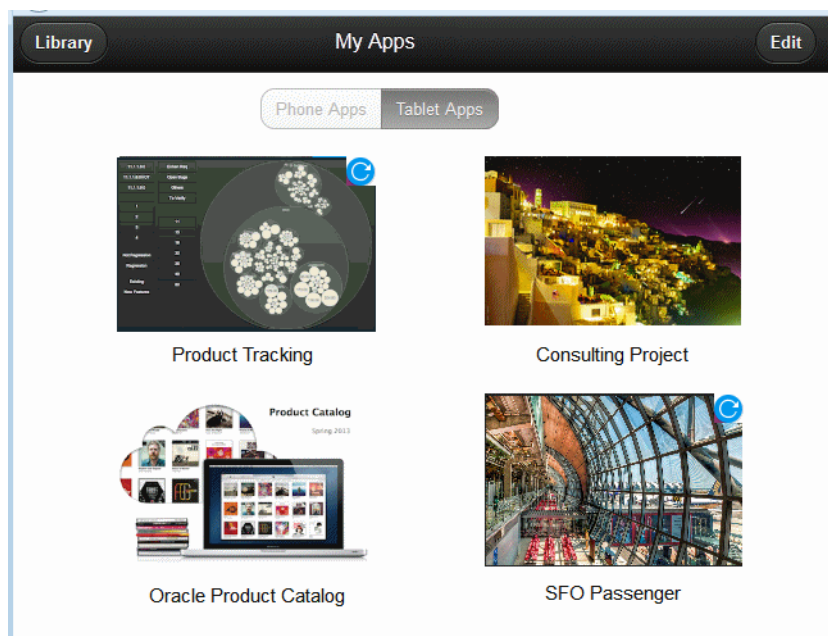
4. Copy the code snippet and include in your Web page code to launch your app from another site.

4.4 Publishing Apps to the Apps Library

The Apps Library is where users access and subscribe to your apps from their mobile devices. When you publish your app it is copied to the Apps Library folder set up by your administrator and is immediately available to users with access to the library and permissions to run your app.

If you update an app that you have already published to the Apps Library, users that have subscribed to your app see a **Refresh** icon in their My Apps library notifying them that the newer version is available. Users must explicitly click the **Refresh** icon to get the new version.

Figure 4–5 *My Apps Library Showing New Versions of Apps Available*



Prerequisites

The Apps library must be configured by an administrator see [Section A.2, "Setting Up the Apps Library Locations."](#)

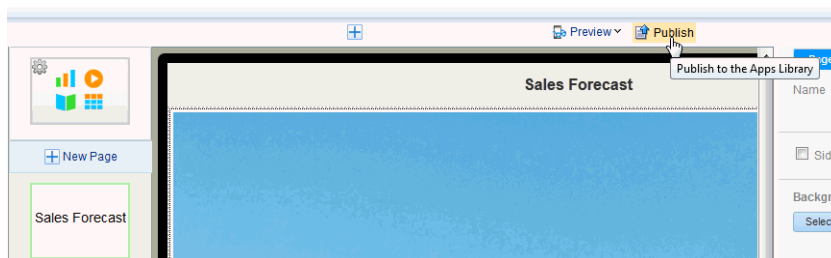
You must have write permission on the Apps Library folder in the catalog to which you are publishing.

Oracle recommends that you apply appropriate permissions to your app before you publish it.

The Publish feature enables you to quickly expose your apps in the Apps Library configured for your environment.

To Publish an app:

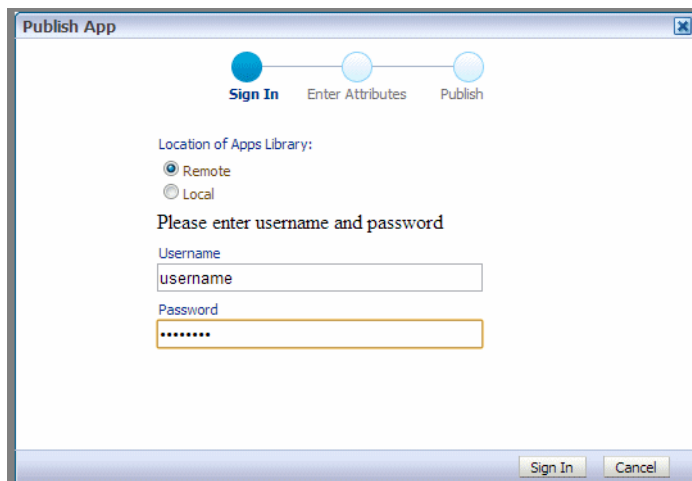
1. From the Mobile App Designer, click **Publish**, as shown in [Figure 4–6](#).

Figure 4–6 Choosing the Publish Command

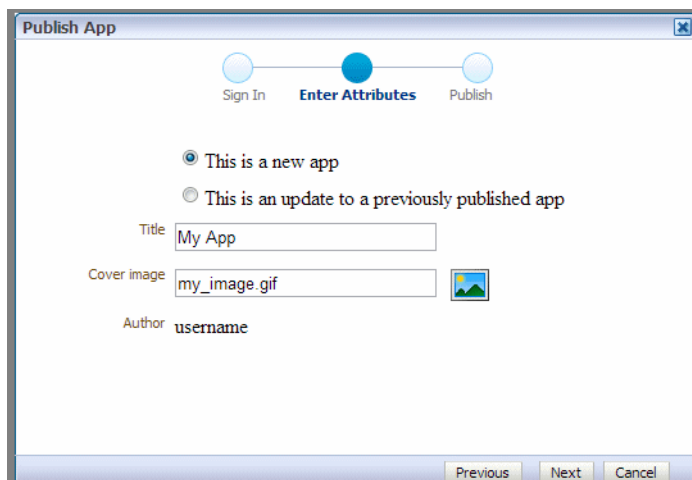
2. In the **Publish App** dialog, choose whether to publish your app to the Apps Library in the current (**Local**) environment or to a preconfigured **Remote** environment.

If you choose **Local**, click **Next**.

If you choose **Remote**, enter the username and password for the target environment.

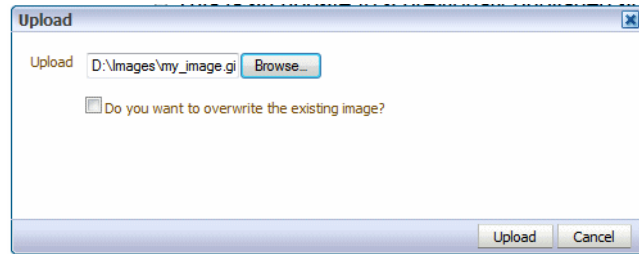
Figure 4–7 Publish Apps Sign In Step for Remote Apps Library

3. Enter the attributes for your app:

Figure 4–8 Publish App Enter Attributes Step

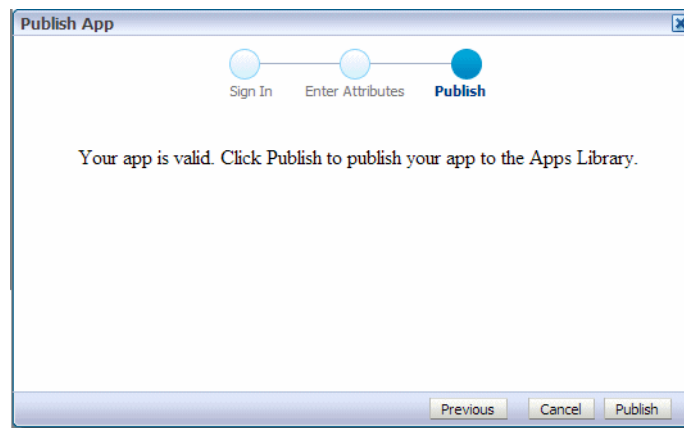
- **Title** - enter the title of the app to display in the Apps Library.
- **Cover Image** - upload a cover image for the App. The cover image is a thumbnail image that displays in the Apps Library for your app. A cover image is required.

Click the image icon to open the **Upload** dialog.



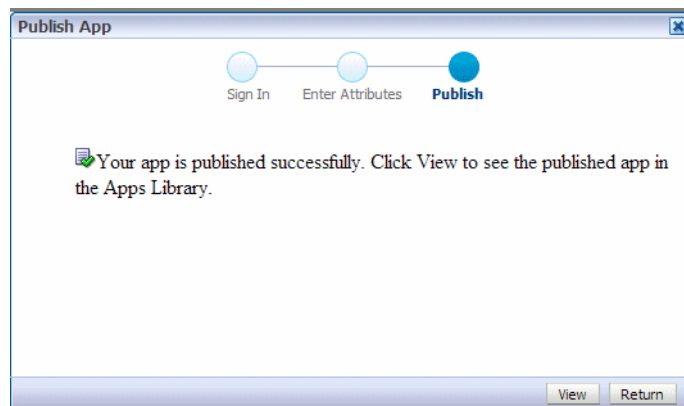
Click Browse to locate the image in the file system, then click **Upload**.

4. Click **Next** to validate your app. The validation process verifies that the data source (BI Publisher data model or BI subject area) exists in the target environment.



If validation fails, ensure that the data model or subject area exists in the target environment and is located in the same path as the source environment.

5. Click **Publish** to publish your app to the target Apps Library.



6. Click **View** to view your App in the Apps Library or click **Return** to return to the Mobile App Designer.

Component Property Features

This chapter describes component properties of the BI Mobile App Designer.

This chapter includes the following sections:

- [Working with Tables](#)
- [Working with Pivot Tables](#)
- [Adding Filters to Tiles, Charts, Tables, and Pivot Tables](#)
- [Adding a Search Filter to Navigation and Accordion Lists](#)
- [Adding Static or Dynamic Links](#)
- [Features of Metric Fields](#)
- [Setting Borders](#)
- [Setting Margins](#)
- [Using the Color Picker](#)
- [Custom Number and Date Formats](#)

5.1 Working with Tables

The following sections describe additional features of tables in the Mobile App Designer:

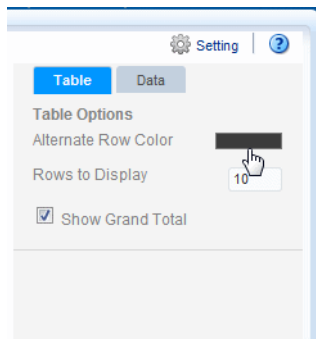
- [Customizing Alternating Row Colors](#)
- [Setting Table Options](#)
- [Customizing Column Headers](#)
- [Customizing Table Data Display](#)
- [Customizing Table Totals](#)

5.1.1 Customizing Alternating Row Colors

By default, alternating table rows in apps display a different gradient for easier viewing. You can customize the alternating row color.

To set an alternating row color:

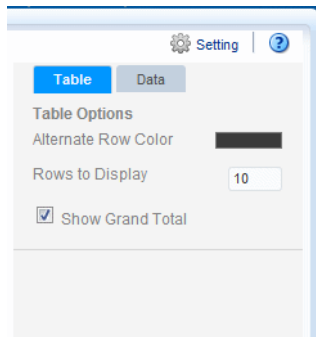
1. Select the table to display its properties.
2. Under **Table Options**, click the value shown for **Alternate Row Color** to launch the color picker. [Figure 5–1](#) shows the **Alternate Row Color** option.

Figure 5–1 Alternating Row Color Property

3. Choose a color and click **OK**.

5.1.2 Setting Table Options

Figure 5–2 shows the Table properties

Figure 5–2 The Table Properties

Use the Table properties to perform the following:

- [Set the Number of Rows to Display](#)
- [Define Filters for Data Displayed in the Table](#)
- [Show or Hide the Total Row](#)

5.1.2.1 Set the Number of Rows to Display

The **Rows to Display** property controls the number of rows of data displayed as follows:

- When designing an app, this property sets the number of rows that are displayed for the table within the app designer.
- When viewing the app, this property sets the size of the scrollable region for the table.

The default is 10 rows of data.

Note: Displaying more rows of data can impact the performance of the app designer.

5.1.2.2 Define Filters for Data Displayed in the Table

See [Section 3.20, "Global Filters."](#)

5.1.2.3 Show or Hide the Total Row

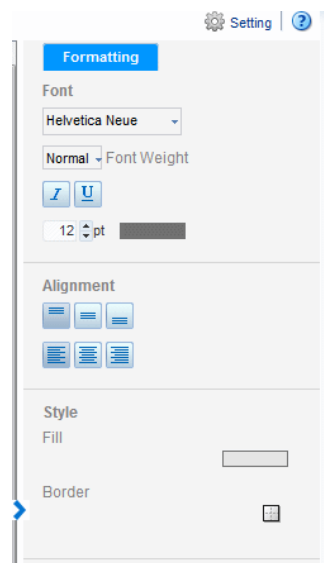
By default, the app designer inserts a total row in a table that sums numeric columns. To remove the total row, deselect the **Show Grand Total Row** option.

The total row can be further customized using the **Total Cell** properties. For more information see [Section 5.1.5, "Customizing Table Totals."](#)

5.1.3 Customizing Column Headers

[Figure 5–3](#) shows the **Table Column Header** properties.

Figure 5–3 The Table Column Header Properties



Use the Table Column Header options to perform the following:

- Edit the font properties of the table header column
- Edit the cell properties of the table header including border weight, style, and color and background fill color
- Set the vertical and horizontal alignment of the table header

5.1.4 Customizing Table Data Display

The Column properties display when you select a specific column in a table.

[Figure 5–4](#) shows the Column properties

Figure 5–4 Table Column Properties

The screenshot shows the 'Table Column Properties' dialog box with the 'Formatting' tab selected. The 'Font' section shows 'Helvetica Neue' as the font, 'Light' as the font weight, and '12 pt' as the font size. The 'Alignment' section has icons for vertical and horizontal alignment. The 'Style' section has a 'Fill' color picker and a 'Border' style icon. The 'Data Formatting' section shows a currency format '-\$1,234.57', '2' decimals, and a checked 'Thousands Separator'. The 'Conditional Formatting' section has 'Highlight' and 'Manage Formats' options. The 'Link' section has an 'Add Link' button.

Use the Column Formatting properties to:

- Edit the font properties of the column including style, size, and color
- Edit the style properties of the column including border weight, style, and color and background fill color
- Set the vertical and horizontal alignment of the column contents
- Apply formatting to the column data (see [Section 5.1.4.3, "Setting Table Data Formatting Options"](#))
- Apply conditional formatting to the column (see [Section 5.1.4.1, "Applying Conditional Formats to a Table Column"](#))
- Create links for table column data (see [Section 5.5, "Adding Static or Dynamic Links"](#))

Use the Column Data properties to:

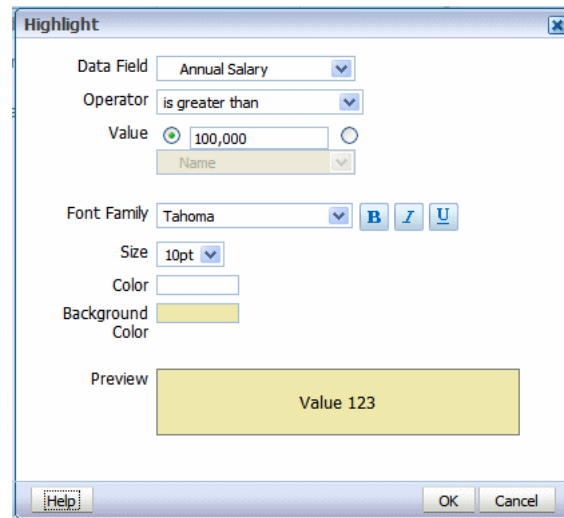
- Apply grouping (see [Section 5.1.4.9, "About Grouping"](#))
- Apply a formula to the column data
- Apply sorting and sort precedence (see [Section 5.1.4.7, "About the Sort Option"](#))

5.1.4.1 Applying Conditional Formats to a Table Column

To apply a conditional format:

1. Select the table column to display its properties.
2. Click the **Highlight** button. This launches the **Highlight** dialog.

Figure 5–5 The Highlight Dialog



3. Enter the fields to define a condition and format to apply, as described in [Table 5–1](#).

Table 5–1 Fields to Define Conditions and Formats

Field	Description
Data Field	Select the data field to apply the condition to. All elements are available regardless of whether they are included as table columns. For example, you may want to highlight in red all employees with salaries greater than \$10,000, but not actually include the salary element in the table.
Operator	Select from the following operators: is equal to, is not equal to, is less than, is greater than, is less than or equal to, is greater than or equal to, is between
Value	Enter the value or values appropriate for the operator selected. You can enter a static text value or choose a data field Important: If entering a date value, use one of the following XSL date or time formats: YYYY-MM-DD or YYYY-MM-DDTHH:MM:SS.
Font Family	Select the font to apply to the row of data that meets the condition. You can also apply bold, italic, or underline emphasis.
Size	Select the size of the font to apply to the row of data that meets the condition.
Color	Click the color box to open the Color Picker . Choose one of the predefined colors or click Custom Color to define a color to apply to the font.
Background Color	Click the color box to open the Color Picker . Choose one of the predefined colors or click Custom Color to define the background color to apply to the row.

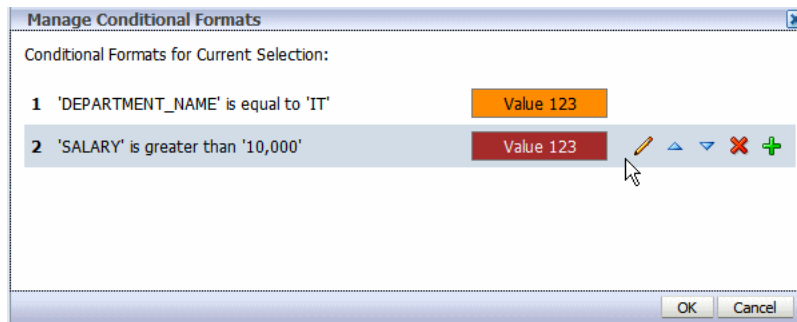
5.1.4.2 Managing Formats

After you have added conditional formats, use the **Manage Formats** command to edit or delete a format.

To manage formats:

1. Click the **Manage Formats** button to launch the **Manage Conditional Formats** dialog.

Figure 5–6 Managing Conditional Formats



2. Hover the cursor over an item to display the actions toolbar. Use the toolbar buttons to edit the format, move the format up or down in the order of application, delete, or add another format. The order of the conditions is important because only the first condition that is met is applied.

5.1.4.3 Setting Table Data Formatting Options

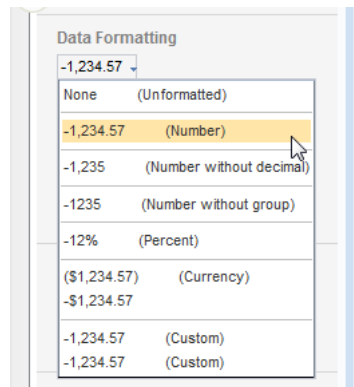
The options you see in the Data Formatting region depend on the data type of the column you selected. These features are described in the following sections:

- [Formatting Numeric Data Columns](#)
- [Formatting Date Type Data Columns](#)
- [Custom Data Formatting](#)

5.1.4.4 Formatting Numeric Data Columns

If the column contains numeric data, the following formatting options are available:

- **Format** - Select one of the common number formats from the list. The format is applied immediately to the table column. The formats are categorized by Number, Percent, and Currency, as shown in [Figure 5–7](#).

Figure 5–7 Number, Percent, and Currency Formats

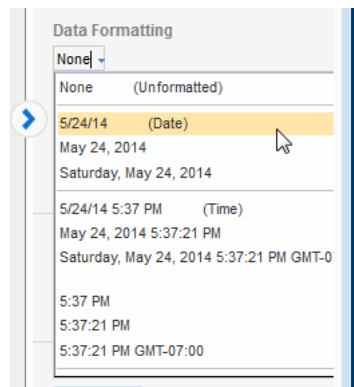
To apply a format not available from this list, see [Section 5.1.4.6, "Custom Data Formatting."](#)

- **Decimal position** - Use the arrow buttons to increase or decrease the number of decimal positions displayed to the right of the decimal point.
- **Thousands Separator** - Select this checkbox to show the grouping separator (for example, 1234.00 displays as 1,234.00).

5.1.4.5 Formatting Date Type Data Columns

If the column contains dates, the following formatting options are available:

- **Format** - Select one of the common date formats from the list. The format is applied immediately to the table column. The formats are categorized by Date and Time, as shown in [Figure 5–8](#).

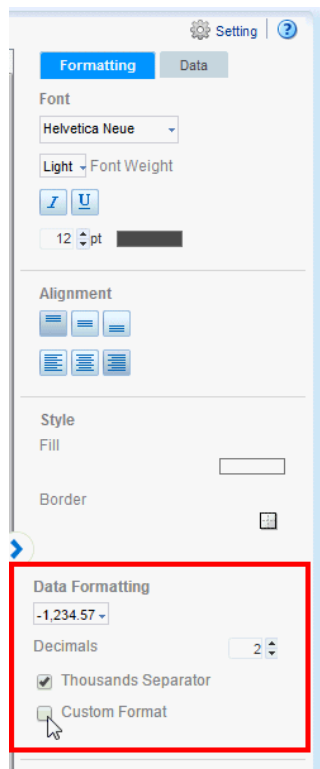
Figure 5–8 Date and Time Formats

5.1.4.6 Custom Data Formatting

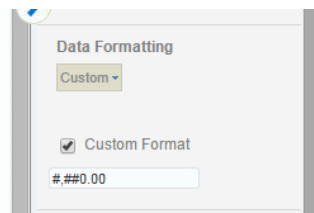
BI Mobile App Designer supports the use of format masks for custom data formatting. The display of the output depends on the locale selected for viewing the app.

To apply custom data formatting:

1. Select a data field or column.
2. The data formatting **Properties** are displayed as shown in [Figure 5–9](#).

Figure 5–9 Data Formatting Properties

3. Under Data Formatting, select **Custom Format**
4. Enter the format mask in the text box. For example, for a column that contains product totals, you can enter `#,###.00` formatting mask to display total values with two zeros to the right of the decimal place. See [Custom Number and Date Formats](#) for the list of supported formats.



5.1.4.7 About the Sort Option

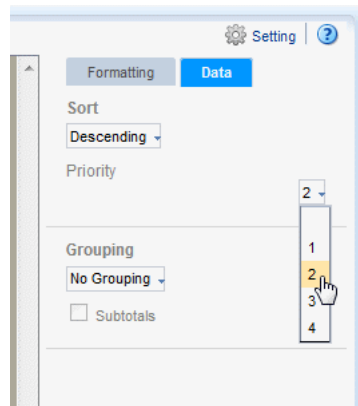
To sort the data in a column, select the column, then in the Properties pane, select the **Data** tab. Under the **Sort** group select **None**, **Ascending**, or **Descending**.

To sort by more than one column, select the column, the sort order, and then assign a **Priority** to each column.

To apply multiple sort orders to a table:

1. Select the column.
2. On the **Column** properties, click the **Data** tab. Under **Sort**, select **Ascending** or **Descending**.
3. From the **Priority** list, select 1.

[Figure 5–10](#) shows the **Priority** list.

Figure 5–10 Priority List

4. Select the next column you want to sort by.
5. On the **Column** properties, click the **Data** tab. Under **Sort**, select **Ascending or Descending**.
6. From the **Priority** list, select 2.
7. Repeat Steps 1 - 6 for each column you want to sort by, applying the appropriate priority for each column.

5.1.4.8 Removing a Sort Order

To remove a sort order applied to a column:

1. Select the column.
2. On the **Column** properties, click the **Data** tab. Under **Sort**, select **None**.

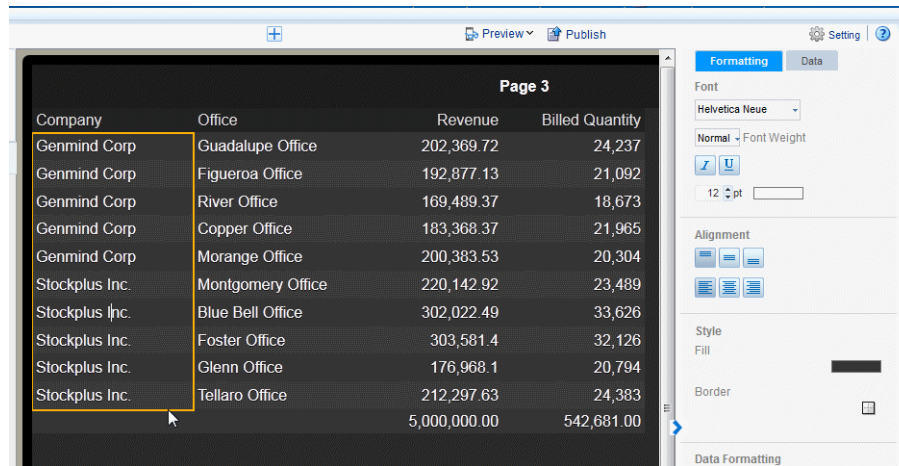
5.1.4.9 About Grouping

"Grouping" groups together elements in the data of the same value. In a table, applying grouping can make the table easier to read. After you group elements in a table, you can also add a subtotal row for each group. Grouping can be applied to a table column header or to a table column.

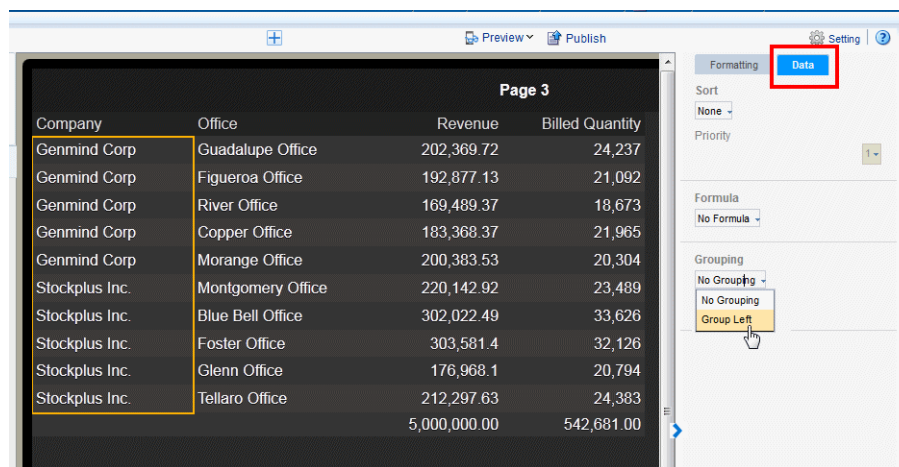
5.1.4.10 Apply Group Left

To apply group left to a table column:

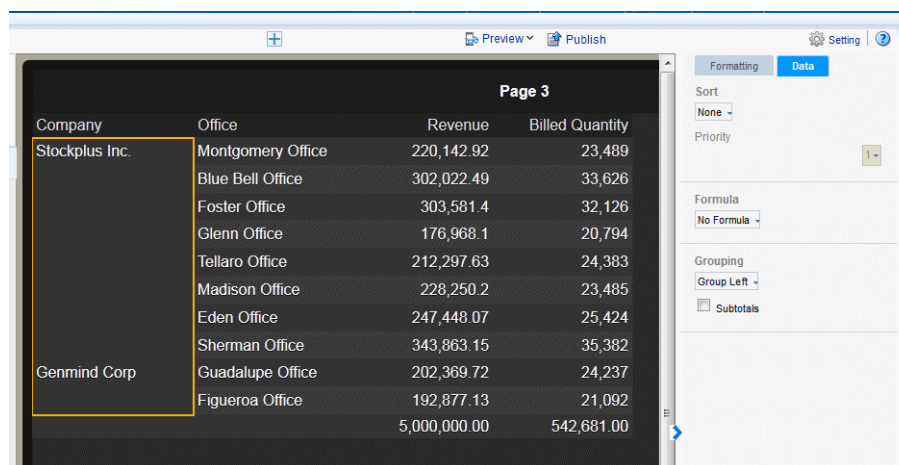
1. Select the Table Column to display its properties.



- Click the Data tab of the properties pane. Under the Grouping properties, select **Group Left**.



Like values are now grouped together for the selected column of your table.

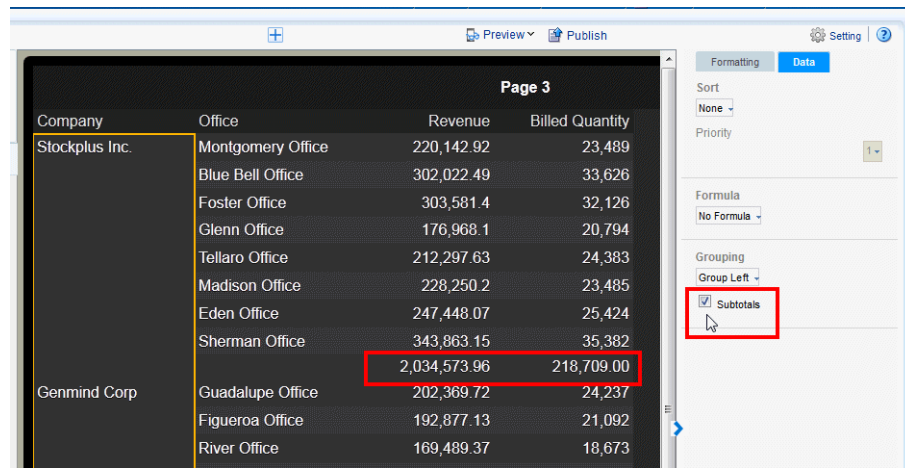


5.1.4.11 Applying Subtotals

To further enhance a table, you can add a subtotal row to display for each grouped occurrence of the element.

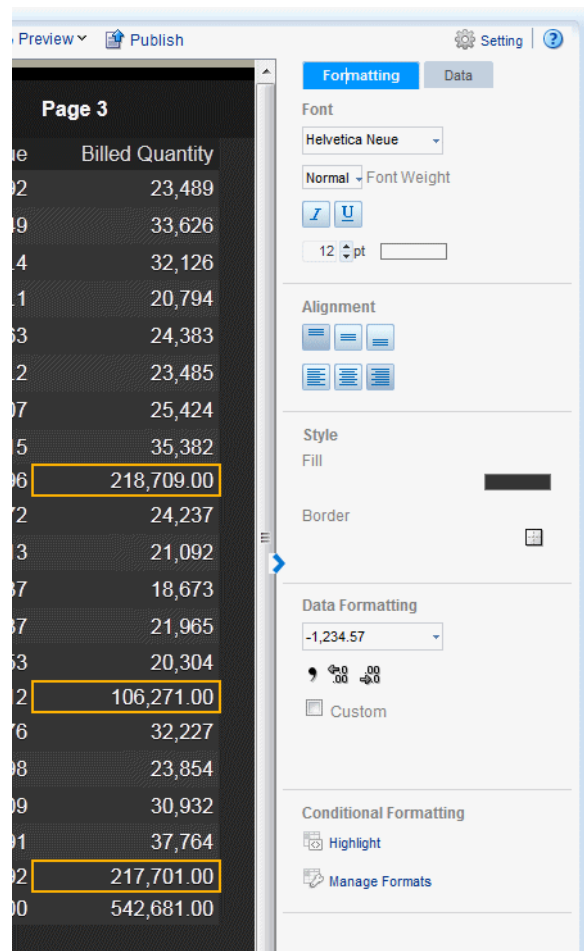
To apply subtotals for a grouped column:

1. Select the table column of the grouped column.
2. On **Data** tab of the properties pane, select **Subtotals**. Your table now shows a subtotal row for each group.



5.1.5 Customizing Table Totals

The app designer automatically inserts a grand total row when you insert a data table to the page. As shown in the section on grouping, you can also insert subtotal rows within the table based on a grouping field. To edit the attributes of the cells in a grand total or subtotal row, select the cell that displays the total and use the options in the Total Cell properties shown in [Figure 5-11](#).

Figure 5–11 The Total Cell Properties

On Total Cell **Formatting** properties tab you can:

- Edit the font properties of the total cell
- Edit the cell properties of the total cell including border weight, style, and color and background fill color
- Set the vertical and horizontal alignment of the cell contents
- Apply formatting to the cell data
- Apply a formula to the cell
- Apply conditional formatting to the cell

On Total Cell **Data** properties tab you can:

- Apply a formula to the data

5.1.5.1 Formatting Total Cell Data

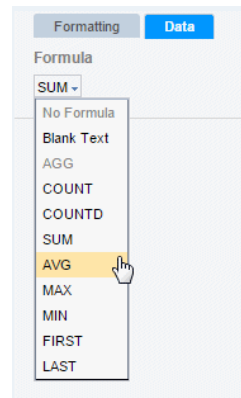
See [Section 5.1.4.3, "Setting Table Data Formatting Options."](#)

5.1.5.2 Applying a Formula

By default, the formula applied to a Total Cell within a numeric column is a sum of the column items. The Formula option enables you to apply a different formula.

On the Total Cell Properties **Data** tab, you can select from a list of alternative formulas.

Figure 5–12 *Formula selection list for a Table Total Cell*



5.1.6 Inserting Links to Table Column Data

See [Section 5.5, "Adding Static or Dynamic Links."](#)

5.2 Working with Pivot Tables

This section includes the following topics about working with pivot tables in the Mobile App Designer:

- [Customizing the Pivot Table](#)
- [Customizing Pivot Table Headers](#)
- [Customizing Pivot Table Data](#)

5.2.1 Customizing the Pivot Table

Use the Pivot Table properties to:

- Apply filters to the pivot table data
- Customize the display of total rows

5.2.1.1 Applying Filters

See [Section 3.20, "Global Filters"](#) for a description of adding filters to tables.

5.2.1.2 Customizing the Display of Totals

You can customize the display of grand total and subtotal rows.

By default, the pivot table displays the grand total rows, but not subtotal rows:

- **Row Grand Total** - Inserted at bottom of table
- **Row Subtotal** - Inserted at top of each subgroup, with no row header
- **Column Grand Total** - Inserted at the far right
- **Column Subtotal** - Inserted to the left of each column subgroup, with no header

Change the positioning and display of totals and subtotals by clicking the appropriate tab in the properties and selecting the desired properties.

5.2.2 Customizing Pivot Table Headers

Select the column or row header of the pivot table and use the Pivot Table Header properties to perform the following:

- Customize the fonts, colors, alignment and other display features of the header
- Apply a sort order (for more information see [Section 5.1.4.7, "About the Sort Option"](#))
- Apply data formatting for numbers and dates (see [Section 5.1.4.3, "Setting Table Data Formatting Options"](#))

5.2.3 Customizing Pivot Table Data

Select the data area of the pivot table and use the Pivot Table Data properties to perform the following actions. The commands in the Pivot Table Data properties are the same as the corresponding commands in the table Column properties. See the references for more information on their use.

- Customize the fonts, colors, alignment and other display features of the data
- Apply conditional formatting to the data (see [Section 5.1.4.1, "Applying Conditional Formats to a Table Column"](#))
- Apply data formatting (see [Section 5.1.4.3, "Setting Table Data Formatting Options"](#))
- Apply a formula (see [Section 5.1.5.2, "Applying a Formula"](#))

5.3 Adding Filters to Tiles, Charts, Tables, and Pivot Tables

Apply a filter to refine the items displayed in tiles, charts, tables, and pivot tables. For example, apply a filter to:

- Display only the top 10 salaries
- Display only the bottom 25 store sales
- Display only employees in the IT department
- Display only sales that are between \$10,000 and \$20,000 and in the Southern region

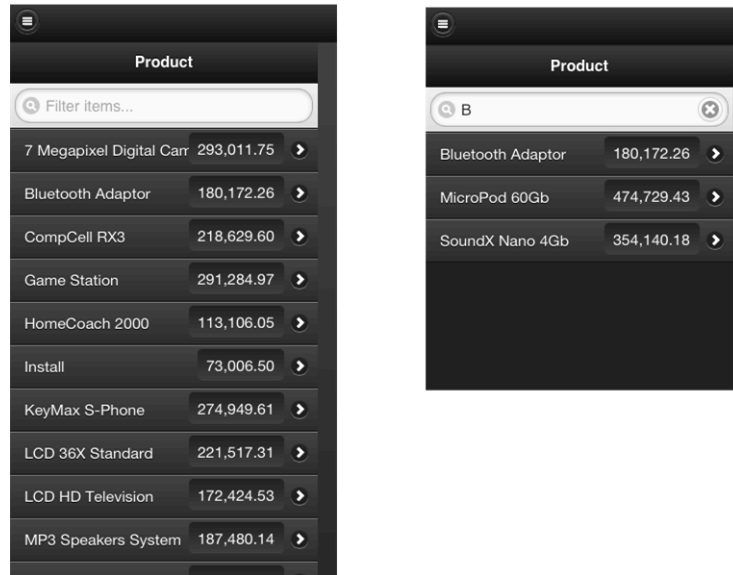
You can add multiple filters and manage the order in which they are applied to the page component. See [Section 3.20, "Global Filters."](#)

5.4 Adding a Search Filter to Navigation and Accordion Lists

If the column that you select to create a navigation list or accordion list has many items, you can add a search filter to enable users to find specific items in the list more quickly. [Figure 5–13](#) shows the data search filter option enabled for a navigation page.

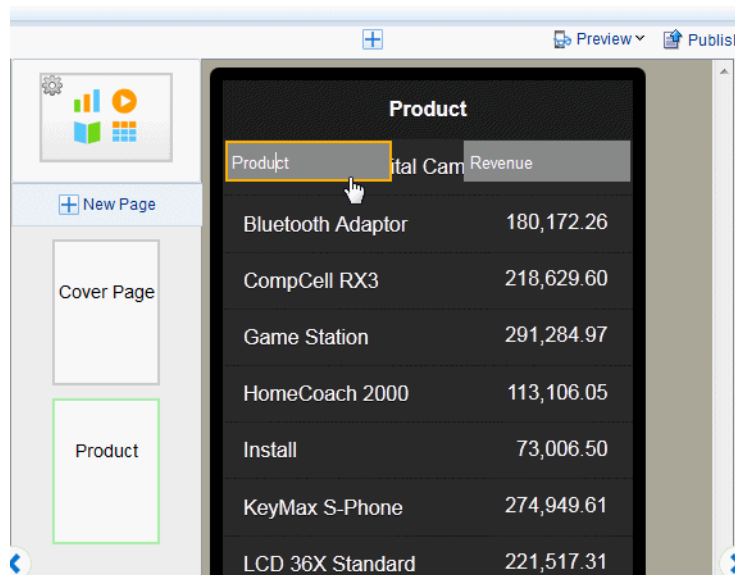
Figure 5–13 Example of Data Filter for a Navigation List

Data Filter provides a text entry field to filter items in the list

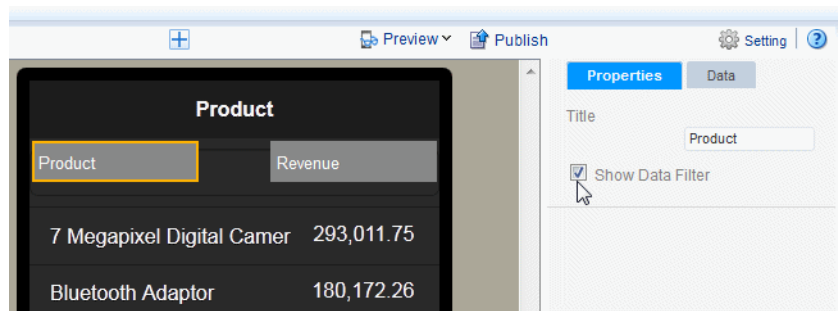


To add a search Data Filter for a Navigation or Accordion list:

1. Select the **Label** field on your Navigation or Accordion page. [Figure 5–14](#) shows selecting the Label field of a Navigation page.

Figure 5–14 Selecting the Label Field

2. On the Properties pane, set the **Show Data Filter** property to True.



Note that the data filter does not display during design time. To see how the data filter displays at run time, **Preview** the app.

5.5 Adding Static or Dynamic Links

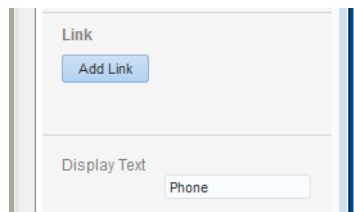
You can add links to Data Field components or table Columns. Links can be to any of the following:

- Web page - tapping the linked item in the app page opens a Web page in your mobile device browser.
- E-mail - opens the mobile device mail app and populates the To: field with the e-mail address provided in the link.
- Phone - when tapped on a phone, displays an alert to initiate dialing the number.
- SMS - launches the message app to initiate a text message to the number.

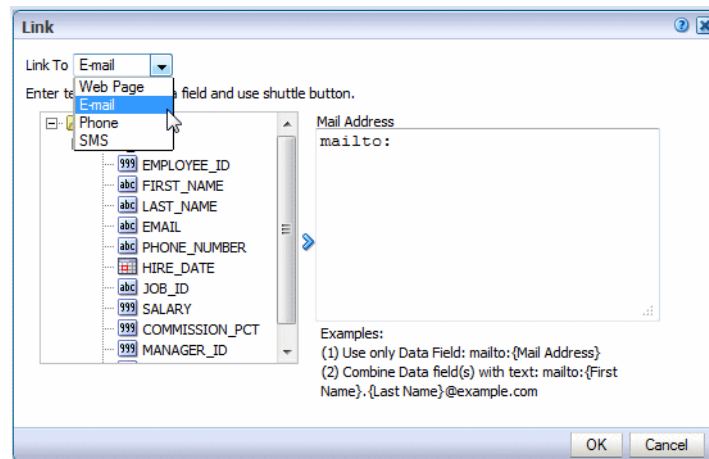
To add links:

1. Select the Data Field or the table Column in the app page for which you want to add a link to display the properties. [Figure 5–15](#) shows the Add Link option from the Properties pane.

Figure 5–15 Link Command Shown in the Properties Pane

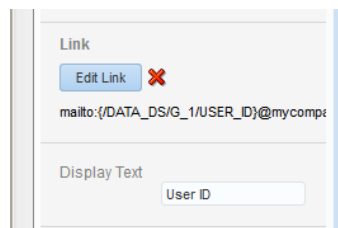


2. In the **Link** dialog, choose the type of link to add from the **Link To** menu. When you make a selection, the region where you define the link content displays the appropriate prompts.



3. Specify the link to associate with the data field or column data. You can specify a static link or build a dynamic link using fields from your data. You can combine static text with data fields. To use a data field in your link definition, select the data field from the list and use the shuttle button to move it to the link definition.

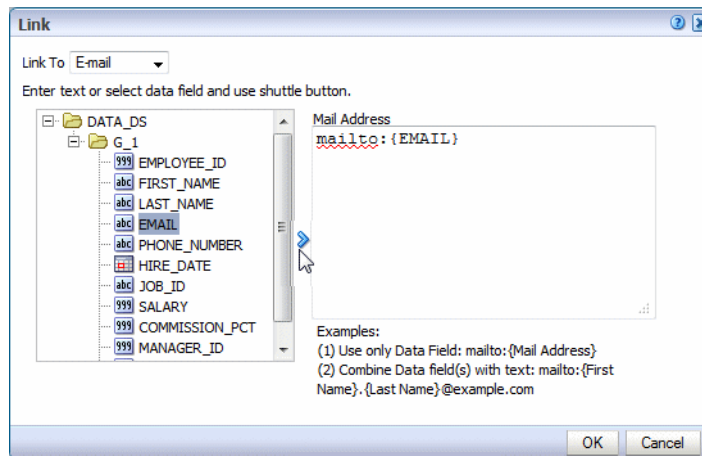
When you click OK, the link displays in the **Link** region of the Properties pane.



Example: Create E-Mail Link with Data Field Source

Assume your app includes a table of employees. Your app data includes e-mail address as a data field. To add the e-mail link to each employee name:

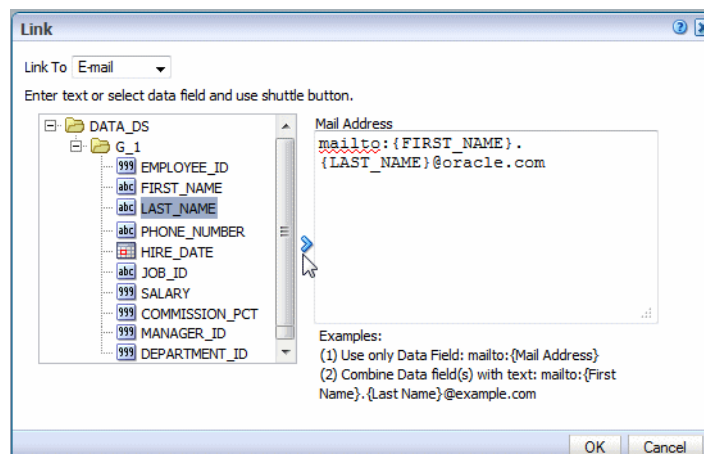
1. Select the table column to enable the **Column** properties
2. Click the **Link** command.
3. In the **Link** dialog, select E-mail from the **Link To** list.
4. Position your cursor to the right of `mailto:`:
5. Select the data field that contains the e-mail address and click the shuttle button to move it to the link definition area. as shown in [Figure 5-16](#).

Figure 5–16 Defining E-Mail Link Using a Data Field**Example: Create E-Mail Link from Combination Static Text and Data Field Source**

Now assume you want to include the e-mail address for each employee, but the data does not contain the e-mail address field. Instead, you must construct the e-mail address from a combination of the FIRST_NAME and LAST_NAME fields and static text.

To construct the e-mail link:

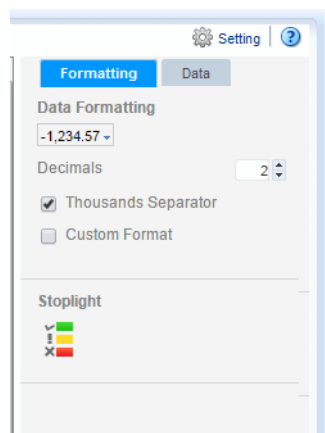
1. Select the table column to enable the **Column** properties
2. Click the **Link** command.
3. In the **Link** dialog, select E-mail from the **Link To** list.
4. Position your cursor to the right of **mailto:**
5. To construct the link:
 - Select the data field that contains FIRST_NAME and click the shuttle button to move it to the link definition area.
 - Enter a period.
 - Select the data field that contains LAST_NAME and click the shuttle button to move it to the link definition area.
 - Enter the domain: @example.com

Figure 5–17 Using Data Fields and Static Text to Define Link

5.6 Features of Metric Fields

Metric fields are fields that calculate a measure, such as sum of revenue or count of orders. Examples of metric fields are the aggregated data fields you define for navigation and accordion pages.

When you select the measure field on an app page, the metric field properties enable you to customize display and calculation options. [Figure 5–18](#) shows the Metric Field properties.

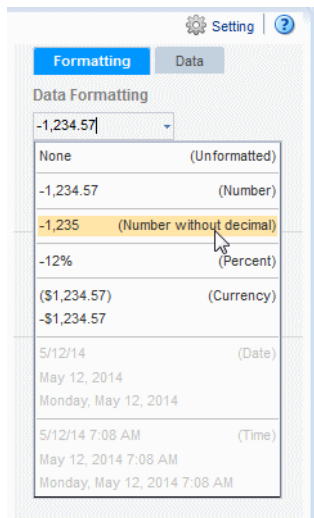
Figure 5–18 The Metric Field Properties

The properties enable you to:

- [Apply Data Formatting](#)
- [Apply Custom Data Formatting](#)
- [Add Stoplight Formatting](#)

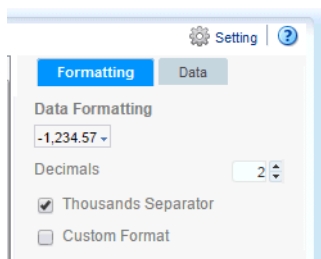
5.6.1 Apply Data Formatting

- **Data Formatting** - Select one of the common number formats from the list. The format is applied immediately to the field. The formats are categorized by Number, Percent, and Currency, as shown in [Figure 5–19](#).

Figure 5–19 Number, Percent, and Currency Formats

To apply a format not available from this list, see [Section 5.6.2, "Apply Custom Data Formatting."](#)

- **Decimal position** - Enter the number of decimal positions to show to the right of the decimal point.

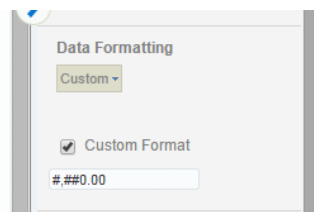


- **Show/Hide Thousands Separator** - Enable this checkbox to show the grouping separator (for example, 1234.00 displays as 1,234.00).

5.6.2 Apply Custom Data Formatting

To apply custom data formatting:

1. Select the data field or table column. to display its properties.
2. Under Data Formatting, select **Custom Format**
3. Enter the format mask in the text box. For example, for a column that contains product totals, you can enter `#,###.00` formatting mask to display total values with two zeros to the right of the decimal place. See [Custom Number and Date Formats](#) for the list of supported formats.



5.6.3 Update the Formula

To update the formula, click the Data tab. The formula option is available for several components including:

- Column
- Table Total Cell
- Chart Measure Field
- Pivot Table Data

Not all options are applicable to each component type.

5.6.3.1 Predefined Formulas Available from the Menu

The menu provides the predefined formulas described in [Table 5–2](#).

Table 5–2 *Predefined Formulas*

Formula	Description
No Formula	Removes any mathematical formula from a numeric column.
Blank Text	Removes the display of any data, text or numeric.
Count	Returns the count of the number of occurrences of the element in the current group.
Count Distinct	Returns a count of the distinct values of an element in the current group.
Summation	Sums the values of the element in the current group.
Average	Displays the average of the values in the current group.
Maximum	Displays the highest value of all occurrences in the current group.
Minimum	Displays the lowest value of all occurrences in the current group.

For non-numeric data, only the following formula options are supported:

- Blank Text
- Count
- Count Distinct

5.6.3.2 Applying Custom Formulas

To apply a custom formula to a field, use the Calculated Fields feature. For more information, see [Section 3.19, "Calculated Fields."](#)

5.6.4 Add Stoplight Formatting

For details on adding stoplight formatting, see [Section 3.10.3, "Stoplight Formatting for Navigation and Accordion Pages."](#)

5.7 Setting Borders

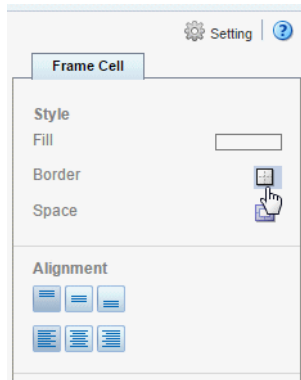
Several components enable setting a border around the item. For example:

- Table columns
- Table column headers

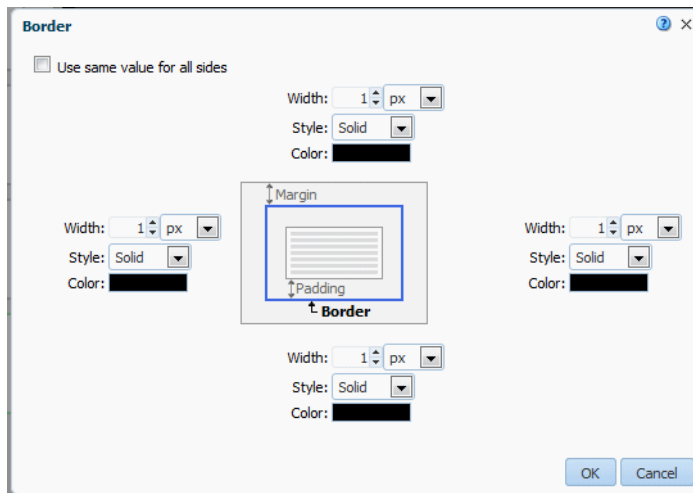
- Table total cells
- Frame cells
- Text items
- Lists

To set the border around a component:

1. Select the component to display its properties.
2. Click the **Border** icon.



3. In the **Border** dialog set the following values for each border:
 - **Width** - enter a value in pixels (px), points (pt) inches (inch), or centimeters (cm)
 - **Style** - select from Solid, Double, Dotted, or Dashed
 - **Color** - click the color to launch the color picker



To use the same values for all borders, select **Use same value for all sides** and set the values for the top border only.

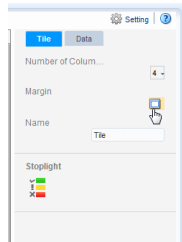
4. Click **OK**.

5.8 Setting Margins

Several components allow you to customize the margins around the component, these include tiles, tables, and charts.

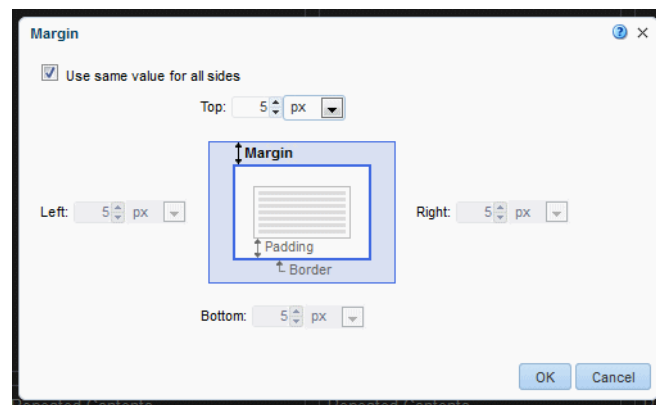
To set margins:

1. Select the component to display its properties.
2. Click the **Margin** tool.



3. Enter the new margin values and select the unit of measurement.

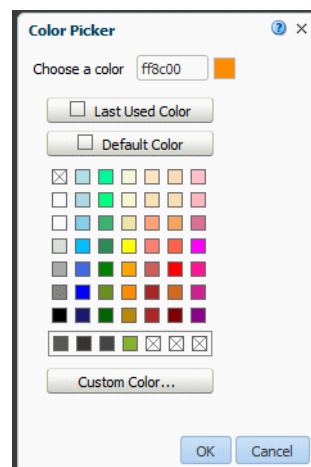
Figure 5–20 Customizing Tile Margins



4. Click OK.

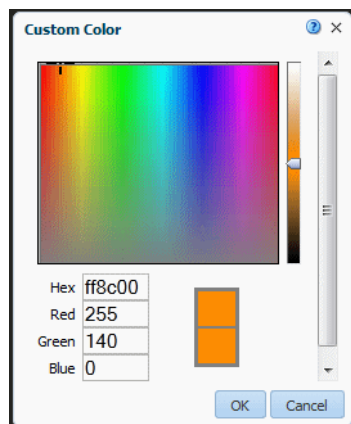
5.9 Using the Color Picker

The **Color Picker** enables you to change the color of the selected item.



You can select a new color in any of the following ways:

- Enter the hexadecimal value for the color in the **Choose a color** text box.
- Select the **Last Used Color**.
- Select the **Default Color**.
- Choose one of the common colors presented or recently used colors.
- Click **Custom Color** to define a custom selection.



To define a custom color:

1. Drag your mouse over the color palette to the color desired; or, enter the Red, Green, and Blue color model values.
2. Use the slider bar to increase or decrease the color saturation.
3. Click **OK**.

5.10 Custom Number and Date Formats

Mobile App Designer supports a set of number and date format masks to enable you to customize how your data displays.

- [Supported Number Format Mask Definitions](#)
- [Supported Date Format Mask Definitions](#)

5.10.1 Supported Number Format Mask Definitions

[Table 5–3](#) lists the supported number format mask definitions.

Table 5–3 Supported Number Format Mask Definitions

Symbol	Location	Meaning
0	Number	Digit. Each explicitly set 0 appears, if no other number occupies the position. Example: Format mask: 00.0000 Data: 1.234 Display: 01.2340

Table 5–3 (Cont.) Supported Number Format Mask Definitions

Symbol	Location	Meaning
#	Number	Digit. When set to #, only the incoming data is displayed. Example: Format mask: ##.#### Data: 1.234 Display: 1.234
.	Number	Determines the position of the decimal separator. The decimal separator symbol used is determined at runtime based on locale. Example: Format mask: ##.##0.00 Data: 1234.56 Display for English locale: 1,234.56 Display for German locale: 1.234,56
-	Number	Determines placement of minus sign for negative numbers.
,	Number	Determines the placement of the grouping separator. The grouping separator symbol used is determined at runtime based on locale. Example: Format mask: ##.##0.00 Data: 1234.56 Display for English locale: 1,234.56 Display for German locale: 1.234,56
E	Number	Separates mantissa and exponent in a scientific notation. Example: 0.###E+0 plus sign always shown for positive numbers 0.###E-0 plus sign not shown for positive numbers
;	Subpattern boundary	Separates positive and negative subpatterns. See the Note that follows the table.
%	Prefix or Suffix	Multiply by 100 and show as percentage
'	Prefix or Suffix	Used to quote special characters in a prefix or suffix.

Note: Subpattern boundary: A pattern contains a positive and negative subpattern, for example, "#,##0.00;(##,##0.00)". Each subpattern has a prefix, numeric part, and suffix. The negative subpattern is optional. If absent, the positive subpattern prefixed with the localized minus sign ("-") in most locales) is used as the negative subpattern. That is, "0.00" alone is equivalent to "0.00;-0.00". If there is an explicit negative subpattern, it serves only to specify the negative prefix and suffix. The number of digits, minimal digits, and other characteristics are all the same as the positive pattern. That means that "#,##0.0#;(##,##0.0#)" produces precisely the same behavior as "#,##0.0#;(##,##0.0#)".

5.10.2 Supported Date Format Mask Definitions

Table 5–4 lists the supported Mobile App Designer date format mask definitions.

Table 5–4 Supported Date Formats

Symbol	Meaning
d	The day of the month. Single-digit days do not have a leading zero.
dd	The day of the month. Single-digit days have a leading zero.
ddd	The abbreviated name of the day of the week.
dddd	The full name of the day of the week.
M	The numeric month. Single-digit months do not have a leading zero.
MM	The numeric month. Single-digit months have a leading zero.
MMM	The abbreviated name of the month.
MMMM	The full name of the month.
yy	The year without the century. If the year without the century is less than 10, the year is displayed with a leading zero.
yyyy	The year in four digits.
gg	The period or era. This pattern is ignored if the date to be formatted does not have an associated period or era string.
h	The hour in a 12-hour clock. Single-digit hours do not have a leading zero.
hh	The hour in a 12-hour clock. Single-digit hours have a leading zero.
H	The hour in a 24-hour clock. Single-digit hours do not have a leading zero.
HH	The hour in a 24-hour clock. Single-digit hours have a leading zero.
m	The minute. Single-digit minutes do not have a leading zero.
mm	The minute. Single-digit minutes have a leading zero.
s	The second. Single-digit seconds do not have a leading zero.
ss	The second. Single-digit seconds do have a leading zero.
f	Displays seconds fractions represented in one digit.
ff	Displays seconds fractions represented in two digits.
fff	Displays seconds fractions represented in three digits.
ffff	Displays seconds fractions represented in four digits.
fffff	Displays seconds fractions represented in five digits.
ffffff	Displays seconds fractions represented in six digits.
fffffff	Displays seconds fractions represented in seven digits.
tt	The AM/PM designator.
z	Displays the time zone offset for the system's current time zone in whole hours only. (This element can be used for formatting only)
zz	Displays the time zone offset for the system's current time zone in whole hours only. (This element can be used for formatting only)
zzz	Displays the time zone offset for the system's current time zone in hours and minutes.
:	The default time separator.
/	The default date separator.

Table 5–4 (Cont.) Supported Date Formats

Symbol	Meaning
'	Quoted string. Displays the literal value of any string between two ' characters.
"	Quoted string. Displays the literal value of any string between two " characters.

Configuring App Settings

This chapter describes how to update data sources and how to configure the app settings such as parameters, caching, and translations.

It includes the following sections:

- [Updating or Refreshing the Data Source](#)
- [Accessing App Settings](#)
- [Setting App Caching Properties](#)
- [Configuring Parameters](#)
- [Adding Translations](#)

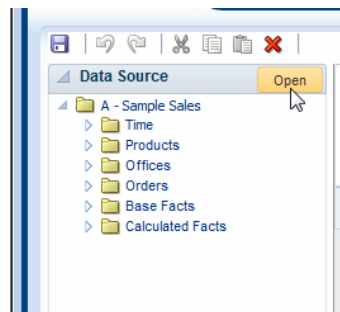
6.1 Updating or Refreshing the Data Source

- [Add Multiple Subject Areas to a Data Source](#)
- [Refresh Excel Data](#)
- [Update a BI Publisher Data Model](#)

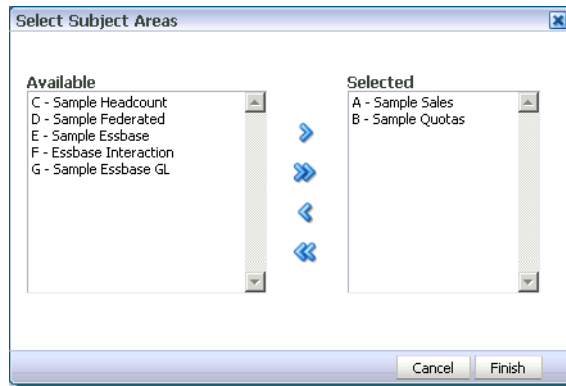
6.1.1 Add Multiple Subject Areas to a Data Source

You can add multiple subject areas to your app data source:

1. In the **Data Source** pane, click **Open**.



2. In the Select Subject Areas dialog, from the **Available** list, select one or more subject areas.

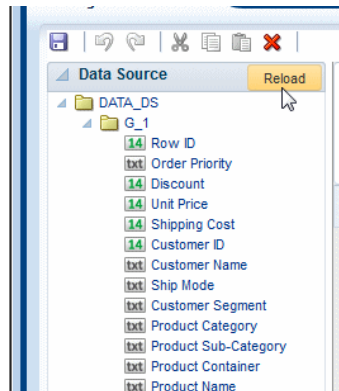


3. Click **Move** to move the subject areas to the **Selected** list.
4. Click **Finish**.

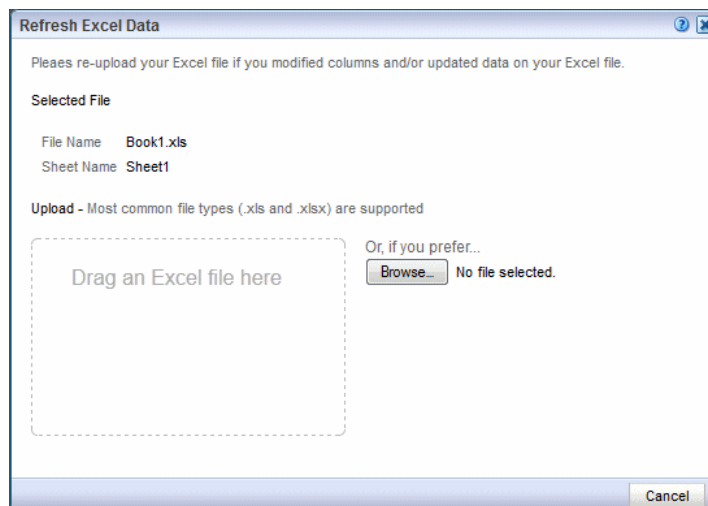
6.1.2 Refresh Excel Data

To refresh or re-upload your Excel data source:

1. In the **Data Source** pane, click **Reload**. (This option is enabled only when the data source is Excel.)

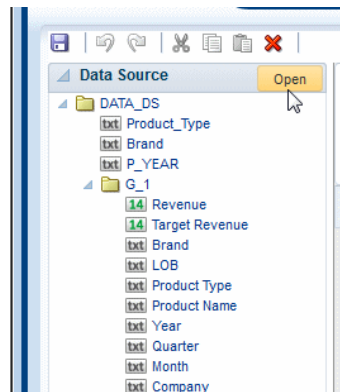


2. In the **Refresh Excel** dialog, either drag the new file from your desktop and drop it on the dialog, or click browse to locate the file in your file system.



6.1.3 Update a BI Publisher Data Model

1. In the **Data Source** pane, click **Open** to launch the BI Publisher data model editor.

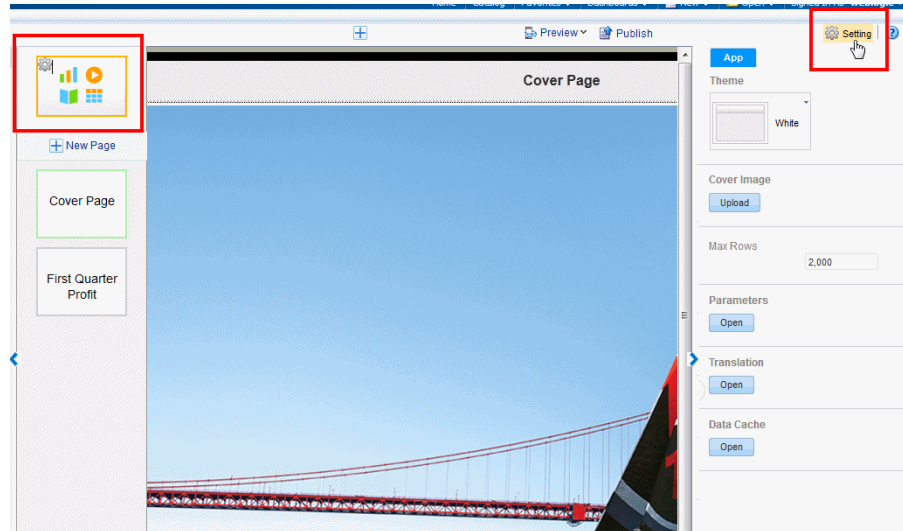


2. Update the data model following the guidelines in the *Oracle Fusion Middleware Data Modeling Guide for Oracle Business Intelligence Publisher*.
3. Click **Return** to return to the Mobile App Designer.

6.2 Accessing App Settings

There are two ways to access the app-level properties and settings:

- Click the app icon
- Click the **Setting** button



See the following sections for tasks you can perform from the **Setting** menu:

- [Configuring Parameters](#)
- [Setting App Caching Properties](#)
- [Adding Translations](#)

6.3 Configuring Parameters

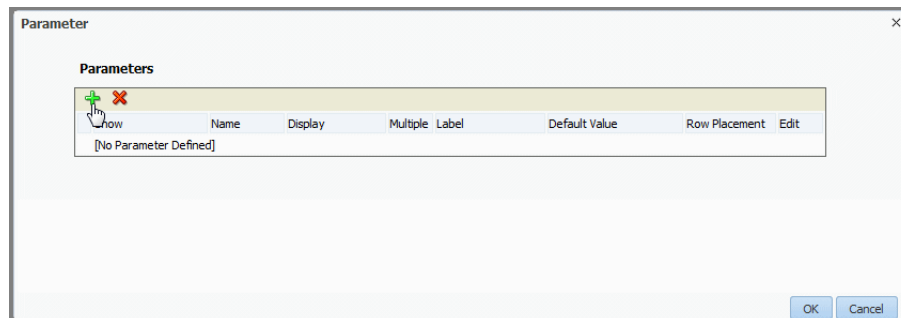
How you configure parameters depends on whether your data source is a BI Publisher data model or a BI subject area.

- [Configure Parameters for a BI Subject Area Data Source](#)
- [Configure Parameters for a BI Publisher Data Model Data Source](#)

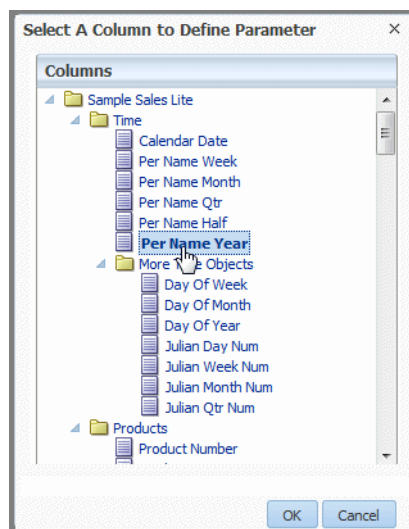
6.3.1 Configure Parameters for a BI Subject Area Data Source

To add parameters to a subject area data source:

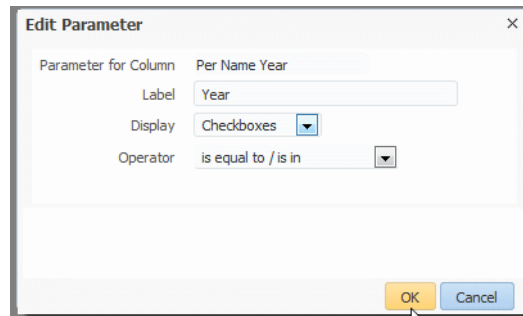
1. In the **Setting** properties, under **Parameters** click **Open**.
2. In the Parameters dialog, click **Add**.



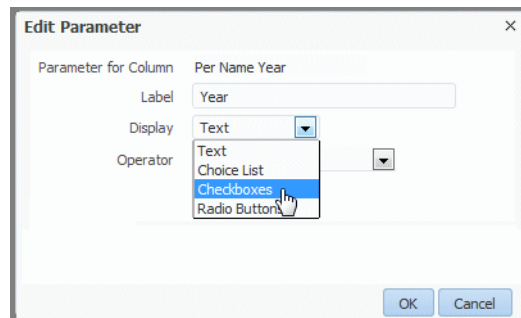
3. In the **Select a Column to Define Parameter** dialog, select the column to define the parameter and click **OK**. In this example, the **Per Name Year** column is chosen to create the parameter.



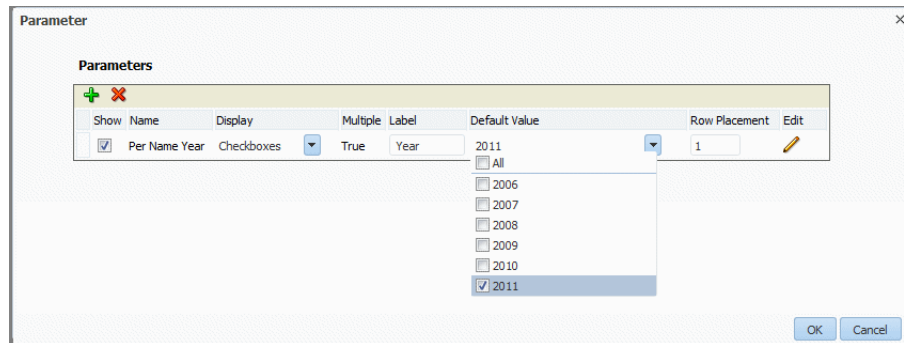
4. Define the parameter settings in the **Edit Parameter** dialog.



5. Enter the label to display for the parameter in the app.
6. Select a parameter display option.



- **Calendar** - Provides users with a field into which they can enter a specific date, as well as a calendar pop-up to select a date. This display option is only available for selection if the parameter has a date column type.
 - **Choice List** - Provides users with a collapsed list of all prompt values. This display option is useful for a long list of values where you want to provide the user with the ability to search for a specific value.
 - **Checkboxes** - Provides users with a visible list of all prompt values where a small, selectable box displays before each value item. This display option is suitable for a prompt that contains a smaller set of data.
 - **Radio Buttons** - Provides users with a visible list of all prompt values where a radio button is displayed before each prompt value. This display option is useful for short lists of values where the user is to select only one prompt value.
 - **Text** - Provides users with a field into which they can enter a specific prompt value. This display option cannot be used for multiple prompt values. Only the field and the field label are displayed for this option.
7. Click **OK**.
 8. In the Parameters dialog, enter a default value for the parameter.



9. Click **OK**.

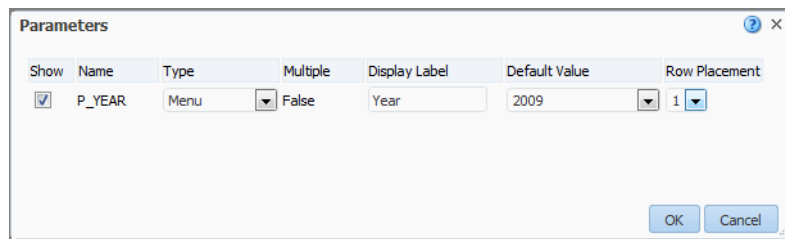
6.3.2 Configure Parameters for a BI Publisher Data Model Data Source

Parameters are defined in the BI Publisher data model. To add a new parameter, see [Section 6.1.3, "Update a BI Publisher Data Model."](#)

The app editor enables you to configure the parameter settings specifically for each app that uses the data model.

To configure the parameters for this app:

1. In the **Setting** properties, under **Parameters** click **Open**.



2. Customize the parameter settings for this app by making selections for the following display options:

Show

This property controls whether the parameter is displayed to the user. Disable the **Show** property if you do not want the user to see or change the parameter values that are passed to the data model.

Type

This property is customizable for menu type parameters only. For menu type parameters, the following display options are available:

- **Menu** - Provides users with a collapsed list of all prompt values. This display option is useful for a long list of values where you want to provide the user with the ability to search for a specific value.
- **Checkboxes** - Provides users with a visible list of all prompt values where a small, selectable box displays before each value item. This display option is suitable for a prompt that contains a smaller set of data.
- **Radio Buttons** - Provides users with a visible list of all prompt values where a radio button is displayed before each prompt value. This display option is useful for short lists of values where the user is to select only one prompt value.

Multiple

This property indicates whether multiple values may be selected for a menu parameter. This property is defined in the BI Publisher data model and is display only here.

Display Label

Use this property to edit the display labels shown for each parameter. The default values are defined in the data model.

Default Value

Use this property to configure the default value. Choose "Default" to pass the default value defined in the data model.

Row Placement

Not used.

6.4 Setting App Caching Properties

For BI Publisher and Excel data sources, you can [Enable Data Caching](#).

6.4.1 Enable Data Caching

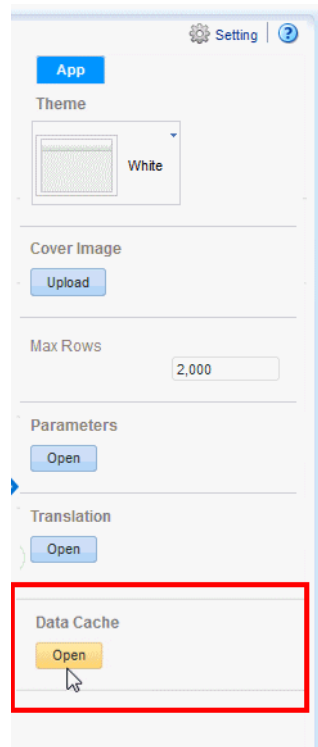
When data caching is enabled, the data generated by the app is stored in the cache. Subsequent requests to view this app with the same parameter selections display the app using the data that is stored in the cache. This setting enhances performance by using stored data to generate the app rather than regenerating the data from the source. The data remains in the cache according to the time limit that is specified in the Cache Duration property.

When you enable caching you can choose between two types of caching:

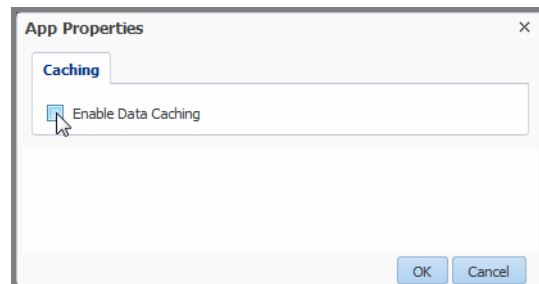
- **User Level cache** - stores a separate cache for each user. The data shown to each user comes only from the private cache. When enabled, this property ensures that each user can only see data that they are authorized to view. Be aware that user-level cache has less efficient performance. If the data is not user sensitive, you can disable this property to enhance performance.
- **Document cache** - stores the app in cache. When a user views the app, the document (data plus app layout) is placed in the cache. When any other user views the same app with the same parameter selections, the app is retrieved from the cache. The app document remains in the cache according to the caching duration specified.

To configure caching for your app:

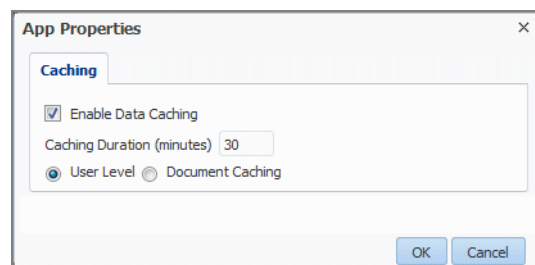
1. On the **Setting** menu select **Data Cache**.



2. On the **App Properties** dialog, select **Enable Data Caching**.



3. Choose **User Level** or **Document Caching** and enter the length of time in minutes for the data to remain in the cache.



4. Click **OK**.

6.5 Adding Translations

This section describes translation support in the Mobile App Designer. It includes the following topics:

- [What Is App Translation?](#)

- [Downloading a Translation File](#)
- [Translating the XLIFF File](#)
- [Uploading the Translated File](#)

6.5.1 What Is App Translation?

App translation is a feature of Mobile App Designer that enables you to extract the translatable strings from an app layout into an industry-standard XLIFF translation file.

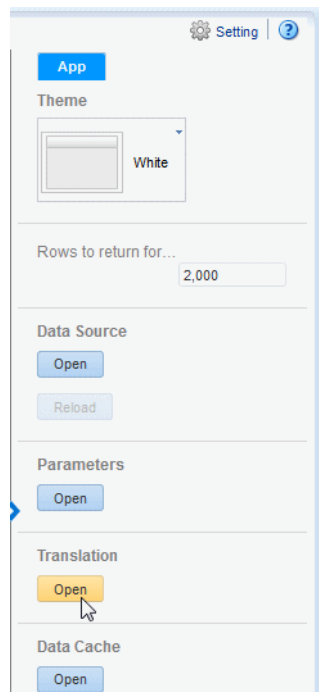
You can translate these strings within your organization or send the file to a localization provider. You then upload the translated XLIFF file back to the app and assign it the appropriate locale.

At run time, when a user's Locale preference is set to the locale of an available XLIFF translation file, the translated strings from the XLIFF are applied to the app and the user sees the translations of the text strings appropriate for his locale.

6.5.2 Downloading a Translation File

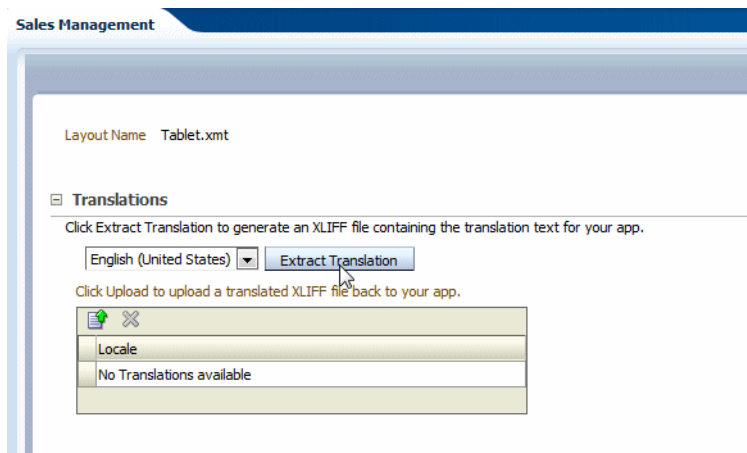
To generate and download the XLIFF file for an app:

1. In the **Setting** properties, under **Translation** click **Open**.



2. In the **Translations** region, click **Extract Translation**.

BI Mobile App Designer extracts the translatable strings from the template and exports them to an XLIFF (.xlf) file.



3. When prompted, save the XLIFF file to a local directory.

6.5.3 Translating the XLIFF File

After downloading the XLIFF file, you can send it to a translation provider, or using a text editor, you can enter the translation for each string. For information about how to edit an XLIFF file, see the chapter "Translation Support Overview and Concepts" in the *Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher*.

A "translatable string" is any text in the app that is intended for display, such as table headers and field labels. Text supplied at run time from the data is not translatable.

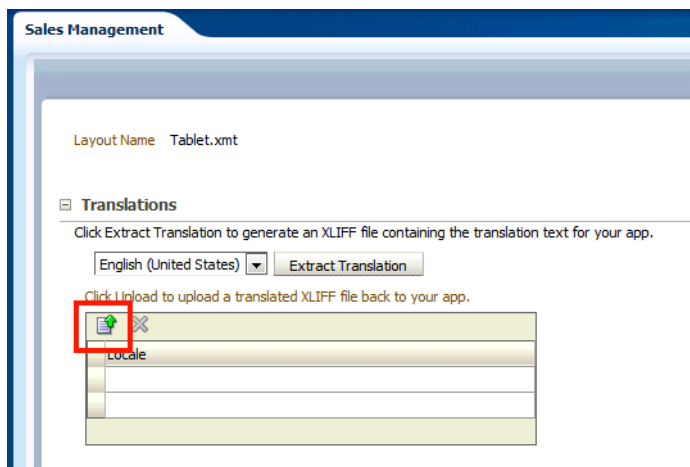
You can translate the app XLIFF file into as many languages as desired and associate these translations to the original app for multiple language support in a single app.

Important: Ensure that when you save your translated file, you save it with UTF-8 encoding.

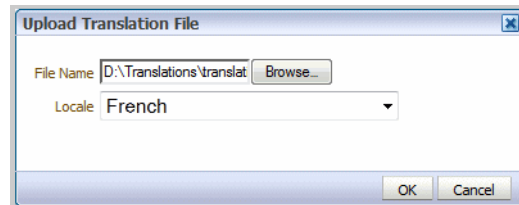
6.5.4 Uploading the Translated File

To upload the translated XLIFF:

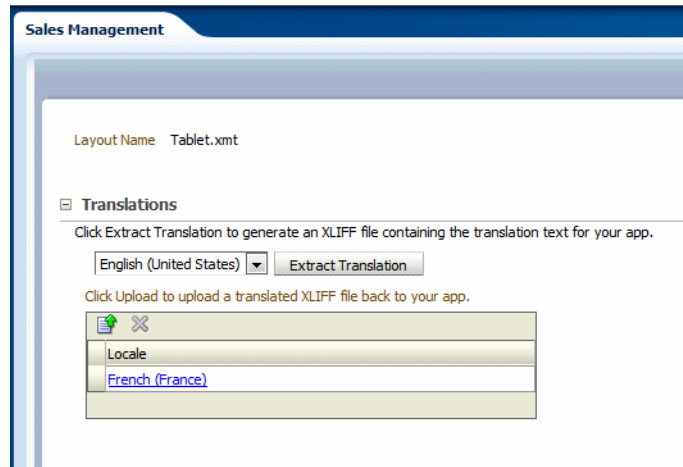
1. Open the app for editing and select **Translation** from the **Settings** menu.
2. In the **Translations** region, click the **Upload** toolbar button.



3. In the **Upload Translation File** dialog locate the file in a local directory and select the **Locale** for this translation.



4. Click OK to upload the file and view it in the **Translations** table.



Now users whose locale setting matches the locale assigned to this translation see the translated strings when viewing this app.

Adding Custom Plugins

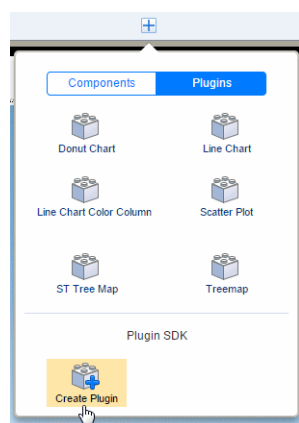
This chapter describes how to extend the BI Mobile App Designer functionality using custom or third-party plug-ins to integrate content generated by other applications into BI Mobile Apps.

This chapter includes the following sections:

- [What Are BI Mobile App Designer Plug-in Extensions](#)
- [Exploring the Sample Plugins](#)
- [Setting Up Your Development Environment](#)
- [Coding the Custom Plug-in](#)
- [Accessing Data at Run Time](#)
- [Testing Your Plug-in](#)
- [Preparing Your Plug-in for Upload](#)
- [Uploading to the Plug-in Gallery](#)

7.1 What Are BI Mobile App Designer Plug-in Extensions

BI Mobile App Designer supports adding JavaScript plug-in extensions to the app editor to add custom components to your apps. The custom components are then included when you run the app. After you add a plug-in, during design time, an icon for your plug-in displays in the layout editor's **Insert** menu. You can then drag and drop the custom component to the app page as you do any other app component.



The Mobile App Designer Plug-in SDK contains an environment to develop Oracle BI Mobile App Designer plugin components. The kit contains the plug-in javascript file,

the development server, and sample data. You can develop and test a plug-in from your desktop.

After you install the prerequisite software and set up your environment, you are ready to download the SDK with sample data. After you download the SDK you are ready to design your plug-in. You can test your plug-in on your desktop before deploying to your environment.

After testing, upload your plugin to the Plugin Gallery and it is immediately available within the Mobile App Designer.

7.2 Exploring the Sample Plugins

Sample plugins are provided with your BI Mobile App Designer installation. These samples demonstrate the custom functionality you can achieve by adding your own plug-in components. Explore these samples to inspire you to design your own.

For more details, see:

- <http://www.oracleappdesigner.com/2015/07/introduction-to-line-bar-chart-plugin.html>
- <http://www.oracleappdesigner.com/2015/07/plugin-donuts-chart.html>

7.3 Setting Up Your Development Environment

This section describes the tasks required to set up your environment and the files included in the SDK.

- [Download and Install the Prerequisites](#)
- [Download the SDK with Sample Data](#)
- [Files Included in the SDK](#)

7.3.1 Download and Install the Prerequisites

To work with the SDK environment, you must install the following software. These steps are only required the first time you work with the SDK:

Node.js

Download Node.js from the Node.js website:

<http://nodejs.org/>

The Plug-in SDK was developed with version v0.10.18; ensure that you install v0.10.0 or later.

Grunt

After installing Node.js, run the following node package manager (npm) command to install the Grunt command:

```
$ (sudo) npm install -g lru-cache grunt-cli
```

If you are behind a firewall, then provide the https proxy server setting to run the command as follows:

For Linux or Mac:

```
$ (sudo) https_proxy=http://<your https proxy server>:<proxy port>/ npm install -g
```

```
lru-cache grunt-cli
```

For Windows:

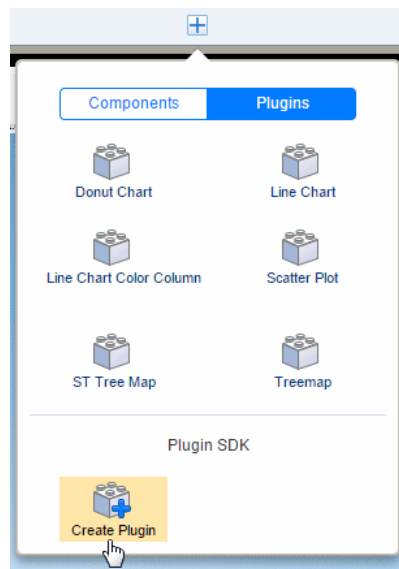
```
c:\> set HTTPS_PROXY=http://<https proxy server>:<proxy port>/
c:\> npm install lru-cache grunt-cli
```

7.3.2 Download the SDK with Sample Data

Now that your environment is set up, the next step is to download the SDK from the Mobile App Designer with some sample data. You download the SDK from within the Mobile App Designer. You can either create a new app or open an existing app. The key is to ensure that the data used by the app you choose is appropriate for the plug-in that you are designing:

To download the SDK:

1. Create a new app or open an app in edit mode that uses sample data appropriate for your plug-in. For example, if your plug-in creates a map visualization that requires country and city codes in the data source, ensure that the app data source includes these fields.
2. Click **Add Component** and choose **Plug-ins**.
3. Under **Plug-in SDK**, click **Create Plug-in**.



4. In the Create Plugin dialog, enter the following
 - Plugin Name - enter a name for this plug-in.
 - Label fields - click the + to select data fields from your data source to provide sample data for label fields
 - Measure fields - click the + to select data fields from your data source to provide sample data for measure fields. When you add a measure field, you must also select an aggregation formula. Sum, Count, and Average are supported.
 - Data Format - select the format for the sample data generation.

5. Click **bimadplugin.zip** to download the SDK to your local environment.

7.3.3 Files Included in the SDK

The following items are included in the SDK download:

Plugin file (<your_plugin_name>.js)

Your plugin file is located under the top level of the current directory. When you download the SDK from the Mobile App Designer, the data fields and data structure that you selected are coded in the Sample.js file. Use any text editor to finish coding the sample plug-in file. Once you start the development server, the changes are immediately applied.

Gruntfile.js

Gruntfile.js controls the development server.

You do not need to make any changes to this file unless you want to rename your plugin file or start with a different .js file in another location. To specify a new file name or location, rename the `pluginFile` variable value in the `gruntfile.js`.

data Folder (data/data01.csv)

The sample data that you chose during the download the step is put in this folder.

If you wish to test additional sample data with your plugin, you can put csv files in the data folder to use for testing. Although the BI Mobile App Designer supports data from various sources, in the SDK environment, the data must be provided through a csv file.

Note that each data file must contain the same data that your plugin specifies. If the plugin definition contains two data field entries, then the csv file must be two columns in the same order as the field entries.

This SDK server does not provide any sorting, reordering or regrouping at the time of serving data.

node_modules, package.json

These files are required to maintain the environment; do not edit or delete them.

assets

Use this folder to include any images, cascading style sheets (.css files), or Javascript files that your plug-in requires.

7.4 Coding the Custom Plug-in

This section provides the specification of the plug-in structure and describes the APIs provided for use with the plug-in. It contains the following topics:

- [Plug-in Structure](#)
- [JavaScript APIs That Can Be Used in Custom Plug-ins](#)

7.4.1 Plug-in Structure

The plug-in file is a simple JavaScript file. It has a basic structure that must be followed to ensure Mobile App Designer can use it. The basic components are:

Table 7–1 Overview of the plug-in file components

Component	Description
id	(Required) Unique ID for the plug-in
component	(Required) Comprises the plug-in name, tooltip and icon
remoteFiles	(Optional) Supported .css and JavaScript files
properties	(Required) The plug-in's defined properties, for example, the fields and data that are used for rendering
fields	The data fields you selected from the data source
dataType	Defines the data structure based on your choice of d3hierachy, CSV, or arrayOfArrays
render	(Required) the main function of the plug-in
refresh	(Optional) Function called to refresh the data when data changed by click event

Following is a basic sample plug-in file:

```
{
  id: 'com.oracle.xdo.Sample', //Change this default id to unique value before
development
  component: {
    'name': 'Sample',
    'tooltip': 'Insert Sample'
  },
  properties: [
    {key: "width", label: "Width", type: "length", value: "500px"},
    {key: "height", label: "Height", type: "length", value: "400px"}
  ],
  //These are the fields you selected from your data source
  fields: [
    {name: "Brand", caption: "Drop Brand Field Here", fieldType: "label",
dataType: "string"},
    {name: "LOB", caption: "Drop LOB Field Here", fieldType: "label", dataType:
"string"},
    {name: "Product Type", caption: "Drop Product Type Field Here", fieldType:
"label", dataType: "string"},
    {name: "Revenue", caption: "Drop Revenue Field Here", fieldType: "measure",
```

```
dataType: "number", formula: "summation"}
  ],
  //dataType can be d3hierarchy or 'arrayOfArrays' depending on your selection
  dataType: 'd3hierarchy',
  render: function (context, containerElem, data, fields, props) {
    containerElem.innerHTML = '<h1>My First Plugin</h1>';

    var tableStr = '<div id="'+this.id+'_table">';
    tableStr += '<table>';

    data.forEach(function(row, rowNo, rows) {
      tableStr += '<tr>';
      row.forEach(function(col, colNo, cols) {
        tableStr += '<td>'+col+'<td>';
      });
      tableStr += '<tr>';
    });
    tableStr += '</table>';
    tableStr += '</div>';

    containerElem.innerHTML += tableStr;
  },
  refresh: function (context, containerElem, data, fields, props) {
    var tableStr = '<table>';

    data.forEach(function(row, rowNo, rows) {
      tableStr += '<tr>';
      row.forEach(function(col, colNo, cols) {
        tableStr += '<td>'+col+'<td>';
      });
      tableStr += '<tr>';
    });
    tableStr += '</table>';

    // update table with a bit of fancy transition
    $(document.getElementById(this.id+"_table")).html(tableStr).hide().fadeIn(1000);
  }
}
```

Following are more detailed descriptions of the JavaScript object fields:

id

The `id` is an identification string. Oracle recommends using the reverse domain name to avoid any naming conflicts, for example: "com.example.helloworld".

component

The following fields comprise the component object:

name

The name of the component. Example: "Hello World"

icon

The icon is the image that displays in the app designer **Insert** menu to represent the plug-in. This field takes a URL that points to the icon image. Example: "http://www.example.com/img/smile.gif"

tooltip

The tooltip message to display for the icon image. Example: "Hello World Plugin".

description

Optional description of the plug-in. The description displays in the Plug-in Gallery.

author

Optional author entry. The author displays in the Plug-in Gallery.

cssClass

(Optional) Component CSS class selector to identify the plug-in components.

render

The render function renders the plug-in contents. The render function passes the following parameters:

context

Object containing the following information:

id

The `id` is an assigned instantiated component ID string. The system assures this ID is unique in the same template. Oracle recommends using this ID as a prefix or suffix to the HTML element that the plug-in code generates. This practice prevents ID conflicts.

reportLocale

The locale assigned to the app layout.

containerElem

The container HTML element. The contents must be set to this element. This provides the Mobile App Designer the pointer to where your plug-in visualization is to be rendered.

data

The variable that data is loaded into depending on the data structure you chose.

fields

See [Fields Support](#).

props

Current properties. See [Property Support](#).

7.4.2 Property Support

To add optional custom properties that display and can be set on the Properties pane of the App Designer, define them under the `properties` component. Array of property definition object can be set to this field. Construct the `properties` definition object from the following values.

key

A string value that specifies the property key. This value must be unique.

label

A string value that specifies the label displayed for this property in the Mobile App Designer Properties pane.

type

A string value that specifies the property type. The App Designer uses this value to open the appropriate editor to edit the property. The following values are supported for type:

- **string** - creates a text entry box to enter string data.
- **number** - creates a text entry box to enter numeric data.
- **bool** - creates a True/False (boolean) choice option.
- **length** - creates text entry box to enter length data and select units in px, in, cm, or pt.
- **color** - displays a color-chooser for color selection.
- **font** - displays the list of supported fonts for selection.
- **fontsize** - displays the font size selector.
- **lov** - creates a list of values. See [options](#) for creating the name-value pairs.

value

The initial value of the property. The value must follow the format of the type specified.

options

This parameter is valid only when the property **type** is "lov". The options parameter contains label-value pairs to define the list of values.

- **label** - the label for each list item
- **value** - the value for the label

7.4.2.1 Predefined Properties

The app editor sets the following property settings:

- **width**: set based on available space
- **height**: set based on available space
- **padding**: 0px 0px 0px 0px
- **margin**: 0px 0px 0px 0px
- **border-top**: 0px none #000000
- **border-left**: 0px none #000000
- **border-right**: 0px none #000000
- **border-bottom**: 0px none #000000

These properties can be updated within the app editor at design time.

7.4.3 Fields Support

The sample JavaScript file generated by the SDK includes the fields definitions for the fields that you specified in the download dialog.

These fields enable users to drag-and-drop data columns from the data source to your plug-in for custom visualizations of the data. The field information is specified in the **fields** component of the plug-in structure. The field definition consists of **name**, **caption**, **fieldType**, and **dataType**. A field can have a **fieldType** of **label** or **measure**. For measure fields, additionally specify a **formula**.

7.4.3.1 Defining the Data Fields

The following example shows data field definitions:

```
{
  id: "com.oracle.xdo...",
  component: {
    name: "Field Test"
  }
  fields:
  [
    {name: "labelField", caption: "Drop Label Field Here", fieldType:"label",
    dataType:"string"},
    {name: "dataField", caption: "Drop Data Field Here", fieldType:"measure",
    dataType: "number", formula: "summation"}
  ],
}
```

For each field the following is defined. You can update these attributes in the plug-in file.

- **name**
- **caption** - text that the app editor displays for the field. For example: "Drop Label Field Here".
- **fieldType** - valid values are "label" and "measure".
- **dataType** - the following data types are supported:
 - string (text string, default)
 - number (number, including integer and float)
 - date (XML date format)

The data type of the element that you drag and drop from the data model structure in the app editor must match the dataType defined here.

- **formula** - when the fieldType is "measure" specify a default formula. The following are supported:
 - count
 - count-distinct
 - summation
 - average
 - maximum
 - minimum
 - first
 - last

Field

Users can update Field properties in the App Designer's property editor. This information can then be accessed by the fields variable in the render function arguments at run time.

- **field** - path to the field in the data source
- **fieldFormula** - the field formula; must be null when fieldType="label"

- `fieldType` - the field type: "label" or "measure" (copied from the definition)
- `dataType` - the data type: "string", "number", or "date" (copied from the definition)
- `label` - the label string specified in the Property editor

7.4.4 JavaScript APIs That Can Be Used in Custom Plug-ins

The following JavaScript APIs are available to use in custom plug-ins:

- [handleClickEvent Method](#)

Captures tapped (selected) field information to send to the system.

- [getPixelValue Method](#)

Returns the pixel value from the length string value. The system uses 96 dots per inch (dpi), which is the same as most browsers.

7.4.4.1 handleClickEvent Method

This method captures the tapped (or selected) field information to send to the system.

Signature

`xdo.api.handleClickEvent(info)`

This method takes the following parameter:

info

Clicked field information object.

The structure of the object is:

Object Structure

```
{
  id: [component id],
  [
    {
      field: [xpath to the element],
      value: [filter value]
    },
    {
      field: [xpath to the element],
      value: [filter value]
    }
  ]
}
```

7.4.4.2 getPixelValue Method

This method returns the pixel value from the length string value. The system uses 96 dpi, which is the same as most browsers.

Signature

`xdo.api.getPixelValue(lengthString)`

The method takes one parameter:

lengthString

A string value that specifies the length. Supported units are "px", "pt", "in", and "cm".

7.5 Accessing Data at Run Time

Typically, calculated data is stored in the `rows` variable of the `render` function. To leverage D3.js libraries, use a JavaScript object.

- [Accessing Data Stored in rows Array](#)
- [Accessing Data Through a JavaScript Object](#)

7.5.1 Accessing Data Stored in rows Array

At runtime, calculated data is stored in the `rows` variable of the `render` function. The `rows` variable is an array type and each `rows` element has another array for keeping column information. The following `render` function implementation displays data in HTML:

```
render: function(context, containerElem, rows, fields, props) {
  // setup column
  var html = '<table>';
  for (var i=0, rowCount=rows.length; i<rowCount; i++)
  {
    html += "<tr>";
    var col = rows[i];
    for (var j=0, colCount=col.length; j<colCount; j++)
    {
      html += "<td>";
      html += col[j];
      html += "</td>";
    }
    html += "</tr>";
  }
  html += '</table>';
  containerElem.innerHTML = html;
}
```

7.5.2 Accessing Data Through a JavaScript Object

To support visualizations using the D3.js JavaScript library, pass data using the "d3hierarchy" datatype. With this data type, the `render` function is called with a JavaScript object instead of arrays of rows. The JavaScript object has `obj.node` and `obj.children` for node or `obj.value` and `obj.value` for leaf.

For example:

```
node example
{
  name: "root"
  children: [
    {name: "child A", value: 1}
    {name: "child B", value: 2}
  ]
}
```

To use this data type, add `dataType: "d3hierarchy"` to the plugin definition.

For example:

```
{
  id: "com.oracle.xdo...",
  component: {
    name: "Field Test"
  }
  dataType: "d3hierarchy",
```

```
fields:
[
  {name: "labelField", caption: "Drop Label Field Here", fieldType:"label",
  dataType:"string"},
  {name: "dataField", caption: "Drop Data Field Here", fieldType:"measure",
  dataType: "number", formula: "summation"}
],
render: function(context, containerElem, node, fields, props) {
  // setup column
  var html = node.value
  ....
}
```

If no data field information is defined in the plug-in definition, null is passed to the node (or rows) argument.

7.6 Testing Your Plug-in

To start the development server, run the following command:

```
$ grunt server
```

This command starts the Web server with port 9000 and opens the plug-in simulator page.

7.7 Preparing Your Plug-in for Upload

To create the archive file appropriate for uploading to the Plugin Gallery, run the following command:

```
$ grunt archive
```

This command packages up the required files and creates a .xmp file that can be uploaded to the Plugin gallery.

7.8 Uploading to the Plug-in Gallery

To make your plug-in available in the BI Mobile App Designer, upload it to the plug-in gallery. You also manage your plug-ins from the Plug-in Gallery. Access the plug-in gallery from the BI Mobile App Designer Administration page. You must have Administrator privileges to access the Administration page.

Alternative Upload Method: If you prefer to upload from your development session, you can run the command

```
$ grunt deploy
```

At the prompts, enter:

- server URL
- administrator username and password
- the destination plugin directory on the server

The deploy command deploys the plugin to the destination folder and creates a .session file in your project. The .session captures the session information you entered enabling you to run the deploy command again without re-entering this information.

7.8.1 Accessing the Plug-in Gallery

To access the BI Mobile App Designer Administration page and the Plug-in Gallery:

1. In your application URL, change the context node to `/mobile`

For example, if you are logged in through Oracle BI EE, your URL will look similar to:

`http://www.hostname:port/analytics.dll`

Replace everything after

`http://www.hostname:port`

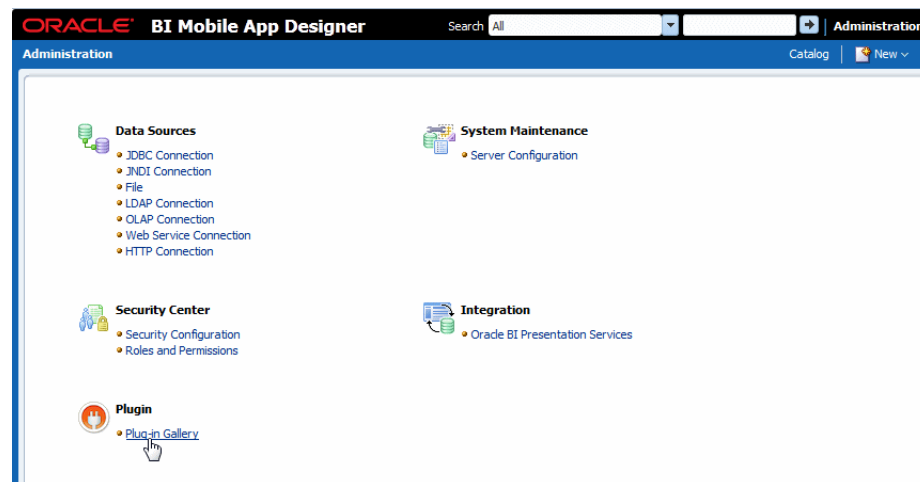
with `/mobile`

that is:

`http://www.hostname:port/mobile`

2. On the Oracle BI Mobile App Designer page, click **Administration** to launch the Administration page.
3. On the Administration page click **Plug-in Gallery** as shown in [Figure 7-1](#).

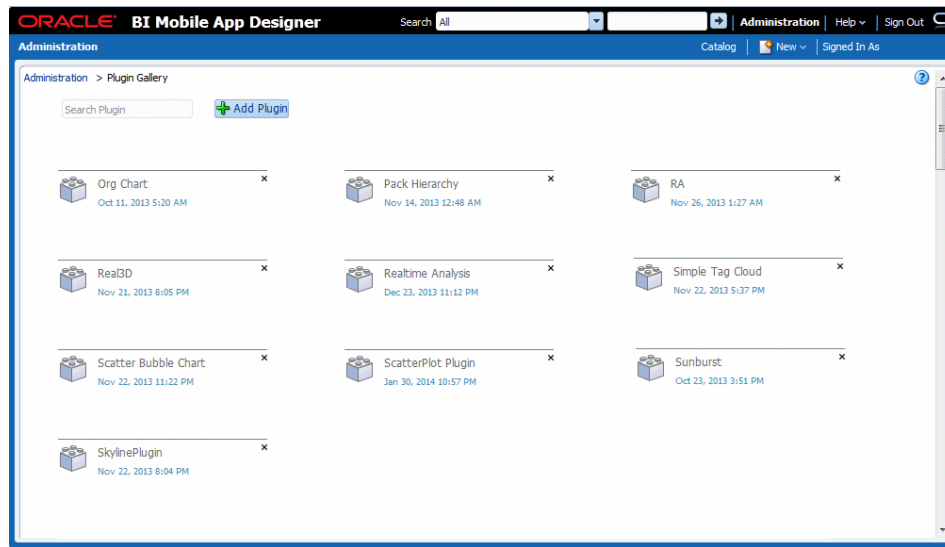
Figure 7-1 Administration Page



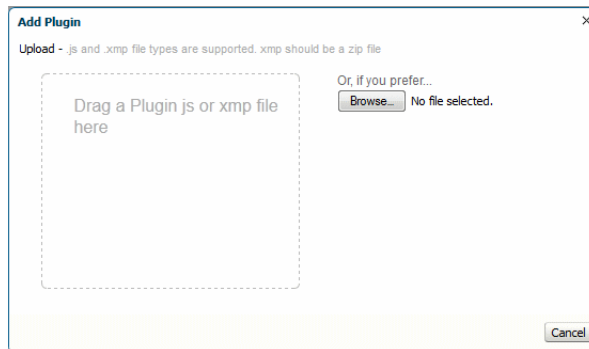
7.8.2 Uploading Plug-ins

To upload a new plug-in to the Plug-in Gallery:

1. On the Plug-in Gallery page, click **Add Plugin**.



2. In the **Add Plugin** dialog drag the plug-in file from your file system to the dialog, or use the Browse button to locate and select it. from your file system.



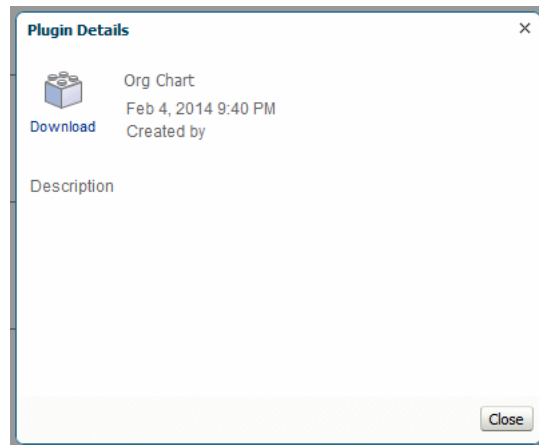
3. Click **Upload**.
4. The plug-in is now visible in the Plug-in Gallery and available in the Mobile App Designer.

7.8.3 Managing Plug-ins

The Plug-in Gallery displays all the plug-ins that are available in your system.

To delete a plug-in from the gallery, click the **Delete** icon next to the plug-in name.

To get more information about a plug-in, click the plug-in name to view the **Plugin Details**.



Reference for Administrators

This appendix contains information for administrators who set up and maintain Oracle BI Mobile App Designer.

It includes the following topics:

- [About the Apps Library](#)
- [Setting Up the Apps Library Locations](#)
- [Sharing Access to the Apps Library with App Consumers](#)
- [Securing Apps in the Apps Library](#)
- [Required Permissions to Use Oracle BI Mobile App Designer and Run Apps](#)
- [Moving Apps Between Environments](#)
- [Adding Custom Maps](#)
- [Activating Metadata Startup Cache to Improve Performance](#)
- [Configuring Single Sign-On](#)
- [Enabling Oracle BI EE 10g-Style Initialization Block Security](#)
- [Protecting Against Out of Memory Errors Using Memory Guard](#)

A.1 About the Apps Library

This section describes the Apps Library. It includes the following sections:

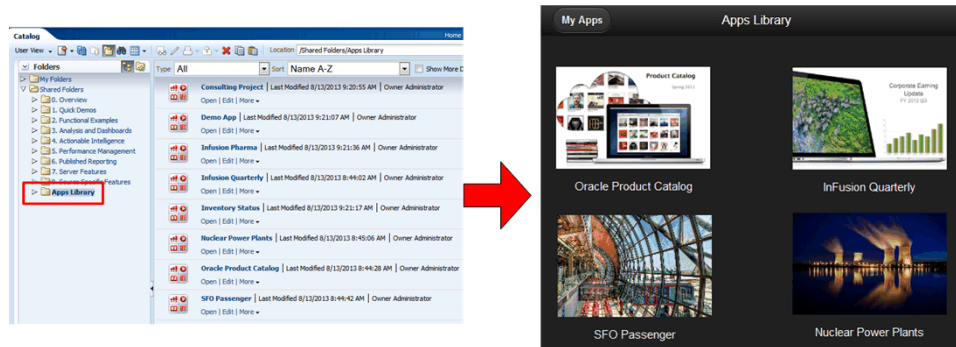
- [What Is the Apps Library?](#)
- [What Happens When a User Subscribes to an App?](#)
- [About Local and Remote Apps Libraries](#)

A.1.1 What Is the Apps Library?

The Apps Library is where users view, manage, and subscribe to published apps from their mobile devices. When users navigate to the Apps Library URL from a browser they can choose to subscribe to the apps they have permission to access.

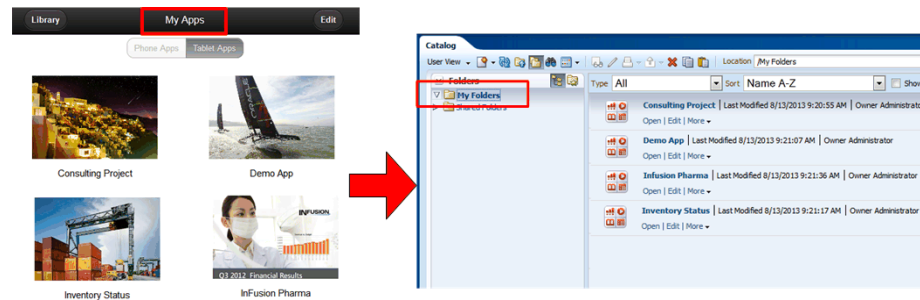
Figure A-1 The Apps Library Viewed on Tablet

From the administrator's perspective, the Apps Library is a folder in the catalog configured to act as the library. When app designers publish an app it is copied to the Apps Library folder. When consumers open the Apps Library URL in their browsers they can interact with the apps that have been published to the folder.

Figure A-2 Apps in the Apps Library Folder Displayed in the Apps Library

A.1.2 What Happens When a User Subscribes to an App?

Subscribing to an app copies the app from the Apps Library to the user's My Folders folder in the catalog. This exposes the app in the user's My Apps library.

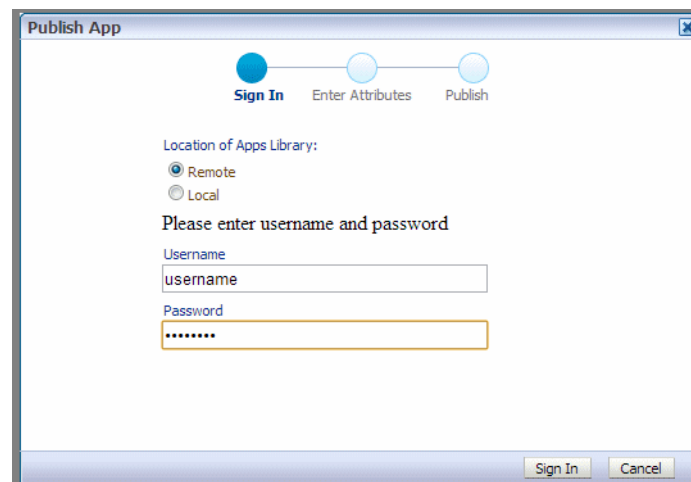
Figure A–3 *Subscribing to an App Copies the App to a User's My Folders Folder*

A.1.3 About Local and Remote Apps Libraries

You can configure a folder on the local instance to be the Apps Library where app designers publish their apps. You can also configure an instance to connect and publish to the Apps Library on another (remote) instance.

For example, suppose you have a development instance and a production instance. Both instances have a local Apps Library defined. The development instance has the production instance configured as a remote library. App designers using the development instance publish their apps to the local (development) instance during design time to test and review. When apps are ready for production you can use the Publish option from the development instance to publish the app to the production instance using the remote Apps Library option.

Figure A–4 shows the Remote and Local options available to select the location of the Apps Library that you publish to.

Figure A–4 *Publish App Dialog Offers Local and Remote Options*

A.2 Setting Up the Apps Library Locations

The following sections describe setting up an Apps Library and enabling connection to publish to a remote server.

- [Setting Up a Local Apps Library](#)
- [Enabling Remote Connection to an Apps Library on Another Instance](#)

A.2.1 Setting Up a Local Apps Library

To set up a local Apps Library to create an Apps Library on the same instance, perform the following:

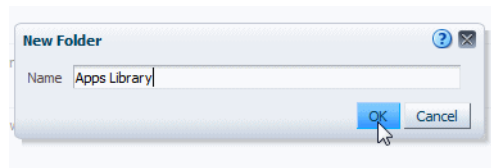
- [Create the Apps Library Folder in the Catalog](#)
- [Set Up the Configuration File](#)

A.2.1.1 Create the Apps Library Folder in the Catalog

To create a folder in the catalog:

1. On the Catalog page, select Shared Folders.
2. In the catalog toolbar, click **New** and select **Folder**.
3. Enter the folder name and click OK, as shown in [Figure A-5](#).

Figure A-5 *Creating the Apps Library Folder in the Catalog*



4. Set Permissions on the Apps Library folder. App designers that will be publishing apps to the folder must have Write permissions. App consumers that will be viewing apps in the library must have Read permissions.

A.2.1.2 Set Up the Configuration File

To register the folder in the catalog designated as the Apps Library you add a property entry to the `xmlp-server-config.xml` configuration file.

To set up the configuration file:

1. Open the `xmlp-server-config.xml` file. It is located under `<DOMAIN_HOME>/config/bipublisher/repository/Admin/Configuration`.
2. Add the following property to `xmlp-server-config.xml`:

Property: `APPS_LIBRARY_FOLDER_LOCAL`

Description: Specifies the folder in the catalog to act as the Apps Library. Enter the path to the folder under Shared Folders that you created in the previous step (do not include "Shared Folders" in the path).

Sample Configuration File Entry:

```
<property name="APPS_LIBRARY_FOLDER_LOCAL" value="/ Apps Library"/>
```

3. Restart the `bimad` (BI Mobile App Designer) application in the WebLogic Server Administration Console.

A.2.2 Enabling Remote Connection to an Apps Library on Another Instance

Set up a remote Apps Library connection when you want to Publish apps that reside on one instance to the Apps Library that resides in the catalog of another instance. To enable connection to a remote Apps Library, register the connection information in the configuration file of the instance from which you want to connect.

To enable connection to a remote Apps Library:

1. Open the `xm1p-server-config.xml` file. It is located under `<DOMAIN_HOME>/config/bipublisher/repository/Admin/Configuration`.
2. Add the following properties to `xm1p-server-config.xml`:

Property	Description	Sample Configuration File Entry
<code>APPS_LIBRARY_FOLDER_REMOTE</code>	Specifies the Apps Library folder in the catalog on the remote server.	<code><property name="APPS_LIBRARY_FOLDER_REMOTE" value="/Apps Library"/></code>
<code>MOBILE_APP_REMOTE_SERVER</code>	Specifies the remote server where the <code>APPS_LIBRARY_FOLDER_REMOTE</code> resides. Setting this property enables connection to Publish apps to the remote Apps Library.	<code><property name="MOBILE_APP_REMOTE_SERVER" value="http://example.com:7001/mobile/" /></code>

The following sample shows example entries when both a local and remote Apps Library are defined:

```
<property name="APPS_LIBRARY_FOLDER_REMOTE" value="/Apps Library"/>
<property name="MOBILE_APP_REMOTE_SERVER"
value="http://example.com:7001/mobile/" />
<property name="APPS_LIBRARY_FOLDER_LOCAL" value="/Apps Library Test"/>
```

3. Restart the `bimad` (BI Mobile App Designer) application from the WebLogic Server Administration Console.

A.3 Sharing Access to the Apps Library with App Consumers

The URL for the Apps Library is of the form:

```
http://<hostname>:<port>/mobile/appstore/
```

After you configure the Apps Library, you can provide users with this URL to open in the device browser and bookmark for later use.

Users can also access the Apps Library URL from the Oracle Business Intelligence home page under the **Browse/Manage** region.

Figure A–6 Apps Library Link on Oracle BI Home Page

A.4 Securing Apps in the Apps Library

Typically you configure the Apps Library to be accessible to all users who will be running apps. To ensure that users can only run apps appropriate for their roles, apply permissions specifically to each app. When using the Publish feature to promote apps to the Apps Library, Oracle recommends applying the permissions to the app before Publish. When the app is promoted to the Apps Library it maintains the original permission settings.

For information on setting permissions in the catalog, see "Managing Objects in the Oracle BI Presentation Catalog" in the *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

A.5 Required Permissions to Use Oracle BI Mobile App Designer and Run Apps

The default permissions assigned to the BI Author role enable the use of the Mobile App Designer. To save apps to a specific folder the user's role must also have write permissions on the target folder.

The default permissions assigned to the BI Consumer role enable viewing apps.

If you create custom roles, the following are required:

Table A–1 Required Permissions for Mobile App Designer

Function	Required Permission	Included In
Create apps	oracle.bi.publisher.developReport	BI Author
Create apps that use BI Subject Area or Excel spreadsheet as the data source	oracle.bi.publisher.developLightDataModel	BI Author
View apps	oracle.bi.publisher.runReportOnline	BI Consumer

App consumers must also have appropriate permissions on the app data sources; for example, if your data source is a BI Publisher data model, the role must be granted access to that data source to view apps that run against the data source.

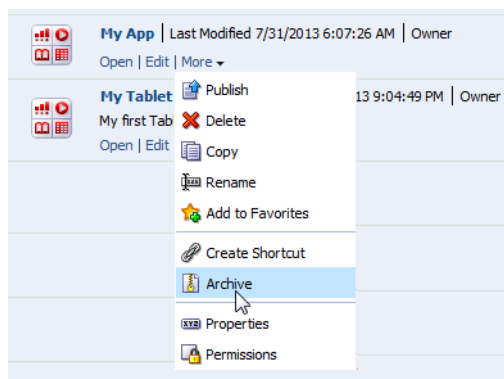
A.6 Moving Apps Between Environments

To move apps between development, test, and production environments, use the archive and unarchive feature available from the BI Presentation catalog. For more information about this feature, see the *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Enterprise Edition*.

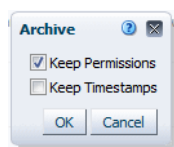
Archiving enables you to bundle the entire catalog, specific folders, or multi-component objects as a `.catalog` file and upload the `.catalog` file to unarchive the data to another location in the catalog. This process enables you to transfer specific data across environments.

To create an archive file:

1. In the Catalog navigate to the folder or app to archive.
2. Select **More**, then **Archive**.



3. In the Archive dialog, specify to maintain or omit the permissions and timestamps for the folder or object.



Keep Permissions: Use this option to maintain the object or folder's existing permissions. If you do not select this option, then the archiving process does not include any permissions. Upon unarchiving, the system assigns the parent folder's permissions to all of the objects and folders.

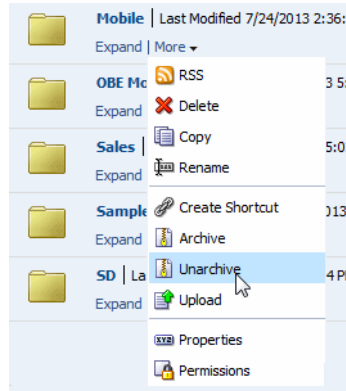
Keep Timestamps: Use this option to maintain the CreationTime, LastModified, and LastAccessed timestamps assigned to the object or folder. Upon unarchiving, the LastModified timestamp is updated to indicate the time at which the object or folder is unarchived. If you select this option, the **Old** option in the Paste Overview area of the Preferences dialog is available when unarchiving. You use the **Old** option to overwrite existing catalog items that are older than the catalog items in the archive.

If you do not select this option, then the archiving process does not include timestamp information and the **Old** option in the Paste Overview area of the Preferences dialog is not available.

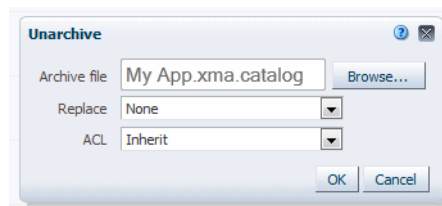
4. Click **OK** to save the archive file.

To unarchive an archive file:

1. In the catalog, select the folder location where you want to upload the archive file.
2. Select **More**, then **Unarchive**.



3. In the Unarchive dialog, browse for and select the archive file.



Replace: Use to specify if and how to replace an existing folder or object with the same name. Note the following options:

- **All** — Select this option to replace any existing folders or objects with the same names as folders or objects included in the archive file that you are uploading.
- **Old** — Select this option to replace folders or objects except those folders or objects that exist, unless they are older than the source.
- **None** — Select this option to add any new folders or objects, but preserve any existing folders or objects.
- **Force** — Select this option to add and replace all folders or objects.

ACL: Use to specify how the folders or objects are assigned permissions using Access Control Lists (ACLs) when unarchived. Note the following options:

- **Inherit** — Inherits the folder or object's permissions (ACL) from its new parent folder.
- **Preserve** — Preserves the folder or object's permissions (ACL) as it was in the original, mapping accounts as necessary.
- **Create** — Preserves the folder or object's permissions (ACL) as it was in the original, creating and mapping accounts as necessary.

4. Click **OK**.

A.7 Adding Custom Maps

The map component supports custom region definitions. Provide the custom region definitions through a custom geoJSON format file; then update the configuration file to make it available to the Mobile App Designer map component.

1. Define your region definitions in a geoJSON format file. The files must be saved with the .json extension. (See <http://geojson.org/>)
2. Copy the geoJSON file to the Map directory in the BI Mobile App Designer repository:

<DOMAIN_HOME>/config/bipublisher/repository/Admin/Map.

3. Update the map-config.xml file to recognize your geoJSON file.

The map-config.xml file is also located in:

<DOMAIN_HOME>/config/bipublisher/repository/Admin/Map.

Sample entries in the map-config file are:

```
<?xml version="1.0"?>
<map-config>
  <geojson-list>
    <geojson file="world_countries.json" display-name="World Countries"
label-column="NAME">
      <matcher type="property" name="NAME" case-sensitive="false"/>
      <matcher type="property" name="ISO2" case-sensitive="true"/>
      <matcher type="property" name="ISO3" case-sensitive="true"/>
    </geojson>
    <geojson file="usa_states.json" display-name="U.S. States"
label-column="STATE_ABRV" projection="AlbersUSA">
      <matcher type="id" case-sensitive="false"/>
      <matcher type="property" name="STATE_ABRV" case-sensitive="false"/>
    </geojson>
    <geojson file="usa_counties.json" display-name="U.S. Counties"
label-column="DISPLAY_NAME" projection="AlbersUSA" outline-file="usa_states_
line.json">
      <matcher type="id" case-sensitive="false"/>
    </geojson>
  </geojson-list>
</map-config>
```

Add another <geojson> element to the map-config file to define the attributes for your custom geoJSON file. Each <geojson> element in the configuration file must have at least one <matcher> child element. The <matcher> element specifies how data is matched from your data source to the location definition in the geoJSON file.

The <geojson> element takes the following attributes:

Table A-2 Attributes of the <geojson> Element

Attribute	Description
file	Required. The custom geojson file name residing in the Admin/Map folder. Example: "world_regions.json"
display-name	Required. The name for the map that displays in the selection list in the app designer. Example: "World Regions"

Table A–2 (Cont.) Attributes of the <geojson> Element

Attribute	Description
label-column	Optional. Specifies the attribute from the geoJSON file to use as the display label for locations on the map. The display label is shown when the Display Labels property is enabled for the map. Example: "STATE_ABRV" might be entered here to display the state abbreviation for each state on the map.
projection	Optional. Applies to maps of the USA only. Valid values are "Mercator" and "AlbersUSA". If not specified, maps of the USA use the "Mercator" projection.
outline-file	Optional. Applies to maps of the USA only. The outline-file provides enhanced border outlines of the US states.

Following are the attributes for the child <matcher> element. You can have multiple <matcher> elements to provide multiple ways to map data fields to locations defined in the geoJSON file.

Table A–3 Attributes of the Child <matcher> Element

Attribute	Description
type	Required. Valid values are "id" and "property". Specifies how to match a data field from the data source to a location defined in the geojson file. When set to "id" the commonly used identifier defined for the geoJSON object is used to match a data field to the location definition. When set to "property" you also must specify "name" attribute to specify the property in the geoJSON file to match to.
name	Required when type is "property". Not required when type is "id". The name of the property to match the data field value to in the geoJSON file.
case-sensitive	Values are: "true" or "false". Specifies whether the matcher field is case sensitive.

For example:

In the World Countries geoJSON file, the commonly used identifier (_id) for each country is a three-letter country code (e.g. USA, JPN, or FRA). Because the map configuration file includes a matcher element of type "id", as long as your data also includes the same three-letter country codes to identify countries, your data will be mapped properly.

If you look more closely at the geoJSON file, you will notice that it includes more options for associating data values with the geographical polygons, specifically: "NAME", "ISO2", "ISO3", "CONTINENT", "ALT_REGION".

The map config file also includes three "property" matcher type entries for the World Countries map. The corresponding "name" attributes are "ISO2", "ISO3" and "NAME". These provide additional valid values for mapping your data to the World Countries map. So if your data includes the matching ISO2 code, ISO3 code, or Name to identify each country, your data will still be mapped properly (for example, JP, JPN, or JAPAN).

A.8 Activating Metadata Startup Cache to Improve Performance

BI Subject Area metadata such as dimension and measure names are cached in the server to improve the performance when opening the Designer or running apps. By default this caching occurs for a Subject Area the first time it is used by Mobile App Designer either for designing or running an app. For large subject areas the initial loading time can seem unacceptable to users.

You can avoid this initial load-time experience by activating metadata start-up cache. With the metadata start-up cache activated, all Subject Area metadata (that you specify) are loaded into cache when the Mobile App Designer server initially starts up. Because the cache is loaded before users begin running or designing apps, users do not experience the first-time load slowdown.

- [Configuring Metadata Cache on Startup](#)
- [Clearing the Subject Area Metadata Cache](#)

A.8.1 Configuring Metadata Cache on Startup

To activate metadata cache on startup, manually update the configuration file, `xmlpserver-config.xml`.

1. Navigate to the `xmlpserver-config.xml` file located in:
`<DOMAIN_HOME>/config/bipublisher/repository/Admin/Configuration/`
2. Locate the following three properties and update them as described in [Table A-4](#):

Note: If you upgraded from the 11.1.1.7 release of Mobile App Designer, you must add these properties to `xmlpserver-config.xml` file.

Table A-4 Business View Cache Properties

Property	Description
BVCACHE_ALLOW_LOADING_AT_STARTUP	Set value="true" to turn on subject area metadata caching. The default is "false".
BVCACHE_LIST_OF_SUBJECT_AREAS	<p>List the Subject Areas to cache at startup. Separate Subject Areas with a comma. For example:</p> <pre><property name="BVCACHE_LIST_OF_SUBJECT_AREAS" value="Sales,Target Sales,Offices" /></pre> <p>To cache all Subject Areas, use "*", for example:</p> <pre><property name="BVCACHE_LIST_OF_SUBJECT_AREAS" value="*" /></pre> <p>This property is active only when BVCACHE_ALLOW_LOADING_AT_STARTUP is set to true.</p>
BVCACHE_DEFAULT_BIEE_USERNAME	<p>Enter a username to connect to the Oracle BIEE system. This user does not need Administration privileges, but must have read access to the Subject Areas.</p> <p>This property is active only when BVCACHE_ALLOW_LOADING_AT_STARTUP is set to true.</p>

The following sample shows how the properties display in the `xmlpserver-config.xml` file:

```
<!-- Turn on/off the automatic subject area metadata cache loading
```

```

at the system startup time. Value can be true or false. The default is false.
->
<property name="BVCACHE_ALLOW_LOADING_AT_STARTUP" value="true"/>

<!-- List of subject areas to load automatically at the startup time.
Separated by comma, or simply '*' to load all subject areas. ->
<property name="BVCACHE_LIST_OF_SUBJECT_AREAS" value="*" />

<!-- The default username to connect to BIEE system to load subject areas.
This is mandatory for the automatic subject area business view loading. ->
<property name="BVCACHE_DEFAULT_BIEE_USERNAME" value="username" />

```

Usage Tips

Consider carefully when deciding the trade-off in slower initial start-up time versus better subsequent performance. If your system has hundreds of Subject Areas, consider defining a list of Subject Areas rather than using '*', otherwise the startup time of the Mobile App Designer application may be unacceptable. Focus on listing those Subject Areas with many tables and measures that cause the biggest hit to performance for end users.

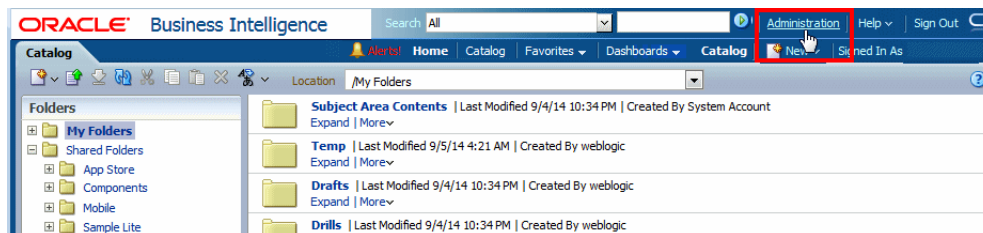
A.8.2 Clearing the Subject Area Metadata Cache

The cached Subject Area metadata remain in cache until you clear it. If you make any changes to your subject areas in the RPD, such as adding or deleting columns, you must follow this procedure for your changes to show up in Mobile App Designer.

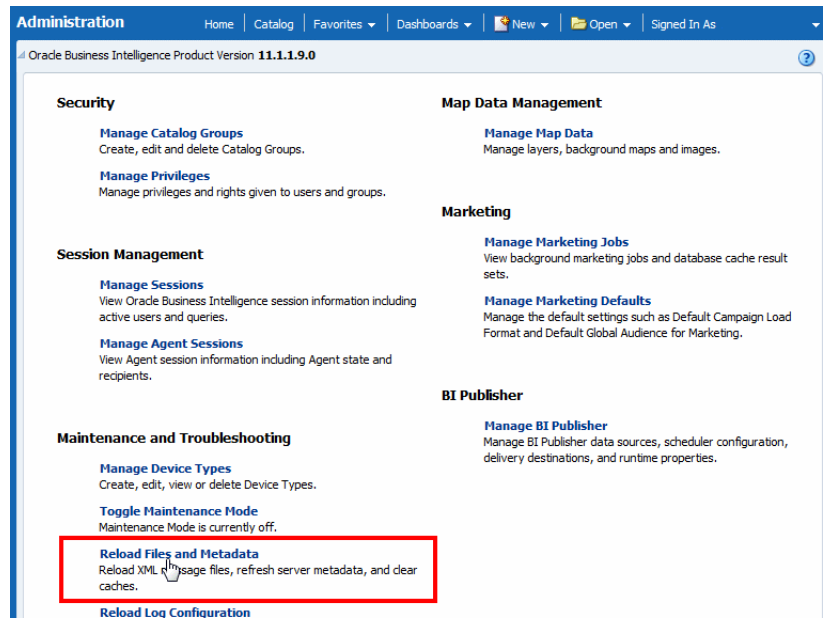
Clearing the cache is a two-step process: You must refresh the cache from the Oracle BI Administration page and then also clear the cache from the BI Mobile App Designer Administration page.

To clear the Subject Area metadata cache:

1. On the Oracle Business Intelligence header, click **Administration**.



2. On the **Administration** page, under **Maintenance and Troubleshooting**, click **Reload Files and Metadata**.

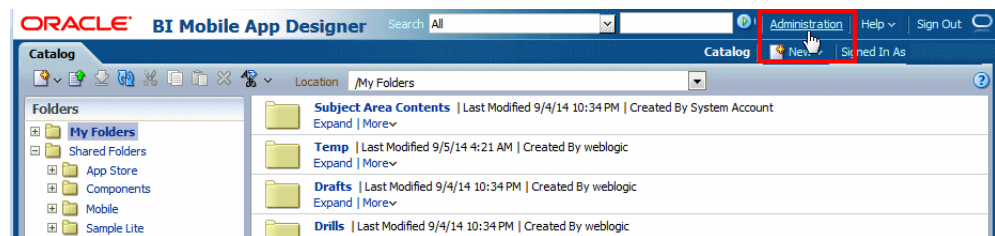


- Next, launch the BI Mobile App Designer application directly from its URL:

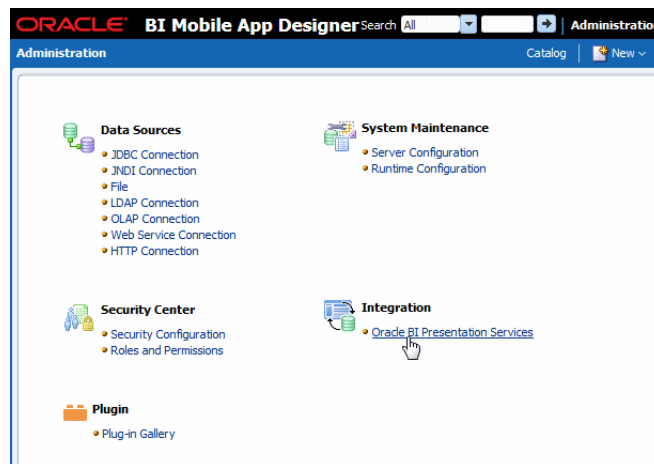
`http://<host>:<port>/mobile/`

and log in with Administrator credentials.

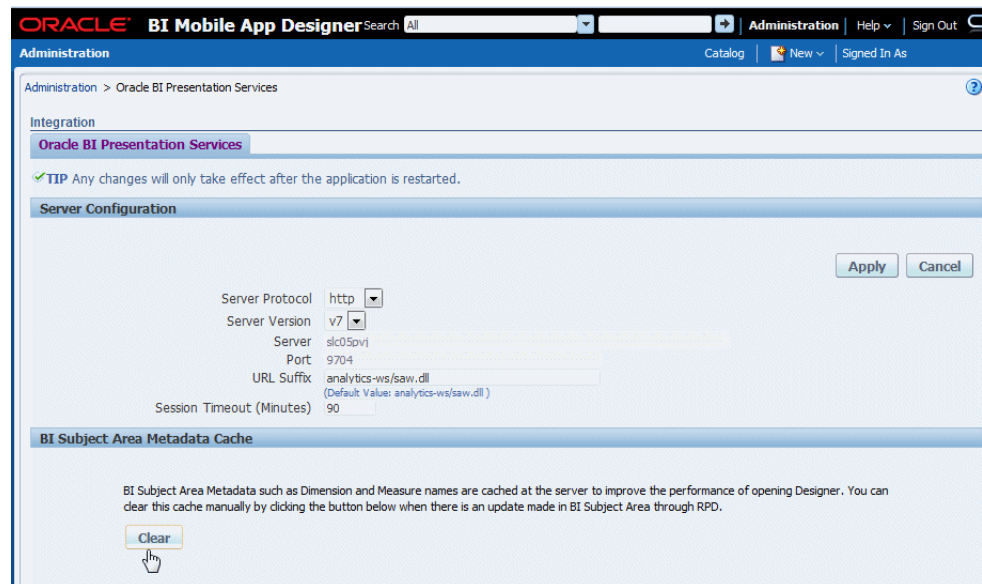
- Click **Administration**.



- On the **Administration** page, click **Oracle BI Presentation Services**.



- Under **BI Subject Area Metadata Cache**, click **Clear**.



A.9 Configuring Single Sign-On

If your Oracle BI Enterprise Edition is configured for single sign-on, you must add BI Mobile App Designer as a protected resource and also unprotect services to allow communication with other BI resources.

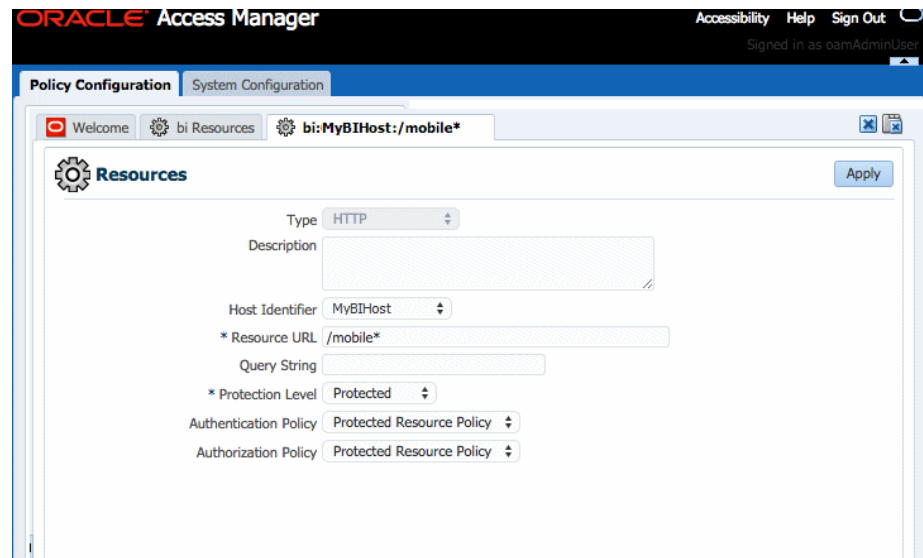
See your single sign-on provider documentation for specific requirements for your single sign-on provider. See also the *Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition*.

A.9.1 Setup Procedure for Oracle Access Manager

To set up Single Sign-On in Oracle Access Manager:

1. Navigate to the Oracle Access Management Console.
2. Under the bi domain, add these BI Mobile App Designer URLs and set the Protection Level to "Protected":

```
/mobile*
/mobile/.../*
```

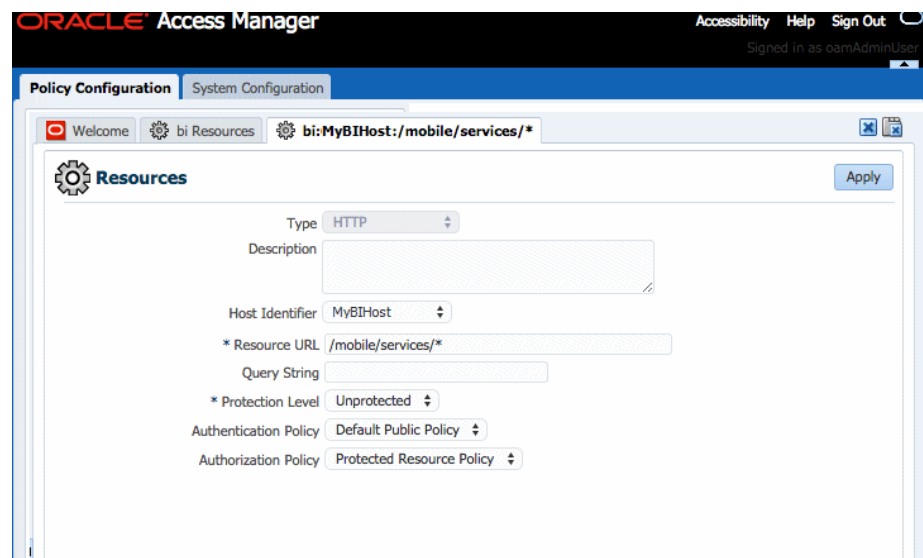
Figure A-7 Protecting the /mobile* Resource URL in Oracle Access Manager

3. To allow internal communication between Mobile App Designer and other components of the BI suite, add these URLs and set the Protection Level to "Unprotected":

```

/mobile/services/*
/mobile/services/.../*
/mobile/report_service/*
/mobile/report_service/.../*

```

Figure A-8 Unprotecting the /mobile/services/* URL in Oracle Access Manager

4. Set up the Single Sign-Off URL on the BI Mobile App Designer Administration Security Configuration page.

Important: BI Mobile App Designer shares this setting with Oracle BI Publisher. If you have already set up the Single-Sign-Off URL in BI Publisher, you can skip this step.

On the **Administration** page, click **Security Configuration**. In the **Authentication** region:

- Select **Use Single Sign-On**
- From the **Single Sign-On Type** list, select **Oracle Access Manager**.
- Enter the **Single Sign-Off URL**.

A sample Security Configuration page is shown in [Figure A-9](#).

Figure A-9 Sample Security Configuration Page

The screenshot shows the 'Administration' page of Oracle BI Mobile App Designer. The 'Authentication' section is active. It includes a note about selecting Single Sign-On or LDAP. The 'Use Single Sign-On' checkbox is checked. Below it, the 'Single Sign-On Type' is set to 'Oracle Access Manager'. The 'Single Sign-Off URL' is 'http://(host):(port)/'. The 'How to get username' is 'HTTP Header' and the 'User Name Parameter' is 'OAM_REMOTE_USER'. The 'How to get user locale' is 'HTTP Parameter' and the 'User Locale Parameter' is 'LOCALE_LANGUAGE'. There is a section for LDAP configuration with the 'Use LDAP' checkbox unchecked.

5. Click **Apply**. Restart the application.

A.10 Enabling Oracle BI EE 10g-Style Initialization Block Security

If your authentication and authorization are configured with the Oracle BI EE 10g style Initialization Blocks (Init Blocks) against external database tables, you must manually update the web.xml file inside the bimad.ear as described here.

To update the web.xml file in the bimad.ear:

1. Back up the existing bimad.ear file in your staging directory:
2. Extract bimad.war from bimad.ear.
3. Update WEB-INF/web.xml in the bimad.war file as follows:

Find SAW_AUTH_INIT_BLOCK_ONLY and change the value from "true" to "false"

```
<servlet>
<servlet-name>xdo</servlet-name>
```

```

<servlet-class>oracle.xdo.servlet.MobileXDOServlet</servlet-class>
<init-param>
<param-name>SAW_AUTH_INIT_BLOCK_ONLY</param-name>
<param-value>false</param-value>

<!-- <description>True to tell BIEE executing auth init block only. False
otherwise </description> -->
</init-param> <load-on-startup>2</load-on-startup>
</servlet>

```

4. Re-pack bimad.ear with modified bimad.war.
5. From the Oracle WebLogic Server Administration Console, update the bimad.ear deployment with the modified bimad.ear file as follows:
 - a. Open your Oracle WebLogic Server Administration Console.
 - b. In the **Change Center** of the Administration Console, click **Lock & Edit**.
 - c. In the left pane of the Console, select **Deployments**. A table in the right pane displays all deployed Enterprise Applications and Application Modules.
 - d. In the table, select bimad.
 - e. Click **Update**.
 - f. Click **Finish** (do not change the source path).
 - g. In the **Change Center** of the Administration Console, click **Activate Changes**.

A.11 Protecting Against Out of Memory Errors Using Memory Guard

BI Mobile App Designer provides a set of properties to protect against out-of-memory errors by blocking app requests that generate excessive amounts of data. The properties set limits on data size and free memory availability. When an app exceeds the set limits, processing of the app terminates and the following error message is returned to the user:

The report you are trying to run exceeds the data limit set for this server.

Memory Guard includes default settings that may not be appropriate for your system. It may take trial and error before you find the appropriate setting for your system. When editing the Memory Guard limits keep in mind that setting them too high may impact the overall health of your system.

- [What are the Memory Guard Settings?](#)
- [Setting Memory Guard Properties](#)

A.11.1 What are the Memory Guard Settings?

Memory Guard protects your system by setting maximum limits on data set size and by setting minimum limits on available memory:

- [Restricting Maximum Data Sizes for App Processing](#)
- [Configuring Free Memory Threshold](#)

A.11.1.1 Restricting Maximum Data Sizes for App Processing

By restricting the data size allowed for app processing you can prevent out of memory errors when a query returns more data than the system can handle.

Specify a Maximum Data Size Allowed for Online Processing

Property: **Maximum report data size for online reports**

Default value: 300 MB

This property limits the data size allowed for online report (or app) viewing. This limit is applied as follows:

1. A user taps (or clicks) an app.
2. The data engine retrieves the data for the app.
3. The size of the data is inspected.
4. If the data generated is larger than the maximum setting, the app processing is ended. The user gets the following message:

The report you are trying to run exceeds the data limit set for this server.

To enable this app to run, you must The user can then either set parameters (if available for the report) to limit the data and resubmit online; or use the BI Publisher scheduler to submit the report.

A.11.1.2 Configuring Free Memory Threshold

This set of properties helps you to protect against out of memory conditions by establishing a minimum available free memory space. This set of properties enables your system to automatically protect free memory availability and intelligently process reports with large data sets based on this availability.

- [Specify A Minimum Free Memory Threshold for App Processing](#)
- [Specify Maximum App Data Size Under the Free Memory Threshold](#)
- [Set Minimum Time Span Between Garbage Collection Runs](#)
- [Set Maximum Wait Time for Free Memory to Come Back Above the Threshold](#)

Specify A Minimum Free Memory Threshold for App Processing

Property: **Free memory threshold**

Default value: 500 MB

This property protects a minimum amount of JVM space while evaluating the ability of the to system to run an app based on two factors: current usage and the size of the report data. This feature requires the setting of several properties that work together. You specify the threshold JVM space, the maximum app size that will be allowed when the JVM falls below the threshold, and the maximum wait time to pause the app to wait for more JVM free space to become available.

These limits are applied as follows:

1. A user taps (or clicks) an app.
2. The data engine generates the data for the app.
3. JVM memory is inspected. If the available JVM memory is above the **Free memory threshold** property value, the app processes as usual and there is no system intervention.

If the available JVM memory is below the threshold value, the size of the app data is inspected and compared to the property setting for **Maximum report data size under the free memory threshold**. If the app data is below this threshold, then the app continues processing.

If the app data size exceeds the threshold, then the app is paused to wait for free memory to become available. The app will wait for the time specified in the property **Maximum Wait Time for Free Memory to Come Back Above Threshold Value**. If the free memory does not rise back above the minimum in the wait period specified, the app request is rejected.

Specify Maximum App Data Size Under the Free Memory Threshold

Property: **Maximum report data size under the free memory threshold**

Default value: (value of Free Memory Threshold)/10

Maximum single app data size allowed when free JVM memory is under the specified threshold value set in **Free memory threshold**. For example (assuming the default setting), if the data generated for an app exceeds one-tenth of the value set for **Free memory threshold**, then processing is terminated. Therefore if the Free memory threshold is set to 100 MB and an app data extract exceeds 10 MB, then the report processing is terminated.

This property takes effect only when **Free memory threshold** is set to be a positive value.

Set Minimum Time Span Between Garbage Collection Runs

Minimum time span in seconds between any two subsequent garbage collection runs. Set this value to avoid overrunning JVM garbage collection. The server enforces the minimum of 120 seconds, which means the value will be reset to 120 seconds if it falls below the minimum.

The default is 300 seconds.

Set Maximum Wait Time for Free Memory to Come Back Above the Threshold

The maximum time in seconds that a run-app request will wait for free JVM memory to come back above the threshold value. This property value takes effect only when a positive value for Free memory threshold is specified.

If the free memory becomes available within the time specified, the request will proceed immediately to generate the app. If free memory is still below the threshold value after the time specified, the request is rejected.

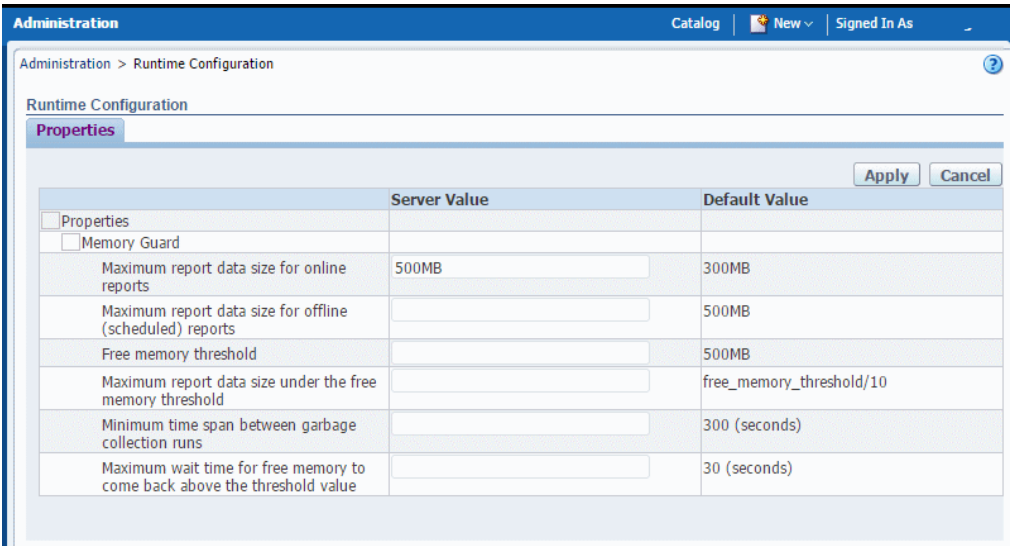
The default for this property is 30 seconds.

A.11.2 Setting Memory Guard Properties

To navigate to the Memory Guard properties:

1. Launch the BI Mobile App Designer application directly from its URL:
`http://<host>:<port>/mobile/`
2. Click **Administration**.
3. From the Administration page, under **System Maintenance**, click **Runtime Configuration**.
4. The default values are shown. To change the value for your server, enter the new value in the Server Value column. See [Table A-5](#) for property details.

Figure A–10 Memory Guard Property Settings



IMPORTANT: These settings are shared with Oracle BI Publisher. Updating the setting here also updates it for BI Publisher.

Table A–5 Memory Guard Properties and Descriptions

Property	Description
Maximum report data size for online reports	<p>Default value: 300MB</p> <p>Sets the maximum allowed app data size. You can set the value in GB, MB, or KB. For example: 1GB, 200MB, or 1500KB.</p> <p>An app request is rejected immediately when the data size returned from the data model exceeds the value of this property.</p> <p>To turn off this property, enter 0 or a negative number.</p>
Maximum report data size for offline (scheduled) reports	<p>Not used by BI Mobile App Designer.</p>
Free memory threshold	<p>Default value: 500MB</p> <p>Threshold value of free JVM memory to trigger possible rejection of app requests. You can set the value in GB, MB, or KB. For example: 1GB, 200MB, or 1500KB.</p> <p>When JVM free memory returned from run time is below the value of this property, the server will check the report data size to decide if a request should be accepted or rejected. This property works together with the three properties:</p> <ul style="list-style-type: none">Maximum report data size under the free memory thresholdMinimum Time Span Between Garbage Collection RunsMaximum Wait Time for Free Memory to Come Back Above Threshold Value <p>If this property value is 0 or a negative number, this condition is ignored.</p>

Table A–5 (Cont.) Memory Guard Properties and Descriptions

Property	Description
Maximum report data size under the free memory threshold	<p>Default value: (value of Free memory threshold / 10)</p> <p>Maximum app data size allowed when free JVM memory is under the specified threshold value set in Free memory threshold. A request will be rejected when its data size exceeds the value of this property.</p> <p>This property takes effect only if Free memory threshold is set to be a positive value.</p> <p>You can set the value in GB, MB, or KB. For example: 1GB, 10MB, or 1500KB. If you do not explicitly set the value, the default value is calculated by dividing the value you set for Free memory threshold by 10. So if you set Free memory threshold to 100MB, the default value for this property is 10MB.</p>
Minimum time span between garbage collection runs	<p>Default value: 300 (seconds)</p> <p>Minimum time span in seconds between any two subsequent garbage collection runs. Set this value to avoid overrunning JVM garbage collection. Note that the server automatically enforces a minimum value of 120 seconds, so if you enter a value less than 120 seconds, the server overrides it.</p>
Maximum wait time for free memory to come back above the threshold value	<p>Default value: 30 (seconds)</p> <p>The maximum time in seconds that an app request will wait for free JVM memory to come back above the threshold value. This property value takes effect only when a positive value of Free memory threshold is specified. If the free memory comes back in the time less than the value of Maximum wait time for free memory to come back above the threshold value, the request will proceed immediately to generate the app. If free memory is still below the threshold value after the time set for this property, the request is rejected.</p>

