

iDashboards Builder Manual

Version 11.3

iDashboards Builder Manual

Version 11.3

No part of the computer software or this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without permission in writing from iDashboards. The information in this document is subject to change without notice. If you find any problems with this documentation, please report them in writing to support@iDashboards.com. iDashboards does not warrant that this document is error free.

Copyright © 2004 - 2022 iDashboards. All rights reserved.

Trademarks:

The iDashboards logo and tagline are trademarks of iDashboards.

All other products and company names referenced herein are the trademarks of their respective owners.

Support information:

iDashboards
900 Tower Drive, 4th Floor
Troy, MI 48098

Phone: (248) 528-7160
Fax: (248) 828-2770

Email: support@iDashboards.com
Web site: <http://www.iDashboards.com>



1. Table of Contents

Contents

1. Table of Contents.....	3
2. OSKAR.....	13
3. Introduction.....	14
3.1 iDashboards URL.....	14
3.2 iDashboards URL Details.....	14
3.3 System requirements for accessing iDashboards.....	15
3.4 Login Requirements and User Roles.....	15
3.5 Login Experience.....	15
3.5.1 Mobile.....	16
3.5.2 Lobby.....	16
3.5.3 View.....	16
3.5.4 Build (this manual).....	17
3.5.5 Admin.....	17
3.5.6 Log out.....	17
4. Build Interface.....	18
4.1 Requirements.....	19
4.2 Menu – Basic.....	19
4.2.1 Go to Lobby.....	19
4.2.2 User Settings.....	19
4.2.3 About.....	22
4.2.4 Help.....	22
4.2.5 Log Out.....	22
4.3 Colors.....	23
4.3.1 Color picker.....	23
4.3.2 Palettes.....	24
4.4 Menu – Dashboard.....	26
4.4.1 Refresh Dashboard.....	26
4.4.2 Identify Favorite Dashboards.....	26
4.4.3 Set as Startup Dashboard.....	26
4.4.4 Enter Dashboard Parameters.....	26

4.5	Opening	26
4.5.1	Sorting	27
4.5.2	By Category	27
4.5.3	By Searching	27
5.	Categories	28
5.1	Personal Category	28
5.2	Creating Categories	28
5.3	Category Sort Order	28
5.4	View-Only Categories	28
5.5	Protected Categories	28
6.	Designing Dashboards	29
6.1	Design Permissions	29
6.2	Create or Edit a Dashboard	29
6.3	The Dashboard Designer	30
6.3.1	Top-Level Properties	30
6.3.2	Page Manager	31
6.3.3	Designer Controls on the Title Bar	35
6.3.4	Designer Dashboard Layouts	37
6.3.5	Designer Toolbar	39
6.3.6	Icon Overview	39
6.3.7	Dashboard Scaling	42
6.4	Frames and Frame Borders	44
6.4.1	Frames	44
6.4.2	Frame Borders	45
6.4.3	Frame Border: Positioning	45
6.4.4	Frame Borders: Show or Hide	46
6.4.5	Frame Tile Effects	47
6.5	Dashboard Backgrounds	47
6.5.1	Dashboard Background Image	47
6.5.2	Dashboard Background Color	48
6.6	Show Chart-Menu Buttons	48
6.7	Synchronized Highlighting	49
6.8	Synchronize Pivots	49
6.9	Mobile Interface	50
6.9.1	Design Layout	50



6.9.2	Frame Ordering (Mobile Device > Display Mode).....	50
6.10	Move Dashboard (via Changing the Dashboard Category)	52
6.11	Rename a Dashboard.....	52
6.12	Hide the Category Name in the Dashboard Title	52
6.13	Dashboard Thumbnails.....	53
6.13.1	Thumbnail Policy	54
6.14	Hide Dashboard.....	55
6.15	Refresh an Open Dashboard	56
6.15.1	Auto-Refresh Dashboard	56
6.16	Saving the Dashboard (or Save-As).....	57
6.16.1	Dashboard Title	57
6.16.2	Dashboard Description	58
6.17	Dashboard Panels	59
6.17.1	Image Panel	59
6.17.2	Text Panel	61
6.17.3	HTML Panel.....	62
6.17.4	IFRAME Panel.....	64
6.17.5	Parameter Panel.....	64
6.17.6	Comments Panel	65
6.17.7	Dashboard Launcher	66
6.17.8	Form Panel	67
6.17.9	Article Panel	68
7.	Chart Overview	69
7.1	Design Permissions	69
7.2	Create or Edit a Chart.....	69
7.3	Chart Designer.....	70
7.3.1	Navigating the Chart Designer Stages	70
7.3.2	Chart Designer Help	71
7.4	Saving the Chart (or Save As)	71
7.4.1	Save as Snapshot.....	72
7.4.2	Chart Title	72
7.4.3	Chart Description	73
8.	Designing Charts	74
8.1	Dynamic Data	75
8.1.1	Table/View.....	75



8.1.2	Custom Query.....	76
8.1.3	Stored Procedure.....	77
8.1.4	Workbook Range	78
8.1.5	Data Feed.....	79
8.2	Static Data	80
8.2.1	Import Data.....	80
8.2.2	Create Data	81
8.2.3	Editing Data	83
8.3	Define Axis List.....	83
8.4	Define Chart Type.....	85
8.5	Chart Properties.....	86
8.5.1	Live Chart Preview.....	86
8.5.2	Basic Settings.....	87
8.5.3	Conditional Basic Settings	97
8.5.4	Formats and Labels	99
8.5.5	Unit Labels.....	101
8.5.6	Colors	102
8.5.7	Drilldown.....	105
9.	Chart Types	116
9.1	Bar Charts	117
9.1.1	Gantt Chart	119
9.2	Column Charts.....	123
9.3	Bubble Charts.....	129
9.4	Speedometer Charts.....	130
9.4.1	Speedometer Themes	139
9.5	Bullet Charts	140
9.5.1	Bullet Themes.....	143
9.6	Thermometer Charts.....	144
9.6.1	Thermometer Themes	146
9.7	Metrics Charts.....	148
9.7.1	Stoplight Themes.....	149
9.7.2	Treemap Charts.....	149
9.8	Line Charts	151
9.9	Pie Charts.....	152
9.10	Pyramid and Funnel Charts	154



9.11	Tabular Charts	155
9.12	Calendar Charts.....	156
9.12.1	Data Layout	156
9.12.2	Calendar Types	156
9.13	Presentation Charts	159
9.13.1	Content Folder	159
9.13.2	Content Macro (non-Detail chart).....	160
9.13.3	Missing Images.....	160
9.13.4	Gallery	161
9.13.5	Slideshow	161
9.13.6	Details.....	162
9.13.7	Infographic.....	166
9.14	Map Charts	168
9.14.1	GeoPlot Map Chart	168
9.14.2	Image Plot Chart.....	182
9.14.3	SVG Drawing	205
9.14.4	US County Map	210
9.15	Radar Charts	213
9.16	3D Image Chart	214
9.16.1	Technical workflow.....	214
9.16.2	Supported Image Types.....	215
9.16.3	Shapes	215
9.16.4	Empty Faces.....	219
9.16.5	Chart Properties.....	220
9.17	Candlestick Chart.....	222
9.18	OHLC Chart.....	223
9.19	Word Cloud Chart	223
9.19.1	Data Layout	223
9.19.2	Excluded Words.....	226
9.19.3	Drilldown.....	227
10.	Advanced Chart Features.....	228
10.1	Using the Pivot Feature	228
10.1.1	Creating a Pivot	228
10.1.2	Customizing Pivot Appearance	230
10.2	Using the Hide-Axis Feature	232



10.2.1	Example 1: Detailed Sorting.....	232
10.2.2	Example 2: Basic Sorting using Expressions	233
10.3	Using the Function Feature.....	235
10.4	Truncate Dates	237
10.5	Using Filters.....	238
10.5.1	Operators.....	238
10.5.2	Setup Process	238
10.6	Using Filter Macros	240
10.6.1	Setting the Proper Date Type and Format.....	241
10.6.2	Testing the Date Format	242
10.7	Adding Macros to Chart Titles	244
10.8	Repeat Macro	245
10.8.1	Requirements	245
10.8.2	Overview.....	245
10.8.3	Value Macro: Recap	245
10.8.4	Repeat Macro: Syntax	246
10.8.5	Repeat Macro: Drilldown.....	246
10.8.6	Repeat Macro: Presentation Charts.....	246
10.9	Filter on User	248
10.10	Export Chart Data.....	249
10.10.1	Export All Chart Data.....	249
11.	Deleting.....	250
11.1	Delete	250
11.2	Delete Sequence	251
12.	Diagnostics.....	252
12.1	Accessing Chart Data Diagnostics	253
12.2	Diagnostics Sections.....	253
12.2.1	Chart Data Selection.....	253
12.2.2	Chart Axis Definitions.....	253
12.2.3	Drilldown Filters	254
12.2.4	Chart Parameters	254
12.2.5	Chart Data Filters.....	254
12.2.6	Chart Data Arguments	254
12.3	Export Diagnostic Details.....	255
13.	Input Parameters.....	256



13.1	Parameter Definitions	256
13.2	Configuring Input Parameters	258
13.2.1	Passing Dashboard Input Parameters to Chart Input Parameters	259
13.2.2	Control Types	259
13.2.3	Import Input Parameters	263
13.2.4	Allow Multiple Values	264
13.3	Working with Input Parameter Values	266
13.3.1	The PARAM Macro	266
13.4	Date Selection Control	267
13.5	Filtering Data with Input Parameters	268
13.6	Displaying Input Parameters in Chart Titles	271
13.7	Prompting the User for Input Parameter Values	273
13.7.1	Dashboard Input Prompts	273
13.7.2	Chart Input Prompts	274
13.8	Rules for Prompting	275
13.9	Blank Parameter Values	276
13.10	Passing Input Parameters into Custom Queries	276
13.11	Passing Input Parameters into Stored Procedures	277
13.12	Passing Input Parameters through Drilldowns	278
13.12.1	Sending Input Parameters through a Drilldown	278
13.12.2	Getting Input Parameters through a Drilldown	278
13.13	Passing Input Parameters through URL	280
13.13.1	URL Syntax	280
13.14	Picklist	282
13.14.1	Picklist Overview	282
13.14.2	Design Permissions	282
13.14.3	Create or Edit a Picklist	282
13.14.4	Picklist Designer	283
13.14.5	Saving the Picklist (or Save As)	284
13.15	Designing a Picklist	286
13.15.1	Dynamic Data	286
13.15.2	Static Data	291
13.15.3	Picklist Parameters	294
13.15.4	Completing the Picklist Definition	295
13.16	Using Cascading Input Parameters	297



13.16.1	Flow Diagram	297
13.16.2	Picklist Parameter	298
13.16.3	Picklist Filter using Parameter	298
13.16.4	Define the Cascading Parameter.....	298
14.	Analytics.....	299
14.1	Derived Columns	299
14.1.1	Introduction.....	299
14.1.2	Derived Column Concept.....	299
14.1.3	Derived Column within iDashboards	300
14.1.4	Built-in Variables.....	303
14.1.5	Data Types of Derived Columns	305
14.1.6	Order of Evaluation.....	306
14.1.7	Applying Built-in Variables to Pivots.....	307
14.1.8	Limitations for Derived Columns	307
14.1.9	JavaScript Fundamentals	307
14.2	Derived Columns Examples.....	310
14.2.1	Example 1 – Calculating differences	310
14.2.2	Example 2 – If Statement.....	310
14.2.3	Example 3 – If...Else Statement	311
14.2.4	Example 4 – Preventing Divide-By-Zero Error.....	312
14.2.5	Example 5 – Calculating Daily Count	312
14.2.6	Example 6 – Deriving a Quarterly Timeframe Pivot.....	313
14.3	Analytics: Input Parameters	316
14.4	Real-Time Analytics Examples	317
14.4.1	Example 1 – Headcount and Expense Projection	317
14.4.2	Example 2 - Sales Commission Rate Analysis.....	318
15.	Alerts.....	321
15.1	Licensing and Enablement.....	321
15.2	Chart Alert Overview.....	322
15.2.1	Alert Components	322
15.2.2	Configuration Permissions	323
15.2.3	Configuration Restrictions.....	323
15.3	Menu Overview.....	324
15.4	Configure a Chart Alert	325
15.4.1	Properties	325



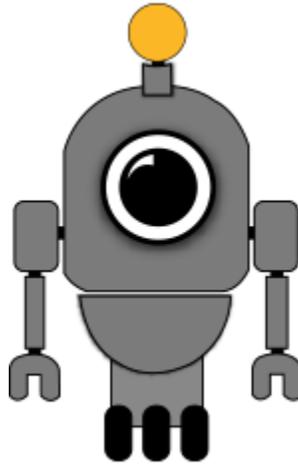
15.4.2	Rules	326
15.4.3	Schedule.....	329
15.4.4	Reactivation	330
15.4.5	Users	331
15.4.6	Summary	331
15.5	Private Alerts	331
15.6	Alert Notifications	332
15.6.1	User Alert Notification Options	332
15.6.2	Administrator Alert Notification Options.....	333
15.7	Navigating Alert Notifications	334
15.7.1	Table View	335
15.8	Viewing Alert Notifications.....	336
15.9	Charts with Alerts.....	337
16.	Reports.....	338
16.1	Licensing and Enablement.....	338
16.2	Reports Overview	339
16.2.1	Report Components.....	339
16.2.2	Configuration Permissions	340
16.2.3	Configuration Restrictions	340
16.3	Generating Reports.....	340
16.3.1	On-Demand Reports.....	340
16.3.2	Scheduled Reports	340
16.4	Accessing Report Configurations and Schedules.....	341
16.5	Configuring a Report.....	342
16.5.1	Cover Pages Sections	342
16.5.2	Headers and Footers	343
16.5.3	Chart Report Pages	343
16.5.4	Dashboard Report Pages	343
16.6	Scheduling a Report	344
16.6.1	Managing Scheduled Reports.....	344
16.6.2	Schedule Properties.....	345
16.6.3	Report Scheduling	345
16.6.4	Report Recipients	347
16.7	Reports via Filter on User	348
17.	Data Forms	349



17.1	Licensing and Enablement.....	349
17.2	Requirements	349
17.3	Creating or Editing a Form.....	350
17.4	Form Types.....	350
17.5	New Form	350
17.5.1	“Create New” Form	351
17.5.2	“From Existing Table” Form	351
17.6	Update Form.....	352
17.7	Form Designer	352
17.7.1	Form Layout	352
17.7.2	Form Footer	352
17.7.3	Form Properties	353
17.7.4	Data Fields	354
17.7.5	Data Field Groups.....	354
17.7.6	Field Properties.....	355
17.7.7	Saving the Form	357
17.8	Form Preview.....	357
17.8.1	Form Footer Buttons.....	358
17.8.2	Search	359



2. OSKAR



OSKAR, the Online Support & Knowledge Acquisition Repository, is the preferred support resource for iDashboards' customers, partners and prospects. The OSKAR Support Portal can be used to submit, review and update support tickets.

<https://oskar.idashboards.com/>

Those who have an active, support and maintenance contract with iDashboards also have access to the following content in our User Community:

- **Knowledge Base** – Numerous product and technology articles for your review.
- **Community** – Forums and discussion groups for customers to discuss various topics and products amongst themselves.
- **Resources** – Many downloadable resources that can be used with iDashboards.
- **Ideas** – Area for customers to submit feature requests and great product ideas.
- **Blog** – Thoughts, stories, and ideas on data and dashboards

3. Introduction

3.1 iDashboards URL

Using a web-browser, access iDashboards via:

Interface	URL
Lobby	http://<servername>/idashboards/

Bookmark
any of
these
URL's!

From the Lobby, there are additional URLs:

Interface	URL
Viewer	http://<servername>/idashboards/view
Builder	http://<servername>/idashboards/build
Admin	http://<servername>/idashboards/admin

Note: Each of the URLs above can be bookmarked for direct access after login!

3.2 iDashboards URL Details

The URL for iDashboards is the web address used to access settings with a web browser. To determine it, you must know three things:

1. The hostname or IP address of the server on which iDashboards is deployed.
2. The port number on which the iDashboards application server is listening, if it is other than port 80, which is the default HTTP port number.
3. The “context root” under which the iDashboards application was deployed. Normally this will be “idashboards.” See the Admin Manual for more information.

Once these three components are known, the URL of the User Application will be:

`http://<servername or IP address>:<port number>/<context root>`

For example, if the hostname is “dashmachine”, the port number is 8080, and the context root is “idashboards”, the URLs for iDashboards would be:

Interface	URL
► User Application	http://dashmachine:8080/idashboards/
Auto Uploader	http://dashmachine:8080/idashboards/
Application Server Console	http://dashmachine:8080/idashboards/admin/
Data Hub Application	http://dashmachine:8080/idbdata/
<i>If the port number is 80, it can be omitted from any of the URLs, as seen below:</i>	
User Application	http://dashmachine/idashboards/
<i>In some cases, such as when the server is being accessed over the Internet, it may be necessary to append a domain name to the hostname, as seen below</i>	
User Application	http://dashmachine.mycompany.com/idashboards/

► = Topic included within this manual



3.3 System requirements for accessing iDashboards

- **Network/Internet connection** – To the iDashboards Application
- **User Application** – Any current browser supporting HTML5 content.
- **Auto Uploader (optional)** – This allows for direct uploading of Excel files from a client machine to an iDashboards instance without requiring the user to log into the iDashboards Application through a browser. Refer to the Administrative Application for additional details.
 - Microsoft Windows 7 or Windows Server 2008 R2
 - Microsoft .NET 4 Framework - Full profile

3.4 Login Requirements and User Roles

Access to iDashboards requires a username and password. Users will be prompted with the iDashboards login prompt when they open the application. A user may change her/his password after logging-in, but not the username. Every user that is created in iDashboards must be assigned to one, and only one, User Role. There are four User Roles, as follows:

- **Viewer** – Users assigned to this role have read-only access to dashboards within the categories where they have permissions. They can, however, fully interact with charts and dashboards. They do not have a “Personal” category and there are limited menu options.
- **Builder** – Users assigned to this role have full dashboard, chart and picklist building capabilities within the categories where they have ‘save’ permissions. They can upload files to the “Content” directory and they do have a “Personal” category.
- **Data Admin** – This role allows all Builder role functions, plus the ability to log into the Data Hub for performing data manipulations and Excel file uploads.
- **Admin** – Users assigned this role have all the permissions of the Data Admin role and are the only users that can log into the Administrator Application and perform administrative functions.

3.5 Login Experience

The iDashboards login experience will vary based on the user role and the device type used during the login process.

Security Warning! Auto-Login

It is possible the browser has been asked to remember the login credentials for an iDashboards user. Therefore, navigating to the iDashboards URL from a newly opening a browser could populate the login fields and complete the login process without manually typing in the login credentials.

This feature is managed by the browser, not iDashboards. Use care when attempting to leverage this feature.



3.5.1 Mobile

The iDashboards Mobile interface offers users a browser-based experience to view dashboards specifically from a small(er) device screen. The mobile interface is automatically determined by the device type used during the login process – often identified as a smartphone or portable/tablet device.

Note: If you do not want to experience the mobile interface, it may be possible to use a browser option to request the desktop site.

The mobile interface is a 'view-only' interface, regardless of your user role within iDashboards. Only one dashboard can be opened at a time. While charts and dashboards cannot be created or edited, dashboards with charts can be viewed and will be fully interactive.

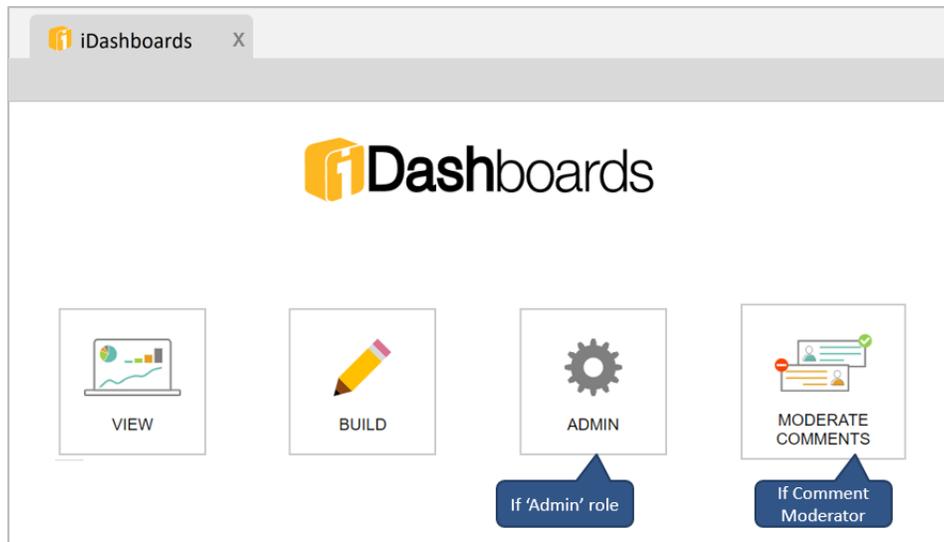
For more information, access the Mobile Help information within the mobile interface.

3.5.2 Lobby

The “Lobby” is a special landing page that only appears when two conditions exist:

1. Device type = Desktop, Laptop and Tablet
2. User role = Anything except a “Viewer” role (Builder, Data Admin or Admin)

The Lobby requires you to navigate into the interface you want to experience. As illustrated in the image below, if your user role is Builder or Data Admin, you will have the Lobby options “View” and “Build”. However, having the Admin user role includes the option for “Admin”. Also, if you are a “Comment Moderator”, that option will be provided.



Note: User's with the “Viewer” role do not experience the Lobby, as these logins will automatically go into the “View” interface.

3.5.3 View

The “View” (or “Viewer”) interface is the primary location for opening and interacting with dashboards, regardless of your device type or user role. Here, this interface will offer the ability to search or navigate through all of the categories you are permitted to see, and then open and interact with the dashboards.



Note: After login, users with the “Viewer” role will automatically go into the “View” interface. Otherwise, all other user roles can manually select the “View” interface.

For more information, refer to the HELP menu option within the View Interface.

3.5.4 Build (this manual)

The “Build” (or “Builder”) interface is where charts, dashboards and picklists are created, edited, saved and deleted. Building dashboards is far more complex than viewing dashboards, so this manual will be particularly useful for those responsible for creating dashboards.

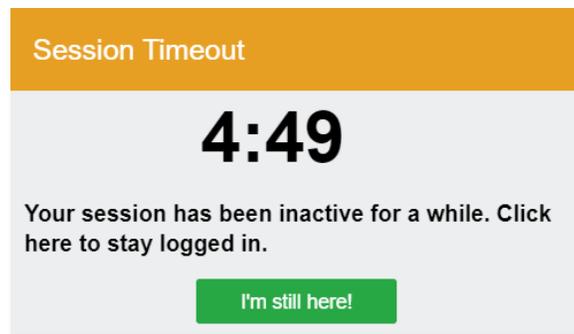
3.5.5 Admin

The “Admin” (or Administrative) interface is where the iDashboards application is fully administered. This navigation option only appears for users with the “Admin” role. This interface manages user accounts, save permissions, data source management, system configuration, security settings and more.

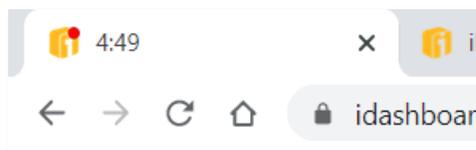
Refer to the Administrative Application and the Administrative Manual for additional details.

3.5.6 Log out

Logging out of iDashboards can help prevent unapproved access on a computer/device that is unattended. Manually logging out is one method to promote a secure environment. Alternatively, iDashboards has a timeout mechanism that can log out a user after a set number of inactive minutes. Prior to a session timeout, the user will be notified via dialog screen, as shown below.



Note: The session timeout might also change the browser tab depending on the browser and device type being used.



4. Build Interface

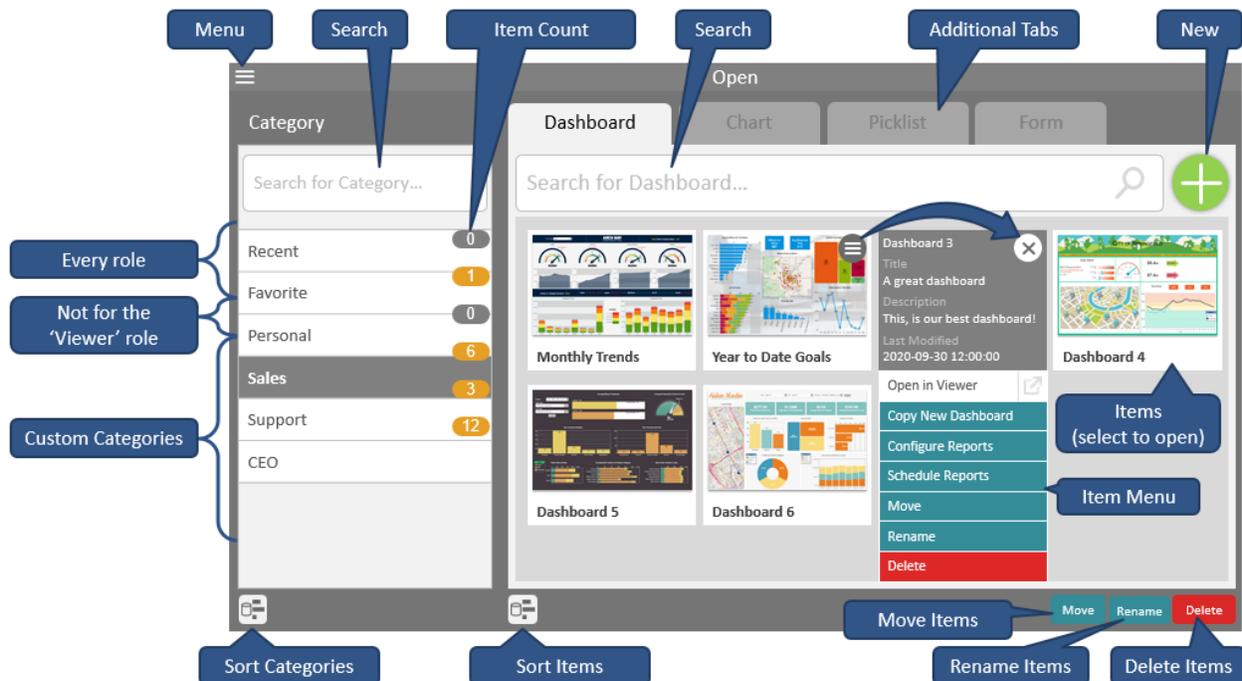
The “Build” (or “Builder”) interface is where charts, dashboards, picklists and forms (if enabled) are created, edited, saved and deleted. Building dashboards is far more complex than viewing dashboards, so this manual will be particularly useful for those responsible for building dashboards.

The “View” interface is a 'view-only' interface, regardless of your user role within iDashboards. While charts and dashboards cannot be created or edited, dashboards with charts can be viewed and will be fully interactive.

The **three** main purposes for using the Build Interface are:

1. **Creating** – Using the “New” button, a builder can create charts, dashboards, picklists or forms (if enabled). These items can only be saved within the Categories where the user has “Save” permissions.
2. **Editing** – Using the “Open” options, a builder can make edits to existing charts, dashboards, picklists or forms (if enabled). These items can only be re-saved into the Categories where the user has “Save” permissions.
3. **Deleting** – Using the “Delete” button, a builder can delete existing charts, dashboards, picklists or forms (if enabled). These items can only be deleted within the Categories where the user has “Save” permissions.

Note: The “Build” interface is not sufficient for viewing or experiencing a dashboard.



Upon navigation to the Build Interface, the user will see the home screen – as seen above. The Build Interface is nearly the same as the View Interface noting the following differences:



	View	Build
Multiple opened dashboards	Yes	No
“Personal” category guaranteed	Based on user role	Yes
Additional navigation tabs for “Chart”, “Picklist” and “Form”	No	Yes
“New” button	No	Yes
Build/Edit dashboard	No	Yes
“Move”, “Rename” and “Delete” buttons	No	Yes
Configure and Schedule Reports	No	Yes
Interactive dashboard	Yes	Yes*
*A temporary ‘preview’ experience is available while building a dashboard.		

4.1 Requirements

The requirements for using the Build Interface include:

- Access to the iDashboards Application URL.
- Authentication via log in.
- An HTML5 browser on a desktop device.
 - Logging in with a mobile device will take you to the view-only mobile interface.

4.2 Menu – Basic

When building dashboards, the menu is only accessible on the home screen and not within the building screens. To access the menu, select the Menu button, as seen in Figure 4-1.

Note: The menu options change when a dashboard is opened.

4.2.1 Go to Lobby

This setting is available all the time. This option will exit the Build Interface and return the user to Lobby.

4.2.2 User Settings

There are a number of settings a user can configure that pertain to the user experience and security of iDashboards.

Note: The username can only be changed using the Admin interface.

- **Username** – This field will display the currently logged in user. Usernames can only be added, edited or deleted by the iDashboards administrator.



- **Change Password** – This option allows you to confirm your current password and then enter a new password. If you forget your password, ask your iDashboards administrator to assign you a new temporary password that you may change after you log in.
- **Language** – Each user may choose which language iDashboards uses for menu items, navigation, and messaging within the dashboard framework. To change the language preference, select the desired language in the ‘Language’ dropdown box.
 - Languages are added, removed and updated by the iDashboards administrator, by means of downloading and installing a language pack. Language packs affect the text that is built in to the iDashboards application. Language packs will not affect any text that is pulled from a Data Source as it comes directly from the data source. The iDashboards documentation is written in US English.

Note: Contact iDashboards for availability of Language Packs, or check OSKAR to download current Language Packs (see Chapter 2, “OSKAR”).

- **Theme Color** – You may choose which color iDashboards uses for the border of the dashboard framework. To change your preference, select the currently defined color, and choose the desired color from the selector.

Note: Colors prefixed with an asterisk () indicate usage may produce difficulties in visually discerning certain elements (dashboard icons, etc.) due to masking. It is suggested to refrain from using these colors except in cases when these elements are not going to be used.*

- **Sort Order** – The Category and Dashboard list can be sorted independently from one another by using the sort icons beneath each section on the home screen. Alternatively, users can save their preferred sort option by selecting one of the options shown below:



Server Order (default): Sort the items as determined by the iDashboards administrator.



Alphabetical Ascending: Sort the items alphabetically in ascending order.



Alphabetical Descending: Sort the items alphabetically in descending order.

Note: The “Recent”, “Favorite” and “Personal” Categories are always displayed at the top when sorting. However, the Dashboards within these categories can be sorted when toggling through the dashboard sort options.

- **Email Address** – This is an optional field but is required for emailing alert notifications.
- **Data Source Caching** – When performing tasks where the “Chart Designer” appears, iDashboards performs a check against the available data sources and tables that are available to the user. This check consumes a negligible amount of time, and the window

will not appear until the checking process is complete. However, in cases where there are excessive numbers of data sources or tables, some users may experience a noticeable amount of time. Therefore, in case-by-case scenarios, it may make sense for a user to cache the available data sources and tables so subsequent launches of the Chart Designer will launch faster. Enabling this feature involves setting a 'minute' value greater than zero. The minute value defines the number of minutes to keep the current cache, ranging from 0-60 minutes.

The tradeoff for caching this information occurs when changes are made to the data sources or tables, because the user will not see these changes until the defined number of minutes or until the user logs out and back in.

- **Periodically check for new Alerts** – Enable this option if you want iDashboards to check for alert notifications as soon as you log in, and at regular intervals afterward. You may want to disable this option if you plan on presenting dashboards to an audience and do not want interruptions from an Alert.
 - **Minutes between checks** – This option will set the frequency with which iDashboards looks for new notifications, from once per minute to once per hour (1-60). Note that this is not the same thing as checking an individual alert to see if it will trigger. This check is for new notifications from any alert which has triggered since the last check.
- **Receive alert notifications by email** – If enabled, this option let you choose to receive alert notifications by email, if an alert is set to send them. The email address field must contain the user email address.
- **Receive alert notifications by text message (SMS)** – If enabled, this option let you choose to receive alert notifications by mobile SMS text message, if an alert is set to send them. The text message fields must contain the user mobile device information.
- **Mobile carrier** – The iDashboards Administrator configures the Mobile carrier list through the administrative application. Select the carrier for your device, or contact your administrator to request additional carriers.
- **Mobile country code** – Enter your specific mobile dialing country code. iDashboards will process each mobile SMS text notification with consideration for the country code. For example, the U.S. has a country code of “1”.
- **Mobile phone number** – This field is for entering the mobile phone number for SMS text messages. For example, the U.S. has a 10-digit phone numbering system, beginning with the area code.
- **Long message behavior** – The iDashboards administrator has an option to set one of the options below as the “System Default”. However, users can override personal settings to experience different results. Each carrier has a configuration for handling long text messages. Keep in mind carriers usually have a text message limit of 160 characters.



- **Send Entire Message** – This option will attempt to send a text message of any character length. If the character length exceeds 160, then the carrier provider will handle message segmentation.
- **Truncate Message** – This option will ensure the iDashboards text message will be reduced at-or-below 160 characters before sending the message.
- **Multiple Messages** – This option will have iDashboards segment any messages greater than 160 characters, sending multiple messages if necessary.
- **Multiple Messages (Reverse Order) <default>** – Same as ‘Multiple Messages’, but the sending order will be in reverse order (sending the last message segment first, and continuing until the first segment is sent last)

4.2.3 About

Select this option to see information about the iDashboards license. Here, the screen will list the exact version of iDashboards installed on the server and some basic license information.

4.2.4 Help

This manual 😊

4.2.5 Log Out

Select this option to log out of iDashboards.

Security Warning!

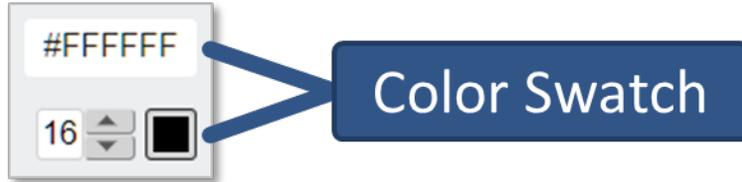
Closing the browser, or browser tab, is not sufficient for logging out of iDashboards. Selecting “Log Out” is the only way iDashboards can ensure the session has ended.

It is possible the browser has been asked to remember the login credentials for an iDashboards user. Therefore, navigating to the iDashboards URL from a newly opening a browser could populate the login fields and complete the login process without manually typing in the login credentials.

4.3 Colors

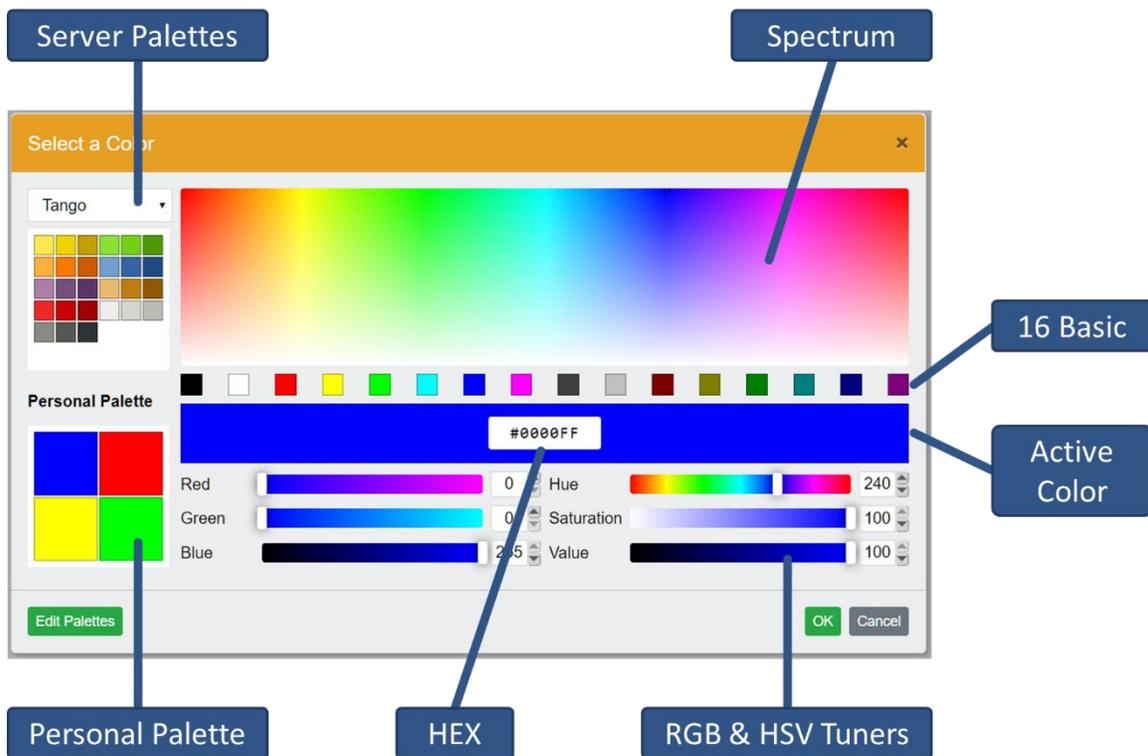
Editing colors is an essential function to customizing the final dashboard. Default colors will appear in many areas of a chart and nearly all are customizable! For example, there are customizable text, background, range colors, frames, notes, and so much more.

Options to change a color exist when a large or small color swatch (or 'button') is available (see below). Clicking on any color swatch will open the color picker.



4.3.1 Color picker

iDashboards offers a variety of predefined colors. Click on a color to use the color. The image below illustrates the various areas where colors can be selected. iDashboards offers predefined colors or the option to discover a new color.



4.3.2 Palettes

If certain colors become common during the dashboard building process, there are two options to save a customized palette. Users can create, edit and save additional color palettes.

4.3.2.1 Personal Palette

The Personal Palette acts a list of the 16-last used colors. These colors are only saved locally and should not be considered a permanent assortment of colors for future use since they have a disposable behavior.

Note: A 'Personal Palette' is only available from the computer used to create the palette. This palette will not reappear when logging in from different computers. Another user who logs into your computer will see the same Personal Palette. Although users should consider this a temporary palette since it is cached, it is likely to remain available for extended periods of time.

4.3.2.2 Server Palettes

If a color palette needs to be preserved and recalled over time and it is laborious to restore each of the customized colors, it makes sense to use a server-saved color palette. These color palettes can be advantageous when all users within the iDashboards environment need access to the same set of colors. The colors may be re-used in different parts of the iDashboards application, and will also be visible to other users in the organization. During the color selection process, users can change the palette to a custom shared palette and standardize the dashboard building process. There can be 100 colors in a shared palette.

iDashboards comes pre-configured with at least one server palette. These palettes cannot be edited or removed.

There are two methods for creating, editing, or deleting palettes:

1. From the Builder home screen, within the menu, select "Edit Palettes".
2. From any color picker dialog, select the button "Edit Palettes".

To create a color palette:

- Begin by navigate to the "Edit Palettes" dialog as described above.
- Next, click on the "New Palette" button (the "+" icon below the pallets list.)
- Next, update the palette name.
- Configure a color using the Spectrum, HEX, RGB or HSV controls.
 - Drag-and-drop the "Active Color" color into the square frame below the palette name.
 - Note: colors are added where they are dropped. Colors can be rearranged using drag-and-drop.
 - Alternatively, to add the "Active Color", click on the "Add Color" button (the "+" icon below the palette colors box.)



- Note: colors are added in the order applied. Colors can be rearranged using drag-and-drop.
- Repeat the process of adding multiple colors until a palette of colors exists.
- Click the “Save” button.
- Lastly, click the “Done” button.

To edit a color palette:

- Begin by navigating to the “Edit Palettes” dialog as described above.
- Next, select the palette name from the list of existing palettes.
- Edit Options:
 - To remove a color, select the color from the palette, then click the “Delete Color” button.
 - To add a color, configure a color using the Spectrum, HEX, RGB or HSV controls.
 - To rearrange a color, use drag-and-drop to reposition any color.
 - To rename a palette, edit the name text of the palette once it is selected.
- Once done altering the palette, click the "Save" button.
- Lastly, click the “Done” button.

To delete a color palette:

- Begin by navigating to the “Edit Palettes” dialog as described above.
- Next, select the palette name from the list of existing palettes.
- Next, click the “Delete Palette” button.
- Lastly, click the “Done” button.



4.4 Menu – Dashboard

This menu is only available when viewing, not building, a dashboard. To access the menu, click the Menu button.

Note: The menu options change when a dashboard is opened.

4.4.1 Refresh Dashboard

Select this option to force each dynamic chart to re-query the data source. This will result in displaying the most recent information for the charts of the current dashboard being viewed.

4.4.2 Identify Favorite Dashboards

iDashboards can organize your favorite dashboards to make them easier to locate in the future.

While a dashboard is open, select “Add as Favorite” to save the currently opened dashboard into your personal list of favorite dashboards. To open a favorite dashboard, select the “Favorites” category and then select the dashboard(s). To remove a dashboard from your list of favorites, first open the dashboard. Then select “Remove as Favorite”. This action will not delete the dashboard.

4.4.3 Set as Startup Dashboard

A Startup Dashboard is a concept similar to a Home Page within a web browser. As soon as you launch iDashboards (and log in), the Startup Dashboard will automatically open within the dashboard viewing area. Only one dashboard can be set as the Startup dashboard.

Select this option to choose the currently opened dashboard as your startup dashboard. The next time a log in occurs, the startup dashboard will automatically open.

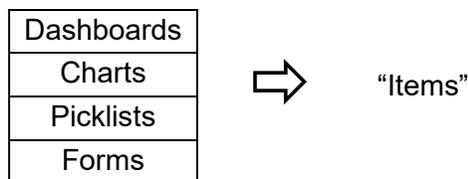
4.4.4 Enter Dashboard Parameters

If a dashboard has been configured with Input Parameters, and those parameters are set to “Always prompt for parameters on dashboard load”, then the user will be prompted to provide parameter values prior to the dashboard being shown. Once the parameters have been submitted, the dashboard will be displayed, and the parameter dialog could close from view.

Selecting this option will open the input parameter prompt after it has been closed, enabling modification of the dashboard parameter values at any time.

4.5 Opening

For the purpose of simplifying this concept, this section of the documentation will refer to “Dashboards”, “Charts”, “Picklists”, and “Forms” (if enabled) as “Items”.



Note: The Build Interface can only open one dashboard at a time.

Note: Hover on an item to see additional details



Note: Opening dashboards or charts can also occur when using various dashboard features (ex. the Dashboard Launcher or Drilldown features.)

Similar methods are used when opening dashboards, charts, picklists or forms (if enabled). First select the tab. Then, from the home screen select the item (i.e. click on it) to open it.

4.5.1 Sorting

The Category and item list can be sorted independently from one another by using the sort icons beneath each section.



Server Order (default): Sort the Categories or Dashboards as determined by the iDashboards administrator. Only Categories and Dashboards can be manually sorted by the administrator.



Alphabetical Ascending: Sort the Categories or Items alphabetically in ascending order.



Alphabetical Descending: Sort the Categories or Items alphabetically in descending order.

Note: The "Recent", "Favorite" and "Personal" Categories are always displayed at the top when sorting. However, the Items within these categories can be sorted when toggling through the dashboard sort options.

4.5.2 By Category

From the home screen, items can be manually located by selecting the various categories on the left which will then display the related items on the right. The number of items within the Category are shown and empty Categories are not hidden from view.

Note: The user role "viewer" does not contain a "Personal" category.

Note: Empty Categories are hidden from view within the View Interface.

4.5.3 By Searching

From the home screen, items can be located by using the dynamic search bar. Categories and Dashboards can be searched independently. Only items with matching values will appear in either list. When a search value has been entered, select the Category name to see the list of related items. Clear each search box to display all Categories and Dashboards.

5. Categories

Note: Refer to the Administrative Application for additional details.

Similar to how file folders organize documents within a computer environment, categories help organize dashboards, charts, picklists or forms (if enabled) within the iDashboards environment. Each iDashboards user could see different Categories when logged into iDashboards; this is due to the permissions and access control determined by the iDashboards administrator.

- All users have a “Recent” and “Favorites” category
- Except for users assigned the “Viewer” role, all other user roles have a “Personal” category
- Custom categories can only be created, edited, or deleted by the iDashboards administrator.
- Empty categories are hidden from the View Interface

5.1 Personal Category

The first Category is always “Personal”. As the name implies, the Personal Category is a personalized category for each user account except for users with a User Role of Viewer, and no other users can see its content. As a user you can create and save dashboards and charts within your Personal Category.

5.2 Creating Categories

Creating Categories is an administrative function, requiring a user to log into the iDashboards Administrator Application. For more information on this process, refer to the Administrative Application for details.

5.3 Category Sort Order

The category display can be sorted. Temporary sorting is achieved when using the sort buttons. Retaining the sort preference with your next login is possible when configuring the user settings.

5.4 View-Only Categories

An ‘eye’ icon denotes that the category is view-only. Any items edited in this category will not allowed to save to any view-only category. Categories that are not view-only will be available.

5.5 Protected Categories

A ‘lock’ icon indicates the content saved within the category has been authored, and protected, by iDashboards. Contact support for more information.



6. Designing Dashboards

Designing a dashboard can be fun! Like many technical projects, the more planning that occurs beforehand, the better your experience will be when it comes time to build each chart and each dashboard. Over time, these tasks will become familiar and easier to use based on repetition and experience. For training and tips, refer to Chapter 2, “OSKAR”.

6.1 Design Permissions

The ability to access the Lobby and then navigate to the Build Interface is allowed with a minimum user role of “Builder”. This user role will allow the minimum functionality of creating charts or dashboards to be saved within the “Personal” category. But the ability to save items into customized categories requires access rights provided by the iDashboards administrator. Such access rights are provided at the Category level, and all dashboards, charts, picklists and forms (if enabled) within any given Category inherit those access rights.

Three access rights:

- Save Access
- View Access
- No Access

Therefore, depending upon the access rights assigned to you, you may or may not have the rights to make changes and save those changes to an existing dashboard. However, you can always make changes to an existing dashboard and save those changes as a different dashboard within your Personal Category or any other Category to which you may have Save permissions.

Note: The remainder of this manual assumes a proper user role and access rights are configured

6.2 Create or Edit a Dashboard

While the goal between creating and editing a dashboard may be quite different, the interface will be the same.

Create – To create a new dashboard, make sure the “Dashboard” tab is selected on the home screen, then select the “New” button as seen in the image below.

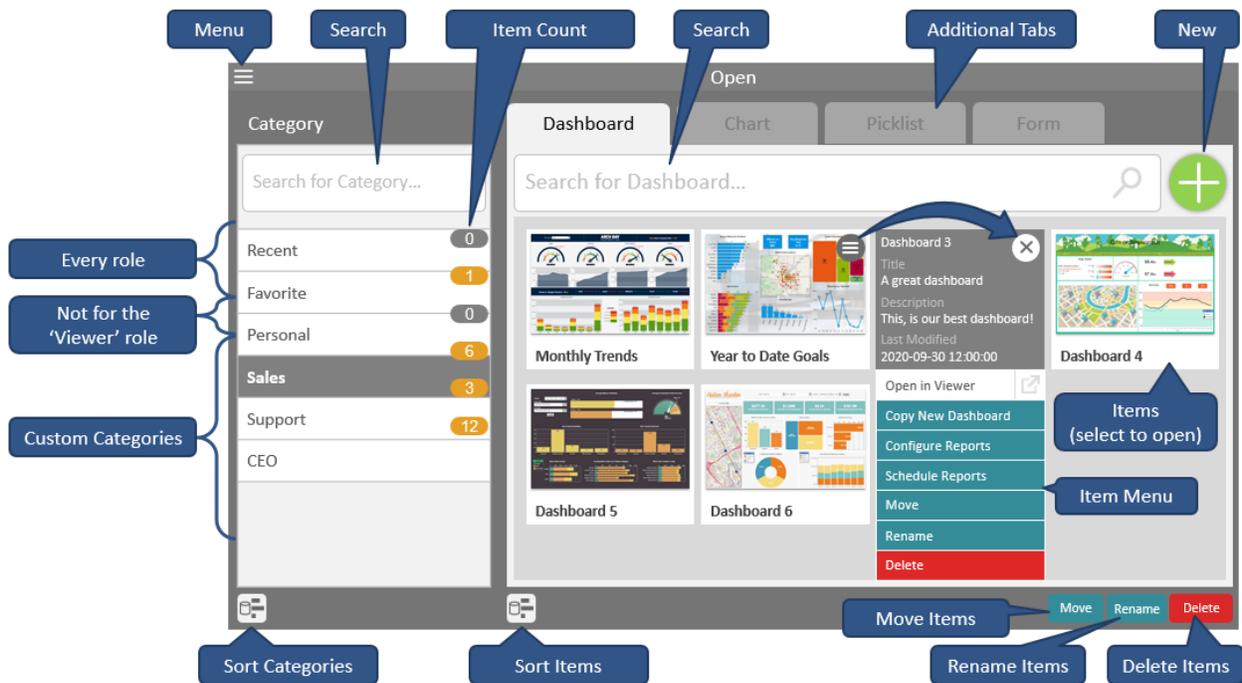
Edit – To edit an existing dashboard, there are two options based on the currently logged-in interface:

- **From the View Interface:** Click the “Edit” button (pencil icon) in the title bar to open the Dashboard Designer for the current dashboard.

Note: This does not indicate the user account has access rights to make modifications to the dashboard.

- **From the Build Interface:** Open the dashboard, and the interface will automatically open the Dashboard Designer.





6.3 The Dashboard Designer

Building dashboards takes place within the “Dashboard Designer”. Here a user can create the dashboard frames and then insert all of the charts, panels and images to complete the entire dashboard.

6.3.1 Top-Level Properties

These are dashboard properties that are applied across all pages, layouts and views.

They are:

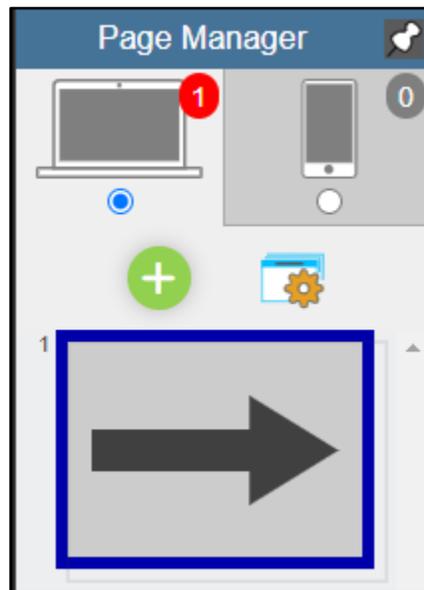
- **Extended Dashboard Properties**
 - Dashboard name for the dashboard selector
 - Dashboard Category
 - Dashboard Description
 - Mobile Display Mode (*defines dashboard’s page scrolling on mobile device*)
 - Mobile Orientation Lock (*controls the dashboard’s orientation on mobile device*)
 - Thumbnail policy
 - Hide in Dashboard Lists
- **Schedule Report**
 - A report can be scheduled on the desktop layout only



6.3.2 Page Manager

Dashboards may have multiple “pages” that can be scrolled through when the dashboard is loaded. Even though a dashboard page is very similar to a standalone dashboard, the page does not exist independently of its containing dashboard and cannot be loaded directly as a standalone dashboard.

When creating a new dashboard, the “Page Manger” panel will open as expanded and pinned into view.

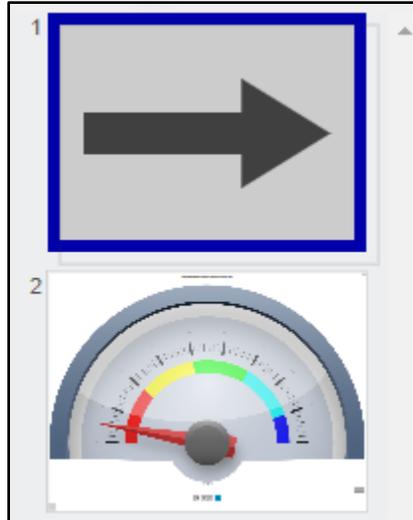


The  button will toggle the collapsing/expanding of the panel. When collapsed, placing the cursor over the collapsed bar will expand it and leave it open as long as the cursor pointer is over the panel. Once moved away from the panel, it will collapse. The  button can be used to lock the panel as expanded.

6.3.2.1 Page Navigation

The dashboard that is active in the main viewport is identified with a page-thumbnail designed as a “Right Arrow” and bold border. Other pages will have a thumbnail of the actual dashboard. Selecting another page makes it active in the designer. Pages can even be re-ordered by using drag-n-drop.

Note: Only page 1 will be used to generate a thumbnail of the dashboard



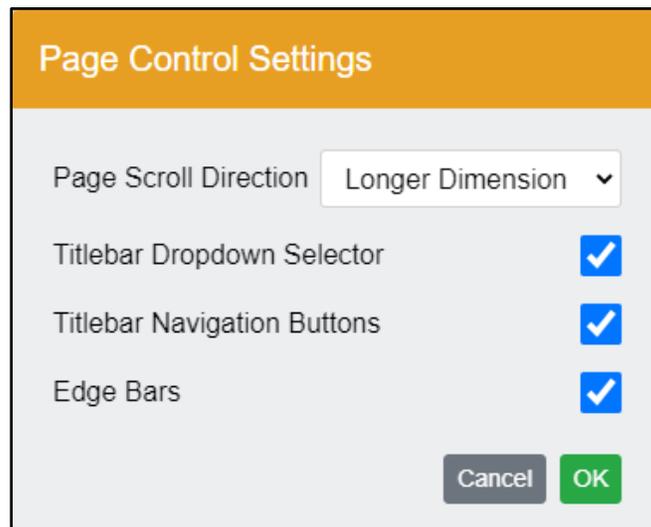
6.3.2.2 Page Manager Controls

Whether using the desktop layout or the mobile layout, multiple dashboard pages can be added, independent of each other. If there are multiple pages, the layout's icon will show a page count. See 6.3.4 Designer Dashboard Layouts for layout details.

	Add new blank page
	Configure the page control settings for the layout
	Create a copy of the page
	Delete the page

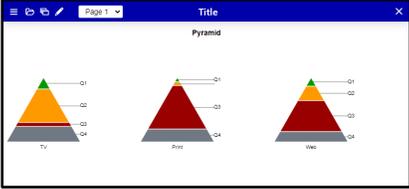
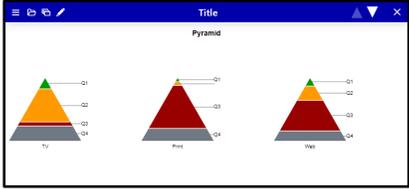
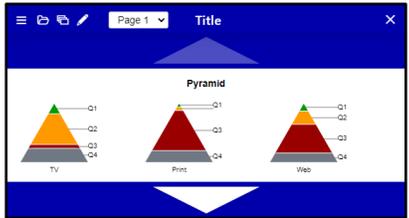
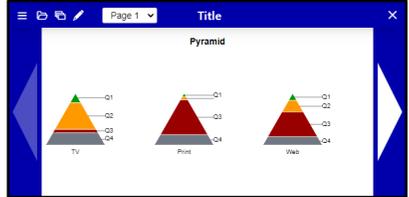
6.3.2.2.1 Page Control Settings

These settings define how the page scrolling will look and work.



- **Page Scroll Direction**
 - Vertical
 - Horizontal
 - Longer Dimension (*default*)
 - Shorter Dimension
- **Page Control Type**
 - Titlebar Dropdown Selector
 - Titlebar Navigation Buttons
 - Edge Bars

Examples:

	<p>Titlebar Dropdown Selector</p>
	<p>Titlebar Navigation Buttons</p>
	<p>Edge Bars with Vertical Scroll</p>
	<p>Edge Bars with Horizontal Scroll</p>



6.3.2.3 Page-Level Properties

These are properties that typically affect the appearance of a dashboard page, and which can be configured in a different way for each page. Basically, any property that is not a layout-level or a top-level property is likely a page-level property.

Here is a list of those properties:

- **Dashboard**
 - Lock Borders
- **Boorders**
 - Color
 - Width
 - Transparency
 - Show
 - Handles
- **Frames**
 - Display Order
- **Background**
 - Color
 - Image
 - Sizing
- **Scaling**
 - Scaling
 - Aspect Ratio

Note: Charts can be created within the Dashboard Designer, or independently. When created independently, charts can later be placed into dashboard frames.

6.3.2.4 Multiple Page Thumbnails

A multiple page dashboard is a Default and/or Mobile layout with more than one page. When dashboard thumbnails are enabled (see *Thumbnail Configuration in Administrator Manual*) a multiple page dashboard thumbnail will be used.



Note: If dashboard thumbnails are disabled, a  icon will indicate multiple pages.

6.3.3 Designer Controls on the Title Bar



6.3.3.1 Switch View Mode

While in the designer's desktop layout the default view of the dashboard is for a desktop interface. While in the mobile layout the default view of the dashboard is for a mobile interface.

To work on the desktop layout, as it would be displayed on a mobile device, you can switch to a mobile view. Conversely, to work on the mobile layout, as it would be displayed on a desktop interface, you can switch to a desktop view.



Switch to mobile view (*the green desktop signifies current view is desktop*). In mobile view, the dashboard displays in a simulated mobile view, and will open in portrait preview mode.



Use the  control to toggle preview between portrait and landscape modes.

Note: This button will be inactive, on the desktop layout, if there is a Mobile layout defined.

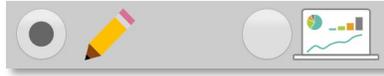


Switch to desktop view (*the green mobile signifies current view is mobile*)

Note: View modes are not the same as dashboard layouts.

6.3.3.2 Live Dashboard Preview

When designing a dashboard, the Build interface doesn't allow for chart interaction (cursor highlighting, drilldown, pivot changes, parameter changes, etc.) because all of the commands are focused on frame editing and dashboard design. Therefore, to experience the View interface without opening a new tab or leaving the build mode, simply toggle the radio button shown below.



While in the View 'mode' it is possible to test the functions of the dashboard to ensure everything is working properly – the cursor will interact, charts will have drilldown capabilities and everything else will be functional. It is also possible to edit charts and forms and charts and dashboards opened through a launch panel or drilldown. This is accomplished through the use of the mouse's right-click, which opens a menu with the appropriate Edit Chart, Edit Dashboard or Edit Form options. A new Browser tab will open with the editor for the item selected.



For a chart, if it is using the 'Right-Click Menu Enabled' setting, the menu will include the standard chart menu options (see 8.5.2.16 Right-Click Menu Enabled.) Once the dashboard is ready for production, toggle back to the Build interface and save the dashboard.

6.3.4 Designer Dashboard Layouts

A dashboard has at least one, and possibly two layouts. A layout is how the dashboard appears to the user, depending of the device with which they view it. The two possible layouts are:

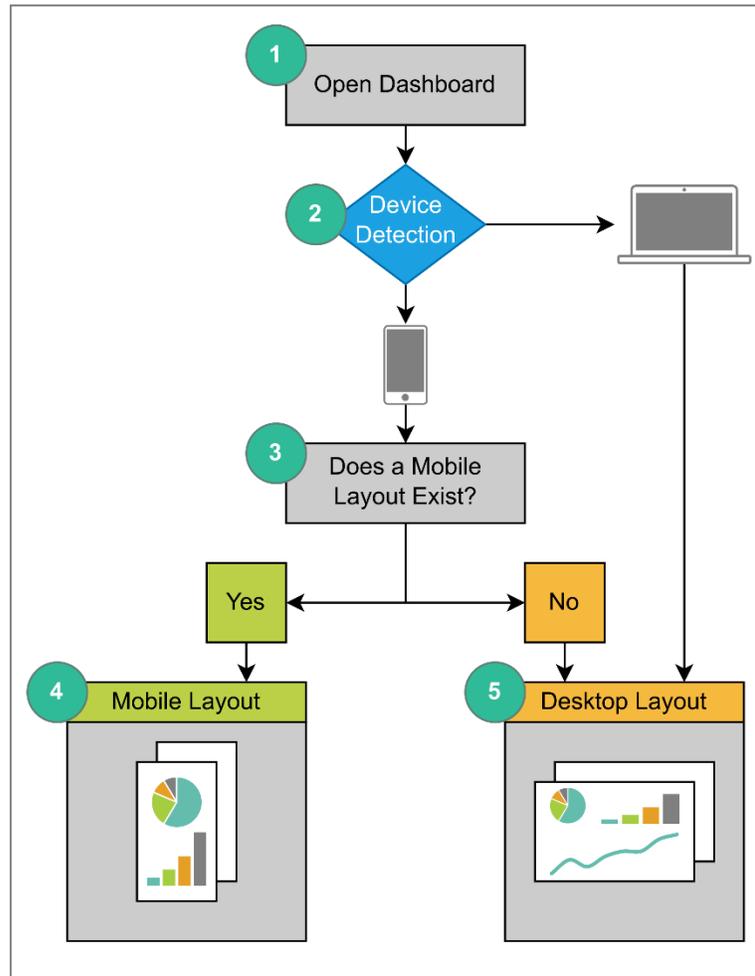


Desktop Layout – This variant always exists and is the default layout for using iDashboards. As named, this layout will be displayed when viewing dashboards from a desktop viewer interface (and will also be displayed in the mobile viewer interface if no mobile variant exists.) Be aware that a report can only be scheduled on the desktop layout.



Mobile Layout – (optional) – Creating a mobile layout will cause all mobile devices to experience only the mobile layout and not the desktop layout. This will allow mobile devices to experience different dashboards that are uniquely designed for mobile devices like phones and tablets.

If a mobile layout does not exist (or all mobile pages are deleted), then all mobile devices will experience the desktop layout.



When first selecting mobile layout there is a prompt for what you would like copied from the desktop layout.

- **Everything** – Copy all desktop pages, including all dashboard parameters.
- **Parameters Only** – Only copy the dashboard parameters and none of the pages or page layouts (including any parameter panels)
- **Nothing** – Start with new, single-page blank dashboard.

In mobile layout, the dashboard displays in a simulated mobile view, and will open in portrait preview mode. In mobile view, use the  control to toggle preview between portrait and landscape modes.

Note: Dashboard layouts are not the same as view modes

6.3.4.1 Layout Level Properties

These are properties that are configured separately for each layouts, if a separate mobile layouts has been created.

- The dashboard title, and whether or not it will include the category name.
- The dashboard parameters.
- The auto refresh settings. For example, a desktop layouts might auto fresh itself every 10 minutes, while the mobile layouts does not auto refresh at all, in order to not burn the user's data plan and battery.
- The page controls that will be displayed for a multipage layouts.
- The report configuration. Reports can be configured separately for the default and the mobile layouts.

6.3.5 Designer Toolbar

The Dashboard Designer uses a graphical interface comprised of icons and fields. Each icon has a mouse-over description and this section will further define the function of the icon. When interacting with the screen, notice the coordinates and status bar. The toolbar can be docked to the top or bottom of the viewport. To reposition the toolbar, drag the vertical grips to a new position on the screen.

Coordinates – Use the X and Y coordinates to accurately position and align frame borders.

Status Bar – Provides information regarding frames.

6.3.5.1 Toolbar Autoshrink

Most buttons have a label to identify their function. However, when the browser is not wide enough to display all of the labels, some labels will automatically disappear and condense to the width of the browser to show all of the buttons. This functionality can be enabled or disabled, per device, by expanding the “Save” dropdown and changing the autoshrink setting. Whether this setting is enabled or disabled, the toolbar can also be scrolled left-right with the mousewheel or via moving the mouse to the left-right of the toolbar.

6.3.6 Icon Overview

The three sections of the Dashboard Designer toolbar are illustrated below. Together, they organize a variety of icons into three sections: DASHBOARD, CHART and TOOLS:



DASHBOARD	
	Undo the last modification <i>Note: (“Save” operations cannot be undone)</i>
  (a red glow will pulsate if there are unsaved changes on the dashboard)	This will save the current state of the dashboard via: <ul style="list-style-type: none"> • Save Dashboard <ul style="list-style-type: none"> ○ Saves the already-named dashboard. • Save Dashboard and Open in Viewer <ul style="list-style-type: none"> ○ Saves the dashboard, then leaves the Build Interface to open the same dashboard within the View Interface. • Save Dashboard As... <ul style="list-style-type: none"> ○ Required for new dashboards. ○ Creates a new dashboard with a new Dashboard ID. This option also allows for overwriting an existing dashboard. <i>Note: (these operations cannot be undone)</i>
	Add, Remove, and edit Dashboard Parameters
	Configure the Dashboard Properties, including mobile display mode

 or 	Divide the selected frame either vertically or horizontally. The spacing is determined by the “auto-space” setting being either enabled or disabled.
 or   or 	Merge the selected frame with an adjacent frame – either vertically or horizontally. Depending on the selected frame, these icons may change directions and a merge direction may not always be possible.
 or 	Lock or Unlock the frame borders (Effective only in the View Interface.)
 or 	Enable or Disable auto-space frame borders when splitting frames
Borders	
<input type="text" value="6"/> Width 0-50	Frame border thickness
<input type="text" value="0"/> Transp. 0-100	Frame border transparency
<input type="color" value="#0070C0"/> Color	Frame border color
 or 	Show or Hide frame borders
 or 	Use, or don't-use, the default frame border color which is determined by the user's currently selected theme color
 or 	Show or Hide frame border handles
Frames	
<input type="text" value="Tag..."/>	Set the frame tag for the selected frame. While naming a frame can be used for documentation purposes, it is a requirement if using “Drilldown to Chart” along with “Target Dashboard Frame”.
	Copy the contents of the selected frame to the clipboard. (Ctrl-C)
	Paste the copied frame contents into the selected frame. (Ctrl-V)
 or 	Mobile Layout Only Configure the order in which frames are navigated in Phone Mode
	Tile Effects Configure the tile effects for the selected frame. Shift-Click removes the tile effects from the selected frame.
Background	
<input type="color" value="#FFFFFF"/> Color	Set the background color for this dashboard
 or 	Fit or Don't-fit the background image to the outer dashboard frame
	Set a background image for this dashboard



 or 	Show or hide the chart menu buttons
Controls	
 or 	Enable or Disable synchronized highlighting in this dashboard (Effective only in the View Interface.)
 or 	Enable or Disable pivot synchronization in this dashboard (Effective only in the View Interface.)
Scaling	
 or 	Enable or Disable dashboard scaling (See section below for more details)
 or 	Enable or Disable aspect ratio locking (when scaling is enabled)
<input data-bbox="266 697 354 751" type="text" value="1228."/> Width 100-5K	Dashboard Width (100-5000 pixels) (when aspect ratio is locked)
<input data-bbox="266 764 354 819" type="text" value="451.5"/> Height 100-5K	Dashboard Height (100-5000 pixels) (when aspect ratio is locked)

CHART	
	Create a new chart in the selected frame
	Place existing charts into frames. Once selected, the open dialog will allow for continuous drag-and-drop of multiple charts into multiple frames. Alternatively, the dialog will open a single chart into a single frame.
	Edit the chart in the selected frame
Panels	
	Place a panel into the selected frame. This includes: Image, Text, HTML, Dashboard Parameters, and the Dashboard Launcher
	Clear the contents of the selected frame

TOOLS	
Dashboard	
	Configure the report schedule on the dashboard
	Configure the report appearance and content on the dashboard
Chart	
	Schedule reports on the chart in the selected frame
	Configure reports on the chart in the selected frame
	Configure alerts on the chart in the selected frame

6.3.7 Dashboard Scaling

The chart graphics within iDashboards use a vector technology which scales virtually perfect across any size screen. These vector-based charts and graphs are pixel perfect with near-infinite zooming without distortion. The text-based components within iDashboards are also vector but have been disabled from scaling with different screen resolutions. The reason for not scaling text is to ensure a text height of 0.25" (1/4 inch) will stay at 0.25" if the User Application window is resized to a smaller dimension; keeping all text readable to the human eye. Lastly, dashboards scale to fill the extents of the entire User Application window. This is the default behavior of a dashboard.

It's become a common practice to view the User Application in full-screen mode to ensure all of the dashboard charts and graphs can be easily interpreted. When resizing the User Application, scaling will independently stretch both the horizontal and vertical directions. Generally, this behavior works very well when a dashboard is designed on Computer "A" and viewed on Computer "B" even if the two computers do not share the same screen size or screen resolution.

However, extreme scaling may not display the dashboard in the most favorable format. Extreme scaling occurs when Computer "A" meticulously designs a dashboard but then Screen "B" views the dashboard. Because of the extreme resolution differences between the two devices, the dashboard no longer has the desired scaling. On a per-dashboard basis, this can be resolved by enabling Dashboard Scaling.

Note: When using a mobile interface, regardless if there is a mobile layout, dashboard scaling will not be applied when using Vertical/Horizontal scroll mode. Dashboard scaling will be applied when using No Scrolling mode. If you change the setting to No Scroll, you'll see the scaling in effect in the live preview, because frame scrolling is no longer in effect.

6.3.7.1 Dashboard Scaling Tips and Tricks

- The elements of a dashboard are made up of various item types; Some can scale infinitely (vector) and some cannot scale infinitely (raster). Make sure to only scale up/down a dashboard within the visual limits of whatever screen sizes will view the dashboard.
 - If the scaling width/height is set to 600x400 and the dashboard is viewed on a screen that is 1920x1080, expect to see FUZZY text or FUZZY chart graphic details. In this scenario, the low-resolution dashboard is being stretched upward into a size that could pixilate some of the text or chart graphics.
- If "Extreme Scaling" is needed on a dashboard (maybe for Wall Display purposes), consider saving a copy of the original dashboard prior to adding the large-scale factor.

6.3.7.2 Text Scaling

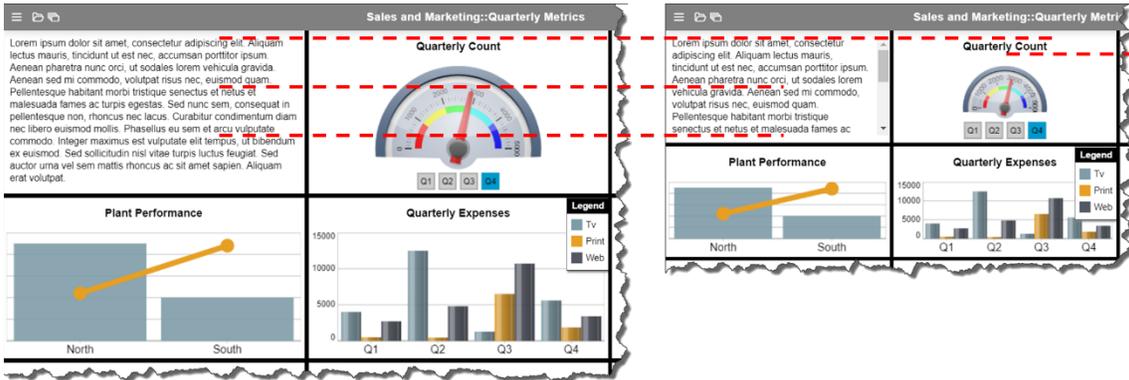
Under extreme scaling the majority of aesthetic issues occur with the 'Text' based elements. It is not helpful to keep 0.25" text heights when displaying a dashboard on a 60" high-definition monitor. To allow the dashboard to scale all text, click on the icon to enable Dashboard Scaling. The current User Application window dimensions will be used when this feature is enabled. This



means the text will scale in the future based on the current dimensions of the User Application window.

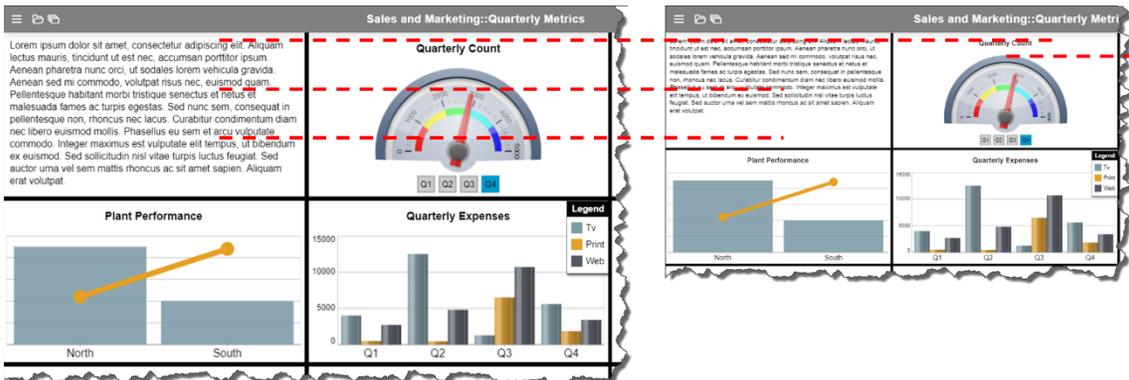
Example (no text scaling):

Below are two screenshots of the same dashboard. The red lines have been added for illustration purpose. On the left, the screen resolution of the User Application was 1200x800. On the right is the same screenshot after the User Application resolution was decreased to 900x600. Notice the graphics scaled down in size and the text did not scale down in size. The text remained the same height which can make the dashboard easier to read when viewed on smaller screens. However, the text panel now requires scrollbars because there is less space to display the same size text.



Example (with text scaling):

Below are two screenshots of the same dashboard. The red lines have been added for illustration purpose. On the left, the screen resolution of the User Application was 1200x800. On the right is the same screenshot after the User Application resolution was decreased to 900x600. Notice the graphics and the text scaled down in size. Here, the text did not remain the same height which made this example more difficult to read. However, the text panel no longer requires scrollbars. This was achieved by enabling Dashboard Scaling.



6.3.7.3 Aspect Ratio Locking

Once Dashboard Scaling has been enabled, an option exists to also lock the aspect ratio of the dashboard. Locking the aspect ratio will force the dashboard to always maintain a scaled ratio based on the Layout Height and Width provided within the settings. This option is helpful when



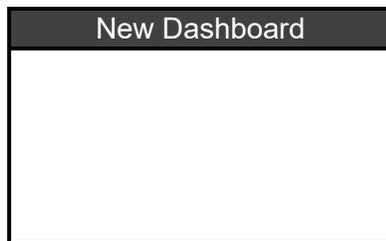
the layout of a dashboard is graphically sensitive and will not provide good results if scaled without using a certain aspect ratio. For example, a background images of circular shapes would appear best if the aspect ratio was locked to ensure the shapes always appear as circles and not ovals.

6.4 Frames and Frame Borders

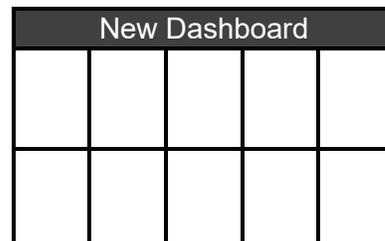
Initially, a new dashboard begins as a large, single frame. Splitting this frame (horizontally or vertically) will:

- Divide a single frame into two frames.
- Reveal a new frame border.

Frames are used to place interactive charts, images, text, and more. The more frames that appear on the dashboards, the more items that can be placed. However, the more frames that appear on a dashboard also mean the frames will be smaller.



Dashboard with 1x Frame



Dashboard with 10x Frames

6.4.1 Frames

Select a frame and you will see it highlight with a red-colored border. Options now include:

- **Create** a new chart in the selected frame
- **Open** a chart within the selected frame
- **Place** a panel into the selected frame. This includes: Image, Text, HTML, Dashboard Parameters, and the Dashboard Launcher
- **Clear** the contents of the selected frame
- **Split** the frame into two (horizontally or vertically)
- **Merge** the frame with the one above, below, to the left or to the right (conditional based on adjacent frame)
- **Swap** the contents of one frame with another using drag and drop
- **Copy** the contents of the selected frame to the clipboard. (Ctrl-C)
- **Paste** the copied frame contents into the selected frame. (Ctrl-V)
- **Tile Effects** configure the tile effects for the selected frame. (Shift-Click removes effects)

6.4.2 Frame Borders

Frame borders can be repositioned on the screen using the mouse to click-and-drag. Frames can be divided (increasing the frame count) or merged (decreasing the frame count), widened or narrowed, locked or unlocked, or visually changed.

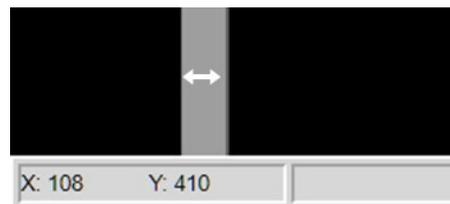
Using the example settings in the table below, the frame borders were customized, and a dashboard background color was added to create the screenshot, also shown below.

Icon	Setting	Value
 Color	Dashboard background color	000000 (Black)
 Color	Frame border color	FFFFFF (White)
 Handles	Frame border handles	Off
<input type="text" value="25"/> Width 0-50	Frame border thickness	25
<input type="text" value="50"/> Transp. 0-100	Frame border transparency	50

6.4.3 Frame Border: Positioning

Frame borders can be repositioned using click-and-drag. While dragging the frame the X and Y coordinates of the cursor are shown near the status bar (see image below). Coordinates can help a designer perfectly align one frame with another, or keep frames consistent across multiple dashboards.

Note: Frame borders can be repositioned under all but one circumstance; if the frame borders are hidden.



If a frame border is going to remain visible, there is also an option to lock the position of all frame borders. This will prevent a viewer from temporarily altering the desired appearance of the dashboard layout.

6.4.3.1 Frame Border: Precision Positioning

With a high-res monitor, it can be difficult to position a frame border at a given pixel location. Precision positioning provides a method of applying a fine-control when adjusting a divider.

Options can be:

- **Mouse-Over Border:** The border's X or Y location (within the dashboard's XY coordinate system) is displayed on the Toolbar's Status Bar.
- **SHIFT+Drag Border:** Enables precision-dragging by decelerating the mouse movement. This results in the border moving 1 pixel for every 10 pixels of mouse movement.



When precision-dragging the frame's "ghost" border will move while you're dragging, and does not stay pinned near the mouse. This is because the mouse is moving at its normal speed in that direction. The best way to use this feature is to drag the border to approximately where you want it, and while you're dragging use SHIFT+Drag, and let up the mouse button before releasing the SHIFT key, when the border is in place.

If you release the SHIFT key while precision-dragging, the border will snap the location of the mouse cursor, if it's within the frame borders available range of motion.

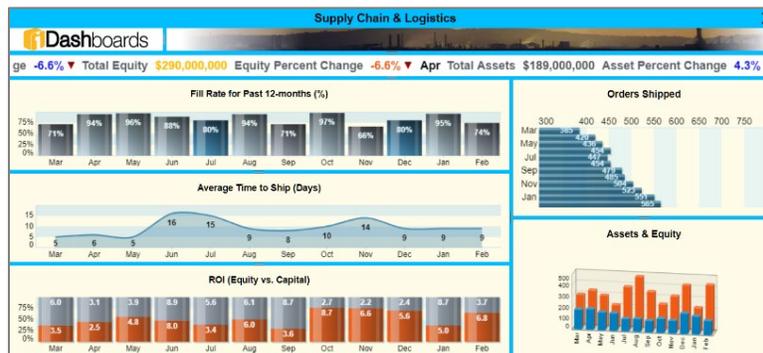
In Summary:

- Release SHIFT First: Border snaps to mouse cursor
- Release Mouse Click First: Border created at "ghost" location

6.4.4 Frame Borders: Show or Hide

Frame borders are the dividers between each frame on a dashboard. Examples of the same dashboard with borders on and off are displayed in the screenshots below. For aesthetic purposes, the borders can be hidden.

Note: The default setting for a new dashboard is to have the frame borders turned ON.



6.4.5 Frame Tile Effects

Configure tile effects for the selected frame. Shift-Click removes the tile effects from the selected frame. Tile effects can be copied (Shift-Ctrl-C) from one frame and pasted (Shift-Ctrl-V) into another frame.

Each time a tile effect is saved, its properties are stored. The next time tile effects are first applied to a frame, those stored effects will be automatically applied, without any of the borders and margins clipped. When splitting a frame with tile effects, those stored effects will also be automatically applied, with borders and margins dynamically clipped. While configuring a tile effect, there are also clear, copy and paste buttons. All of this is to facilitate rapidly configuring the same tile effects on multiple tiles.

Note: Properties are stored in the browser's local storage and will persist as long as that storage is retained.

Starting from the outside to inside, the Tile Effects are made of 3 sections, with controllable colors, widths and sizes.

- **Margin** The outer edge around the tile. You can define the dimensions, color and transparency.
- **Tile** The inside of the tile with ability to round the corners, along with 8 border styles, and option to add shadow effects.
- **Content** The area where a chart or panel would be placed, with option to add shadow effects.

Each side of the effects' preview, uses a "clip" checkbox to remove the border and margin from that side of the tile. All the content will remain, along with shadow effects. The border and margin can be easily restored by unselecting the checkbox.

6.5 Dashboard Backgrounds

Customizing the dashboard background can greatly enhance the overall appearance of a dashboard. The background of a dashboard can either be a single, customized color or an image.

Note: The default background for new dashboards is a white colored background.

6.5.1 Dashboard Background Image

A background image can be applied to the entire dashboard. This image will be visible in any empty dashboard frame as well as in any chart with transparency. An example can be seen below.

Note: Images can be uploaded to iDashboards through the Build Interface or the Administrator Application. Images may also be referenced by URL, but this setting requires the administrator to enable the Proxy Server or Cross-Domain policy. See the Administrator's Manual for details on adjusting these settings.





From the Dashboard Designer, click on the  button to set the dashboard background image. An image selection window will appear, allowing an image to be selected from the iDashboards server.

Note: If the proxy server settings have been enabled by the Administrator, a checkbox will appear within this window. This feature allows for linking out to an internet-hosted image file and won't require uploading the file into iDashboards.

By default, the image will be stretched to fit the image to the outer dashboard frame. If this is not the desired behavior, click on the  button to toggle the stretch behavior.

6.5.2 Dashboard Background Color

A background color can be applied to the entire dashboard. This color will be visible in any empty dashboard frames as well as in any chart with transparency. From the Dashboard

Designer, click on the icon  to set the dashboard background color. This color will be the back-most layer when viewing the dashboard and considering the variety of transparency options.

6.6 Show Chart-Menu Buttons

Chart menu buttons can display additional actions to perform on a chart. For aesthetic purposes, the default setting hides these buttons. To toggle this setting, click on the  button to reveal these buttons for all charts on the dashboard.

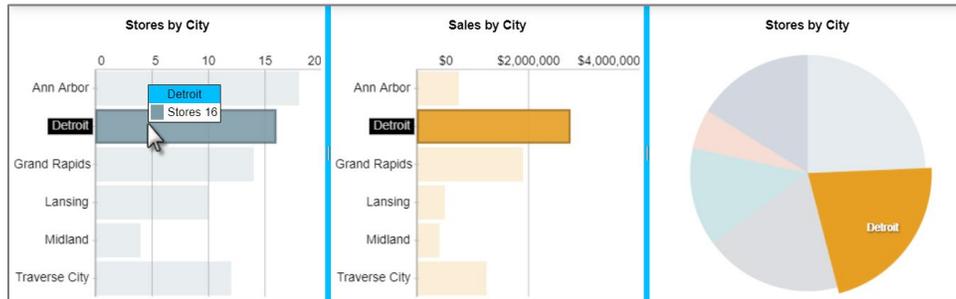
Note: Chart Properties can individually control the menu visibility settings. The dashboard setting discussed here cannot override the chart property.

6.7 Synchronized Highlighting

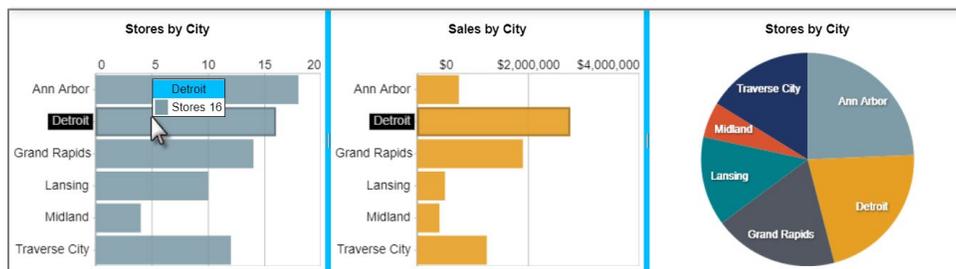
When mousing over various data points within a chart, other data points in that chart will fade out. Also, through “Interactive Intelligence” related data points in other charts on the dashboard are simultaneously highlighted and other data points in those charts are faded out. This is a powerful feature of using iDashboards.

See Section 8.5.2.16, “Match Type” for more information.

To toggle synchronization on and off, click on the  button. Note the behavior of the cursor with the same dashboard set to ‘Synchronize highlighting’ enabled and disabled (see below).



“Enabled”



“Disabled”

6.8 Synchronize Pivots

Synchronize pivots forces the dashboard to automatically synchronize related pivots in all charts that exist in that dashboard. For synchronization to work, the pivots must have the same Y-axis label name. This does not require identical data in the Y-axis, but the pivots will not synchronize on a data point if both pivots don’t have matching data.

From the Dashboard Designer, click on the  button to toggle the dashboards pivot synchronization.

Example: If two charts have a pivot named “State” and Synchronize Pivots is enabled, they will be synchronized. If a user changes the pivot value in one of the charts, the pivot value in the other chart will automatically change to the same value and both charts will be refreshed to show data relating to that pivot value. If the Synchronize Pivots is disabled, they will not be synchronized. The pivot in the other chart will not automatically change and that chart will not be refreshed.

6.9 Mobile Interface

The iDashboards Mobile interface offers users a browser-based experience to view dashboards specifically from a small device. The mobile interface is a 'view-only' interface, regardless of your user role within iDashboards. Only one dashboard can be opened at a time. While charts and dashboards cannot be created or edited, dashboards with charts can be viewed and will be fully interactive.

6.9.1 Design Layout

There are multiple tools to assist in designing dashboards that will be viewed through a mobile interface.

- The designer can switch from the 'Desktop Layout' to 'Mobile Layout'. This will create a clone of the existing dashboard, which can then be configured separately. When the dashboard is loaded from the mobile application, you'll get the mobile layout. The mobile application will show a thumbnail based on the mobile layout, in its selected display mode. That thumbnail is always a 1/2 aspect ratio.
- 'Toggle Mobile View' will allow building and viewing the dashboard's desktop layout through a mobile interface. The  icon will toggle between portrait and landscape modes.
- Once a mobile layout is created, the option to remove it is available through selecting 'Delete Mobile Layout'.

6.9.2 Frame Ordering (Mobile Device > Display Mode)

There are three display modes within the mobile interface. The purpose of this option allows a user to choose the best experience when transitioning from a large-format dashboard onto a small-format mobile device.

Setting one of these modes, known as "Mobile Display Mode", is possible within the Dashboard Properties – otherwise the mobile device's default scrolling mode will be used.

Note: "Mobile Display Mode" refers to when you can select scrolling dashboard frames vertically or horizontally, one at a time.

- **No Scrolling** – This setting renders the entire dashboard inside of the mobile browser. While this view offers the purest experience of the dashboard layout, as it was designed on a desktop, it may not offer the best experience if the charts and graphics are too small to interpret.
- **Vertical Scroll** – This setting displays one dashboard frame at a time and transitions to the next (or previous) frame using an up-down vertical scrolling direction.
- **Horizontal Scroll** – This setting displays one dashboard frame at a time and transitions to the next (or previous) frame using a horizontal scrolling direction.

When using the Dashboard Builder, frame scrolling will work in the dashboard designer's Live Preview mode, but since the single mobile page only has a single frame, the frame's navigation arrows do not appear. Regardless if there is a mobile layout, the Live Preview



mode does not offer the option to experience frame scrolling (Vertical/Horizontal scroll mode).

Note: When using a mobile interface, regardless if there is a mobile layout, dashboard scaling will not be applied when using Vertical/Horizontal scroll mode. Dashboard scaling will be applied when using No Scrolling mode. If you change the setting to No Scroll, you'll see the scaling in effect in the live preview, because frame scrolling is no longer in effect.

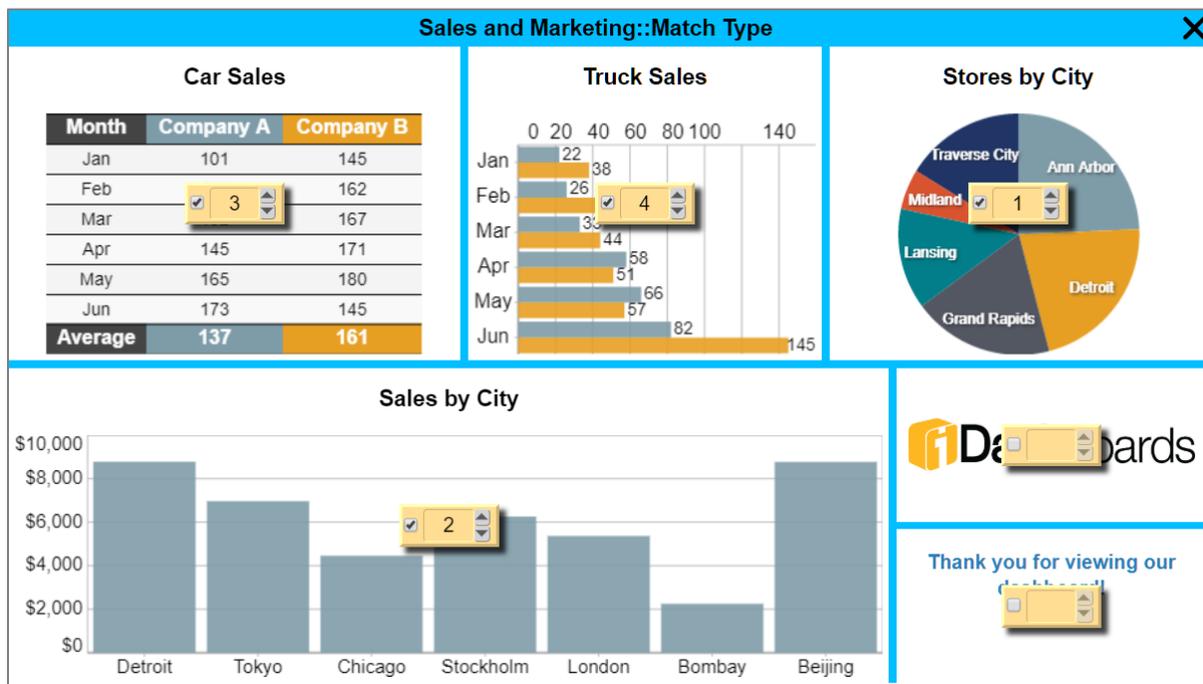


By default, the 'Frame Ordering' button is disabled. Once enabled, this setting will display a number-box within the center of each frame on the dashboard:

- The number represents the order in which the frame will appear, when using "Vertical Scroll" or "Horizontal Scroll" mode on a phone.
- The checkbox enables or disables the frame from appearing altogether, when using "Vertical Scroll" or "Horizontal Scroll" mode on a phone.

In the example below, if this dashboard were viewed on a phone, using "Vertical Scroll" or "Horizontal Scroll" mode:

- The pie chart would appear first. Followed by the blue bar chart, then the multi-colored bar chart, then the tabular chart.
- The logo and text panels won't appear at all, as the checkboxes have been removed.



Note: The number-box(s) will never appear in the View Interface. They are only shown when designing the dashboard.



6.10 Move Dashboard (via Changing the Dashboard Category)

A user can move an existing dashboard from the current category into another category. The user must have 'Save' permissions on the source and destination categories.

To change the category for a dashboard:

- Open the dashboard.
- Click on the Dashboard Properties button .
- Change the Category using the drop-down list and click "OK".
- Lastly, the dashboard must be saved to complete this task.

Note: Moving a dashboard will not change the Dashboard ID. However, performing a 'Save Dashboard As...' on the dashboard creates a new dashboard with a new Dashboard ID.

6.11 Rename a Dashboard

Dashboards that have already been saved can be renamed. The user must have 'Save' permissions on the category where the dashboard resides.

To rename a dashboard:

- Open the dashboard.
- Click on the Dashboard Properties button .
- Change the "Dashboard Name" and click "OK".
- Lastly, the dashboard must be saved to complete this task.

Note: The Dashboard ID will not change. If you perform a 'Save Dashboard As...' on the dashboard then a new dashboard will be created with a new Dashboard ID.

6.12 Hide the Category Name in the Dashboard Title

By default, a dashboard title includes the Category name followed by the Dashboard name.

<Category Name> :: <Dashboard Name>

ex: Sales Dashboards :: Monthly Sales

The Category name (Sales Dashboards) can be hidden from view through the Dashboard Title control.

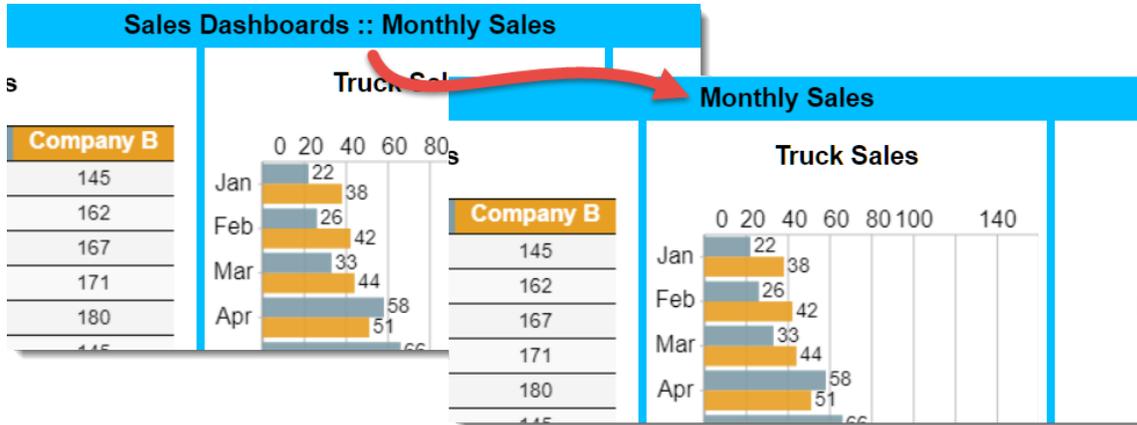


Show Dashboard Category In Title

Dashboard Title

Cancel OK

Note: Changes to showing the Dashboard Category will not display until the dashboard is saved.



6.13 Dashboard Thumbnails

The thumbnails are primarily used within the View Interface to help locate a dashboard by appearance. If the administrative thumbnail policy is enabled and configured, dashboard thumbnails are created each time the dashboard is saved. If thumbnails are missing from your environment and everything is properly enabled and configured, then it is likely the dashboard needs to simply be Opened and Saved within the Build Interface.



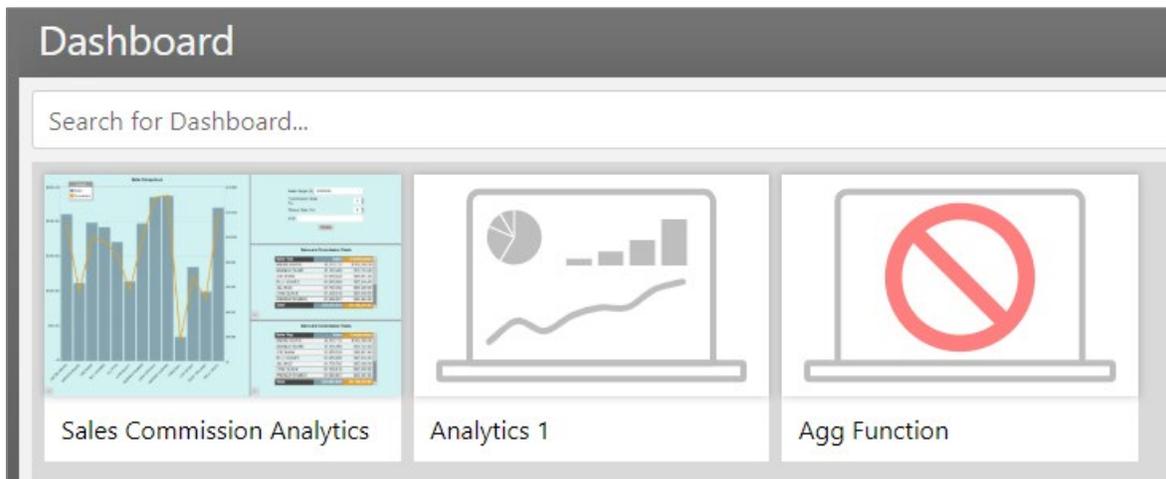
6.13.1 Thumbnail Policy

The iDashboards administrator has global control whether thumbnails are permitted. However, if they are permitted it is still possible to not see thumbnails on every dashboard tile by means of restricting the thumbnail on a per-dashboard basis:

- Open the dashboard.
- Click on the Dashboard Properties button  .
 - **System Default:** A dashboard thumbnail will adhere to the administrative value of either:
 - Automatic, Prompt or Never
 - **Automatic:** A dashboard thumbnail will automatically be saved each time the dashboard is saved.
 - **Prompt:** The user will be prompted when they save a dashboard to indicate if they would like to save a dashboard thumbnail.
 - **Never:** The dashboard thumbnail will never be saved. These dashboards will be represented by a generic thumbnail.

The three types of dashboard thumbnails are seen in the example shown below:

- An actual thumbnail of the dashboard
- A dashboard still requiring a thumbnail (Analytics 1)
- A dashboard with thumbnails disabled (Agg Function)



6.14 Hide Dashboard

Within the “Builder” interface, access to a category (view or save) will show all of the dashboards. However, within the “Viewer” interface, dashboards within a category may be hidden from view.

Hiding a dashboard within a category is helpful if the goal is to prevent users from opening the dashboard natively. For example, this is especially beneficial when hiding a dashboard that should only appear during the drilldown process.

To hide a dashboard:

- Open the dashboard.
- Click on the Dashboard Properties button  .
- Enable the setting “Hide in Dashboard Lists” and click “OK”.
- Lastly, the dashboard must be saved to complete this task.

Note: Dashboards can also be hidden (or unhidden) by the administrator, using the Administrator Application.

6.15 Refresh an Open Dashboard

Refresh a dashboard to force each dynamic chart to re-query the data source. This will result in displaying the most current information since the dashboard was last opened. The manual task of updating a dashboard can only take place from the View Interface.

6.15.1 Auto-Refresh Dashboard

From the Build Interface, it is possible to configure an automatic update on a dashboard or on the individual chart(s) displayed on a dashboard. Once enabled, the refresh time interval option can be set anywhere between 1-second and 9999 minutes. When the setting is disabled, the dashboard will not automatically refresh.

To enable Auto-Refresh on a dashboard:

- Open the dashboard.
- Click on the Dashboard Properties button .
- Enable and set the property “Refresh Interval” to any value ranging from 1-second to 9999 minutes and then click “OK”.
- Lastly, the dashboard must be saved to complete this task.

Note: The preferred way to set up auto-refresh is at the chart level. Refer to the Chart Properties for more info on chart level auto-refreshing.

Note: The default setting is to have the ‘Auto-refresh’ option disabled within new dashboards.

Note: Take caution when using a small period of time as the refresh interval. If the dashboard query takes 5-seconds and the refresh interval is 2-seconds, it is possible the dashboard will never fully render on the screen.

6.16 Saving the Dashboard (or Save-As)

The dashboard is edited and created locally within the application, until the user chooses to save the dashboard to the server.

Note: Dashboards and charts are saved independent of one another. Therefore, if all the charts are saved but the dashboard is not, the dashboard may not open the charts the next time it is opened.

Saving a new dashboard has the following minimum requirements:

- A selected Category (in which you have 'save' permissions).
- A dashboard "Name".
 - In the event the dashboard name is not unique within the category, you will be asked to provide a new name or overwrite the existing dashboard.

To Save a dashboard:

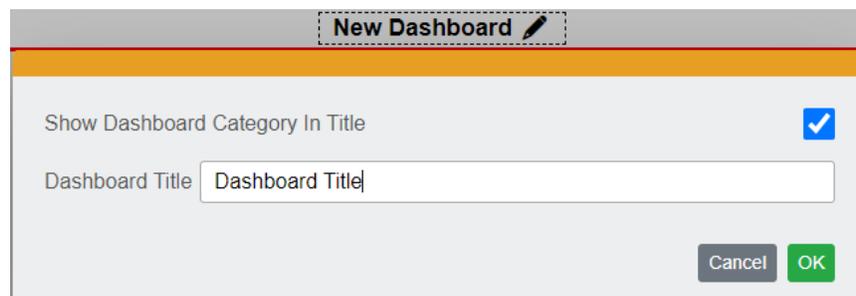
- Start a new dashboard or open an existing dashboard.
- Click on the  button and select either "Save Dashboard" or "Save Dashboard As...".
- In the save dialog window, select a category then enter a dashboard "Name" (Title and Description are optional).
- Click "Save".

Note: "Save As..." is an option used to create a copy, or clone, of the current dashboard. This operation will not copy the linked charts or picklists, as those will require their own "Save As..." operation.

6.16.1 Dashboard Title

The purpose of a dashboard title is to allow the dashboard to be saved using a 'technical' name while allowing the display name of the dashboard to remain user friendly. Each dashboard page will show the same title, but it can be change on any of the pages. The title is also displayed at the time a dashboard is being opened.

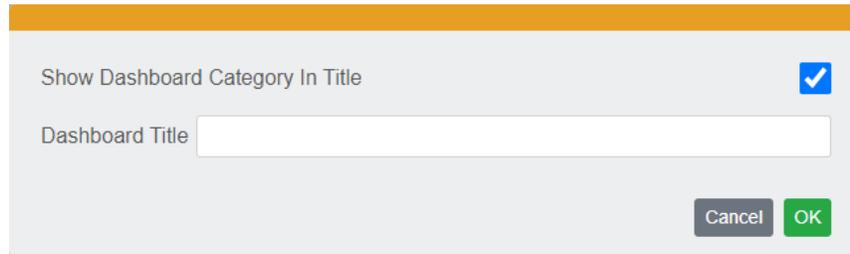
A new dashboard will start with the 'New Dashboard' title. Select the pencil icon to edit the dashboard title. Controlling whether the Dashboard Category prefixes the Dashboard Name is accomplished through the checkbox.



Note: Changes to showing the Dashboard Category will not display until the dashboard is saved.

Category Name::Dashboard Title 

The dashboard title is optional when saving a new dashboard. If left blank, the dashboard name will be displayed in the title bar of the dashboard.



Category Name::Dashboard Name 

To change the title after the dashboard has been saved, click on the pencil icon.

Note: The title field is not searchable.

6.16.2 Dashboard Description

The purpose of a dashboard description is to allow the dashboard to contain notes which are displayed at the time a dashboard is being opened. The dashboard description is optional when saving a new dashboard. To change the description after the dashboard has been saved, click

on the Dashboard Properties button  .

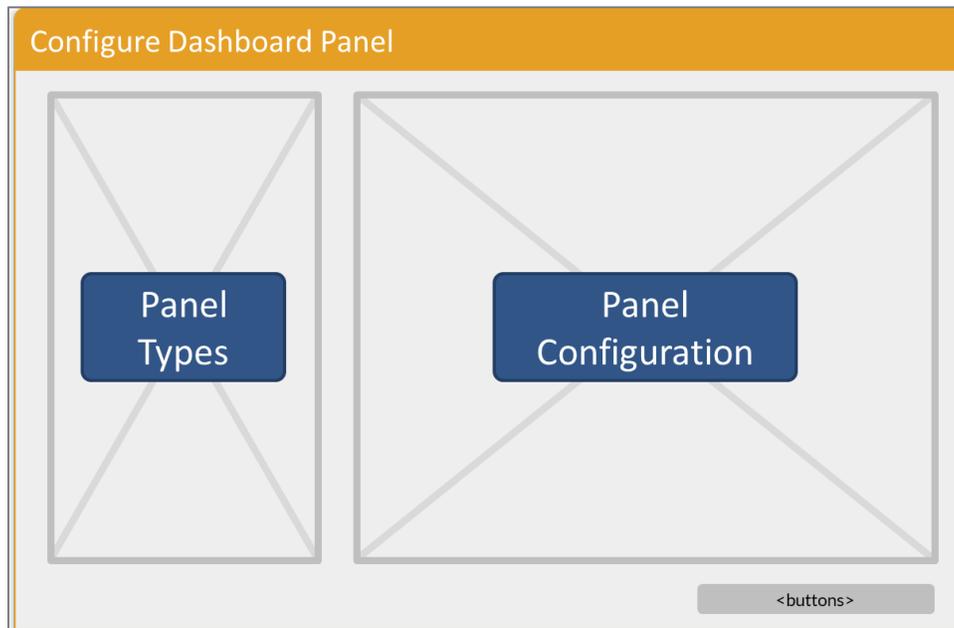
Note: The description field is not searchable.

6.17 Dashboard Panels

A Panel is a special kind of dashboard object that is not a chart. There are a number of options to choose from when creating a panel as they can be comprised of an image, plain text, HTML, and more. Panels are not saved individually but are saved as a component of the Dashboard. Therefore, they cannot be opened in other frames and will need to be recreated if they are needed within another dashboard.

To place a panel into a frame, select the desired frame and then click the panel  button.

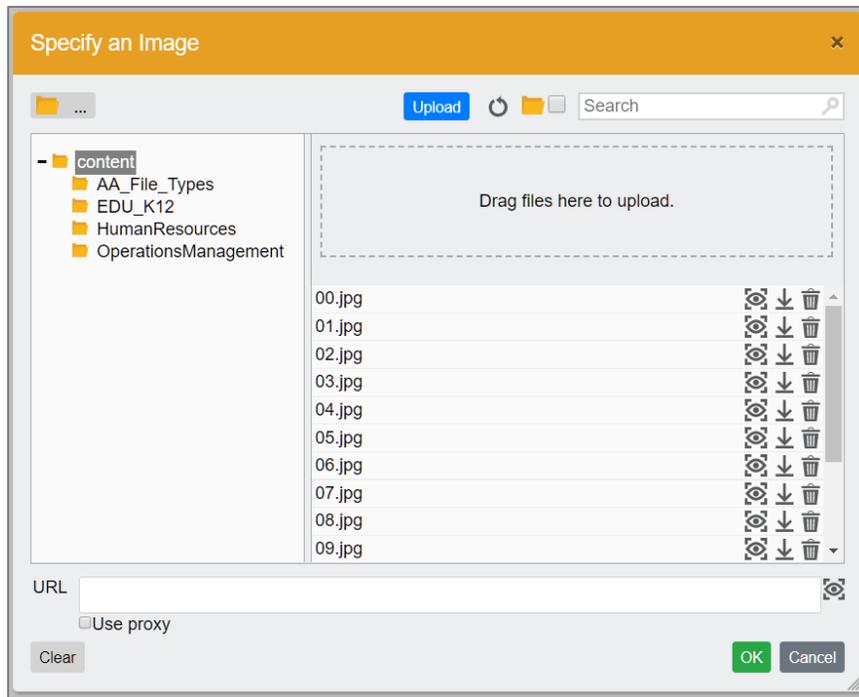
To clear a panel from a frame, select the desired frame and then click the clear  button.



6.17.1 Image Panel

Dashboard frames can contain static images for aesthetic purposes. This feature allows dashboard builders the ability to add logos, graphics, and other external graphics to a dashboard. Click the “Set...” button and a new window will appear. When specifying an image, the user can select an existing image or upload a new image. Images may also be referenced by URL, but this setting may also require the administrator to enable the Proxy Server.

Note: Images referenced by URL may also require the administrator to enable the Proxy Server. If the proxy server settings have been enabled by the administrator, a checkbox will appear within this window. This feature allows for linking out to an internet-hosted image file and won't require uploading the file into iDashboards.



6.17.1.1 Supported Image Types

- PNG
- JPG (JPEG, PJPEG)
- GIF
- SVG (SVGZ)
- MP4 (M4V) (specifically for use when using the HTML panel)
- XML (specifically for use with the Image Plot Chart and GeoPlot Chart)

6.17.1.2 Image Properties

Aside from the background color, background transparency, and image URL, the Image Panel comes with additional unique properties.

- **Tool Tip Text** – This is the text that will appear when the cursor hovers over the image.
- **Link URL** – This is the URL that will open a new browser tab when the image is clicked.
- **Link Target Window** – This is the name of the new browser tab when using the “Link URL”. Naming the window will cause all hyperlinking to reuse the same Target Window instead of creating new tabs.
- **To fit image in frame** – The default option is ‘Stretch’, which will stretch the image inside of the frame. This option will resize the image/move as necessary to fit a smaller frame and will make the image as large as possible in larger frames, sometimes distorting the image.
- **Maintain Aspect Ratio** – This option is only available when the ‘Stretch’ option is selected. This setting will maintain the height-to-width ratio of the image no matter the size of the frame. You also have the option to choose the background color of the frame.



- **Window Settings** – These are any standard HTML modifiers that can be assigned to a new browser window. Each modifier is comma separated, and take the form as shown below. See the table below for a list of example modifiers.

`<setting name>=<setting value>`

For example:

`width=1000,height=600,resizable=yes,left=0,top=0,status=yes,toolbar=no,menubar=no`

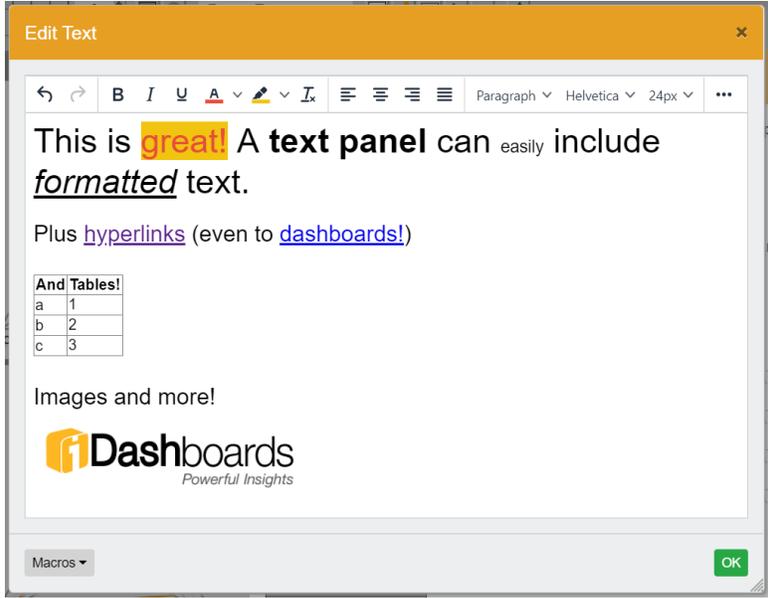
Modifier	Value	Browser Specific	Description
width	number in pixels		Specifies the width of the new window.
height	number in pixels		Specifies the height of the new window.
resizable	yes or no	Possibly	Specifies whether or not the new window will be resizable.
scrollbars	yes or no	Possibly	Specifies whether or not the new window will have scrollbars
toolbar	yes or no	Possibly	Specifies whether or not the new window should show the browser navigation bar at the top (The back, forward, stop buttons etc.)
location	yes or no	Possibly	Specifies whether or not the new window will show the location box with the current URL (The place to type <code>http://address</code>).
directories	yes or no		Specifies whether or not the new window should show the extra buttons (personal buttons, etc...).
status	yes or no		Specifies whether or not the new window will show the window status bar at the bottom.
menubar	yes or no		Specifies whether or not the new window will show the menus at the top (File, Edit, etc...).
copyhistory	yes or no		Specifies whether or not the new window will copy the old browser window's history list.
left	number in pixels		Specifies the position of the new window from the left of the screen.
top	number in pixels		Specifies the position of the new window from the top of the screen.

6.17.2 Text Panel

A Text Panel can be used to display one or more lines of text within a frame. Click the “Set...” button and a text editor will appear. A number of options can control the appearance of the Text Panel such as: background color/transparency, text size/color/alignment, scrollbars, and more.

Note: To see additional options click the “...” button to expand the toolbar.





6.17.3 HTML Panel

An HTML Panel can display formatted text, graphics or more, by using a subset of the HTML markup language. Click the “Edit” button and a new window will appear, containing a raw-HTML editor. Using this method will require knowledge of writing in the HTML language. A number of options can control the appearance of this panel such as: background color/transparency, scrollbars, margins, and more.

6.17.3.1 HTML Security

When user-created HTML is inserted directly into the HTML panel within iDashboards, steps are taken to insure that:

- The HTML cannot visually corrupt the iDashboards UI. For example, <style> tags could be used to alter existing CSS classes, therefore they are prohibited.
- JavaScript will not be executed, as that would create a serious security vulnerability.

Step #1: Remove prohibited elements				
applet	element	html	object	template
base	embed	iframe	script	textarea
body	form	input	select	title
button	frame	link	shadow	
canvas	frameset	main	slot	
content	head	meta	style	

Note: An “a” tag with an “href” attribute that begins with “javascript” is also considered a forbidden element, and removed.

Step #2: Remove prohibited attributes			
onabort	ondragexit	onloadeddata	onreset
onautocomplete	ondragleave	onloadedmetadata	onresize
onautocompleteerror	ondragover	onloadstart	onscroll
onblur	ondragstart	onmousedown	onseeked



oncancel	ondrop	onmouseenter	onseeking
oncanplay	ondurationchange	onmouseleave	onselect
oncanplaythrough	onemptied	onmousemove	onshow
onchange	onended	onmouseout	onsort
onclick	onerror	onmouseover	onstalled
onclose	onfocus	onmouseup	onsubmit
oncontextmenu	oninput	onmousewheel	onsuspend
oncuechange	oninvalid	onpause	ontimeupdate
ondblclick	onkeydown	onplay	ontoggle
ondrag	onkeypress	onplaying	onvolumechange
ondragend	onkeyup	onprogress	onwaiting
ondragenter	onload	onratechange	

Note: Any forbidden attributes will have their value set to an empty string.

6.17.3.2 HTML Macros

This panel can utilize macros to assist with advanced implementations:

`\${user}` - If necessary, it is possible to populate a portion of the dashboard frame with the username of the person logged in. The example below shows the HTML value and the rendered dashboard panel when finished:

Using this HTML	Results in this
<code><p>Welcome `\${user}`</p></code>	Welcome admin

`\${content}` - If an image is needed within the context of an HTML panel, it is possible to link to a file located on the iDashboards server or a file on the internet. The HTML code can be copied from the example below (requiring an adjustment the image path) The text highlighted in yellow was placed into the panel by clicking on the macro button.

Using this HTML	Results in this
<pre><textformat leading="2"><p align="CENTER">The <u>HTML panel</u> is able to show <i>formatted text</i>, </p></textformat><textformat leading="2"><p align="CENTER">plus images!</p></textformat><textformat leading="2"><p align="LEFT"> </p></textformat><textformat leading="2"><p align="LEFT"></p></textformat></pre>	<p><i>The HTML panel is able to show formatted text, plus images!</i></p> 

Note: Images may be stored within sub-folders on the iDashboards server. It may be necessary to designate the folder structure when using the content macro. Example: "\${content}/Maps/States/Michigan.png".

Note: The macro `#{uecontent}`, will un-escape the expanded value of the macro.

6.17.4 IFRAME Panel

Using an iFrame, this panel can display the contents of an external web page within an iDashboards dashboard frame. There are many restrictions beyond the control of iDashboards that may not allow a web page to render within the iFrame panel. In those situations, consider using the feature “Drill to webpage” to open the web page in a new browser tab.

Security Warning!

While the iFrame feature is contained in what is commonly known as a sandbox, iDashboards has no control over which URLs are configured to run within this panel type. Therefore, take caution when implementing this feature.

6.17.4.1 IFRAME Properties

Aside from the background color and background transparency, the IFRAME Panel comes with additional unique properties.

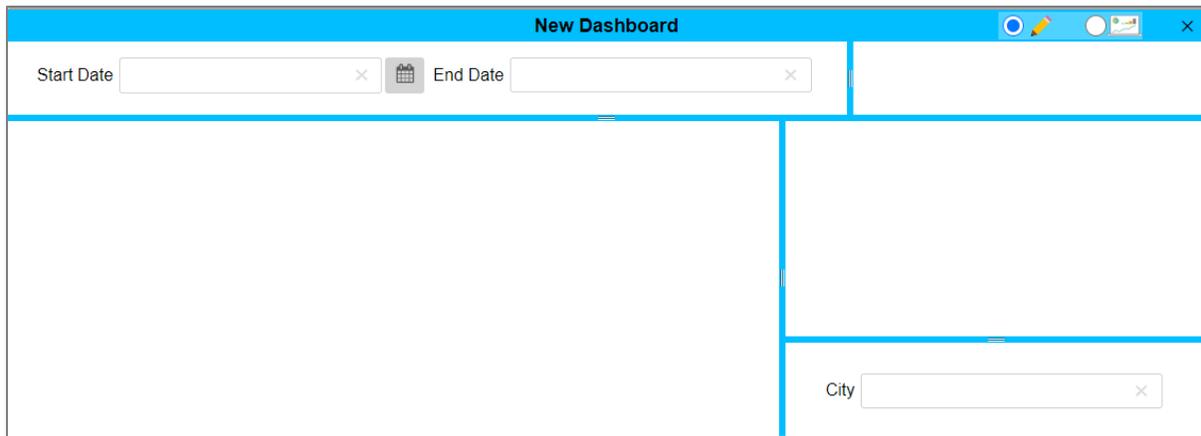
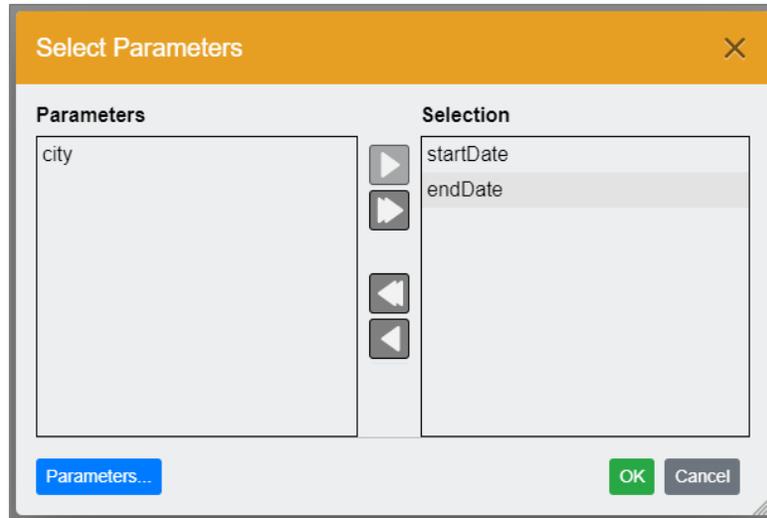
- **URL** – This is the URL to the web page that will appear within the iDashboards frame.
- **Referrer Policy** – These options reflect the HTML ‘referrerpolicy’ attribute of the <iframe> element which define how the referrer is sent when fetching the resource.
- **IFRAME Sandbox Settings** – At the tradeoff of allowing what might be an unknown behavior, these settings can allow/enable various elements to run within your iDashboards session. In most cases, if troubleshooting why an iFrame will not render, these settings can resolve many issues.
 - Allow Form Submission, Allow Scripts, Allow Popups, Allow SAME-ORGIN, Enable Pointer Lock API.
 - Allow Presentation will get rid of the error message and allow presenting content from the iframe.
- **IFRAME 'allow' Settings** – Allow Autoplay enables video content to autoplay in View and Preview mode, but it will be disabled in Build mode.

6.17.5 Parameter Panel

The Dashboards Parameters Panel is used to display and control pre-configured Dashboard Input Parameters. A number of options can control the visibility of the panel such as: background color/transparency, direction of parameter layout (vertical or horizontal), alignment, and more.

It is possible to customize the Selected Parameters when configuring this frame. In the example below, the configuration will allow the ‘city’ parameter to be placed into a different frame – or not displayed at all!





Note: In order for a Dashboard Parameter panel to display, the dashboard needs to have Dashboard Input Parameters defined.

6.17.6 Comments Panel

This panel is used to add and display the comments associated with the dashboard. A number of options can control the appearance of the Comments Panel such as: background color/transparency, header text size/color/alignment, post button color, and more. If used, the header can be configured to, when selected, open a separate Comments Dialog. A comment can contain up to 1000 characters, with formatting and emoticons.

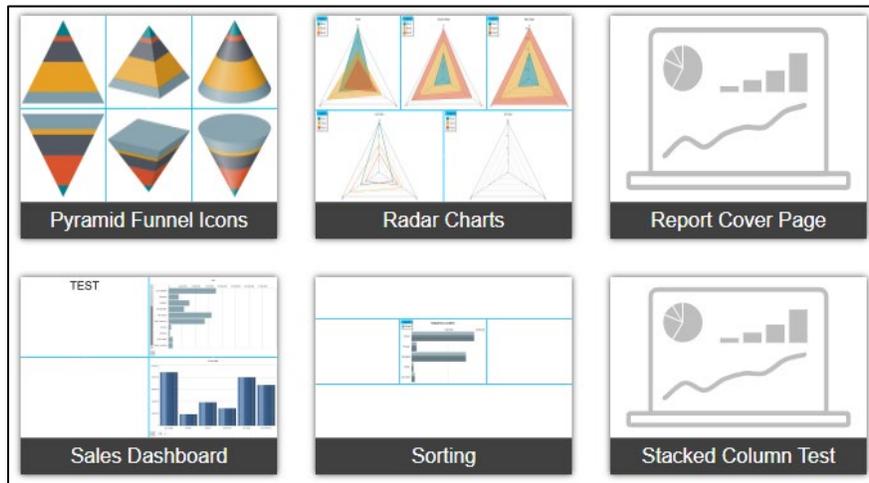
6.17.7 Dashboard Launcher

The Dashboard Launcher Panel displays a list of dashboards that can be used for easy navigation to each of the dashboards. Once configured, the Dashboard Launcher will function similarly to the drilldown option “Drilldown to Dashboard” where the target dashboard needs to be closed to return to the dashboard with the launcher panel. A number of options can control the appearance of the panel such as: background color/transparency, header text, alignment, and more.

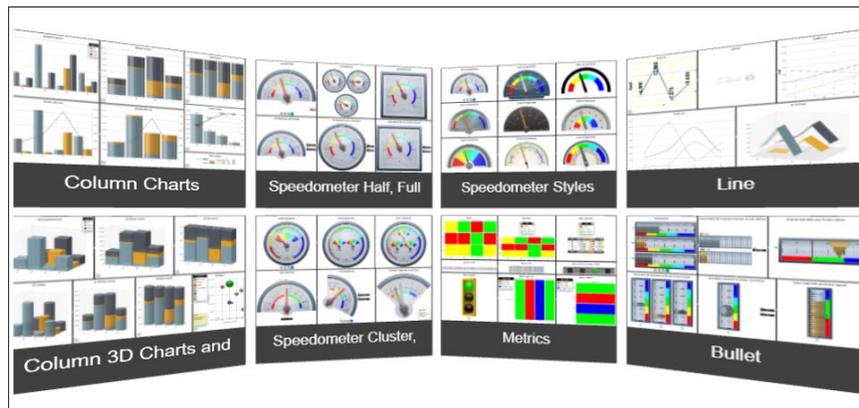
Text example:



Thumbnail example:



3D Thumbnail example:

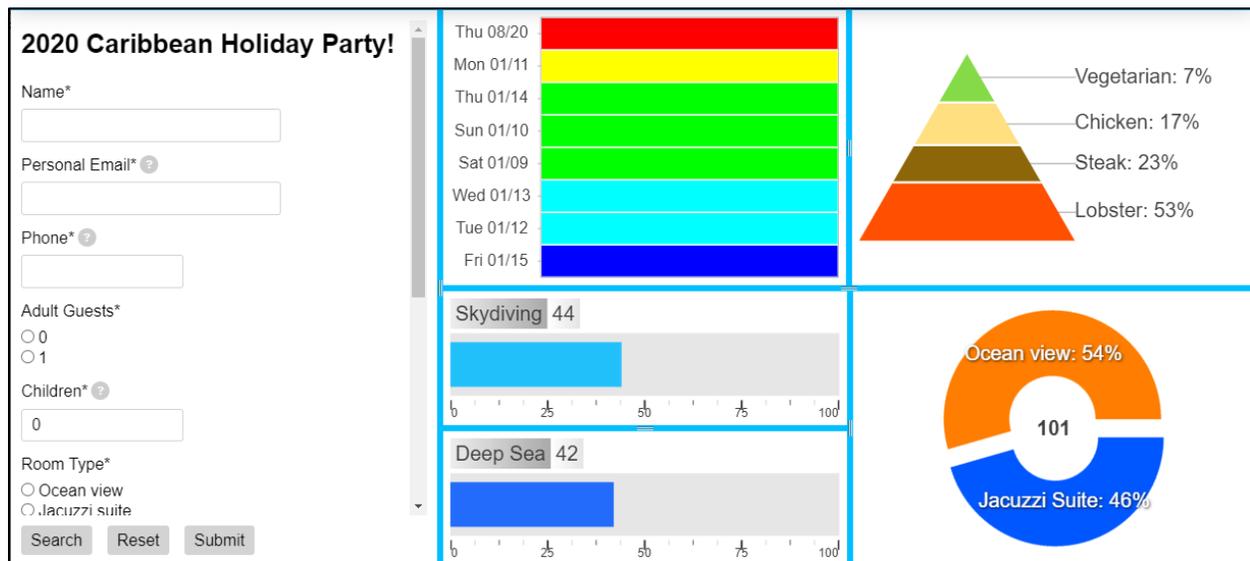


6.17.8 Form Panel

Note: The Forms feature must be enabled within the iDashboards license. The administrator has the option to disable the entire Forms feature.

Building forms is covered in a later section. Once a form is created and saved, it can be placed into a dashboard using the Form Panel option. A form can be placed into the same, or different dashboard, multiple times. When placing a form, there are visual settings that can change the appearance of the form, such as background color, form layout, and form footer button colors. Customizations made to the appearance of a form in this dialog are not saved with the form.

Below is an example of a dashboard with a form in the left panel, along with charts associated with the same database table.



6.17.8.1 Form Properties

Aside from the background color, background transparency, and margins, the Form Panel comes with the following unique properties:

- **Text Color:** Defines the color of the text that will be used on the form.
- **Form:** Use the 'Select form...' button to open the interface to select a form.
- **Handle Form Permission Error:** How to handle permission errors when displaying the form in the panel.
 - **Show Default Message:** 'You do not have access rights to the form:' with the form name on the next line.
 - **Show Blank Frame:** No message will show. The frame will be empty.
 - **Show Custom Message:** Use the message defined in the Custom Permission Error Message.



- **Custom Permission Error Message:** Define the message that will be used for an error message if the “Show Custom Message” is selected in the previous setting.
- **Text Size:** Defines the size of the text that will be used on the form.
- **Override Form Layout:** The form designer defines the forms layout, but it can be changed for it use in the form panel.
 - **Blank:** Use the Form’s defined layout.
 - **Flow Down:** Fields are laid out vertically.
 - **Flow Across:** Fields are laid out horizontally.
 - **Grid:** Fields are a grid-based layout.
- **Button Color:** Each of the form footer button colors can be customized here.

6.17.9 Article Panel

Note: The Knowledge Base feature must be enabled within the iDashboards license. The administrator has the option to disable the entire Knowledge Base feature.

This panel is used to display Knowledge Base articles. Click the “Select...” button and a new window will appear for selecting the Article. A number of options can control the appearance of the panel such as: background color/transparency and margins. The article’s title can be configured to be a link to the article in the Knowledge Base.



7. Chart Overview

Building charts can be fun! Like many technical projects, the more planning that occurs beforehand, the better your experience will be when it comes time to build each chart and each dashboard. Over time, these tasks will become familiar and easier to use based on repetition and experience. For training and tips, refer to Chapter 2, “OSKAR”.

7.1 Design Permissions

The ability to access the Lobby and then navigate to the Build Interface is allowed with a minimum user role of “Builder”. This user role will allow the minimum functionality of creating charts or dashboards to be saved within the “Personal” category. But the ability to save items into customized categories requires access rights provided by the iDashboards administrator. Such access rights are provided at the Category level, and all dashboards, charts, picklists and forms (if enabled) within any given Category inherit those access rights.

Categories have three levels of access rights:

- Save Access
- View Access
- No Access

Therefore, depending upon the access rights assigned to you, you may or may not have the rights to make changes and save those changes to an existing dashboard. However, you can always make changes to an existing chart and save those changes as a different chart within your Personal Category or any other Category to which you may have Save permissions.

Note: The remainder of this manual assumes a proper user role and access rights are configured to perform each operation described.

7.2 Create or Edit a Chart

Creating or editing charts can occur in two locations:

1. Directly from the Home screen
2. From within the Dashboard Designer

While the goal for creating and editing a chart may be quite different, the interface will be the same.

Create

1. To create a new chart, make sure the “Chart” tab is selected on the home screen, then select the “New” button as seen in the image below.
2. Alternatively, through the Dashboard Designer, click the “New Chart” icon.



Edit

1. To edit an existing chart, make sure the “Chart” tab is selected on the home screen. Then, using manual navigation or searching, locate the chart to open.
2. Alternatively, through the Dashboard Designer, click the “Edit Chart” icon (or double-click the chart)

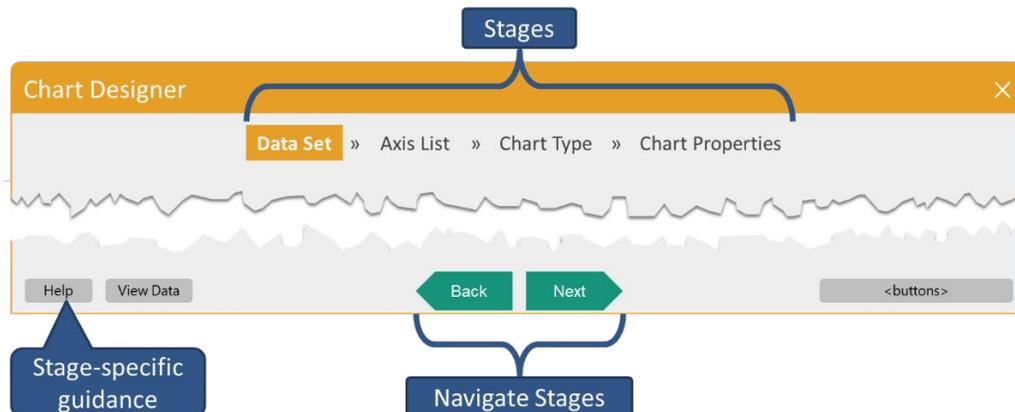


7.3 Chart Designer

The Chart Designer is a window used for creating and editing all aspects of a chart. The designer is comprised of various steps and stages, with minimum requirements, prior to the ability of saving and viewing a chart. Only one Chart Designer can be opened at a time.

7.3.1 Navigating the Chart Designer Stages

The Chart Designer progresses in a left-to-right direction as the chart is being built or edited. The stages appear at the top of the window, where the current stage will be highlighted and previous stages will appear as ‘visited’. To transition between the stages, click on the buttons along the bottom of the window. Refer to the following image for the interface essentials.



- **Data Set** – This stage includes defining where the chart data will come from, as well as what parameter and filters the chart will use.
- **Axis List** – This stage includes defining which data columns, from the data set, are included in the chart.
- **Chart Type** – This stage specifies which chart type will be used to present the data. First, select the chart group on the left, then the chart type on the right.
- **Chart Properties** – This stage is used to customize all properties and settings available for the chosen chart type. Additional settings include: number formatting, date formatting, drilldown configuration, and all color assignments.
- **Preview** - This stage offers a real-time preview of the chart based on the configurations of the previous stages.

Note: If the chart designer created or opened a chart directly from the Home screen, then the Preview stage will display the chart at the dimensions of the Chart Designer.

Note: If the chart designer created or opened a chart from within the Dashboard Designer, then the Preview stage will display the chart at the dimensions of the frame where the chart resides – at that time.

7.3.2 Chart Designer Help

Click on the “Help” button to receive in-product, stage-specific documentation. While this information may not provide all of the details of the product, it can be helpful to receive guidance to encourage the fundamentals of creating a chart.

7.4 Saving the Chart (or Save As)

The chart is edited and created locally within the application, until the user chooses to save the chart to the server. Saving a chart has requirements and validations which must be met before the “Save” button will become enabled. Progress to the later stages will enable the “Save” button.

Note: Dashboards and charts are saved independent of one another. Therefore, if all the charts are saved but the dashboard is not, the dashboard may not open the charts the next time it is opened.

Saving a new chart has the following minimum requirements:

- A selected Category (in which you have ‘save’ permissions).
- A chart “Name”.
 - In the event the chart name is not unique within the category, you will be asked to provide a new name or overwrite the existing chart.

To Save a chart:

- Start a new chart or open an existing chart.



- Progress to the “Preview” stage, then click on the “Save” button.
- In the save dialog window, select a category then enter a chart “Name” (Title and Description are optional).
- Click “Save”.

Note: “Save As...” is an option used to create a copy, or clone, of the current chart. This operation will not copy the linked charts or picklists, as those will require their own “Save As...” operation.

7.4.1 Save as Snapshot

The default save option for a Dynamic Data chart, is to remain a Dynamic Data chart. This implies that every time this chart is opened, it will load dynamic, real-time data from its data source. Therefore, during the Save-As command, the radio toggle will be set to “Dynamic Data Load”, as shown below.

Saving a “Snapshot” will sever the dynamic connection from the chart to the data source, and therefore return the same data ever after. The data that is represented in the chart at the time the chart is saved will be saved along with the chart. Every time the chart is opened afterward, the chart will display static data saved within the chart and not a real-time dynamic data loaded from the dynamic data source like an uploaded Excel file or database table/view.

Note: Take caution saving a Snapshot chart.

Note: Saving a Snapshot chart is irreversible.

Note: Some users find it safer to rename the chart while saving a Snapshot, so the original, dynamic chart is not overwritten.

7.4.2 Chart Title

The purpose of a chart title is to allow the chart to be saved using a ‘technical’ name while allowing the display name of the chart to remain user friendly. Chart titles are visible on the chart, and therefore will be visible on the dashboards containing the chart. There are properties that allow customization of the text and appearance of the title. The title is also displayed at the time a chart is being opened.

The chart title is optional when saving a new chart.

Note: The title field is not searchable.



7.4.3 Chart Description

The purpose of a chart description is to allow the chart to contain notes which are displayed at the time a chart is being opened.

The chart description is optional when saving a new chart. After the chart has been saved, there are properties to edit the value of the description.

Note: The description field is not searchable.



8. Designing Charts

The first stage of designing a chart requires configuring the source, and technique, for how the chart will be populated with Data. A new chart can be created by providing data using the two basic states for data:

1. Dynamic Data

- a. Query data from your local database
- b. Query data from Excel files (uploaded via Workbook Database)

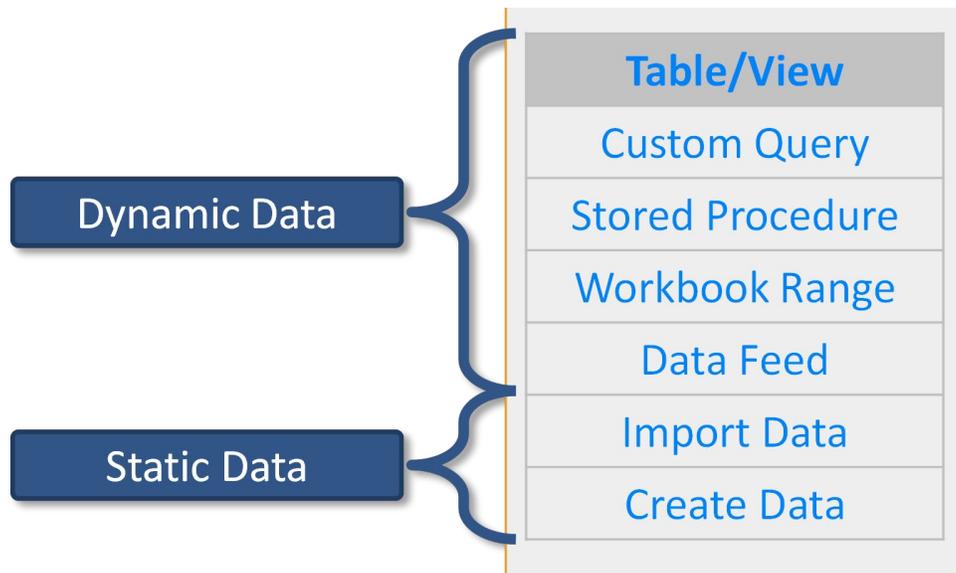
2. Static Data

- a. Entering the data manually
- b. Importing the data from an Excel spreadsheet (this is a one-time import)
- c. Importing the data from a (CSV) comma separated values file (this is a one-time import)

The image below, identifies the first decision to make when creating a new chart.

Note: Refer to Chapter 7, "Chart" to learn how to initiate a new chart.

Note: Creating charts by dynamically retrieving data requires knowledge of Data Structure, Named Ranges, Tables, Views, Queries and Data Columns.

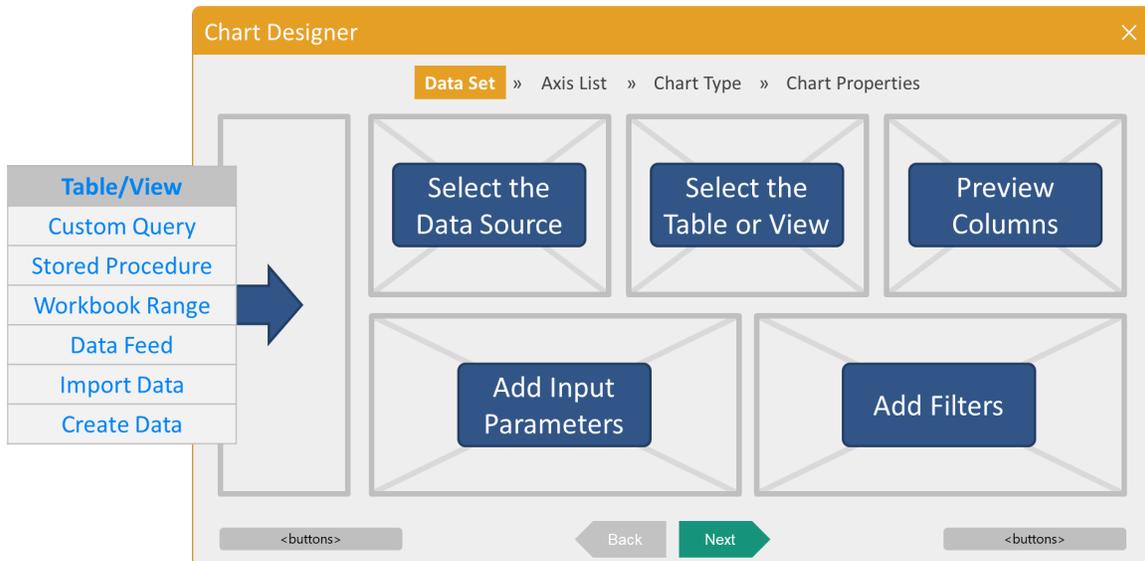


8.1 Dynamic Data

This section describes the dynamic data methods for creating a chart. Creating a static data chart is discussed in Section 8.2, “Static Data”.

8.1.1 Table/View

When first creating a new chart, selecting the “Table/View” data option will allow a chart to dynamically connect to existing Tables or Views. The image below shows the framework for this stage. The minimum requirements include selecting the Data Source and Selecting the Table or View.



8.1.1.1 Define Data Set

The Data Source section of the window lists all the available data sources for which you have permission to use. Data Sources are configured in the iDashboards Administrator Application by an iDashboards administrator. If “Data Source Access Control” is enabled by the administrator then it is possible there are data sources hidden from view for particular users.

- **Select a Data Source** – Click on an item in the Data Source list, and the Table or View options will appear.
- **Select a Table or View** – Here, a list of database schemas within that data source will appear as folders. These schemas/folders can be expanded or collapsed to display the tables or views. Click on an item in this list, and the remaining fields will appear.
- **Data Column Preview** – Column names and the associated column data type are displayed in this section of the window.
- **Specify Parameters** – See Chapter 13, “Input Parameters” for details.
- **Specify Data Filters** – See Section 10.5, “Using Filters” for details.

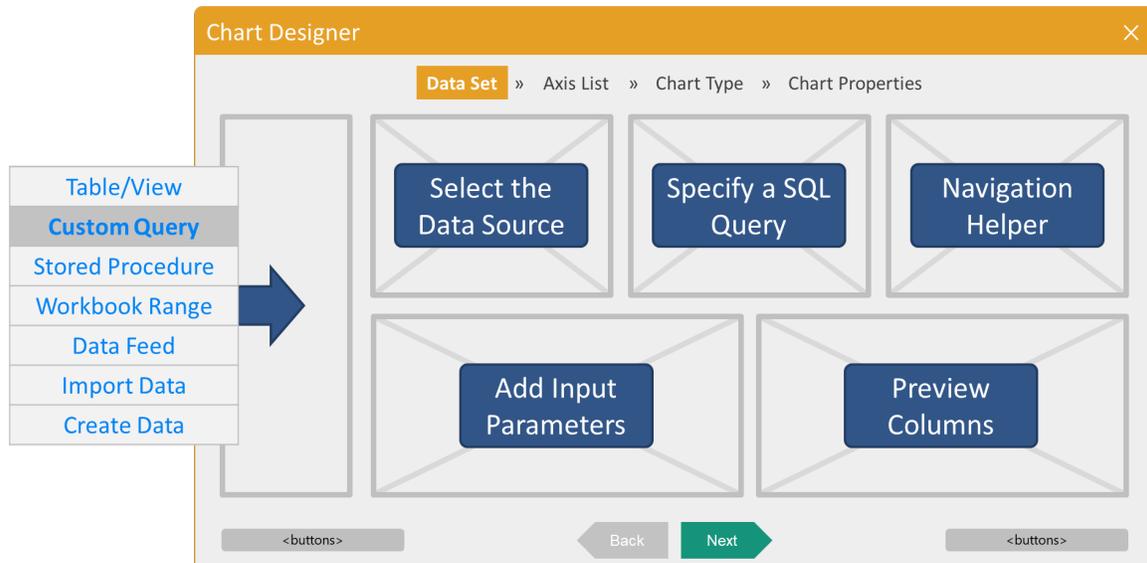
Click “Next” to continue to the next stage called ‘Axis List’.



8.1.2 Custom Query

This section describes how to use a “Custom Query” to retrieve chart data. Using this option requires a certain knowledge to understand the required syntax and expectation for using this powerful option. The image below shows the framework for this stage. The minimum requirements for the query demands returning a single column.

Note: Queries defined here, within a chart, cannot use the SQL JOIN clause. It is only possible to select a single database table or view. Filtering must take place within the Query.



8.1.2.1 Define Data Set

The Data Source section of the window lists all the available data sources for which you have permission to use. Data Sources are configured in the iDashboards Administrator Application by an iDashboards administrator. If “Data Source Access Control” is enabled by the administrator then it is possible there are data sources hidden from view for particular users.

- **Select a Data Source** – Click on an item in the Data Source list, and the SQL Query editor will appear.
- **Specify a SQL Query** – The first word must be “SELECT”, followed by the remaining query. Some options include selecting data from a single table or joining tables.
- **Navigation Helper** – This window allows you to see a list of tables and views, along with their associated columns, within the selected data source. Drag the Table name or Column name into the SQL dialog box. You may use the list to build your SQL statement or write it manually.
- **Retrieve Columns** – Clicking this “RED” button will process the current query. If there are errors, a window will appear stating the general problem. If there are no errors, the button will become “GREEN” and the validated columns from the query will appear in the bottom-right portion of the Chart Designer. The “Data Type” column will be automatically chosen by iDashboards to represent one of three data types: String, Number or



Datetime. iDashboards will examine the data type of the column that is selected under the “Data Column” and apply the most appropriate data type.

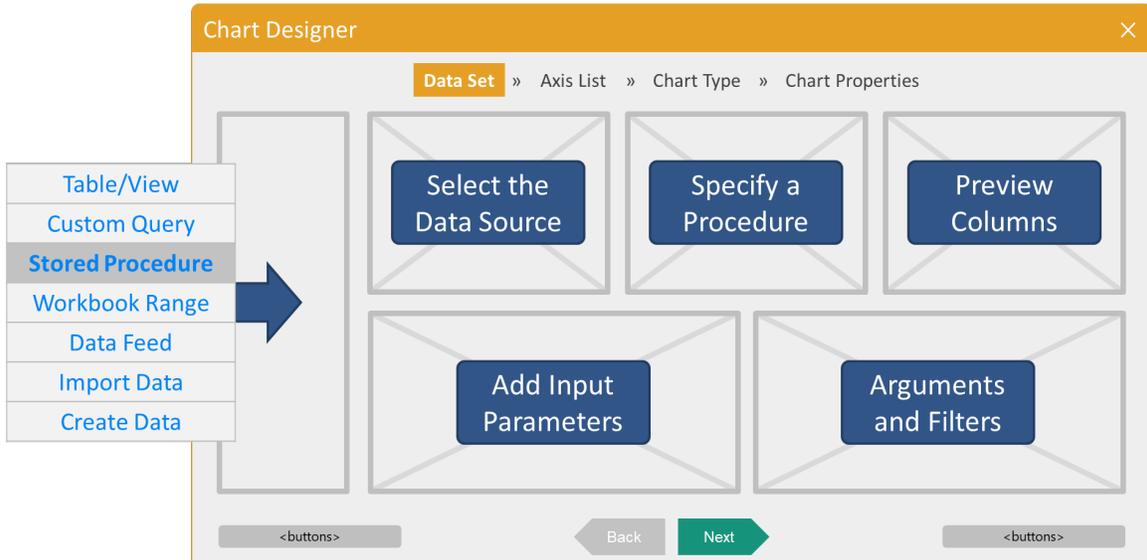
- **Specify Parameters** – See Chapter 13, “Input Parameters” for details.
- **Specify Data Filters** – See Section 10.5, “Using Filters” for details.

Note: When creating Custom SQL queries, the use of fully qualified table names is encouraged. If wildcards are used to collect columns, changes in the table may result in a chart error. Charts with an error will require re-mapping between the Axis Name and Data Columns.

Click “Next” to continue to the next stage called ‘Axis List’.

8.1.3 Stored Procedure

This section describes how to use a “Stored Procedure” to retrieve chart data. The Administrator’s Manual outlines the process of adding a stored procedure to iDashboards. Once that is completed, users can build a chart that utilizes a stored procedure.



8.1.3.1 Using a Macro as an Argument

A variety of macros exist which can be used as input arguments for a stored procedure; $\${user}$, $\${value:<Axis Name>}$, or any of the date macros (or their derivations) listed below. It is essential that the stored procedure was written to accept the macro data type offered as an input argument.

Date Macros		
$\${ds}$	$\${currmonth, "MM"}$	$\${currmonth:begin + 3, "MMMM dd, yyyy"}$
$\${ts}$	$\${currmonth, "MMMM"}$	$\${curryear}$
$\${currdate}$	$\${currmonth, "MMMM"}$	$\${curryear-3}$
$\${currdate-30}$	$\${currmonth + 2, "MMMM"}$	$\${curryear + 2}$
$\${currdate+30}$	$\${currmonth + 2, "MMMM d, yyyy"}$	$\${curryear + 2, "dd MMM yy"}$
$\${currdate+90, "MMMM dd, yyyy"}$	$\${currmonth:begin}$	$\${curryear:begin}$
$\${currmonth}$	$\${currmonth:end}$	$\${curryear:end}$
$\${currmonth-3}$	$\${currmonth:end - 2}$	$\${curryear:begin + 1, "dd MMM yy"}$
$\${currmonth+5}$	$\${currmonth:begin + 3, "MMMM d, yyyy"}$	

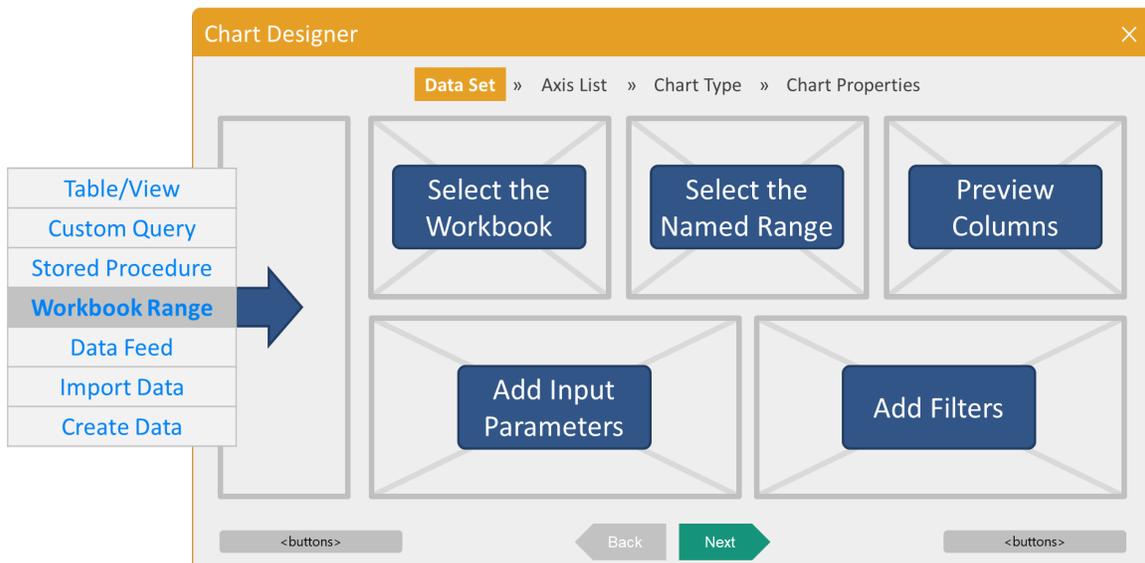


The `#{user}` macro simply returns the username of the currently logged in user. Therefore, when a chart that uses this macro as an input argument is opened, the username of the current user is sent to the stored procedure as an input argument.

Drilldown Note: The `#{value:<Axis Name>}` macro must be used in a chart that acts as a target chart for a drilldown. When a chart that uses this macro is opened, it will evaluate any drilldown parameters being sent to it for a parameter called `<Axis Name>`. If it finds a parameter called `<Axis Name>`, the macro will return the value of the `<Axis Name>` parameter.

8.1.4 Workbook Range

This option is visible only when the iDashboards administrator has configured an Excel workbook database (Refer to the Administrator Manual for details). If the Workbook Range option is selected, a list of Excel files will appear. These files have been uploaded into the workbook database prior to creating the chart. When a file is selected, the named ranges within the file will then be displayed. The image below shows the framework for this window.



8.1.4.1 Define Data Set

From this window, access to the Workbook Range is allowed. Alternatively, access to the Auto Uploader (for scheduling Excel file uploads) is controlled by a system setting within the Administrator Application. See the Administrator's Manual for details.

Note: Users will have access to all uploaded Excel files and the associated Named Ranges - "Data Source Access Control" does not affect this screen.

- **Select a Workbook** – Select a file name in the Workbook list, and the Named Range window will appear.
- **Select a Named Range** – Select the name of a Named Range, and the remaining fields will appear.
- **Data Column Preview** – Column names and the associated column data type are displayed in this section of the window.



- **Specify Parameters** – See Chapter 13, “Input Parameters” for details.
- **Specify Data Filters** – See Section 10.5, “Using Filters” for details.

Click “Next” to continue to the next stage called ‘Axis List’.

8.1.5 Data Feed

A relational database contains one or more data tables. Each table has a name, and a specific set of columns. Each column has a name and a specific data type.

An iDashboards data service provides something similar to a database table called a data feed. Like a table, a data feed has a name to distinguish it from other data feeds within the same data service. Also like a table, a data feed has a specific set of data columns, each with a unique name (within the feed) and data type.

Note: Setting up Data Feeds is an administrative task.



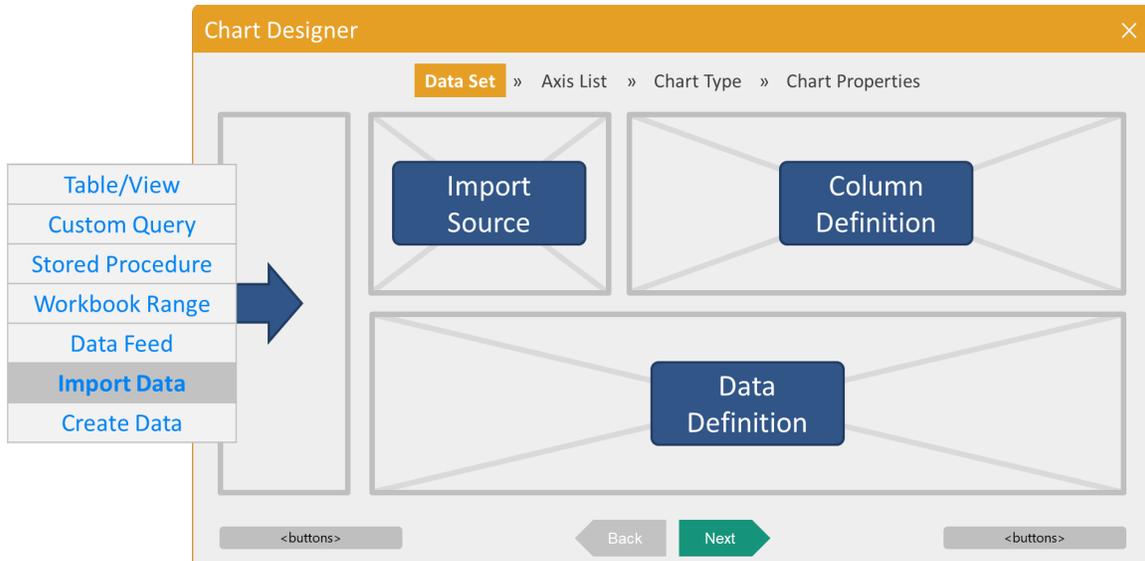
8.2 Static Data

This section describes the static data methods for creating a chart. Creating a dynamic data chart is discussed in Section 8.1, “Dynamic Data”.

Note: Input Parameter and Filtering is not functional with charts using a Static data source.

8.2.1 Import Data

When first creating a new chart, selecting the “Import Data” option will allow a one-time import of data from either an Excel file or a comma separated value file (.CSV).



8.2.1.1 Import from Excel

Importing data from an Excel file or Delimited file are essentially the same, with the exception being Excel files can refer to specific Worksheets or Named Ranges; whereas a delimited file can only contain a single dataset.

Selecting the “Excel” option in the dropdown will display options related to using an Excel spreadsheet as a data source. Importing Excel data using this method will import the data as static data into the iDashboards application.

Note: The imported data will only get saved within iDashboards when the chart is saved.

Notice

Excel files can also be used as a dynamic data source, meaning that every time you upload an updated spreadsheet into iDashboards, the new data will be reflected in every chart that is pointing to it. To set this up, an iDashboards administrator must use the Workbook Database (See the iDashboards Administrator’s Manual for more information.)

8.2.1.1.1 Define Data Set

Check the box ‘Data includes header row:’ if the first row of the Excel data contains the name of each column. When a header row is used, the Excel column names are retained and stored

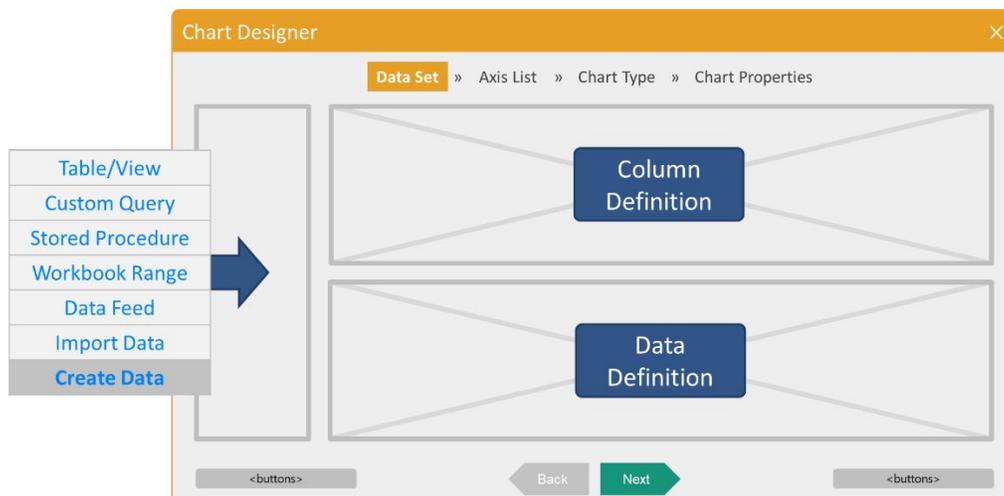


along with the data in iDashboards. Column names should be less than 30 characters long, should begin with a letter and should only contain letters, digits and spaces. If the box is not checked, the first row in the Excel data is imported as the first data row.

- **Select** – Click the “Select...” button to locate the file.
- **Browse** – Files can be imported from local or network storage locations. Select one file.
- **Import Method** – There are essentially three methods for defining the import method:
 - **<none>** - If the Excel file is very ‘clean’, no further action is required prior to import.
 - **<Worksheet Name>** - If the Excel file contains multiple worksheets, by default, the first worksheet will be imported unless the worksheet name is entered.
 - **<Named Range>** - *Preferred* If the Excel file contains Named Ranges, enter its name into this field.
- **Import** – Click the “Import” button to extract the data.
- **Review/Edit** – Review the results:
 - If the column preview or the data values appear incorrect, examine the results and troubleshoot as necessary. Ensure the Excel data is in an appropriate format, shape, and structure. Best results come from using the Named Range options. Make changes to the Excel file and attempt to import the data again.
 - If the column preview or the data values appear mostly correct, consider utilizing the options to add/remove individual columns or rows. Editing the actual values can take place at this time.

8.2.2 Create Data

Creating data manually might be an appropriate option when no other option offers the same benefit. One reason not to use this option is due to the time it takes to manually identify all columns and all row-values for an entire data set. However, one reason to use this option is to quickly create a chart with a few values that never, or rarely, change.



8.2.2.1 Define Columns

A basic minimum requirement for all iDashboards charts is one X-axis and one Y-axis. However, if the intention is to have the same values appear in both the X and Y axis, then it is valid to have a data source with only a single column. By default, the “Create Data” option has a single pre-defined column, and therefore requires no additional information prior to configuring all other stages of the chart. However, doing so, will be of little value because there won’t be any data values to help render the chart graphics!

Columns can be added, deleted, or sorted (for personal preference). To reorder the columns, click-and-drag the column into the desired position. To create or delete columns, use the “Add” and “Remove” icons are shown below.

Data Set Column	Data Type
State	String
Population	Number

Enter Column Name – The image above shows two columns, each with a customized name. Some charts have visible labels and therefore merit grammatically correct names. However, at this time it is not *necessary* to be strict about the grammatical value because the next stage of creating a chart allows for using a mapped name. However, it is often found to be helpful to have accurate names at this time.

Select Column Type – The user must select one of the three standard iDashboards data types; String, Number or Datetime for each Column Name added to the chart.

8.2.2.2 Create Data

For each Column Name defined, the user will be able to enter data associated to the data column, by using the lower portion of the Chart Designer. The image below shows three records manually added by the user. To reorder the rows, click-and-drag the row into the desired position using the ≡ icon. To create or delete rows, use the “Add” and “Remove” buttons shown below.

	State	Population
≡	MI	10000000
≡	FL	21000000
≡	CA	40000000

8.2.3 Editing Data

After a chart has been configured and saved, it is entirely possible to make additional changes to a “Static Data” chart.

- Columns and data can be added without causing issues to the existing chart.
- Columns that are removed or renamed will lose their mapping within the “Axis List” stage, requiring additional steps to resolve the mapping.
- Data values can generally be added without causing issues to the existing chart.
- Data values can generally be removed without causing issues to the existing chart.

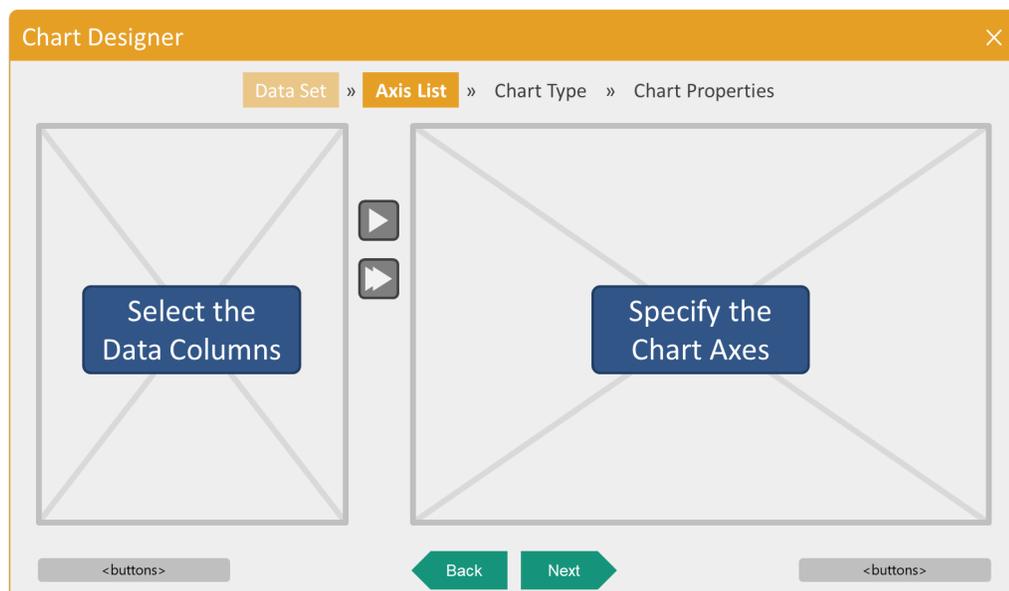
8.3 Define Axis List

At the Axis List stage, available data columns from the Data Set are displayed on the left. Columns that are needed for the chart design are added to the axis list on the right.

Note: It is not necessary to specify all columns. Best Practices suggest specifying the least number of columns needed to support the desired chart type.

In the image below, chart axes can be populated using any of the following methods:

- Double-click the data column
- Drag-and-drop the data column
- Select column then click single arrow
- Multi-select column then click single arrow
 - Use CTRL+click to multi-select
- Click multi arrow to add all columns
- Manually add and map each column



- **Axis** – A basic minimum requirement for all iDashboards charts is one X-axis and one Y-axis. Some charts require additional axes or specific data structures.
- **Axis Names** – As Column Names become Chart Axes, the ‘Name’ value will be made proper with an UPPER-CASE first letter. However, the label name may not be a friendly name if displayed on the chart. Once the axes appear on the right-hand side, double-click the name and update the label to a friendly name.
- **Axis Data Column** – If the user manually created the axis, then the users will have to manually map each axis to the data columns from the data source.
- **Axis Data Type** – If the data type is automatically determined by recognizing the type of data associated to the mapped column, it will be set when the axis is added to the list, otherwise it should be manually set.
- **Axis Function** – Depending on the Data Type, options may include “SUM”, “AVG”, “MAX”, “MIN”, or “COUNT”.
- **Axis Pivot** – Define a column(s) to use as a pivot.
- **Axis Hide** – Hide a column.

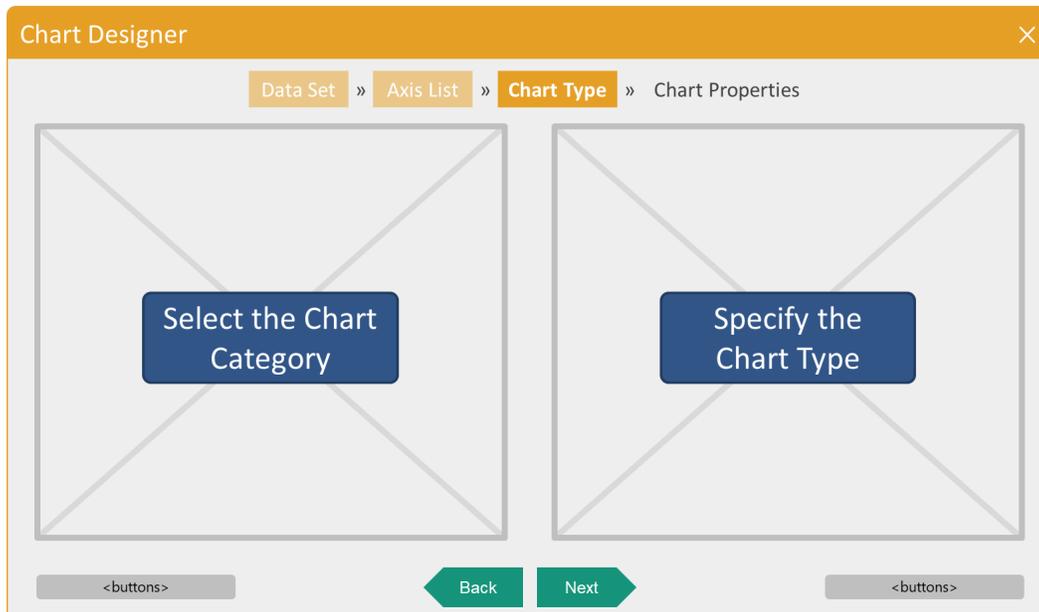
Note: At this time, the minimum requirements for configuring chart data using a Tabular chart is complete. To preview the data before viewing the chart, click the ‘View Data’ button.

Click “Next” to continue to the next stage called ‘Chart Type’.



8.4 Define Chart Type

This stage is used to select the desired chart type. As noted in the previous section, every chart requires a minimum of two axes. Some charts require additional axes or specific data structures. To select a chart, first browse the Chart Category on the left and then select the actual Chart Type on the right.



Based on the Axis List definition, some charts will immediately be unavailable for selection, as shown by the strikethrough within the name of the chart, and a darkened chart tile. If the invalid chart type is selected, a notice with additional details will appear.

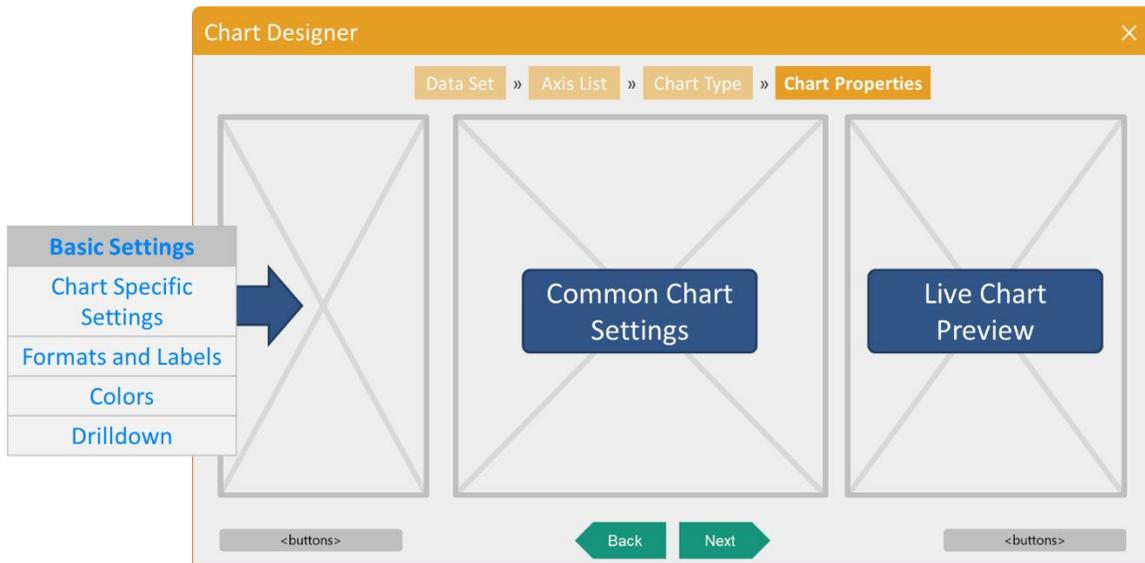
Note: Sometimes, the chart is available to select, but until the data is rendered in the chart it is too early to determine if the chart will provide the desired results. Refer to the “Preview” stage for experiencing a live preview of the chart and data.

Click “Next” to continue to the next stage called ‘Chart Properties’.

8.5 Chart Properties

Chart Properties determine the various display characteristics/settings of the chart, such as the Chart Title, Number/Date settings, Colors, and many other display options within the chart. This stage allows you to change any of the chart settings.

Note: If you don't have Save permission for a chart, you may still change that chart's properties and then save it as a different chart within a Category for which you have Save privilege such as your Personal category.



As seen in the image above, there are three main sections to navigate within the Chart Properties. The sub-menu on the left will update the “Common Chart Settings” and the “Chart-Specific Settings” will always be displayed.

Note: “Formats and Labels” will not appear in the sub-menu if the chart data contains no numbers or dates.

The majority of the settings are either self-explanatory or can be understood with a little experimentation. For this reason, and because the various “Chart Specific” settings sections can contain 100’s of unique settings when taking into account all of the chart types, these settings are not documented in this manual.

8.5.1 Live Chart Preview

While adjusting the visual properties of a chart, the live chart preview will assist with the chart design. Using this preview can help verify all of the settings have been configured correctly, and the visual elements are positioned properly, prior to leaving the Chart Designer dialog.

The following preview functions are available:

	<ul style="list-style-type: none"> • Normal – (default) This mode will display the live chart preview adjacent to the chart properties. The divider that separates the preview from the properties can be repositioned.
---	---

	<ul style="list-style-type: none"> • Maximized – This mode will display the live chart preview using the largest frame possible. This will collapse the properties. If the chart is being designed/edited from within the Dashboard Designer, then this mode will use the same frame size and aspect ratio when viewing the chart. However, if the chart is being designed/edited from outside the Dashboard Designer, then the live preview will use a generic full-window. Either situation allows the user to make temporary adjustments to the frame borders to experience the behavior of the chart if resized.
	<ul style="list-style-type: none"> • Minimized – This mode will close the live chart preview and only display the chart properties.
	<ul style="list-style-type: none"> • Information – This button will re-open the dialog that contains scenario-specific notes about what can be adjusted on the chart, within the live preview. For example, positioning the legend is only possible within the live preview. <p><i>Note: This informational dialog, by default, is set to appear the first time a new chart build reaches the Chart Properties, and every time the chart is edited. This can be turned off by unselecting the 'Show this message automatically' checkbox. This change will persist across all chart building and login sessions. Be aware that the unselect is temporary and it will revert to being selected when some event clears the browser's Local Storage, like closing the browser or clearing its cache.</i></p>

Functions of the live preview include:

- View and interact with the chart (see note above)
- Reposition the legend (if chart type utilizes a legend)
- Test the pivot selection (if “Show Pivot Selectors” is selected)
 - Set the default pivot value (if shown)
- Test the input parameter (if “Show Parameter Input” is selected)
 - Reposition the parameter legend (if shown)
- Reposition the note (if chart is enabled to use the note)
- Change the visibility state of various elements to ‘minimized’
- Adjust the column widths (if using a tabular chart)
- Adjust the frame boundaries (top, bottom, left and right) to simulate how the chart will react if the dashboard (or browser) are adjusted to a different size.

8.5.2 Basic Settings

Basic Settings are available for all charts regardless of the chart type. These settings allow you to change chart features such as title appearance, legend appearance, mouse-over value, note appearance, sort order, chart refresh interval, etc.



Note: As with all other edits in the Chart Designer, changes will only get applied once the chart is saved.

8.5.2.1 Naming a Chart

All charts require a name upon the initial save. However, using this property it is possible to name or rename a chart (Refer to Section 7.4, “Saving the Chart (or Save As)”). Chart names are required within iDashboards and the name of a chart can be different from the title. Refer to the nearby property ‘Chart Title’, which is optional, and has additional formatting options. Users must save the chart after changing the name for the rename to take effect.

Note: Performing a ‘Save As ...’ on the chart will similarly allow edits to the chart name, title and category.

8.5.2.2 Moving Chart to other Categories

A user can move an existing chart to another category by changing the Category value using the dropdown list. The dropdown will contain all categories for which the user has “Save” access. Users must save the chart after selecting a new category from the dropdown for the category change to take effect.

8.5.2.3 Chart Title & Size

The title of a chart can be entered into the ‘Chart Title’ text box. This property is optional and has the following formatting options:

Chart Title Size – Font size is controlled by adjusting the slider bar or editing the numerical value of the font size itself.

Chart Title Alignment – The chart title will always be top-aligned. This option controls the horizontal alignment options: ‘Center’ (default), ‘Left’ and ‘Right’

Chart Title Word-Wrap – By default, lengthy chart titles do not word wrap, and will truncate to fit the width provided by the associated frame. When truncated, the user can position the mouse over the title to read the entire value. Enabling this option will allow the chart title to word-wrap, consuming up to 100-characters. When chart frame widths are narrow and chart titles are long and word-wrap is enabled, it is possible to have the title infringe upon the graphics of the chart as the title section increases in height and the chart graphics decrease in height.

8.5.2.4 Chart Background Color

The chart background and default text color can be controlled under Basic Settings as well as the ‘Colors’ tab.

8.5.2.5 Chart Background Transparency

This setting allows the user to set the transparency percentage of the chart background. If it is set to 0, then the chart will display the chart background color. If it is set to 100, then any dashboard color or image will show through the chart background.

8.5.2.6 Chart Background Image

Any chart type can set a custom background image for the frame. Setting the transparency and stretching of the image is easily controlled with two options. Background Transparency settings will control transparency of the background color. If left at zero, the chart background image will



not be displayed. To view the image, increase the transparency. Setting the value to 100% will ignore the chart background color to display the pure image.

Click the “Set...” button. An image selection window will appear, allowing an image to be selected from the iDashboards server. Navigate the content folders on the left and select the file. Supported file types are .PNG, .JPEG, .SVG and .GIF formats.

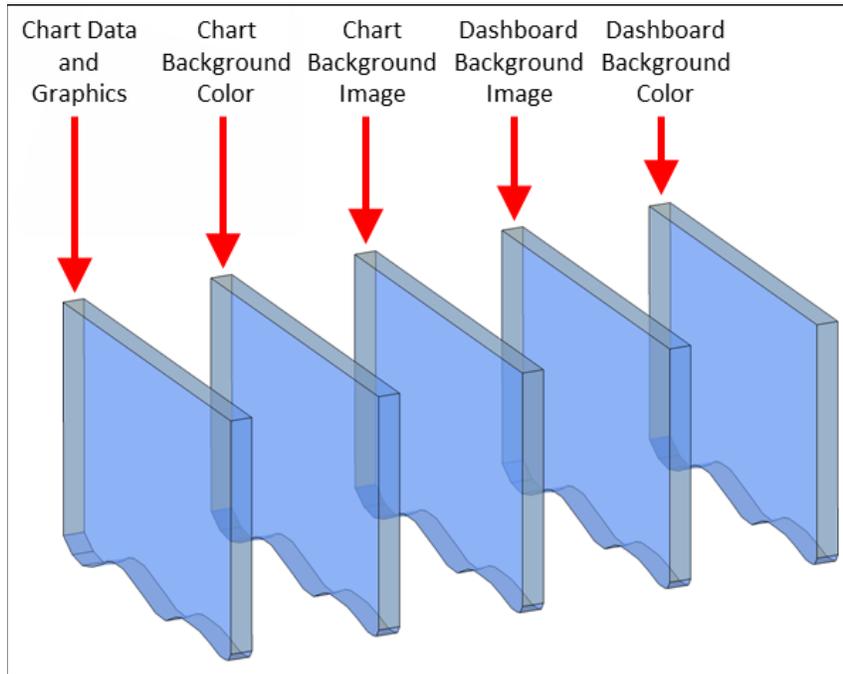
Note: Images can be uploaded to iDashboards through the Administrator Interface or from the image selection dialog. Images may also be referenced by URL, but this setting requires the administrator to enable the Proxy Server or Cross-Domain policy. See the Administrator’s Manual for details on adjusting these settings.

8.5.2.7 Fit Chart Background Image to Frame

By default, the chart background image will stretch to fill the entire frame. A checkbox controls this setting which allows for stretched or native image display. If the option is turned off, the image will display at the native resolution.

8.5.2.8 Color Layers and Transparency

Regardless of background color or background image, the chart data and graphics will always be presented on the ‘top’ layer. Some image types support transparency. Using images with transparencies allow for ‘lower’ layers to be seen by the user (like dashboard background images and dashboard background colors.) Illustrated below is the layer sequence.



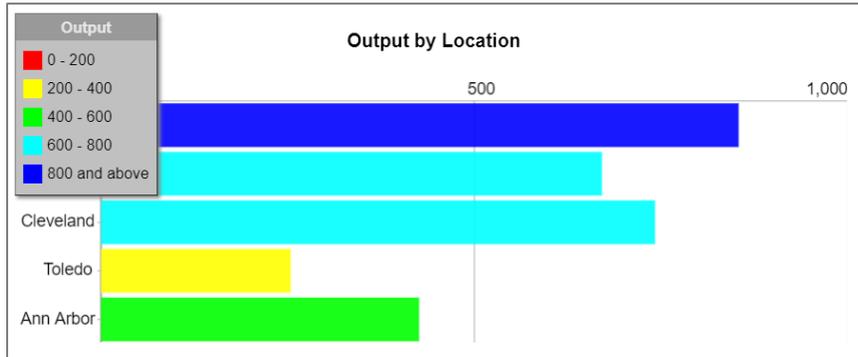
8.5.2.9 Animation

By default, most charts have a unique animation when the chart is first opened or refreshed. This animation can be turned off by un-checking the ‘Enable Animation’ checkbox. Turning off an animation is sometimes helpful when hundreds or thousands of data points need to render and the end user does not want to wait.

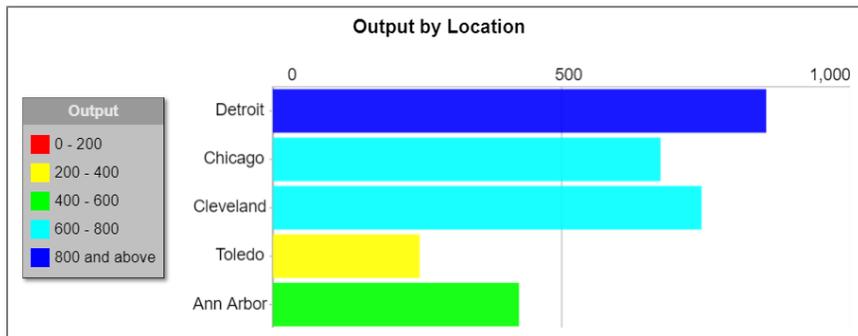
8.5.2.10 Graph Margins

Graph Margins allow for padding of whitespace in the regions above, below, to the right, and to the left of a chart. The padding above the chart will always be below the title to keep the title at the top of the dashboard frame.

Below, the chart has a legend that has a grey background and covers part of the chart.



By changing the left Chart Margin to 120 pixels the same chart can contain the whole Legend and chart without blocking any of the chart's data points.

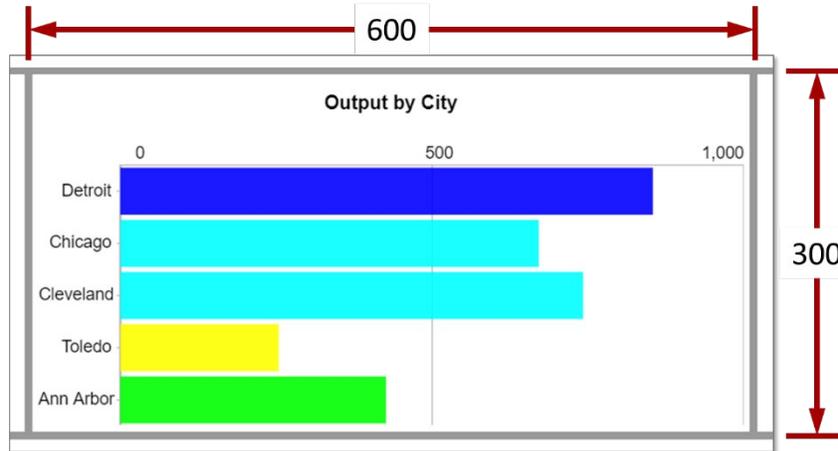


Resizing the frame borders or application window will always stretch/shrink the size of the chart, but will always keep the same margin values.

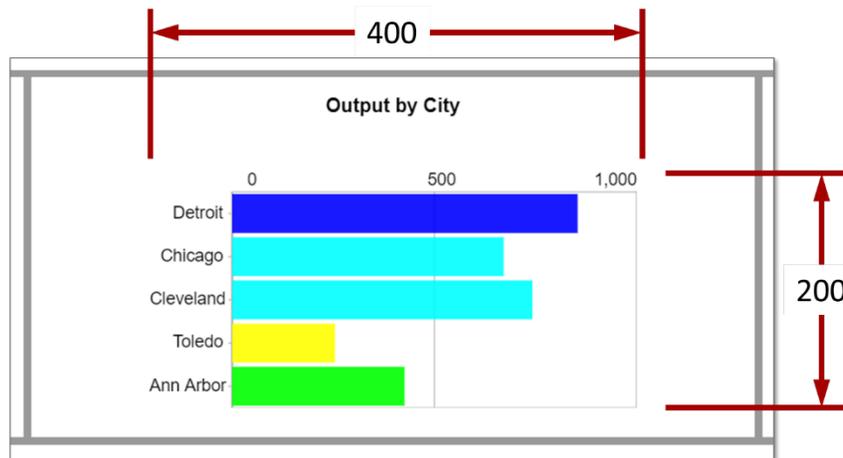
Note: A chart cannot become smaller than 100 pixels by 100 pixels.

8.5.2.11 Max Graph Width/Height

As an alternative to adjusting the graph margins, which often change the size of the displayed charts, a user can adjust the maximum graph width or height. Setting the 'Max Graph Width' or 'Max Graph Height' will restrict a chart from growing beyond a set size. By default, both values are set to 'blank', indicating the property is not being used. Below is an example of a chart within a frame that is 600 pixels wide and 300 pixels tall. Aside from the space occupied by the chart title, the chart will consume as much of the frame as possible. Dimensions have been placed over the screen capture for illustration purposes only.



The same chart is also shown below, but using the Max Width = 400 and Max Height = 200. By default, the chart will remain centered within the frame.



Note: Chart margins can also be applied to position the chart in an offset location.

8.5.2.12 Chart Legend Controls

By default, the chart legend box is displayed next to a chart and displays the chart labels with their associated colors.

Options include: **Show Legend** (default), **Hide Legend**, and **Minimize Legend**

Builder features include:

1. Ability to completely hide the legend.
2. Set the legend background color, text color and the degree of transparency of the legend box, from 0 to 100%.
3. Set the initial view of the legend to maximized or minimized.
4. Set the initial position of the legend (within the boundaries of the chart frame).

Viewer features include:

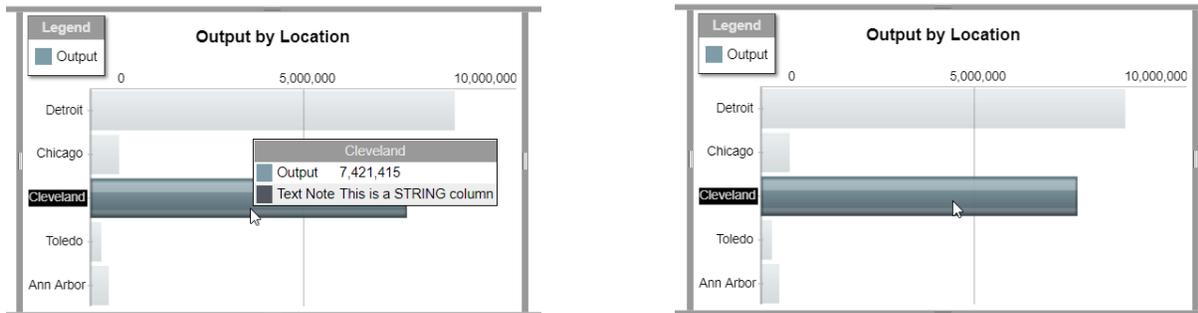


1. Temporarily maximize or minimize a legend.
2. Temporarily reposition a legend.

8.5.2.13 Mouse-Over Value Display

When a user places the mouse cursor over a chart, most chart types will display a small pop-up that contains the chart label(s) and their associated values for the X-axis data point they are hovering over. This is default functionality by design. However, under certain circumstances (such as too many data points on the chart or a pie chart where data is already visible on the slices), a user may want to turn this pop-up off. The 'Show Mouse-over Value' checkbox of the Chart Features tab allows you to turn on/off the mouse-over pop-up.

Note: the "Text Note" label shown below. This comes from an additional string-based Y-axis (Y2 column is "Text Note"). Since it is a string, no value for it can be displayed in the column chart (only numeric data can be displayed). However, it is displayed in the pop-up. This functionality allows you to add text for each X-axis value that is only displayed when you mouse-over the values.



8.5.2.14 Chart Notes

Chart Notes simulate a "sticky note" behavior and can be used to display a custom note about the chart's state (such as "low is good" or "high is bad").

Options include: **Hide Note** (default), **Show Note**, and **Minimize Note**

To display a Chart Note, simply choose "Show Note" in the Note Visibility dropdown in Chart Features. Then, select "Edit..." to populate the contents of the note. During the "Preview" stage, the Chart Note can be repositioned and resized.

Note: On the lower right corner, the resize handle can be clicked and dragged to resize the dimensions of the box.

Note: A chart cannot have more than one note.

Note: The text color of a chart note is black.

Builder features include:

1. Ability to completely hide the chart note.
2. Set the value of the note title and note body



3. Set the chart note background color and the degree of transparency of the chart note, from 0 to 100%.
4. Set the initial view of the chart note to maximized or minimized.
5. Set the initial position and size of the chart note (within the boundaries of the chart frame).

Viewer features include:

1. Temporarily maximize or minimize a chart note.
2. Temporarily reposition or resize a chart note.

8.5.2.15 Menu Button Visibility

By default, this setting uses the dashboard setting in determining whether to show the chart menu buttons or to hide them. Alternatively, this option can override the dashboard setting for chart-by-chart customization.

Options include: **Use Dashboard Setting** (default), **Show Menu Button**, and **Hide Menu Button**.

8.5.2.16 Right-Click Menu Enabled

By default, for new charts, this option is selected. It enables the mouse's right-click to open the chart menu, at the location of the click.

8.5.2.17 Match Type

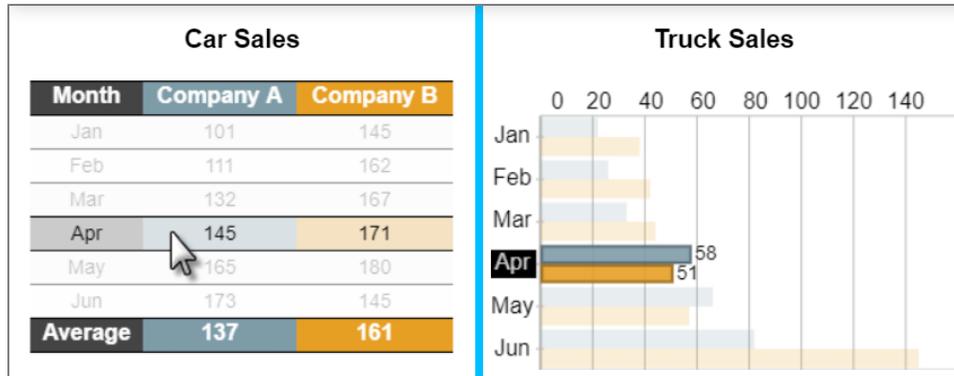
Synchronized highlighting works in conjunction with the Match Type settings of a chart. Match Types help manage the underlying matching rules that in-turn control the visual display of related data points within the same chart and across different charts. This technology, called 'Interactive Intelligence', is built-in functionality of iDashboards. It allows you to hover over a data point in one chart and have related data points in all charts within the same dashboard appear highlighted. This is a powerful feature to highlight the data relevance across various charts in a meaningful fashion and help users get a better insight of the information being presented.

See Section 6.7, "Synchronized Highlighting" for more information.

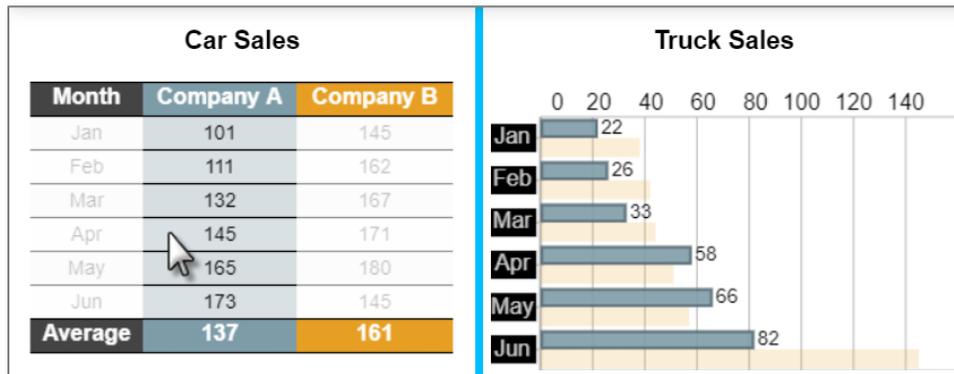
The possible options for Match Type are:

- **X Value** – Matching highlights all data points in all charts that have the same value for the X-axis (independent variable) as the data point you are hovering over.

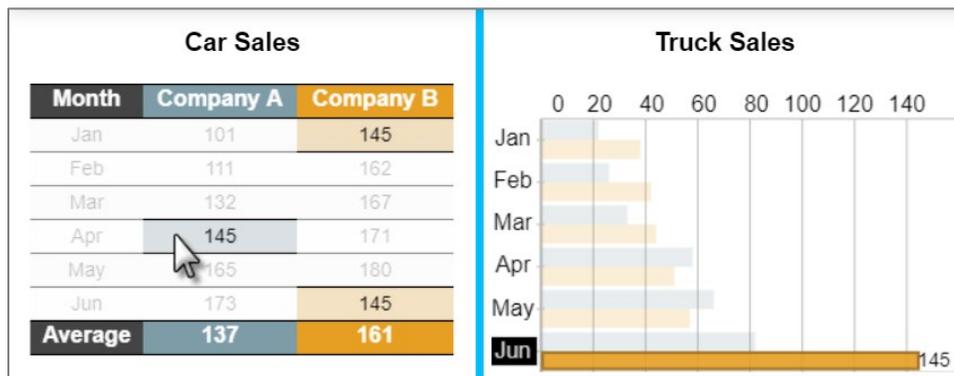




- Y Axis Name** – Matching highlights all data points in all charts that have the same label of the Y-axis (dependent variable) as the data point you are hovering over. As you hover over the label “Company A” in the tabular chart, all data represented by the same axis label of the bar chart is highlighted. This is particularly noticeable on charts with multiple Y-Axes.

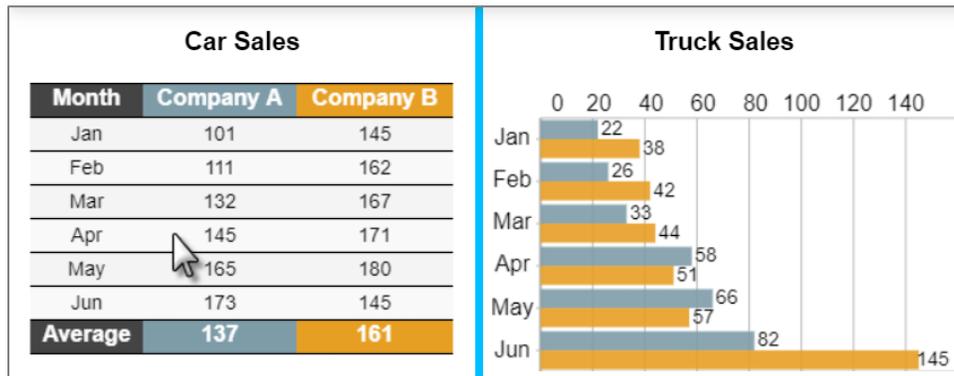


- Y Value** – Matching highlights all data points in all charts that have the same value of the Y-axis (dependent variable) as the data point you are hovering over.



- None** – The chart will no longer highlight the data point you are hovering over or the matching data point within related charts. Interactive Charts, like Speedometers, Bullets,

Thermometers and others with multiple rows of data will require the Button Row be enabled to control the row being displayed within the chart.



Note: Full utilization of Match Type exists only within the chart types: Bar, Column, Line, Tabular and Metric. Other chart types may not fully utilize all Match Type capabilities.

8.5.2.18 Sorting Data Results

Sorting allows chart data to be sorted in a pre-defined order. The data can be sorted ascending, (a-z, 1-10) or descending (z-a, 10-1). Chart data can be sorted on any column by choosing the column in the “Sort On” dropdown menu. Leaving the “Sort On” dropdown blank will draw a chart without sorting, instead pulling the data in the order the data is retrieved from the data source.

Sort On options include: **<none>** (default), <list of all axes>

Sort Order options include: **Ascending** (default), and **Descending**

8.5.2.19 Maximum Number of Data Rows

The purpose of the ‘Maximum Number of Data Rows’ feature is to limit the number of rows returned from a chart’s data source for performance and visual-integrity purposes.

This will allow the user to create a chart that will display such things as the ‘Top X’ or ‘Bottom X’ rows of data from a data source. For example, a user may want to create a chart that only returns the Top 5 sales associates based on sales volume from a department of 100 sales associates. In this case the user can set the ‘Maximum Number of Data Rows’ setting to 5, resulting in a dataset consisting of only the first 5 rows from the data source. To achieve the actual top 5 associates based on sales the dataset would need to be properly sorted in the data source so that the first 5 rows were also the 5 associates with the highest sales volume. This would be a descending sort on sales volume in the data source.

Note: Sorting can also be performed via the ‘Sort On’ chart property, however this sorting takes place after the ‘Maximum Number of Data Rows’ dataset has been returned to the client and will thus sort only the 5 values returned, which may not be the intended values.

The ‘Maximum Number of Data Rows’ will also allow you to prevent the ‘max rows exceeded’ error from displaying when the number of rows for a given chart exceeds system-defined



limitations for both pivoted and non-pivoted charts. In many cases, a cropped dataset by iDashboards may be an expected result, and an error alerting the user to this issue may not be desired. In general, on any pivoted or non-pivoted chart the 'Maximum Number of Data Rows' option can be set equal to the respective system-defined limitations resulting in the same dataset being returned as would be if 'Maximum Number of Data Rows' was not configured, however in this case the user will not be presented with an error notifying them that the maximum number of rows has been exceeded.

By default, the 'Maximum Number of Data Rows' setting is blank and will result in all of a given chart's data being returned up to the iDashboards system-defined limitations for pivoted/non-pivoted charts. If populated, however, the value can be any integer from 1 to 3,000 and will limit the number of rows returned to the chart accordingly. If the 'Maximum Number of Data Rows' setting exceeds the system-defined limitation for the number of rows that can be returned, the 'Maximum Number of Data Rows' value will be ignored (without error) and the system-defined limitation will instead be enforced.

8.5.2.20 Export All Chart Data

Note: Refer to the Administrative Manual for information on allowing end-users to have the option of exporting data. This option can be turned off by the Administrator Application.

Note: The default 'Maximum Chart Data Rows to Export' is 50,000 rows and can be adjusted by the iDashboards administrator within the iDashboards Administrator Application. See the iDashboards Administrator's Manual for details on adjusting these settings.

By default, charts within iDashboards have a 1,000-row limit, and pivot charts have a 3,000-row limit. The row limit of a chart is for reducing the size of the result set and the number of visual data points that will be displayed (there is no row limit on the data source).

Note: iDashboards does not have a limit on how much data can be queried in your data source.

Note: Administrative settings can increase the 1,000-row limit to 3,000 rows

If, however, it is necessary to obtain the entire data set of the chart, then this setting could allow the viewer to download the entire data set – unlimited rows. The option to 'Export Chart Data' is selected from the chart menu button within the View Interface. This setting will enable a dialog offering the following options:

- **Displayed** – This will export only the data that the chart is currently rendering.
- **All** – This will refer to the server setting (default = 50000) and will export the data until the server limitation has been reached.

8.5.2.21 Auto-Refresh Chart

The Refresh Interval option allows a dynamic chart to be refreshed automatically at the time interval specified (static charts by nature cannot be refreshed). Once enabled, the refresh time interval option can be set anywhere between 1-second and 9999 minutes. When the setting is disable, the chart will not automatically refresh.



Note: Take caution when using a small period of time as the refresh interval. If the chart query takes 5-seconds and the refresh interval is 2-seconds, it is possible the chart will never fully render on the screen.

8.5.2.22 Customized 'No Chart Data' Message

By default, the message that appears when a chart returns no data is "No data was found for this chart." This message can be customized by clicking 'Edit...'

The message can be customized to include more details about the specific chart that may help a user to understand why no data was displayed. Utilize Macros to better illustrate the issue and to make the message dynamic, i.e. based on Input Parameter selections.

8.5.3 Conditional Basic Settings

These settings appear within the "Basic Settings" only when configured in the previous stages of "Data Set" and "Axis List".

8.5.3.1 Chart Pivots

The available Pivot options will only show up in chart settings if the chart contains one or more pivots. If the chart doesn't contain a pivot(s), these options will not be visible.

- **Show Pivot Selectors** – This option is checked by default when designing a chart with Pivots. If unchecked, your pivots will still exist in the chart but they will not be visible for a user to select and change their value(s). An example situation would be wanting to hide the pivot selectors, so they don't take up screen real estate on your chart, along with the chart being a target chart for a drilldown. With the chart acting as a target chart for a drilldown, its pivot values can be selected even with the pivot selectors hidden. Another example is when using the Drilldown feature "Change Other Charts' Pivots"
- **Motion Pivot Enabled** – Only for certain chart types, currently Cluster, Stacked and Ratio Bar and Column charts. When enabled, along with Enable Animation (See 8.5.2.9 Animation), this animation automatically cycles through the pivot values causing the graphics of the chart to animate while maintaining the proper sorting of any numeric Y-Axis. By default, the first numeric Y-Axis will be automatically selected. When the sorting is on the X-Axis, there will be animation, but with no Y-Axis re-ordering. Not having any sorting is similar to using the X-Axis, except the data is the order it was retrieved. (See 8.5.2.17 Sorting Data Results). When Motion Pivot is not used the chart simply recreates without animation.

Note: Motion Pivot is not appropriate for all data sets. Use when desired visual performance is acceptable.

- **Last Pivot Control Type** – This option defaults to dropdown. The last Y-axis, which is designated as a pivot, will be shown as the control type selected. If only one pivot exists, for the chart, this controls that one pivot.
 - Dropdown (default) – Normal dropdown selection.
 - Button Row – A row of buttons, one for each value. When required buttons are provided for horizontal scrolling. A single click on the button will move the list by



one value, but will not change the select value unless it scrolls off the list. Using CTRL-Click will scroll one page of values at a time.

- Slider – A player-type control, with Play/Pause and Auto-Repeat buttons. The slider’s color theme is defined by the ‘Default Text Color’ settings, also under ‘Basic Setting’.
 - **Slider Play - Time Interval Between Changes** – For an animated Slider control, the amount of time, in seconds and milliseconds, between pivot value changes. Setting it to 0 causes continuous animation as it iterates through the pivot values.
 - **Autoplay When Chart Loads** – For an animated Slider control, when the chart loads begin to execute the slider play. While playing, and when stopped after playing, changes to other chart’s pivots will not occur.
 - **Auto-Repeat** – For an animated Slider control, automatically repeat the slider play.

Note: All pivot control types before the last pivot control are dropdown.

- **Pivot Location** – You can designate where the pivot selector(s) is placed; bottom of the chart (default), top of the chart or under the chart title.
- **Pivot Alignment** – For Dropdown control type only, you can designate how the pivot selector(s) is aligned; left (default), right or center.
- **Pivot Sort Order** – Sorting the pivot allows chart data to be sorted in a pre-defined order. The data can be sorted ascending, (a-z, 1-10) or descending (z-a, 10-1). Pivot data can be sorted by choosing the column in the “Pivot Sort Order” dropdown menu. The sort order is ascending by default.
- **Save Pivot Selections** – This option is unchecked by default. With it unchecked, every time you open or refresh a chart that contains a pivot(s), the pivot selectors will display the first pivot selection from the data source, no matter what values were selected the last time the chart was saved. If you check this option, every time you open or refresh a chart that contains a pivot(s), the pivot selectors will be set to the values that were selected the last time you saved the chart.
- **Opaque Pivot Bar** – Unchecked, the pivot bar background uses the same color and transparency as the chart background. If checked, the pivot bar will be 100% opaque, and will never have a transparency option. However, the pivot bar color will be the same as the chart background color.
- **Show Pivot Value Banner** – By default, this is not selected. If selected the last pivot’s value will be displayed. This can be useful when using the Slider control, since the selected pivot value is not always displayed.
 - **Pivot Value Banner Location** – You can designate where the pivot value is placed; above the chart (default), under the chart.



- **Pivot Value Banner Alignment** – You can designate how the pivot value is aligned; center (default), left or right.
- **Pivot Value Banner Size** – The default value’s text size is 20. It can be between 5 and 100, inclusive.
- **Pivot Value Banner Color** – The value’s text color can be controlled. The default is a medium dark shade of gray (it could be called ‘Eclipse’.)

8.5.3.2 Chart Parameters

- **Parameter Input Visibility** – By default, this option is set to ‘Show Parameter Input’ when designing a chart with Parameters. Other options include ‘Hide’ and ‘Minimize’.
- **Parameter Input Location** – You can designate where the parameter(s) is placed; top, bottom, left, right (default) or floating.
- **Parameter Legend Visibility** – Parameters can utilize a floating window to organize each parameter and value applied within the chart. This feature is sometimes helpful when the parameter display is hidden.
 - **Parameter Legend Settings** – The parameter legend can be visually controlled by adjusting the text color, the legend window color, and the legend window transparency settings.

8.5.4 Formats and Labels

Formats and Labels are available for all charts regardless of the chart type. Yet, based on the data set for the chart, not all sections may appear. These settings allow you to change chart features associated to the way numbers are displayed (\$, %, decimal places, etc.), the display format for dates, and unit labels for designating notes like “in millions”

8.5.4.1 Number Formatting

Since most charts have a numeric Y-Axis and many charts display numeric labels in the background of the chart-graphic, number formatting is a critical part of any chart design. Individually, an axis can be customized with a number format. However, editing the “Default Format” could also have a significant impact on the chart labels. Clicking on any “Edit” button will open a Number Format window. This window will be titled either “Chart Default” or titled specific with the axis names configured for the chart.

- **Separators** – You have the option of configuring the “Thousands Separator” and “Decimal Separator” fields. These values cannot remain blank. The value will likely be a “,” (comma) or “.” (decimal point).
- **Decimal Places** – Leaving this value blank will use a ‘floating decimal’ calculation, which automatically suppresses decimal places if the data set returns only whole numbers. Alternately, all decimal places will be displayed if the data set returns multiple decimal places. However, if a number value, like “1”, is entered, then the chart will force the display of a 1/10th decimal place. Therefore, the number entered will display the same number of decimal places.
- **Rounding** – You can set the “Decimal Places” and have the option of rounding up, down, nearest or not rounding at all.



- **Negative Indicator** – Charts can be configured to display the “Negative Indicator” as either ‘-’ or ‘()’.
- **Unit Symbol** – Adding a symbol to the “Unit Symbol” field will display the symbol in the chart’s mouse over pop-up bubble depending on the chart type, in the chart data.
- **Large Number Abbreviation** – Apply this to the “Default Format” for this property to be effective. For charts that display labels, this feature will abbreviate large numbers by removing zeros and adding the letter "K" (thousands), "M" (millions), "B" (billions), or "T" (trillions). Options for this feature include
 - **None:** No abbreviation will be applied
 - **Automatic:** "K", "M", "B", or "T" will be dynamically and automatically applied. If the chart data is ever filtered, pivoted or changes, it is possible to ‘step-up’ or ‘step-down’ in size. Additionally, a single chart might display a mixed set of values like “0.5M, 1.0B, 1.5B, 2.0B”.
 - **Thousands:** Will forcefully make the chart display thousands.
 - **Millions:** Will forcefully make the chart display millions.
 - **Billions:** Will forcefully make the chart display billions.
 - **Trillions:** Will forcefully make the chart display trillions.

8.5.4.2 Date Formatting

There are a variety of ways to display the value of a date. Date Formatting settings are available for charts that have an axis with the “Datetime” data type. These settings help you customize how a chart displays dates.

The following are various date formatting options and their translations:

Format	Display	Description
yyyy-MM-dd	2019-03-07	Year, 4-digits. Month and Day, 2-digits.
M/d/yy	3/7/19	Month and Day, 1 or 2-digits. Year, 2-digits.
MM/dd/yy	03/07/19	Month, Day, and Year, 2-digits.
MM/dd/yyyy	03/07/2019	Month and Day, 2-digits. Year, 4-digits.
E M/d/yy	Thu 3/7/19	Day of Week, 3-letter abbreviation. Month and Day, 1 or 2-digits. Year, 2-digits.
E M/d/yyyy	Thu 3/7/2019	Day of Week, 3-letter abbreviation. Month and Day, 1 or 2-digits. Year, 4-digits.
MM-yyyy	03-2019	Month, 2-digits. Year, 4-digits.
MMM-yy	Mar-19	Month, 3-letter abbreviation. Year, 2-digits.
M/d	3/7	Month and Day, 1 or 2-digits.
MM/dd	03/07	Month and Day, 2-digits.
MMM d	Mar 7	Month, 3-letter abbreviation. Day, 1 or 2-digits.
E M/d	Thu 3/7	Day of Week, 3-letter abbreviation. Month and Day, 1 or 2-digits.
E MM/dd	Thu 03/07	Day of Week, 3-letter abbreviation. Month and Day, 2-digits.
M/d/yy H:mm:ss	3/7/19 15:09:07	Month and Day, 1-digit. Year, 2-digits. Hour, 1 or 2-digits. Minute and Second, 2-digits.



Format	Display	Description
M/d/yy h:mm:ss a	3/7/19 3:09:07 PM	Month and Day, 1 or 2-digits. Year, 2-digits. Hour, 1 or 2-digits. Minute and Second, 2-digits. AM or PM marker.
MM/dd/yyyy HH:mm:ss	03/07/2019 15:09:07	Month and Day, 2-digits. Year, 4-digits. And Hour, Minute, and Second, 2-digits.
yyyy	2019	Year, 4 digits.
MMMM	March	Month, full name spelled out.
MMM	Mar	Month, 3-letter abbreviation.
MM	03	Month, 2-digits.
E	Thu	Day of Week, 3-letter abbreviation.
H:mm	3:09	Hour, 1 or 2-digits and Minute, 2-digits.
HH:mm	15:09	Hour, 2-digits and Minute, 2-digits.
HH:mm:ss	15:09:07	Hour, Minute, and Second, 2-digits.
h:mm:ss a	3:09:07 PM	Hour 1 or 2-digits. Minute, and Second, 2-digits. AM or PM marker.
&&	1440-07-01	Conversion to Hijri: Date.
&&&&	1440-07-01 15:09:07	Conversion to Hijri: Date and Time.

Note: When previewing the chart data via 'View Data', all date data will be displayed in the default system date format. However, when the chart is viewed in the Preview window or on the dashboard, the date format will reflect what is specified in 'Chart Properties | Formats and Labels | Date Formatting'.

Note: The '&&' and '&&&&' date formats convert dates stored in Gregorian format to Hijri using the Kuwaiti algorithm.

The "Default Date Format" will be used for all axes unless a specific axis has a format defined. This default format is 'M/d/yy' and will display the date in numeric form only.

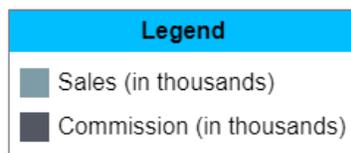
Users can select the arrow button that appears next to the date formatting text box to view the list. Selecting a format from the list will populate the text box with pre-defined date formats.

If the date format for the "Default Date Format" is changed to 'MMM dd, yyyy', then all axes date values will be displayed in the format "Jan 01, 2020". If no format is defined for any axis (including the default format) the system date format will be used.

8.5.5 Unit Labels

Unit Labels are available for all charts regardless of the chart type. These settings allow you to add additional text to a chart's legend. Leaving the Unit Labels blank will force the legend to only show its Y-axis labels.

For example, adding text "in thousands" to the Unit Labels will append that text to the end of its Y-axis labels and display as shown below.



8.5.6 Colors

Setting colors is an integral part of designing attractive and effective dashboards. Color properties are available for all charts regardless of the chart type. Yet, based on the selected chart type, varying sections may appear. These settings allow you to change common chart color settings.

Refer to section 4.3, “Colors” for instructions on using colors and color palettes.

8.5.6.1 Colors: Basic Settings

When looking at the “Basic Settings” within the Common Chart Settings, there will be a variety of color settings to change the color of common chart elements. Based on the layering principles, a color setting might also include a transparency value. These settings are available in all charts regardless of their type. The features are as follows:

Setting	Description	Default
Background Color	The chart’s background.	White (FFFFFF)
Background Transparency	The chart’s background transparency level. Often used when an image is used as a dashboard background. (0 = no transparency & 100 = full transparency)	Value = 100
Default Text Color	The chart’s title and any text color that is not explicitly configurable.	Black (000000)
Legend Text Color	The legend text color, displaying axis labels and values.	Black (000000)
Legend Background Color	The legend window background color.	White (FFFFFF)
Legend Transparency	The legend window transparency level. (0 = no transparency & 100 = full transparency)	Value = 100
Parameter Legend Text Color	The parameter legend text color, displaying axis labels and values.	Black (000000)
Parameter Legend Background Color	The parameter legend window background color.	White (FFFFFF)
Parameter Legend Transparency	The parameter legend window transparency level. (0 = no transparency & 100 = full transparency)	Value = 100
Note Background Color	The note text color, displaying user-defined information.	Tan (EEEEAA)

8.5.6.2 X Value Colors

Note: “X Value Colors” properties only display with certain chart types. This feature is available in all of the ratio charts (pie, pyramid, funnel) and the bubble chart.

Charts supporting X Value Colors will have pre-defined colors automatically applied. To change the default colors, select the option ‘Select Custom Colors’. Customizing the X Value color is beneficial to assign a specific color, to a specific data value, and then keep that value-color



combination consistent within other charts on the same dashboard or the same colors across different pivot selections!

On bubble charts, there is one bubble per non-unique X value, and the color of each bubble is its corresponding X value color. On a pie chart in sum down mode, the X value colors are used for pie slices, each of which represents a non-unique X value. Note that there can be two distinct slices for a repeating value on the same pie, and both will have the same color.

The colors selected by the user from the “Select Custom Colors” option will be persisted and used for those X values the next time the chart is loaded. If the dataset contains new X values on the next chart load, for which there are no saved colors, they’ll be given system assigned colors that will be saved as user-selected colors the next time the chart is saved.

The X value colors will display in the legend along with the date or number formatted X values.

Note: The system may duplicate colors already in use by other X values. Users should utilize System-assigned colors whenever the set of X values is likely to change (as is often the case with Date-based X values).

8.5.6.3 Colors: Axis Colors

When looking at the “Colors” settings within the Common Chart Settings, there are specific color settings based on the chart type and the axes within the chart. By default, initial colors will automatically be applied.

These features are available in those charts that display data for each Y variable (examples include bar, column and line charts).

8.5.6.3.1 Chart Axis Colors using Shared Color Palette

A user can customize the axis colors one-at-a-time. Alternatively, shared palette colors can be applied to each axis. To use palette colors, select the name of the color palette from the list and then click ‘Apply’.

Note: Colors are applied in the exact order as their appearance in the shared palette. The first color of the selected palette will be applied to the first axis, the second color will be applied to the second axis, etc.

For example, “U.S.A.” begins with Red, White, and then Blue. When applied to the chart with 2-axes, Red and White will be used.

8.5.6.4 Range Sets

Note: Not all charts support Range Colors.

Note: Not all charts that support Range Colors are enabled to display the range colors.

Charts supporting Range Colors will have calculated colors automatically applied. This means, the colors, the number of color increments, the color thresholds and the color labels, will be computer generated (assigned by evaluating the highest data value the chart returns and dividing it in to equal parts from zero to that value). Range sets are colors allow you to configure a chart so that various portions of the chart will change color based on the value(s) the chart is



pulling from its data source. To change the default settings, select the option 'Manually Define Colors'.

Color	Lower	Upper	Label
#FF0000	0	2000	0 - 2000
#FFFF00	2000	4000	2000 - 4000
#00FF00	4000	6000	4000 - 6000
#00FFFF	6000	8000	6000 - 8000
#0000FF	8000		8000 and above

The first consideration when customizing the range colors is whether or not to apply the same changes to one, or all, of the axes. For example, if scores for TEST-A range from 700 to 1200 are within the same chart as the scores from TEST-B having a range from 10 to 40, then the range colors probably should not share the same color thresholds and the option to customize each axis independently should be used. Using the dropdown shown at the top of the image above, choose the option that will deliver the desired results.

Once the Axis has been decided, there are two methods for defining a new range set:

1. **Manually configure** – Using the 'trash' and 'Add' buttons, an entirely new color set can be defined. To assist with the labels, try enabling 'Auto Label' to automatically generate labels based on each number threshold.
2. **Auto Ranges** – Using the 'Auto Ranges' button, a multitude of colors can be automatically created by defining the following items. Prior to leaving the 'Auto Ranges' dialog, click 'Preview' to see the colors applied within the main dialog. When using this option, the Labels will be automatically generated based on each number threshold. These can be overridden afterwards if desired.
 - a. A Lower and Upper Boundary number.
 - b. A Range Size (the incremental value between the lower and upper values)
 - c. A Progression Type of: Spectrum, Reverse-Spectrum or Blend
 - d. A Beginning and Ending color

Within the Range Set Colors, a user can customize the range colors one-at-a-time. However, shared palette colors can be automatically applied to the range colors.

Using the dropdown shown at the bottom of the image above, select the palette containing the desired colors and click 'Apply'.

Note: Colors are applied in the exact order as their defined order within the palette. The first color of the selected palette will be applied to the first range, the second color will be applied to the second range, etc.

8.5.7 Drilldown

The drilldown functionality of iDashboards allows a user to click on a chart and have that click-action perform one of the following functions:

- **Display a different chart** – Uses the concept of a Source chart and Target chart.
- **Display a different dashboard** – Uses the concept of a Source chart and Target dashboard.
- **Display a web page** – Uses the concept of a Source chart and Target window.
- **Change other chart's pivot** – Uses the concept of a Source chart and Target charts constructed with a matching pivot axis.
- **Change dashboard parameter** – Uses the concept of a Source chart and Target dashboard parameters.

When the user clicks on the chart (called the “**source chart**”), the value they clicked on can be passed as a parameter during the drilldown so the “**target chart**” (or “target dashboard”, “target web page”, etc.) can consume the value. In the scenario where a source chart drills down to a dashboard, all charts within that dashboard act as target charts.

The value being passed can be the value from the X-axis, Y-axis or combination of both values from the metric that was clicked on. Also, any pivot values that exist and are selected in the source chart will be passed as parameters to the target chart. This combination of drilling down and passing parameters is typically implemented when you would like to see more detailed information about the data point you clicked on displayed in the target chart, dashboard or web page. Within the View Interface, you will be able to simply navigate through drilldown paths already created within dashboards.

A hierarchy of multiple levels of drilldown may be created. For example, the top-level chart may have top level metrics, which may drill down to secondary level metrics, which may drill down to item level metrics, which may drill down to SKU level metrics. During each drilldown, the metric you clicked will be passed to the next chart to be filtered.

Note: As you place your cursor over a chart and see the cursor icon as hand shaped, it is indicative of the chart having a drilldown.

8.5.7.1 Drilldown Parameters

iDashboards offer a technique for passing values from a source chart into a Target chart/dashboard. When clicking on a chart to perform a drilldown, certain information is passed to the drilldown target. This information includes the axis label and clicked-on axis value for the



axes that were selected to be sent down during drilldown. If the drilldown target is a chart, the chart can use the axes values passed to it to auto-select its pivots or to populate a macro in the chart title or even get consumed into the target chart parameters. If the drilldown target is a URL, the axes label/value pairs can be included in the URL so that the destination web page can use them.

If you are performing drilldowns that cascade down through multiple charts, it is important to note that the information passed during each drilldown is stored and maintained throughout all the drilldowns. Every time a chart is clicked on to perform a drilldown, the selected axes label/value pairs are stored so that they may be used by subsequent drilldown targets. Even if the drilldown target doesn't use one of the axes label/value pairs, they are still recognized and stored by the drilldown target. Therefore, when a new drilldown is performed, the new drilldown target can recognize and use the axes label/value pair that was previously ignored.

8.5.7.1.1 Traditional Conceptual Example

For example, in the example below, the drilldown sequence would be STATE > VEHICLE > DETAILS. If you click on the row for Michigan in the "State" chart (source), the X-axis label 'State' and the X-axis value 'Michigan' will be sent as drilldown parameters to the "Vehicle" chart (target). The "Vehicle" chart first verifies a matching column name (valid). Next the "Vehicle" chart will use the value "Michigan" to filter the data. Continuing the drilldown process, within the "Vehicle" chart (now the source chart) you now click on "Car". Lastly, the "Details" chart (target) will verify both matching columns ("State" and "Vehicle") and then filter the "Details" chart based on both columns, "State" and "Vehicle" and the related values "Michigan" and "Car".

State		Vehicle		
State	Amount	State	Vehicle	Amount
Florida	\$1,381,009.44	Michigan	Truck	\$499,075.79
Texas	\$1,268,909.02	Michigan	Motorcycle	\$102,852.01
California	\$932,640.41	Michigan	Car	\$76,474.61
Michigan	\$678,402.41			

Details				
State	Vehicle	Driver	Claim	Amount
Michigan	Car	Male	#41932	\$43,855.66
Michigan	Car	Female	#64579	\$32,618.95

8.5.7.1.2 Alternate Conceptual Example

In a similar example below, the drilldown sequence would also be STATE > VEHICLE > DETAILS. This time, the unconventional difference is how the "Vehicle" chart does not contain the column "State". If you click on the row for Michigan in the "State" chart (source), the X-axis label 'State' and the X-axis value 'Michigan' will be sent as drilldown parameters to the "Vehicle" chart (target). The "Vehicle" chart will accept and store this information even though there is no matching column name for "State". Continuing the drilldown process, within the "Vehicle" chart



(now the source chart) you now click on “Car”. Lastly, the “Details” chart (target) will verify both matching columns (“State” and “Vehicle”) and then filter the “Details” chart based on both columns and the values “Michigan” and “Car”.

State		Vehicle	
State	Amount	Vehicle	Amount
Florida	\$1,381,009.44	Truck	\$1,866,485.04
Texas	\$1,268,909.02	Car	\$1,530,829.18
California	\$932,640.41	Motorcycle	\$863,647.06
Michigan	\$678,402.41		

Details				
State	Vehicle	Driver	Claim	Amount
Michigan	Car	Male	#41932	\$43,855.66
Michigan	Car	Female	#64579	\$32,618.95

8.5.7.1.3 The Value Macro `#{value:<Axis Name>}`

`#{value:<Axis Name>}` is a macro that will return the value of the axis named “<Axis Name>”. The axis name is case sensitive. Often times, the dashboard builder will place the value macro into the title of the Target chart.

The example below illustrates the outcome of using the value macro. The Source chart, on the left, has an X-axis of “State”. The Target chart, on the right, has a pivot Y-Axis of “State”.

Using the value macro in the title of the Target chart would look like this:

Vehicle Claims in `#{value:State}`

State	
State	Amount
Florida	\$1,381,009.44
Texas	\$1,268,909.02
California	\$932,640.41
Michigan	\$678,402.41

Vehicle Claims in Michigan	
Vehicle	Amount
Truck	\$499,075.79
Motorcycle	\$102,852.01
Car	\$76,474.61

Michigan

When the drilldown occurs in this example, iDashboards will use the value “Michigan” and place it into the title wherever the dashboard builder wanted the state name to appear. The value will only ‘expand’ when the chart is viewed after a drilldown, not during the chart building process.



8.5.7.1.4 Passing the Format of Dates or Numbers

iDashboards will always translate a simple date into a complex date during the chart building process. During the drilldown process, the clicked-on date can be passed into the target chart. As the complex date value is passed, the source-chart date formatting will also be passed along. This is important when the dashboard builder chooses to use the date 'value macro' within the target chart.

Therefore, if the Source chart is formatting dates like this: "MMM-yy", then the value will be passed into the Target chart like this: "MMM-yy".

Number formatting will also pass down to the Target chart. Therefore, if the Source chart has number formatting like this: "\$12,345.00", then the value will be passed into the Target chart like this: "\$12,345.00".

8.5.7.2 Drilldown to Chart

By default, new charts are not configured with a drilldown option. To configure a chart-to-chart drilldown, open the Chart Designer, then navigate to the Chart Properties and expand the "Drilldown" menu prior to selecting the "Drilldown to Chart" button.

The first step is to select the target chart. Click on the "Select..." button to open the Select Chart window which will show all Categories you have access to. In this window, select the Category and the specific Chart you want linked as the target chart. When the source chart is clicked, the drilldown can have the target chart display in the same dashboard frame as the source chart or in a different frame within the dashboard. To select which frame to have the target chart displayed in, select the desired frame "tag name" in the "Target Dashboard Frame" dropdown box. To save this drilldown link, save the chart.

8.5.7.2.1 Selecting the Axis Value for Target Chart Filtering

iDashboards will allow users to select any of a chart's axes to be used for drilldown filtering; this includes hidden axes. The default options will suggest using the X-axis, all pivoted Y-axes, and sending the input parameters to the target chart – all for the purpose of drilldown filtering. However, any the default options can all be changed to customize the drilldown behavior.

If the target chart does not contain the filtered value that was selected from the source chart, then the target chart is presented to the user without any data filtering.

8.5.7.2.2 Always send value for clicked-on axis

When drilling down to a chart, in addition to being able to select the axes to be used for drilldown filtering, a designer can also check a box labeled "Always send value for clicked-on axis." When this is checked, the clicked-on axis will be used for drilldown filtering, in addition to any axes which are specifically selected for drilldown filtering. In some cases, the clicked-on axis might be the only one used for filtering.

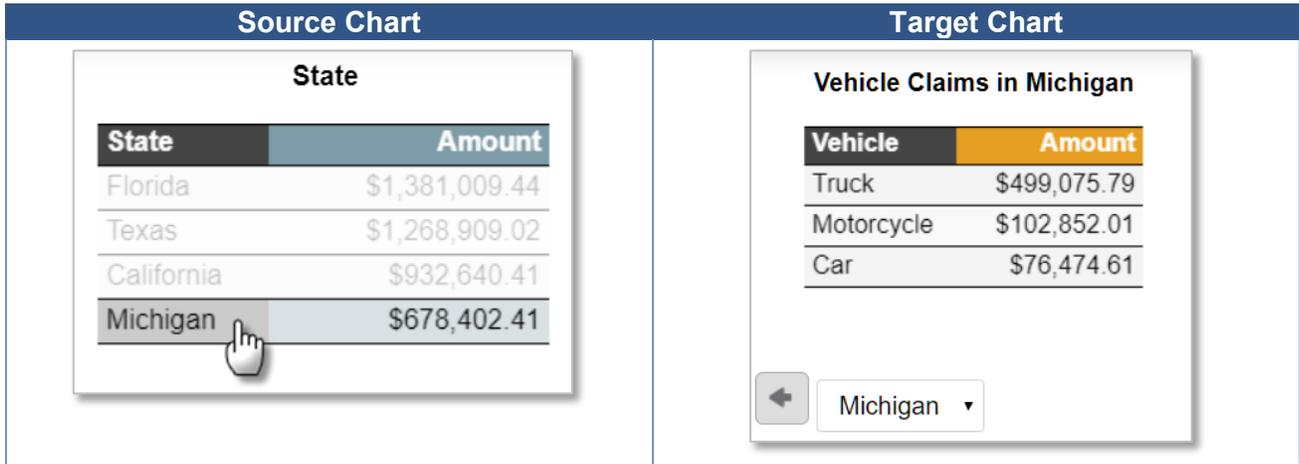
8.5.7.2.3 X-Axis Value Example

A common use for using drilldowns is to pass along a value from the X-axis that will help filter the data in the target chart. The images below show examples of a source chart and target chart respectively. When a user clicks on "Michigan" in the source chart, the target chart will be displayed and will automatically filter its data on "Michigan".

Note: The value macro was used in the chart title on the target chart.



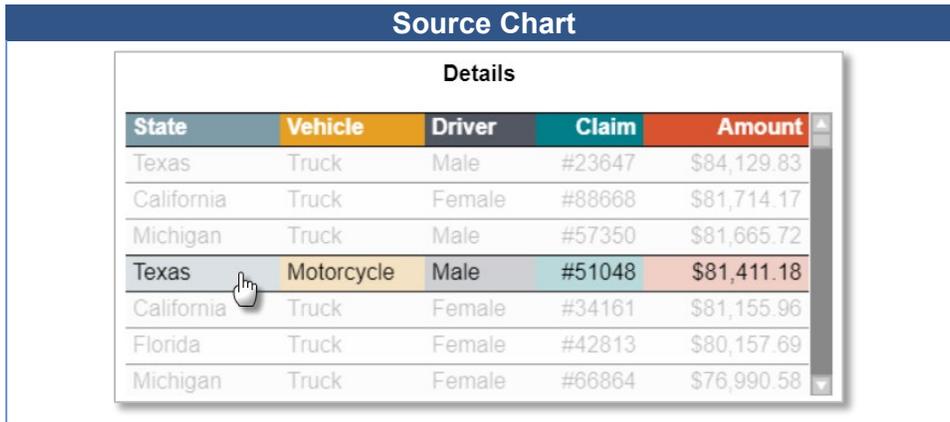
Note: The “Drill Back” button on the target chart allows the user to drill back up the drilldown path.



8.5.7.2.4 Y-Axis Value Example

Another common use for using drilldowns is to pass along a value from the Y-axis that will help filter the data in the target chart. The example below shows a source chart and target chart respectively. When a user clicks on the row that contains a Y-axis value of “Motorcycle” in the source chart, the target chart will be displayed and will automatically filter its data on “Motorcycle” to show just his sales figures. It is even possible to drill into the same chart barring any row limitations.

Note: The “Drill Back” button on the target chart allows the user to drill back up the drilldown path.



Target Chart				
Motorcycle Details				
State	Vehicle	Driver	Claim	Amount
Texas	Motorcycle	Male	#51048	\$81,411.18
Florida	Motorcycle	Female	#11071	\$68,307.62
Texas	Motorcycle	Male	#70284	\$64,398.29
Florida	Motorcycle	Male	#19483	\$62,985.62
Florida	Motorcycle	Male	#49772	\$62,855.20
Florida	Motorcycle	Male	#64000	\$64,000.40

8.5.7.3 Drilldown to Chart or Dashboard Report

There are two drilldown options for reporting: “To Chart Report” and “To Dashboard Report”. Either selection will function similar to the core feature of a drilldown, meaning, the chart can be clicked on and a value *can* be used to filter whatever type of report is seen afterward (filtered reports are optional). The report that is seen afterward uses report properties that were configured with the target dashboard or target chart. The PDF report will be generated in a new browser tab.

8.5.7.4 Drilldown to Dashboard

By default, new charts are not configured with a drilldown option. To configure a chart-to-dashboard drilldown, open the Chart Designer, then navigate to the Chart Properties and expand the “Drilldown” menu prior to selecting the “Drilldown to Dashboard” button.

The first step is to select the target dashboard. Click on the “Select...” button to open the Select Chart window which will show all Categories you have access to. In this window, select the Category and the specific Dashboard you want linked as the target dashboard. When the source chart is clicked, the target dashboard will pop up and get displayed over the dashboard that contains the source chart you clicked on. To save this drilldown link, save the chart.

The recommended approach to creating effective drilldowns to dashboards is to have the target dashboard with charts having Pivots with Pivot labels identical to the X-label of the drill origination chart. iDashboards will automatically filter the Pivot columns in each chart of the target dashboard. This way, the entire target dashboard delivers filtered values within its contained charts. If a chart within the target dashboard does not contain a label matching the drill origination chart, that chart will display in the dashboard with unfiltered data.

The ability to filter the target dashboard based on X or Y-axis values are available, just as it is with drilldowns to charts. The ‘Always send value for clicked-on axis’ option is as well.

8.5.7.5 Drilldown to Web Page

By default, new charts are not configured with a drilldown option. To configure a chart-to-webpage drilldown, open the Chart Designer, then navigate to the Chart Properties and expand the “Drilldown” menu prior to selecting the “Drilldown to Webpage” button.

It can be very useful to create a drilldown link out to a web page. You may simply want the user to link out to a web page for information purposes, or, if you have development control of the



target web page, you can have that target web page perform a filter based on the data point(s) the user clicks on in the source chart. A drilldown link to a web-based reporting tool is a good example.

In the 'URL' field, provide the complete web site address to which the chart should be linked. Clicking on the chart will launch the linked web site using your default browser. To save this drilldown link, save the chart.

Note: If a user has Pop-up Blocker activated within their browser, then the linked web site won't be able to open on drilldown.

When setting up a URL as the destination for a chart drilldown, the drilldown can be configured to send iDashboards macros to the target URL. These macros should be added to the URL.

The following macros (in the table below) can be used to configure a drilldown URL within the 'Drilldown' tab. Clicking on the macro button will display all available macros. Here, all axes from the chart are displayed for easy selection. A value macro, like `#{value:<Axis Name>}`, can also be used with the URL string.

Macro	Expands to:
<code>#{user}</code>	Username value of user currently logged in user.
<code>#{value:<Axis Name>}</code>	The value of the chosen <Axis Name>, which may be the X-axis or any Y-axis that is a pivot.
<code>#{uevalue:<Axis Name>}</code>	Use 'ue' (unescaped) option when axis value is a URL.
<code>#{chartid}</code>	chartID value.
<code>#{charttitle}</code>	chartTitle value.
<code>#{categoryid}</code>	categoryID value.

8.5.7.5.1 Passing the Value Macro (Non-URL)

State	Vehicle	Driver	Claim	Amount
Texas	Truck	Male	#23647	\$84,129.83
California	Truck	Female	#88668	\$81,714.17
Michigan	Truck	Male	#57350	\$81,665.72
Texas	Motorcycle	Male	#51048	\$81,411.18
California	Truck	Female	#34161	\$81,155.96
Florida	Truck	Female	#42813	\$80,157.69
Michigan	Truck	Female	#66864	\$76,990.58

For example, if the chart above has been configured to drill down into a web page and the 'URL:' field contains:

`http://www.mywebpage.com/?state=#{value:State}&claim=#{value:Claim}`



(Replace with actual URL)

When the chart is clicked and the drilldown is initiated, the URL will be transformed into:

```
http://www.mywebpage.com/?state=Texas&claim=51048
```

Note the replacement of “`#{value:State}`” with “Texas” and “`#{value:Claim}`” with “51048”

Note: The values that get passed down can be any combination of X and Y axes within the chart. Any spaces in the value will be replaced with '%20' and colons with '%3A'.

8.5.7.5.2 Passing the Value Macro (When Axis is URL)

If the drilldown axis contains a URL with a prefix of ‘http://’ or ‘www’, then the `#{uevalue...}` macro must be used. The ‘ue’ stands for ‘unescaped’, which will help circumvent browsers from substituting special characters when launching a new browser or browser tab. It is only necessary to use the ‘ue’ when the data column contains the URL.

For each `#{value:<AxisName>}` macro, there is a corresponding `#{uevalue:<AxisName>}` macro. For example, if the chart below has been configured to drill down into a web page and the ‘URL:’ field contains:

`#{uevalue:Website}`

Details	
State	Website
Texas	https://www.txdot.gov/
California	http://www.caltrans.ca.gov/
Michigan	https://www.michigan.gov/mdot/
Florida	https://www.fdot.gov/

Because the unescaped value was used, the chart will drill down to a new web page without transforming special characters within the browser.

```
https://www.michigan.gov/mdot/
```

8.5.7.5.3 Windows Settings & Target Window

There are several ‘Window settings’ and ‘Link target window’ options that can be set when launching the separate browser. These are any standard HTML modifiers that can be assigned to a browser window. These modifiers take the form:

`<settingname>=<setting value>`

Modifiers are comma separated. The table below shows a list of modifiers available.



Modifier	Value	Browser Specific	Description
width	number in pixels		Use this to define the width of the new window.
height	number in pixels		Use this to define the height of the new window.
resizable	yes or no	Possibly	Use this to control whether or not you want the user to be able to resize the window.
scrollbars	yes or no	Possibly	This lets you decide whether or not to have scrollbars on the window.
toolbar	yes or no	Possibly	Whether or not the new window should have the browser navigation bar at the top (The back, forward, stop buttons etc.).
location	yes or no	Possibly	Whether or not you wish to show the location box with the current URL (The place to type http://address).
directories	yes or no		Whether or not the window should show the extra buttons (personal buttons, etc...).
status	yes or no		Whether or not to show the window status bar at the bottom of the window.
menubar	yes or no		Whether or not to show the menus at the top of the window (File, Edit, etc...).
copyhistory	yes or no		Whether or not to copy the old browser window's history list to the new window.
left	number in pixels		Sets the position of the window in pixels from the left of the screen.
top	number in pixels		Sets the position of the window in pixels from the top of the screen.

An example of a “Windows Settings” option is as follows:

```
toolbar=no,width=1008,height=655,status=no,menubar=no,location=no
```

The ‘Target Window’ allows for an identification of the launched browser window so that the same browser window will be used every time a web page is linked. In order for this to work, enter a word or set of characters in the field. Every time the chart is clicked the URL will be launched in the same browser, not multiple browsers. Any word or set of characters can be used in the ‘Target Window’ field, however spaces are not allowed. If this field is left blank, each click on the same drilldown chart will launch a new browser displaying the target URL (see below).

Drilldown to Webpage

URL

Link URL Settings

Target Window



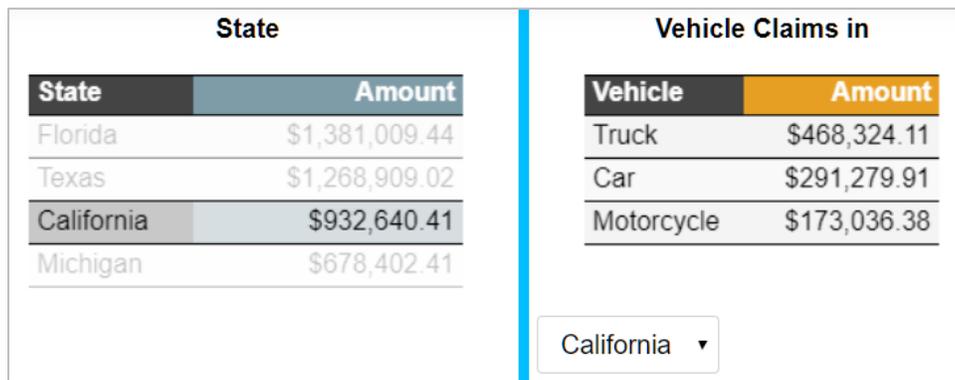
8.5.7.6 Changing Other Charts' Pivots

For this feature to function there must be other charts on the dashboard using pivots.

Note: Pivot charts do not filter data. Changing Pivots cannot alter the title of a chart.

To configure a chart to change the pivot value of another chart, open the Chart Designer, then navigate to the Chart Properties and expand the “Drilldown” menu prior to selecting the “Change Other Chart’s Pivot” button.

This option will allow the user to click on a value in the source chart and have the pivots in adjacent charts, on that same dashboard, update to reflect the axis value that was clicked on. Below, when “California” is clicked on in the source chart, the other charts in the dashboard update to have “California” selected in their pivots. To configure this feature, the name of the axis on the Source chart (for example, “State”) must match the name of the pivot value on the Target chart (for example, “State”).



8.5.7.6.1 Use name and value of clicked-on axis

If this option is checked, the clicked-on axis will be sent during the drilldown instead of the axis selected via the radio buttons.

8.5.7.7 Dashboard Parameters

For this feature to function there must be other charts on the dashboard using input parameters. There must also be a dashboard input parameter.

Note: Input Parameter charts can filter data. Changing input parameter values can alter the title of a chart if the parameter macro is used.

This drilldown option lets a user click on a chart and automatically change the dashboard parameters that correspond to the selected axes names. In order for the dashboard parameter to be modified with a given axis value, the parameter display name must match the axis name and the value must be an acceptable value for the parameter. Only the values of the selected axes, or the clicked-on axis if the checkbox below is selected, will be applied to the dashboard parameters.

8.5.7.8 Setup Example

Item	Value
Source Chart	X-Axis = State



	Y-Axis = Amount
	Drilldown Type = Change Dashboard Parameters X-Axis = <checked>
Dashboard	Input Parameter Name = theState Input Parameter Label = State
Target Chart(s)	Input Parameter Name = theState

8.5.7.9 Removing Drilldown

To remove the drilldown, navigate to the “Chart Designer > Chart Properties”, and select the option “No Drilldown” from the menu.

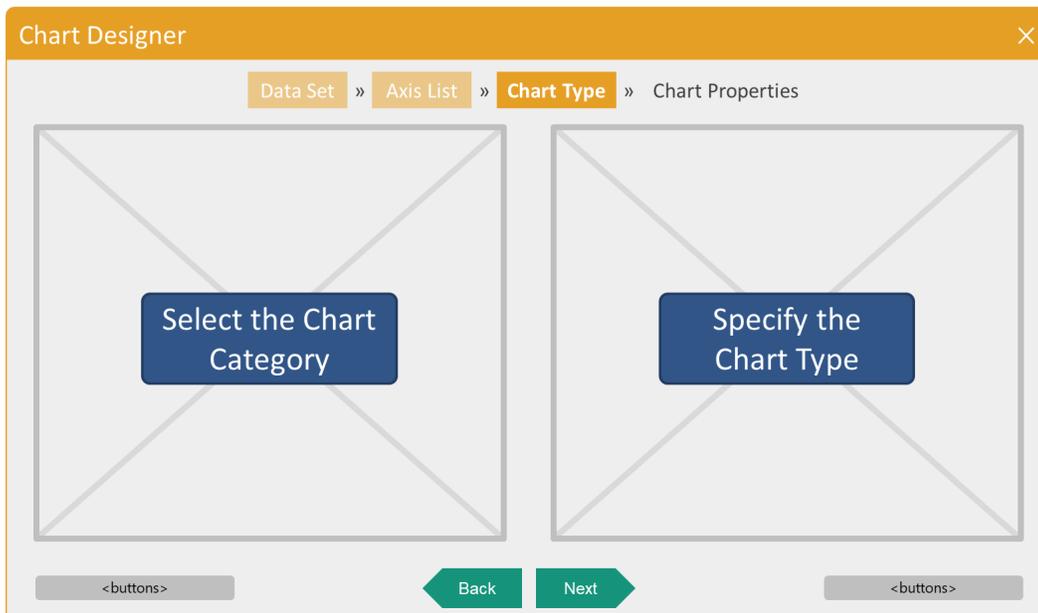


9. Chart Types

When building new charts, users will automatically progress to this option. However, if the chart already exists on the screen, users can continue to make changes to the Chart Type.

To configure a chart type, open the Chart Designer, then navigate to the Chart Type. To select a chart, first browse the chart category on the left and then select the actual chart type on the right.

Note: every chart requires a minimum of two axes.

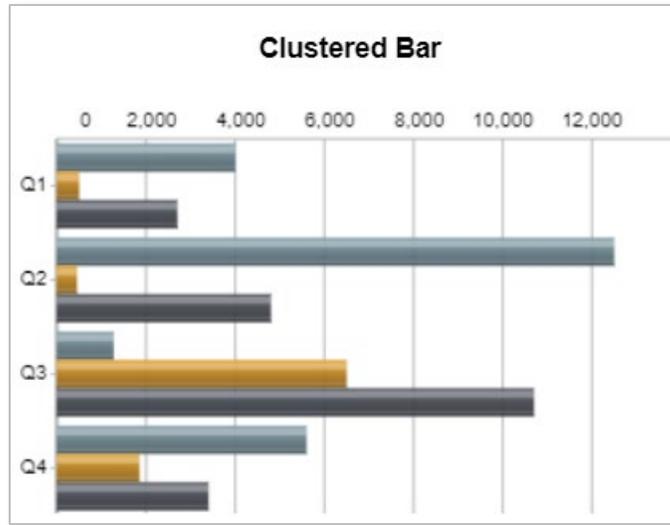


Based on the Axis List definition, some charts will immediately be unavailable for selection, as shown by the strikethrough within the name of the chart. If the invalid chart type is selected, a notice with additional details will appear.

Note: Sometimes, the chart is available to select, but until the data is rendered in the chart it is too early to determine if the chart will provide the desired results. Refer to the “Preview” stage for experiencing a live preview of the chart and data.

9.1 Bar Charts

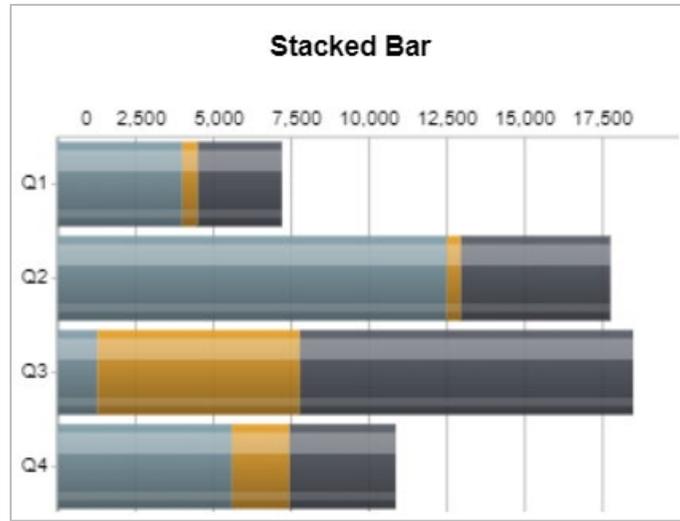
- **Clustered Bar** – Each Y-variable is drawn as a colored horizontal bar. Compares Y-values across multiple Y-variables (see image below).



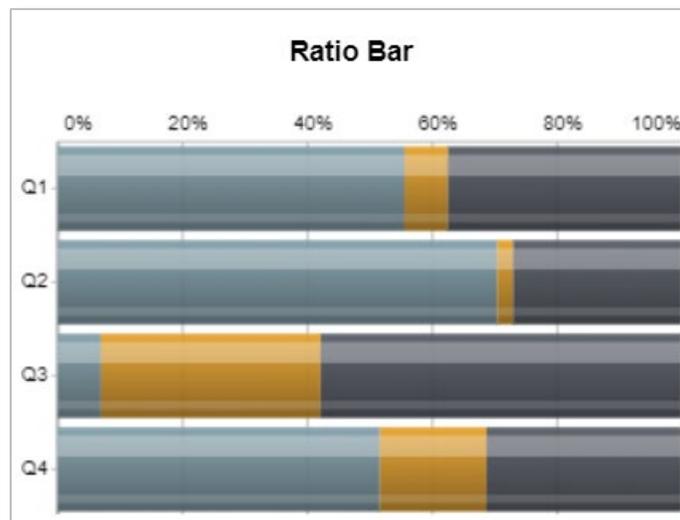
Note: The 'Show Bar Gradient' option is available on the Clustered Bar Chart. This option allows for the bars to display with a variety of gradient options.

Note: The 'Use Mixed Y-Axis Ranges' option is available on the Clustered Column (and Bar) Chart, 3D Column and 3D Cylinder charts. This option allows for the graph to plot the values in their proportions. For example, the values 100, 200, and 300 would plot the same as 10, 20, and 30 since they have the same proportions. The 'Use Mixed Y-Axis Ranges' option on the Clustered Bar Chart will work only if three conditions are met:

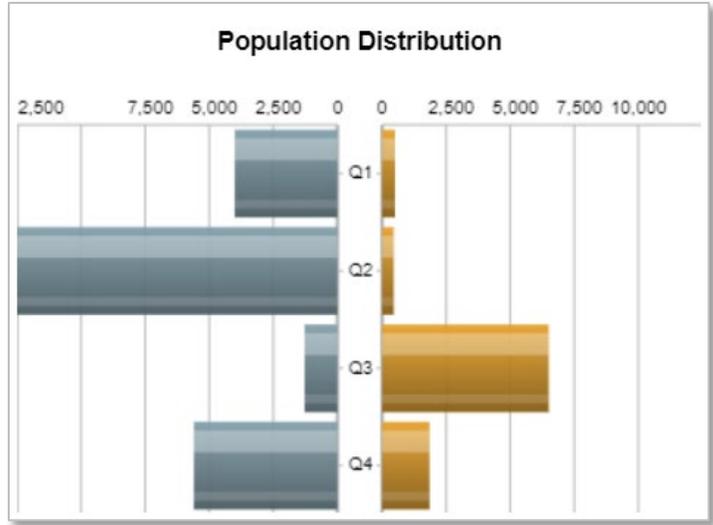
1. *The chart has more than one numeric Y value.*
 2. *The 'Use Mixed Y-Axis Ranges' option is checked.*
 3. *The 'Match Type' option in the basic setting is set to "Y Axis Name".*
- **Stacked Bar** – Each Y-variable is drawn as a colored stack on a multicolor horizontal bar. Compares the contribution of each Y-variable to a total across multiple Y-variables (see image below).



- **Ratio Bar** – Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a stacked horizontal column (see image below).



- **Population Distribution** – Displays distribution of various groups in a population, with horizontal bars comparing two numeric Y axes against a common X axis (see image below).



The first numeric Y axis found will be displayed on the left; the second numeric Y axis found will be displayed on the right. Unlike some other charts, this chart will not graphically display a third (or more) numeric axis. Instead the additional axes will only display in the mouse-over legend.

9.1.1 Gantt Chart

Graphically illustrates the start and end dates of tasks within a project.



Gantt charts are for project scheduling. Project details, like tasks or phases, can be visually interpreted to determine the expected/actual start and finish dates.

This chart type is divided vertically into two components. The left half displays tabular data and the right half displays horizontal, graphical bars along a timeline. Each X-axis value is represented as a row (task) and the horizontal bar will illustrate length of time between the start and finish date for each task.

Using two Y-Axes: If Y1 represents the expected start date for a task then Y2 should represent the expected end date. If an additional Y3 & Y4 are added to communicate actual start and end dates, then each task will display two horizontal bars. The pattern of 'pairing dates' can continue

an unlimited number of times. However, the first pair is specifically designated as the expected period of time.

By default, a vertical line appears at the 'Current Date' under the 'Data Date Line' settings. If there is an unpaired Y-axis containing dates, the vertical line can appear in a different location. The data structure to support this axis is non-typical and doesn't follow the traditional tabular-data format. iDashboards will locate the first date (querying from the top-down) to position the date line. Therefore, it may be necessary to organize data in a particular format for this column. If the Data Date Line is not unpaired, then a specific label name can be used to leverage this feature. The Axis Label must begin with the word "data" followed by an optional: space, hyphen, or underscore and followed by the word "date" (case-insensitive), and lastly followed by optional characters. Ex: "DATA_DATE", "datadate", "Data-date For admin". The label must also be hidden during the setup of 'Chart Data Column'.

Milestones exist when the start date and finish date are identical.

If a data column contains additional data which groups the tasks together, the chart builder has multiple options to summarize and group tasks. Grouped tasks will display a horizontal summary bar with visual arrows at each end.

The color of each horizontal bar can be adjusted by customizing the start date axes. For example, Y1 (expected start) and Y2 (expected end) are paired dates which display as one horizontal bar. Editing the start date (Y1 axis) color will change the color of that bar. If the summary bars are enabled, the colors can be adjusted by changing the end date (Y2 axis).

9.1.1.1 Gantt Data Requirements

- Like all iDashboards charts, one X and one Y axis are required. However, two Y Axes, as dates, are required for the Gantt chart to display a period of time.
- The first Y-axis containing dates should always be a date which precedes the second Y-axis date.

Example Data Structure #1

Task	Expected Start	Expected End	Actual Start	Actual End	...
X-Axis	Y1-Axis	Y2-Axis	Y3-Axis	Y4-Axis	...
Task 1	01/01/2020	06/01/2020	02/01/2020	11/01/2020	
Task 2	02/22/2020	08/08/2020	03/22/2020	05/01/2020	
Task 3	10/10/2020	12/30/2020	12/01/2020	12/15/2020	

Example Data Structure #2

Task	Division	Team	Expected Start	Expected End	...
X-Axis	Y1-Axis	Y2-Axis	Y3-Axis	Y4-Axis	...
Task 1	X	A	01/01/2020	06/01/2020	
Task 2	X	A	02/22/2020	08/08/2020	
Task 3	X	B	05/01/2020	05/10/2020	
Task 4	Y	C	05/01/2020	07/01/2020	
Task 5	Y	D	10/10/2020	12/30/2020	



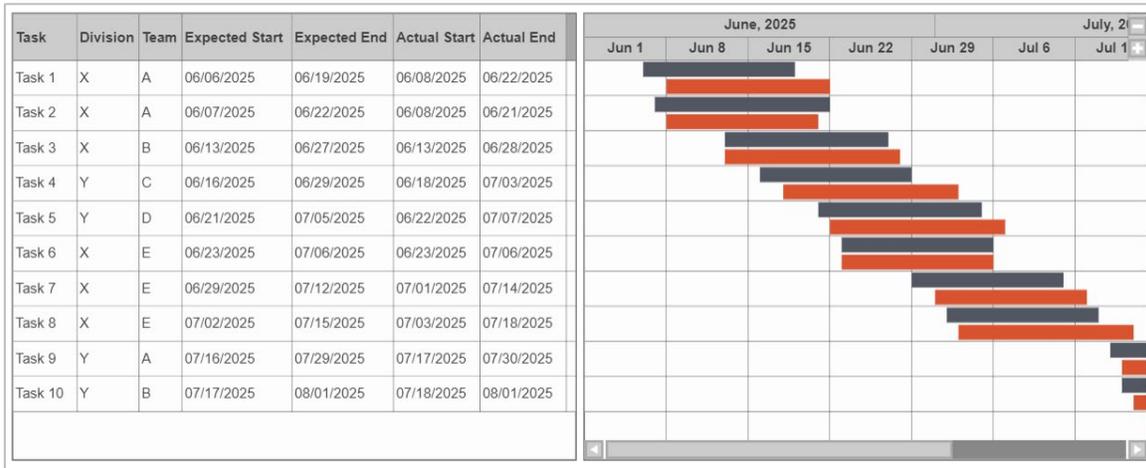
Example Gantt Solution #1

This section includes a progressive concept explaining one example for developing a Gantt chart within iDashboards. The data assumes 10 tasks, 2 corporate divisions, and 5 teams. All tasks were completed because the last task has a populated 'Actual End' date. The goal is to develop a Gantt chart which is capable of displaying each task detail, along with grouping options at the Division and Team level.

Data:

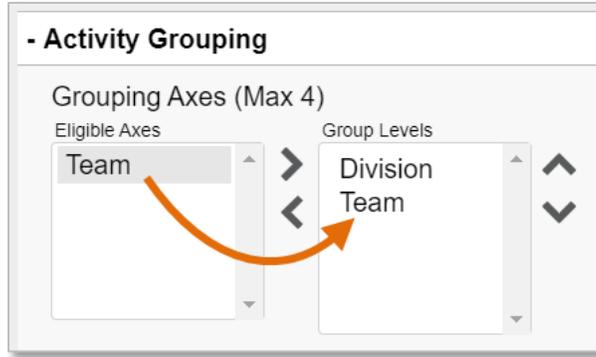
X-Axis	Y1-Axis	Y2-Axis	Y3-Axis	Y4-Axis	Y5-Axis	Y6-Axis
Task 1	X	A	6/6/2025	6/19/2025	6/8/2025	6/22/2025
Task 2	X	A	6/7/2025	6/22/2025	6/8/2025	6/21/2025
Task 3	X	B	6/13/2025	6/27/2025	6/13/2025	6/28/2025
Task 4	Y	C	6/16/2025	6/29/2025	6/18/2025	7/3/2025
Task 5	Y	D	6/21/2025	7/5/2025	6/22/2025	7/7/2025
Task 6	X	E	6/23/2025	7/6/2025	6/23/2025	7/6/2025
Task 7	X	E	6/29/2025	7/12/2025	7/1/2025	7/14/2025
Task 8	X	E	7/2/2025	7/15/2025	7/3/2025	7/18/2025
Task 9	Y	A	7/16/2025	7/29/2025	7/17/2025	7/30/2025
Task 10	Y	B	7/17/2025	8/1/2025	7/18/2025	8/1/2025

Default Chart:

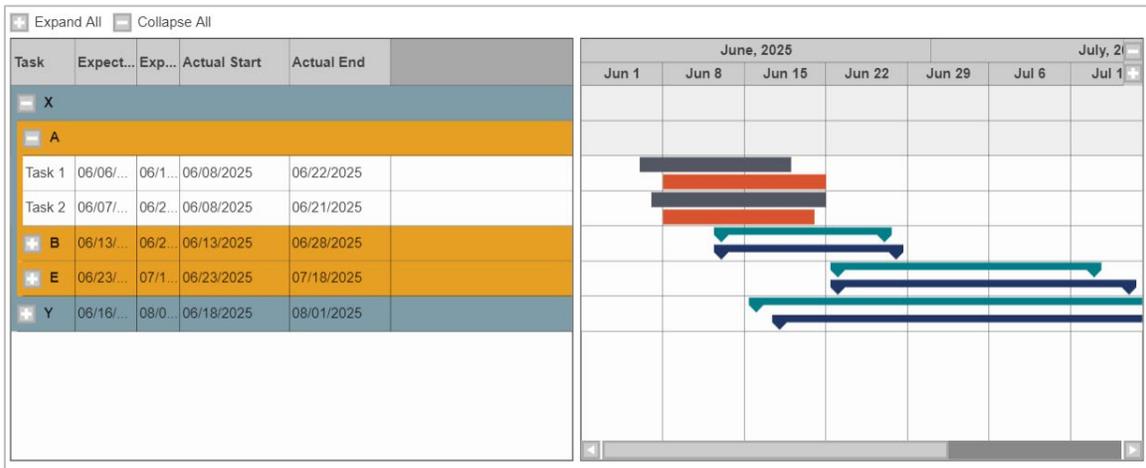


Update Chart Properties:



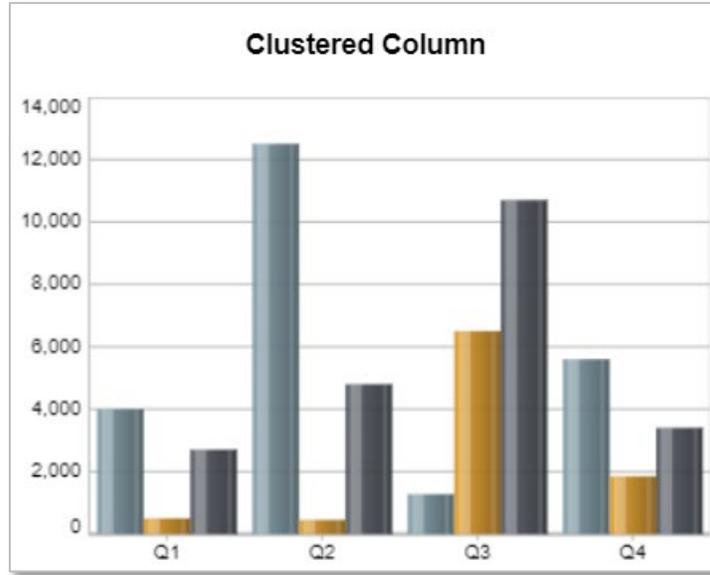


Final Chart:



9.2 Column Charts

- **Clustered Column** – Each Y-variable is drawn as a colored Vertical bar. Compares Y-values across multiple Y-variables (see image below).

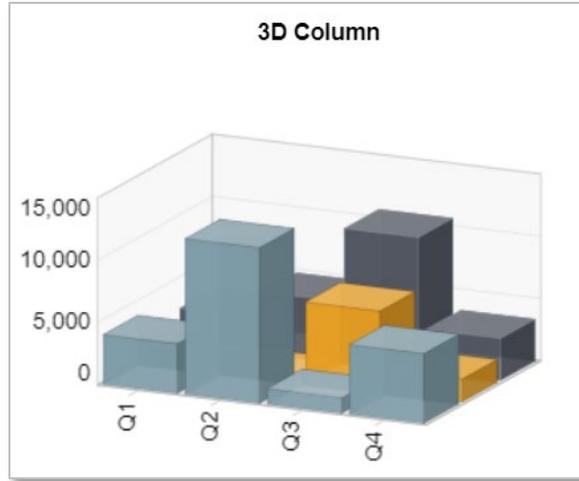


Note: The 'Show Bar Gradient' option is available on the Clustered Column Chart. This option allows for the columns to display with a variety of gradient options.

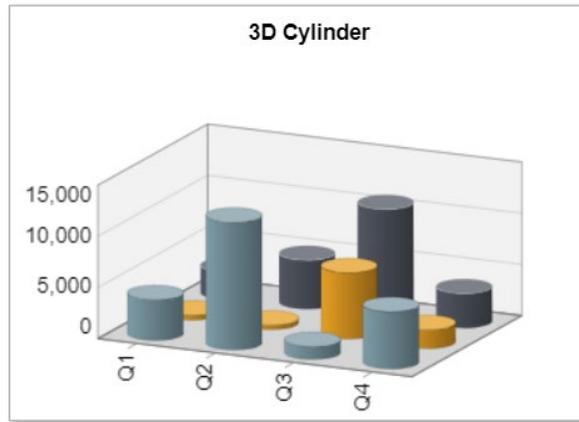
Note: The 'Use Mixed Y-Axis Ranges' option is available on the Clustered Column (and Bar) Chart, 3D Column and 3D Cylinder charts. This option allows for the graph to plot the values in their proportions. For example, the values 100, 200, and 300 would plot the same as 10, 20, and 30 since they have the same proportions. The 'Use Mixed Y-Axis Ranges' option on the Clustered Bar Chart will work only if three conditions are met:

1. *The chart has more than one numeric Y value.*
2. *The 'Use Mixed Y-Axis Ranges' option is checked.*
3. *The 'Match Type' option in the basic setting is set to "Y Axis Name".*

- **3D Column** – Each Y-variable is drawn as a colored 3D vertical column and arranged in a three-dimensional formation. Dark tops on the column will indicate a negative value (see image below).



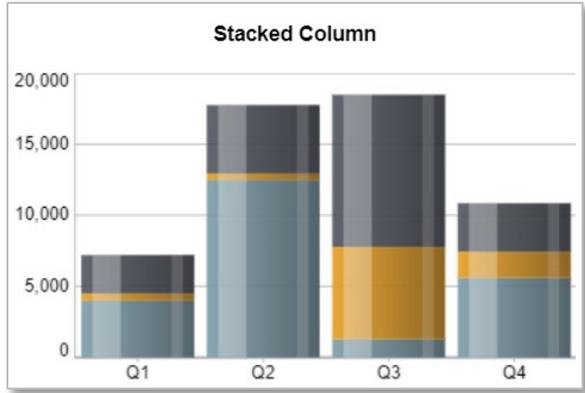
- **3D Cylinder** – Each Y-variable is drawn as a colored 3D vertical cylinder and arranged in a three-dimensional formation. Dark tops on the cylinder will indicate a negative value (see image below).



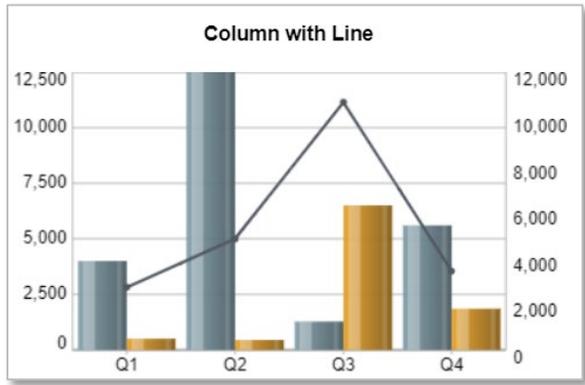
Note: The 'Use Mixed Y-Axis Ranges' option is available on the Clustered Column (and Bar) Chart, 3D Column and 3D Cylinder charts. This option allows for the graph to plot the values in their proportions. For example, the values 100, 200, and 300 would plot the same as 10, 20, and 30 since they have the same proportions. The 'Use Mixed Y-Axis Ranges' option on the Clustered Bar Chart will work only if three conditions are met:

1. *The chart has more than one numeric Y value.*
2. *The 'Use Mixed Y-Axis Ranges' option is checked.*
3. *The 'Match Type' option in the basic setting is set to "Y Axis Name".*

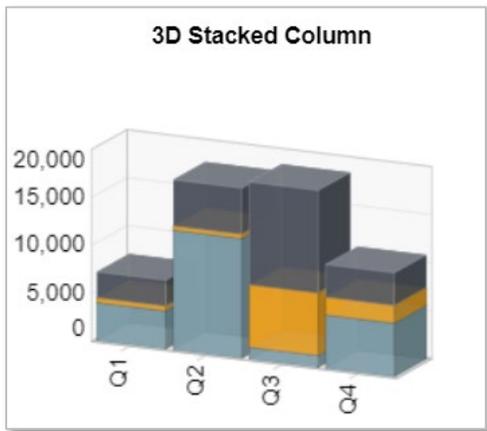
- **Stacked Column** – Each Y-variable is drawn as a colored stack on a multicolor Vertical bar. Compares the contribution of each Y-variable to a total across multiple Y-variables (see image below).



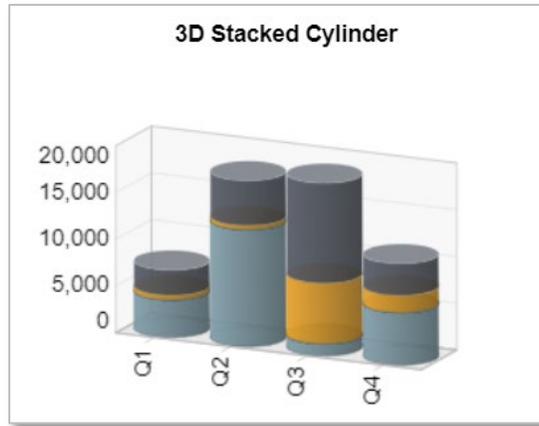
- **Stacked with Line** – Each Y-variable is drawn as a colored stack on a multicolor vertical column with the last y-variable(s) plotted as line(s) on a separate scale (see image below).



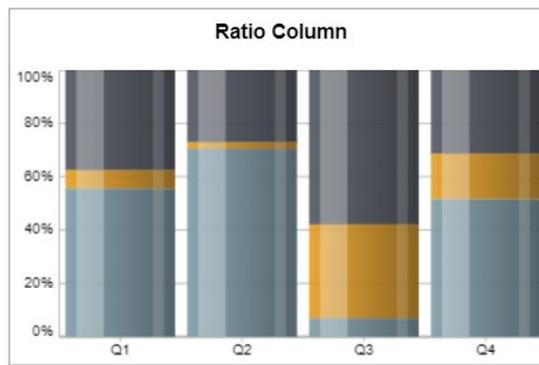
- **3D Stacked Column** – Each Y-variable is drawn as a colored stack on a multicolor vertical 3D column (see image below).



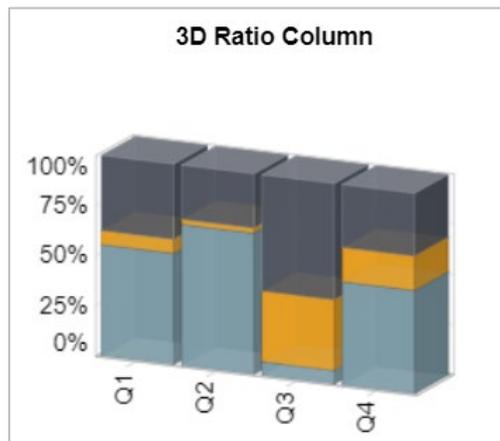
- **3D Stacked Cylinder** – Each Y-variable is drawn as a colored stack on a multicolor vertical cylinder (see image below).



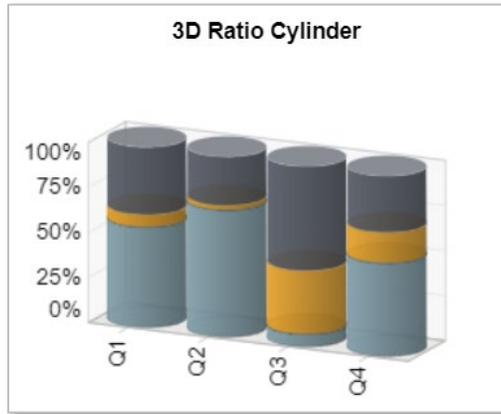
- **Ratio Column** – Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a stacked vertical column (see image below).



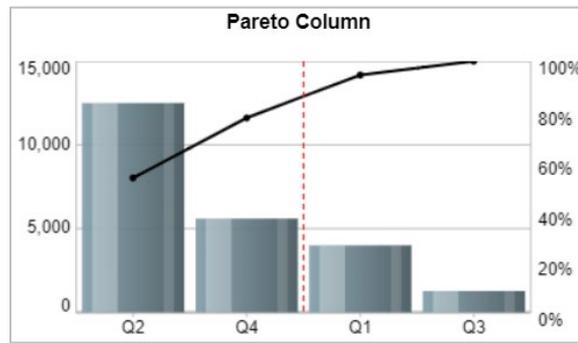
- **3D Ratio Column** – Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a stacked vertical 3D column (see image below).



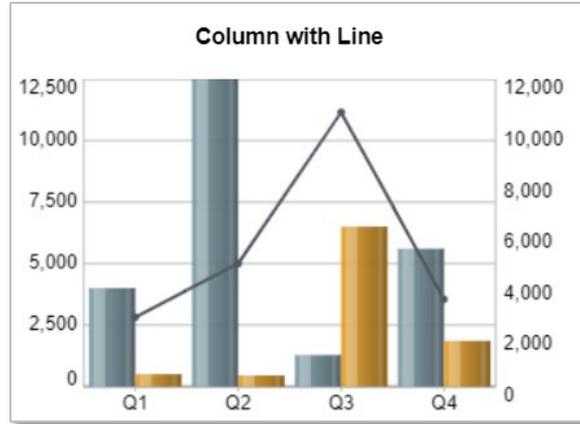
- **3D Ratio Cylinder** – Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a stacked vertical cylinder (see image below).



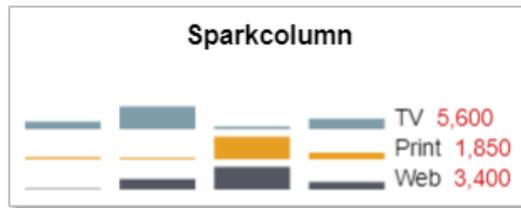
- **Pareto Column** – A special type of histogram, arranged in order of severity from largest to smallest, reflecting the frequency or impact of entities. The line indicates cumulative percentage. User specified sort order will be ignored for this chart type, as Pareto data items are arranged from largest to smallest. The left-side vertical axis will display the values of the individual entity. The right-side vertical axis will always be the cumulative percentage from 0 to 100%. The vertical dashed line represents the user-defined breakpoint (see image below).



- **Column with Line** – Clustered column chart with last Y-variable plotted as a line on a separate scale that is drawn on the right vertical axis. The last Y-axis is represented with a line rather than a column. The user can control how many columns should be displayed as lines. The first Y-axis displayed as a column must remain a column and cannot be displayed as a line. However, if all axes need to be a line, the user can change the chart type to a Line Chart (see image below).

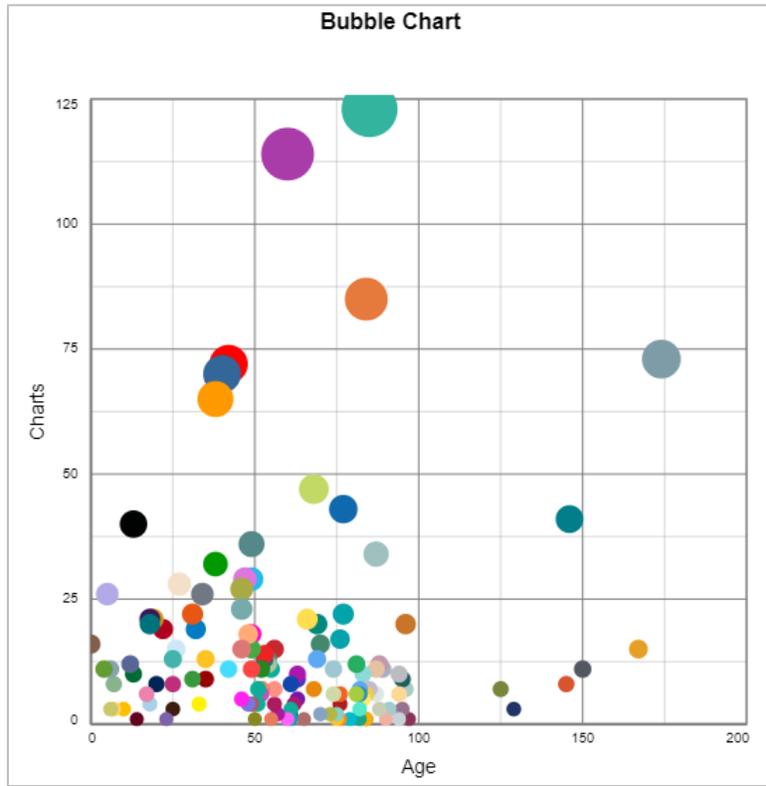


- **Sparkcolumn** – Variations of each Y-variable are represented in individual column charts, stacked into a simple and condensed chart (see image below).



9.3 Bubble Charts

- **Bubble Chart** – Each bubble is an intersection of two measures, and a third measure determines the size of the bubble. Each bubble is displayed as a 3D sphere. However, if the chart has only two measures (2 sets of Y-variables), then all the bubbles are identical in size. The location of each bubble is determined by the intersection of the two measures.



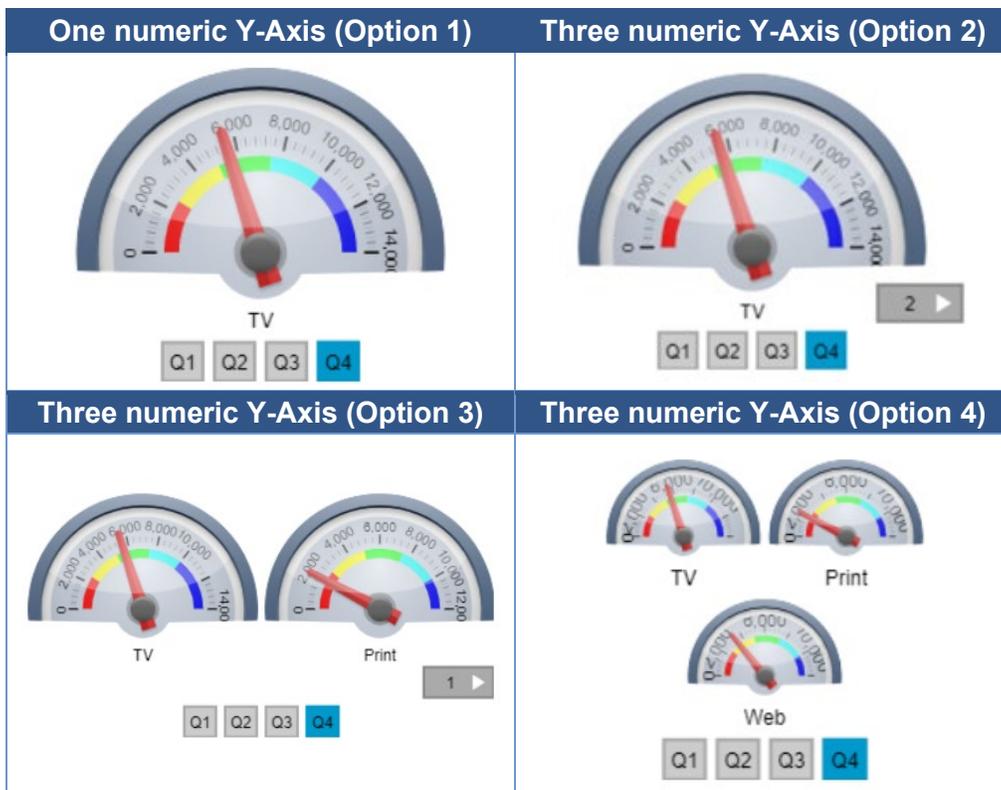
9.4 Speedometer Charts

Features include ability to change dial, needle and text colors besides the colors and thresholds for the displayed scale within each speedometer.

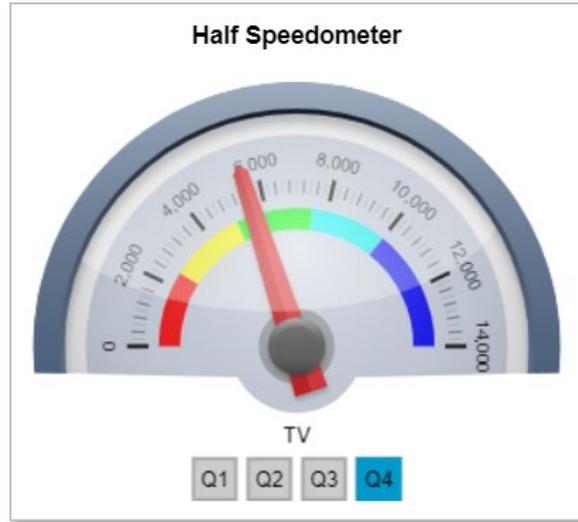
Clustered Display

If more than one Y-Axis is part of the data set, most of the speedometer chart types offer a clustered display option. When the number of displayed speedometers is greater than one, then multiple speedometer 'charts' will appear within a single frame.

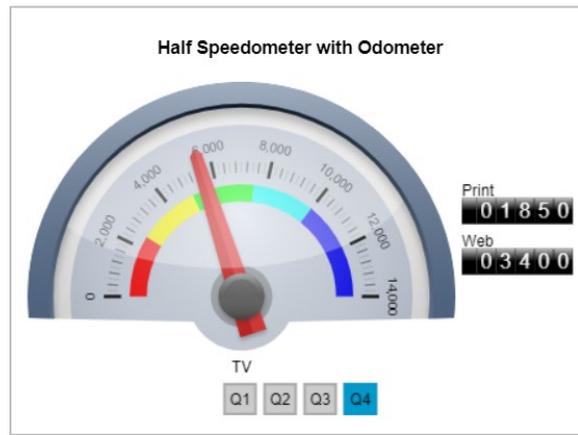
Note: There is no limit to the number of clustered charts that can appear within a single frame. The maximum limit is usually defined aesthetically by the person building the chart.



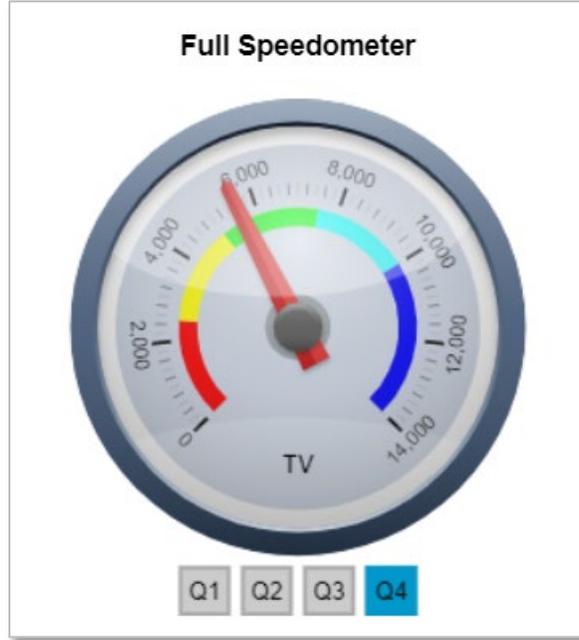
- **Half Speedometer** – Each Y-variable is displayed as an animated needle on a half-round speedometer (see image below).



- **Half Speedometer with Odometers** – The first Y-variable is displayed as an animated needle on a half-round speedometer and the rest of the Y-variables are displayed in an odometer like interface (see image below).



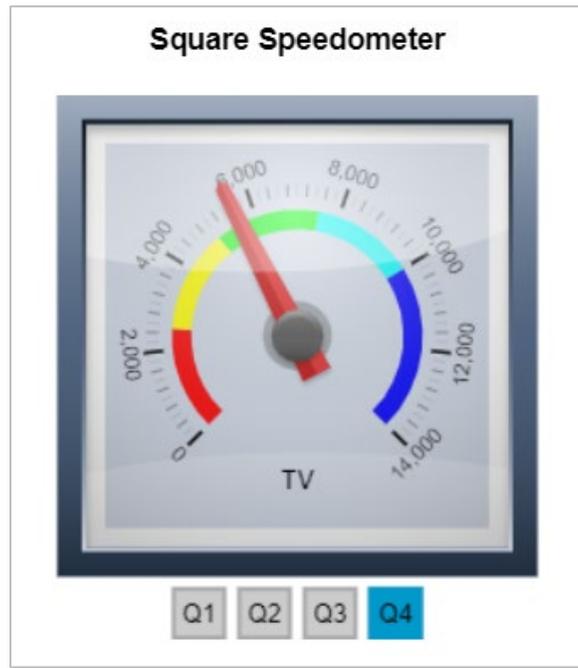
- **Full Speedometer** – Displays through an animated needle the value of each Y-data point in a full round Speedometer interface (see image below).



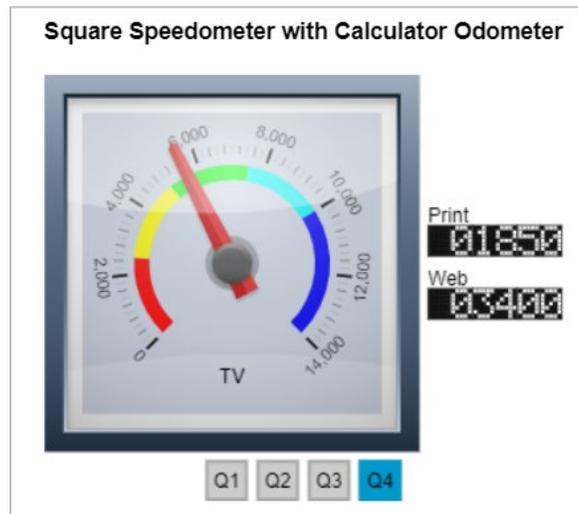
- **Full Speedometer with Odometers** – Displays through an animated needle the value of first Y-data point in a Speedometer interface. Remaining Y-data points are displayed as text on the right side of the speedometer. The text and speedometer needle animate with any change of an x-value due to cursor location (see image below).



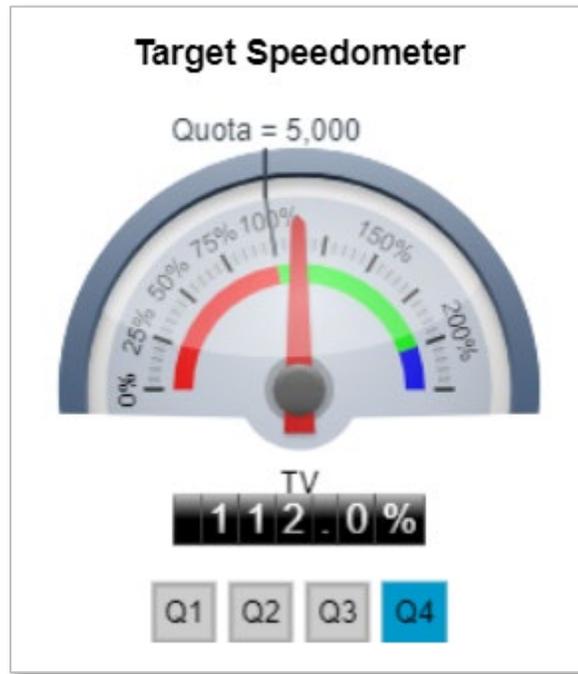
- **Square Speedometer** – Y-variables are displayed as an animated needle on a square speedometer (see image below).



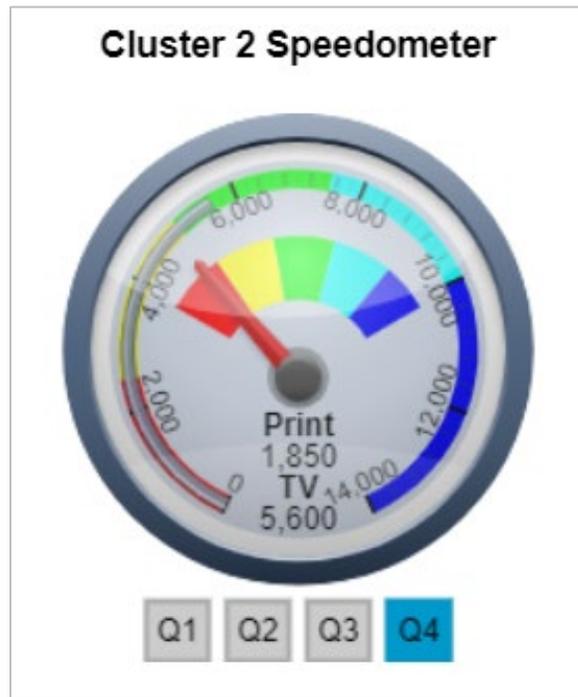
- **Square Speedometer with Odometers** – Specified number of Y-variables is displayed as an animated needle on a square speedometer and the rest of the Y-variables are displayed in an odometer-like interface (see image below).



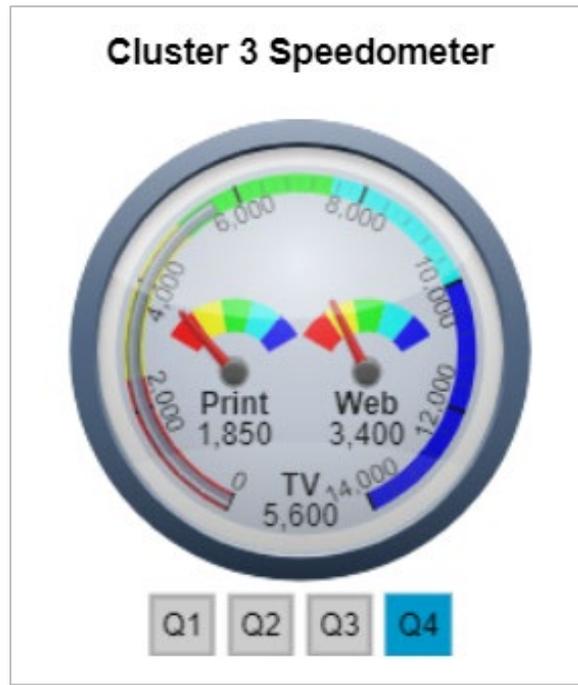
- **Target Speedometer** – Display the first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in horizontal thermometer gauges (see image below).



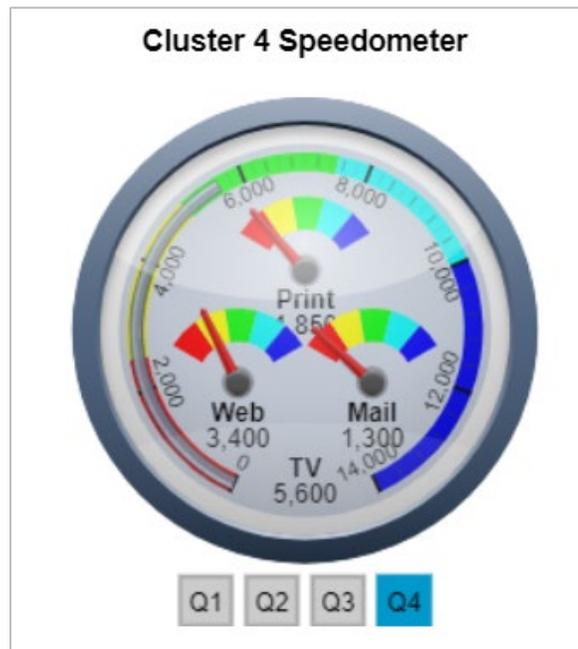
- **Cluster-2 Speedometer** – Y-variables are displayed as an animated needle on a full speedometer with an inner speedometer (see image below).



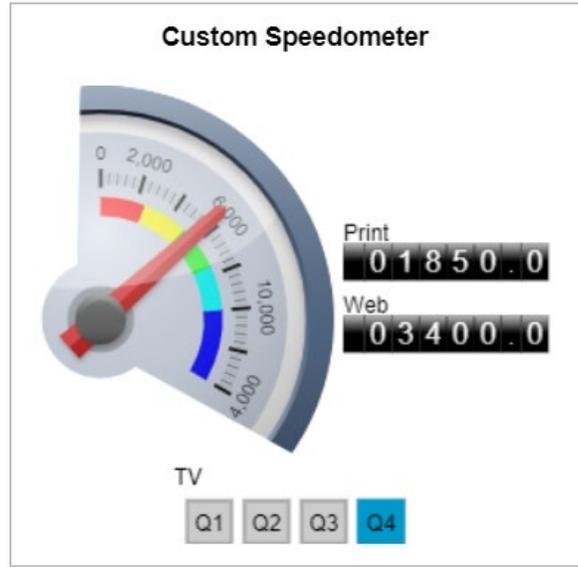
- **Cluster-3 Speedometer** – Y-variables are displayed as an animated needle on a full speedometer with two inner speedometers (see image below).



- **Cluster-4 Speedometer** – Y-variables are displayed as an animated needle on a full speedometer with three inner speedometers (see image below).



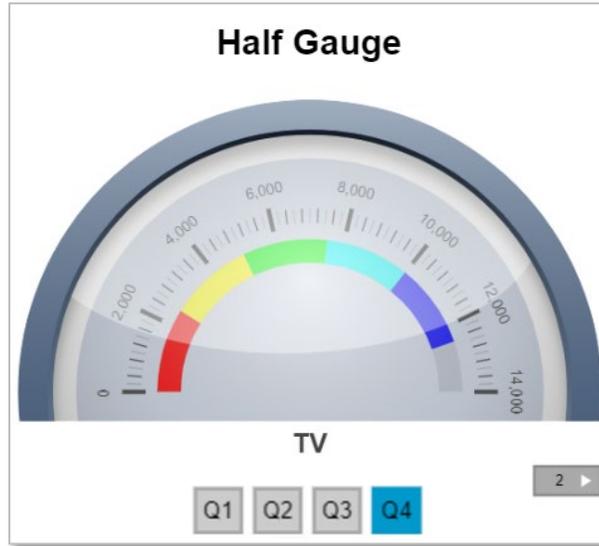
- **Custom Speedometer** – Y-variables are displayed as an animated needle with a variety of speedometer face, border, shape, needle and meter options (see image below).



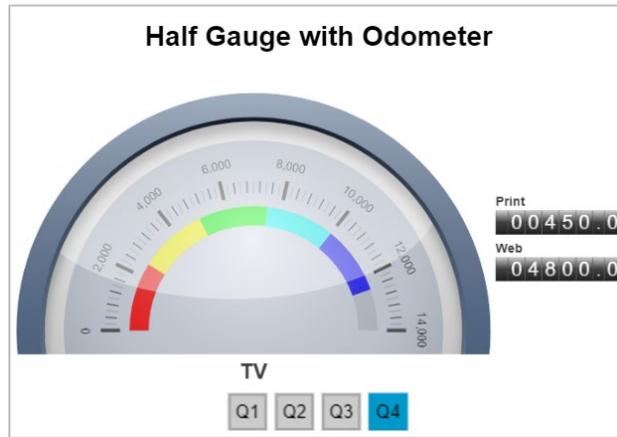
- **Multi-needle Speedometer** – Similar to the Custom Speedometer chart including the option to display multiple needles for when there are multiple Y-axes. The user has options to display a combination of multiple needles, multiple displayed speedometers and multiple odometers. Colors of each needle can be customized under the Colors tab (see image below).



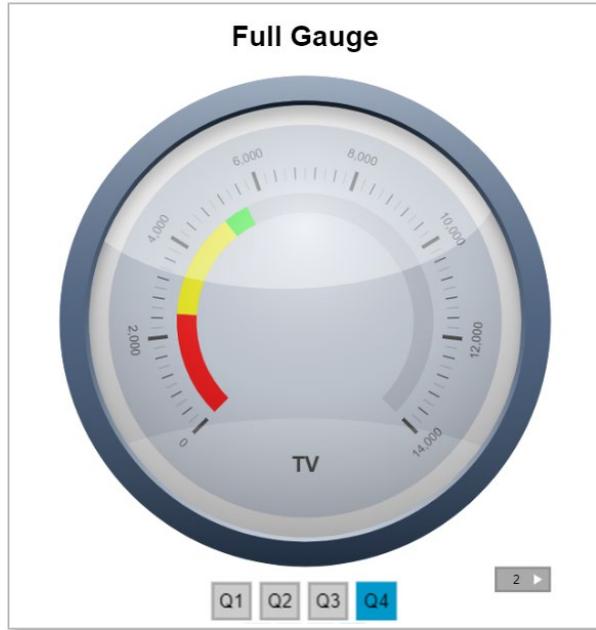
- **Half Gauge** – Y-variables are displayed as an animated arc on a half-round speedometer (see image below).



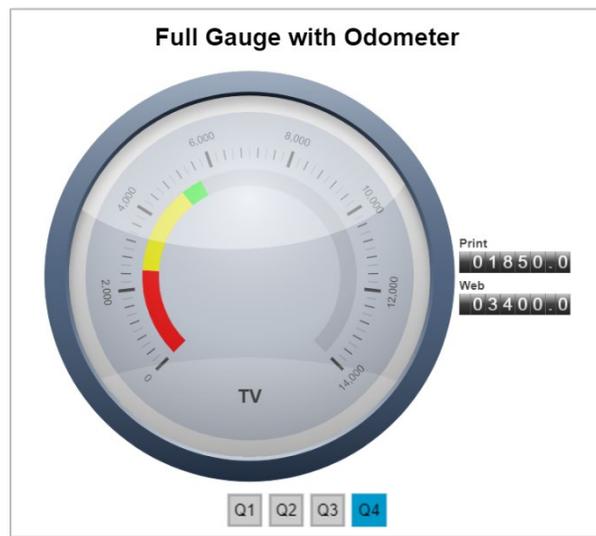
- **Half Gauge with Odometer** – The first Y-variable is displayed as an animated arc on a half-round speedometer and the rest of the Y-variables are displayed in an odometer-like interface (see image below).



- **Full Gauge** – Y-variables are displayed as an animated arc on a full speedometer (see image below).



- **Full Gauge with Odometer** – The first Y-variable is displayed as an animated arc on a full speedometer and the rest of the Y-variables are displayed in an odometer-like interface (see image below).



9.4.1 Speedometer Themes

All Speedometer charts have Themes. Themes apply a certain value to a variety of properties which change the appearance of the speedometer. Manually, these same properties could be configured but selecting a Theme will certainly ease this task. Including 'Default', there are nine pre-defined themes: Classic, Modern, Chrome, Hot Rod, Gadget, Ranger, Precision, and Pointer.



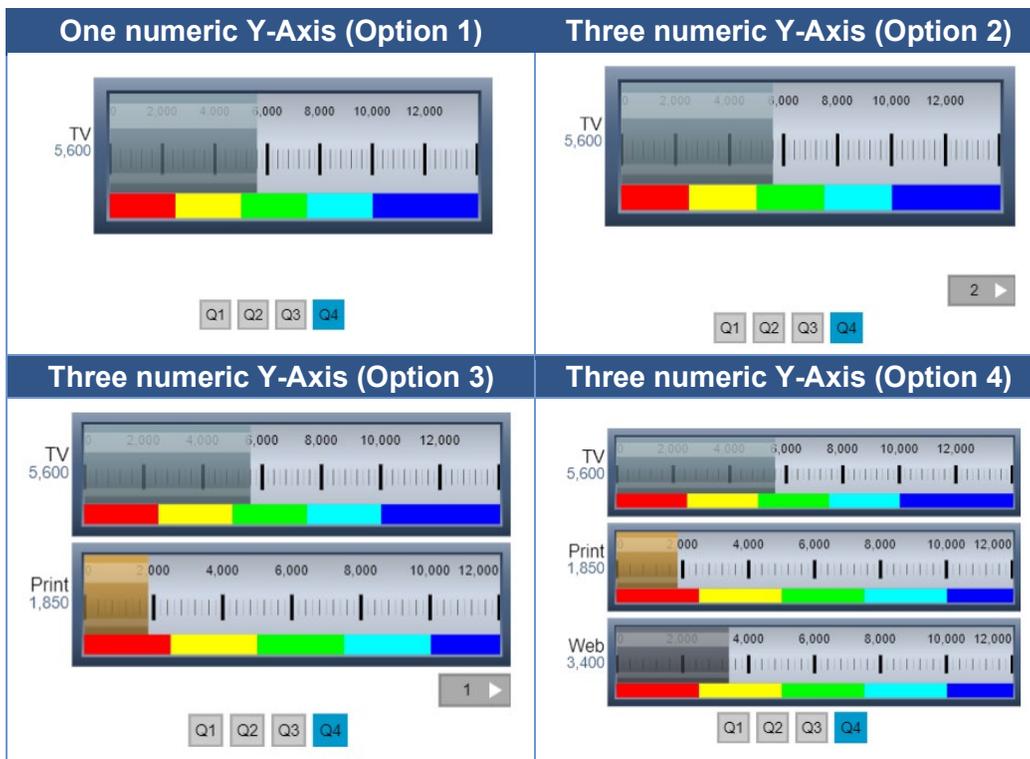
9.5 Bullet Charts

Features include ability to change indicator-type and color, background-track type and color, thresholds, odometer options and much more.

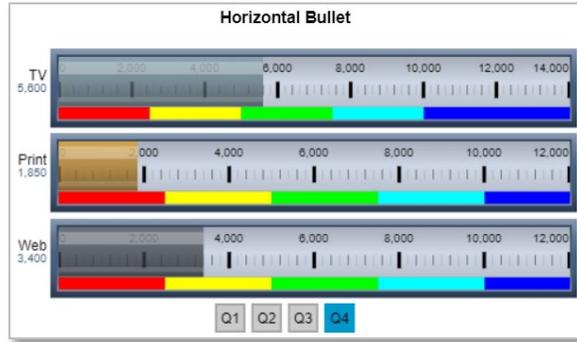
Clustered Display

If more than one Y-Axis is part of the data set, most of the bullet chart types offer a clustered display option. When the number of displayed bullets is greater than one, then multiple bullet 'charts' will appear within a single frame.

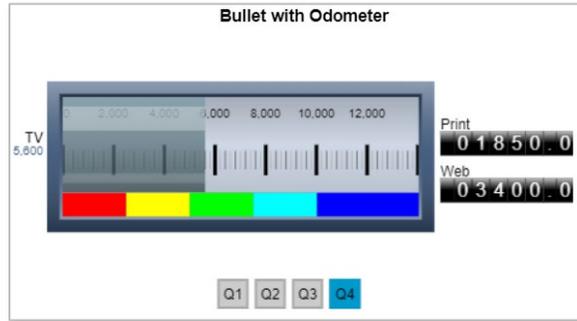
Note: There is no limit to the number of clustered charts that can appear within a single frame. The maximum limit is usually defined aesthetically by the person building the chart.



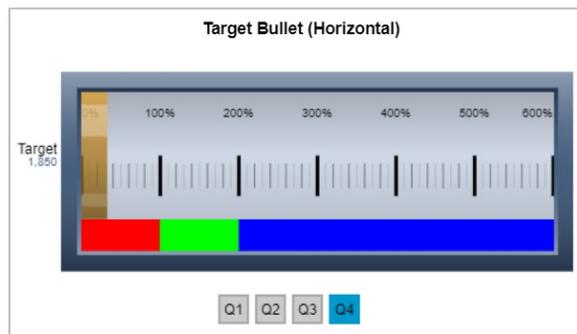
- **Horizontal Bullet** – Y-variables are displayed in linear horizontal gauges. Features include the ability to change the range colors and range values of the bar.



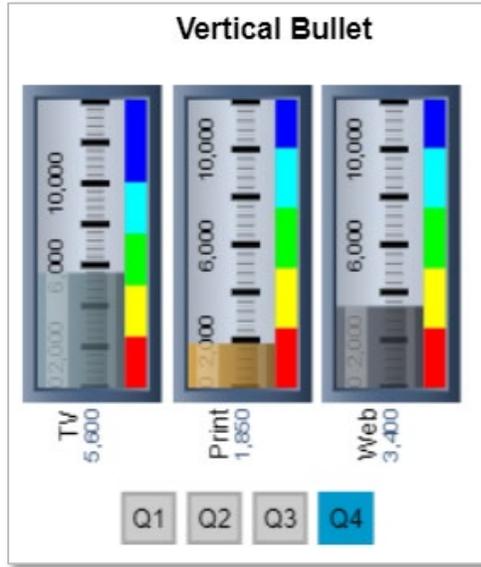
- **Bullet with Odometer (horizontal icon)** – The first Y-variable is displayed as a linear horizontal gauge and the rest of the Y-variables are displayed in an odometer like interface.



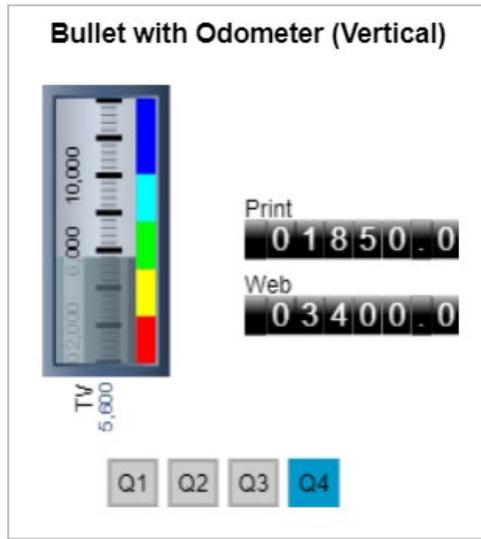
- **Target Bullet (horizontal icon)** – Displays first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in linear horizontal gauges.



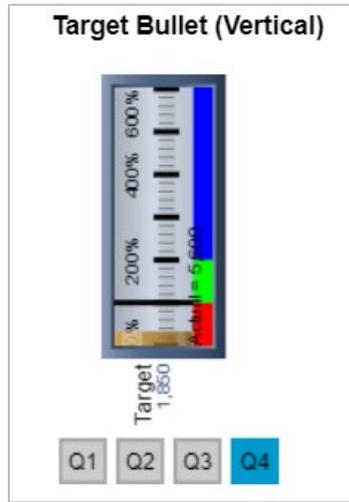
- **Vertical Bullet** – Y-variables are displayed in linear vertical gauges.



- **Bullet with Odometer (vertical icon)** – The first Y-variable is displayed as a vertical gauge and the rest of the Y-variables are displayed in an odometer like interface.

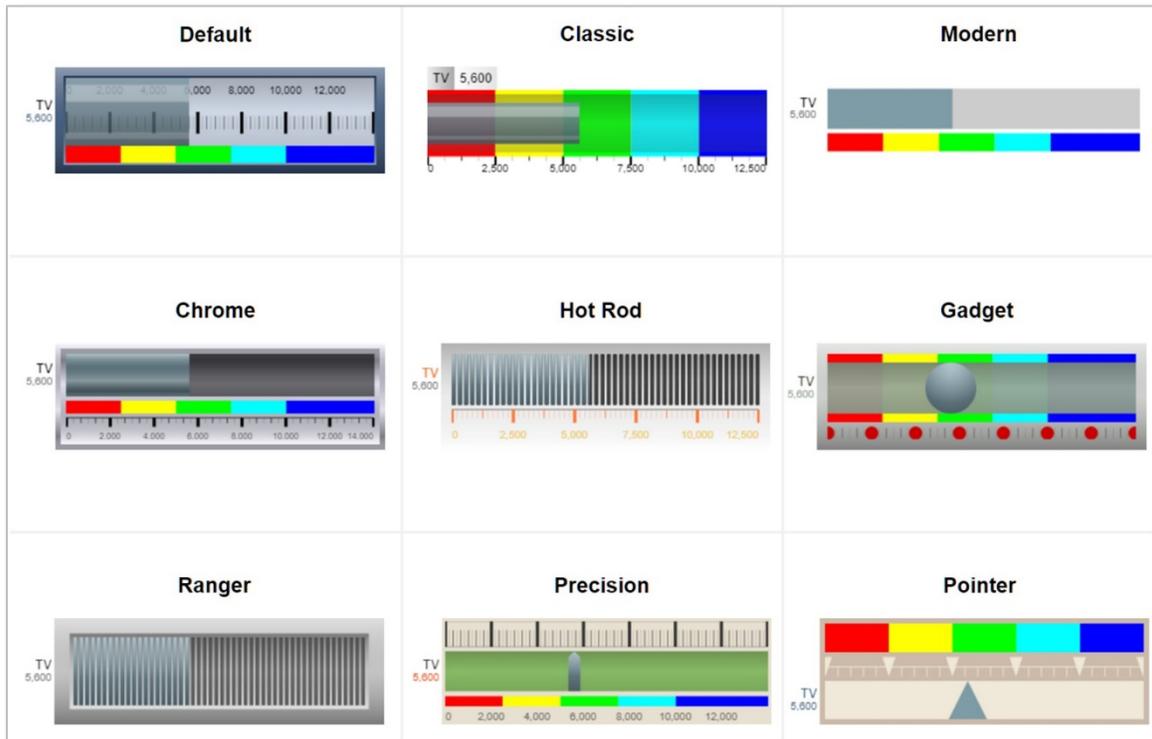


- **Target Bullet (vertical icon)** – Displays first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in linear vertical gauges.



9.5.1 Bullet Themes

All Bullet charts have Themes. Themes apply a certain value to a variety of properties which change the appearance of the bullet. Manually, these same properties could be configured but selecting a Theme will certainly ease this task. Including 'Default', there are nine pre-defined themes: Classic, Modern, Chrome, Hot Rod, Gadget, Ranger, Precision, and Pointer.



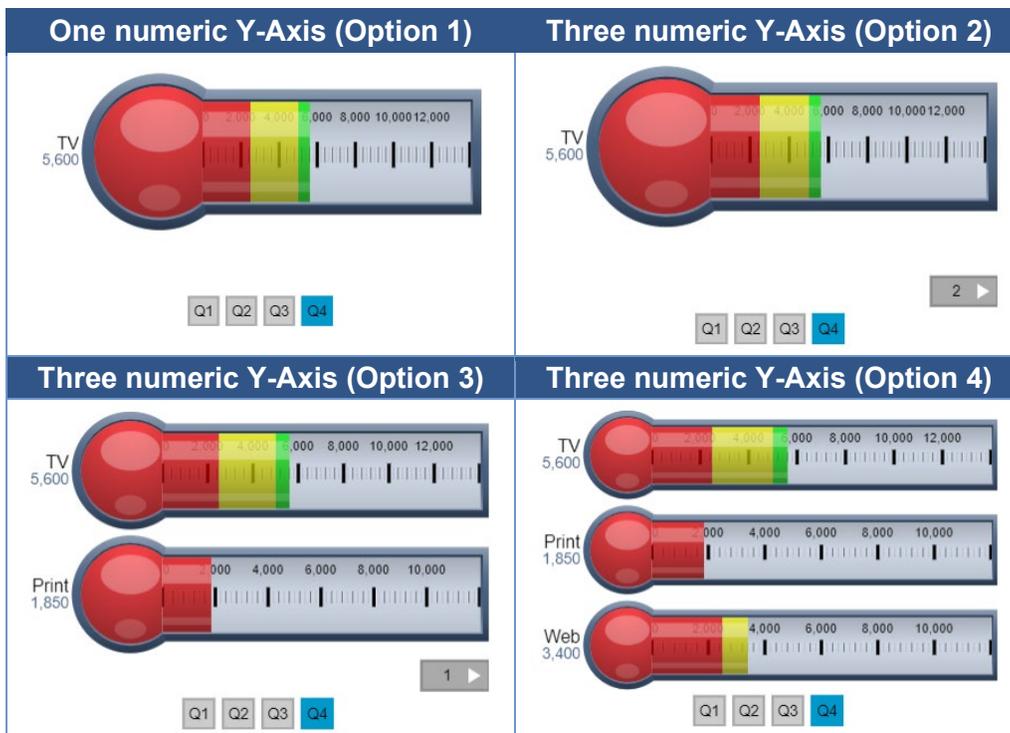
9.6 Thermometer Charts

Features include ability to change indicator-type and color, background-track type and color, thresholds, odometer options and much more.

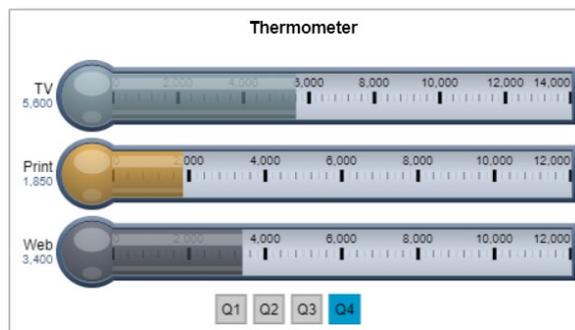
Clustered Display

If more than one Y-Axis is part of the data set, most of the thermometer chart types offer a clustered display option. When the number of displayed thermometers is greater than one, then multiple thermometer 'charts' will appear within a single frame.

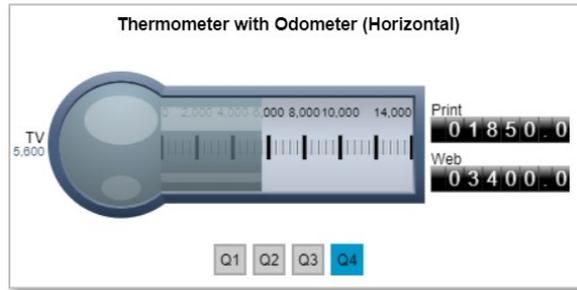
Note: There is no limit to the number of clustered charts that can appear within a single frame. The maximum limit is usually defined aesthetically by the person building the chart.



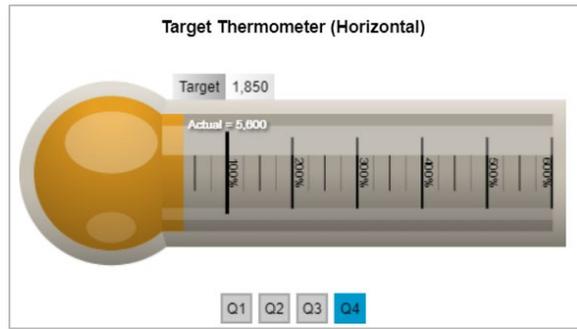
- **Thermometer** – Y-variables are displayed in horizontal thermometer gauges.



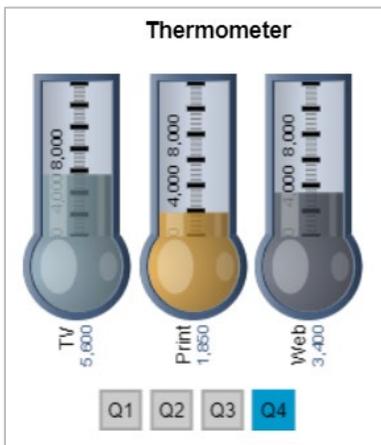
- **Thermometer with Odometer (horizontal icon)** – The first Y-variable is displayed as a horizontal thermometer gauge and the rest of the Y-variables are displayed in an odometer like interface.



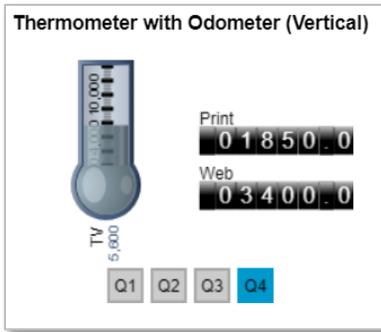
- **Target Thermometer (horizontal icon)** – Displays first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in horizontal thermometer gauges.



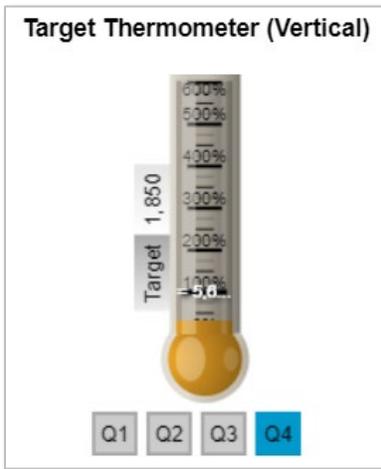
- **Vertical Thermometer** – Y-variables are displayed in vertical thermometer gauges.



- **Thermometer with Odometer (vertical icon)** – The first Y-variable is displayed as a vertical thermometer gauge and the rest of the Y-variables are displayed in an odometer like interface.

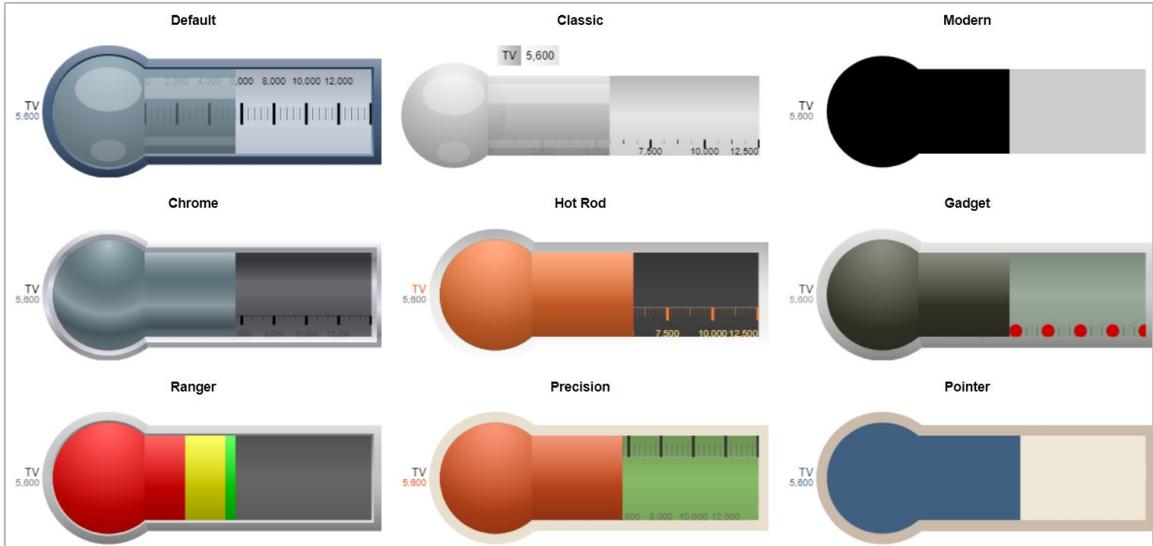


- **Target Thermometer (vertical icon)** – Displays first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in vertical thermometer gauges.



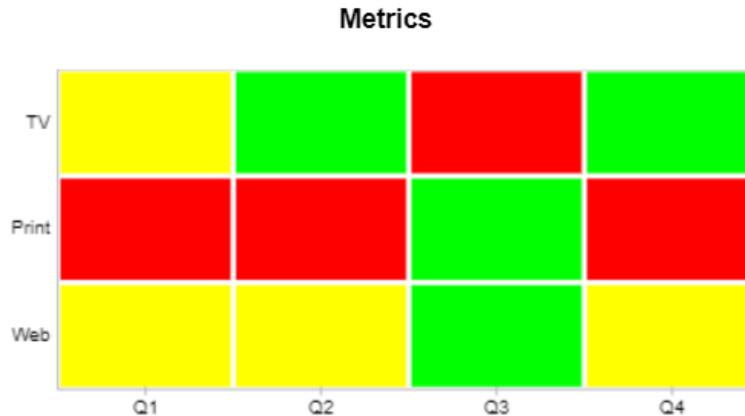
9.6.1 Thermometer Themes

All Thermometer charts have Themes. Themes apply a certain value to a variety of properties which change the appearance of the thermometer. Manually, these same properties could be configured but selecting a Theme will certainly ease this task. Including 'Default', there are nine pre-defined themes: Classic, Modern, Chrome, Hot Rod, Gadget, Ranger, Precision, and Pointer.



9.7 Metrics Charts

- **Metrics Chart** - Displays colored pixels to show relative performance of one or more metrics against corresponding performance benchmarks. Features include the ability to change the range colors and Datapoint shape.



- **Metrics Bar Chart** - The same as a Metrics chart, except that the X values are displayed along the left side of the matrix instead of the bottom.
- **Metrics Target Chart** - Displays first, third, fifth, etc. Y-variables as target values; and the second, fourth, sixth, etc. Y-variables as percentages of those targets, respectively in colored shapes. Use the mouse-over legend to see the percentages.
- **Metrics Bar Target Chart** - The same as a Metrics target chart, except that the X values are displayed along the left side of the matrix instead of the bottom.
- **Metrics Scorecard** - This chart displays all the values of X and Y variables in a multi-row, multi-column Tabular format with colored pixels showing relative performance of one or more metrics against corresponding performance benchmarks. All numeric Y-axes will expect a range value set for defining the box color. Similar to the Tabular chart, but with additional color options.
- **Horizontal Metrics Ticker** - Displays metrics data scrolling from right to left as a ticker, with option to display colored up or down arrows for positive and negative values respectively.
- **Vertical Metrics Ticker** - Displays metrics data scrolling from bottom to top as a ticker, with option to display colored up or down arrows for positive and negative values respectively. Ticker data is refreshed automatically per configurable frequency.
- **Stoplight** – Similar to a traffic light, this chart highlights one of three colored pixel in a spectrum of benchmark pixels arranged horizontally indicating performance of one or more metrics.
- **Vertical Stoplight** – Highlights a colored pixel in a spectrum of benchmark pixels indicating performance of one or more metrics.

Note: Stoplight chart types have the option to display Y-axis variables as odometer displays or to represent each Y-axis variable in a separate stoplight.

9.7.1 Stoplight Themes

Both Stoplight charts have Themes. Themes apply a certain value to a variety of properties which change the appearance of the stoplight. Manually, these same properties could be configured but selecting a Theme will certainly ease this task. Including 'Default', there are nine pre-defined themes: Classic, Modern, Chrome, Hot Rod, Gadget, Ranger, Precision, and Pointer.

9.7.2 Treemap Charts

A treemap chart displays rectangle 'boxes' that have an area proportional to their data values. Using the examples below: these boxes can be organized into groups by selecting a non-numeric Y-axis. Another example of the treemap chart has a dataset containing seven (7) rows of data. Each row of data will become a box-slice within the entire rectangular shape. There are many visual customizations available with the treemap chart and many of the guidelines will be described within this section.

Note: Ratio charts can only display values greater than zero.

The treemap chart combines features available in other charts from iDashboards. For example, like a Pie Chart, the treemap is a ratio-chart using percentage values to define the area of each rectangular slice (aka 'box') as part of the entire graph. However, unlike a Pie Chart, the treemap chart is better at spatial data placement and is better at displaying dozens or hundreds of data points at the same time. Like a Column Chart, the color of each box can be defined using range colors. And like the Report Chart, there is a grouping option to visually associate X-axis values based on a Y-axis grouping value.

Labels

- Name, Value and Percentage labels can be turned on or off by customizing the chart properties.
- Each of these labels have a size, color and boldness setting
- Labels will appear within a box if the box size is large enough to display the label. If the box is too small to display the label, the chart will attempt to display at least one (1) character along with an ellipsis (...). If the box is too small for displaying any characters then the label will not be used.

Grouping

- Grouping boxes with a data axis is available when a third (3rd) axis is mapped to the chart using the data type 'String'.
- Range Colors define the color of each box unless the 'Grouping Axis' property is being used. If the Grouping Axis property is used, the box grouping colors are generated by iDashboards and assigning specific colors is not permissible.
- Grouping does not reduce the number of total boxes shown in the chart. Instead, using a Group Axis restructures the sorted order of boxes to first consider the grouping total, the



<p>5</p>	<p>Similar to Example 4 but the chart is using a shade called “Chrome” and the “Range Axis” is disabled, causing the transparency to disappear.</p>	 <table border="1"> <thead> <tr> <th>City</th> <th>Population</th> <th>State</th> </tr> </thead> <tbody> <tr> <td>Oakland</td> <td>6,769,205</td> <td>CA</td> </tr> <tr> <td>Sacramento</td> <td>6,370,753</td> <td>CA</td> </tr> <tr> <td>Los Angeles</td> <td>4,282,208</td> <td>CA</td> </tr> <tr> <td>San Francisco</td> <td>5,526,442</td> <td>CA</td> </tr> <tr> <td>San Diego</td> <td>2,035,491</td> <td>CA</td> </tr> <tr> <td>Modesto</td> <td>2,470,186</td> <td>CA</td> </tr> <tr> <td>Grand Rapids</td> <td>2,933,387</td> <td>MI</td> </tr> <tr> <td>Traverse ...</td> <td>2,609,111</td> <td>MI</td> </tr> <tr> <td>Detroit</td> <td>2,264,710</td> <td>MI</td> </tr> <tr> <td>Ann Arb...</td> <td>2,236,605</td> <td>MI</td> </tr> <tr> <td>Midland</td> <td>1,315,891</td> <td>MI</td> </tr> <tr> <td>Lansing</td> <td>1,079,135</td> <td>MI</td> </tr> <tr> <td>Bloomingt...</td> <td>1,421,385</td> <td>IL</td> </tr> <tr> <td>Chicago</td> <td>1,381,381</td> <td>IL</td> </tr> <tr> <td>Joliet</td> <td>1,194,783</td> <td>IL</td> </tr> <tr> <td>Bell...</td> <td>593,570</td> <td>IL</td> </tr> <tr> <td>Spr...</td> <td>570,000</td> <td>IL</td> </tr> <tr> <td>S...</td> <td></td> <td></td> </tr> <tr> <td>E...</td> <td></td> <td></td> </tr> </tbody> </table>	City	Population	State	Oakland	6,769,205	CA	Sacramento	6,370,753	CA	Los Angeles	4,282,208	CA	San Francisco	5,526,442	CA	San Diego	2,035,491	CA	Modesto	2,470,186	CA	Grand Rapids	2,933,387	MI	Traverse ...	2,609,111	MI	Detroit	2,264,710	MI	Ann Arb...	2,236,605	MI	Midland	1,315,891	MI	Lansing	1,079,135	MI	Bloomingt...	1,421,385	IL	Chicago	1,381,381	IL	Joliet	1,194,783	IL	Bell...	593,570	IL	Spr...	570,000	IL	S...			E...		
City	Population	State																																																												
Oakland	6,769,205	CA																																																												
Sacramento	6,370,753	CA																																																												
Los Angeles	4,282,208	CA																																																												
San Francisco	5,526,442	CA																																																												
San Diego	2,035,491	CA																																																												
Modesto	2,470,186	CA																																																												
Grand Rapids	2,933,387	MI																																																												
Traverse ...	2,609,111	MI																																																												
Detroit	2,264,710	MI																																																												
Ann Arb...	2,236,605	MI																																																												
Midland	1,315,891	MI																																																												
Lansing	1,079,135	MI																																																												
Bloomingt...	1,421,385	IL																																																												
Chicago	1,381,381	IL																																																												
Joliet	1,194,783	IL																																																												
Bell...	593,570	IL																																																												
Spr...	570,000	IL																																																												
S...																																																														
E...																																																														

9.8 Line Charts

- **Trend Line** - Plots Y-variables as lines against the vertical axis.
- **Sparkline** - Variations of each Y-variable are represented in individual line charts, stacked into a simple and condensed chart.
- **Scatter Chart** - Plots Y-variables as individual data points with an option to have a regression line for each Y-variable.
- **Smooth Line** - Plots Y-variables as smooth lines against the vertical axis.
- **3D Trend Line** - Plots Y-variables as 3D lines against the vertical axis.
- **Area** - Plots Y-variables as lines with shading of the area between the lines and the X-axis.
- **Smooth Area** - Plots Y-variables as smooth lines with shading of the area between the lines and the X-axis.
- **Stacked Area** - Plots Y-variables as consecutive stack of lines with shading of the area between the lines and the X-axis.
- **Stacked Ratio Area** - Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a stacked area chart.
- **3D Area** - Plots Y-variables as 3D lines with shading of the area between the lines and the X-axis.
- **3D Stacked Area** - Plots Y-variables as consecutive stack of 3D lines with shading of the area between the lines and the X-axis.
- **3D Stacked Ratio Area** - Displays the relative contribution of each Y-variable as a percentage of the total that is represented in a 3D stacked area chart.



9.9 Pie Charts

Note: Pie Charts can only display a maximum of 70 rows.

Note: Ratio charts can only display values greater than zero.

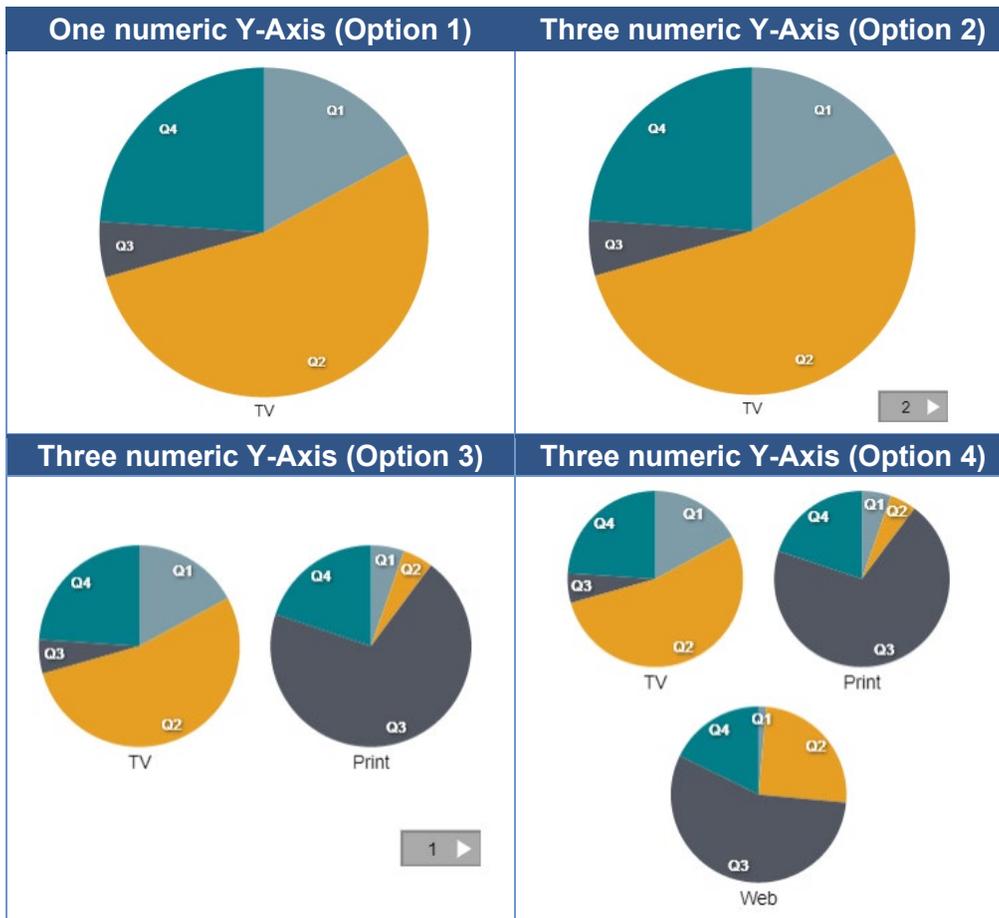
Note: There will always be a possibility that the pie slice percentages do not appear to add up to exactly 100 percent. Take for example the following 5 slices: 20.6, 20.6, 20.6, 20.6, and 17.6. They add up to exactly 100%. However, if you round each to its nearest whole number, you have 21, 21, 21, 21 and 18, which add up to 102%.

Note: "Sum Label Options" can be used to display the total numeric value.

Clustered Display

If more than one Y-Axis is part of the data set, most of the pie chart types offer a clustered display option. When the number of displayed pies is greater than one, then multiple pie 'charts' will appear within a single frame.

Note: There is no limit to the number of clustered charts that can appear within a single frame. The maximum limit is usually defined aesthetically by the person building the chart.



- **2D Pie** - Displays the relative contribution of each Y-variable to the total that is represented in a circular pie.
- **Exploded 2D Pie** - A 2-dimensional pie with gaps between the slices. Displays the relative contribution of each Y-variable to the total that is represented in a circular pie.
- **2D Donut** - A donut-shaped pie chart. Displays the relative contribution of each Y-variable to the total that is represented in a circular donut-shaped pie.
- **Exploded 2D Donut** - A donut-shaped pie chart with gaps between the slices. Displays the relative contribution of each Y-variable to the total that is represented in a circular donut-shaped pie.
- **Single-Slice 2D Pie** - Uses the pie chart to display only a single record of data in comparison to "100", another axis value, or a custom number.
- **Single-Slice 2D Donut** - Uses the donut chart to display only a single record of data in comparison to "100", another axis value, or a custom number.
- **3D Pie** - A 3-dimensional pie chart that displays the relative contribution of each Y-variable to the total that is represented in a circular pie.
- **Exploded 3D Pie** - A 3-dimensional pie with gaps between the slices. Displays the relative contribution of each Y-variable to the total that is represented in a circular pie.
- **3D Donut** - A 3-dimensional donut-shaped pie chart. Displays the relative contribution of each Y-variable to the total that is represented in a circular donut-shaped pie.
- **Exploded 3D Donut** - A 3-dimensional donut-shaped pie chart with gaps between the slices. Displays the relative contribution of each Y-variable to the total that is represented in a circular donut-shaped pie.

Single-Slice 2D

These types of charts use a custom data set, where only the first 2 rows of data can be part of the chart.

Using this example data set:

X	Y1	Y2
Record1	10	50
Record2	20	60
Record3	30	70

For Record1, by the default Y1 axis (10) is calculated against a 'From Data' value of 100, resulting in 10% pie slice on the chart. Select 'Custom' for 'From Data' and the value can be anything that is appropriate for the data set. Selecting 'Use Second Y-Axis', will calculate against Record2's Y1 axis (20), resulting in a 50% pie slice. Use the 'Transpose X/Y Axes' checkbox to calculate against Record1's Y2 axis (50), resulting in a 20% pie slice. Any records after Record2 are ignored for these chart types.



9.10 Pyramid and Funnel Charts

Note: Each height of an area represents the relative weight of each data point.

Note: Ratio charts can only display values greater than zero.

Note: Refer to pie charts for how clustering options work with the pyramid and funnel charts.

- **Pyramid** - Displays the relative contribution of each Y-variable to the total that is represented in a 2-dimensional pyramid.
- **Funnel** - Displays the relative contribution of each Y-variable to the total that is represented in a 2-dimensional funnel



9.11 Tabular Charts

All tabular charts have the option to set alignments (left, centered or right) for each individual column. Axis summaries, aggregation, sorting column headers and word-wrap are other shared properties within all tabular charts.

Tabular Chart – Displays all the values of X and Y variables in a multi-row, multi-column Tabular format. Applying a color to a numeric Y-axis is an option if upper and lower limits are enabled. A maximum of three colors is possible: numbers above the ‘upper limit’, numbers below the ‘lower limit’, and numbers in-between. All numeric Y-axes will use the same limits. Similar to the Metrics Scorecard chart, but with limited color options.

Report – Displays all the values of X and Y variables in a grid format with the ability to group and sub-group data rows. Option to collapse and expand rows of data groups. Note that you can click on a column header to sort the data on that column.

Scorecard – The same as a Tabular chart, including the ability to use a variety of shapes to visually indicate multiple range sets. Colored indicators are optional, and can be used on one or more numeric X or Y columns. See below for examples of each available shape.

Range Shapes				
Square	Arrow	Arrow Inv.	Triangle	Triangle Inv.
 & 	 & 	 & 	 & 	 & 

9.12 Calendar Charts

iDashboards includes a set of calendar charts that can be used to track events. The Calendar Chart group has four chart types; Monthly Calendar, Monthly Event Calendar, Weekly Calendar and Daily Calendar.

9.12.1 Data Layout

For a calendar chart to function properly the X-Axis has to be of type 'datetime'. The datetime data may or may not include the time. The Y-Axes of a calendar chart can be of type number, string or datetime.

In the sample data below, the 'Date_Time' column will be the X-Axis while the other four columns will be Y-Axes. Each row of data will be displayed in the calendar corresponding to the date in the 'Date_Time' column. A Monthly Event Calendar chart is used in this example.

Date_Time	Type	Deposit	Withdrawal	PostingDate
6/8/2020 11:57 AM	Deposit	5000		6/11/2020 6:46 AM
6/11/2020 8:51 PM	Deposit	1342.11		6/14/2020 12:51 PM
6/15/2020 8:37 PM	Multi	1382.34	-877.71	6/18/2020 11:26 AM
6/16/2020 4:59 AM	Multi	469.29	-882.58	6/18/2020 7:31 PM
6/19/2020 1:35 AM	Withdrawal		-1100.69	6/21/2020 5:19 AM

In all four calendar chart types, hovering your mouse over a date will pop-up a display that shows full data for that date. In order for the pop-up to work the "Show Mouse-over Value" in the 'Basic Settings' of the 'Chart Properties' needs to be checked.

9.12.2 Calendar Types

9.12.2.1 Monthly Calendar

The Monthly Calendar chart displays multiple months at one time. Each day within a month will display configurable colored shapes to show performance of one or more metrics against defined range sets.

Specific behaviors of the Monthly Calendar chart are as follows:

- The chart will map one row of data for each date cell. If there is a time element of the date, it will be ignored.
- If there are multiple rows of data with the same date, the chart will use the first date occurrence and ignore the others.
- Each date cell will display one colored square for each numeric Y-Axis. Hovering over the shape will display the data.
- Each date cell will display a triangle in the upper left when there are string-based Y-Axes present. Hovering over the triangle will display the values.

9.12.2.2 Monthly Event Calendar

The Monthly Event Calendar chart displays a monthly calendar with one or more metrics for each day. Its behavior is similar to the Monthly Calendar except that it only displays one month at a time and that all data is displayed in the date cell without the need to mouse-over.



9.12.2.3 Weekly Calendar

The Weekly Calendar chart displays the days of the week. Its behavior is similar to the Monthly Event chart except that you can control the number of days being displayed at a time.

9.12.2.4 Daily Calendar

The Daily Calendar chart displays one or more days at a time and can display multiple events for each day. Similar to the other calendar charts, it can display numeric, string and date data. Numeric data will display an associated colored square to show performance against defined range sets.

The Daily Calendar chart expects a time element in the date field. Every unique time within the day will display in the chart. If there are multiple rows of data with the same datetime, the chart will use the first datetime occurrence and ignore the others.

9.12.2.5 Calendar Navigation

All calendar charts provide a navigation feature to advance back and forth between dates. Depending on the chart type it will move the date forward or backward one period of time. For example, a Daily Calendar chart will move the chart one day while a Monthly Calendar chart will move the chart one month. Holding the 'Control' (CTRL) button on the keyboard while clicking the navigation button will advance multiple periods forward or backwards.

9.12.2.6 Calendar Holiday Settings

Calendar charts can display holidays using a custom color shaded effect on the day. The ability to define business-specific holidays is flexible and will depend on your data. Within the Calendar Chart properties, notice the option to set a specific color.

To connect your holidays with the chart, there needs to be a Y-axis column labeled 'isholiday' with Boolean values (1 or 0). The iDashboards chart will recognize the axis label 'isholiday' and will ensure the numbers 1 & 0 do not display within the chart; instead, the dates with a '1' will be colored with the 'Cell Holiday Color'. If the same date repeats within the dataset, either mark the holiday in the top-most position with "1", or mark all instances with "1".

9.12.2.7 Base Date/Position Selection

When opening a Calendar chart there may be many dates that are returned by the dataset. As part of the chart design, a determination will need to be made as to what set of calendar dates will be displayed first. The Calendar charts offer the ability to select a starting or ending date via three settings: "Base Position", "Custom Date", and "Base Type". These settings should be used in coordination with one another to achieve the desired result; Edit the chart, and within the Chart Designer navigate to 'Chart Properties'.

The "Base Position" will determine which date the chart will use when it initially loads. This setting provides several options to choose from:

- **End of Data** - The latest date available in the dataset.
- **Start of Data** - The earliest date available in the dataset.
- **Yesterday** - The day before the current date.
- **Today** - The current date.



- **Tomorrow** - The day after the current date.
- **Sunday through Saturday** - The nearest weekday, this includes the current date.
- **Custom Date** - The date entered in the "Custom Date" setting.

The "Custom Date" setting becomes available only when the Custom Date option is chosen from the "Base Position" setting. This date value must be entered in the YYYY-MM-DD date format. iDashboards date macros are not allowed in this field.

The "Base Type" setting has two options, Start Point and End Point. These options instruct the graph to use the selected base date/position as either the first or last cell in the Calendar display.

Note: The Monthly Calendar uses only the month and year portion of a base date. For example, the Monthly Calendar will interpret both 2009-12-01 and 2009-12-29 as a December 2009 base date.

9.12.2.8 Calendar Drilldowns

A Calendar chart can only be drilled into when there is data for the date clicked on. Clicking on a populated date will trigger the drilldown by sending the X-value (date and time) if a drilldown is configured on the chart.



9.13 Presentation Charts

Presentation charts are unique to most other charts within iDashboards. These charts have the ability to reference data, text and image-references in a graphical format rather than through a numeric-only chart. Their unique ability to display images using interactive intelligence allows dashboards to integrate data with image libraries within a single screen. When creating a presentation chart, the same creation method is used when compared to numeric graphs and charts: X & Y axis requirements, data source, mapping, etc. A requirement for presentation charts to function is to have data which matches the filename of an image file. It is preferred to have the file type extension within the data.

Example data (X or Y axis)	Example image filename
Building6	Building6.png
Building7.jpg	Building7.jpg
SusanneJennings	SusanneJennings.jpeg
Dennis_White.gif	Dennis_White.gif

9.13.1 Content Folder

Any images you wish to use within a presentation chart will first need to be uploaded to iDashboards through the Administrator Application. Images may be stored within sub-folders on the iDashboards server to help with file organization. See the iDashboards Administrator’s Manual for more information on uploading images.

Each presentation chart has the ability to dynamically display multiple images. The user will need to have a working knowledge of the location of images on the iDashboards server. Below are some examples of how some Image URL’s may be referenced.

‘Image URL Template’ Examples	
Administrator ‘Content’ location	URL within chart
Image at the root level (data has file extension)	content:\${value:Image}
Image at the root level (data does not have file extension)	content:\${value:Image}.PNG
Image in a subfolder	content:blueprints/\${value:Image}
Image in a deep subfolder	content:blueprints/2d/\${value:Image}
** The word ‘Image’ is an axis name (X or Y) within the chart	

Note: Image referencing is different with the Detail chart due to HTML syntax. See the Detail Chart section for additional details.



9.13.2 Content Macro (non-Detail chart)

iDashboards provides a content macro which allows you to replace the base URL for your iDashboards installation. The content macro makes referencing images easier. See below for examples on the standard format for using the content macro.

Example Introduction	Using the macro 'content:' will replace the iDashboards installation URL
This:	content:
Represents this:	<p><u>http://www.yourwebserver.com/idashboards/content/</u></p>  <p>Imagine 'your' URL for iDashboards</p>

Example 1	Start with the content macro and then suffix with an image name
Use this:	content:Building6.PNG
To represent this:	<u>http://www.yourwebserver.com/idashboards/content/Building6.PNG</u>

Example 2	Same as Example 1, but the image exists in a subfolder called 'blueprints'
Use this:	content:blueprints/Building6.PNG
To represent this:	<u>http://www.yourwebserver.com/idashboards/content/blueprints/Building6.PNG</u>

Example 3	Same as Example 2, but the image name comes from a chart value within the axis labeled "Image". This technique results in an interactive image.
Use this:	content:blueprints/\${value:Image}
To represent this:	<u>http://www.yourwebserver.com/idashboards/content/blueprints/Image Name</u>
	 <p>Imagine 'your' image name</p>

9.13.3 Missing Images

All presentation charts have the option to reference single or multiple images. If an image does not exist, or if a match does not exist between the data and the image directory, the chart will display a broken image icon. This icon does not indicate a broken chart, but it does represent a broken link between the chart and the image it was expecting to reference.



9.13.4 Gallery

The Gallery chart is used for displaying multiple thumbnail images within rows and columns. iDashboards will first determine the number of rows/columns and thumbnails displayed. Short and wide frames will generally display multiple columns and few rows. Tall and narrow frames will generally display few columns and multiple rows. However, the chart developer can customize each of these settings by configuring the 'Max Frame Row' and 'Max Frame Column' settings.

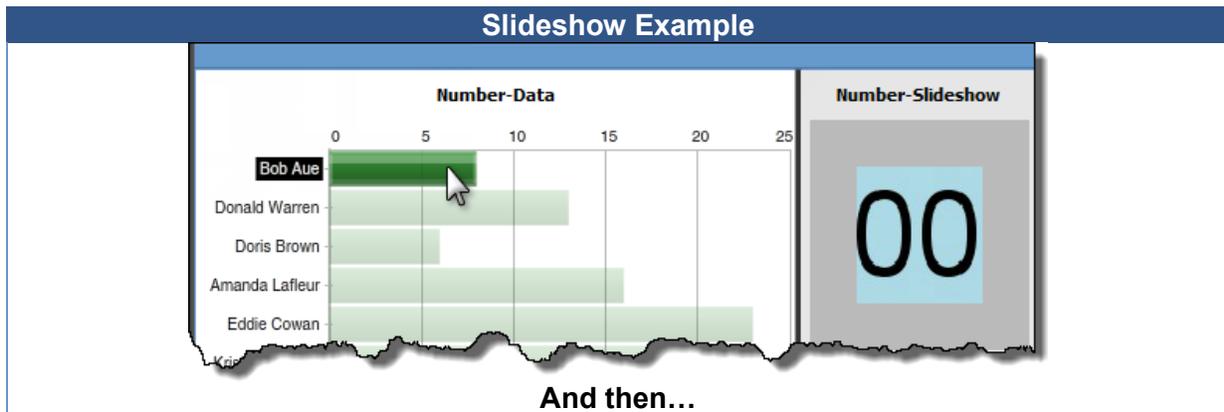
9.13.4.1 Scrolling

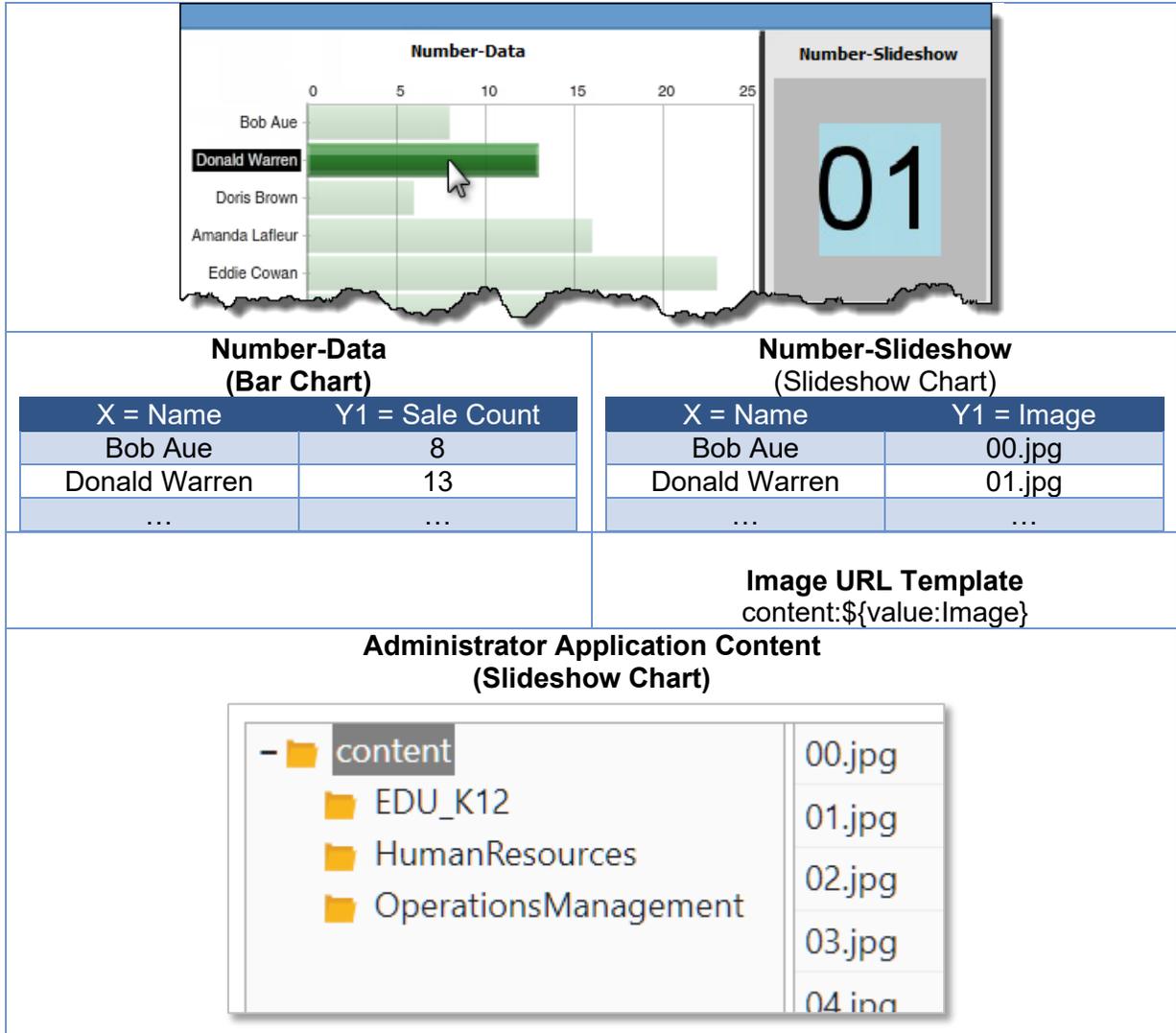
Because Gallery charts can link to hundreds of images, it may be necessary to scroll through various pages. The downward facing arrow indicates a Vertical scrolling direction. Clicking this arrow will scroll up/down one row at a time; CTRL-Click will scroll one page at a time, Shift-Click will scroll from beginning to end, and holding the click will auto-scroll at a speed determined within the chart properties. The chart developer has the option to flip the scrolling direction from Vertical to Horizontal, with the same scrolling functionality but using a left/right direction. Manual scrolling is not necessary if this chart is used with Interactive Intelligence from another chart.

9.13.5 Slideshow

The slideshow chart is used for displaying one image at a time. By default, the slideshow buttons are turned on, allowing the user to click on the 'Forward', 'Backward' or 'Play' buttons. This chart is unique because it can be set to 'Autoplay', potentially driving other charts with the use of interactive intelligence.

Through some customization, a slideshow chart can also display just a single image from a collection of images and automatically change based on interactive intelligence with another chart. Below, observe the relationship between the two charts, and then examine the data structure.





9.13.6 Details

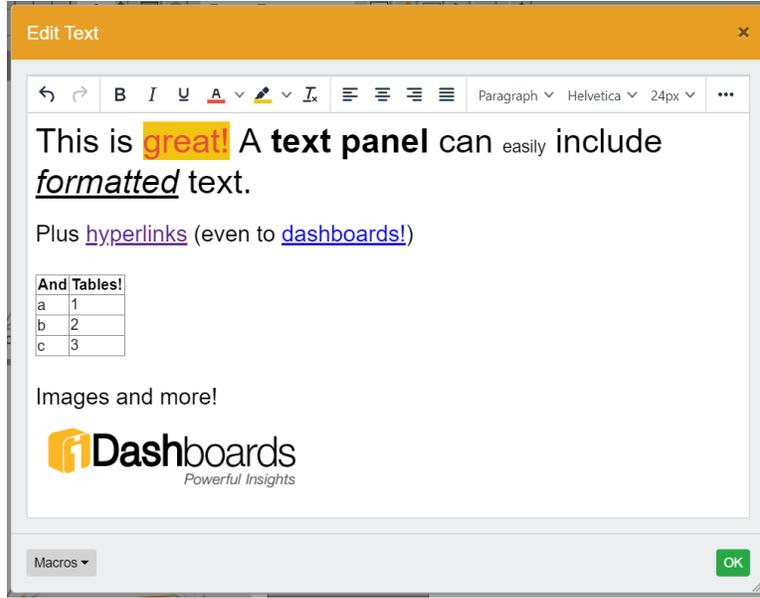
The Details chart can display a combination of chart data and unstructured content - including text and images. Detail charts have the unique ability to dynamically display data through a visually formatted frame. Users have the option to configure the content using “Rich Text” or “HTML”. Detail charts can show the same scrolling options as a Slideshow chart, or users can change data values with interactive intelligence.

9.13.6.1 Rich Text Frame Style

This option is the easy method for populating the Details chart. Rich Text can be used to display one or more lines of text within a frame. A number of options can control the appearance of the content such as: background color/transparency, text size/color/alignment, table options and more.

Note: To see additional options click the “...” button to expand the toolbar.





9.13.6.2 HTML Frame Style

Using this option will require knowledge of writing in the HTML language. An HTML frame can display formatted text, graphics or more, by using a subset of the HTML markup language. Click the “Edit” button and a new window will appear, containing a raw-HTML editor. A number of options can control the appearance of this panel such as: background color/transparency, scrollbars, margins, and more.

Although many HTML features are supported, the chart has may have limitations for advanced HTML elements.

Note: The Detail chart allows for use of the tag.

9.13.6.2.1 HTML Macros

Macros can assist with advanced implementations:

`\${user}` – for example, it is possible to populate a portion of the frame with the username of the person logged in. The example below shows the HTML value and the rendered dashboard panel when finished:

Using this HTML	Results in this
<code><p>Welcome `\${user}`</p></code>	Welcome admin

`\${content}` - If an image is needed within the context of an HTML frame, it is possible to link to a file located on the iDashboards server or a file on the internet. The HTML code can be copied from the example below (requiring an adjustment the image path) The text highlighted in yellow was placed into the panel by clicking on the macro button.

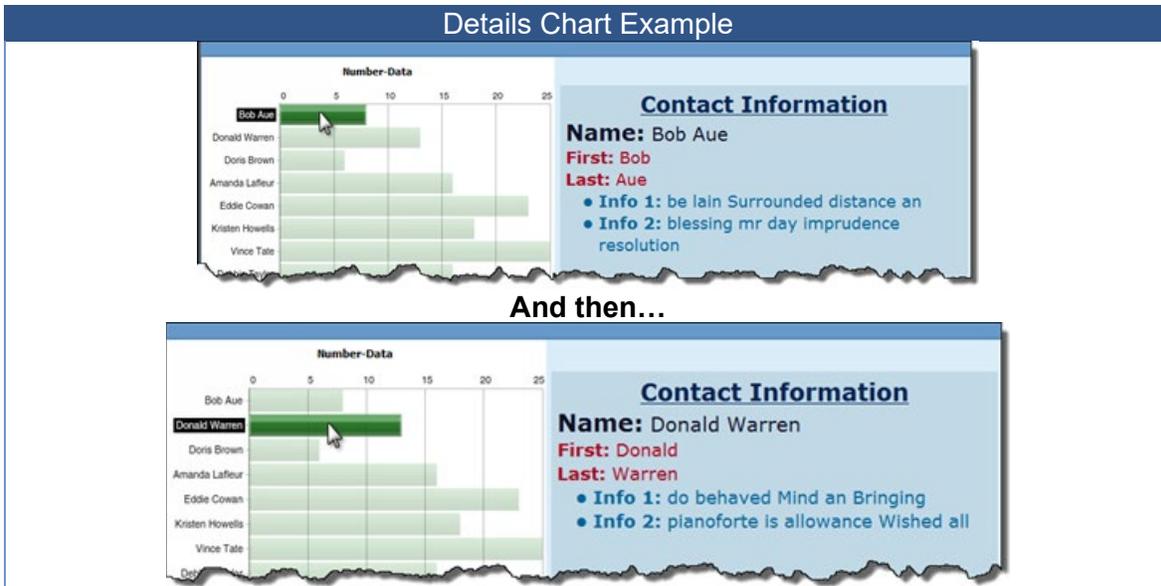
Using this HTML	Results in this
<pre><textformat leading="2"><p align="CENTER">The <u>HTML panel</u> is able to show <i>formatted text</i>, </p></textformat><textformat leading="2"><p align="CENTER">plus images!</p></textformat><textformat leading="2"><p align="LEFT"> </p></textformat><textformat leading="2"><p align="LEFT"></p></textformat></pre>	<p data-bbox="971 283 1419 338"><i>The HTML panel is able to show formatted text, plus images!</i></p> 

Note: Images may be stored within sub-folders on the iDashboards server. It may be necessary to designate the folder structure when using the content macro. Example: "{content}/Maps/States/Michigan.png".

Note: The macro {uecontent}, will un-escape the expanded value of the macro.

The following steps show the process to create a Detail chart to illustrate the connection between the chart, data, and HTML. The key steps are illustrated below, including a results table showing the usefulness of the chart. To set up a Details chart:

1. Create a chart that contains enough axes to contain your data
 - a. Observe the image below to understand the data for this example (Random text was used in the comment fields).
2. When choosing the chart type: Presentations > Details
3. Click on the 'Edit...' button for 'HTML Template'. Enter static text, and then use the function button to help place value macros.
4. When finished, the chart will appear with slideshow-type scrolling options, including the option to auto-play. Like all presentation charts, the values can change using interactive intelligence with other charts on the same dashboard.
5. Below, observes the relationship between the two charts.



9.13.6.2.2 Detail Chart Images

The Details chart does not use the same image referencing as the Slideshow or Gallery chart. Because the Details chart is essentially an HTML editor, the chart-builder must enter the HTML tag that will control image placement and properties within the chart.

The minimum requirement for placing an image into a Details chart, using the HTML option looks like:

```
<IMG SRC="{content}"/<image information">
```

...where the bold text above is replaced with image information. Then the chart-builder could modify the image tag to modify the image properties.

```
<IMG SRC="{content}/Building6.PNG">
```

Or...

```
<IMG SRC="{content}/blueprints/Building6.PNG">
```

Or...

```
<IMG SRC="{content}/{$value:Image}">
```

Or...

```
<IMG SRC="{content}/{$value:Image}" WIDTH="75" HEIGHT="75">
```

9.13.6.2.3 HTML Security

When user-created HTML is inserted directly into the HTML Details chart within iDashboards, steps are taken to insure that:

- The HTML cannot visually corrupt the iDashboards UI. For example, `<style>` tags could be used to alter existing CSS classes, therefore they are prohibited.
- No JavaScript can be executed. That would create a serious security vulnerability.



Step #1: Remove prohibited elements				
applet	element	html	object	template
base	embed	iframe	script	textarea
body	form	input	select	title
button	frame	link	shadow	
canvas	frameset	main	slot	
content	head	meta	style	

Note: An “a” tag with an “href” attribute that begins with “javascript” is also considered a forbidden element, and removed.

Step #2: Remove prohibited attributes			
onabort	ondragexit	onloadeddata	onreset
onautocomplete	ondragleave	onloadedmetadata	onresize
onautocompleteerror	ondragover	onloadstart	onscroll
onblur	ondragstart	onmousedown	onseeked
oncancel	ondrop	onmouseenter	onseeking
oncanplay	ondurationchange	onmouseleave	onselect
oncanplaythrough	onemptied	onmousemove	onshow
onchange	onended	onmouseout	onsort
onclick	onerror	onmouseover	onstalled
onclose	onfocus	onmouseup	onsubmit
oncontextmenu	oninput	onmousewheel	onsuspend
oncuechange	oninvalid	onpause	ontimeupdate
ondblclick	onkeydown	onplay	ontoggle
ondrag	onkeypress	onplaying	onvolumechange
ondragend	onkeyup	onprogress	onwaiting
ondragenter	onload	onratechange	

Note: Any forbidden attributes will have their value set to an empty string.

9.13.7 Infographic

The Infographic chart is an easy to use, all-inclusive label + graphic + value chart option within iDashboards. The chart displays a Y-Axis name and value along with a configurable images or shapes; and can easily accommodate one or more Y-Axes. When combined with interactive intelligence, this chart can dynamically display updated labels and values. The main advantage of using the Infographic chart versus a combination of other Presentation Charts (ex. Slideshow + Details) is access to the included library of over 400+ Clip Arts!

Similar to other Presentation Charts, the data requirements are unique because the Y-Axis values are not required to be numeric. Each Y-Axis in the chart can be defined by a label, value and graphic. With multiple Y-Axes, the chart will appear in clustered form. Clustering will be calculated by iDashboards on whether or not the graphics will be linear-aligned in a vertical or horizontal rectangular shape, or if the clustering will stack and stagger if the frame is square shaped.

Notes: Not all graphics can aesthetically accommodate all label placement options. Graphics cannot dynamically change or change based on the data.

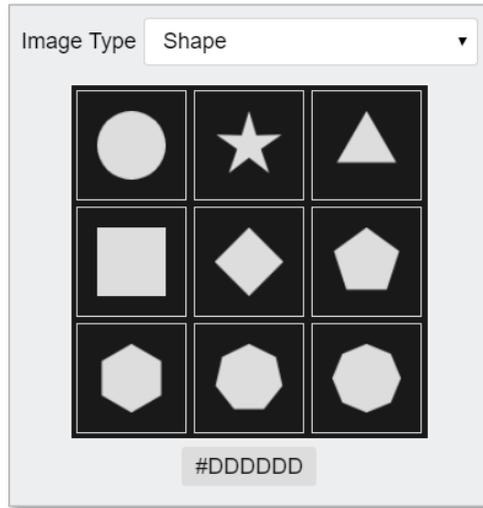


While Clip Art cannot be customized or changed, the Infographic chart can use any custom image available within the iDashboards 'Content' directory. The purpose of this chart is to display a graphic so once a graphic is selected it is not possible to un-associate a graphic with the axis.

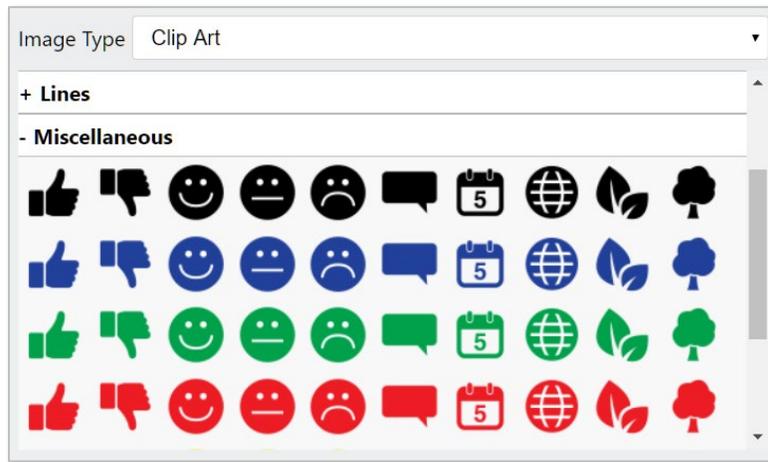
9.13.7.1 Axis Images

Within the Infographic Settings of the chart properties, an Axis Image can be selected for each Y-Axis. There are three (3) methods for associating the axis with a graphic: use a Shape, Clip Art or Image URL. To select the axis image, click the 'Edit' button associated to the axis.

- **Shape** – Nine (9) geometric shapes with an option to select a customized, fixed color.



- **Clip Art** – 400+ pre-made categorized graphics. When applicable, transparency is used in-and-around the detailed parts of the graphic.



- **URL** – This option allows the selection of any image within iDashboards, or the option to use a website URL referencing an image.

9.13.7.2 Axis Properties

Each Y-Axis can also be individually customized in regards to the text labeling and other settings. First, select the axis from the drop-down menu. Next, configure the various properties on how the graphic size and associated label will appear.

9.13.7.3 Non-Axis Properties

Non-axis properties are simply the remaining infographic properties not directly related to an axis. The X Value settings allow the chart to display the X-Axis value, which is often tied to other charts using interactive intelligence. For the 'Label Template', it is possible to use static or dynamic text.

9.14 Map Charts

There are two chart options for displaying map data within iDashboards. The GeoPlot Map chart and the Image Plot chart.

9.14.1 GeoPlot Map Chart

The GeoPlot Map chart is capable displaying data points upon a dynamic world map. The experience allows users to navigate the map using common panning and zooming. While increasing the zoom level, the map will display increasing details; even to the point of showing small streets or possibly even biking trails!

The GeoPlot Map chart is similar to the Image Plot chart in many ways. However, the GeoPlot Map chart uses a pre-defined dynamic map for the background layer. The Image Plot chart uses a custom-defined, static, raster image for the background layer.

Note: If the GeoPlot Map is not an available chart type, please contact your iDashboards administrator or iDashboards account manager.

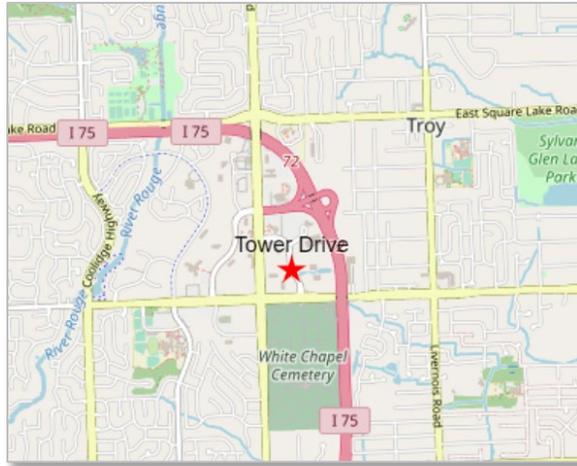
9.14.1.1 Concept

There are 2-primary layers to the GeoPlot Map. The background layer consists of a dynamic map and the foreground layer provides an infrastructure to overlay points. Similarly, the Image Plot chart is a 2-layer chart.

Like most charts within iDashboards, the GeoPlot Map is a combination of data with an illustration, to provide a dynamic experience to relay information visually. The GeoPlot Map chart allows latitude and longitude (lat/lon) points to be plotted on top of a map. iDashboards hosts a map tile server and the connection is pre-configured within the chart. The points that overlay the map will illustrate geographic locations for things like cities, buildings, events, or more.

Data must be provided in the format of latitude and longitude. When lat/lon data is provided, the intersecting coordinate will result in a displayed point. Additional details within this section will describe a multitude of ways to make various graphics appear on the map aside from lat/lon data points.





9.14.1.2 Requirements

Because this chart relies upon an additional network connection to a map tile server, some requirements are necessary for the chart to operate.

9.14.1.2.1 Point Definition

The point-layer of the GeoPlot Map will be the interactive component. Points must be defined using Latitude and Longitude coordinates.

- Method for locating each point
 - From data set
 - From XML configuration file
 - If using XML as a data source is desired, then users need a basic text/XML editor capable of reading and saving Text or XML Documents that are UTF-8 encoded
- Method for defining the properties of each point
 - From Chart Properties (globally assigns properties to all points)
 - From data set (allows unique point-property definition)
 - From XML configuration file (allows unique point-property definition)

9.14.1.2.2 Technical

Contact iDashboards Support to discuss these details:

- Maintenance and Support is required to continue using this chart
- Internet connection from the client to the iDashboards map tile server
 - If this is not possible, then an internet connection from the client to a non-iDashboards map tile server (ex. OpenStreetMaps®, Google®, etc.)
 - If this is not possible, then it will be necessary to internally host a map tile server on the intranet



9.14.1.2.3 License to enable this chart

Contact your iDashboards administrator for details on licensing.

- The GeoPlot Map chart requires iDashboards v9.1 or later
- A new license file specifically created for v9.1 or later

9.14.1.2.4 Licensed to use iDashboards Map Tile Server

While all licensed versions of iDashboards v9.1 will have this chart, there are two settings which control the ability to connect to the iDashboards map tile server:

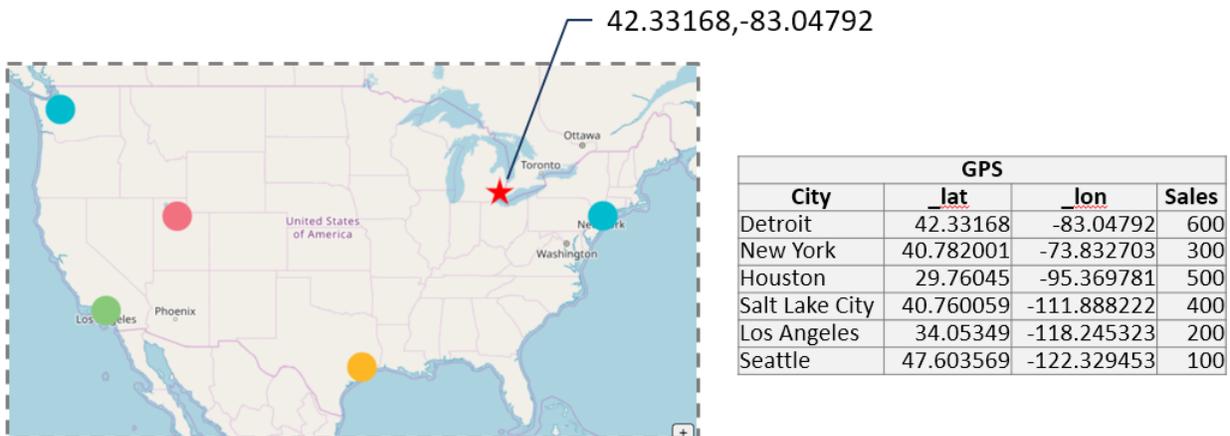
1. This chart CAN use the iDashboards map tile server
2. This chart CANNOT use the iDashboards map tile server

Note: If the iDashboards map tile server is not an available option, please contact your iDashboards administrator or iDashboards account manager.

9.14.1.3 Methods for Defining Coordinates

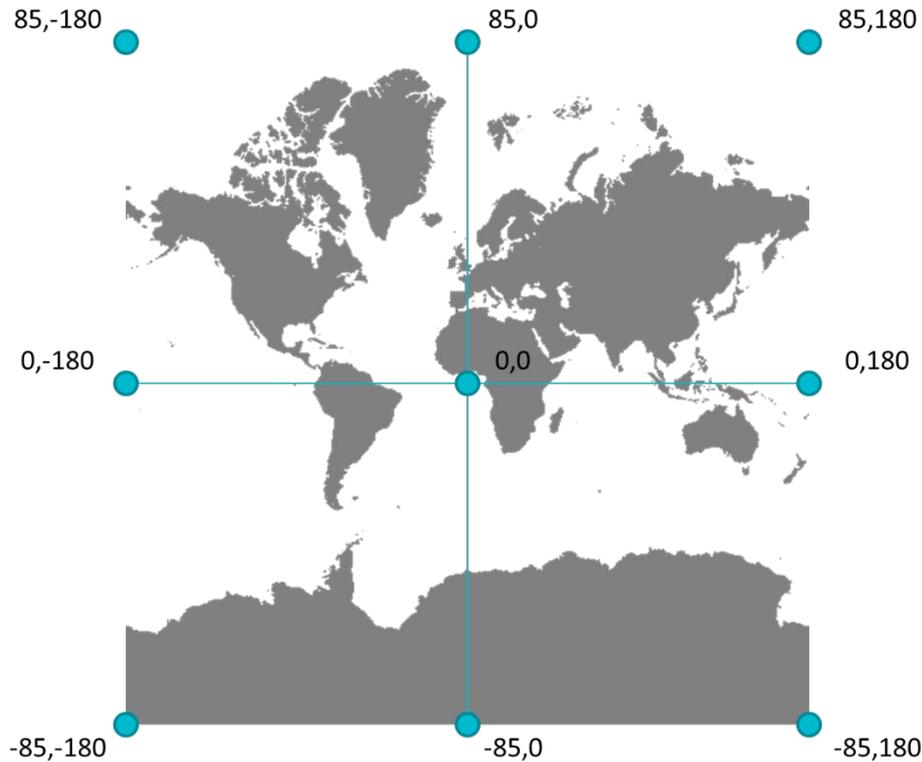
Data must be provided in the format of latitude and longitude. When provided, the intersecting coordinate will result in displayed point. One row of data can only contain the coordinates for one point. Therefore, to plot 27 points on the map will require the chart dataset to be at least 27 rows.

- **Latitude and Longitude Decimal Degree** – This method requires latitude and longitude values, in decimal degree format, for each point.
 - Ex. “25” indicates either the 25th latitude or longitude coordinate on the globe.
 - **Latitude** - Valid latitude range is: -85.0511287798066 to 85.0511287798066
 - Approx: -85 to 85
 - **Longitude** - Valid longitude range is: -180 to 180
- Latitude and longitude values will need decimal places to achieve accurate point location, particularly if zoom level is showing details



9.14.1.3.1 Latitude and Longitude (lat,lon) world extents

Below illustrates the world map and the basic latitude/longitude coordinates considering (0,0). All points on the GeoPlot Map chart will be accurately located at the exact geographic location based on this coordinate system.



Note: -85.0511287798066 and 85.0511287798066 approximated as -85 and -85

9.14.1.3.2 Manually register a coordinate (SHIFT+CLICK)

While there may be alternative methods for identifying the coordinates of a point, this iDashboards feature was built into the GeoPlot chart to assist with identifying the location of points if these values do not exist within the data. Capturing points can be used for a single click or for multiple clicks, as the results will be temporarily stored in the clipboard. For this feature to work, enable the property 'SHIFT-CLICK Copies Location'. Next, review the 'Click Location Template' and leverage static text, along with the following macros, to construct the syntax for capturing information. Next, leave the properties window. Position the map view to allow capturing one or all of the necessary points, hold down SHIFT, place the cursor at the desired location, and CLICK!

- **`\${lat}** - The macro to record the latitude coordinate
- **`\${lon}** - The macro to record the longitude coordinate
- **`\${tlat}** - The macro to record the latitude coordinate of the containing tile (upper-left corner)
- **`\${tlon}** - The macro to record the longitude coordinate of the containing tile (upper-left corner)



A Click Location Template like this:

```
<latlon>${lat},${lon}</latlon>
```

Can result in a clipboard like this (formatted for XML):

```
<latlon>42.33155973336517,-83.04653087959565</latlon>
```

```
<latlon>42.3289581796185,-83.039750255206</latlon>
```

```
<latlon>42.32616614858348,-83.04711023674287</latlon>
```

```
<latlon>42.3368894098674,-83.05086532936372</latlon>
```

9.14.1.4 Y-Axis Label Names: Point Positioning

Using the GeoPlot Map within iDashboards requires specific axis label names to position the points on the chart. When labels are created using the options listed below, the chart will expect data in that specific format.

Method for locating each point

- From data set
 - Require specific Y-Axis label names
- From XML configuration file
 - Does not require specific Y-Axis label names
 - If using XML as a data source is desired, then users need a basic text/XML editor capable of reading and saving Text or XML Documents that are UTF-8 encoded
 - Points defined by XML require the chart to have matching X-Axis values
 - Markers defined by XML do not have data requirements

There are two axis-labeling options for positioning points using latitude and longitude coordinates. The underscore prefix is a required character.

Two (2) Y-Axes	
_lat	A decimal, numeric value to position the point along a latitude
_lon	A decimal, numeric value to position the point along a longitude

One (1) Y-Axis	
_latlon	A comma-separated string value, containing the decimal/numeric value to position the point at a latitude and longitude coordinate

Note: The data-column name does not need to follow this naming convention

9.14.1.5 Y-Axis Label Names: Point Properties

The GeoPlot Map within iDashboards has a variety of options to customize the appearance of the points within the map. When labels are created using the options listed below, the chart will expect data in that specific format.



- Method for defining the properties of each point
 - From Chart Properties (globally assigns properties to all points)
 - Does not require specific Y-Axis label names
 - From data set (allows unique point-property definition)
 - Requires specific Y-Axis label names
 - From XML configuration file (allows unique point-property definition)
 - Does not require specific Y-Axis label names

Note: The data-column name does not need to follow this naming convention.

Note: There are two methods for ignoring Point Properties: Remove the underscore or delete the axis.

Point Properties	
Color _color	<p>Visual properties to alter the appearance of individual points Defines the color of a point. HEX (as string) or Decimal value (as number). Example Values:</p> <ul style="list-style-type: none"> • #FF0000 (String equals Red) • 0xFF0000 (String equals Red) • 16711680 (Number equals Red) <p>Notes: A chart property exists to apply a specific color to ALL undefined points.</p>
Shape _shape	<p>Defines the shape of a point. Values include:</p> <ul style="list-style-type: none"> • circle • square • star • poly_#_# <p>Poly must include elements to define the number of sides and the rotation angle. Sides can range from 3-8 and angle can range from 0-359, as in poly_[3-8]_[0-359] (ex. poly_3_45)</p> <p>Notes: A chart property exists to apply a specific shape to ALL undefined points, but lacks the full variety of polygon shapes and angle rotation.</p>
Size _size	<p>Defines the size, in Pixels. Values include:</p> <ul style="list-style-type: none"> • 5-100 (as numeric) <p>Notes: A positive, numeric value to resize a point. A chart property exists to apply a specific size to ALL undefined points.</p>
Blink	Defines the blink capabilities, in milliseconds



<code>_blink</code>	Determines the blinking state of a point. <code>_blink</code> is the duration (in milliseconds) that a point is in the visible state. If values are NULL, the point will not blink.
<code>_blinkfactor</code>	<code>_blinkfactor</code> is the factor of the <code>_blink</code> value in defining how much time to spend in the invisible state. For example, a blink factor of 1.0 will make the point invisible for the same duration as defined in the <code>_blink</code> axis. Using a blink factor of 0.5 will make the point invisible for a duration half of the amount defined in the <code>_blink</code> axis. If left blank, the blink factor will use "1".
	Notes: Any positive, numeric values are allowed, but the blinking will never be less than 100ms, regardless of the provided value.

9.14.1.6 Consider the Data Source

- The GeoPlot Map chart only positions points when latitude and longitude coordinates are provided in the numeric, decimal format
- Geocoding based on addresses or ZIP code is not possible without pre-processing the information prior to iDashboards, this is referred to as Geocoding, and can be processed when using the ETL task 'Geocode'.
- Empty values result in no displayed point. There is no warning when this occurs.

First, emphasis must be placed on how the dataset will contain latitude and longitude coordinates. Next, consider how the points should appear, on the map, using visual properties.

In a simple scenario, Excel would be used as the type of data source because of the easy data structuring options. To achieve the minimum requirements for the chart the data source should have two or three columns. Later, the document discusses additional chart properties which can be defined using additional Y-axes.

The X-axis would be any pertinent data for defining the point on the image. This could be data for display on the chart such as City, Location or Name in a geographical map. The Y-axes for the point on the image must be X and Y relational coordinates. The tables below represent a dataset for locating specific cities in the United States.

Below, are three tables providing examples of a dataset containing latitude and longitude coordinates within a dataset for locating specific cities in the United States. Each dataset will position the points in the exact same location on the map.

This table provides an example of latitude and longitude coordinates each being assigned to a Y-Axis.

GPS			
Axis Position	X axis	Y1 axis	Y2 axis
Data Type	String	Number	Number
Axis Label	City	<code>_lat</code>	<code>_lon</code>
	Detroit	42.33168	-83.04792
	New York	40.782001	-73.832703



Houston	29.76045	-95.369781
Salt Lake City	40.760059	-111.888222
Los Angeles	34.05349	-118.245323
Seattle	47.603569	-122.329453

This table provides an example of latitude and longitude coordinates being combined into a single Y-Axis.

GPS		
Axis Position	X axis	Y1 axis
Data Type	String	String
Axis Label	City	_latlon
	Detroit	42.33168, -83.04792
	New York	40.782001, -73.832703
	Houston	29.76045, -95.369781
	Salt Lake City	40.760059, -111.888222
	Los Angeles	34.05349, -118.245323
	Seattle	47.603569, -122.329453

This table provides an example of latitude and longitude coordinates each assigned to a Y-Axis, along with aesthetic properties for each point also being defined by the data.

GPS						
Axis Position	X axis	Y1 axis	Y2 axis	Y3 axis	Y4 axis	Y5 axis
Data Type	String	Number	Number	String	String	Number
Axis Label	City	_lat	_lon	_color	_shape	_size
	Detroit	42.33168	-83.04792	#FF0000	star	14
	New York	40.782001	-73.832703	#00BBCE	circle	10
	Houston	29.76045	-95.369781	#FDB724	circle	10
	Salt Lake City	40.760059	-111.888222	#F27180	circle	10
	Los Angeles	34.05349	-118.245323	#87C979	circle	10
	Seattle	47.603569	-122.329453	#00BBCE	circle	10

9.14.1.7 GeoPlot Map Settings: Chart Designer

- Colors can be defined via data column



- Colors can be defined with a global property
- Colors can be defined via numeric value and Range Colors

9.14.1.7.1 Range Colors (Single Numeric Y-Axis)

If 'Range Colors' is enabled, the point colors will be identified with the colors defined within the chart Range Colors.

Show Y-axis Name – Will display the numeric Y-axis label name beneath the graphic.

Legend Priority – When Range Colors and an XML configuration file are used, see the property "Legend Priority" to determine which color option should have priority.

9.14.1.7.2 Point Label Template

Any axis value from the dataset or static value can be used as the visual label associated with each point. When left blank, the point will display the X-Axis value. The syntax defined here will be applicable for all points.

Tip: Enable "Point Labels Always Visible" to quickly preview the Point Label Template settings while developing the label template.

- Macro – Label values can be static values, or can come from a variety of dynamic macros. Clicking on the macro button '\${}' will list the following options as described below.

Macro	Description
<code>\${selected:axis}</code>	Label display would be 'Day Op Cost' or 'Strength' Requires multiple numeric Y-axes and is used in conjunction with enabling "Use Range Colors".
<code>\${selected:value}</code>	Label display would be '875' or '10,000' Requires multiple numeric Y-axes and is used in conjunction with enabling "Use Range Colors".
<code>\${value:<x>}</code>	Label would display the X-Axis value
<code>\${value:<y>}</code>	Label would display the designated Y-Axis value
<code>\${user}</code>	Label display would be '<current logged in user>'
<code>\${charttitle}</code>	Label display would be '<current chart title>'
<code>\${chartid}</code>	Label display would be '<current chart ID>'
<code>\${categoryid}</code>	Label display would be '<current category ID where chart is saved>'
<code>\${br}</code>	Use this 'Break Line' macro to add multi-line labeling

9.14.1.8 Navigation

Common to similar map applications, the GeoPlot Map has an intuitive experience for navigating on the screen. Based on the device type, it may be necessary to use touch-screen techniques versus mouse-pointer.

For improved performance, iDashboards will 'load' and 'unload' tiles based on the minimum necessity for the viewport. This process frequently occurs while zooming and panning. While zooming or panning, it is possible for the map image tiles to appear as 'unavailable'. This is a temporary state. It indicates the tile server does not currently have the requested tile. The wait time should be minimal (1-30 seconds) before the image appears on the screen. Wait time in



excess of 30-seconds are likely due to excessive new-tile requests or poor connectivity speed to the internet.

9.14.1.8.1 Pan

Panning is always allowed. Traditionally, data points will automatically be displayed when the GeoPlot Map is displayed, but panning allows the user to see areas around the map points. Using either click-and-drag or 'finger' swiping, it is possible to navigate in all directions in and around the map, even to areas where no data points appear. While panning, iDashboards will display a red crosshair to indicate the center of the map prior to panning and a black crosshair to indicate the new center of the map.

9.14.1.8.2 Zoom

Generally speaking, zooming is the processes of increasing or decreasing the zoom level of a map. The zoom range is from 0-54 (for a total of 55 zoom levels) where '0' shows the entire world map in a small 256x256px tile and '54' shows the closest, and most detailed street level detail.

Zoom techniques include:

- **Mouse wheel scrolling** – Zooms in/out based on cursor location.
- **Finger pinch** – Zooms in/out based on mid-point of fingers (on some PC's with touchscreen).
- **Button** – Zooms in/out based on the center of the viewport when the "+" or "-" buttons are clicked. While the cursor is above the buttons a crosshair appears at the center of the viewport, the zoom level is displayed and the lat, lon coordinates are displayed (for the crosshair).

Regardless of the device or technique, zooming may not always be allowed, as described by two explanations: First, there are chart properties which can restrict the minimum and maximum zoom levels of a map. Second, configuring iDashboards to use a different map tile server may result in using a map with limited or only one zoom level.

9.14.1.9 XML

Using a configuration XML file is an advanced option and not a chart requirement. Items defined by XML must be positioned in the format of latitude and longitude. Unique capabilities of using a configuration XML file include:

- Cosmetically enhance the GeoPlot Map experience
- Add points to the map when the latitude and longitude don't exist within the chart data
- Add points/markers, labels, leaders and images
- Make items appear or disappear at a particular zoom level
 - Labels
 - Leader lines
 - Images



- Points (interactive with cursor)
- Markers (not-interactive with cursor)

Note: Items displayed within the GeoPlot Map are not part of the chart dataset, and therefore do not export or appear within reports.

9.14.1.9.1 XML Tree structure

On the XML file, the first line should include a standard XML declaration, defining the XML version (1.0). The next line is the required root element `<geoplot version="1.0">` and the last line must be the closing root element `</geoplot>` (as shown below).

Line 1	<code><?xml version="1.0" encoding="UTF-8"?></code>
Line 2	<code><geoplot version="1.0"></code>
Line 3-#	<code><!--XML content goes here--></code>
Last Line	<code></geoplot></code>

9.14.1.9.2 XML Tags

The most complex function of the XML Configuration file includes the syntax and structure of tags and attributes.

Tags

```

<objects></objects>
<object></object> <!--append attributes then repeat for each group-->
<marker></marker> <!--repeats for each marker-->
<latlon></latlon> <!--comma-separated numeric coordinates-->
OR <lat></lat> AND <lon></lon> <!--a numeric coordinate-->
<shape></shape> <!--a shape value-->
<size></size> <!--a number between 0-100-->
<color></color> <!--a color value-->

<label></label> <!--repeats for each marker-->
<text></text> <!--any text value-->
<align></align> <!--"center", "left" or "right"-->
<color></color> <!--a color value-->
<line></line> <!--"true" or "false"-->
<vo></vo> <!--a number (pixel) value to vertically offset from top-
left corner-->
<ho></ho> <!--a number (pixel) value to horizontally offset from top-
left corner-->

<images></images>
<image></image> <!--repeats for each image-->
<tile></tile> <!--"true" or "false" value if image should be fixed to
the tile-->
<url></url> <!--path to image file-->
<vo></vo> <!--a number (pixel) value to vertically offset from top-
left corner-->
<ho></ho> <!--a number (pixel) value to horizontally offset from top-
left corner-->

<points></points>

```



```

<point></point>
<value></value>  <!--any X-Axis value-->
<latlon></latlon> <!--comma-separated numeric coordinates-->
OR <lat></lat> AND <lon></lon> <!--a numeric coordinate-->

<legend></legend> <!--only one-->
<items></items>
<item></item>
<icon></icon>
<shape></shape> <!--a shape value-->
<size></size> <!--a number between 0-100-->
<color></color> <!--a HEX color value in "0x" (zero x) notation-->

```

Attributes

```

zmin="" <!--a number between 0-54-->
zmax="" <!--a number between 0-54-->
tile="" <!--true or false-->
marker-shape="square" <!--a shape value-->
marker-size="" <!--a number between 0-100-->
marker-color="" <!--a HEX color value in "0x" (zero x) notation-->
label-size="" <!--a number between 0-100-->
label-color="" <!--a HEX color value in "0x" (zero x) notation-->

```

9.14.1.9.3 XML Example

```

<objects> <!--this tag can repeat-->
  <object zmin="0" zmax="25">
    <value/>
    <color/> <!--also applied to line-->
    <size/>
    <x/> <!--or <xpct></xpct>-->
    <y/> <!--or <ypct></ypct>-->
    <line>
      <x/> <!--or <xpct></xpct>-->
      <y/> <!--or <ypct></ypct>-->
    </line>
  </object>
  <label/> <!--repeat for additional labels-->
</objects>

```

9.14.1.10 Priority Notes

The GeoPlot Map chart has multiple areas in which to define multiple items. For example, point positions can be defined by the data or through an XML Configuration file. In circumstances where values are defined by two or more techniques, the following rules apply:

- If Range Colors, and an XML configuration file are used, see the property “Legend Priority” to determine which color option should have priority.
- Point Properties defined by data will override properties available within the Chart Designer.
 - **Priority 1:** (only for Color properties): Range Color, if Range Colors are configured to be used



- **Priority 2:** Data Column
- **Priority 3:** XML Configuration File
- **Priority 4:** UI Properties within Chart Designer

9.14.1.11 Map Tile Server

A map tile server hosts a gigantic number of small static images (256x256 pixel PNG files) that, when automatically stitched together in a tiled format, represent an illustration of a certain geographic area of a map - at a certain zoom level.

- http://wiki.openstreetmap.org/wiki/Slippy_map_tilenames#Resolution_and_Scale
- Tiles are 256 × 256 pixel PNG files
- Each zoom level is a directory, each column is a subdirectory, and each tile in that column is a file
- Filename(url) format is /zoom/x/y.png
- Mercator projection

For the mapping experience in the GeoPlot Map chart, iDashboards hosts an OpenStreetMap® map tile server. OpenStreetMap is open data, licensed under the Open Data Commons Open Database License (ODbL) by the OpenStreetMap Foundation (OSMF), as described here:

<http://www.openstreetmap.org/copyright>

This map is a 'street' map that also shows Points of Interests (POI); both are displayed based on the zoom level.

9.14.1.11.1 Updates to map

Changes, suggestions, or corrections, to the map graphics cannot be reported to iDashboards. If changes, suggestions, or corrections are necessary, OpenStreetMaps is a 'community driven' technology and allows anyone to submit changes online or host their own map tile server. Information about this process can be found at:

<http://www.openstreetmap.org/about>

<http://www.openstreetmap.org/help>

9.14.1.11.2 Considerations for using a different map tile server

Outside the control of iDashboards, it is possible to heavily customize the GeoPlot Map to leverage various map tile server technologies. Configuring a different map tile server:

- may negate default values discussed within this documentation
 - ex: some map tile servers only host zoom levels 10-30
- may result in different appearances from the screenshots used within this documentation
 - ex: some map tile servers allow different styles of maps like aerial, dark, light, black/white, etc.



- may require additional customization of the 'Tile URL Template'

9.14.1.11.3 Tile URL Template

Configuring the GeoPlot Map chart to use a different map tile can be performed within the property 'Tile URL Template'. The goal for this field is to connect to a URL where images can be streamed and tiled into a map. However, it will be necessary to position the tile macros into the correct location within the URL. Because all tile servers are different, there is no definitive URL structure.

- [] - The macro to allow switching between different servers based on server availability
- \${z} - The macro to define the zoom level
- \${x} - The macro to define the x-position of the image within the viewport
- \${y} - The macro to record the y-position of the image within the viewport

Example:

```
http://[ab].tile.exempletileservname.com/geodata/${z}/${x}/${y}.png
```

9.14.1.12 GeoPlot Map Technical Tips

Information in this section is new, or continued, from topics of the GeoPlot Map chart. Refer to the entire GeoPlot Map section to understand the basic functionality.

9.14.1.12.1 Zoom Tips

The zoom range is from 0-54, for a total of 55 zoom levels. Technically, these zoom levels are comprised of 19 Coarse zoom levels and 36 Fine zoom levels. The Coarse zoom levels are in multiples of 3 (0, 3, 6, 9...) and are indicated with square brackets around the zoom number as shown in the image below. The 36 Fine zoom levels simply enlarge (zoom) the Coarse zoom map images and therefore may result in slight pixilation.



9.14.1.12.2 Geocoding

The process of converting logical addresses into latitude/longitude coordinates is called Geocoding. For example, geocoding the address "1600 Pennsylvania Ave., Washington, DC 20500" will translate into "38.897674, -77.036528". The manual process for registering a coordinate through the GeoPlot Map chart interface was discussed earlier (SHIFT+CLICK). However, there are options across the internet to automate the conversion, even if conversion requests are reoccurring. For support with implementing a geocoding solution, contact your iDashboards account manager.

9.14.2 Image Plot Chart

The Image Plot chart is a customizable chart, unique to most other charts within iDashboards. The chart is developed using the common chart building process, and all features within iDashboards can be integrated into this chart (Interactive Intelligence, Pivots, Drilldowns, Input Parameters, etc.). However, unlike traditional data sets for numeric charts, the Image Plot requires specific data structuring to function.

The Image Plot chart is similar to the GeoPlot Map chart in many ways. However, the GeoPlot Map chart uses a pre-defined dynamic map for the background layer. The Image Plot chart uses a custom-defined, static, raster image for the background layer.

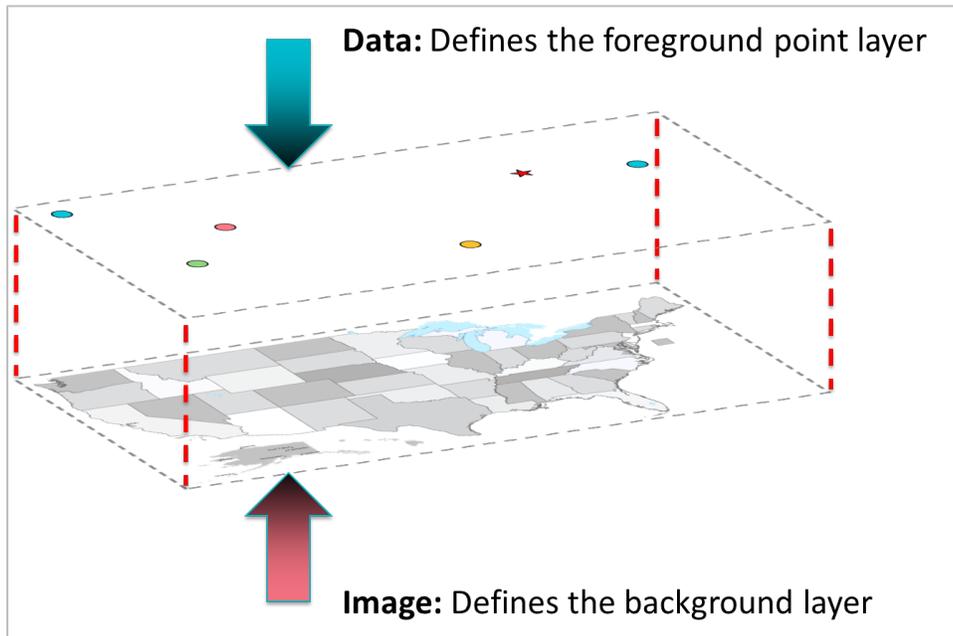
9.14.2.1 Concept

There are 2-layers to the Image Plot. The background layer consists of a raster image and the foreground layer provides an infrastructure to overlay points. Similarly, the GeoPlot Map chart is also a 2-layer chart.

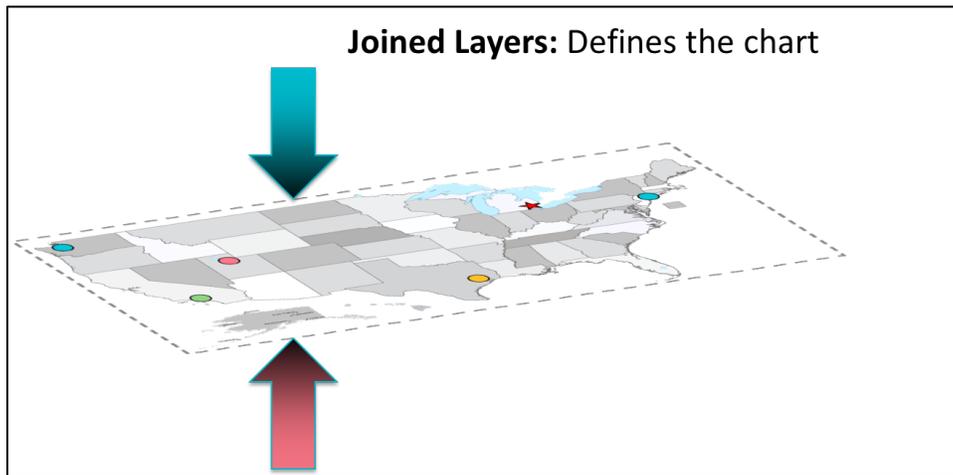
Like most charts within iDashboards, the Image Plot is a combination of data with an illustration, to provide a dynamic experience to relay information visually. The Image Plot chart allows custom points to be plotted over an image file. Users will be able to supply images of anything, like geographies, building floor plans, or process steps; the options are endless. The points that overlay the image will illustrate things like cities, locations, or tasks; these options are endless too.

Supporting files are required to use the Image Plot chart. The files can be uploaded to the installation of iDashboards and be used an unlimited number of times within any dashboard. The two typical components include: a raster image for the background and the coordinate data for positioning points.

Below illustrates an angle to help identify the conceptual 2-layers:



Below illustrates the 'joined' effect from the conceptual 2-layers:



Below illustrates a real screenshot of the final Image Plot chart, as seen within a dashboard.



9.14.2.2 Requirements

Users should have a firm understanding of the chart building process and how to access images within the 'Content' directory. Administrator access to iDashboards may be helpful.

- **Raster Image:** Image Plots can be created using any image file, available for use within iDashboards.
- **Point Definition:** The point-layer of the Image Plot will be the interactive component.
 - Method for locating each point
 - From data set



- From XML configuration file
 - If using XML as a data source is desired, then users need a basic text/XML editor capable of reading and saving Text or XML Documents that are UTF-8 encoded
 - Configuring an XML file is required if GPS coordinates are used
- Method for defining the properties of each point
 - From Chart Properties (globally assigns properties to all points)
 - From data set (allows unique point-property definition)
 - From XML configuration file (allows unique point-property definition)

9.14.2.3 Methods for Defining Coordinates

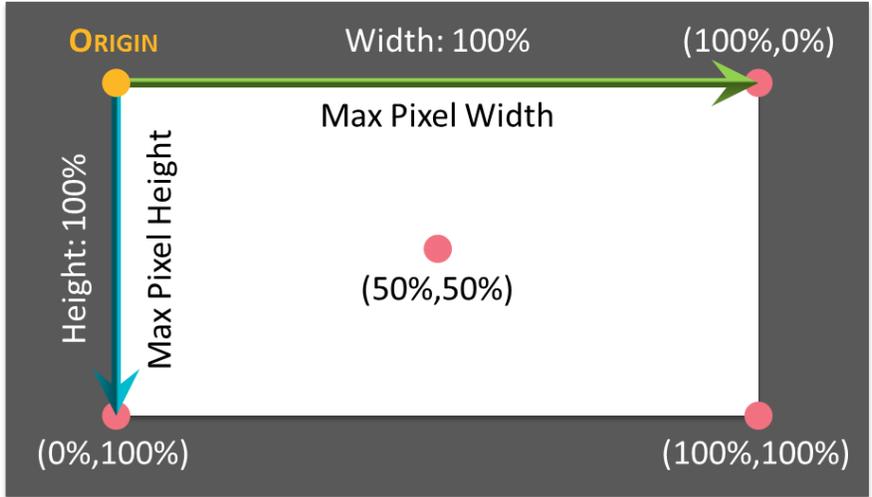
There are three methods to define the coordinates for a plotted point on an image. For all methods, image scaling is automatically managed by iDashboards.

- **Pixel** – This method requires the data to provide the X and Y position for each point. Values are calculated using the image resolution with the origin at the upper-left corner. Ex. “25” indicates 25-pixels
- **Percent** – This method requires the data to provide the X and Y percentage for each point. Values are calculated using the image width and height with the origin at the upper-left corner. Ex. “25” indicates 25%
- **Latitude and Longitude Decimal Degree** – This method requires latitude and longitude values, in decimal degree format, for each point. This can be used with any geospatial raster image of ‘Equirectangular’ or ‘Plate Carrée’ projection. Ex. “25” indicates either the 25th latitude or longitude coordinate on the globe.

9.14.2.3.1 Pixel and Percent positioning concept

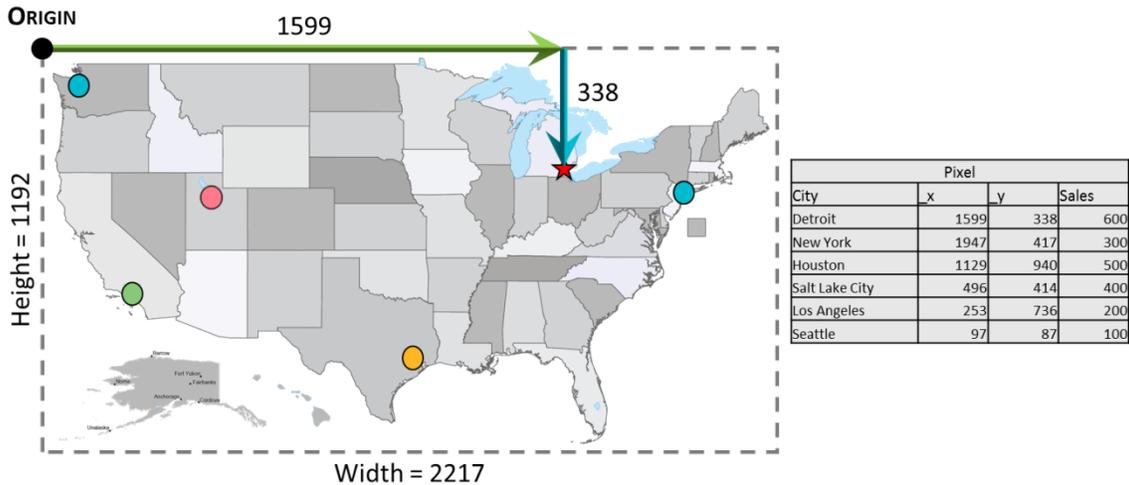
The simplest method for locating points is either Pixel or Percent. The image below shows the origin of an image when determining the point location. The origin, in the top-left corner, will always be (0,0) and then each point can be placed over any area of the rectangle boundaries of the image.





9.14.2.3.2 Detroit by Pixel example

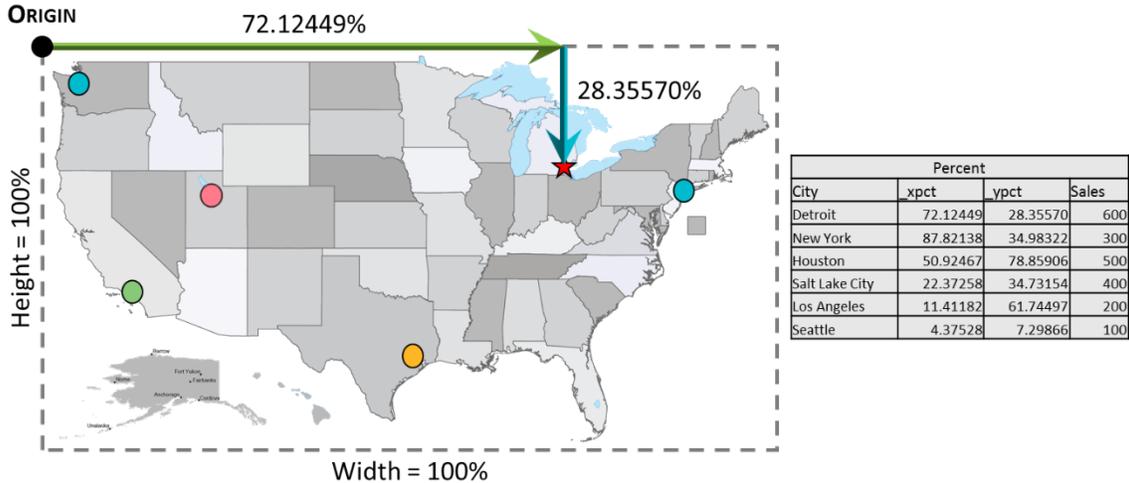
Below shows an example dataset, and how a point, representing Detroit, is positioned using pixel dimensions. The dimensions for locating Detroit in this example are specific to the image and image resolution.



9.14.2.3.3 Detroit by Percent example

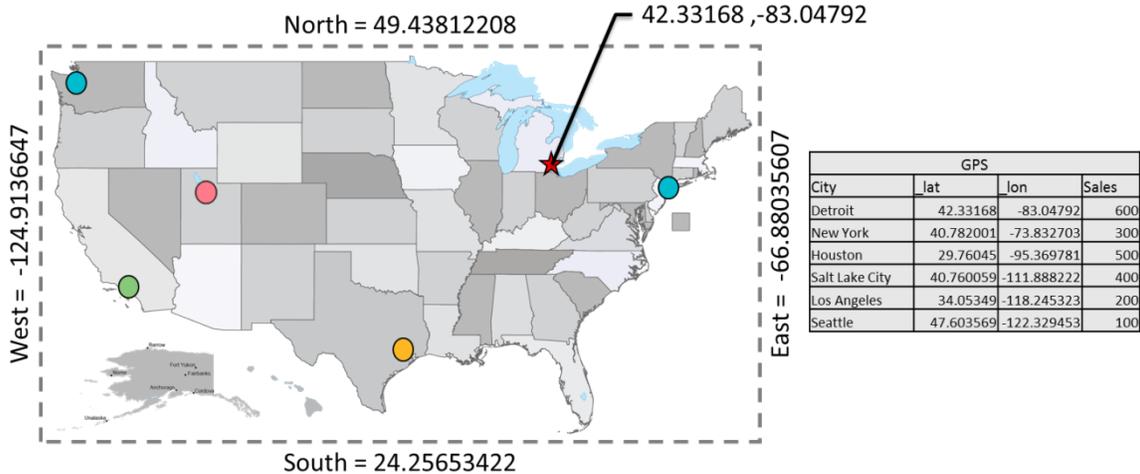
Below shows an example dataset, and how a point, representing Detroit, is positioned using percentage values. The values for locating Detroit in this example are specific to the image and image resolution.





9.14.2.3.4 Detroit by GPS example

Below shows an example dataset, and how a point, representing Detroit, is positioned using latitude and longitude values. The values for locating Detroit in this example are specific to the image and the North, South, East, West (N, S, E, W) boundaries of the image. Scaling the image would not require recalculating the data point location values, but it would require redefining the N, S, E, W boundaries of the image in the associated XML file.



9.14.2.4 Y-Axis Label Names: Point Positioning

Using the Image Plot within iDashboards requires specific axis label names to position the points on the chart. When labels are created using the options listed below, the chart will expect data in that specific format.

Method for locating each point

- From data set
 - Require specific Y-Axis label names
- From XML configuration file



- Does not require specific Y-Axis label names
- If using XML as a data source is desired, then users need a basic text/XML editor capable of reading and saving Text or XML Documents that are UTF-8 encoded
- Configuring an XML file is required if GPS coordinates are used

Note: The data-column name does not need to follow this naming convention

9.14.2.4.1 Pixels

Position points using Pixel distances.

<code>_x</code>	A positive, numeric value to position the point in the X direction
<code>_y</code>	A positive, numeric value to position the point in the Y direction

Notes:

- (0,0) is the upper-left corner of the image.
- Scaling the image would require recalculating the data point location values.
- A pixel axis can be paired with a percent axis.

9.14.2.4.2 Percent

Position points using Percent values where the image dimensions are 100%.

<code>_xpct</code>	A positive, numeric value to position the point based on the total width of the image
<code>_ypct</code>	A positive, numeric value to position the point based on the total height of the image

Notes:

- (0,0) is the upper-left corner of the image and (100%,100%) is the lower-right corner.
- Scaling the image would not require recalculating the data point location values.
- A percent axis can be paired with a pixel axis.

9.14.2.4.3 GPS

Position points using GPS coordinates.

<code>_lat</code>	A decimal, numeric value to position the point along a latitude
<code>_lon</code>	A decimal, numeric value to position the point along a longitude

Notes:

- This technique also requires the use of an XML Configuration file.
- This is the only type of positioning that allows negative values.



9.14.2.5 Y-Axis Label Names: Point Properties

Using the Image Plot within iDashboards there are a variety of options to customize the points that overlay the image. When labels are created using the options listed below, the chart will expect data in that specific format.

Method for defining the properties of each point

- From Chart Properties (globally assigns properties to all points)
 - Does not require specific Y-Axis label names
- From data set (allows unique point-property definition)
 - Requires specific Y-Axis label names
- From XML configuration file (allows unique point-property definition)
 - Does not require specific Y-Axis label names

Note: The data-column name does not need to follow this naming convention.

Note: There are two methods for ignoring Point Properties: Remove the underscore or delete the axis.



Point Properties

Color	Visual properties to alter the appearance of individual points
<code>_color</code>	<p>Defines the color of a point. HEX (as string) or Decimal value (as number).</p> <p>Example Values:</p> <ul style="list-style-type: none"> • #FF0000 (String equals Red) • 0xFF0000 (String equals Red) • 16711680 (Number equals Red) <p>Notes: A chart property exists to apply a specific color to ALL undefined points.</p>
Shape	Defines the shape of a point.
<code>_shape</code>	<p>Values include:</p> <ul style="list-style-type: none"> • circle • square • star • poly_#_# <p>Poly must include elements to define the number of sides and the rotation angle. Sides can range from 3-8 and angle can range from 0-359, as in poly_[3-8]_[0-359] (ex. poly_3_45)</p> <p>Notes: A chart property exists to apply a specific shape to ALL undefined points, but lacks the full variety of polygon shapes and angle rotation.</p>
Size	Defines the size, in Pixels.
<code>_size</code>	<p>Values include:</p> <ul style="list-style-type: none"> • 5-100 (as numeric) <p>Notes: A positive, numeric value to resize a point. A chart property exists to apply a specific size to ALL undefined points.</p>
Blink	Defines the blink capabilities, in milliseconds
<code>_blink</code>	<p>Determines the blinking state of a point. <code>_blink</code> is the duration (in milliseconds) that a point is in the visible state. If values are NULL, the point will not blink.</p>



`_blinkfactor` `_blinkfactor` is the factor of the `_blink` value in defining how much time to spend in the invisible state. For example, a blink factor of 1.0 will make the point invisible for the same duration as defined in the `_blink` axis. Using a blink factor of 0.5 will make the point invisible for a duration half of the amount defined in the `_blink` axis. If left blank, the blink factor will use "1".

Notes: Any positive, numeric values are allowed, but the blinking will never be less than 100ms, regardless of the provided value.

9.14.2.6 Consider the Data Source

First, emphasis must be placed on how the point(s) will be positioned. Then, consider how the points should appear using visual properties. Because of the unconventional data structure necessary for an Image Plot chart, the best way to apply plot locations to the Image Plot chart is to add them to the associated chart data source. In a simple scenario, Excel would be used as the type of data source because of the easy data structuring options. To achieve the minimum requirements for the chart the data source should have three columns; one X-axis and two Y-axes. Later, the document discusses additional chart properties which can be defined using additional Y-axes.

The X-axis would be any pertinent data for defining the point on the image. This could be data for display on the chart such as City, Location or Name in a geographical map. The Y-axes for the point on the image must be X and Y relational coordinates. Below, are three tables which represent a dataset for locating specific cities in the United States. Each dataset will position the points in the exact same location on the image, but using each of the positioning options of pixel, percent, and GPS.

Pixel			
Axis Position	X axis	Y1 axis	Y2 axis
Axis Label	City	_x	_y
	Detroit	1599	338
	New York	1947	417
	Houston	1129	940
	Salt Lake City	496	414
	Los Angeles	253	736
	Seattle	97	87

Percent			
Axis Position	X axis	Y1 axis	Y2 axis
Axis Label	City	_xpct	_ypct
	Detroit	72.12449	28.35570
	New York	87.82138	34.98322
	Houston	50.92467	78.85906



Salt Lake City	22.37258	34.73154
Los Angeles	11.41182	61.74497
Seattle	4.37528	7.29866

GPS			
Axis Position	X axis	Y1 axis	Y2 axis
Axis Label	City	_lat	_lon
	Detroit	42.33168	-83.04792
	New York	40.782001	-73.832703
	Houston	29.76045	-95.369781
	Salt Lake City	40.760059	-111.888222
	Los Angeles	34.05349	-118.245323
	Seattle	47.603569	-122.329453

The table below illustrates a dataset, using pixels to locate the points and specific visual properties for each point.

Pixel						
Axis Position	X axis	Y1 axis	Y2 axis	Y3 axis	Y4 axis	Y5 axis
Axis Label	City	_x	_y	_color	_shape	_size
	Detroit	1599	338	#FF0000	star	14
	New York	1947	417	#00BBCE	circle	10
	Houston	1129	940	#FDB724	circle	10
	Salt Lake City	496	414	#F27180	circle	10
	Los Angeles	253	736	#87C979	circle	10
	Seattle	97	87	#00BBCE	circle	10

9.14.2.7 Overview of Creating an Image Plot Chart using Pixels

Before creating an Image Plot chart, have an image and dataset prepared within iDashboards. Then, using the data from the table above, create a chart using axis names shown below. Keep in mind, an Image Plot only requires one X-Axis (defining the point(s) and two Y-Axes (for defining each point position). Therefore, this example will also demonstrate Point Properties defined in the Data Set.



Data Set » **Axis List** » Chart Type » Chart Properties » Preview

Specify Chart Axes:

Axis	Name	Data Column
X	<input type="text" value="City"/>	City ▼
Y1	<input type="text" value="_x"/>	x_position ▼
Y2	<input type="text" value="_y"/>	y_position ▼
Y3	<input type="text" value="_color"/>	the_color ▼
Y4	<input type="text" value="_shape"/>	the_shape ▼
Y5	<input type="text" value="_size"/>	the_size ▼

When selecting the chart type, click “Maps” on the left and “Image Plot” on the right. Lastly, the Image Plot requires referencing an image. This takes place in the chart specific settings.

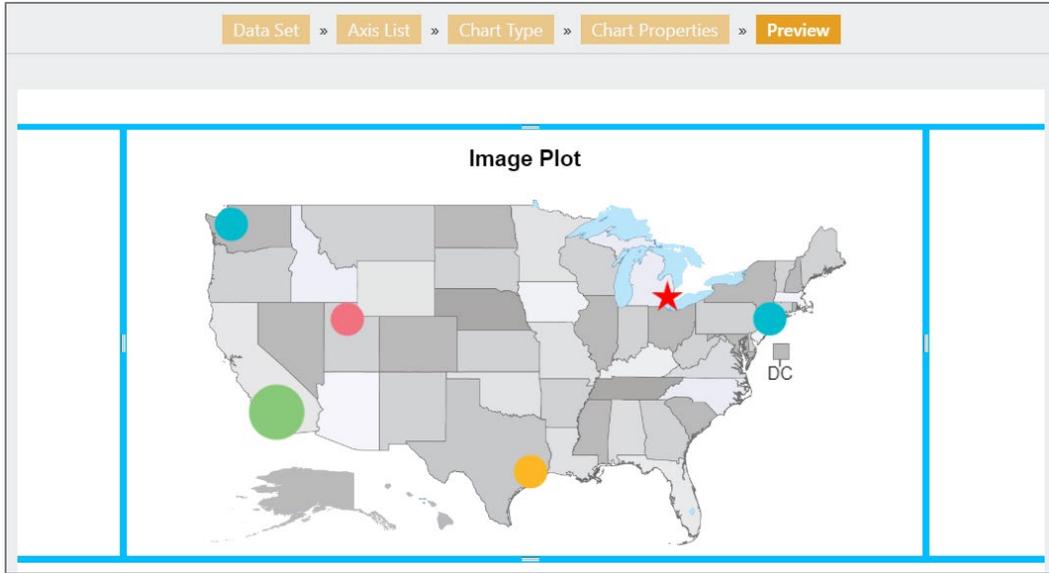
Image Plot Settings Reset

Built-in Map

Image URL x {}

Config XML URL x {}

The chart Preview should now demonstrate the image shown below. Hovering the cursor over the Image Plot points or the bar chart on the right will immediately show Interactive Intelligence.



9.14.2.8 Content Folder

The image you wish to use within an Image Plot chart will first need to be uploaded to iDashboards. Images may be stored within sub-folders on the iDashboards server to help with file organization.

iDashboards provides a content macro which allows you to replace the base URL for your iDashboards installation. The content macro makes referencing images easier. See below for examples on the standard format for using the content macro.

Example Introduction	Using the macro 'content:' will replace the iDashboards installation URL
This:	
content:	
Represents this:	
<u>http://www.yourwebserver.com/idashboards/content/</u>	
 <p>Imagine 'your' URL for iDashboards</p>	

Example 1	Start with the content macro and then suffix with an image name
Use this:	
content:USA_map.png	
To represent this:	
<u>http://www.yourwebserver.com/idashboards/content/USA_map.png</u>	



Example 2	Same as Example 1, but the image exists in a subfolder called 'gray_maps'
Use this:	
content:gray_maps/ <u>USA_map.png</u>	
To represent this:	
<u>http://www.yourwebserver.com/idashboards/content/gray_maps/USA_map.png</u>	

Note: Unlike Presentation charts, the Image Plot cannot use a value macro for the image filename.

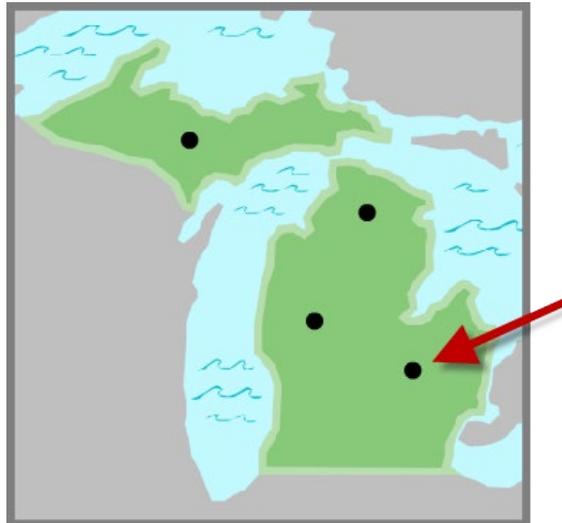
9.14.2.9 Image Plot Settings: Chart Designer

Within this section are Image Plot Settings that are unique and not common across other charts in the iDashboards library.

9.14.2.9.1 Example

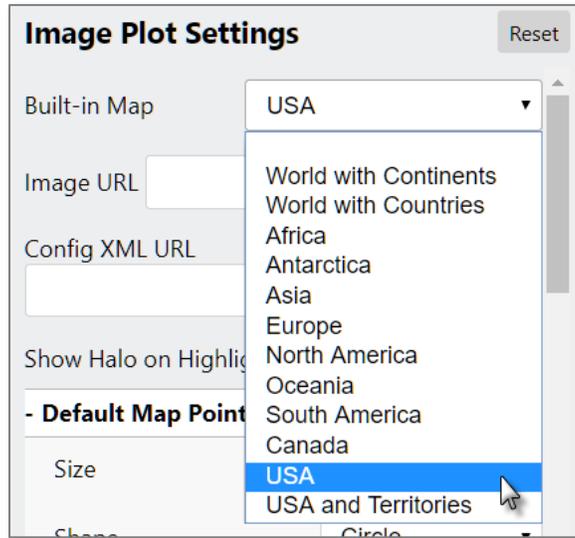
Some properties will reference the following example. Consider the Dataset shown below, and the resulting chart shown below (which has an 800 x 800 dimension).

Name	Tower	_x	_y	City	Day Op Cost	Strength
Axis	X	Y1	Y2	Y3	Y4	Y5
	tower	x_pos	y_pos	city	day_op_cost	strength
	Tower 100	620	560	Flint	875	10000
	Tower 200	470	485	Big Rapids	643	8000
	Tower 300	550	320	Gaylord	225	6000
	Tower 400	280	210	Gwinn	180	5000



9.14.2.9.2 Built-in Map

Selecting a built-in map will disable the 'Image URL' and 'Config XML URL' settings. The dropdown list will display the iDashboards 12 pre-installed maps.



9.14.2.10 Image URL

The required image URL path and image name. For this example, the image file "Michigan.png" was placed at the root level, within the Administrator Application.

content:Michigan.png

9.14.2.11 Config XML URL

The optional XML URL path and file name. An XML file is required for use with GPS data values.

9.14.2.11.1 Point Label Template

Any axis value from the dataset or static value can be used as the visual label associated with each point. When left blank, the point will display the X-Axis value. The syntax defined here will be applicable for all points.

Tip: Enable "Point Labels Always Visible" to quickly preview the Point Label Template settings while developing the label template.

- **Macro** – Label values can be static values, or can come from a variety of dynamic macros. Clicking on the macro button "{\$}" will list the following options for the current example:

Macro	Description
`\${selected:axis}`	Label display would be 'Day Op Cost' or 'Strength' Requires multiple numeric Y-axes and is used in conjunction with enabling "Use Range Colors".
`\${selected:value}`	Label display would be '875' or '10,000' Requires multiple numeric Y-axes and is used in conjunction with enabling "Use Range Colors".

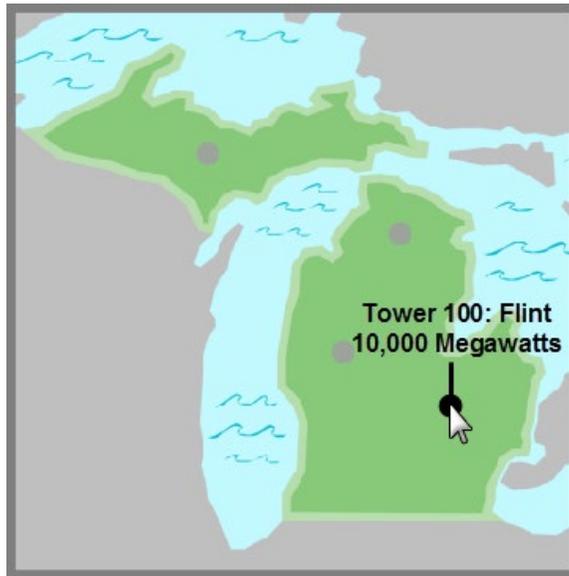


Macro	Description
<code>#{value: Tower}</code>	Label display would be 'Tower 100'
<code>#{value: _x}</code>	Label display would be '620'
<code>#{value: _y}</code>	Label display would be '560'
<code>#{value: City}</code>	Label display would be 'Flint'
<code>#{value: Day Op Cost}</code>	Label display would be '875'
<code>#{value: Strength}</code>	Label display would be '10,000'
<code>#{user}</code>	Label display would be '<current logged in user>'
<code>#{charttitle}</code>	Label display would be '<current chart title>'
<code>#{chartid}</code>	Label display would be '<current chart ID>'
<code>#{categoryid}</code>	Label display would be '<current category ID where chart is saved>'
<code>#{br}</code>	Use this 'Break Line' macro to add multi-line labeling
<code>#{blank}</code>	Use this to omit the label entirely (must be the only value entered)

Point Label Template, with Macros, Example:

`#{value: Tower}: #{value: City}#{br}#{value: Strength} Megawatts`

Results:



9.14.2.11.2 Range Colors (Single Numeric Y-Axis)

If 'Range Colors' is enabled, the point colors will be identified with the colors defined within the chart Range Colors.

Show Y-axis Name – Will display the numeric Y-axis label name beneath the graphic.

Legend Priority – When Range Colors and an XML configuration file are used, see the property "Legend Priority" to determine which color option should have priority.

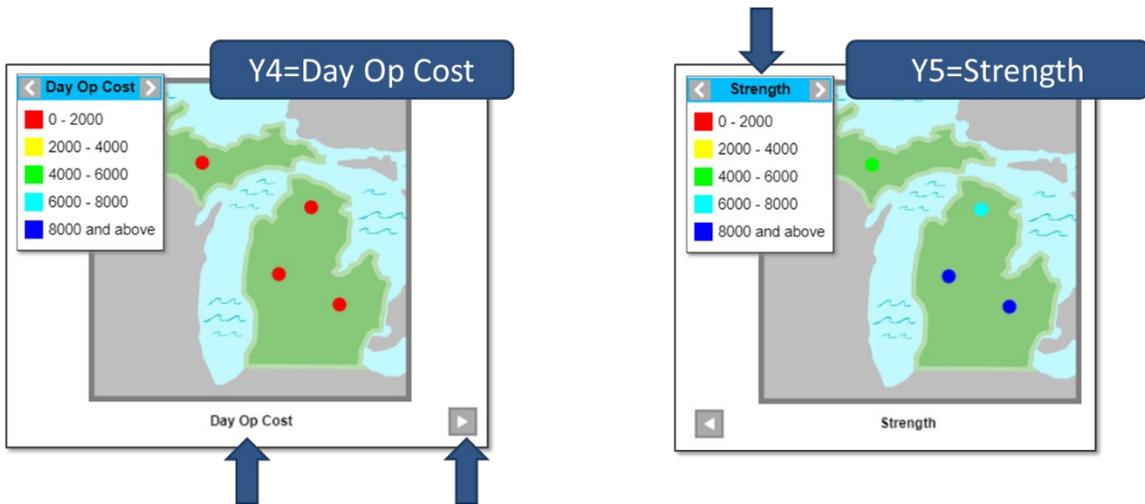


9.14.2.11.3 Range Colors (Multiple Numeric Y-Axis)

If 'Range Colors' is enabled, the point colors will be identified with the colors defined within the chart Range Colors. Because there are multiple numeric Y-Axis values, the chart will have scroll left-right options to see each additional Y-axis, and each Y-axis can have its own range color definition. The image below shows the cursor location where clicking on the left, or right, arrow will scroll through the multiple numeric Y-axes. While clicking on the arrows, the map data, map value (beneath the map) and the map legend values will also change.

Show Y-axis Name – Will display the numeric Y-axis label name beneath the graphic.

Legend Priority – When Range Colors and an XML configuration file are used, see the property "Legend Priority" to determine which color option should have priority.



9.14.2.11.4 Show Mouse Location

While there may be alternative methods for identifying the coordinates of a point, this iDashboards feature was built into the Image Plot chart to assist with identifying the location of points if using pixel or percent values. It works in conjunction with the property 'Copy Click Location For'.

Show Mouse Location – Has three options: None, Pixels, or Percent.

- **None** – Disables the feature
- **Pixels** – Enables the on-screen ability to display and capture the pixel coordinates of the mouse pointer. Values range from 0,0 to <max width>,<max height>
- **Percent** – Enables the on-screen ability to display and capture the percent coordinates of the mouse pointer. Values range from 0,0 to 100,100.

Copy Click Location – Has three options: SQL, XML, or Excel.

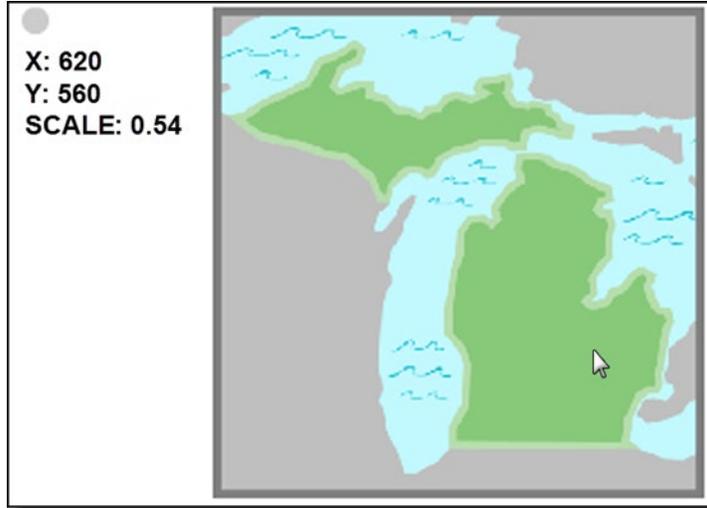
- **SQL** – Captures the coordinates of the mouse pointer using an SQL syntax. Where the x/y coordinates are separated by a comma.
 - 537, 642
- **XML** – Captures the coordinates of the mouse pointer using an XML syntax. Where the x/y coordinates are identified within x and y tags.
 - <x>537</x>
 - <y>642</y>
- **Excel** - Captures the coordinates of the mouse pointer using an Excel syntax. Where the x/y coordinates are separated by a tab-character.
 - 537 642

9.14.2.11.5 Show Mouse Location Example

1. First, begin by building an Image Plot chart containing at least one X-axis and one Y-axis. Creating the chart using the required Point Position axes is also an option.
 - a. Within the Chart Properties, make sure to define the 'Image URL'.
 - b. Within the Chart Properties, enable 'Show Mouse Location' using either Pixel or Percent.
 - c. Within the Chart Properties, set 'Copy Click Location For' to any desired value.
2. Click 'Finish' on the Chart Designer. The Image Plot chart should appear with the image, but no points.
3. Enlarge the chart frame to a large size (for added accuracy)
4. Now, position the cursor over a location where a data point needs to appear. Notice the upper-left portion of the frame shows the coordinates and image scaling. Left-click the mouse.
 - a. This process has captured the coordinates into the computer clipboard, allowing the values to be pasted into the desired data source location.
5. Repeat this process until all points have been registered.

The image below illustrates how the point location for 'Tower 100', in Flint, Michigan was obtained. In the upper-left corner of the chart frame, the pixel dimensions are shown. When a click is performed, those dimensions are saved into the clipboard (using the pre-determined format), so the user can ultimately paste the values into the data set.





9.14.2.12 XML Structure

An XML file is required for use with GPS data values. Otherwise, an XML file is optional and provides an extended set of customizations with the Image Plot chart.

Note: When developing the XML file, it may be necessary to make changes to the file and re-upload. Due to some browser cache settings, it may be necessary to log out of iDashboards and then log back in to see the changes.

The XML file requires a specific syntax for use within iDashboards. The body of the xml file includes many tags. Some tag values are described in this section and other tag values can be found in the Point Property Values from Section 12.16.5, “Y-Axis Label Names: Point Properties”. Below are the various body elements and tags.

9.14.2.12.1 XML Tree structure

On the XML file, the first line should include a standard XML declaration, defining the XML version (1.0). The next line is the required root element `<imageplot version="1.0">` and the last line must be the closing root element `</imageplot>` (as shown below).

Line 1	<code><?xml version="1.0" encoding="UTF-8"?></code>
Line 2	<code><imageplot version="1.0"></code>
Line 3-#	<code><!--XML content goes here--></code>
Last Line	<code></imageplot></code>

9.14.2.12.2 Labels (and Lines)

Use this section to place text on the Image Plot chart. Labels can be placed next to points or on the Image Plot as a chart note.

- **Lines** – Adding a line to a label is optional. Use this section to construct lines connected to label values. The line tag needs to be nested within the Label element. Lines are drawn from the x/y endpoint toward the center of its parent label. The line will not appear in the space of the label value, and will automatically disappear once text begins to interfere with the line.

```

<labels> <!--do not repeat this tag-->
  <label>
    <value/>
    <color/> <!--also applied to line-->
    <size/>
    <x/> <!--or <xpct></xpct>-->
    <y/> <!--or <ypct></ypct>-->
    <line>
      <x/> <!--or <xpct></xpct>-->
      <y/> <!--or <ypct></ypct>-->
    </line>
  </label>
  <label/> <!--repeat for additional labels-->
</labels>

```

9.14.2.12.3 Points

Use this section to place points/shapes on the Image Plot chart

```

<points> <!--do not repeat this tag-->
  <point>
    <value/> <!--X-axis match value-->
    <color/> <!--hex or decimal-->
    <size/>
    <x/> <!--or <xpct></xpct>-->
    <y/> <!--or <ypct></ypct>-->
    <shape/> <!--any of the shape values-->
    <blink/> <!--number-->
    <blinkfactor/> <!--number-->
  </point>
  <point/> <!--repeat for additional points-->
</points>

```

9.14.2.12.4 Legend

Use this section to create a custom legend for an Image Plot chart. Do not repeat the 'legend' or 'items' tag, each XML file can only contain one legend.

```

<legend> <!--do not repeat this tag-->
  <items> <!--do not repeat this tag-->
    <item>
      <text/>
      <icon>

```



```

        <shape/> <!--If a polygon shape is
desired, the value must appear in the form poly_<sides> or
poly_<sides>_<angle>-->
        <color/>
    </icon>
</item>
<item/> <!--repeat for additional legend items-->
</items>
</legend>

```

9.14.2.12.5 GPS

Use this section to define the latitude and longitude values for the North, South, East, and West boundaries of an image file. This is a required tag if data contains latitude and longitude values. Do not repeat the 'gps' or 'mapboundaries' tag, each XML file can only contain one set of boundary references.

```

<gps> <!--do not repeat this tag-->
  <mapboundaries> <!--do not repeat this tag-->
    <north/> <!--decimal top latitude value-->
    <south/> <!--decimal bottom latitude value-->
    <east/> <!--decimal right longitude value-->
    <west/> <!--decimal left longitude value-->
  </mapboundaries>
</gps>

```

9.14.2.12.6 GPS Map Inset

In addition to the GPS element, it is possible to implement 'inset' values into an Image Plot chart. The common use for an inset can be seen below. Alaska and Hawaii don't easily appear on maps of the United States because of their obscure distance from the 48 contiguous states. Therefore, maps of the United States often place Alaska and Hawaii into a visually appealing location, without concern for accurate GPS positioning. Within iDashboards, this is known as a map inset. The Image Plot chart supports one or more insets (insets cannot overlay) if the XML Configuration file is developed correctly.

```

<gps>
  <mapboundaries> <!--do not repeat this tag-->
    <north/> <!--decimal top latitude value-->
    <south/> <!--decimal bottom latitude value-->
    <east/> <!--decimal right longitude value-->
    <west/> <!--decimal left longitude value-->
  </mapboundaries>
  <inset frame="false"> <!--Alaska: To enable, set value
to 'true'. Inset values use pixel coordinates-->
    <topleft>
      <x/> <!--or <xpct></xpct>-->
      <y/> <!--or <ypct></ypct>-->
    </topleft>
    <bottomright>
      <x/> <!--or <xpct></xpct>-->

```

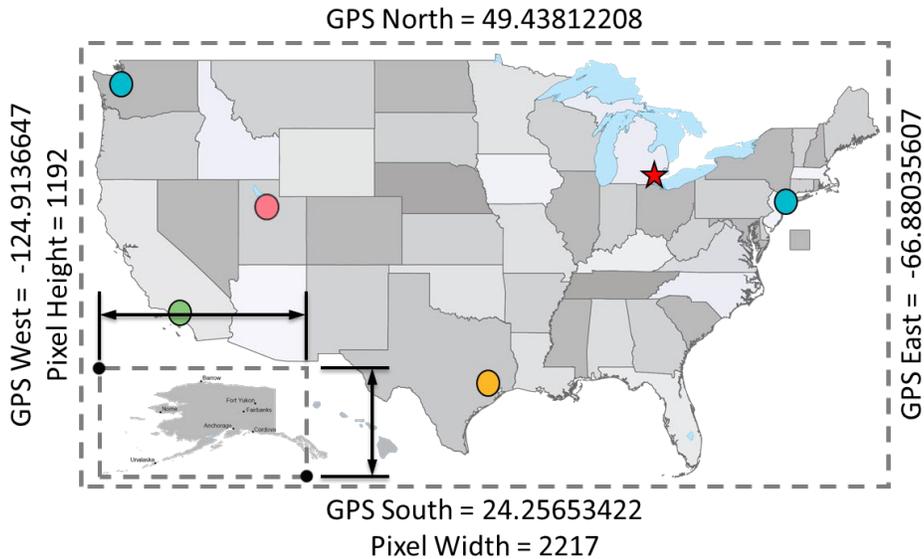


```

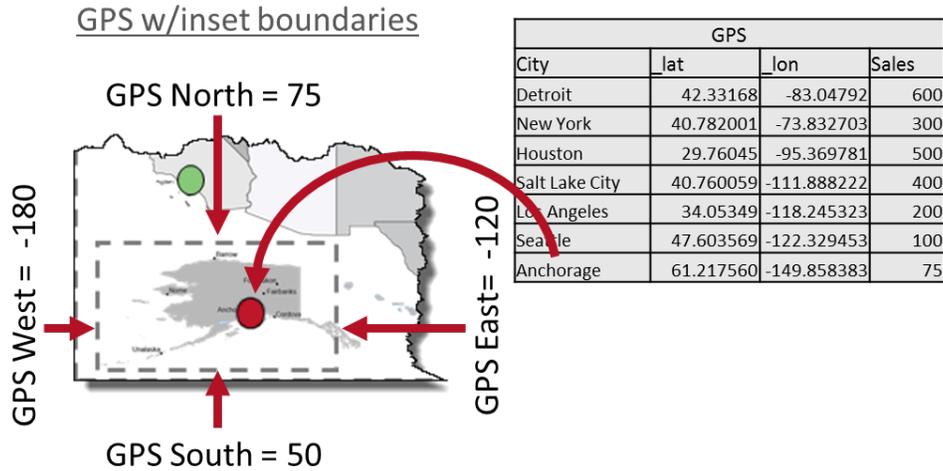
        <y/> <!--or <ypct></ypct>-->
    </bottomright>
    <mapboundaries> <!--do not repeat this tag-->
        <north/> <!--decimal top latitude value-->
        <south/> <!--decimal bottom latitude value-->
        <east/> <!--decimal right longitude value-->
        <west/> <!--decimal left longitude value-->
    </mapboundaries>
</inset>
</gps>

```

The Image Plot chart can position points based on GPS location if the North, South, East and West boundaries are defined. If the geography exists within one of the rectangular insets, then additional points can be located via pixel location. When the data set can include GPS values for any location (using `_lat` and `_lon` axes); if the location is included by the major map boundary, the point will be plotted there, but if the location is only included within the inset boundaries then the point will be plotted in the inset portion of the map.



Refer to the image below for an example how Anchorage, Alaska can be accurately plotted within the inset portion of the map.



9.14.2.13 Priority Notes

The Image Plot chart has multiple areas in which to define multiple items. For example, point positions can be defined by the data or through an XML Configuration file. In circumstances where values are defined by two or more techniques, the following rules apply:

- If Range Colors, and an XML configuration file are used, see the property “Legend Priority” to determine which color option should have priority.
- Point Properties defined by data will override properties available within the Chart Designer.
 - **Priority 1:** (only for Color properties): Range Color, if Range Colors are configured to be used
 - **Priority 2:** Data Column
 - **Priority 3:** XML Configuration File
 - **Priority 4:** UI Properties within Chart Designer

9.14.2.14 Image Plot Tips

Using the Image Plot will likely require data structuring, then even the most advanced data sets will be able to fully utilize the potential of the Image Plot. Therefore, with creative thinking, and combining multiple features within iDashboards, it makes sense to explore additional options.

Consider using derived columns for point positioning or point properties. Consider the data set shown below. There are only 2 traditional looking data columns, ‘City’ and ‘Sales’. The following example will construct various elements necessary for an Image Plot using derived columns.

Axis Position	X axis	Y1 axis	Y2 axis	Y3 axis	...
Axis Label	City	Sales	<derived column>	<derived column>	...
	Detroit	1599			
	New York	1947			
	Houston	1129			



Salt Lake City	496			
Los Angeles	253			
Seattle	97			

9.14.2.14.1 Define the X-Pixel value for Detroit and New York

```
if (x=='Detroit') {1599}
else if (x=='New York') {1947}
else {null}
```

9.14.2.14.2 Define shape of Detroit as a Star and all other cities as a Circle

```
if (x=='Detroit') {'star'}
else {'circle'}
```

9.14.2.14.3 Make point blink if numeric value is less than 1000

```
if (y1<1000) {1}
else {null}
```

9.14.2.14.4 Make point blink if numeric value is less than <Input Parameter>

This example uses an input parameter in the expression. As a user changes the input parameter value on the chart, different cities will begin to blink if their sales value is less.

```
if (y1<param_value) {1}
else {null}
```

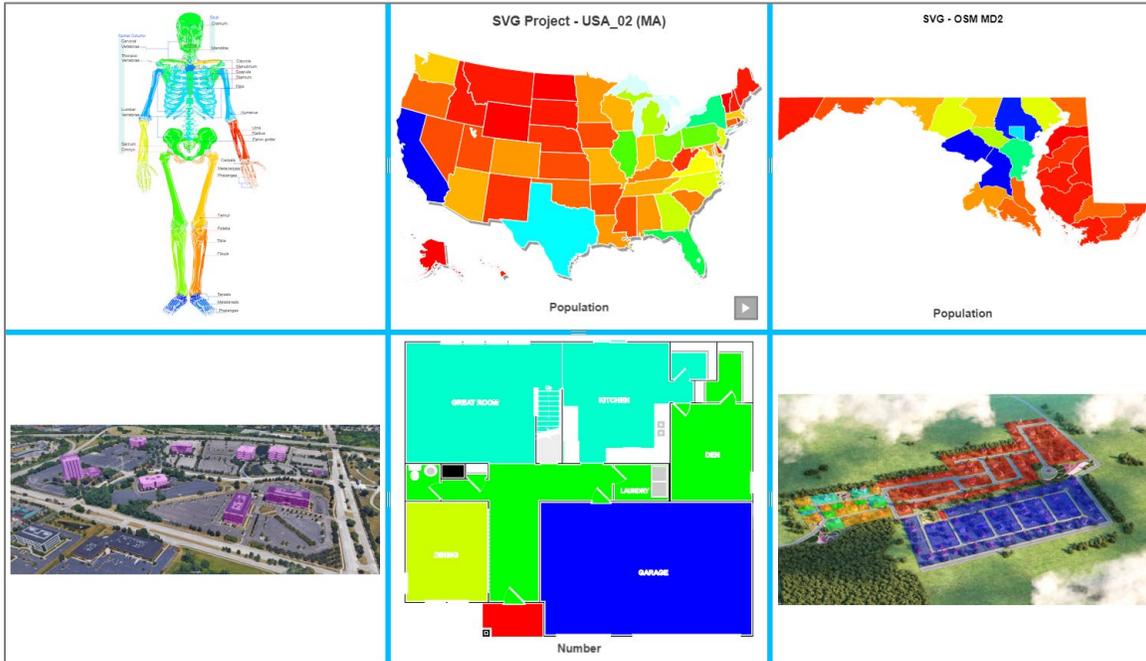
9.14.2.14.5 Define each city with a specific HEX color

```
if (x=='Detroit') {'#FF0000'}
else if (x=='New York') {'#00BBCE'}
else if (x=='Houston') {'#FDB724'}
else if (x=='Salt Lake City') {'#F27180'}
else if (x=='Los Angeles') {'#87C979'}
else {'#00BBCE'}
```



9.14.3 SVG Drawing

Use an SVG file to display interactive and high-quality regions of any drawing. Each region shape can use range colors, a global color or no color at all. This chart is used for displaying custom charts using a dynamic “illustration” experience. The creative uses are endless and some examples include geographic maps, floorplans, property management, scientific illustration, and much more!



9.14.3.1 SVG File

The core feature of the SVG Drawing chart is the Scalable Vector Graphic (SVG) file. Similar to the Image Plot chart, iDashboards doesn't provide the file needed to use this chart because it is a 100% custom chart.

The SVG element, which is the root of the SVG document, must, at minimum, have a viewBox attribute. as shown in the example above. The first two numbers should be 0 and the third (width) and fourth (height) should be positive.

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 600 400">
```

If the SVG element has width and height attributes, they should adhere to the following requirements:

- Either both or neither should be present.
- The only unit suffix they may have is "px", for example "600px", or they should have no unit suffix at all, for example, "600".
- In most cases the width and height attribute should match the third and fourth numbers of the viewBox attribute, respectively. Otherwise, their ratio to each other (the aspect ratio) should match that of the third and fourth viewBox numbers.

An example SVG element with conforming width and height attributes would look like:

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 600 400"
width="600" height="400">
```

Any other attributes or inline CSS styling on the SVG element may potentially cause unwanted behavior when the SVG image is sized and positioned, and they should be examined when troubleshooting any such unwanted behavior.

9.14.3.2 ID Attributes and shape-to-data mapping

For the iDashboards SVG Drawing chart to interact with a polygon shape ('object') within the SVG file, a mapping process must be used. This process utilizes an 'id' value within the SVG file and a matching data value within the X-Axis or Y-Axis of the chart.

The objects in the map that are filled in with range colors, and which the labels are positioned relative to, must have id attributes. These objects will be referred to herein as "datapoint objects." The values assigned to the id attributes must adhere to the following requirements:

- Each value may appear only once, as an id attribute, in the entire document.
- The value may contain only letters, digits, underscores and hyphens.

It is not a requirement, but the id attribute values, ideally, should contain either lowercase or uppercase letters, but not both.

In cases where multiple objects form a single map object (For example, the Hawaiian Islands form the state of Hawaii), these objects should be grouped together using a <g> element, and the id attribute should be placed on the <g> element.

9.14.3.3 Label Positioning

Each datapoint object in the map has an imaginary bounding rectangle, which is the smallest possible rectangle that contains all of the datapoint object. A datapoint label, which consists of one or more lines of text of a particular font and size, also has an imaginary bounding rectangle. By default, a label is positioned so the center of its bounding rectangle is at the same coordinate as the center of its datapoint object's bounding rectangle.

Advanced

In most cases, the default label positioning is satisfactory, but if not, the label's position can be offset, up or down or left or right, by a specific number of pixels. This is accomplished by adding one or both of the following attributes to the datapoint object's element:

- data-label-offset-x - negative values move the label to the left, and positive values move it to the right.
- data-label-offset-y - negative values move the label up, and positive values move it down.



9.14.3.4 Label Lines

Advanced

Sometimes, an object's label will be positioned far enough away that a line, drawn from the label to the center point of the object, is desirable. To make the line appear, add this attribute with a value of "true":

- data-label-line

The line's endpoint can be offset from the center point of the object with one or both of these attributes:

- data-label-line-offset-x - negative values move the line's endpoint to the left, and positive values move it to the right.
- data-label-line-offset-y - negative values move the line's endpoint up, and positive values move it down.

If the label's bounding rectangle contains the line's endpoint, the line will not appear.

9.14.3.5 Label CSS styling

A label's font size and color are configurable through the SVG chart's configuration panel. Also, the configuration panel provides a setting for a "shadow" which gives the label text an outline effect.

Advanced

The labels for individual datapoint objects can be styled, however, by applying data-label-css-*<stylename>* attributes to the datapoint object's element. Here's an example where the label is given a border and a partially transparent background:

```
<g id="BALTIMORE_CITY" data-label-offset-x="-195"
  data-label-offset-y="195"
  data-label-line="true"
  data-label-line-offset-y="-5"
  data-label-css-padding="10px"
  data-label-css-border="1px solid blue"
  data-label-css-background-color="rgba(0,0,255,0.3)"
  data-label-css-border-radius="10px">
```

For each one of an object's attributes whose names begin with "data-label-css-", the part of the name that follows "data-label-css-" is used as a CSS style name, and it's value is used as a value, and these styles are applied to the <div> element used as the label. They are applied after the default styles configured in the chart designer are applied, so they can be used to override those defaults.



9.14.3.6 Hideable Map Objects

Advanced

A map can contain up to 10 groups of one or more "hideable map objects." These are objects that a user may want removed from the map (visually speaking) for a less cluttered appearance. Examples might be text labels or extraneous graphics such as legends or map scales.

The groups are numbered 1 to 10. A map object is added to a group by assigning a CSS class to its root element. The class name consists of "idb-hide-" followed by the group number. For example:

```
<text class="idb-hide-1" transform="translate(597.13
270.15)">Worcester</text>
```

Or

```
<text class="cls-4 idb-hide-2" transform="translate(597.13
270.15)">Worcester</text>
```

Note: unlike an id attribute, the value of a class attribute does not have to be unique within the file. In fact, they most typically are not.

Also, the value of a class attribute can contain multiple class names, separated by spaces. In the example shown above, "cls-4" is also a class name, separate from, and unrelated to, the "idb-hide-2" class name.

When a map SVG is loaded in the chart designer (to display the live preview) the checkboxes for the "Hideable Map Objects" control are enabled/disabled based on whether or not objects with the corresponding class are found in the file.

9.14.3.7 Cell Highlighting

Advanced

When a map cell is highlighted, its root element has a CSS class named "idb-cell-highlight" applied to it, and it is removed when the cell is no longer highlighted.

A map developer can define, inside the SVG file, what this class does to a map cell. This is done by creating a CSS class named idb-cell highlight. For example:

```
<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 657.27 319.43">
  <defs>
    <filter id="idb-glow">
      <feColorMatrix type="matrix"
        values="
          0 0 0 0 0
          0 0 0 0.9 0
          0 0 0 0.9 0
          0 0 0 1 0"/>
      <feGaussianBlur stdDeviation="2.5"
        result="coloredBlur"/>
      <feMerge>
        <feMergeNode in="coloredBlur"/>
        <feMergeNode in="SourceGraphic"/>
      </feMerge>
    </filter>
  </style>
```



Advanced

```
.idb-cell-highlight {  
  filter:url(#idb-glow)  
}  
</style>  
</defs>
```

In the above example, a filter is defined that puts a glow around the edges of objects to which it is applied. Below that, in the `<style>` block, a rule set is created that says that any objects with the class `idb-cell-highlight` will have that filter applied to them.

9.14.3.8 Zoom and Pan

The SVG Drawing has multiple settings to control zooming in and out of the drawing, and for panning around once zoomed in. Zooming in and out, along with resetting to the initial zoom level can be accomplished using the image's **+** **-** and center icon buttons, in the lower left corner of the chart. For a desktop interface zooming is also accomplished using the mouse's scroll wheel, and the left click button for panning around. On a mobile device pinch-n-zoom and two-finger panning can also be used.

9.14.4 US County Map

9.14.4.1 Concept

There are 2-primary layers to the US County Map chart. The background layer consists of a dynamic map and the foreground layer provides an infrastructure to overlay county shapes.

Like most charts within iDashboards, the US County Map chart is a combination of data with an illustration, to provide a dynamic experience to relay information visually. The US County Map chart will overlay county shapes over an interactive map of the United States. iDashboards hosts a map tile server and the connection is pre-configured within the chart.

9.14.4.2 Requirements

Because this chart relies upon an additional network connection to a map tile server, some requirements are necessary for the chart to operate.

- Internet connection from the client to the iDashboards map tile server
- Maintenance and Support is required to use this chart
- Data structure requirements, as further defined in this section

9.14.4.2.1 License to enable this chart

Contact your iDashboards administrator for details on licensing.

- This chart requires iDashboards v9.1 or later
- A new license file specifically created for v9.1 or later

9.14.4.2.2 Licensed to use iDashboards Map Tile Server

While all licensed versions of iDashboards v9.1 will have this chart, there are two settings which control the ability to connect to the iDashboards map tile server:

1. The chart CAN use the iDashboards map tile server
2. The chart CANNOT use the iDashboards map tile server

Note: If the iDashboards map tile server is not an available option, please contact your iDashboards administrator or iDashboards account manager.

9.14.4.3 Data Format

In the United States, county names are unique within each state. However, county names are not unique between multiple states. Therefore, a mechanism needs to be used to identify which county to highlight when the data value is “Washington”. The solution for this depends on the data set or if the counties are going to be identified with single state.

Data must be provided in one of the following formats, as registered through the USDA Natural Resources Conservation Service website:

https://www.nrcs.usda.gov/wps/portal/nrcs/detail/national/home/?cid=nrcs143_013697

County Name	County Name with suffix
Oakland	Oakland_County
Jackson	Jackson_County



Switzerland	Switzerland_County
Ford	Ford_County
Stark	Stark_County
Grand Traverse	Grand Traverse_County

Note: The suffix can either be “_county” or “county”, and the word “county” is not case-sensitive.

9.14.4.4 Example 1

If the data uses the following structure and contains the county name and does not contain the state name, then it will be necessary to set the chart property to set a static/fixed state value. Not assigning a state value means that “Jackson” county will appear over twenty times because the United States has over twenty “Jackson” counties. However, because this data set requires setting a specific state, there will be counties that won’t render at all if they do not associate with the selected state.

County Name	Number
Oakland	100
Jackson	200
Switzerland	300
Ford	400
Stark	500
Grand Traverse	600

Note: The above data example will only provide accurate results when a single state value is selected.

9.14.4.5 Example 2

If the data uses the following structure and contains the county name and the state name, then it will be necessary to set the chart property to identify the column containing the state name. This data set will allow the counties to span across multiple states, with each county name specifically associated to the appropriate state

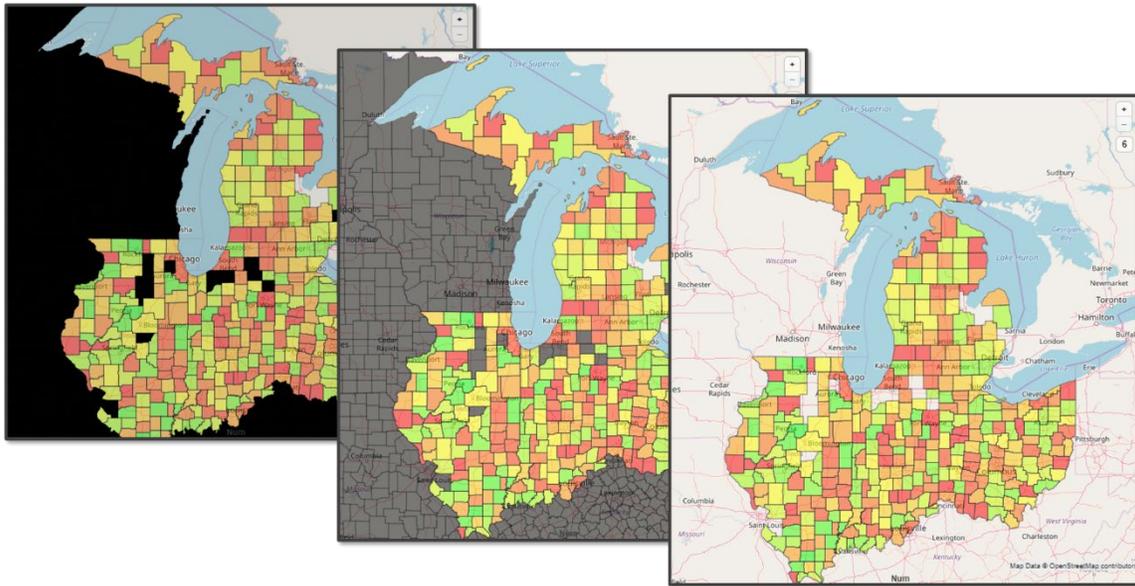
County Name	State Name	Number
Oakland	MI	100
Jackson	MI	200
Jackson	OH	300
Jackson	IL	400
Jackson	IN	500
Switzerland	IN	600
Ford	IL	700
Stark	IL	800
Stark	OH	900
Grand Traverse	MI	10000

Note: The above data example will provide accurate results even when the counties span multiple states.



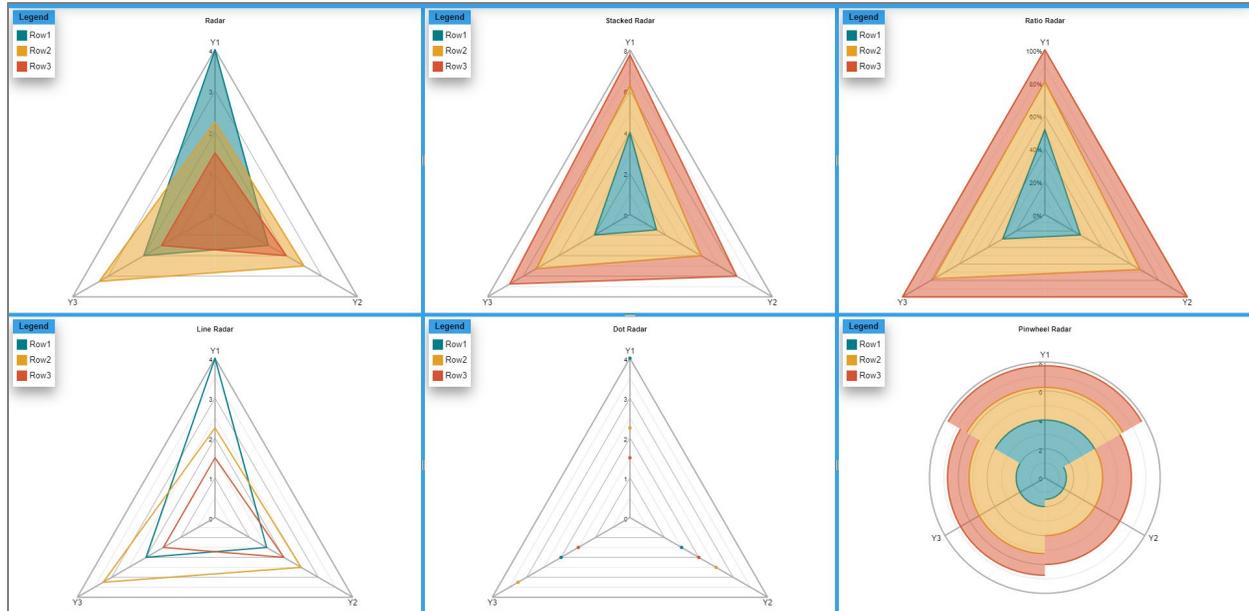
9.14.4.6 No Data Cells

The County chart has a unique property to drastically change the appearance and surprisingly, the property is associated to the county shapes where there is no data found. By default, the counties without data will appear black, using a 50% transparency. However, both the color and transparency can be changed when customizing the appearance.



9.15 Radar Charts

There are five radar charts, closely related in purpose and features. A radar chart is unique as each data row is displayed on a linear-radial or circular-radial grid using a certain shape type.



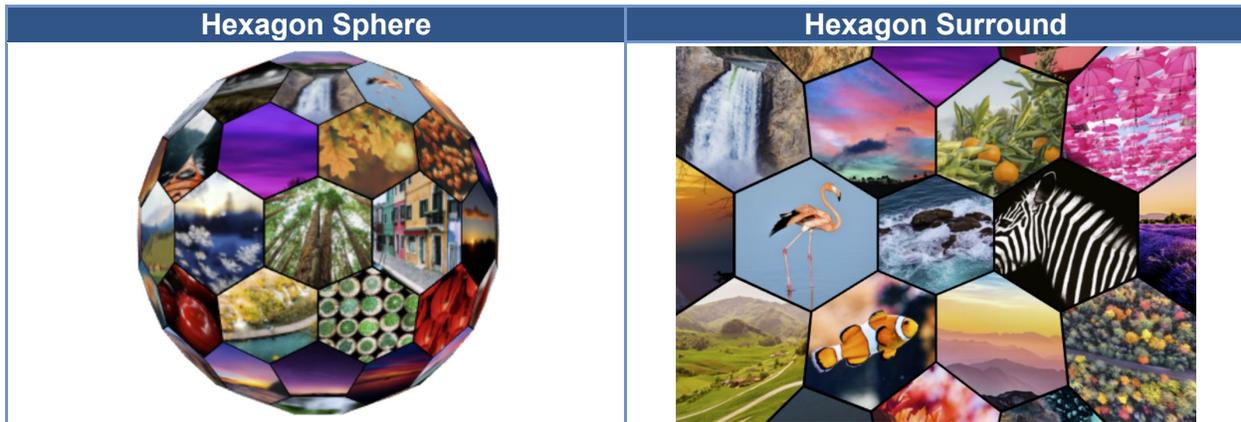
- **Radar** – Displays a data row using a filled polygonal shape.
- **Stacked** – Similar to a Radar chart, except that each shape builds off the previous shape by adding the current row's Y Axis values to the previously-plotted Y Axis values.
- **Ratio** – Similar to a Stacked Radar chart, except that the values along each Y Axis line are normalized to fit within 100%.
- **Line Radar** – Similar to a Radar chart, except that the polygonal shapes are not filled.
- **Dot Radar** – Similar to a Radar chart, except that the polygonal shapes do not have lines and fills; shapes display only their vertex dots.
- **Pinwheel Radar** – Similar to a Stacked Radar chart, except the shape is not always continuous between data points and the shape is circular.

9.16 3D Image Chart

This chart displays images onto the 3-dimensional faces of a particular shape. Some of the shapes are shown from the viewpoint of looking at a 3D object from the **outside** perspective, while the 'surround' shapes are shown from the viewpoint of looking at a 3D object from the **inside** perspective. Each shape (cube, sphere, etc.) has an adjustable number of faces (selectable from a list of values) which are used for displaying an image. When the complexity of the shape increases, the number of faces increase too. Currently, the number of faces range from 6 to 320, depending on the shape!

In addition to a configurable, automatic rotation, this chart mostly comes alive when grabbed with the cursor and spun around. Similarly, a touch device or mobile phone is especially exciting to use. However, the functional purpose of this chart is to provide visual navigation of multiple images. And due to the obscure angles and polygon renderings, all faces displaying an image can be selected – at which time the face will be centered to the screen and zoomed. There are a few ways to 'unzoom' and return to the original shape: click the "X" close button, select another face (in the background), or drag-n-drop the selected image.

Like all presentation charts, the dashboard builder is responsible for providing the images that will be displayed on the 3D shape. It is possible to use any variation of the chart if there are more or less images than the exact number of 3D faces; and using the configuration options, there are properties to control the outcome of such a discrepancy.



9.16.1 Technical workflow

Due to the massive bandwidth this chart is capable of consuming, it may be helpful to understand the working components. Imagine a Hexagon Sphere with 122 faces and 122 unique images. When opened, this chart will first retrieve the data set. When properly configured, the chart will contain enough data to reference one image per data row. As the chart begins to render on the screen, 122 requests for each image are made to the server where the images are stored. When the first image file is received, it is converted into a low-resolution image so that it can be rendered on the 3D shape without any performance issues. It's likely you will have to wait for each image to load, and there is a 'waiting' animation on each face to occupy that time and demonstrate the progress being made. When the images load, they will be placed randomly on a face (front, back, side, etc.) When all the images are loaded and processed, then the bandwidth is returned to normal but now the browser is using a chunk of

memory and your computer/device will need to continue processing the 3D animations. Lastly, when any face is selected, a full-resolution version of the image will be shown within the chart frame boundaries. This full-size image will immediately appear without any latency because the file was previously cached when the chart was opened.

Any image type listed below is supported. Additionally, the portrait or landscape projection of the image can remain as-is and will be programmatically cropped and centered while displayed on a 3D face – yet will render as-is when selected (and the image is centered and zoomed).

9.16.2 Supported Image Types

- PNG
- JPG (JPEG, PJPEG)
- GIF
- SVG (SVGZ)

9.16.3 Shapes

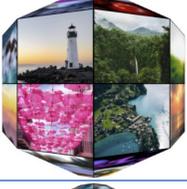
The following tables illustrate an actual screenshot of each shape and face-number combination for all chart options.

Cube	Faces
	6
	24
	96
-	-

Triangle	Faces
	20
	80
	180
	320

Hexagon	Faces
	32
	122
	272

Globe	Faces
	130
	202
	290

Jewel	Faces
	20
	26
	62
	74

Jewel	Faces
	128
	146
	164
-	-

Cylinder	Faces
	57
	100
	155
	164

Cube Surround	Faces
	6
	24
	96
	-

Triangle Surround	Faces
	20
	80
	180
	320

Hexagon Surround	Faces
	32
	122
	272

Globe Surround	Faces
	130
	202
	290

Jewel Surround	Faces
	20
	26
	62
	74

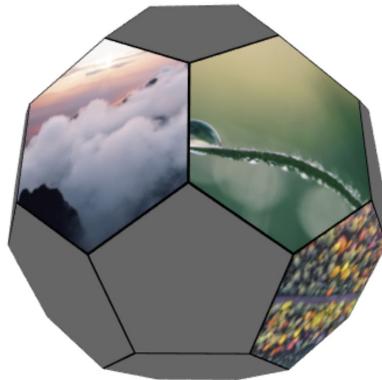
Jewel Surround	Faces
	128
	146
	164
-	-

Cylinder Surround	Faces	Thumbnail Carousel	Faces
	57		1 Row
	100		2 Rows
	155		3 Rows
	164		4 Rows

9.16.4 Empty Faces

When there are more faces, than images, it is possible to have empty faces appear on the chart. This situation occurs in one of two ways:

- The chart data set exists but the image does not
- The number of rows in the chart data set are less than the number of available faces on the shape



Note: If empty faces are not desirable, and the data set is accurate and the number of images is accurate, then consider enabling the property 'Allow Empty Faces'. This will place copies of existing images on the shape until all faces are filled.

9.16.5 Chart Properties

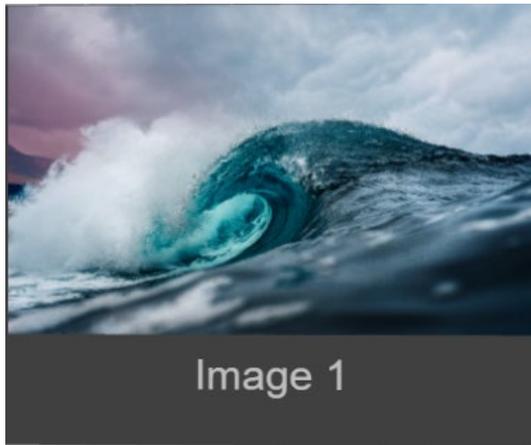
- **Faces** – Each shape has a predetermined number of faces that geometrically make up the shape. All faces are flat, regardless of the shape. Faces are interactive with the cursor (or finger via touchscreen) and can be centered by clicking on the face.
- **Face Color** – This property defines the color of a face when no image appears on the face. This color is always seen during the initial rendering of the chart but might be completely hidden when all faces are displaying an image.
- **Border Color** – This property defines the color of the border between each face (face edges). This color is always seen during the initial rendering of the chart and after all images have been displayed.
- **Highlight Color** – This property customizes a subtle effect which indicates interactivity with the cursor or another chart. The color appears as a transparent layer over the image and temporarily overrides the border color.
- **Allow Empty Faces** – If the data set is accurate and the number of images is accurate, but not all faces have been populated with an image, then consider enabling this property. When enabled, this will place copies of existing images on the shape until all faces are filled. This could result in a shape that does not have unique images. For the Thumbnail Carousel chart, this setting is the default section and hidden.
- **Allow Faces to Switch Images** – If the data set (and available images) contains more combinations than the number of faces, then this option can be used to load additional images every time a face is rotated away from view. For the Thumbnail Carousel chart, this setting is not selected and hidden.
- **Click Behavior** – This property controls how to customize the experience when a face is selected and the chart has been configured with a drilldown target.
 - **Zoom then Drilldown** – With this setting, the first click will center and zoom upon the selected image. The second click will perform the drilldown.
 - **Zoom and Drilldown** – With this setting, the first click will center and zoom upon the selected image and simultaneously perform the drilldown. This may not be desirable if the target chart is going to appear within the same frame. However, this might be desirable if the drilldown is not going to remove the 3D chart from its frame.
 - **Drilldown Only** – With this setting, the first click will perform the drilldown and in no circumstance with the image ever be centered or zoomed upon. For the Thumbnail Carousel chart, this setting is the default section and hidden.
- **Gap Between Faces** (*Thumbnail Carousel Only*) – The property controls the gap space between faces on the Thumbnail Carousel charts. Values are from 0 to 100, with 0 being the default.
- **Selection Speed / Selection Bounce / Rotational Friction** – These properties change the experience of the chart when the shape is being clicked or dragged. Each one simulates a physical property and can also adjust the animation effect.



- **Use Auto Rotation** – By default, this chart is configured to automatically rotate. The Delay, Speed and Direction settings can be adjusted to provide a variety of visual effects.
- **Image Label Settings** – These settings control if, what, and how, the label appears when interacting with a face. The label settings are divided whether the face is being 'hovered' over or if the face has been selected and appears centered and zoomed.
 - Hover Label



- Face Label (*Thumbnail Carousel Only*)



- Zoom Label



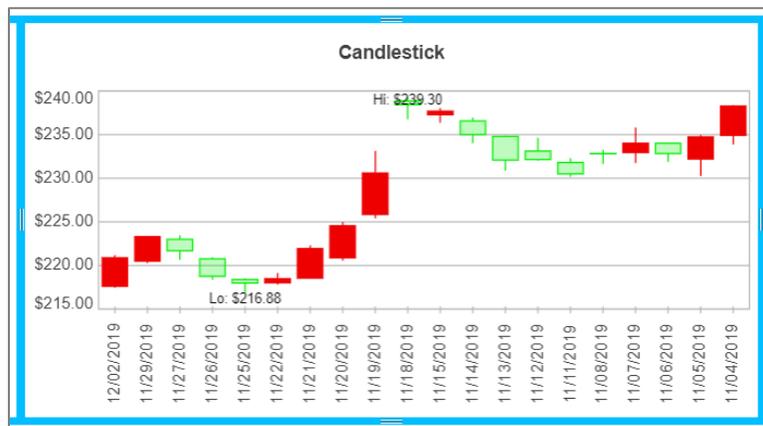
9.17 Candlestick Chart

A candlestick chart is commonly used to track the fluctuation of a stock's price over time.

Each row of data is rendered as a “candle,” which is a red or green rectangle, with possibly a “wick”, which is a same-colored line, protruding from its top and/or bottom. Red rectangles are filled, while green rectangles are displayed as outlines only.

For green candles, the bottom of the candle indicates the opening price of the stock and the top indicates the closing price for the indicated time period. For red candles, the opposite is true. (In other words, green candles indicate a net price move upward during the time period, while red candles indicate a net move downward.) Although the red and green colors can be changed, outline candles are always used to indicate net increases and solid candles are always used to indicate net decreases. This makes the chart useful in situations where distinguishing colors is difficult.

For both green and red candles, the tip of the upper wick indicates the period's high price, and the tip of the lower wick indicates the period's low price. The absence of a wick means the high or low occurred at the opening or closing of the period, whichever is appropriate given the color of the candle.

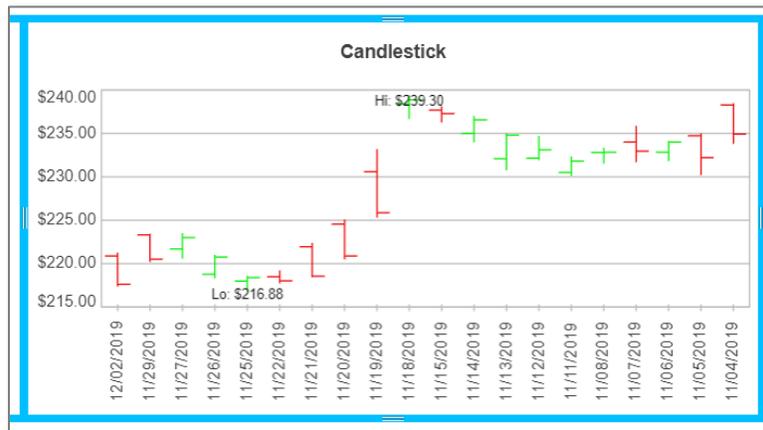


9.18 OHLC Chart

“OHLC” is shorthand for Open-High-Low-Close. This chart type is very similar to a candlestick chart. It is used to track the fluctuation of a stock’s price over time.

Each row of data is rendered as a thin vertical bar, with a short horizontal line intersecting its left side and its right side. The top of the bar indicates the stock’s high price during the indicated time period, and its bottom indicates the low price during the period. The horizontal line intersecting on the left indicates the opening price for the period, and the one intersecting on the right indicates the closing price.

The color of the bars can be configured, with different colors used to indicate net upward or downward moves. The default colors are green for upward moves or red for downward moves.



9.19 Word Cloud Chart

This chart displays individual words in a size represented by the word occurrence count.

9.19.1 Data Layout

A major characteristic which makes this chart unique is the ability to consume two different types of data sets: pre-processed data or unprocessed data. The only thing to distinguish the difference between each technique is the structure of the incoming data set, configured within the Chart Designer.

9.19.1.1 Pre-processed Data

Pre-processed data can be helpful if there are special characters, or phrases, that must be preserved. Pre-processed data requires the following data conditions:

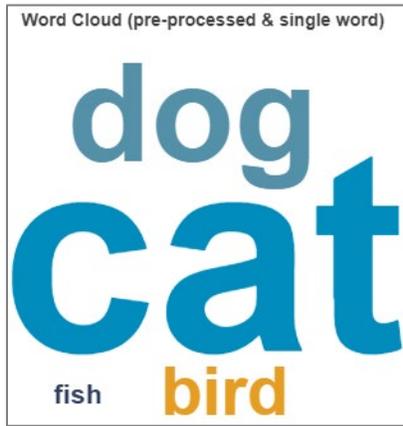
- **X-Axis:**
 - Must be the data type STRING
 - The value can contain multiple words separated by any character (like space) and will be considered a single value within the chart
- **Y1-Axis:**
 - Must be the data type NUMBER



- The value of the number will be used to determine the size of the word

9.19.1.1.1 Example 1

Axis Position	X axis	Y1 axis
Data Type	String	Number
Axis Label	Word	Count
	Cat	100
	Dog	50
	Bird	25
	Fish	10



9.19.1.1.2 Example 2

Axis Position	X axis	Y1 axis
Data Type	String	Number
Axis Label	Word	Count
	Fierce Cat	100
	Huge Dog	50
	Big-Bird	25
	Slippery, Wet Fish	10

Note: Using the pre-processed method will preserve words, phrases, sentences, etc. and will not alter the X-Axis value.



9.19.1.2 Unprocessed Data

Unprocessed data uses the iDashboards algorithm for counting individual words. Using this method will introduce a secondary dataset, which will drastically change the entire structure, and function, of the chart. Drilldowns, Highlighting, Match-Type, X/Y axis reference, will all change during the processing.

- **X-Axis:**
 - While the chart still requires an X-Axis, the data column will be ignored.
- **Y1-Axis:**
 - Must be the data type STRING
 - Can contain single words or multiple words (like sentences or paragraphs)

9.19.1.2.1 Example 1 (unprocessed – “original” data set)

Axis Position	X axis	Y1 axis
Data Type	(anything)	STRING
Axis Label	Anything	Words
	1	The cat is a cat
	2	The cat is not a dog
	3	The cat is not a bird or dog
	4	The cat is not a fish or bird or dog

9.19.1.2.2 Example 1 (Processed – “secondary” data set)

The table below is the “secondary” data set, automatically created by iDashboards to support the Word Cloud chart.

Axis Position	X axis	Y1 axis
Data Type	STRING	NUMBER
Axis Label	Anything	Count
	cat	5
	the	4
	is	4
	a	4



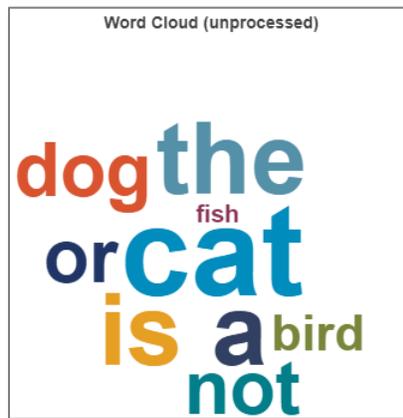
dog	3
or	3
not	3
bird	2
fish	1

Note how the axis data types have changed state!

Note how the X-Axis value was discarded!

Note how the string-words were broken into individual values and moved from the Y-Axis into the new X-Axis!

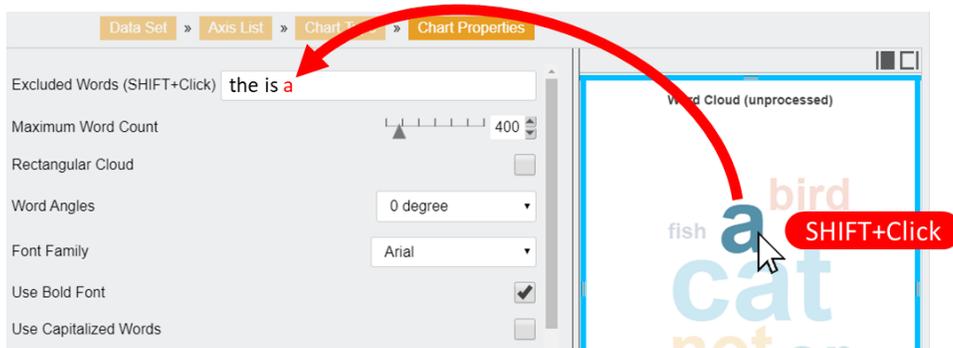
Note how the new Y-Axis has calculated the word count and has a new label!



9.19.2 Excluded Words

When words are displayed, it may be surprising how often a word, of no value, appears. In the above example, the words “the”, “is”, “a”, “or”, and “not” might be unnecessary for the purpose of the chart. Therefore, this chart can exclude these words to emphasize other words.

To exclude a word, type the word into the “Excluded Words” field, and separate each word with a space-character. Alternatively, hold SHIFT and Left-CLICK on the word in the live preview and the word will automatically be removed from view and added to the exclusion list.



9.19.3 Drilldown

Notice!

The Word Cloud chart does not follow normal drilldown practices. Refer to section 8.5.7, “Drilldown” for information on configuring chart drilldowns.

For pre-processed data, this chart follows the same protocol as all other charts. However, charts using unprocessed data have a very different behavior because of the “secondary” data set.

Consider the following example:

Unprocessed		Processed	
X axis	Y1 axis	X axis	Y1 axis
A Number	Words	A Number	Words
1	The cat is a cat	cat	5
2	The cat is not a dog	the	4
3	The cat is not a bird or dog	is	4
4	The cat is not a fish or bird or dog	a	4
		dog	3
		or	3
		not	3
		bird	2
		fish	1

In the example above, the drilldown setup will continue to show the original axis names, yet the data values will NOT be from the “original” (unprocessed) data set, but rather from the “secondary” (processed) data set. So, to drilldown on a word will require sending the axis “A Number”. Similarly, in rare situations the number may be the desired drilldown value and would require sending the axis “Words”.

9.19.3.1 Drilldown on Word Example

Drilldowns within iDashboards have the ability to take the value of an axis (or axes) and send it onto another entity like a chart, dashboard, URL, or even a parameter. The unique behavior of using drilldowns with unprocessed data sets was discussed above. However, it is possibly desirable to drilldown on a word, like “fish” and see the “original” (unprocessed) data set in the form of a chart. To achieve this requires a unique setup, dynamic data sets, and custom queries.

Note: For a “how to” guide on this process, check OSKAR (see Chapter 2, “OSKAR”).



10. Advanced Chart Features

10.1 Using the Pivot Feature

Pivots are a feature that can be added to any chart that allows a user to, in real time, select values to filter the chart on. This gives the user the ability to view various slices of the multidimensional data the chart represents. When a user selects a specific value in a pivot, the chart refreshes with information filtered for that specific value. The number of pivots that can be designed into any chart is unlimited.

Using pivots is optional. A pivot is created when you designate one or more of your Y-axis as a Pivot. The ability to assign a pivot is only activated when there are at least 2 or more Y-axes. If there are 3 or more Y-axes, multiple pivots may be applied.

10.1.1 Creating a Pivot

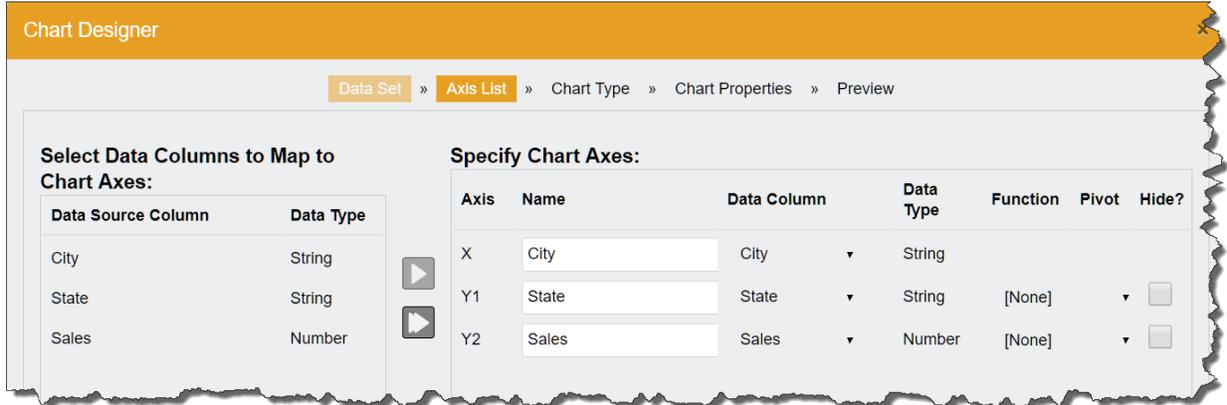
To create a chart with a pivot, you must have the Builder or Admin role. The table shows sample data with “Sales” as a measure for various branch offices in various states. The objective is to create a chart with “State” being a pivot, and depending upon the selected value of “State” in the pivot, the chart displays the Sales in the cities with that specific “State”.

City	State	Sales
Los Angeles	CA	4282208
Modesto	CA	2470186
Oakland	CA	6760205
Sacramento	CA	6370753
San Diego	CA	2035491
San Francisco	CA	5526442
Denver	CO	138792
Boulder	CO	89841
Fort Collins	CO	134465
Grand Junction	CO	248351
Englewood	CO	81726
Lafayette	CO	68702
Belleville	IL	593629
Bloomington	IL	1421385
Chicago	IL	1381381
Joliet	IL	1194179
Rockford	IL	782143
Springfield	IL	570421
Ann Arbor	MI	2238605
Detroit	MI	2264710
Grand Rapids	MI	2933387
Lansing	MI	1079135
Midland	MI	1315891
Traverse City	MI	2609111

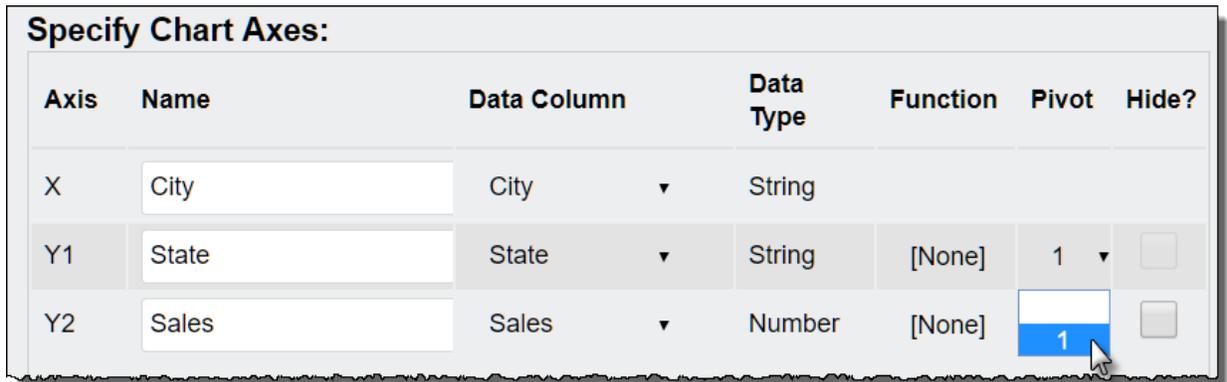
City	State	Sales
Deluth	MN	295975
Eagan	MN	192835
Minneapolis	MN	187543
Rochester	MN	292374
St. Cloud	MN	254584
St. Paul	MN	184154
Albany	NY	2649662
Buffalo	NY	2253849
New York City	NY	2632417
Rochester	NY	2547807
Syracuse	NY	3515153
Utica	NY	3344057
Austin	TX	132243
Dallas	TX	87885
Fort Worth	TX	117291
Houston	TX	119071
Plano	TX	83890
San Antonio	TX	55598



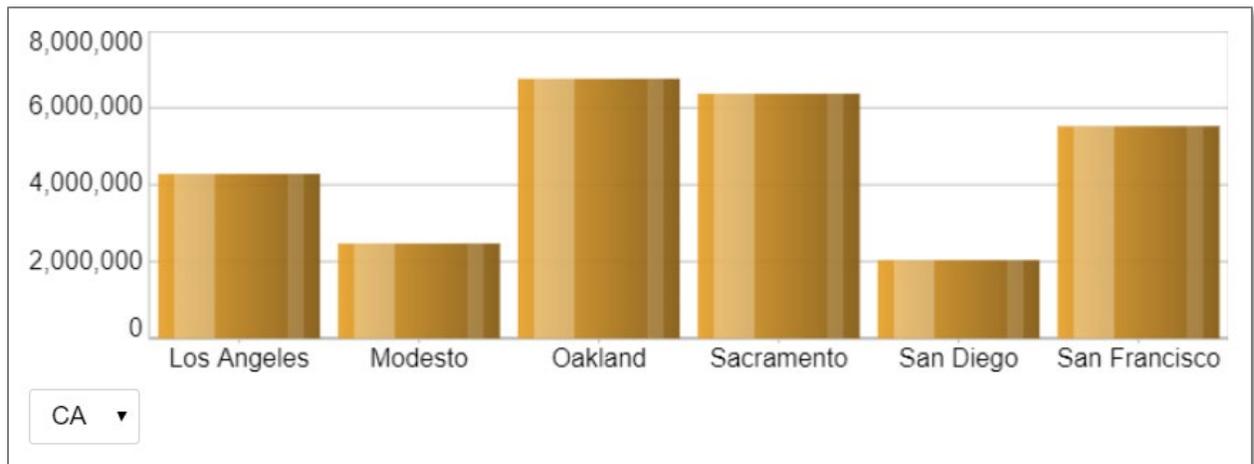
First, connect to the Data Set that includes the sample information. For the Axis List, add all of the columns. The Chart Designer should appear as shown below:



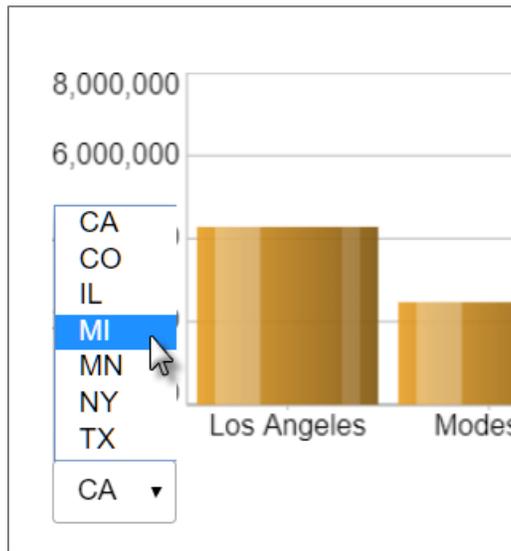
To create a pivot on “State”, you need to expand the Pivot dropdown box for the Y-axis “State” and select “1”, as shown below:



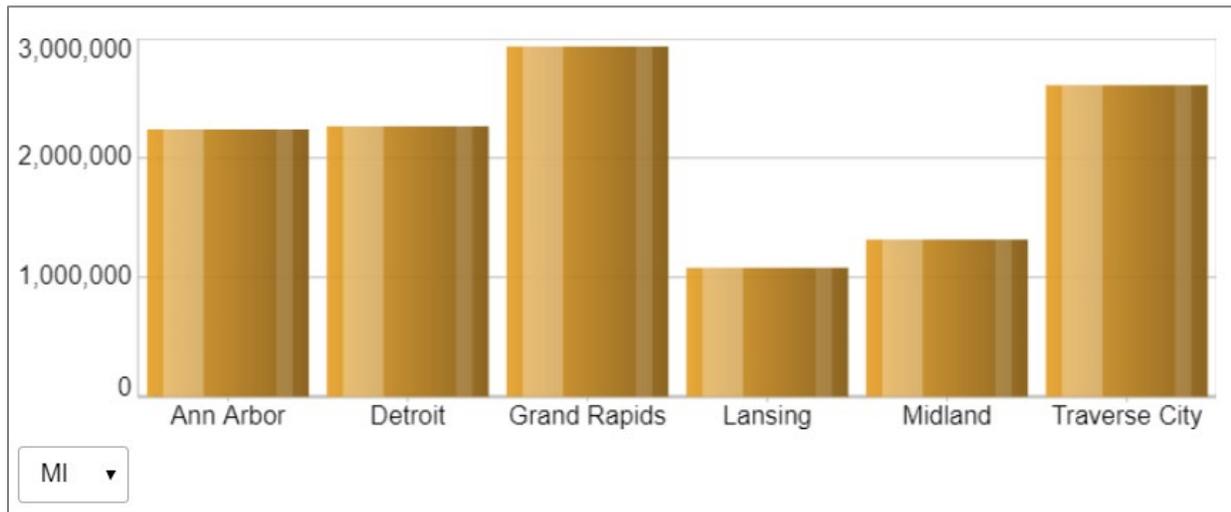
Complete the chart creation process by selecting a chart type. This example shows a Clustered Column chart. When the chart is drawn, there will be a pivot on the “State” data (see below).



Clicking on the pivot will bring up a popup menu of all distinct items in the pivot column (see below)

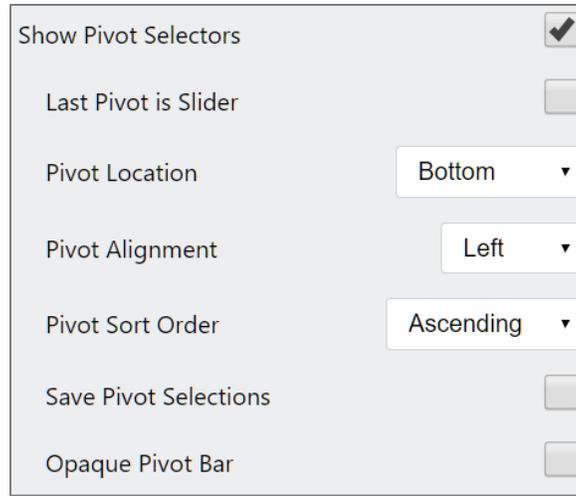


If you select a different value in the menu, the chart reloads to reflect data pertinent to the new pivot value (see below)



10.1.2 Customizing Pivot Appearance

The appearance of Pivots can be customized within the common chart properties. The available Pivot options will only show up in chart settings if the chart contains one or more pivots. If the chart doesn't contain a pivot(s), these options will not be visible.



- **Show Pivot Selectors** – This option is checked by default. If you uncheck it, your pivots will still exist in the chart but they will not be visible for a user to select and change their value(s). The example situation you would want to hide the pivot selectors is if you don't want the pivot selectors to take up screen real estate on your chart AND the chart is a target chart for a drilldown. With the chart acting as a target chart for a drilldown, its pivot values can be selected even with the pivot selectors hidden. Another example is when using the Drilldown feature "Change Other Charts' Pivots"
- **Last Pivot is Slider** - This option is unchecked by default. If you check it, the last Y-axis that is designated as a pivot will be shown as a slider bar, not a dropdown list. If only one pivot exists in the chart and you check this option, it will be shown as a slider bar.
- **Pivot Location** – You can designate where the pivot selector(s) is placed; bottom of the chart (default), top of the chart or under the chart title.
 - **Relocate** - Pivot location can also be set using the chart menu button. If dashboard properties are set to 'Show chart-menu buttons', select the button and the select 'Pivot Location'.

Changing the pivot location using these methods will not invoke a save to the chart. Therefore, viewers could temporarily change the pivot location. If the updated location needs to be permanent, save the chart.

- **Pivot Alignment** - You can designate how the pivot selector(s) is aligned; left (default), right or center.
- **Pivot Sort Order** - Sorting the pivot allows chart data to be sorted in a pre-defined order. The data can be sorted ascending, (a-z, 1-10) or descending (z-a, 10-1). Pivot data can be sorted by choosing the column in the "Pivot Sort Order" dropdown menu. The sort order is ascending by default.
- **Save Pivot Selections** - This option is unchecked by default. With it unchecked, every time you open or refresh a chart that contains a pivot(s), the pivot selectors will display the first pivot selection from the data source, no matter what values were selected the

last time the chart was saved. If you check this option, every time you open or refresh a chart that contains a pivot(s), the pivot selectors will be set to the values that were selected the last time you saved the chart. To set the default pivot value, navigate to the Preview stage and pick the desired value.

- **Opaque Pivot Bar** - If checked, this option will make the pivot bar transparent, therefore allowing any dashboard color or image to show through where the pivot selectors are displayed. If unchecked, the chart background color will be displayed.

10.2 Using the Hide-Axis Feature

Hiding an axis is optional when building charts. While not a common function, there are certain circumstances which greatly benefit from hiding an axis. Hiding an axis requires creating a data column within the chart, but then suppressing the visibility of the data within the chart. Hiding an axis occurs within the Chart Designer when configuring the Axis List. Hiding is sometimes used when sorting data or in conjunction with an Analytics feature.

10.2.1 Example 1: Detailed Sorting

Examine the dataset below. This business has 5 locations, where Detroit is the headquarters. The goal is to have Detroit appear at the topmost position within a Bar Chart, and the other locations sorted based on other factors. All charts within iDashboards have the ability to sort. However, alphabetically (using the 'Locations' axis), the data point "Detroit" will not correctly sort. Numerically (using the 'Output' axis), Detroit could appear in the correct position, unless the output becomes less than the output of another location.

locations	output
Chicago	671395
Detroit	8542672
Cleveland	7421415
Toledo	254003
Ann Arbor	426853

One solution is to add an additional column to the dataset, specifically detailing the exact sort order for each location.

locations	output	sorting
Chicago	671395	2
Detroit	8542672	1
Cleveland	7421415	5
Toledo	254003	4
Ann Arbor	426853	3

Then, within the Chart Designer > Axis List, add the sorting column followed by checking the "Hide" setting on the same sorting column.



Specify Chart Axes:

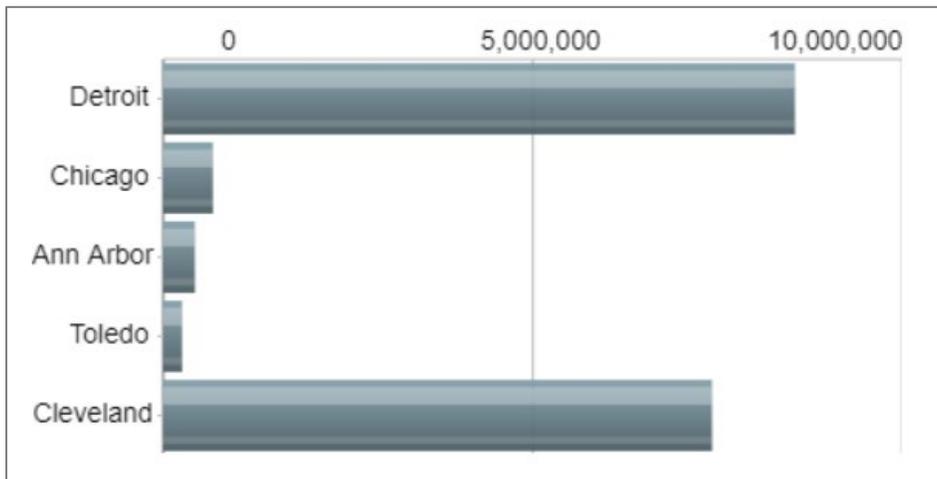
Axis	Name	Data Column	Data Type	Pivot	Hide?
X	Locations	locations ▼	String		
Y1	Output	output ▼	Number	▼	<input type="checkbox"/>
Y2	Sorting	sorting ▼	Number	▼	<input checked="" type="checkbox"/>

Lastly, within the basic chart settings, configure the sorting as shown below. The outcome will show Detroit at the top, followed by the sorting values as they appear in the data set.

Sort On Sorting ▼

Sort Order Ascending ▼

Sorting Results

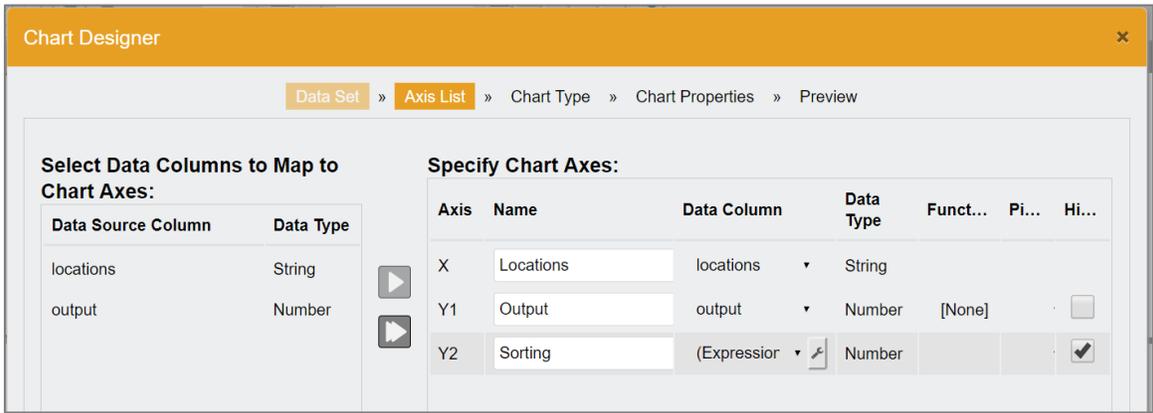


10.2.2 Example 2: Basic Sorting using Expressions

This example provides a simple technique which acts as the foundation for more advanced concept. Examine the previous example. Imagine there is a simpler need for placing Detroit at the top of the chart, and all other locations can appear in non-sorted fashion. Or, maybe it is not possible to adjust the dataset to include an additional column for sorting.

Within iDashboards, while designing the chart, an axis can be created even though there is no data column to map to. Using an expression, Detroit can be assigned a sort value of “1” while all other locations are assigned a sort order of “2”.

Axis List:



Expression:

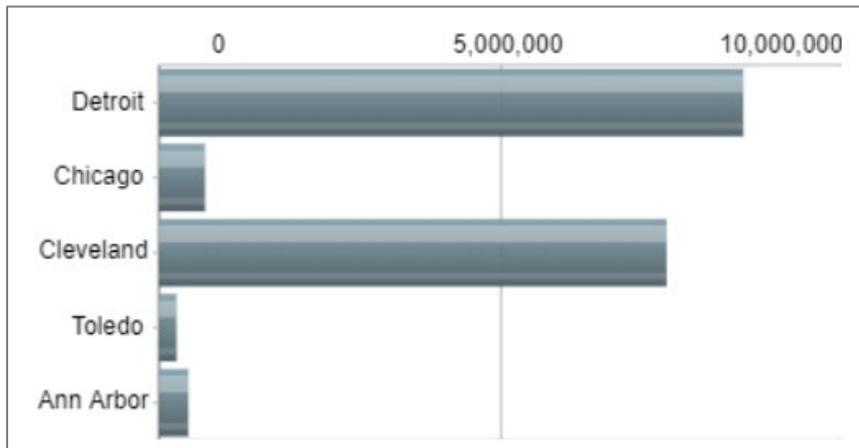
```
if (x=="Detroit") {1}
else {2}
```

Data Output:

locations	output	sorting
Chicago	671395	2
Detroit	8542672	1
Cleveland	7421415	2
Toledo	254003	2
Ann Arbor	426853	2

Chart Results:

Once sorting is applied on the 'sorting' column, this technique can create the chart shown below. Detroit is at the top, and the other locations are unsorted.



10.3 Using the Function Feature

iDashboards provides the ability to aggregate Y-axis data columns by using the “Function” feature within the Chart Designer. For each of the available Functions (SUM, AVG, MAX, MIN, and COUNT), the function will be performed on the Y-axis values, which are automatically grouped by each unique X-axis value. For example, if you have a table with 10,000 rows of data and a “City” X-axis with only 6 unique cities, the function will aggregate all of the rows of data for each city, resulting in a result set of only 6 rows.

Note: The Datetime or String column types can only use the COUNT function option

Consider the following dataset. It shows sales figures for various cities, in 7 states.

City	State	Sales	City	State	Sales
Los Angeles	CA	4282208	Deluth	MN	295975
Modesto	CA	2470186	Eagan	MN	192835
Oakland	CA	6760205	Minneapolis	MN	187543
Sacramento	CA	6370753	Rochester	MN	292374
San Diego	CA	2035491	St. Cloud	MN	254584
San Francisco	CA	5526442	St. Paul	MN	184154
Denver	CO	138792	Albany	NY	2649662
Boulder	CO	89841	Buffalo	NY	2253849
Fort Collins	CO	134465	New York City	NY	2632417
Grand Junction	CO	248351	Rochester	NY	2547807
Englewood	CO	81726	Syracuse	NY	3515153
Lafayette	CO	68702	Utica	NY	3344057
Belleville	IL	593629	Austin	TX	132243
Bloomington	IL	1421385	Dallas	TX	87885
Chicago	IL	1381381	Fort Worth	TX	117291
Joliet	IL	1194179	Houston	TX	119071
Rockford	IL	782143	Plano	TX	83890
Springfield	IL	570421	San Antonio	TX	55598
Ann Arbor	MI	2238605			
Detroit	MI	2264710			
Grand Rapids	MI	2933387			
Lansing	MI	1079135			
Midland	MI	1315891			
Traverse City	MI	2609111			

Displayed as a tabular chart below, the configuration would return chart data as a SUM of SALES, grouped by STATE; displayed using only 7 rows of data. If SALES wasn’t summed, the chart would display a row of sales data for each city; looking the same as the original data set.



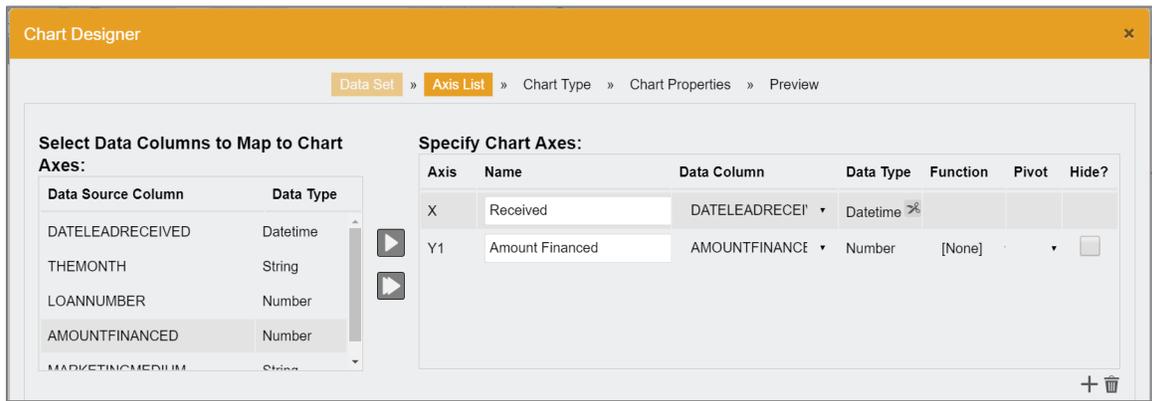
State	Sales
CA	\$27,445,285
CO	\$761,877
IL	\$5,943,138
MI	\$12,440,839
MN	\$1,407,465
NY	\$16,942,945
TX	\$595,978

10.4 Truncate Dates

This task is used to truncate at least one Datetime column in either the X or Y-Axis position. This task is only available when the data type of a column is Datetime. A truncation will return the datetime value nearest to the selection option.

Date truncation returns the date (or time) portion on the date part that was selected; Such as year, month, day, hour or minute. Selecting 'Year' returns the first day of the specified year. Selecting 'Month' returns the first day of the specified month. This will generally create duplicate date values which can later be aggregated.

To set up date truncation for a column, click the scissors icon that appears next to Datetime columns.



- **Year:** turns this "2020-03-12 10:45:15.012" into this "2020-01-01 00:00:00.000"
- **Month:** turns this "2020-03-12 10:45:15.012" into this "2020-03-01 00:00:00.000"
- **Day:** turns this "2020-03-12 10:45:15.012" into this "2020-03-12 00:00:00.000"
- **Hour:** turns this "2020-03-12 10:45:15.012" into this "2020-03-12 10:00:00.000"
- **Minute:** turns this "2020-03-12 10:45:15.012" into this "2020-03-12 10:45:00.000"
- ***Quarter:** Similar to 'Month' but only identifies with January, April, July and October
- ***Week:** Similar to 'Day' but only identifies with the weekday Monday

*Note: * Indicates the truncation is available only with Oracle and SQL Server data sources.*

10.5 Using Filters

Filters can be added to charts which have a dynamic data source (not imported or created data). If using a custom query, the filter needs to be specified within the query. Filters can be a logical AND or OR condition if filtering on multiple columns. The default logic is AND. Filters can be created during the creation of a new chart, or added to existing charts.

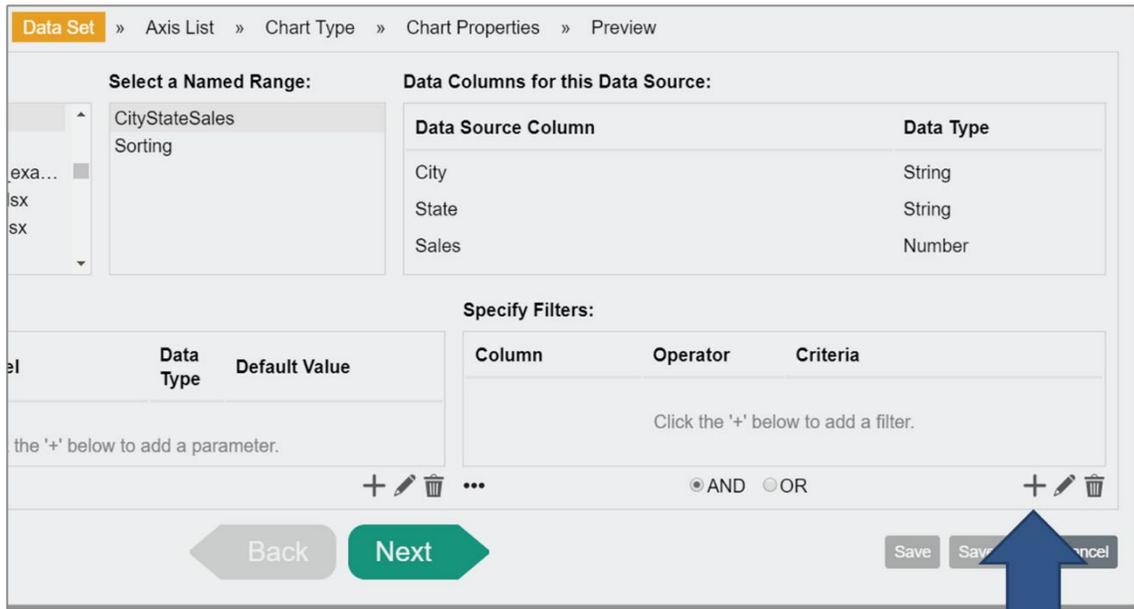
10.5.1 Operators

Assuming $x = 5$

Operator	Description	Example
=	Is equal to	$x=8$ is false
<>	Is not equal to	$x<>8$ is true
>	Is greater than	$x>8$ is false
<	Is less than	$x<8$ is true
>=	Is greater than or equal to	$x>=8$ is false
<=	Is less than or equal to	$x<=8$ is true
in	Is equal to a value from multiple possible values	$x=8,7,6,5,4,3$ will result in the true match =5
not in	Is not equal to a value from multiple possible values	$x=8,7,6,5,4,3$ will result in the true match =8,7,6,4,3
is null	Is equal to a field with no value	
is not null	Is not equal to a field with no value	

10.5.2 Setup Process

1. Navigate to the Chart Designer > Data Set
2. Click the 'plus' icon to add a filter

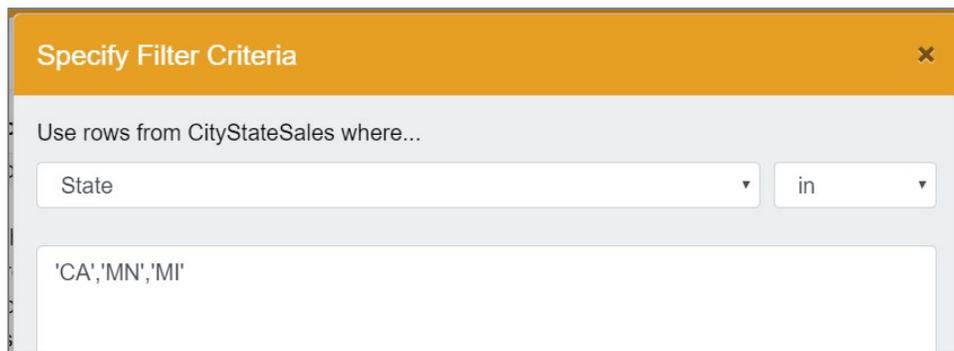


3. In the Specify Filter Criterion window choose the column you want to apply the filter:



4. Make sure the filter operator has the correct option selected (>, <, =, etc.)
5. Either type the filter criteria in the text box provided or click the “List values” button which will display a list of distinct values that exist in the data column you selected for the filter. From this list, you can click on the values you would like to use and they will be added to the text box.

Note: When using “IN” or “NOT IN”, multiple values can be selected from the “Values” list. Alternatively, each value can be provided within single-quotes and comma separated.

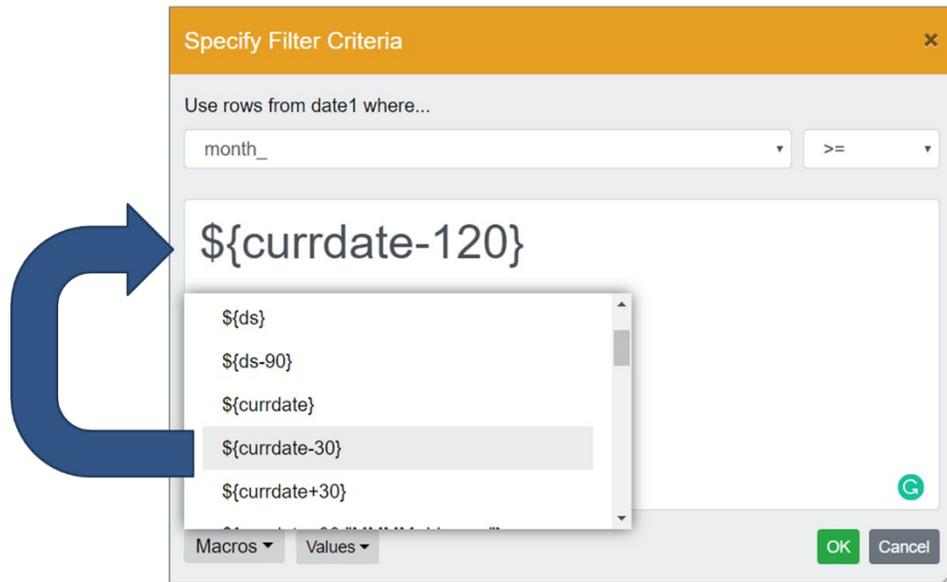


6. Finally click the “OK” button to get back to the Chart Designer window to add more filters as necessary.

10.6 Using Filter Macros

Macros are a feature within iDashboards that allow for dynamic substitution of values. Macros are useful when you would like to indicate a value that changes over time or by context as the criteria for the filter. For example, you want to create a chart that will show you data for a specific period of time relative to today's date and have that date period rolling.

As an example, let's work with revenue figures for a 120-day rolling timeframe. The chart should recognize which day it is and update itself with every new day's data (and exclude the data for dates greater than 120-days ago). The image below shows how this macro would be entered when creating a macro filter.



All dynamic charts within iDashboards essentially send a SQL statement to their data source to retrieve the data they will represent. The SQL statement is automatically built based on how the chart was created. A date macro is used in the filter to create the WHERE clause of this SQL statement.

The following is a list of available date macros within iDashboards are displayed in the table below. All values can be modified to alter the length of time they represent. Macros have the form:

`$$${keyword:shifter +/- modifier, format}`

The 'shifter' will set the keyword value to a specific point, generally the first or last day of a month or year. The 'modifier' is a whole number that can be added or subtracted from the result of the 'keyword' or 'keyword:shifter' value. The 'format' determines how the date or time format will look.

Note: You may edit the numerical modifier in any macro.

Date Macros		
<code>\$\$\${ts}</code>	<code>\$\$\${currhour+5, "hh:mm:ss a"}</code>	<code>\$\$\${currmonth:end - 2}</code>
<code>\$\$\${ts-180}</code>	<code>\$\$\${currmonth}</code>	<code>\$\$\${currmonth:begin + 3, "MMMM d, yyyy"}</code>
<code>\$\$\${ds}</code>	<code>\$\$\${currmonth, "MM"}</code>	<code>\$\$\${curryear}</code>
<code>\$\$\${ds-90}</code>	<code>\$\$\${currmonth, "MMM"}</code>	



<code>#{currdate}</code>	<code>#{currmonth, "MMMM"}</code>	<code>#{curryear-3}</code>
<code>#{currdate-30}</code>	<code>#{currmonth-3}</code>	<code>#{curryear + 2}</code>
<code>#{currdate+30}</code>	<code>#{currmonth + 2, "MMMM"}</code>	<code>#{curryear + 2, "dd MMM yy"}</code>
<code>#{currdate+90, "MMMM dd, yyyy"}</code>	<code>#{currmonth + 2, "MMMM d, yyyy"}</code>	<code>#{curryear:begin}</code>
<code>#{currhour}</code>	<code>#{currmonth:begin}</code>	<code>#{curryear:end}</code>
<code>#{currhour-3, "HH:mm:ss"}</code>	<code>#{currmonth:end}</code>	<code>#{curryear:begin + 4, "dd MMM yy"}</code>

10.6.1 Setting the Proper Date Type and Format

For date macros to work they must be written to work with the data type and format of the column they will be filtering against within the database. The date macro can filter against data columns that are configured as a number or date/datetime. It is common to filter against columns that are set as a date/datetime. However, various databases treat date/datetime columns differently. An error notification should appear if there is a data type or date format conflict between the iDashboards' filter and the database column you are filtering against.

Some databases are more forgiving about date formats than others. The remainder of this section illustrates how to use date macros with two popular data sources; SQL Server and Oracle. Examples are provided.

10.6.1.1 SQL Server

SQL Server is forgiving when it comes to date fields. To filter on a date macro, you may use any of the date macros without concern as to date formatting.

10.6.1.2 Oracle

Oracle is very particular about the format of its date data. If your date data is stored as a date data type, you can use one of the 'timestamp' macros, `#{ds}` or `#{ts}`, or you can use any of the other macros in combination with the Oracle "to_date" function to convert the date macro to the same date format Oracle is using. If you are writing a filter to filter against a data column that stores date information as numbers (i.e. "2013" as the year or "15" as the 3pm hour), you can simply treat these values as numbers and use the date macros as they are without modifying the date format.

Following are common abbreviations used to format dates in Oracle:

Macro	Result
MM	Numeric month (e.g., 07)
MON	Abbreviated month name (e.g., JUL)
MONTH	Full month name (e.g., JULY)
DD	Day of month (e.g., 24)
DY	Abbreviated name of day (e.g., FRI)
YYYY	4-digit year (e.g., 2013)
YY	Last 2 digits of the year (e.g., 13)
AM/PM	Meridian indicator
HH	Hour of day (1-12)
HH24	Hour of day (0-23)
MI	Minute (0-59)
SS	Second (0-59)



10.6.1.3 Date Macro Examples for: SQL Server & Oracle

Current date (5/23/2019) minus 365 days (via 'ds')		
	Macro	Result
SQL Server	<code>\${ds-365}</code>	2018-05-23
Oracle	<code>\${ds-365}</code>	

Current date (5/23/2019) minus 5 days (via 'currdate')		
	Macro	Result
SQL Server	<code>\${currdate-5}</code>	2019-05-18
Oracle	<code>to_date('\${currdate-5}','YYYY-MM-DD HH24:Mi:SS')</code>	

Current time (5/23/2019 10:36:09) minus 120 minutes		
	Macro	Result
SQL Server	<code>\${ts-120}</code>	2019-05-23 8:36:09
Oracle	<code>\${ts-120}</code>	

The first day of the month onward (Today=5/23/2019)		
	Macro	Result
SQL Server	<code>\${currmonth:begin}</code>	2019-05-01
Oracle	<code>to_date('\${currmonth:begin}','YYYY-MM-DD HH24:Mi:SS')</code>	

Current date (5/23/2019) minus 3 years		
	Macro	Result
SQL Server	<code>\${curryear-3}</code>	2016-05-23
Oracle	<code>\${curryear-3}</code>	

Current hour (5/23/2019 10:36:09)		
	Macro	Result
SQL Server	<code>\${currhour}</code>	10
Oracle	<code>\${currhour}</code>	

10.6.2 Testing the Date Format

iDashboards provides a web page that allows you to test the output of various macros you wish to use. Enter the macro you wish to use, along with any special date formatting, and click 'Expand Macro'. The page is located at:

<http://www.yourwebserver.com/idashboards/macrotest.jsp>

Replace with actual URL



Macros have the form:

`${keyword:shifter, format}`

Use this form to test iDashboards macros.

Enter text containing one or more macros:

Macro value: August 01, 2019

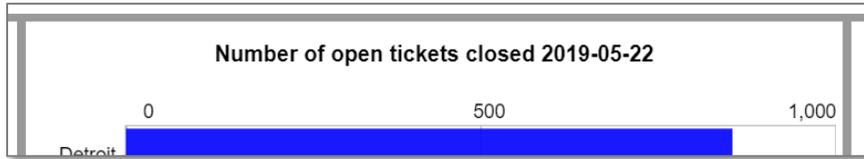
10.7 Adding Macros to Chart Titles

Note: Macros DO NOT work within the chart Name or Description fields.

You can add macros into a chart title, enabling that chart title to display dynamic content. For example, to create a chart with today's date displayed as part of the title, `#{currdate}` could be added to the chart title. For example, the chart title below:

Number of open tickets closed `#{currdate-1}`

...would render as



10.8 Repeat Macro

The repeat macro helps construct complex website URLs for various functions within iDashboards. This macro has the ability to continuously loop through multiple data rows to construct a delimited list for a website URL. In addition, there is also intelligence on the currently active data point and support for leveraging multiple data columns.

10.8.1 Requirements

A requirement for the repeat macro is the website must support delimited list syntax within the URL. Sometimes this functionality is categorized as 'Website API' or 'Website Application Programming Interface'.

10.8.2 Overview

A website URL can be used within certain chart types or chart properties. Charts can use a website URL in the chart properties during the drilldown process. Charts can also use a website URL in the image content reference of the Presentation Charts. The difference in these two options is the destination of the function. Drilldowns to a web page will launch a browser window (outside of iDashboards) and using the presentation charts will keep the next function within the iDashboards instance.

Using the options mentioned above, the repeat macro is an extension onto the ability to use a website URL. The repeat macro will perform a macro substitution to generate a delimited list of values within the URL string.

Note: The iDashboards application handles the URL-encoding so it doesn't need to be explicitly handled in the URL.

10.8.3 Value Macro: Recap

A regular value macro replacement would take this URL string:

```
http://www.search1234.com/search?q=${value:item1}+${value:item2}
```

...and construct a URL like this:

```
http://www.search1234.com/search?q=cat+dog
```

...and the required dataset would need to support this format:

code	item1	item2
0012	Cat	Dog

Notice the URL was constructed from multiple Y-Axes and each animal type would require a unique data column. Therefore, if there are changes to the number of data columns used within the chart, then there will need to be changes made to the URL string also.



10.8.4 Repeat Macro: Syntax

The format of the macro is:

```
[[[directives]toRepeat]]
```

The directives section is optional and specifies aspects of how the 'toRepeat' component will be repeated for every data set row. Knowledge of the website URL is necessary when selecting the proper repeat macro syntax. The directives are:

delim – This directive indicates the delimiter to be used between repeated instances within the resulting expansion. The default is the pipe '|' character.

Inc – This directive indicates whether or not the associated data point should be included in expansion. The data point will be included by default or when the value of this directive is 'all'. If the value is 'others', then the data point will be omitted from the expansion.

Max – This directive indicates the maximum number of data rows that will be used in the expansion. The default is no limit.

10.8.5 Repeat Macro: Drilldown

Implementing the repeat macro occurs when using the 'Drilldown to Web Page' properties of a chart. Clicking on the configured chart will launch an additional browser window based on the URL and repeat macro that was configured. It is only beneficial to use the repeat macro if the dataset within the chart contains more than 1 row of data. The minimum data requirements are anything which could become the X-Axis and Y-Axis of the chart. For example:

code	item1
0012	Cat
0027	Dog

Building off the example in the previous section, if the desired output is:

```
http://www.search1234.com/search?q=cat+dog
```

...then the delimiter needs to be a plus (+) sign and the repeating list of values needs to come from the 'item1' column:

```
http://www.search1234.com/search?q=\[\[delim=+\]\${value:item1}\]\]
```

Notice how the repeat macro will construct a delimited list as long as the number of rows within the dataset.

10.8.6 Repeat Macro: Presentation Charts

iDashboards has the ability to dynamically stream a website image into a Slideshow Chart (or Detail Chart with additional HTML knowledge). The image can change based on every record of data within the chart data set. Using the repeat macro will allow these two chart types to send a delimited list of data points to a website. The result of the website URL must be able to provide an image based on knowledge of all data values.



Note: The website API must be able to dynamically construct and deliver a static image.

Note: Using the Slideshow Chart will likely require enabling the Proxy Server setting within the chart properties.

10.8.6.1 Example 1

Implementing the repeat macro in the image URL first requires knowledge on the necessary output. If the desired output is:

```
http://www.online.Map1234.com/?city=Detroit|city=Grand%20Rapids
```

It is only beneficial to use the repeat macro if the dataset within the chart contains more than 1 row of data. The minimum data requirements are anything which could become the X-Axis and Y-Axis of the chart. For example:

city	routes
Detroit	7
Grand Rapids	3

```
http://www.online.Map1234.com/?city=[[ [ndp=true] ${value:city} ]]
```

10.8.6.2 Example 2

Implementing the repeat macro in the image URL first requires knowledge on the necessary output. If the desired output is:

```
http://www.image1234.com/logo.jpg&blend=checkerboard.jpg&mask=watermark.png&w=600&h=600
```

It is only beneficial to use the repeat macro if the dataset within the chart contains more than 1 row of data. The minimum data requirements are anything which could become the X-Axis and Y-Axis of the chart. For example:

code	item1
blend	blend=checkerboard.jpg
mask	mask=watermark.png
width	w=600
height	h=600

```
http://www.image1234.com/logo.jpg?[[ [delim=&] ${value:Item1} ]]
```

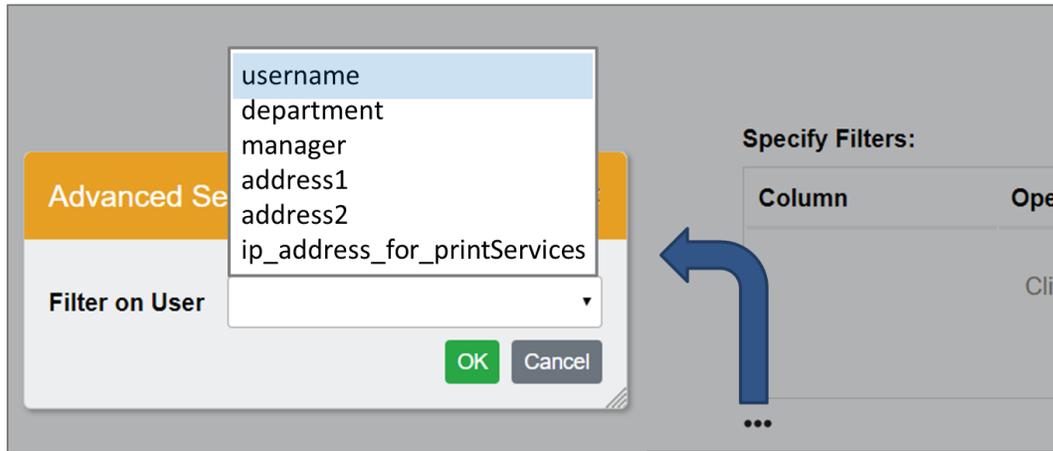


10.9 Filter on User

The majority of the Filter-on-User functionality is configured in the iDashboards Administrator Application and the iDashboards Repository itself.

Note: Refer to the Administrator's Manual for feature overview and details

Once the administrative tasks have been performed, Filter on User can be found within the Chart Designer > Data Set > Specify Filters, as shown below:

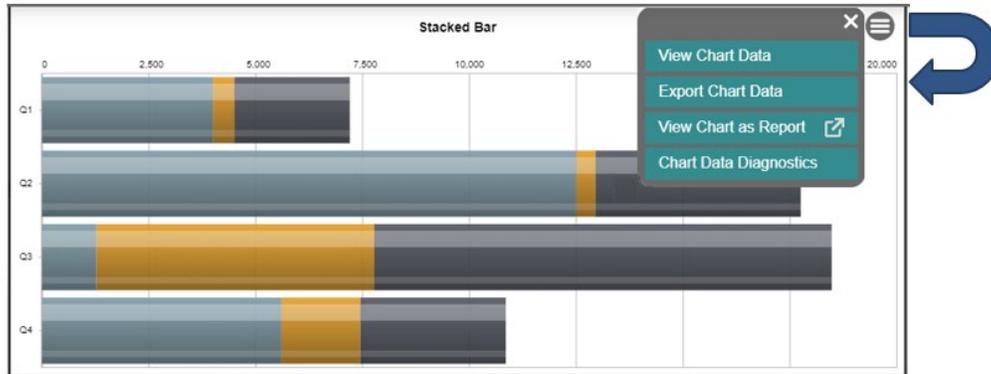


If the dropdown for 'Filter-on-User' is not shown, then the feature has not been configured (or was configured improperly). This setup requires someone who has the function of an iDashboards administrator before the feature can be configured within a chart.

The Filter-on-User functionality allows multiple users to view the same dashboard but the data they see will be pre-filtered to only show data for their logged in username. For example, two Regional Managers can log on to iDashboards and view the same "Sales" dashboard, but manager A will only see data for his territory and manager B will only see data for her territory.

10.10 Export Chart Data

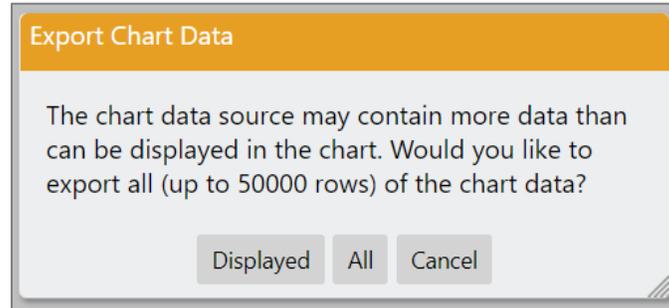
iDashboards provides an easy method to export data into a comma separated value (CSV) for use with other applications. To export chart data, select the chart menu button and choose 'Export Chart Data'



Note: The "Menu Button Visibility" setting can be determined within the Dashboard Designer or the Chart Designer.

10.10.1 Export All Chart Data

When selecting 'Export Chart Data' on a chart that returned more than the maximum number of data rows (1,000 standard or 3,000 with pivots), users will receive an option to export 'All' or only the 'Displayed' rows.



Note: The maximum rows during export is controlled by an administrator setting.

Note: Administrative settings can increase the 1,000-row limit to 3,000 rows

11. Deleting

Warning!

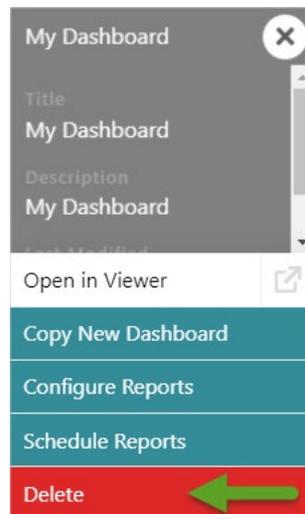
You may delete an existing chart or dashboard only from Categories to which you have Save permissions. There is no mechanism within iDashboards to recover deleted chart or dashboards.

11.1 Delete

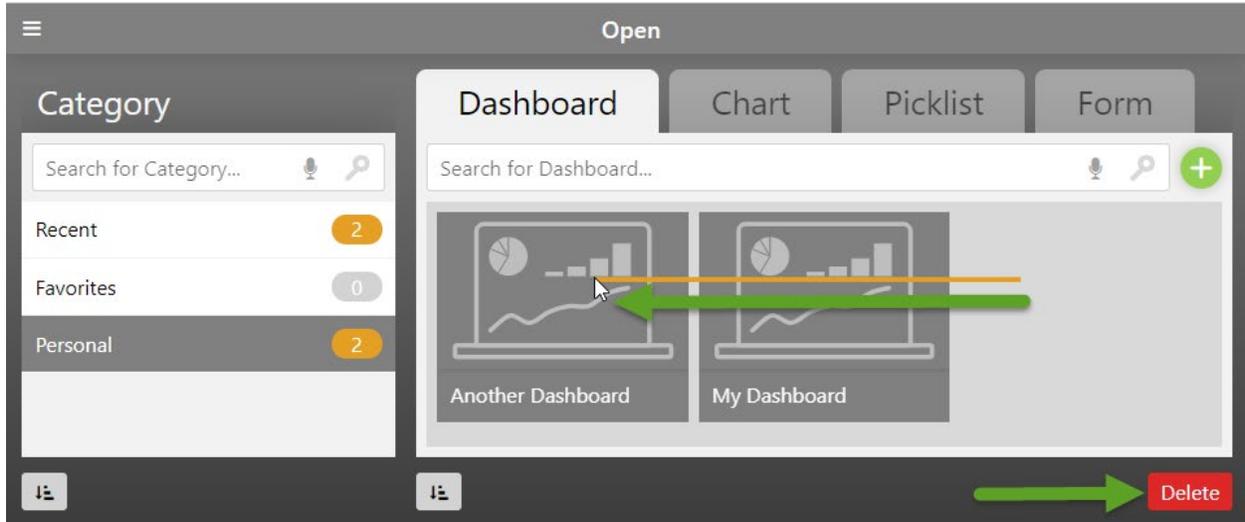
From the Build Interface, dashboards, charts, picklists and forms (if enabled) can be deleted. Deleting is not allowed from the “Recent” or “Favorites” categories. Deleting Categories is not possible from this interface (see “Admin”).

To delete an item, first select the appropriate tab (Dashboard, Chart, Picklist or Form). Next select the category. Next, select the item to delete (multiple items can be selected, searching can be used, etc.).

For a single item select “Delete” from the items control list.



For multiple items, click and drag the cursor over the items and select the “Delete” button at the bottom of the screen.



Either way, a confirmation dialog will appear for each item selected.

Note: Alternatively, items can be deleted from the Administrative interface (accessible only by those with the 'admin' role).

11.2 Delete Sequence

Linked items may result in the inability to delete an item. For example, a picklist that is used by a chart(s) or dashboard(s) cannot be deleted until the 'unlinked' from those charts, dashboards and forms (if enabled).

12. Diagnostics

Diagnostics provide information about how the data displayed within a given chart was obtained. The diagnostics reported vary based on the data source type, whether the chart has filters or parameters, and whether the chart is a drilldown target.

Chart Data Diagnostics
✕

Chart Data Selection

Data Source: KenRose_idbData
Schema: dbo
Table Name: GOV_City_Planning

Executed SQL Statement
SELECT [bldg_name], SUM([agency_revenue]) FROM [dbo].[GOV_City_Planning]
WHERE [agency] <> 'Airport' AND [bldg_name] is not null GROUP BY [bldg_name]

Number of Rows Selected: 8

Chart Data Filters

Defined Data Filter agency <> 'Airport' AND bldg_name is not null
Applied Data Filter [agency] <> 'Airport' AND [bldg_name] is not null

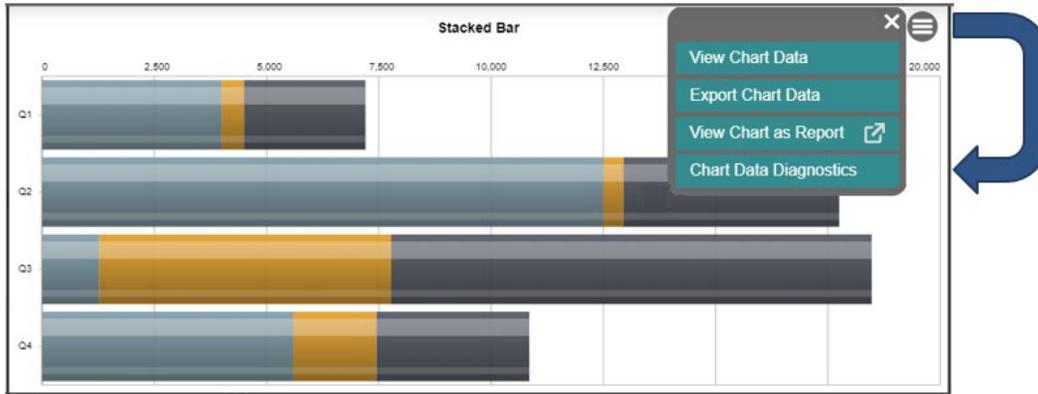
Chart Axis Definitions

Name	Data Type	Value Definition
Agency	String	bldg_name
Agency Revenue	Number	SUM(agency_revenue)

Save...
Copy
Close

12.1 Accessing Chart Data Diagnostics

To access the diagnostics, select the chart menu button and choose 'Chart Data Diagnostics'



Note: The “Menu Button Visibility” setting can be determined within the Dashboard Designer or the Chart Designer.

Note: Diagnostics can be disabled for guest users or users with the ‘viewer’ role within the Administrative console.

12.2 Diagnostics Sections

The ‘Chart Data Selection’ section and the ‘Chart Axis Definitions’ section always appears in the diagnostics information. Other types of chart features will provide additional details in the diagnostics. Listed below are details on each section that may appear in the diagnostics.

Section Name	Display
Chart Data Selection	All Data Source Types
Chart Axis Definitions	All Data Source Types
Drilldown Filters	All Data Source Types, Drilldown Target Only
Chart Parameters	All Dynamic Data Source Types, Parameters Only
Chart Data Filters	All Dynamic Data Source Types except Custom Query, Filters Only
Chart Data Arguments	Stored Procedures and Feeds only, Arguments Only

Note: For example, if a user views the diagnostics on a drilldown target chart, then the Drilldown Filters section will display any values used as a filter.

12.2.1 Chart Data Selection

This section always appears in the diagnostics for all data types. It contains general information about how the data is selected for the chart. At a minimum, items include: ‘Data Source’, ‘Executed Query’, and ‘Number of Rows Selected’. Additional items appear based on the construction of the chart (i.e. ‘Schema’, ‘Table Name’, ‘Named Range’, etc.)

12.2.2 Chart Axis Definitions

This section always appears in the diagnostics for all data types. The axis name and data type are displayed as well as the value definition used to provide the data for the axis. When the axis is from a direct column mapping, the name of the column is given. When it is an aggregate function, the applied aggregate function is displayed. When it is an expression, the defined



expression is shown and the expanded expression with parameter substitution is shown in the tooltip

12.2.3 Drilldown Filters

This section is present when the chart is a drilldown target. It lists the drilldown filters indicating the name of the filter, the value of the filter and whether or not it was applied (i.e. whether it matches an axis of the current chart). More information regarding the drilldown filter value is available via the information button “i”. The additional information displays the drilldown chain and where in the chain the filter value came from.

12.2.4 Chart Parameters

This section is present when the chart has parameters defined. The parameters are listed indicating the name of the parameter, the parameter label, the data type of the parameter and the value of the parameter.

Note: Parameters are not available on static charts.

More information regarding the parameter value is available via the information button “i”. A message indicating the source of the parameter value is indicated. Possible parameter value sources are:

- Source chart parameter (Drilldown target only)
- Source chart drilldown filter (Drilldown target only)
- Source dashboard drilldown parameter (Drilldown target only)
- Dashboard parameter
- Dashboard parameter default
- User supplied
- Parameter default

If the chart is a drilldown target, the additional information displays the drilldown chain and where in the chain the parameter value came from.

12.2.5 Chart Data Filters

This section is present when the chart has data filters defined and displays the defined filter criteria, the applied filter criteria with parameter values substituted and macros expanded.

Note: Filters are not available for static data sources or custom queries.

12.2.6 Chart Data Arguments

This section is present when the chart data source has arguments associated with it. Only data feeds and stored procedures have arguments associated with them. The arguments are listed indicating the name of the argument, the defined value of the argument and the applied value of the argument with parameter values substituted and macros expanded.



12.3 Export Diagnostic Details

Diagnostics can be saved as a JSON file or copied to the clipboard. Some use cases for this include archiving the query instructions of a chart. However, this information is primarily used during technical support.



13. Input Parameters

Input Parameters provide the ability to create charts and dashboards whose displayed data is dependent on Input Parameter values provided by the user. The input values are used by charts and dashboards in a variety of ways, including:

- Filtering chart data based on the input values
- Displaying the input values in chart titles
- Using the input values in a chart's custom SQL queries
- Using the input values as parameters for stored procedures
- Passing the input values down during drilldowns
- Implement parameter controls into the URL of a dashboard (for sharing/launching dashboards with specific parameter values)
- Use input parameter values when generating the dashboard screenshot

Input Parameters (single or multiple) can be defined at both the chart and dashboard level. Chart Input Parameters only affect their associated chart. Dashboard Input Parameters can affect every chart contained in the associated dashboard. Dashboard Input Parameters effectively pass their values down to the Chart Input Parameters of the charts within the dashboard.

Note: Input Parameters are only available for charts that use dynamic data, not static data.

13.1 Parameter Definitions

Input Parameters are defined using the following attributes:

- **Parameter Name** – This is the name to be used internally in the chart or dashboard definition to identify the parameter. For example, when you add parameters to a chart filter or chart title, you use this name. It can consist of 1 to 25 letters, numbers or underscores, and must begin with a letter. Blank spaces are not allowed.
- **Parameter Label** – This is the label displayed to iDashboards when the user supplies parameter values. For example, when a user is prompted to enter a value for an Input Parameter, this is the parameter label that will be displayed. Therefore, the label should be user-friendly and formatted nicely. It can contain up to 50 printable characters without leading or trailing whitespaces.
- **Data Type** – This can be one of the three standard iDashboards data types; String, Number or Datetime.
- **Control Type** – This is the type of input selector that will be presented to the user. The available types are Text Box, Dropdown List, Slider, Spinner and Slider/Spinner.



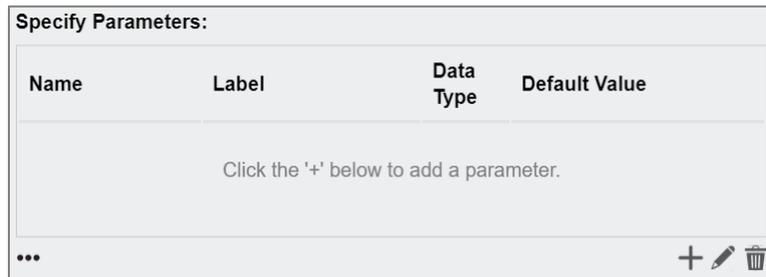
- **Hidden** – Show/Hide chart or dashboard parameters as needed. This will allow parameters to be used but hidden from view. It will also allow some dashboard parameters to appear in one panel and other dashboard parameters to appear in another panel.
- **Required Flag** – This indicates whether or not a value is required for the Input Parameter before the chart or dashboard is displayed. If it is required, and a value is not present via a drilldown or initial value, the user is prompted to enter a value.
- **Replace blank with (optional, charts only)** – If a user submits an Input Parameter with a blank value (i.e. they don't enter a value but still submit), the value in the 'Replace blank with' field will be used instead.
- **Initial Value (optional)** – This is the value that will be used for the Input Parameter if no other value is present.

Note: When a picklist is used with an input parameter, the user can identify the 'Initial Value'. The value entered must come from the 'Value' data column and not the 'Display' data column.

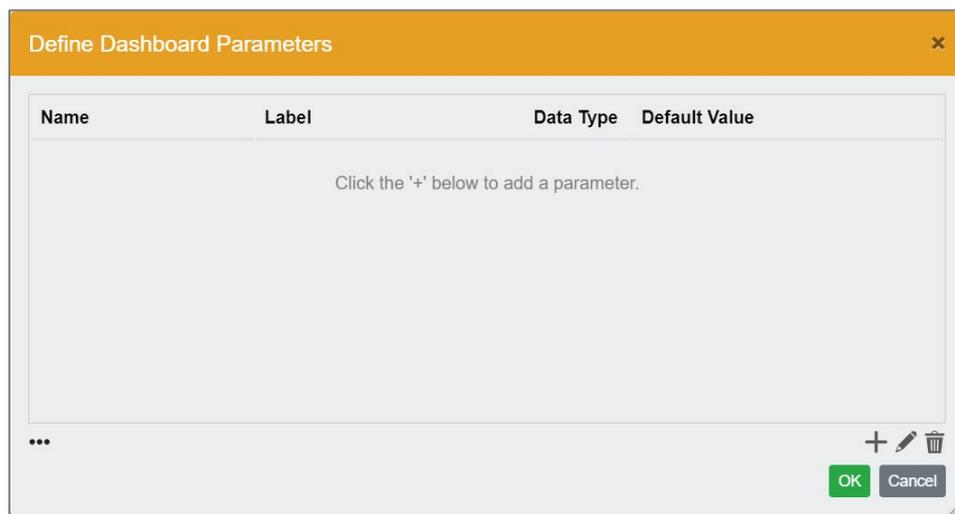


13.2 Configuring Input Parameters

Input Parameters can be configured for both dashboards and charts. Chart Parameters are configured when specifying the chart data; either when identifying the data source columns or defining a custom query. More specifically, Chart Input Parameters can be configured when you first create or edit a chart.



Dashboard Parameters are defined in a similar manner to Chart Parameters. They are accessed within the Dashboard Designer by selecting the “...” icon and then “Define Dashboard Properties”.



Each Input Parameter definition will be displayed as a row in the window. You can order the Input Parameters by drag-and-drop. The order is for display purposes only; It does not affect the parameter behavior.

To configure a new Input Parameter for a chart or dashboard, click the ‘Add’ button. To edit an existing Input Parameter, select the Input Parameter and click ‘Edit’. Either method will open the same Parameter Definition window.

Note: When defining Input Parameters for a dashboard, the Parameter Definition window will not contain the ‘Replace blank with’ field. The ‘Replace blank with’ field only pertains to Chart Input Parameters, and only in certain circumstances.



Note: Parameters can be set to “Hidden” at either the chart or dashboard level. Therefore, it is possible to use parameters without them being displayed on the dashboard.

The required attributes are ‘Name’, ‘Label’, ‘Data Type’ and ‘Control Type’ are required, whereas the other properties are optional. In some circumstances, a certain property could make additional properties to become required.

13.2.1 Passing Dashboard Input Parameters to Chart Input Parameters

As stated earlier, the only real purpose of Dashboard Input Parameters is to pass their values down to the Chart Input Parameters of the charts within the dashboard. To configure a Dashboard Input Parameter that will be passed to the charts within the dashboard, the name and data type of the Dashboard Input Parameter must match the name and data type of a Chart Input Parameter. All matching Input Parameters will be passed from the dashboard to its charts, while non-matching Input Parameters have no impact on the values of the Chart Input Parameters.

13.2.2 Control Types

The available Control Types depend on the Data Type selected.

	Control Types				
	Text Box	Dropdown	Slider	Spinner	Slider/Spinner
String	X	X			
Number	X	X	X	X	X
Datetime	X	X			

For Chart Input Parameters, if you select “Always prompt for parameters on chart load”, every time the chart is loaded, it will prompt the user to enter values for Input Parameters, regardless



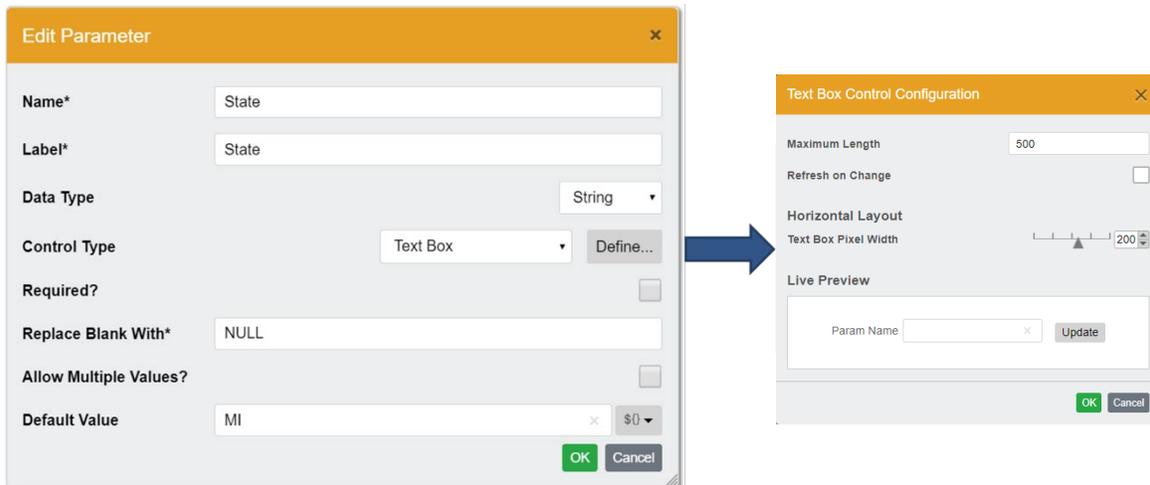
of whether the individual Input Parameters are required or not. For Dashboard Input Parameters, “Always prompt for parameters on dashboard load” works the same way.

13.2.2.1 Text Box

Used for data type String, Number or Datetime, a Text Box allows a user to enter virtually any value.



The Control Configuration window allows for setting a maximum length for the input text as well as to turn on ‘Refresh on Change’. If ‘Refresh on Change’ is selected, the ‘Update’ button will be removed from the chart and the update action will trigger when a user enters a value and hits Tab while in the Text Box or when the Text Box loses focus. A pixel width can also be defined, and previewed, for when the input parameter is displayed using a horizontal layout.

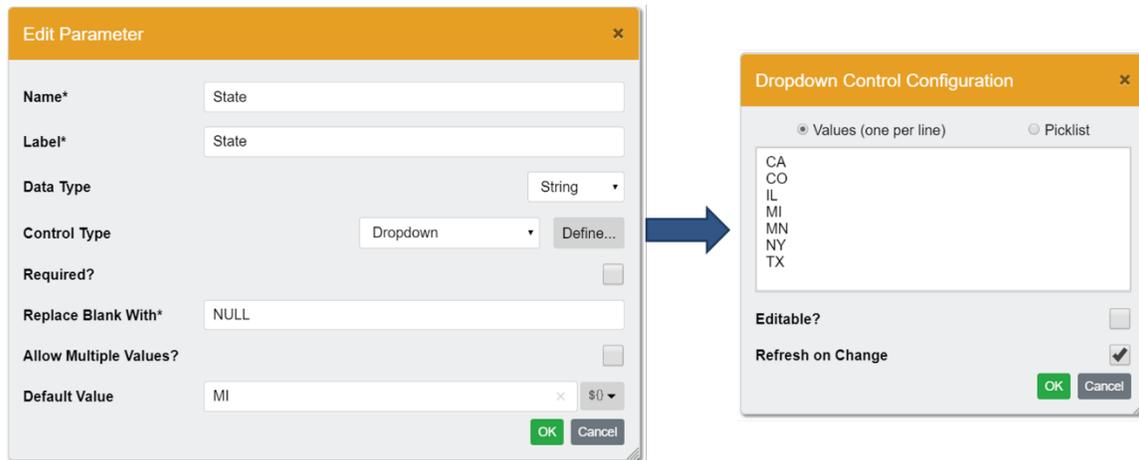


If ‘Allow Multiple Values’ is selected, users will be able to enter multiple values into the Text Box via multiple line items. iDashboards will automatically generate a comma-delimited list from the list of values.

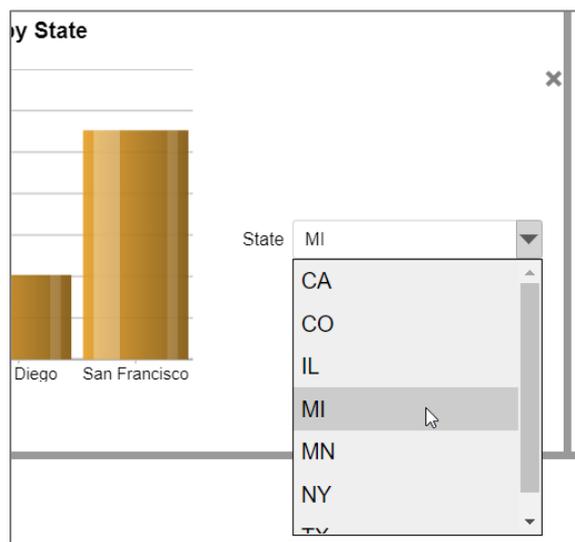
Note: Do not create the comma-delimited list within the Text Box. Each value needs to be entered on a new line item.

13.2.2.2 Dropdown List

Used for data type String, Number or Datetime, a Dropdown List allows a user to select from a predefined list of available values. For horizontal layouts, the width of the dropdown menu will be determined by the longest character set, ranging from a minimum character set of approx. 4 and a maximum of approx. 20.



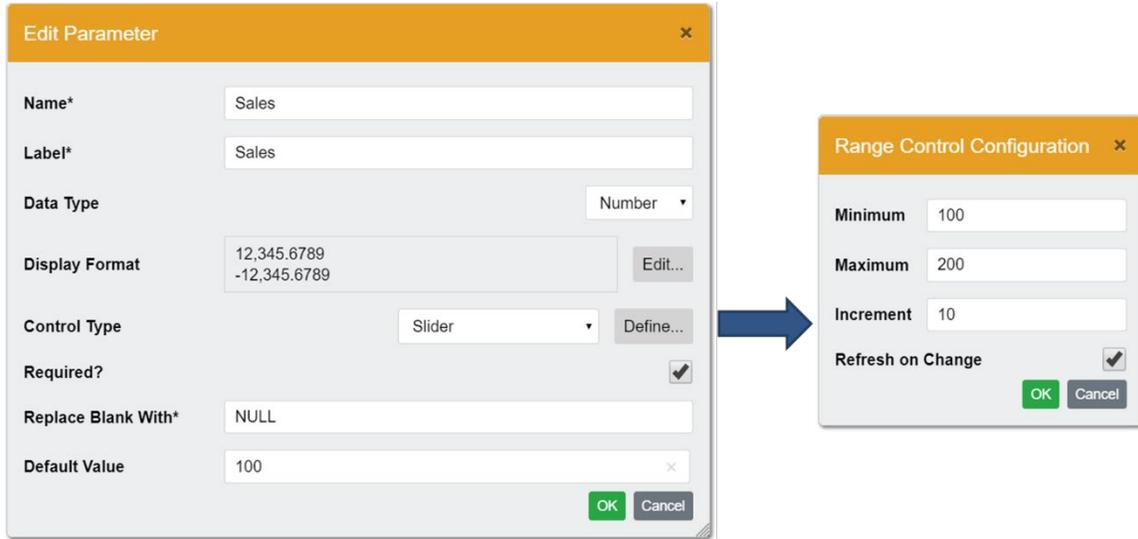
The resulting parameter appears as shown below:



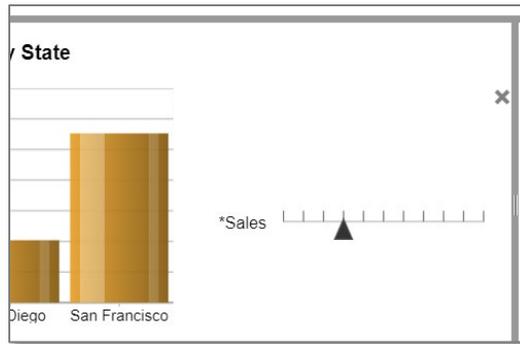
13.2.2.3 Slider (Spinner and Slider/Spinner)

Used for data type Number or Datetime, a Slider allows a user to move a slider through a set of predefined values and select a particular value in between the minimum and maximum defined limits. Shown below are the setup screens for the Spinner (but applicable to the Spinner and Slider/Spinner).

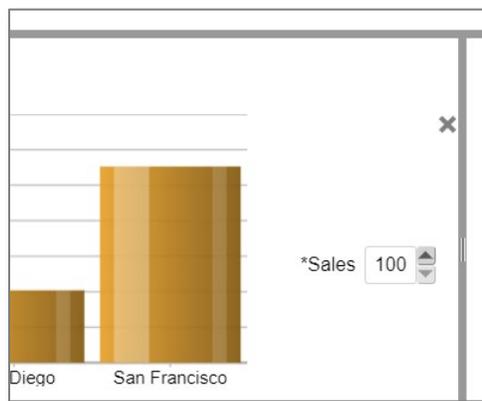
Note: The Slider control and the Spinner control can only increment by the value set in within the configuration. The Slider/Spinner only increments the Slider and not the Spinner.



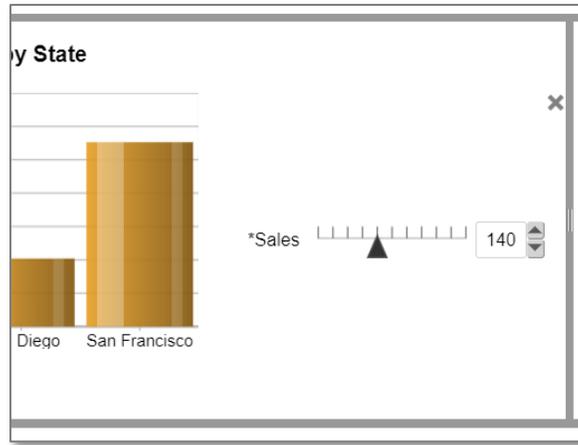
The resulting Slider parameter appears as shown below:



The resulting Spinner parameter appears as shown below:



The resulting Slider/Spinner parameter appears as shown below:



The Parameter Value Definition window, accessed by clicking “Define” on the Parameter Definition window, allows you to define the values that will be available along the Slider bar. For Number, you do this by setting the Minimum, Maximum and Increment values. For Datetime, you do this by setting the Beginning Date, Ending Date and Day Increment. If ‘Refresh on Change’ is selected, the ‘Update’ button will be removed from the chart and the update action will trigger when the user moves and releases the slider.

13.2.3 Import Input Parameters

When there is a need to establish the same input parameter across multiple charts and the dashboard, there is a helpful process requiring a few conditions:

1. The Chart Designer must be used within the Dashboard Designer (not from the Builder home screen).
2. The Dashboard Parameters need to be defined first.

Within the Chart Designer, if the above conditions are met, there is a button that will appear allowing the option to import the dashboard parameters.

Specify Parameters:			
Name	Label	Data Type	Default Value
Sales*	Sales	Number	100



A new window will appear allowing the selection, and importation, of a single, multiple or all dashboard parameters. Multiple selections are accessed using CTRL+click. The import options allow some or all of the parameters to be imported:

Name	Label	Data Type	Default Value
state_name	State	String	MI
sales_amt	Sales	Number	5

13.2.4 Allow Multiple Values

Input parameters can be enabled to allow the use of multiple values. This setting is available to all Data Types, and to the Control Types: Text Box and Dropdown List.

Filtering Data: Enabling the setting occurs in the Definition of the input parameter. If the desired use of the input parameter is to filter data, then it is required to use the 'IN' or 'NOT IN' clause.

JavaScript Expression: Multiple values can also be used within a JavaScript expression. One example refers to the JavaScript array where delimited values are iterated.

Note: A multi-valued parameter can only provide a value for another multi-valued parameter. However, a single-valued parameter can provide the value for a multi-valued parameter.

13.2.4.1 Setting the Initial Value

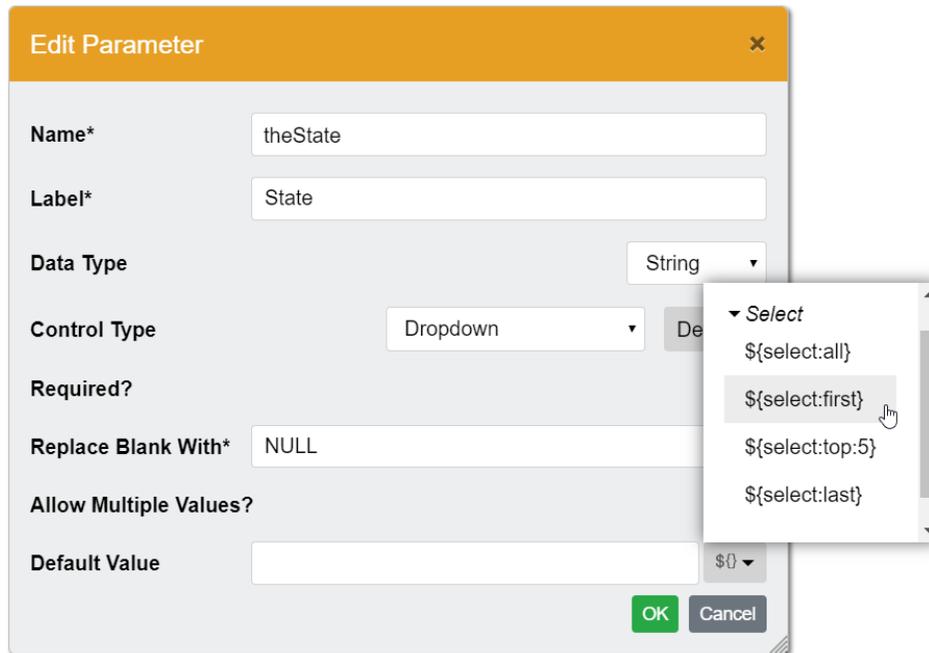
When charts or dashboards use input parameters, there is often an option to define the static initial value to be used upon loading the dashboard. If the desire is to have a single value as the initial value, then simply enter the static value into the 'Initial Value' field. If the desire is to apply multiple values as the initial value, then enter each value on a line item as shown below.

Note: When manually defining multiple values as an initial value, do not create the comma-delimited list within the 'Initial Value' field. Each value needs to be entered on a new line item.

When using Dropdown lists (as the Control Type), then it is possible to use a macro to achieve certain initial values. Macros can be typed in or click on '\${}' to see the list.



- **`#{select:all}`** – Sets the initial value to all available values.
 - This is selected by default when 'Allow Multiple Values' is enabled.
- **`#{select:first}`** – Sets the initial value to the first row in the list
- **`#{select:last}`** – Sets the initial value to the last row in the list
- **`#{select:top:5}`** – Sets the initial values to a defined number of rows from the top of the list. Number of rows are defined by changing the number, ex: `#{select:top:<number of rows>}`.
 - This is selected by default when 'Allow Multiple Values' is enabled.



The screenshot shows the 'Edit Parameter' dialog box with the following fields and values:

- Name*: theState
- Label*: State
- Data Type: String
- Control Type: Dropdown
- Required?: (checkbox)
- Replace Blank With*: NULL
- Allow Multiple Values?: (checkbox)
- Default Value: (dropdown menu)

The 'Default Value' dropdown menu is open, showing the following options:

- Select
- `#{select:all}`
- `#{select:first}` (highlighted)
- `#{select:top:5}`
- `#{select:last}`

Buttons: OK, Cancel

13.3 Working with Input Parameter Values

When a user enters a value for an Input Parameter, that value is stored by iDashboards and can be used in a number of ways. To use an Input Parameter value, you will make use of the “param” macro.

13.3.1 The PARAM Macro

The Input Parameter macro takes the following format:

```
${param:<Parameter Name>}
```

For example, if you want to use the value of the Input Parameter named “theState”, the macro would look like:

```
${param:theState}
```

Any legitimate place you use this macro, the macro will be replaced by the current value of the “theState” Input Parameter.

Note: If a parameter macro uses the option ‘Allow Multiple Values’ and the same parameter macro is being used to filter data, the clause must use ‘IN’ to ensure comma-delimited values are acceptable.

Note: If a parameter macro uses the option ‘Allow Multiple Values’ and the same parameter macro is being used within the chart title, then a comma-delimited list of expanded values will appear in the title. The values will be separated with commas and no-spaces.

13.4 Date Selection Control

Input parameters that use the datetime control type can leverage the date and time selection windows to define initial values. The date icon allows the user to select a short date (yyyy-MM-dd). The time selection component allows the user to additionally select a time portion for their datetime value. Date macros are also supported as an initial value.

Note: "Datetime" requires all dates to be submitted with the time stamp.

Note: The Administrator Application specifies whether or not military time format is displayed. The default value for this setting is 'true', indicating the time selection will display the time in military format.



13.5 Filtering Data with Input Parameters

A primary use of Input Parameters is to filter data displayed in charts. Note that we did not say “charts and dashboards.” This is because there is no data to filter within a dashboard, only the charts that make up the dashboard have data to filter. Therefore, this section pertains specifically to Chart Input Parameters. Recall that Dashboard Input Parameters are passed down to Chart Input Parameters.

The filtering process works as follows. Every time a chart receives an Input Parameter value (either via user input, drilldown or an initial value), the chart refreshes, filtering its data based on the value. Every time this happens, the chart performs a new query against its data source, using the parameter value as a filter. The data returned to the chart is a result of the filtered query.

To set up this filter functionality, you need to add the Input Parameter as a filter on the chart. You do this by creating a chart filter and using the PARAM macro as its value.

For example, let’s say you want to create a “Sales by City” chart that will accept ‘State’ as an Input Parameter and filter the chart data based on the state abbreviation entered.

1. Create the chart

2. Create the Input Parameter

3. Create the filter

Specify Filter Criteria ✕

Use rows from CityStateSales where...

State =

`$(param:theState)`

Macros

4. Enjoy the results!

Sales by City, by State

City	Sales (Approx.)
Ann Arbor	2,200,000
Detroit	2,200,000
Grand Rapids	3,000,000
Lansing	1,100,000
Midland	1,300,000
Traverse City	2,600,000

State ✕

Once the chart is configured as described, the chart will take the state value “MI” and pass it to the chart filter through the PARAM macro. The value will then be used as a filter criterion when the chart queries its data source, therefore only displaying data filtered for the Input Parameter value.

Since Input Parameters and Chart Filters are two distinct features of iDashboards, there is no way to validate that an Input Parameter value entered by a user will be a valid filter for that chart. Therefore, it may be possible for a user to submit an Input Parameter value that will result in a chart with no data, as shown below.

Sales by City, by State

No data was found for this chart.

State ✕

Note: The “No data was found for this chart.” default message can be customized by clicking ‘Edit...’ under ‘No Chart Data Message’ within the Basic Settings of the Chart Designer.



It is very important to make sure that when a chart is first loaded, it has a parameter value for every Input Parameter it uses for filtering. Otherwise, the chart may return no data. You can force an Input Parameter value in any of the following ways:

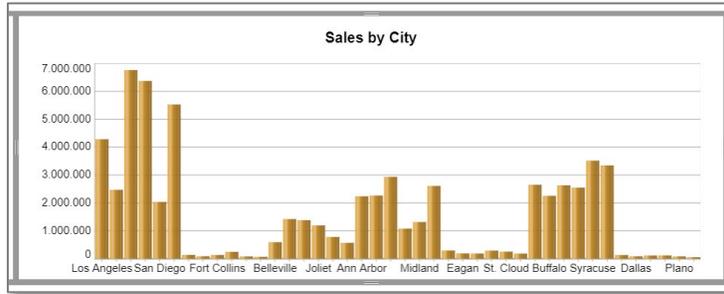
1. Create an initial value for the Input Parameter.
2. Check the Required Flag for the Input Parameter.
3. Check the “Always prompt for parameters on dashboard load:” option on the Input Parameters tab.



13.6 Displaying Input Parameters in Chart Titles

Input Parameter values can be displayed in a chart title by adding the PARAM macro to the chart title. For example, let's say you want to create a "Sales by City" chart that will display the parameter value within the title like "Sales by City (MI)".

1. Create the chart



2. Create the Input Parameter

New Parameter

Name*

Label*

Data Type

Control Type Define...

Required?

Replace Blank With*

Allow Multiple Values?

Default Value

OK Cancel

3. Create the filter

Specify Filter Criteria

Use rows from CityStateSales where...

=

Macros Values

OK Cancel

4. Define the Chart Title

Chart Title

5. Enjoy the results!

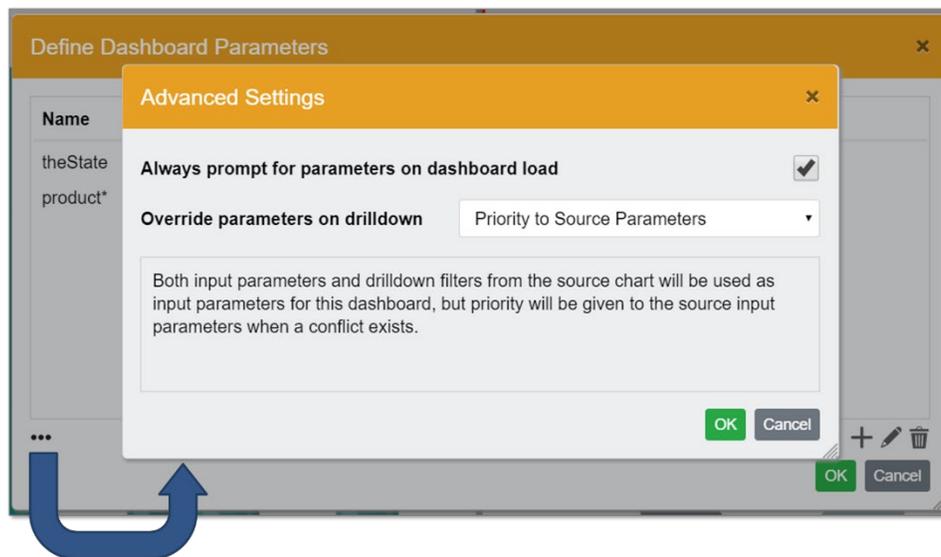
City	Sales
Ann Arbor	2,200,000
Detroit	2,200,000
Grand Rapids	3,000,000
Lansing	1,000,000
Midland	1,300,000
Traverse City	2,500,000

13.7 Prompting the User for Input Parameter Values

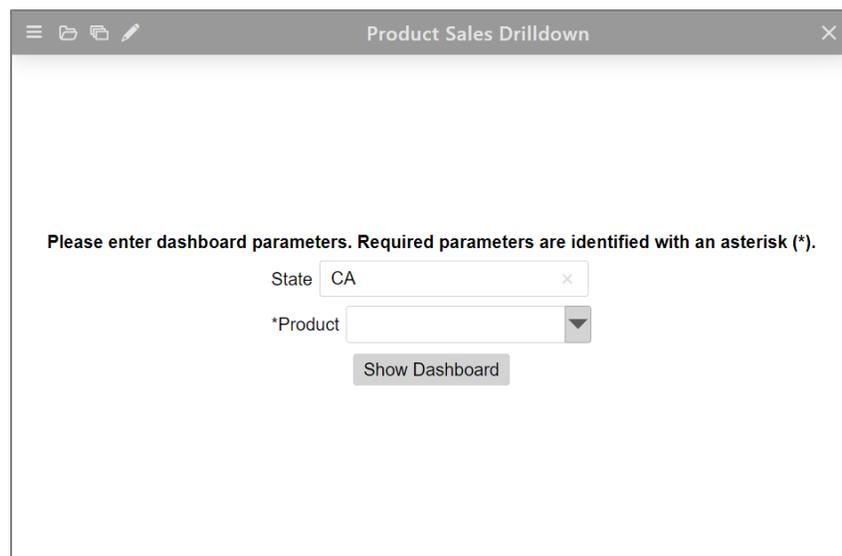
Input Parameter values are entered through Input Parameter “prompts.” Prompts work the same for dashboards and charts, but the circumstances in which they are presented to the user differ.

13.7.1 Dashboard Input Prompts

When a dashboard is first opened, the user may or may not be presented with a pop-up prompt to enter Input Parameter values. The appearance of a prompt depends on the situation (as related to the input parameters). If a prompt is necessary, it will look similar to the image below. Required parameters are identified with an asterisk (*). To enable the prompt, go to the Dashboard Designer and configure additional properties via the “...” button and “Define Dashboard Parameters”. Then, via the “...” button, again, enable “Always prompt for parameters on dashboard load.



The result of doing this can be experienced in the View Interface when opening the dashboard:

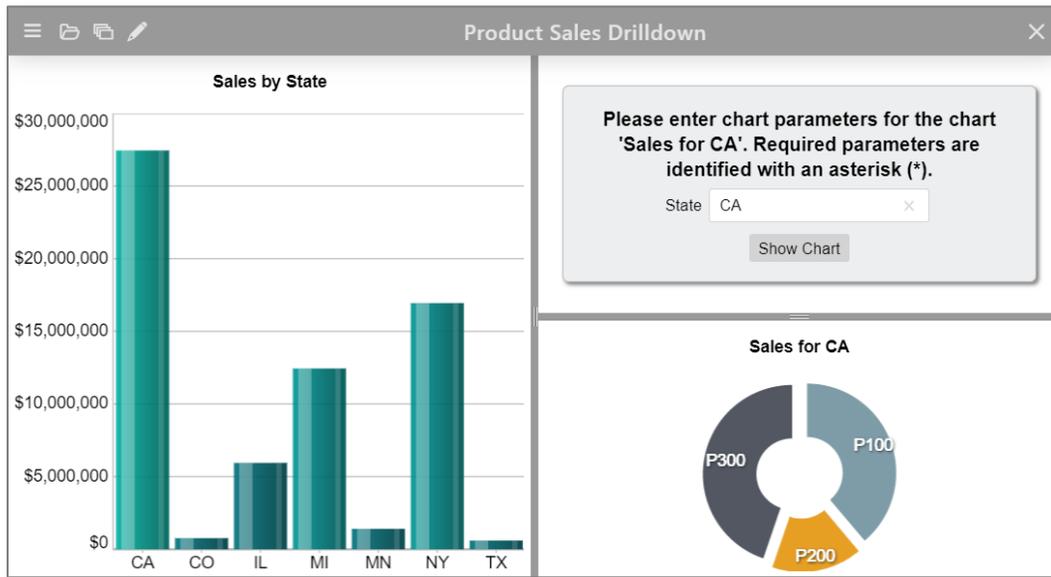


Note: The user may also open the prompt themselves. This is done by navigating to the menu and selecting “Enter Dashboard Parameters...”

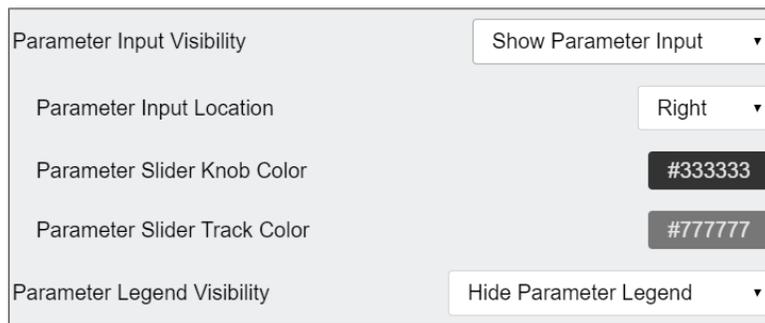
The Dashboard Input Parameter prompt may be permanently added to a dashboard so that it is visible at all times. This is done by adding a “Dashboards Parameter” Panel to one of the frames of the dashboard.

13.7.2 Chart Input Prompts

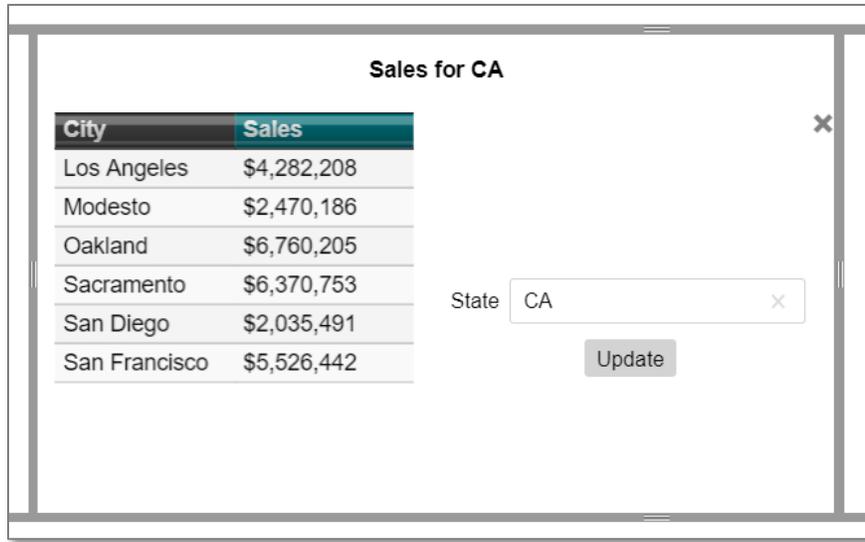
When a chart is first opened, the user may or may not be presented with a pop-up prompt to enter Input Parameter values. The appearance of a prompt depends on the situation (as related to the input parameters). If a prompt is necessary, it will look similar to the image below. Required parameters are identified with an asterisk (*). Required parameters are identified with an asterisk (*).



By default, Chart Input Parameter prompts are visible as part of the chart, as shown below. You can control how prompts are displayed via the Parameters settings in the chart’s properties.



You can control whether the Input Parameter prompt is shown, hidden or minimized. You can also control the location of the prompt; left, right, top, bottom or floating. If you select floating, the prompt will act as a floating window that you can move around the dashboard and minimize.



City	Sales
Los Angeles	\$4,282,208
Modesto	\$2,470,186
Oakland	\$6,760,205
Sacramento	\$6,370,753
San Diego	\$2,035,491
San Francisco	\$5,526,442

State: CA

The Parameter Legend displays the Input Parameter values the chart is currently using. You can control the visibility of the legend as well as its text color, background color and transparency.

13.8 Rules for Prompting

A user will be prompted to enter Input Parameter values when a dashboard and its charts are first loaded and one or more of the following conditions exist. This holds true for Dashboard Input Parameters and Chart Input Parameters.

1. An Input Parameter is marked as required, does not have an initial value and does not receive a value via a Drilldown Filter or Source Parameter.
2. The option “Always prompt for parameters on chart load” has been checked within the Input Parameter configuration.

13.9 Blank Parameter Values

A blank parameter value refers to a parameter value that has been provided, but is “blank”. For example, a user is prompted for an Input Parameter value, but they leave the value blank and submit anyway. Therefore, a blank value is submitted and recognized by iDashboards.

iDashboards will then replace that blank value with the value in the Input Parameters ‘Replace blank with’ field. If the ‘Replace blank with’ field is also blank (which it could be because it’s optional), then the Input Parameter value will ultimately remain an empty string or null numeric or date value, depending on the parameter data type.

When a blank Input Parameter value is replaced with the ‘Replace blank with’ value, it is actually the PARAM macro that takes on the value of the ‘Replace blank with’ field. There are two situations where this can take place. (1) When the PARAM macro is used in a filter for a chart built using Data Source Columns, and (2) when the PARAM macro is used for a chart built using a Custom Query. PARAM macros will not be replaced with the ‘Replace blank with’ value when used in a chart title or stored procedure input.

When the PARAM macro is replaced with the ‘Replace blank with’ value, it is done so verbatim. Therefore, if the ‘Replace blank with’ value is a string, you need to make sure you include single quotes in the Input Parameters ‘Replace blank with’ field so that the SQL query used by the chart runs properly.

Since a chart’s filter value is really part of the SQL WHERE clause, the string-based filter value needs to be singled quoted. If the ‘Replace blank with’ value is a number or datetime, no quotes are needed. When the PARAM macro is used in situations other than expanding to the ‘Replace blank with’ value, iDashboards takes care of adding the single quotes (if needed).

13.10 Passing Input Parameters into Custom Queries

Input Parameters can be used in Custom Queries by adding the PARAM macro to the query.

For example, if you create a chart with Input Parameters, you could create a Custom Query and Filter as shown below in the comparison table below.

Note: An Input Parameter’s ‘Replace blank with’ value will not replace the PARAM macro when used in a chart title.

Original Query	Query with Input Parameter
<pre>SELECT City, Sales FROM dbo.dw_ETL_SALES WHERE State = 'MI'</pre>	<pre>SELECT City, Sales FROM dbo.dw_ETL_SALES WHERE State = \${param:theState}</pre>



13.11 Passing Input Parameters into Stored Procedures

Input Parameter values can be used as stored procedure values when a chart uses a stored procedure as its data source. You do this by adding the PARAM macro as the stored procedure value when setting up the stored procedure arguments.

Note: Since an Input Parameter's 'Replace blank with' value cannot be used as a stored procedure input, this field is absent from the Parameter Definition window when creating Input Parameters for charts using a stored procedure as a data source.

When designing a stored procedure chart with input parameters, use the parameter macro in the Arguments field.

Specify Parameters:

Name	Label	Data Type	Default Value
theState	State	String	MI

Arguments Filters

Name	Data Type	Value
stateName	String	<input type="text" value="\${param:theState}"/>

... + ✎ 🗑

13.12 Passing Input Parameters through Drilldowns

Input Parameters can be passed through a drilldown just as standard Drilldown Filters can. When the user clicks on a drilldown chart (called the “**source chart**”), the value they clicked on can be passed as a parameter during the drilldown so the “**target chart**” (or “target dashboard”, “target web page”, etc.) can consume the value. In the scenario where a source chart drills down to a dashboard, all charts within that dashboard act as target charts.

13.12.1 Sending Input Parameters through a Drilldown

The default behavior of a source chart is to pass its Input Parameters when a drilldown is initiated (when it is clicked on). This can be disabled by unchecking the “Send Input Parameters to Target Chart” checkbox on the drilldown properties.

Note: Similar to standard Drilldown Filters, Input Parameters accumulate and are stored by iDashboards over multiple drilldown levels.

13.12.2 Getting Input Parameters through a Drilldown

The target chart of a drilldown can accept any combination of the Drilldown Filters and Input Parameters from a source chart. This is controlled via the “Override parameters on drilldown:” dropdown box when accessing the parameter advanced settings.

Name	Label	Data Type	Default Value
product	Product	String	P100

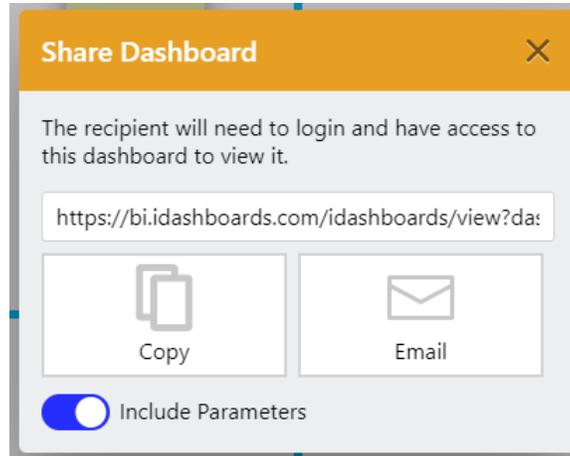
The options are as follows:

- **Priority to Source Parameters** - Both Input Parameters and Drilldown Filters from the source chart will be used as Input Parameters for this chart, but priority will be given to the source Input Parameters if a conflict exists.
- **Priority to Drilldown Filters** - Both Input Parameters and Drilldown Filters from the source chart will be used as Input Parameters for this chart, but priority will be given to the Drilldown Filters if a conflict exists.
- **Only Source Parameters** - Only Input Parameters from the source chart will be used as Input Parameters for this chart and Drilldown Filters from the source chart will be ignored.
- **Only Drilldown Filters** - Only Drilldown Filters from the source chart will be used as Input Parameters for this chart and Input Parameters from the source chart will be ignored.
- **Neither Parameters nor Drilldown Filters** - Neither the Drilldown Filters or Input Parameters from the source chart will be used as Input Parameters for this chart.



13.13 Passing Input Parameters through URL

The ability to share a dashboard is a feature within the viewer interface (as shown below). One of the options, when sharing, is the ability to include the parameters. If parameters are included, the URL will contain the last used parameter values.



13.13.1 URL Syntax

The easiest method for constructing a URL with parameter values is described above using the “Share Dashboard” method. However, it’s possible to manually construct the URL using the following syntax:

For dashboard parameters:

```
p_<paramName>=<value>
```

For chart parameters

```
p_<chartId>_<paramName>
```

- If a parameter is multi-valued, then multiple occurrences of it can occur in the URL.
- If the same chart occupies more than one frame, then all charts will receive the parameters from the URL.
- Blank or empty parameters will not appear in the URL.
- If a frame contains a drilldown, the chart parameters will not appear in the URL.

Once the URL is generated, it can be shared with another iDashboards user (if access permits). Alternatively, the URL can be manually manipulated for other purposes. The examples below will all open a single dashboard ‘6064’ using the viewer interface:

Example 1 – This is a regular URL for sharing a dashboard

```
https://bi.idashboards.com/idashboards/view?dashId=6064
```

Example 2 – This URL will open the dashboard and populate all parameter instances named ‘city’ using the value ‘Detroit’

```
https://bi.idashboards.com/idashboards/view?dashId=6064&p_city=Detroit
```



Example 3 - This URL will open the dashboard and populate all parameter instances named 'city' using the value 'Detroit' and populate all parameter instances named 'startdate' using the value '2016-01-01'

```
https://bi.idashboards.com/idashboards/view?dashId=6064&p_city=Detroit&p_startdate=2016-01-01%2000%3A00%3A00.000
```

Example 4 – This URL will open the dashboard and populate the chart (Chart ID: 1234) parameter named 'city' using the value 'Detroit'. This URL did not populate the dashboard parameter also named 'city'.

```
https://bi.idashboards.com/idashboards/view?dashId=6064&p_1234_city=Detroit
```

Example 5 – This URL will open the dashboard and populate all parameter instances named 'city' using the values 'Detroit' and 'Fort Wayne'. Note: iDashboards will handle words separated by spaces.

```
https://bi.idashboards.com/idashboards/view?dashId=6064&p_city=Detroit&p_city=Fort Wayne
```

```
https://bi.idashboards.com/idashboards/view?dashId=6064&p_sdate=2016-01-01%2000%3A00%3A00.000&p_edate=2019-12-26%2000%3A00%3A00.000&p_city=asdf&p_8824_sdate=2016-01-01%2000%3A00%3A00.000&p_8824_edate=2019-12-26%2000%3A00%3A00.000&p_9747_city=asdf
```



13.14 Picklist

Picklists are similar to charts in terms of creation and management. Knowing how charts are designed within iDashboards will help understand the workflow for designing a picklist.

13.14.1 Picklist Overview

A picklist is a defined list that can be used to populate dropdown parameters. Picklists are managed independently from parameter creation; therefore, the same picklist may be used multiple times across several charts and dashboards. The values in the list can be either static or dynamic.

13.14.2 Design Permissions

The ability to access the Lobby and then navigate to the Build Interface is allowed with a minimum user role of “Builder”. But the ability to save items into customized categories requires access rights provided by the iDashboards administrator. Such access rights are provided at the Category level, and all dashboards, charts, picklists and forms (if enabled) within any given Category inherit those access rights.

Three access rights:

- Save Access
- View Access
- No Access

Therefore, depending upon the access rights assigned to you, you may or may not have the rights to make changes and save those changes to an existing dashboard. However, you can always make changes to an existing picklist and save those changes as a different picklist within any other Category to which you may have save permissions.

Note: The remainder of this manual assumes a proper user role and access rights are configured

13.14.3 Create or Edit a Picklist

A picklist can be created or edited directly from the Home screen in a manner similar to that of creating or editing a chart.

Note: A second method for creating a picklist occurs during the creation of an input parameter.

To create a new picklist, make sure the “Picklist” tab is selected on the Home screen, then select the “New” button as seen in the image below.

To edit an existing picklist, make sure the “Picklist” tab is selected on the Home screen. Then, using manual navigation or searching, locate the picklist to open.





13.14.4 Picklist Designer

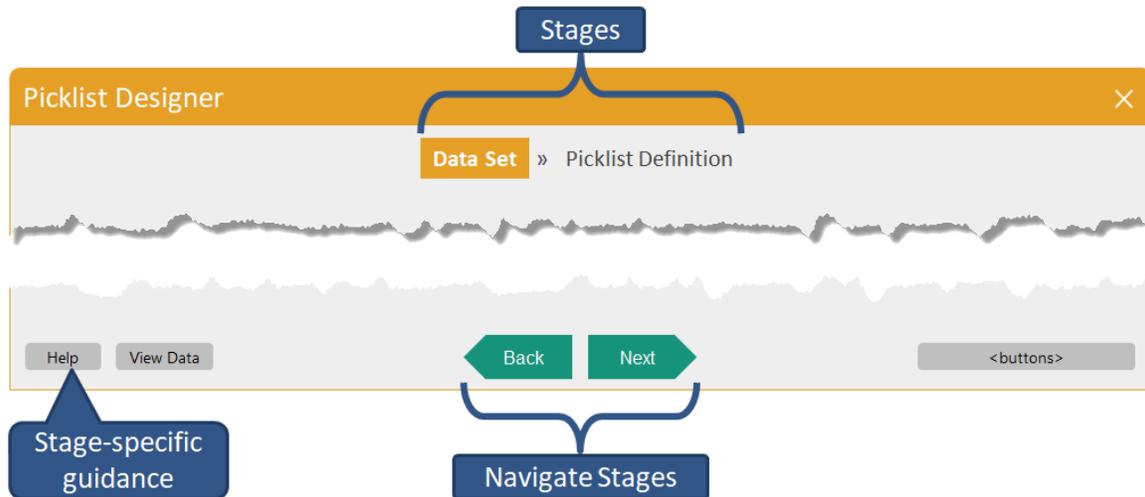
The Picklist Designer is a window used for creating and editing all aspects of a picklist. The creation of a picklist is very similar to the creation of a chart. The designer is comprised of various steps and stages, with minimum requirements, prior to the ability of saving picklist. Only one Picklist Designer can be opened at a time.

As with chart data, picklist data can be obtained from a dynamic data source or may be statically defined. Dynamic picklist data can be obtained from a database table or view, a stored procedure, a data feed, a custom query or from a named range using Excel. Static picklist data can be manually defined or imported from an Excel workbook or a CSV file.

The primary difference between picklist data and chart data is that a picklist has exactly two picklist column definitions (rather than axes like a chart) associated with it; one for the actual value of a picklist item and one for the display value. Another difference from defining chart data is that expressions are not available to define picklist column values.

13.14.4.1 Navigating the Picklist Designer Stages

The Picklist Designer progresses in a left-to-right direction as the picklist is being built or edited. The stages appear at the top of the window, where the current stage will be highlighted and previous stages will appear as 'visited'. To transition between the stages, click on the buttons along the bottom of the window (as shown below). Picklist design only consists of two stages; specifying the data set for the picklist and then completing the picklist definition.



- **Data Set** – This stage includes defining where the data will come from, as well as the picklist parameters and filters.
- **Picklist Definition** – This stage includes defining which data columns from the data set will be used to specify the value column and the display column for the picklist. Formatting of the display value and properties of the picklist are also specified in this stage.

13.14.4.2 Picklist Designer Help

Click on the “Help” button to receive in-product, stage-specific documentation. While this information may not provide all of the details of the product, it can be helpful to receive guidance to encourage the fundamentals of creating a picklist.

13.14.5 Saving the Picklist (or Save As)

The picklist is edited and created locally within the application, until the user chooses to save the picklist to the server. Saving a picklist has requirements and validations which must be met before the “Save” button will become enabled. Progressing to the later stages will enable the “Save” button.

Saving a new picklist has the following minimum requirements:

- A selected Category (in which you have ‘save’ permissions).
- A picklist “Name”.
 - In the event the picklist name is not unique within the category, you will be asked to provide a new name or overwrite the existing picklist.

To save a picklist:

- Start a new picklist or open an existing picklist.
- Progress to the “Picklist Definition” stage, then click on the “Save” button.
- In the save dialog window, select a category then enter a picklist “Name”.



- Click “Save”.

Note: “Save As...” is an option used to create a copy, or clone, of the current picklist.



13.15 Designing a Picklist

The first stage of designing a picklist requires configuring the source, and technique, for how the picklist will be populated with data. A new picklist can be created by providing data using the two basic states for data:

1. Dynamic Data

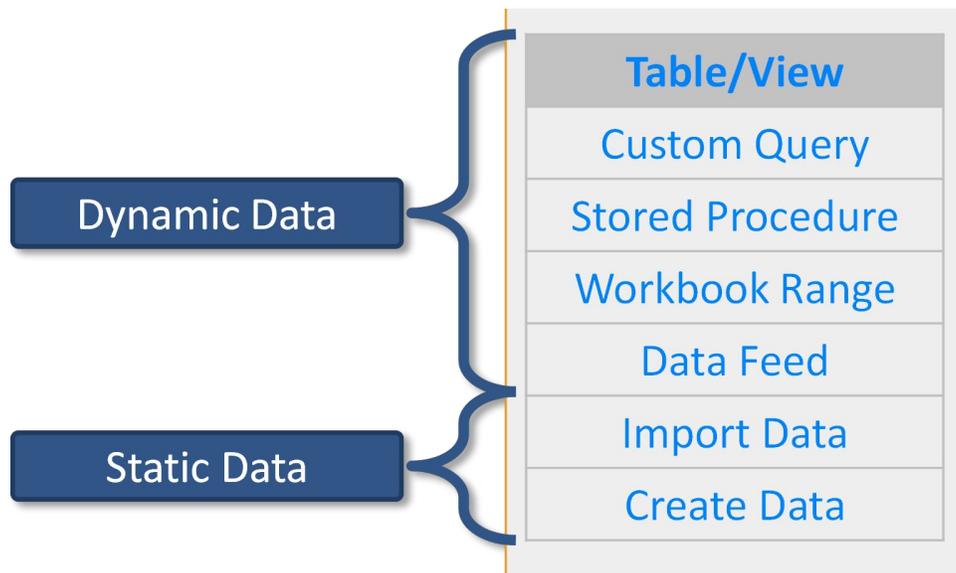
- a. Query data from your local database
- b. Query data from Excel files (uploaded via Workbook Database)

2. Static Data

- a. Entering the data manually
- b. Importing the data from an Excel spreadsheet (this is a one-time import)
- c. Importing the data from a (CSV) comma separated values file (this is a one-time import)

The image below identifies the first decision to make when creating a new picklist.

Note: Creating picklists by dynamically retrieving data requires knowledge of Data Structure, Named Ranges, Tables, Views, Queries and Data Columns.



13.15.1 Dynamic Data

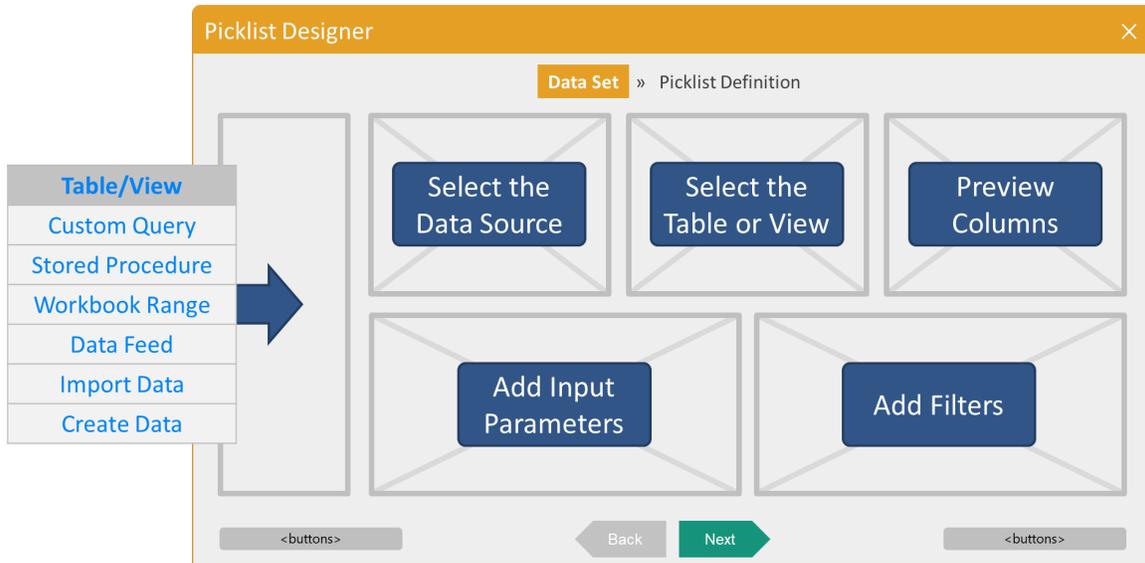
This section describes the dynamic data methods for creating a picklist.

13.15.1.1 Filtering Picklist Data

Similar to a chart filter, the values used within a picklist can be filtered to include specific values or exclude specific value. An unlimited number of filters to be applied to the picklist result set.

13.15.1.2 Table/View

When first creating a new picklist, selecting the “Table/View” data option will allow a picklist to dynamically connect to existing Tables or Views. The image below shows the framework for this stage. The minimum requirements include selecting the Data Source and Selecting the Table or View.



13.15.1.2.1 Define Data Set

The Data Source section of the window lists all the available data sources for which you have permission to use. Data Sources are configured in the iDashboards Administrator Application by an iDashboards administrator. If “Data Source Access Control” is enabled by the administrator then it is possible there are data sources hidden from view for particular users.

- **Select a Data Source** – Click on an item in the Data Source list, and its list of Table and Views will appear.
- **Select a Table or View** – Here, a list of database schemas within that data source will appear as folders. These schemas/folders can be expanded or collapsed to display the tables or views. Click on an item in this list, and the remaining fields will appear.
- **Data Column Preview** – Column names and the associated column data type are displayed in this section of the window.
- **Specify Parameters** – See Chapter 13, “Input Parameters” for details.
- **Specify Data Filters** – See Section 10.5, “Using Filters” for details.

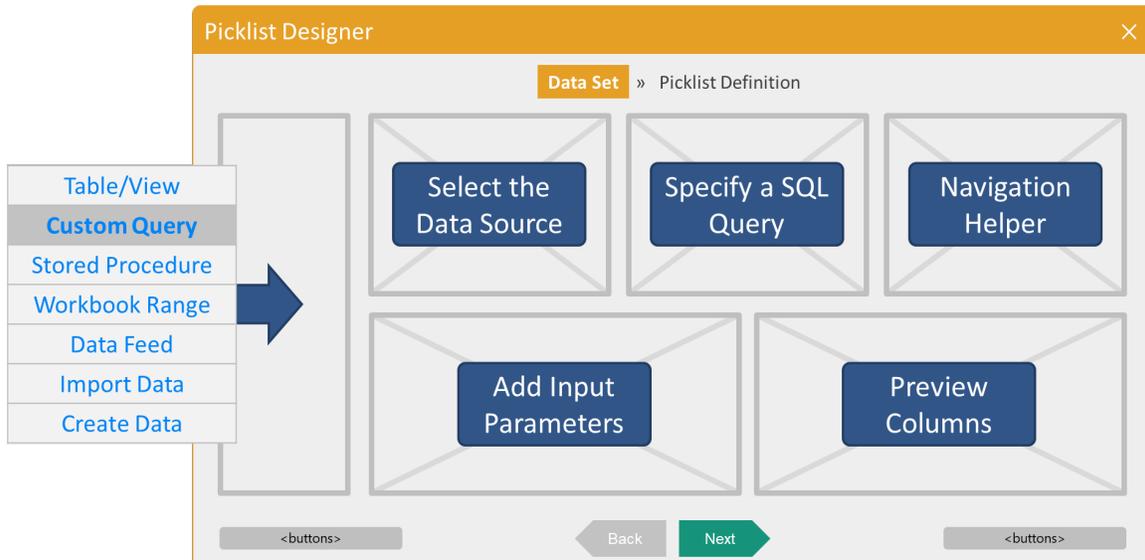
Click “Next” to continue to the next stage called ‘Picklist Definition’.

13.15.1.3 Custom Query

This section describes how to use a “Custom Query” to retrieve picklist data. Using this option requires a certain knowledge to understand the required syntax and expectations for using this powerful option. The image below shows the framework for this stage. The minimum requirements for the query demands returning a single column.



Note: Queries defined here, within a picklist, cannot use the SQL JOIN clause. It is only possible to select a single database table or view. Filtering must take place within the Query.



13.15.1.3.1 Define Data Set

The Data Source section of the window lists all the available data sources for which you have permission to use. Data Sources are configured in the iDashboards Administrator Application by an iDashboards administrator. If “Data Source Access Control” is enabled by the administrator then it is possible there are data sources hidden from view for particular users.

- **Select a Data Source** – Click on an item in the Data Source list, and the SQL Query editor will appear.
- **Specify a SQL Query** – The first word must be “SELECT”, followed by the remaining query. Some options include selecting data from a single table or joining tables.
- **Navigation Helper** – This window allows you to see a list of tables and views, along with their associated columns, within the selected data source. Drag the Table name or Column name into the SQL dialog box. You may use the list to build your SQL statement or write it manually.
- **Retrieve Columns** – Clicking this “RED” button will process the current query. If there are errors, a window will appear stating the general problem. If there are no errors, the button will become “GREEN” and the validated columns from the query will appear in the bottom-right portion of the Picklist Designer. The “Data Type” column will be automatically chosen by iDashboards to represent one of three data types: String, Number or Datetime. iDashboards will examine the data type of the column that is selected under the “Data Column” and apply the most appropriate data type.
- **Specify Parameters** – See Chapter 13, “Input Parameters” for details.
- **Specify Data Filters** – See Section 10.5, “Using Filters” for details.

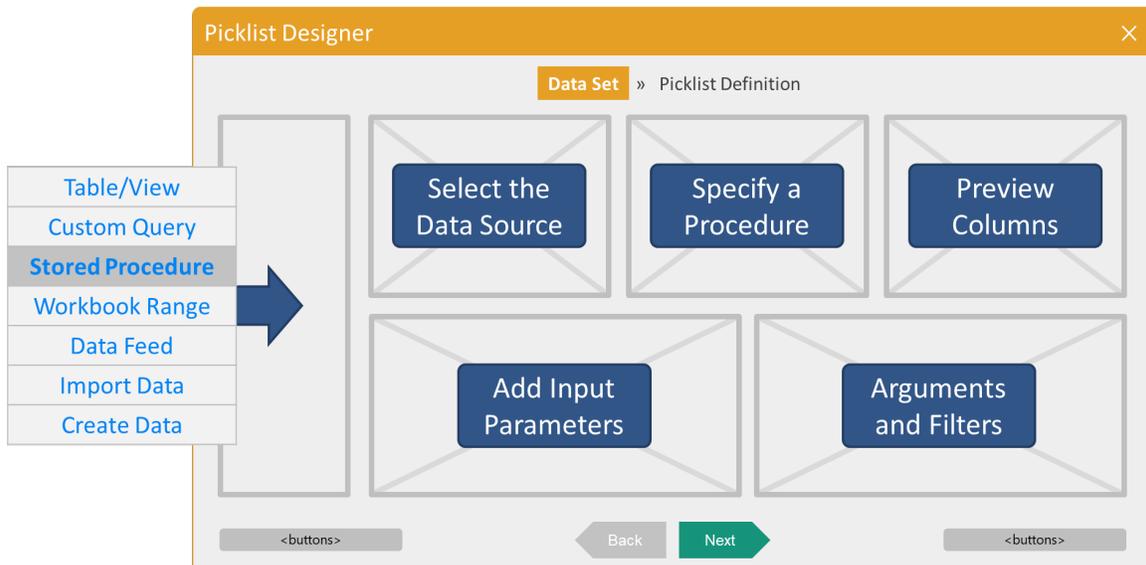


Note: When creating Custom SQL queries, the use of fully qualified table names is encouraged. If wildcards are used to collect columns, changes in the table may result in a picklist error. Picklists with an error will require re-mapping between the Picklist Columns and the Data Columns.

Click “Next” to continue to the next stage called ‘Picklist Definition’.

13.15.1.4 Stored Procedure

This section describes how to use a “Stored Procedure” to retrieve picklist data. The Administrator’s Manual outlines the process of adding a stored procedure to iDashboards. Once that is completed, users can build a picklist that utilizes a stored procedure.



13.15.1.4.1 Using a Macro as an Argument

A variety of macros exist which can be used as input arguments for a stored procedure; $\${user}$, $\${param:<Parameter Name>}$, or any of the date macros (or their derivations) listed below. It is essential that the stored procedure was written to accept the macro data type offered as an input argument.

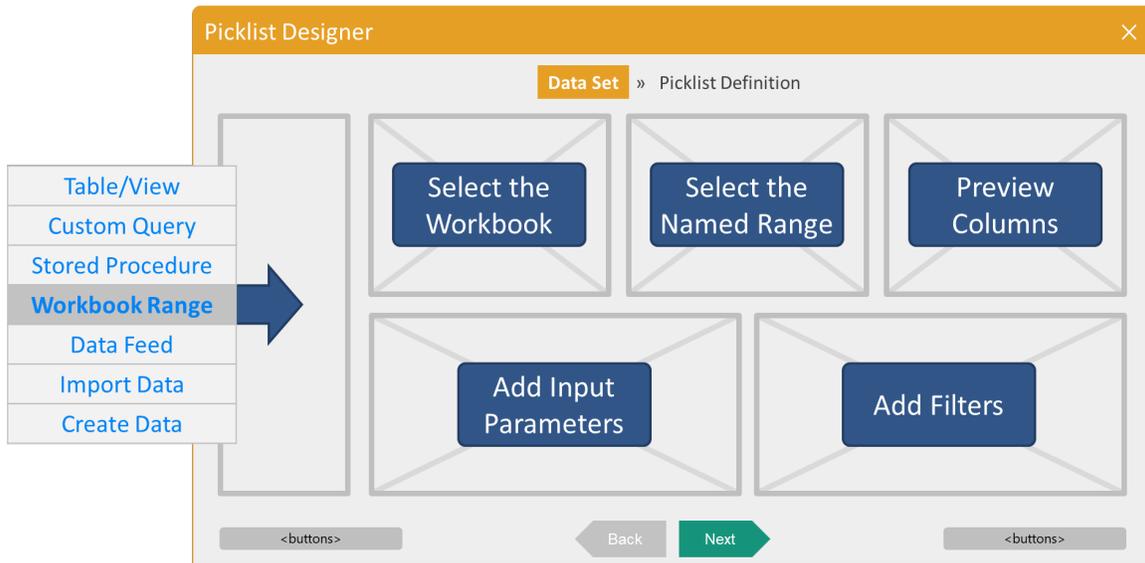
Date Macros		
$\${ds}$	$\${currmonth, "MMM"}$	$\${curryear}$
$\${ts}$	$\${currmonth, "MMMM"}$	$\${curryear-3}$
$\${currdate}$	$\${currmonth + 2, "MMMM"}$	$\${curryear + 2}$
$\${currdate-30}$	$\${currmonth + 2, "MMMM d, yyyy"}$	$\${curryear + 2, "dd MMM yy"}$
$\${currdate+30}$	$\${currmonth:begin}$	$\${curryear:begin}$
$\${currdate+90, "MMMM dd, yyyy"}$	$\${currmonth:end}$	$\${curryear:end}$
$\${currmonth}$	$\${currmonth:end - 2}$	$\${curryear:begin + 1, "dd MMM yy"}$
$\${currmonth-3}$	$\${currmonth:begin + 3, "MMMM d, yyyy"}$	
$\${currmonth+5}$	$\${currmonth:begin + 3, "MMMM dd, yyyy"}$	
$\${currmonth, "MM"}$		

The $\${user}$ macro simply returns the username of the currently logged in user. Therefore, when a picklist that uses this macro as an input argument is opened, the username of the current user is sent to the stored procedure as an input argument.

The $\${param:<Parameter Name>}$ macro refers to the value of the named picklist parameter.

13.15.1.5 Workbook Range

This option is visible only when the iDashboards administrator has configured an Excel workbook database (Refer to the Administrator Manual for details). If the Workbook Range option is selected, a list of Excel files will appear. These files have been uploaded into the workbook database prior to creating the picklist. When a file is selected, the named ranges within the file will then be displayed.



13.15.1.5.1 Define Data Set

From this window, access to the Workbook Range is allowed. Alternatively, access to the Auto Uploader (for scheduling Excel file uploads) is controlled by a system setting within the Administrator Application. See the Administrator's Manual for details.

Note: Users will have access to all uploaded Excel files and the associated Named Ranges - "Data Source Access Control" does not affect this screen.

- **Select a Workbook** – Select a file name in the Workbook list, and its list of Named Ranges will appear.
- **Select a Named Range** – Select the name of a Named Range, and the remaining fields will appear.
- **Data Column Preview** – Column names and the associated column data type are displayed in this section of the window.
- **Specify Parameters** – See Chapter 13, "Input Parameters" for details.
- **Specify Data Filters** – See Section 10.5, "Using Filters" for details.

Click "Next" to continue to the next stage called 'Picklist Definition'.

13.15.1.6 Data Feed

A relational database contains one or more data tables. Each table has a name, and a specific set of columns. Each column has a name and a specific data type.



An iDashboards data service provides something similar to a database table called a data feed. Like a table, a data feed has a name to distinguish it from other data feeds within the same data service. Also like a table, a data feed has a specific set of data columns, each with a unique name (within the feed) and data type.

Note: Setting up Data Feeds is an administrative task.

13.15.1.7 Filtering Picklist Data

Similar to a chart filter, the values used within a picklist can be filtered to include specific values or exclude specific values. An unlimited number of filters to be applied to the picklist result set.

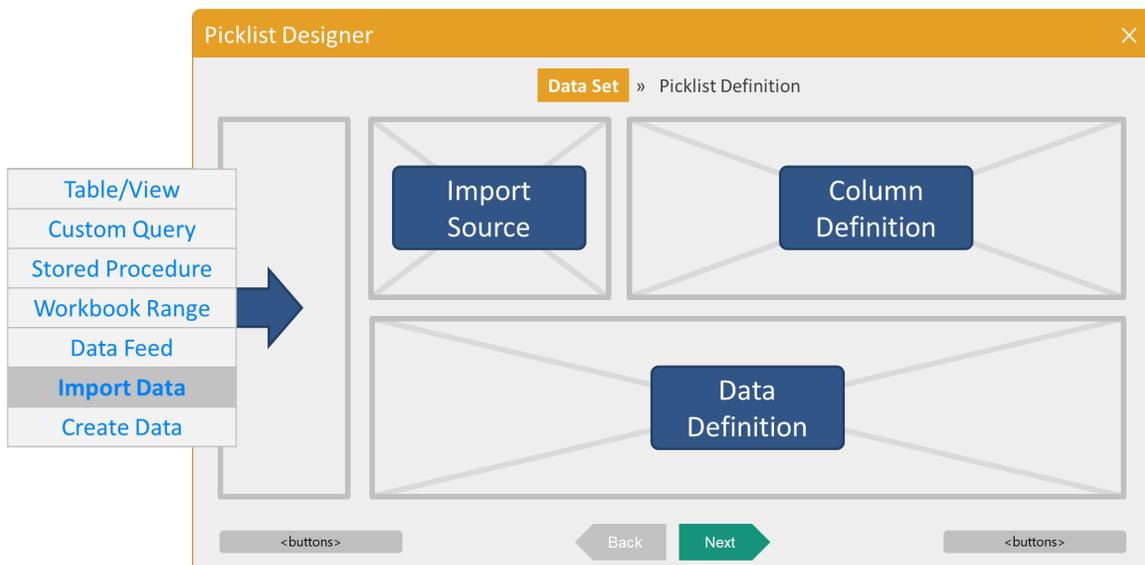
13.15.2 Static Data

This section describes the static data methods for creating a picklist. Creating a dynamic data picklist is discussed in Section 2.1, “Dynamic Data”.

Note: Picklist Parameter and Filtering is not functional with picklists using a Static data source.

13.15.2.1 Import Data

When first creating a new picklist, selecting the “Import Data” option will allow a one-time import of data from either an Excel file or a comma separated value file (.CSV).



13.15.2.1.1 Import from Excel

Importing data from an Excel file or Delimited file are essentially the same, with the exception being Excel files can refer to specific Worksheets or Named Ranges; whereas a delimited file can only contain a single dataset.

Selecting the “Excel” option in the dropdown will display options related to using an Excel spreadsheet as a data source. Importing Excel data using this method will import the data as static data into the iDashboards application.

Note: The imported data will only get saved within iDashboards when the picklist is saved.



Notice

Excel files can also be used as a dynamic data source, meaning that every time you upload an updated spreadsheet into iDashboards, the new data will be reflected in every chart that is pointing to it. To set this up, an iDashboards administrator must use the Workbook Database (See the iDashboards Administrator's Manual for more information.)

13.15.2.2 Define Data Set

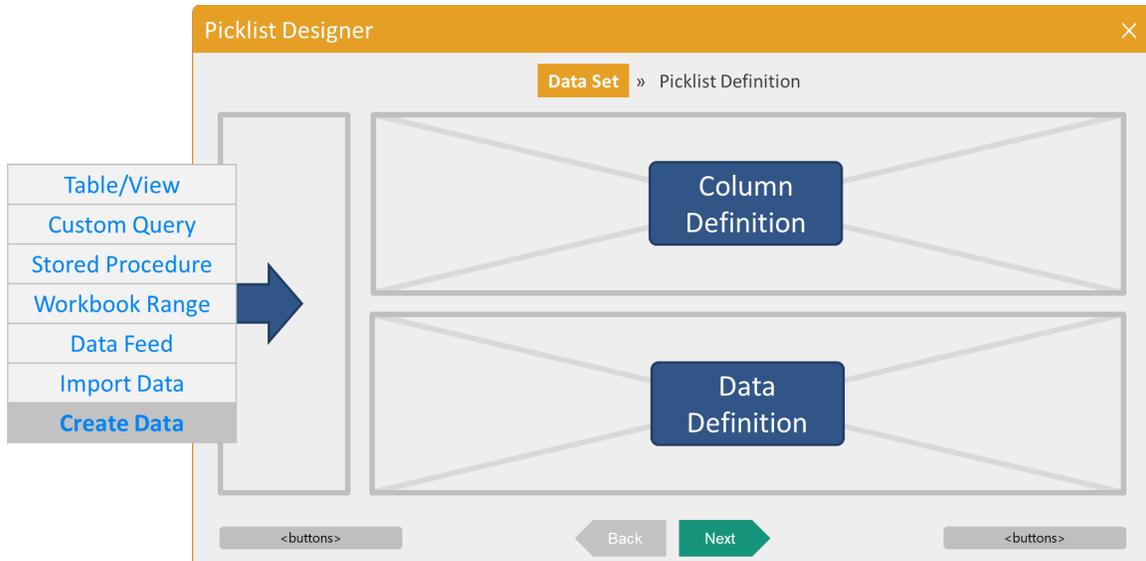
Check the box 'Data includes header row:' if the first row of the Excel data contains the name of each column. When a header row is used, the Excel column names are retained and stored along with the data in iDashboards. Column names should be less than 30 characters long, should begin with a letter and should only contain letters, digits and spaces. If the box is not checked, the first row in the Excel data is imported as the first data row.

- **Select** – Click the “Select...” button to locate the file.
- **Browse** – Files can be imported from local or network storage locations. Select one file.
- **Import Method** – There are essentially three methods for defining the import method:
 - **<none>** - If the Excel file is very 'clean', no further action is required prior to import.
 - **<Worksheet Name>** - If the Excel file contains multiple worksheets, by default, the first worksheet will be imported unless the worksheet name is entered.
 - **<Named Range>** - *Preferred* If the Excel file contains Named Ranges, enter its name into this field.
- **Import** – Click the “Import” button to extract the data.
- **Review/Edit** – Review the results:
 - If the column preview or the data values appear incorrect, examine the results and troubleshoot as necessary. Ensure the Excel data is in an appropriate format, shape, and structure. Best results come from using the Named Range options. Make changes to the Excel file and attempt to import the data again.
 - If the column preview or the data values appear mostly correct, consider utilizing the options to add/remove individual columns or rows. Editing the actual values can take place at this time.

13.15.2.3 Create Data

Creating data manually might be an appropriate option when no other option offers the same benefit. One reason not to use this option is due to the time it takes to manually identify all columns and all row-values for an entire data set. However, one reason to use this option is to quickly create a picklist with a few values that never, or rarely, change.





13.15.2.3.1 Define Columns

A basic minimum requirement for all iDashboards picklists is one picklist value column. However, if the intention is to have the same values appear as both the picklist value column and the picklist display column, then it is valid to have a data source with only a single column. By default, the “Create Data” option has a single pre-defined column, and therefore requires no additional information prior to configuring all other stages of the picklist. However, doing so, will be of little value because there won’t be any data values in a dropdown parameter that is associated with this picklist.

Columns can be added, deleted, or sorted (for personal preference). To reorder the columns, click-and-drag the column into the desired position. To create or delete columns, use the “Add” and “Remove” icons as shown below.

Define Columns for this Data Set:

Data Set Column	Data Type
State	String ▼
Population	Number ▼

+
🗑️

- Enter Column Name** – The image above shows two columns, each with a customized name. Some picklists have visible labels and therefore merit grammatically correct names. However, at this time it is not necessary to be strict about the grammatical value because the next stage of creating a picklist allows for using a mapped name. However, it is often found to be helpful to have accurate names at this time.

- **Select Column Type** –. The user must select one of the three standard iDashboards data types; String, Number or Datetime for each Column Name added to the picklist.

13.15.2.3.2 Create Data

For each Column Name defined, the user will be able to enter data associated to the data column, by using the lower portion of the Picklist Designer. As shown below, three records have been manually added by the user. To reorder the rows, click-and-drag the row into the desired position. To create or delete rows, use the “Add” and “Remove” buttons shown below.

State	Population
MI	10000000
FL	21000000
CA	40000000

13.15.2.4 Editing Data

After a picklist has been configured and saved, it is entirely possible to make additional changes to a “Static Data” picklist.

- Columns and data can be added without causing issues to the existing picklist.
- Columns that are removed or renamed will lose their mapping within the “Picklist Definition” stage, requiring additional steps to resolve the mapping.
- Data values can generally be added without causing issues to the existing picklist.
- Data values can generally be removed without causing issues to the existing picklist.

13.15.3 Picklist Parameters

Picklist parameters are similar to chart parameters in the way they can be used to filter picklist data and supply the value of a stored procedure or data feed argument. The primary purpose of picklist parameter is to establish a relationship to another chart or dashboard parameter via cascading parameters. The definition of chart and dashboards parameters are similar, so this description will refer to chart parameters, but the same concept applies to dashboard parameters. When a chart parameter is defined, it may be desired that the value of that chart parameter determines the possible values for a dropdown associated with another chart parameter. For example, if one chart parameter is ‘State’ and another is ‘City’, the value of the ‘State’ chart parameter can be used to filter the dropdown choices displayed for the ‘City’ parameter.



The user creating the picklist defines the parameter name and type. These parameter definitions must match the name and type of the chart parameter that will be used to provide the value for picklist parameter. Therefore, the value of the picklist parameter will be provided by the value of a chart parameter that has been defined earlier in the sequence of chart parameter definitions. This concept becomes clearer in the context of defining cascading parameters.

13.15.4 Completing the Picklist Definition

The last component of a picklist is the 'Picklist Definition' stage. This panel is where the association of the picklist data columns is made to the actual picklist value column and the picklist display columns. Additional properties of the picklist are specified in this panel as well.

The screenshot shows the 'Picklist Designer' window with the 'Picklist Definition' tab selected. The 'Picklist Name' is 'State List' and the 'Category' is 'Sales'. The 'Picklist Data Type' is 'String'. Under 'Specify the Picklist Columns', there is a table with the following data:

	Data Column	Data Type	Function
Value	STATE	String	[None]
Display	<Same as Value		

Below the table, there are additional settings: 'Sort On' (empty), 'Sort Order' (Ascending), 'List Limit (1-10,000)' (empty), and 'Remove Duplicate Rows' (checked).

Picklist Name – The picklist must be given a name of less than 50 characters. The name will later be referenced when building input parameters at the chart or dashboard level.

- **Category** – Unlike charts and dashboards, a picklist must be saved into a non-Personal category. Role-based permissions will determine if a user is able to save within the category. The category drop-down menu will also allow a picklist to be moved from one Category to another.
- **Data Column** – Like charts, a picklist must connect to columns of data. The value name must be mapped to a data column. The display name can either duplicate the value or connect to a second column to represent the data shown in the picklist. Functions for each column can be used to aggregate the data if needed.

Note: When a picklist is used with an input parameter, the user can identify the 'Initial Value'. The value entered must come from the 'Value' data column and not the 'Display' data column.

- **Formatting** – Picklists created from a column data type of Datetime, Number or String has an option to format their display. For example, the setting for removing duplicate rows is on by default. A row limit may also be used to restrict lists that are too long.
 - **Sorting**: Any picklists, regardless of the data type, have the ability to display values in a particular order. Options for sorting are Ascending or Descending, based on either the value or display column.
 - **Date Formatting**: Picklists created from a column data type of Datetime can be formatted to display certain date formats. Changes can be made when the Edit button is selected for the Display Format.
 - **Number Formatting**: Picklists created from a column data type of Number can be formatted to display certain numeric formats. Changes can be made when the Edit button is selected for the Display Format.
- **Preview List** – During the picklist definition, clicking on 'View Data' will open a preview window illustrating the Value and Display fields.
- **Save Picklist** – Clicking on Save within the Picklist Definition window will ensure the picklist is available in the future while building input parameters.



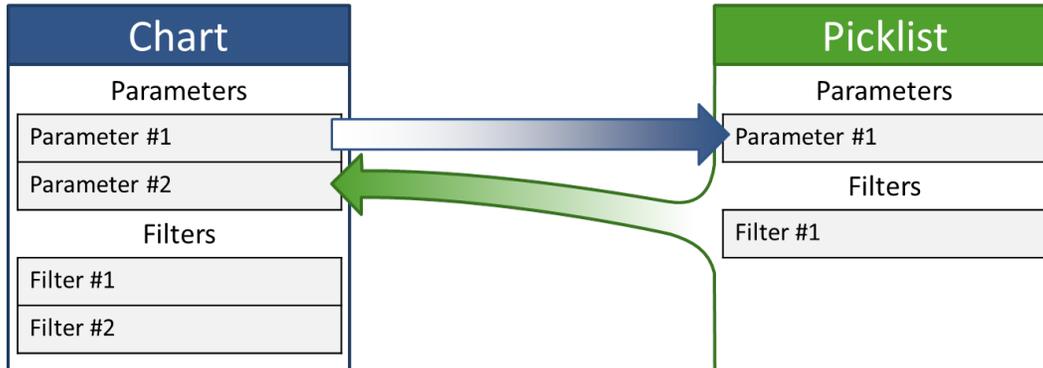
13.16 Using Cascading Input Parameters

A cascading input parameter requires a combination of using picklists with input parameters. When multiple parameters exist within a chart or dashboard, cascading parameters offer a hierarchy when users interact with the values. The configuration of this feature is intentional and will allow the options of a parameter to be determined by another parameter. A parameter with values determined by another parameter is called a cascade target and the parameter supplying the value is called a cascade source.

Only parameters that are dropdowns with their values specified by a picklist can be cascade targets.

13.16.1 Flow Diagram

In the image below is an example for this entire section. The goal of the cascading input parameter is to affect the values within the cascade target (referred to as “Parameter #2”) by the first selected cascade source (referred to as “Parameter #1”). This diagram is showing the minimum number of components needed to implement a cascading input parameter.



A best-practice for the design sequence, is as follows:

1. Create the Chart
 - a. Add the first input parameter (ex. name = state)
 - b. Add the first filter using the first parameter (ex. Column 'STATE' = `#{param:state}`)
2. Create the Picklist
 - a. Add the parameter (ex. name = state)
 - b. Add the filter using the parameter (ex. Column 'STATE' = `#{param:state}`)
3. Edit the Chart
 - a. Add the second input parameter (**Note: This must use the Control Type of 'Dropdown List'**)

- i. During the Parameter Value Definition, select 'Picklist' and locate the previously created picklist. Parameterized picklists are designated with a 'P' icon next to the name of the picklist.
- b. When this parameter is finished getting configured, the cascading hierarchy should be recognized with an indented arrow between the first parameter and the second parameter.
- c. Add the second filter using the second parameter

13.16.2 Picklist Parameter

A picklist parameter has the ability to connect to a matching parameter from either a chart or dashboard. The user creating the picklist defines the parameter name and type. These parameter definitions must match the name and type of the chart or dashboard that will be associated to the picklist. Therefore, spelling and punctuation of these components is critical.

13.16.3 Picklist Filter using Parameter

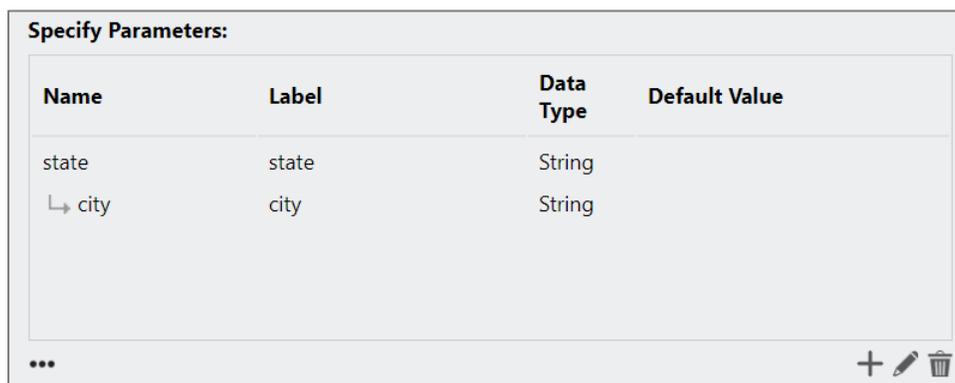
In addition to filtering the values of a picklist, "Picklist Filtered Data", picklist filters are necessary to use cascading parameters. Once the parameter within the picklist has been defined, a corresponding filter must be added. Assume there are two filters. The first filter 'MONTHX' is configured to show data within the last 120-days. The second filter 'STATE' is configured to show a particular geography within the United States.

13.16.4 Define the Cascading Parameter

A cascade target is defined by associating it with a parameterized picklist (a picklist with a parameter). The relationship between a cascade target and a cascade source is determined by the order in the chart or dashboard parameter list. In order for the definition of a cascade target to be valid, all of the picklist parameters must match parameters that appear earlier in the parameter list. A picklist can have multiple picklist parameters, but only the values of parameters that are part of a 'cascade chain' to the cascade target will be used to provide values of the picklist parameters. The cascade source name and data type must match that of the picklist parameter associated with the cascade target.

Note: The typical use of cascading parameters will have a single cascade source that provides a single value for the picklist parameter associated with the target.

A cascade target is identified with a hierarchy arrow in the Chart Designer:



14. Analytics

Real-Time Analytics will allow end-users to perform what-if analysis in real-time. It provides the ability to apply calculations on real-time data feeds using user-specified input parameters, and create any number of what-if scenarios that can deliver extremely powerful insight. Furthermore, Real-Time Analytics will give the ability to derive real-time statistics on incoming data, and help determine exceptions and outliers. Real-Time Analytics is based on two iDashboards technologies; Derived Columns and Input Parameters.

- **Derived Columns** – is an iDashboards feature that is only available with Real-Time Analytics. Derived Columns allow you to add one or more columns (Y-axes) to a chart that is the result of a calculation (or “Expression”) done on one or more other columns.
- **Input Parameters** – is standard functionality in iDashboards. Input Parameters allow dashboard developers to create charts and dashboards that present end users with input prompts (textboxes, dropdowns, sliders, etc.). End users then enter values into these input selectors and click update. The dashboards/charts then take these input values and perform some action. For the purposes of the Real-Time Analytics, charts take these input values and use them in expressions to calculate what-if results to display to the end user.

When these two features are used together, the Input Parameters values are passed into the expressions of the Derived Columns. This is the basis of Real-Time Analytics in iDashboards.

Note: It is assumed that the reader has knowledge of the dashboards/chart creation process.

14.1 Derived Columns

14.1.1 Introduction

This section explains how Derived Columns work without any mention of Input Parameters. It is meant to give you a clear understanding of Derived Columns. Later you will read about Input Parameters in this manual and in the iDashboards User’s Manual. From there you are presented with many examples of Derived Columns being used with Input Parameters (which is the basis for Real-Time Analytics).

14.1.2 Derived Column Concept

A derived column does not connect to a data source. Instead, a derived column is the result of a formula/equation, or JavaScript expression. As an example, review the table below. The data source currently contains 3 columns: ‘City’, ‘Male_Admits’ and ‘Female_Admits’. The data source does not contain the total number of admits for each city. Therefore, a derived column, dependent on other data, has been created ‘Total_Admits’. In this example, the formula would look like: $B+C=D$. Derived columns within iDashboards can be used to help provide charts with information that does not come directly from the data source. In this example, the dashboard builder might decide to use column D ‘Total_Admits’ with a Speedometer or Bar chart, while hiding column B and C.

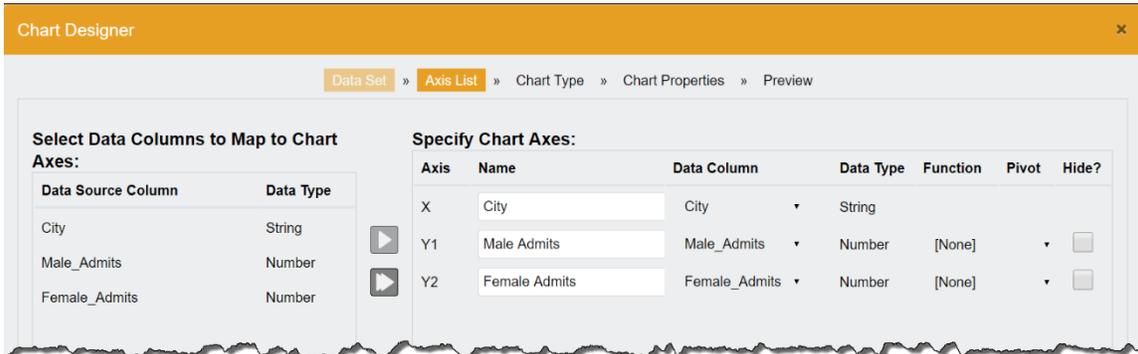


Data Source has 3x Columns			Derived Column
A	B	C	D
City	Male_Admits	Female_Admits	Total_Admits
Atlanta	47	66	113
Chicago	52	53	105
Detroit	36	40	76
New York	45	45	90
Raleigh	29	31	60
Seattle	38	40	78

14.1.3 Derived Column within iDashboards

When creating a chart in iDashboards, it is common to assign each Y-axis to a column from the data source. Configured in this way, each Y-axis will simply display the data stored in each column. However, some situations may require one or more Y-axes to display data that is a result of a calculation performed on data from the data source. To accomplish this, you would configure each of these Y-axes as a Derived Column. Derived columns display data that is derived from other data in the data set and will be the result of a mathematical expression written in JavaScript.

Consider the following example where the X-axis maps to the 'City' column, the Y1-axis maps to the 'Male_Admits' column, and the Y2-axis maps to the 'Female_Admits' column. The image below shows how the Chart Designer window would be configured and the table below is a representation of the data set.



X	Y1	Y2
City	Male_Admits	Female_Admits
Atlanta	47	66
Chicago	52	53
Detroit	36	40
New York	45	45
Raleigh	29	31
Seattle	38	40



In addition to displaying admissions by gender details in the chart, suppose you would like to display the total admissions, for each city. This can be accomplished by adding a derived column to the chart in the Y3 position, called 'Total Admits', whose values are derived from a calculation performed on Y1 and Y2. The steps below describe the setup details.

- Create a new chart and select the Data Set and add the necessary X, Y1 and Y2 data columns shown.
- From the 'Axis List' stage of the Chart Designer, click the "+" button to add an additional data column in the Y3 position. Name the column "Total Admits".
- Map Y3 to the "(Expression)" option using the drop-down list from the "Data Column"
- In the Expression window, enter the mathematical JavaScript expression:
 - `y1 + y2`
 - Select the Data Type "Number"

Specify Chart Axes:						
Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	City	City	String			
Y1	Male Admits	Male_Admits	Number	[None]		<input type="checkbox"/>
Y2	Female Admits	Female_Admits	Number	[None]		<input type="checkbox"/>
Y3	Total Admits	(Expression)	Number			<input type="checkbox"/>

Note: The 'Total Admits' column has "(Expression)" selected for its Data Column. This indicates it is a derived column. When you select "(Expression)" for a Data Column, the ability to add/edit the expression becomes available. The expression window is where you enter the JavaScript expression that will be used for the Derived Column.

Note: The Data Type for the expression must be selected from the list of options: Number, String, Datetime.

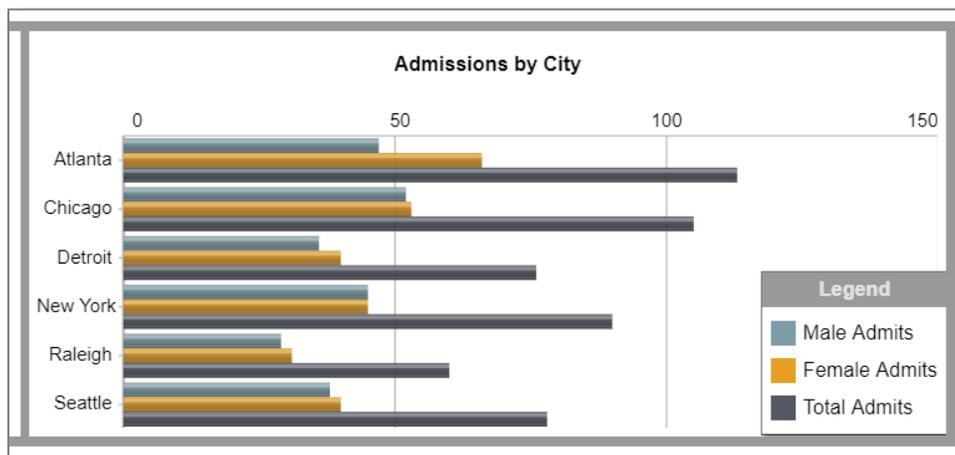
Note: Using "y1 + y2" is a relative equation. If either of the 'admits' columns were to change positions by the addition of new axes values, the equation would no longer be correct. Therefore, review the section below for using an alternative equation that is based on the axis name:

```
$val["Male Admits"]+$val["Female Admits"]
```



When the chart is loaded, the server will evaluate the expression for every row in the data set and insert the result in the Y3 column. The resulting data set would look like the table below. The resulting chart is shown in the figure below.

X	Y1	Y2	Y3
City	Male_Admits	Female_Admits	Total Admits
Atlanta	47	66	113
Chicago	52	53	105
Detroit	36	40	76
New York	45	45	90
Raleigh	29	31	60
Seattle	38	40	78

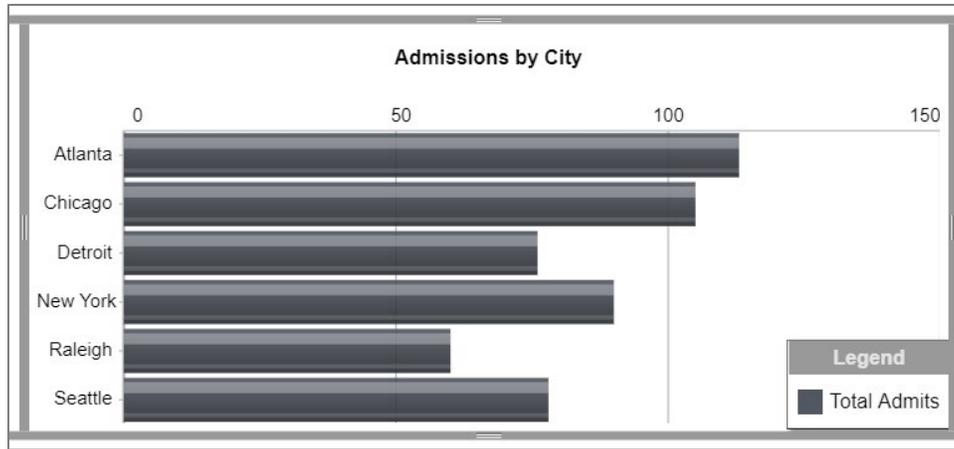


14.1.3.1 Hiding axes

This example provides an ample opportunity to implement hidden axes within a chart. Within the Chart Designer > Axis List, the Y1 and Y2 axes can be hidden. Add a checkmark to each axis as shown below.

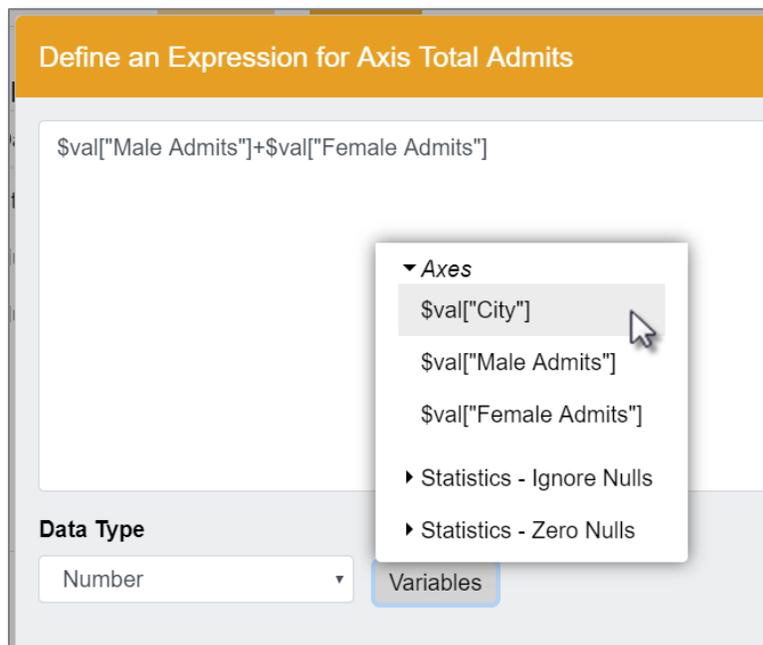
Specify Chart Axes:						
Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	City	City	String			
Y1	Male Admits	Male_Admits	Number	[None]		<input checked="" type="checkbox"/>
Y2	Female Admits	Female_Admits	Number	[None]		<input checked="" type="checkbox"/>
Y3	Total Admits	(Expression)	Number			<input type="checkbox"/>

Hiding these axes will result in a chart that only displays the derived value, as shown below:



14.1.4 Built-in Variables

Expressions used in derived columns can use a number of built-in variables that are recognized by iDashboards. When populating the JavaScript expression, the user can select from the “Variables” list menu:



14.1.4.1 Built-in Axis Variables

Corresponding to each “X”, “Y1”, “Y2”, etc. axis are variable names using the following syntax:

```
$val[\"<axis name>"]
```

See the image above for an example of how to use variables to achieve “y1 + y2”.

14.1.4.2 Built-in Statistic Variables

Furthermore, there are built-in variables that provide common statistics. There are two options for using these variables:

- **Ignore NULLs** – Will ignore NULL values in the dataset when applying the statistical calculation
- **Zero NULLs** – Will recognize NULL values in the dataset as zeros when applying the statistical calculation

In its simplest form, a statistical variable name is constructed as follows:

```
<column id><underscore><stat name>
```

OR

```
y2_mean
```

The column id would be an Axis (x, y1, y2, etc.), and signifies the column to which the statistic applies. Alternatively, using the axis variable would use the following syntax:

```
$(stat name>["<axis name>"]
```

Or

```
$mean["Female Admits"]
```

The statistic name is one of the following:

- **min** – minimum value in the column
- **max** - maximum value in the column
- **mean** – mean (average) of all values in the column
- **med** – median of all values in the column
- **stdev** – standard deviation of all values in the column
- **total** – the summed total of all values in the column

For example, y2_mean would return the mean, or average, of all non-null values in the Y2 column. y4_max would return the maximum value in the Y4 column.

If a statistic variable name is prefixed by an underscore, this means the statistic is calculated by counting all null values as zero. For example, _y2_med would be the median of all values in the Y2 column, counting all nulls as zero.

Note: iDashboards recognizes its built-in variables in all upper case or all lower case, but not mixed case (such as Y3P_Mean, which would be unrecognized).

The following tables show example source data and the twelve statistical variables with their resulting values when applied to the source data. Both the non-underscored variables (ignoring nulls) and the underscored variables (treating nulls as zeros) are shown.



Source Data		Non-Underscored Variables (ignoring nulls)		Underscored Variables (treating nulls as zeros)	
X	Y1				
Month	Count	Variable	Value	Variable	Value
Jan	2	y1_min	2	_y1_min	0
Feb	3	y1_max	5	_y1_max	5
Mar	5	y1_mean	3	_y1_mean	2
Apr	null	y1_med	2.5	_y1_med	2
May	null	y1_stdev	1.225	_y1_stdev	1.732
Jun	2	y1_total	12	_y1_total	12

Note: Each of the above variables applies to all rows of data in the column they refer to. iDashboards also provides a set of variables that apply only to the subset of rows a pivot selects.

14.1.5 Data Types of Derived Columns

In most cases, derived columns will perform calculations on numeric data and will return a number. However, they may also work with string data from other columns and return a string. For example, if the X column contains 'Atlanta' and the Y1 column contains the number 47, the following expression will result in the string 'Atlanta47'.

```
x + y1;
```

The following is another example of a derived column returning a string. This example uses a JavaScript If...Else statement. The derived column will return the string SMALL, MEDIUM or LARGE depending on the Y2 column value.

```
if (y2 < 100) {
    'SMALL';}
else if (y2 < 200) {
    'MEDIUM';}
else {
    'LARGE';}
```

When configuring a derived column, the user must select the data type that will be returned. The choices are Number, String and Datetime.



14.1.6 Order of Evaluation

Derived columns are evaluated in ascending order of Y-axes (Y1 first, then Y2, and so on). For example, a derived column in the Y2 column will be evaluated before a derived column in the Y5 column. Due to this evaluation order, a derived column that references another derived column must exist at a higher Y-axis position than the one it references. The following example demonstrates this requirement.

Suppose a chart contains two derived columns, the Y4 and Y5 columns. Suppose also that the expression for the Y4 column is:

```
y1 + y2 + y3;
```

Suppose that the Y5 column is as follows, which uses the y4 variable to access the value in the Y4 column:

```
y4 / 24;
```

This example will work because the Y4 column is evaluated before the Y5 column. Because this is the case, the y4 variable used in the Y5 expression will have already been evaluated and will contain a useable value.

On the other hand, if the Y4 and Y5 expressions were switched, the derived column in the Y4 column would fail because it references the derived column in the Y5 column, which hasn't been evaluated yet.

To recap, the first set of derived columns below will work, the second set will not.

Valid	Invalid
Y4 derived column = y1 + y2 + y3 Y5 derived column = y4 / 24	Y4 derived column = y5 / 24 Y5 derived column = y1 + y2 + y3

It is also important to note that derived columns can contain multiple expressions and that it is the last evaluated expression that is used as the derived column value. For example, if a derived column was configured to use the following expressions, the derived column would result in the value returned from y2 + 34.

```
y2 / 24;
```

```
y3 - 65;
```

```
y2 + 34;
```



14.1.7 Applying Built-in Variables to Pivots

A chart within iDashboards can contain one or more pivots. When a user selects a value in a pivot, the pivot performs a filter on the entire data set and only displays the data it relates to.

When configuring a chart to use a derived column, a user may want to have that derived column perform its calculation(s) on the subset of data the pivot selects, rather than the entire data set. To accommodate this functionality, iDashboards provides a variation on the twelve statistical variables already introduced. To apply a statistical variable to a pivot, add the letter “p” immediately after the column id. For example, the variable `_y3p_total` would contain the total of the values in the Y3 column (counting nulls as zero) for which the pivot selects.

Note: It is common to configure a derived column as a pivot column. However, when this is the case, the variables outlined in this section (statistical variables that pertain to pivots) should not be used in the derived column expression.

14.1.8 Limitations for Derived Columns

There are two limitations when working with derived columns:

- No filtering on derived columns

It is not possible to apply a filter to a derived column due to the fact that derived column expressions are evaluated after all filtering has taken place. If you are configuring a filter on a table that uses a derived column, the derived column will simply not show up as an option when selecting the filter’s data column.

- No drilldown filtering on derived columns

It is common to configure one chart to drill down into another chart and to have the second chart perform a “drilldown filter”. A drilldown filter automatically selects a value in the second chart’s pivot that corresponds to the value the user clicked in the first chart. Since drilldown filtering is a form of filtering, it cannot be used in conjunction with a derived column.

14.1.9 JavaScript Fundamentals

Derived columns use JavaScript for their expressions. The following is a brief outline of common JavaScript functionality that may be used in a derived column.

14.1.9.1 Declaring Variables

JavaScript variables are used to hold values or expressions. Variables must be declared using the `var` statement before they are used. Variable can also be set when declared.

```
var InputText;  
  
var InputText = 'Hello World';
```

14.1.9.2 Conditional Statements

In your derived column expression, you may want to perform different actions for different conditions. JavaScript provides a number of conditional statements that achieve this functionality. The following are three of the most common conditional statements that can be used in derived column expressions.



If Statement - Use this statement if you want to execute some code only if a specified condition is true. The syntax of the statement is:

```
if (condition) {  
    code to be executed if condition is true  
}
```

If...Else Statement - Use this statement if you want to execute some code if the condition is true and another set of code if the condition is false. The syntax of the statement is:

```
if (condition) {  
    code to be executed if condition is true  
}  
else {  
    code to be executed if condition is not true  
}
```

Switch Statement - Use this statement to select one of many blocks of code.

```
switch(n) {  
    case 1:  
        execute code block 1; break;  
    case 2:  
        execute code block 2; break;  
    default:  
        code to be executed if n is different from case 1 and 2  
}
```

Note: If the code in the above “code to be executed” sections contains only one line, the preceding and following curly brackets can be excluded. For example:

```
if (x > 10)  
    x = x + 5;
```



14.1.9.3 Operators

There are many JavaScript operators that can be used in expressions that will allow for arithmetic, assignment, comparison and logical comparison operations. Samples of these operators are shown below:

Assuming $y = 5$

Operator	Description	Example	Result
+	Addition	$x=y+2$	$x=7$
-	Subtraction	$x=y-2$	$x=3$
*	Multiplication	$x=y*2$	$x=10$
/	Division	$x=y/2$	$x=2.5$
++	Increment	$x=++y$	$x=6$
--	Decrement	$x=--y$	$x=4$

Arithmetic

Assuming $y = 5$ and $x = 10$

Operator	Example	Same As	Result
=	$x=y$	$x=y$	$x=5$
+=	$x+=y$	$x=x+y$	$x=15$
-=	$x-=y$	$x=x-y$	$x=5$
=	$x=y$	$x=x*y$	$x=50$
/=	$x/=y$	$x=x/y$	$x=2$

Assignment

Assuming $x = 5$

Operator	Description	Example
==	is equal to	$x==8$ is false
!=	is not equal	$x!=8$ is true
>	is greater than	$x>8$ is false
<	is less than	$x<8$ is true
>=	is greater than or equal to	$x>=8$ is false
<=	is less than or equal to	$x<=8$ is true

Comparison

Assuming $y = 3$ and $x = 6$

Operator	Description	Example
&&	and	$(x < 10 \ \&\& \ y > 1)$ is true
	or	$(x==5 \ \ y==5)$ is false
!	not	$!(x==y)$ is true

Logical Comparison



14.2 Derived Columns Examples

This chapter shows many examples of using Derived Columns without the use of Input Parameters. This chapter is meant to strengthen the understanding of Derived Columns.

14.2.1 Example 1 – Calculating differences

When multiple Y-Axes represent data captured from different timeframes, it is possible to use a derived column to calculate the change. In this example, the data source contains Sales information on three products from a produce company, and two columns of sales data (representing the previous and current sales year).

X	Y1	Y2
Product	Sales_Previous_Year	Sales_Current_Year
Apples	25142	28574
Bananas	14587	15247
Oranges	32415	28560

During the chart design, additional axes are created as an expression:

Desired	Expression
Y3 - Difference	$y2 - y1;$
Y4 – Percentage of Previous	$y2 / y1 * 100;$
Y5 - Percent Change	$(y2 - y1) / y1 * 100;$

X	Y1	Y2	Y3	Y4	Y5
Product	Sales_Previous_Year	Sales_Current_Year	Diff	% of Prev	% Change
Apples	25142	28574	3432	113.6505	13.6505
Bananas	14587	15247	660	104.5246	4.5246
Oranges	32415	28560	-3855	88.1074	-11.8926

14.2.2 Example 2 – If Statement

You would like to create a Tabular chart that will return the 2-digit version of set of numbers from 1 to 12. The return value will be of type string.

letters	numbers
x	y1
a	1
b	2
c	3
d	4
e	5
f	6
g	7
h	8
i	9
j	10
k	11
l	12



Expression for column 'y2'

```
var retval = y1.toString();
if (retval.length < 2)
    retval = "0" + y1;
else
    retval;
```

letters	numbers	new text column
x	y1	y2
a	1	01
b	2	02
c	3	03
d	4	04
e	5	05
f	6	06
g	7	07
h	8	08
i	9	09
j	10	10
k	11	11
l	12	12

The expression should return a string.

14.2.3 Example 3 – If...Else Statement

You would like to create a chart that will include a column to display “Pass” if a value is equal to or greater than 90% and “Fail” if the value is less than 90%. You can use a Tabular chart and a derived column with an If...Else Statement to achieve this.

Student	Grade	Pass/Fail
Student 1	78	Fail
Student 2	93	Pass
Student 3	96	Pass
Student 4	89	Fail
Student 5	90	Pass
Student 6	76	Fail

Expression for “Pass/Fail” Column (as String):

```
if (y1 >= 90) {
    'Pass';
}
else {
```



```
'Fail';
}
```

14.2.4 Example 4 – Preventing Divide-By-Zero Error

An expression that gives the number of standard deviations above or below the mean for a particular column would be:

```
(y2 - y2_mean) / y2_stdev;
```

To guard against the divide-by-zero error that would occur if `y2_stdev` is zero (as when all the values in the column are identical), the JavaScript `If...Else` conditional statement can be used. The following expression will return zero if `y2_stdev` equals zero, otherwise it will return the result of the calculation $(y2 - y2_mean) / y2_stdev$.

```
if (y2_stdev == 0) {
  0;
}
else {
  (y2 - y2_mean) / y2_stdev;
}
```

14.2.5 Example 5 – Calculating Daily Count

This example will display the number of sales that occurred on each day of the week (Monday, Tuesday, etc.) for a small period of time. The source data contains two columns, Date and Sales Count, which will be the X and Y1 axes, respectively.

After selecting a data source, the Data Columns for the axes should be configured as shown. Note the pivot on the 'Day' column. This will allow the user to select which day of the week to view at any one time. All dates that fall on that day (i.e. Thursday) will be displayed along with a total sales count for that day.

Specify Chart Axes:						
Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	Date	month_ ▾	Datetime ⚙			
Y1	Sales Count	Value ▾	Number	[None] ▾	▾	<input type="checkbox"/>
Y2	Day	(Expression) ▾ ⚙	String		1 ▾	<input type="checkbox"/>

The expression for the 'Day' column will use a JavaScript switch statement and should be written as follows:

```
var TextDay;
var NumericDay = x.getDay();
```

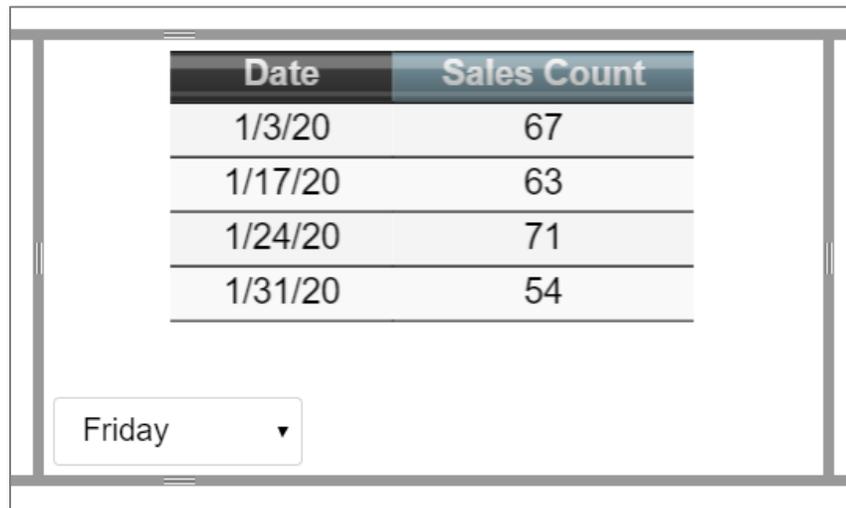


```

switch(NumericDay) {
case 0: TextDay = 'Sunday'; break;
case 1: TextDay = 'Monday'; break;
case 2: TextDay = 'Tuesday'; break;
case 3: TextDay = 'Wednesday'; break;
case 4: TextDay = 'Thursday'; break;
case 5: TextDay = 'Friday'; break;
case 6: TextDay = 'Saturday'; break;
}

```

The expression should return a string. The resulting chart would look like the image below:



14.2.6 Example 6 – Deriving a Quarterly Timeframe Pivot

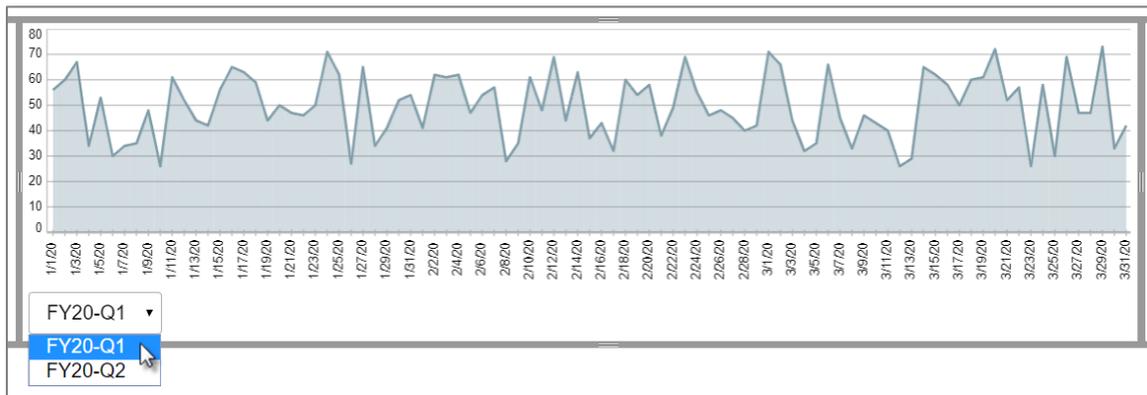
This example will create quarterly groupings for any range of dates. The original data set appears as follows:

month_	Value
1/1/2020	36
1/2/2020	38
1/3/2020	45
1/4/2020	34
1/5/2020	71
1/6/2020	42
1/7/2020	28
...	...

The chart Axes:

Specify Chart Axes:						
Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	Date	month_	Datetime			
Y1	Closing	Value	Number	[None]		<input type="checkbox"/>
Y2	Quarter	(Expression)	String		1	<input type="checkbox"/>

The resulting chart appears as follows:



There is a pivot on the quarter, where those values are dynamically generated based on the data! The expression for this example is:

```
var month = x.getMonth() + 1;
var qtr;
if(month <= 3) qtr = 1;
else if(month <= 6) qtr = 2;
else if(month <= 9) qtr = 3;
else qtr = 4;
var yr = x.getYear() - 100;
yr = (yr < 10) ? '0' + yr : yr.toString();
'FY' + yr + '-Q' + qtr;
```

In this code we set a variable called “month” with the JavaScript `getMonth()` function to be the month that the X-axis falls into and adding 1 since JavaScript starts with January as month 0 (zero). We are creating another variable called “qtr” that will be evaluated based on the month. A logical ‘IF’ statement evaluates the month to determine what quarter the month belongs. Another variable called “yr” is created and set to the year that the X-axis falls using the JavaScript `getYear()` function. We subtract 100 from the year since the JavaScript `getYear()`



function will return the year minus 1900. This means that 2013 will return 113 and our variable “yr” will be set to 13. If the year is evaluated to be less than 10 a 0 (zero) will be appended to the front of the variable and this variable will be converted to a string value with the JavaScript toString() function. Finally, a string of ‘FY’ plus the year (12, 13 or 14 in this instance) with a ‘-Q’ and quarter of 1-4 is created.



14.3 Analytics: Input Parameters

Input Parameters are explained in detail Chapter 19, “Input Parameters”. The reader should familiarize themselves with the core functionality of Input Parameters by reading that chapter before moving forward in this chapter.

That chapter explains that charts can use Input Parameter values to:

- Filter chart data based on the input values.
- Display the input values in chart titles.
- Use the input values in a chart’s custom SQL queries.
- Use the input values as parameters for stored procedures.
- Pass the input values down during drilldowns.

That chapter does not explain that Input Parameters can also be passed into the expressions of Derived Columns. This is the basis for Real-Time Analytics. To do this, you will make use of a special Derived Columns “parameter” macro that acts very much like the standard Input Parameters “PARAM” macro. The Derived Columns parameter macro is of the form:

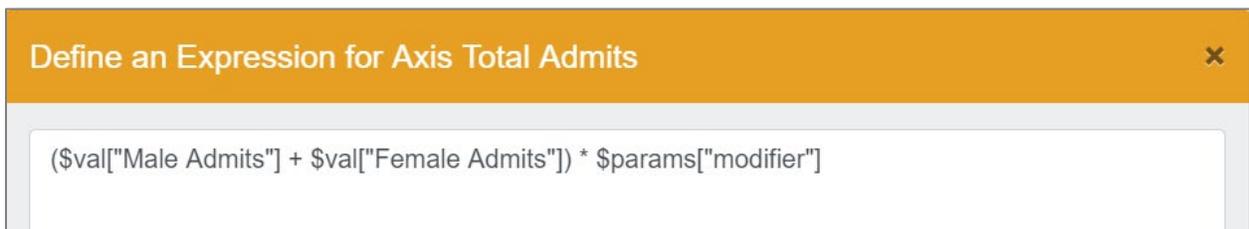
```
$params["<parameter name>"]
```

For example, if an Input Parameter is called “modifier”, the Derived Columns parameter macro would appear as:

```
$params["modifier"]
```

Just as the PARAM macro in chart filters (for example), you can use the \$params in the JavaScript expressions that make up Derived Columns. For example, if you would like to add an Input Parameter named “modifier” to the Derived Column example, you would update the expression as:

```
($val["Male Admits"]+$val["Female Admits"])*$params["modifier"]
```



14.4 Real-Time Analytics Examples

14.4.1 Example 1 – Headcount and Expense Projection

The following example uses Real Time Analytics within a single chart. This dashboard allows the user to project future headcount and training expenses based on current departments and current headcount.

Data Set	
department	headcount
Sales	250
HR	10
Engineering	60
Support	20

The image below illustrates the two input parameters needed to adjust projected changes in headcount and the projected cost to train each person.

Specify Parameters:			
Name	Label	Data Type	Default Value
change_percent*	Change (%)	Number	2
training	Training Expense	Number	2800

... 

The image below illustrates the axis columns added as expressions:

Specify Chart Axes:						
Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	Department	department ▼	String			
Y1	Headcount	headcount ▼	Number	[None]	▼	<input type="checkbox"/>
Y2	Change	(Expression) 🔗	Number		▼	<input type="checkbox"/>
Y3	Cost	(Expression) 🔗	String		▼	<input type="checkbox"/>

Axis	Expression
Y2	$\$val["Headcount"] * ((\$params["change_percent"] / 100) + 1)$
Y3	$\$val["Headcount"] * \$params["training"]$



The results of this example appear below. The user can adjust the “Change” values using a spinner-input (ranging from -10% to +10%) along with the ability to make changes to the projected training expense per-person. Ultimately, the “Cost” is calculated.

Department	Headcount	Change	Cost
Sales	250	255	\$700,000
HR	10	10.2	\$28,000
Engineering	60	61.2	\$168,000
Support	20	20.4	\$56,000

*Change (%)

Training Expense

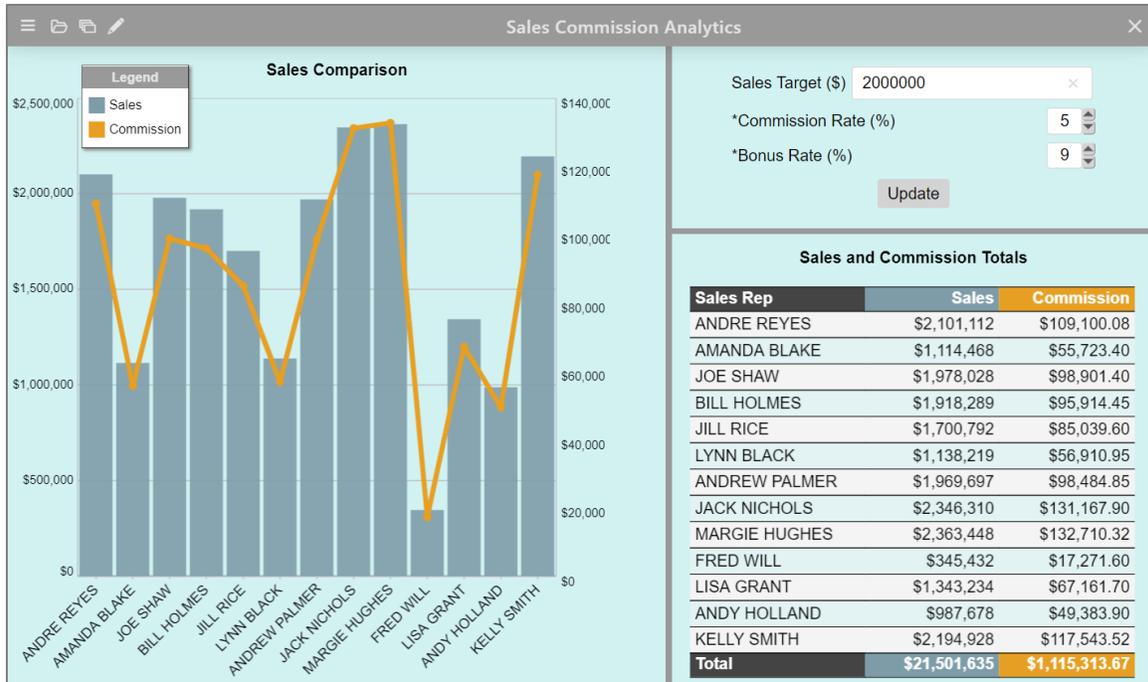
TIP: The columns “Headcount” and “Change” can be hidden if the chart only needs to show “Cost”.

14.4.2 Example 2 - Sales Commission Rate Analysis

The following example uses Real Time Analytics across the entire dashboard. This dashboard allows the user to enter various values that are used to calculate sales commissions. The user can try out different values for Sales Target, Commission Rate and Kicker Rate to see how they would affect total commissions paid.

Data Set	
SalesRep	Sales
ANDRE REYES	2101112
AMANDA BLAKE	1114468
JOE SHAW	1978028
BILL HOLMES	1918289
JILL RICE	1700792
LYNN BLACK	1138219
ANDREW PALMER	1969697
JACK NICHOLS	2346310
MARGIE HUGHES	2363448
FRED WILL	345432
LISA GRANT	1343234
ANDY HOLLAND	987678
KELLY SMITH	2194928





There are two charts used in this dashboard; “Sales Comparison” and “Sales and Commission Totals”. There is also a Dashboard Panel in the upper right. The two charts are essentially the same, they just used different chart types. All of their Chart Data settings are identical, which includes their axes names, data source, Input Parameters, Derived Columns, etc.

14.4.2.1 Input Parameters Setup

Below shows how the Dashboard Input Parameters for this dashboard are configured. The charts both have the same parameter names, labels and data types, but were created as text boxes with no default value.

Define Dashboard Parameters			
Name	Label	Data Type	Default Value
sales_target	Sales Target (\$)	Number	2000000
commission_rate*	Commission Rate (%)	Number	5
bonus_rate*	Bonus Rate (%)	Number	9

14.4.2.2 Derived Columns Setup

Both charts are configured using the same axes. You can see from the image below that the “Commission” Y-axis is configured as a Derived Column.



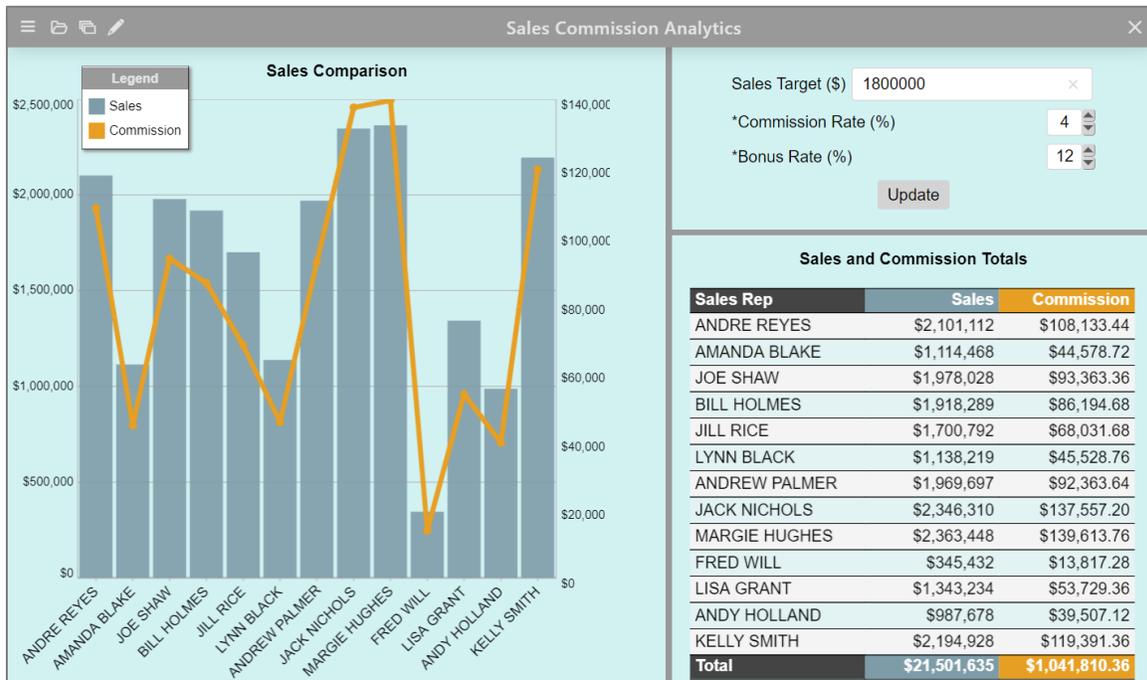
Specify Chart Axes:

Axis	Name	Data Column	Data Type	Function	Pivot	Hide?
X	Sales Rep	SalesRep	String			
Y1	Sales	Sales	Number	[None]		<input type="checkbox"/>
Y2	Commission	(Expression)	Number			<input type="checkbox"/>

This example will use the following JavaScript expression:

```
if ($val["Sales"] <= $params["sales_target"]) {
    $val["Sales"] * $params["commission_rate"]/100;}
else {
    ($params["sales_target"] * $params["commission_rate"]/100) +
    ($val["Sales"] - $params["sales_target"]) *
    $params["bonus_rate"]/100;}
```

By using the param_ macros, this formula will accept values entered into the three Input Parameters and then calculate the commissions for each Sales Rep. Like most analytical examples, the raw data will never change but the 'what-if' projections will be flexible. Using different values compared to the beginning of this exercise, the results could look like the following image:



15. Alerts

Warning!

ASSOCIATED COSTS OF MOBILE TEXT MESSAGES

SMS is essentially the text messaging service offered by all major mobile carriers. The iDashboards Alerts feature has an option to send an SMS text message to users when an alert is triggered. SMS text messages are not enabled by default. The iDashboards administrator and each user partake in setting up each user enrollment in receiving SMS alerts.

Not all users subscribe to SMS or may incur a cost for all SMS messages received. Before using Alerts with SMS, review the SMS capabilities and costs of any iDashboards user configured in the system.

When each user enables Mobile notifications, they will see the following warning:

WARNING: By choosing to receive text messages, you may incur additional charges from your mobile phone provider.

iDashboards assumes no responsibility for fees incurred by utilizing the SMS text messaging feature. Any mobile SMS text messaging fees that are incurred will be billed on 'your' individual mobile provider bill.

15.1 Licensing and Enablement

Contact your iDashboards administrator for details on licensing.

- The Alerts feature must be enabled within the iDashboards license.
- The administrator has the option to disable the entire Alerts feature, disable the schedule of an Alert, or disable the schedule of all Alerts.



15.2 Chart Alert Overview

Alerts are functionality within iDashboards to notify you when certain conditions arise in your chart data. Utilizing rules and schedules, you set the conditions to monitor on a chart and Alerts will notify users when those criteria have been met. Alerts also run as a service and therefore can monitor chart data and notify users via email or text whether logged into iDashboards or not.

iDashboards Alerts provides the following capabilities:

- Near/Real-time monitoring of changing data values;
- Highly flexible monitoring criteria, based on user-defined rules;
- Robust scheduling for monitoring by month, day, hour and minute;
- Seamless integration into the dashboard/chart framework;
- Flexible notification options, sending on-screen, email and SMS messages to individuals or groups.

Every alert includes:

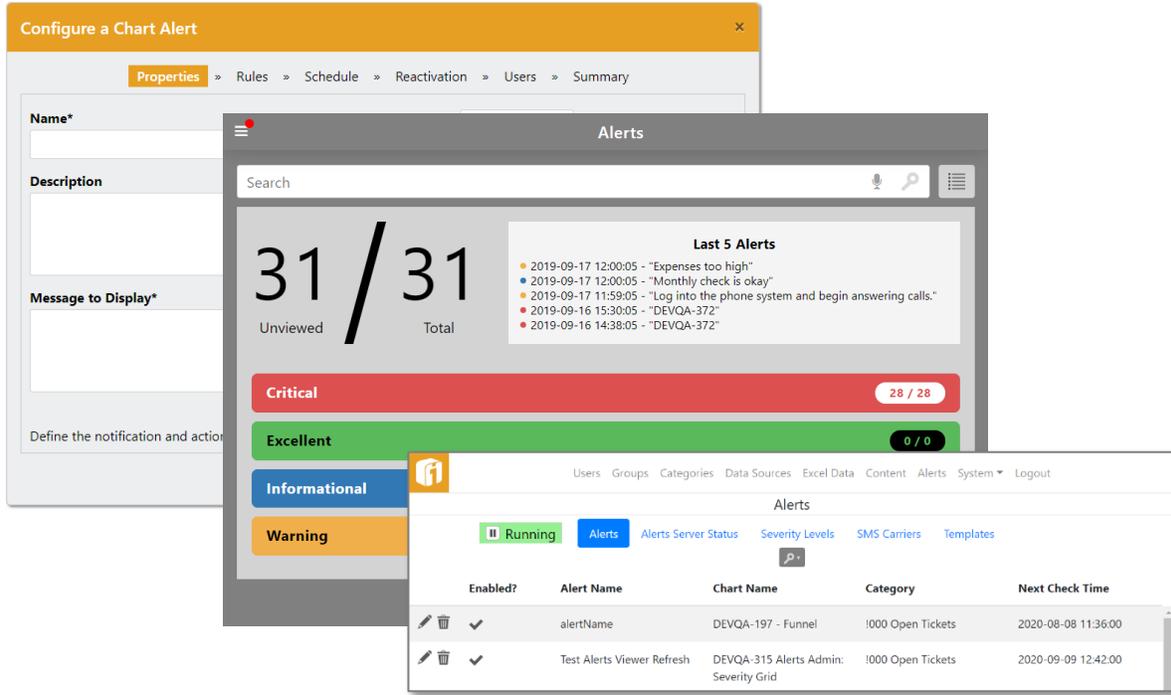
- A **severity** level to categorize the urgency of the alert (ex. Critical, Information, etc.)
- One or more **rules**, describing the data values to watch for;
- A **row threshold**, for defining how many rows in the data set must meet the condition of the rule;
- A **schedule**, describing when to check the data values, and how often;
- A **reactivation schedule** (or time period) in which to resume checking for a condition after an alert has been triggered;
- A list of **groups**, defining the **users** who will be notified when the rules are satisfied;

15.2.1 Alert Components

There are three main parts of the iDashboards Alert feature:

1. **Configure:** Define the rules, properties, schedule and user groups. This is only for the builder role or above.
2. **Alert:** Upon being 'triggered' the alerts can notify via email, text, and on-screen. This is for all user roles.
3. **Administer:** Administration includes the setup of how to send emails and texts, configuring the notification templates, globally identifying all alerts, and troubleshooting.





15.2.2 Configuration Permissions

- Only users with the builder role, or above, can create alerts.
- An alert can only be associated to a single chart. However, a single chart can have one, multiple, or no alerts.
- Alerts can only be created from within the Build Interface.
- Alerts can be created on any chart which a builder role has either “view” or “save” permissions.

Note: Alerts can be configured to charts with Dynamic or Static data sources. Please note, charts with a dynamic data source are far more useful with alerts.

15.2.3 Configuration Restrictions

- Alerts cannot be added to charts within categories where the builder has no permissions.
- Alerts cannot be added to a chart if there are required input parameters without a default value.
 - Similarly, charts with required parameters and no default value cannot be checked.



15.3 Menu Overview

From the Build Interface, Alert configurations can be added/deleted/edited by using two methods:

1. Directly to a chart from the Build home screen
 - a. Select the chart, then select the button “Configure Alerts”
2. From the Dashboard Designer
 - a. Select the chart, then select the toolbar button “Configure Alerts”

Using either method, the Alerts dialog will open, and display the names of all the alerts associated with this chart (if any). From this dialog, you can create a new alert, or edit, or delete an existing alert. Existing alerts will appear with a square color swatch representing the severity level, a circular green symbol representing ‘enabled’, and will be grouped based on being a Private or Group alert.



Private Alerts: These are alerts that when their conditions are satisfied, they notify only you. You can make any alert private. Any chart in your Personal category can have only private alerts.

Deleting Alerts: If you select an existing alert and click the “Delete” button, you’ll be warned that deleted alerts cannot be retrieved. If you choose to continue, the selected alert will be deleted.

Edit Alerts: Double-click (or single-click and then click “Edit”) on an alert to open the Configure a Chart Alert screen.

New Alert: When you click on the “New” button, the Configure a Chart Alert dialog opens. This is where you set up an alert to meet your requirements, and is described in the following sections.



15.4 Configure a Chart Alert

The Configure a Chart Alert dialog box gives you access to all aspects of a single alert. There are click-able stages used to navigate through the alert building process and the last stage, 'Summary' will always provide a textual description of the alert's current definition.

*Note: There are minimum requires for defining an Alert. Not meeting the criteria will result in **red** highlights to help identify the issue.*

Note: If the chart resides in the Personal category, the 'Users' stage will not be available. The alert can only be configured as a private alert.

15.4.1 Properties

The 'Properties' stage has places for the alert's name, description and a few other details. They are:

- **Name** (required): Up to 100 characters. This is the name by which the alert appears in the Alerts dialog.
- **Description**: Up to 255 characters. This is a brief description of the alert, for your convenience.
- **Message to Display** (required): Up to 500 characters. This is the message that will appear, when the alert's conditions are satisfied, in both the on-screen and email notifications.

Note: When the Knowledge Base feature is enabled, within the iDashboards license, there will be a  icon. Selecting it assists in inserting, into the message, the text for a link to a specific article.

- **Severity**: This list provides a method for classifying the alert. Each severity level is associated with a color. By default, Critical is associated with red, Warning is associated



with yellow, Informational is associated with blue, and Excellent is associated with Green but your iDashboards Administrator can add/remove levels and fully customize the color. Severity colors display as part of the chart notification and help to visually categorize different types of alerts.

- **Action:** Describes what happens when the alert's conditions are satisfied. On-Screen alerting is always available.
 - **Send an Email:** In addition to the screen notification, the Alert can be configured to also send an email. Any email messages are sent to the user defined in the User Settings dialog.
 - **Send a text message (SMS):** In addition to the screen notification, the Alert can be configured to also send an SMS text message. Any text messages are sent to the user defined in the User Settings dialog.
 - **Temporarily disable this alert:** When this option is selected, iDashboards stops checking the alert's conditions. Any existing notification messages from this alert are unaffected. If you deselect this option, iDashboards resumes checking the alert's conditions according to its schedule.

15.4.2 Rules

Rules describe the conditions that trigger the alert, and cause it to notify you that they have been satisfied. Rules consists of several component:

- **Expression:** The mathematical condition to analyze against the data set of the chart. When this condition is met, an alert will be triggered.
- **Row Threshold:** The number of rows within the data set that must meet the condition of the expression
- **Variables:** All charts are built using columns (axes). Variables are a way to reference the name (and syntax) of each column. Variables can be used in both the Expression and the Row Threshold.

15.4.2.1 Expression Overview

When the alert is checked, the rule compares the value in the selected column to the specified value provided within the expression. This dialog uses a JavaScript expression to determine a Boolean TRUE or FALSE outcome when each record of the charts data set is analyzed.

Note: Expressions are run against each record of data and must be entirely satisfied within that record of data.

Expression components:

- **Variable:** The axis/column name, or the value to compare
- **Comparison Operator:** This controls the comparison between the column value and the specified value. The choices are Equal (==), Not Equal (!=), Greater Than (>), Less Than (<), Greater Than or Equal (>=), and Less Than or Equal (<=).



- Logical Operators:** There are three logical operators in Javascript, && (AND), || (OR), and ! (NOT). These are used in addition to a comparison operator.

Assuming x = 5 & y = 4

Operator	Description	Example
==	Is equal to	x=8 is false
!=	Is not equal to	x<>8 is true
>	Is greater than	x>8 is false
<	Is less than	x<8 is true
>=	Is greater than or equal to	x>=8 is false
<=	Is less than or equal to	x<=8 is true
&&	AND	(x < 8 && y > 1) is true
	OR	(x == 5 y == 5) is false
!	NOT	!(x == y) is true

The following section provides JavaScript examples based on the following chart using the following data set:



City	State	Sales
Los Angeles	CA	4282208
Modesto	CA	2470186
Oakland	CA	6760205
Sacramento	CA	6370753
San Diego	CA	2035491
San Francisco	CA	5526442
Denver	CO	138792
Boulder	CO	89841
Fort Collins	CO	134465
Grand Junction	CO	248351
Englewood	CO	81726
Lafayette	CO	68702
Belleville	IL	593629
Bloomington	IL	1421385
Chicago	IL	1381381
Joliet	IL	1194179
Rockford	IL	782143
Springfield	IL	570421
Ann Arbor	MI	2238605

City	State	Sales
Midland	MI	1315891
Traverse City	MI	2609111
Deluth	MN	295975
Eagan	MN	192835
Minneapolis	MN	187543
Rochester	MN	292374
St. Cloud	MN	254584
St. Paul	MN	184154
Albany	NY	2649662
Buffalo	NY	2253849
New York City	NY	2632417
Rochester	NY	2547807
Syracuse	NY	3515153
Utica	NY	3344057
Austin	TX	132243
Dallas	TX	87885
Fort Worth	TX	117291
Houston	TX	119071
Plano	TX	83890



Detroit	MI	2264710
Grand Rapids	MI	2933387
Lansing	MI	1079135

San Antonio	TX	55598

15.4.2.2 Expression Examples

Alert Description	Alert Expression
If any city has sales greater than \$3,000,000	<code>\$val["Sales"] > 3000000</code>
If any city in Michigan has sales greater than \$3,000,000	<code>\$val["Sales"] > 3000000 && \$val["State"] == "MI"</code>
If sales are greater than \$3,000,000 in Michigan OR if sales are greater than \$4,000,000 in California	<code>(\$val["Sales"] > 3000000 && \$val["State"] == "MI") (\$val["Sales"] > 4000000 && \$val["State"] == "CA")</code>

15.4.2.3 Row Threshold

A Row Threshold is the number of rows within the data set that must meet the condition of the expression, prior to triggering the alert. The default number is “1”, meaning, upon a 200 row data set, the first occurrence to result in a TRUE will be enough to trigger the alert. Whereas, a threshold value of “3” indicates 3-rows of data must result in TRUE to trigger the alert. Additionally, an expression can be used to define the row threshold – possibly for circumstances where the variable “\$numRows” is used in the equation as shown below:

$$\text{\$numRows} * 0.1$$

This means, 10% of all the rows must satisfy the expression-based rule.

15.4.2.4 Expression Notes

An alert can have as many rules as you wish to create. Each time the alert is checked, iDashboards evaluates all of its rules to determine if the alert will be triggered. Between any two adjacent rules is a relation, either “AND” or “OR”. When iDashboards checks an alert, it evaluates all the rules, and then combines the results logically, according to the relations, to arrive at a result.

The “AND” relation takes precedence over the “OR” relation. You can see this, and how rules are logically combined, by looking at the example below.

Alert Description	Alert Expression
If sales are greater than \$3,000,000 in Michigan AND if sales are greater than \$4,000,000 in California	<code>(\$val["Sales"] > 3000000 && \$val["State"] == "MI") && (\$val["Sales"] > 4000000 && \$val["State"] == "CA")</code>



Result:	Always = FALSE
<p>When iDashboards checks this alert, it evaluates all four rules. Then it combines the results of the first two rules with a logical “AND”, and does the same with the results of the last two. Since “MI” and “CA” are only found within the column “State”, it would not be possible to get a match.</p>	

Alert Description	Alert Expression
If sales are greater than \$3,000,000 in Michigan AND the state can also be California OR the sales can also be greater than \$4,000,000	<pre>\$val["State"] == "MI" && \$val["Sales"] > 3000000 && (\$val["State"] == "CA" \$val["Sales"] > 4000000)</pre>
Simplifies as:	<pre>\$val["State"] == "MI" && \$val["Sales"] > 4000000</pre>
<p>At first, the condition is just “MI” & “3000000”. Using another AND operator, two additional values will also be acceptable: “CA” and “4000000”. Because the expression was already looking for “MI” & “3000000”, adding State=“CA” would always result in FALSE. However, the OR condition between “CA” and “4000000” means that we could actually find a TRUE result if we consider the simplified expression.</p>	

15.4.3 Schedule

The schedule determines how often the rules should be calculated against the chart data. Schedule frequencies range from checking every 1-minute to once per year. Related to the Schedule is the Reactivation, which controls how soon iDashboards resumes checking the alert after it’s been triggered.

Note: A private alert - one that notifies only the user who created the alert - can be checked at most once per hour. Therefore, when selecting the “Minutes” value, only one value can be selected.

Below, is the scheduling interface, representing a calendar year. On the left, navigate to each unit to select a time value from the right. Each section begins with no selected timeframe until the user changes a setting. You can select as many boxes as you like in each section but in order for the schedule to be valid, at least one box in each timeframe must be selected.

Selection Methods:

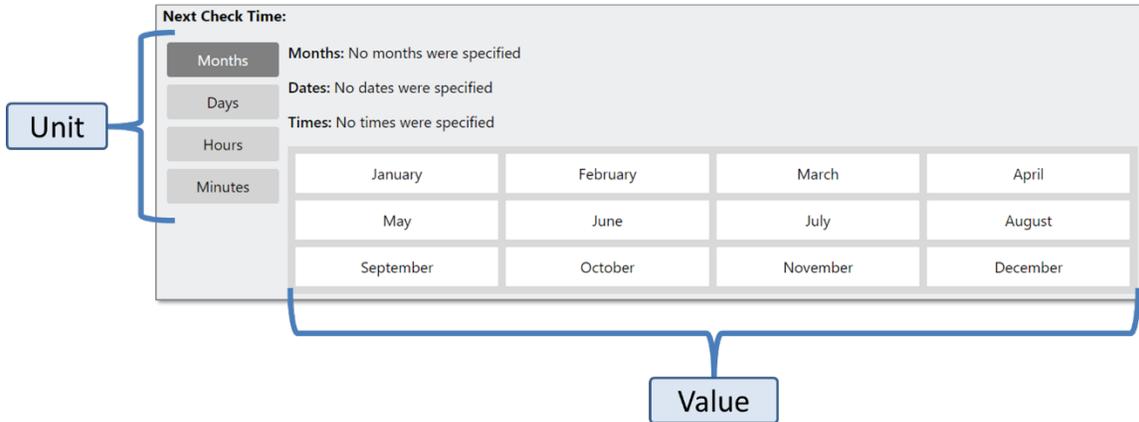
- Left-Click single value
- CTRL-Click multiple unique values
- SHIFT-Click multiple values in a row



- Click-and-drag multiple values

UnSelect:

- CTRL-Click on a selected value



Note: Days of the week and days of the month work together in selecting the days in which an alert is checked. It will be checked on any day that satisfies either section.

15.4.4 Reactivation

After an alert is triggered, the alert is automatically, and immediately, disabled. However, by using the default settings, the alert is reactivated immediately and will resume checking on the previously defined schedule. When it is necessary to frequently monitor the rules of an alert, but not desirable to receive excessive notifications for each timeframe thereafter, a reactivation timeframe can be adjusted.

There are only a few methods for reactivating the alert. Only through the use of reactivating the alert will it be possible for the alert to resume its checking schedule.

- **Automatically reactivate:** Using the alert reactivation schedule
- **Manually reactivate:** Modify and save the alert

Note: This section will focus on the automatic reactivation schedule.

There are three techniques for identifying the reactivation timeline:

- **Immediately** (default): Based on the alert schedule, this is the most aggressive method to keep receiving alerts every minute if the alert is continuously triggered.
- **After a fixed period of time:** While this method can be similarly aggressive compared to “Immediately”. This option allows for a waiting period spanning from every 1-minute to every 31-days.
- **According to a more complex timeframe:** This option uses a similar interface as the schedule definition. The timeframe is based on a calendar year, and multiple timeframes can be specified.



15.4.5 Users

The 'Users' stage is used to configure who will be notified when an alert is triggered. The options included:

- **Only Me (private alert):** When selected, the schedule will be restricted to checking no more than once per hour. Therefore, if multiple minutes had been previously selected, all values will be deselected except for the lowest interval.
- **All groups with access to this chart's category:** This option will leverage the group permissions (defined by the administrator). This will allow an alert notification to be sent to anyone who has "View" or "Save" access to the category of the chart (by means of using group permissions). The right side of the dialog will list each group with access to the category where the chart has been saved.
- **Only groups selected at the right:** This option will leverage the group permissions (defined by the administrator). This will allow an alert notification to be sent to specific groups of users who have "View" or "Save" access to the category of the chart (by means of using group permissions). The right side of the dialog will list each group with access to the category where the chart has been saved, then by using CTRL+Click, groups can be selected and deselected.

15.4.6 Summary

The 'Summary' stage simply contains a textual description of all aspects of the alert, as defined by all previous stages of the configuration. Its purpose is for reviewing all elements of an alert configuration.

15.5 Private Alerts

These are the methods for creating a private alert:

1. **Chart Configuration:** Configure the 'Users' stage to "Only me (private alert)".
2. **Personal Category:** Charts in a user's Personal category can only have private alerts. When configuring an alert on these chart's, the 'Users' stage will not appear on the configuration dialog.
3. **Filter on User:** When a chart is configured with the Filter on User feature (refer to section 10.9, "Filter on User"). When configuring an alert on these chart's, the 'Users' stage will not appear on the configuration dialog.



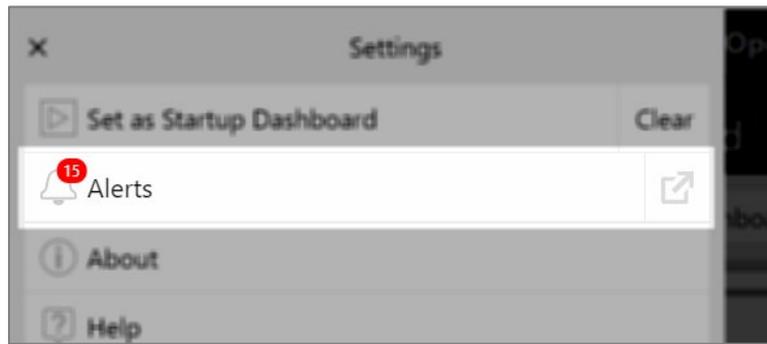
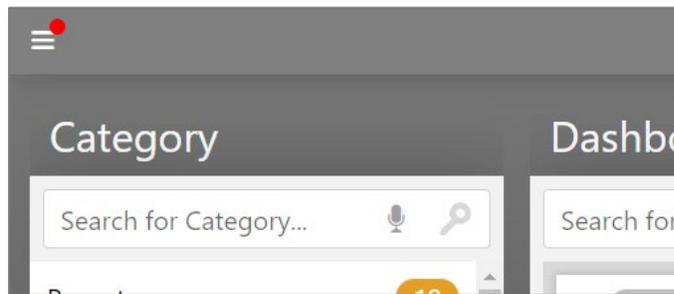
15.6 Alert Notifications

Alerts run as a service and therefore can monitor chart data and notify users via email or text whether logged into iDashboards or not. Here are the notification options with Alerts:

15.6.1 User Alert Notification Options

These notifications come from a chart when there is a change in data.

- **On-Screen Notification**
 - Requires no setup
 - If you are logged into iDashboards, evidence of a new alert will appear as red notifications within the Lobby, Builder and View Interface.



- **Email**
 - Requires administrative setup (See the Admin Manual for more information)
 - Uses a template, configured by the iDashboards administrator
 - Requires User Settings setup
- **Text message (SMS)**
 - Requires administrative setup (See the Admin Manual for more information)
 - Uses a template, configured by the iDashboards administrator
 - Requires User Settings setup

Note: When an alert is triggered, if all methods of notification are configured, then all methods will be used.



15.6.2 Administrator Alert Notification Options

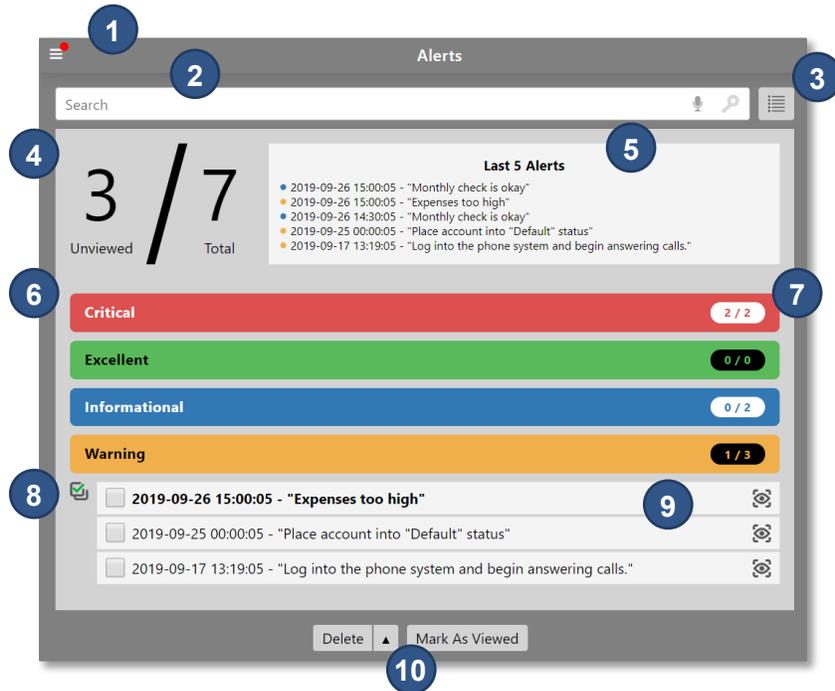
These notifications come from a server error or event based on the Alerts infrastructure (ex. Services were paused, Data Source went offline, etc.) Refer to the Admin Manual for details.

- **Error Notification**
 - Requires administrative setup
 - Uses a template, configured by the iDashboards administrator
- **Event Notification**
 - Requires administrative setup
 - Uses a template, configured by the iDashboards administrator



15.7 Navigating Alert Notifications

Once a logged-in user navigates to the Alerts option in the menu, they will see the alerts viewer page. This page is unique for each iDashboards user account. An alert notification that has been viewed or deleted by one user does not change the status of the same notification for another user. This page is described below:



1. **Menu:** To access the Alerts viewer, first select the Menu button. A red dot here indicates an unviewed notification.
2. **Search Bar:** Search for any part of the Alert message.
3. **Toggle View:** Change the view between “Grouped” and “Table”
 - a. **Grouped:** First, the notifications are grouped and counted, then sorted by datetime
 - b. **Table:** In this view (see image below) the notifications are displayed in a table which can be sorted by: Viewed, Triggered, Severity and Alert
4. **Total Counts:** View the total number of unviewed alerts (on the left) compared to the total number of alerts (on the right).
5. **Last 5 Alerts:** In chronological order, this section will display the last 5 alerts, regardless of severity.
6. **Severity:** Expand these sections to display the related alert notifications.
7. **Severity Counts:** For each severity, view the total number of unviewed alerts (on the left) compared to the total number of alerts (on the right).



8. **Select/Deselect-All:** Manually select multiple notifications, or use the “Select All/None” button, prior to deleting or marking as ‘Viewed’
9. **Notification:** Click a notification to view the chart and other notification details.
10. **Decision Options:** Notifications can be cleaned up from view by either marking them as viewed, or by deleting them entirely. Delete options include:
 - a. **Delete:** Delete the selected notification(s)
 - b. **Delete All:** Delete all notifications, regardless of selection
 - c. **Delete Viewed:** Delete all viewed notifications, regardless of selection
 - d. **Delete All For Alert:** Delete all notifications, for the type of notification that is selected
 - e. **Mark As Viewed:** Notifications will automatically change their status from “Unviewed” to “Viewed” when opened. To set the status to “Viewed” without opening each notification, use this option to change all selected items.

15.7.1 Table View

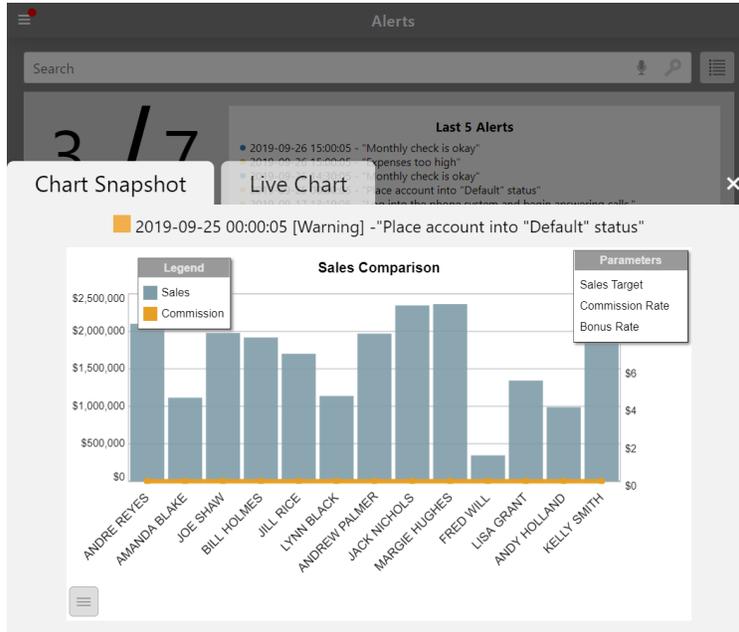
The screenshot displays the Alerts interface. At the top, there is a search bar and a summary showing 3 unviewed alerts out of a total of 7. Below this is a section titled "Last 5 Alerts" listing recent alerts with their timestamps and severities. The main part of the interface is a table with the following data:

Viewed	Triggered	Severity	Alert
<input type="checkbox"/>	2019-09-26 15:00:05	Informational	Monthly check is okay
<input type="checkbox"/> •	2019-09-26 15:00:05	Warning	Expenses too high
<input type="checkbox"/>	2019-09-26 14:30:05	Informational	Monthly check is okay
<input type="checkbox"/>	2019-09-25 00:00:05	Warning	Place account into "Default" status
<input type="checkbox"/>	2019-09-17 13:19:05	Warning	Log into the phone system and begin answering calls.
<input type="checkbox"/> •	2019-09-16 15:30:05	Critical	DEVQA-372
<input type="checkbox"/> •	2019-09-11 08:50:05	Critical	Kmessage

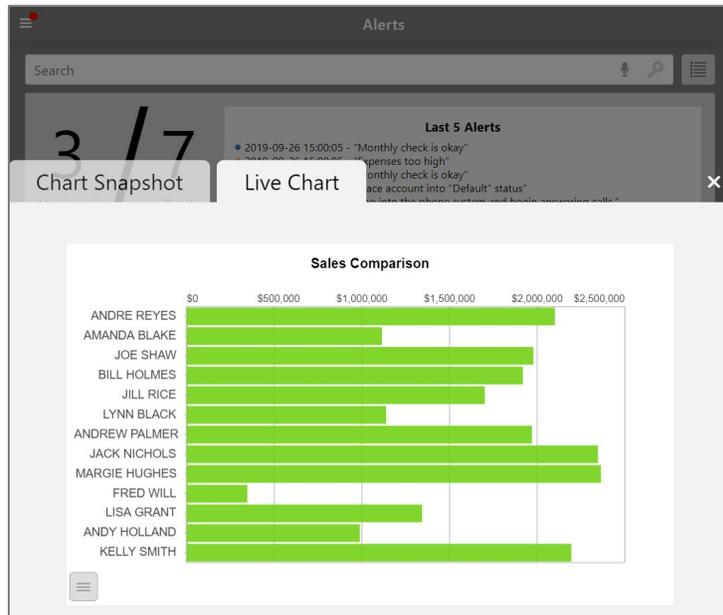
At the bottom of the table, there are two buttons: "Delete" and "Mark As Viewed".

15.8 Viewing Alert Notifications

When a notification is selected, information about the chart will appear. The first tab “**Chart Snapshot**”, shows the exact chart in the state of when the alert was triggered. In addition to the interactive chart, the lower-left menu can reveal finer details about the chart data or chart data diagnostics, as shown below.



The second tab “**Live Chart**”, shows a current version of the same chart. In addition to the interactive chart, the lower-left menu can reveal finer details about the chart data or chart data diagnostics. As shown below, one would likely observe that the vertical “Column with Line” chart from above (the chart when triggered) has since been changed into a bar chart using different colors (the current chart as viewed today).



Note: Some chart features, like drilldown, will be disabled from the notification dialog.

15.9 Charts with Alerts

A chart on which an associated alert is different than a non-alert chart. Since alerts depend on a non-changing data definition within the chart, dialogs - which you could otherwise modify – may display the following message:

This chart has alerts configured on it. Altering the axis names may make the alert condition invalid if the axes are referenced in the expression.

Therefore, take caution when changing the Axis Labels of a chart when there are associated alerts. As for Chart Type and Chart Properties, those remain fully open for edit since the



16.2 Reports Overview

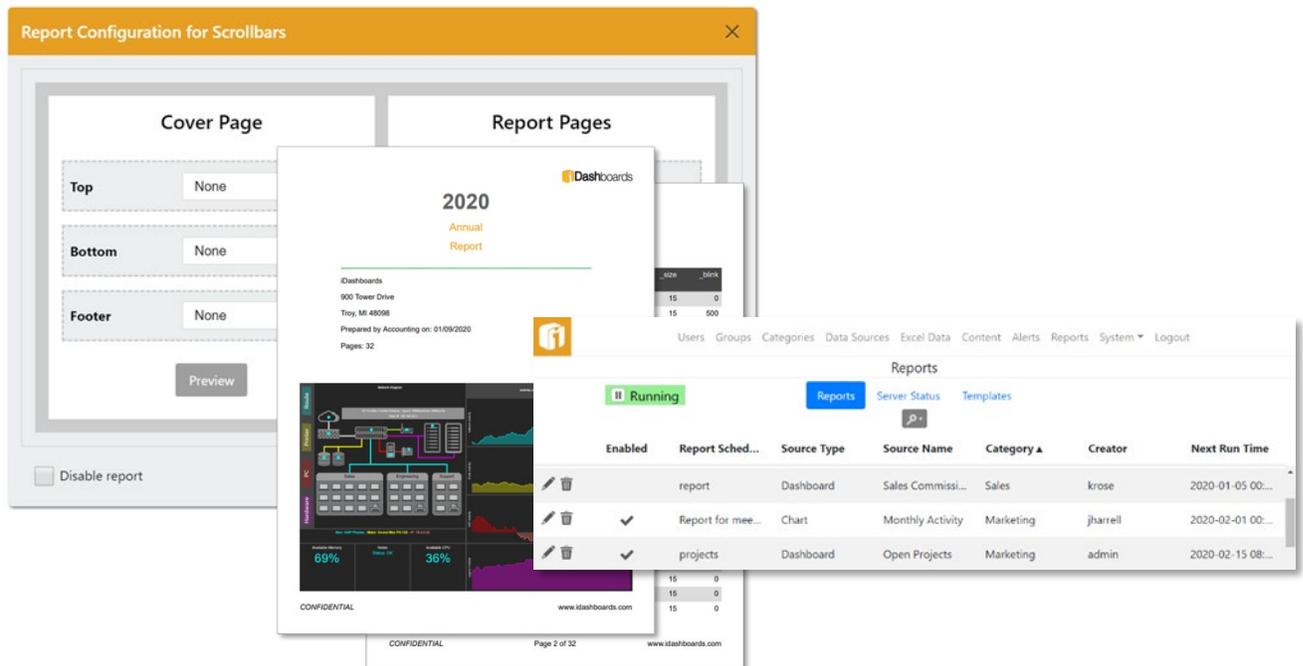
iDashboards Reports provides the following capabilities:

- On-demand report generation from the client interface (desktop or mobile).
- Automated report publishing using a customized schedule.
- Security to determine who is able to generate and receive reports.
- Report generation on a single chart or an entire dashboard.
- Report data can be filtered using the Filter on user
- Customizations to refine the report data and the appearance.

16.2.1 Report Components

There are four main parts of an iDashboards Report:

1. **Designing a report:** Reports are immediately available on every chart and dashboard and can be run without configuration. However, configuration settings are available for each chart and dashboard which specify report properties controlling what information the report should display and how it should look.
2. **Viewing a report:** Running a report on a chart or dashboard will use the configuration settings assigned to it, or reasonable defaults when it has not been configured.
3. **Scheduling a report:** Without logging into iDashboards, a Report can be automatically created and emailed based on a custom schedule of your choice. Scheduled Reports can be for personal use or for groups of iDashboards users.
4. **Administration:** Includes customizable email templates, global list all report schedules, and troubleshooting.



16.2.2 Configuration Permissions

- Only users with the builder role, or above, can customize Reports.
- Report configurations or schedules can only be created from within the Build Interface.

Note: Reports can be configured to charts with Dynamic or Static data sources.

16.2.3 Configuration Restrictions

- Report configurations or schedules are not available for charts within categories where the builder has no permissions.
- Scheduled Reports cannot be processed if there are required input parameters without a default value.

16.3 Generating Reports

Users within the View interface are the ultimate recipients of a report. Dashboard reports are a single PDF containing information on the specified charts within the dashboards report configuration. Chart reports are also a single PDF containing information on that specific chart.

16.3.1 On-Demand Reports

On-demand reports are not published (emailed), but are generated immediately and the report PDF is loaded into a new browser tab. There are multiple ways to generate an on-demand report.

To generate an on-demand report for a dashboard being viewed, use the application menu and select “View Dashboard as Report”. To generate an on-demand report for a specific chart in a dashboard being viewed, use the charts menu and select “View Chart as Report”.

Other on-demand methods occur when charts have been configured using the Drill-to-Chart-Report or Drill-to-Dashboard-Report setting.

Users with only View permissions may run on-demand reports on any dashboard or chart that they can access. Builders may disable this functionality through the report configuration settings.

16.3.2 Scheduled Reports

A scheduled report differs from on-demand reports in the following manner:

- Scheduled reports are sent via email to eligible recipients;
- Scheduled reports can include the report PDF, and/or zip file of chart data;
- Scheduled reports use a customizable email template;
- Scheduled reports are typically triggered by a Report Schedule.

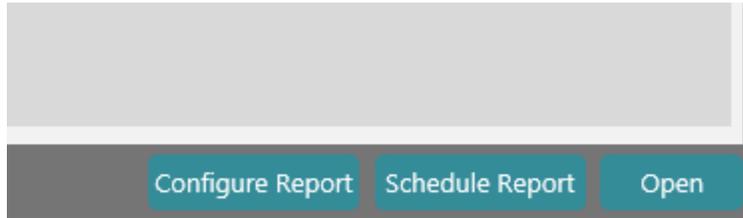
Note: During the configuration of a report schedule, the user can immediately trigger the scheduled report as a testing mechanism. This is not an option for a user with only Viewer privileges



16.4 Accessing Report Configurations and Schedules

From the Build Interface, both Report Configurations and Report Schedules can be accessed by two methods:

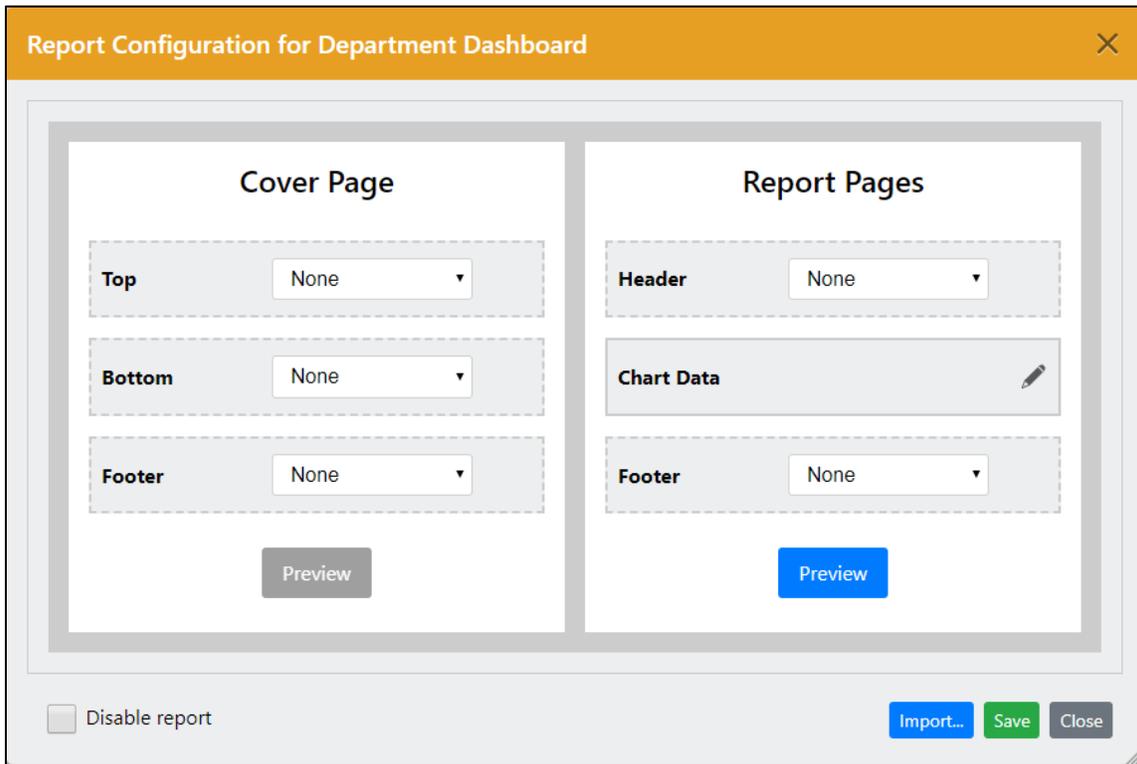
1. Directly from the Build home screen by selecting a dashboard or chart, then pressing the “Configure Report” or the “Schedule Report” button.



2. From the Dashboard Designer
 - a. **Dashboard:** Click the Additional Properties toolbar button, then choose “Configure a Dashboard Report...” or “Schedule a Dashboard Report...” This is only available if there are no unsaved changes.
 - b. **Chart:** Select the desired chart, then click the Chart Reports toolbar button and choose “Configure a Chart Report...” or “Schedule a Chart Report...”.

Using either method, the appropriate Reports dialog will open for the chosen dashboard or chart.

16.5 Configuring a Report



A report consists of an optional cover page followed by a series of report pages. The cover page will only be included in the report if it has been configured. Cover pages are limited to a single page, user content can be checked for overflow using the preview button. The sections of the cover page vary depending on whether the report is for a dashboard or chart – each section provides composition using a rich-text editor which allows macros for dates and page numbering.

From the configuration dialog, report configurations can be imported from another chart or dashboard, as appropriate, using the “Import...” button. Report configurations are embedded in each chart or dashboard and are included during copy (Save As...) as well as the import and export of charts and dashboards through the administration page, “System, Import/Export”.

16.5.1 Cover Pages Sections

- **Chart Reports** – contain *content* and *footer* sections
- **Dashboard Reports** – contain *top*, *bottom* and *footer* sections
 - A dashboard screenshot may appear in either the top or bottom sections and its image may be linked to the dashboard by selecting “Screenshot” from the dropdown and clicking the edit (pencil) icon and selecting “Link to dashboard”



16.5.2 Headers and Footers

Options are provided to create one, two or three column headers and footers. Cover page footers and report page footers can be shared by selecting the “Same as Header/Footer” options.

16.5.3 Chart Report Pages

Editing the Report Columns Section of a chart report provides control over the report summary (totals) along with the output widths and aggregation functions to be used in generating the charts' report.

Note: Dashboard reports use their own configuration for each of their charts and do not rely on the individual charts configurations. A charts particular configuration only applies when an individual Chart Report is run.

16.5.4 Dashboard Report Pages

Editing the Chart Data Section of a dashboard report provides control over which charts are to be included in the dashboards' report and offers configuration of each individual charts' report summary (totals) along with the column output widths and aggregation functions to be used in generating that charts' section of the dashboards' report.



16.6 Scheduling a Report

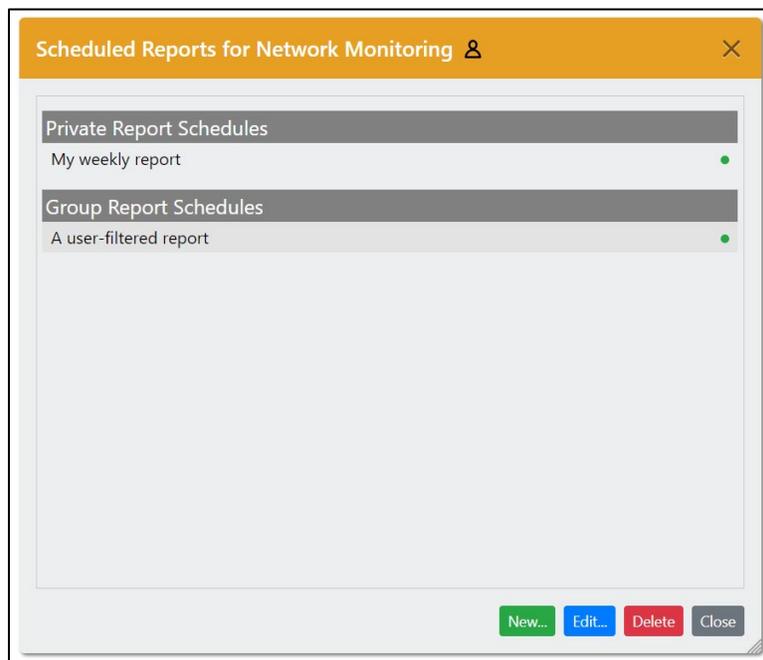
Each dashboard or chart contains configuration for one report, and that report can be generated on-demand, or by scheduling it to run at specified intervals or dates. A single chart or dashboard report can have many schedules.

When a scheduled report is run or generated, eligible recipients will receive the report via email which can contain attachments of the report PDF, a zip file containing the report data, or both. Email templates are available for customization through the Administration Interface, under Reports, Templates. These “published” email reports can also be run at any time manually via the report scheduling dialog.

Report schedules are embedded in the dashboard/report, but do not transfer when copying (Save as...) and will not persist through the Administration, Import/Export of a dashboard or chart.

16.6.1 Managing Scheduled Reports

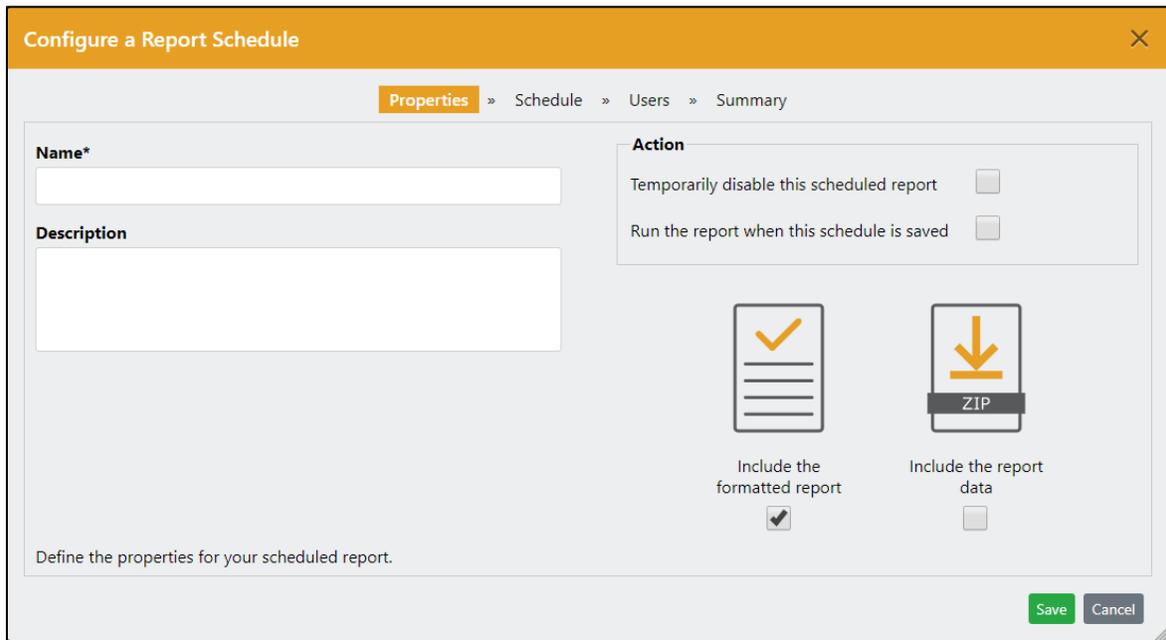
As each chart or dashboard may have many scheduled reports, selecting “Schedule Report” will provide a list of existing report schedules and allow creation, deletion and editing of scheduled reports for the selected chart or dashboard.



The scheduled reports list contains two sections:

- **Private Report Schedules** – schedules that will specifically email the report only to the current user.
- **Group Report Schedules** – all other schedules; those emailing the report through specified groups or all groups with access to the dashboard/charts category.

16.6.2 Schedule Properties

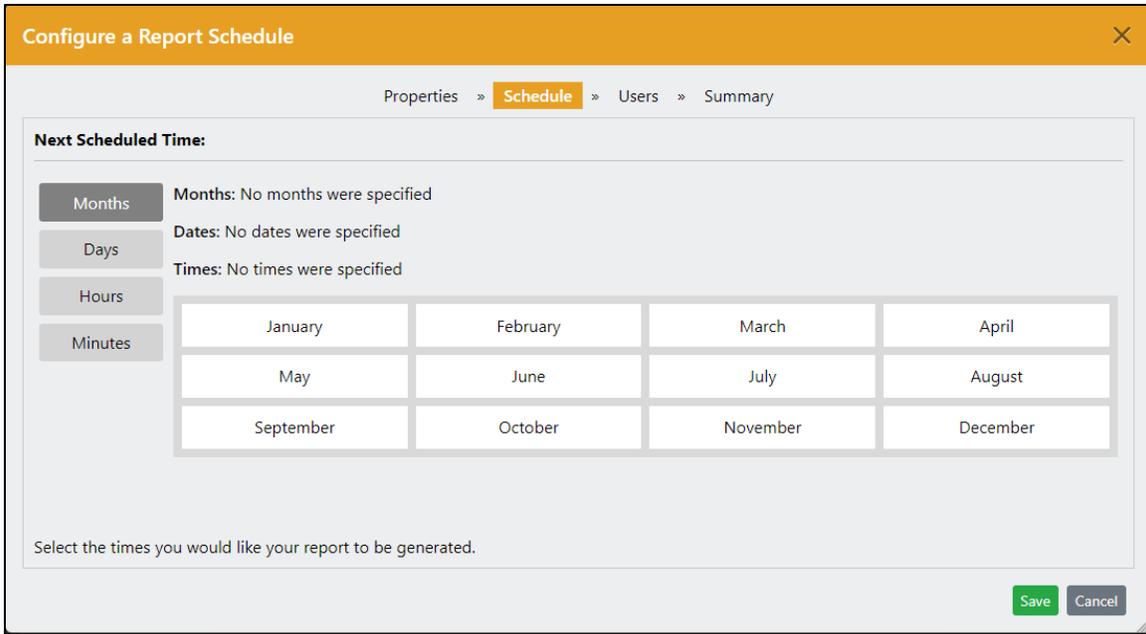


- **Name** – *required*, a unique schedule name for this report
- **Description** – an optional description for the schedule
- **Action**
 - **Temporarily disable this scheduled report** – do not allow this schedule to trigger the report
 - **Run the report when this schedule is saved** – run the report when dialog is saved
- **Include the formatted report** – include the PDF in the email
- **Include the report data** – include a zip file of the data in the email

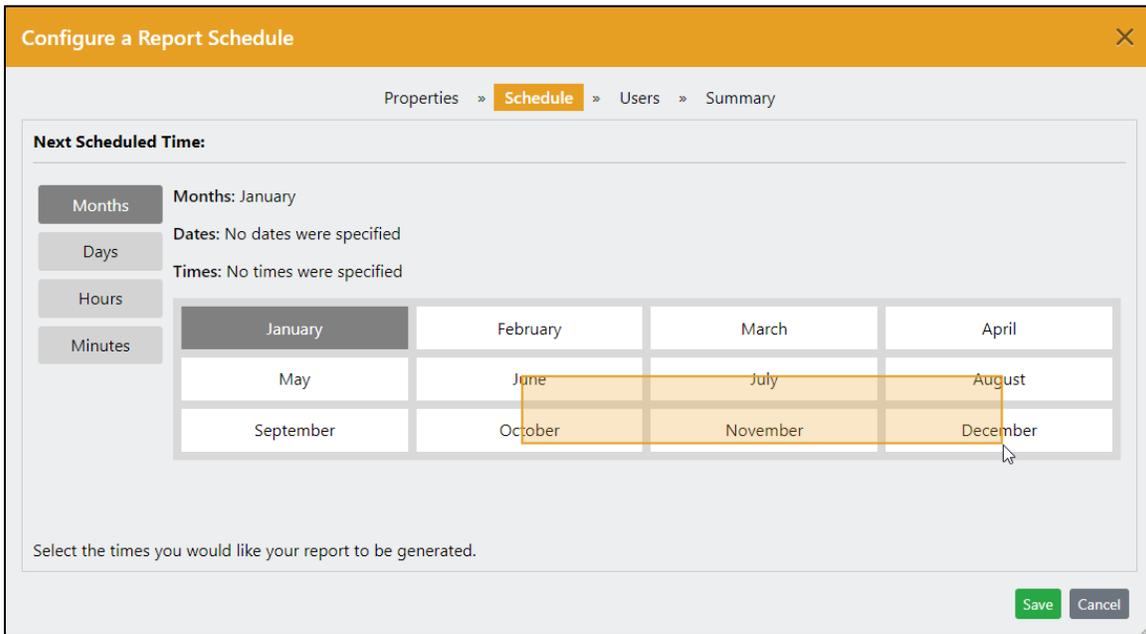
16.6.3 Report Scheduling

The schedule determines how often the report will be published (emailed to the eligible recipients). Schedule frequencies range from publishing every minute to once per year.





Use CTRL+Click to select multiple items, individually or while holding down click draw a box over the items and release.



Note: A report schedule that is private - one that specifically publishes only to the user who created the schedule – is restricted to publishing at most once per hour. Therefore, when selecting the “Minutes” value, only one value can be selected.

16.6.4 Report Recipients

The 'Users' stage is used to configure who will be receive the report at the scheduled time. The options included:

- **Only me (private report):** When selected, the report schedule will be restricted to publishing no more than once per hour. Therefore, if multiple minutes had been previously selected, all values will be deselected except for the lowest interval.
- **All groups with access to this chart's/dashboard's category:** This option will leverage the group permissions (defined by the administrator). This will allow a report to be sent to anyone who has "View" or "Save" access to the category of the chart/dashboard (by means of using group permissions). The right side of the dialog will list each group with access to the category where the chart/dashboard has been saved.
- **Only groups selected at the right:** This option will leverage the group permissions (defined by the administrator). This will allow the report to be sent to specific groups of users who have "View" or "Save" access to the category of the chart/dashboard (by means of using group permissions). The right side of the dialog will list each group with access to the category where the chart/dashboard has been saved, then by using CTRL+Click, groups can be selected and deselected.



16.7 Reports via Filter on User

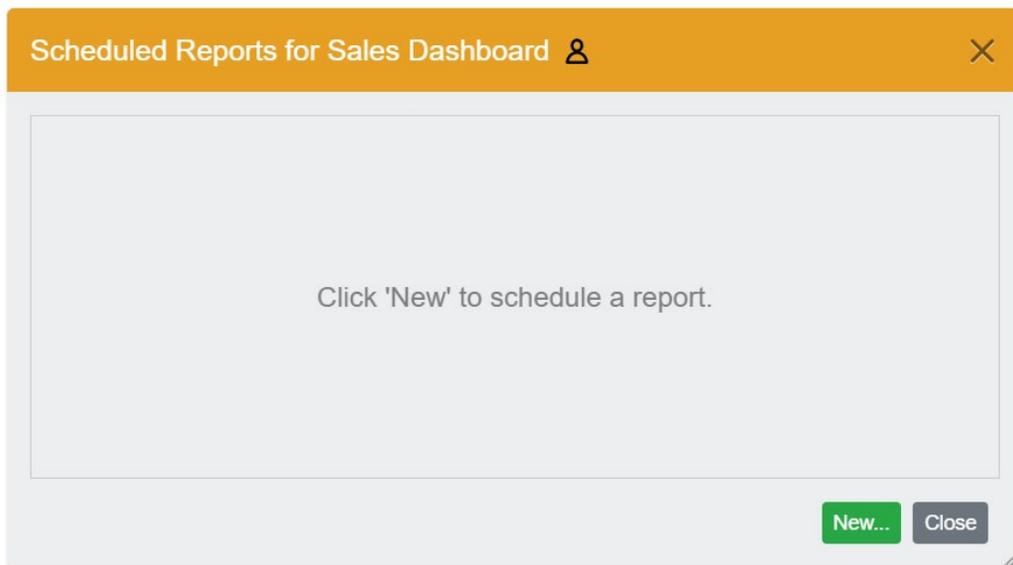
Refer to sections 10.6, “Using Filter Macros” and 10.9, “Filter on User” and the Administrator manual for information on the Filter on User feature.

If a dashboard contains a chart that is somehow being filtered by the current user login, then the report will honor the filtered data and the user will receive a custom report. For on-demand reports, this behavior seems reasonable and the performance between the server and client is based on generating a single, custom report. However, for scheduled reports, each user included in the schedule will also receive a custom report – potentially including a custom screenshot and custom ZIP file containing the chart data.

CAUTION:

Scheduling a Report on a dashboard containing user-filtered data, can affect the performance of your entire server. This task likely needs to query chart data, render a screenshot, generate the PDF, compile the chart data ZIP file, then send an email containing this entire package. For example, 100 users receiving filtered data reports will require the server to generate 100 custom reports.

When scheduling a report where there is user-filtered data, the titlebar of the scheduling window will contain a user icon. Mouse-over the icon to learn “The sent reports will be individualized for each recipient.”



Additionally, when defining the “Users”, the same icon will appear with another note stating “The sent reports will be individualized for each recipient.”

17. Data Forms

Everything about iDashboards charts, alerts, reports and more, has been designed to consume existing data. If data exists, a chart can be built, an alert can be triggered, a report can be processed, etc. Data Forms is the only feature of a dashboard that has the ability to create data using a structured mechanism to collect and store that data within a database table.

Data Forms begin with the ability to create a form using a variety of input controls (text, number, date, etc.) and then place the form within a dashboard. Each form must be configured to save data into a single database table, which could then be used to build charts or for further processed via the Data Hub.

Additionally, data entered into the form can be search, edited or deleted if the form permissions are customized.

17.1 Licensing and Enablement

Contact your iDashboards administrator for details on licensing.

- The Forms feature must be enabled within the iDashboards license.
- The administrator has the option to disable the entire Forms feature.

17.2 Requirements

Each data form will be related to a single database table – this is a 1:1 relationship. Additional system requirements are necessary to use data forms. **Please refer to the iDashboards Administrative Manual for details.**

- **Databases and user privileges**
 - Supported databases include: SQL Server (2005 or later), Oracle (9i or later), MySQL (5.0.3 or later) and PostgreSQL (9 or later). These types of databases are required for the storage of form data.
 - A database user with read/write privileges to at least one data store database.
 - A database user with write, create table, update table, and drop table privileges to the data form database(s).
 - You can combine all privileges into one user, or use one user for read access and another user for write, create table, and drop table privileges.
- **Data Source**
 - A minimum of one data source, configured as a data store. This is where the auto-generated form tables will be automatically created.

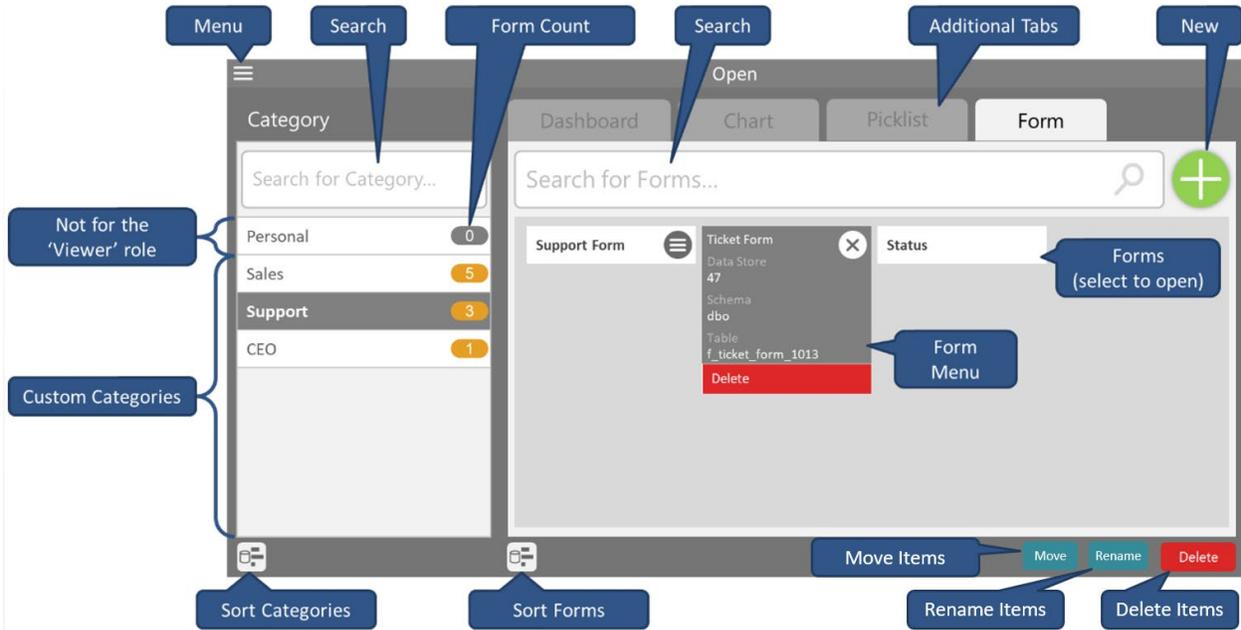


17.3 Creating or Editing a Form

A form can be created or edited directly from the Home screen in a manner similar to that of creating or editing a chart.

To create a new form, make sure the “Form” tab is selected on the Home screen, then select the “New” button as seen in the image below.

To edit an existing form, make sure the “Form” tab is selected on the Home screen. Then, using manual navigation or searching, locate the form to open.



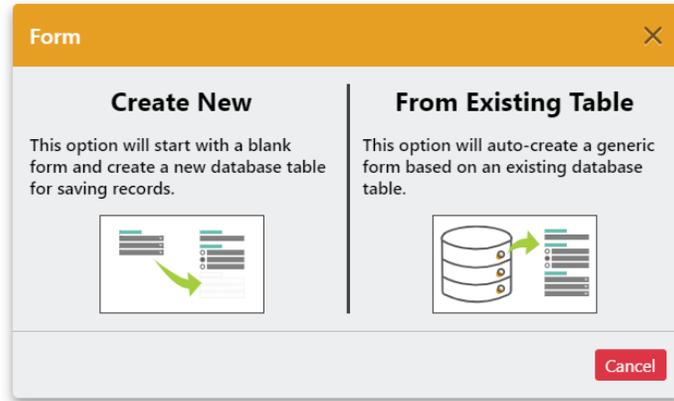
17.4 Form Types

There are two types of forms:

- **Create New** – This option will start with a blank form and create a new database table for saving records.
- **From Existing Table** – This option will auto-create a generic form based on an existing database table.

17.5 New Form

From the builder’s Form tab, selecting the “New” button will ask for the form type (as shown below).



17.5.1 “Create New” Form

This option will start with a blank form and create a new database table for saving records. These forms can be thought of as “**Auto**” forms because the table will always automatically adjust to whatever you do within the form designer. Other characteristics of this form type include:

- The form can always be changed to add or remove fields.
- Table names are auto-created using the syntax:
 - “f_” + <form name> + “_” + <form ID>
 - Ex: “f_survey_results_1035”
- Some changes to an existing form may result in the entire column, and all of its data, being dropped.
- Some changes to an existing form may result in the entire table, and all of its data, being dropped.

The first step for this type of form is selecting where the form’s database table will be created. This is similar to the process and defining a data set, where you select the data store and schema. When the form is first saved, the form’s table will be auto-created at this location in the database.

After Data Store information is defined the Form Designer will open.

17.5.2 “From Existing Table” Form

This option will auto-create a generic form based on an existing database table. These forms can be thought of as “**Static**” forms because the table will never adjust based on whatever you do within the form designer. Other characteristics of this form type include:

- The form can **ONLY** display fields if it can be mapped to an existing table column. Therefore, if a new field needs to be added, and there isn’t a data column available, then the database administrator must first add the column to the table.
- The form does not have to use all columns from the existing table.

The first step for this type of form is selecting where the database table that the form will connect to. The Form Designer will open on the Data Store definition, where you select that table.

17.6 Update Form

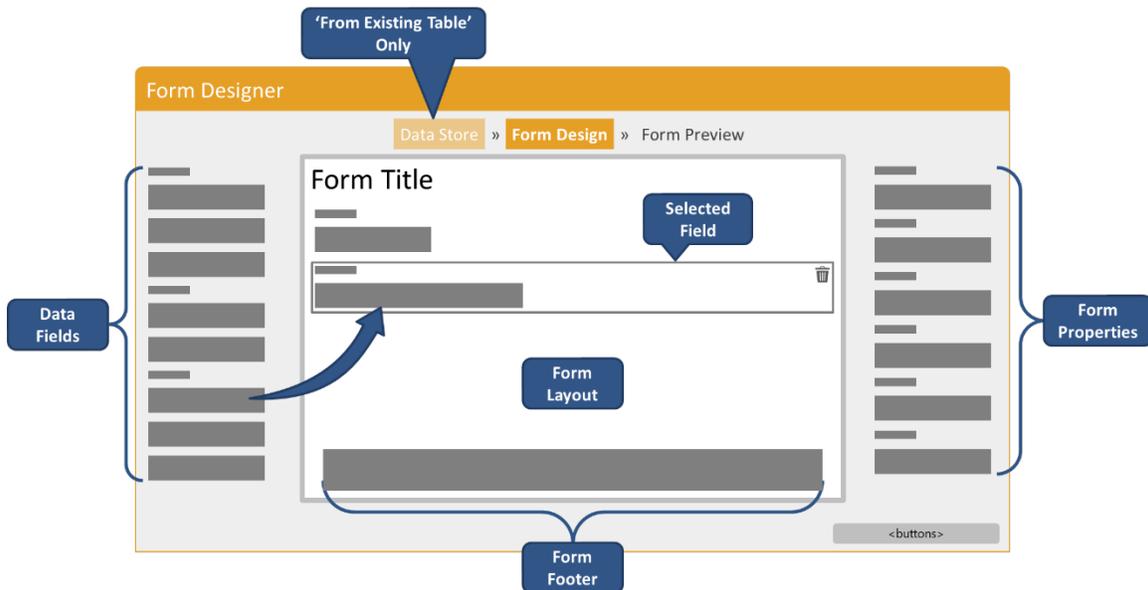
When opening a form, to edit it, the Form Designer opens to the Form Design, showing the current layout with the form's properties.

Changes to the forms layout and properties can be made and saved back to the same form, although some changes to a 'Create New' (Auto) form may result in all current data in the column or in the table maybe be lost.

Other options are to use Save-As to create a 'Copy of' the form, or change the form's Category.

17.7 Form Designer

The Form Designer consists of 5 sections: Form Layout, Form Footer Form Properties, Data Fields and Field Properties.



17.7.1 Form Layout

This is where the Form is put together. Click on a data type field or drag the field onto the form layout.

For a "From Existing Table" form, the Auto-Field button can be used to select some or all of the existing table's fields.

To remove a field from the layout select its  icon.

17.7.2 Form Footer

This is the bottom of the form, where control buttons are automatically placed. Except for the Clear button, they are only place holders when on Form Design. Here the Clear button will clear

the form when testing validation, and helpful for Regex Pattern testing. Also when on Form Design only, the **Form Properties** button, will also be found here.

17.7.3 Form Properties

The form's properties consist 5 property types. If they are not currently showing on the Form Design, use the **Form Properties** button, found on the right side of the form footer.

17.7.3.1 Form Definition

- **Category:** Like dashboards, charts and picklist, used to help organize forms, within the iDashboards environment, similar to file folders organize documents within a computer environment.
- **Name:** Maximum size is 100 Characters. "Create New" forms use the form name in the database table name, using the syntax: "f_" + <form name> + "_" + <form ID>. Ex: "f_survey_results_1035".
- **Title:** Maximum size is 100 Characters. This is displayed, left justified, on the top line of the form.
- **Description:** Maximum size is 500 Characters.

17.7.3.2 Database Setup

- **Data Store:** An iDashboards Data Source configured as a Data Store.
- **Schema:** The Data Store's database schema where the table is located.
- **Table:** The Data Store's database table name.

17.7.3.3 Form Layout

- **Flow Down:** Fields are laid out vertically.
- **Flow Across:** Fields are laid out horizontally.
- **Grid:** Fields are a grid-based layout
- **Numbered Fields:** The field labels will be prefixed with a number indicating their order on the form.

17.7.3.4 Record Control Permissions (*Edit, Delete, View*)

- **None:** No control permissions
- **Own Records:** Only records the current user created. In order for the permission "Own Records" to function properly, this form must have a field with Auto-Populate set to "Create User ID" or "Create Username".
- **Any Record:** Able to control any record
- **By Group:** Set control permissions at the group level. When selected, or when selecting the  icon, an interface opens allowing the setting of permissions individually for each group. To determine what the effective permissions are, the permissions from



all the groups, that the user belongs to, are looked at, and the maximum of all of all of those permissions is what is used.

17.7.3.5 Search Case Sensitivity

- **Enable Choice:** Triggers a dropdowns with two options, Case-Insensitive and Case-Sensitive.
- **Database Default:** This will hide the dropdowns on the search form, and all searches will use the normal SQL syntax for queries.
- **Force Case Insensitivity:** This will hide the dropdowns on the search form, and all searches will use the UPPER function to make searches case-insensitive. This will often be desirable, because in most cases, you will not want searches to be case-sensitive.

17.7.4 Data Fields

Data fields, and field groups, can be added to the form by the following options:

- **Drag-n-Drop** – This option allows the new field to be placed into the desired position.
- **Selection** – This option will add the new field to the bottom of the form. However, if an existing field is selected within the layout, then the new field will be placed after the selected field.

Note: Repositioning fields can be achieved by using drag-n-drop.

The following fields are available:

- **Text** A single line of characters with length control.
- **Text Area** A multi-line resizable character box with length control.
- **Email Address** A properly formatted email address.
- **Dropdown** A predefined list of values, either static or from a picklist.
- **Radio Button List** A predefined list of values, either static or from a picklist.
- **Checkbox** Selected is TRUE. Not selected is FALSE.
- **Number** A numeric value with range and decimal place control.
- **Date** A date value entered or selected from a calendar.
- **Date and Time** A date and time value entered or selected from a calendar.

17.7.5 Data Field Groups

Field groups are a special type of component which is placed onto a form in the same manner as a regular data field. The difference is that a field group allows for a clustered set of fields to appear as a grouped section within the form. As a grouped section, these fields can have a different layout and can moved/deleted in a single effort. Once a field group is placed, the fields within the group can be further customized or even removed.

The following field groups are available:



- **Address** - A collection of fields which are generally used for collecting address information. Field types include Text and Number, of varying sizes and utilizing a grid layout.
- **Checkbox List** – A special section where the form builder can add any number of Checkbox fields
- **Multiple Text** – A special section where the form builder can add any number of Text fields
- **Metadata** – A collection of fields (Hidden by default) which are generally used for collecting available metadata information when submitting form values.

17.7.6 Field Properties

Each field will have properties specific to it.

- **Column Name:** This is the name of the field in the database table, and is required. The default is to generate the column name automatically, or this can be turned off and manually provided.
- **Label:** This is the name of the field on the form, and is required. Macros are available to provide dynamic values.
- **Help:** When hovering over the  icon, displayed after the field label, the help text is displayed. If there is no help text defined, then there is no  icon.
- **Field Size:** Small, Medium, Large, Full choice are available. The size is relative to the form layout and the sizes of other fields on the form.
- **Default Value:** A value to automatically use in the field, but can be changed. If dropdown, radio button, or checkbox are used, the value must be in that predefined list of values. Macros are available to provide dynamic values.
- **Hidden:** On the Form Design's layout the field will have an  icon is displayed after the field's label. The field will not be visible on the form when viewed.
- **Required:** Identified with a * after field's label, signifying that a value must be provided.
- **Primary Key:** Makes the field a unique identifier in the database table. More than one field can be part of the Primary Key, collectively identifying a unique record in the table. Automatically selected as a required field, because the table must have one primary key.
- **Browser Autocomplete:** Enables the field's ability to utilize the browser's autocomplete feature. When enabled the browser is allowed to automatically complete input into the form's fields using its own judgement.
- **Auto-Populate:** a read-only extrapolated value to automatically use in the field. Not available for dropdown, radio button list, checkbox and email address fields.
 - Available Choices



- **Number Field**
 - **From External System:** A value coming from outside of iDashboards Forms. It can be used to make fields that have system generated values, like a serial number, read-only.
 - **Big Increasing Number:** The current time expressed in milliseconds (since 1/1/70 GMT). This will always increase each time a record is submitted. Allows a value that can be converted into a date and time using an ETL job.
 - **Increment-On-Update:** Generates a sequence of integers auto-incremented for each updated record.
 - **Concurrent Update Detector:** A load-time version of the record. If some other transaction manages to commit a newer record version, the update is prevented.
 - **Update User ID:** The iDashboards User ID for the user who update the record.
 - **Create User ID:** The iDashboards User ID for the user who created the record.
- **Datetime Field**
 - **From External System:** A value coming from outside of iDashboards Forms. Can be used to make fields that have system generated values, like a create timestamp, read-only.
 - **Create Timestamp:** The current date and time the record was created.
 - **Update Timestamp:** The current date and time the record was updated.
- **String Field**
 - **UUID:** A universally unique identifier value. While the probability that a UUID will be duplicated is not zero, it is close enough to zero to be negligible.
 - **From External System:** A value coming from outside of iDashboards Forms. Can be used to make fields that have system generated values, like a UUID, read-only.
 - **Update Username:** The iDashboards User ID for the user who updated the record.
 - **Create Username:** The iDashboards User ID for the user who created the record.



- **Number Range** (*Number field*): Available when Auto-Populate is not used. Defines the allowed numeric value range.
- **Decimal Places** (*Number field*): Available when Auto-Populate is not used. Defines the number's decimal place accuracy. If not provided the values are saved as an integer.
- **Length Range** (*String fields*): Available when Auto-Populate is not used. Defines the maximum number of characters the field will allow. Defines the size of the VARCHAR for the database column.
- **Regex Pattern** (*Text, Text Area and Email Address fields*): Available when Auto-Populate is not used. Used to force the value entered to be a specific format. When the format is not used, when the record is submitted, a 'Please match the requested format' message will prompt for correction. The field's Help text can be used to provide what the value's specific format should be.
- **Date Range** (*Datetime fields*): Defines the allowed date and time value range.

17.7.7 Saving the Form

When the form is saved, a check occurs to make sure there are one or more required primary key fields.

If a primary key has not been defined, a prompt will ask about adding a new field to be the primary key. If yes, the new field is a text field, with label 'ID', auto-populated with a UUID value, and is hidden. The form still has not saved, providing the option to make any final adjustments. When everything is correct, select the Save button again. Or, you can stop the save and go back to the form layout and define one or more primary key fields.

17.8 Form Preview

Form Preview provides a working form to be used, just as if it was in a form panel on a dashboard. To access Form Preview, select it from above the form, right after Form Design.

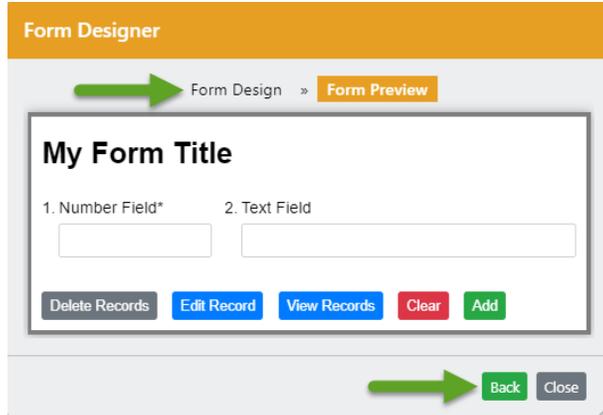


The form must always be saved on Form Design before being allowed to move on to Form Preview. If the form has not been saved, a prompt will provide a reminder.



Testing can be accomplished, allowing a quick return to Form Design, to make adjustments. Moving back to Form Design can be accomplished using the Back button, or selecting "Form Design" above the form.





17.8.1 Form Footer Buttons

This is the bottom of the form, where control buttons are automatically placed.

Note: The Delete, Edit and View Record(s) button visibility is controlled through the Form Design, under the form property's record control permission setting.

Button	Available When	Functionality
Delete Records	Delete Permission setting allows it	Launches an interface used to search the table based on the provided criteria. Retrieved records are displayed. One or more records can be selected, and deleted using the button.
Edit Record	Edit Permission setting allows it	Launches an interface used to search the table based on the provided criteria. Retrieved records are displayed. One record can be selected, retrieve, and populated on the form, using the Edit Record button. The record can be modified and saved to the table using the Update button.
View Record	View Permission setting allows it	Available when form property View Permission setting allows it. Launches an interface used to search the table based on the provided criteria. Retrieved records are displayed.
Clear	Always	Clears all values in the form, and resets form to Add mode.
Add	The form is in Add mode	Adds the record to the table.
Update	The form is in Update mode	Updates the table record.

Note: Multiple visual properties about a form can be customized when placing the form into a dashboard panel.



17.8.2 Search

The Delete, Edit and View Record controls will launch an interface used to search the table based on the provided criteria.

The interface will show the name of the control that was selected.

The **First 100 Records** button will find and display the first 100 records in the table. If the form's record control permissions restrict search to a user's records that will also be applied here.

Each data type, number, string and datetime, each have their own condition statements.

Data Type	Condition	Case
Number	Equals	
	Greater Than	
	Greater Than or Equal To	
	Less Than	
	Less Than or Equal To	
String	Contains	Case-Insensitive
	Begins With	Case-Sensitive
	Equals	
	End With	
	Is LIKE *	
	Is NOT LIKE *	
	Is NULL	
	Is NOT NULL	
Datetime	Equals	
	After	
	Equal To or After	
	Before	
	Equal To or Before	

* Denotes that the database's wildcard characters can be used. i.e. % for zero or more characters, and _ for a single character. If the searched character is also a special character, that is part of the search syntax, it can be 'escaped' using the '\ ' character.



The form's record control permissions, for edit, delete and view, will determine the type of records that can be accessed.

Own Records	The form must use one or both Auto Populate 'Create User Name' or 'Create User ID'. The search defaults to the create user records, of the current user, only.	
By Group	Any Record	All records are available to search.
	Own Records	The form must use one or both Auto Populate 'Create User Name' or 'Create User ID'. The search defaults to the create user records, of the current user, only. As defined at the group level.

