



DATAZEN

Product Documentation

Version 2.5 | © 2014 ComponentArt Inc.

Table of Contents

INTRODUCTION	4
PRODUCT OVERVIEW	5
PHILOSOPHY & APPROACH	6
PRODUCT ELEMENTS	7
ARCHITECTURE OVERVIEW	8
DATAZEN ENTERPRISE SERVER	9
INTRODUCTION AND KEY CONCEPTS	10
SERVER COMPONENTS	11
SECURITY ARCHITECTURE	12
SERVER SCALING	13
INSTALLATION GUIDE	14
<i>Installation Requirements</i>	15
<i>Installing Datazen Enterprise Server</i>	17
<i>Distributed Deployment</i>	22
INITIAL CONFIGURATION	23
<i>Authentication</i>	24
<i>Initial Users</i>	25
<i>Hub Configuration</i>	26
<i>Publishing Initial Content</i>	29
USING THE CONTROL PANEL	32
<i>Managing KPIs</i>	33
<i>Managing Dashboards</i>	36
<i>Data Access</i>	38
<i>Data Source Types</i>	41
<i>Managing Custom Maps</i>	45
<i>Managing Hub Users and User Groups</i>	46
<i>Managing Permissions</i>	50
<i>Managing Custom Branding</i>	52
USING THE WEB VIEWER	53
<i>Accessing Dashboards and KPIs</i>	54
<i>Configuring Public Access</i>	56
<i>Configuring Integrated Windows Authentication</i>	57
<i>Embedding Datazen Content Into Custom Apps</i>	58

<i>Embedding Datazen Content Into SharePoint</i>	59
SERVER MAINTENANCE	62
<i>Windows Services, IIS Sites and Application Pools</i>	63
<i>Configuration Settings</i>	64
<i>Server Core Settings</i>	65
<i>Rendering Service Settings</i>	68
<i>Control Panel Settings</i>	77
<i>Web API Configuration</i>	78
<i>Batch Creating Users</i>	79
<i>Server Logs</i>	80
<i>Email Templates</i>	81
<i>Backup and Restore</i>	83
CREATING AND PUBLISHING DASHBOARDS	85
DASHBOARD DESIGNER	86
DASHBOARD RUNTIME	88
DATA	90
<i>Data Model</i>	91
<i>Preparing Dashboard Data</i>	93
<i>Working with Excel Data</i>	96
<i>Working with Live Data Sources</i>	99
<i>Working with Simulated Data</i>	100
CONFIGURING NAVIGATORS	101
CONFIGURING VISUALIZATIONS	105
SAVING & PUBLISHING DASHBOARDS	112
ADVANCED TOPICS	114
<i>Parameterized Queries and Load on Demand</i>	115
<i>Drill-throughs to Other Dashboards or Custom URLs</i>	117
<i>Hierarchical Selections</i>	119
<i>Cascading Selection Lists</i>	120
<i>Working with Custom Maps</i>	121
EXTENDING DATAZEN	123
IMPLEMENTING CUSTOM BRANDING	124
IMPLEMENTING CUSTOM MAPS	131

Introduction

Welcome to the Datazen product documentation. This material is designed to provide comprehensive coverage of the entire product feature set, including detailed instructions on how to setup and configure Datazen Enterprise Server as well as create and publish dashboards and KPIs.

Target Audience

Product documentation is divided into two main categories:

1. [Datazen Enterprise Server](#): Written for system administrators who will be installing and configuring Datazen Enterprise Server, as well as power users who will be configuring BI hubs, connecting to live databases and creating data queries;
2. [Creating and Publishing Dashboards](#): Written for Business Intelligence professionals who will be creating dashboards based on the data queries created by system administrators or power users.

Prerequisites

All users of this documentation should already have a high-level understanding of the product, obtained through the official website and the included product videos. System administrators need be proficient in configuring Windows Server, IIS as well as their target data sources.

Product Overview

This section contains the following documents:

- [Philosophy & Approach](#)
- [Product Elements](#)
- [Architecture Overview](#)

Philosophy & Approach

Datazen was designed to enable rapid publishing of Business Intelligence content in a way that delivers premium user experience on any device. All product features are carefully designed to simplify the process of connecting to enterprise data sources and delivering insight securely to any device type and form-factor. The following points outline our high-level philosophy and approach.

Mobile First

Datazen was designed from the ground up for the next generation of devices. All aspects of the entire product stack - including the dashboard designer - are designed for touch interfaces, variable screen form-factors and mobile CPUs. In order to provide best possible experience to all users, Datazen offers native apps for Windows, iOS, Android and Windows Phone. In addition, Datazen scales well to traditional workstations as well as large screens.

Rapidly Develop and Publish

Every part of the Datazen product stack has been designed to enable rapid development and publishing. From quick data access to the integrated dashboard designer, you can build once and publish with confidence that your content will work flawlessly on any platform type or device form-factor.

Integrate with Existing Infrastructure

Customers need to be able to leverage their existing infrastructure investments. Datazen integrates with your existing databases, data warehouses and the security infrastructure. It is designed to be an effective data analytics and BI presentation layer that works with the backend products that you already use today.

Let Your Users Self-Serve

Data should be discoverable, but also controlled by corporate governance. Datazen enables administrators and power users to publish collections of data views and set granular access permissions. Users are able to browse the data views that have been published and easily create effective dashboards for their own analysis or for sharing with other users in the organization.

Always Know Your Data Is Secure

The entire Datazen product stack implements enterprise-grade security. Data is encrypted on the server, in transmission between the server and devices, as well as on the actual devices. Active Directory authentication is supported and granular access permissions can be configured for each KPI, dashboard or data view.

Have Fun

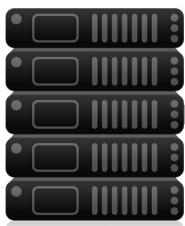
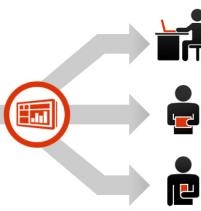
Yes, you can deliver a state-of-the art BI presentation and data analytics to your organization and have fun doing it!

Product Elements

Viewer Apps	Supported OS	Key Features
 	Windows 8 Windows 8.1+	<ul style="list-style-type: none"> • Connect to Datazen Server securely • Access published dashboards
 	iOS 7+	<ul style="list-style-type: none"> • Access published KPIs • Collaborate with team members in your BI hub
 	Android 4.1+	<ul style="list-style-type: none"> • Store KPI data on the device for offline access
 	Windows Phone 7.1 Windows Phone 8+	<ul style="list-style-type: none"> • Store dashboard data on the device for offline access*

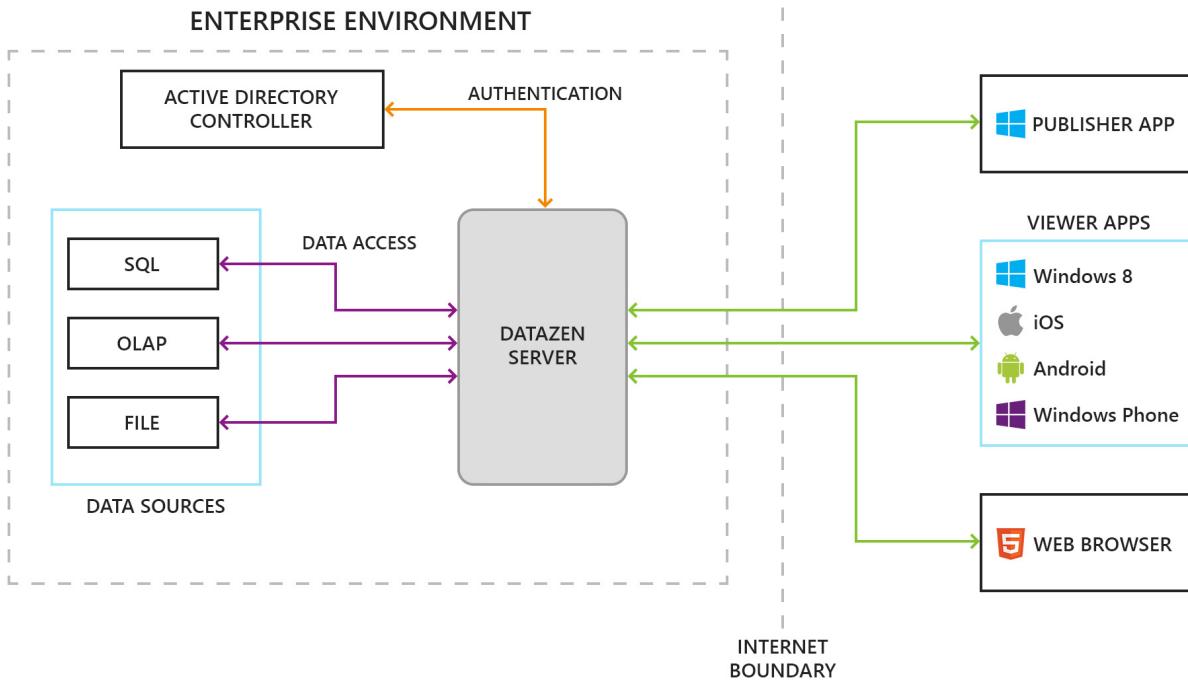
*Available only for Windows 8 in the current version.

Publisher App	Supported OS	Key Features
 	Windows 8 Windows 8.1+	<ul style="list-style-type: none"> • Create and share dashboards from local Excel data • Create dashboards from cloud & enterprise data sources and publish to any device

Server	Supported OS	Key Features
 	Windows Server 2008 R2 (64-bit) Windows Server 2012 (64-bit)	<ul style="list-style-type: none"> • Connect to enterprise data sources • Integrate with Active Directory for user authentication • Publish dashboards for access by any device • Personalize data queries for each user • Integrate corporate branding

Architecture Overview

The following diagram shows a typical Datazen deployment:



Datazen Enterprise Server is typically deployed within an enterprise IT environment, integrated with Active Directory for user authentication and connected to internal data sources for data access. All client applications (various viewer apps and the publisher app) communicate with Datazen Enterprise Server only; in other words they don't access the data sources directly. Client applications can access Datazen Server either from within the IT environment or outside the IT environment, if Datazen Server is configured for external access.

For more information on product architecture please refer to the following sections:

- [Server Components](#)
- [Security Architecture](#)
- [Server Scaling](#)
- [Dashboard Runtime](#)

Datazen Enterprise Server

This section contains the following documents:

- [Introduction and Key Concepts](#)
- [Server Components](#)
- [Security Architecture](#)
- [Server Scaling](#)
- [Installation Guide](#)
- [Initial Configuration](#)
- [Using the Control Panel](#)
- [Using the Web Viewer](#)
- [Server Maintenance](#)

Introduction and Key Concepts

Datazen Enterprise Server is a collection of applications, web applications and Windows services which:

- Acts as a repository for storing and sharing dashboards and KPIs
- Performs acquisition and caching of dashboard data
- Renders HTML5, PDF and image versions of dashboards to support various viewers
- Secures dashboards and data for the organization
- Provides an administrative Control Panel for managing and maintaining all aspects of the deployment

BI Hubs

BI hubs are containers for grouping together an organization's users, dashboards, KPIs, data and security into a logical unit. A single Datazen Server instance can have any number of BI hubs created and deployed. Users can be members of multiple hubs at the same time, potentially with different roles: hub owners, hub publishers or simply view-only users.

Dashboard and KPI Repository

One of the primary purposes of the Datazen Enterprise Server is to act as a central repository for dashboards and KPIs. Without Datazen Enterprise Server, dashboards can only be shared by e-mailing dashboard files and opening them with the Windows client on another machine. With Datazen Enterprise Server, dashboards and KPIs can be published to the server, organized and secured there, and retrieved at a later time by Datazen clients.

Data Acquisition and Caching

Datazen dashboards use tabular data to power their rich visualizations. Without Datazen Enterprise Server, this data is limited to Microsoft Excel spreadsheets that are available to the author when the dashboard is created. With Datazen Enterprise Server data can be configured on the server and backed by various data providers. This data can then be added to visualizations by the dashboard author and is always up to date with the server, providing dashboard viewers with near real-time data capability.

Dashboard Rendering Features

Datazen dashboards are authored in the Designer Windows application. With Datazen Enterprise Server published dashboards can be rendered to HTML5, PDF and image formats becoming viewable on almost any device, including the various Datazen mobile apps (available for iOS, Android and Windows Phone).

Dashboard and Data Security

Datazen Enterprise Server provides configurable membership and security features, allowing administrators to provide tight granularity over who can access dashboards and the data that drives them. User authentication can optionally be controlled via an existing Active Directory infrastructure.

Server Components

Datazen Enterprise Server consists of the following software components:



Server Core

- Windows Service with an embedded no-SQL database
- Metabase stores users, KPIs, dashboards, data views, custom maps, permissions, custom brand packages, cached data
- Data acquisition service performs scheduled polling of all data sources and caches the results for quick access
- Server core is designed for high-availability, fast reads with a large number of objects

Web API

- REST-based API for communication with all native clients
- Performs user authentication, obtaining KPIs and dashboards, data source browsing, dashboard publishing and hub activity participation

Web Viewer

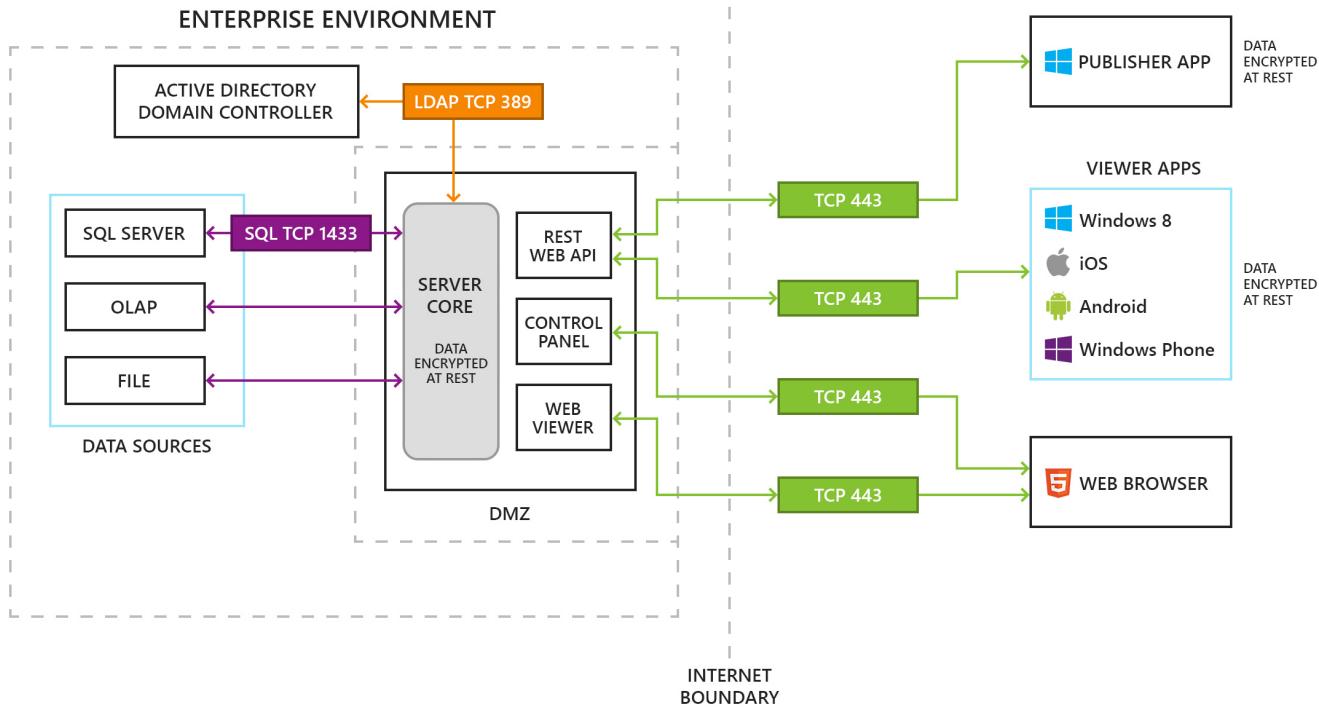
- ASP.NET application providing browser access to all Datazen Enterprise Server content

Control Panel

- ASP.NET application for browser-based server administration

Security Architecture

The following diagram describes the security architecture of a typical Datazen Enterprise Server deployment:



Data Encryption

Datazen implements data encryption at all levels of the product stack:

1. Data is encrypted at rest when stored in the Datazen Enterprise Server metabase;
2. Data is encrypted via SSL when transmitted from Datazen Enterprise Server to client devices;
3. Data and user credentials are encrypted on each device.

User Authentication

User authentication is performed through a secured REST-based API. Based on the authentication settings, actual user validation is executed either through an LDAP call behind the corporate firewall, or internally within the Datazen Enterprise Server metabase. For more information, please refer to the [Authentication](#) section.

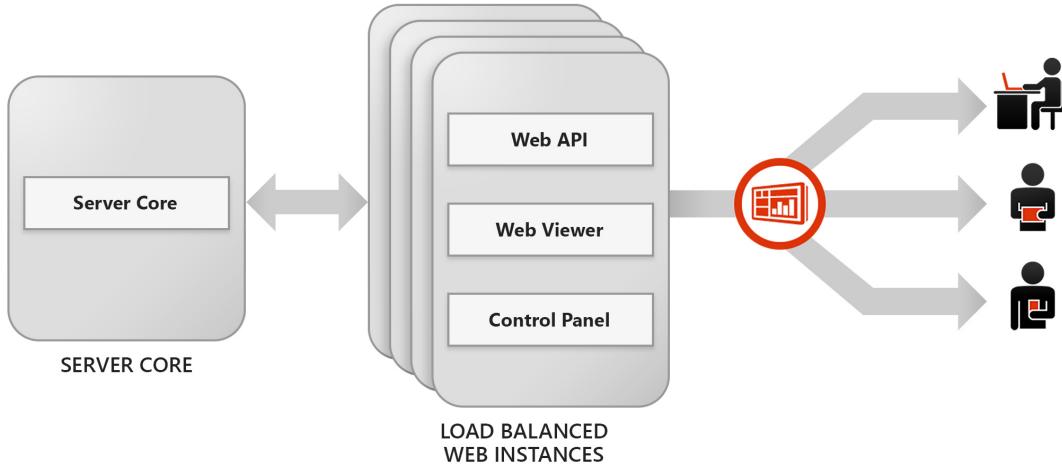
Managing Access Permissions

Highly granular user permissions can be setup for the following Datazen Enterprise Server objects: KPIs, dashboards, data views. For details, please refer to the [Managing Permissions](#) section.

Server Scaling

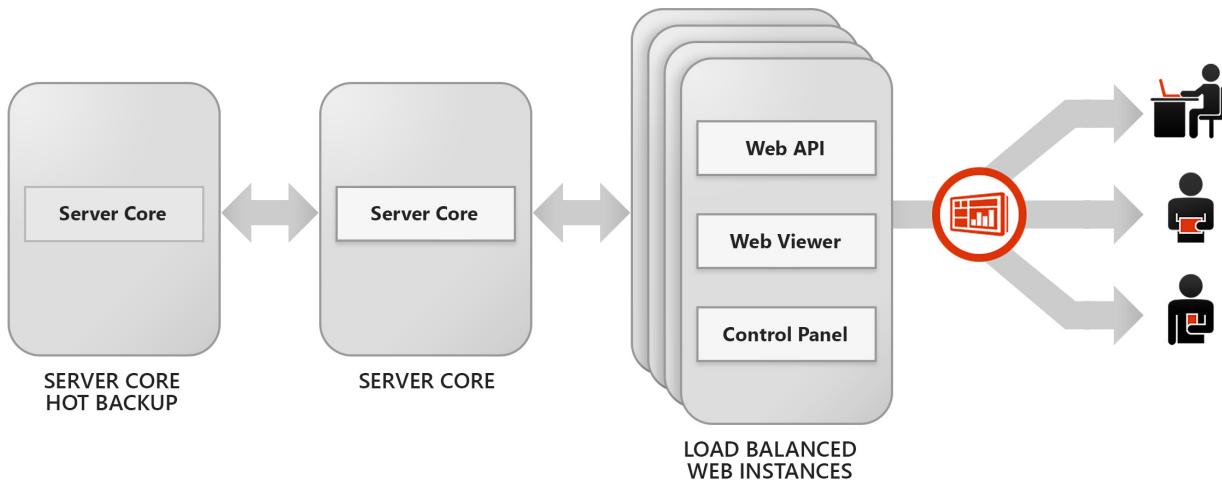
All components of the Datazen Enterprise Server can be distributed to different server machines and scaled. However, in practice it is recommended to keep the Server Core component on a single machine (since it is able to handle tens of thousands of users) and to distribute "Web Instances". Here are some common deployment scenarios:

Distributed Deployment



1. Server Core is kept on a single machine;
2. Multiple load balanced "Web Instances" are deployed, each containing Web API, Web Viewer and Control Panel components. The load balancing technology for "Web Instances" needs to support sticky sessions.

High Availability Distributed Deployment



1. A second Server Core machine is deployed as a "hot backup";
2. In case of a failure of the primary Server Core machine, the "hot backup" machine automatically takes over.

For information on setting up and configuring distributed scenarios, please refer to the [Distributed Deployment](#) installation section.

Installation Guide

This section contains the following documents:

- [Installation Requirements](#)
- [Installable Server Features](#)
- [Simple Deployment](#)
- [Distributed Deployment](#)

Installation Requirements

Datazen Enterprise Server can be installed in standard Windows Server environments on physical machines, virtual machines or in the cloud. This document covers the exact software and hardware requirements, as well as the recommended deployment configuration for common scenarios.

Hardware Requirements

The following are minimum hardware specifications for server machines running Datazen Enterprise Server:

- CPU: 8 cores
- RAM: 16GB
- Available Disk Space: 50Gb

Software Requirements

Datazen Enterprise Server requires one of the following:

- Windows Server 2008 R2 64-bit with: IIS and .NET 4.0 (KB2468871), or
- Windows Server 2012 64-bit with IIS installed and configured as follows:
 - Windows Features – ASP.NET 4.5
 - Windows Roles – IIS
 - Windows IIS Feature - Application Development -> ASP.NET 4.5
 - Windows IIS Manager GUI -> Select Server -> "Feature Delegation"
 - Set Handler Mappings -> read/write
 - Set Modules -> read/write

Deployment Configuration for Common Scenarios

Please refer to the table below as a high-level guideline regarding the recommended server configuration for various deployment scenarios:

Scenario	Server Configuration
Development – up to 50 registered users	1 Server with all features installed
Production – 200 registered users	1 Server with all features installed
Production – 500 registered users	1 Server Core Machine 2 Web Server Machines
Production – 1,000 registered users	1 Server Core Machine 4 Web Server Machines
Production - 10,000 registered users	1 Server Core Machine 10 Web Server Machines

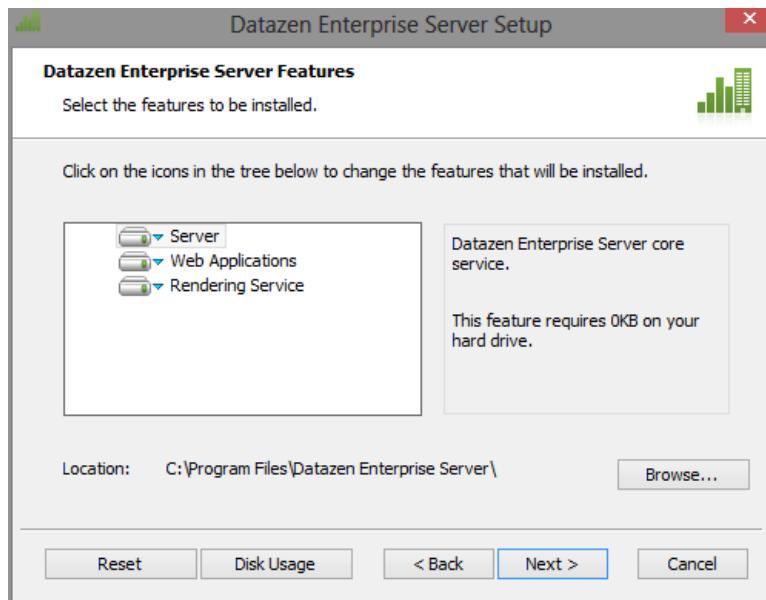
Notes:

1. Each server machine needs to meet the hardware and software requirements listed above;
2. These high-level recommendations are not based on concurrent users, but total registered users. Estimate is that 20% of registered users will access the system during peak times;
3. For deployments that require multiple server machines, please refer to the [Distributed Deployment](#) document.

Installing Datazen Enterprise Server

The Datazen Enterprise Server installation executable contains the three major installable server features, Server service, Web Applications, and [Rendering Services](#).

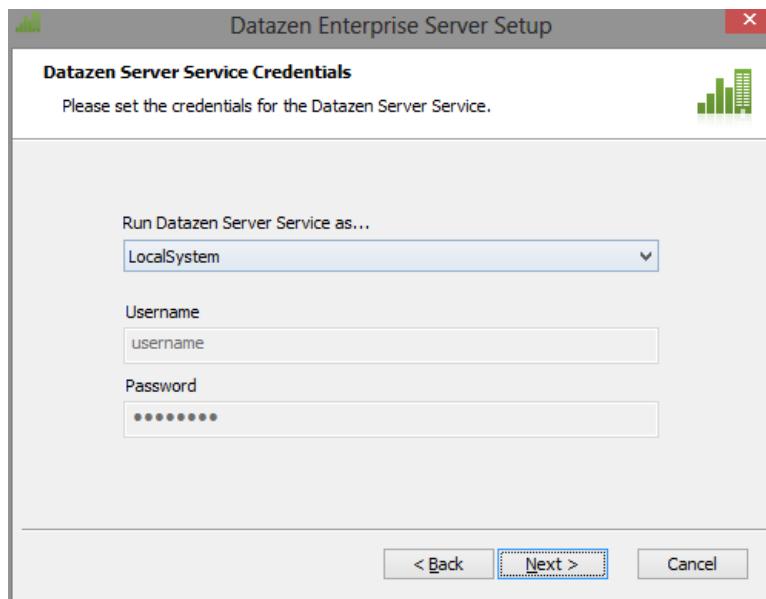
All of these features are required in order for the Datazen Enterprise Server to function. However, they do not need to be installed on the same machine. For various reasons it may be desirable to install the features together on a single-machine, or [distribute them among several machines](#).



Server service

This is the Datazen Enterprise Server core service. It contains the database and **must** be installed before any other features. This feature may **not** be distributed.

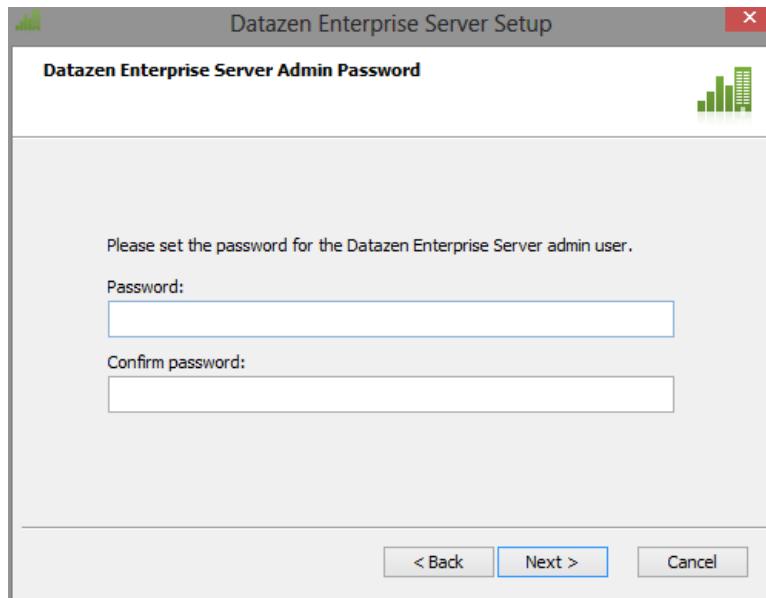
Service credentials



Use this dialog to specify under which credentials the Server service will run.

In a single-server, [default authentication](#) scenario, this should be left as *LocalSystem*. If you are planning on using [Active Directory authentication](#) and this server is a domain member, the Server service **must** be configured to run as a domain account.

Admin password

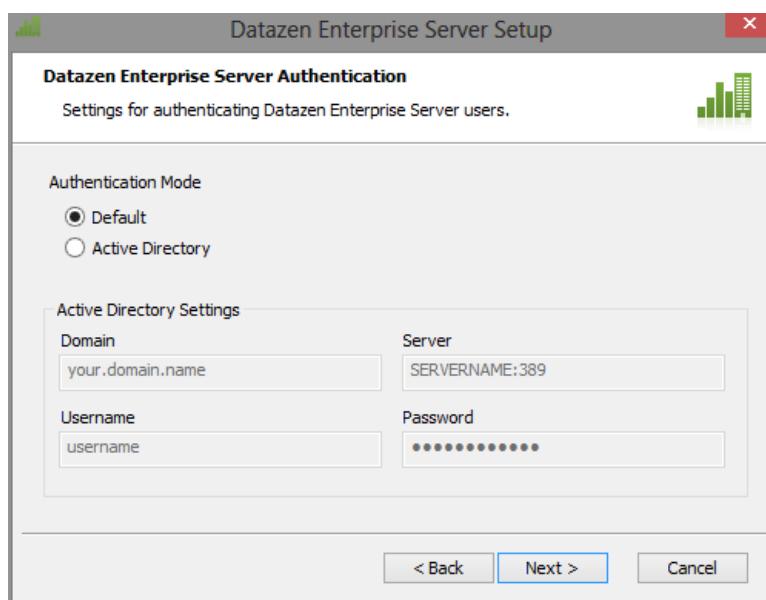


Enter the password for the Datazen Enterprise Server *admin* user. The password may be changed later via the [Control Panel](#).

Authentication mode

Datazen Enterprise Server users can be authenticated in one of two modes:

- *Default mode* - usernames and passwords are stored in the Server service database.
- [Active Directory](#) - usernames are stored in the Server service database. Passwords are authenticated against a local or remote Active Directory environment.



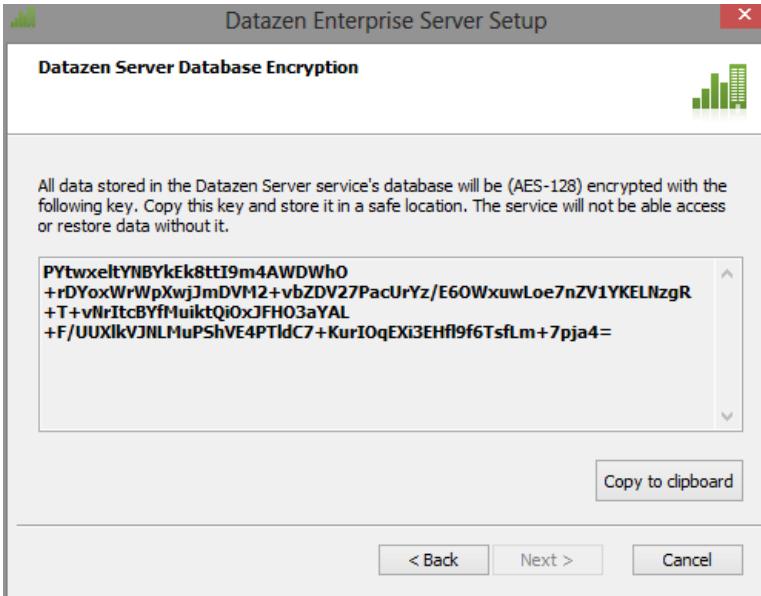
When Active Directory is selected, the following options are available:

- *Domain* - If all of the users are in a single domain, enter the domain name here. Users can then be created by

username only, instead of *username@domain.name*. If users are part in multiple domains, leave this blank.

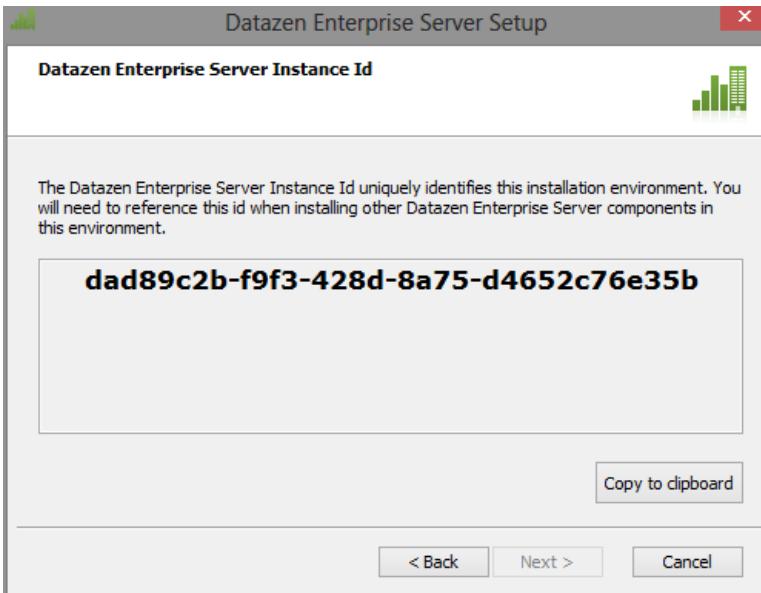
- **Server** - If this machine is not a domain member, the Server service will use LDAP to query Active Directory. Enter the LDAP server and port here. If this machine is a domain member, leave this blank.
- **Username** - If using LDAP, enter the username of the account that will perform Active Directory query. Otherwise, leave this blank and the query will be performed with the service credentials provided earlier.
- **Password** - If using LDAP, enter the password of the account above. Otherwise, leave blank.

Encryption Key



This is the database encryption key. This key is required in order to perform a [database restore](#). Ensure it is stored safely.

Instance Id



This is the Datazen Enterprise Server instance id. This key is required if [distributed components] will be installed in the future.

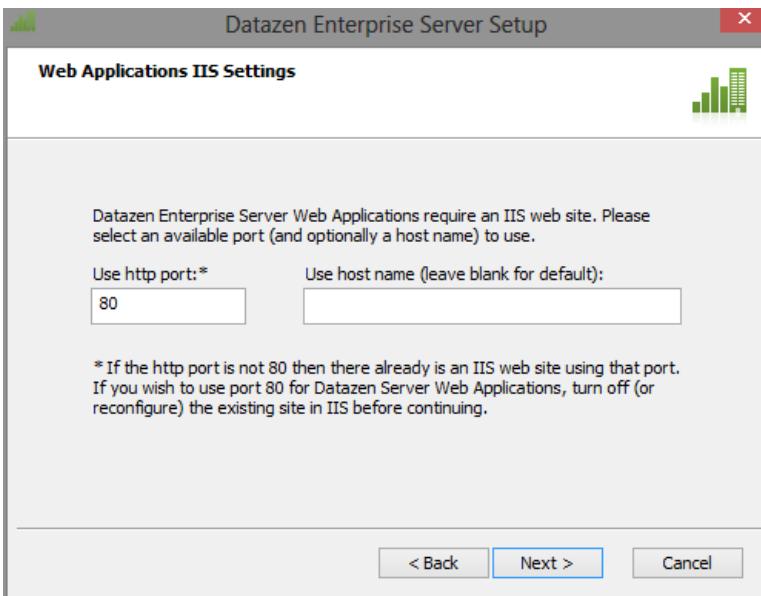
Web Applications

This feature provides the following Web Applications to Datazen Enterprise Server client applications:

- *Web API* - all of the Datazen Enterprise Server client applications communicate directly with this service.
- [Control Panel](#) - the management portal used to manage users, dashboards, data, security and other aspects of the Datazen Enterprise Server installation.
- [Web Viewer](#) - a web application to facilitate viewing published KPIs and dashboards in any modern web browser.

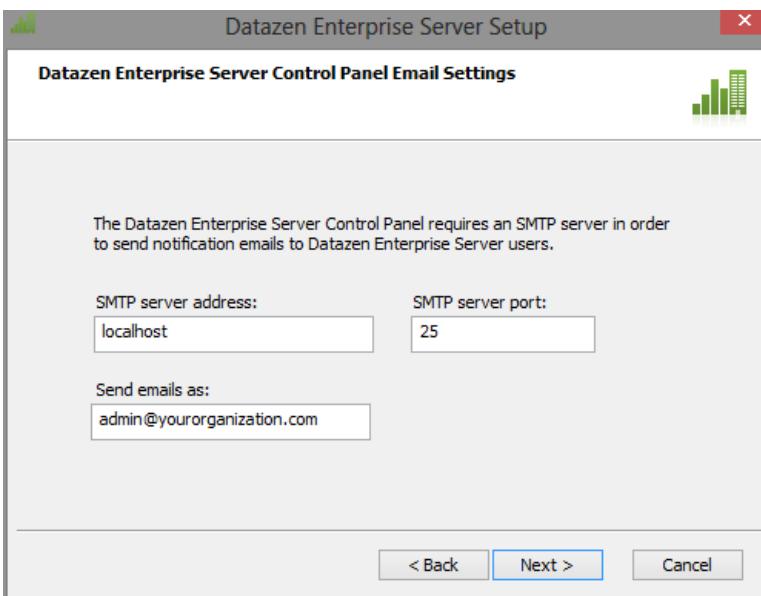
This feature may be [distributed](#).

Internet Information Server



Specify an available port (and optional hostname) on which Web Applications will run. A new website and several IIS application pools will be created. The website settings can be configured anytime with the Windows IIS Manager (for enabling SSL security, for example).

Control Panel Email



The [Control Panel](#) requires an SMTP server in order to send notification emails to Datazen Enterprise Server users.

Rendering Service

The [Rendering Service](#) is a Windows service that is responsible for:

- Accepting requests from the [Web Viewer](#) and generating the HTML5 response.
- Generating thumbnails for published dashboards.

This feature may be [distributed](#).

This feature does not present any installation dialogs. For post-installation configuration of this service, please see the [Rendering Service configuration documentation](#).

Distributed Deployment

The Datazen Enterprise Server environment can be distributed among several machines. Distributing server features provides many benefits over a single-machine installation.

Only the Web Applications and Rendering Service features may be distributed. The Server service cannot be distributed.

Web Applications and Rendering Services may be installed together (both running on the same machine) or separately, depending on the installation requirements.

Distributing Web Applications

Distributing the Web Applications feature among several machines will provide the following benefits:

- More client applications can be simultaneously connected to the Datazen Enterprise Server.
- The Datazen Enterprise Server environment will still be accessible to client applications if any single Web Application machine becomes unavailable.

To create a distributed Web Applications node:

1. Run the Datazen Enterprise Server installation executable.
2. On the feature selection screen, ensure Web Applications is selected to be installed, and Server is not.
3. On the *Server Service Location* dialog, enter the location and Instance Id of the existing Server service.
4. Proceed with the installation as outlined in the [installation guide](#)*.

* If the Rendering Service is not selected to be installed you will be prompted for its location, since the Web Applications feature depends on it.

Distributing Rendering Service

Distributing the Rendering Service feature among several machines will provide the following benefits:

- An increased number of dashboards are able to be rendered simultaneously for the [Web Viewer](#).
- An increased number of thumbnails are able to be rendered simultaneously.
- The Datazen Enterprise Server environment will be able to continue rendering dashboards and thumbnails if any single Rendering Service machine becomes unavailable.

To create a distributed Rendering Services node:

1. Run the Datazen Enterprise Server installation executable.
2. On the feature selection screen, ensure Rendering Services is selected to be installed, and Server is not.
3. On the *Server Service Location* dialog, enter the location and Instance Id of the existing Datazen Server service.
4. Proceed with the installation as outlined in the [installation guide](#).

Initial Configuration

This section contains the following documents:

- [Authentication](#)
- [Initial Users](#)
- [Hub Configuration](#)
- [Publishing Initial Content](#)

Authentication

Users can be authenticated by either the Datazen server itself (default) or by an existing Active Directory infrastructure.

Datazen authentication

The default authentication mechanism stores and validates user passwords within the Datazen server. When a user is created a notification email is sent to the user which contains a URL to initialize the account and set the password. Users may change their password at any time by managing their profile via the Control Panel.

Active Directory Authentication

Active Directory authentication mode utilizes an existing Active Directory infrastructure for username and password validation. In this mode, when a user attempts to log in the following validations are performed:

1. The username is checked first within the Datazen server users list.
2. If found the username and password are forwarded to Active Directory for further validation.

In this mode, passwords are not stored within the Datazen server.

For more information on configuring Active Directory authentication mode, please refer to the Datazen Enterprise Server service and installation documentation.

Initial Users

The Admin User

When you first install Datazen Enterprise Server, a special admin user is created. This user has full administrative rights to perform any operation in the server instance, and is the only user that can create other users or hubs. When you log in to the Control Panel as the admin user, you will see several options for managing and configuring the server instance that would otherwise not be available.

The Guest User

If created, the user with username guest has a special significance. The account with this name is used for public web access to dashboards. When maintaining a guest account, ensure that it can only access those dashboards which are intended to be viewed by anyone.

Server Users

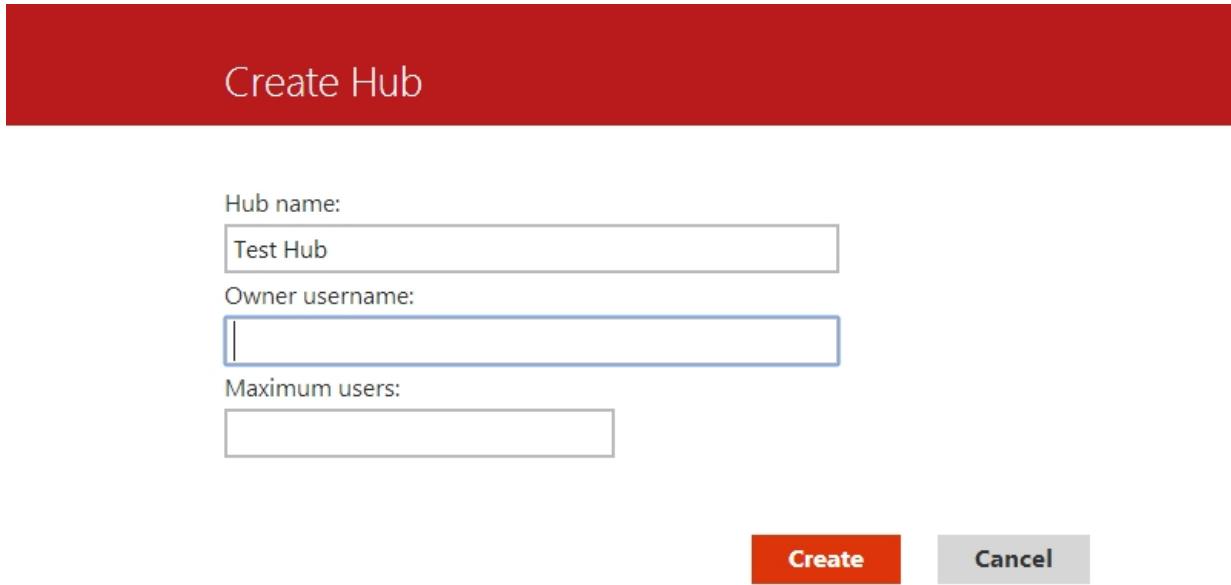
Before Datazen can be used, at least one initial server user should be created. To create the first user, log into the Control Panel as Admin, and go under Server Users to click on Create User.

For more information about using the Control Panel to manage user accounts, see [Managing Hub Users and User Groups](#).

Hub Configuration

Before any content can be published to the server, the first hub must be created and configured.

1. On the Home screen of the Control Panel, click on "Create Dashboard Hub".



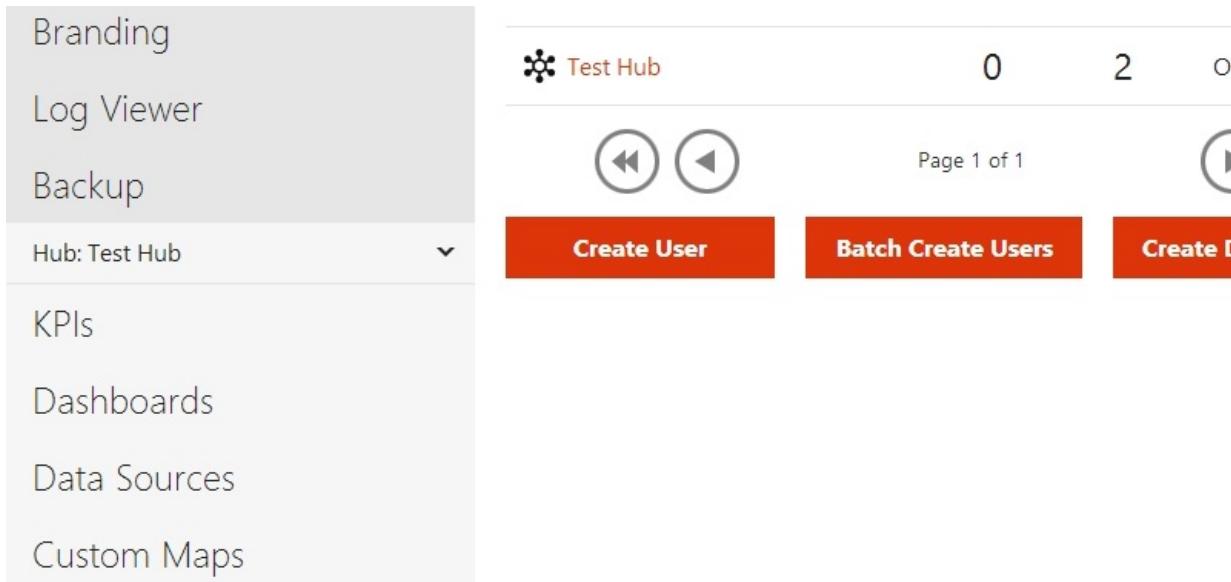
The screenshot shows the "Create Hub" dialog box. It has a red header bar with the title "Create Hub". Below the header are three input fields: "Hub name:" containing "Test Hub", "Owner username:" with an empty input field, and "Maximum users:" with an empty input field. At the bottom right are two buttons: a red "Create" button and a grey "Cancel" button.

Create Dashboard Hub dialog

1. Enter "Test Hub" for the Hub name.
2. Enter a user you've created for the Owner username.
3. Enter 50 under Maximum users.
4. Click "Create". When this is done, click "Done".

Next, we may want to add more users to the newly created hub.

1. Switch to the new hub using the hub dropdown in the navigation pane on the left-hand side of the Control Panel.



The screenshot shows the left navigation pane and the hub selector area. The navigation pane includes links for Branding, Log Viewer, Backup, KPIs, Dashboards, Data Sources, and Custom Maps. A dropdown menu labeled "Hub: Test Hub" is open. The hub selector area shows a list with one item: "Test Hub" (with a gear icon) followed by the numbers 0, 2, and 0. Below this are navigation arrows, a page indicator "Page 1 of 1", and three red buttons: "Create User", "Batch Create Users", and "Create D".

Hub selector

1. Click on Users. You should see the hub owner you've specified as the sole user in the new hub.
2. Click on "Invite More Users".
3. Enter email addresses of users to be added to this hub as a delimited list in the text input area and click "Invite Users".

Users already registered on the server will automatically be added to the hub. Others will be emailed with instructions on how to join the server and access the hub.

In order to be able to publish dashboards to the new hub, we need to create at least one dashboard group.

1. Click on "Dashboards" on the left-hand side nav.
2. Click on "Create a New Group".

Create a New Group - Step 1 of 2

Create a new dashboard group in **Test Hub**.

Group name:

Next **Cancel**

Create Dashboard Group dialog

1. Enter "Test Dashboards" for the name. Click Next.

Create a New Group - Step 2 of 2

Set Permissions:

User Groups

User Group	Allow Access
 Everyone	<input checked="" type="checkbox"/>

Previous

Create

Cancel

Dashboard Group access permissions dialog

1. Give "Everyone" access rights for now.
2. Click "Create".

You should now be able to log into the server from a publisher application and create and publish a dashboard to the new hub.

Publishing Initial Content

Once an initial user and hub have been established, we can publish some content to the server to make sure everything is in order.

Create a Test KPI

Log into the Control Panel under the user account you've created. We will first need to create an initial KPI group.

1. Under KPIs, click on Create a New Group.
2. Select "No Connection" for the data connection. We will manually enter the data for this first KPI. Enter "Test KPIs" for the title. Click OK.
3. Give "Everyone" access.

You should now see an empty KPI group named "Test KPIs". We'll create a test KPI within that.

1. Click on "Create New KPI" on the group header.

New KPI

Preview:

Test KPI 1
\$123,000 (+11%)



KPI Name:

Value Format:

Value: Enter Value:

Goal: Enter Goal value:

Status: Enter Trend value:

Trend Set: Enter Trend Set value:

Visualization:

Create **Cancel**

New KPI dialog

1. Enter "Test KPI 1" for the KPI name.
2. Select Currency for the Value Format.
3. Enter 123000 for the Value.
4. Select "Set manually" for the Goal, and enter 111000.
5. Select "Set manually" for the Status, and select +1 (green) for its value.
6. Select "Set manually" for the Trend Set, and accept the default random values.
7. Click "Create".

You should see your test KPI in the group.

Connect with a Publisher App

Using the Datazen Publisher, connect to the newly configured server.

1. On the bottom app menu, click "Connect".
2. On the Server Connect dialog, enter the server address and the credentials for a user you've created.
3. Click "Connect".

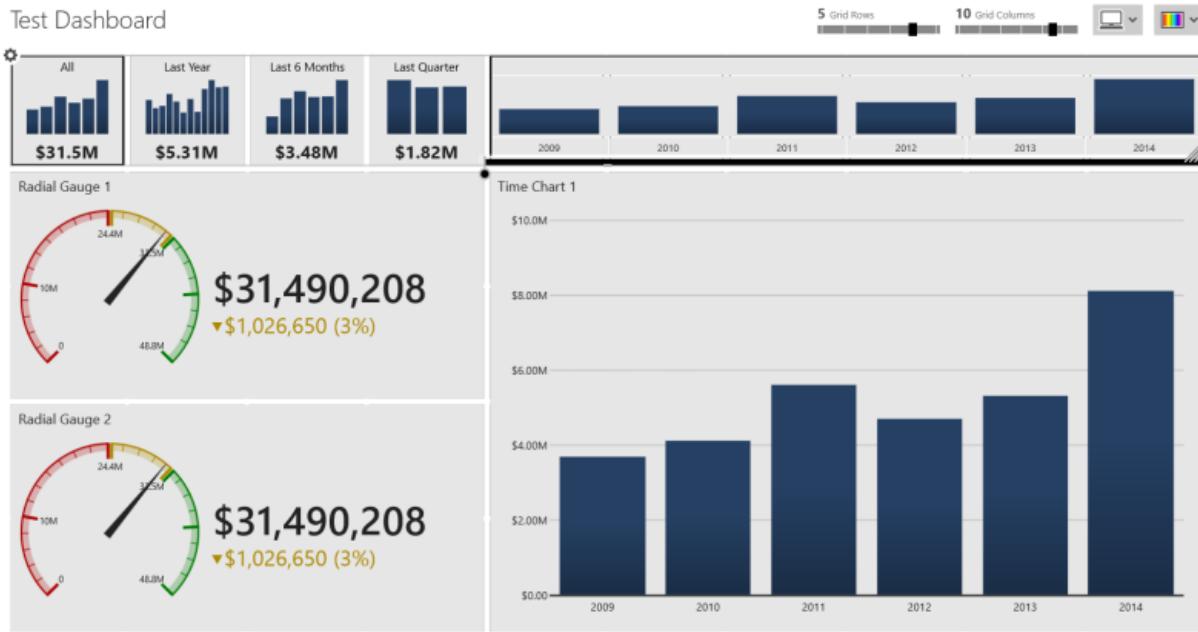
You should now be connected to the server. If you created a Test KPI in a previous step, you should now see it on your home screen.

Publish a Test Dashboard

To create a simple test dashboard, activate the bottom app menu again and click "Create".

1. In the Dashboard Designer, start by entering the dashboard name at the top.
2. Drag a few dashboard elements onto the design surface: a Time Navigator, two Radial Gauges and a Time Chart.

As you do this, simulated data is generated to produce an approximation of the dashboard's appearance.



Test dashboard

You can "Run Preview" to see your test dashboard in action. Return to the designer to publish the dashboard to the server.

1. On the bottom app menu, click "Publish to Server".

First, the dashboard must be saved on the local file system.

Verify the file name and default *local* dashboard group (*My Dashboards* is fine for now) and click "Save".

The first time you do this, you will be prompted to select a working folder for the publisher. This is the folder where

your local dashboards will be stored.

Next, we are prompted for the server destination.

Make sure your server is selected, as well as the right hub and dashboard group. Click "Publish".

Your test dashboard should now be published on the Datazen server. Back on the home screen, it should appear under *Test Dashboards*.

Verify Initial Content Using Web Viewer

The content you have created is also accessible via the web viewer, at `http(s)://your.server.address/viewer`. Use your browser to navigate there.

Log in using the credentials you used to publish your first dashboard. You should see your initial content.

Clicking on the dashboard tile should present the interactive web view of the dashboard.

Use the top menu to switch between dashboards and KPIs.

For more information on the Web Viewer, see the [section dedicated to its use](#).

Using the Control Panel

This section contains the following documents:

- [Managing KPIs](#)
- [Managing Dashboards](#)
- [Data Access](#)
- [Data Source Types](#)
- [Managing Custom Maps](#)
- [Managing Hub Users & User Groups](#)
- [Managing Permissions](#)
- [Managing Custom Branding](#)

Managing KPIs

Control Panel's *KPIs* page allows users to create and organize KPIs on a hub.

KPI Groups

KPI groups contain information of how their member KPIs will connect to remote data. Unlike [Data Connections](#), they can also be static and not require a network connection.

Edit, delete, or reorder KPIs and KPI Groups

The screenshot shows a management interface for KPI groups. At the top, there is a header bar with a dropdown arrow, the title "Kpi Group 1", and three buttons: "Create New KPI", "Sub-KPIs", and "Related Dashboards". Below the header is a list of four KPI groups, each represented by a small icon and a name:

KPI Group	Action Icons
Total Sales	Edit, Delete, Related Dashboards
Total Expenses	Edit, Delete, Related Dashboards
Profit	Edit, Delete, Related Dashboards
Profit Margin	Edit, Delete, Related Dashboards

At the bottom right of the list, the text "KPI edit dialog." is visible.

Creating a KPI group

1. Click on *Create a New Group*.
2. Select the desired KPI group type and complete the form. See [Data Source Types](#) for more information.
3. Fill out the name and other properties for the connection. Click *Test Connection* to verify that the data provider can be reached by the Datazen server. Click *Next* when done.
4. Grant access permissions to user groups or users. Access permissions can be set at any time from the [Permissions page](#).
5. Click *Finish* when done.

The new KPI group will now be listed among the available KPI groups.

Editing a KPI group

1. Click the *Edit* icon for the KPI group.
2. Update the properties in the presented dialog.
3. **Optional** Click *Test Connection* to verify the connection status.
4. Click *Next* when done.
5. Update the permissions. Click *Finish* when done.

The KPI group will now be updated.

Removing a KPI group

1. Click the *Delete* icon for the KPI group.
2. Click *Delete* in the presented dialog.

The KPI group will now be removed from the system.

KPIs

KPIs are defined and managed in the Control Panel. Each KPI can consist of a primary setting, Value, as well as optional secondary settings:

- Goal, which is used for comparison against the Value
- Status, which defines the background of the KPI visual
- Trend Set, which defines the graph of the KPI

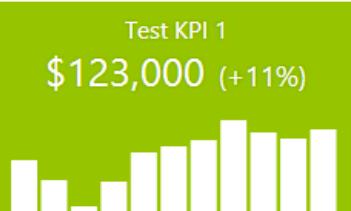
Each of these settings can be set manually (a static value), or be defined in a query which will be applied against the KPI group's data connection. These queries, like [Data Views](#), may be personalized.

KPIs can have sub-KPI children, which themselves are full KPIs, inheriting the KPI group data connection. KPIs can also be assigned related dashboards, which will be presented when the KPI is viewed in a client interface.

New KPI

Preview:

Test KPI 1
\$123,000 (+11%)



Value:	Set manually	Enter Value:	123000
Goal:	Set manually	Enter Goal value:	111000
Status:	Set manually	Enter Trend value:	+1 (green)
Trend Set:	Set manually	Enter Trend Set value:	40, 25, 5, 23, 45, 50, 55, 70, 61, 56, 63
Value Format:	Currency	Visualization:	<input type="checkbox"/> NONE <input checked="" type="checkbox"/> Bar <input type="checkbox"/> Line <input type="checkbox"/> Step <input type="checkbox"/> Area
		<input type="button" value="Create"/> <input type="button" value="Cancel"/>	

KPI and KPI groups listing.

Creating a KPI

1. Scroll to the *KPI group* to which the new KPI should be added. If this will be a sub-KPI, navigate down the *Sub-KPIs* link from the parent KPI.
2. Click *Create New KPI* on the KPI group/KPI. A form presenting the KPI properties will appear.
3. Complete the form. The KPI *name* must be unique to the KPI group.
4. Click *Create* to complete the KPI creation process.

The new KPI will now be listed among the other KPIs.

Changing a KPI's related dashboards

1. Navigate to the KPI via the sub-KPIs if necessary.
2. Click the *Related Dashboards* icon for the KPI.
3. Add related dashboards by clicking them in the left column; reorder them by dragging the dashboard in the right column; remove them by clicking the X.
4. Click *Update* when done.

The KPI's related dashboards will now be updated.

Edit a KPI

1. Navigate to the KPI via the sub-KPIs if necessary.
2. Click the *Edit* icon for the KPI.
3. Update the properties in the presented dialog. Click *Update* when done.

The KPI will now be updated.

Remove a KPI

1. Navigate to the KPI via the sub-KPIs if necessary.
2. Click the *Delete* icon for the KPI.
3. Click *Delete* in the presented dialog.

The KPI has now been removed from the system.

Managing Dashboards

Control Panel's *Dashboards* page enables Hub owners to reorder the Hub's dashboards and dashboard groups, create new dashboard groups, edit the names of dashboards or dashboard groups, and delete dashboards or dashboard groups.

Edit, delete, or reorder dashboards and dashboard groups

Dashboard	Action
Sales vs. Expenses	<input type="checkbox"/>
Sales by Region	<input type="checkbox"/>
Sales by Industry	<input type="checkbox"/>
Product Scores	<input type="checkbox"/>

Dashboards and dashboard groups listing.

Dashboards

Renaming a dashboard

1. Click on the *Edit* icon next to the dashboard.
2. Update the name of the dashboard in the dialog that appears.
3. Click *Apply* when done.

The dashboard has now been renamed.

Deleting a dashboard

1. Click on the *Delete* icon next to the dashboard.
2. Click *Delete* when prompted.

The dashboard has now been deleted.

Reorder/move a dashboard

1. Click on the drag-handle of the dashboard, and drag the dashboard to its new position. The dashboard can also be dragged into a different dashboard group.
2. Release the drag-handle.

The dashboard has now been reordered/moved.

Hide a dashboard

1. Click on "Hide on Main Screen" checkbox for the dashboard.

The dashboard is now hidden from the Main Screen of client applications. This is intended to prevent users from directly viewing dashboards that are intended to only be drilled into.

Dashboard Groups

Creating a dashboard group

1. Click the *Create a New Group* button.
2. Enter a group name in the dialog that appears. Click *Next* to continue.
3. Grant access permissions to user groups or users. Note that these access permissions can be set at any time from the [Permissions page](#).
4. Click *Create* to complete the process.

The dashboard group will now be added to the selected hub.

Renaming a dashboard group

1. Click on the *Edit* icon next to the dashboard group.
2. Update the name of the dashboard group in the dialog that appears.
3. Click *Apply* when done.

The dashboard group has now been renamed.

Deleting a dashboard group

1. Click on the *Delete* icon next to the dashboard group.
2. Click *Delete* when prompted.

The dashboard group, and any dashboards it may have contained, has now been deleted.

Reorder a dashboard group

1. Click on the drag-handle of the dashboard group, and drag the group to its new position.
2. Release the drag-handle.

The dashboard group has now been reordered.

Data Access Concepts

Datazen dashboards use tabular data (*Data Views*) to power their visualizations. Datazen Enterprise Server data are configured on the server and backed by various data providers (*Data Sources*). This data can then be added to visualizations by the dashboard author, having connected to the Datazen Server. Without a connection to Datazen Enterprise Server, data is limited to local Microsoft Excel spreadsheets that are available to the author when the dashboard is created.

Data Connections

All Folders and Data Connections

New Folder...

Folder/Data Connection	Edit	Delete
 Marketing Documents		
 Operations Documents		
 Corporate Intranet SharePoint lists		
 Sql Azure DemoDb		

Data connection listing.

Data connections contain information (i.e. address, credentials) about how to connect to remote data so that it can be retrieved and be made available to dashboard authors. Depending on the [data connection type](#), different configuration settings are available.

Only hub owners are allowed to create or manage data connections. Access permissions must be set on a data connection before other hub members are allowed to use it.

Data Views

 Sql DemoDb

Data View	Edit	Delete	View	Status
 Daily Sales				Last Update: 11: Next Update: 12: Last Status: Suc
 Expense Category Lookup				Last Update: 11: Next Update: 12: Last Status: Suc
 Expense Time Lookup				Last Update: 11: Next Update: 12: Last Status: Suc
 Expenses By Category LOD				Last Update: 7:1: Next Update: 7:0: Last Status: Suc

Data view listing.

Some [data connection types](#) support child data views. A data view uses the parent data connection to connect to the remote data, but has its own settings for selecting which data to acquire from the connection. For example: A Microsoft SQL Server data connection would contain a connection string to locate the data store, and its child data view would contain a T-SQL statement that returns a specific table of data.

Data connection types that support data views must have one or more data views defined in order to begin acquiring remote data.

Data views inherit the access permission of the parent data container. A hub member with access to the data container will have access to all of the data views contained within it.

It is possible to author a dashboard that a hub member has access to, but contains data that the hub member does not have access to. In this scenario the dashboard would display on a Datazen viewer but without any data.

Cached Data Views

Edit Data View - Step 1 of 2

Data View Name:

Refresh Frequency: Real-time [Define parameters...](#) 

Allow Client Data Caching
 Personalize for each member 

Data View Query:

```
SELECT [Date] ,[Amount] FROM [dbo].[SalesByDay]
```

 Editing a data view.

Once data connections and views have been configured in a hub, the Datazen Enterprise Server will periodically retrieve the data from the original data source and store it internally, caching it for client use. The schedule for how often the data is refreshed is configurable either for the individual data view (for [data connection types](#) that support child data views) or on the data connection itself.

Data for views can be manually refreshed at any time via the Control Panel's *Data Sources* page.

Real-time Data Views

The data retrieval schedule can be bypassed by setting the real-time flag on the data connection or view. By turning on real-time data retrieval, every time a dashboard that utilizes the data is viewed the original data store is re-queried for the data.

Real-time views can use parameters in their query to filter data before it is sent to a dashboard.

Hub owners should be aware of the increased network utilization incurred for real-time data, particularly where the original data resides over a metered connection or in a fee-based service such as Windows Azure.

Personalized Data Views

By default, the data provided to a specific dashboard by a data connection or view is the same for all users. Data personalization allows this data to be specific to each user. When a dashboard is viewed that utilizes personalized data, the server will return only the data specific to the logged-in user.

When this feature is enabled the data retrieval mechanism will issue and store a personalized data table for every user that has been granted [explicit access permission](#) to the data connection. Note that data is not retrieved when the access permission are modified, but according to the defined data refresh interval.

The data personalization feature usage is different for the various data connection types that support it. For more information, please see the [data connection reference](#).

Parameterized Data Views

Define Parameters

Name	Type	Default Value	Edit /
@categoryCode	Text	RU	
@startTime	Date / Time	2009-01-01	
@endTime	Date / Time	2010-01-01	

Defining data view parameters.

Real-time views can use parameters to filter data before it is sent to a dashboard, which would otherwise retrieve the full set of data. These parameters can be of type Text, DateTime or Number. A default value can also be defined, for cases where the parameter is otherwise not set or available.

When a dashboard is built using a real-time view with parameters, the designer will provide the user with the option to bind the dashboard's filterable properties to these parameters.

Data Source Types

The following data source types are available to allow hub owners to provide data to dashboard authors from various sources.

Microsoft SQL Server

Accesses data stored in a Microsoft SQL Database.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **yes**

Supports personalization: **yes**

Supports real-time: **yes**

Properties:

- *Connection String* - The connection string to access the Microsoft SQL Database.

View properties:

- *Command Text* - The SQL query to perform. Must return a single table of data.

Microsoft SQL Server Analysis Services

Accesses data stored in a Server Analysis Services Server.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **yes**

Supports personalization: **yes**

Supports real-time: **yes**

Properties:

- *Connection String* - The connection string to access the Microsoft SQL Server Analysis Service. The connection string must access the [SSAS via HTTP](#). Please ensure the SASS instance has been configured for [IIS/MSMDPUMP](#).

View properties:

- *Command Text* - The MDX query to perform. Must return a single table of data. Any [Dimension hierarchy](#) which needs to be returned as part of the view must be expressed as a defined [Member](#) in the MDX Query.

Microsoft SharePoint List

Accesses data stored in a Microsoft SharePoint list.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **no**

Supports personalization: **no**

Supports real-time: **yes**

Properties:

- *Server* - The base URL of the [SharePoint 2010 REST service](#) or [SharePoint 2013 REST service](#).
- *Username* - The username of the SharePoint user accessing the list.
- *Password* - The password of the SharePoint user accessing the list.

View properties:

- *List Name* - The name of the SharePoint list to retrieve.

Microsoft SQL Azure

Accesses data stored in a Windows Azure SQL Database.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **yes**

Supports personalization: **no**

Supports real-time: **yes**

Properties:

- *Connection String* - The connection string to access the Windows Azure SQL Database instance.

View properties:

- *Command Text* - The SQL query to perform. Must return a single table of data.

OData (Open Data Protocol) Web Service

Accesses data returned from an online [OData](#) web service.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **no**

Supports personalization: **yes**

Supports real-time: **yes**

Properties: none

View properties:

- *OData view URL* - The URL of the OData web service.

Generic XML Web Service

Performs an HTTP request for a single XML document of data.

Has child data views: **no**

Supports client caching: **yes**

Supports parameterization: **no**

Supports personalization: **no**

Supports real-time: **yes**

Properties:

- *URL of XML Web Service* - The URL of the web service return the XML data.

Generic ODBC Data Source

Accesses data returned though a ODBC driver.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **yes**

Supports personalization: **yes**

Supports real-time: **yes**

Properties:

- *Connection String* - The ODBC driver-specific connection string. The underlying ODBC driver referenced by this connection must be installed on the Datazen Enterprise Server.

View properties:

- *Command Text* - The query to perform. Must return a single table of data.

Generic OLE DB Data Source

Accesses data returned though a ODBC driver.

Has child data views: **yes**

Supports client caching: **yes**

Supports parameterization: **yes**

Supports personalization: **yes**

Supports real-time: **yes**

Properties:

- *Connection String* - The OLE DB driver-specific connection string. The underlying ODBC driver referenced by this connection must be installed on the Datazen Enterprise Server.

View properties:

- *Command Text* - The query to perform. Must return a single table of data.

Excel documents on file system

Reads Microsoft Excel documents from a local folder or network share.

Has child data views: **no**

Supports client caching: **yes**

Supports parameterization: **no**

Supports personalization: **no**

Supports real-time: **no**

Properties:

- *Path on Server* - The fully qualified local or remote (UNC) path *from the server* to the folder containing Microsoft Excel documents. The folder or share must be accessible by the Datazen Enterprise Server account.

Managing Custom Maps

Map files (as used by the Datazen map controls) can be uploaded and stored on a hub via the Control Panel's *Custom Maps* page. These files are of the same type as the local version.

All Folders and Map Shapes

 New Folder...

Folder/Map	Edit	Delete
  Custom		
 california		
 ukraine		

Custom map listing.

Upload a map

1. Click on the **Upload Map Shape...* button.
2. Type a name for the map.
3. Select a shape (.SHP) and attribute (.dbf) file.
4. Click *Upload*; the dialog will close once the upload is complete.

The custom map is now available to users connected to the hub.

Removing a map

1. Navigate to the map via the folders if necessary.
2. Click the *Delete* icon for the map.
3. Click *Delete* in the presented dialog.

The map will now be removed from the system.

Moving a map

1. Navigate to the map via the folder if necessary.
2. Click the *Move* icon for the map.
3. Navigate to the destination folder in the presented dialog. Click *Move* when done.

The map will now appear in the new location.

Managing Hub Users & User Groups

As a publisher (or admin), one may administrate hub membership in Control Panel's *Users* page. User groups provide a way to apply permissions to sets of users, and can be administrated in the *User Groups* page.

Datazen Enterprise Server users are created via the Control Panel. At least one user must be added to the instance before any hubs can be created. Users must be added to hubs before they can access or publish dashboards. See the hubs documentation for more information.

Users can manage their profiles and passwords by logging in to the Control Panel.

The Admin User

When you first install Datazen Enterprise Server, a special *admin* user is created. This user has full administrative rights to perform any operation in the server instance, and is the only user that can create other users or hubs.

When you log in to the Control Panel as the *admin* user, you will see several options for managing and configuring the server instance that would otherwise not be available.

The Guest User

If created, the user with username *guest* has a special significance. The account with this name is used for public web access to dashboards. When maintaining a guest account, ensure that it can only access those dashboards which are intended to be viewed by anyone.

Users listing.				
#	Username	Email	Is Publisher	Is Owner
1	breon	breon@componentart.com	<input type="checkbox"/>	<input type="checkbox"/>
2	brian	brian@componentart.com	<input type="checkbox"/>	<input type="checkbox"/>
3	chris	chris@componentart.com	<input type="checkbox"/>	<input type="checkbox"/>
4	corey	corey@componentart.com	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Users listing.

Adding users to a hub

1. Navigate to the *Users* screen. Click *Add Users*.
2. Enter the usernames, delimited by space, comma, semicolon, or new line, of the users to be added to the current hub.
3. Click *Add Users to [current hub]*.

An e-mail will be sent to the user notifying them that they've been added to the hub.

Removing users from a hub

1. Navigate to the *Users* screen.
2. Find the user(s) to be removed, either through paging the table or by searching for them via the *Search Users* search box.
3. Click the *Remove User* icon next of the user(s).
4. Click *Remove* when prompted.

The user(s) will now be removed from the hub.

Assigning publishers

1. Navigate to the *Users* screen.
2. Find the user(s) to become a publisher, either through paging the table or by searching for them via the *Search Users* search box.
3. Click on the *Is Publisher* checkbox for that user(s).

The user(s) will now be a publisher for the hub.

Re-assigning hub ownership

1. Navigate to the *Users* screen.
2. Find the user to become the new owner, either through paging the table or by searching for them via the *Search Users* search box.
3. Click on the *Is Owner* checkbox for that user.
4. Click *Set Owner* when prompted.

The user will now be the hub owner. The previous owner will lose their ownership status.

Edit, delete, or create user groups:

User Group	Edit Group Name	Delete Group
Everyone	 Abc	
Marketing	 Abc	
Sales	 Abc	

User groups listing.

Creating a user group

1. Navigate to the *User Groups* screen. Click *Create New User Group*.
2. Enter a name for the group when prompted. The name should be unique against the other user groups in the hub.
3. Click *Create*.

The user group will be added to the hub, and should appear in the list.

Managing user group membership

1. Navigate to the *User Groups* screen.
2. Click *Users...* for the user group to be managed.
3. To add users to the user group, click their checkbox(es) in the table on the left and then click *Add*. To remove users from the user group, click their checkbox(es) in the table on the right and click *Remove*.

The users will be added/removed when the button is clicked, without prompting.

Removing a user group

1. Navigate to the *User Groups* screen.
2. Click the *Delete Group* icon for the user group.
3. Click *Delete* when prompted.

The user group will now be removed from the hub.

Renaming a user group

1. Navigate to the *User Groups* screen.
2. Click the *Edit Group* icon for the user group.
3. Update the name of the user group in the appropriate text box.
4. Click *Apply* when done.

The user group has now been renamed.

The Datazen admin may create or remove users entirely within the Control Panel. At least one user must be created before a hub may be created.

The screenshot shows the 'Create User' dialog box. It has a red header bar with the text 'Create User'. Below it is a section titled 'Required Info:' containing three input fields: 'Username' (with placeholder text 'User'), 'Email Address' (with placeholder text 'Email'), and 'Full Name' (with placeholder text 'Name'). Below this is a section titled 'Optional Info:' containing a single input field for 'Phone Number'. The entire form is contained within a light gray box.

Create User dialog.

Creating a user

1. Click the *Create User* button on the home page.
2. Fill out the user details form. Username, email and full name are required fields. The username and email address

must not already be in use.

Note: If the server is operating in Active Directory authentication mode, the username must match the user's Active Directory username.

3. Click *Create User*.

The user has now been added to the system.

- In default authentication mode, the user will receive an e-mail notification (see below). They must click the supplied link in order to complete their account creation process.
- In Active Directory authentication mode, the user can use the system as soon as they become a member of a hub.

Batch creating users

See [Batch Creating Users](#).

Deleting a user

1. Navigate to the *Server Users* screen.
2. Find the user to be deleted, either through paging the table or by searching for them via the *Search Users* search box. Click the *Delete User* icon next of the user.
3. Click *Delete* when prompted.

The user is now deleted from the system.

Managing the user profile

1. Log in as the user, if not already. Click the *Account* dropdown in the upper-right-hand corner.
2. Click *Edit* from the presented options.
3. Update text boxes as necessary.
4. Click *Update My Account*.

The user's profile is now updated.

Email notifications

Control Panel uses e-mail to notify users when they've been invited to the system, have been added to a hub, or require a new password. These e-mails use templates which can be edited in [Email Templates](#). The email system settings can be updated in [Control Panel's configuration](#).

Managing Permissions

Control Panel's *Permissions* page enables the hub owner to give or revoke edit permissions to:

- * Users and User Groups on Dashboards
- * Users and User Groups on Dashboard Groups
- * Users and User Groups on Data Connections
- * Users and User Groups on KPIs
- * Users and User Groups on Activity

The screenshot shows the Control Panel's Permissions page with the 'Dashboards' tab selected. At the top, there are four tabs: Dashboards, Data Sources, KPIs, and Activity. Below the tabs, there are two dropdown menus: 'Dashboards & Dashboard Groups:' set to 'Public Dashboards' and 'Set permissions for:' set to 'User Groups'. A large dark grey bar contains the text 'User Group'. Below this bar is a list of user groups: 'Everyone', 'Marketing', and another group whose name is partially visible. The entire interface has a clean, modern design with a light grey background and white text.

Permissions page.

Change permissions on a dashboard or dashboard group

1. Select the dashboard or dashboard group in the *Dashboards & Dashboard Groups* dropdown.
2. Update the permissions by toggling the *Allow Access* checkboxes. Use the second dropdown to switch between user groups and users.

Changes made are saved immediately.

Change permissions on a data connection

1. Click the *Data Sources* tab.
2. Select the data connection in the *Folders & Data Connections* dropdown.
3. Update the permissions by toggling the *Allow Access* checkboxes. Use the second dropdown to switch between user groups and users.

Changes made are saved immediately.

Change permissions on a KPI Group

1. Click the *KPIs* tab.
2. Select the KPI group in the *KPI Groups* dropdown.
3. Update the permissions by toggling the "Allow Access" checkboxes. Use the second dropdown to switch between user groups and users.

Changes made are saved immediately.

Change permissions on activity

1. Click the *Activity* tab.

2. Update the permissions by changing the *Access Level* dropdown.
3. *No Access* hides activity from the user/user group.
4. *Read Only* allows the user/user group to see activity.
5. *Full Access* allows the user/user group to add new comments, or update or remove their old comments. Use the second dropdown to switch between user groups and users.

Changes made are saved immediately.

Managing Custom Branding

Branding refers to the graphical resources used by the Datazen clients when drawing their user interface, such as the background image and color of buttons. Control Panel's *Branding* page allows the administrator to assign a possible default brand for each hub, as well as a default brand for the Datazen Server.

The screenshot shows the 'Branding' section of the Datazen Control Panel. At the top, there are tabs for 'Server/Hub' (selected), 'Branding', 'Upload', and 'Download'. Below the tabs is a table listing four branding packages:

Page	Name	Action
1	Server	
2	Corporate Overview	
3	Examples	
4	KPI Testing	

At the bottom of the table are navigation buttons: back, forward, and search. The text 'Page 1 of 1' is centered below the table. To the right, the text 'Branding listing.' is displayed.

Upload a branding

1. **Optional** Page to the hub if necessary.
2. Click on the *Upload* icon for the hub/server.
3. Click *Browse* to find and select the branding package on the local file system.
4. Click *Upload*; the dialog will close once the upload is complete.

The branding will now be applied by default to dashboards and clients connected to this hub/server.

Removing a branding

1. **Optional** Page to the hub if necessary.
2. Click the *Remove* icon for the branding.
3. Click *Delete* in the presented dialog.

The branding will now be removed from the system.

Download a branding

1. **Optional** Page to the hub if necessary.
2. Click the *Download* icon for the branding.
3. Accept the file from the browser's download dialog.

The branding package will now be available on the local file system.

Web Viewer

This section contains the following documents:

- [Accessing Dashboards and KPIs](#)
- [Configuring Public Access](#)
- [Configuring Integrated Windows Authentication](#)
- [Embedding Datazen Content Into Custom Apps](#)
- [Embedding Datazen Content Into SharePoint](#)

Accessing Dashboards and KPIs

The viewer is available by browsing the `/viewer` folder of Datazen Enterprise Server's base URL (e.g. `http://local.server.com/viewer`). To bring up a graphical shell with various frame options, use the `/viewer/shell` path (e.g. `http://local.server.com/viewer/shell`).

The Web Viewer is a secure portal and requires users to log in with valid Datazen Enterprise Server credentials. All dashboards and KPIs visible to the logged-in user are then presented for viewing.

The top navigation bar can be hidden by adding the parameter `?header=false` to the query string. It is also possible to link directly to the dashboards or KPIs lists, at `/viewer/dashboards` and `/viewer/kpis`, respectively.

It is also possible to filter the view by hub, group, and even down to an individual element. See bottom for examples of this.

Examples of tailored views

To link to public KPIs while omitting the navigation header:

`http://local.server.com/viewer/public/kpis?header=false`

To link to dashboards in the hub "Marketing" and the group "Regional Overviews":

`http://local.server.com/viewer/dashboards?hub=Marketing&group=Regional%20Overviews`

To link to a single specific KPI tile, and no navigation header:

`http://local.server.com/viewer/kpis?hub=Sales&group=Salespeople&name=Julie%20Tam`

To link directly to a named dashboard in a specific group:

`http://local.server.com/viewer/dashboard?hub=Marketing&group=Regional%20Overviews&name=Middle%20East`

To link to dashboards accessible to the logged-in Windows domain user:

`http://local.server.com/viewer/user/dashboards`

Full list of supported URL patterns

`/viewer`

`/viewer/Login`

`/viewer/Shell`

`/viewer/Home`

`/viewer/Dashboards`

`/viewer/Dashboards?hub=Hub&group=Group`

/viewer/Dashboard?hub=Hub&group=Group&name=Name

/viewer/KPIs

/viewer/KPIs?hub=Hub&group=Group

/viewer/KPI?hub=Hub&group=Group&name=Name

/viewer/public/Home

/viewer/public/Dashboards

/viewer/public/KPIs

/viewer/user/Home

/viewer/user/Dashboards

/viewer/user/KPIs

Configuring Public Access

If a [*guest account*](#) exists on the Datazen server, dashboards accessible to it can be viewed at `/viewer/public` (e.g. `http://server/viewer/public`).

If a URL linking directly to a dashboard's HTML5 view is shared, the *guest* account will be used, if available, to access it when no credentials are provided. Therefore to easily share a dashboard with a large audience:

1. Make it accessible to the *guest* user,
2. Get its URL from the `/viewer/public` view,
3. Email it to a list of recipients.

If a dashboard is not viewable by *guest*, it can still be shared but only to users who have permission to access it and are logged into the web portal. To link directly to a dashboard, the URL pattern is `http://local.server/viewer/dashboard?hub=HubName&group=GroupName&name=DashboardName`.

Integrated Windows Authentication

Automatic pass-through to the Web Viewer for authenticated Windows users is possible using Integrated Windows Authentication.

To enable this feature:

1. Ensure that Datazen Enterprise Server is configured to use Active Directory authentication.
2. Ensure that the Datazen Enterprise Server IIS application is running in the same domain as the users to be authenticated.
3. Enable both "anonymous" and "windows" authentication on the Datazen Enterprise Server IIS application. The latter may require installation of the IIS Windows Authentication feature through the Control Panel.

Clients on the network may now access `http://local.server.com/viewer/user` to view their dashboards. If clients are configured to include `http://local.server.com` in the "Intranet" zone under their local Internet Settings, they will pass through automatically without a login prompt.

To link directly to a specific dashboard, the URL pattern is `http://local.server.com/viewer/user/dashboard?hub=HubName&group=GroupName&name=DashboardName`. Note that this URL will only be accessible by others if they are authenticated on the domain and have access to the dashboard via their Datazen account.

Embedding Datazen Content Into Custom Apps

Dashboards and KPIs which are publically accessible, or are in an environment which supports [integrated Windows authentication](#) can be embedded into custom web apps. This is typically done by including an IFRAME element with a link to the desired Datazen content.

Index Views

To produce index views with tiles for each KPI and dashboard, use the below URL format.

Public access: server.com/viewer/public/home

User Content: server.com/viewer/user/home (via integrated Windows authentication)

Optional Parameters:

Parameter	Values	Meaning
active	kpis/dashboards	Which section to display by default
group	group name	Only show content from the KPI or dashboard group with the specified name
header	true/false	Whether to display the navigation header at the top
hub	hub name	Only show content from the given hub
name	item name	Only show the KPI or dashboard with the specified name

Dashboards

To directly include an interactive dashboard, use the below URL format.

Public access: server.com/viewer/public/dashboard

User Content: server.com/viewer/user/dashboard (via integrated Windows authentication)

Parameters:

Parameter	Values	Meaning
group	group name	The name of the dashboard group in which the dashboard is found (optional)
hub	hub name	The name of the hub in which the dashboard is found (optional)
name	item name	The name of the dashboard (required)

Embedding Datazen Content Into SharePoint

1. Click *Site Contents*;

Sharepoint Demo

EDIT LINKS

Home

Documents

Recent

Levy's List

Operations Dashboard

Site Contents

EDIT LINKS

Corporate Scorecards

European Division

Scorecard 1

Scorecard 2

Scorecard 3

2. Click *Site Pages*;

Site Contents

Lists, Libraries, and other Apps

add an app

Documents
0 items
Modified 5 months ago

Form Tem
0 items
Modified

Style Lib
5 items
Modified

Site Assets
0 items
Modified 5 months ago

Site Pages
3 items
Modified 3 hours ago

3. Click *New Wiki Page*;

SharePoint

BROWSE FILES LIBRARY

Datazen Sharepoint Demo EDIT LINKS

Site Pages

+ new Wiki page

All Pages By Author By Editor ... Find a file

Home Documents Recent Levy's List Operations Dashboard Site Contents

Name	Modified By	Modified	Created By
Home	System Account	3 hours ago	System Account
Operations Dashboard	System Account	4 days ago	System Account
How To Use This Library	System Account	June 21	System Account

4. Give the page a name;
5. Click *Create*;
6. Click the *Insert* tab and then *Embed Code*;

SharePoint

BROWSE PAGE FORMAT TEXT INSERT

Tables Media Video and Audio Links Upload File App Part Web Part Embed Code

Home Documents Recent Levy's List

7. In the embed code add an iframe of the desired size with the appropriate SRC attribute, for example:

```
<iframe width="1280" height="768" src="http://demo-server.datazen.com/viewer/public/home?active=dashboards&header=false&bg=fff"></iframe>
```

8. Click *Insert*;



9. Save the page.

Server Maintenance

This section contains the following documents:

- [Windows Services, IIS Sites and Application Pools](#)
- Configuration Settings
 - [Server Core Settings](#)
 - [Rendering Service Settings](#)
 - [Control Panel Settings](#)
 - [Web API Configuration](#)
- [Batch Creating Users](#)
- [Server Logs](#)
- [Email Templates](#)
- [Backup and Restore](#)

Windows Services, IIS Sites and Application Pools

Datazen Server Service

The Datazen Server Service is a Windows service that is responsible for storing dashboards, data and security information. This service is a critical component of the Datazen Enterprise Server and the environment will not function if the service is not running.

When Datazen Enterprise Server is first installed, the Windows service datazen is created and configured as per the options chosen in the installer. It is possible to configure the service settings after the service has been installed provided the datazen service is not running.

Configuration for the service is via the Server Service .config file.

Datazen Rendering Service

The Windows service listens for requests from the Datazen Rendering Web Application. Upon receiving a request, the assets required to render the dashboard are fetched from the Datazen Server Service, rendered by one of the viewer processes available in the rendering pool and then passed back to the Datazen Rendering Web Application.

Configuration for the service is via the Rendering Service .config file.

Datazen Rendering Web Application

The web application accepts HTTP requests for dashboards and their assets. It forwards these requests to the Datazen Rendering Service and returns the appropriate HTML, PDF or image content to the client. For HTML requests, images are generated and cached on the server and subsequently downloaded by the client.

Configuration for the web application is via the Datazen Rendering Web Application web.config file.

Control Panel Web Application

An administrative web site for managing various aspects of the Datazen Enterprise Server deployment.

Configuration for the web application is via the Datazen Control Panel web.config file.

Web API

A REST API to which the various Datazen clients and viewers connect to. The Web API application contains the Web Viewer.

Configuration for the web application is via the Datazen Web API web.config file.

Configuration Settings

This section contains the following documents:

- [Server Core Settings](#)
- [Rendering Service Settings](#)
- [Control Panel Settings](#)
- [Web API Configuration](#)

Server Core Settings

The Datazen server configuration settings are set during the installation procedure. The configuration may be modified at any time by stopping the Datazen Enterprise Server service and editing the *Datazen.Server.Service.config* file located in the *\service* sub-folder of the installation folder.

Configuration settings

datapath

The location of the database folder. If this value is preceded with a tilde (~) the location is relative to the *\service* sub-folder of the installation folder.

Location:

```
<configuration>
  <appSettings>
    <add key="datapath" value="[path]" />
```

Default value: ~\Data

dataport

The TCP port used for the data REST service.

Location:

```
<configuration>
  <appSettings>
    <add key="dataport" value="[port]" />
```

Default value: 28952

apiport

The TCP port used for the data REST management service.

Location:

```
<configuration>
  <appSettings>
    <add key="apiport" value="[port]" />
```

Default value: 28953

authtimeout

The duration (in seconds) to keep remote clients logged in before they must re-authenticate.

Location:

```
<configuration>
  <appSettings>
    <add key="authtimeout" value="[value]" />
```

Default value: 900

authtype

The authentication mode to use for servicing Datazen Enterprise Server login requests. May have one of the following values:

- default - users are authenticated using their Datazen Enterprise Server credentials
- ad - users are authenticated using Active Directory.

Location:

```
<configuration>
  <appSettings>
    <add key="authtype" value="[default|ad]" />
```

Default value: default

ad_server

In Active Directory authentication mode, the server name and port number of the LDAP service. Typically this will be the host name of a domain controller, e.g. dc1.corp.myorg.com:389.

Location:

```
<configuration>
  <appSettings>
    <add key="ad_server" value="[server:port]" />
```

ad_domain

In Active Directory authentication mode, the fully qualified domain name, e.g. corp.myorg.com.

Location:

```
<configuration>
  <appSettings>
    <add key="ad_domain" value="[value]" />
```

ad_username

In Active Directory authentication mode, the username of an account that will perform the AD query.

Location:

```
<configuration>
  <appSettings>
    <add key="ad_username" value="[username]" />
```

ad_password

In Active Directory authentication mode, the password of the account specified in *ad_username*.

Location:

```
<configuration>
  <appSettings>
    <add key="ad_password" value="[password]" />
```

can_hub_owners_create_users

When set to 'false', the server will run in "Enterprise Mode" and hub owners cannot directly add users; they may only invite them. When set to 'true', the server will run in "Demo Mode" and hub owners can add users to a hub.

Location:

```
<configuration>
  <appSettings>
    <add key="can_hub_owners_create_users" value="[true|false]" />
```

rundataupdates

When set to false pauses all [remote data fetching](#). Location:

```
<configuration>
  <appSettings>
    <add key="rundataupdates" value="[true|false]" />
```

Default value: true

HTML Rendering Settings

The Datazen HTML rendering feature allows published dashboards to be rendered to HTML, PDF or image formats. It consists of two parts, a web application and a Windows service. While not a required component of the Datazen Enterprise Server, installing the Datazen Rendering Service will allow users to [view dashboards in web browsers](#) as well as utilize Datazen mobile device applications (Android, iOS, Windows Phone).

Datazen Rendering Web Application

The [web application](#) accepts HTTP requests for dashboards and their assets. It forwards these requests to the Datazen Rendering Service and returns the appropriate HTML, PDF or image content to the client. For HTML requests, images are generated and cached on the server and subsequently downloaded by the client.

Datazen Rendering Service

The [Windows service](#) listens for requests from the Datazen Rendering Web Application. Upon receiving a request, the assets required to render the dashboard are fetched from the Datazen Server Service, rendered by one of the viewer processes available in the rendering pool and then passed back to the Datazen Rendering Web Application.

Datazen Rendering Service Configuration

The Datazen Rendering Service Configuration file (Datazen.Rendering.Service.config) is located in the `\renderer\svc` sub-folder of the installation folder. It provides the following configurable settings:

Datazen Server REST location

The location of the REST service (data) of the Datazen Server. This setting should not be modified unless the address or port of the Datazen Server machine is changed.

Location:

```
<configuration>
  <connectionStrings>
    <add name="datazenserver" connectionString="[value]" />
```

Default value: `Url=http://datazenserver:28952`

ProcessPoolSize

Number of DashboardViewer processes to maintain. For production set this to the logical CPU count - 1

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ProcessPoolSize" serializeAs="String">
```

```
<value>[int]</value>
```

Default value: 1

ProcessTimeOut

Time afforded to DashboardViewer to complete a request.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ProcessTimeOut" serializeAs="String">
        <value>[hh:mm:ss]</value>
```

Default value: 00:00:35

KillZombiesAtStart

Kill all {DashboardViewerFilename} processes on service start.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="KillZombiesAtStart" serializeAs="String">
        <value>[true|false]</value>
```

Default value: true

ProcessKillMemoryLimit

Size in Mb at which a viewer process will be killed and replaced.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ProcessKillMemoryLimit" serializeAs="String">
        <value>[int]</value>
```

Default value: 984

ResponseMemoryFileSize

Size in Kb for the response MemoryMappedFile (images & html).

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ResponseMemoryFileSize" serializeAs="String">
        <value>[int]</value>
```

Default value: 2048

DashboardViewerProcessName

System name for the Viewer process.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="DashboardViewerProcessName" serializeAs="String">
        <value>[value]</value>
```

Default value: DashboardViewer

SocketPort

The port the service listens on for incoming requests from the server.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="SocketPort" serializeAs="String">
        <value>[int]</value>
```

Default value: 9000

SocketBufferSize

The size in bytes of the socket buffer.

Location:

```
<configuration>
  <applicationSettings>
```

```
<Datazen.Rendering.Service.Properties.Settings>
    <setting name="SocketBufferSize" serializeAs="String">
        <value>[int]</value>
```

Default value: 1048576

SocketMaxConnections

Maximum number of simultaneous connections the server supports.

Location:

```
<configuration>
    <applicationSettings>
        <Datazen.Rendering.Service.Properties.Settings>
            <setting name="SocketMaxConnections" serializeAs="String">
                <value>[int]</value>
```

Default value: 1000

RequestMemoryFileSize

Size in Kb for the request MemoryMappedFile.

Location:

```
<configuration>
    <applicationSettings>
        <Datazen.Rendering.Service.Properties.Settings>
            <setting name="RequestMemoryFileSize" serializeAs="String">
                <value>[int]</value>
```

Default value: 16192

DashboardViewerPath

Absolute path to the DashboardViewer release executable.

Location:

```
<configuration>
    <applicationSettings>
        <Datazen.Rendering.Service.Properties.Settings>
            <setting name="DashboardViewerPath" serializeAs="String">
                <value>[path to viewer executable]</value>
```

Default value: set by installer

ServiceTimerInterval

Milliseconds interval to check the Datazen Server for requests.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ServiceTimerInterval" serializeAs="String">
        <value>[int]</value>
```

Default value: 3

ConsoleTimerInterval

Interval to check the request queue for requests.

Location:

```
<configuration>
  <applicationSettings>
    <Datazen.Rendering.Service.Properties.Settings>
      <setting name="ConsoleTimerInterval" serializeAs="String">
        <value>[int]</value>
```

Default value: 3

Datazen Rendering Web Application

The Datazen Rendering Web Application is responsible for fulfilling dashboard rendering requests from the non-Windows Datazen viewers (Windows Phone, iOS, Android, browser) in HTML, PDF, or image formats.

Communication with other components

The Datazen Rendering Web Application communicates with the Datazen Rendering Service over UDP port 9000. If the Datazen Rendering Service is located on a remote machine or network, this port must be made available for inbound and outbound traffic to the server hosting the Datazen Rendering Service.

Extra Datazen Rendering Service instances may be run on other machines and can be provisioned during runtime by editing the web.config file of the Datazen Rendering Web Application.

Changing the communication UDP port

If the default communication UDP is changed both the Datazen Rendering Service Configuration file and in the Datazen Rendering Web Application Configuration file must be modified.

Datazen Rendering Web Application Configuration

The Datazen Rendering Web Application Configuration file (web.config) is located in the /renderer/web sub-folder of the installation.

renderer

Address of a Datazen.Rendering.Service. There can be multiple renderer values defined, each must start with the word *renderer*. Requests are distributed in a round-robin fashion.

Changing these values in a running server causes the collection to be refreshed.

Location:

```
<configuration>
  <appSettings>
    <add key="renderer[id]" value="[address:port]" />
    ...
  </appSettings>
</configuration>
```

Default value: 127.0.0.1:9000

Example:

```
<add key="renderer01" value="127.0.0.1:9000" />
<add key="renderer02" value="render02.somedomain.com:9000" />
<!-- if both defined, these two rendering servers will be used in a round-robin fashion -->
```

nojsimagesize

In the absence of either height or width parameters, use this size as the default for non-Javascript enabled clients.

Location:

```
<configuration>
  <appSettings>
    <add key="nojsimagesize" value="[widthxheight]" />
  </appSettings>
</configuration>
```

Default value: 640x480

ajaxverb

Verb to use for AJAX loading.

Location:

```
<configuration>
  <appSettings>
    <add key="ajaxverb" value="[GET|POST]" />
  </appSettings>
</configuration>
```

Default value: POST

cache.enabled

Enables or disables caching. Note that even when caching is disabled, images are still written to cache so that they can be retrieved via the ImageCacheHandler; they are deleted upon retrieval.

Location:

```
<configuration>
  <appSettings>
    <add key="cache.enabled" value="[true|false]" />
```

Default value: false

cache.optipng

When optipng.exe is in the */bin* directory of the Datazen Rendering Web Application then PNG compression is enabled for file-type caching. This setting is the command parameters passed to optipng.exe.

Please refer to the [optipng documentation](#) for more information.

Location:

```
<configuration>
  <appSettings>
    <add key="cache.optipng" value="[optipng parameters]" />
```

Default value: -preserve -silent -o1

cache.type

The cache storage mechanism to use. Currently, only *file* is available.

Location:

```
<configuration>
  <appSettings>
    <add key="cache.type" value="[value]" />
```

Default value: file

cache.interval

The default cache interval in seconds. The default value when none is specified is 1200 (20 minutes).

Location:

```
<configuration>
  <appSettings>
```

```
<add key="cache.interval" value="[seconds]" />
```

Default value: 604800

cache.maxsize

The maximum size in megabytes of the cache before older (and potentially unexpired) items are removed. A value of *0* will disable unexpired content from being removed as the cache grows. The default value when none is specified is *0* (disabled).

Location:

```
<configuration>
  <appSettings>
    <add key="cache.maxsize" value="[size in megabytes]" />
```

Default value: 750

cache.virtualpath

The virtual path to the ImageCacheHandler. All images are requested via this path.

Location:

```
<configuration>
  <appSettings>
    <add key="cache.virtualpath" value="[value]" />
```

Default value: vcache

Example:

```
<add key="cache.virtualpath" value="vcache" />
<!-- http://server/vcache/image.png -->
```

cache.file.path

The path where cached content is saved to disk. The path can be absolute or relative to the Datazen Rendering Web Application installation location.

Location:

```
<configuration>
  <appSettings>
    <add key="cache.file.path" value="[path to cache folder]" />
```

Default value: cache

Examples:

```
<add key="cache.file.path" value="cache" /> <!-- relative -->
<add key="cache.file.path" value="c:\cache" /> <!-- absolute -->
<add key="cache.file.path" value="\server\share" /> <!-- remote -->
```

Control Panel Settings

The Control Panel configuration settings are set during the installation procedure. The configuration may be modified at any time by editing the *Web.config* file located in the *\controlpanel* sub-folder of the installation folder.

Datazen Enterprise Server connectionString

The location of the REST service (data) of the Datazen Enterprise Server. This setting should not be modified unless the address or port of the Datazen Enterprise Server machine is changed.

Location:

```
<configuration>
  <connectionStrings>
    <add name="datazenserver" connectionString="[value]" />
```

Default value: Url=http://localhost:28952

SMTP from address

The sender address for all email notifications sent from the Control Panel.

```
<configuration>
  <system.net>
    <mailSettings>
      <smtp from="[smtp email address]">
```

Default value: no-reply@datazen.com

SMTP host name/port

The host name (or IP address) and port of the SMTP server to use for sending email notifications from the Control Panel.

```
<configuration>
  <system.net>
    <mailSettings>
      <smtp>
        <network host="[smtp host name]" port="[smtp port]" />
```

Default Value: <network host="localhost" port="25" />

Web API configuration

The Datazen Web API is a REST web service to which the various Datazen clients and viewers connect to. The Web API configuration settings are set during the installation procedure. The configuration may be modified at any time by editing the *Web.config* file located in the *\webapi* sub-folder of the installation folder.

Datazen Server connectionString

The location of the REST service (data) of the Datazen Server. This setting should not be modified unless the address or port of the Datazen Server machine is changed.

Location:

```
<configuration>
  <connectionStrings>
    <add name="datazenserver" connectionString="[value]" />
```

Default value: Url=http://localhost:28952

Batch Creating Users

To create more than one user at a time, a CSV ("comma-separated values") file containing the user details for a set of new users can be uploaded. The CSV file must contain a header row and one or more data rows. The header row/column values are:

- Email (mandatory)
- Username (mandatory)
- Name (mandatory)
- PhoneNumber (optional)
- Company (optional)
- Position (optional)

e.g.

```
Email,Username,Name  
michael.z.madsen@gmail.com,mmadsen,Michael Z. Madsen  
linda.b.lacelove,llacelove,Linda B. Lacelove
```

The screenshot shows a red dialog box titled "Batch Create Users". At the top, there is a label "Select the CSV file:" followed by a file input field. To the right of the input field are two buttons: "Browse" (in red) and a question mark icon. Below the input field are two large buttons: "Upload" (in red) and "Cancel" (in grey). The entire dialog is set against a white background with a dark grey header bar.

The Batch Create Users dialog.

1. Navigate to *Home*. Click on *Batch Create Users*.
2. Click on *Browse* in the dialog that appears; select the file to be uploaded.
3. Click *Upload*.
4. Click *OK*.

The users have now been added to the system.

Server Logs

Many of the various elements of Datazen Enterprise Server log error, trace and debug events in their own log files on the server machine. They can be accessed directly on the server or, for some of them, viewed via a Control Panel interface.

Service and Web API Logs

Server Service logs are stored under `\service\logs` in the Datazen Enterprise Server installation folder. They record notable events related to the operation of the service and roll over daily.

Web API logs can be found under `\webapi\logs` in the Datazen Enterprise Server installation folder.

Events of "debug" or higher priority can also be viewed by the server Admin user in the [Control Panel](#) under the Log Viewer option.

Renderer Logs

Each element of the rendering system maintains its own set of logs. These can be found, respectively, under

`\renderer\svc\logs` for the Rendering Service,

`\renderer\viewer\logs` for the rendering system's internal viewer element, and

`\renderer\web\logs` for the Rendering Web Server.

E-mail Templates

Control Panel's *Email Templates* page is used to edit the subject and body of e-mails that are sent to users and prospective users.

- The "Add User" template is used for e-mails sent to existing users (i.e. have already registered with the system), notifying them that they have been added to a Hub.
- The "Invite User" template is used for e-mails sent to new users (i.e. have not yet registered), to begin their registration process.
- The "Forgot Password" template is used for e-mails sent to existing users who have forgotten their password and need a new one.

The screenshot shows the Control Panel's Email Templates page. At the top, there are three tabs: 'Add User' (highlighted in red), 'Invite User', and 'Forgot Password'. Below the tabs, the 'Add User email template' is selected. The page is divided into two main sections: 'Subject' and 'Body'. The 'Subject' section contains the placeholder text 'Welcome to the '[HubName]' Hub!'. The 'Body' section contains the following text: '[AdderName] has added you to the '[HubName]' dashboard hub. You can use your existing login information to access this hub.' and '[AddLink]'. The left side of the page features a vertical sidebar with a red and grey gradient.

The Email Templates page.

Update the Add User template

1. Navigate to *Email Templates*.
2. Update the subject and body text. You can define where the user's name, the hub name, and link appear using this syntax:
 - 3. [AdderName]: The name of the member who added the recipient.
 - 4. [AddLink]: The URL that the recipient should visit in order to continue the adding process.
 - 5. [HubName]: The name of the hub.
6. Click "Update Templates" when done. To cancel your changes navigate to another section of the Control Panel site.

The Add User template has now been updated, and subsequent e-mails sent to added users will use this version.

Update the Invite User template

1. Navigate to *Email Templates*.
2. Click the *Invite User* tab.
3. Update the subject and body text. You can define where the user's name, the hub name, and link appear using this syntax:
 - 4. [InviterName]: The name of the member who invited the recipient.
 - 5. [InviteLink]: The URL that the recipient can visit to see the hub.
 - 6. [HubName]: The name of the hub.

7. Click "Update Templates" when done. To cancel your changes navigate to another section of the Control Panel site.

The Invite User template has now been updated, and subsequent e-mails sent to invited users will use this version.

Update the Forgot Password template

1. Navigate to *Email Templates*.
2. Click the *Forgot Password* tab.
3. Update the subject and body text. You can define where the link appears using this syntax:
4. [ForgotPasswordLink]: The URL that the recipient should visit in order to continue the password reset process.
5. Click "Update Templates" when done. To cancel your changes navigate to another section of the Control Panel site.

The Forgot Password template has now been updated, and subsequent e-mails sent to users will use this version.

Backup and Restore

The Datazen service contains an embedded database that should be periodically backed up for disaster recovery. Backing up the database can be performed while the Datazen service is online. Control Panel's *Backup* page can be used to manage and perform system backups. Performing a backup will send a request to the Datazen Enterprise Server to read and store all Datazen database data:

- Dashboards and dashboard groups
- Users and user groups
- Data connections and data views
- Permissions

The backup will be created in a specified directory; the folder will be created if it does not already exist. If the path is a relative one, the backup will be created relative to the Datazen Enterprise Server's "datopath" folder.

Home
Configuration
Server Users
Email Templates
Branding
Log Viewer
Backup
Hub: ComponentArt

Status Schedule

Backup running: **false**
Last backup started: --
Last back finished: --

Backup directory (e.g. C:\Backups):
Please enter a folder path

Backup Now

Backup and restore page in Control Panel.

Performing a backup

1. Enter a directory to write the backup files to.
2. Click *Backup Now*.
3. Click *Backup* when prompted to do so.

A backup request has now been sent to the Datazen Enterprise Server. The status of the backup can be monitored on this screen, as well as in the Log Viewer.

Scheduling a backup

1. Click on the *Schedule* tab.
2. Check the *Scheduled backup enabled* checkbox.
3. Define the schedule for the backup in the dropdowns provided.
4. Enter a directory to write the backup files to.
5. Click *Save Settings*.

A backup has now been scheduled to occur. If you make some changes but want to start over, clicking *Load Settings* will reset the form to its last saved settings.

Restoring a backup

Restoring the database is an *offline* operation, meaning the Datazen service **must be stopped** before a restore is attempted.

The database restore utility is the command-line executable *Raven.Database.exe* located in the *\service* sub-folder of the installation folder. The format of the command to perform a restore is:

```
Raven.Database.exe -src [path to backup folder] -dest [path to data folder] -restore
```

The default *data folder* is the *\service\Data* sub-folder of the installation folder. Restoring over an existing database is not allowed, therefore **the data folder must be empty** in order for the restore operation to succeed.

Once the restore operation is finished, the Datazen service may be restarted.

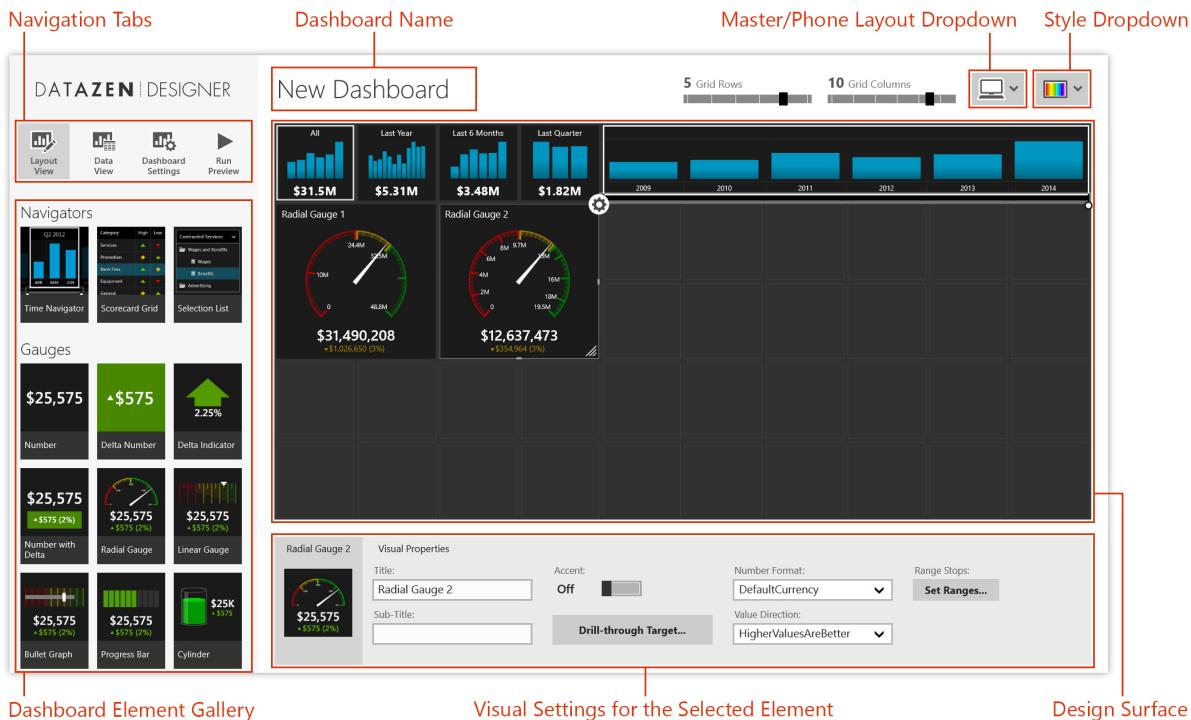
Creating and Publishing Dashboards

This section contains the following documents:

- [Dashboard Designer](#)
- [Dashboard Runtime](#)
- [Data](#)
- [Configuring Navigators](#)
- [Configuring Visualizations](#)
- [Saving & Publishing Dashboards](#)
- [Advanced Topics](#)

Dashboard Designer

Datazen Designer enables quick creation of dashboards that scale well to any screen size. The design paradigm is based on a design surface with adjustable grid rows and columns and flexible dashboard elements that adjust perfectly to any amount of real estate. The following diagram shows all the components of the Datazen Designer layout view:



Data-First vs. Design-First

When creating dashboards, two basic approaches may be taken: data-first, and design-first. Datazen Designer supports both. The data-first approach is to import all required data first and then move next to designing the dashboard and setting data properties on the dashboard elements. This has the advantage of being able to connect each element to real data when it is added to the layout. When using a data-first approach one should be sure that their real data is formatted correctly for use with Datazen.

The design-first approach, on the other hand, is to create a dashboard layout first without importing any data. This is a good way to create a dashboard when you are not sure if the data is formatted correctly. Without real data gallery elements will automatically be bound to generated simulated data, which can be exported and used as a template to describe the required data. The following diagram shows all the components of the Datazen Designer data view:

Dashboard Control Instances

Data Commands

New Dashboard

Control Instances

Selected Data View

Data Settings for the Selected Element

Data Views

The screenshot shows the Datazen Designer application. At the top, there's a toolbar with 'Add Data', 'Refresh All Data', and 'Export All Data' buttons. Below the toolbar is a table with columns: Date, Metric1, Metric2, Metric3, Metric4, Metric5, and Comparison. The table contains 15 rows of data from January 1, 2009, to March 1, 2010. A red box highlights the first three rows of the table. Below the table is a navigation bar with tabs: SimulatedTable, ExpenseData, and SalesData. A red box highlights the 'ExpenseData' tab. On the left, there's a sidebar with icons for Layout View, Data View, Dashboard Settings, and Run Preview. A red box highlights the 'Data View' icon. The main area displays several controls: a bar chart labeled 'Time Navigator', two radial gauges labeled 'Radial Gauge 1' and 'Radial Gauge 2', and a table. A red box highlights the 'Radial Gauge 2' control. A detailed view of its data properties is shown in a modal window. The modal has tabs for Radial Gauge 2 and Data Properties. Under Data Properties, it shows 'Main Value:' set to 'SimulatedTable' and 'Metric2', and 'Comparison Value:' set to 'SimulatedTable' and 'Comparison2'. Both have dropdown menus and 'Options...' buttons.

	Date	Metric1	Metric2	Metric3	Metric4	Metric5	Comparison:
1	1/1/2009 12:00:00 AM	305,272.80	121,034.46	248,877.24	185,956.32	122,057.47	282,707.42
2	2/1/2009 12:00:00 AM	239,047.20	96,668.08	191,059.96	145,494.46	96,429.71	282,707.42
3	3/1/2009 12:00:00 AM	250,192.80	100,829.67	196,013.32	153,011.16	99,255.11	282,707.42
4	4/1/2009 12:00:00 AM	267,127.20	107,521.72	213,431.07	162,170.91	107,604.90	274,442.79
5	5/1/2009 12:00:00 AM	297,712.80	117,382.72	244,408.86	182,033.62	120,205.43	274,442.79
6	6/1/2009 12:00:00 AM	337,672.80	136,690.33	277,815.41	202,221.60	133,233.59	274,442.79
7	7/1/2009 12:00:00 AM	417,960.00	166,430.08	332,790.26	252,506.43	167,548.08	342,899.06
8	8/1/2009 12:00:00 AM	393,832.80	153,747.60	317,831.94	238,400.46	157,910.30	342,899.06
9	9/1/2009 12:00:00 AM	335,512.80	131,290.30	269,042.81	199,078.06	132,191.51	342,899.06
10	10/1/2009 12:00:00 AM	301,687.20	122,029.85	247,326.81	181,429.69	121,394.45	320,221.42
11	11/1/2009 12:00:00 AM	251,272.80	103,350.00	201,780.73	151,493.69	99,780.60	320,221.42
12	12/1/2009 12:00:00 AM	275,032.80	107,892.67	222,933.39	159,903.24	109,937.91	320,221.42
13	1/1/2010 12:00:00 AM	476,992.80	188,244.59	372,660.29	285,187.67	193,662.25	371,550.00
14	2/1/2010 12:00:00 AM	222,480.00	90,155.13	177,915.69	134,875.50	88,383.83	371,550.00
15	3/1/2010 12:00:00 AM	267,472.80	105,866.55	208,293.86	159,517.42	107,242.80	371,550.00

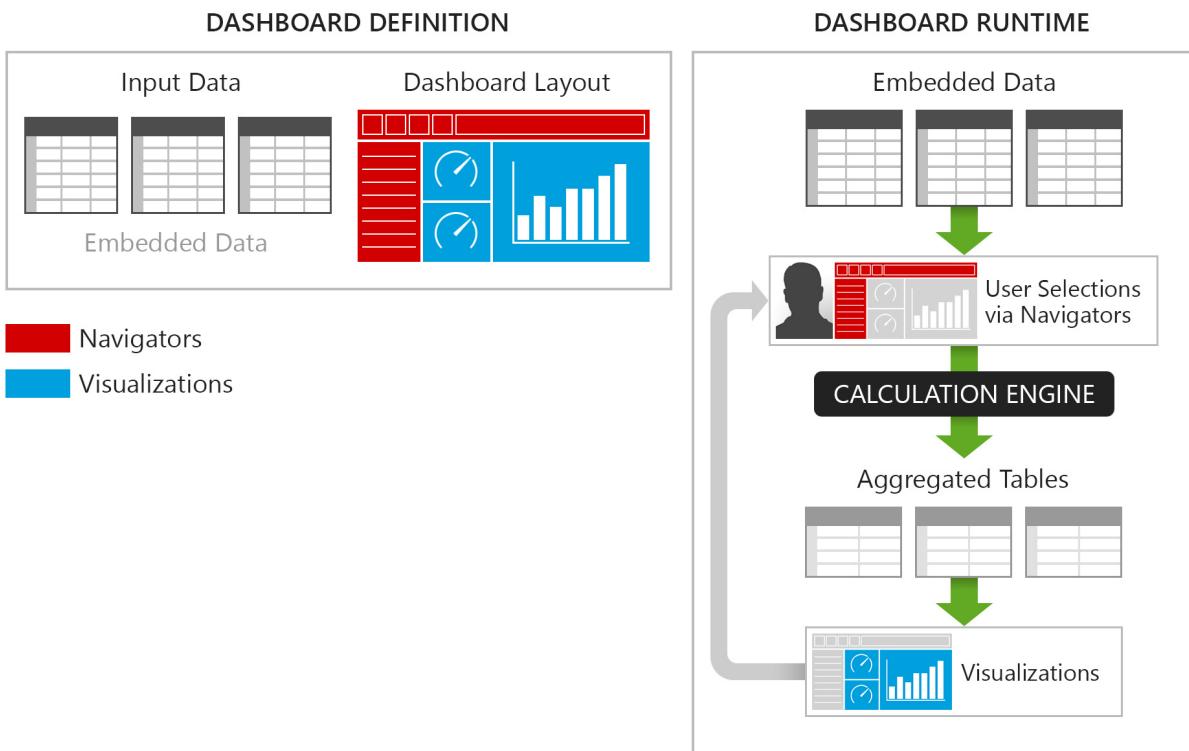
Dashboard Runtime

The purpose of the Datazen dashboard runtime is to instantiate a dashboard, load the required data and process user actions. Processing user actions – like filter or date/time selections – involves aggregating the original data updating all visual elements of the dashboard. The Datazen dashboard runtime can operate in three different modes:

1. Disconnected mode
2. Connected mode
3. Load on Demand mode

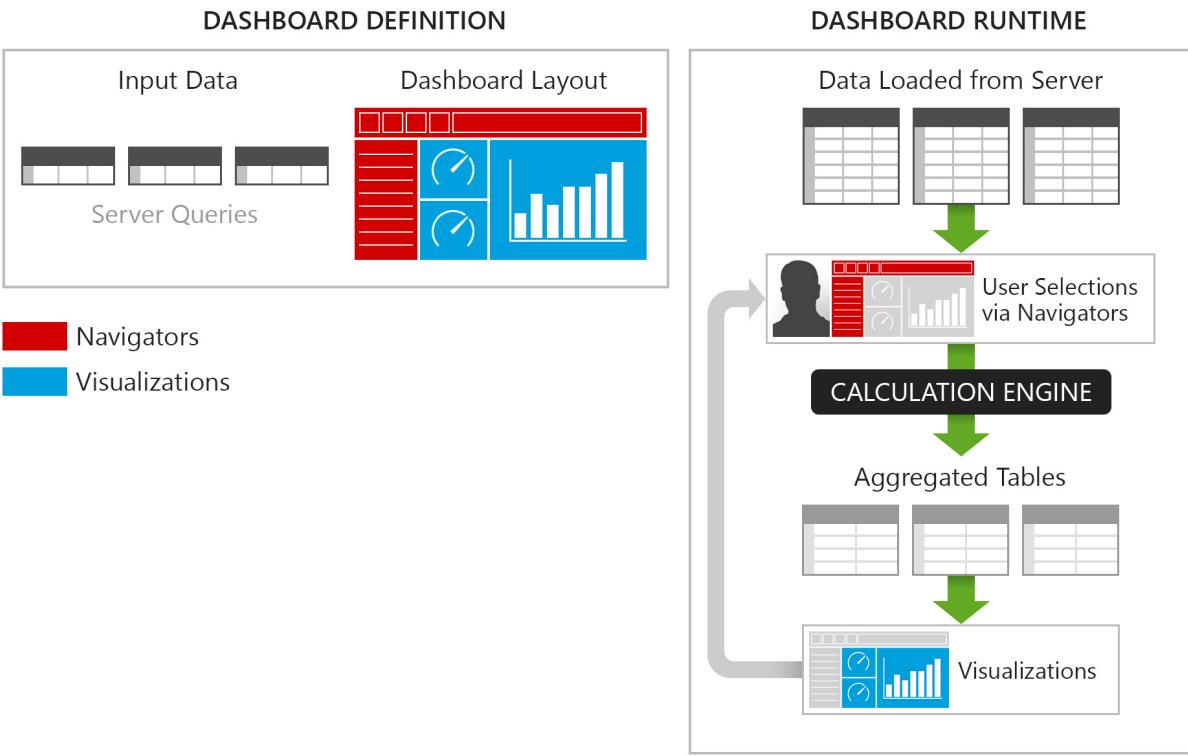
Disconnected Mode

In the disconnected mode, the device running Datazen is not connected to the server. The device may actually be completely disconnected from the network, or running in “airplane mode”. In this mode, the runtime loads the dashboard definition (containing offline data) instantiates the dashboard and processes user actions by performing filtering and date/time aggregations on the client:



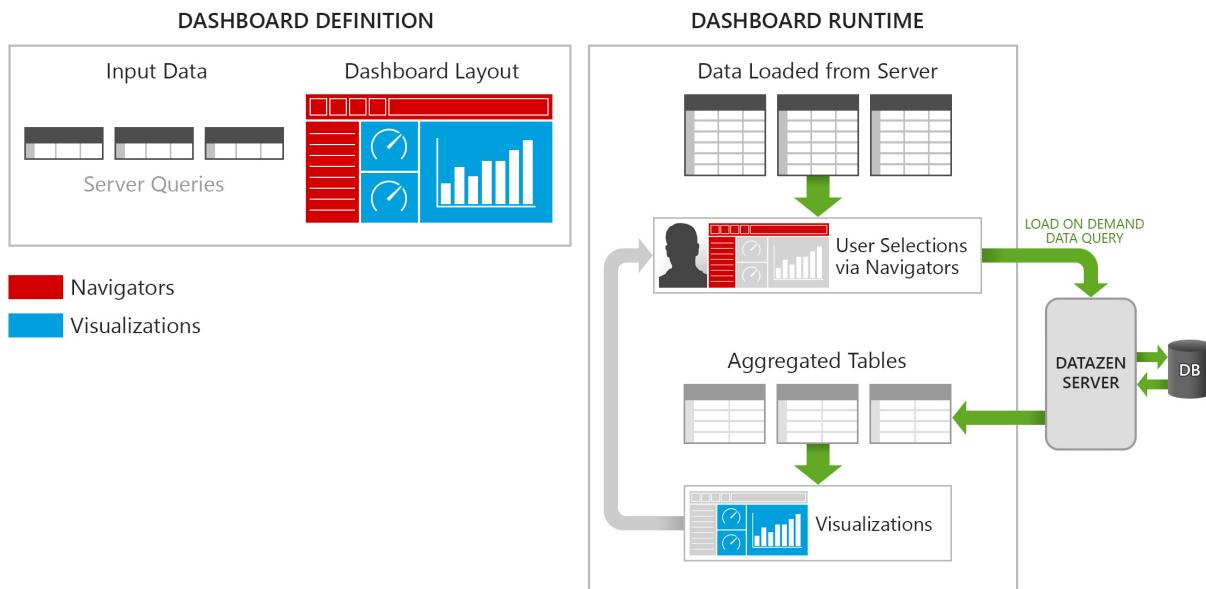
Connected Mode

In the connected mode, the device running Datazen is connected to the Datazen Enterprise Server. After instantiating a dashboard, the runtime loads fresh data views from the server. From that point on, all user actions are handled on the client, same as in the disconnected mode.



Load on Demand Mode

The load on demand mode is appropriate when working with very large datasets. In this mode, the device running Datazen is connected to the Datazen Enterprise Server. After instantiating a dashboard, the runtime loads fresh data views from the server. Following that, the runtime handles user actions by issuing new requests to the server – passing selection parameters to parameterized data views – and sending the aggregated results to visualization elements on the dashboard.



Data

This section contains the following documents:

- [Data Model](#)
- [Preparing Dashboard Data](#)
- [Working with Excel Data](#)
- [Working with Live Data Sources](#)
- [Working with Simulated Data](#)

Data Model

Datazen data model is extremely simple. Data is imported into the Designer as a collection of "data views". Formal relationships between data views are not required. Lookups from one data view to another will work as long as the key values match. Date/time aggregations are handled by the dashboard runtime and they will match between different data views, even if the date/time data granularity is different between the data views.

There are two types of data views that can be imported:

- Local Excel data:** The user selects an Excel document and picks which worksheet(s) will be imported. Once imported, the data is stored within the dashboard definition. To refresh the data from the original Excel file, the Refresh Data command must be used in the designer data view;
- Datazen Enterprise Server data:** The user browses the list of data views that have been published on the server and selects the ones to be added to the dashboard. Supported data types include: SQL queries, MDX queries, SharePoint lists, web services, generic ODBC data sources and others. Dashboards published based on server data will always stay connected to the original server data view(s) and reflect the latest state of the data on the server.

Once imported into the Designer, data views are data type agnostic: they are treated equally regardless of their origin. The rest of the dashboard creation and design experience is the same regardless of where the data came from.

Connecting Dashboard Elements to Data

Each Datazen dashboard element contains one or more data settings. For example, the Radial Gauge element contains two data settings: Main Value and Comparison Value. Each of these settings points to exactly one field (column) in a specific data view:

The screenshot shows the Datazen Designer interface with a 'Corporate Scorecard' dashboard. The top navigation bar includes 'Add Data', 'Refresh All Data', and 'Export All Data'. On the left, there's a sidebar with 'Layout View', 'Data View' (selected), 'Dashboard Settings', and 'Run Preview'. Below this are 'Control Instances' and three radial gauge charts for 'Revenue', 'Expenses', and 'Profit', each showing a value of '\$25,575 + \$575 (2%)'. At the bottom, there are tabs for 'CorporateStructure', 'PerformanceByDivision', 'HumanCapital', and 'IncidentsPerProductLine'. The main area displays a table with columns: Division, Y (Revenue), Revenue Target, Expenses, Expense Projections, Profit, and Target Profit. The 'Revenue' column for Division A1 is highlighted with a red border. The 'Revenue Target' column for Division A1 is highlighted with a green border. The 'PerformanceByDivision' tab is selected, showing data properties for the 'Revenue' gauge. The 'Main Value' dropdown is set to 'PerformanceByDivision' and 'Revenue'. The 'Comparison Value' dropdown is set to 'PerformanceByDivision' and 'Revenue Targ'.

Division	Y	Revenue	Revenue Target	Expenses	Expense Projections	Profit	Target Profit
1	Division A1	30,527,280.00	22,967,145.87	23,849,640.00	15,907,867.00	6,677,640.00	7,059,278.87
2	Division A2	23,904,720.00	22,967,145.87	11,124,000.00	15,907,867.00	12,780,720.00	7,059,278.87
3	Division A3	25,019,280.00	22,967,145.87	13,373,640.00	15,907,867.00	11,645,640.00	7,059,278.87
4	Division A4	26,712,720.00	27,441,243.12	20,285,640.00	14,848,752.00	6,427,080.00	12,592,491.12
5	Division B1	29,771,280.00	27,441,243.12	16,470,000.00	14,848,752.00	13,301,280.00	12,592,491.12
6	Division B2	33,767,280.00	27,441,243.12	15,587,640.00	14,848,752.00	18,179,640.00	12,592,491.12
7	Division B3	41,796,000.00	42,178,783.20	19,386,000.00	13,889,247.00	22,410,000.00	28,289,536.20
8	Division B4	39,383,280.00	42,178,783.20	16,794,000.00	13,889,247.00	22,589,280.00	28,289,536.20
9	Division C1	33,551,280.00	42,178,783.20	11,807,640.00	13,889,247.00	21,743,640.00	28,289,536.20
10	Division C2	30,168,720.00	29,137,151.94	12,924,360.00	19,615,238.00	17,244,360.00	9,521,913.94
11	Division C3	25,127,280.00	29,137,151.94	23,400,360.00	19,615,238.00	1,726,920.00	9,521,913.94
12	Division C4	27,503,280.00	29,137,151.94	20,268,360.00	19,615,238.00	7,234,920.00	9,521,913.94

The dashboard runtime provides aggregated values for the gauge, based on user selections. Note that the Comparison Value of the same Radial Gauge instance can be bound to a field from a different data view. The following example shows how multiple chart series can be bound to a collection of fields in a data view:

Sales vs. Expenses

Add Data
Refresh All Data
Export All Data
Layout View
Data View
Dashboard Settings
Run Preview

Control Instances



	Month Start	Product A	Product B	Product C	Total
1	1/1/2008 12:00:00 AM	143,999.52	214,999.57	389,999.48	748,998.57
2	2/1/2008 12:00:00 AM	146,999.51	204,999.59	292,499.61	644,498.71
3	3/1/2008 12:00:00 AM	125,999.58	229,999.54	262,499.65	618,498.77
4	4/1/2008 12:00:00 AM	89,999.70	104,999.79	202,499.73	397,499.22
5	5/1/2008 12:00:00 AM	71,999.76	104,999.79	179,999.76	356,999.31
6	6/1/2008 12:00:00 AM	74,999.75	209,999.58	277,499.63	562,498.96
7	7/1/2008 12:00:00 AM	116,999.61	199,999.60	217,499.71	534,498.92
8	8/1/2008 12:00:00 AM	65,999.78	164,999.67	269,999.64	500,999.09
9	9/1/2008 12:00:00 AM	95,999.68	99,999.80	232,499.69	428,499.17
10	10/1/2008 12:00:00 AM	47,999.84	114,999.77	97,499.87	260,499.48
11	11/1/2008 12:00:00 AM	62,999.79	89,999.82	97,499.87	250,499.48
12	12/1/2008 12:00:00 AM	47,999.84	94,999.81	112,499.85	255,499.50
13	1/1/2009 12:00:00 AM	65,999.78	104,999.79	127,499.83	298,499.40
14	2/1/2009 12:00:00 AM	68,999.77	139,999.72	187,499.75	396,499.24
15	3/1/2009 12:00:00 AM	110,999.63	219,999.56	307,499.59	638,498.78

MonthlySalesByProduct
MonthlyExpenses
MonthlyProfit

Profitability Analysis

Data Properties

Main Series: MonthlySalesByProduct Product A;... Options...

Comparison series: MonthlyExpenses Expenses Options...

Preparing Dashboard Data

Datazen supports a number of complex data features including filtering, aggregation, and time slicing; however certain points should be kept in mind while preparing data. Pre-aggregating data can optimize both dashboard creation and use and is, on occasion, required by some dashboard designs.

For best results, queries or documents should be pre-aggregated to 10,000 records or fewer. For record sets larger than 10,000 records configuring load-on-demand is suggested. See [Data Access](#) for information on when load-on-demand and parameterized data views and dashboards should be used.

When dealing with date and time intervals for use in a dashboard, particularly with the TimeNavigator, it's important that the date/time column is properly formatted so that it can be identified as such. The following are examples of valid date/time formats:

```
05/01/2009  
2009-05-01  
05/01/2009 14:57:32.8  
2009-05-01 14:57:32.8  
2009-05-01T14:57:32.8375298-04:00  
5/01/2008 14:57:32.80 -07:00  
1 May 2008 2:57:32.8 PM  
Fri, 15 May 2009 20:10:57 GMT
```

Date- and time-based datasets can, in most cases, be described by one or more date/time intervals, such as hourly, daily, monthly, quarterly, and yearly. Datazen is able to combine multiple tables of different granularity and display them on a single dashboard however it's good to keep in mind the relevant intervals from the original dataset(s) as this can help when deciding what date/time filter options to present to the user in the final dashboard.

Datazen is able to filter data based on both date/time fields and key fields. While key fields can be numeric in most cases they will be either an ID or a string value. In order to prepare a filter field for use with a navigator element such as the Selection List the filter key should be contained within a single column of the desired data table. In this way the table rows may be grouped according to the value in the filter column. Having multiple columns contain different filter keys, or filter criteria, allows for dashboards with multiple filter navigators to be used together in a cascading fashion, or individually applied.

Industry	Country	Region
Banks	AFGHANISTAN	ASIA
Commercial & Professional Services	AFGHANISTAN	ASIA
Food, Beverage & Tobacco	AFGHANISTAN	ASIA
Media	AFGHANISTAN	ASIA
Pharmaceuticals	AFGHANISTAN	ASIA
Food & Staples Retailing	ALBANIA	EUROPE

Key-based filters may also be hierarchically structured for use with a Selection List in a tree structure. To prepare data for use in this type of scenario a look-up table should be created describing the hierarchical structure. This is easily accomplished by using a table structure that includes a Key column and a Parent Key column to indicate where a node belongs in the overall hierarchy.

ItemKey	ParentItemKey
Financial	
Industrials	
Consumer Staples	
Consumer Discretionary	
Health Care	
Information Technology	
Banks	Financials
Real Estate	Financials
Diversified Financials	Financials
Insurance	Financials
Commercial & Professional Services	Industrials
Capital Goods	Industrials
Transportation	Industrials
Food, Beverage & Tobacco	Consumer Staples
Food & Staples Retailing	Consumer Staples
Household & Personal Products	Consumer Staples
Media	Consumer Discretionary
Automobiles and Components	Consumer Discretionary
Consumer Durables and Apparel	Consumer Discretionary
Consumer Services	Consumer Discretionary
Retailing	Consumer Discretionary

Pharmaceuticals	Health Care
Health Care Equipment & Services	Health Care
Software & Services	Information Technology
Technology Hardware & Equipment	Information Technology
Telecommunication Services	Information Technology

Working with Excel Data

Here are some things to keep in mind when preparing an Excel file and worksheets for use with a dashboard:

DO

- Have one worksheet per data set
- Have column headers in the first row
- Keep data types consistent within each column
- Format cells as proper types in Excel
- When using formulas, ensure that the entire column is calculated using the same formula
- Use Excel 2007 or later
- Save Excel files with the extension XLSX

DO NOT

- Include images, graphs, pivot tables or other embedded objects in data set worksheets
- Include total or calculated rows
- Keep the file open in Excel when importing
- Format numbers manually by adding currency or other symbols
- Use Excel 2003 or earlier
- Use files with the extension XLS

Worksheets

When preparing an Excel file for use with a dashboard one should enforce a policy of having one dataset per worksheet. Each individual worksheet will be imported into the dashboard designer as a separate table. Identically named worksheets from multiple Excel sources will be renamed upon importing by appending incrementing numbers, for example if three worksheets entitled "MyWorksheet" are imported the second and third will be renamed to "MyWorksheet0" and "MyWorksheet1". The screenshot below illustrates the first few rows of an ideal Excel worksheet ready for import.

A	B	C	D	E	F	G
1	Date	Region	Product A	Product B	Product C	Total
2	1/1/08 12:00 AM	Africa	143999.52	214999.57	389999.48	748998.57
3	2/1/08 12:00 AM	North America	146999.51	204999.59	292499.61	644498.71
4	3/1/08 12:00 AM	Asia	125999.58	229999.54	262499.65	618498.77
5	4/1/08 12:00 AM	North America	89999.7	104999.79	202499.73	397499.22
6	5/1/08 12:00 AM	Africa	71999.76	104999.79	179999.76	356999.31
7	6/1/08 12:00 AM	Europe	74999.75	209999.58	277499.63	562498.96
8	7/1/08 12:00 AM	Caribbean	116999.61	199999.6	217499.71	534498.92
9	8/1/08 12:00 AM	Asia	65999.78	164999.67	269999.64	500999.09
10	9/1/08 12:00 AM	Europe	95999.68	99999.8	232499.69	428499.17
11	10/1/08 12:00 AM	Central America	47999.84	114999.77	97499.87	260499.48
12	11/1/08 12:00 AM	North America	62999.79	89999.82	97499.87	250499.48
13	12/1/08 12:00 AM	Europe	47999.84	94999.81	112499.85	255499.5
14	1/1/09 12:00 AM	North America	65999.78	104999.79	127499.83	298499.4
15	2/1/09 12:00 AM	Asia	68999.77	139999.72	187499.75	396499.24
16	3/1/09 12:00 AM	Middle East	110999.63	219999.56	307499.59	638498.78
17	4/1/09 12:00 AM	North America	131999.56	234999.53	239999.68	606998.77
18	5/1/09 12:00 AM	North America	140999.53	189999.62	337499.55	668498.7

Example data in Excel.

As you can see in the example above, it is preferable that the first row contain the name of the metric contained in that

column. Upon adding this worksheet to the designer, these column headers will be preserved for easy reference in gallery element settings. Column headers are not required, however, and if missing will be auto-generated using the Excel A,B,C,...,AA,BB,... convention.

Cells

It is important that cell data is consistent within each column of a worksheet's dataset. Each column will be assigned a data type upon importing. Data types will be automatically detected as either string, double (numeric), boolean (true/false) or datetime. Mixing data types within the same column of a worksheet can cause this detection to be inaccurate or fail completely. This detection accounts for possible column headers being of string type. Cells should be formatted as the correct type in Excel to ensure that dashboard designer detects the desired types. In the above example the six columns would be typed as a datetime column, a string column, and four double columns.

If a worksheet contains calculated cells or formulas, only the resulting display value will be imported into the dashboard designer.

Key Metrics

Dashboards have the ability to display key metrics. These will be displayed on the dashboard's main screen tile as well as the dashboard's pinned tile if it has been added to your Windows 8 start screen. When using Excel, key metrics should reside on a separate worksheet consisting of a value and a target.

File Location and Refreshing Excel Data

There are no restrictions as to where on your file system you can store Excel files to be imported into the dashboard designer. However, if the file is moved or renamed after importing, it will prevent the ability to refresh that data through the "refresh all data" command found in the Data View. It's important to note that the dashboard designer does not automatically refresh Excel data; this must be done by the dashboard author through the designer refresh command. If you wish to take advantage of this feature, ensure that any imported Excel files reside in a permanent location on your file system.

Dates

Dates fields are essential to many dashboards thus it is important to ensure that cells are properly formatted as dates in Excel. In some cases this means a conversion is necessary. This section describes some techniques for converting cells from text to dates in Excel.

```
Week 24-2013=DATE(MID(A2,9,4),1,-2)-WEEKDAY(DATE(MID(A2,9,4),1,3))+MID(A2,6,2)*7
```

```
2013/03/21=DATEVALUE(A1)
```

```
2013-mar-12=DATEVALUE(RIGHT(A1,2)&"-"&MID(A1,6,3)&"-"&LEFT(A1,4))
```

Once a conversion like this has been applied the cells must be formatted as dates by selecting them, or the entire column, selecting Format Cells from the Context Menu, and Date from the Category list in the Format Cells dialog. Excel's text-to-columns wizard may also be used to convert text cells to properly formatted dates.

Unsupported

Worksheet data in formats other than those described above could cause undesired results when imported. It is a good

idea to restrict worksheets in an Excel document to only those which are in the correct format for use with a dashboard.

Custom objects embedded into Excel worksheets including pivot tables, visualizations, and images will not be imported into the dashboard designer.

Working with Live Data Sources

In addition to being able to load data from local [Excel](#) files Datazen can, in conjunction with a Datazen Enterprise Server, also access live data from almost any source. Accessing live data requires a Datazen Enterprise Server; each live data source is configured on that Datazen Enterprise Server through its control panel by an administrator.

Once connections and data views, or queries, are configured, live data sources are proxied through the Datazen Enterprise Server and made available to Datazen clients for inclusion in dashboards.

After a user has connected to a Datazen Enterprise Server from the client, importing live data views into a dashboard is as easy as selecting Datazen Enterprise Server instead of local excel from the “add data” dialog. Selecting Datazen Enterprise Server will present the user with the data sources available sorted into folders by the server administrator. Selecting a data source will present the user with the available data views in that source, one or more of these views may then be checked off and imported into the dashboard designer.

← Add Data to New Dashboard – Step 4

Sql Datazen DemoDb

<input type="checkbox"/> Monthly Product Sales	<input type="checkbox"/> Monthly Product Sales and Targets	<input type="checkbox"/> Quarterly Sales Targets	<input type="checkbox"/> Quarterly Expense Projections by Category
<input type="checkbox"/> Sales Breakdown	<input type="checkbox"/> Regional Sales by Industry	<input type="checkbox"/> Monthly Sales	<input type="checkbox"/> Monthly Sales Targets by Product
<input type="checkbox"/> Daily Sales	<input type="checkbox"/> Sales Last Hour	<input type="checkbox"/> Monthly Expenses	<input type="checkbox"/> Monthly Expenses by Category

Cancel

Live data sources are available from servers.

Thanks to Datazen Enterprise Server, once a live data source has been imported into the dashboard designer the design process may proceed in exactly the same as with local data.

Real-time data and Parameters

Data sources can be configured to perform real-time data retrieval – every time a dashboard that utilizes real-time data is viewed, the original data store is queried for data.

These real-time sources may use parameters in their query to filter data before it is sent to a dashboard. When a parameterized source is added to a dashboard, the “Parameters” option will become available in the source’s settings rose. Invoking this option will present a dialog that allows you to override the default value for each parameter with either a static value or the property of a navigator on the dashboard.

Working with Simulated Data

When a gallery element is placed on the design surface, simulated data for that control is immediately generated. This data serves a number of useful purposes when creating dashboards.

Category	Metric1	Metric2	Metric3	Metric4	Metric5	Comparison1
1	Item A1	145,748.00	241,698.00	560,122.00	3,430.00	759,778.00
2	Item A2	122,024.00	773,750.00	156,859.00	156,615.00	485,458.00
3	Item A3	231,842.00	103,066.00	863,856.00	891,691.00	704,546.00
4	Item A4	991,127.00	565,861.00	332,030.00	997,866.00	641,611.00
5	Item B1	934,386.00	183,375.00	424,880.00	550,112.00	2,992.00
6	Item B2	8,644.00	234,560.00	766,129.00	365,567.00	983,919.00
7	Item B3	674,421.00	650,233.00	231,684.00	487,082.00	399,658.00
8	Item B4	194,827.00	466,473.00	451,987.00	41,587.00	23,109.00
9	Item C1	565,675.00	847,905.00	331,714.00	241,004.00	334,066.00
						756,567.00



An example of simulated data.

Firstly, simulated data can help immensely when taking a design-based approach to dashboard creation. Initially populating elements with simulated data allows the rapid creation of dashboard prototypes without having to address specific data requirements. These dashboards can then be evaluated for the overall aesthetics and effectiveness.

Secondly, simulated data provides a template which accurately represents the data requirements of a particular dashboard design. By using the Export All Data command located at the top right of the Data View, an Excel document containing the simulated data is generated, allowing for quick substitution of actual data, ready for import.

Simulated Data Behavior

The simulated data generated is tailored specifically to the dashboard being created. As more elements are placed on the design surface, the associated simulated data will grow and change to provide the best possible experience short of real data. This evolution ensures that extra fields and filter possibilities are available in case the user adds extra series to chart visualizations or expands the scope of one or more dashboard elements in another way.

Simulated data may be exported to an Excel file, as mentioned previously, creating a perfect data template for the associated dashboard. Users may simply swap in their own real data into the Excel file and import it back into the dashboard designer.

Once all controls have been bound to real data, simulated tables which are no longer in use are automatically removed from the dashboard. Simulated tables still referenced by elements on the design surface cannot be removed.

It's important to note that simulated data does not add to the overall dashboard footprint as they are not serialized with the dashboard but generated on-the-fly at runtime.

Configuring Navigators

Navigators & Filtering Concepts

Navigators are used to filter input data tables into smaller subsets. Time-based navigators filter tables by selecting rows that fall into the desired time range. Selection-based navigators filter tables by selecting rows where a particular column value matches the desired key value; or, in cases of hierarchical trees, where a particular column value belongs to the sub-tree of the desired key value.

Navigators can be used to provide filter values in offline datasets, or for load-on-demand and parameterized queries in connected scenarios. For more information on which functionality you should consider please refer to [Data Access](#).

Time Navigator

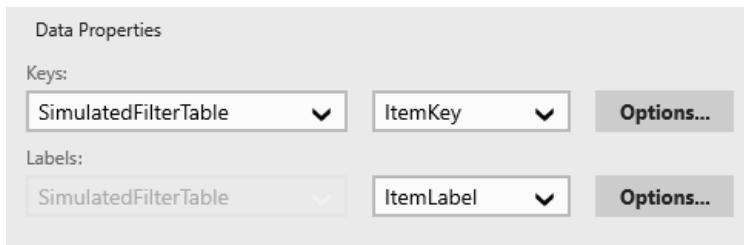
As its name implies, the time navigator filter is used to select a range of data bounded by a time range. Any control that is connected to time-based data can opt-in (default) to the time navigator filter. If a table contains more than one time-based column, only the first is used for filtering. The series table drives the embedded visualization and determines the overall date range of the dashboard.



Time Navigator series properties.

Selection List

The selection list filters input data by matching the selected key value of the filter to the value of a specified column for each row of a filtered table. In the data properties panel, first select a table that will provide the keys to be filtered and choose the key column, then select a column for the label that will be displayed. The key column and label column can be the same.



Selection List data properties.

In the case of hierarchical tree data, a parent key column must be selected.

Data Properties

Keys:

Parent Keys:

Labels:

Selection List tree structure data properties.

Once the data properties have been set, tables that are to be filtered must be configured. In the filtered tables panel, check each table to be filtered and select the column to filter by. This column will be used to match values with the selection list's key column.

Tables Filtered by Selection List 1

SimulatedFilterTable

SimulatedTable

Selection List filter configuration.

Scorecard Grid

The scorecard grid filter functions very much like the selection list filter, but in addition, it also displays score indicators and value columns. After selecting the key, label, and optional parent key columns, an input table and column must be selected to provide the scorecard with data. The scorecard data column should be filterable by the key column.

Data Properties

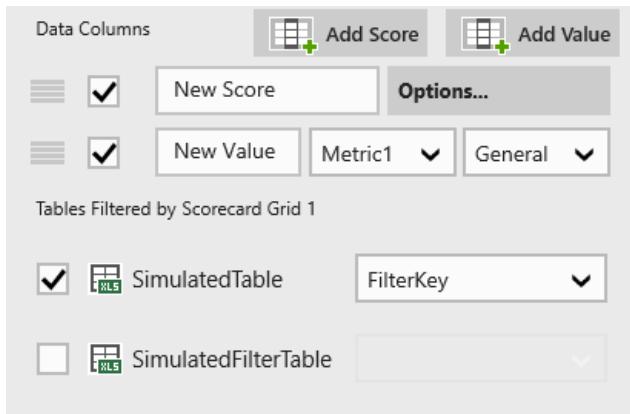
Keys:

Labels:

Data Keys:

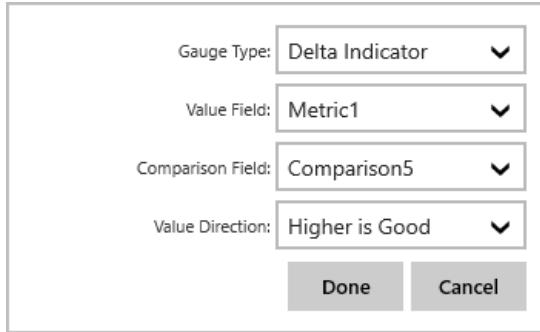
Scorecard Grid data properties.

To configure the score indicators, switch to the data view and locate the panel in the lower-right corner. From here, a score indicator can be added by clicking the "add score" button.



Scorecard Grid column configuration.

Name the score indicator as desired and configure its properties by clicking the options button.



Scorecard Grid visualization configuration.

- Gauge type: Choose a gauge type from the list; valid gauges types are delta arrow, delta indicator, delta background, delta foreground, and progress bar.
- Value field: Choose the desired value field from the table.
- Comparison field: Choose the desired comparison field from the table.
- Value direction: Choose whether higher values are good or lower values are good.

To add a value indicator, click the “add value” button. Name the value indicator as desired, choose its source column from the table, and select how it will be formatted.

Configuring Gallery Elements

Gallery elements are configured to use filters by clicking the options button for a particular input in data view. Filters are enabled by toggling the appropriate checkbox.



Gallery element filter configuration.

Cascading Filters

Filters can also be cascaded together so that the selected value of one will filter the available values in a second. To cascade filters, apply the filters to the Key column as you would a regular gallery element.

Configuring Visualizations

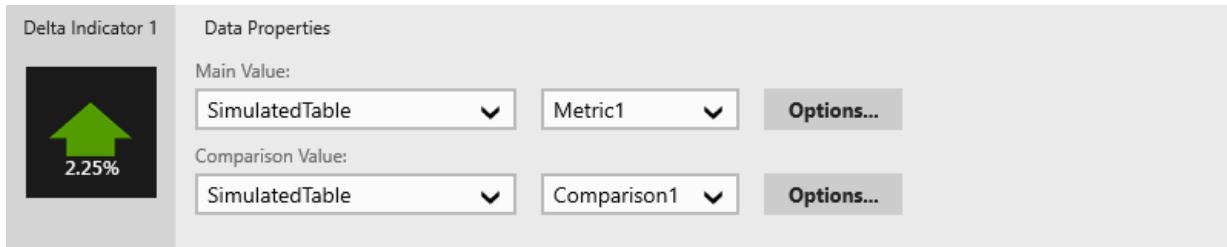
Gauges



Gauge visualizations.

Gauges are the most basic and most widely used visualization controls in Datazen. All gauges share at least one common property, value. In order for a gauge to function with real data, its value property must be set to a numeric field in one of the data tables available to the dashboard in which it resides. The value property can be set by selecting the gauge visualization on the design surface, switching to the data view and using the drop-down lists in the data properties panel to select a data table and a numeric field.

Some gauges, such as the delta number and delta indicator visualizations, are able to display a comparison value or the delta relationship between the value and comparison value. The comparison value, when supported, may also be set by selecting the gauge on the design surface and using the drop-down lists in the data properties panel to select the desired data table and numeric field which represents the comparison value.



Gauge data properties.

Aggregation options are also available for gauge elements. The default is sum, which will display the total of all values contained within the current filtered data available to the gauge control. When selecting an alternate aggregation method for the value property keep in mind that you may also want to change the aggregation method on the comparison value property, though in some cases mixing aggregation methods may be desired.

If the dashboard has any navigators, a gauge element may be bound to one or more of these by checking off the desired Navigators in the "filtered by" section of the option popup found in the data properties panel. For the sake of versatility, a gauge's value and comparison value may be bound to one or more different navigators. This powerful feature allows virtually endless options for delta gauge elements.

Along with the data properties which connect gauge elements to data fields, there are a number of functional and visual properties which can be configured to customize an element. The value direction property is one of these and may be set to either HigherValuesAreBetter or LowerValuesAreBetter. This determines whether positive values will be colored green, indicating a desirable change for the better, or red, indicating an undesirable change for the worse. The value

direction property relates only to Gauge elements which support a comparison value and the color of the delta gauge will be determined by the sign of the delta integer and the value direction property setting.

The second gauge-specific non-data property is range stops. Range stops can be adjusted via a popup accessed below the design surface once a gauge element has been selected. This property determines at what percentage of its comparison value visualization should be presented as on-target (green), neutral (amber) and off-target (red) – a gauge's comparison value being the target.

Another non-data property of gauge element, and one shared by many other elements, is number format. The number format property can be accessed below the design surface when an element is selected. It determines how numbers displayed in the element should be formatted – standard versus currency, for instance. This property is not dashboard-wide and should be set on each element.

Charts



Chart visualizations.

An essential part of dashboard design and data visualization, Datazen offers a variety of chart elements which can be configured to cover a wide range of scenarios. Some properties apply to all chart elements, and others only to specific chart elements.

The number format property can be accessed below the design surface when an element is selected; this property determines how numbers displayed in the chart element should be formatted – standard versus currency for instance. This will apply to axis annotations, as well as data point popups. This property is not dashboard-wide and should be set on each element.

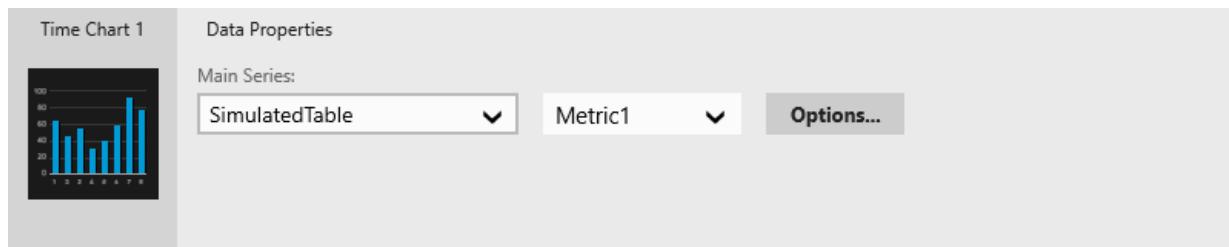
A legend may be enabled for chart elements by toggling the show legend property on the layout view.

Each individual metric, or value, displayed on a chart element is referred to as a series; multiple series can, and do, share both a common x-axis and a common y-axis. Series are defined in the data properties panel of the data view by selecting one or more data tables and fields. Each field will result in an individual series of datapoints on the chart visualization with its own color.

There are three basic types of chart types in Datazen: time, category, and totals. These three chart types have matching comparison chart elements, which allow the chart's visualization to show two distinct sets of series for comparison.

Time Chart

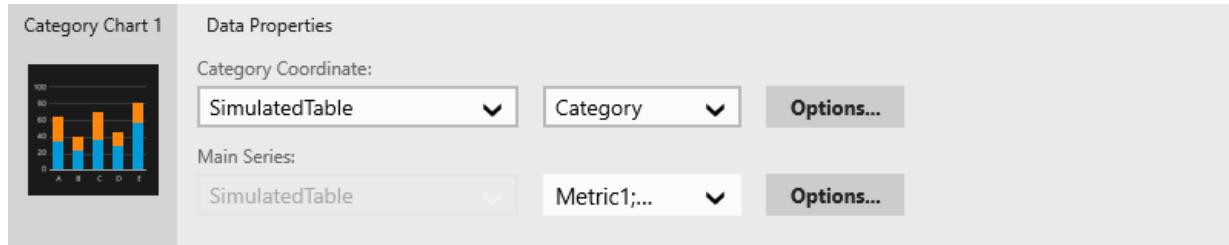
The time chart is the most basic chart element in Datazen and can be configured by setting its main series property to a numeric field in a data table. The time (and date) axis of the chart will automatically be set to the first valid date/time field in the data table.



Time Chart data properties.

Category Chart

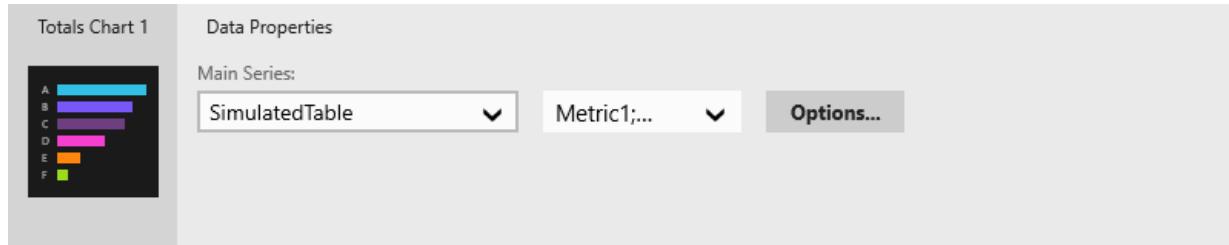
The category chart, unlike the time chart, will allow an x-axis relating to a grouping rather than a date/time field. This grouping is assigned through the category coordinate property accessible in the data properties panel of the data view. Category coordinates are restricted to fields of string type. The data will be grouped by this field and displayed on the resulting chart with the category coordinate field comprising the x-axis. The main series of category charts is set in the same way as the time chart and is restricted to numeric values.



Category Chart data properties.

Totals Chart

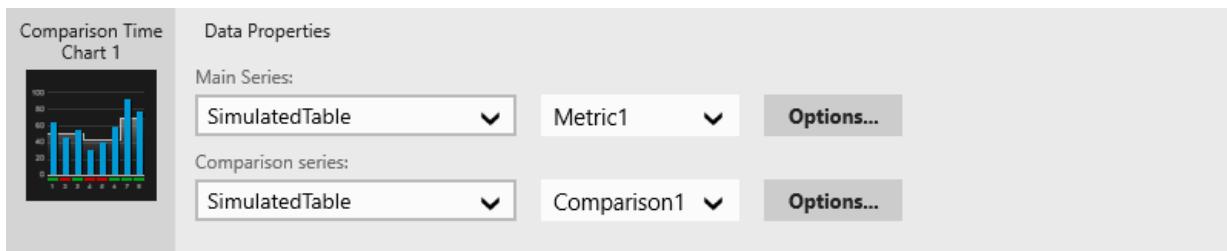
The totals chart accomplishes two unique things: firstly, it does not present multiple series but rather only the sum, or total, of the defined main series. Secondly, it has the option to use data table columns as a grouping when its data structure property is set to ByColumns. This can be quite useful when dealing with flattened data. In the latter mode, only the main series property is available as the category column is automatically determined by the number of fields selected for the main series property.



Totals Chart data properties.

Comparison Chart

Each of the three chart types are available as comparison charts. This allows the user to specify not only a main series, but also a second comparison series which can be displayed in different ways by adjusting the series visualization to one of the following: BarVsThinBar, LineVsBar, and BarVsStepArea. These visualization types describe how the comparison series should be displayed in relation to the bars of the main series.



Comparison Chart data properties.

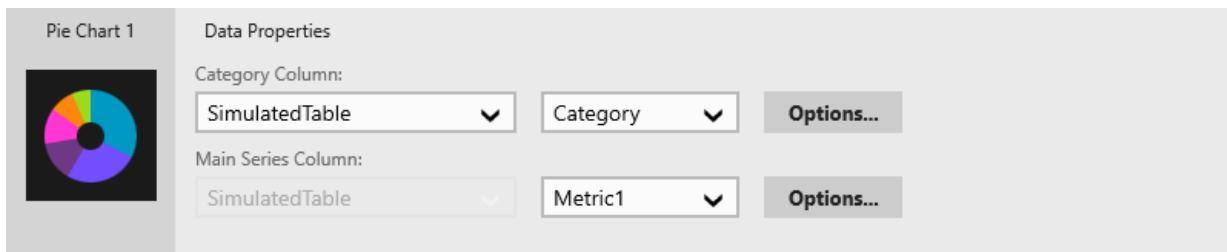
Comparison charts also introduce a unique visual property: reuse colors on comparison series. If set to true, the color palette used for the series will be reset between drawing the main and comparison Series, in order to coordinate between comparison series and similarly-sorted main series. If set to false, the color palette will continue its normal rotation when drawing the main series after the comparison series, preventing potentially misleading color coordination between the two sets of series.

Pie and Funnel Charts



Pie and Funnel Chart visualizations.

Pie and funnel charts are among the simplest of visualizations. The data structure, found on the visual properties panel of the layout view, can either be set to ByRows or ByColumns. When set to ByColumns simply select the columns to aggregate and display on the chart in the data properties of the data view. The field names will be used to label each area of the resulting chart. Conversely, when using the ByRows data structure, a field must be specified for grouping and labels, the category column, as well as a numeric field for the main metric or the main series column of the chart.



Pie and Funnel Chart visualizations.

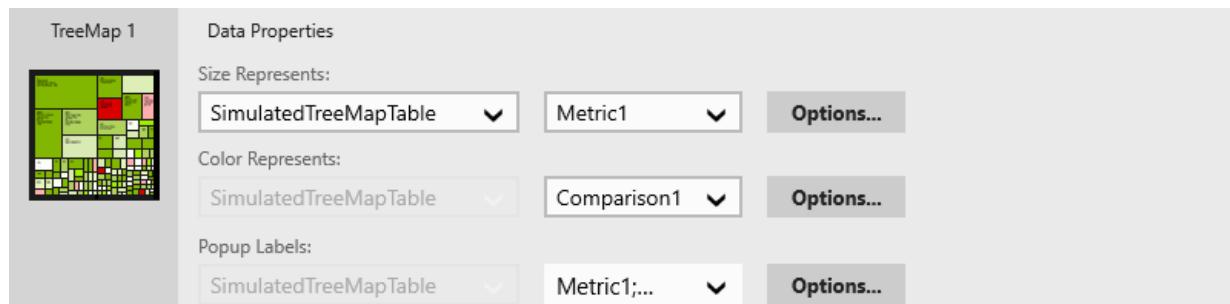
TreeMap



TreeMap visualization.

The treemap element displays metrics by applying their values to the area and color of tiles within a rectangular grid. In

the data properties panel of the data view, a designer can select a numeric field to be represented by the tile's size and a numeric field to be represented by the tile's color. In addition to these properties, the designer can also set which fields should be displayed both in the popup of each tile and, when space permits, on the tile itself.



TreeMap data properties.

A hierarchical treemap can be display by selecting the two-level treemap option in visual properties panel of the layout view. This will add the “group by” property to the data properties that should be set to a field by which the other metrics will be grouped and aggregated. Enabling two-level treemap works very well with show item headers enabled.

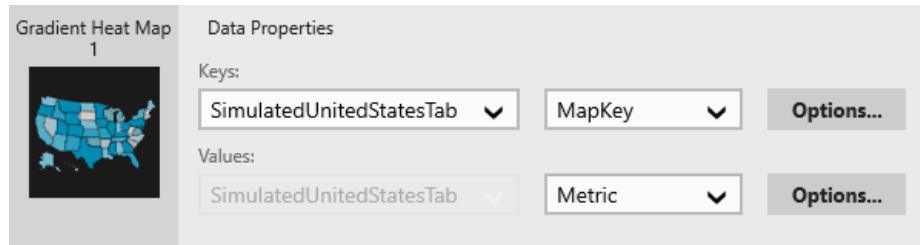
Maps



Map visualizations.

Maps are a fantastic way to visualize geographical data, and Datazen provides three types of map visualization along with 10 maps. These three elements require a few key data properties be set in order to properly display real data.

The first is the keys property which will connect the data to specific map regions. Ensure that you have the correct map selected – USA, for instance, if the data is indexed by USA states; or WorldCountries if the data is indexed by country name. The selected map may be changed in the visual properties panel in the layout view. Once the correct map has been selected the keys property in the data properties panel of the data view should be set to the data table and field which contains the region names, state or country for example.



Map data properties.

The second property is the values property, which should be set to a numeric field in the same table as the selected keys field. These values will be represented in different ways depending on which map element is being used. The gradient map will use these values to color each region with a varying shades based on the range of values whereas the bubble map will base the size of a bubble visualization over each region on the value property. The bubble map also allows the dashboard designer to specify whether each region's bubble visualization should be a different color, or whether all should be the same color via the Use Different Colors toggle in the visual properties panel in the Layout

View.

The range stop heat map element allows the user to visualize a value's relation to a target. Because of this, the targets property must be set on the data properties panel of the data view, in addition to both values and keys. The targets property should be set to a numeric field in the same data table as both the values and keys. The range stop heat map also supports the value direction property, which may be set to either HigherValuesAreBetter or LowerValuesAreBetter. This determines whether positive values will be colored green, indicating a desirable change for the better, or red, indicating an undesirable change for the worse. The color of the regions will be determined by the sign of the delta between the value and the target as well as the value direction property setting.

DataGrids



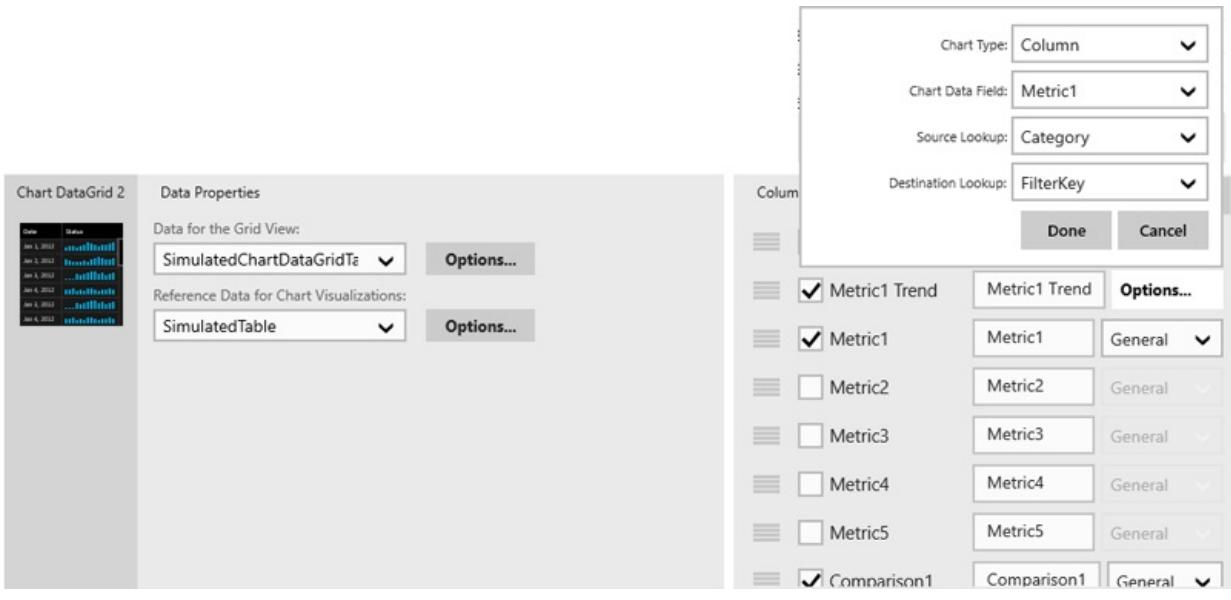
DataGrid visualizations.

In cases when the best visualization is the data itself, Datazen includes three datagrid elements to accomplish this.

The most basic datagrid, the simple datagrid, is capable of displaying multiple columns of data with custom formatting and headers. The simple datagrid can be connected to real data by selecting a data table in the data properties panel of the data view. Once a data table has been selected, the fields to be included can be selected by checking them off on the right hand side of the data properties panel. In addition to including column header text can be entered in the textboxes adjacent to the field checkboxes, and, in the case of date/time fields and numeric, formatting options can be specified.

The remaining two datagrid elements, the indicator datagrid and chart datagrid, allow the designer to integrate gauge and chart columns, respectively. This can be accomplished by selecting the "add gauge (or chart) column" in the columns panel of the data view.

Once a gauge column has been added to a datagrid some properties must be set in the options panel accessed by clicking the options button beside the gauge column in question. First, select a gauge type from the first drop-down list in the options panel, then continue to configure the gauge as normal. The datagrid will automatically feed the gauge only the data specific to that row of the datagrid.



DataGrid data properties.

Adding chart columns to datagrids requires additional considerations as a separate data table is needed to provide chart data for each row. This can be set in the data properties panel under "reference data for chart visualizations". This second data table must share a field which may be joined in order to link each row to associated chart data.

After adding a chart column and selecting a chart type from the chart column options panel, each chart column must be assigned a data field, a source lookup, and a destination lookup. These three properties determine how the datagrid should provide data to each chart in the column.

Firstly, the source lookup property is set to a field in data table from which the datagrid itself is getting data. This field can be considered a per-row-filter which will be applied to the chart reference data table in order to provide data to the embedded chart for each row. After selecting a source lookup field from the datagrid's source data table, select a destination lookup. This is the field in the "reference data for chart visualizations" data table. The data for the chart in each row will be joined on those two fields.

The last property that needs to be set is the chart data field. This determines which metric in the "reference data for chart visualizations" data table will be used as the y-axis value or series in the chart in each row.

Publishing Dashboards

Publishing Steps

To publish a dashboard to a server, open the desired dashboard in the designer, swipe up from the bottom of the screen (or right-click to activate the bottom menu) and click the publish button.

If there are no server connections configured, one must first be configured. Enter the appropriate connection credentials and click Connect.

If the dashboard has been edited since it was last saved, the first step in publishing a dashboard is to save it to the local folder. At this point, the title, filename, local group, culture, and fiscal start date can be modified. Make any desired changes and click Save.

Publish to Server, Step 1

Before being published to the server, your dashboard must be saved locally.
Save dashboard as **New-Dashboard.datazen** ?

Dashboard Title:

File Name:

Dashboard Group:

Dashboard Culture:
 Affects default currency, date & number formatting, and static labels.

Fiscal Year Start:
 Affects time navigators and time charts.

Publishing a Dashboard to a Server Step 1.

The next step in publishing a dashboard is to configure the server settings. Server configuration options are the server to connect to, dashboard title on the server, hub, and server group. Make any desired changes and click Publish.

Publish to Server, Step 2

Datazen Server:

Dashboard Title:

Dashboard Hub:

Server Group:

Publishing a Dashboard to a Server Step 2.

At this point the dashboard has been published and will be accessible from any dashboard viewer client.

Saving a Dashboard Locally

When publishing a dashboard it must first be saved locally as described above in Step 1. If you wish to save a dashboard locally without publishing this can be done at any point by swiping up or right clicking in the dashboard designer and selecting Save Locally.

Dashboards saved locally in your working directory will automatically appear on the main screen of Datazen. They may also be opened through the Windows file explorer by double-clicking on them.

Advanced Topics

This section contains the following documents:

- [Parameterized Queries & Load on Demand](#)
- [Drill-throughs to Other Dashboards or Custom URLs](#)
- [Hierarchical Selections](#)
- [Cascading Selection Lists](#)
- [Working with Custom Maps](#)

Parameterized Queries & Load on Demand

Creating Dashboards with Load On Demand Data

Creating dashboards that load data on demand is a two-step process. First you need to create the appropriate data views that allow parameters from the dashboards to be passed into the queries. Once this is done, you have to connect the parameters to the appropriate filters in the dashboards in order for the view to be properly used by the dashboard.

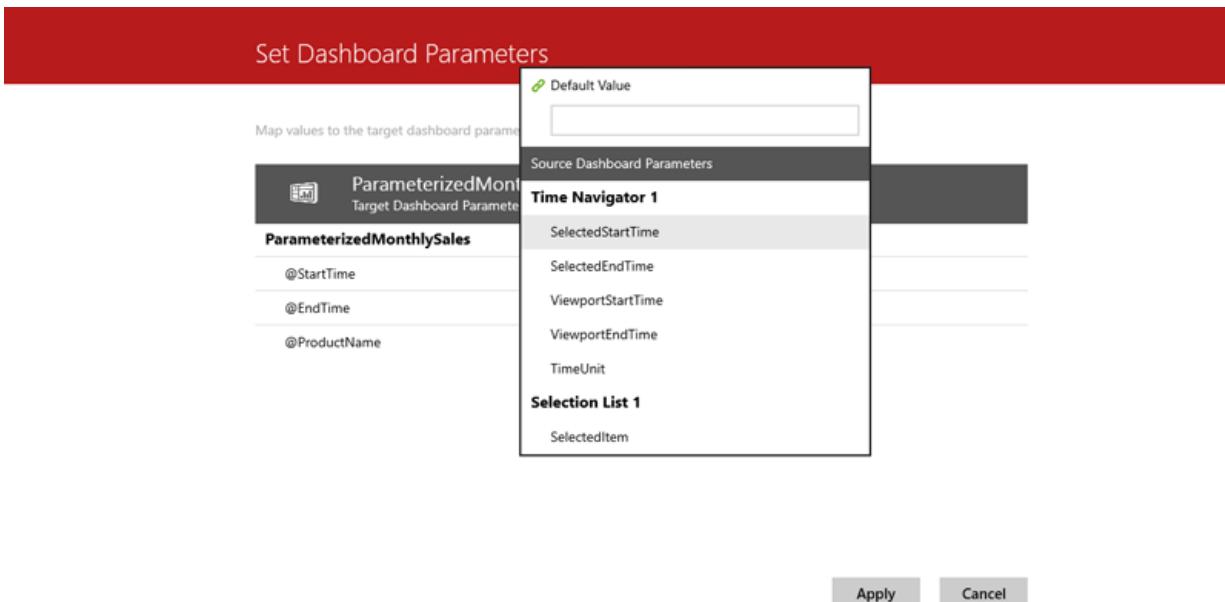
Creating Parameterized Views

Currently parameterized views are supported for SQL and MDX queries and must be created using the Control Panel. Please refer to [Data Access](#) for reason why you might consider using a parameterized view and how to get started.

Using Parameterized Views in a Dashboard

Once a parameterized view has been created using the Control Panel, a dashboard can be configured to use it with the Windows 8 client

In the Data View section of the Dashboard Designer parameterized data tables can be imported from a server. Once imported a green P on the right of the data table's tab denotes that parameters are available to be configured for it. Clicking on the gear icon beside the green P and selecting the Param option will display the parameter configuration screen.



Connecting data view parameters with navigators.

The parameter configuration screen allows you to bind the data table's query parameters to parameters available from elements currently on dashboard such as Selection Lists and Time Navigators.

Once the parameter setup has been completed, each time one of the parameters changes through end user input, any data table with bound parameters will be refreshed, along with all the controls that use its data. With this approach, you do not need to configure filtering on the client since the filtering is done on the database through parameters and only the filtered results are returned.

Using Parameterized Views with Load On Demand Data

Parameterized views can be used to load data on demand by allowing the user to select filter parameters through the use of Selection Lists or Scorecard Grid, as well as allowing the user to select a time range by using a Time Navigator. Queries structured with start and end datetime parameters support load on demand data for any dashboards with a Time Navigator bound to those start and end datetime parameters.

Drill-throughs to Other Dashboards or Custom URLs

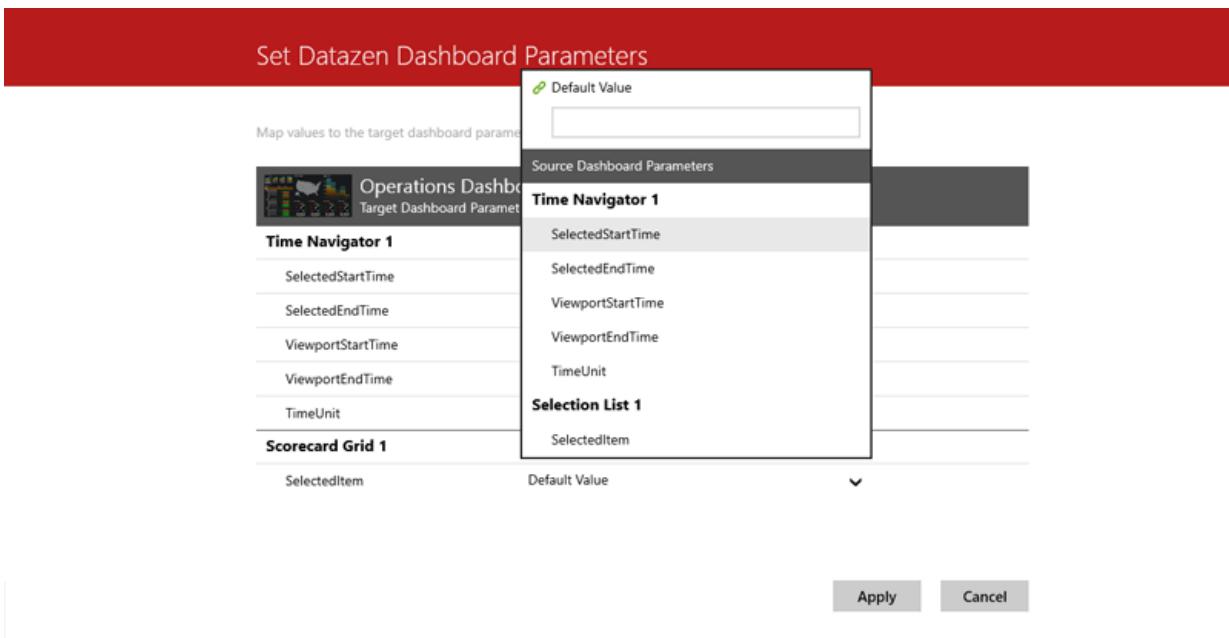
Any Gauge, Chart or Datagrid can be configured to initiate a drill-through action to another dashboard or custom URL. Depending on the source dashboard one or more parameters can be passed to the target dashboard or integrated into a custom URL.

At runtime a touch or click on an element configured with a drill-through target will send the user to the that target, either a dashboard or URL.

Configuring a drill-through to another dashboard

To configure a drill-through to another dashboard, in the dashboard designer first select an element that support drill-through targets on the design surface. Open the Drill-through Target drop down in Visual Properties pane and select Dashboard, this will open the drill-through configuration dialog.

Select the desired target dashboard from the drill-through configuration dialog. Once a target dashboard has been selected it's available input parameters will be displayed, these parameters include properties which can be bound on navigator controls as well as any parameters configured on data tables of the target dashboard.



Drill-through to another dashboard properties.

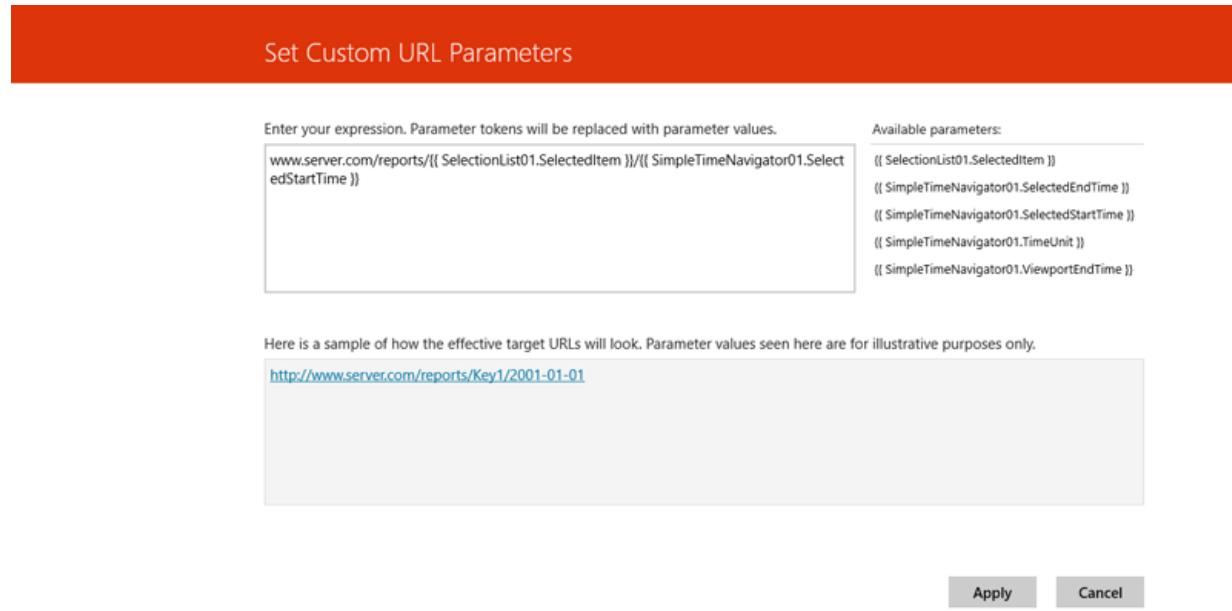
Properties of matching data types can then be connected to available output properties on the source dashboard by using the dropdown to the right of each property. Defaults for each output may also be configured here in the event a user has not interacted with the dashboard before initiating a drill-through action.

When publishing dashboards drill-through target dashboards should be published before drill-throughs are configured on source dashboards.

Configuring a drill-through to a custom URL

To configure a drill-through to a custom URL, in the dashboard designer first select an element that support drill-through targets on the design surface. Open the Drill-through Target drop down in Visual Properties pane and select CustomURL, this will open the drill-through configuration dialog.

In the Custom URL Parameter dialog a URL can be combined with available parameters by pasting, or typing the URL as well as touching or clicking on available parameter tokens listed to the right of the URL textbox. Parameters can also be typed or pasted manually into the URL textbox. A preview of the Custom URL combined with sample resolved parameters (if included) will be displayed in the panel below the input textbox.



Drill-through to a custom URL properties.

While previewing dashboards in the dashboard designer a dialog will be shown when drill-throughs are triggered. Drill-throughs can only be initiated when a dashboard is saved or published and viewed, not from within the dashboard designer or preview.

Hierarchical Selections

Preparing Hierarchical Data

The best practice for implementing hierarchical selections is to prepare data in a normalized fashion, specifically in the case of parent and child data relations.

Queries and tables should be structured to return data rows with a key, a parent key (or null) and a label. This data can then be used to create a hierarchical user interface with a hierarchically structured Selection List.

Implementing a Hierarchical Selection List

Once data is prepared a hierarchical selection list may be implemented in a dashboard by dragging a Selection List onto the dashboard design surface and selecting Tree from the Data Structure dropdown in the Visual Properties panel below the design surface.

When a Selection List's Data Structure has been set to Tree the visual display of the element on the design surface will change to reflect a tree-like hierarchical structure. If the Selection List occupies only one cell vertically and thus displays as a dropdown the tree structure will not be visible at design time.

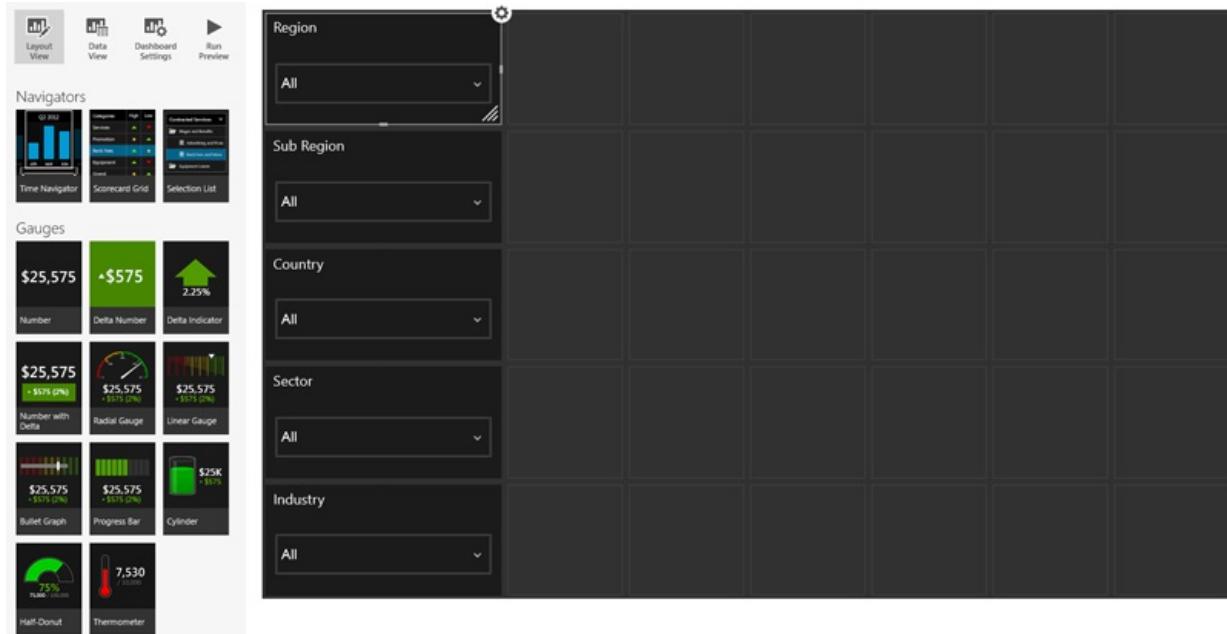
On the Data View of the dashboard designer a Selection List has properties for selecting columns to provide Key and Label. When set to use a Tree Data Structure an additional property is available for specifying a data table fields to provide Parent Keys. Once Keys, Parent Keys and Labels data properties have been set to appropriate data table fields the Selection List is ready to display a hierarchical structure.

While flattened or denormalized data may also be used to implement a hierarchical selection list, the preferred method is to use normalized data as outlined above.

Cascading Selection Lists

When dealing with hierarchical data sets one of the options to allow user interaction is two or more cascading selection lists where each selection list is populated with one level of the hierarchy.

In the dashboard designer add a Selection List for each user selectable level in the hierarchy to the design surface. Selection lists occupying only one vertical cell on the design surface will be displayed as dropdowns.



Cascading Selection Lists arranged on the Design Surface.

In the Data View each Selection List instance's Key property, which can be found in the Data Properties panel in the lower portion of the screen, should be set to the Key or Id of the desired associated hierarchy level. The Label property can be set to the Key, or a related user friendly string field.

Each Selection List instance should also be set to be filtered by its parents in the hierarchy. This can be done by selecting the Options button to the right of the Key property for each Selection List and ensuring that each and every ancestor, preceding (or parent) Selection Lists is checked and therefore will filter the current Selection List. This must be done for each Selection List in the hierarchy.

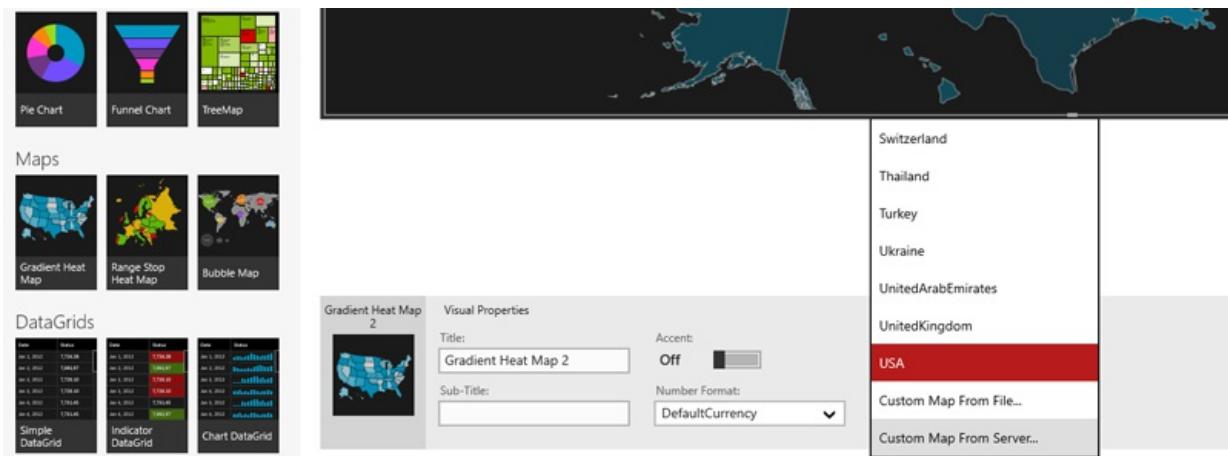
Setting filters for each Selection List should be done after all lists are in place and from the highest parent in the hierarchy to the lowest child. Cascading selection lists should be filtered by their parents and not by any child Selection Lists. Keeping this in mind, the top most Selection List should not be filtered by any others in the hierarchy. In this way a change to any Selection List in the hierarchy will propagate the filter throughout its descendants.

Working with Custom Maps

Connecting data to Custom Maps requires that the data be prepared in advance so that at least one field matches the region keys which are used in the Custom Map. In order to use a Custom Map it must first be uploaded to the server by the Hub owner. Please refer to [Implementing Custom Maps](#) and [Managing Custom Maps](#) for information on creating and installing custom maps on a Datazen Enterprise Server.

Loading a Custom Map

Once a Map gallery element has been placed on the dashboard design surface it can be connected to a Custom Map by selecting either Custom Map From File or Custom Map from Server. Selecting one of these options will allow you to browse either the server or your local file system for a Custom map.



Loading a custom map from a server or local file.

When loading a custom map from your local file system you must select two files, a file with the extension .SHP (shape) and a matching file with the extension .DBF (data), these two files are required for connecting data to a custom map.

When loading a custom map from a server each map will be presented as a single option, selection of individual .SHP and .DBF files is not required from server sources.

Connecting Data to a Custom Map

Three data properties are required for configuring Map instances, these are accessible from the Data View page of the dashboard designer.

In order to connect data to a custom Map the Keys property of the Map instance must be set to a data table field whose content matches the keys contained in the Map's data file. Map data keys in most cases will be region names.

Once Map keys are connected to data a Value field can be set, and in the case of the Range Stop Heat Map a Comparison field can be set as well.

Gradient Heat Maps display the Value properties as shades of a single color.

Bubble Map Value properties determine the radius of a bubble visualization displayed over the associated region.

Comparison fields for Range Stop Heat Maps will be used to determine the delta between the Comparison and the

Value. The resulting delta will determine the color within a range which will fill the associated region of the map.

Extending Datazen

This section contains the following documents:

- [Implementing Custom Branding](#)
- [Implementing Custom Maps](#)

Implementing Custom Branding

A brand package can be created to modify the application's user interface – and, optionally, create a custom dashboard style – to match an organization's branding and aesthetics. To do so, a number of assets are required. An assortment of images and text-based configuration files are expected by the application.

Each package must contain images for all supported clients: Windows 8, Android, iOS, Windows Phone, and HTML. While perhaps daunting at first, many of the required files are similar – a large number of them consist of differently-sized and positioned logos, while many others may only require a simple color change. Furthermore, most image files are named using the same logical naming convention of Target OS-File description-Image dimensions.

In addition to images, a number of text-based configuration files must be edited in order to specify, among other things, the colors of various user interface elements – buttons, borders, text fields, etc. – and, optionally, a custom dashboard style.

Once all the images have been prepared and all configuration files edited, the assets will need to be assembled in a ZIP file and uploaded via the Control Panel web application.

Image assets

Each supported client uses a variety of specifically-sized and named assets. While some of these files are shared between clients, most of them exist solely for use in a discrete client. Supported clients are available on the following platforms: Windows 8, Android, iOS, and Windows Phone. For complete brand integration, the customization of several key HTML pages is also possible.

As mentioned above, the majority of the images are logos of various sizes, while others are solid-color icons. These icons can be easily modified to fit any custom brand by changing their color.

Note: Unless otherwise specified, all graphic files are assumed to be 32-bit PNG images.

Common

The following assets are shared between two or more different clients:

Filename	Dimensions	Description
Activity-42x42.png	42x42	Standard resolution activity button icon
Activity-42x42@2x.png	84x84 (42x42 * 200%)	High resolution activity button icon
BlankAvatarImage-80x80.png	80x80	Default user picture
Hub-Connected-32x32.png	32x32	Hub connection indicator
Hub-Disconnected-32x32.png	32x32	Hub "not connected" indicator
Server-Connected-64x64.png	64x64	Server connection indicator
Server-Default-64x64.png	64x64	Default server icon

Android

The following unique files are required by the Android client. Several different screen resolutions and configurations are supported.

Filename	Dimensions	Description
Android-AddNewServerIcon-30x30.png	30x30	Add New Server button icon
Android-Background-1200x1900.png	1200x1900	High resolution background image
Android-Background-768x1280.png	768x1280	Lower resolution background image
Android-Logo-320x64.png	320x64	Logo

iOS

The following unique files are required by the iOS clients. The application is a universal binary that runs on both iPhone and iPad. High-resolution ("Retina") graphics are identified by the filename suffix @2x.

Filename	Dimensions	Description
iOS7-AddNewServerIcon-42x42.png	42x42	Standard resolution Add Server button icon
iOS7-AddNewServerIcon-42x42@2x.png	84x84	Retina resolution Add Server button icon
iOS7-Background-320x568.png	320x568	Standard resolution iPhone background (portrait)
iOS7-Background-320x568@2x.png	640x1136	Retina resolution iPhone background (portrait)
iOS7-Background-568x320.png	568x320	Standard resolution iPhone background (landscape)
iOS7-Background-568x320@2x.png	1136x640	Retina resolution iPhone background (landscape)
iOS7-Background-768x1024.png	768x1024	Standard resolution iPad background (portrait)
iOS7-Background-768x1024@2x.png	1536x2048	Retina resolution iPad background (portrait)
iOS7-Background-1024x768.png	1024x768	Standard resolution iPad background (landscape)
iOS7-Background-1024x768@2x.png	2048x1536	Retina resolution iPad background (landscape)
iOS7-BackIcon-42x42.png	42x42	Standard resolution Back button icon
iOS7-BackIcon-42x42@2x.png	84x84	Retina resolution Back button icon

iOS7-Logo–225x45.png	225x45	Standard resolution logo
iOS7-Logo–225x45@2x.png	550x90	Retina resolution logo
iOS7-Menu–32x32.png	32x32	Standard resolution Menu button icon
iOS7-Menu–32x32@2x.png	64x64	Retina resolution Menu button icon
iOS7-Search–32x32.png	32x32	Standard resolution Search button icon
iOS7-Search–32x32@2x.png	64x64	Retina resolution Search button icon
iOS7-Settings–42x42.png	42x42	Standard resolution Settings button icon
iOS7-Settings–42x42@2x.png	84x84	Retina resolution Settings button icon

Windows 8

The following unique files are required by the Windows 8 client. Note that in addition to the standard naming convention, a resolution identifier (.100, .140, and .180) immediately precedes the file extension. These numbers represent the size of each image as a percentage of the standard resolution image and each asset should therefore be sized appropriately: 100%, 140%, and 180%. These images are then loaded automatically based on the client's display capabilities.

Filename	Dimensions	Description
Windows8-Background–1366x768.100.png	1366x768	Standard resolution background image
Windows8-Background–1366x768.140.png	1912x1075 (1366x768 * 140%)	Medium resolution background image
Windows8-Background–1366x768.180.png	2459x1382 (1366x768 * 180%)	High resolution background image
Windows8-Footer–220x22.100.png	220x22	Standard resolution "Powered by..." logo
Windows8-Footer–220x22.140.png	308x31 (220x22 * 140%)	Medium resolution "Powered by..." logo
Windows8-Footer–220x22.180.png	396x40 (220x22 * 180%)	High resolution "Powered by..." logo
Windows8-Logo–450x90.100.png	450x90	Standard resolution main logo
Windows8-Logo–450x90.140.png	630x126 (450x90 * 140%)	Medium resolution main logo
Windows8-Logo–450x90.180.png	810x162 (450x90 * 180%)	High resolution main logo
Windows8-SmallLogo–150x30.100.png	150x30	Medium resolution main logo

Windows8-SmallLogo -150x30.140.png	210x42 (150x30 * 140%)	Medium resolution smaller logo
Windows8-SmallLogo -150x30.180.png	270x54 (150x30 * 180%)	High resolution smaller logo
Windows8-Style-120x110.100.png	120x110	(Optional) Standard resolution style preview image
Windows8-Style-120x110.140.png	168x154 (120x110 * 140%)	(Optional) Medium resolution style preview image
Windows8-Style-120x110.180.png	216x198 (120x110 * 180%)	(Optional) High resolution style preview image

Important note

If no custom dashboard style is required, ensure that all Window8-Style files are omitted from the final brand package ZIP file. This includes the three PNGs as well as the Window8-Style.xaml config file.

Windows Phone

The following unique files are required by the Windows Phone clients. Note that while high-resolution image filenames are appended with the resolution identifier .150, this number does not reflect the image's scale. And unlike Windows 8 assets, standard resolution image filenames do not use a .100 identifier.

Filename	Dimensions	Description
WP-Background-480x853.png	480x853	Standard resolution background image
WP-Background-480x853.150.png	1080x1920	Very high resolution background image
WP-Logo-280x40.png	280x40	Standard resolution logo
WP-Logo-280x40.150.png	560x80	High resolution logo

HTML

Only three images are required for the two HTML pages: a standard resolution logo, a high resolution logo, and a spinner image. For the sake of ease, the two required logos can be renamed copies of the similar iOS assets, iOS7-Logo-225x45.png and iOS7-Logo-225x45@2x.png. The spinner image, a 32-bit PNG, is animated using a CSS rotation transform and is displayed very briefly, so a high resolution version of the asset is unnecessary. It is recommended that the spinner image be kept as it is, but recolored as appropriate.

Filename	Dimensions	Description
Html-Logo-225x45.png	225x45	Standard resolution logo
Html-Logo-225x45@2x.png	550x90	High resolution logo

Configuration files

In addition to the above image assets, a brand package requires two specific text-based configuration files. A third optional file, Windows8-Style.xaml, is needed if a custom dashboard style is desired.

- theme.json: a JSON file containing the name & value pairs for setting the colors of the client application's main user interface;
- Html-Style.css: a CSS file for styling the browser-based login and dashboard list pages;
- Windows8-Style.xaml: an optional XAML file for styling for creating custom dashboard styles & color palettes.

The first two config files, theme.json and Html-Style.css, should be clear enough to avoid the need for detailed instructions – the JSON file uses obvious property names and the CSS file is thoroughly commented throughout. The optional Windows8-Style.xaml file, however, is quite a bit more complex and therefore explained in the following section.

Creating a custom dashboard style

A custom brand package can be easily extended with the addition of a custom dashboard style: color palettes and UI styles that affect the appearance of the dashboards themselves.

In order to create a custom a style, a configuration file will require editing: Windows8-Style.xaml. XAML (Extensible Application Markup Language) is a Microsoft-developed XML-based language; if you're at all familiar with XML you'll recognize the syntax immediately. In addition to this configuration file, three PNG image assets, mentioned in the above Windows 8 section, are required. These are the style thumbnails used in the dashboard style dropdown menu.

The XML object described within the file is divided into three discrete parts: The main properties, the color palette, and the dashboard panel styles.

Important note

Properties not described in the list below should not be modified.

Main properties

The main, or root, properties of the style definition are further divided into three groups: general style colors, standard dashboard style colors, and accent dashboard style colors.

General style colors

Three hexadecimal properties handle general colors: TileBrush, PositiveValueBrush, and NegativeValueBrush. TileBrush controls the color of the dashboard tile, the small rectangle containing the title & thumbnail that appears in the dashboard list view. PositiveValueBrush and NegativeValueBrush control the colors for positive values and negative values throughout a dashboard. These are generally shades of green and red, respectively. Standard dashboard style colors

These properties handle the default appearance of dashboards using this style. Background, a hexadecimal value, is the dashboard's background color. TitleBrush and SubTitleBrush, both hex colors, are used for the main dashboard's title and subtitle. PanelVariant and PanelBackground, handle the appearance of dashboard panels – the boxes that surround dashboard controls. A dashboard panel can have its own title and subtitle colors, as described further below. Finally,

`SelectedForeground` and `SelectedBackground` handle the appearance of selected elements in Selection Lists (both List and Tree styles).

A string value, `PanelVariant`, tells the dashboard which base theme variant to use to draw the dashboard panels. Valid properties are "Standard" or "Accent". A "Standard" panel is considered to have a light-on-dark appearance, while an "Accent" panel has a "dark-on-light" color appearance. Choose the value most appropriate for this particular style.

Accent dashboard style colors

The properties handle the accent appearance of dashboards using this style. `AccentBackground`, a hexadecimal value, is the dashboard's background color. `AccentTitleBrush` and `AccentSubTitleBrush`, both hex colors, are used for the main dashboard's title and subtitle. `AccentPanelVariant` and `AccentPanelBackground`, handle the appearance of dashboard panels – the boxes that surround dashboard controls. A dashboard panel can have its own title and subtitle colors, as described further below. Finally, `AccentSelectedForeground` and `AccentSelectedBackground` handle the appearance of selected elements in Selection Lists (both List and Tree styles).

A string value, `AccentPanelVariant`, tells the dashboard which base theme variant to use to draw the dashboard panels. Valid properties are "Standard" or "Accent". A "Standard" panel is considered to have a light-on-dark appearance, while an "Accent" panel has a "dark-on-light" color appearance. Choose the value most appropriate for this particular style.

The color palette

At the heart of a dashboard style is the color palette definition. These are one or more comma-separated hexadecimal values representing the colors used by various controls – charts, gauges, maps – to visually identify data.

PaletteName

A globally unique, valid ID to represent the color palette, i.e. `PaletteName="MyCompanyGreen"` or `PaletteName="MyCompanyBlue"`.

ChartingDataPoints

These collections of hexadecimal values represent the colors used by a chart when it contains a specific number of data point series. If, for example, the control is given four series, it will draw the data points using the colors from the `ChartingDataPoints4` collection.

It's generally a good idea to create the largest collection, `ChartingDataPoints12`, first and then remove color for the lesser properties.

GaugeStops & MappingStops

These collections of three colors handle the colors for their respective controls.

`GaugeStops` contains the colors for the three different gauge sections in the following order: Positive, Caution, Negative. These are generally green, yellow, and red.

`MappingStops` contains the colors used by the map control to differentiate different data regions.

Using the same values found in `ChartingDataPoints3` is generally a good idea, aesthetically- speaking.

Dashboard panel styles

Due to the way the dashboard style parser is architected, four properties must be set to complete a dashboard styles, two each for the default dashboard panel style and the accent dashboard panel style. These properties handle the color of the title and subtitle within an individual panel. These properties are set among the two template code blocks, `PanelHeaderTemplate` and `AccentPanelHeaderTemplate`.

Title

To set the color of the title of a dashboard panel, locate and change the `Foreground` property of a `TextBlock` with the property `Text="{Binding Title}"` found within the corresponding `PanelHeaderTemplate` block.

SubTitle

To set the color of the subtitle of a dashboard panel, locate and change the `Foreground` property of a `TextBlock` with the property `Text="{Binding SubTitle}"` found within the corresponding `PanelHeaderTemplate` block.

Style thumbnail images

Once the XAML file has been edited, you'll need to make two images. These will act as thumbnails within the dashboard style dropdown menu. Standard resolution, medium resolution, and high-resolution images are required. Please refer to the image information described in the Windows 8 images list elsewhere in this document.

Creating the final brand package

Once all the various edits have been made, a single brand package file must be assembled. This is done by creating a ZIP archive containing all required files. These files must all be on the same level; the archive cannot contain sub-folders.

Once this file is created, it should be uploaded and assigned using the Control Panel web application. See [Managing Custom Branding](#) for more information.

Implementing Custom Maps

Geographic maps in Datazen are defined in a format known as *ESRI shapefiles*.

Initially designed by a private company, this is now a widespread semi-open format used in a large portion of GIS applications. In accordance with this format, Datazen requires two files to be provided when defining a map: a .SHP file for shape geometries, and a .DBF file for meta data. The base files names must match (e.g. canada.shp and canada.dbf). The meta data must include the field NAME with the value of the corresponding shape's name (key), to be used when populating the map with data.

For information on how to make custom maps available for inclusion in dashboards, see [Managing Custom Maps](#).

Information & Resources

More specific information can be found by visiting some of the following links.

Technical information

- The official specification: <http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf>
- The Wikipedia shapefile article: <http://en.wikipedia.org/wiki/Shapefile>

Creating & editing map geometry

Creation and editing shapefiles is a complex process that is beyond the scope of this document. Here are some resources and applications to help you get started:

- ArcGIS: <http://www.arcgis.com/>
- MAPublisher plug-in for Adobe Illustrator: <http://www.avenza.com/mapublisher>
- QuantumGIS (free): <http://www.qgis.org/>
- Manco ShapeFile Editor: <http://www.mancosoftware.com/ShapeFileEditor>

Existing shapefiles

Many existing shapefiles can be downloaded from the Web, from sites like these:

- Diva-GIS: <http://www.diva-gis.org/Data>
- OpenStreetMap: <http://openstreetmapdata.com/data>
- GeoCommons: <http://www.geocommons.com/>