Table of Contents

Glossary

A

attribute tree
Displays the attributes that are associated with the entity selected under the **entity tree**. For example, the Customer entity has attributes like Address, Phone, Country, etc. If no entity is selected, the attribute tree is empty.

D

drill down
Allows users to expand collapsed table rows to reveal more data. See also drill through.

drill through
Allows users to click links to reports with more detailed data. Located in the Design tab toolbar, the Drilldown button is enabled when you select a table cell or chart data point. Specify parameters in the target report to supply relevant detail data.

R

report design surface
Visible on the Design tab and the Report tab, it is a visual page designer where you can drag and drop entities and attributes to create tables and charts and design your reports.

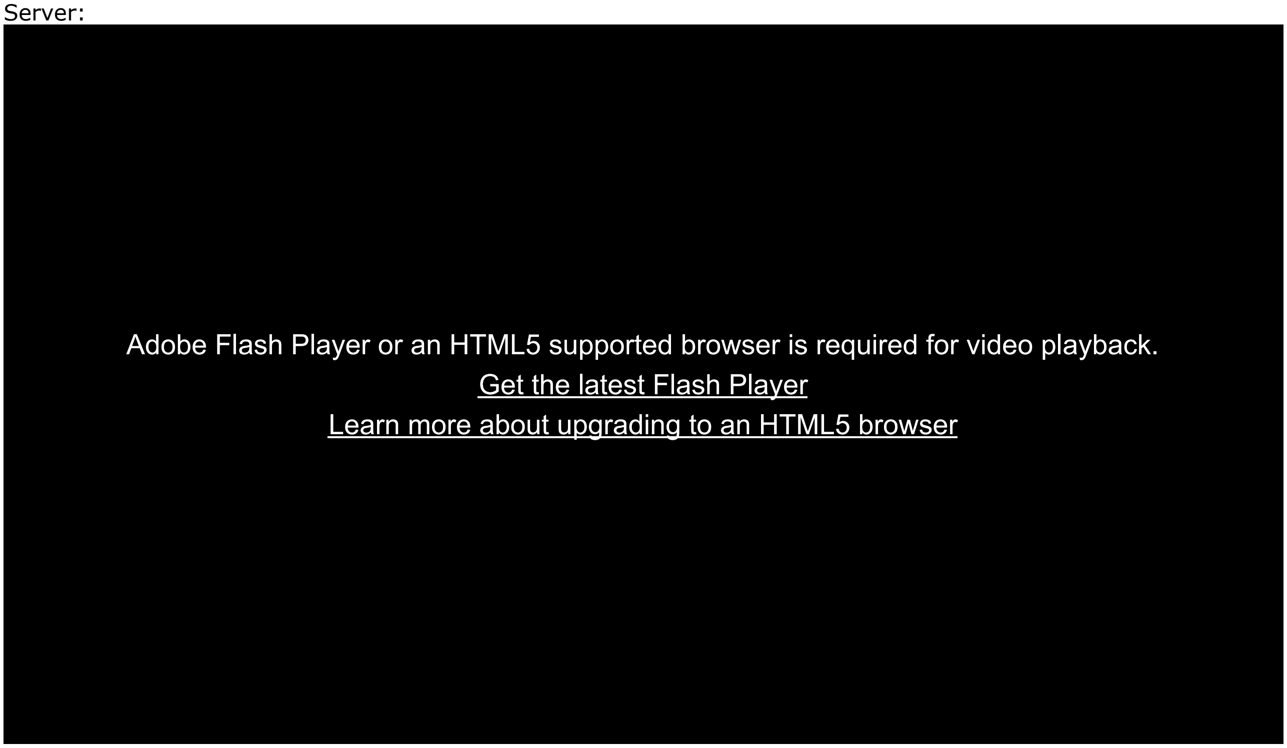
Report Info
An insert that you can add to a textbox to show page numbering, report name, or run time. Located in the Report tab toolbox, it is enabled when you click inside a textbox.

Installation

ActiveReports 8 Server allows you to provide a browser-based report designer for end users and to serve up both end user reports and your more sophisticated developer reports. This section helps you to get started.

> **Important**: During installation, if you choose to create new users for the Web App Pool and Service, be sure to update the user properties to match your password policy, as the default setting is to use an expiring password.

For a short video of the ActiveReports 8 Server installation process, please see Installing and configuring ActiveReports 8 Server:



In This Section

System Requirements
View the ActiveReports 8 Server managed agent system and server requirements.

AppPool Permissions (SQL Server)
Learn how to set up ActiveReports 8 Server so that you can use Windows authentication to create new data models with SQL Server.

Installed Files
Browse the files installed with ActiveReports 8 Server and brief descriptions of how they are used.

Logging On to the Administrator Site
Find out how to access the first page of the Administrator site.

Backup and Recovery
Learn which files to back up when you set up ActiveReports 8 Server and how to recover your data in the event of a problem.

Configuring ActiveReports 8 Server on SSL and HTTPS Web Sites
Find information on using ActiveReports 8 Server with secure Web sites.

> **Important**: If you use a virtual directory, you must copy the crossdomain.xml file (from C:\ActiveReports 8 Server\Site) to the full root of the parent application. If you have it only in the virtual directory, a Security sandbox violation occurs when you try to preview a report.

Using the Product

System Requirements

To install and use ActiveReports 8 Server, you must have compatible software installed on your server. If you have multiple servers set up for load balancing, the Managed Agent machines have fewer requirements than the Master Server machine. If you have a single-server deployment, the Site consists of a Master Server and a Managed Agent on the same machine.

Operating Systems

- **Server**: Windows Server 2003, 2008 and above (32- or 64-bit)
- **Managed Agent**: Windows XP and above; .NET Framework 3.5 SP1 and above

Windows 8

The following features must be turned on in Windows 8 in order to run ActiveReports 8 Server. From Control Panel, Programs and Features, open **Turn Windows features on or off** to ensure that these features are turned on. All are on by default.

- Internet Information Services / Web Management Tools / IIS Management Console
- Internet Information Services / World Wide Web Services / Application Development Features / ASP.NET 4.5
- Internet Information Services / World Wide Web Services / Common HTTP Features / Static Content
- .NET Framework 4.5 Advanced Services / WCF Services / HTTP Activation

Windows Server 2012

The following features must be turned on in Windows Server 2012 in order to run ActiveReports 8 Server. From Control Panel, Programs and Features, open **Turn Windows features on or off** to ensure that these features are turned on. All are on by default.

- Internet Information Services / World Wide Web Services / Application Development Features / ASP.NET 4.5
- .NET Framework 4.5 Advanced Services / WCF Services / HTTP Activation

Web Servers

- IIS6 with Windows Server 2003 (needs WWW Service and ASP.NET 2 registered)
- IIS7 with Windows Server 2008 (needs ASP.NET role service)
- IIS Management Console must be turned on.
- **.NET Framework**: 3.5 SP1 or higher

Internet Information Service (IIS) Ports

TCP/IP

- ActiveReports 8 Server 4040
- ActiveReports Managed Agent 5050

UDP

- ActiveReports 8 Server 7923, 26637, or 57323
- ActiveReports Installer 7979, 37637, and 15377

Database

- Microsoft SQL Server 2005, 2008 and above
- MySQL
- Oracle 10g and above

Note: For Oracle, the Oracle client must be installed on all agent machines.

Installand Uninstall

Installed Files

When you install ActiveReports 8 Server and use all of the default settings, files are installed in the following folders:

C:\ActiveReports 8 Server

File (or Folder) Name	Description
Agent	Contains the files that you use to deploy your solution to separate machines that host your Managed Agents if you do a multi-server deployment for load balancing.
Data	Contains user information and reports and models. This is the folder that you need to back up to prevent loss of data. For more information, see **Backup and Recovery**.
DataProviders	Contains data provider assemblies.
SDK	Contains server report controls and a class library for the software developer kit.
SecurityProviders	Contains the ActiveDirectorySecurityProvider assembly used to create a security provider.
Server	Contains the files used to run the server.
Site	Contains the files used to run both the Report Portal and the Admin sites, as well as all associated help files.
ActiveReports 8 Server License Agreement.rtf	The license agreement that dictates the terms under which you may use this software.

C:\ActiveReports 8 Server\SDK

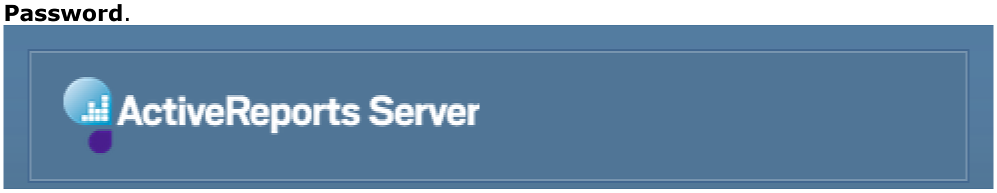
File Name	Description
ActiveReports.Server.ReportControls.dll	Assembly that supports the ReportDesigner and ReportList control for use in your ASP.NET solutions.
ActiveReports.Server.ReportControls.xml	Intellisense for the ReportControls assembly.
sdk.chm	Class library help file that provides information about the ReportControls assembly classes and their public methods and properties, and walkthroughs on their use.
Samples	Contains sample Visual Studio projects that you can use to evaluate the software, or customize and extend to use in your own solutions. For more information on samples, see the SDK help.

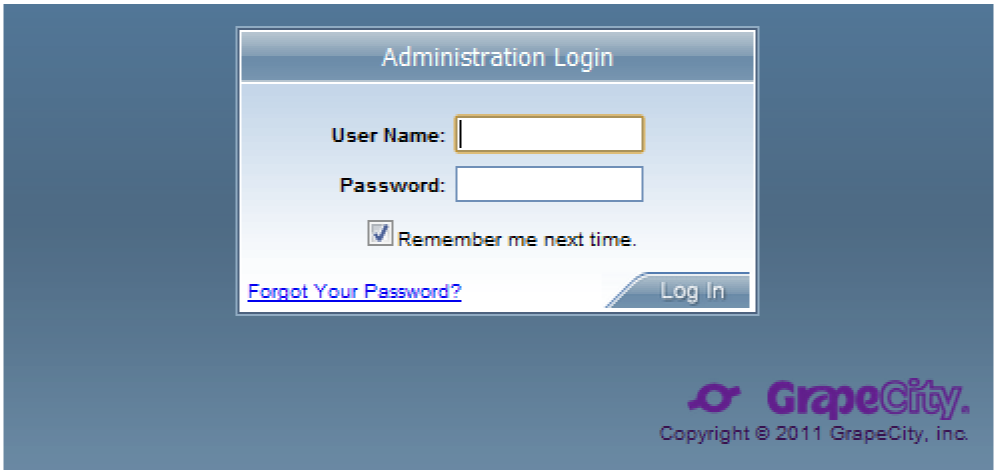
C:\ActiveReports 8 Server\Site

Folder Name	Description
Admin	Contains the files and folders used to run the Admin site, as well as the associated help file.
Bin	Contains ActiveReports and Data Dynamics Reports assemblies as well as SQLite and DotNetZip assemblies for use on the site.
Help	Contains the help file and supporting files for the Report Portal site.
web.config	The main file that contains configuration information for the Designer Web application.

Logging On to the Administrator Site

1. From the Start menu, in All Programs, select **GrapeCity**, then **ActiveReports 8 Server**, **Administration**.
2. In the Administration Login dialog that appears in your browser, enter your registered **User Name** and **Password**.





3. Click **Log In**. The Administrator Dashboard appears.

License Authentication

Backup and Recovery

It is always a good idea to back up your data so that you have a safe copy of it in the event of a disaster.

To back up and recover your data

1. Copy the contents of the `C:\ActiveReports 8 Server\Data` folder to a safe location.
2. Install a fresh instance of ActiveReports 8 Server with the same version as the one from which you backed up your data.
3. Copy the contents of the Data folder into the new `C:\ActiveReports 8 Server\Data` folder. The data is recovered.

Configuring ActiveReports 8 Server on SSL and HTTPS Web Sites

ActiveReports 8 Server does not require any special configuration to work under SSL. You can set up SSL and HTTPS on the ActiveReports 8 Server Web site just as you would for any Web site on your Web server. For help with this, see Microsoft's article on [Configuring SSL on a Web Server or Web Site (IIS 6.0)] or [How to Set Up SSL on IIS 7].

Once you set up the certificate and start using https:// to access the Web site on which ActiveReports 8 Server is installed, it begins working in secure mode.

Administrator Guide

ActiveReports 8 Server allows you to provide a browser-based report designer for end users and to serve up both end user reports and your more sophisticated developer reports.

To learn about what's new, please see the [ActiveReports 8 Server Releases] page on our web site.

In This Documentation

Features and Benefits
Read about the most important ActiveReports 8 Server features and how they make an administrator's life easier.

What's New In Version 8
Discover what has changed in this version of ActiveReports Server.

ActiveReports Limitations
Learn what limitations you will come across when working with ActiveReports in ActiveReports 8 Server.

Migration from Previous Versions
This section describes how to migrate all of your existing reports to the new version of ActiveReports Server.

License Types
Learn what features are included with each type of ActiveReports 8 Server license.

Installation
View the server and managed agent system requirements, installed files, and learn to log on.

Concepts
This section has a list of major features, and reviews general concepts to help you understand ActiveReports 8 Server.

Managing Models
This section has topics that show you how to work with models, including how create a new model and how to modify a model or its elements.

Managing Reports
This topic describes your options with each type of reports that are supported.

Managing Schedules
This topic describes shared schedules that you can set up to allow specified user roles to schedule reports.

Managing Themes and Styles

This section describes how to work with themes for entire reports, and styles for individual report items.

Managing Security
This section contains information about managing users, roles and permissions and about how to set up SSL/HTTPS on the ActiveReports 8 Server Web site.

Managing Configuration
This section teaches you about the ActiveReports 8 Server Agents, Licenses, Security, SMTP settings, notification URL, Audit settings and server diagnostics for your site.

How To
Quickly learn how to perform specific tasks with ActiveReports 8 Server.

Troubleshooting
Learn how to resolve some common issues with ActiveReports 8 Server.

Licensing Agreement

Please see the [ActiveReports 8 Server Licensing Agreement] on our web site for full details about licensing for each edition.

Acknowledgements

Microsoft, Windows, Visual Studio, and Microsoft SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Features and Benefits

Here are the most important ActiveReports 8 Server features that make an administrator's life easier.

Feature	**Benefit**
A report designer designed especially for end users that don't use SQL.	End users can create and modify reports without depending on IT.
Completely browser-based designer. No installation required.	Fast and easy deployment in production. No updates to users' desktops.
A logical data model from your database that users understand.	End users can query, sort, and filter data using familiar terms. No knowledge of database schema or SQL required.
Automatic generation of logical data models from your database.	IT can generate and then customize and extend the data models shown to end users.
Schedule & distribute reports to specific people at specific times.	Timely distribution of information to business users when they need it most.
Self-managed clustering & load-balancing.	Cost-effective and easy to manage scalability for a few or a hundred users.
Role-based security for reports and data models.	IT makes sure that only authorized users can create and view different reports.
Single sign-on security provider.	Allow users to sign in with their existing user name and password.
A set of ASP.NET controls for embedding reports and report designer.	Programmers can rapidly integrate reporting features into ASP.NET applications.
Flexible web service access to the reporting platform.	Programmers can exercise greater control integrating reporting into their applications.
Compatibility with ActiveReports.	ActiveReports 8 Server supports all types of reports provided by ActiveReports. Publish existing ActiveReports report files for centralized storage, distribution, security, and scalability.
Install and configure in less than 10 minutes.	Install and configure a working server in your own environment to evaluate in minutes - not days.

To see what's new and view the release notes for each build, please see Releases on our web site.

What's New In Version 8

We have made a number of changes since the last version of ActiveReports Server. Here are the major ones.

New Licensing

Our licensing now focuses on which features you use, rather than how many users you have. In ActiveReports 8 Server, you can scale up as much as you need to without purchasing additional agents, and the browser-based Designer is now sold as a separate Addon for those who need it. For more information, see **License Types**.

Support for Code-Based Section Reports

In addition to page reports (*.rdlx format) and XML-based section reports (*.rpx format), you can now serve up traditional code-based ActiveReports. For more information, see Using Code-Based Section Reports.

Parameters Support for Section Reports

In addition to support for parameters in page reports and end-user-created reports, we have added support for SQL query parameters as well as Report Explorer parameters in all section reports. For more information, see **Setting a Parameter in the Preview (on-line documentation)**.

HTML5 Viewer

The new HTML5 Viewer is a javascript component that you can use to show reports in a web browser. It includes a Mobile UI that is optimized for mobile devices, a traditional Desktop UI, and a Custom UI that developers can configure using the public API methods and properties. For more information, see **Previewing Reports in HTML5 Viewer (on-line documentation)** and How to Work with the HTML5 Viewer using Javascript.

Migration from Previous Versions of ActiveReports Server

We have ensured that you have an easy path to migrate all of your existing reports by allowing you to install the new version in the same folder over the previous version. The installer detects the previous version and suggests uninstalling it, and then adapts any existing agents, models, roles, groups, users, schedules, permissions, themes, reports, styles, audit settings and SMTP settings to work with ActiveReports 8 Server. For more information, see **Migration from Previous Versions**.

ActiveReports Limitations

If your developers use ActiveReports, they will come across the following limitations when working with ActiveReports in ActiveReports 8 Server.

Shared Data Source

The Shared Data Source, the ActiveReports file in RDSX format that contains data connection information, is not currently supported in ActiveReports 8 Server.

Drill-Down in FPL Reports

The drill-down in ActiveReports Fixed Page Layout reports is not supported in ActiveReports 8 Server, so you can view such reports without using this interactive feature.

Subreport Absolute Path

ActiveReports 8 Server does not support using the absolute path for subreports because ActiveReports 8 Server saves reports in the internal storage. In ActiveReports 8 Server you have to specify the subreport name to locate the subreport.

Master Reports

Master reports are not supported and cannot be uploaded to the Report Portal of ActiveReports 8 Server.

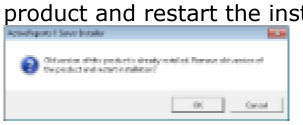
Migration from Previous Versions

You are provided with an easy migration from previous versions when you install the ActiveReports 8 Server.

> **Important:** Before following the upgrade steps in the live environment, we suggest that you test the upgrade in the test environment.

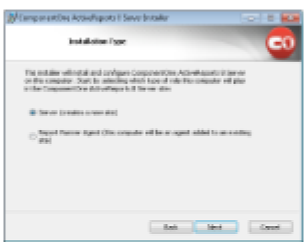
The new version of ActiveReports 8 Server installs in the same folder over the previous version. The installer detects the previous version and suggests uninstalling it, and then adapts any existing agents, models, roles, groups, users, schedules, permissions, themes, reports, styles, audit settings and SMTP settings to work with ActiveReports 8 Server.

1. Run the ActiveReports 8 Server installation file.
2. In the **ActiveReports 8 Server Installer** message that appears, click **OK** to remove the old version of the product and restart the installation.

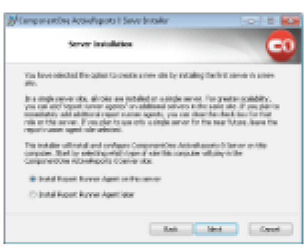


3. On the **End-User License Agreement** screen that appears, go through the terms in the License Agreement, select the check-box to accept them and click **Next** to continue with installation.

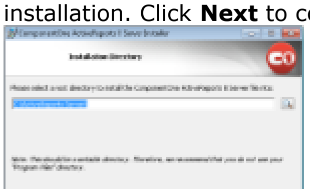
4. On the **Installation Type** screen that appears, select out of two options on the role the computer will play in the ActiveReports 8 Server site and click **Next** to continue with installation.
   - Server (creates a new site).
   - Report Runner Agent (this computer will be an agent added to the existing site). If you select this option, you will be taken directly to the **Installation Directory** screen (see below).



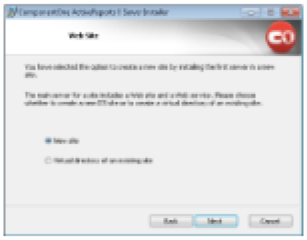
5. On the **Server Installation** screen, select one of the two options on the Report Runner Agent installation and click **Next** to continue with installation.
   - Install Report Runner Agent on this server.
   - Install Report Runner Agent later.



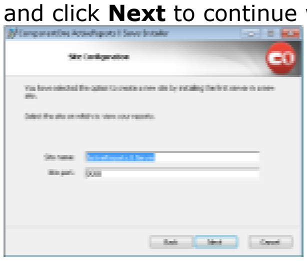
6. On the **Installation Directory** screen that appears, select the root directory for the ActiveReports 8 Server installation. Click **Next** to continue with installation.



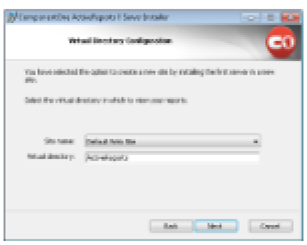
7. On the **Web Site** screen that appears, select one of the two options on how to create a new site and click **Next** to continue with installation.
   - New site. If you select this option, you will be taken to the **Site Configuration** screen.
   - Virtual directory of an existing site. If you select this option, you will be taken to the **Virtual Directory Configuration** screen.

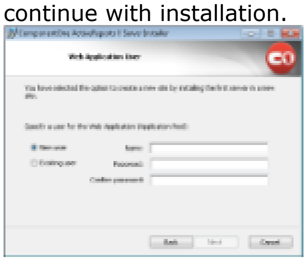
8. (The New site option) On the **Site Configuration** screen that appears, select the site to view the reports and click **Next** to continue with installation.

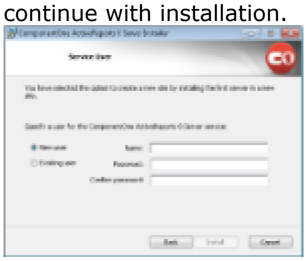
   (The Virtual directory of an existing site option) On the **Virtual Directory Configuration** screen that appears, select the virtual directory to view the reports and click **Next** to continue with installation.

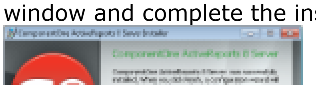
9. On the **Web Application User** screen that appears, specify a user for the web application and click **Next** to continue with installation.

10. On the **Service User** screen that appears, specify the user for the ActiveReports 8 Server service. Click **Next** to continue with installation.

11. Once the installation finishes, a screen notifying the completion of installation appears. Click **Finish** to close the window and complete the installation process.

Once the installation process is complete, you will be taken to the Administrator site to complete the configuration of ActiveReports 8 Server. There, you need to activate the license as the ActiveReports 8 Server license is not compatible with the ActiveReports Server product keys. For details, see **Activate a License**.

If you need a new product key for ActiveReports 8 Server, contact our sales team at sales@componentone.com.

For information on the ActiveReports 8 Server installation process, see **Installation**.

Note: The custom security provider settings get overwritten by the installation process, thus you have to manually reapply these settings after the installation is complete.

Note: We recommend clearing the cache after you have completed the upgrade to ActiveReports 8 Server.

License Types

There are three types of licenses for ActiveReports 8 Server:

ActiveReports 8 Server Trial Edition

This is a 30 day free trial version of the product that you can use to determine how ActiveReports 8 Server can fulfill your needs. The Trial Edition includes all of the features in the Core Edition plus the Designer Addon.

If the initial trial expires before you have had a chance to fully evaluate the product, you may contact sales@componentone.com to request an extension of the trial period.

ActiveReports 8 Server Core Edition

This license covers use of the server, the report portal, the HTML5 report viewer, and as many managed agents as you need. Once you purchase the Core Edition, there is no need to purchase additional licenses as you scale up to handle increased server loads.

With this license, you can schedule and distribute reports, take advantage of self-managed clustering and load-balancing, and provide role-based security for reports and data models. You can even allow users to sign in with their existing user name and password to access centrally stored ActiveReports report files.

ActiveReports 8 Server Designer Addon

In addition to serving up existing reports, if you want to offer your users the ability to create their own ad hoc reports right in the browser, you can purchase the optional Designer Addon.

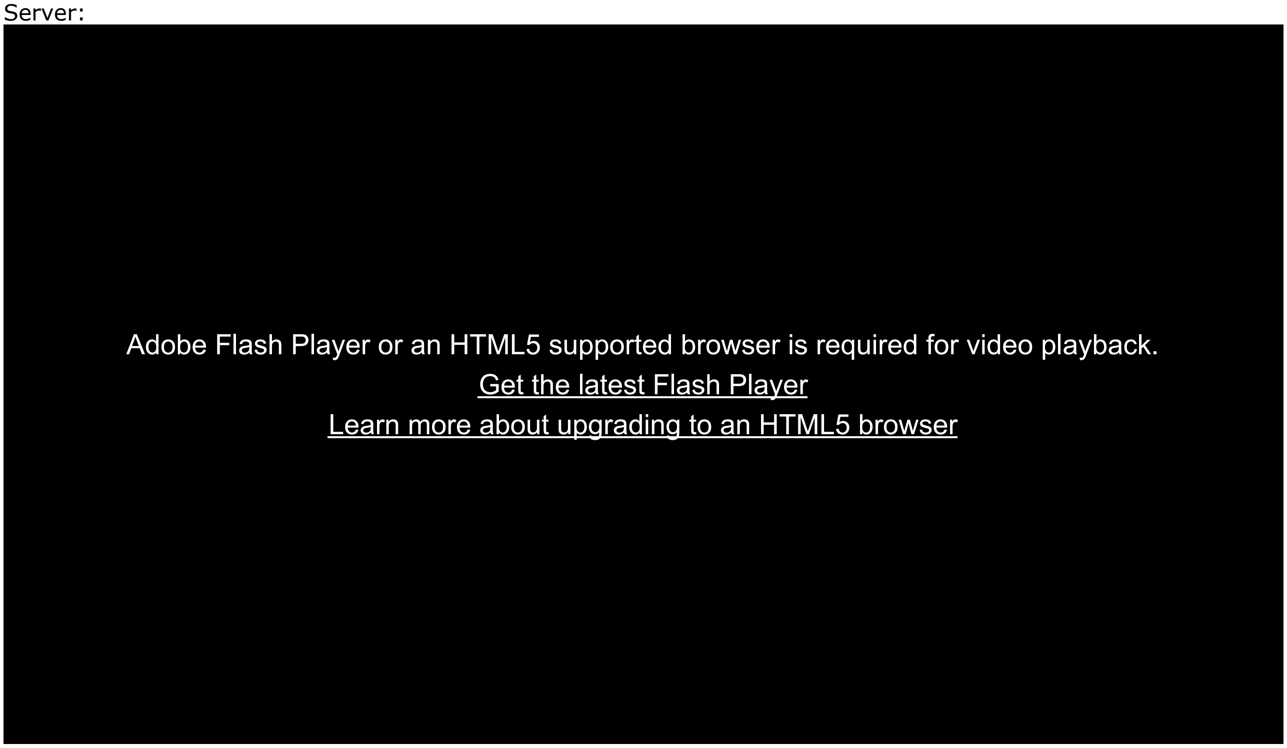
The Designer Addon is a completely browser-based report designer for end users who do not use SQL. End users can create and modify reports and even query, sort, and filter data with no special knowledge of database schema or SQL.

Installation

ActiveReports 8 Server allows you to provide a browser-based report designer for end users and to serve up both end user reports and your more sophisticated developer reports. This section helps you to get started.

> **Important**: During installation, if you choose to create new users for the Web App Pool and Service, be sure to update the user properties to match your password policy, as the default setting is to use an expiring password.

For a short video of the ActiveReports 8 Server installation process, please see Installing and configuring ActiveReports 8 Server:



In This Section

System Requirements
View the ActiveReports 8 Server managed agent system and server requirements.

AppPool Permissions (SQL Server)
Learn how to set up ActiveReports 8 Server so that you can use Windows authentication to create new data models with SQL Server.

Installed Files
Browse the files installed with ActiveReports 8 Server and brief descriptions of how they are used.

Logging On to the Administrator Site
Find out how to access the first page of the Administrator site.

Backup and Recovery
Learn which files to back up when you set up ActiveReports 8 Server and how to recover your data in the event of a problem.

Configuring ActiveReports 8 Server on SSL and HTTPS Web Sites
Find information on using ActiveReports 8 Server with secure Web sites.

> **Important**: If you use a virtual directory, you must copy the crossdomain.xml file (from C:\ActiveReports 8 Server\Site) to the full root of the parent application. If you have it only in the virtual directory, a Security sandbox violation occurs when you try to preview a report.

System Requirements

To install and use ActiveReports 8 Server, you must have compatible software installed on your server. If you have multiple servers set up for load balancing, the Managed Agent machines have fewer requirements than the Master Server machine. If you have a single-server deployment, the Site consists of a Master Server and a Managed Agent on the same machine.

Operating Systems

- **Server**: Windows Server 2003, 2008 and above (32- or 64-bit)
- **Managed Agent**: Windows XP and above; .NET Framework 3.5 SP1 and above

Windows 8

The following features must be turned on in Windows 8 in order to run ActiveReports 8 Server. From Control Panel, Programs and Features, open **Turn Windows features on or off** to ensure that these features are turned on. All are on by default.

- Internet Information Services / Web Management Tools / IIS Management Console
- Internet Information Services / World Wide Web Services / Application Development Features / ASP.NET 4.5
- Internet Information Services / World Wide Web Services / Common HTTP Features / Static Content
- .NET Framework 4.5 Advanced Services / WCF Services / HTTP Activation

Windows Server 2012

The following features must be turned on in Windows Server 2012 in order to run ActiveReports 8 Server. From Control Panel, Programs and Features, open **Turn Windows features on or off** to ensure that these features are turned on. All are on by default.

- Internet Information Services / World Wide Web Services / Application Development Features / ASP.NET 4.5
- .NET Framework 4.5 Advanced Services / WCF Services / HTTP Activation

Web Servers

- IIS6 with Windows Server 2003 (needs WWW Service and ASP.NET 2 registered)
- IIS7 with Windows Server 2008 (needs ASP.NET role service)
- IIS Management Console must be turned on.
- **.NET Framework**: 3.5 SP1 or higher

Internet Information Service (IIS) Ports

TCP/IP

- ActiveReports 8 Server 4040
- ActiveReports Managed Agent 5050

UDP

- ActiveReports 8 Server 7923, 26637, or 57323
- ActiveReports Installer 7979, 37637, and 15377

Database

- Microsoft SQL Server 2005, 2008 and above
- MySQL
- Oracle 10g and above

> **Note**: For Oracle, the Oracle client must be installed on all agent machines.

AppPool Permissions (SQL Server)

In order to generate data models using Windows authentication with Microsoft SQL Server data, the AppPool used for ActiveReports 8 Server needs proper credentials to access the database.

This is set during installation with the user that you assign for the Web Application Pool.

If the user you assign for the application pool does not have permission to access the database, when you try to create a new model using Windows Authentication, you get an error like:

> "The connection test failed. Login failed for user 'YourUserName'."

You can avoid this error by setting the AppPool account with a user who has permissions to the data you want to use, or by opting to use Server Authentication when you create a data model. See **Creating a New Model** for more information.

Installed Files

When you install ActiveReports 8 Server and use all of the default settings, files are installed in the following folders:

C:\ActiveReports 8 Server

File (or Folder) Name	Description
Agent	Contains the files that you use to deploy your solution to separate machines that host your Managed Agents if you do a multi-server deployment for load balancing.
Data	Contains user information and reports and models. This is the folder that you need to back up to prevent loss of data. For more information, see **Backup and Recovery**.
DataProviders	Contains data provider assemblies.
SDK	Contains server report controls and a class library for the software developer kit.
SecurityProviders	Contains the ActiveDirectorySecurityProvider assembly used to create a security provider.
Server	Contains the files used to run the server.
Site	Contains the files used to run both the Report Portal and the Admin sites, as well as all associated help files.
ActiveReports 8 Server License Agreement.rtf	The license agreement that dictates the terms under which you may use this software.

C:\ActiveReports 8 Server\SDK

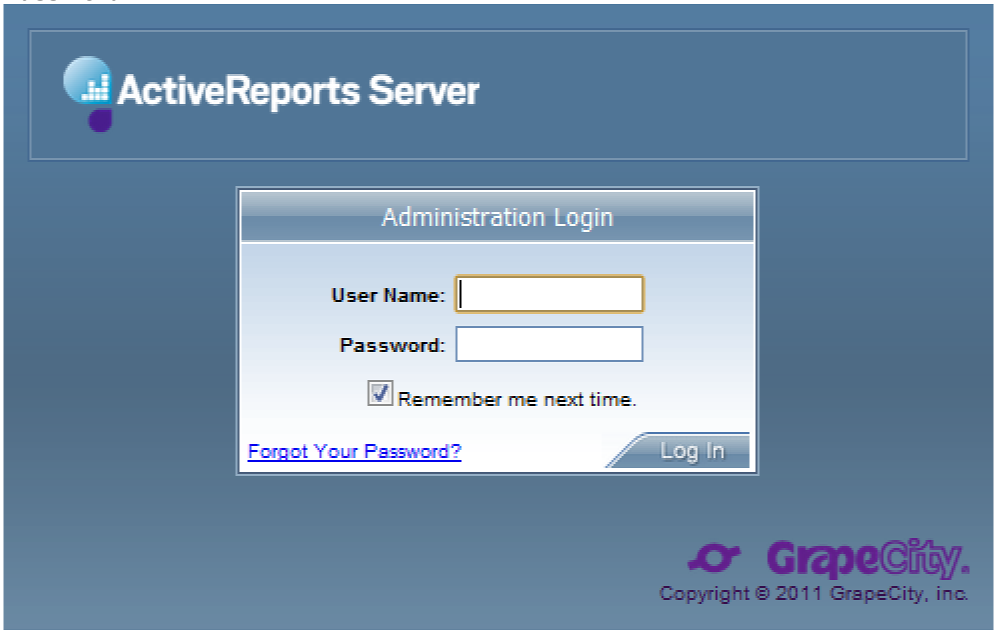
File Name	Description
ActiveReports.Server.ReportControls.dll	Assembly that supports the ReportDesigner and ReportList control for use in your ASP.NET solutions.
ActiveReports.Server.ReportControls.xml	Intellisense for the ReportControls assembly.
sdk.chm	Class library help file that provides information about the ReportControls assembly classes and their public methods and properties, and walkthroughs on their use.
Samples	Contains sample Visual Studio projects that you can use to evaluate the software, or customize and extend to use in your own solutions. For more information on samples, see the SDK help.

C:\ActiveReports 8 Server\Site

Folder Name	Description
Admin	Contains the files and folders used to run the Admin site, as well as the associated help file.
Bin	Contains ActiveReports and Data Dynamics Reports assemblies as well as SQLite and DotNetZip assemblies for use on the site.
Help	Contains the help file and supporting files for the Report Portal site.
web.config	The main file that contains configuration information for the Designer Web application.

Logging On to the Administrator Site

1. From the Start menu, in All Programs, select **GrapeCity**, then **ActiveReports 8 Server**, **Administration**.
2. In the Administration Login dialog that appears in your browser, enter your registered **User Name** and **Password**.



3. Click **Log In**. The Administrator Dashboard appears.

Backup and Recovery

It is always a good idea to back up your data so that you have a safe copy of it in the event of a disaster.

To back up and recover your data

1. Copy the contents of the `C:\ActiveReports 8 Server\Data` folder to a safe location.
2. Install a fresh instance of ActiveReports 8 Server with the same version as the one from which you backed up your data.
3. Copy the contents of the Data folder into the new `C:\ActiveReports 8 Server\Data` folder. The data is recovered.

Configuring ActiveReports 8 Server on SSL and HTTPS Web Sites

ActiveReports 8 Server does not require any special configuration to work under SSL. You can set up SSL and HTTPS on the ActiveReports 8 Server Web site just as you would for any Web site on your Web server. For help with this, see Microsoft's article on Configuring SSL on a Web Server or Web Site (IIS 6.0) or How to Set Up SSL on IIS 7.

Once you set up the certificate and start using https:// to access the Web site on which ActiveReports 8 Server is installed, it begins working in secure mode.

Concepts

This section reviews the most common concepts to help you understand ActiveReports 8 Server and allow you to create and serve reports more efficiently.

Logical Data Models
In ActiveReports 8 Server, data is organized in data models. This topic gives a general overview of a data model and its components.

Reports
End users can create and save reports in the ActiveReports 8 Server Reporting Portal. Administrators can serve both developer reports and end user reports.

Styles and Themes
Review the elements that contribute to the visual presentation of a report.

Security: Users and Roles
Browse an overview of the ActiveReports 8 Server role-based security.

Agents and Scalability
Find out how ActiveReports 8 Server uses agents to handle more concurrent users and higher report execution loads.

Single Sign-On with LDAP or Active Directory
Find information on mapping LDAP or Active Directory users to ActiveReports 8 Server.

Logical Data Models

A **logical data model** is an abstract model that organizes data into separate entities with defined relationships between entities. Each entity has its own set of attributes. A user selects a data model from a list of available data models, and then uses it as a basis for creating reports.

Entities

Entities represent business objects (e.g. Contact, Product, etc.) that are included in a data model. Entities are represented to the end user in a multi-level tree. Each entity has an associated set of attributes.

You can create **Ad Hoc Entities** by writing SQL queries and binding them to a logical table, but certain restrictions apply.

Attributes

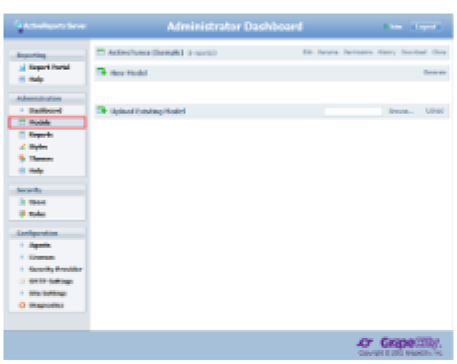
Attributes are associated with an entity and represent values that are used for creating a report. A **calculated field** is a type of attribute which calculates a value based on other attributes.

Relations

A relation is an element of a data model that represents relationships between entities.

Relation Cardinality dictates how relations are created.

As administrator, you can create new models, delete models, and edit models to provide what your end users need in order to create meaningful reports. See the **Managing Models** section for information on how to perform these tasks.



Ad Hoc Entities

An ad hoc entity is one that is bound to a dynamic SQL query, for example, a stored procedure or a view, rather than a query built by ActiveReports 8 Server. You can use an ad hoc entity in the same ways as any other entity, except that you cannot add a logical column, a calculation, to the base table of an ad hoc entity. You can, however, add calculated attributes in the SQL query itself. You can also add attributes that bind to the existing columns, and you can add relations to other entities.

An example of a query that creates an ad hoc entity from the included ActiveTunes (Sample) model is as follows.

SQL Query

```
SELECT Address, City, State FROM Customer
```

Creating Ad Hoc Entities

Any limitations that apply to the underlying data also apply to your ad hoc entity.

- In SQL subqueries, you can have up to 32 levels of nesting (or less depending on memory resources and the complexity of other expressions in the query).
- Microsoft SQL Server does not allow the TOP function in nested queries.

If you try to create an entity using a query that exceeds your underlying data's limits, ActiveReports 8 Server displays an appropriate error.

Tip: When editing queries for ad hoc entities, provide all attributes that are currently used by the entity. This way, if you want to exclude an attribute from the SQL query of the ad hoc entity at run time, you can delete it before the edit operation.

Take care when changing these queries, because you can change the actual data for attributes without changing the semantic model (except the entity edits). This could cause data in some reports to change dramatically.

Relation Cardinality

Relation cardinality dictates how relations are generated when you create a logical model, and how they are used to generate underlying SQL JOIN queries when you run reports. Generally, the cardinality is set in the underlying database, but if it is not set up properly there, you can set the cardinality of a relation from the Administrator Dashboard in the Model Editor. For more information, see **Modifying a Relation**.

Relation Types

When you add relations between entities, ActiveReports 8 Server uses rules to determine cardinality. There are four ways that entities can relate:

- **One**: One-to-one
- **Many**: Many-to-many
- **OptionalOne**: One-to-zero or one
- **OptionalMany**: Many-to-zero or more

If you add relations between entity A (from table A) and entity B (from table B) using the corresponding sets of tables in AC and BC, the rules for determining cardinality are as follows.

Adding relation AB to entity A

- If there is a unique constraint on table B that is nullable and exactly matches the columns in BC: **OptionalOne**.
- If there is a unique constraint on table B that is not nullable and exactly matches the columns in BC: **One**.
- Otherwise: **OptionalMany**. (ActiveReports 8 Server does not use **Many** in creating relations, but you can set it explicitly.)

Adding relation BA to entity B

- If there is **no** unique constraint on table A that matches the columns in AC: **OptionalMany**.
- If there is a unique constraint on table B that matches the columns in BC, and one of those columns is nullable: **OptionalOne**.
- If there is a unique constraint on table B that matches the columns in BC, and **none** of those columns is nullable:

One.

Example

We can add a relation between the Album and Artist tables using the ArtistID column. Assuming this relation does not exist in the database yet, if we add the relation from the Album entity, we get an Album-to-Artist relation with OptionalOne cardinality and an Artist-to-Album relation with OptionalMany cardinality.

SQL JOIN Types

ActiveReports 8 Server does not take role cardinality into account in determining what JOIN type to use. The relation in the data source controls the behavior.

- If the relation belongs to the table that the query joins to, then it uses a LEFT JOIN.
- If there is at least one column in the relation that accepts a null value, then it uses a LEFT OUTER JOIN.
- Otherwise, it uses an INNER JOIN.

Basically, if it accepts null values, it uses outer joins, otherwise it uses inner joins.

Examples

The Album table has a foreign key constraint linked with the Artist table, there is a relation created for it, and it belongs to the Album table. So long as Album.ArtistID does not accept null, then for queries joining the Album table to the Artist table, we use an INNER JOIN, but for queries joining the Artist table to the Album table, we use a LEFT JOIN. However, if Album.ArtistID is nullable, then even in the case of queries joining the Album table to the Artist table, we use a LEFT OUTER JOIN.

Consider relations based on foreign constraints between Playlist, Track and PlaylistTrack tables via PlaylistID and TrackID. The relations belong to the PlaylistTrack table as they refer to Playlist and Track. Assume for this purpose that CollapseInRelations is not set for PlaylistTrack.

If you drop Playlist.Name and Track.Name to create a table, then the query tree looks like this:

PlaylistTrack

> Playlists

> Tracks

In both cases, it uses an INNER JOIN, as both relations belong to the table (PlaylistTrack) from which the query pulls.

In the case of the Album and Artist tables, the relation connects an Album to an Artist by the ArtistID. So you can drop the Artist.Name attribute and a "Total Albums" aggregate to create a table. If you drop the Name attribute first, then the query looks like this:

Artist

> Album

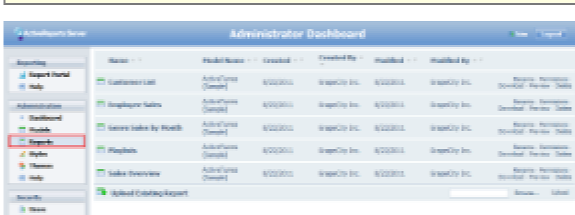
In this case, it uses a LEFT OUTER JOIN because the relation belongs to the table (Album) from which the query pulls.

Reports

End users can create reports in the ActiveReports 8 Server Reporting Portal. When they save them in the Reporting Portal, you can see them listed on the **Reports** page of the Administrator Dashboard, where you can modify, delete, download, or set permissions on them.

In addition to end user reports, you can also serve more complex developer reports created with ActiveReports. See **Managing Reports** for information on adding developer reports.

> **Note**: Developer reports are visible to end users, but they can only download or preview them, and cannot open them in Design view.

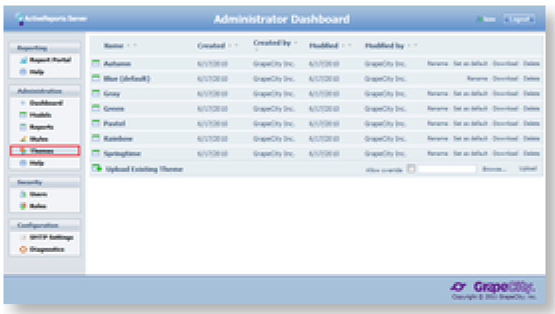


For more information on report options in the Administrator Site, refer to **Managing Reports**.

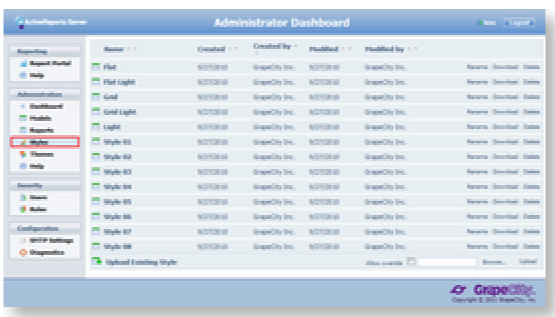
Styles and Themes

Styles and themes allow your end users to control the visual presentation of the reports they create. You can control which styles and themes are available to your users.

Themes determine the overall look of a report, and change the appearance of all report items: tables, charts, etc. The end user accesses themes on the Report tab of the Reporting Portal.



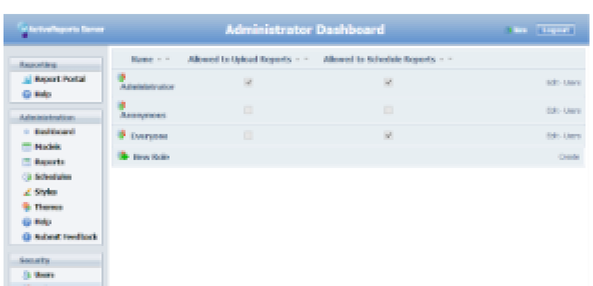
Styles can be applied to individual report items. Some styles apply only to the Table report item, and the end user accesses them on the Table Design Tools tab. Others apply only to the Chart report item, and the end user accesses them on the Chart Design Tools tab.



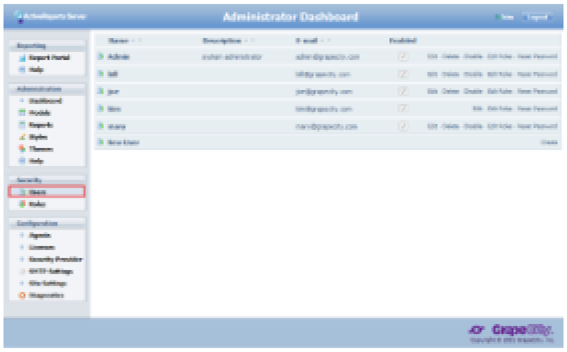
You can upload, download, modify, and delete styles and themes, and control which are used by default. For more information, see **Managing Themes and Styles**.

Security: Users and Roles

ActiveReports 8 Server employs role-based security, which means that permissions are assigned to roles, rather than to individual users. To allow a user to access specific report options, you must include them in a role group that has the proper set of permissions. In the Security section of the dashboard, click **Roles** to administer roles.



As Administrator, you can create roles and users. You then add users to each role and control which roles are allowed to upload ActiveReports and which roles have Read access to each model. In the Security section of the dashboard, click **Users** to administer users. For more information, see **Managing Security**.



Agents and Scalability

Agents allow you to serve reports to more concurrent users, and respond to higher report execution loads.

Scale-Out Deployment

Using multiple report server instances that share a single report server database is called scale-out deployment. You can add Managed Agents to your deployment that the central controller server uses to automatically load-balance report execution tasks.

Licensing

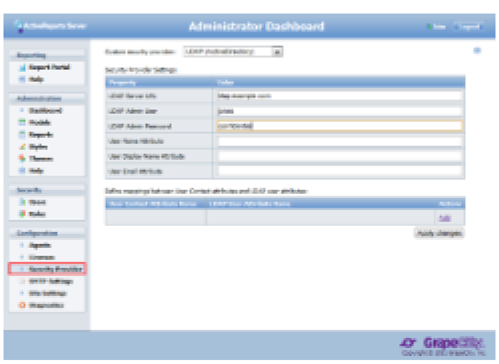
When you purchase the ActiveReports 8 Server Edition, you can use as many agents as you need.

Deployment Scenarios and Licensing Requirements

For optimal scalability, it is best practice to run agents on separate machines from the main host. But if your focus is on having less hardware to manage and keeping costs down, you can keep the agent on the same machine as the host.

Single Sign-On with LDAP or Active Directory

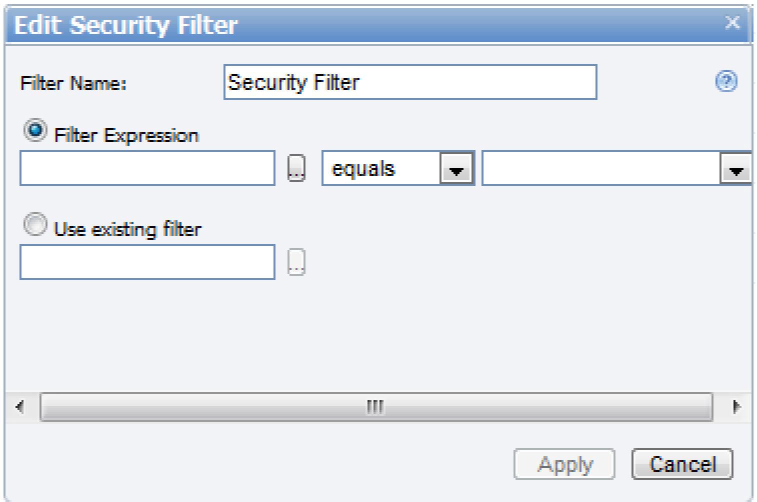
If your user information is already stored in a corporate directory that uses Lightweight Directory Access Protocol (LDAP) or Active Directory®, you can connect your ActiveReports 8 Server users and roles to the directory by configuring it on the Security Provider page.



For information on how to set up your LDAP or Active Directory in ActiveReports 8 Server, see the **Managing Security Providers** and **Using LDAP and Active Directory** topics.

> **Note**: ActiveReports 8 Server supports OpenLDAP v2.3 or later. Earlier versions are not supported because they do not have a MemberOf attribute.

Once you have the settings and mappings in place, when you edit a model, you can add a security filter to each model entity. (See the **Modifying an Entity** topic for more information.)



In the Filter Expression, the value on the right side gives you a list of the attributes that are mapped to your LDAP or Active Directory.

Managing Models

This section contains information that helps Administrators with data management tasks.

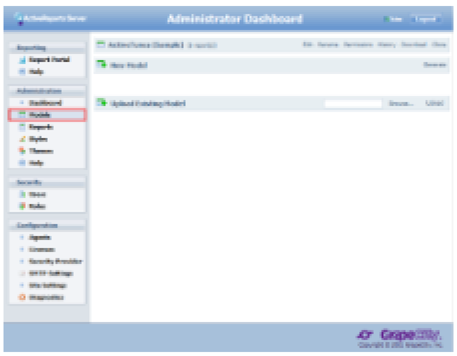
Model List Overview
This section describes how to maintain the list of data models from which Reporting Portal users create their reports.

Model Editor Overview
Refer to this section to learn about the ways in which you can modify individual data models.

> **Note**: You can create row-level security by creating custom security providers. See the SDK documentation and the **Managing Single Sign-On Security Providers** topic for more information.

Model List Overview

You can view and modify all of the data models you have available for creating reports on the Administrator Dashboard Models page.



Creating a New Model
Learn to create a new data model with the wizard.

Cloning a Model
Learn to copy an existing data model.

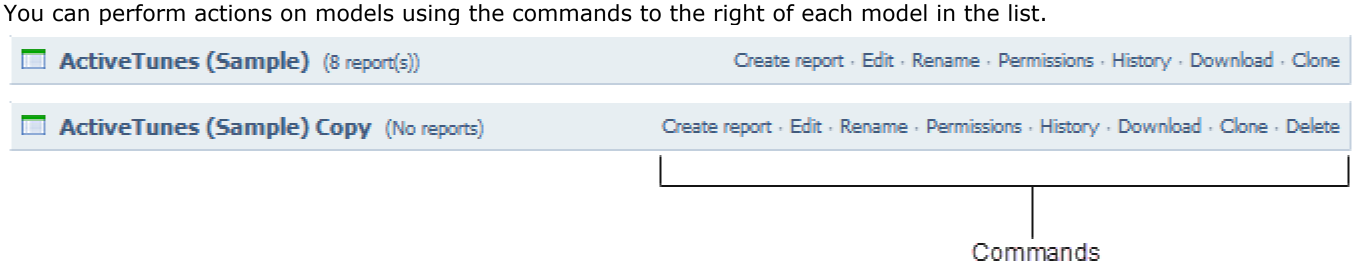
Working with Model Versions
Learn to use versioning to track changes to your data model. This allows you to revert to previous versions.

Deleting a Model
Learn to delete data models.

Uploading and Downloading Models
Learn to upload and download data models for backing up or transferring them.

Model Commands

You can perform actions on models using the commands to the right of each model in the list.



Command Descriptions

Command	Description
Create report	Opens the report designer with the data model already selected.
Edit	Opens the Model Editor so that you can modify the model's properties. For more information, see

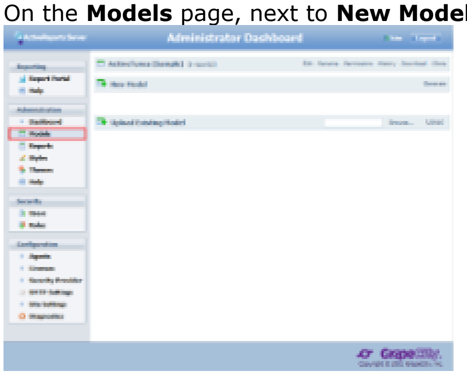
	Model Editor Overview.
Rename	Allows you to change the model's name.
Permissions	Opens the **Change Permissions** dialog where you can select which roles have access to the model.
History	Displays a list of modifications by date. If you do not supply comments when you modify it, a description of the modification is automatically generated.
Download	Downloads and saves a model to the local file system.
Clone	Creates a copy of the latest version of the model, without any of the reports.
Delete	Deletes the model. This command is not available for the sample model, ensuring that there is always at least one model available.

Creating a New Model

Create a new data model using the model generation wizard.

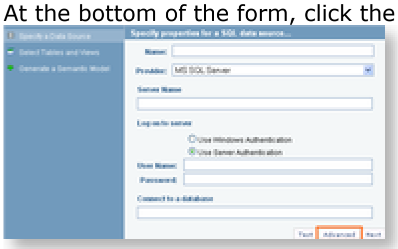
To create a new model

1. From the Administrator Dashboard, select **Models**.
2. On the **Models** page, next to **New Model**, click **Generate**.



3. In the new model wizard that appears, enter a **Name** for this new data source and the database server information in the provided fields.
To enter the connection string directly in the Connection String Editor dialog

   a. At the bottom of the form, click the **Advanced** button.



Sample connection string for SQL server

```
Provider=SQLCLIENT;Data Source=HQ;Initial Catalog=PUBS;User Id=myUsername;Password=myPassword;
```

Sample connection string for Oracle

```
Provider=ORACLE;Data Source=MyServiceName;User Id=myUsername;Password=myPassword;
```

   b. Click **Accept** to save the connection string for the new model.

Tip: You can use any UserContext attribute in the connection string by putting the attribute name between percent signs. For example,

```
Provider=SQLOLEDB.1;Data Source=HQ;Initial Catalog=%TenantDatabase%;
User Id=myUsername;Password=myPassword;
```

4. Click **Next** to proceed to the **Select Tables and Views** page of the wizard.
5. Click check boxes to select items from the list of available tables and views. You can select all visible items at once by selecting the **Select/Deselect All Visible Items** check box.

Notes: If you have a lot of items, you can filter what is shown by entering a value in the **Filter Items By Name** box.

If you do not select any items in this step, you can generate an empty model without any data, and later add data to the model manually. See **Adding a New Entity**, **Adding a New Logical Column (Attribute)** and **Adding a New Logical Relation** for details.

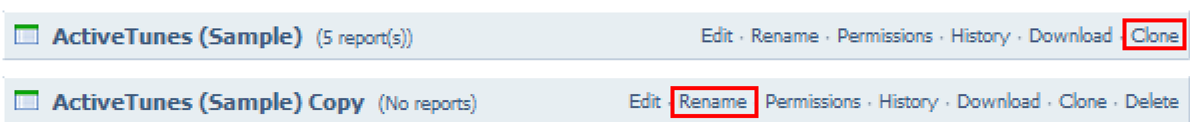
6. Click **Next** to finish the model generation wizard. A new model is generated and appears in the **Models** list, available to users for creating reports in the Reporting Portal.

Cloning a Model

Another way to create a new model is by cloning. In this case, we choose a model from the list of current models and create its precise copy as an independent model that we can edit.

To clone a model

1. From the Administrator Dashboard, select **Models**.
2. Next to the model in the list that you want to copy, click **Clone**. A new model is created and displayed in the Models list.



3. To rename the newly created model, click **Rename**, and enter new text.
4. Click **Update** to apply the change.

Note: When you clone a model, the reports remain bound to the initial model. A newly cloned model is created without any bound reports.

Working with Model Versions

Model versioning allows you to keep and manage various versions of a model, so that you can track modifications and easily revert to any version of the model at any time. ActiveReports 8 Server always uses the latest version of the model to create reports in the Reporting Portal.

To save a model version

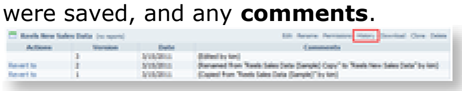
When you upload an existing model, ActiveReports 8 Server saves it in the version history the same as when you modify it in the model editor.

1. After you finish **Saving a Modified Model**, click the **Save & Publish** button. This creates a new version of the model. The old version, as well as all other previous model versions, is kept in the **History** section of the model.
2. All reports associated with the model automatically bind to the latest model version. Model changes, especially deletions, may break some reports, so compatibility is checked automatically.

Note: See **Managing Model Breaking Changes** for more information on how broken reports are handled.

To view the model history

1. In the Models list, next to the model whose history you want to view, click **History**.
2. The **History** section of the model drops down to display a numbered list of model versions, the dates the changes were saved, and any **comments**.

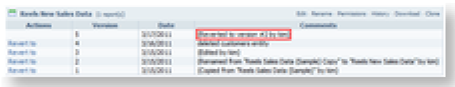


3. To hide the expanded History section, click **History** again.

To revert to an historic model version

1. Click **History** to expand the History section of the model you want to revert.
2. In the list of model versions, next to the version that you want, click **Revert to**.
3. A copy of the selected version of the model becomes the latest version, and a compatibility check runs for all associated reports.

> **Note**: If a report is not compatible with the selected model version, the compatibility check continues until a compatible model version is found.



Deleting a Model

You can delete any model that does not have any associated reports.

To delete a model

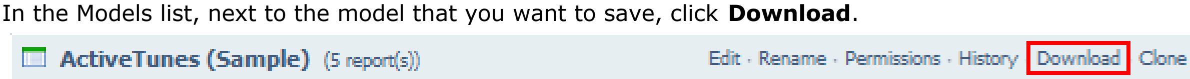
1. Delete any reports associated with the model to enable the delete function. (See **Managing Reports** for more information.)
2. In the Models list, next to the model that you want to delete, click **Delete**.

Uploading and Downloading Models

The upload and download model operations offer an additional means of backing up or transferring models between servers without requiring access to the server console. To streamline upload and download times, models are saved as specially formatted and compressed xml with a file extension of .datamodel.

To download a model

1. In the Models list, next to the model that you want to save, click **Download**.



2. The compressed xml model file is copied onto your machine in .datamodel format.

> **Tip**: If your browser does not automatically offer you a choice of where to save the file, you can right-click **Download** and select **Save As**.

To upload a model

1. In the Models list, next to **Upload Existing Model**, click **Browse** and navigate to the location of a locally saved compressed xml model file.

> **Tip**: Supported file extensions are .datamodel, .xml, and .smdl, so long as the file uses valid semantic

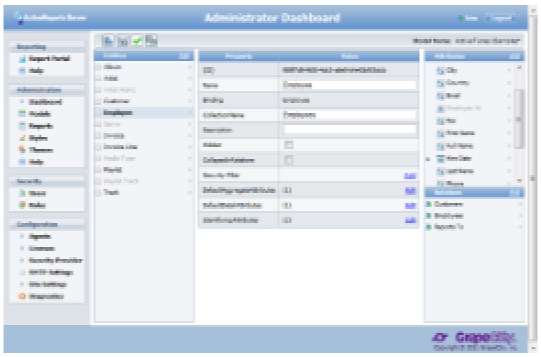
model data.

2. Select the file to upload and click **Open**. The dialog closes and the file name appears in the box.
3. Click **Upload** to add the model from your computer to the server.

Model Editor Overview

To access the Model Editor, next to a model in the Models list, click **Edit**.

The Model Editor allows you to modify a model's entities, attributes and relations. An asterisk (*) by the model name in edit mode indicates that the model has been modified. Select an entity from the **Entities** list on the left to edit it.



Modifying an Entity
Learn which properties you can modify on each entity.

Adding a New Logical Table (Entity)
Learn to add new logical tables and to base new tables on existing entities.

Modifying an Attribute
Learn which properties you can modify on an entity's attributes.

Adding a New Logical Column (Attribute)
Learn to add new logical columns and to base new columns on existing attributes.

Modifying a Relation
Learn which properties you can modify on an entity's relations.

Adding a New Logical Relation
Learn to add new logical relations and to base new relations on existing ones.

Deleting an Item
Learn to delete any entity, attribute, or relation.

Saving a Modified Model
Learn to save a working draft of a model and to publish model changes.

Entering Modification Comments
Learn to add comments that help you to identify versions in the model history.

Managing Model Breaking Changes
Learn about changes that may break reports, and how ActiveReports 8 Server handles them.

Model Editor Toolbar Buttons

- **Save & Publish** validates and, if no errors are found, saves modifications as a new **version** of the model.
- **Save Draft** saves modifications as a local draft copy available only to the user making the modifications. (Use this button to save a draft of a model that does not pass validation.)

- **Validate** checks modified elements for errors.
- **Properties** displays model properties such as the model name and connection string information.

Changing the Connection String

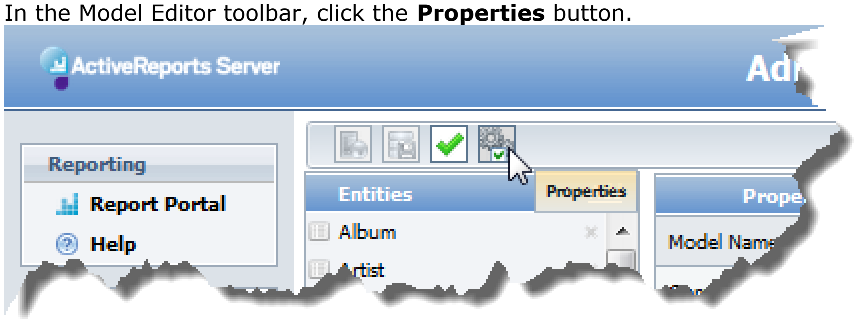
You can modify the connection string of an existing model. This is useful if you want to substitute a UserContext attribute for a value in the connection string when you use LDAP and ActiveDirectory or a custom security provider.

You can also modify the name and command timeout value of the model.

To change the connection string

1. Next to a model in the Models list, click **Edit**. The model editor appears.

2. In the Model Editor toolbar, click the **Properties** button.



3. In the workspace, change properties as necessary. The table below details the properties that you can change.
   Model Properties

Property	Description
Model Name	Changes the name of the model that appears in the Models list. Note that if you change this name, you must update it in any reports based on the model.
Command Timeout	Specifies the wait time before terminating an attempt to execute a command and generating an error. Leave this value blank to use the provider's default timeout.
Connection String	Specifies the connection string used to access data for the model. This value is disabled for the sample model, but with an actual model, you can edit the connection string here. You can put a UserContext attribute name between percent signs in place of any value in the connection string.

Modifying an Entity

You can modify the properties of an existing entity within a model.

> **Caution**: Modifying a model may break reports attached to it. See **Managing Model Breaking Changes** for more information.

To modify an entity

1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity that you want to modify. The properties appear in the workspace and the Attributes and Relations lists populate with any associated attributes and relations.
3. In the workspace, change properties as necessary. The table below details the properties.
   Entity Properties

Property Name	Type	Description
(Id)	ID	A unique identifier for the entity. Used internally by the system.
Name	String	The name of the entity.
Binding	ID	The database object that the entity represents.
CollectionName	String	The name to be used for a collection of instances of the entity. Use this property to ensure that plural nouns are spelled correctly.
Description	String	A description of the entity that appears as tooltip text in the Reporting Portal when a user hovers the pointer over the entity.
Hidden	Boolean	Indicates whether the entity is hidden from end users.
CollapseInRelations	Boolean	Indicates whether to collapse the entity when it is displayed within a relation.
Security Filter	Expression	Allows you to use custom security providers to filter the data displayed for the entity. See **Managing Single Sign-On Security Providers** for more information.
DefaultAggregateAttributes	Object	Determines the attributes to display by default when the entity appears in an aggregate.
DefaultDetailAttributes	Object	Determines the attributes to display by default. For example, when you drag the entity onto the designer surface and create a table or chart, these attributes display by default.
IdentifyingAttributes	Object	The set of attributes to use to identify a unique instance of the entity. This is used instead of DefaultDetailAttributes to build a table or chart if that collection is empty, since this one is never empty. This is also used to create the grouping label when you group an entity.

Adding a New Logical Table (Entity)

You can create and add a new logical table to a model, or base a new one on an existing entity from the list.

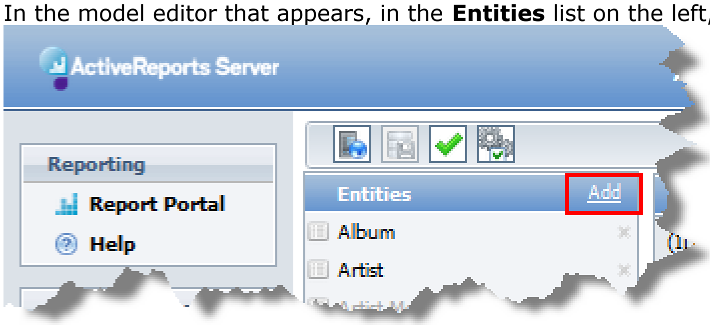
To create a new logical table (entity)

Follow these steps to create a new entity using a SQL query.

Create a new entity from a query

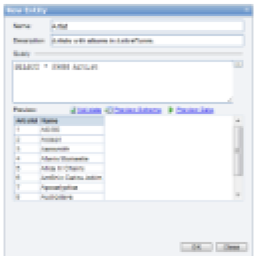
1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, click **Add**. The New Entity dialog appears.



3. In the **New Entity** dialog, enter a **Name** for the new entity. This value appears in the list of entities for the model.
4. In the **Description** box, enter a description that provides users with additional information about the entity, such as when it is appropriate to use it.
5. In the **Query** box, define the entity with a SQL query that pulls the data that you want.

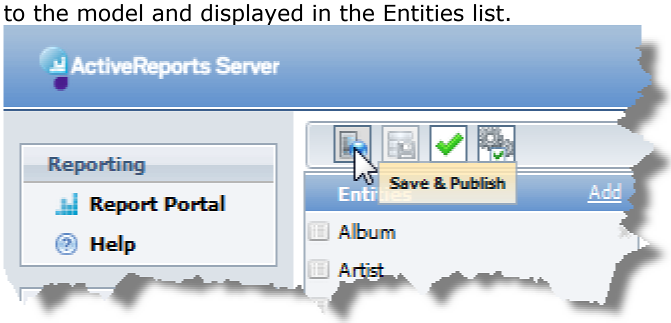
6. Check your query using the commands above the preview box.
   - **Validate** verifies the query and displays any errors it finds in red at the top of the dialog.
   - **Preview Schema** populates a table with information pulled by the query (column name, size, data type, and is nullable).
   - **Preview Data** populates a table with a sampling of the data pulled by the query.



7. Click **OK** to add the new logical table to the model as a new entity.

> **Note**: In the Entities pane, when you select an entity that you created, an **Edit** command appears next to the **Add** command at the top of the pane. Click **Edit** to re-open the New Entity dialog and make changes to the name, description, or query.

8. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical table is added to the model and displayed in the Entities list.

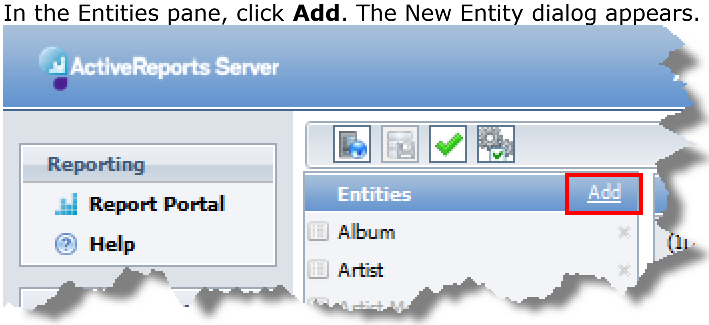


To base a new entity on an existing logical table

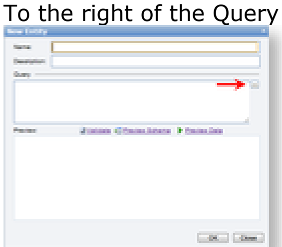
Follow these steps to select an existing logical table to use as a new entity.

Base a new entity on an existing logical table

1. To add an existing entity
2. In the Entities pane, click **Add**. The New Entity dialog appears.



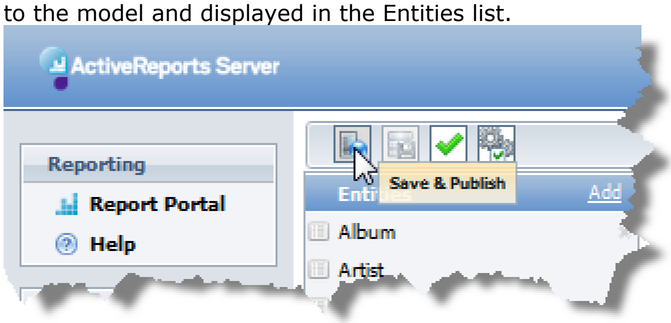
3. To the right of the Query box, click the ellipsis button.



4. In the **Select an Existing Table** dialog that appears, select the table that you want and click **OK**. The Query box

in the New Entity dialog is populated with the table name, for example **_Expenses**, and the Name box is populated with the user friendly table name, for example **Expenses**.

5. You can check your results using the commands above the preview box.
   - **Validate** verifies the query and displays any errors it finds in red at the top of the dialog.
   - **Preview Schema** populates a table with information pulled by the query (column name, size, data type, and is nullable).
   - **Preview Data** populates a table with a sampling of the data pulled by the query.
6. In the **Description** box, enter a description that provides users with additional information about the entity, such as when it is appropriate to use it.
7. Click **OK** to add the new logical table to the model as a new entity.
8. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical table is added to the model and displayed in the Entities list.



Modifying an Attribute

You can modify the properties of an existing attribute within a model.

Caution: Modifying a model may break reports attached to it. See **Managing Model Breaking Changes** for more information.

To modify an attribute

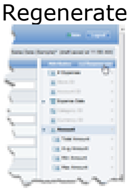
1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity whose attribute you want to modify. The properties appear in the workspace and the Attributes and Relations lists populate with any associated attributes and relations.
3. In the **Attributes** list on the right, select the attribute whose properties you want to modify. The properties appear in the workspace.
4. In the workspace, change properties as necessary. The table below details the properties that you can change.
Attribute properties

Property Name	Type	Description
(Id)	ID	A unique identifier for the attribute. Used internally by the system.
Name	String	The name for the attribute.
Binding	ID	The database object that the attribute represents.
DataType	Enum	The data type for the attribute.
DiscourageGrouping	Boolean	Indicates whether data is grouped by entity instead of by attribute.
Format	String	The default format for the attribute.
Hidden	Boolean	Indicates whether the attribute is hidden from end users.
IsAggregate	Boolean	Indicates whether the attribute is an aggregate.
Nullable	Boolean	Indicates whether the attribute can have a null value and is used in filter and parameter editors to provide special values.

ValueSelection	Enum	Determines what control is used to display attribute values in the filter and parameters editors. • **Dropdown** shows lookup values in a dropdown control. • **List** shows more values at once than Dropdown. • **None** disables the entire lookup process. By default, ActiveReports 8 Server selects this value based on the UAVC (unique attribute values count): • Less than 200 - Dropdown • Between 200 and 1,000 - List • Otherwise - None

5. If the attribute that you are modifying has a list of variations under it such as date parts (e.g. Expense Day, Expense Month) or aggregations (e.g. Total Amount, Avg Amount), that you have modified with format changes or deletions, you can click the **Regenerate** command at the top right of the Attributes list to generate all of the available variations. Note that the attribute for which you want to regenerate variations must have focus for the Regenerate command to appear.



Adding a New Logical Column (Attribute)

You can create and add a new logical column to an entity, or base a new attribute on an existing logical column from the list.

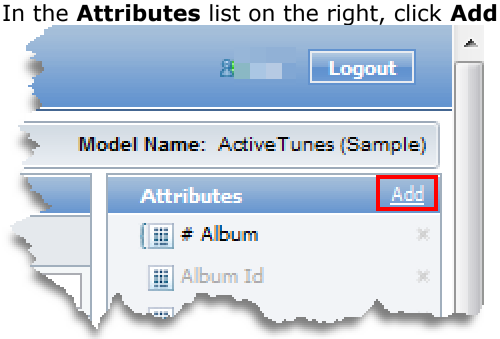
To base a new attribute on a related entity

Follow these steps to select an existing attribute on a related entity.

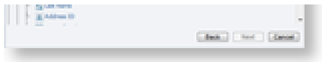
Base a new attribute on an existing logical column

1. Next to a model in the Models list, click **Edit**.

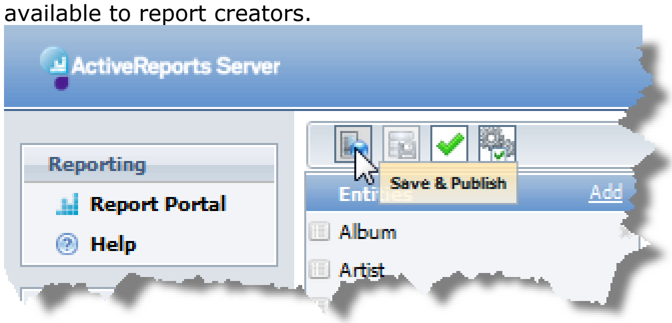
2. In the model editor that appears, in the **Entities** list on the left, select the entity to which you want to add an attribute.
3. In the **Attributes** list on the right, click **Add**. The Add Attribute wizard appears.



4. In the **Add Attribute** wizard, click the **Related Entity** selection.
5. In the list of existing attributes that appears, a white triangle indicates that you can expand the node to reveal related attributes. (A black triangle indicates that a node is expanded.)



6. Select the one that you want to use and click **Next**.
7. In the **Name** and **Description** boxes that appear, enter the name and description for the new attribute.
8. Click **Finish**. The new attribute appears in the Attributes list.
9. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical column is available to report creators.



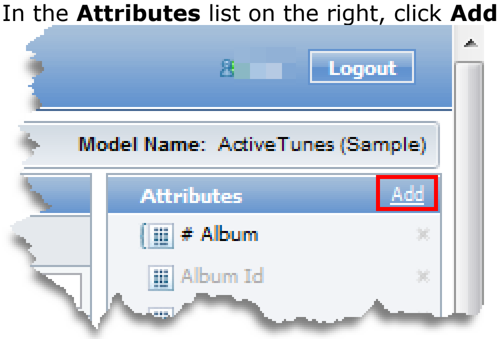
To base a new attribute on a database field

Follow these steps to select a database field to use as an attribute.

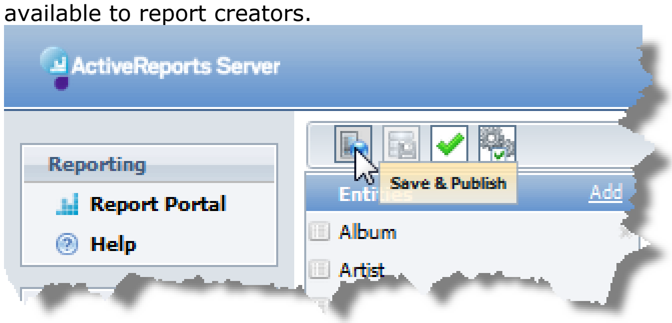
Base a new attribute on a database field

1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity to which you want to add an attribute.
3. In the **Attributes** list on the right, click **Add**. The Add Attribute wizard appears.



4. In the **Add Attribute** wizard, click the **Database Field** selection.
5. In the list of existing columns that appears, select the one that you want to use and click **Next**.
6. In the **Name** and **Description** boxes that appear, enter the name and description for the new attribute.
7. Click **Finish**. The new attribute appears in the Attributes list.
8. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical column is available to report creators.

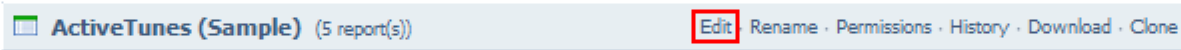


To create a new logical column (attribute)

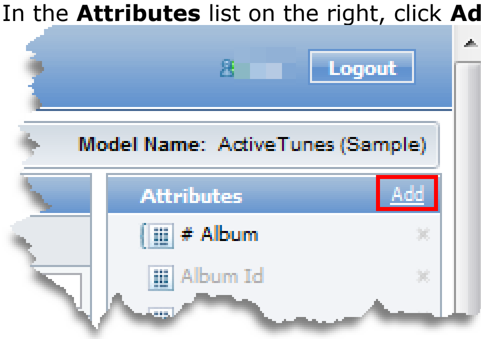
Follow these steps to create a new attribute using a SQL expression definition.

Creating a new attribute from an expression

1. Next to a model in the Models list, click **Edit**.



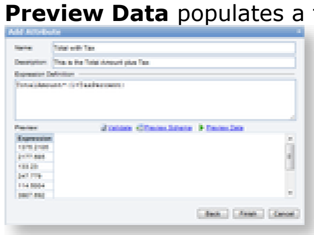
2. In the model editor that appears, in the **Entities** list on the left, select the entity to which you want to add an attribute.
3. In the **Attributes** list on the right, click **Add**. The Add Attribute wizard appears.



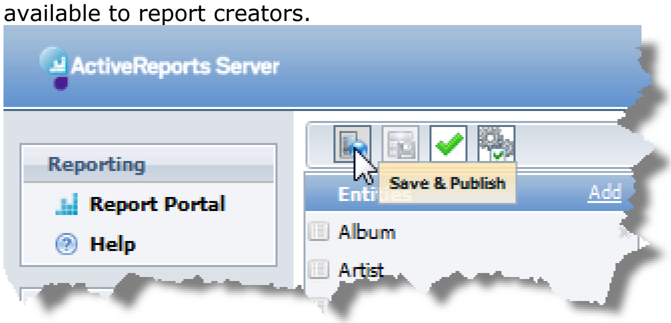
4. In the **Add Attribute** wizard, click the **SQL Expression** selection.
5. In the **Name** box that appears, enter the name for the new attribute. This value appears in the list of entities for the model.
6. In the **Description** box, enter a description that provides users with additional information about the attribute, such as when it is appropriate to use it.
7. In the **Expression Definition** box, define the expression for the logical column with a SQL expression that pulls the data that you want.

> **Note:** The syntax to use for expressions is T-SQL for SQL data sources or PL-SQL for Oracle data sources.

8. Check your query using the commands above the preview box.
   - **Validate** verifies the expression and displays any errors it finds in red at the top of the dialog.
   - **Preview Schema** populates a table with information pulled by the expression (column name, size, data type, and is nullable).
   - **Preview Data** populates a table with a sampling of the data pulled by the expression.



9. Click **Finish**. The new attribute appears in the Attributes list.
10. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical column is available to report creators.



Modifying a Relation

A relation item in a model describes the relationships between entities. You can modify the properties of an existing relation within a model.

Caution: Modifying a model may break reports attached to it. See **Managing Model Breaking Changes** for more information.

To modify a relation

1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity whose relation you want to modify. The properties appear in the workspace and the Attributes and Relations lists populate with any associated attributes and relations.
3. In the **Relations** list on the lower right, select the relation whose properties you want to modify. The properties appear in the workspace.
4. In the workspace, change properties as necessary. The table below gives details about the properties.
Relation properties

Property Name	Type	Description
(Id)	ID	A unique identifier for the relation. Used internally by the system.
Name	String	The name for the relation.
Binding	ID	The database object that the relation represents.
Cardinality	Enum	Defines the rules and restrictions for the relation's behavior in the entity tree and in semantic query building. Possible values are One, Many, OptionalOne, or OptionalMany. This property determines whether the Recursive property has an effect on the relation.
Description	String	A description of the relation that appears as tooltip text in the Reporting Portal when a user hovers the pointer over the relation.
Hidden	Boolean	Indicates whether the entity reached by the relation is hidden from end users.
Recursive	Boolean	Indicates whether the relation can be added more than once to the same path. This applies only to relations with a Cardinality of One or OptionalOne. For relations with a Cardinality of Many or OptionalMany, this property is ignored.
Include self	Boolean	Indicates whether to allow the relation to be accessible to itself in the same recurring path. This applies only in the second level of an entity.
HiddenFields	Object	Defines the set of entity fields that is hidden from end users when they reach the entity through the relation.
RelatedRelation	Object	Indicates the backward relation (the relation for the related entity) of the relation. You cannot edit this value, but you can use the navigate link to go to the related relation.
RelatedEntity	Object	The entity to which the relation is bound. You cannot edit this value, but you can use the navigate link to go to the related entity.

Adding a New Logical Relation

You can create and add a new logical relation to an entity, or base a new relation on an existing logical relation from the list.

To create a new relation

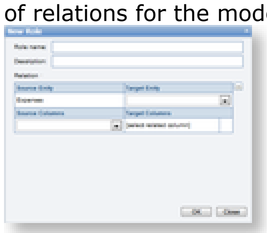
Follow these steps to create a new logical relation by specifying a target entity, and source and target attributes.

Creating a new logical relation

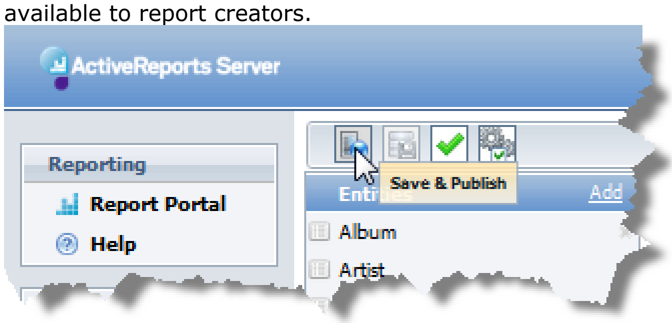
1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity to which you want to add a relation.
3. In the **Relations** list on the lower right, click **Add**. The New Relation dialog appears.

4. In the **New Relation** dialog, in the **Name** box, enter the name for the new relation. This value appears in the list of relations for the model.



5. In the **Description** box, enter a description that provides users with additional information about the attribute, such as when it is appropriate to use it.
6. In the **Relation** section, the **Source Entity** is already set (based on the entity you selected to which to add a relation). Drop down the **Target Entity** list and select a target for the relation. This enables the Target Columns drop-down list below.
7. Drop down the **Source Columns** list and select a source attribute for the relation. This adds a new pair of drop-down lists for additional sources and targets below.
8. Drop down the **Target Columns** list and select a target attribute for the relation.
9. Click **OK**. The new relation appears in the Relations list.
10. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical relation is available to report creators.



To base a new logical relation on an existing relation

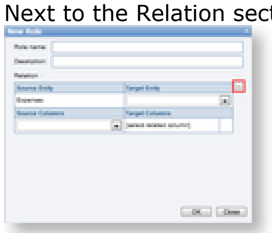
Follow these steps to select a relation to use as a new logical relation.

Base a new attribute on an existing logical column

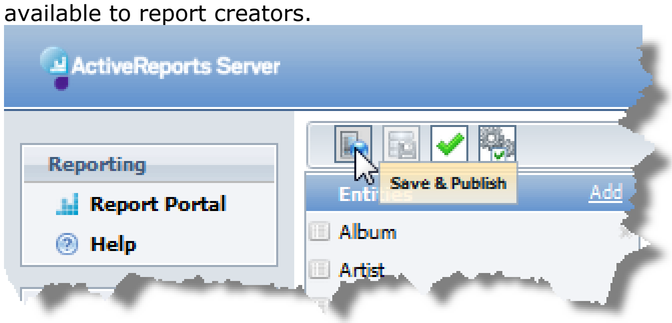
1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity to which you want to add a relation.
3. In the **Relations** list on the lower right, click **Add**. The New Relation dialog appears.

4. In the **New Relation** dialog, in the **Relation name** box, enter the name for the new relation. This value appears in the list of relations for the model.
5. In the **Description** box, enter a description that provides users with additional information about the attribute, such as when it is appropriate to use it.
6. Next to the Relation section, click the ellipsis button.



7. In the Select an Existing Relation dialog that appears, select the one that you want to use and click **OK**.
8. In the New Relation dialog, click **OK**. The new relation appears in the Relations list.
9. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The new logical relation is available to report creators.



Deleting an Item

You can delete any model item, whether entity, attribute or relation, by clicking the X button to the right each item in the Model Editor.

Caution: Modifying a model may break reports attached to it. See **Managing Model Breaking Changes** for more information.

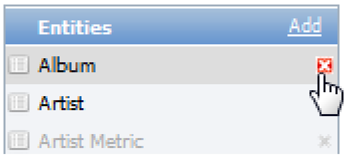
To delete an entity

Follow these steps to delete an entity from a model.

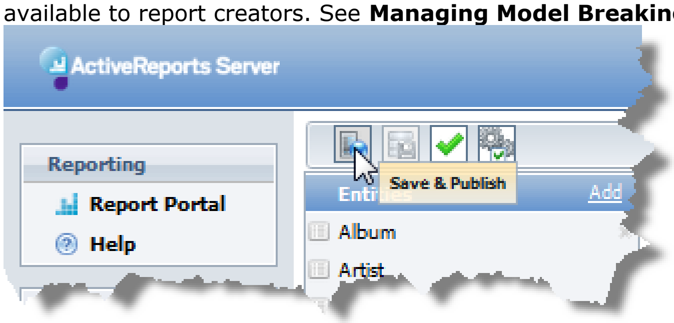
Delete an entity

1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, to the right of the entity that you want to delete, click the X button.



3. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The deleted entity is no longer available to report creators. See **Managing Model Breaking Changes** for more information.



Note: It is recommended that you hide an entity or an attribute of the model instead of deleting it. This way, the entity or attribute is not available for use in new reports, but existing reports are not broken.

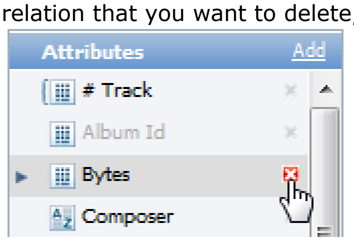
To delete an attribute or relation

Follow these steps to delete an attribute or relation from an entity.

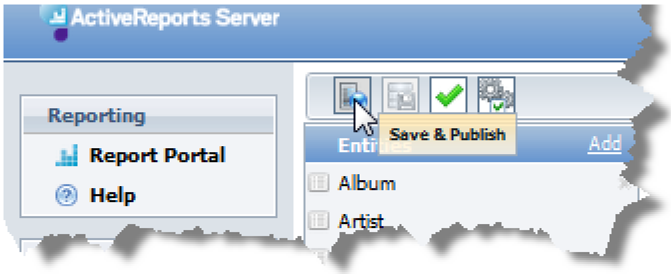
Delete an attribute or relation

1. Next to a model in the Models list, click **Edit**.

2. In the model editor that appears, in the **Entities** list on the left, select the entity that contains the attribute or relation that you want to delete. The Attributes and Relations lists populate with any associated attributes and relations.
3. In the **Attributes** list on the right, or the **Relations** list on the lower right, to the right of the attribute or relation that you want to delete, click the X button.



4. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. The deleted attribute or relation is no longer available to report creators. See **Managing Model Breaking Changes** for more information.



Saving a Modified Model

When you modify a model in the Model Editor, you can save the modifications and make them available to users right away, or if you have many changes to make, or a specific roll-out date, you can save a draft until you are ready to make your changes public.

Caution: Modifying a model may break reports attached to it. See **Managing Model Breaking Changes** for more information.

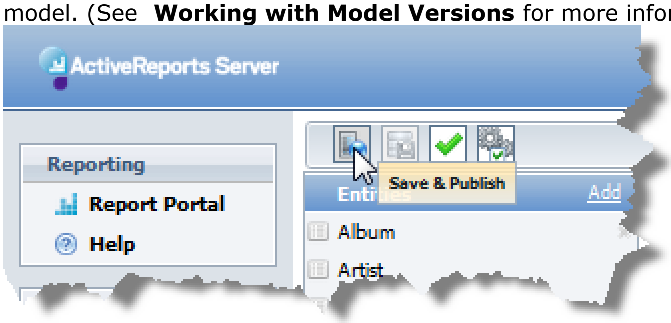
To save changes and make them public

Follow these steps to publish your changes so that report creators can use the updated model.

Publishing changes

1. Next to a model in the Models list, click **Edit**.

2. Modify, add, or delete any entities, attributes, or relations that you need to modify. (See **Model Editor Overview** for topics relating to these actions.)
3. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. This creates a new version of the model. The old version, as well as all other previous model versions, is kept in the **History** section of the model. (See **Working with Model Versions** for more information.)



4. All reports associated with the model automatically bind to the latest model version. Model changes, especially deletions, may break some reports, so compatibility is checked automatically. (See **Managing Model Breaking Changes** for more information.)

To save a draft of your changes

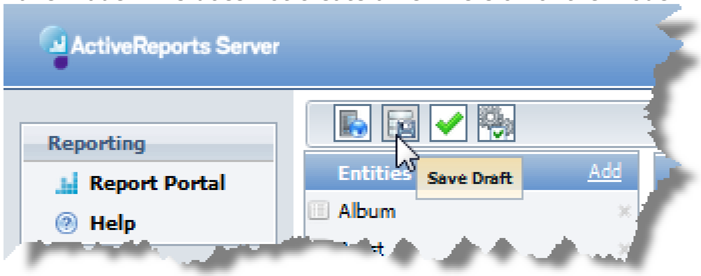
Follow these steps to save a working draft that you can return to later, without releasing the changes to report creators. ActiveReports 8 Server automatically saves a draft while you are working, but you can save the draft any time.

Saving a working draft

1. Next to a model in the Models list, click **Edit**.

2. Modify, add, or delete any entities, attributes, or relations that you need to modify. (See **Model Editor Overview** for topics relating to these actions.)
3. In the Model Editor toolbar, click the **Save Draft** button to save a draft copy of the model with the changes you

have made. This does not create a new version of the model. The old version is still what report creators use.



4. The draft copy is available only to the user who made the modifications, and is marked in the Models list with (Draft Saved).

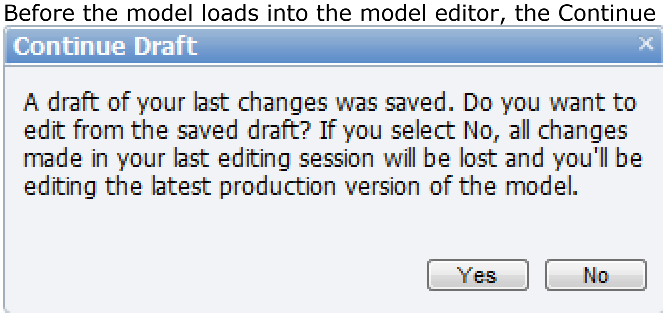
To discard a draft

ActiveReports 8 Server automatically saves a draft while you are working, and you can save it as you make changes, but if you do not want to keep the changes, you can discard them.

Discarding a draft

1. In the Model list, click **Edit (Draft Saved)**.

2. Before the model loads into the model editor, the Continue Draft message box appears.



3. Click **No**. The draft changes are discarded and the unchanged version of the model loads into the model editor.

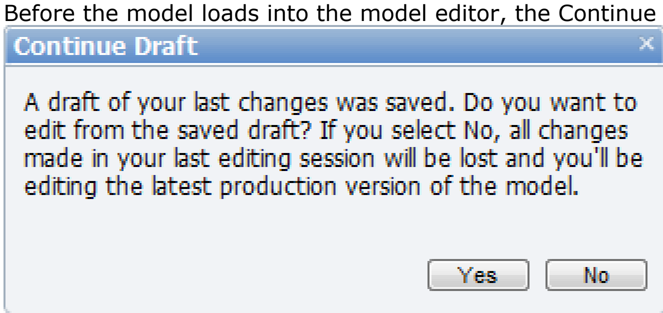
To apply the modifications of the draft copy to the model

Follow these steps to keep changes saved to a working draft, and publish the changes for report creators.

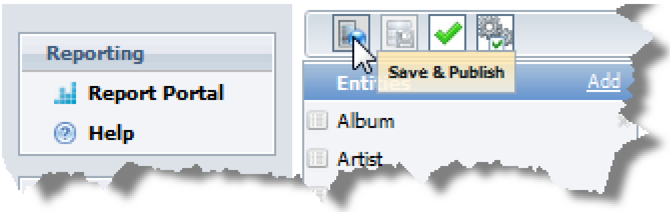
Publishing a draft

1. In the Model list, click **Edit (Draft Saved)**.

2. Before the model loads into the model editor, the Continue Draft message box appears.



3. Click **Yes**. The changed draft version of the model loads into the model editor.
4. In the Model Editor toolbar, click the **Save & Publish** button to save the changes. This creates a new version of the model. The old version, as well as all other previous model versions, is kept in the **History** section of the model. (See **Working with Model Versions** for more information.)



5. All reports associated with the model automatically bind to the latest model version. Model changes, especially deletions, may break some reports, so compatibility is checked automatically. (See **Managing Model Breaking Changes** for more information.)

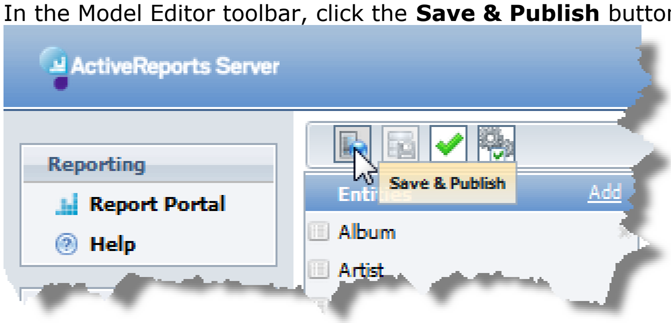
Entering Modification Comments

ActiveReports 8 Server automatically adds a comment with the name of the administrator who published a change, or you can enter your own comments after you modify and save a model. These comments appear in the history of model versions so that you can easily find the version you want. See **Working with Model Versions** for more information.

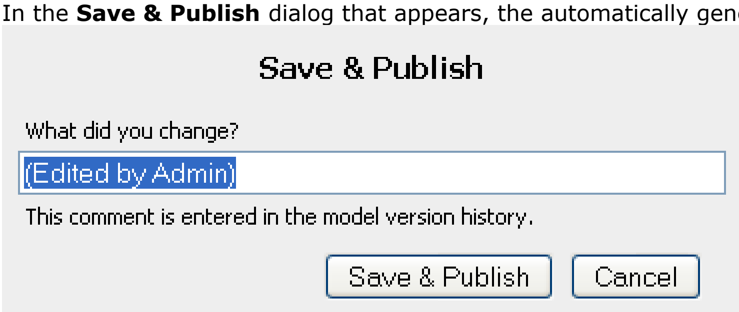
To enter modification comments

1. Next to a model in the Models list, click **Edit**.

2. Modify, add, or delete any entities, attributes, or relations that you need to modify. (See **Model Editor Overview** for topics relating to these actions.)
3. In the Model Editor toolbar, click the **Save & Publish** button to save the changes.



4. In the **Save & Publish** dialog that appears, the automatically generated comments are highlighted.



5. Enter your comments and click **Save & Publish**. This creates a new version of the model. The old version, as well as all other previous model versions, is kept in the **History** section of the model.

> **Note:** If you do not wish to enter your own comments, click **Save & Publish** anyway to save the automatically generated comments.

6. The comments are displayed in the model history. (See **Working with Model Versions** for more information.)

Managing Model Breaking Changes

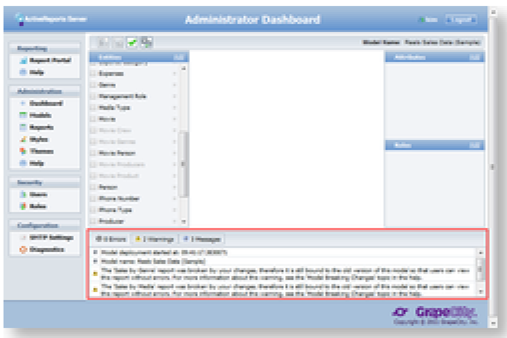
When you modify a model, certain changes may break reports that are associated with the model. Such model changes are:

1. **Deleting** an entity, attribute or relation.
2. Modifying the **Cardinality** property of a model's relation.

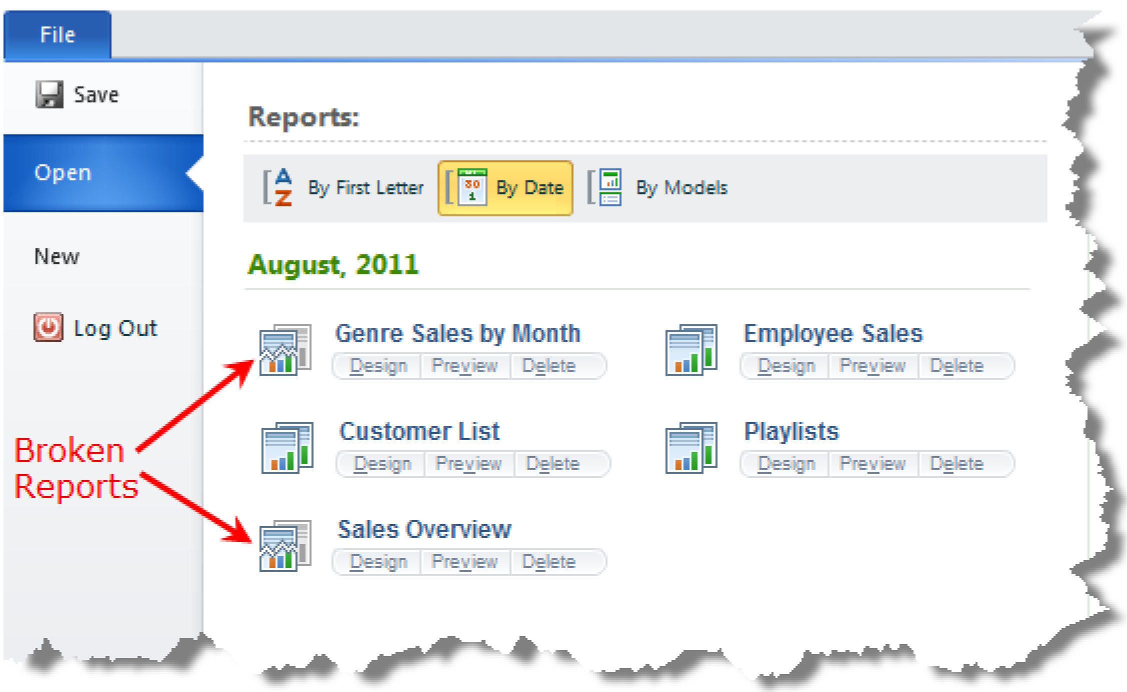
> **Note:** It is recommended that you hide an entity or an attribute of the model instead of deleting it. This way, the entity or attribute is not available for use in new reports, but existing reports are not broken.

When you save a change that breaks a report, a warning appears below the work area for each broken report.

> *The "ReportName" report was broken by your changes, therefore it is still bound to the old version of this model so that users can view this report without errors. For more information about this warning, see the "Model Breaking Changes" topic in the help.*



Any broken reports associated with the model are displayed under **Reports** in ActiveReports 8 Server Reporting Portal with a special icon, indicating that the report is not compatible with the latest version of its bound model.



Reports that are incompatible with the current version of the model open in the Preview using the old version of the model.

If you open the report in the Reporting Portal, you are required to fix any errors caused by the model breaking change so that the report will work with the new version of the model. To continue working with a broken report, perform one of the following actions:

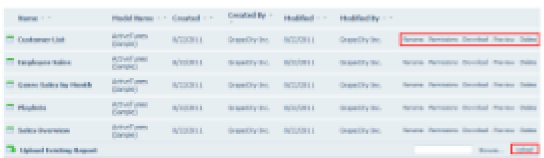
1. Go to ActiveReports 8 Server Reporting Portal and fix the report.
2. **Revert** the model to a previous version in ActiveReports 8 Server Administrator.

Managing Reports

End users create reports based on a data model in the ActiveReports 8 Server Reporting Portal. Reports saved in the Reporting Portal are displayed on the Administrator Dashboard in the **Reports** list. Administrators can rename, set permissions, download, preview, delete, or upload them.

Report Commands

You can perform actions on reports using the commands to the right of each report in the list.



Command Descriptions

Command	Description
Rename	Allows you to change the report's name. Click this command to make the Name property editable. The commands are replaced with Upload and Cancel commands for the report. Click Upload to save the new name, or Cancel to discard any changes.
Permissions	Opens the **Change Permissions** dialog where you can select which roles have access to the model. See **Managing Permissions to Reports and Models** for more information.
Download	Downloads and saves the report to the local file system in .rdlx format.
Preview	Runs the report and displays it in the Designer Preview window.
Delete	Deletes the report and removes it from the list of reports of the corresponding model.
Upload	Uploads a report from your local files. First click Browse to locate a valid report file in one of the supported formats: .rdl, .rdlx, .rpx, .dll or .exe. (Reports in other formats are not uploaded.) Select a report, click Open, and then click the Upload command. The report is added to the Reports list. Using Developer ActiveReports **Notes**: An uploaded report is automatically bound to a compatible model. If there is no compatible version of the model that the report is bound to on the server, the report is not uploaded. The user who uploads the report is the Owner of the report, with full, irrevocable permission.

Special Reports

Some types of reports require special consideration. These include drill-through reports, and reports created with the developer version of ActiveReports or Data Dynamics Reports. Find out how to handle these reports in the following topics.

- **Using Code Based Section Reports from ActiveReports Developer**
- **Using RPX Reports from ActiveReports Developer**
- **Using Data Dynamics Reports**

- **Uploading Drill-Through Reports**

Using Code Based Section Reports

If your developers use ActiveReports, you can upload your existing ActiveReports code-based files that are compiled into a .NET assembly to take advantage of the ActiveReports 8 Server scheduling, security, and scalability features. Note that if you also use the Designer Addon, these developer reports cannot be edited by end users.

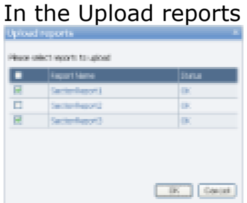
.NET Assembly Requirements

In order to upload your code-based section reports to ActiveReports 8 Server, first compile them into a .NET assembly. Your compiled report assembly must meet all of the following requirements in order to avoid errors.

- It must be a valid .NET assembly with a strong name signature.
- It must have the target platform set to Any CPU.
- It must be compiled for .NET Framework 3.5.
- All its dependencies must be resolvable on the Server side.
- The version of ActiveReports 8 assemblies that you reference in the compiled report assembly cannot be newer than the version of ActiveReports 8 assemblies installed on the Server.
- It must contain at least one code-based section report.
- All its reports must have a parameterless constructor.

To upload developer ActiveReports code-based files

Developer ActiveReports appear in the list of available reports in the Reporting Portal as well as in the Reports list in the Administrator Dashboard. In the Open page of the Reporting Portal, they have only Preview and Delete buttons, with the Design button omitted.

1. From the Administrator Dashboard, select the **Reports** page.
2. Below the list of reports, next to **Upload Existing Report**, click the **Browse** button.
3. In the Open dialog that appears, navigate to the .NET assembly file (dll or exe) with the code-based section reports that you want to distribute via ActiveReports 8 Server, select it, and click **Open**.
4. Next to the **Browse** button, click **Upload**.
5. In the Upload reports dialog that appears, select reports in the list by selecting the corresponding check boxes.

6. Click **OK** to add the reports to the Reports list in both the Administrator Dashboard and the Reporting Portal.

> **Note**: Developer ActiveReports cannot be edited in the ActiveReports 8 Server designer. To edit such reports, please use ActiveReports in Visual Studio or in an End User Designer application created with the Pro Edition of ActiveReports.

To schedule developer ActiveReports

As with any report, you can schedule code-based section reports to run at certain times, and have them automatically sent to stakeholders. For details, please see the Scheduling Reports and Schedule Options topics in the User Guide.

Using RPX Reports from ActiveReports

You can upload your existing ActiveReports RPX files to take advantage of the ActiveReports 8 Server scheduling, security, and scalability features. Of necessity, there are some limits to what is supported. Here are some tips to make your RPX files compliant with ActiveReports 8 Server.

Only self-contained RPX files are supported

- Convert any Visual Basic or C# code used in the original ActiveReport into script in the RPX file.
- Convert any references to external classes, Web services, or application code to a self-contained script inside the RPX file.
- Remove any function that requires access to the local file system, or the .NET security policies used to ensure server stability will cause errors.
- Connect reports to databases such as SQL Server or Oracle normally, using only a connection string.
- Connect to XML data sources on a remote host using the FileURL property to enable unbound data source scenarios.

To upload developer ActiveReports RPX files

Developer ActiveReports appear in the list of available reports in the Reporting Portal as well as in the Reports list in the Administrator Dashboard. In the Open page of the Reporting Portal, they have only Preview and Delete buttons, with the Design button omitted.

1. From the Administrator Dashboard, select the **Reports** page.
2. Below the list of reports, next to **Upload Existing Report**, click the **Browse** button.
3. In the Open dialog that appears, navigate to the RPX file that you want to distribute via ActiveReports 8 Server, select it, and click **Open**.
4. The dialog closes and a shortened path to the file appears in the Browse box. Click **Upload** to add the report to the Reports list in both the Administrator Dashboard and the Reporting Portal.

Note: Developer ActiveReports cannot be edited in the ActiveReports 8 Server designer. To edit such reports, please use ActiveReports in Visual Studio or in an End User Designer application created with the Pro Edition of ActiveReports.

To schedule developer ActiveReports

As with any report, you can schedule RPX reports to run at certain times, and have them automatically sent to stakeholders. For details, please see the Scheduling Reports and Schedule Options topics in the User Guide.

Using Data Dynamics Reports

You can upload your existing Data Dynamics Reports RDLX files to take advantage of the ActiveReports 8 Server scheduling, security, and scalability features.

Caution: In order to run RDLX reports in ActiveReports 8 Server, the server must have access to the data source. Shared data sources are not supported for Data Dynamics Reports in ActiveReports 8 Server.

To upload Data Dynamics Reports RDLX files

Data Dynamics Reports appear in the list of available reports in the Reporting Portal as well as in the Reports list in the Administrator Dashboard. In the Open page of the Reporting Portal, they have only Preview and Delete buttons, with the Design button omitted.

1. From the Administrator Dashboard, select the **Reports** page.
2. Below the list of reports, next to **Upload Existing Report**, click the **Browse** button.
3. In the Open dialog that appears, navigate to the RDLX file that you want to distribute via ActiveReports 8 Server, select it, and click **Open**.
4. The dialog closes and a shortened path to the file appears in the Browse box. Click **Upload** to add the report to the Reports list in both the Administrator Dashboard and the Reporting Portal.

Note: Data Dynamics Reports cannot be edited in the ActiveReports 8 Server designer. To edit such reports, please use Data Dynamics Reports in Visual Studio or in an End User Designer application created with Data Dynamics Reports.

To schedule Data Dynamics Reports

As with any report, you can schedule RDLX reports to run at certain times, and have them automatically sent to stakeholders. For details, please see the Scheduling Reports and Schedule Options topics in the User Guide.

Uploading Drill-Through Reports

You can download and upload drill-through reports like any other report created with ActiveReports 8 Server, but to keep the drill-through functionality intact, you must keep two things in mind.

- The linked report must always be kept with the main report.
- If you change the linked report's name, you must also change it in the main report.

When you upload a report with drill-through links, if the linked reports do not exist on the server, an error message appears.

Managing Schedules

Shared schedules allow you to specify a recurring interval at which to run a report. Each schedule has a name, an optional description, and a list of roles that defines which users have permission to assign reports to it.

Once you add a shared schedule, all users in each role that is granted permission to the schedule can assign reports to run on it.

Recurrence

You can set up recurrence options on shared schedules in the same way that users set recurrence options on individual schedules for reports.

Recurrence options

Option	Description
Starting On	Set the date and time for the task to start.
Repeat Task Every	Enter a number and select **Minutes**, **Hours**, **Days**, or **Weeks** to set the frequency of the report execution.
On days	Select check boxes to specify which day or days of the week the task runs, or leave all blank to run the task daily.
Result Expires After	Enter a number and select **Minutes**, **Hours**, or **Days** to specify how long to keep the report in the History. When the specified time period expires, you can no longer open the report from the **History** tab by clicking **Open**.

Note: The **Repeat Task Every** and **On days** settings work in conjunction to define the dates for a scheduled task to run.

For example, if you set the **Repeat Task Every** setting to *2 Days* and the **On days** setting to *Fri*, the scheduled task runs every second day that falls on a Friday.

Another example: if you set the **Starting On** setting to *Thursday*, the **Repeat Task Every** setting to *2 Weeks*, and the **On days** setting to *Thu*, the scheduled task runs every second Thursday. However, if a day other than Thursday is selected in the **On days** setting, the scheduled task will NOT run, as the days for the schedule to run include the starting date.

Delivery

You can optionally set delivery options for your users, or you can leave this section blank and allow your users to choose these settings.

Option	Description
Format	Select the format in which to deliver the report. **Formats**: Allow User to Choose, Excel file (xls), Mht document (archived Web page), Image file, PDF document, Word document, or XML file (xml).
Delivery Type	Select **Allow User to Choose**, **Email**, or **Windows File Share**. Below are more options that appear with these delivery types.

Delivery Type options

Note: ExecutionTime token is specified as **&{ExecutionTime:<TokenFormat>}**. Different **DateTime** format strings for .NET framework such as "d", "F", "T" etc. can be specified for the TokenFormat. For example, **&{ExecutionTime:d}** will be displayed as **06/10/2014**.

Caution: When **"g"** DateTime format is specified in the "&ExecutionTime" token, the generated attachment that is received through scheduled e-mail is downloaded in an unsupported format.

Option	Description
Email report to	Enter one or more e-mail addresses where the report is to be sent. Separate multiple e-mail

	addresses with commas.
Subject	Enter the text to use for the e-mail's subject line. You can use the following syntax in the subject: • ?*ParameterName:* To specify parameter values for the report in the subject. *R*eplace *ParameterName* with the actual name of the parameter. • &ReportName: To specify the report name. • &ExecutionTime: To specify the execution date and time of the report.
Body	Enter text to use in the body of the e-mail. You can use the following syntax in the e-mail body: • ?*ParameterName:* To specify parameter values for the report in the e-mail body. *R*eplace *ParameterName* with the actual name of the parameter. • &ReportName: To specify the report name. • &ExecutionTime: To specify the execution date and time of the report.
Delivery method	Select **Allow User to Choose**, or **As Link** to insert a hyperlink to the report in the body of the message, or **As Attachment** to attach the report to the message in the format specified above. If you select As Link, please see **Site Settings Notification URL**.
Attachment template	Enter the resource name template. You can use the following syntax in the attachment template: • ?*ParameterName:* To specify parameter values for the report in the attachment template. *R*eplace *ParameterName* with the actual name of the parameter. • &ReportName: To specify the report name. This is default template. • &ExecutionTime: To specify the execution date and time of the report.

Windows File Share Delivery Type Options

Option	Description
File name	Enter the name of the report file. You can can use the following styntax in the file name: • ?*ParameterName:* To specify parameter values for the report in the file name. *R*eplace *ParameterName* with the actual name of the parameter. • &ReportName: To specify the report name. This is default template. • &ExecutionTime: To specify the execution date and time time of the report.
Add file extension	Select to add the file extension for the report's format to the file name.
Path	Enter the fully qualified UNC path of the location at which to share the report.
User name	Enter the user name to connect to the UNC path.
Password	Enter the password of the user for the UNC path.
Overwrite	Select **Allow User to Choose**, **Overwrite the existing file if it exists** , or **Increment filename as newer versions are added** to control how stored versions of the report are handled.

Managing Themes and Styles

Themes and styles allow you to control the appearance of reports. Themes determine the overall look of reports and all of their tables and charts, whereas styles are applied to individual report items.

Working with Themes
Learn to delete themes and styles from the server, rename them, and set your default theme.

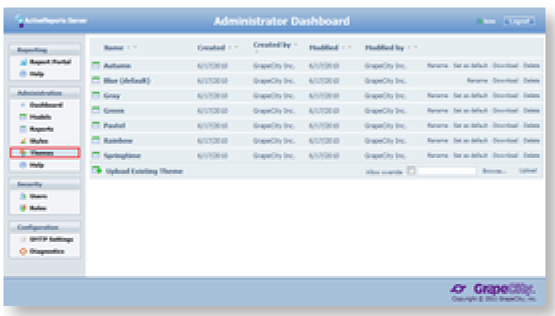
Working with Styles
Learn to download themes and styles to your local machine and upload them to the server.

Working with Themes

The upload and download operations offer an additional means of backing up or transferring themes between servers without requiring access to the server console. Themes are specially formatted xml files with a file extension of .rdlx-theme.

> **Tip**: If you also use Data Dynamics Reports, .rdlx-theme files automatically open in that product's Theme Editor dialog, giving you a way to visually edit your theme files.

You can delete and rename themes, and change the default theme to control the look and feel of all of the reports your users create.



To download a theme

Save a working copy of a theme to your local machine

1. From the Administrator Dashboard, select **Themes**.
2. On the **Themes** page, next to the theme that you want to save, click **Download**.
3. The xml theme file is copied onto your machine in .rdlx-theme format.

   > **Tip**: If your browser does not automatically offer you a choice of where to save the file, you can right-click **Download** and select **Save As**.

To upload a theme

Save a file from your local machine onto the server

1. From the Administrator Dashboard, select **Themes**.
2. On the **Themes** page, next to **Upload Existing Theme**, click **Browse**.
3. In the Open dialog that appears, navigate to the file with the extension **.rdlx-theme** to upload and click **Open**. The dialog closes and the file name appears in the box.
4. If you want to replace an existing theme with your locally modified version of the same name, select the **Allow override** check box.
5. Click **Upload** to add the theme from your computer to the server.

To delete a theme

You can delete any themes that you do not want your users to access when they design reports.

Remove a theme from the list

1. From the Administrator Dashboard, select **Themes**.
2. On the **Themes** page, next to the theme that you want to delete, click the **Delete** command. The theme is removed from the list.

To rename a theme

You can name themes after departments to let them know which one to use, or use any name that suits your business needs.

Make a theme name relevant for your users

1. From the Administrator Dashboard, select **Themes**.
2. On the **Themes** page, next to the theme that you want to rename, click **Rename**. The Name box for the theme is enabled and becomes editable.
3. In the **Name** box, enter the name to use for the theme and click the **Update** command.

To change your default theme

You can select which theme is used by default when your users create new reports. Users can still select any other theme that you have in the themes list.

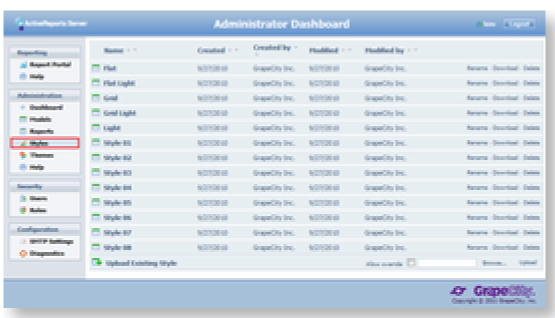
Designate the most commonly used theme for your company

1. From the Administrator Dashboard, select **Themes**.
2. On the **Themes** page, next to the theme that you want to set as your default, click **Set as default**. The Set as default command is removed from the selected theme, and added back to the theme that was previously the default theme.

Working with Styles

The upload and download operations offer an additional means of backing up or transferring styles between servers without requiring access to the server console. Styles are specially formatted xml files with a file extension of .style.

You can delete and rename styles to control the look of text in the reports your users create, as well as to suggest where they are used for a more consistent company-wide look.



To download a style

Save a working copy of a style to your local machine

1. From the Administrator Dashboard, select **Styles**.
2. On the **Styles** page, next to the stylee that you want to save, click **Download**.
3. The xml style file is copied onto your machine in .style format.

> **Tip**: If your browser does not automatically offer you a choice of where to save the file, you can right-click **Download** and select **Save As**.

To upload a style

Save a file from your local machine onto the server

1. From the Administrator Dashboard, select **Styles**.
2. On the **Styles** page, next to **Upload Existing Style**, click **Browse**.

3. In the Open dialog that appears, navigate to the file with the extension **.style** to upload and click **Open**. The dialog closes and the file name appears in the box.
4. If you want to replace an existing style with your locally modified version of the same name, select the **Allow override** check box.
5. Click **Upload** to add the style from your computer to the server.

To delete a style

You can delete any styles that you do not want your users to access when they design reports.

Remove a style from the list

1. From the Administrator Dashboard, select **Styles**.
2. On the **Styles** page, next to the style that you want to delete, click the **Delete** command. The style is removed from the list.

To rename a style

You can name styles to reflect where and how they are used in your company to keep your users consistent.

Make a style name relevant for your users

1. From the Administrator Dashboard, select **Styles**.
2. On the **Styles** page, next to the style that you want to rename, click **Rename**. The Name box for the style is enabled and becomes editable.
3. In the **Name** box, enter the name to use for the style and click the **Update** command.

Managing Security

This section contains information that helps administrators manage users, roles, and permissions.

Managing Users
Learn to add and delete users, and to modify user credentials, password, and roles.

Managing Roles
Learn to add and delete roles, and to modify permissions for roles.

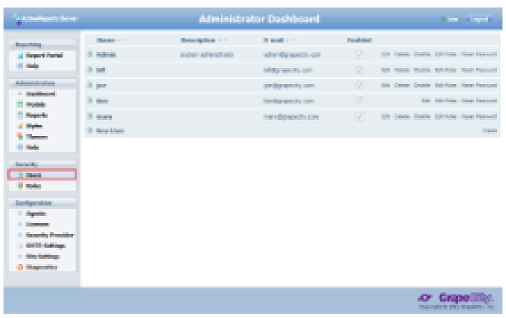
Managing Permissions: Reports, Models, Schedules
Learn what permissions you can set on reports and models.

Using LDAP and Active Directory for Single Sign-On
Learn about how you can integrate your LDAP and Active Directory users into ActiveReports 8 Server security.

> **Note**: You can also create row-level security by creating custom security providers. See the SDK documentation and the **Managing Single Sign-On Security Providers** topic for more information.

Managing Users

An administrator can add and delete users, and modify existing users' credentials, passwords, and roles. All of these functions are in the Users list, which you can access from the Administrator Dashboard by clicking Users in the Security section.

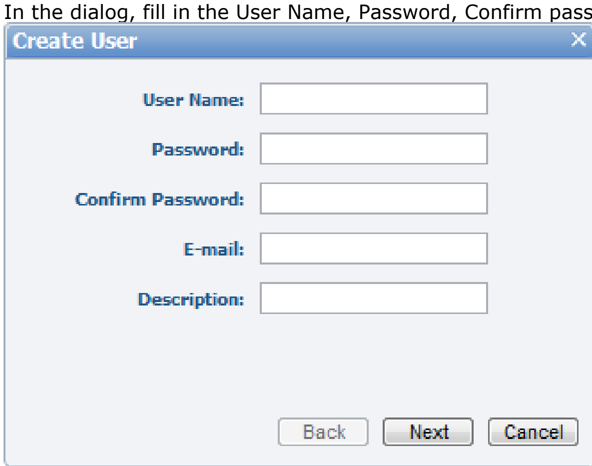


To add a new user

Each user needs an e-mail address, password, and user name.

Create a new user

1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to New User, click **Create**. The Create User dialog appears.
3. In the dialog, fill in the User Name, Password, Confirm password, E-mail, and Description fields.



4. Click **Next**. The Select roles for new user dialog appears.
5. Select the check box under any roles to which to assign the user.
6. Click **Finish**. The new user is added to the Users list and any selected roles, and the user can log in to the Reporting Portal with the user name and password specified.

To delete a user

You can remove users from ActiveReports 8 Server so that they no longer have access.

Delete a user

1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to the user that you want to remove, click **Delete**. The user is removed from the list.

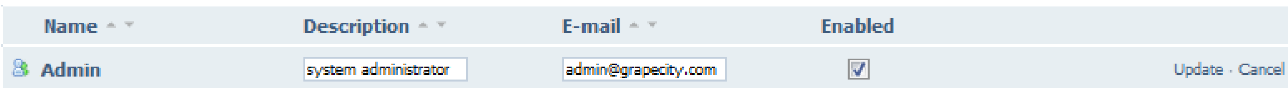
> **Note:** The currently signed-in user Admin cannot be deleted, so there is no Delete command for this user.

To modify credentials of an existing user

In addition to changing a user's e-mail address or description, you can lock or unlock user accounts.

Change user credentials

1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to the user whose credentials you want to change, click **Edit**. The editable fields are enabled and the commands to the right are replaced with Update and Cancel commands.
3. Modify the information in the activated fields of the **Users** table.



The **Description** allows you to enter a description of the user.
The **E-mail** address of the user.
The **Enabled** check box is cleared automatically if the user has made several unsuccessful log-in attempts.

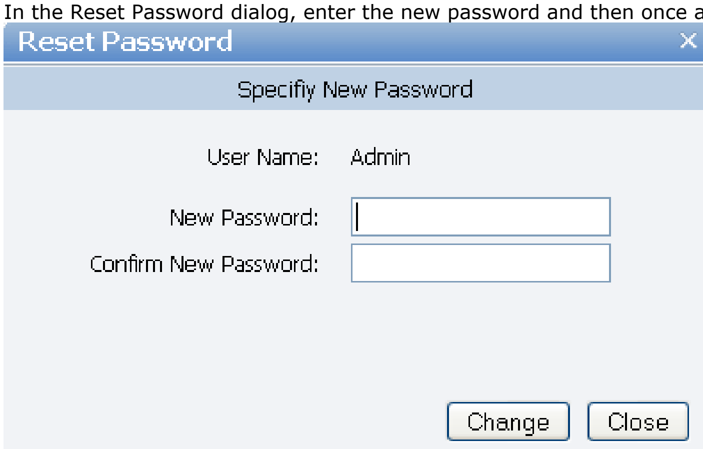
4. Click the **Update** command to apply the changes to the user, or to discard the changes, click the **Cancel** command.

To change a user's password

You can change a user's password when they forget it, or if their password security is in doubt.

Reset a password

1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to the user whose password you want to change, click **Reset Password**. The Reset Password dialog appears.
3. In the Reset Password dialog, enter the new password and then once again to confirm it.



4. Click the **Change** button to confirm the password change.

To manage a user's roles

In addition to editing the user's credentials, you can add or remove the user from roles.

Edit roles

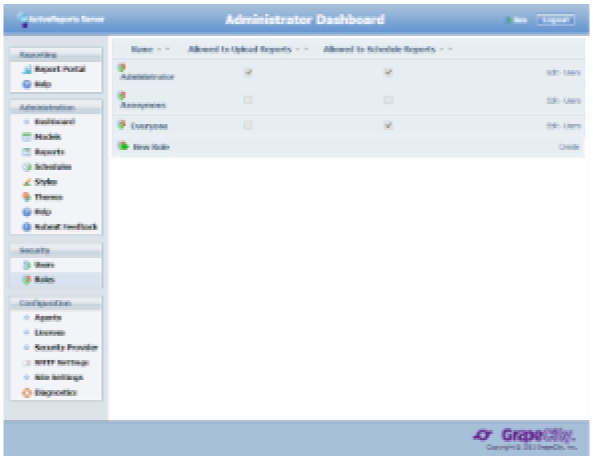
> **Note:** ActiveReports 8 Server has role-based security, which means that permissions are assigned to roles, not users.

1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to the user whose roles you want to change, click **Edit Roles**. The Manage User dialog appears.
3. In the Manage User dialog, select or clear the roles to assign to the user.
4. Click the **Apply** button to apply the changes to the user's profile.

Managing Roles

All permissions are granted to a Role. To grant permission to users, add the user to one or more roles. By default, four roles are configured. These roles cannot be deleted, but you can add new roles to the list.

- **Administrators** contains users with the permission to enter the Administration area.
- **Anonymous** contains no users, but it allows users to work with the Reporting Portal without credentials.
- **Everyone** contains all users.
- **Owner** contains the user who originally created or uploaded the report. By default, this is the only role with full permissions on a newly created or uploaded report.

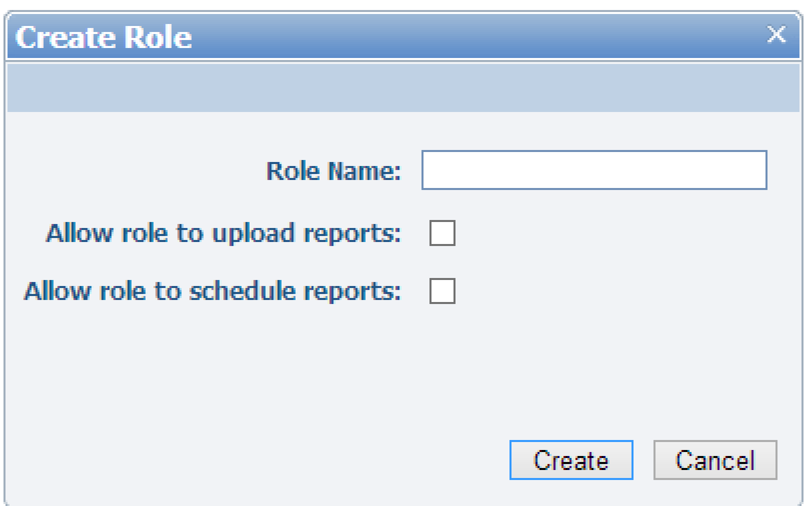


To create a new role

You can add as many roles as you need to the list. Then if you decide to grant permission for a report or model to an entire department, you can make the change once instead of for each individual.

Create roles with specified permissions

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**. The Roles list appears.
2. Next to New Role, click **Create**. The Create Role dialog appears.



3. In the **Role Name** box, enter a name for the new role.
4. If you want to allow the role to upload reports, select the **Allow role to upload reports** check box.
5. If you want to allow the role to schedule reports, select the **Allow role to schedule reports** check box.
6. Click **Create**. The new role is added to the list, with an extra command that allows you to delete it. To begin using the role, add users to it. To add more permissions, see **Managing Permissions: Reports, Models, Schedules**.

To add a user to a role

When you create a new role, it has no users.

Grant permissions by adding users to roles

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**.
2. In the Roles list that appears, to the right of the role to which you want to add a user, click the **Users** command. The Manage Role dialog appears.
3. In the Manage Role dialog, select check boxes next to users to assign them to the role.

   > **Note:** To remove a user from the role, clear the corresponding check box. You cannot add or remove users from the Owner role, which has a user for each report based on the Created By value for that report.

4. Click the **Apply** button to save the changes.

To grant permission to upload reports

By default, only the Administrator role has permission to upload reports.

Uploading ActiveReports

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**.
2. In the Roles list that appears, to the right of the role to which you want to grant upload permission, click the **Edit** command. The options are enabled.
3. Select the **Allow role to upload reports** checkbox.
4. To the right of the role, click the **Update** command. All users assigned to the role are able to upload reports.

To grant permission to schedule reports

By default, only the Administrator role has permission to schedule reports.

Scheduling ActiveReports

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**.
2. In the Roles list that appears, to the right of the role to which you want to grant scheduling permission, click the **Edit** command. The options are enabled.
3. Select the **Allow role to schedule reports** checkbox.
4. To the right of the role, click the **Update** command. All users assigned to the role are able to schedule reports.

To delete a role

To delete a role

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**.
2. In the Roles list that appears, to the right of the role that you want to delete, click the **Delete** command.
3. If the role does not contain any users, it is removed from the list immediately. If the role does contain users, you are asked to confirm.
4. Click **OK** to delete the role.

> **Note:** The roles **Admin**, **Everyone**, and **Owner** cannot be deleted; therefore the **Delete** command is not available for these roles.

You can grant other permissions to reports and models by selecting roles to allow for each individual report and each model. See **Managing Permissions: Reports, Models, Schedules** for more information.

Managing Permissions: Reports, Models, Schedules

Administrators can modify permissions to data models, reports, and shared schedules by assigning rights to a role. Once a role has permission to use a model, report, or shared schedule, all users added to that role have the same permission.

Table defining what permissions are granted for each combination of settings for reports and models

Report Permission Setting	Model Read	Model Read & Create	Model None
Report None	No operations	No operations	No operations

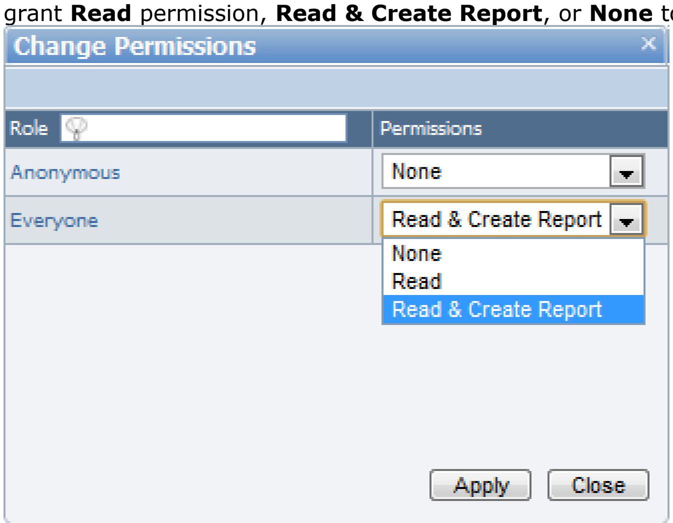
Report Read	Preview only	Preview only	Preview only
Report Read & Write	Preview, delete, and modify report	Preview, delete, and modify report	Preview and delete report
Report Full	Preview, delete, and modify report Change access	Preview, delete, and modify report Change access	Preview and delete report Change access
Create New Report	No	Yes	No

To manage permissions to a model

You can grant read or read and create permission to different roles for each model.

Allow or deny read permission to a model

1. From the Administrator Dashboard, select **Models**.
2. On the **Models** page, next to the model on which you want to set permissions, click **Permissions**.
3. In the **Change Permissions** dialog that appears, next to each role, drop down the box to select whether to grant **Read** permission, **Read & Create Report**, or **None** to hide the model from users in that role.



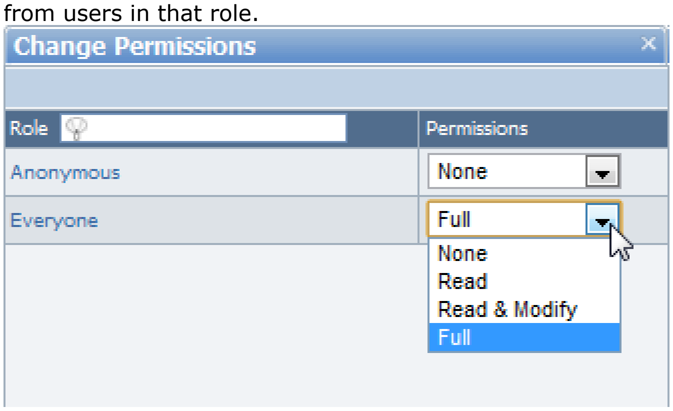
4. Click the **Apply** button to save the changes. The selected roles have permission to read or read and create reports on the model.

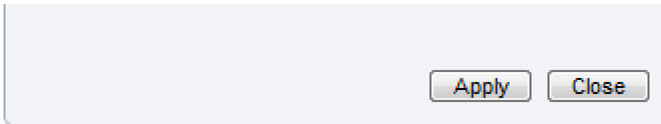
To manage permissions to a report

You can grant different roles permission to read, modify, or change permissions on each report. The Owner of the report (the user who created or uploaded it) has full permission which cannot be modified.

Allow or deny read, modify, or change permissions

1. From the Administrator Dashboard, select **Reports**.
2. On the **Reports** page, next to the report on which you want to set permissions, click **Permissions**.
3. In the **Change Permissions** dialog that appears, next to each role, drop down the box to select whether to allow its users **Read**, **Read & Modify**, or **Full** permissions on the report, or **None** to hide the report from users in that role.





- **Read** allows users in the role to see the report in the Reporting Portal **Open** page and preview the rendered report, but does not allow them to modify it in the Reporting Portal.
- **Read & Modify** allows users in the role to modify the report in the Reporting Portal in addition to Read permissions.
- **Full** allows users in the role to modify permissions on the report in addition to Read & Modify permissions.

4. Click the **Apply** button to save the changes. All users in the selected roles are granted the specified access.

To change ownership of a report

By default, the user who created or uploaded the report has full permission to it. The Owner role always has full permission to the report, so the only way to change that is to change ownership by deleting the report and having a new owner upload it to the server.

Change ownership

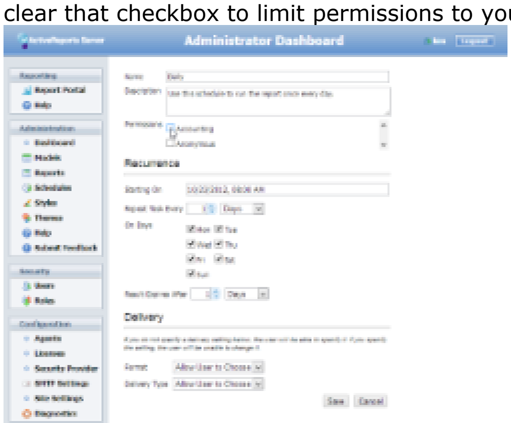
1. From the Administrator Dashboard, select **Reports**.
2. On the **Reports** page, next to the report for which you want to change ownership, click **Download**. The RDLX file of the report is copied to your local machine.
3. Next to the same report, click **Delete**. In the message box that appears, click **Yes** to confirm that you want to delete the report. The report is removed from the server.
4. Have the new owner of the report log in.
5. On the Reports page, next to **Upload Existing Report**, click **Browse**.
6. In the Open dialog that appears, navigate to the report and click **Open**. A fake file path appears in the text box.
7. Click **Upload**. The report appears in the list with the new owner listed under the **Created By** column.

To manage permissions to a shared schedule

You can grant different roles permission to use shared schedules to schedule reports to run. You can also set options in the Delivery section to allow or prevent users from changing the settings. For more information, see **Managing Schedules**.

Allow or deny read, modify, or change permissions

1. From the Administrator Dashboard, select **Schedules**.
2. On the **Schedules** page, next to the schedule on which you want to set permissions, click **Edit**.
3. In the schedule settings, next to Permissions, scroll through the list of roles and select the checkbox for each role whose users you want to allow to schedule reports. By default, **Everyone** is allowed to schedule reports, so clear that checkbox to limit permissions to your selected roles.



4. Click the **Save** button to save the changes. All users in the selected roles are granted scheduling permissions.

Using LDAP and Active Directory for Single Sign-On

You can use your existing LDAP or Active Directory® with ActiveReports 8 Server so that your users can log in with their existing user names and passwords, and so that you can use existing groups to provide access to specific data.

> **Note**: ActiveReports 8 Server supports OpenLDAP v2.3 or later. Earlier versions are not supported because they do not have a MemberOf attribute.

To map your user directory to ActiveReports 8 Server, you specify LDAP (Active Directory) as your custom security provider, and set properties to give ActiveReports 8 Server access to it. These properties are described in the table below.

Security provider properties table

Property	Description
LDAP Server URL (required)	The URL to the directory where you can find users. The syntax is: `ldap://host:port/domain` • **ldap://** is an optional prefix • **host** is the LDAP server name or IP address • **port** is the optional LDAP server port number (default is 389) • **domain** is the optional distinguished name of the domain in which to search for users **Example:** `ldap://example.org:888/DC=grapecity,DC=net` In this example, *example.org* is the name of the server answering LDAP queries, *888* is a non-standard port used on the LDAP server, and *grapecity.net* is the ActiveDirectory domain.
LDAP Admin User (required)	The name or DN (distinguished name) of the user under which the LDAP connection is established. This user is also used to implement the ISecurityProvider.GetAdminContext method when you add a custom entity. **Note**: Most non-Active Directory LDAP servers require a DN (distinguished name) for this, so a DN in the administrator account would be the directory manager. **Example:** `cn=Directory Manager,cn=Root DNs,cn=config`
LDAP Admin Password (required)	The password associated with the administrative user on the LDAP server. **Example:** `*****`
User Name Attribute (optional)	By default, this value is **samaccountname** and specifies the LDAP attribute to use in resolving the user name passed in the ISecurityProvider.GetUserToken method. **Example:** `uid`
User Display Name Attribute (optional)	By default, this value is **displayName** and specifies the LDAP attribute to use in resolving the user-friendly name with the ISecurityProvider.GetUserDescription method. **Example:** `sn`
User Email Attribute (optional)	By default, this value is **mail** and specifies the LDAP attribute to use in resolving the user e-mail with the ISecurityProvider.GetUserDescription method. **Example:** `mail`
Define Mappings	**Description**
User Context Attribute Name	LDAP User Attribute Name

To configure ActiveReports 8 Server to use your LDAP or Active Directory

1. From the Administrator Dashboard, in the Configuration section on the lower left, click **Security Provider**. The Custom security provider drop-down list appears.
2. Drop down the list and select **LDAP (Active Directory)**.
3. In the Security Provider Settings that appear, enter a value for each of the properties.
4. Below the Security Provider Settings, in the Attribute Name mapping section, under **Actions**, click **Add**.
5. In the boxes that appear, enter a valid **LDAP User Attribute Name**, and a **User Context Attribute Name** that you will be able to select when you create security filters in the model editor. (See the **To create a new security filter** section below.)

To test your custom security provider

You can place percent signs around a UserContext attribute to use it in a connection string when you **create a model**.

This is useful when each tenant in a multi-tenant application has a separate database, and you need to supply a dynamic value for the database.

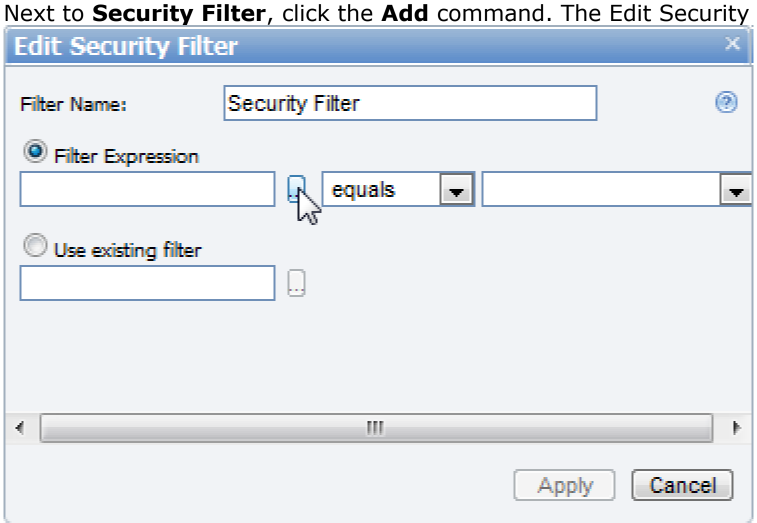
```
Provider=SQLOLEDB.1;Data Source=HQ;Initial Catalog=%TenantDatabase%;
User Id=myUsername;Password=myPassword;
```

When you create a security filter, you can re-use it for other entities that contain the attribute on which the filter expression is based.

To create a new security filter for row-level security

From the Administrator Dashboard, in the Administration section on the left, click **Models**. The Models list appears.

1. Next to the model to which you want to add a filter, click **Edit**. The model editor appears.
2. Select the entity that you want to filter. The editable properties appear in the center workspace.
3. Next to **Security Filter**, click the **Add** command. The Edit Security Filter dialog appears.



4. In the box next to **Filter Name**, enter a name for your filter.
5. With **Filter Expression** selected, click the ellipsis button to open the Select Attribute dialog where you can select an attribute of the entity on which to filter data, and click **OK**.
6. Drop down the list next to the attribute to select whether to display data that **equals** or does **not equal** the following value.
7. Drop down the list to the right, and select a User Context Attribute Name mapping against which to filter the selected attribute.
8. Click the **Apply** button to create the filter, or to discard the changes, click **Cancel**.

To use UserContext attributes from LDAP or ActiveDirectory in a connection string

You can place percent signs around a UserContext attribute to use it in a connection string when you **create a model**. This is useful when each tenant in a multi-tenant application has a separate database, and you need to supply a dynamic value for the database.

```
Provider=SQLOLEDB.1;Data Source=HQ;Initial Catalog=%TenantDatabase%;
User Id=myUsername;Password=myPassword;
```

User Context for Multi-Tenancy

If you use ActiveReports, you can set UserContext attribute in your existing Page Report, RDL report or Section report and uload the report on ActiveReports 8 Server to provide information that is specific to the user that is logged into the server. You can set UserContext attribute as a value or in a connection string. Using this feature you can filter your report data depending on the user roles and attributes that you set. UserContext attribute is particularly helpful when you have multiple tenants who are logged on to the same portal and you want to restrict or filter the data available to each tenant.

Setting UserContext attribute as a Value

UserContext attribute can be set as a value in reports created in ActiveReports. These reports can then be uploaded on the ActiveReports 8 Server to view the output based on the UserContext attribute value that has been set. For example, if ActiveDirectory is set as a Security Provider on the Server, **displayName** securiy token can be used as a UserContext object.

Following steps will guide you through the procedure to set UserContext values in the Value property of a control.

> **Caution:** Make sure the security token specified for the UserContext object is consistent with your Sercurity Provider settings.

For Page Report, RDL Report

1. Add or select the control for which you want to set the UserContext.
2. In the Properties window, enter the following expression in the Value property of the control.

   Expression

   ```
   =Code.UserContext.GetValue("Your_Security_Token")
   ```

For Section Reports

1. Add the control for which you want to set the UserContext.
2. In the script tab of Section Report, add the following script. In this example, UserContext has been set in the Text property of TextBox control.

   Script

   ```
   public void ActiveReport_ReportStart() {TextBox1.Text =
   UserContext.GetValue("Your_Security_Token"); }
   ```

Setting UserContext in Connection String

You can use any UserContext attribute in a connection string for reports created using ActiveReports. This is useful when each tenant in a multi-tenant application has a separate database, and you need to supply a dynamic value for the database.

In order to implement UserContext in a connection string, it is necessary to first complete your connection settings that is, connecting your report to an appropriate datasource, creating a data set and then re-placing datasource connection string with an expression having a Usercontext attribute. This is necessary because the ActiveReports does not have events to execute script code used for setting UserContext before data request.

For Page Report, RDL Report

1. Complete the connection setting for your report. See **Connecting a Report to a DataSource<CREATE LINK>** for details.
2. Re-open the Datasource dialog and in the **Connection String** tab, replace the existing connection string with an expression having the UserContext attribute. See the following expression for SQL Server connection string.

   Connection String

   ```
   "data source=ServerName;initial catalog=" +
   Code.UserContext.GetValue("Your_Security_Token") + ";user
   id=UserName;password=ServerPassword;"
   ```

For Section Reports

1. Complete the connection setting for your report. See **Connecting a Report to a DataSource<CREATE LINK>** for details.
2. Open the Script tab and add the following script to set UserContext attribute in an expression for SQL Server connection string.

   Script

   ```
   public void ActiveReport_DataInitialize()
   {
   var _ds = rpt.DataSource as GrapeCity.ActiveReports.Data.SqlDBDataSource;
   _ds.ConnectionString = "data source=ServerName;initial catalog=" +
   UserContext.GetValue("Your_Security_Token")

   + ";persist security info=False;user id=UserName;password=ServerPassword";
   }
   ```

Setting UserContext as a Value for Parameter

UserContext attribute can be set as a Value for a Parameter to retrieve specific information for the user that is logged into the server. Following steps will guide you through the procedure to set UserContext attribute as a Value for Parameters.

For Page Report, RDL Report

1. Open the Report Parameter dialog. See Add Parameters in a Page Report for details on How to add parameters.
2. In the Available Values tab, select the select the **Non-queried** radio button and click the Add button.
3. Enter the following expression in the Value property.

   Expression

   ```
   =Code.UserContext.GetValue("Your_Security_Token")
   ```

For Section Reports

> **Note:** These steps assume that a Section Report is already bound to a DataSource. See Bind Reports to a Data Source for details.

1. Open the Script tab and add the following script in the ReportStart event to set UserContext attribute as a Value for parameter in an expression for SQL Server connection string.

   Script

   ```
   public void ActiveReport_ReportStart()
   {
    report1.Parameters["paramName"].Value = UserContext.GetValue("Your_Security_Token");
    GrapeCity.ActiveReports.Data.SqlDBDataSource ds =
   (GrapeCity.ActiveReports.Data.SqlDBDataSource) report1.DataSource;
    ds.SQL = "Select * from TableName where FieldName ='" +
   rpt.Parameters["paramName"].Value + "'";
   }
   ```

Managing Configuration

This section contains information that helps administrators manage SMTP settings and diagnose problems.

Managing Agents
This section describes how ActiveReports 8 Server uses agents to handle scale-out deployment so that you can serve up more reports to more users as needed.

Managing Licenses
Learn about different licenses for ActiveReports 8 Server and how to manage them.

Managing Single Sign-On Security Providers
Learn about using custom security providers as well as LDAP or Active Directory® security providers.

Configuring Email
Learn to configure SMTP settings for your site to enable the error reporting and password reminder functions.

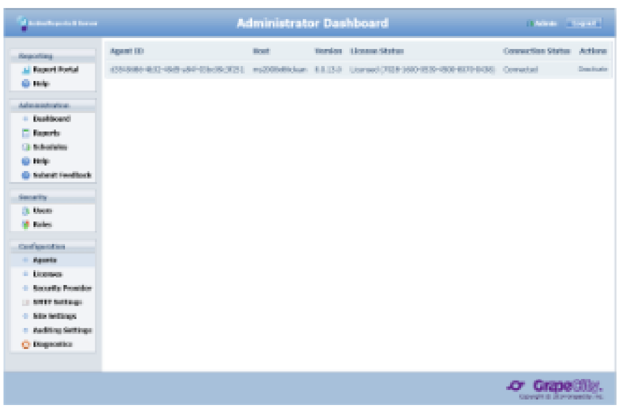
Site Settings Notification URL
Learn how to set a base URL for links sent in notifications for lost passwords and scheduled reports.

Managing Audit Settings
This section describes how to set up auditing on your site, and reports and data are automatically set up.

Server Diagnostics
Learn about monitoring your server with the ActiveReports 8 Server Diagnostics page.

Managing Agents

You can use agents to manage your report execution load and concurrent users in ActiveReports 8 Server. The agents list displays information about each agent to help you manage them. For more information, see **Agents and Scalability**.



Agent Information and Command

You can obtain information on agents and activate or deactivate them using the command to the right of each agent in the list.

Agent ID	Host	Version	License Status	Connection Status	Actions
3a256927-1ca4-4fb3-bded-c8f41fadb2ad	US-DOC-KIMN	0.6.1980.0	Licensed (0992-5600-6117-4600-9373-0333)	Connected	Deactivate

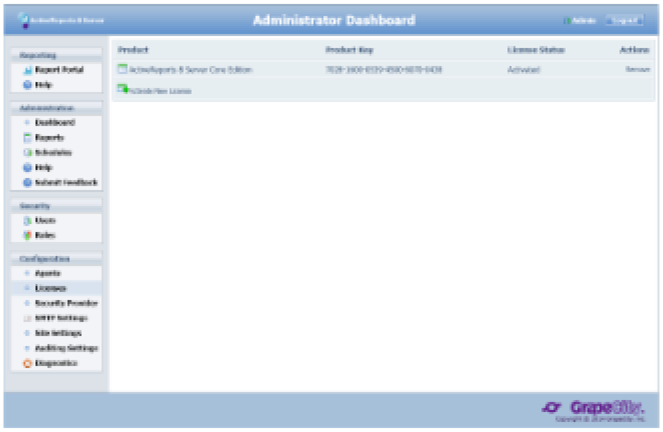
Column Descriptions

Column	Description
Agent ID	Displays a unique internal identification code for the agent.
Host	Displays the name of the hosting machine.
Version	Displays the ActiveReports 8 Server version number.
License Status	Displays licenses and serial numbers for licensed agents, or lets you know when no license is available. If the agent is disabled or cannot connect, the license goes back into circulation, and the license status is N/A.
Connection Status	Displays the state of the connection: **Connected** or **Cannot Connect**.

Deactivate (or Activate)	Takes the agent out of action, but allows you to enable it later. It remains connected, but no tasks are assigned to it, and its license is freed up for use by another agent. You can deactivate an agent and re-activate it to force it to check for an available license. When a new agent is installed, it is deactivated until an Administrator activates it.

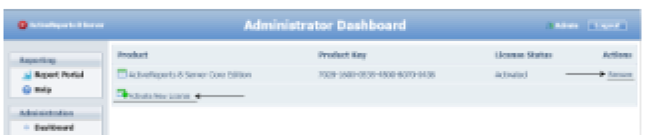
Managing Licenses

When you purchase an initial license, you receive a product key to unlock them. You can manage these licenses on the Administrator Dashboard Licenses page. For more information, see **Agents and Scalability**.



License Information and Commands

You can obtain information on licenses and remove them using the command to the right of each license in the list. You can add a new license using the command below the list.



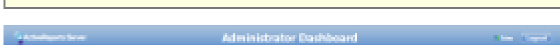
Column Descriptions

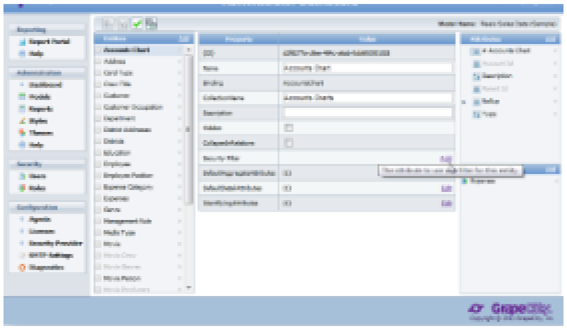
Column	Description
Product	Displays the type of license you purchased, which may be a Trial, Core, and Designer Addon Edition.
Product Key	Displays the product key that you received when purchasing the license.
License Status	Displays the status of the license, which may be Activated, or may show the number of days left on a trial license.
Actions	Click **Remove** to remove a license from the list. This allows you to clean up any expired evaluation licenses.
Activate New License	Click this button to add a new license. It opens a dialog where you can enter your product key and activate the license online. See **Activate a License** for more information.

Managing Single Sign-On Security Providers

You can provide row-level security with security filters. While editing a model, you can add and delete security filters from entities, and modify existing ones, as well as configure and use an LDAP security provider. All of these functions are in the Model Editor, which you can access from the Administrator Dashboard.

Note: In order to use single sign-on security filters, you must first create a custom security provider. For information on creating custom security providers, please see the SDK documentation at Create a Custom Security Provider.





To use an Active Directory® or LDAP security provider

When using a Lightweight Directory Access Protocol (LDAP) security provider, you can create User Context attributes that map to LDAP user attributes, and use them in security filter expressions to enable single sign-on for your users.

Use an LDAP security provider

1. From the Administrator Dashboard, in the Configuration section on the lower left, click **Security Provider**. The Custom security provider drop-down list appears.
2. Drop down the list and select **LDAP (Active Directory)**.
3. In the Security Provider Settings that appear, enter a value for each of the properties.
4. Below the Security Provider Settings, in the Attribute Name mapping section, under **Actions**, click **Add**.
5. In the boxes that appear, enter a valid **LDAP User Attribute Name**, and a **User Context Attribute Name** that you will be able to select when you create security filters in the model editor. (See the **To create a new security filter** section below.)

To add your new security provider to the server

Once you have compiled your custom security provider into a DLL, you can begin using it to filter entities in your models.

Add a new security provider

1. In Windows Explorer, create a SecurityProviders folder on the same level as your site, for example, if you have C:\ActiveReports 8 Server, create the folder C:\ActiveReports 8 Server\SecurityProviders.
2. Copy your custom security provider DLL file into the new folder.
3. From the Administrator Dashboard, in the Configuration section on the lower left, click **Security Provider**. The Custom security provider drop-down list appears.
4. Drop down the list and select your new security provider.
5. Depending on your security provider, there may be settings that you can select.
6. When you have the settings the way that you want them, click the **Apply changes** button. The security filter becomes available to your model entities.

To delete a security provider

You can stop the server site and remove security providers from ActiveReports 8 Server.

Delete a security provider

1. Open Internet Information Services (IIS) Manager, and expand the **Sites** node.
2. Select the **ActiveReports 8 Server** site, and on the right, in the Manage Web Site section, click **Stop**.
3. In Windows Explorer, locate your SecurityProviders folder.
4. Delete the DLL containing the security provider that you long longer wish to use. The provider is removed from the list.
5. Back in IIS, with the ActiveReports 8 Server site selected, in the Manage Web Site section, click **Start**.

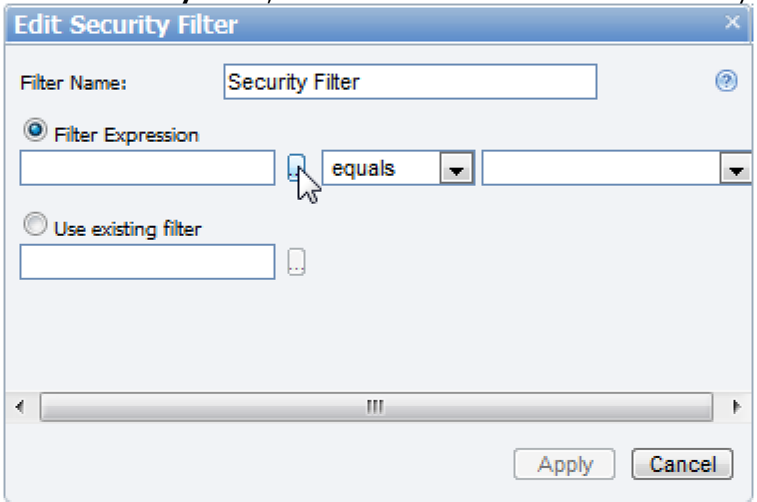
To create a new security filter

When you create a security filter, you can re-use it for other entities that contain the attribute on which the filter expression is based, providing row-level security for your data.

Create a new filter

1. From the Administrator Dashboard, in the Administration section on the left, click **Models**. The Models list appears.
2. Next to the model to which you want to add a filter, click **Edit**. The model editor appears.

3. Select the entity that you want to filter. The editable properties appear in the center workspace.
4. Next to **Security Filter**, click the **Add** command. The Edit Security Filter dialog appears.



5. In the box next to **Filter Name**, enter a name for your filter.
6. With **Filter Expression** selected, click the ellipsis button to open the Select Attribute dialog where you can select an attribute of the entity on which to filter data, and click **OK**.
7. Drop down the list next to the attribute to select whether to display data that **equals** or does **not equal** the following value.
8. Drop down the list to the right, and select a value against which to filter the selected attribute. If you are using an LDAP (Active Directory) security provider, any User Context Attribute Name mappings that you have created appear in this list.
9. Click the **Apply** button to create the filter, or to discard the changes, click **Cancel**.

To use an existing filter

You can re-use security filters with other entities that use the same attributes.

Use an existing filter

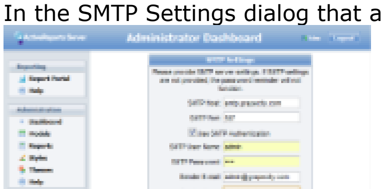
1. From the Administrator Dashboard, in the Administration section on the left, click **Models**. The Models list appears.
2. Next to the model to which you want to add a filter, click **Edit**. The model editor appears.
3. Select the entity that you want to filter. The editable properties appear in the center workspace.
4. Next to **Security Filter**, click the **Add** command. The Edit Security Filter dialog appears.
5. In the box next to **Filter Name**, enter a name for your filter.
6. With **Use existing filter** selected, click the ellipsis button to open the Select Existing Filter dialog where you can select a previously created filter (if it is based on an attribute shared by this entity), and click **OK**.
7. Click the **Apply** button.

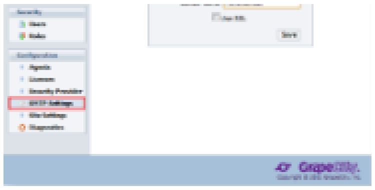
Configuring Email

To ensure that error reporting and password reminders function properly for your site, you can configure the SMTP settings. The report server also uses these settings to connect with the e-mail server to deliver scheduled reports. You can configure e-mail SMTP settings from the Administrator Dashboard.

To configure e-mail from the Administrator Dashboard

1. From the Administrator Dashboard, at the bottom left under Configuration, click **SMTP Settings**.
2. In the SMTP Settings dialog that appears, enter values for your SMTP server.





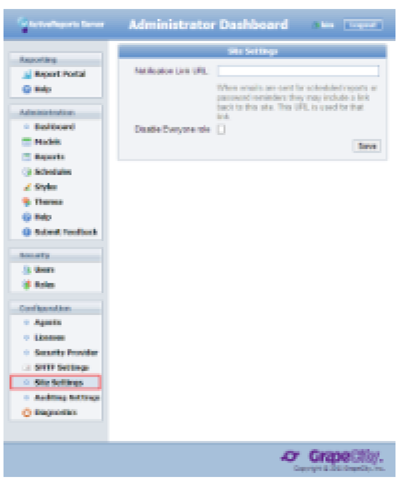
- **SMTP host** Enter the host address for your server, usually with syntax like "smtp.yourdomainname.com."
- **SMTP port** Enter the port number for your server, for example, 587.
- **Use SMTP authentication** Select this check box to add a user name and password.
- **Sender e-mail** Enter the e-mail address to use as the return address for error reporting, password reminders, and scheduled report delivery.
- **Sender name** Enter the name to use for the sender.
- **Use SSL** Select this check box to enable the Secure Sockets Layer protocol.

3. When you have finished entering the values for your server, click **Save**.

Site Settings Notification URL

When you schedule reports and choose **Email** as the delivery type, and **As Link** as the delivery method, the email that goes out to your users contains a link to the report location. Likewise, you may include a link in password reminder emails.

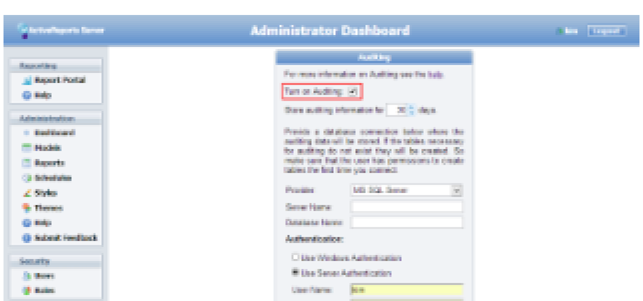
The Site Settings page allows you to set the base URL for creating these links.

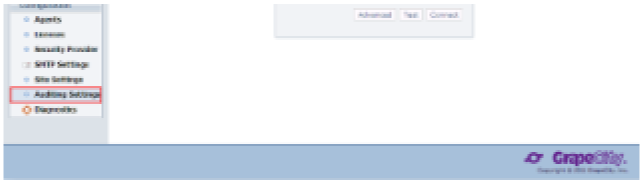


Managing Audit Settings

Whether you need to be HIPAA compliant and able to track who accessed personally identifiable data, or you just want to know which reports your clients actually use and which ones they do not use, auditing allows you to track which reports are accessed and what is done with them.

When you turn on auditing, in addition to setting the number of days to store audit data, you also specify a database in which to store the data.





Depending on the Provider you select, a different set of fields appears, allowing you to provide the information needed to connect to the database. Once you provide all of the information, click **Connect** to save it and begin auditing report usage statistics. ActiveReports 8 server asks you whether to create its own tables, and when you OK the action, it takes care of everything else, including adding an **Audit Model** and two **Audit Reports**.

Connection String Editor

If you just want to enter a connection string, you can click the **Advanced** button to open the Connection String Editor, where you can also specify a connection timeout period.

> **Tip**: You can use any UserContext attribute in the connection string by putting the attribute name between percent signs. For example:
>
> ```
> Provider=SQLOLEDB.1;Data Source=HQ;Initial Catalog=%TenantDatabase%;
> User Id=myUsername;Password=myPassword;
> ```

Provider

Here are descriptions of the fields that you need to fill in for each type of provider.

MS SQL Server

Field	Description
Server Name	Enter the name of the server on which to store auditing data, for example, HQ.
Database Name	Enter the name of the database in which to store auditing data, for example, REPORTAUDITS.
Authentication	Select whether to use Windows authentication or server authentication to provide access to the database.
User Name	Enter the user name under which to log in to the database.
Password	Enter the password to use for logging in to the database.

MySQL Server

Field	Description
Server Name	Enter the name of the server on which to store auditing data, for example, localhost.
Database Name	Enter the name of the schema object in your SQL Editor Object Browser in which to store auditing data, for example, ARServerAudit.
User Name	Enter the user name under which to log in to the database.
Password	Enter the password to use for logging in to the database.

Oracle

Field	Description
Service Name	Enter the name of the service on which to store auditing data.
User Name	Enter the user name under which to log in to the database.
Password	Enter the password to use for logging in to the database.

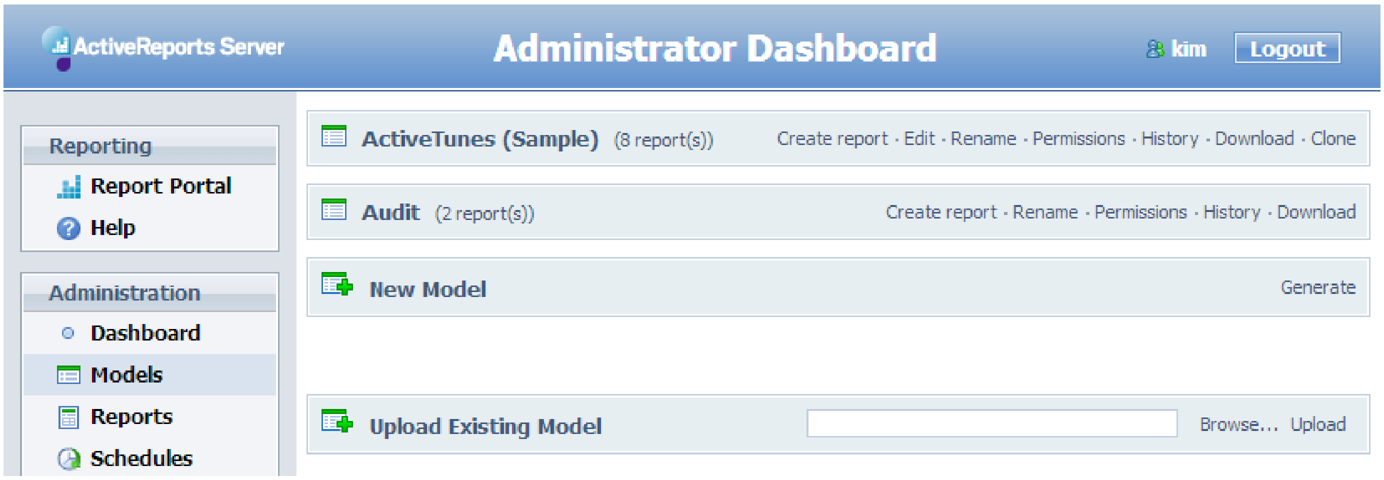
Postgre SQL

Field	Description
Host	Enter the name of the host machine on which to store auditing data, for example, localhost.
Port	Enter the port number to use on the host machine, for example, 8080.

Database Name	Enter the name of the database in which to store auditing data, for example, REPORTAUDITS.
User Name	Enter the user name under which to log in to the database.
Password	Enter the password to use for logging in to the database.

Audit Model

Once you turn on auditing, provide the information needed to connect to a database, and click **Connect** to save it, ActiveReports 8 server asks you whether to create its own tables. When you OK that action, it immediately begins auditing report usage statistics, saving the data in the specified database, and automatically creates a data model called Audit.



In the Models list, the Audit model has **two reports** by default, and differs from other models in that you cannot edit or clone it. This is to satisfy requirements for secure audit data.

Like other models, you can create new reports using its data, rename it, set permissions on it to allow other users to access it, view its history, and download it.

Audit Reports

Once you turn on auditing, provide the information needed to connect to a database, and click **Connect** to save it, ActiveReports 8 server asks you whether to create its own tables. When you OK that action, it immediately begins auditing report usage statistics, saving the data in the specified database, and automatically creates two auditing reports.

Reports contain data for the number of days that you specify when you set up auditing.

Auditing Access Summary

The Auditing Access Summary report logs all of the following types of access for the most recent seven day period:

- Recently accessed reports, including date, report name, user, and client IP address.
- Recently accessed models, including date, model name, action type, user, and client IP address.
- User logon and logoff attempts, including date, user, client IP address, and success or fail.

Auditing Activity Summary

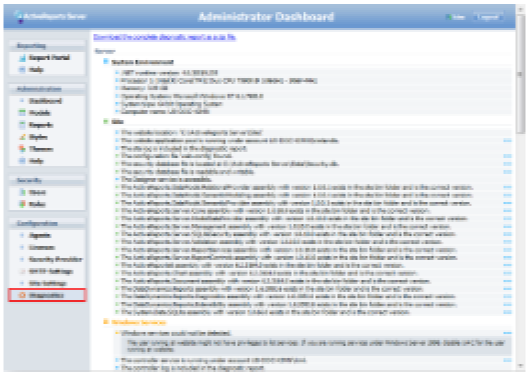
The Auditing Activity Summary report logs all of the following action types for every report for the most recent 30 day period:

- Download
- Logon
- ModelGeneration
- ParametersResolving
- Rendering
- Upload

You can also create custom reports on your auditing data, as you would on any other data model.

Server Diagnostics

The **Diagnostics** page allows you to perform run-time monitoring of the server that hosts ActiveReports 8 Server. This page provides information on the application system environment, its installed files, assemblies and services.



To open the Diagnostics page

1. From the Administrator Dashboard, at the bottom left under Configuration, click **Diagnostics**.
2. In the item description list that appears, click the Toggle visibility icon next to an item to display details.
3. If you want to send a diagnostic report in for support, you can click the command at the top of the page to **Download the complete diagnostic report as a zip file.**

If you encounter an error, you are presented with the Error Reporting dialog. The information we collect from this dialog is protected by our privacy policies. For more information, see **Privacy**.

Privacy

Privacy Statement for the ActiveReports 8 Server Error Reporting Service

Last updated: 6/2/2011

GrapeCity is committed to helping protect your privacy. This statement explains how the ActiveReports 8 Server Error Reporting Service collects information and how collected information can be used. This statement does not apply to other online or offline GrapeCity Web sites, software, or services.

Why does GrapeCity collect information about errors and problems?

The information helps GrapeCity and ActiveReports 8 Server diagnose problems in the software you use and provide solutions. Not all problems have solutions but when solutions are available, they are offered as steps for solving a problem you've reported or as updates to install. To help prevent problems and make software more reliable, some solutions are also included in service packs and future versions of the software.

How is information collected?

GrapeCity ActiveReports 8 Server includes its own error reporting service. If a problem occurs in this software, you are asked if you want to report it. You can view the details of the report before sending it, although some files might not be in a readable format.

Some software also allows you to report problems automatically instead of requesting your consent each time a problem occurs. If you use automatic reporting, you are not prompted to review the information in a report before it is sent. However, no information is collected unless you (or your system or network administrator) choose to report problems. You can choose to stop reporting problems at any time.

What types of information can be collected?

The reporting service can collect information about problems that interrupt you while you work and about errors that occur behind the scenes. It is important to diagnose errors that occur behind the scenes because these problems, if left unsolved, may cause additional problems such as performance or program failures.

Reports contain information that is most useful for diagnosing and solving the problem that has occurred, such as:

- Where the problem happened in the software or hardware. Occasionally, empty files might be included as an initial indication of a problem.
- Type or severity of the problem, if known.
- Files that help describe the problem (typically system or report-generated files about software behavior before or after the problem occurred).
- Basic software and hardware information (such as operating system version and language, device models and manufacturers, or memory and hard disk size).

Your Internet Protocol (IP) address is also collected because you are connecting to an online service (web service) to send error reports. However, your IP address is used only to generate aggregate statistics. It is not used to identify you or contact you.

Reports might unintentionally contain personal information, but this information is not used to identify you or contact you. For example, a report that contains a snapshot of memory might include your name, part of a document you were working on, or data that you recently submitted to a Web site. If you are concerned that a report might contain personal or confidential information, you should not send the report.

Who can use the information and how can it be used?

GrapeCity uses information about errors and problems to improve ActiveReports 8 Server. GrapeCity employees and contractors may be provided access to information collected by the reporting service. However, they may use the information only to repair or improve the products that they publish or manufacture.
For example, if an error report indicates that a third-party product is involved, GrapeCity may send that information to the vendor of the product. The vendor may provide the information to sub-vendors and partners. If a third-party product is involved, and information is transferred to the vendor of that product, that vendor will be subject to its own policies. GrapeCity cannot monitor such vendor's compliance with its policies.

About surveys and report tracking

After you report a problem, you might be asked to complete a survey about the error experience. If you choose to provide a phone number or e-mail address in response to the survey, your error report will no longer be anonymous. GrapeCity may contact you to request additional information to help solve the problem you reported.

You might also be provided with the opportunity to track some error reports so you can check to see if a solution has been found. If you choose to track an error report, the report is associated with your e-mail address and is no longer anonymous.

Information storage, processing, and release

Information that is collected by or sent to GrapeCity may be stored and processed in the United States or any other country in which GrapeCity or its affiliates, subsidiaries, or agents maintain facilities. GrapeCity may disclose this information if required to do so by law or in the good faith belief that such action is necessary to: (a) conform to the edicts of the law or comply with legal process served on GrapeCity or the site; (b) protect or defend the rights or property of GrapeCity and its affiliates, or (c) act in urgent circumstances to protect the personal safety of GrapeCity employees, users of GrapeCity software or services, or members of the public.

GrapeCity occasionally hires other companies to provide limited services on its behalf, such as providing customer support, processing transactions, or performing statistical analysis of reports. GrapeCity will provide these companies only the information they need to deliver the service. They are required to maintain the confidentiality of the information and are prohibited from using it for any other purpose.

Security practices

GrapeCity is committed to helping protect the security of the information we collect. The Error Reporting Service uses multiple security technologies and procedures to help protect information from unauthorized access, use, or disclosure. For example, reports are sent to GrapeCity from your computer using encryption technology. The information is then stored on computer servers with controlled access.

Changes to this statement

GrapeCity may update this privacy statement. If we do, we will revise the "Last updated" information at the top of the statement. To stay informed about how we are helping protect the information collected by the Error Reporting Service, we encourage you to periodically review the privacy statement.

For more information

GrapeCity welcomes your comments regarding this privacy statement. If you believe that GrapeCity has not adhered to this statement, please contact us. We will use commercially reasonable efforts to promptly determine and remedy the problem.

GrapeCity Contact Information

720, 4th Avenue, Suite 220, Kirkland, WA 98033-8136
Tel : +1 (425) 828-4440 Fax : +1 (425) 828-4590

To find contact details for the GrapeCity subsidiary or affiliate in your country or region, see the GrapeCity corporate Web site at http://www.componentone.com.

How To

This section provides quick answers to your questions about how to perform specific tasks with ActiveReports 8 Server.

Create Users and Roles
This topic explains how to set up roles and groups so that users only have access to specified functions.

Activate a License
This topic explains how to license ActiveReports 8 Server with your product key.

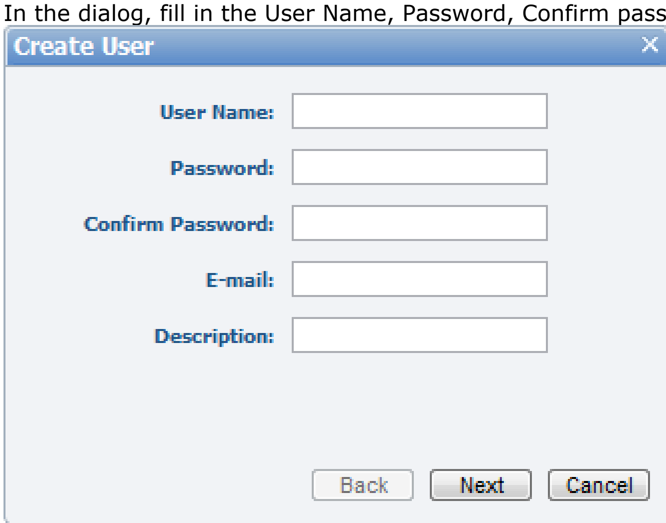
Add Row-Level Security
This topic explains how to license ActiveReports 8 Server with your product key.

Create Users and Roles

To control which users have access to data models and reports, you can set up users and group them by roles. You can assign roles access to each report and data model, and you can designate which roles are able to upload reports.

To set up a user

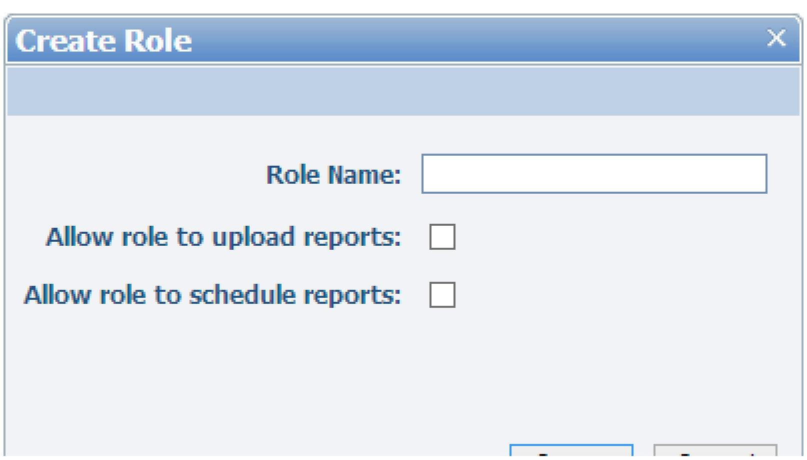
1. From the Administrator Dashboard, in the Security section on the lower left, click **Users**. The Users list appears.
2. Next to New User, click **Create**. The Create User dialog appears.
3. In the dialog, fill in the User Name, Password, Confirm password, E-mail, and Description fields.

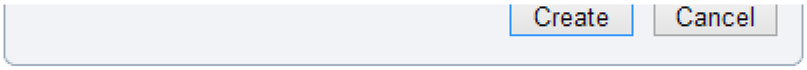


4. Click **Create**. The new user is added to the Users list, and the user can log in to the Reporting Portal with the user name and password specified.

To set up a role

1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**. The Roles list appears.
2. Next to New Role, click **Create**. The Create Role dialog appears.





3. In the **Role Name** box, enter a name for the new role.
4. If you want to allow the role to upload reports, select the **Allow Upload Report** check box.
5. Click **Create**. The new role is added to the list, with an extra command that allows you to delete it. To begin using the role, add users to it.

To add a user to a role

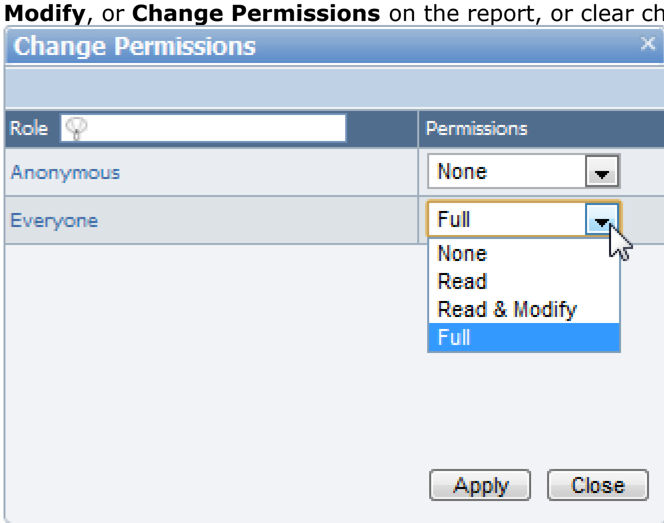
1. From the Administrator Dashboard, in the Security section on the lower left, click **Roles**.
2. In the Roles list that appears, to the right of the role to which you want to add a user, click the **Users** command. The Manage Role dialog appears.
3. In the Manage Role dialog, select check boxes next to users to assign them to the role.

> **Note:** To remove a user from the role, clear the corresponding check box.

4. Click the **Apply** button to save the changes.

To give a role permissions on a report

1. From the Administrator Dashboard, select **Reports**.
2. On the **Reports** page, next to the report on which you want to set permissions, click **Permissions**.
3. In the **Change Permissions** dialog that appears, select check boxes next to each role to allow its users to **Read**, **Modify**, or **Change Permissions** on the report, or clear check boxes to deny permission.

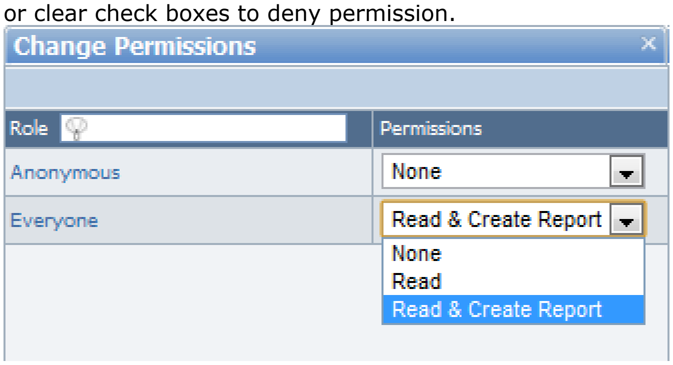


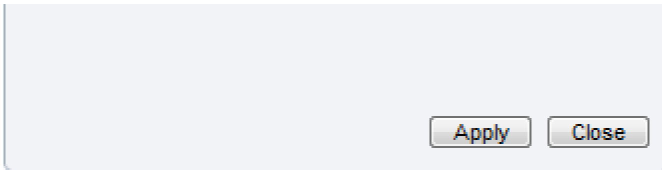
- **Read** allows users in the role to see the report in the Report Portal **Open** page and preview the rendered report, but does not allow them to modify it in the Reporting Portal.
- **Modify** allows users in the role to modify the report in the Reporting Portal.
- **Change Permissions** allows users in the role to modify permissions on the report.

4. Click the **Apply** button to save the changes. All users in the selected roles are granted the specified access.

To give a role permissions on a model

1. From the Administrator Dashboard, select **Models**.
2. On the **Models** page, next to the model on which you want to set permissions, click **Permissions**.
3. In the **Change Permissions** dialog that appears, select check boxes next to each role to allow **Read** permission, or clear check boxes to deny permission.

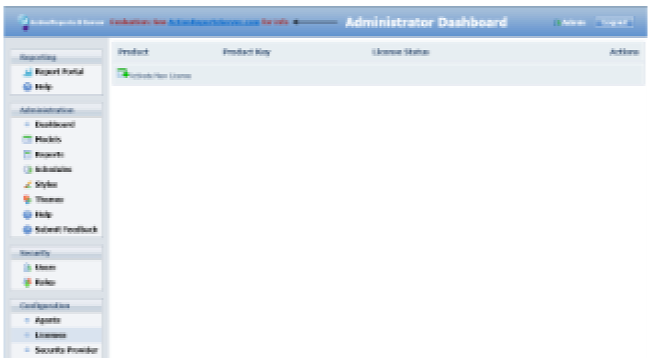




4. Click the **Apply** button to save the changes. The selected roles have permission to create reports on the model.

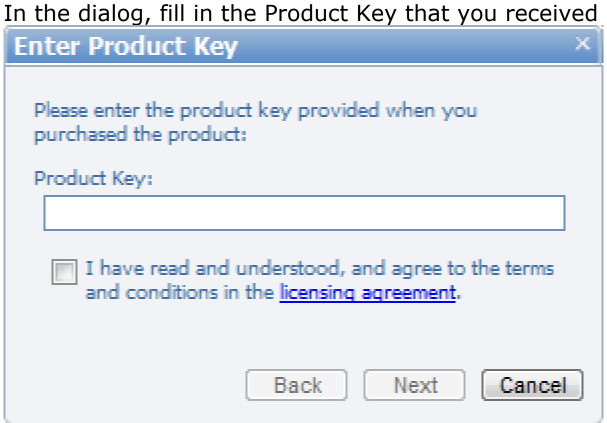
Activate a License

If you have not yet licensed ActiveReports 8 Server, you will see an evaluation message in the top bar of both the Administrator Dashboard and the Reporting Portal. To remove the evaluation message, license the server using the product key you received via e-mail when you purchased the product.



To license your server

1. From the Administrator Dashboard, in the Configuration section on the lower left, click **Licenses**. The Licenses list appears.
2. Click **Activate New License**. The Enter Product Key dialog appears.
3. In the dialog, fill in the Product Key that you received when you purchased ActiveReports 8 Server.



4. Select the check box to indicate your acceptance of the licensing agreement terms, and click **Next**. The Authentication Number fills in automatically.
5. Click the link that appears in Step 2 to open the activation site in a new browser tab. Your Product Key and Authentication Number fill in automatically.
6. Click **Send Request**. Your License Key appears, along with your activation date. Select the License Key number and copy it.
7. Return to the browser tab with the Administrator Dashboard, paste the License Key into Step 3 of the dialog and click **Finish**. The evaluation message is removed, and the product key is listed in the Licenses list with a License Status of Activated.

Add Row-Level Security

To control access to row-level data based on user permissions, you must create security filters and apply them to the model entities that you want to secure. Security filters can be based on LDAP or Active Directory, or you can create a custom security provider.

For information on creating custom security providers, please see the SDK documentation at Create a Custom Security Provider, or the sample in C:\ActiveReports 8 Server\SDK\ActiveTunes.SecurityProvider.

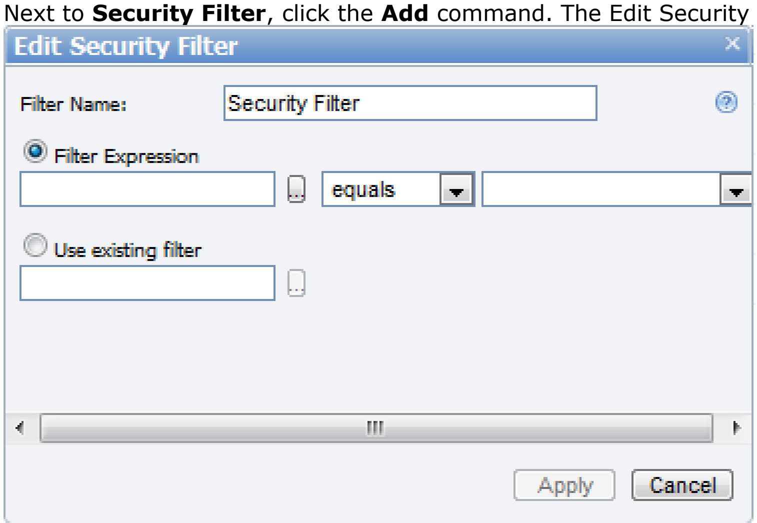
If you use LDAP or Active Directory, please see **Using LDAP and Active Directory for Single Sign-On**. If you have created a custom security provider, follow the steps below.

To configure ActiveReports 8 Server to use your custom security provider

1. From the Administrator Dashboard, in the Configuration section on the lower left, click **Security Provider**. The Custom security provider drop-down list appears.
2. Drop down the list and select your custom security provider.
3. In the Security Provider Settings that appear, enter a value for any properties that appear.
4. When you have entered all of the information, click the **Apply changes** button. Your custom security provider becomes available for use in security filters.

To create a new security filter for row-level security

1. From the Administrator Dashboard, in the Administration section on the left, click **Models**. The Models list appears.
2. Next to the model to which you want to add a filter, click **Edit**. The model editor appears.
3. Select the entity that you want to filter. The editable properties appear in the center workspace.
4. Next to **Security Filter**, click the **Add** command. The Edit Security Filter dialog appears.



5. In the box next to **Filter Name**, enter a name for your filter.
6. With **Filter Expression** selected, click the ellipsis button to open the Select Attribute dialog where you can select an attribute of the entity on which to filter data, and click **OK**.
7. Drop down the list next to the attribute to select whether to display data that **equals** or does **not equal** the following value.
8. Drop down the list to the right, and select a User Context Attribute Name mapping against which to filter the selected attribute.
9. Click the **Apply** button to create the filter, or to discard the changes, click **Cancel**.

Troubleshooting

Here you can find solutions to issues that you may encounter while working with ActiveReports 8 Server. Click a description below to view the symptoms, cause, and solution.

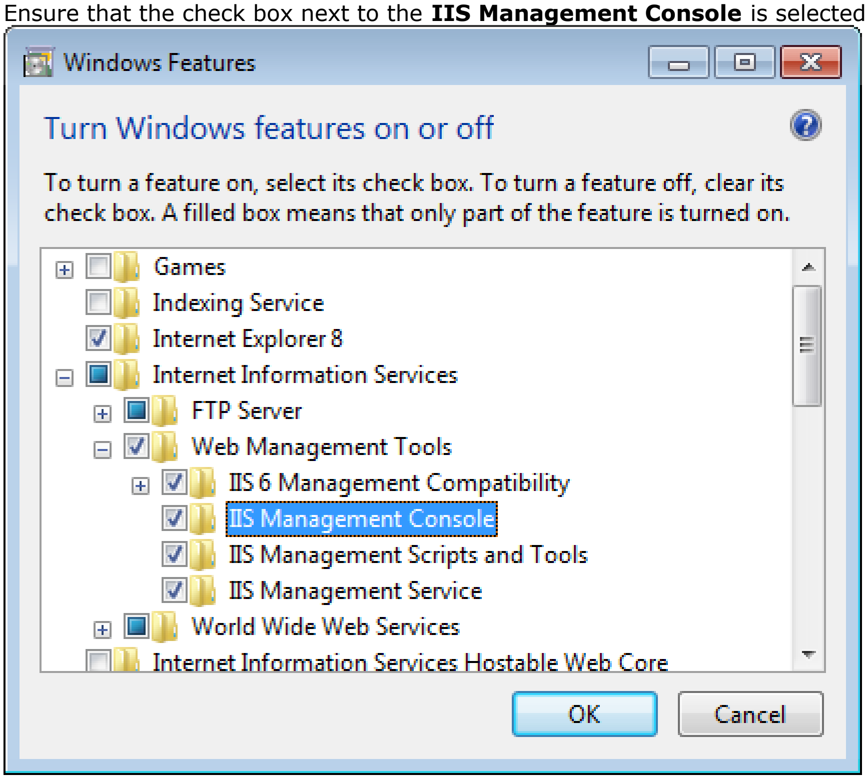
When trying to open the Reporting Portal page, you get the Server Error "Could not connect to endpoint DesignerService.svc on the server."

Symptoms: The Reporting Portal page will not open. Instead, you get a Server Error that reads: "Could not connect to endpoint DesignerService.svc on the server. Please verify connectivity to the server and check the server configuration. HTTP request error"

Causes: The IIS Management Console is turned off by default, or WCF is not configured on your machine.

Solution 1: Turn on the IIS Management Console.

1. Open Control Panel and select **Programs and Features**.
2. On the left, select **Turn Windows features on or off**.
3. In the Windows Features window that appears, expand the **Internet Information Services** node, then the **Web Management Tools** node.
4. Ensure that the check box next to the **IIS Management Console** is selected and click **OK**.



Solution 2: Configure WCF on your machine.

1. From the Start menu, open a command prompt (cmd.exe).
2. In the command window, enter **cd %SystemRoot%\Microsoft.Net\Framework\v3.0\Windows Communication Foundation** to change to the directory where the ServiceModel Registration Tool is installed.
3. For Framework 4 or 4.5, enter the command: **ServiceModelReg.exe -ia** to run the tool to install all WCF and WF components. (For Framework 3 or 3.5, the command is: **ServiceModelReg.exe -ir**.)

Solution 3: Add features to WCF Services on the server.

1. From the Start menu, select **Server Manager**.
2. On the Manage menu, select **Add Roles and Features**.
3. In the Add Roles and Features window, select the **Features** tab.
4. Under Features, expand the .NET Framework 4.5 (Installed) node, then the **WCF Services (Installed)** node.
5. Under WCF Services (Installed), select the following three features:
   - HTTP Activation
   - Named Pipe Activation
   - TCP Activation

6. Click the **Install** button.

When trying to open the Administrator Dashboard or Report Portal page, you get HTTP Error 503 Service Unavailable.

Symptoms: The Administrator Dashboard and Report Portal pages will not open. Instead, you get HTTP Error 503.

Cause: The application pool for the service is stopped.

Solution: Restart the application pool.

1. Open IIS and expand your localhost node.
2. Select **Application Pools**.
3. In the Application Pools page that appears, find **ASP.NET v4.0 AR_AppPool** and check its Status.
4. If the Status is Stopped, right-click **ASP.NET v4.0 AR_AppPool** and select **Start**.

When trying to reinstall, you get the error: "Service 'ActiveReports 8 Server' failed to start. Verify that you have sufficient privileges to start system services."

Symptoms: The service will not start when you try to reinstall. Instead, you get "Service 'ActiveReports 8 Server' failed to start."

Cause: The previous version did not uninstall completely.

Solution: Uninstall the remnants of the previous version before reinstalling.

1. In Windows Explorer, delete the directory C:\ActiveReports 8 Server.
2. Open IIS, expand the localhost, then the **Sites** node, and delete the **ActiveReports 8 Server** site.
3. Right-click the installer and select **Run as administrator** and follow the installation instructions.

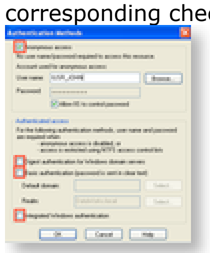
The reports list is not displayed on the sample web page

Symptoms: The report list is not displayed on the sample web page.

Cause: When deploying samples, two access options in the Internet Information Services (IIS) settings are selected: "Enable anonymous access" and "Integrated Windows authentication."

Solution: Change the IIS settings to use only a single authentication scheme, "Enable anonymous access."

1. In the Control Panel, open **Administrative Tools**, then **Internet Information Services Manager**.
2. In the **Internet Information Services** window that appears, expand the tree view in the left pane until you see the **Web sites/sample** node.
3. Right-click **Web Sites/sample** to open the **Web Sites Properties** dialog.
4. On the Directory Security tab, in the **Authentication and access control** section, click **Edit**.

5. In the Authentication Methods dialog that appears, ensure that only one authentication scheme is selected in the corresponding checkbox.

6. Click **OK** to apply the changes, and **OK** again to close the dialog.
7. Back in Visual Studio, run the sample again.

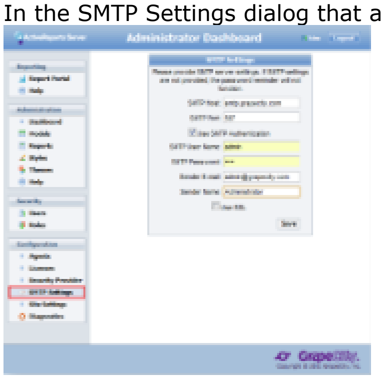
Errors

Configuring Email

To ensure that error reporting and password reminders function properly for your site, you can configure the SMTP settings. The report server also uses these settings to connect with the e-mail server to deliver scheduled reports. You can configure e-mail SMTP settings from the Administrator Dashboard.

To configure e-mail from the Administrator Dashboard

1. From the Administrator Dashboard, at the bottom left under Configuration, click **SMTP Settings**.
2. In the SMTP Settings dialog that appears, enter values for your SMTP server.



- **SMTP host** Enter the host address for your server, usually with syntax like "smtp.yourdomainname.com."
- **SMTP port** Enter the port number for your server, for example, 587.
- **Use SMTP authentication** Select this check box to add a user name and password.
- **Sender e-mail** Enter the e-mail address to use as the return address for error reporting, password reminders, and scheduled report delivery.
- **Sender name** Enter the name to use for the sender.
- **Use SSL** Select this check box to enable the Secure Sockets Layer protocol.

3. When you have finished entering the values for your server, click **Save**.

Support Options

We provide a number of technical support options to help you succeed with our products.

Free Community Support

Our community web site is available free of charge. Community members and ActiveReports 8 Server team members provide answers to your most common questions.

In addition, we offer the following support options:

Annual Maintenance Package

Customers who purchase annual maintenance can benefit from the full suite of technical support services for one year. These benefits include:

1. **All major releases.**
2. **All service releases and minor upgrades.**
3. **Email support**
   For e-mail support, please submit your requests via the Helpdesk section on the community web site.
   Please include the build number, the environment you are using, and a detailed description of the problem.
   This will help speed up the support process.
4. **Phone support**

Support telephone number: (425) 952-6362
Support hours: Monday through Friday, 9:00 am-5:00 pm US Pacific Standard Time (PST)
When contacting support, please be prepared to provide your serial number, a complete description of the problem, as well as hardware and operating environment specifications.