Pilot Data Server Administrator Guide

Version 9.8



Copyright ©2026 Vizrt. All rights reserved.

No part of this software, documentation or publication may be reproduced, transcribed, stored in a retrieval system, translated into any language, computer language, or transmitted in any form or by any means, electronically, mechanically, magnetically, optically, chemically, photocopied, manually, or otherwise, without prior written permission from Vizrt. Vizrt specifically retains title to all Vizrt software. This software is supplied under a license agreement and may only be installed, used or copied in accordance to that agreement.

Disclaimer

Vizrt provides this publication “as is” without warranty of any kind, either expressed or implied. This publication may contain technical inaccuracies or typographical errors. While every precaution has been taken in the preparation of this document to ensure that it contains accurate and up-to-date information, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained in this document.

Vizrt’s policy is one of continual development, so the content of this document is periodically subject to be modified without notice. These changes will be incorporated in new editions of the publication. Vizrt may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time. Vizrt may have patents or pending patent applications covering subject matters in this document. The furnishing of this document does not give you any license to these patents.

Antivirus Considerations

Vizrt advises customers to use an AV solution that allows for custom exclusions and granular performance tuning to prevent unnecessary interference with our products. If interference is encountered:

- **Real-Time Scanning:** Keep it enabled, but exclude any performance-sensitive operations involving Vizrt-specific folders, files, and processes. For example:
  - C:\Program Files\[Product Name]
  - C:\ProgramData\[Product Name]
  - Any custom directory where [Product Name] stores data, and any specific process related to [Product Name].
- **Risk Acknowledgment:** Excluding certain folders/processes may improve performance, but also create an attack vector.
- **Scan Scheduling:** Run full system scans during off-peak hours.
- **False Positives:** If behavior-based detection flags a false positive, mark that executable as a trusted application.

Technical Support

For technical support and the latest news of upgrades, documentation, and related products, visit the Vizrt web site at www.vizrt.com.

Created on

2026/11/17

Contents

1 Introduction 6
1.1 Pilot Data Server Launcher 6
1.2 Pilot Data Server Service 6
1.3 Script Runner Service 6
1.4 Crop Service 7
1.5 Thumbnail Generator 7
1.6 Related Documents 7
1.7 Feedback and Suggestions 7
1.8 Support 7
2 Installation 8
2.1 Installation Prerequisites 8
2.2 Installing Pilot Data Server 8
2.2.1 ShowEdge Installer 8
2.2.2 Scripted Installation 9
2.3 Installing the Thumbnail Generator 10
2.3.1 Silent Installation 10
2.4 Alternative Installations 10
2.5 Firewall Exceptions 11
2.6 Antivirus 11
3 Default Ports and Protocols 12
3.1 HTTP 12
3.2 HTTPS 13
4 Configuration 15
4.1 Launcher 15
4.2 Status Panel 16
4.3 Log Panel 17
4.4 License Panel 18
4.5 Authentication 19
4.5.1 Configuring Pilot Data Server 19
4.5.2 Roles 20
4.5.3 Identity Configuration 21
4.5.4 Newsroom Requirements 28
4.6 Script Runner 29

4.6.1 Configuring Script Runner 29
4.7 Licensing 31
4.7.1 Setting Up Licenses 31
4.7.2 Configuring Alerts 33
4.8 Pilot Data Server Settings 35
4.8.1 The Settings File 35
4.8.2 The Settings UI 36
4.8.3 Configuring the Pilot Data Server HTTP Port 39
4.8.4 STOMP over TCP 40
4.8.5 STOMP over WebSocket 41
4.9 Log Files 42
4.9.1 Logging Configuration 42
4.10 Crop Service 44
4.10.1 Configuration via Launcher UI 44
4.10.2 Pilot Database Configuration 48
4.10.3 Security 48
4.10.4 Default INI Values 49
4.10.5 Operational Notes 50
4.10.6 Health endpoint 50
4.11 Pilot Update Service 53
4.11.1 Configuring Update Service 53
4.11.2 Configuring Certificate Authorities (CAs) 54
4.11.3 Logging 56
4.12 Audit Logging 57
4.12.1 What is Audited 57
4.12.2 What Information is Recorded 57
4.12.3 How Audit Data is Stored 57
4.12.4 Viewing Audit Data 57
4.12.5 Configuration 58
4.12.6 Data Protection 58
5 Web Interface 59
5.1 Accessing the Pilot Data Server Web Interface 59
5.2 Service 59
5.3 Template Feed 60
5.4 Data Element Feed 60
5.5 Settings 60
5.6 Examples 60

5.7 Documentation 60
6 RestVOS 61
6.1 Comparing RestVOS and Conventional VOS Access 61
6.2 Advantages of RestVOS 61
7 Change Notifications 62
7.1 REST API 62
7.2 STOMP API 62
8 Proxy Support 63
8.1 Using X-Forwarded Headers 63
8.2 Configure Base URL 63
8.3 Redundancy and Failover 64
8.3.1 Load Balancer Quick Setup 64
9 MOS XML 66
9.1 Graphics with No Timing Info 66
9.2 Graphics with Timing Info 66
9.3 Videos 67
9.4 Stillstore Images 67
9.5 Overriding MOS XML Values 67
10 Data Server Config 68
10.1 Panels 68
10.2 Parameters panel 68
10.2.1 Most important settings for a Pilot Edge system 69
10.3 Search Providers panel 70
10.4 Tags panel 70
10.5 Asset Guests panel 70
10.6 Pilot Edge panel 71
10.7 Database Parameters 73
10.7.1 Settings used by Pilot Edge and Template Builder 73
10.7.2 Newsroom and system settings 75
10.7.3 Legacy settings (not used by Pilot Edge or Template Builder) 76
10.7.4 Settings managed internally by Pilot Data Server 80
10.8 Optional Parameters 82
10.8.1 env- parameters as environment variables 82

1 Introduction

Pilot Data Server is installed as an application layer on top of the Viz Pilot database, and acts as an application server for accessing Viz Pilot's database and other services, and enables the use of features including Crop Service, Template Tagging, Update Service, Person Search, search on Viz One, Thumbnail Generator and Timeline Editor in Viz Pilot workflows.

It has a web interface with a REST API, accessible for 3rd party integrations. Pilot Data Server is required when using Graphic Hub as the Viz Pilot database, and for integrating with Viz Story, Viz Pilot Edge and Template Builder.

The full Pilot Data Server system contains the following parts:

Type	Component
Application	Pilot Data Server Launcher
Services	Vizrt Pilot Data Server Vizrt Script Runner Vizrt Crop Service Vizrt Thumbnail Generator

1.1 Pilot Data Server Launcher

The Pilot Data Server Launcher is a separate application that can be used to configure, start and stop the Pilot Data Server service.

1.2 Pilot Data Server Service

The Pilot Data Server service is the main module of this system. It provides a REST API for web applications to access the Pilot database, and it has a changelog event mechanism for broadcasting changes to templates, elements and concepts inside the database.

1.3 Script Runner Service

The Script Runner Service comes packaged with Pilot Data Server and is automatically installed. It can be configured, started and stopped with the Pilot Data Server Launcher. The Script Runner Service provides a simple way for users to use the Media Sequencer's *Live Update* mechanism without needing to create a custom service. The Script Runner acts as an External Update Service for Media Sequencer. See Update Script Editor in the Viz Pilot User Guide.

Copyright © 2026 Vizrt

1.4 Crop Service

The Crop Service provides a way to crop images. The Crop Service is provided in a separate installer. It can be installed and run on the same physical host as Pilot Data Server. If installed on the same host, it can be configured, started and stopped through the Pilot Data Server Launcher.

1.5 Thumbnail Generator

Thumbnail Generator is a Windows service that automatically generates thumbnails of data elements when they are created or changed. The thumbnails are stored in the Viz Pilot database. Thumbnail Generator requires a Preview Server, which is used to serve the thumbnails in the data element feed.

1.6 Related Documents

The Viz Pilot User Guide provides complete documentation of the Viz Pilot system. For more information and documentation for all Vizrt products, please visit the links available under Support.

1.7 Feedback and Suggestions

We encourage suggestions and feedback about our products and documentation. To give feedback and/or suggestions, please contact your local Vizrt customer support team at www.vizrt.com.

1.8 Support

Support is available at the Vizrt Support Portal.

For more information about all Vizrt products, visit:

- www.vizrt.com
- Vizrt Documentation Center
- Vizrt Training Center
- Vizrt Forum

Copyright © 2026 Vizrt

2 Installation

This section covers:

- Installation Prerequisites
- Installing Pilot Data Server
    - ShowEdge Installer
    - Scripted Installation
- Installing the Thumbnail Generator
    - Silent Installation
- Alternative Installations
- Firewall Exceptions
- Antivirus

2.1 Installation Prerequisites

- Visual C++ Redistributable Packages for Visual Studio 2015

The packages are included in the Pilot Data Server bundle installer, and can also be found on Microsoft's website.

2.2 Installing Pilot Data Server

There are two options to install Pilot Data Server:

- Use the bundle installer: *PilotDataServerBundleInstaller-x64-[version].exe*, or
- The MSI installer: *PilotDataServerInstaller-x64-[version].msi*.

Installer	Content
Bundle installer	Pilot Data Server Visual Studio C++ 2015 redist Codemeter runtime
MSI	Pilot Data Server Thumbnail Generator Service (optional)

To install with a UI, double click one of these installers and follow the on-screen instructions.

> **Note:** The default location is *%ProgramFiles%\Vizrt\Pilot Data Server*.

2.2.1 ShowEdge Installer

The ShowEdge installer is not bundled into the Pilot Data Server installer, nor into the Viz Pilot Edge installer, it has been available from the Pilot Data Server end point `/showedge`. To still be able to serve out the ShowEdge

Copyright © 2026 Vizrt

installer from this end point, the file `ShowEdge.[version].exe` needs to be downloaded from the FTP, and copied to the `app/showedgeinstaller` folder of the Pilot Data Server installation (for example, to `C:\Program Files\Vizrt\Pilot Data Server\app\showedgeinstaller`).

2.2.2 Scripted Installation

For a scripted installation, Pilot Data Server can be installed without using the installer GUI. This can be done with the MSI installer found on the Vizrt's FTP, or extracted from the bundle installer with the *-msi* command line option.

> **Note:** For silent install, the step for configuring the license is skipped. The license configuration of Pilot Data Server is stored at *%ProgramData%\Vizrt\Pilot Data Server\licenses.json*, and the file itself can be copied from one installation to another as a way to apply known license configurations without using the GUI.

In a command shell, run the following command:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi /quiet
```

If required to set properties for the silent installer for the base url, use the following MSI properties:

Property	Example value
BASE_URL	https://myserver.com:443/pds
BASE_PATH	pds
SECURE_PORT	7373
CONNECTION_METHOD	GH or ORACLE
ORACLE_USERNAME	pilot
ORACLE_PASSWORD	pilot
ORACLE_DATA_SOURCE	localhost/vizrtdb
GH_HOST	localhost
GH_PORT	19398
GH_USER_ID	Admin
GH_PASSWORD	VizrtDb123

Copyright © 2026 Vizrt

Property	Example value
INSTALLDIR	"c:\\Program Files\\Vizrt\\Pilot Data Server"

Examples

```
msiexec /i <application.msi> CONNECTION_METHOD="ORACLE" ORACLE_USERNAME="pilot"
 ORACLE_PASSWORD="pilot" ORACLE_DATA_SOURCE="hostname/vizrtdb" BASE_PATH="pds" /log
install.log
msiexec /i <application.msi> CONNECTION_METHOD="GH" GH_HOST="hostname"
 GH_PORT="19398" GH_USER_ID="admin" GH_PASSWORD="some password123" BASE_URL="https://
something.com/pds" SECURE_PORT="7373" /log install.log
```

2.3 Installing the Thumbnail Generator

The Thumbnail Generator (see the Viz Pilot User Guide) can be installed as an optional component during the Pilot Data Server installation. The option for Thumbnail Generator installation is disabled by default. To install it, select the Thumbnail Generator manually when installing Pilot Data Server through the MSI installer.

2.3.1 Silent Installation

Since the Thumbnail Generator is not installed by default, the *INSTALLLEVEL=2* needs to be added to the command string for the Thumbnail Generator to be included during a silent Pilot Data Server installation:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi INSTALLLEVEL=2 /quiet
```

To upgrade an existing Pilot Data Server to include the Thumbnail Generator without uninstalling it, the following command needs to be run:

```
msiexec /i VizrtPilotDataServer-x64-x.x.x.x-Release.msi
ADDLOCAL=ThumbnailGeneratorService /quiet
```

2.4 Alternative Installations

- If **Crop Service** installation is required (see the relevant section in the Installation chapter in the Viz Pilot User Guide), it must be installed separately, after Pilot Data Server.
- If support for **multiple Viz Pilot Database schemas** is required, *separate* Pilot Data Servers must be installed on *separate* machines.

Copyright © 2026 Vizrt

2.5 Firewall Exceptions

For Pilot Data Server to be accessible from other machines, Windows firewall exceptions are required. Pilot Data Server adds the following inbound rules to the Windows firewall during installation:

Allow connections on port:

- 8177 - Pilot Data Server HTTP port
- 7373 - Pilot Data Server HTTPS port
- 9876 - STOMP event notifications port
- 1981 - Script Runner HTTP port
- 1982 - for Pilot Script Runner HTTPS port

These ports are the recommended ports for the service. They are removed when Pilot Data Server is uninstalled.

2.6 Antivirus

Vizrt does not recommend or test antivirus systems in combination with Vizrt products, as the use of such systems can potentially lead to performance losses. The decision for the use of antivirus software and thus the risk of impairments of the system is solely at the customer's own risk.

There are general best-practice solutions, these include setting the antivirus software to not scan the systems during operating hours and that the Vizrt components, as well as drives on which clips and data are stored, are excluded from their scans (as previously stated, these measures cannot be guaranteed).

A list of services and folders which should be excluded from scanning can be found below:

Services:

- vizrt_crop_server
- VizrtMist
- vizrt_pilot_data_server
- vizrt_script_runner
- vizrt_thumbnail_generator

Folder:

- *C:\Program Files (x86)\Vizrt*
- *C:\ProgramData\vizrt*
- *C:\Program Files\Vizrt*

Processes:

- PilotCropServiceHost.exe
- PilotDataServerHostService.exe
- PilotScriptRunnerHostService.exe
- PilotThumbnailGeneratorHostService.exe
- caddy-service.exe
- caddy.exe

Copyright © 2026 Vizrt

3 Default Ports and Protocols

The table below shows all of the default Protocols and Ports used with Pilot Data Server.

3.1 HTTP

Name	Default Value	Comments
Pilot Data Server	http://pds-host:8177	The port used for HTTP is configurable in the *DataServerConfig.ini* file on *%ProgramData%\Vizrt\Pilot Data Server*.
Script Runner	http://pds-host:1981	Installed with Pilot Data Server and runs on the same host machine. The port used for HTTP is configurable in the file *PilotScriptRunnerHostService.exe.config* in the Pilot Data Server installation folder.
Crop Service	http://pds-host:8178	Separate installer that normally runs on the same host as Pilot Data Server. The port used for HTTP is configurable in the file *PilotCropServiceHost.exe.config* in the Crop Service installation folder.
Graphic Hub REST	http://gh-host:19398	Graphic Hub uses the same port for HTTP and HTTPS.
Viz One	http://vizone-host:80/thirdparty	
Preview Server	http://ps-host:21098	
REST VOS / GH Image Extension	http://imex-host:19390/restvos/service	The image extension UI can be accessed on *http://imex-host:19390/imagelibrary/index.html*.
Viz Pilot Preview Port	50008	Viz Engine preview port used for snapshot previews in Viz Pilot.

Copyright © 2026 Vizrt

Name	Default Value	Comments
Viz Arc MOS Panel	http://arc-host:9004/mos-plugin	
PDS changelog notifications	stomp://pds-host:9876/?destination=/changelog&lastKnownId=-1	

3.2 HTTPS

Name	Default Value	Comments
Pilot Data Server	https://pds-host:7373	The port used for HTTPS is configurable from the Pilot Data Server Launcher. The service needs to restart.
Script Runner	https://pds-host:1982	The port used for HTTPS is configurable from the Pilot Data Server Launcher. The service needs to restart.
Crop Service	https://pds-host:8179	The port used for HTTPS is configurable from the Pilot Data Server Launcher. The service needs to restart.
Graphic Hub REST	https://gh-host:19398	Graphic Hub uses the same port for HTTP and HTTPS.
Viz One	https://vizone-host.example.com:443/thirdparty	
Preview Server	https://ps-host:4443	
REST VOS / GH Image Extension		No support for HTTPS.
Viz Pilot Preview Port	50008	Viz Engine preview port used for snapshot previews in Viz Pilot.

Copyright © 2026 Vizrt

Name	Default Value	Comments
Viz Arc MOS panel	https://arc-host:9005/mos-plugin	
PDS changelog notifications	stomp://pds-host:9876/?destination=/changelog&lastKnownId=-1	

Copyright © 2026 Vizrt

4 Configuration

This section covers the following topics:

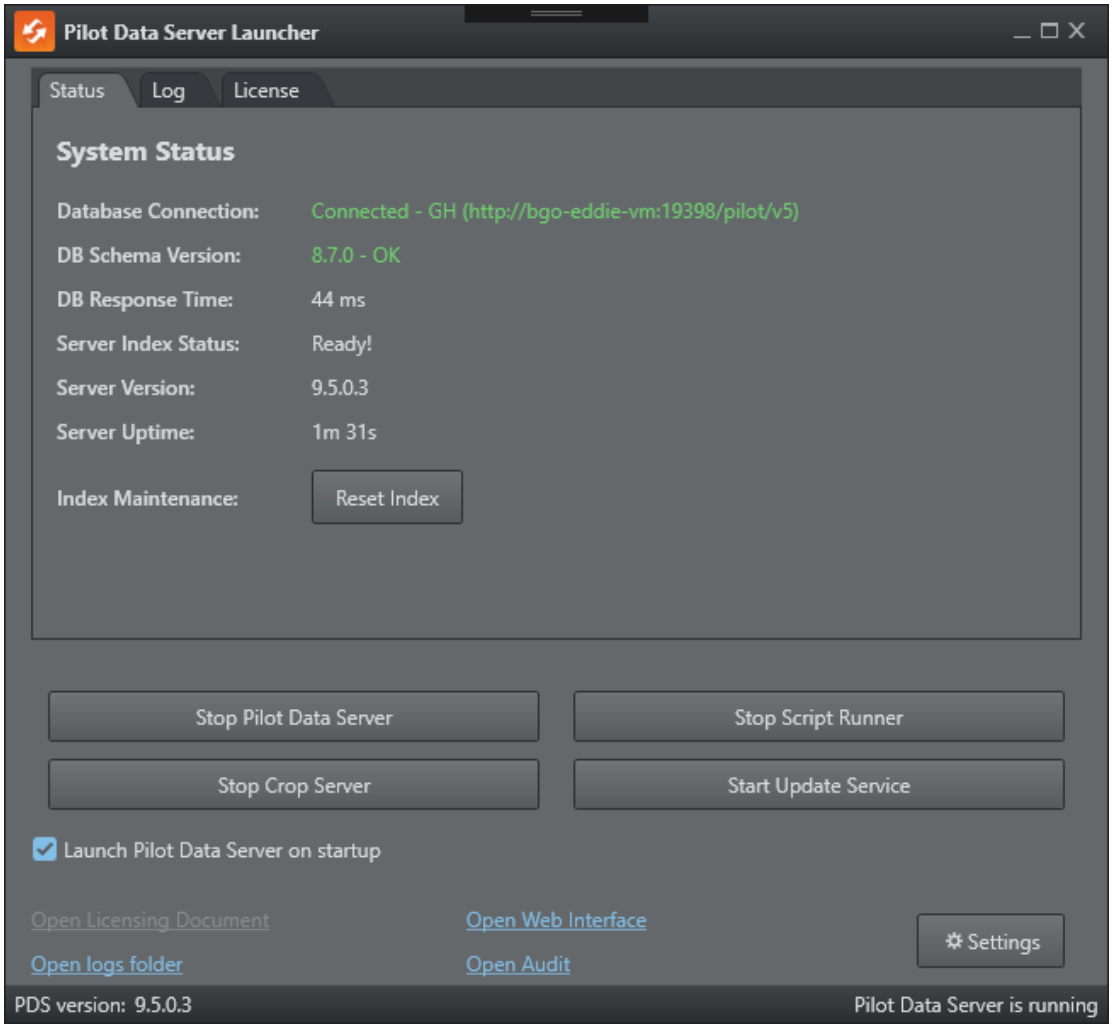
- Launcher
- Status Panel
- Log Panel
- License Panel

4.1 Launcher

The Pilot Data Server Launcher is a minimal dashboard to start and stop services, and offers a UI to configure the system.

Copyright © 2026 Vizrt

4.2 Status Panel



The Status panel provides an overview of the current Pilot Data Server state. The information shown is not taken from a single authoritative status source. Instead, it is derived from two runtime signals: the launcher continuously monitors the **server log** to infer index-building progress, and it periodically queries the **/service/health** endpoint to retrieve health information such as **database connectivity**, **schema compatibility**, **server version**, and **uptime**. Because these values are gathered asynchronously and from different sources, the panel should be understood as an operational summary for troubleshooting and monitoring, not as a strict real-time guarantee of internal service state.

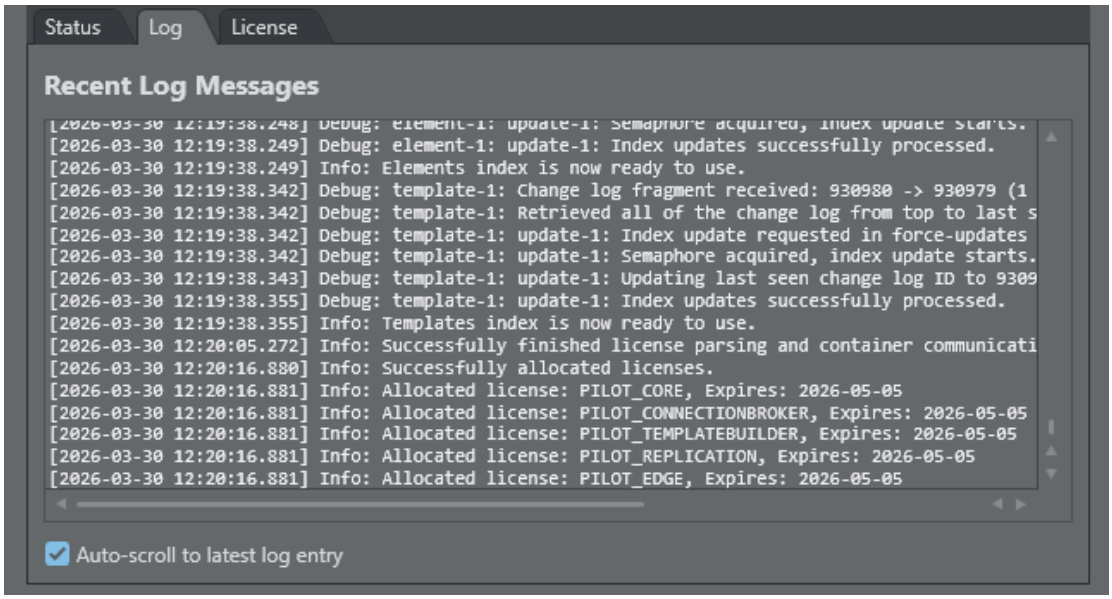
In addition, the following functions are available:

- **Settings:** Opens the settings window, with the main configuration topics:
  - Authentication
  - Script Runner
  - Licensing
  - Pilot Data Server Settings

Copyright © 2026 Vizrt

    - Log Files
    - Crop Service
    - Pilot Update Service
    - Audit Logging
- **Start/Stop:** Start and stop services.
- **Launch Pilot Data Server on startup:** Guarantees the Pilot Data Service and additional services are started when the computer starts.
- **Open Licensing Document:** Opens the local documentation to configure the licenses for Pilot Data Server.
- **Open logs folder:** Opens the logs and ini file folder, normally *%ProgramData%\Vizrt\Pilot Data Server*.
- **Open Web Interface:** Opens the web page of the currently running Pilot Data Server.
- **Open Audit:** Opens the Audit window.

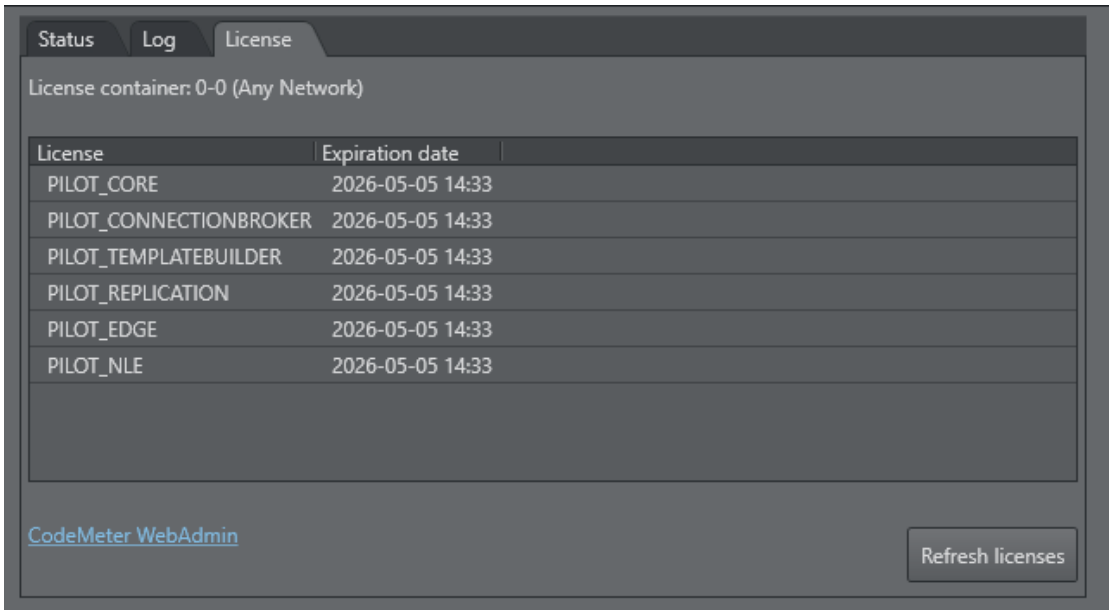
4.3 Log Panel



The Log panel displays a live view of recent Pilot Data Server log messages, to support troubleshooting and operational monitoring. It reads from the local server log file and shows the latest entries in chronological order, including timestamp, log level and message text. The panel is intended as a convenience view inside the launcher and only shows a limited rolling window of recent messages, not the complete log history. For a full investigation, use the Open logs folder link to inspect the original log files.

Copyright © 2026 Vizrt

4.4 License Panel



- **Currently activated licenses:** List of the currently activated licenses on the license server/container.
- **CodeMeter WebAdmin:** Opens Wibu license administration in a browser, and requires the Codemeter service to run.
- **Refresh licenses:** Reconnects to the license system and does another attempt to fetch the current licenses.

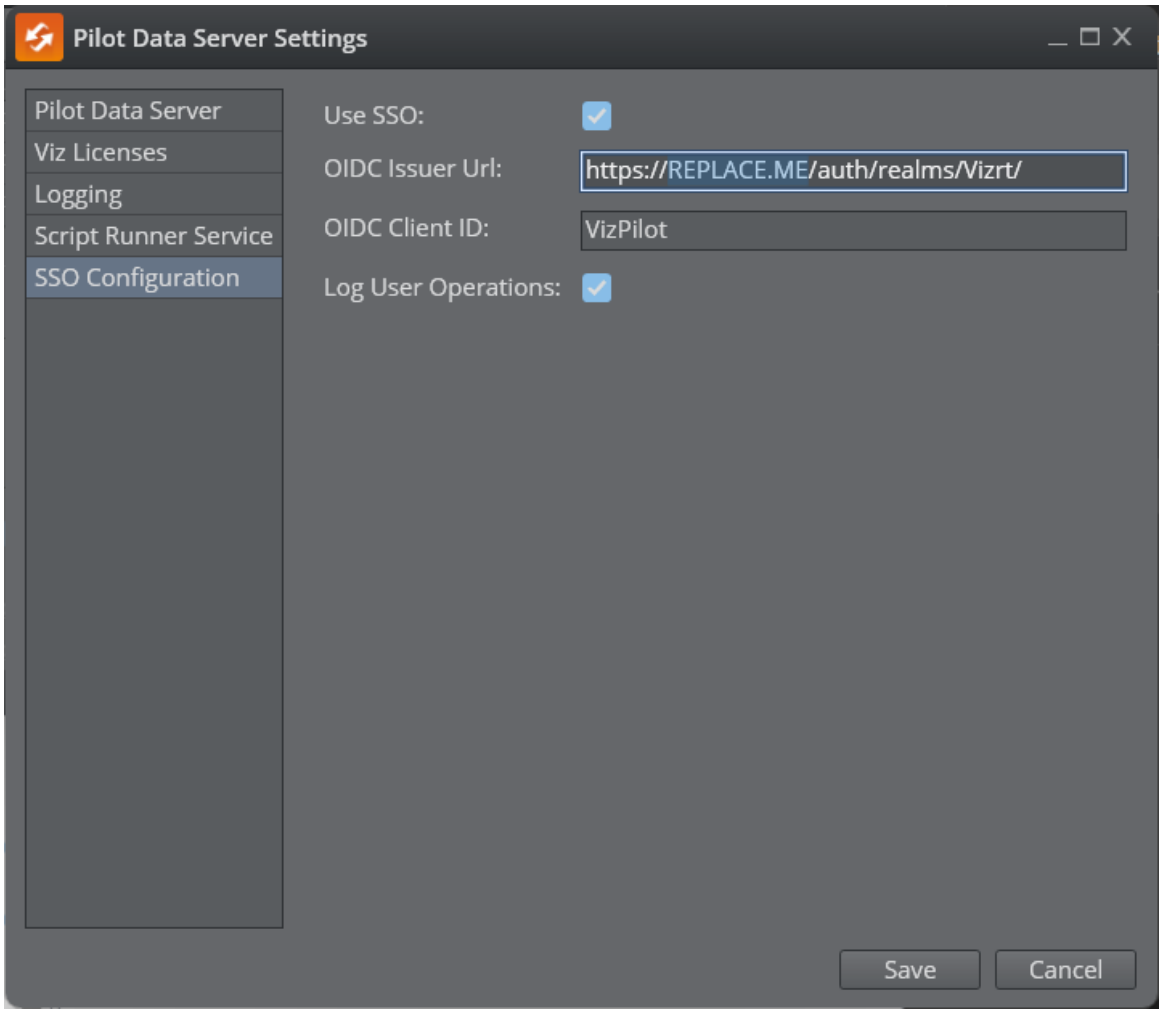
Note: The Codemeter.exe service of the Wibu Licensing system, must be running in order to start Pilot Data Server.

Copyright © 2026 Vizrt

4.5 Authentication

- Configuring Pilot Data Server
- Roles
- Identity Configuration
    - Microsoft Entra ID
    - Vizrt SSO (Keycloak)
- Newsroom Requirements
    - Known Limitations

4.5.1 Configuring Pilot Data Server



- **Use SSO:** Decides whether the system enables authentication or not. It requires TLS/SSL enabled, and the full system needs to communicate over HTTPS.
- **OIDC Issuer URL:** URL to the OIDC compliant identity provider. See below.

Copyright © 2026 Vizrt

- **OIDC Client ID:** Public identifier for the applications. See below for a detailed explanation.
- **Log User Operations:** If enabled, user operations that add, delete or modify templates and data elements, are added to the regular log. The user ID and name or email (depending on the identity provider) is added to the log. For instance, a user modifies a data element:

```
2024-11-06 11:36:23.189 [21] INFO: john (3709b32c-032a-4b78-af83-c82b65c90868)
modified resource at https://host-pilot:7373/dataelements/168 (PUT)
```

4.5.2 Roles

Pilot Data Server can be configured to use Authentication to control access to Viz Pilot Edge, Template Builder and DataServerConfig.

There are 4 different roles defined:

- pilot-journalist
- pilot-editor
- graphic-designer
- pilot-admin

It is important to understand the distinction between roles and groups. Roles are what grants a user privileges, and a role can be assigned to a user directly, or to all members of one or more groups depending on the identity provider being used. The role names are fixed and defined by Vizrt, but you are free to define or reuse groups as you like.

These are the access rights for the different roles:

	Viz Pilot Edge	Template Builder	Data Server Settings
pilot-journalist	✓	✗	✗
pilot-editor	✓	✗	✗
graphic-designer	✓	✓	✗
pilot-admin	✓	✓	✓

In addition, a user's profile or group can determine which Pilot Collections they can access, based on role-to-collection name matching. Roles prefixed with **pilot-collection-** are used to match authenticated users to specific collections. For example:

- A user with the roles “pilot-journalist” and “pilot-collection-news”, only has access to the Pilot Collection named **news.**
- Users can have multiple roles, allowing them to access multiple Pilot Collections.

See the section **Pilot Collections** in the Template Builder user manual.

Copyright © 2026 Vizrt

4.5.3 Identity Configuration

Pilot Data Server uses the OpenID Connect Protocol (OIDC) to authenticate users. It can, in theory, be configured to integrate with any OIDC compliant identity provider.

The following providers have been tested and are supported:

- Microsoft Entra ID (formerly Azure Active Directory)
- Vizrt SSO (Keycloak)

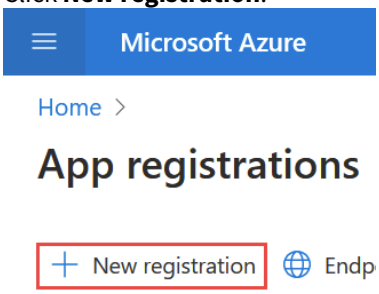
Microsoft Entra ID

The Entra ID application configuration consists of two parts: an App Registration for the technical details of the application, and an Enterprise Application that defines user and group access and their role assignments, as well as other aspects of how the app integrates with your tenant.

Warning: Use the new Microsoft Graph app manifest format. The enclosed configuration file *entra-id-pilot-sso.json* is updated to this format.

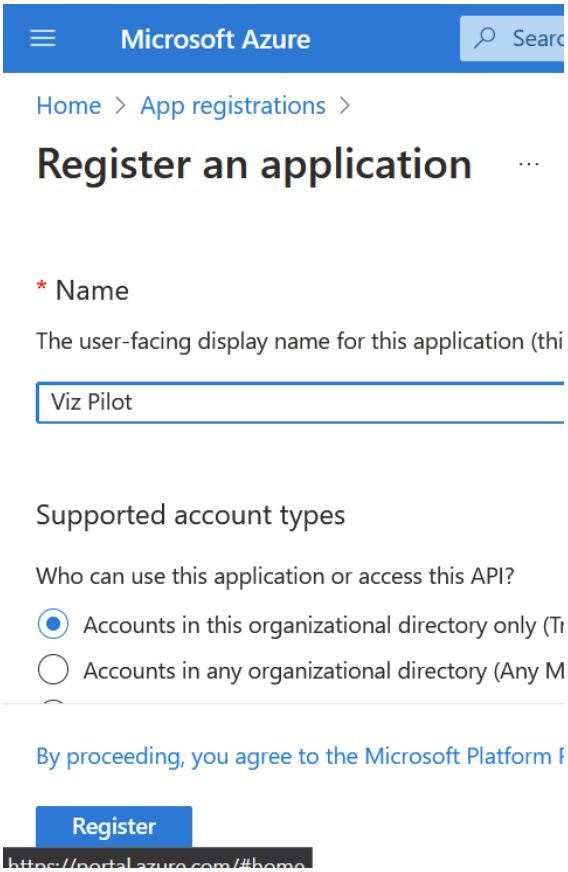
To connect Viz Pilot to Microsoft Azure Entra ID (formerly Azure AD), log in to your tenant in the Azure Portal, then perform the following steps:

1. Go to the App Registrations blade.
2. Click **New registration**:



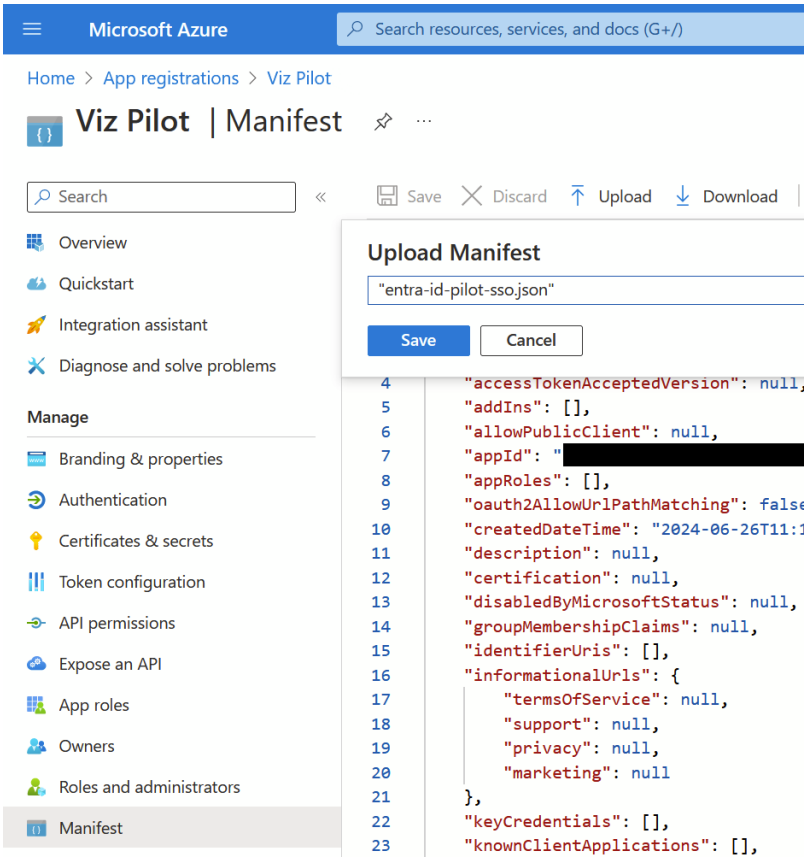
3. Type a name, for example Viz Pilot, and leave the other settings as is. Then click **Register**.

Copyright © 2026 Vizrt

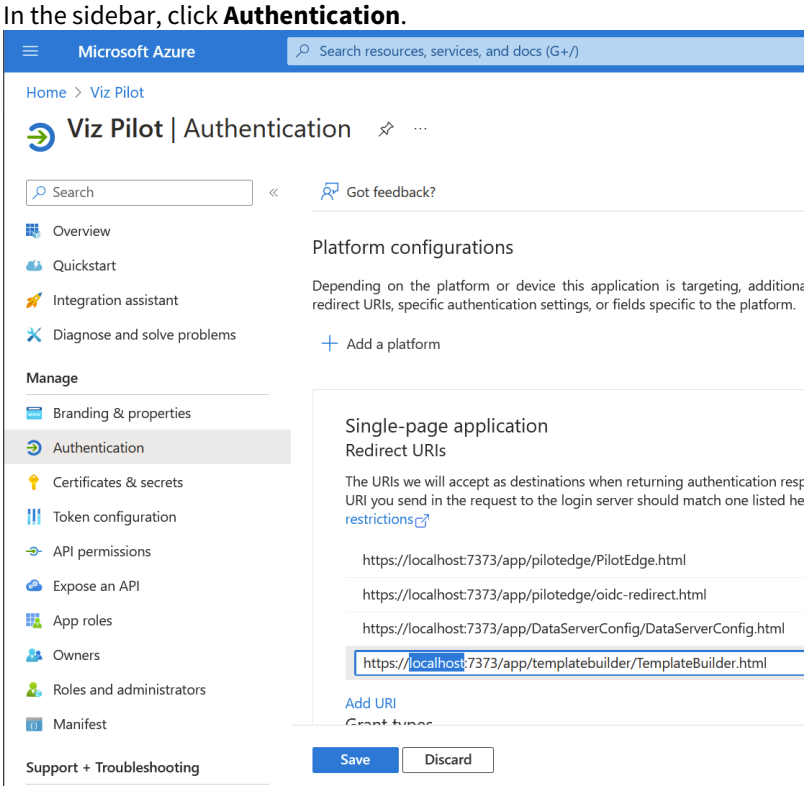


4. In the sidebar, click **Manifest**.

Copyright © 2026 Vizrt

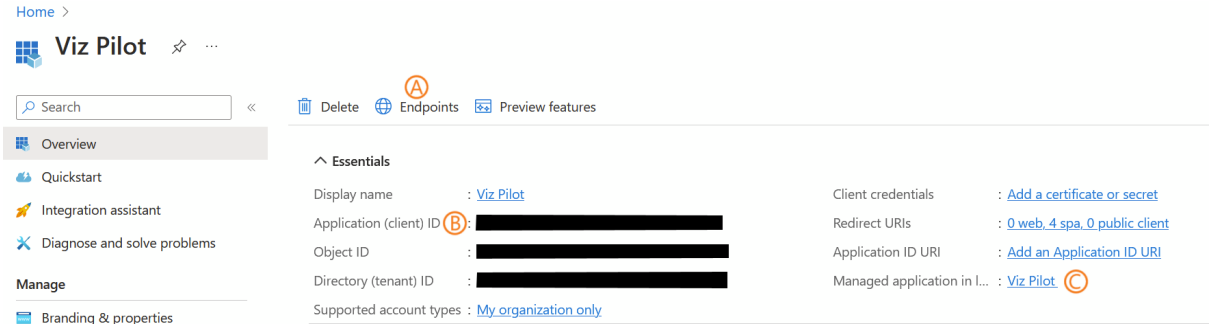


5. Click **Upload**, select the provided JSON file, then click **Save**.
6. In the sidebar, click **Authentication**.

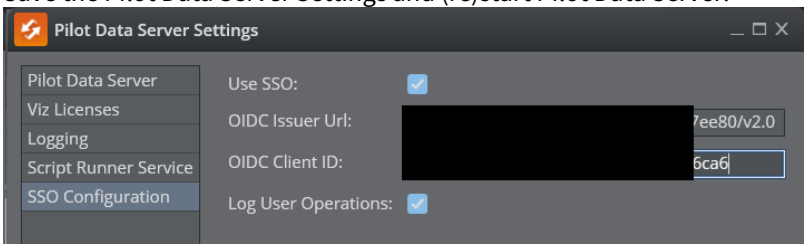


Copyright © 2026 Vizrt

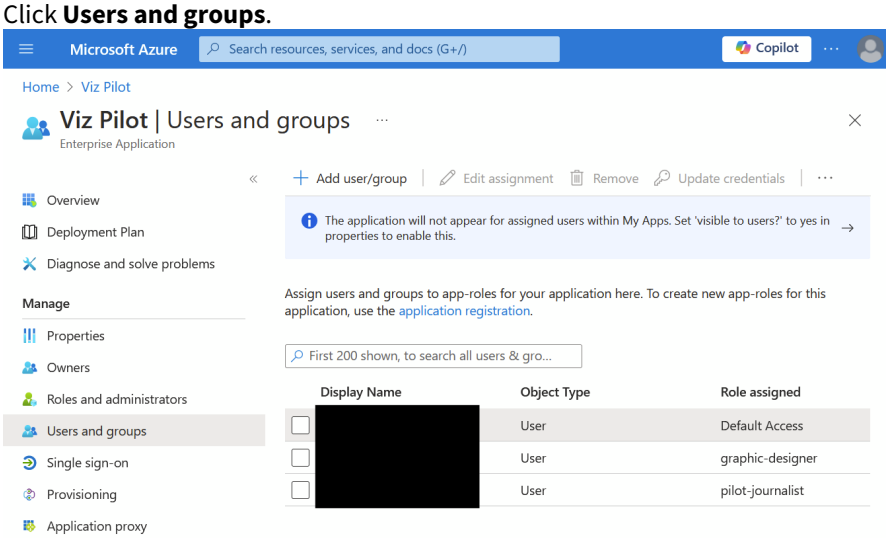
7. Edit the URLs to have the correct hostname, and click **Save**.
   Optional:
   a. In the sidebar, click **API permissions**, then click **Grant Admin Consent for <your tenant name>**.
   b. In the sidebar, click **Branding & properties**, then upload the provided logo.
   c. In the sidebar, click **Owners**, then assign yourself and/or other users that should have access to manage this app registration.
8. In the sidebar, click **Overview**:



9. Open the Pilot Data Server Launcher, click **Settings** and select the SSO page (or use the notepad to collect the following values for a later entry).
10. Click **Endpoints** (A) and find the **OpenID Connect metadata document** field. Copy the left part up (but not including */.well-known/openid-configuration)*, and paste it into the **OIDC Issuer Url** field in Pilot Data Server Settings.
11. Find the **Application (client) ID** (B) value and copy it to the **OIDC Client ID** field in Pilot Data Server Settings.
12. Save the Pilot Data Server Settings and (re)start Pilot Data Server.



13. Click the link with the label **Managed application in local directory** (C), which takes you to the Enterprise Application blade.
14. Click **Users and groups**.



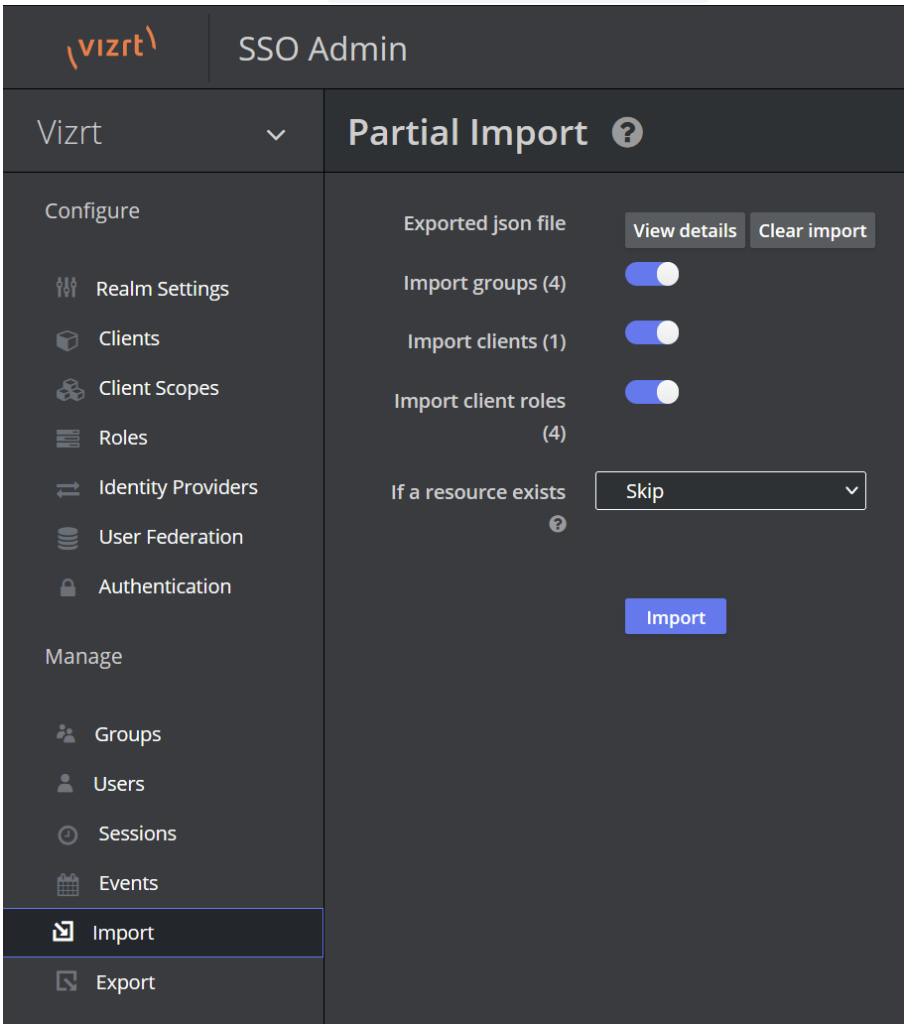
Copyright © 2026 Vizrt

15. Add users and groups to roles as you wish (**Default Access** is added automatically but does not grant access to any Viz Pilot apps, so you may remove it).
16. You may also inspect the other pages in the sidebar to customize behavior and security, and assign other users that should have access to manage this Enterprise Application, as owners . You do not need to perform Provisioning for Viz Pilot.

Vizrt SSO (Keycloak)

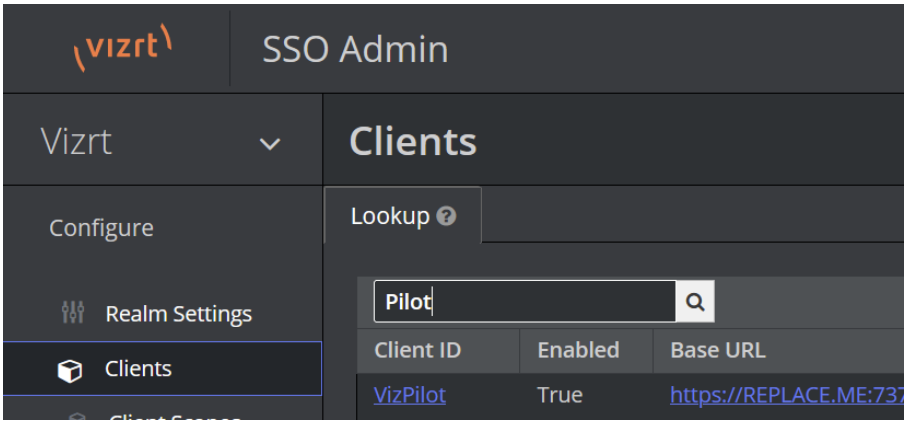
To connect Viz Pilot to Vizrt SSO running in an existing Viz One deployment, perform the following steps:

1. Download the `keycloak-pilot-sso.json` file from the same folder that you got the installer on Vizrt's FTP server.
2. Log in to the Viz One SSO Admin pages, and go to the **Import** tab in the left menu.
3. Click **Select file** and pick the `keycloak-pilot-sso.json` file.

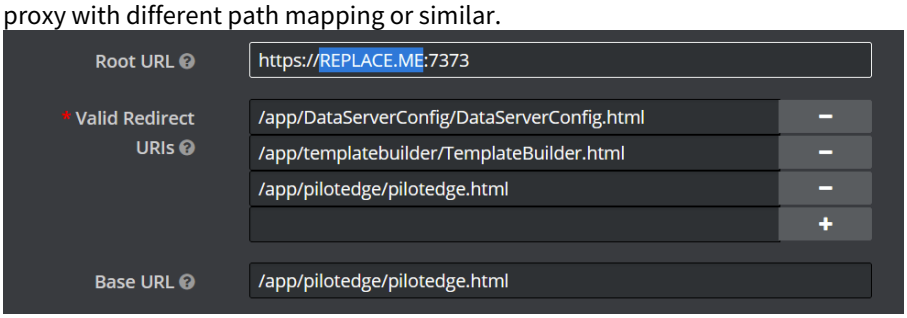


4. Review the options and click **Import** when ready.
5. Go to the **Clients** tab, find and open the **VizPilot** client that was just imported.

Copyright © 2026 Vizrt

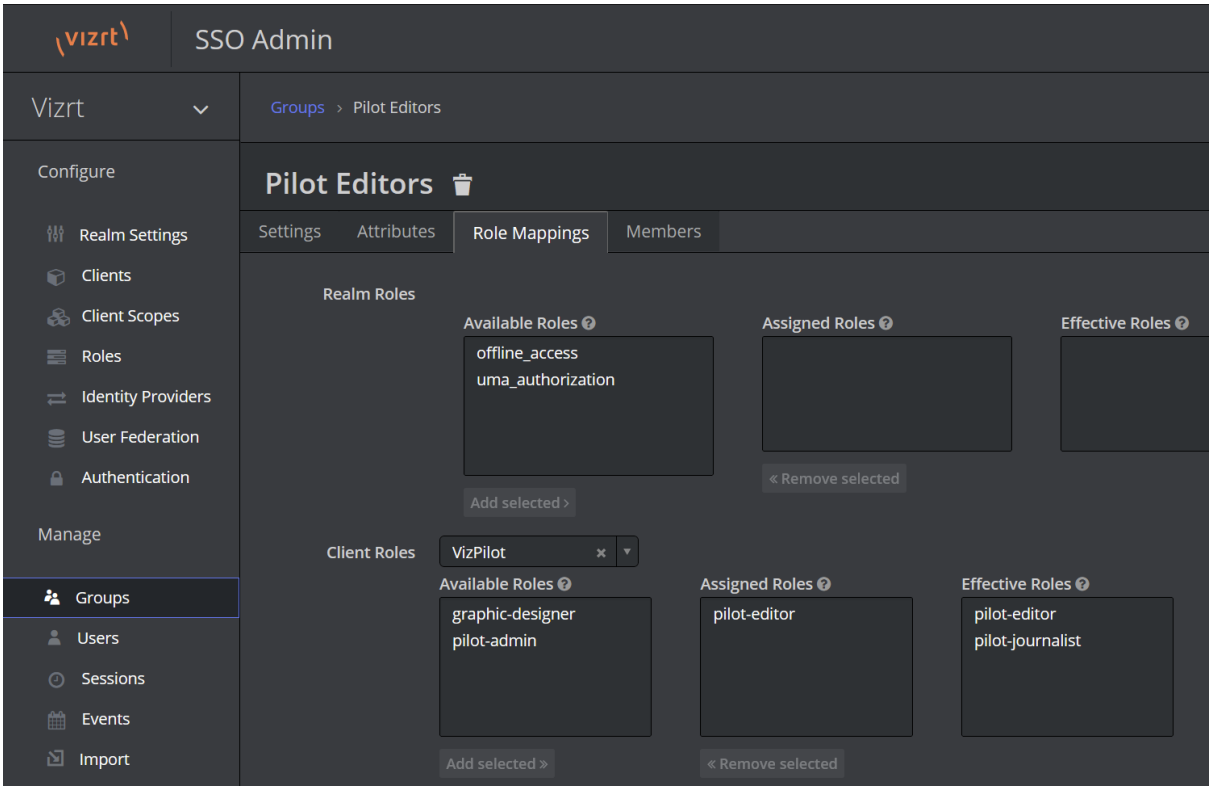


6. Update the Root URL field with the fully-qualified domain name of the Viz Pilot server. Also check and adjust the Valid Redirect URIs in case you have installed multiple versions of the web apps, or are using a reverse proxy with different path mapping or similar.



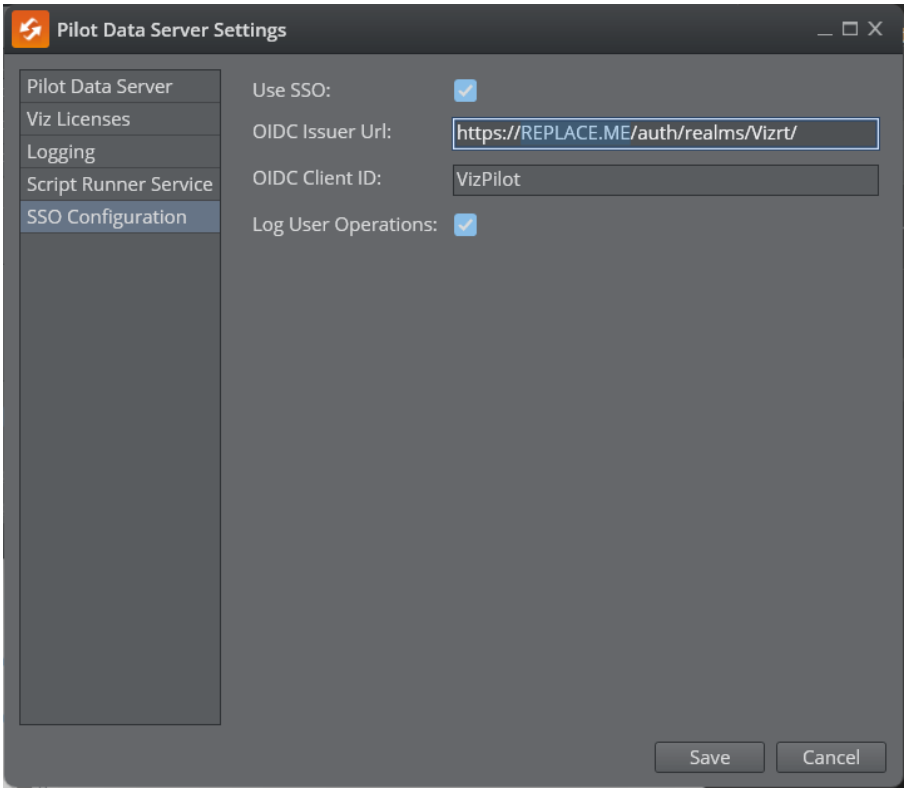
7. Manage users and groups to assign the four Viz Pilot roles as desired. The roles are defined by the **VizPilot** client, as shown in the screenshot below. The imported config includes example groups which you can use as-is, customize, or replace with your own groups. Remember that these groups don't directly grant any privileges, they just assign the corresponding roles to all group members, and it is the roles that grant a user privileges. The **graphic-designer** and **pilot-editor** roles also implicitly grant the **pilot-journalist** role. The **pilot-admin** role implicitly grants all the other Pilot roles.

Copyright © 2026 Vizrt



8. Use the Pilot Data Server Launcher to configure OIDC settings for the target server. The **Issuer Url** is typically the Viz One server's base url followed by `/auth/realms/Vizrt`. The Client ID should be **VizPilot** when using the imported config.

Copyright © 2026 Vizrt



4.5.4 Newsroom Requirements

The Newsroom system hosts MOS plugins like Viz Pilot Edge in their own web browser environment. This environment must be considered secure by the authentication technology we are using in Viz Pilot Edge.

The following requirements must be met:

1. The newsroom web hosting environment for MOS plugins must communicate over secure socket (using the **HTTPS protocol**). This must be configured in the newsroom system.
2. The web browser hosting the MOS plugin guest must support **popup windows** to allow the login dialog to be displayed.
3. The web browser must support the **SubtleCrypto** web API as specified here: https://developer.mozilla.org/en-US/docs/Web/API/SubtleCrypto. All modern browsers support this API, but it may be that the newsroom system still deploys an older browser internally, which can be incompatible with modern authentication technology.

Known Limitations

The web browser in **Avid iNEWS** does not support popup windows. To use authentication in iNEWS, the ShowEdge wrapper must be used to host Viz Pilot Edge.

Copyright © 2026 Vizrt

4.6 Script Runner

- Configuring Script Runner
  - Use Secure Communication
    - Server Certificates and PFX Files
  - Configure the HTTP Port

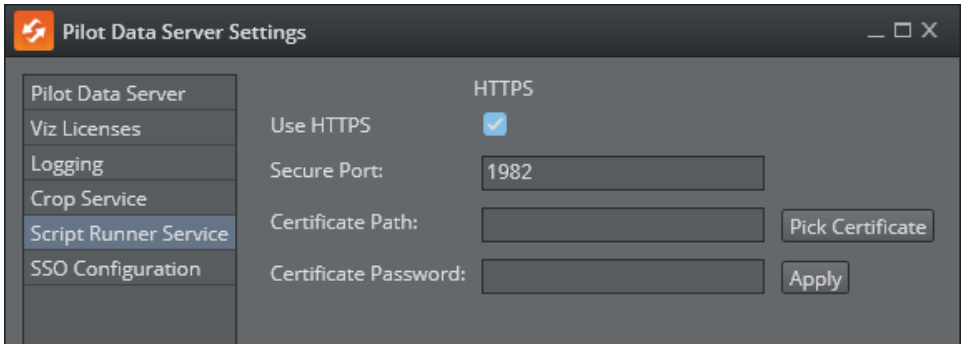
4.6.1 Configuring Script Runner

Script Runner is Viz Pilot's update service that lets you update template data, such as stock values and player statistics, right before going on air.

> **Note:** You can write Visual Basic In Pilot scripts to modify the data, and the scripts run on the Script Runner. It is also possible to create your own external update service. For more information, see *External Update Service* in the Media Sequencer Manual bundled with the software.

Use Secure Communication

Script Runner can be configured to use the SSL transport layer when communicating. Click **Settings** > **Script Runner Service** to enter the configuration UI.



To configure Script Runner for SSL communication:

1. Check **Use HTTPS**.
2. The default port suggested is 1982. This can be changed, but it is recommended to use this port.
3. Configure a certificate by either entering a path to the *.pfx file or picking a certificate from the file system.
4. Enter the certificate password.
5. Click **Apply** to register the certificate information.
6. Click **Save**.
7. Restart the Script Runner Service.

Server Certificates and PFX Files

A server certificate, typically issued by a trusted Certificate Authority (CA), contains the server's public key and identifying information. Adding a certificate for Script Runner or Crop Service is done by importing a PFX file . PFX

Copyright © 2026 Vizrt

(Personal Information Exchange) file, also known as PKCS#12. A PFX file consolidates the server certificate, its private key, and any intermediate and root certificates into a single, secure, password-protected file.

> **Note:** Viz Pilot and Media Sequencer with an Oracle backend, do not support Script Runner on HTTPS.

Configure the HTTP Port

The Script Runner uses port 1981 by default for HTTP. To change the port:

1. On the Pilot Data Server machine, open the configuration file in a text editor: *%ProgramFiles%\vizrt\Data Server\PilotScriptRunnerHostService.exe.config.*
2. Change the port by modifying the following line:

```
<setting name="DefaultPort" serializeAs="String">
    <value>1981</value>
</setting>
```

3. Save the file.
4. Restart the **Vizrt Script Runner** service from the Windows Services window.

Copyright © 2026 Vizrt

4.7 Licensing

Licenses in Pilot Data Server are managed by using WIBU Systems. This manages the use of PDS, Viz Pilot Edge, Template Builder and Viz Pilot Edge through the NLE plugins.

WIBU works through the CodeMeter application, therefore, it is required to be installed in the same machine as Pilot Data Server (it is not necessary to have CodeMeter installed in the same machine as Viz Pilot Edge, Template Builder or the graphic plugins).

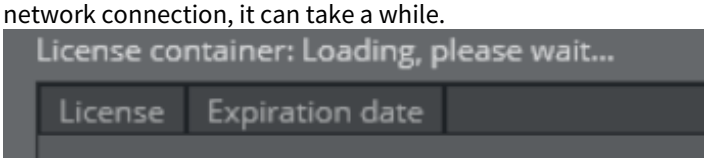
The CodeMeter local installation may have the WIBU containers set up or they can be connected to another CodeMeter license server. Please read the Viz Licensing documentation on how to set up CodeMeter containers.

> **Note:** The Codemeter.exe service of the Wibu Licensing system, must be running in order to start Pilot Data Server.

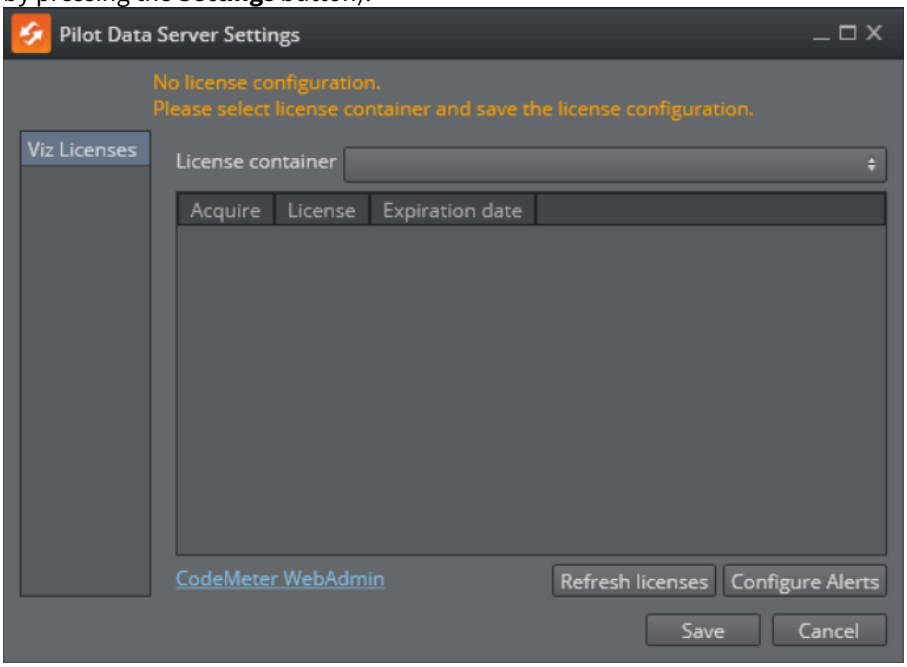
- Setting Up Licenses
- Configuring Alerts
  - Configuring SMTP Alert

4.7.1 Setting Up Licenses

1. After starting Pilot Data Server launcher, it tries to connect to a valid CodeMeter server. Depending on the network connection, it can take a while.



2. If there are no settings saved in the local machine, a new window is opened (this window can also be opened by pressing the **Settings** button).

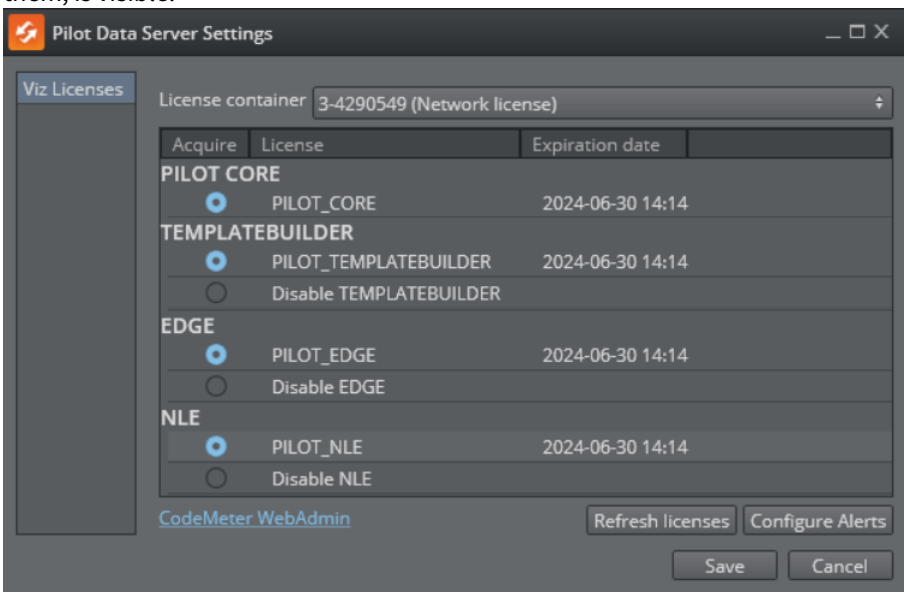


Copyright © 2026 Vizrt

3. Select a container from the drop-down list:

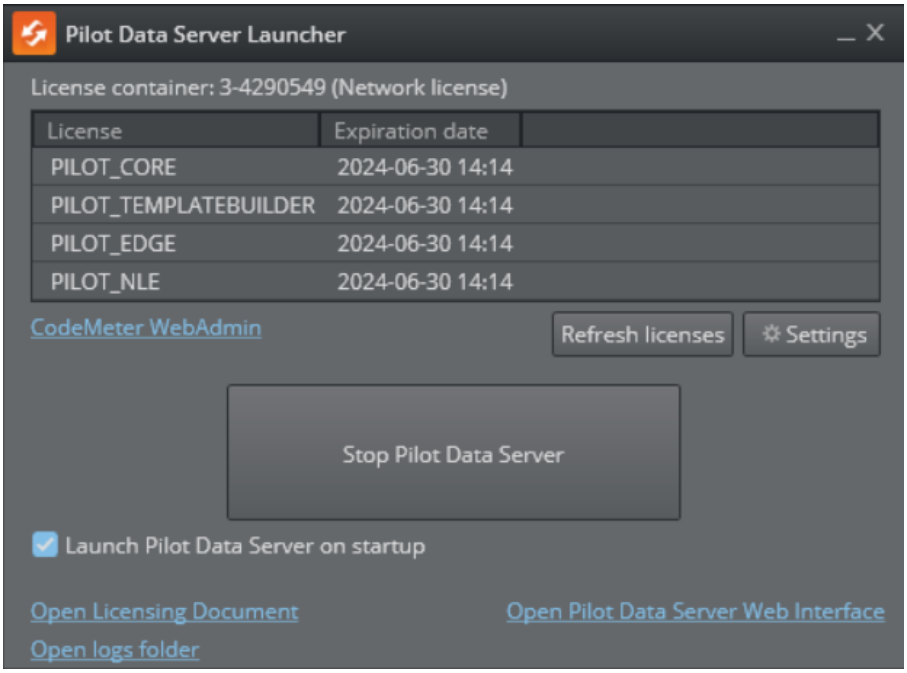


   If no containers are available, press the link for CodeMeter WebAdmin at the bottom left of the window to open the CodeMeter administration page. After solving any issues on CodeMeter, you can press the **Refresh Licenses** button.
4. When a valid container is set, a list of licenses with the available licenses and the expiration date for each of them, is visible:



   The *PILOT_CORE* license is for PDS and it is mandatory. The remaining licenses can be enabled or disabled manually.
5. After selecting the licenses, click **Save** to go back to the initial window. This window shows the state of the selected licenses:

Copyright © 2026 Vizrt



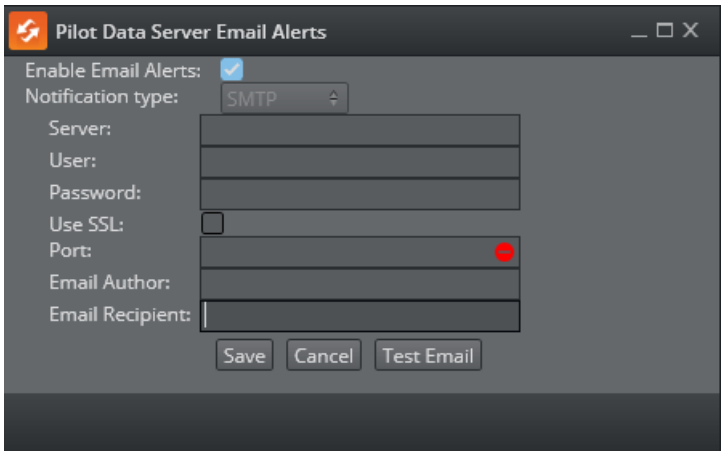
6. Click **Launch as Windows Service** to launch PDS. To start PDS at system startup, tick the **Launch on system startup** box.

4.7.2 Configuring Alerts

When PDS is running, alerts about license expiration can be configured by clicking on the **Configure Alerts** button from the MSE launcher. The PDS license alerter can be configured to either send email with SMTP, or to send journal messages via the Graphic Hub. There is a 7 days grace period on expiry of the PDS license.

Configuring SMTP Alert

To configure email alerts about license expiration, select **SMTP Server** from the **Notification type** dropdown, and configure the SMTP server settings.



Copyright © 2026 Vizrt

Enable Email Alerts: Option to use email alerts when license is expiring.
Notification type: Only SMTP email configuration is possible.
Server: SMTP server to use.
User: SMTP server user.
Password: SMTP server password.
Use SSL: Check if your server uses SSL, otherwise leave unchecked.
Port: SMTP server port.
Email Author: Email account which sends the email (usually user must have account permissions with this account).
Email Recipients: Email account that receives the alert email.

When finished, press the **Save** button.

Copyright © 2026 Vizrt

4.8 Pilot Data Server Settings

- The Settings File
- The Settings UI
    - HTTPS
    - Configuring SSL Certificates
        - Understanding SSL Certificates
        - Configuring the Certificate for Pilot Data Server
    - STOMP Notifications
    - Proxy
    - Database Connection
        - Graphic Hub
        - Oracle
    - Enable Indexing
- Configuring the Pilot Data Server HTTP Port
- STOMP over TCP
    - Changing the STOMP TCP Address
    - URLs Embedded in STOMP Notifications
- STOMP over WebSocket

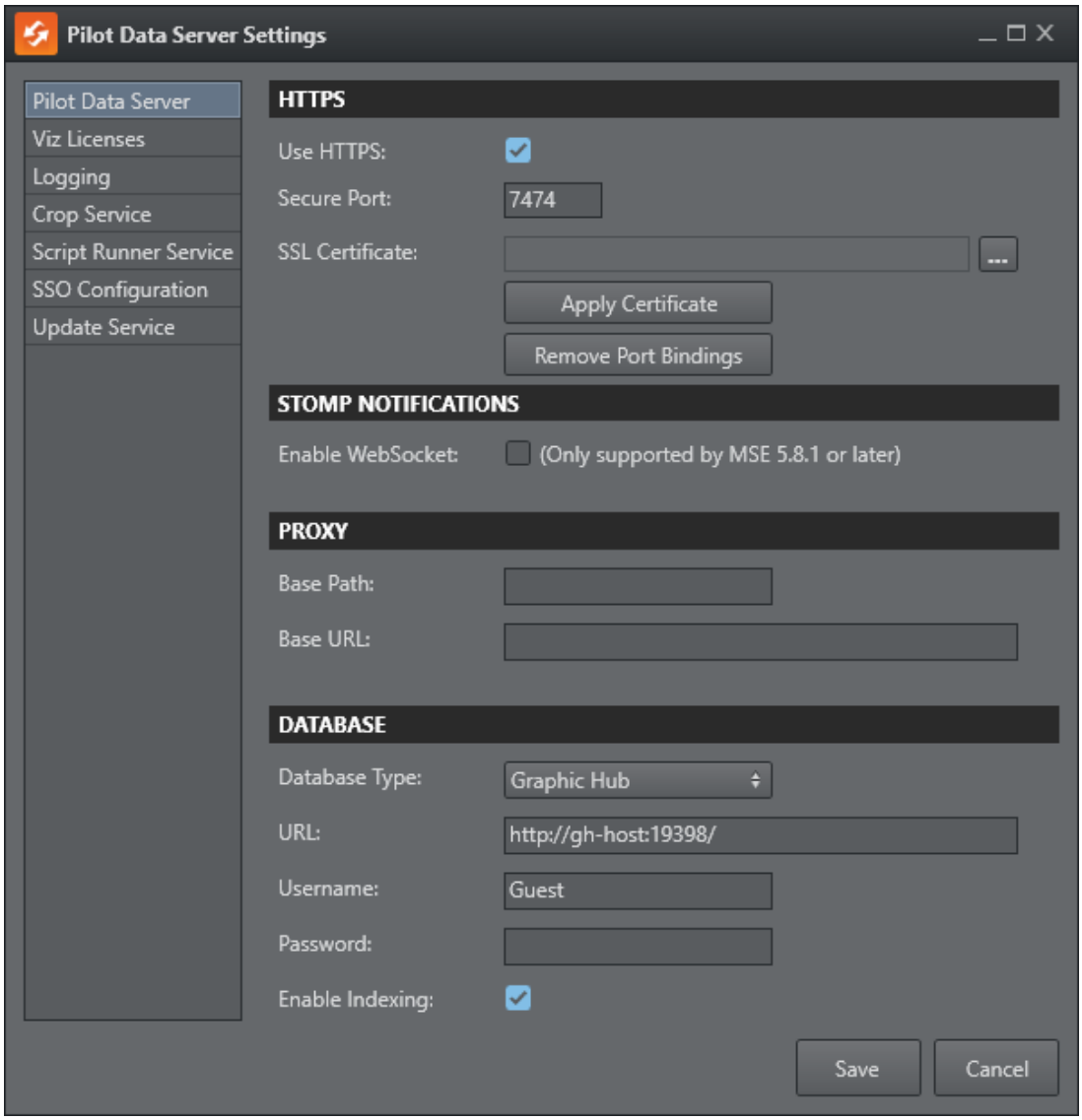
4.8.1 The Settings File

Most of the Pilot Data Server settings are stored in a single file: `%ProgramData%\Vizrt\Pilot Data Server\DataServerConfig.ini`.

This file backs most of the settings in the Launcher Settings. This file can be manipulated without the Launcher running, but the Pilot Data Server service needs to restart if the settings are changed in the file. This file can also be copied to a target machine to pre-configure Pilot Data Server without entering the UI.

Copyright © 2026 Vizrt

4.8.2 The Settings UI



HTTPS

Pilot Data Server can be configured to use the SSL transport layer when communicating. If HTTPS is configured, all endpoints of Pilot Data Server can be accessed using the HTTPS protocol.

1. Check **Use HTTPS**.
2. The suggested default port is 7373. Although it can be changed, it is recommended to use this port.

Copyright © 2026 Vizrt

Configuring SSL Certificates

To enable HTTPS for Pilot Data Server, you must configure an SSL certificate used by the server to secure communications. Select a certificate from the Windows Certificate Store, bind it to the configured port and application ID, and optionally, remove existing port bindings if necessary.

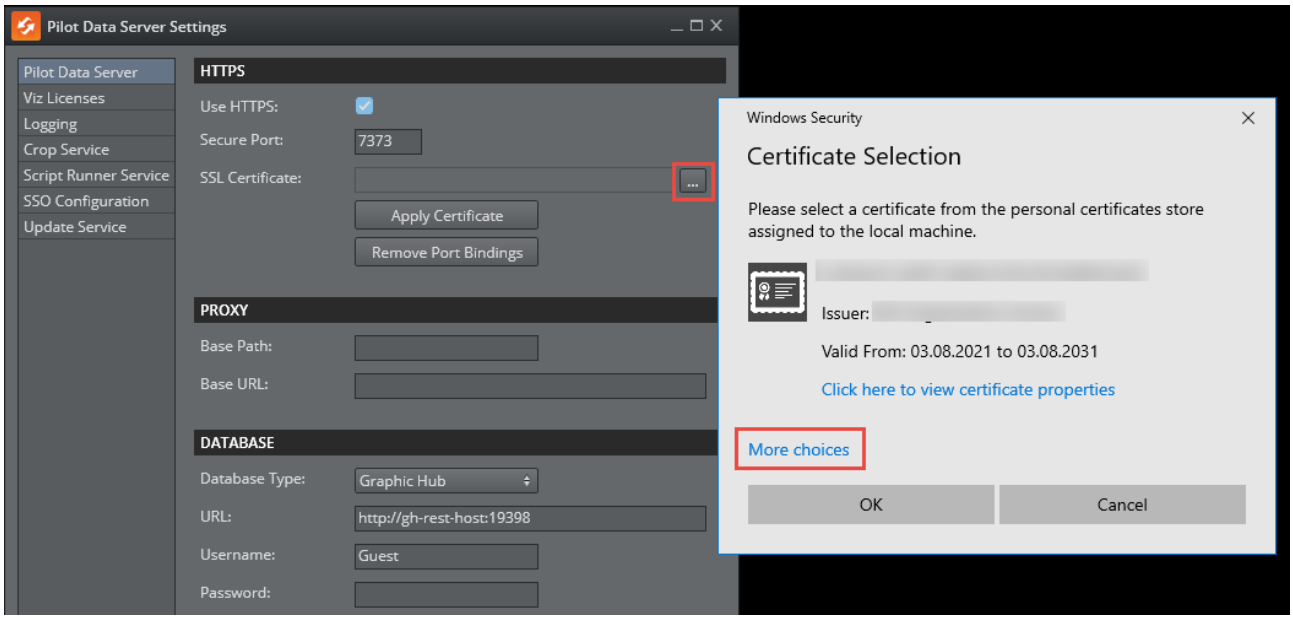
Understanding SSL Certificates

An SSL certificate is a digital credential that enables a secure communication, by encrypting data transmitted between clients and the server. It verifies the server's identity to clients, ensuring trustworthiness. Certificates are stored in the Windows Certificate Store and can be accessed via specific logical stores and locations. Pilot Data Server uses the following:

- **Store Location**
  - `LocalMachine` : This location stores certificates available to all applications and services on the machine.
  - Certificates in this location typically require administrative privileges to access.
- **Store Name**
  - `My` : Also known as the **Personal** store, it contains certificates intended for server-level use, such as those for HTTPS.

Configuring the Certificate for Pilot Data Server

To configure a certificate for Pilot Data Server, follow these steps:



1. **Browse for a Certificate**: Click the ... button next to the **SSL Certificate** text box to browse for a certificate.
2. **Select a Certificate**:

Copyright © 2026 Vizrt

- The Windows Certificate Store browser opens, defaulting to `My` store under the `LocalMachine` location.
- It may be necessary to click **More choices** in the dialog box to expand the list of certificates in the store.
- Select a certificate that matches your server's domain or hostname. Ensure the certificate:
  - Is valid (not expired).
  - Contains the private key.
  - Is issued by a trusted Certificate Authority (CA), or is a custom certificate, such as a self-signed certificate, that you trust and intend to use.
- Click **OK** to confirm your selection.

3. **Apply the Certificate**:
   - Once a certificate is selected, its “friendly name” appears in the text box. Hovering over the “friendly name” of the certificate shows more comprehensive info on the selected certificate.
   - Click **Apply Certificate**. This action binds the certificate to the configured port and application ID using the `netsh` command. The command associates the certificate’s thumbprint with the application and ensures HTTPS is enabled.
4. **Remove Existing Port Bindings (Optional)**:
   - If you need to remove existing certificate bindings for the configured port and application ID, click **Remove Port Bindings**. This unregisters any current SSL certificate bindings associated with the application. This operation only locates bindings done by the PDS launcher. Any manual bindings previously done are not detected. See note below.

⚠ **Notes:**

- Search providers and Preview Server must also be in HTTPS environments in order for calls to them to work. To enable support for HTTPS requests for Preview Server, see the Preview Server Administrator Guide.
- If necessary, to manually clear the HTTPS port configured using the Pilot Data Server Launcher (Pilot Data Server, Script Runner, Crop Server), clear from the sslcert and urlacl lists with the following commands:
  - netsh http delete sslcert ipport=0.0.0.0:7373
  - netsh http delete urlacl https://+:7373/

STOMP Notifications

- **Enable WebSocket:** This setting enables STOMP notifications over the WebSocket protocol. See below.

Proxy

Pilot Data Server can be installed along with a proxy/reverse-proxy server. See Proxy Support for more information.

- **Base Path:** This setting modifies the internal URL only when no proxy server is configured. It appends a path segment to the internal URL, defining the endpoint where Pilot Data Server listens for requests. For example, if the internal Pilot Data Server URL is `http://mypds:8177`, setting "pds" as the base path, changes the internal URL to `http://mypds:8177/pds`.

Copyright © 2026 Vizrt

- **Base URL:** This setting rewrites the internal URL to an external-facing URL in every response, making it accessible through the proxy. This ensures clients interact with a consistent URL matching the proxy's external address. The base URL must be provided as an absolute URL string. For example, setting the base URL to `https://example.com/pds` causes the server to rewrite response URLs to match this external address.

> ⚠ **Note:** These two settings cannot be combined. Base URL takes precedence over Base Path.

Database Connection

Pilot Data Server can connect to either an Oracle or a Graphic Hub database. Select your preferred choice in the dropdown.

Graphic Hub

- **URL:** Full URL to the Graphic Hub REST service.
- **Username:** Username when Pilot Data Server connects to Graphic Hub.
- **Password:** Password when Pilot Data Server connects to Graphic Hub.

Oracle

- **Data Source:** A connection string with format *hostname/servicename (SID)* to the Oracle database, for example *localhost/vizrtdb.*
- **Username:** Username when Pilot Data Server connects to the Oracle Database.
- **Password:** Password when Pilot Data Server connects to the Oracle Database.

Enable Indexing

When enabled, Pilot Data Server creates a local indexing file under:

```
%PROGRAMDATA%\Vizrt\Pilot Data Server\[GUID]\
```

The `[GUID]` is a hash generated from the connection parameters (such as hostname, port and username), used to connect to Graphic Hub or Oracle. As a result, each unique database configuration has its own folder with a corresponding index folder.

The indexing file maintains a local mirror of data element names, template names, and concept names. With indexing enabled, searches across concepts for templates and data elements are significantly faster. However, Pilot Data Server consumes more resources, as it must actively query the database to keep the index in sync.

If indexing is disabled, all client search queries are forwarded directly to the database backend, resulting in slower performance, especially for cross-concept searches.

4.8.3 Configuring the Pilot Data Server HTTP Port

The default Pilot Data Server port is 8177. To change the port:

Copyright © 2026 Vizrt

1. On the Pilot Data Server machine, open the ini file in a text editor: *%ProgramData\Pilot Data Server\DataServerConfig.ini*.
2. Replace the following setting: *Port = 8177*.

4.8.4 STOMP over TCP

Pilot Data Server sends change notifications to clients through the STOMP protocol (the default STOMP TCP port is 9876). This is used to form a URL to the STOMP server, sending change notifications to Media Sequencer. This URL is served out as a link in the `/changelog` endpoint of Pilot Data Server.

> ⚠ **Note:** This STOMP endpoint cannot be placed behind a HTTP proxy. The host name of the STOMP server must be accessible over TCP from the Media Sequencer host.

On the Pilot Data Server machine, open the ini file in a text editor: *%ProgramData\Pilot Data Server\DataServerConfig.ini*.

The STOMP address consists of the machine's fully qualified domain name (FQDN) and the STOMP port.

Changing the STOMP TCP Address

The STOMP TCP server uses the machine's FQDN as its base address (the default port is 9876). The port can be changed by setting `ChangeServerPort` in the ini file:

```
ChangeServerPort = 9988
```

The resulting STOMP URL is *stomp://hostname.domain.com:9988/*.

If the auto-detected FQDN is incorrect or unreachable from client machines (for example, because the machine picks up the wrong DNS suffix), the hostname used in the STOMP URL can be overridden by setting `ChangeServerReportAddress` to the correct hostname.

```
ChangeServerReportAddress = pdshost.server.example.com
```

The resulting STOMP URL becomes *stomp://pdshost.server.example.com:9988/*. If `ChangeServerReportAddress` is left empty (default), the auto-detected FQDN is used.

URLs Embedded in STOMP Notifications

The base URL used inside STOMP change messages (for links included in the data of the notifications) is determined separately from the STOMP connection address. It is controlled by the `BaseUrl` setting:

```
BaseUrl = https://baseurl.domain.com
```

If `BaseUrl` is not set, the server constructs a base URL from the configured protocol (HTTP/HTTPS), port, base path, and the machine's FQDN.

Copyright © 2026 Vizrt

Note: `ChangeServerReportAddress` does not affect the URLs embedded inside STOMP messages. It only controls the hostname that clients use to connect to the STOMP service.

4.8.5 STOMP over WebSocket

Pilot Data Server also provides STOMP over WebSocket for clients that prefer WebSocket transport.

- The WebSocket STOMP endpoint URL is derived from the server's HTTP base URL (not from the STOMP TCP hostname).
- It uses the configured `BaseUrl` (and `BasePath`) to build the endpoint.
  - If `BaseUrl` is set, WebSocket STOMP uses it as the base. Otherwise, the server constructs from protocol (HTTP/HTTPS), port, `BasePath`, and the machine's FQDN.
- `ChangeServerReportAddress` does not affect the WebSocket STOMP endpoint. It only overrides the hostname used in the TCP STOMP URL embedded in the changelog feed.

Info: The WebSocket STOMP endpoint can run behind a reverse proxy, provided the proxy supports the WebSocket upgrade (HTTP 101 Switching Protocols). Ensure the proxy forwards the path/stomp, and preserves query parameters.

Copyright © 2026 Vizrt

4.9 Log Files

Script Runner, Crop Service and Pilot Data Server services logs to different files.

Service	Log file path	Configurable
Pilot Data Server	%ProgramData% \Vizrt\Logs\PilotDataServer\server.log	Log file size and log level can be configured in the Launcher UI.
Crop Service	%ProgramData% \Vizrt\Logs\CropService\cropservice.log	Log file size and log level can be configured in the Launcer UI if CropServer is installed on the same host as Pilot Data Server. The settings are written to *cropservice.ini*.
Script Runner	%ProgramData% \Vizrt\Logs\ScriptRunner\scriptrunner.log	Log file size and log level can be configured in *PilotScriptRunnerHostService.exe.config* in the installation path.
Thumbnail Generator	%ProgramData%\Vizrt\Logs\Viz Pilot Thumbnail Generator	Log file size and log level can be configured in *PilotThumbnailGeneratorHostService.exe.config* in the installation path.

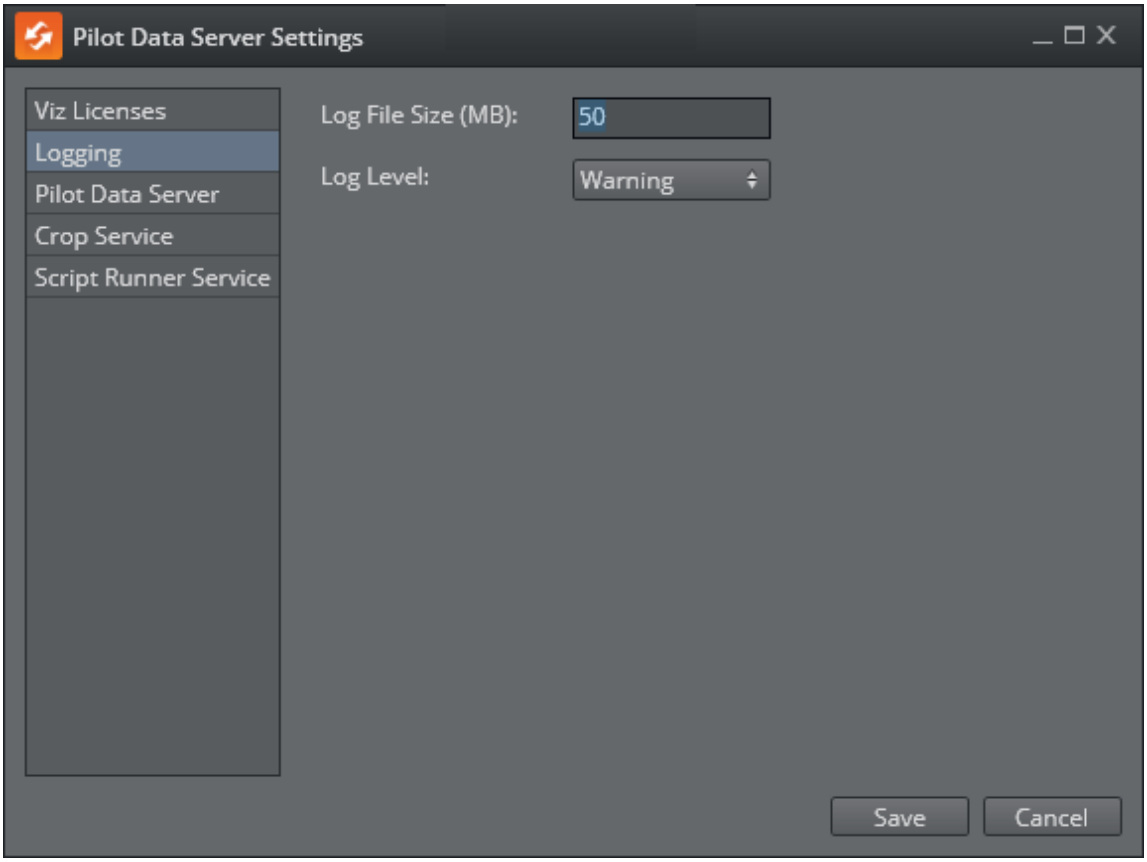
Note: The maximum default size of the log is 1000Mb, with one active file and one *.old* file.

4.9.1 Logging Configuration

The configuration of Logging for Pilot Data Server is done using the Pilot Data Server Launcher.

1. On the initial window of the launcher, click the **Settings** button.
2. Select **Logging Config** to show the configuration settings.

Copyright © 2026 Vizrt



Log File Size: Defines the maximum file size for log files (in mb) for PDS. When the size is exceeded, a new file is created in the same folder.

Log Level:

a. **Debug:** Log message for debug purposes.
b. **Info:** Log for general messages, for example "Connected to host", etc.
c. **Warning:** Log message for irregular situations that may cause unexpected results.
d. **Error:** Used for serious and unexpected errors, that may cause situations with unwanted results.

3. After configuring, click **Save**.

Warning: Clicking **Save** applies changes to all settings in this window, including logging, licensing, PDS, etc.

Info: Each log level, logs all of the messages of the level below. Therefore, debug logs all of the messages while the error level only logs errors.

Copyright © 2026 Vizrt

4.10 Crop Service

Vizrt Crop Service is a Windows service that enables on-the-fly image cropping, and is delivered as a separate installer. If installed on the same machine as Pilot Data Server, configuration is performed through the **Launcher UI**.

All runtime configuration is stored in `%ProgramData%\Vizrt\CropService\cropservice.ini`.

> **Note:** The Crop Service no longer reads port settings from exe.config files. Only **cropservice.ini** is used for runtime configuration.

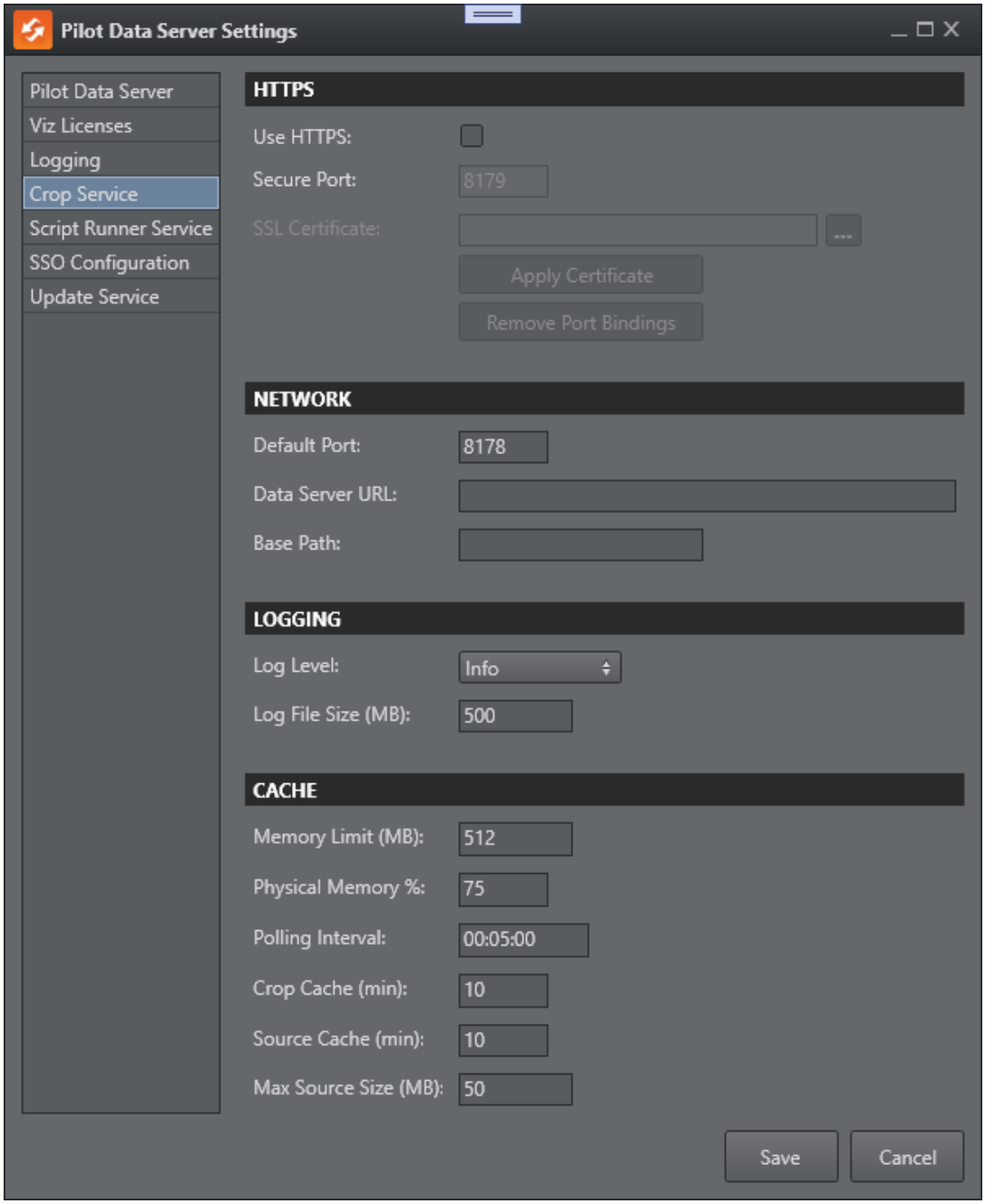
This page contains the following sections:

- Configuration via Launcher UI
  - HTTPS
  - Network
  - Logging
  - Cache
- Pilot Database Configuration
- Security
- Default INI Values
- Operational Notes
- Health endpoint

4.10.1 Configuration via Launcher UI

The Launcher UI is the authoritative way to configure the Crop Service. Each field maps directly to a key in the INI file.

Copyright © 2026 Vizrt



HTTPS

Used to configure secure communication and certificate binding.

- **Use HTTPS:** Check this option to enable HTTPS.

```
UseHttps=True | False
```

  - If `True` service listens on `SecurePort`.

Copyright © 2026 Vizrt

    - If `False` service listens on `DefaultPort`.
- **Secure Port:** The default port is **8179**.

  ```
  SecurePort=<port>
  ```

  It is recommended to:
    - Keep 8179 unless your environment requires a different port.
    - Ensure the port is open in host and network firewalls.
- **SSL Certificate:** Used to select a certificate from the Windows certificate store.
    - Click **Apply Certificate** to bind the certificate to the HTTPS port.
    - Click **Remove Port Bindings** to remove existing HTTP.SYS bindings.

> **Info:** The Launcher manages HTTP.SYS bindings. Do not use manual `netsh` commands unless explicitly required.

Network

- **Default Port** (HTTP only)**:** The default port is **8178**. Only used when `UseHttps=False`.

  ```
  DefaultPort=<port>
  ```
- **Data Server URL:** Set the URL to the Pilot Data Server service document (this must be an absolute URL). The Crop Server retrieves the Pilot Data Server settings to extract credentials to pre-authenticated hosts, and reads the *croptool_max_image_area* setting from the database.

  ```
  PilotDataServerUri=<absolute-url>
  ```
- **Base Path** (Reverse Proxy Support)**:** This setting is only used when the Crop Service is published under a sub-path behind a reverse proxy.

  ```
  BasePath=<path>
  ```

  For example, the public URL is https://proxy.company.com/vizrt/cropservice. Therefore, the base path value can be set to *vizrt/cropservice*. In most modern deployments, `BasePath` should NOT be needed.
  If your reverse proxy forwards standard headers and preserves the full request path, the Crop Service can derive the correct public URL from headers.

Logging

Controls log verbosity and file size.

- **Log Level**

Copyright © 2026 Vizrt

```
LogLevel=<level>
```

  - Debug
  - Info
  - Warning
  - Error
- **Log File Size (MB):** Maximum size before log cycling.

```
LogFileSize=<megabytes>
```

Cache

These settings control memory usage and caching behavior.

- **Memory Limit (MB):** Total shared memory budget across crop result cache, and source image cache. The default is 512.

```
[Cache]
CacheMemoryLimitMegabytes=<mb>
```

- **Physical Memory %:** Hard upper cap based on system RAM. The default is 75.

```
[Cache]
PhysicalMemoryLimitPercentage=<0-100>
```

  This prevents the service from exhausting system memory even if **Memory Limit** is set high.
- **Polling Interval:** Frequency of cache resource checks (hh:mm:ss). The default is 00:05:00.

```
PollingInterval=00:05:00
```

- **Crop Cache (min):** Retention time for processed crop results. The default is 10 minutes.

```
CropResultCacheMinutes=<minutes>
```

- **Source Cache (min):** Retention time for downloaded source images. The default is 10 minutes.

```
SourceImageCacheMinutes=<minutes>
```

- **Max Source Size (MB):** Do not cache source images larger than this threshold. The default is 50.

```
MaxSourceImageCacheSizeMegabytes=<mb>
```

Copyright © 2026 Vizrt

4.10.2 Pilot Database Configuration

In the Pilot Database, configure these settings as follows.

- **crop_service_uri:** Must point to the externally reachable Crop Service endpoint. For example:
  - **HTTP:** `http://hostname:8178/`
  - **HTTPS:** `https://hostname:8179/`

> **Info:** This must match:
> - Protocol (HTTP/HTTPS)
> - Public hostname
> - Public port

- **croptool_max_image_area** (optional)**:** Defines the maximum served image area.

```
width × height (pixels)
```

Images larger than this are scaled down proportionally, even if no crop is applied.

4.10.3 Security

The Crop Service retrieves remote images on request. These settings let you optionally restrict which sources it may contact and bound the resources a single request can consume.

By default the allow-list is empty, so any `HTTP` / `HTTPS` source is permitted — the Crop Service is designed to run on a **trusted, segmented network**, where the primary controls are network-level (restrict which systems can reach the service, and which destinations it may contact). When stricter application-level control is required, configure an allow-list of approved sources. Regardless of the allow-list, the service returns only generic error messages to callers (full detail is kept in the protected server-side log), rejects responses that are not images, follows a limited number of redirects (each re-validated against the allow-list), and never forwards pre-authenticated credentials across a redirect to a different host.

Allowed Hosts: Optional comma-separated allow-list of approved image sources. When empty, the allow-list is disabled and any host is permitted. When set, only matching sources are retrieved; all others are rejected.

```
[Security]
AllowedHosts=<comma-separated entries>
```

Each entry is a host, optionally qualified with a scheme and/or port. A missing scheme or port matches any:

Entry	Permits
`images.company.com`	that host, any scheme, any port
`images.company.com:8443`	that host, only on port 8443

Copyright © 2026 Vizrt

Entry	Permits
`https://images.company.com`	that host, only over HTTPS
`https://images.company.com:8443`	that host, HTTPS, port 8443 only

`Example: AllowedHosts=https://images.company.com, mam.internal:4444`

Max Download Size (MB): Maximum size of a source image the service will download. Larger responses are rejected before processing. The default is 100.

```
[Security]
MaxDownloadSizeMegabytes=<mb>
```

Request Timeout (seconds): Connection and read timeout for outbound source-image requests. The default is 100.

```
RequestTimeoutSeconds=<seconds>
```

Max Redirects: Maximum number of HTTP redirects to follow for a source-image request; each redirect target is re-validated. The default is 10.

```
MaxRedirects=<count>
```

Info. Each outbound fetch is recorded in the log at **Info** level (caller address, source host and path, result and size), so keep LogLevel=Info to retain request/audit logging. The full source URL, including any query string, is logged only at **Debug**.

4.10.4 Default INI Values

When created initially:

```
[Output]
JpegQuality=95

[Logging]
LogLevel=Info
LogFileSize=500

[Network]
UseHttps=False
DefaultPort=8178
SecurePort=8179
BasePath=
```

Copyright © 2026 Vizrt

```
PilotDataServerUri=

[Security]
AllowedHosts=
MaxDownloadSizeMegabytes=100
RequestTimeoutSeconds=100
MaxRedirects=10

[Cache]
CacheMemoryLimitMegabytes=512
PhysicalMemoryLimitPercentage=75
PollingInterval=00:05:00
CropResultCacheMinutes=10
SourceImageCacheMinutes=10
MaxSourceImageCacheSizeMegabytes=50
```

4.10.5 Operational Notes

- Restart the Crop Service after changing settings.
- Ensure configured ports are open in firewalls.
- For HTTPS:
  - Certificate Subject or SAN must match the public hostname.
  - Certificate must be trusted by clients.
- Prefer reverse proxy header forwarding over manual BasePath configuration.
- Do not edit legacy EXE `.config` files.

4.10.6 Health endpoint

The Crop Service exposes a lightweight health and diagnostics endpoint that reports basic runtime information, cache statistics, and key configuration paths. It supports content negotiation and returns JSON when requested via the Accept header. Otherwise, it responds with human-readable plain text.

```
Route: /health
```

The purpose is to:
• Verify the service is up and responding.
• Inspect basic process metrics (memory, handles, uptime).
• Inspect cache item counts.
• Discover important configuration paths (for example, INI and log file).

Example (JSON)

```
GET /health
Accept: application/json
```

Returns:

```
{
    "cache": {
        "cropResultCacheEntries": 0,
        "sourceImageCacheEntries": 0
    },
    "configFilePath": "C:\\ProgramData\\Vizrt\\CropService\\cropservice.ini",
    "logFilePath": "C:\\ProgramData\\Vizrt\\Vizrt Logging\\CropService\
\cropservice.log",
    "process": {
        "handleCount": 898,
        "memoryWorkingSet": "107 MB",
        "privateMemoryBytes": "74 MB",
        "uptime": "0.00:02:41"
    },
    "timestamp": "2026-02-13T13:33:55.1973069Z",
    "version": "9.5.0.0022"
}
```


Example (plain text)

```
GET /health
```

Returns:

```
Timestamp: 2026-02-13T13:35:01.6806058Z
Log file: C:\ProgramData\Vizrt\Vizrt Logging\CropService\cropservice.log

Crop Service version: 9.5.0.0022

Process info:
PrivateMemoryBytes: 74 MB
MemoryWorkingSet: 107 MB
HandleCount: 953
Uptime: 0.00:03:48

Cache runtime stats:
Crop result cache entries: 0
Source image cache entries: 0

Configuration:
Product version: 0.0.0.11569
Config file: C:\ProgramData\Vizrt\CropService\cropservice.ini

[Logging]
LogLevel: Info
LogFileSize: 500 MB

[Network]
UseHttps: False
DefaultPort: 8178
```

Copyright © 2026 Vizrt

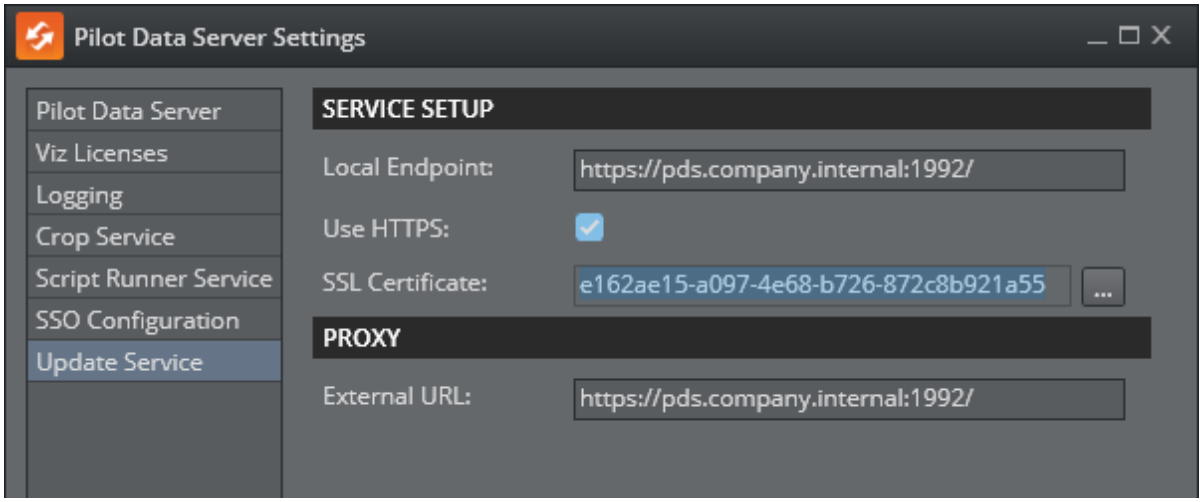
```
SecurePort: 8179
BasePath: (none)
PilotDataServerUri: (none)

[Security]
AllowedHosts: (none - allow-list disabled)
MaxDownloadSizeMegabytes: 100
RequestTimeoutSeconds: 100
MaxRedirects: 10

[Cache]
CacheMemoryLimitMegabytes: 511
PhysicalMemoryLimitPercentage: 75
PollingInterval: 00:05:00
CropResultCacheMinutes: 10
SourceImageCacheMinutes: 10
MaxSourceImageCacheSizeMegabytes: 50
```

Copyright © 2026 Vizrt

4.11 Pilot Update Service

Pilot Update Service is compatible with scripting in Viz Pilot Edge templates. It runs as a service on the same host as Pilot Data Server, and accepts requests to run the internal template's JavaScript in the **onUpdate** method, or auto-update the data through feed bound fields for the given data element. The service is usually called from Media Sequencer when the data element is being played out in the Viz Engine. The following topics are covered:

- Configuring Update Service
- Configuring Certificate Authorities (CAs)
    - Configuring a Certificate for the Windows Service
    - Configuring a Custom CA for Outgoing Connections
- Logging



4.11.1 Configuring Update Service

The configuration of Pilot Update Service, is written to and retrieved from *appsettings.json* in the installation folder, for example `%ProgramFiles%\Vizrt\Pilot Data Server\UpdateService`.

The default ports for the service are **1991** for HTTP and **1992** for HTTPS.

- **Local Endpoint:** URL where the service is running and reachable on the localhost. Use the existing, suggested URL or type in a different URL. The port of this URL is used when the service is starting.
- **Use HTTPS:** Check this box if the service is expected to run in secure mode, communicating with SSL/TLS over the HTTPS protocol.
- **SSL Certificate:** Certificate chosen from the Windows Certificate Store, used by the server to allow clients to connect securely.
- **External URL:** If the service runs behind a proxy, this is the URL the service should be reachable from outside the proxy. Media Sequencer is typically the device that calls Update Service, therefore, this URL has to be reachable from the Media Sequencer host. The URL should be the same as the Local Endpoint if there is no proxy present.

Copyright © 2026 Vizrt

4.11.2 Configuring Certificate Authorities (CAs)

The Pilot Update Service runs as a Windows service and wraps a Node.js server internally. While the Windows service can utilize certificates from the Windows Certificate Store to allow clients to connect securely, the Node.js component also needs certificates for its internal functionality. Specifically, the Node.js server connects to a Pilot Data Server to download the script and the VDF model, and the update script itself may fetch resources from servers on HTTPS, which requires properly configured certificate authorities (CAs) to validate those connections.

Configuring a Certificate for the Windows Service

The **Pilot Update Service** allows secure client connections by binding a selected certificate from the Windows Certificate Store. To configure the certificate:

1. **Access the Certificate Selection Dialog:**
   - Click the button with the three dots (`...`) in the service configuration UI. This opens the Windows Certificate Store browser.
2. **Select a Certificate:**
   - In the certificate selection dialog, click **More choices** to view all registered certificates under the **Personal (My)** store.
   - Select the desired certificate from the list. Ensure the certificate is valid and matches the intended domain or service usage.

After selecting the certificate, it is automatically used by the service for HTTPS communication without further manual steps.

Configuring a Custom CA for Outgoing Connections

Please follow these steps to configure the internal Node.js server to use custom CAs for its internal HTTPS communication:

> ⚠ **Note:** These steps are only required when using **custom Certificate Authorities (CAs)**. If the certificates are issued by an officially recognized CA, no additional configuration is necessary.

Step 1: Export the Required Certificate(s) from the Windows Certificate Store

1. **Open Certificate Manager:**
   - Press `Win + R` , type `certmgr.msc` , and press `Enter` .
2. **Locate the Certificate:**
   - Navigate to the appropriate certificate store (for example, `Trusted Root Certification Authorities` or `Intermediate Certification Authorities`).
3. **Export the Certificate:**
   - Right-click on the desired certificate and select **All Tasks > Export**.
   - In the Certificate Export Wizard:
     - Choose **Base-64 encoded X.509 (.CER)** as the export format.
     - Save the exported file to a known location (for example, `C:\certificates\mycert.cer`).
4. **Repeat for Additional Certificates** (if necessary):

- If multiple certificates are required, export each one individually.

Step 2: Combine Certificates (Optional)

- If you need to provide multiple certificates, combine them into a single PEM file:
  - Open each exported `.cer` file in a text editor (for example, Notepad).
  - Copy and paste the content of each file into a new file, ensuring they are in the correct order:

```
-----BEGIN CERTIFICATE-----
<Certificate 1 Content>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<Certificate 2 Content>
-----END CERTIFICATE-----
```

  - Save the combined file as a `.pem` file (for example, `C:\certificates\custom-ca.pem`).

Step 3: Configure Node.js to Use the Custom CAs

1. **Set the Environment Variable:**
   - Define the `NODE_EXTRA_CA_CERTS` environment variable and point it to the PEM file:
     - Open the **System Properties** dialog:
       - Press `Win + Pause/Break` > **Advanced system settings** > **Environment Variables**.
     - Under **System variables**, click **New**.
     - Set the following:
       - **Variable name**: `NODE_EXTRA_CA_CERTS`
       - **Variable value**: Path to the PEM file (for example, `C:\certificates\custom-ca.pem`).
2. **Restart the Pilot Update Service:**
   - Restart the service to apply the changes.

> ⚠ **Note:**
> - The Windows service does not require additional configuration for CAs if certificates are already available in the Windows Certificate Store for client connections.
> - The Node.js server requires the custom CA configuration described above, to connect to other HTTPS resources securely.
> - When combining certificates, maintain the order: root certificates should appear last, preceded by intermediate and leaf certificates.
>
> By following these steps, the Pilot Update Service handles both secure client connections via the Windows service and internal HTTPS communication within the Node.js server.

Copyright © 2026 Vizrt

4.11.3 Logging

Pilot Update Service uses Serilog, a robust and flexible logging library, to capture and manage application logs. Serilog supports structured logging and can output logs to various sinks, such as files and the console, making it easier to monitor and troubleshoot the service. The system applies the following Serilog settings:

1. **Outputs:**
   - Logs are written to both the console and a file located at `%ProgramData%\Vizrt\logs\Pilot Update Service\service<date>.log`.
2. **File Retention and Size Management:**
   - A maximum of **31 log files** are retained. When the limit is exceeded, the oldest file is deleted.
   - Each log file has a size cap of **1GB**. If this limit is reached, logging to the file stops. This safeguard prevents excessive file creation, particularly in scenarios such as infinite loops, ensuring key log data remains intact.
3. **Log Levels:**
   - The default log level is **Information**.
   - For non-lifetime, Microsoft-based events, the log level is set to **Warning** to reduce noise from routine request processing.

Copyright © 2026 Vizrt

4.12 Audit Logging

Pilot Data Server includes an audit feature (available to customers) that records write activity, providing insight into how the system is used over time. The audit is intended for operational overview and transparency.

- What is Audited
- What Information is Recorded
- How Audit Data is Stored
- Viewing Audit Data
- Configuration
- Data Protection

4.12.1 What is Audited

The audit records **write operations** to Pilot Data Server:

- HTTP POST
- HTTP PUT

Typical examples include creating or updating templates and elements.

Read operations (GET) are not audited.

4.12.2 What Information is Recorded

Each audit entry represents a single write operation, and includes metadata such as:

- Time of the operation.
- Type of operation (POST / PUT).
- User identity when single sign-on (SSO) is enabled.
- Client IP address when SSO is not enabled.

Template payloads and full data content are not recorded.

4.12.3 How Audit Data is Stored

Audit data is stored locally on the machine running Pilot Data Server, in an internal database file.

Key characteristics:

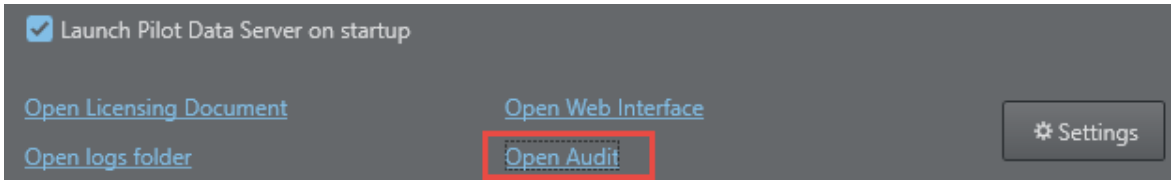
- Audit data is stored locally and not exposed through any API.
- Audit data is not transmitted outside the PDS host.
- Access to audit data requires operating system–level access to the server.

The audit log uses an internal integrity mechanism that manually modifies detectable entries.

4.12.4 Viewing Audit Data

An Audit Viewer is available from the Pilot Data Server Launcher.

Copyright © 2026 Vizrt



The Audit Viewer can be used to:

- Review audit activity for a selected time period.
- See the number of unique users performing write operations.
- See the total number of write interactions.
- Copy audit summary to clipboard.
- Delete existing audit data and restart audit logging from scratch.

Deleting audit data does not affect normal operation of Pilot Data Server.

4.12.5 Configuration

Audit logging is enabled by default, but it can be disabled by setting the following option in the Pilot Data Server configuration file:

```
DisableAudit=True
```

This setting disables all audit logging and is intended as an exception mechanism.

4.12.6 Data Protection

Audit entries may contain personal data (as defined in the Vizrt Data Processing Addendum), such as user identifiers or IP addresses, depending on the authentication setup and use of the audit feature. Customers act as controllers in respect to any personal data contained in audit entries, and are solely responsible for ensuring its compliance with Applicable Data Protection Law in connection with its use.

Customers acknowledge and accept that:

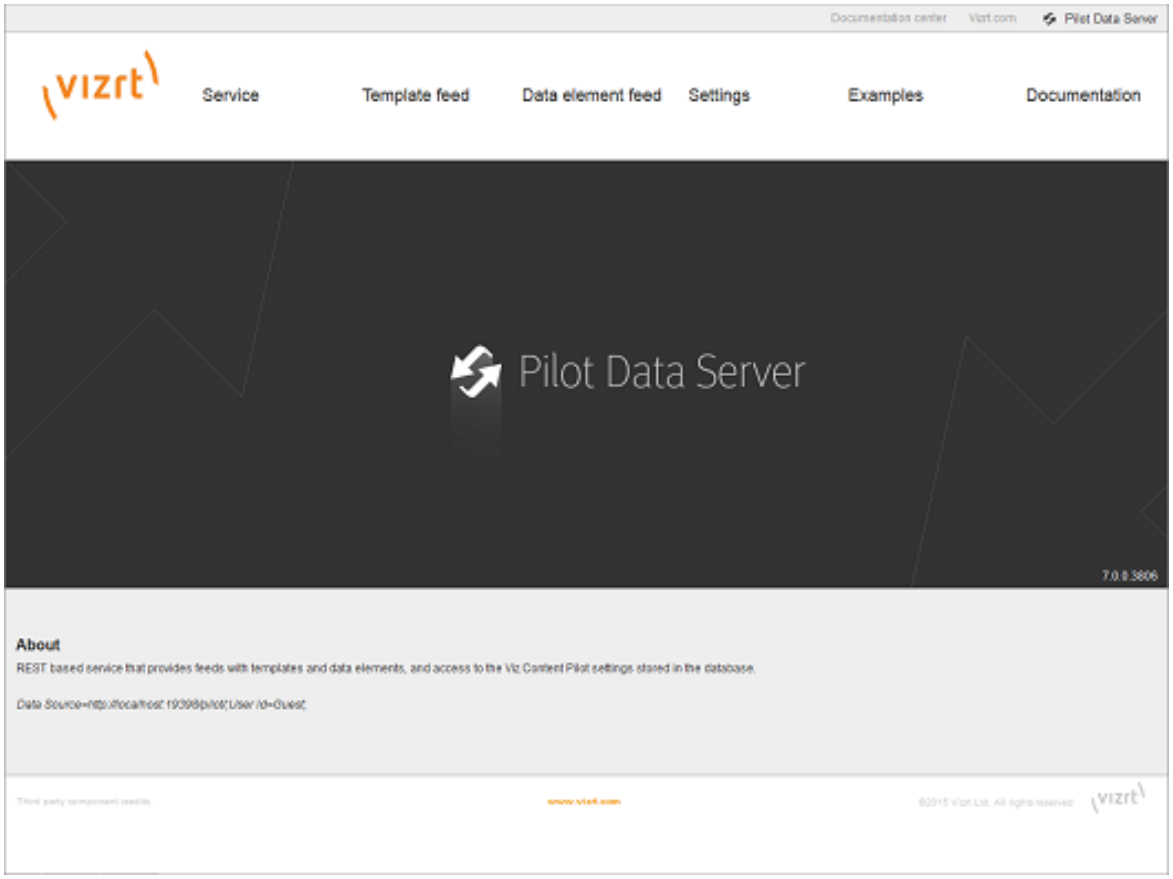
- Audit data is stored locally on the Pilot Data Server host, and is controlled by the customer.
- No automatic reporting or external transmission of audit data is performed, unless expressly agreed between Vizrt and the customer.
- Customers are responsible for retention control, access control, and deletion policies.

This feature is intended to provide usage insight and operational transparency, not employee monitoring or compliance enforcement.

Copyright © 2026 Vizrt

5 Web Interface

The Pilot Data Server web interface offers access to features including the Service Document, Template and Data Element Feeds, Viz Pilot Settings and REST API documentation.



5.1 Accessing the Pilot Data Server Web Interface

- Select the **Pilot Data Server Web Interface** from the Windows Start Menu.
- Alternatively, start the **Pilot Data Server** in a browser, using the host name of the machine running the Pilot Data Server, and port `8177`. For example: *http://dataserver.example:8177*

5.2 Service

The Service Document resource (for example: *http://<dataserver.example>:8177/service*) is an Atom Service Document that can be used by a client to discover the capabilities of a Pilot Data Server and the locations of the available Atom Publishing Protocol collections hosted on it.

Copyright © 2026 Vizrt

5.3 Template Feed

The Template Feed resource (for example: *http://<dataserver.example>:8177/templates*) is an atom feed that contains entries for each template stored in the Pilot Data Server. Metadata may include details of the template such as description, creation date, link to a thumbnail image and link to the Viz Data Format (VDF) model document describing the template.

5.4 Data Element Feed

The Data Element Feed resource (for example: *http://<dataserver.example>:8177/dataelements*) is an atom feed that contains entries for each data element stored in Pilot Data Server. Metadata may include details of the data element such as description, creation date, link to thumbnails and link to the Viz Data Format (VDF) payload document describing the data element.

5.5 Settings

The Settings page (for example: *http://<dataserver.example>:8177/settings*) is used to configure VCP Parameters and Tag Settings. See Settings Parameters for a full description of the Settings tab.

5.6 Examples

The Examples page (for example: *http://<dataserver.example>:8177/examples*) contains examples on how to interact with the Viz Pilot system.

5.7 Documentation

The Pilot Data Server Documentation page (for example: *http://<dataserver.example>:8177/help*) describes the REST API provided by Pilot Data Server. The page includes information on the Resource Types and Content Types that are used in the interface. The API itself lets you access template information, perform image search and person search, and provides a programing interface for other systems.

Copyright © 2026 Vizrt

6 RestVOS

6.1 Comparing RestVOS and Conventional VOS Access

- The fundamental difference is in the protocol used to search for images and deliver the results to the client.
- When accessing an Object Store in the conventional Viz Object Store (VOS) way, the client application goes directly to the database and the network share in order to get the data.
- When using RestVOS, the client accesses the same images and storage, but via the Pilot Data Server. This uses the HTTP protocol, the same protocol used when talking to Viz One systems.
- From the point of view of client applications, accessing RestVOS is no different from accessing a Viz One or any other asset search provider.
- The existing Object Store (see the Object Store section in the Viz Pilot User Guide) application is used to manage the images and storage, whether using standard VOS or RestVOS.

6.2 Advantages of RestVOS

- **Flexibility:** HTTP is an easier protocol to use and manage. By using the Pilot Data Server and the OpenSearch protocol, other applications can also perform the same searches, and there is no need to make the network share available on all clients.
- **Extended features:** Certain features such as Crop Service and Filter Media by Person Name (see the Viz Pilot User Guide), are only available when using RestVOS.
- **Future oriented:** As development on RestVOS continues, features such as load balancing and failover capabilities are added, benefiting image transfers. Once direct database access from Viz Pilot News is no longer required, management of Oracle clients is not necessary. Conventional access to VOS is phased out at some stage.

Copyright © 2026 Vizrt

7 Change Notifications

Change notifications are available through the REST API and the STOMP protocol on Pilot Data Server.

7.1 REST API

A list of database changes ordered from most recent to oldest:

- Located at *http://pdshost:8177/changelog*.
- Parameters:
  - **startId:** The ID of the least recent change, or -1 for no limit. Default is -1.
  - **lastKnownId:** The ID of the most recent change, or -1 for no limit. Default is -1.

7.2 STOMP API

In a Viz Pilot system, the text-based message Streaming Text Oriented Messaging Protocol (STOMP) is used to subscribe to change notifications from Graphic Hub REST and Pilot Data Server. STOMP provides push notifications of new changes.

- Located at: *stomp://pdshost:9876/?destination=/changelog&lastKnownId=-1.*
- Parameters:
  - **lastKnownId:** Mandatory, should be the ID in the STOMP link that is provided by the REST API. For example, when the data stored in the Pilot database changes, a message is delivered to all of the STOMP clients that are subscribed to the server.

Note: The default change notification port for the STOMP protocol is `9876`. Other services using this port may suppress the change notifications.

Copyright © 2026 Vizrt

8 Proxy Support

When deploying Pilot Data Server behind a proxy server, it is often necessary to rewrite requests, to ensure the application interprets and responds to client requests correctly. This process typically involves adjusting elements of the request to account for modifications introduced by the proxy, such as altered hostnames, schemes, or paths.

There are two primary methods for aligning the application's URL with the external-facing URL used by the proxy.

8.1 Using X-Forwarded Headers

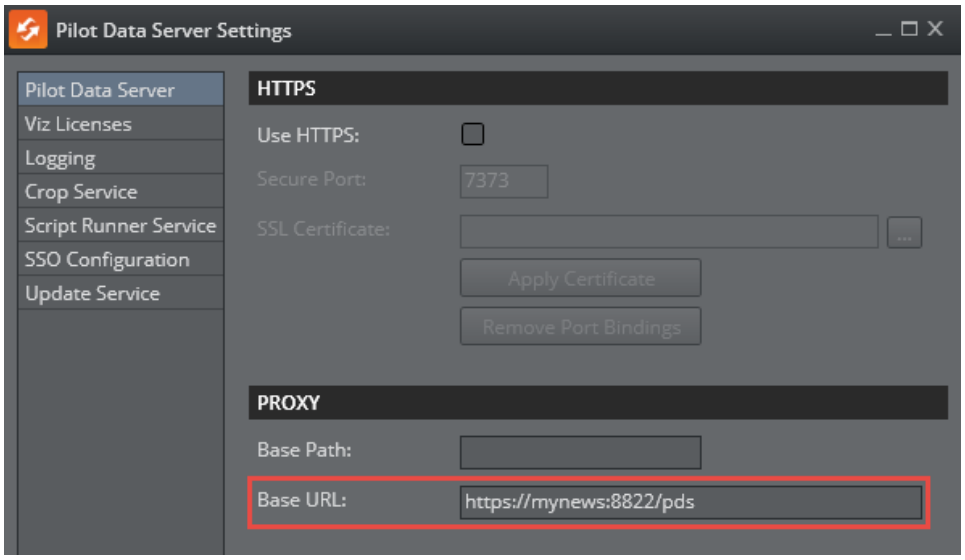
The proxy server must either populate the X-Forwarded headers or the Forwarded header in order for Pilot Data Server to serve the appropriate hostname, port and protocol in its responses:

- `Forwarded`
- `X-Forwarded-Host`
- `X-Forwarded-Port`
- `X-Forwarded-Proto`

Use a proxy server with the above headers configured. If the `Forwarded` or `X-Forwarded` headers are not configured, the proxy server can be configured to pass the `Host` header it received from the client to Pilot Data Server.

8.2 Configure Base URL

An alternative method is to explicitly configure the application's Base URL to match the external-facing URL provided by the proxy.



This cannot be combined with the forwarded-headers mentioned above.

Copyright © 2026 Vizrt

> **Note:** The **Base Path** cannot be combined with neither forwarded-headers nor the Base URL. The base path is only applied to the internal URL when no proxy is configured.

8.3 Redundancy and Failover

Although Pilot Data Server has been designed without built-in redundancy, it does support being used with off-the-shelf third-party HTTP load balancers. As the name suggests, a load balancer's main function is to distribute requests among multiple servers. Multiple servers provide redundancy, and most load balancers come with an option to only route requests to servers that respond, which provides failover.

Both Barracuda Load Balancer ADC and HAProxy have successfully been used in front of Pilot Data Server.

8.3.1 Load Balancer Quick Setup

Quickly set up a HAProxy load balancer:

Prerequisites

- Two (or more) hosts running Pilot Data Server, connected to the same database. They have host names *pds1.example* and *pds2.example* for the purpose of these instructions. Static IP addresses may be used instead.
- A Linux host that turns into a load balancer. It has the host name *proxy.example* for the purpose of these instructions. This is the host used in the installation steps below.

Installation Steps

1. On the Linux host, install HAProxy with the command `sudo apt-get install haproxy`.
2. Edit */etc/haproxy/haproxy.cfg* (for example with `sudo nano /etc/haproxy/haproxy.cfg`) and append the following lines:

```
frontend http-in
    bind :8177
    default_backend servers

backend servers
    server server1 pds1.example:8177 check
    server server2 pds2.example:8177 check

listen stats
    bind :80
    mode http
    stats enable
    stats uri /
```

3. Restart HAProxy with the command `sudo service haproxy reload`.

Copyright © 2026 Vizrt

Notes

- After following the steps above, it should be possible to use *proxy.example:8177* in place for *pds1.example:8177* and *pds2.example:8177*. Check whether the setup works by opening http://proxy.example:8177/ in a browser. The Pilot Data Server index page should load.
- The above configuration also provides a monitoring page at http://proxy.example/, that shows which Pilot Data Server hosts are responding. This does not tell you whether Pilot Data Server is working correctly, as hosts still respond when they cannot reach the database.

Copyright © 2026 Vizrt

9 MOS XML

The MOS (Media Object Server) protocol is a communication standard specifically designed for newsroom computer systems (NRCS) in the broadcast industry. It enables interoperability and seamless integration between various software, devices and plugins used within a newsroom environment. It is important to note that while the MOS protocol provides a standard framework for communication, the specific implementation and feature set may vary between different vendors and systems.

From Viz Pilot Edge 3.0 and Pilot Data Server 9.0, our philosophy in Vizrt is to use these guidelines for how to implement certain parts of the MOS protocol:

1. We want to be as compliant as possible to the MOS protocol version 2.8.5.
2. Where the protocol specification is vague or inadequate, we try to stick to the MOS XML values of existing released Vizrt software.
3. We want to be as internally consistent as possible between Viz Pilot News, Viz Pilot Edge and Pilot Data Server.

The following MOS XML values can be used by newsroom vendors and other consumers as a guide:

> **Info:** Please note that the value 50 below depends on the **video_mode** setting in Pilot Data Server, being 50 for PAL and 60 for NTSC.

The value for **<itemEdDur>** and **<itemEdStart>** are given in frames. **<objTB>** is given in frames per second and should be used to calculate the duration in minutes and seconds.

9.1 Graphics with No Timing Info

	Viz Pilot News 8.9 plugin	Viz Pilot Edge 3.0 / PDS 9.0
<objType>	PILOT	PILOT
<objTB>	0	50
<objDur>	1	1

9.2 Graphics with Timing Info

	Viz Pilot News 8.9 plugin	Viz Pilot Edge 3.0 / PDS 9.0
<objType>	PILOT	PILOT
<objTB>	50	50
<objDur>	1	1

Copyright © 2026 Vizrt

	Viz Pilot News 8.9 plugin	Viz Pilot Edge 3.0 / PDS 9.0
<itemEdDur>	250	250
<itemEdStart>	50	50

9.3 Videos

	Viz Pilot News 8.9 plugin	Viz Pilot Edge 3.0 / PDS 9.0
<objType>	PILOT	VIDEO
<objTB>	50	50
<objDur>	298	298

9.4 Stillstore Images

	Viz Pilot News 8.9 plugin	Viz Pilot Edge 3.0 / PDS 9.0
<objType>	PILOT	STILL
<objTB>	0	0
<objDur>	1	1

9.5 Overriding MOS XML Values

For backwards compatibility with existing newsroom system implementations, it is possible to override some of these values when served out from Pilot Data Server and using Viz Pilot Edge. This should only be done to overcome potential critical issues when upgrading Viz Pilot Edge or Pilot Data Server, and not be a permanent solution. These are the database parameters that can be overridden:

`mos_override_objtype`

`mos_override_objtb`

`mos_override_objdur`

See Database Parameters on how to add optional database parameters.

Copyright © 2026 Vizrt

10 Data Server Config

Data Server Config is the web-based administration interface for the Pilot Database Settings. It is used to configure the system-wide settings that Pilot Edge, Template Builder and other Vizrt clients read from the Pilot Data Server.

See Also

- Database Parameters
- Optional Parameters

You open it in a browser at:

`https://hostname:port/app/DataServerConfig/`

Replace `<hostname>` with the machine running Pilot Data Server, and the port if you have changed it from the default `8177`. If Single Sign-On is enabled, you are prompted to log in first.

Caution: These settings apply to every client connected to this Pilot Data Server. Changes should only be made by administrators. Some changes require a Pilot Data Server restart before they take effect.

10.1 Panels

Data Server Config is organized into panels, selectable from the segmented control at the top of the window:

Panel	Purpose
Parameters	The central list of Pilot Data Server database parameters (the "VCP parameters"). This is where you set core system values such as the Preview Server, Graphic Hub and Crop Service addresses.
Search Providers	Configure the asset search providers (for example Viz One) and the pre-authenticated hosts used when searching for and browsing media.
Tags	Administer the tags that are assigned to templates in Template Builder and used to organize templates in the clients.
Asset Guests	Register external asset sources ("guests") that appear as extra tabs in the Pilot Edge asset browser. Each guest has an ID, a display name and a URL.
Pilot Edge	Default user-interface settings for Pilot Edge, such as the default template sorting and which top-bar tabs are shown. This is the recommended place to configure Pilot Edge behavior for all users.

10.2 Parameters panel

The **Parameters** panel is the modern replacement for the old *VCP Parameters* page. It shows the Pilot Data Server database parameters as an editable list with a description for each parameter.

Copyright © 2026 Vizrt

Common tasks:

- **Filter** – type in the filter box to quickly locate a parameter by name.
- **Edit a value** – change the value directly in the list and click **Save**.
- **Add New Optional Parameter** – add a parameter that is not present by default (for advanced/optional settings).
- **Delete Selected Optional Parameter** – remove an optional parameter you previously added.
- **Reset all** – discard your unsaved edits and reload the values currently stored on the server.

A curated reference of the most relevant parameters for a modern Pilot Edge system is documented on the sub-page.

10.2.1 Most important settings for a Pilot Edge system

For a modern Pilot Edge deployment, these are the parameters you are most likely to need:

Parameter	What it does
`data_server_url`	The public URL of this Pilot Data Server (or the reverse proxy in front of it), e.g. `http://pds.example.com:8177`. This is the address clients and search providers use to reach PDS. Replaces the legacy `app_server` parameter.
`graphic_hub_url`	URL of the Graphic Hub REST API used to provide scenes, e.g. `http://gh.example.com:19398/`.
`preview_server_uri`	Base URL of the Preview Server used to generate preview images in the editor, e.g. `http://preview.example.com:54000`.
`crop_service_uri`	URL of the image Crop Service, e.g. `http://cropservice.example.com:8178/`.
`mos_plugin_url`	URL of the Pilot Edge HTML MOS plug-in. Trio will also use this as the default Pilot Edge Editor.
`env-vizrt-spellcheck-language`	Spell check language code.
`env-vizrt-spellcheck-path`	Relative path or URL to the spellcheck dictionaries.
`video_mode`	Channel video mode, `PAL` or `NTSC`.

Copyright © 2026 Vizrt

Note: Legacy ActiveX / Pilot News desktop parameters (the many `ax_*` options, MOS-plugin tuning, Object Store / VOS password settings, and the old Script Runner update service) are still present for backwards compatibility but are generally **not** relevant when running a modern Pilot Edge system.

10.3 Search Providers panel

Both the clients and the Pilot system read their asset search configuration from Pilot Data Server. On this panel you can:

- **Asset Search Providers** – add, rename, enable/disable and delete search providers by entering the provider's **Service Document URL** and a short **Label** shown in search results (for example a Viz One `thirdparty` service document).
- **Authenticated Hosts** – register hosts together with a **URL**, **username** and **password** so that searches and the Feed Browser can authenticate against them (typically required for Viz One).

Note: Changing a provider's short name, or changing pre-authenticated hosts, requires a Pilot Data Server (and Crop Server) restart before the change takes effect.

10.4 Tags panel

The **Tags** panel is used to administer the tags assigned to templates in Template Builder. Tags are then used in the clients to filter and organize the template list.

Function	Description
Add tag	Add a new tag name.
Rename tag	Select a tag, enter a new name and save.
Delete tag	Delete a selected tag. Deleting a tag does **not** delete the templates that reference it.
Filter	Type to filter the tag list by name.

10.5 Asset Guests panel

The **Asset Guests** panel registers external asset sources that appear as additional tabs in the Pilot Edge asset browser. Each entry consists of:

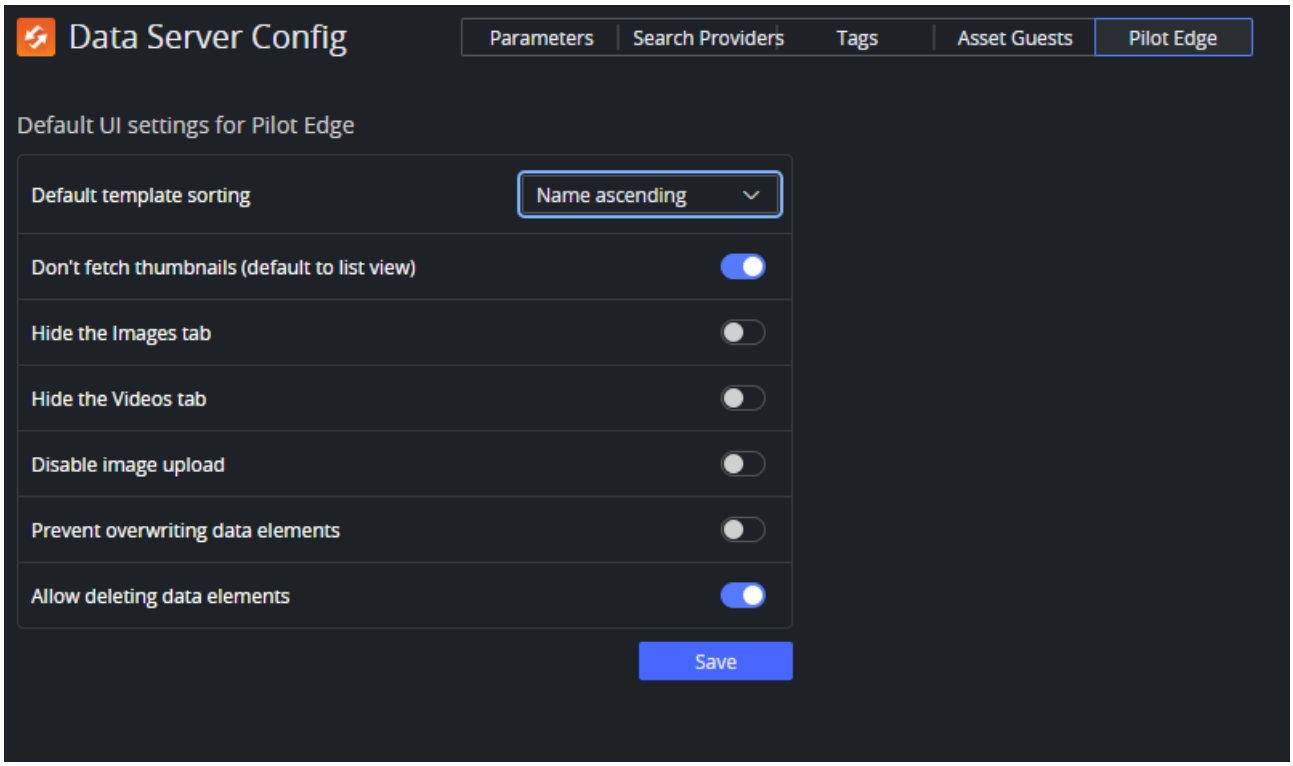
- **ID** – a unique identifier for the guest.
- **Name** – the label shown on the tab in the asset browser.
- **URL** – the address of the asset guest.

IDs must be unique and non-empty; the panel validates this before allowing you to save.

Copyright © 2026 Vizrt

10.6 Pilot Edge panel

The **Pilot Edge** panel configures the default user-interface behavior of Pilot Edge for all users on this server. Individual users can still override most of these afterwards in their own session.



Setting	Description
Default style	The style from the Template Builder Style Editor that Pilot Edge and Template Builder forms use by default. *Built-in style* uses no custom style. Stored in the `env-style` parameter, so the `$style` URL parameter can override it. See Styling.
Themes	The themes users can switch between in Pilot Edge and Template Builder, including the palettes of the Style Editor: **Dark and light** (default), **Dark only** or **Light only**. With only one theme, the theme toggle is hidden.
Default template sorting	The sort order used for the template list the first time a user opens Pilot Edge.

Copyright © 2026 Vizrt

Setting	Description
Don't fetch thumbnails (default to list view)	Show the template and element browser as a list without thumbnails by default.
Hide the Images tab	Hide the Images tab in the Pilot Edge top bar.
Hide the Videos tab	Hide the Videos tab in the Pilot Edge top bar.
Disable image upload	Disable uploading and pasting images into image fields.
Prevent overwriting data elements	Prevent users from overwriting existing data elements when saving (only *Save As* is available).
Allow deleting data elements	Allow users to delete data elements from the element browser.
Show playout instructions	Show the playout instructions when editing a graphic in Pilot Edge and in Template Builder run mode. On by default. Template scripting can override it per graphic with `playoutInstructions.showPlayoutInstructions` (`show`, `hide` or `useDefault`).
Hide tags in media search	Hide the tags filter when searching for images and videos in Pilot Edge. Off by default.
Hide tags in template search	Hide the tags filter when searching for templates in Pilot Edge. Off by default. Profiles that restrict the available tags still show it.
Viz Maps	Settings for the Viz Maps service used by the map editors in Pilot Edge and Template Builder, stored in the `vizmaps_*` parameters. See Viz Maps.

These options are stored in Pilot Data Server database parameters (for example `env-style`, `ax_hide_images`, `ax_hide_videos`, `ax_disableupload`, `ax_disable_data_overwrite`, `delete_data_from_activex`, and the `pilotedge_settings` value that holds the template sorting, themes, playout instructions and tag-hiding settings). You normally do not need to edit those parameters directly – use this panel instead.

Copyright © 2026 Vizrt

10.7 Database Parameters

The database parameters (also called *VCP parameters*) are stored in the Pilot database and served by Pilot Data Server to every connected client. You edit them from the **Parameters** panel in DataServerConfig.

Caution: These settings apply to every client connected to this Pilot Data Server and should only be changed by administrators. Some changes require a server restart before they take effect.

The parameters are grouped below by relevance:

- **Settings used by Pilot Edge and Template Builder** – the settings you configure for a modern system.
- **Newsroom and system settings** – newsroom (MOS) integration and system-level settings such as on-air updates and spell checking.
- **Legacy settings** – kept for backwards compatibility but not used by Pilot Edge or Template Builder.
- **Settings managed internally by Pilot Data Server** – the server reads or maintains these; do not change them unless instructed.

10.7.1 Settings used by Pilot Edge and Template Builder

These are the parameters that affect a modern Pilot Edge / Template Builder deployment. Several of the Pilot Edge UI toggles are more conveniently set from the **Pilot Edge** panel in Data Server Config, but are listed here for completeness.

Parameter	Description
`graphic_hub_url`	Host or reverse-proxy URL of the Graphic Hub REST API used to provide scenes (for example `https://<graphichubserver>:19398/`). May include a base-path prefix, and an explicit port if it differs from the default 19398.
`preview_server_uri`	Base URL of the Preview Server used to render preview images in the editor (for example `https://<previewserver>:4443/`). Published by Pilot Data Server in its service document.
`crop_service_uri`	URL of the image Crop Service used when cropping images in templates and data elements (for example `https://<cropservice>:8179/`).
`shared_curious_server`	Viz World (maps) server IP or host name. In a multi-server setup, use a comma-separated list of IPs/host names.

Copyright © 2026 Vizrt

Parameter	Description
`vizmaps_baseurl`, `vizmaps_apikey`, `vizmaps_language`, `vizmaps_style`	Settings for the Viz Maps service used by the map editors in Pilot Edge and Template Builder. Set from the **Viz Maps** section of the Pilot Edge panel. See Viz Maps.
`moseditor_arc_url`	URL of the Viz Arc MOS plug-in panel. When set, the Viz Arc actions tab is shown in Pilot Edge.
`moseditor_arc_schema`	Qualifier string that identifies the Viz Arc MOS editor when reopening Viz Arc items from the newsroom system. Default `http://www.vizrt.com/mosObj/vizarc/action`.
`moseditor_arc_label`	Title of the Viz Arc tab in Pilot Edge. Default *Viz Arc*.
`moseditor_mosart_url`	URL of the Mosart MOS plug-in panel. When set, the Mosart tab is shown in Pilot Edge.
`moseditor_mosart_schema`	Qualifier string that identifies the Mosart MOS editor when reopening Mosart items from the newsroom system. Default `http://www.mosartmedialab.no/schema/mositem.dtd`.
`moseditor_mosart_label`	Title of the Mosart tab in Pilot Edge. Default *Mosart*.
`ax_hide_images`	If true, the Images tab is hidden in the Pilot Edge top bar. Also configurable from the Pilot Edge panel.
`ax_hide_videos`	If true, the Videos tab is hidden in the Pilot Edge top bar. Also configurable from the Pilot Edge panel.
`ax_disableupload`	If true, uploading and pasting images into image fields is disabled in Pilot Edge. Also configurable from the Pilot Edge panel.
`ax_dont_fetch_thumbnails`	If true, the template and element browser default to list view without thumbnails. Also configurable from the Pilot Edge panel.
`ax_disable_data_overwrite`	If true, users cannot overwrite existing data elements when saving (only *Save As* is available). Also configurable from the Pilot Edge panel.

Copyright © 2026 Vizrt

Parameter	Description
`delete_data_from_activex`	If true, users may delete data elements from the Pilot Edge element browser. Also configurable from the Pilot Edge panel.
`pilotedge_settings`	JSON value holding default Pilot Edge UI settings under the `default` key: `default_template_sorting`, `theme_modes`, `show_playout_instructions`, `hide_media_tags` and `hide_template_tags`. Managed from the Pilot Edge panel – not normally edited by hand.
`collection_settings`	JSON value holding per-profile settings (default concept, available concepts and tags) used by Pilot Edge. Managed from the Template Builder profile editor.
`mosart_configuration`	JSON value holding the Mosart timing configuration. Each group can set a **Default Duration** (`defaultDuration`): the Mosart timing panel fills it in for an item without a duration, and applies it when the user selects the group. Managed from the Template Builder **Viz Mosart Configuration** dialog.
`order_mgt_uri_template`	URL template used to open the Viz One order-management form for image/asset orders.
`env-vizrt-spellcheck-language`	Default language used for spell checking (for example `en_US`). Replaces the legacy `spellcheck_dict_filename` parameter.
`env-vizrt-spellcheck-path`	URL that serves the dictionary files used for spell checking. Replaces the legacy `spellcheck_dict_path` parameter.
`env-style`	Name of the default style for template forms in Pilot Edge and Template Builder. Empty uses the built-in style. Set from the Template Builder Style Editor or the Pilot Edge panel; the `$style` URL parameter overrides it. See Styling.

10.7.2 Newsroom and system settings

These settings configure newsroom (MOS) integration and system-level behavior. They are relevant to a modern deployment but are set less often than the core settings above.

Copyright © 2026 Vizrt

Parameter	Description
`ax_MOSID`	Overrides the MOSID used in the MOS items produced for the newsroom system. Default `PILOT`.
`data_server_url`	Public URL of this Pilot Data Server, or of the reverse proxy in front of it (for example `http://exampleserver.com:8177`). May include a base-path prefix at the end of the URL (for example `http://exampleserver.com:8177/pds`). Clients and search providers use this address to reach the server. Replaces the legacy `app_server` parameter.
`mos_plugin_url`	URL of the Pilot MOS HTML plug-in that Trio/Director uses to open elements and templates that do not have a proper legacy form. Must point at the final (non-redirecting) Pilot Edge URL – for example `http://pds:8177/app/pilotedge/PilotEdge.html` rather than a URL that issues an HTTP redirect.
`live_update_interval`	Poll interval, in seconds, for update scripts. When *Update at regular intervals while on air* is enabled for a template, the Media Sequencer invokes the update service using this interval. Minimum 2 seconds; default 5.
`video_mode`	Channel video mode, `PAL` or `NTSC`. Also used by the server to derive MOS timing (see `mos_override_objtb`).

10.7.3 Legacy settings (not used by Pilot Edge or Template Builder)

These parameters are used by the legacy ActiveX Newsroom Component (Viz Pilot News / Director), the legacy Template Wizard, or Viz Object Store. They are kept for backwards compatibility and have **no effect** in a Pilot Edge / Template Builder system.

Parameter	Description
`activex_preview_timeout`	Timeout, in milliseconds, waiting for the Viz Preview Engine connection before continuing. Default 10000.
`allow_subfield_in_view`	Used by the legacy client to decide whether sub-fields can be added to custom layouts.

Copyright © 2026 Vizrt

Parameter	Description
`ax_dataelement_timer_enabled`	Default for showing/hiding the Graphic Event Timing options in the newsroom component when saving an element. Could be overridden per template in Template Wizard.
`ax_disable_clear_layers`	Disables clearing all layers as the first commands sent when sending to preview.
`ax_disable_dragdrop`	Disables drag-and-drop in the newsroom component (for newsroom systems that do not support it).
`ax_disable_media_drag`	Disables dragging media from search results to the rundown.
`ax_disable_overlay_saving`	Disables saving of overlay timelines for clip assets that have an `overlay_timeline` link.
`ax_enable_refresh_button`	Shows the refresh button in the ActiveX external preview form.
`ax_enableMediaSendToRundown`	Enables the *Add to Rundown* option in media search (for newsroom systems without drag-and-drop).
`ax_force_detached_pvw`	Forces a detached snapshot-preview window in the ActiveX/VCP client.
`ax_hide_data_elements`	Hides the data element list in the legacy newsroom component. Pilot Edge does not read this database parameter; the equivalent behavior is controlled by the `showpilotelements` URL parameter (`?showpilotelements=false`).
`ax_hide_media_tab`	Hides the media tab in the newsroom component.
`ax_include_mosobj_tag`	Adds a `mosobj` node to the MOS XML generated client-side. Required for some (Dalet) integrations.
`ax_mos_edit_save_returns`	Controls whether the ActiveX returns to the template list after saving (when editing).

Copyright © 2026 Vizrt

Parameter	Description
`ax_preview_host`	Host name of the Viz Engine used by the legacy client to generate snapshot previews and fetch resource icons.
`ax_preview_img_protocol`	Requests snapshots in different image formats.
`ax_preview_port`	Port of the Viz Engine used by the legacy client for snapshot previews and resource icons.
`ax_preview_tl_legacy_support`	Sends values to the legacy transition-logic location (`$other$object`) in addition to the normal behavior when previewing.
`ax_preview_tl_show_next_sub`	Uses the `\Toggle*continue_noanim` command when previewing transition logic.
`ax_remove_continue_count`	Disables sending `continueCount` in `mosExternalMetadata`.
`ax_remove_objSlug`	Removes `objSlug` from the MOS XML generated client-side (needed when hosted in AvidCommand).
`ax_required_version`	Minimum required version of the Vizrt newsroom component; a warning is shown if the client is older.
`ax_savesas_cancels_edit`	Changes *Save As New* behavior so the original item in the script is not replaced.
`ax_show_hints`	Turns tooltips in the ActiveX on/off (tooltips could crash the Dalet newsroom system).
`ax_showsavechanges_dlg`	Asks users whether to save the open element or template when opening another.
`ax_use_custom_gui_dlg`	Embeds the Viz Curious Maps Editor inside the newsroom component window.

Copyright © 2026 Vizrt

Parameter	Description
`clip_default_itemchannel`	Default channel for video clips, written to the `itemChannel` element of the MOS XML by the legacy client.
`country_language`	Language used for the legacy country list. English is supported by default.
`DELETE_DATA_PASSWORD`	Password protecting data-element deletion in the legacy client. Empty means no password.
`disable_built_in_vos_search`	Disables the built-in Viz Object Store search (used when moving to RESTful VOS search).
`image_order_uri_template`	Placeholder URL template for image orders.
`image_share`	Path the legacy image Crop Tool uses when saving a cropped image (UNC or Windows path).
`MediaSearch_itemPerPage`	Number of media-search items per page in the legacy client.
`mse_script_runner`	Overrides the (legacy VBScript) script runner used by the Media Sequencer. Superseded by the modern Update Service.
`script_runner_uri`	URL of the legacy VBScript script runner that executed update scripts. Superseded by the modern Update Service.
`show_playlist_element_index`	Shows an index column in the legacy VCP playlist.
`slim_mos_xml`	Reduces the client-generated MOS XML by removing optional tags.
`spellcheck_dict_filename`	Dictionary/affix file base name for the legacy spell checker (default `en_US`). Superseded by `env-vizrt-spellcheck-language` in Pilot Core Service.
`spellcheck_dict_path`	Path to the legacy spell-checker dictionary files. Superseded by `env-vizrt-spellcheck-path` in Pilot Core Service.

Copyright © 2026 Vizrt

Parameter	Description
`timeline_update_service`	Service-document URL for the legacy Timeline Update Service.
`VOS_PASSWORD`	Encrypted Viz Object Store password. Set it through the VOS change-password dialog, not here.
`vos_allow_edit_from_dll`	Enables/disables the Object Store toolbar for editing images and person information (`Y` / `N`).
`vtw_disable_unique_name_check`	Allows saving templates with the same name in different concepts in the legacy Template Wizard.

10.7.4 Settings managed internally by Pilot Data Server

Pilot Data Server reads or maintains these parameters itself. **Do not change them** unless a Vizrt representative instructs you to.

Parameter	Description
`app_server`	Legacy `hostname:port` of the running Pilot Data Server. Set by the server at startup and kept for backwards compatibility – use `data_server_url` instead.
`schema_version`	Major version of the database schema. Read by the server on startup. Do not modify.
`schema_build`	Minor build version of the database schema. Do not modify.
`vcp_schema_name`	Database schema name (for example `PILOT`). Do not modify.
`compatible`	Internal schema/version compatibility marker. Do not modify.
`model_uri_template`	URL template the server uses to build the model-document link for a template. Used by the Maps Preview Server and certain external systems. Do not change.
`payload_uri_template`	URL template the server uses to build the payload-document link for a data element. Used by certain external systems. Do not change.

Copyright © 2026 Vizrt

Parameter	Description
`croptool_max_image_area`	Maximum area (width × height, in pixels) of a cropped image served by the server; larger images are resized while keeping the crop aspect ratio. `0` = disabled. Safe for administrators to tune.
`ax_add_itemchannel`	When enabled, the server adds the element's default channel (`itemChannel`) to the generated MOS XML.
`ax_add_dbdata_to_mosexternal`	When enabled, the server adds the element's stored page data to the MOS item (used by some Mosart integrations).
`mos_override_objtb`	Overrides the `<objTB>` value in the MOS XML served by the server. Normally 50 or 60; setting it to 0 breaks the timing information in Playout Instructions.
`mos_override_objtype`	Overrides the `<objType>` value in the MOS XML for Pilot graphics elements. Default `PILOT` .
`mos_override_objdur`	Overrides the `<objDur>` value in the MOS XML for Pilot graphics elements. Default `1` .

Copyright © 2026 Vizrt

10.8 Optional Parameters

The **Parameters** panel ships with the standard set of database parameters, but you can add extra **optional** parameters that aren't present by default:

1. Open **Data Server Config > Parameters**.
2. Choose **Add New Optional Parameter**.
3. Enter the parameter **name**, an initial **value**, and an optional **description**.
4. Click **Save**.

To remove one you added, select it and choose **Delete Selected Optional Parameter**. Note that you must save or reset any pending edits before adding or removing optional parameters.

Optional parameters are stored and served exactly like the built-in ones, so any custom key you add is included in the settings document that clients fetch from Pilot Data Server.

10.8.1 `env-` parameters as environment variables

Any parameter whose name starts with `env-` is automatically exposed to **Pilot Edge** and **Template Builder** as an **environment variable**, with the `env-` prefix removed. For example, the VCP parameter `env-location` becomes the environment variable `location`.

You then reference that variable differently depending on where you use it:

Context	Syntax	Example (for `env-location`)
Template scripting (TypeScript)	`vizrt.$env.<name>`	`const loc = vizrt.$env.location ?? "default"`
Field/panel visibility expressions	`{{$<name>}}`	`{{$location}}==="oslo"`
Custom HTML (iframe) panel URLs	`{$<name>}`	`<https://myserver/panel.html?loc={$location}>`
Feed Browser URLs	`{$<name>}`	`<https://feeds.example.com/{$location}/latest.atom>`
Inline HTML fragments	`{{$<name>}}`	`Region: {{$location}}`

Two things to keep in mind:

- **Which braces:** URL-type fields (custom HTML panel URLs and Feed Browser URLs) use **single** braces — `{$location}`. Scripting-expression and inline-HTML-fragment contexts use **double** braces — `{{$location}}`. Actual TypeScript scripts use `vizrt.$env.location`.

Copyright © 2026 Vizrt

- A variable can appear anywhere within a string, for example `<https://myserver/{$location}/panel.html>`.

In addition to your `env-` parameters, the applications always provide the built-in variables `APP` (the application name), `PDS` (the Pilot Data Server URL) and `GH` (the Graphic Hub URL).

This lets administrators define site-specific configuration values once in the database and consume them across many templates without any code changes.

Copyright © 2026 Vizrt