



Preview Server Administrator Guide

Version 4.4



Preview Server



Copyright © 2020 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.

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

2020/05/28

Contents

1	Introduction.....	4
1.1	Related Documents.....	4
1.2	Feedback	4
2	About Preview Server	5
2.1	Basic Workflow.....	5
3	Requirements and Recommendations	7
3.1	Software Requirements	7
3.2	Hardware Requirements.....	7
3.3	Virtualized Environments	7
4	Installation and Configuration.....	8
4.1	Installation.....	8
4.1.1	Installing Preview Server	8
4.2	Configuration	9
4.2.1	Preview Server Service.....	9
4.2.2	Viz Engines	9
4.2.3	Service Configuration.....	9
4.2.4	Setting Up HTTPS.....	13
4.2.5	Viz Pilot Data Server.....	13
4.2.6	Configuring Preview Server Settings in Viz Trio	13
5	Working with Preview Server	14
5.1	Preview Server Web Interface.....	14
5.1.1	Home.....	14
5.1.2	Frame API	15
5.1.3	About	15
5.2	Common Procedures.....	15
5.2.1	Startup and Shutdown.....	15
5.2.2	Viz Engine Configuration	16
6	Troubleshooting	18
6.1	Troubleshooting Tips.....	18
6.1.1	Services Don't Start After Installing the Latest Preview Server	18
6.2	Known Limitations	18

1 Introduction

This Administrator Guide covers the configuration and operation of the Preview Server.

1.1 Related Documents

See the following documents at the [Documentation Center](#):

- *Graphics Plugin Administrator and User Guides*
 - *Viz Engine Administrator Guide*
 - *Viz Multichannel User Guide*
 - *Viz Pilot User Guide*
 - *Viz Trio User Guide*
-

1.2 Feedback

We welcome feedback on Vizrt products and documentation. Contact your local Vizrt customer support team at <http://www.vizrt.com>.

2 About Preview Server

Preview Server is a Windows service that manages a pool of **Viz Engine** processes. Client software, such as **Viz Multichannel**, **Viz Pilot**, and **Viz Trio**, send requests to **Preview Server** for snapshots of **Viz Engine** graphics. The content and animation in and out points of the graphics can be specified in the request.

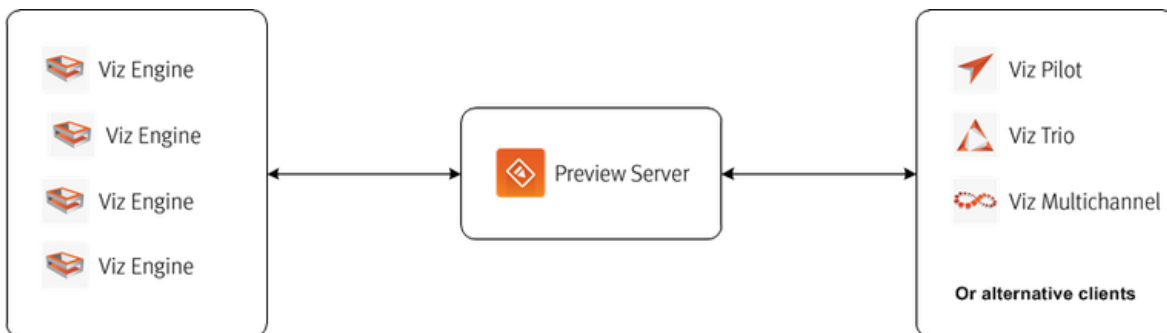
Preview Server provides snapshots of scenes during the creative preview process. The preview snapshots provide an indication of the look of the graphics that will ultimately be played out in high resolution on a **Viz Engine**.

Preview Server provides load balancing and can be used to create a redundant renderer pool.

Preview Server is a Windows service. Upon installation, the service is started and the startup type is set to *Automatic* by default. The service will then start every time the operating system is started, and will continue to run in the background as long as Windows is running. For more details, see [See the Status of the Preview Server Service](#).

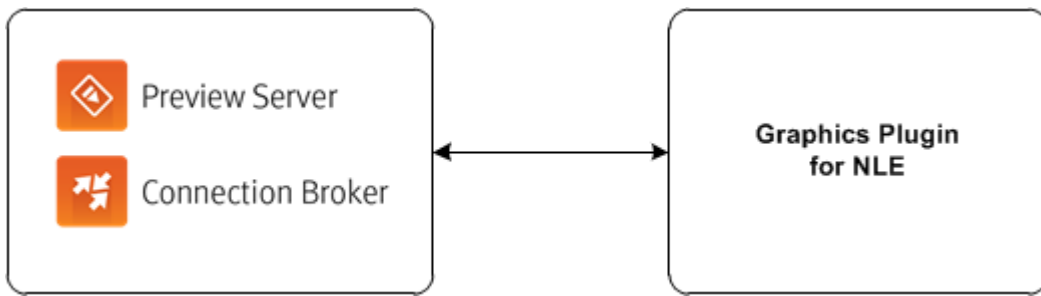
2.1 Basic Workflow

A typical use case is as follows: A **Viz Trio** user wants to add graphics to a video in **Viz Trio**'s Timeline editor. A video clip and one or more data elements (graphics) are dragged to the timeline. In the Timeline editor, the graphics elements are added as placeholders on the video timeline. Since the user wants to display snapshots of the graphics on these placeholders, a **Preview Server** connection has been set up (for details, see [Configuring Preview Server Settings in Viz Trio](#)). **Viz Trio** sends a request to **Preview Server** for an available **Viz Engine** to generate the necessary preview snapshots. The **Preview Server** passes the task to a suitable **Viz Engine**, which in turn generates the snapshots that were requested by **Viz Trio**.



When used together with Vizrt control applications like **Viz Multichannel**, **Viz Pilot**, and **Viz Trio**, **Preview Server** is installed from a standalone installer.

Alternatively, when used in a workflow together with the Graphics Plugin for NLE products, **Preview Server** is installed from a joint installer package, containing both **Preview Server** and **Connection Broker**.



For more details on this joint Preview Server and Connection Broker setup, see the *Graphics Plugin Administrator Guide*.


3 Requirements And Recommendations

3.1 Software Requirements

Microsoft .NET Framework 4.6 or later

3.2 Hardware Requirements


Licensed Viz Engine

 **Note:** Preview Server can run on the same machine as a Viz Engine, but it is recommended to specify additional Viz Engines in order to spread the load.

3.3 Virtualized Environments

Preview Server can run in a virtual environment. However, if you plan to run Viz Engine on the same virtual machine, make sure that the virtual machine is set up with the requirements to run Viz Engine, see chapter 4 and 5 in the [Viz Engine Administrator Guide](#).

Since Preview Server is configured via a [web browser](#) on localhost and the web browser and Preview Server must run on the same machine, you must use remote desktop to configure your system.

 **Tip:** When running in a virtual environment it can be convenient to configure preview server for several Viz Engines.

See Also

- [Preview Server Release Notes](#)

4 Installation And Configuration

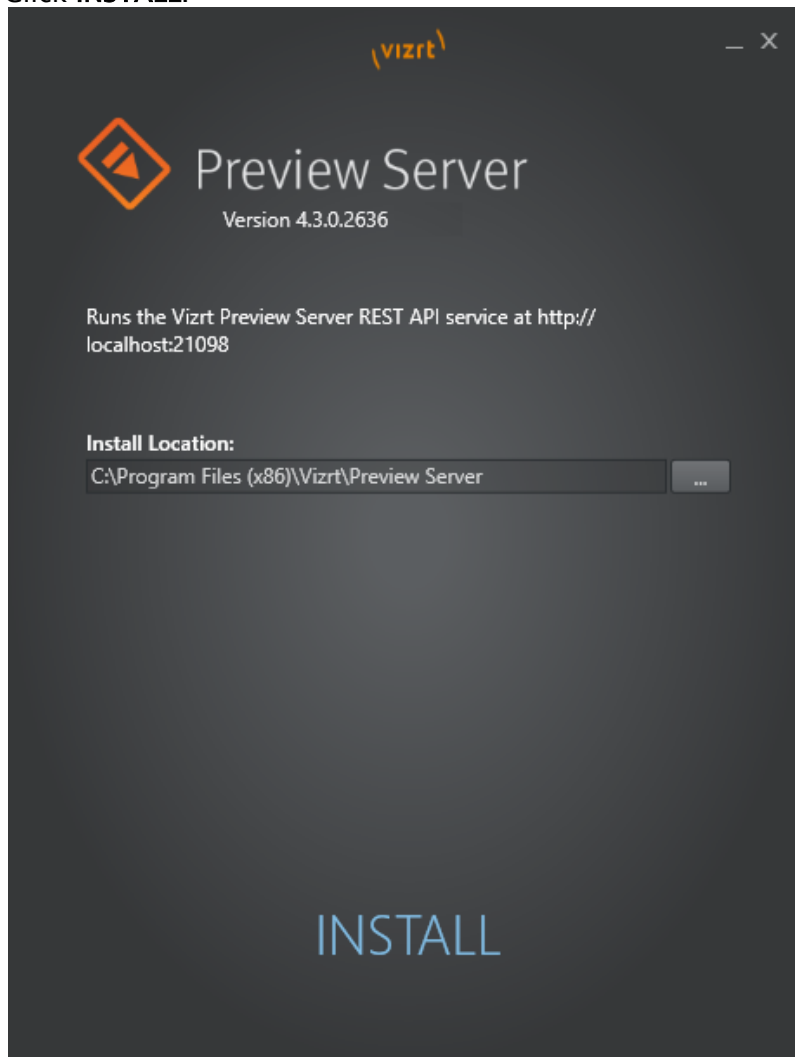
This section covers the following topics:

- [Installation](#)
 - [Configuration](#)
-

4.1 Installation

4.1.1 Installing Preview Server

1. Run the Preview Server installer file (*.exe) to start the *Preview Server Setup*.
2. Make sure that the **Install Location** is correctly defined.
3. Click **INSTALL**.



4. When the installation is complete, click **FINISH**.

Note: The Preview Server service is automatically started, and the startup type is set to *Automatic*, meaning that the service will start every time the operating system is started, and then run in the background for as long as Windows is running. For more details, see [See the Status of the Preview Server Service](#).

4.2 Configuration

This section covers the following topics:

- [Preview Server Service](#)
- [Viz Engines](#)
- [Service Configuration](#)
- [Setting Up HTTPS](#)
- [Viz Pilot Data Server](#)
- [Configuring Preview Server Settings in Viz Trio](#)

4.2.1 Preview Server Service

Preview Server is a Windows Service that sets up an HTTP REST service on port 21098 on the host on which it is running. This service accepts requests for graphical snapshots of playout elements from a Viz Engine.

4.2.2 Viz Engines

Although Preview Server can run on the same machine as a Viz Engine (also used for the data element preview in Viz Pilot's newsroom client), it is also possible to specify additional Viz Engines in order to spread the load. Frame requests will be balanced across available Viz engines in a way that will minimize the memory load of each engine. The pool of Viz Engines is configured in the [Preview Server Web Interface](#) Home page.

Note: Scaling of the system should be monitored to avoid excessive client waiting time and any potential overload of the Viz Engines.

4.2.3 Service Configuration

You can modify the Preview Server service's behavior using the following settings in the `PreviewServer.exe.config` file available in the [install location](#):

`VizEnginePollIntervalMilliseconds`

This setting determines how often a render engine's availability is checked.

`VizEngineTimeoutMilliseconds`

After the specified amount of milliseconds, the render engine list stops attempting to obtain the availability status and declares the engine unresponsive.

CacheMaxAge

Frames are cached in memory, allowing quicker responses when requesting frames that have been requested previously. By default, frames are cached for a maximum of one day. You can set a different time limit by modifying this setting, which specifies the maximum number of seconds to retain cached results.

VizEngineCleanupIntervalMilliseconds

Once a scene in Graphic Hub is modified, all cached snapshots using that scene must be evicted from the cache. This interval specifies how often the scene modification dates are checked to detect scene changes.

UseRunLengthEncoding

It's recommended to activate run length encoding if relatively high-resolution snapshots are requested.

CorsAllowedOrigin

This setting can be used to restrict the allowed cross-origin resource sharing pattern utilized by Preview Server.

DebugMode

Debug mode enables extended logging (for troubleshooting).

ServicePort

The service port setting lets you change the port number for regular HTTP requests for the Preview Server service. The service does not accept any HTTP requests if 0 is specified.

SecurePort

The secure port setting lets you change the port number for secure HTTPS requests for the Preview Server service. The service does not accept any HTTPS requests if 0 is specified.

Redraws

This number specifies how many redraws a renderer must perform before taking a snapshot of the graphic.

Note: Although increasing the redraw number slows down the overall performance of snapshot requests, it might be required for some scenes, for example when using data pool plugins to render correctly.

ExternalBaseUrl

Changes the default URL base for snapshot requests, for example. This setting also overrides any Host, X-Forwarded-Proto and X-Forwarded-Port headers in requests. An empty string uses the default URL base and enables recognition of such headers in requests.

AlwaysRunPreviewScript

If set to True running OnPreview script hooks in the scene before taking a snapshot is enabled. Note that this might delay snapshot requests for scenes containing such hooks.

OpenIdConnectServerUrl

The URL to the OpenId Connect server that is trusted to handle user authentication. When this is set, all REST communication with the server must include an Authorization HTTP header value with a valid bearer token signed by this OpenId Connect server.

ValidAudience

When `OpenIdConnectServerUrl` is set, the bearer token must contain this value in its list of valid audiences. Leave empty to allow any authenticated audience.

AdministrationFromLocalhostOnly

If set to `True` this setting will lock down the administration REST APIs (including documentation and web pages) so that they can only be used from the machine on which this server is running. This is set to true by default.

The following is an example of a Configuration Section:

```

<configuration>
  <applicationSettings>
    <Vizrt.Frames.Properties.Settings>
      <setting name="VizEnginePollIntervalMilliseconds" serializeAs="String">
        <value>5000</value>
      </setting>
      <setting name="VizEngineTimeoutMilliseconds" serializeAs="String">
        <value>1000</value>
      </setting>
      <setting name="CacheMaxAge" serializeAs="String">
        <value>86400</value>
      </setting>
      <setting name="VizEngineCleanupIntervalMilliseconds"
serializeAs="String">
        <value>60000</value>
      </setting>
      <setting name="UseRunLengthEncoding" serializeAs="String">
        <value>False</value>
      </setting>
      <setting name="CorsAllowedOrigin" serializeAs="String">
        <value>*</value>
      </setting>
      <setting name="DebugMode" serializeAs="String">
        <value>False</value>
      </setting>
      <setting name="ServicePort" serializeAs="String">
        <value>21098</value>
      </setting>
      <setting name="SecurePort" serializeAs="String">
        <value>4443</value>
      </setting>
      <setting name="Redraws" serializeAs="String">
        <value>0</value>
      </setting>
      <setting name="ExternalBaseUrl" serializeAs="String">
        <value />
      </setting>
      <setting name="AlwaysRunPreviewScript" serializeAs="String">
        <value>False</value>
      </setting>
      <setting name="OpenIdConnectServerUrl" serializeAs="String">
        <value>https://ourstory.vizrt.internal:7500</value>
      </setting>
      <setting name="ValidAudience" serializeAs="String">
        <value>api.vizrt.com</value>
      </setting>
      <setting name="AdministrationFromLocalhostOnly" serializeAs="String">
        <value>True</value>
      </setting>
    </Vizrt.Frames.Properties.Settings>
  </applicationSettings>

```

```
</configuration>
```

4.2.4 Setting Up HTTPS

- To open the HTTPS port, run the following command once for each port you want to test: `netsh http add urlacl url=https://+:4443/ user=YOURUSERNAME`
- You can change the https port by editing the `.config` file and changing the `SecurePort` value.
- In addition, you must set the SSL certificate for each port with this command: `netsh http add sslcert ipport=0.0.0.0:4443 certhash=HASH_WITHOUT_SPACESappid={81e86c2d-63c0-4841-b72c-2189604b4245}`

To see if you have a hash that can be used for the command above, run one of the following commands:

```
certutil -store My
or
powershell -Command Get-ChildItem -path cert:\LocalMachine\My
```

4.2.5 Viz Pilot Data Server

To grant applications connected to the Viz Pilot database access to Preview Server, see the [Pilot Data Server Administrator Guide](#).

4.2.6 Configuring Preview Server Settings in Viz Trio

If Preview Server is used together with Viz Trio, some Preview Server settings must be defined in Viz Trio:

1. In **Trio Configuration**, navigate to the **Connectivity > Viz One** panel.
2. Enter the hostname and port number in the **Preview Server Host** box (format `hostname:port`). This enables snapshots of graphics to be displayed as video overlays in Viz Trio's Timeline editor.

To enable the Preview Server to load actual thumbnails in the playlist, rather than displaying the default template thumbnails throughout, perform the following steps:

1. In **Trio Configuration**, navigate to the **User Interface > Page List/Playlist** panel.
2. Enable the **Use Preview Server to load thumbnails** box.

For more details, see the *Viz Trio User Guide*.

5 Working With Preview Server

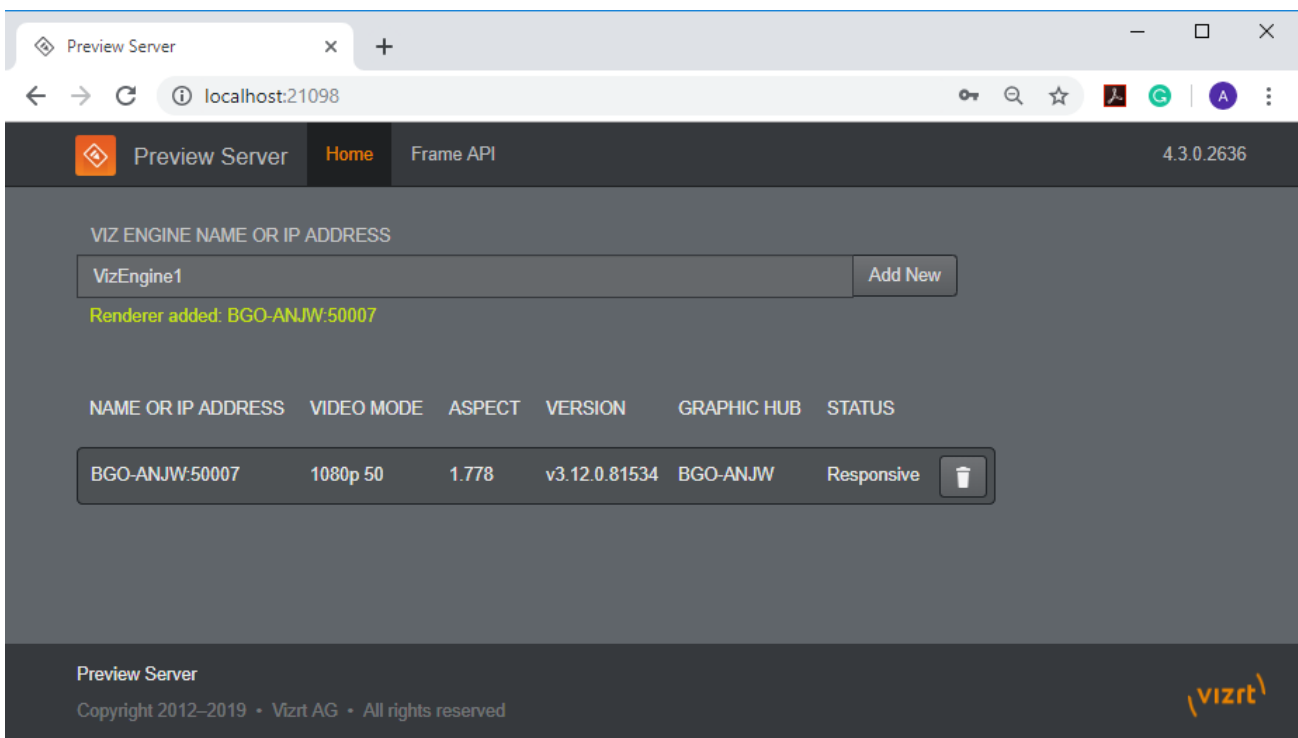
This section covers the following topics:

- [Preview Server Web Interface](#)
- [Common Procedures](#)

5.1 Preview Server Web Interface

Preview Server has a web interface for accessing features.

For details on how to open this interface, see [Accessing the Preview Server Web Interface](#).



The Preview Server web interface contains a menu with links to the following resources:

- [Home](#)
- [Frame API](#)
- [About](#)

5.1.1 Home

You can add, monitor and remove Viz Engines from a pool of renderers on the **Home** page. Preview Server can interact with one or more Viz Engines.


See Also

- [Viz Engine Configuration](#)

5.1.2 Frame API

Click the **Frame API** link in the Preview Server web interface to open the **REST Frame API** documentation, which describes the REST (Representational State Transfer) API over HTTP, provided by the Preview Server for requesting rendered frames from a Viz Engine.

The document also lists the *Resources* and *Data Types* that are used in the interface.

 **Example:** `http://<hostname>:21098/doc/rest_manual`

5.1.3 About

Clicking the *Preview Server* or *version number* link in the Preview Server web interface opens the **About Preview Server** window.

This window contains information about the installed Preview Server version. A link to the list of *third-party component credits* is also available here.

5.2 Common Procedures


This section covers the following topics:


- [Startup and Shutdown](#)
- [Viz Engine Configuration](#)

5.2.1 Startup and Shutdown

Accessing the Preview Server Web Interface

- In the **Start** menu search for **Preview Server Config**, or
- Double-click the **Preview Server Config** desktop icon (if a shortcut has been created), or
- Navigate to the Preview Server web interface in a browser, using the hostname of the machine running the Preview Server, and port *21098*.

 **Example:** `http://<hostname>:21098/`

 **Note:** The default web browser is used when opening the Preview Server web interface from the Windows Start menu or desktop.

See the Status of the Preview Server Service

1. In the **Start** menu search for **Services**.
2. Scroll down to the **Vizrt Preview Server** service.
The status of the Preview Server service is listed in the **Status** column.

Starting the Preview Server Service

1. Perform the steps in [See the Status of the Preview Server Service](#).
2. Make sure that the Preview Server status is empty.
3. Right-click the Preview Server entry in the list.
4. From the menu that appears, click **Start**.

Note: Upon installation, the Preview Server service is started, and the startup type is set to *Automatic*. This means that the service will start every time the operating system is started, and will continue to run in the background as long as Windows is running. However, if the Preview Server service has been manually stopped, the service can be restarted by following the steps in this procedure.

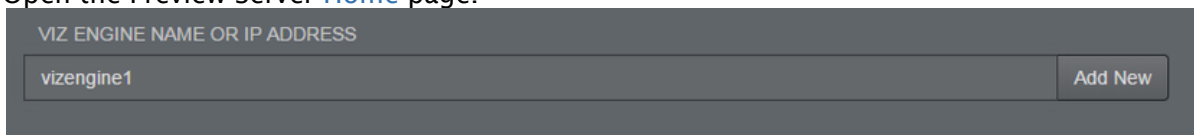
Stopping the Preview Server Service

1. Perform the steps in [See the Status of the Preview Server Service](#).
2. Make sure that the Preview Server status is set to *Started*.
3. Right-click the Preview Server entry in the list.
4. From the menu that appears, click **Stop**.

5.2.2 Viz Engine Configuration

Adding a Viz Engine

1. Open the Preview Server [Home](#) page.



VIZ ENGINE NAME OR IP ADDRESS

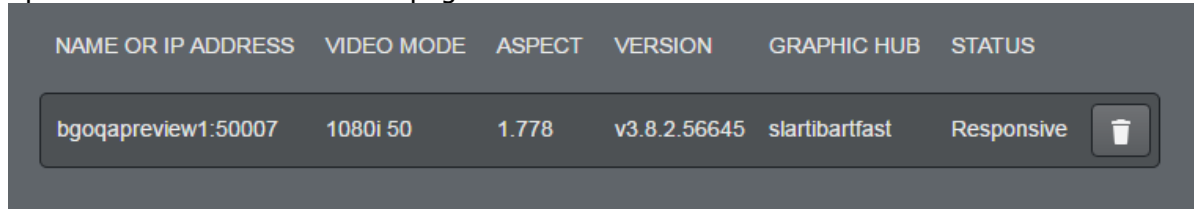
vizengine1 Add New

2. Enter the Viz Engine hostname or IP address in the text box.
3. Click the **Add New** button.

Note: If not specified, the Viz Engine is added with the default port number *50010*. When using Viz Engines in a so-called dual channel setup, it is possible to run multiple instances of the Viz Engine on a single host. Each instance uses a unique port, which means that two Viz Engines should only be considered duplicates when both the hostnames and port numbers are identical. When referring to Viz Engines in a dual channel setup, type the port number after the hostname (for example *VizEngine1:51010*).

Removing a Viz Engine

1. Open the Preview Server [Home](#) page.



NAME OR IP ADDRESS	VIDEO MODE	ASPECT	VERSION	GRAPHIC HUB	STATUS	
bgoqapreview1:50007	1080i 50	1.778	v3.8.2.56645	slartibartfast	Responsive	

2. Click the **trash can** next to a Viz Engine entry.

6 Troubleshooting

If you don't find answers to your issue here, please contact Vizrt support at <https://www.vizrt.com/support>.

6.1 Troubleshooting Tips

6.1.1 Services Don't Start After Installing the Latest Preview Server

- Machines with Apple's Bonjour installed sometimes do not permit Preview Server to be installed. Uninstalling Bonjour may resolve this.
-

6.2 Known Limitations

- It's possible to add the same renderer multiple times on the Preview Server [Home](#) page if you use different spellings. For example, a renderer can be added by entering its hostname, and if the IP address is subsequently entered, the renderer will be added again. Adding the same renderer twice will produce incorrect snapshots.
- Snapshots of scenes with no key defined will result in empty images.
- Adding video mode Viz Engines as renderers could produce incorrect results in scenes using ControlVideo.

See Also

- *Preview Server Release Notes*