Graphics Plugin Administrator Guide

Version 3.0

Disclaimer

Vizrt provides this publication "as is" without warranty of any kind, either expressed or implied. his 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

2025/06/04

Contents

1   Introduction

Graphics Plugin integrates Vizrt's graphics workflow into the *Non-Linear Editing (NLE)* workflow. Use your normal workflow with your preferred NLE system, while adding real-time rendered Vizrt graphics.

Plugins are included for a variety of NLE systems:

- Adobe Premiere Pro
- Avid Media Composer
- Grass Valley Edius

Although Graphics Plugin versions are tailored to specific platforms, the overall setup and user interface is the same.

> ⚠️ **Note:** Viz Trio, Viz Pilot Edge and Viz Pilot, are the editors that allow NLE users to interact with the Vizrt graphics workflow and edit graphics elements. Graphics Plugin supports NLE systems on both Windows and Apple platforms. Viz Trio, Viz Pilot Edge and Viz Pilot are available for Windows-based NLE Systems. Viz Pilot Edge and remote access to Viz Pilot, are also available for Mac-based NLE Systems.

1.1   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.

1.2   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

2 Requirements and Recommendations

2.1 Supported NLE Systems
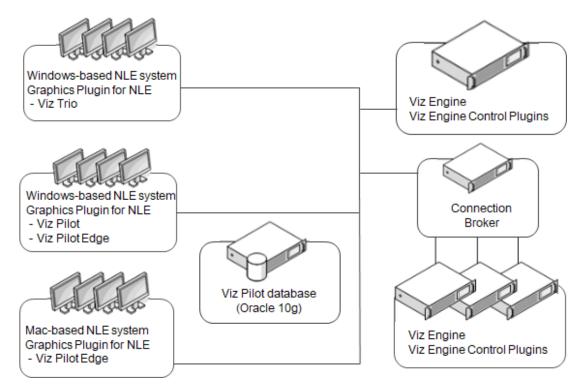
NLE Plugin	Available Editors	Plugin Version	Metagraphics Support	Stop Point Editing
Graphics Plugin for Avid	Viz Pilot, Viz Pilot Edge, Viz Trio (Win), Graphics Plugin Editor (Mac - not supported on macOS Catalina and newer versions.)	3.0.2	**Mac** - No **Win** -Export to Viz One only Requires Avid Assistant 1.16.1 Trio does not support metagraphics.	Yes
Graphics Plugin for Adobe	Viz Pilot, Viz Pilot Edge, Viz Trio (Win), Graphics Plugin Editor (Mac - not supported on macOS Catalina and newer versions.)	2.4.1	**Yes**. Win and Mac full support. Export to / import from Viz One. Import requires Adobe Assistant 3.2. Trio does not support metagraphics.	Yes
Graphics Plugin for Edius 8 and 9	Viz Pilot, Viz Pilot Edge, Viz Trio	2.3.1	No	No
Graphics Plugin for Edius X	Viz Pilot, Viz Pilot Edge, Viz Trio	2.5.0	No	No

2.2   Hardware Requirements

All installations require a Viz Engine in combination with a database (either a Media Sequencer, used by Viz Trio, or a Viz Pilot Database used by Viz Pilot or Viz Pilot Edge). Several NLE systems are supported on both Windows and Mac, and the setup is basically the same for both platforms. The Graphics Plugin configuration tool is used to configure the plugin, Viz Engine, Media Sequencer and editors.

> ⚠️ **Note:** Advanced setups use Connection Broker to select engines from a pool of Viz Engines. See Connection Broker for more information about this setup.

2.2.1   Overview of NLE System Setups

Viz Engine is a real-time graphics renderer, and at the core of the Vizrt graphics workflow. Viz Engine must handle multiple simultaneous requests. In Viz Engine, the multiplexing functionality is a built-in feature.

Connection Broker performs connection brokering, providing all Graphics Plugin system users with a single connection point for graphics rendering.

Viz Pilot database is an Oracle 10g or 11g database that stores template data and references to the template graphics. The Viz Pilot database is mainly used in conjunction with Viz Pilot.

2.2.2   Viz Trio, Viz Pilot Edge and Viz Pilot

Viz Trio is the Graphics Editor for browsing and previewing Viz Trio templates and pages. Support is only available on Windows.

Viz Pilot Edge and **Viz Pilot** support browsing, previewing templates and data elements stored on the Viz Pilot database. Viz Pilot Edge is web-based, and compatible with both Windows and Mac OS. Viz Pilot is Windows based, but may be accessed remotely when using Mac OS systems.

> ⚠️ **Note:** Effects are bound to the editor that made them, even if you subsequently switch to a different editor. This binding happens the first time **Editor** is clicked for a specific effect, not when the effect is dragged onto the timeline.

> ⚠️ **Note:** NLE system-specific installations are outside the scope of this guide.

2.3   Software Requirements

This section covers the following topics:

- Client Software Requirements
- Server Software Requirements
- Software Limitations

2.3.1   Client Software Requirements

Graphics Plugin has been tested with the following software:

Adobe 2.4

> ⚠️ **Note:**
> - The metagraphics import workflow is only possible with Premiere Pro New World option turned **off** when using Premiere Pro 14.0.4 or older versions. It is required to have Graphics Plugin for Adobe version 2.3.0 and Adobe Assistant 3.0 or newer.
> - The metagraphics import workflow is only possible with Premiere Pro New World option turned **on** when using Premiere Pro 14.2 or newer. It is required to have Graphics Plugin for Adobe version 2.3.2 and Adobe Assistant 3.2 or newer.
> - The metagraphics import workflow is not possible in Premiere Pro 14.1.

- Graphics Plugin for Adobe on Windows:
  - Microsoft Windows 10, 64-bit
  - Adobe Premiere Pro CC 2021 or newer (please check the previous release notes for version specific compatibility issues)
  - Viz Pilot 6.0.0 or newer
  - Viz Pilot Edge 1.5 or newer
- Graphics Plugin for Adobe on Mac:
  - Apple macOS 12 or newer
  - Adobe Premiere Pro CC 2021 or newer (please check the previous release notes for version specific compatibility issues)

Avid 3.0

- Graphics Plugin for Avid on Windows:
  - Microsoft Windows 10, 64-bit
  - Avid Media Composer 2021 (up to 2024.10)
  - Viz Pilot 6.0.0 or newer
  - Viz Pilot Edge 1.5 or newer
- Graphics Plugin for Avid on Mac:
  - Avid Media Composer 2021 (up to 2024.10)
  - Apple macOS 14 or newer

EDIUS 2.5

- Microsoft Windows 10, 64-bit or newer
- EDIUS X
- Viz Pilot 6.0.0 or newer

2.3.2   Server Software Requirements

The server components used by Graphics Plugin have different system requirements. Although a summary is provided here, please refer to the individual release notes and manuals for details.

> ⚠ **Note:** It is not recommended to install all components on a single server.

- **Connection Broker 4.0**
    - Optional, it is still possible to specify a specific Viz Engine directly.
    - Requires Microsoft .NET 4.5.
    - It can be installed on the same server as a Media Sequencer or a Viz Engine, or on a separate computer.
    - Connection Broker 1.0.0 can be used, but some of the new features in Viz Engine 3.5 will not work. This may impact performance.
    - Connection Broker 1.1 can be used but some of the new features in Viz Engine 3.7 will not work.
- **Media Sequencer 2.0.3 and 3.0.0**
    - Requires Microsoft .NET 4.0.
    - Connecting multiple Graphics Plugin workstations to the same Media Sequencer makes it easier to share graphics elements between Graphics Plugin clients.
    - Installing it on a Viz Engine server is not recommended since high memory usage can interfere with Viz Engine performance.
- **Viz Pilot 6.0.0, Pilot Data Server 6.0.0**
- **Viz Engine 3 / Viz Artist 3**
    - Viz Engine 3.7.0 or newer
    - If you need to use older versions the recommended versions are 3.2.8, 3.3.1 and 3.5.1.
    - Viz Engine 3.7 or newer is required to take advantage of the new aspect independent frame fetching.
    - Viz Engine 3.5 or newer is required to take advantage of the new custom resolution frame fetching and anti-aliasing.
    - Viz Engine 3.5.0, 3.3.0, and 3.2.7 or older have some known issues that can cause (rare) rendering artifacts when multiple NLE clients are using the same engine. We recommend Viz Engine 3.2.8, 3.3.1 and 3.5.1 or newer.
    - It is highly recommended to have a fast (Gigabit) network connection between each NLE workstation and Viz Engine.
    - This version of Graphics Plugin only supports UTF-8 encoding for Viz Engine.

2.3.3   Software Limitations

There are certain limitations as to what a Graphics Plugin solution can do compared to a full SDI in-and-out Viz Engine renderer.

General Limitations on Windows and Mac

Media Sequencer 1.19 or newer, Viz Engine 3, and Connection Broker 3.0 or newer, do not announce their presence on the computer network. Therefore, the Graphics Plugin configuration tool does not automatically discover these services.

Specific Limitations on Windows

For Avid installations, not all Graphics Plugin versions prior to 1.5 are automatically removed when upgrading.

Specific Limitations on Mac

Only Viz Pilot Edge, or a remote connection to Viz Pilot running remotely, can be used as graphics editors on Mac OS NLE Systems.

> ⚠️ **Note:** For more details on software limitations, see the Graphics Plugin Release Notes.

2.4   Supported Operating Systems

Graphics Plugin supports two operating systems:

- Microsoft Windows 10 or later.
- Mac OS X 12 or later.

Although the basic setup is in principle the same, there are some differences in how the systems are used and what software is used in addition to the plugin.

2.4.1   Windows-based Systems

Three editors are available for adding graphics to a video clip: Viz Trio, Viz Pilot and Viz Pilot Edge. Viz Trio is bundled with the Graphics Plugin installer, whereas Viz Pilot must be installed separately, as it connects to the Viz Pilot database for template data and graphics. Viz Pilot Edge is a web application that is accessed through the Graphics Plugin by adding the URL in the Configuration Tool.

2.4.2   Mac-based Systems

There are two editors available for adding graphics to a video clip:

- Viz Pilot Edge, a web application that is accessed through Graphics Plugin by adding the URL in the Configuration Tool.
- Viz Pilot can be used remotely as an editing tool by configuring the host machine and details of the Viz Pilot executable.

See also

- The Graphics Plugin User Guide for more information on how to use the different editors.
- Hardware Requirements

2.5   Supported NLE Systems

Versions other than those listed here are not officially supported, unless expressly stated in the documentation.

The following systems are supported:

- Adobe Premiere Pro
- Avid Media Composer
- Grass Valley EDIUS
- Apple Final Cut Pro
- Quantel Rio

2.5.1   Adobe Premiere Pro

Graphics Plugin for Adobe has been tested against, and supports the following versions:

- Premiere Pro CC 2024 (24)
- Premiere Pro CC 2023 (23)

⚠️ **Note:** This plugin works on both Windows and Mac machines.

2.5.2   Avid Media Composer

Graphics Plugin for Avid supports the following versions:

- Avid Media Composer 2024 (up to 2024.10)
- Avid Media Composer 2023
- Avid Media Composer 2022
- Avid Media Composer 2021

⚠️ **Note:** This plugin works on both Windows and Mac machines.

The following systems support the Avid Extensions (AVX) 2.0 System Development Kit (SDK):

- Avid Digital Studio (DS).
- Media Composer (Older Release and Post Adrenaline HD release that can run software only), Media Composer Adrenaline, and Media Composer Adrenaline HD.
- Symphony and Symphony Nitris.
- Xpress, Xpress Pro, and Xpress Pro HD.

2.5.3   Grass Valley EDIUS

Graphics Plugin for EDIUS supports the following versions:

- EDIUS X

⚠️ **Note:** The latest version supporting Edius 8 and 9, is version 2.3.1.

> ⚠️ **Note:** This plugin only works on Windows machines.

2.5.4   Apple Final Cut Pro

Graphics Plugin for Final Cut works together with Apple Final Cut Pro X. This plugin only works on Mac-machines.

2.5.5   Quantel Rio

Graphics Plugin for Quantel works together with Quantel Rio 4.3, or later. Version 2.0 of Graphics Plugin supports earlier versions of Quantel QEffects/Rio (version 4.2).

> ⚠️ **Note:** This plugin only works on Windows machines.

2.6   Network Requirements

A Viz Engine can be shared by multiple simultaneous NLE clients; however, there is no recommended number of users for SD and HD as this usually involves a performance issue and is not a fixed limitation.

Windows and Mac-based NLE system setups typically operate on a single Viz Engine. Since this requires transportation of frames (images) over the network, a fast network connection is recommended - preferably a gigabit switched network connection; however, a megabit (100 Mbit/s) connection will also work as the plugin supports lossless data compression using Portable Network Graphics (PNG) or Run-length Encoding (RLE) instead of uncompressed raw images.

> ⚠️ **Note:** RLE is only supported by Viz Engine 3.2.2 or later.

2.6.1   Port Numbers

The table below describes all listeners and port numbers, and applications that use the port numbers to establish a connection:

Listener	Port	Description
Viz Engine	6100	Default port used for graphics rendering.
Viz Trio	6210	Used by the Graphics Plugin to establish a connection to Viz Trio.
Viz Pilot	6220	Used by the Graphics Plugin to establish a connection to Viz Pilot.
Viz Pilot Edge	6230	Used by the Graphics Plugin to establish a connection to Viz Pilot Edge.
Graphics Plugin Config	6240	Used by the Graphics Plugin to establish a connection to the Graphics Plugin configuration tool.
Media Sequencer	8594	Port used by all clients that connect to the Media Sequencer.
Viz Engine	14300	Alternative port used to avoid conflicts with port `6100`. Port `6100` is normally used by renderers that are On Air, hence, it is (for example when running Viz Pilot version 4) recommended to use another port.

Listener	Port	Description
Viz Engine (multiplexed)	50007 50008 50009	Used by clients that connect to a multiplexed Viz Engine. `50007` : Isolated port. All connections to this port get their own session. The port is used by the Graphics Plugin to deliver frames to the host NLE system. For example: rendering or scrubbing of video clips with graphics. Using this port will also suppress bounding box commands. Cannot be used by Viz Pilot. `50008` : Shared port. All connections from one single host share one session. Used by Viz Trio, Viz Pilot and Viz Pilot Edge to display preview frames. `50009` : Fixed port. Same as shared port, except that allocated resources are never cleared from memory. Used by Viz Pilot 4.
Viz Engine (Dual Channel)	50107 50108 50109	Second default port for a dual channel Viz Engine. `50107` : Isolated port. All connections to this port get their own session. The port is used by the Graphics Plugin to deliver frames to the host NLE system. For example: rendering or scrubbing of video clips with graphics. Using this port will also suppress bounding box commands. Cannot be used by Viz Pilot. `50108` : Shared port. All connections from one single host share one session. Used by Viz Trio, Viz Pilot and Viz Pilot Edge to display preview frames. `50109` : Fixed port. Same as shared port, except that allocated resources are never cleared from memory. Used by Viz Pilot 4.
Connection Broker	21098	Connection to the Connection Broker configuration interface (for example http://localhost:21098/).

3   Installation and Configuration

The following sections describe configuration procedures. The sections follow the order of a new installation.

- Graphics Rendering
- Connection Broker
- Windows-based NLE Systems
- Mac-based NLE Systems
- Configuring Metagraphics
- Configuring Server-side Rendering for EDIUS
- Connecting to Viz Pilot Edge with HTTPS

3.1   Graphics Rendering

A successful connection to Viz Engine must be established for a NLE system to render graphics. Off-screen rendering is the default rendering mode for Viz Engine.

3.1.1   Setting Up a Viz Engine for Graphics Rendering

- Start the installer and select **VGA** mode.

3.2   Connection Broker

Connection Broker is a component that manages a pool of Viz Engines. Clients can query it for a Viz Engine that matches a specific video mode. Connection Broker provides load balancing and video mode matching, and can be used to create a redundant renderer pool.

- Open the interface using the address http://localhost:21098/. It is possible to open Connection Broker configuration from other machines. In order to do so, replace "localhost" with the actual hostname:
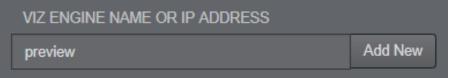

> ⚠️ **Note:** Preview Server is bundled together with Connection Broker. Although both components have their own service document URL, they share the same pool of Viz Engines. If both services are used extensively, the load on the joint pool may be higher than recommended.

This section describes the following procedures:

- Adding a Viz Engine
- Removing a Viz Engine
- Modifying the Ping Interval from Connection Broker to Viz Engine
  - Code Example

3.2.1   Adding a Viz Engine

1. In the Connection Broker web interface, enter the Viz Engine hostname or IP address in the text box.

2. Click **Add New**.

> ⚠️ **Note:** If not specified, Viz Engine is added with the default port `50010` . When using Viz Engines in a so-called dual channel setup, multiple instances of Viz Engine can run 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, *VizEngine1:51007*).

> ⚠️ **Viz Engine host port number restrictions when previewing templates on Editor:** Previewing templates on Viz Trio, Viz Pilot or Viz Pilot Edge requires the Viz Engine port to be `50007` or `50107` . This will match with preview ports `50008` and `50108` . See Network Requirements for port numbering information.

3.2.2   Removing a Viz Engine

In the Connection Broker web interface, click the **trash can** next to the Viz Engine entity that is to be removed.

NAME OR IP ADDRESS	VIDEO MODE	ASPECT	VERSION	GRAPHIC HUB	STATUS	
bgo-anjw:50007	unknown	1.600	v3.9.1.71532	DEMO-GH2	Responsive	🗑️

3.2.3   Modifying the Ping Interval from Connection Broker to Viz Engine

1. Open the *Connection Broker.exe.config* file, typically located at *C:\Program Files (x86)\Vizrt\Connection Broker*.
2. In the configuration file, set a new value for the *VizEnginePollIntervalMilliSeconds* option, which controls the interval for how often the Connection Broker polls each Viz Engine.
3. Set a new value for the *VizEngineTimeoutMilliSeconds* option, which controls the timeout in milliseconds when trying to connect to a Viz Engine.

Code Example

```xml
<?xml version="1.0" encoding="utf-8"?>
<configuration>
    <configSections>
        <sectionGroup name="applicationSettings" type="System.Configuration.ApplicationSettingsGroup, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089">
            <section name="Vizrt.Frames.Properties.Settings" type="System.Configuration.ClientSettingsSection, System, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" requirePermission="false" />
        </sectionGroup>
    </configSections>
    <applicationSettings>
        <Vizrt.Frames.Properties.Settings>
```

```xml
        <setting name="VizEnginePollIntervalMilliSeconds" serializeAs="String">
            <value>10000</value>
        </setting>
        <setting name="VizEngineTimeoutMilliSeconds" serializeAs="String">
            <value>10000</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>
    </Vizrt.Frames.Properties.Settings>
</applicationSettings>
<system.serviceModel>
    <services>
        <service behaviorConfiguration="Vizrt.Frames.AtomBrokerService" name="Vizrt.Frames.AtomBrokerService"></service>
    </services>
    <behaviors>
        <serviceBehaviors>
            <behavior name="Vizrt.Frames.AtomBrokerService">
                <serviceDebug includeExceptionDetailInFaults="true" httpHelpPageEnabled="false" httpsHelpPageEnabled="false" />
            </behavior>
        </serviceBehaviors>
    </behaviors>
</system.serviceModel>
<uri>
    <schemeSettings>
        <add name="http" genericUriParserOptions="DontUnescapePathDotsAndSlashes" />
        <add name="https" genericUriParserOptions="DontUnescapePathDotsAndSlashes" />
    </schemeSettings>
</uri>
<startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5.2" />
</startup>
<runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
        <dependentAssembly>
            <assemblyIdentity name="System.Web.Http" publicKeyToken="31bf3856ad364e35" culture="neutral" />
            <bindingRedirect oldVersion="0.0.0.0-5.2.2.0" newVersion="5.2.2.0" />
        </dependentAssembly>
        <dependentAssembly>
            <assemblyIdentity name="Microsoft.Owin" publicKeyToken="31bf3856ad364e35" culture="neutral" />
```

```
                <bindingRedirect oldVersion="0.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
            </dependentAssembly>
            <dependentAssembly>
                <assemblyIdentity name="System.Net.Http.Formatting" publicKeyToken="3
1bf3856ad364e35" culture="neutral" />
                <bindingRedirect oldVersion="0.0.0.0-5.2.2.0" newVersion="5.2.2.0" />
            </dependentAssembly>
        </assemblyBinding>
    </runtime>
</configuration>
```

3.3   Windows-based NLE Systems

The Configuration Tool and a NLE version of Viz Trio are also installed when you install Graphics Plugin for Windows-based NLE systems. Viz Trio is an advanced character generator used to produce content to use with a Windows-based NLE system. Viz Trio requires a Viz Engine and a Media Sequencer, see the illustration in Hardware Requirements.

3.3.1   Typical Viz Trio Setup

- NLE machines
    - NLE system
    - Graphics Plugin (includes bundled Viz Trio)
- Media Sequencer
- Viz Engine
    - Graphic Hub with graphic scenes
    - *Optional*: Datapool plugins for Viz Engine

Although Viz Pilot and Viz Pilot Edge can also be used as the graphics editor for NLE systems, they are not bundled with the installation, and require a Viz Engine and a Viz Pilot database. The Viz Pilot database (Oracle 10g) is where templates, template data elements and associated data are stored.

3.3.2   Typical Viz Pilot Setup

- NLE machines
    - NLE system
    - Viz Pilot/Viz Pilot Edge
    - Graphics Plugin
- Viz Pilot database installation
    - Oracle 10g
- Viz Engine
    - Graphic Hub with graphic scenes
    - *Optional*: Datapool plugins for Viz Engine

⚠️ **Note:** Viz Engine and Media Sequencer both run on Windows versions 7 or newer. Viz Engine 3.7 and later do not support Windows XP. Although it's recommended to run Media Sequencer on a separate machine, Media Sequencer can run on the Viz Engine machine in small production environments.

This section covers the following topics:

- Installation
- Configuration
- Setting Up a Shared Default Configuration

3.3.3   Installation

When installing Graphics Plugin for EDIUS, Adobe or Avid on Windows, the Configuration Tool and the NLE version of Viz Trio are also installed.

Notes

- The installer requires a Windows user with administrator privileges.
- Adobe Premiere Pro must be installed on the workstation before installing Graphics Plugin for Adobe.
- Separate installers apply for 32-bit and 64-bit NLE systems. The 32-bit installer should be used together with Media Composer 5/NewsCutter 9 (or older); the 64-bit installer with Media Composer 6/NewsCutter 10.

3.3.4   Configuration

A Configuration Tool is installed with Graphics Plugin. See the Graphics Plugin User Guide on how to access the Configuration Tool in Adobe, Avid and Edius, and the Configuration Tool section for settings.

3.3.5   Setting Up a Shared Default Configuration

This section explains how to set up a default configuration for all users of a computer or network by adding a master settings file.

A master settings file is useful in large installations where a system administrator can collect common settings into a shared file, and have this picked up by all users on a computer or all users on a network (by putting the file on a Windows share). Although it's currently not possible to use the Configuration Tool to edit such a shared settings file, an advanced user could copy the *%APPDATA%\Vizrt\VizNLE\vizplugin.xml* file that the Configuration Tool produces into the desired shared location.

If a user settings file is loaded after a master settings file has been loaded, the user setting file takes precedence in case of any conflicts. For example, if the loglevel defaults to "none", and the master settings file sets the loglevel to "report" and the user settings file sets the loglevel to "extreme", then the value of "extreme" is respected by Graphics Plugin. However, if the master settings file sets the Viz Engine hostname, and the user settings file is silent on this point, then the Viz Engine hostname in the master settings file is used.

Before loading the user settings, the master settings file is looked for and potentially loaded as follows:

- Graphics Plugin asks the Windows registry for the *SettingsDir* string in the registry path under the listed registry locations.
- If the string is found, and its content is a path to a folder, then the filename *vizplugin.xml* is searched for at that location.
- If present, *vizplugin.xml* is loaded as a default configuration before loading the individual user settings file.

Registry Paths

Host System	Registry Path
Adobe	*HKLM\SOFTWARE\[vizrt]\NLE\After Effects*

Host System	Registry Path
Avid	*HKLM\SOFTWARE\[vizrt]\NLE\AVX2*
Edius	*HKLM\SOFTWARE\[vizrt]\NLE\EDIUS*

3.4   Mac-based NLE Systems

3.4.1   Installation

The Configuration Tool is installed with Graphics Plugin. The installer requires an OS X user with administrator privileges.

3.4.2   Configuration

A Configuration Tool is installed with Graphics Plugin. See the Adobe, Avid and Configuration Tool sections in the Graphics Plugin User Guide, for more information.

3.4.3   Configuring NLEs to use Editors on a Windows Remote Host

To allow a Mac-based NLE program to use other editors, such as Viz Pilot, on a Windows remote host, do the following:

1. Start *C:\Program Files(x86)\vizrt\Pilot 6.1\VCPAxNle.exe*.
2. In Premiere/Avid on the Mac, create a new effect and click **Edit**.
3. NLE now starts communicating with Viz Pilot on the Windows host.
4. Open a data element, and click **Save**.
5. The effect may now be rendered in Premiere/Avid.

> ℹ **Info:** If you have used Graphics Plugin Editor on your Mac before, you must delete the existing plugin. You may also need to exit and re-open Avid Media Composer for the setting to be applied.

3.4.4   Uninstalling

The following table show which files to remove to uninstall Graphics Plugin for the different Mac NLE applications. Use finder, Terminal or another tool to remove the following files:

Adobe	• */Library/Plug-ins/MediaCore/viznle_plugin.plugin* • */Library/Application Support/Adobe/Common/Plug-ins/[CS version]/MediaCore/viznle_plugin*. Plugin per installed CS version (for example CS5 and CS6)
Avid	• */Library/Application Support/Avid/AVX2_Plug-ins/Vizrt/AvidGraphicsPlugin.avx/*

3.5   Configuring Metagraphics

The metagraphics workflow is available for the Avid and Adobe platforms:

- Avid lets you export sequences with metagraphics (for Windows only, as Avid Assistant is for Windows only).
- For Adobe, both import and export metagraphics is available (for Windows and Mac).

3.5.1   Avid Media Composer

Exporting Metagraphics

For a Vizrt graphics effect to be exported as a metagraphic, the **Burn-in** checkbox must be unchecked in the Vizrt Graphics Plugin effect:

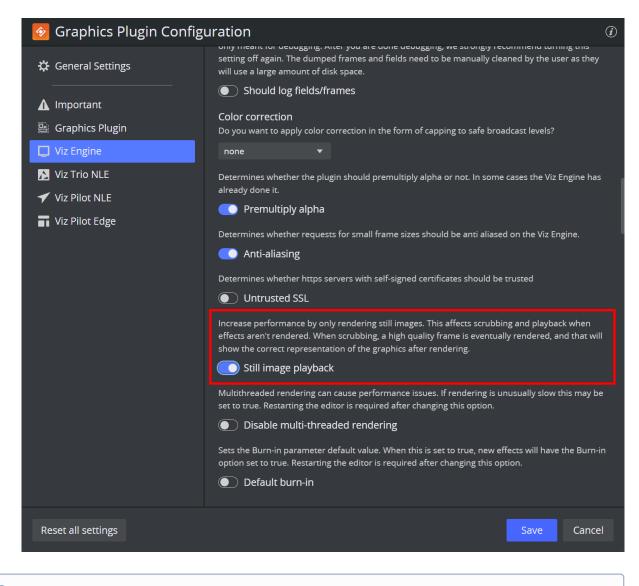If this checkbox is checked, the normal workflow of graphics burn-in is used.

> ℹ️ **Info:** For more information about the metagraphics workflow, see the **Exporting Metagraphics** section of the **Graphics Plugin User Guide**.

> ⚠️ **Note:** Exporting sequences with metagraphics is only available if Avid Assistant is installed.

Previewing Metagraphics

You can enable a static preview when metagraphics is enabled, saving resources.

1. Open the Configuration tool.
2. Select **Graphics Plugin**.
3. Check the **Still image playback** checkbox.

> **ⓘ Info:** For more information about still image playback, see the **Adobe/Avid/Edius Settings** section in the *Graphics Plugin User Guide*.

3.5.2   Adobe Premiere

For Adobe Premiere, metagraphics are configured in Adobe Assistant. See the Adobe Assistant documentation for more information.

3.6   Configuring Server-side Rendering for EDIUS

Server-side rendering is available for the Edius platform. This section covers the following topics:

- Setting up a XRE Client/Server
  - Server Machine
  - Client Machine

3.6.1   Setting up a XRE Client/Server

Server Machine
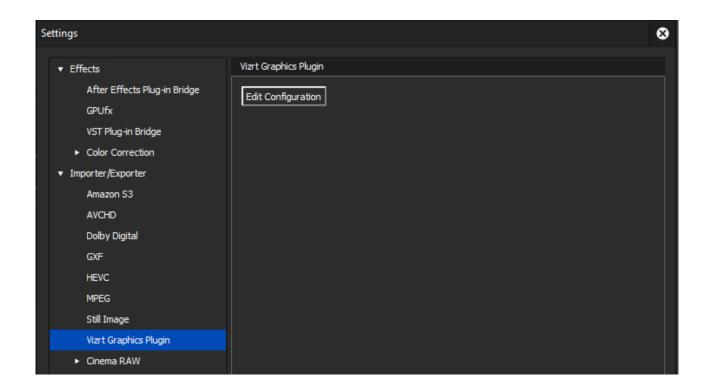
The following components must be installed:

- Grass Valley XRE Server
- Edius Vizrt Graphics Plugin

> ⚠️ **Note:** Make sure the Grass Valley XRE Server is properly configured. You can find more information in the Edius XRE User Guide.

To configure the Vizrt Graphics Plugin, open the XRE Admin Console and select **Plug-in**:

In the **Settings** window go to **Importer/Exporter** and select **Vizrt Graphics Plugin**. Click **Edit Configuration** to open the plugin configuration:

Client Machine

The following components must be installed:

- Grass Valley Edius (any version capable of remote rendering)
- Grass Valley XRE Client
- Edius Vizrt Graphics Plugin

> ⚠️ **Note:** Make sure the Grass Valley XRE Client is properly configured. You can find more information in the Edius XRE User Guide.

Configure both the client and server Edius Vizrt Graphics Plugin using the Configuration Tool.

> ℹ️ **GrassValley Installers:** Make sure that all Grass Valley installers (Edius / XRE Client / XRE Server) are the same version.

See also

- Using Graphics Plugin with Edius in the *Graphics Plugin User Guide*.

3.7   Connecting to Viz Pilot Edge with HTTPS

This page shows how Viz Pilot Edge can be configured to work with HTTP or HTTPS.

3.7.1   HTTP

When using HTTP, the **URL** path must have a value. To configure this, go to the Graphics Plugin Configuration tool and select **Viz Pilot Edge** from the settings view drop-down.
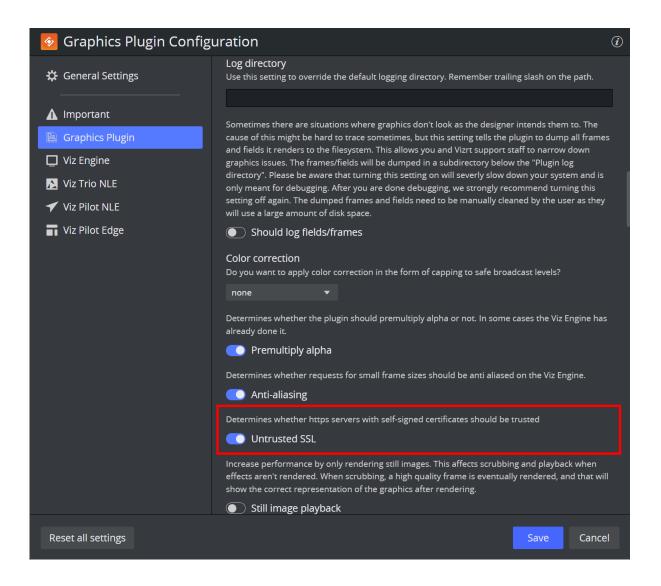

3.7.2   HTTPS

To use HTTPS, besides defining the URL, it is necessary to do the following:

- Use the appropriate (or newer) Graphics Plugin versions:
    - **Adobe:** 2.3.2
    - **Avid:** 2.3.2
    - **Edius:** 2.3.1
- Install the appropriate certificate on the **Trusted Root Certificate Authorities**:

- On the **Graphics Plugin Configuration** tool, go to **Graphics Plugin** and check the **Untrusted SSL** option.

Troubleshooting

When opening Viz Pilot Edge with HTTPS and the message "Failed to load URL" shows, it is likely that one of the steps mentioned above was not followed correctly.

To validate this issue, open the Viz Pilot Edge URL on Chrome or another browser, and check if the connection is secure.

If **Not Secure** ← → C ⚠ Not Secure | is showing on the browser, usually the certificate is not correctly set up. By clicking **Not Secure**, it shows why it is not working properly (usually the certificate is not installed in the specific folder).

> ⚠ **Note:** Please make sure to correctly follow all of the certificate steps to avoid this issue.

4   Fine-tuning the System

Graphics Plugin is a complex product that involves several applications communicating over a network. These applications and the communication between them needs to be robust. Once everything is working after the installation, you can configure the system to work reliably in the environment in which it will be used.

This section covers the following topics:

- Introduction to Fine-tuning the NLE System
- File Compression
- Prefetching
- Testing for Concurrent Use
- Optimizing System Processing

4.1 Introduction to Fine-tuning the NLE System

Graphics Plugin gives the NLE system direct access to preview, edit and to apply graphics to the NLE project and the timeline. In addition, Graphics Plugin allows one renderer to be shared by several concurrent users on a high-speed network. Since the complexities of this design are hidden from the user, performance issues may arise when rendering effects.

4.1.1 Common NLE System Workflow

1. The NLE system requests a frame or field from Graphics Plugin.
2. Graphics Plugin checks its configuration and connects to the Viz Engine over the network.
3. When connected, Graphics Plugin queries Viz Engine for basic information, loads the relevant graphic scene, sets the correct values in the scene, and stretches the graphics to fit the effect in the NLE project.
4. Graphics Plugin requests fields and frames from Viz Engine according to the NLE system requests.
5. Viz Engine receives the requests and ensures that the right scene is loaded and moved to the right frame or field, and sends the graphics back over the network.
6. Graphics Plugin receives the graphics and sends them back to the NLE system.
7. The NLE system processes the data it received before requesting more graphics, repeating the process.

Although the workflow described above is easy to follow, functional and easy to test, it is slow since the NLE system only requests a single frame or field at a time and is blocked until a reply is received. Graphics Plugin sends the request to Viz Engine and must wait for a response, before sending its response back to the NLE system.

4.1.2 Challenge 1 - A Big Volume of Data

A big amount of data is being sent (for example, a simple interlaced HD (1080i) field with full alpha is 4MB (1920*540*4 bytes)), which takes some time to transmit. Rendering a five-second interlaced HD clip at 50 frames a second requires more than one gigabyte of raw data to be rendered and processed by Viz Engine, sent over the network, received by Graphics Plugin, and then moved into the buffers provided by the NLE system, at which point the NLE system must process the data.

4.1.3 Challenge 2 - Latency

In addition, every network has latency and is affected by other network traffic. Waiting for 4MB requests to return and for the NLE system to process the graphics further, takes time and is inefficient. To reduce this delay, Graphics Plugin therefore strives to fields or frames ready when the NLE system sends a request. The plugin tries to accomplish this by keeping the network and Viz Engine continuously busy. It attempts to predict the NLE system's next request by requesting graphics from Viz Engine in advance. Once Viz Engine has received the request, it renders and sends the graphics back to Graphics Plugin, which processes it to fit the NLE system.

A number of factors affect this workflow:

- The TV-standard, for example, PAL is a lot smaller than 1080i and is quicker to render, process and send over the network.
- The CPU and GPU of the Viz Engine define how fast it can deliver graphics. As always, a better graphics card, faster CPU and more memory helps.
- The network card on Viz Engine and the NLE client machine.
- The network capacity defines how much data can be sent over the network.

- The compression of the data determines both how much CPU is used on Viz Engine to compress and on the client to decompress, as well as how much data is sent over the network.
- The amount of ask-in-advance requests keeping the renderer and the network busy.
- The processing the NLE system needs to do with the data it receives (affected by CPU and/or GPU, memory, software, and so on).

4.1.4   Improving Performance

Some things can be improved by adding hardware to the renderer and the clients, and by using as fast a network as possible. In addition, NLE system processing greatly impacts performance. Other things can be configured, either by the plugin or by convention:

- File Compression
- Prefetching
- Testing for Concurrent Use
- Optimizing System Processing

4.2 File Compression

Graphics Plugin supports two different types of compression. You can also send uncompressed data.

4.2.1 No Compression (Raw)

- **Advantages:** No CPU spent on the renderer to compress and no CPU spent on the client to decompress. Usually very good for SD on good networks. Very good on clients with weak CPU and a good network card.
- **Disadvantages:** HD requires a big amount of data, it typically takes longer to transfer over the network than compressed alternatives, and it uses a lot of network capacity.

4.2.2 Portable Network Graphics Compression (PNG)

- **Advantages:** Compresses graphics very well. PNG compression has a very small footprint on very slow networks; for example, this is useful for Digital Subscriber Line (DSL) networks, 10Mbit networks, or district offices that render in a central location.
- **Disadvantages:** May use a lot of CPU to compress and decompress - a problem on slower computers. Generally outperformed by the other two options and mostly useful for special cases.

4.2.3 Run-length Encoding Compression (RLE)

- **Advantages:** Compresses graphics well. Is easy and fast to compress and decompress. Uses a small fraction of the space for a 1080i field compared to no compression - network footprint is quite good. It often performs best for HD on fairly updated computers. Also does very well on 100Mbit networks. Has special handling that exploits the nature of TV-graphics and NLE graphics that contain big open areas, for example lower thirds, over the shoulders, and so on.
- **Disadvantages:** Uses CPU on the renderer and the client (less powerful machines might get better performance with no compression). The smaller amounts of data with SD negates some of the compression benefits, but can often be best for SD too. Might not provide much benefit for graphics with big horizontal gradients.

> ⚠ **Note:** Testing has shown that the advice above may not be suitable in every instance. Only on-site testing can determine what works best in a specific network.

4.3   Prefetching

As shown in File Compression, Graphics Plugin predicts what those frames and fields the NLE system will ask for next and requests them from the renderer in advance (*prefetching*). Although it's possible to disable this feature, this would require safe mode, which is only used if there are problems with prefetching.

4.3.1   What Level Should You Use?

When fine-tuning the performance, the question is how aggressive the pre-fetching should be. Current releases use more aggressive prefetching than earlier, so it's essential to configure how many fields to prefetch, i.e. the number of graphics requests on the network from one client at any given moment. You can also pre-fetch the whole clip, which would likely provide adequate performance; however, this blocks the Viz Engine for other users who want to edit their effects, scrub the timeline or render. In general, the default setting of prefetching 20 fields is a good compromise, although your setup might require a different value (between `5` and `30` are acceptable values).

> ⚠️ **Note:** If rendering performance is too slow when prefetching is enabled, you can try to disable multithread rendering. See the Graphics Plugins section of the Configuration Tool for more information.

See Also

- File Compression

4.4   Testing for Concurrent Use

Graphics Plugin lets multiple users edit, scrub the timeline, and render at the same time. Users require a responsive system, and each user requires its own state on Viz Engine. Viz Engine has its own built-in multiplexer functionality.

When fine-tuning performance, it is important to remember that:

- Testing on a single machine does not reflect the reality of all machines working at the same time.
- Big video files transferred over the network take up network capacity.

4.5   Optimizing System Processing

Despite an optimally configured Graphics Plugin, users may still experience performance issues. One important factor is the size of the project. Working on a fully interlaced HD (1080i) project uses more resources than PAL or NTSC.

When there is a mismatch between video size and the timeline/project settings, the NLE system might need to re-scale graphics and video, which is a costly process. Rendering on a video that resides on a shared drive on the network also utilizes valuable network capacity.

For example, an interlaced HD project with a well-configured Graphics Plugin has an effect added to a layer above the video, and rendering is started. The following happens:

- The NLE system requests a frame;
- Receives it from Graphics Plugin;
- Processes the graphics and possibly caches the data on disk.

These steps are then repeated for each frame. Steps 1 to 2 might only take 20 milliseconds (ms), however, the time from 2 to 3 can be as much as 600-1000ms. Ensuring that the video size, format, project, and timeline settings correspond with each other, and that rendering is done locally on a fast disk is, therefore, very beneficial. Although the performance analysis reports that everything is fine, the system as a whole might be slow due to slow hardware on the NLE system.

5 Troubleshooting and Logging

The following subsections describe common questions and answers and how to log Graphics Plugin activities. It can be useful to provide log files when submitting a support request.

This section contains information on the following topics:

- Log Files
- Troubleshooting

5.1   Log Files

Logging is performed by Graphics Plugin and the different applications it integrates with. The following paragraphs describe the different log files, their log levels, and where they are stored.

Graphics Plugin generates log files that are located in the directory specified by Graphics Plugin (see below).

The Windows-based log files are placed by default in the following directory:

- *%APPDATA%/Roaming/Vizrt/VizNle/Logs*
    - *viznle-log-<date_time>.txt*
    - *nleconfig-<date_time>.txt*
    - *vcp_ax_nle.log*
    - *trio-<date_time>.log*

The Mac-based log files are placed in the following directory:

- *~/Library/Logs/VizNLE/*
    - *viznle_log--<date_time>.txt*
    - *VizNLEConfig--<date_time>.log*
    - *VizNLE_Editor--<date_time>.log*

The log directory can be modified from the Configuration tool. Go to the Graphics Plugin Settings view in the Advanced tab, and specify a new path in the **Log directory** setting.

The path for Viz Trio and Viz Pilot log files can also be modified from the configuration tool, under **Viz Trio NLE** and **Viz Pilot NLE** settings view respectively.

By default, these log files are defined to be placed in the same folder as Graphics Plugin specific log files, using the `-logfile-path LOG_PATH` command in the options text box (`LOG_PATH` refers to the Plugin log directory setting).

To place Viz Trio and Viz Pilot log files in a separate folder from the Graphics Plugin log files, replace `LOG_PATH` in the **Options** setting with a folder name.

Graphics Plugin Configuration

General Settings

- ⚠ Important
- 🖾 Graphics Plugin
- 🖥 Viz Engine
- 🖻 Viz Trio NLE
- ➤ Viz Pilot NLE
- 🗔 Viz Pilot Edge

Viz Trio NLE

Host
Host name of the machine that is running Viz Trio NLE. This should usually be localhost

| http:// | localhost | 6210 |

Executable
Name of the Viz Trio NLE process executable, in case you want to use another filename than the usual.

trio.exe

Options
The exact options string that should be passed to the Viz Trio NLE process at startup.

-nle-mode -control -mse MSE_HOST -logfile-path "LOG_PATH"

Network timeout
How long should Viz Trio NLE connection attempt to wait for replies. 30s is 30 seconds, 8m is 8 minutes, 2h is 2 hours, ...

| 90 | seconds |

Grace period
How long should the plugin give Viz Trio NLE to start everything neatly. 30s is 30 seconds, 8m is 8 minutes, 2h is 2 hours, ...

| 2 | seconds |

Shared folder
Shared path used for temporary files during trioshow import. Both the NLE plugin and the external Viz Engine should have full access to this folder.

Reset all settings

Save   Cancel

5.2   Troubleshooting

This section contains troubleshooting and FAQs and covers the following topics:

- General
- Using the NLE System
- Configuring the NLE System
- Tuning the NLE System
- Rendering with Viz Engine

> ⚠️ **Note:** Differences between the Mac and Windows-based versions are noted where relevant.

5.2.1   General

- **How do I contact Vizrt with my Graphics Plugin questions?**
    - Before sending a request, please check the sections below. See Feedback and Support for how to report an issue.
- **Why are my start, stop and continue buttons in Viz Trio disabled?**
    - When using an external Viz Engine, the start, stop and continue buttons in Viz Trio are disabled. This also applies the **Restart Viz Engine** option available from the context menu in the Render window.

5.2.2   Using the NLE System

- **Why does the Vizrt graphics stop showing in the preview window?**
    - This can occur when still image playback is turned on and a non-existing graphic has been added to the timeline. If this happens, remove the faulty graphic, then temporarily turn off still image playback. Turn still image playback back on after verifying that the graphics preview is correct. The bug does not affect rendered output. Still image playback is turned on/off in the **Configuration Tool**, see the *Graphics Plugin User Guide*.
- **Why am I seeing double-sized graphics?**
    - There is a difference between broadcast and consumer resolution. Several of the plugins are confused by for example consumer HD, producing graphics of the wrong sizes. If the camera says 1080i and has a resolution of 1440x1080 or 1280x1080, it's consumer HD as opposed to broadcast HD of 1920x1080.
    - Check that the software versions are correct. For details, see Requirements and Recommendations.
- **I have edited and saved a scene but I cannot see the graphics on the video. Why is this?**
    - When editing a scene in Viz Trio or Viz Pilot, select to see the scene with *key* turned on (available above the renderer window). If the scene goes black, there is a design flaw in the scene. Redesign the scene by adding key to it. For details, see the **Scene Design** section in *Graphics Plugin User Guide*.
- **Why does the preview in Viz Pilot only display PreviewPanel?**
    - This might be a network bandwidth problem. In most cases this happens when Graphics Plugin's compression is set to No Compression and there is a bandwidth mismatch between Viz Engine and the NLE machine. Usually, this happens when for example the NLE machine is connected to a 1Gbps network and the Viz Engine is connected to a 100Mbps network.

- Try to set compression to *PNG Compression* or configure both machines to use the same network with the same bandwidth. Remember to check that the network cards are compatible with the network it is connected to.
- Viz Engine should ideally be connected to a fast network. If the NLE machine is connected to a slower network it should work fine.
- In order to preview templates in Viz Trio, Viz Pilot and Viz Pilot Edge, the Viz Engine port on the Graphics Plugin Configuration Tool must be 50007 (preview port 50008) or 50107 (preview port 50108). See Network Requirements for more information.

> ⚠️ **Note:** The recommended bandwidth for No Compression is 1Gbps.

- **How do I set Viz Pilot's preview aspect ratio?**
  - The preview aspect ratio is set in Config's **Output Format** section (Viz Engine's configuration tool).
- **Why is Avid locked while using Graphics Plugin?**
  - Avid and most NLE systems are designed to block whenever a plugin is executing, including when the user has pressed **Edit**, the editor is open and the plugin is waiting for a response.
- **Why are the Graphics Plugin editors and configuration programs not closed when the NLE system is closed?**
  - The NLE system is unable to handle a shutdown of the plugin itself (see the previous question).
  - If the NLE system is accidentally terminated or closed this might cause the plugin process to still be running, and a manual and forced shutdown of the plugin is necessary. Force shutdown by restarting Avid or using the Task Manager.

5.2.3   Configuring the NLE System

- **Which ports are used in a typical installation?**
  - See Port numbers.
- **On a Windows system, how do I configure the Viz Engine host and port?**
  - For details about installing Viz Engine, see Graphics Rendering.
- **On a Windows system, how do I configure the output from the Viz Engine?**
  - Some settings related to the output from the Viz Engine are configurable. For example, the image compression protocol (*No, PNG or RLE*).
  - If some of the Viz Engine settings are incorrect, the preview is disabled and a message is displayed stating the cause. Incorrect settings may include if the host is not found, cannot connect to the port, and so on. In addition, the user is notified if the Viz Engine connection is lost during runtime

> ⚠️ **Note:** Some GUI settings are only valid for Windows-based NLE integrations using Viz Trio and Viz Pilot.

- **On a Windows system, how do I configure the user interface that Graphics Plugin should use?**
  - Open the Graphics Plugin configuration tool and set the **Plugin user interface** setting to **Viz Pilot** or **Viz Trio**.
  - The setting ensures that the NLE application sends the extra data that the user interface needs. This includes, but is not limited to, the scene length in milliseconds.
  - An error message is always displayed if the NLE system is not configured properly. The log files also show warnings for non-critical problems.

5.2.4   Tuning the NLE System

- **How can I find out how much time is used between each call from the NLE system to the plugin?**
    - Empty the log directory. Start a project, make an effect, render it and exit the NLE system. Go to the log directory (see the Log Files section) and run the following command:

```
grep --binary-files=text PERFORMANCE viznle-log*.txt
```

- Disregard the first few numbers and you might find for example:

```
14:42:03 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 397 msecs.
14:42:03 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 170 msecs.
14:42:03 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 394 msecs.
14:42:04 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 161 msecs.
14:42:04 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 157 msecs.
14:42:04 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 411 msecs.
14:42:04 [T:2689318816] [/Users/<username>/fcp-1.0/plugins/fcp/
VizNlePlugin.mm,876] QUEUE PERFORMANCE: Time between renders 161 msecs.
```

5.2.5   Rendering with Viz Engine

- **Which Viz Engine version should I run for HD?**
    - A standard-PC version for all HD output is recommended due to a number of issues in the INCA version.
- **Can I include Datapool in the Graphics Plugin workflow?**
    - When including the Datapool mechanism in the Graphics Plugin workflow, you are always working on a scene-copy. The built-in multiplexer functionality of Viz Engine rewrites references to `RENDERER` to the scene-copy you are using. This means that if your scene has DataActions that refer to `RENDERER`, they are not rewritten to the right scene, and they will fail fairly often. Referring to `$SCENE` will likely also not work.

See Also

- Graphics Rendering
- Tuning the NLE System