Media Sequencer Release Notes

Version 5.8

Copyright ©2025 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

2025/12/11

Contents

1 Media Sequencer 5.8.1 5
1.1 Fixed Issues 5
2 Media Sequencer 5.8.0 6
2.1 System Requirements 6
2.2 Silent Installation 8
2.3 Running the Media Sequencer 8
2.4 Removed Functionality 8
2.5 Media Sequencer Core Enhancements and Fixes 9
2.6 New Plugins 10
2.7 Plugin Enhancements and Fixes 10
2.8 Known Issues 11
3 Documentation 13
4 Support 14

- Media Sequencer 5.8.1
  - Fixed Issues
- Media Sequencer 5.8.0
  - System Requirements
  - Silent Installation
  - Running the Media Sequencer
  - Removed Functionality
  - Media Sequencer Core Enhancements and Fixes
  - New Plugins
  - Plugin Enhancements and Fixes
  - Known Issues
- Documentation
- Support

1 Media Sequencer 5.8.1

Release Date: 2025-12-11

These are the release notes for the Media Sequencer (MSE) version 5.8.1. This document describes the user-visible changes since 5.8.0.

1.1 Fixed Issues

- Fixed an issue where creating a template element from model with material_definition fielddef is not recognize and skipped (MSE-9929).
- Fixed an issue where logging a message about an unexpected field in a list payload could cause a read access violation, if the list field name was a nullptr. The log now correctly reports the name of the corresponding field (MSE-9834).
- Fixed an issue where the Media Sequencer did not validate STOMP URLs before using them. The MSE could attempt to use STOMP URLs with unsupported protocols when multiple STOMP monitor links were present in a document, causing failures and fallback to polling instead. The supported protocol is STOMP over TCP. This issue was observed with newer versions of Pilot Data Server (MSE-9862).
- Fixed an issue in which superchannel_manager could resolve the incorrect background scene when playing a transition logic element with an alternative concept defined. This can cause superchannel_manager to wait for the wrong scene to transition, resulting in a timeout error and delay in the next taking element (MSE-9851).
- Fixed an issue where MSE would fail to parse a URL with query parameters but an empty path (MSE-9869).
- Fixed Gateway stops responding when a failover pair is activated (MSE-9897).
- Fixed a possible unhandled exception when sending a POST or PATCH request to a Hierarchy Collection while also removing the collection node (MSE-9891, MSE-9896).
- Fixed a possible crash when disabling the vcp_service actor while it waits for HTTP responses (MSE-9912).
- Update WIBU CodeMeter Runtime to version 8.40a (MSE-9923).
- Fixed invalid URI replied by Peptalk uri command for a Hierarchy Collection resource representing a show or playlist (MSE-9914).

Copyright © 2025 Vizrt

2 Media Sequencer 5.8.0

Release Date: 2025-06-30

Note:

This release introduces two new licenses: MSE_PLAYOUT and MSE_FRAME_PREVIEW.

If you're upgrading a system used for playout, it will stop functioning after the update unless the *MSE_PLAYOUT* license is added in the MseLauncher's License settings. Please contact the Vizrt license team to obtain the required license.

Note:

Refer to comprehensive documentation of the Media Sequencer (MSE) in the *MSE Manual*. When the sequencer is running, this manual is available at http://localhost:8580/mse_manual.html. You can also access the manual by opening the local file www/mse_manual/index.html in the MSE install directory.

2.1 System Requirements

As of version 3.1, Media Sequencer is only available as a 64-bit software.

Software

Supported operating systems:

- **Server:** Windows Server 2016 or higher.
- **Workstation:**
  - Recommended: Windows 11.
  - Windows 10 (LTSC 21H2) is also supported.

Media Sequencer 5.8.0 is 64-bit software, it can only be run on 64-bit versions of Windows.

Operating systems that have reached their end of life cycle and are no longer officially supported by their manufacturers, are not supported for use with the MSE.

A Media Sequencer used with a Viz Pilot system needs to access the database which Viz Pilot uses. For Viz Pilot version 6 or older, this is an Oracle database, which previous Media Sequencers had to connect to directly.

While this configuration remains supported, Media Sequencer 4.0 and Viz Pilot 7 introduced the ability for the Media Sequencer to connect to the Pilot Data Server instead. The Pilot Data Server introduced in Viz Pilot 7 can in turn, run with either an Oracle backend or a Viz Graphic Hub backend.

When a Media Sequencer is set to connect directly to the Oracle database, an Oracle Database Client (Runtime or Administrator) is needed. Administrators must take care to install the 64-bit version of the Oracle Database Client.

Microsoft .Net

The Media Sequencer requires Microsoft .Net framework 4.7.2 (full) or newer to be installed on the machine. The installer notifies the user if .Net 4.7.2 is not found.

Licensing

As of version 5.4.0, Media Sequencer requires WIBU licensing. The WIBU CodeMeter runtime must be installed and configured, to make Media Sequencer licenses available. CodeMeter is installed as part of the new bundle installer.

Selecting which license container to use and which licenses to enable, is configured from the Media Sequencer launcher with elevated privileges. The launcher generates a license configuration file *licenses.json* in the Media

Sequencer data directory. A Media Sequencer core license is required to launch. Additional licenses must be enabled for running the sequencer as a central Gateway, or for licensing the Viz Multiplay client.

As of version 5.5.0, Media Sequencer supports Viz License version 3 (movable licenses), in addition to the already supported Viz License version 2 introduced with 5.4.0. The launcher has been changed to allow selecting between version 2 and version 3 licenses, in the case where both are available.

The MSE can be configured to alert about license expiration in two ways, either via email (SMTP) or via Graphic Hub journal messages. This alert ability can be configured from the License Alert configuration, available from the launcher when the MSE is running.

For information about how to configure MSE licenses and license alerts, refer to the MSE WIBU licensing documentation (available from the Windows Start menu).

Hardware

System requirements for hosting the Media Sequencer:

Workload	Processor	Memory
Standard	2 cores / 4 threads	6 GB
Demanding	4 cores / 8 threads	12 GB

Running other resource-intensive software on the same machine is not recommended as it can slow down the Media Sequencer.

Virtualization

The Media Sequencer may be run inside a virtual machine. We recommend following the table above when allocating resources to the virtual machine. For time-critical operations we recommend using the Element Scheduling REST API to trigger frame accurate playout.

Applications and Components Recommended with this Release

- Viz Trio 4.4.3 or higher
- Viz Multiplay 3.3.0 or higher
- Viz Pilot 8.9.3 or higher
- Viz Pilot Edge 3.3.0 or higher
- Template Builder 3.3.0 or higher
- Pilot Data Server 9.3.1 or higher
- Preview Server 4.7.1 or higher (Preview Server 5.0.0 requires MSE Frame Rendering API)
- Graphic Hub 3.9.2 or higher
- Graphic Hub REST 2.9.2 or higher
- Graphic Hub ImEx Agent 2.1.2 or higher
- Viz Engine and Viz Artist 5.4.1 or higher
- Viz One 7.6.2 or higher

The Media Sequencer has been designed to have backward compatibility with older versions of these components. We work hard to keep the Media Sequencer backward compatible, however, for some older versions of the Delphi-based clients (for example, *Director* and *Viz Trio*), we highly recommend upgrading these clients to a more recent version.

Copyright © 2025 Vizrt

2.2 Silent Installation

Silent installation allows the Media Sequencer to be installed without user interaction.

To perform silent installation, open a command shell as an administrator and run the Media Sequencer bundle installer with the `-s` or `--silent` option.

It is also possible to select features to be installed from the command line.

To get all options available, run the Media Sequencer bundle installer with the `--help` option.

Example command:

```
MediaSequencer.BundleInstaller-5.x.x.xxxxx.exe -s
```

Example command to install Media Sequencer with Gateway controller:

```
MediaSequencer.BundleInstaller-5.x.x.xxxxx.exe -s --mse-gw-controller
```

A license configuration file *licenses.json* in the Media Sequencer data directory (*C:\ProgramData\Vizrt\Media Sequencer*) is required to run Media Sequencer. If *licenses.json* is missing, the installer will prompt the user to create it using the Media Sequencer launcher. Note that license configuration cannot be changed using the non-admin Media Sequencer launcher.

To avoid the automatic license configuration prompt before running silent installation, copy *licenses.json* from another successful Media Sequencer installation to the Media Sequencer data directory. See the License section for more information.

2.3 Running the Media Sequencer

It is highly recommended that the Media Sequencer is run as a system service, rather than in a console. This is the default behavior after installing.

To start the Media Sequencer, licenses must be configured. This can be done from the launcher with elevated privileges. For more information about licensing see the License section, and the MSE WIBU licensing documentation (available from the Windows start menu).

The Media Sequencer can be started and stopped using the Media Sequencer launcher. You can do both with and without elevated privileges. In order to start and stop the Media Sequencer without elevated privileges, open the shortcut `Media Sequencer (non admin)` available from the Windows Start menu. Note that it is *not* possible to configure any settings from the launcher without elevated privileges.

2.4 Removed Functionality

Deprecated Plugins

Plugins are sometimes deprecated with a new release of the MSE. These plugins will no longer be available in the next release. You need to migrate away from using them.

The following plugins are deprecated in this release:

- **websocket:** The websocket plugin mostly used for providing peptalk, treetalk and STOMP protocols over a websocket connection, is deprecated with this release of the Media Sequencer. Clients using these websocket APIs should migrate to using the APIs served by the HTTP server with discovery via the service document.

2.5 Media Sequencer Core Enhancements and Fixes

Frame Rendering API

A new Frame Rendering API is now available in Media Sequencer 5.8. This REST API allows clients to request snapshot previews of graphics scenes rendered by Viz Engine directly from the Media Sequencer. The API supports both GET and POST requests, returning PNG images of the rendered frames. Snapshot requests can be customized with VDF payloads, scene paths, and image properties such as size and aspect ratio. Frame Rendering API requires *MSE_FRAME_PREVIEW* license.

Frame Rendering resources are managed as a collection, represented by the Frame Renderer Collection resource. The collection contains Frame Renderer Collection Entry resources, which correspond to Viz Engine endpoints used for frame rendering. To use the API, clients create a new Frame Renderer Collection Entry resource via a POST request to the Frame Renderer Collection, then follow the atom links provided by the Frame Renderer Collection Entry resource for further operations. For example, the atom link with rel='snapshot' is used to request snapshot previews.

The Frame Rendering API also provides STOMP monitoring for real-time updates on the status of Frame Renderer Collection Entries. Clients can subscribe to the atom link with rel='monitor' in the Frame Renderer Collection document to receive these updates (MSE-9290, MSE-9516).

For more details, see the REST API documentation.

Initial support for Pilot Core Service

Pilot Core Service (PCS) will be the next generation Pilot backend, based on a new scalable and secure document database (Viz Asset Management). This release of the Media Sequencer includes initial support for PCS, with support for resolving Pilot elements from PCS with Playable Cache (MSE-9262).

WIBU CodeMeter Runtime

Update WIBU CodeMeter Runtime to version 8.20a (MSE-9559).

Support Single Sign-on authorization in Playable Cache

Support has been added to set up Single Sign-on (SSO) authentication with client credentials for the MSE. This enables Playable Cache to obtain SSO access tokens for authorizing requests for protected resources from trusted services.

SSO configuration is done from the MSE launcher in the settings window. An OpenID issuer URL, client ID, client secret, scope, as well as a list of trusted services that require SSO authorization must be configured. The MSE will not attempt to authorize requests to untrusted services (MSE-9416).

New Configuration Option: Ignore take_duration on Graphic Elements

A new global setting has been introduced: /config/take_duration/ignore

When set to 'yes', this option disables the automatic 'take out' of graphic elements based on the take_duration attribute. This is particularly useful in Channel Branding workflows where the automation system controls when elements are taken out, but the take_duration value is still displayed in the user interface for reference (MSE-9677).

Copyright © 2025 Vizrt

Playable Cache Backend Added as Option in Pilot Database Config GUI

Playable Cache backend has been added as an option in the PilotDbConfig web app, as an alternative to legacy Oracle database or Pilot Database with vcp_service (MSE-9557).

2.6 New Plugins

2.7 Plugin Enhancements and Fixes

viz

- Handler setting without both host and port is now invalid instead of use default host and port (MSE-9508).

http_server

- Expose viz layer in Output Entry for viz_video handlers. Clients can select viz layer by specifying a 'viz_layer' field in VDF payload of the Output Entry. Default value is 'BACK' (MSE-9544).
- Provide a 'name' field in VDF payload of the Output Entry and Output Settings resources. Clients may specify the created handler name when POSTing to Output Collection resources by setting the 'name' field (MSE-9554).
- Fixed an issue where GET requests to Videowall Collection resources containing invalid videowall entries return 404 Not Found (MSE-9596).
- Added a read-only 'mode' field to the VDF payload of Output Entry and Output Settings resources for superchannel viz and superchannel video handlers. This field shows if the handler operates in 'superchannel_arm_and_fire' mode. Clients cannot create, delete, or modify 'superchannel_arm_and_fire' handlers via Output Collection, Output Entry, or Output Settings, as these handlers belong to video walls (MSE-9709).
- Added support for providing template descriptions for scene alternatives upon POST/PUT of BGFX mastertemplate documents (MSE-9501).
- Added support for POSTing a scene to a video wall scene collection without requiring Graphic Hub credentials. This is done by POSTing a VDF payload that contains all needed information about the scene, as an alternative to the MSE resolving this information from Graphic Hub REST (MSE-9670).
- Added support for BGFX expertmode markup upon POST/PUT of BGFX mastertemplate documents (MSE-9672).
- Two new Profile Commands have been added, "direct_take" and "direct_update". They allow directly taking or updating an element on a specific channel and renderer layer. These commands schedule temporary elements and are suited for simpler integrations that do not require the normal playlist-based workflow (MSE-9654).
- Added support for creating Trio Shows through the MSE REST API (MSE-9647).
- Made the 'alternative-concept' field for outputs in the Profile REST resource available, even when the concept has not been overridden for the output (MSE-9612).
- Empty video handlers are now excluded from the output collection feed (MSE-9553).
- Added support for inserting MOS objects from Pilot Edge into a playlist with a PATCH request to the hierarchy collection feed (MSE-9599).

Copyright © 2025 Vizrt

- Added support for inserting multiple entries (atom entries or MOS objects) into a playlist with a single PATCH request to the hierarchy collection feed (MSE-9623).

playable_cache

- Allow to change 'poll_interval' configuration on-the-fly. When changing, the current polling task will be rescheduled with the new interval (MSE-9569).
- Fix a crash when building a template with generators in VDF model (MSE-9648).
- Added support for monitoring template atom entries with inline BGFX mastertemplate as part of the document (MSE-9438).
- Added support for resolving pilot templates with execution logic provided as an auto-convertible document (MSE-9579).
- Added support for live-update functionality for pilot templates resolved by Playable Cache. This includes support for live-update provided as an auto-convertible document (MSE-9540).
- Added support for context variables to category channel mapping in Playable Cache, provided as an auto-convertible document. Channel names are resolved with a new Playable Cache namespace handler (MSE-9534).
- Added support for resolving pilot templates containing concept to label mapping provided as an auto-convertible document. The mapping can be used by control clients to show user readable labels for concepts instead of the concept IDs (MSE-9541).
- Added support for authorizing requests for protected resources using access tokens obtained with client credentials (MSE-9416).
- Added pilot metadata node to data elements resolved with Playable Cache. The metadata node will contain an editor link if provided in the atom entry for the pilot element. This editor link can be used by clients to open the element for editing in Pilot Edge (MSE-9600).

gateway

- Added support for keepAlive messages between MSE Client and Gateway in order to keep the connections alive when idle (MSE-9606).

element

- Added support for resolving category channel from channelselector in mastertemplate during playout of elements (MSE-9556).

relative

- Added support for resolving category channel from channelselector in mastertemplate during playout of filled presets (MSE-9602).

2.8 Known Issues

This section describes issues that have been reported but not resolved.

Media Sequencer does not support live-updates from Pilot Core Service (PCS)

Currently, Playable Cache attempts to resolve the auto-convertible live-update link in Pilot templates. In future versions of the Media Sequencer, Playable Cache will in addition support inline live-update document needed to support live-update from PCS. This will allow all template information to be retrieved in a single request (MSE-9749).

Cannot Configure WIBU SmartBind Loose Licenses in MSE Launcher

When activating a Software Container on a virtual machine or in a cloud environment via the license portal, additional manual configuration is required (MSE-9666).

Workaround Steps

1. Assign the required licenses in **MSE Launcher**.
2. Close the **MSE Launcher** application.
3. Navigate to the following file location: *C\:ProgramDataVizrtMedia Sequencerlicenses.json*.
4. Open the `licenses.json` file in a text editor **with administrative privileges**.
5. Locate the relevant license container and change its `"location"` value from `"0"` to `"1"`.

Note:

Editing the file requires the text editor to be run as an Administrator.

Incorrectly displayed WIBU License in Media Sequencer Launcher

If multiple versions of a license are available on a license server, the license displayed in the main window of the Media Sequencer Launcher may show an incorrect *Count* and *Expiration date* compared to what is selected in the Media Sequencer Settings (MSE-9522).

Issue with "Out" Command when Performing Frame Accurate Playout of Transition Logic with Video

If you do frame accurate `out` and then two `take` operations of transition logic elements with video, the second video will not start from the beginning of the clip (MSE-7046).

MSE may Report the Main Thread Slow on the Re-initialization Process

MSE may report the main thread to be slow while it is reinitializing. This issue typically occurs when the Viz Pilot database is at a remote location.

Inaccurate Video Availability Progress if the Video Clip is Deleted from a Publishing Point

Availability progress of a video clip is not immediately reset to zero when the related video clip is deleted from a publishing point. This is due to the limitation of the Viz One API feedback (ME-664).

Ordering of Fields in Payloads and Models Not Respected

Media Sequencer does not respect ordering of fields and fielddefs in VDF payloads and models, meaning that for example, a payload inserted using the REST interface may have the fields in a different order when later obtained from the REST interface. This has no consequence for playout, but may cause unexpected reordering of fields and fielddefs in software used to display or edit payloads or models (MSE-6565).

Superchannels May Become Stuck in Transitioning State when Arm/Fire Non-Existing Video Clips

When operating in superchannel arm/fire mode, the Media Sequencer requires feedback from the Viz Engine to ensure that the Viz Engine has finished transitioning between playing and pending subchannels before sending next commands; otherwise, the next commands may affect the incorrect subchannel. The Media Sequencer holds any new commands during transitioning until it receives transition completed feedback from the Viz Engine. When a user arms a non-existent video clip, the Viz Engine does not notify the Media Sequencer that the clip is not available. If the user continues to fire this non-existent video clip, the Media Sequencer will stop executing new commands until the `viz_feedback_timeout_seconds` is reached. This `viz_feedback_timeout_seconds` can be configured in superchannel_manager handler settings (MSE-8414).

Copyright © 2025 Vizrt

3 Documentation

Documentation for the Media Sequencer is available in the bundle installer on the Vizrt FTP.

Once installed, documentation is available within the MSE, as described above.

4 Support

Support is available at the Vizrt Support Portal.