



Media Sequencer Release Notes

Version 5.6



Copyright ©2024Vizrt.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. 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

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

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

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

2024/05/15

Contents

1	Media Sequencer 5.6.0	4
1.1	System Requirements	4
1.2	Silent Installation	6
1.3	Running the Media Sequencer	6
1.4	Removed Functionality	7
1.5	Media Sequencer Core Enhancements and Fixes.....	7
1.6	New Plugins.....	8
1.7	Plugin Enhancements and Fixes	8
1.8	Known Issues.....	10
2	Documentation	11
3	Support	11

These are the release notes for the Media Sequencer (MSE) version 5.6. This document describes the user-visible changes that have been done to the software since release 5.5.

1 Media Sequencer 5.6.0

Release Date: 2024-05-15

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.

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

1.1 System Requirements

Software

Supported operating systems:

- Server: Windows Server 2012 R2 (with Update 2999226) or higher.
- Workstation: Windows 10 or higher.

Since Media Sequencer 5.6.0 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 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 4.2.0 or higher
- Viz Multiplay 3.2.0 or higher
- Viz Pilot 8.9.2 or higher
- Viz Pilot Edge 3.0.2 or higher
- Template Builder 3.0.2 or higher
- Pilot Data Server 9.0.2 or higher
- Preview Server 4.6.0 or higher
- 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 5.2.1 or higher
- Viz One 7.5.3 or higher

The Media Sequencer has been designed to have backward compatibility with the older version of the 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.

1.2 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-5.6.0.xxxxx.exe -s
```

Example command to install Media Sequencer with Gateway controller:

```
MediaSequencer.BundleInstaller-5.6.0.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 Media Sequencer launcher. Note that license configuration can not 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 [Licensing](#) section for more information.

1.3 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.

1.4 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:

- None.
-

1.5 Media Sequencer Core Enhancements And Fixes

Allow MseLauncher to run without administrative rights

For more detail, see section [Running the Media Sequencer](#) (MSE-9169).

Remove command line option to exclude Media Sequencer from being installed

`--no-install-mse` and `--install-mse` option was removed from the Media Sequencer bundle installer. The Media Sequencer is always included (MSE-8856).

Improved error message when acquiring license(s) fail

Fixed an issue with unclear error message when WIBU license could not be acquired. The error message now contains correct use count and expiration date information, and additional licenses no longer show as `NOT_SET` (MSE-8967).

Create an element by taking the default alternative on the master template into account

MSE creates a new element with the default alternative concept and variant specified on the master template when an element is created via REST API or Intelligent Interface without specifying the alternative concept or alternative variant (MSE-7883).

Fixed an issue where an invalid HTTP request could cause MSE to hang

(MSE-8964).

Fixed a possible program failure when renaming a channel in a superchannel arm/fire profile

(MSE-9003).

Fixed an issue where MSE does not always relay Viz command timeout error to Multiplay

This bug caused Multiplay to hang when an error occurred while saving a preset scene composed of multiple presets (MSE-9005).

Fixed an error when installing Media Sequencer on a clean machine

(MSE-9055).

Fixed a possible hang when running Media Sequencer installer with `--silent`

(MSE-9052).

Update WIBU Codemeter Runtime to version 7.60d

(MSE-9090).

Update REST API example to Visual Studio 2022

(MSE-9084).

1.6 New Plugins

1.7 Plugin Enhancements And Fixes

channel_state_actor

- Removed asset entry of video from state:layer in channel_state feed. The asset entry can be retrieved from Element Collection Entry resource provided as `based_on` attributes (MSE-9010).
- Fixed `based_on` links in layer state and arm state when scheduling a ref node pointing to a filled-preset element stored in playable cache. The `based_on` link should point to a child content of the filled-preset assigned to that channel, not the ref node (MSE-9244).
- Prevent excessive channel state event messages when running the `out` command on GFX channels (MSE-9038).

superchannel_manager

- Removed the channel naming restriction. Users can rename channels with any name they want (MSE-9006).
- Improved internal states and command handling during superchannel transitions (MSE-9014, MSE-9015, MSE-9016, MSE-9017).
- Ensure `arm` operations received during superchannel transitions retain their command orders when being processed after transitions stopped (MSE-9030).
- Improved `arm` operations by creating copies of the armed elements and maintaining them in an MSE's internal storage. This improvement ensures that armed elements will be valid for `fire` operations later (MSE-8935).
- Allow playing a live source on multiple superchannels, as Viz Engine 5.2.0 now supports it. Users must be aware that playing the same live source on multiple superchannels does not work with Viz Engine versions earlier than 5.2.0 and may cause incorrect states in superchannel and channel_state (MSE-9163).
- Provide fade-out animation when OUT superchannel in `arm_and_fire` mode. Previously, OUT superchannel directly removed content on the playing sub-channel without transition effect. The reason was to preserve armed content on the pending sub-channel. In this version, OUT a superchannel sets pending sub-channel to empty and toggles transition. This allows the playing content to be faded-out. After that, if there was any content previously armed on the pending sub-channel before OUT, MSE re-arms the contents (MSE-9164).
- Support playing-out using program channel and video program channel in superchannel `arm_and_fire` mode (MSE-9171).

- Fixed a problem where MSE failed to resolve `content_channel` for filled-preset elements if the main channel is not specified by context variables or set as a `channel` attribute on the filled preset elements. This affects `loaded` and `available` statuses of the filled-preset elements (MSE-9194).
- Allow to specify list of channels affected by a command. In superchannel arm/fire mode, some commands may affect multiple channels, for example, `fire_all` a videowall or `take` a filled preset. Users may specify which channels to affect by supplying channel names as environment variable `su_mgr_allow_channels` when scheduling the command or specify as query parameters `allow_channel` when POST the command as `profile_command_resource` (MSE-8925).
- Support taking, arming, firing, and firing all an element created from `<bgfx:operation type="clearchannel" />`. This type of element is used to clear the superchannel or the viz renderer where this element is taken onto (MSE-9245).
- Fixed an issue where the program could unexpectedly terminate due to unhandled exceptions from the format library (MSE-9053).
- Fixed an issue where the superchannel_manager could incorrectly track inactive videowalls (MSE-8978).

treetalk

- Fixed possibility that MSE becomes unresponsive. This problem can occur when a PepTalk connection is disconnected while serializing a VDOM node (MSE-9032).

vcp_service

- Use fewer threads in vcp_service actor to improve overall performance on CPUs with fewer cores (MSE-9045).

viz

- Check and prevent invalid renderer node configurations to send any command (MSE-9012).
- Fixed an issue where the `command_timeout` setting on viz handlers did not take effect when superchannel arm and fire mode were enabled (MSE-9254).

http_server

- Ensure content-type of error responses are `text/plain` to prevent browser from executing any unintentionally scripts injected into response body (MSE-9100).
- Introduce an `id` attribute to layer states and arm states. The `id` attribute is a unique id which is updated each time an element is assigned to a layer/arm state. This allows users to detect changes when the same element is re-taken or re-armed (MSE-9072).
- When creating a video wall through the REST interface, the video wall name, main channel name, and channel name prefix can all be left out of the POST request. MSE will generate a new wall with sensible default names (MSE-9155).
- Allow setting `publishing_point_uri` and `publishing_point_atom_id` when creating a videowall using REST API. These publishing point settings are used when generating handlers for the created videowall (MSE-9158).

- Remove links to bgfx mastertemplate resource from Template Collection Entry representing default templates. Default templates are special templates and do not comply with bgfx mastertemplate schema (MSE-9261).
- Improve the completeness of the profile REST API (MSE-9237).
- Fixed memory leak in the HTTP server (MSE-9227).
- When creating a videowall using the REST API, the main channel is now created as the first channel in the videowall. This allows for better visualization on the Multiplay user interface (MSE-9156).
- Support setting the timecode attribute on a ref node with `vaext:start` markup in the POST request (MSE-8998).

gateway

- The Media Sequencer, whether configured as a central gateway (mosgw) or a client (msemos), now supports the MOS protocol up to 4.0. MOS 4.0 enhances security and introduces passive MOS, enabling bi-directional communication with devices outside of firewalls (MSE-8721).
- Viz Gateway should not accept incoming connections from inactive NRCSSs (MSE-9230).
- Fixed an issue where the mosbuilder inserted an extra newline before the end-tag. This extra newline caused Showmaker's MOS parser to fail in the Showmaker NCS (based on MOS 4.0) (MSE-9270).
- Fixed an issue where the program could unexpectedly terminate when enabling "Send video status changed back to NCS" with a profile containing superchannel_video actors (MSE-9294)

playable_cache

- playable_cache now understands `bgfx:live-source` markup (MSE-9243).

1.8 Known Issues

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

Issue with "Out" Command when Performing Frame Accurate Playback 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 re-initializing. 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 playback,

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).

2 Documentation

Documentation for the Media Sequencer is available in the bundle installer on the [Vizrt FTP](#).

Once installed, documentation is available within the MSE, see the [note](#) above.

3 Support

Support is available at the [Vizrt Support Portal](#).