Vizrt MSE/IGH component (ExternalGraphics Controller)

Description of the mechanism of integration with Vizrt graphics systems via GsCtrl module. Function added in 2.8.0 version.

Licensing

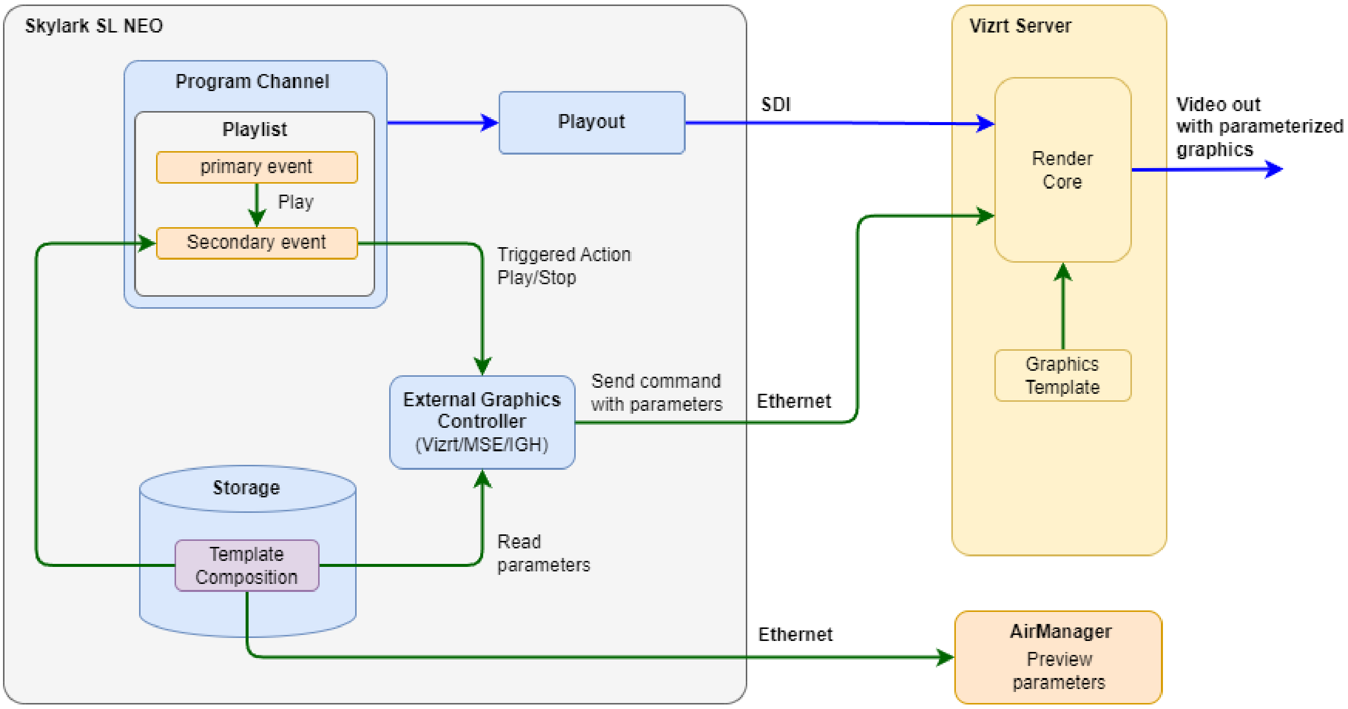
An additional license is required to use this component: Vizrt Software Option “Vizrt Software Option for Vizrt control (control from automation system, from secondary events). License for 1 software channel”

Terms

- **Vizrt scene** - a graphical scene with all animations, textures and scripts.
- **Vizrt template** - created on the basis of Vizrt scene, contains a reference to Vizrt scene, address in GH and a list of control objects of this scene.
- **Page Vizrt** - created based on a Vizrt template, contains a reference to the template.
- **Graphic composition** - basic element, has a name equal to the name of the Vizrt template. It contains elements corresponding to all fields of the Vizrt template. Elements can be of type either Text Object or PIP Frame.
- **Text Parameterization Template** - Refers to a graphic composition. In this template, you can select the fields to be filled in. Each field can have a fixed value, or be able to be changed by the operator. Each field can have a default value. The playlist loaded into the Program module is collected from these templates.

Component Interaction Scheme

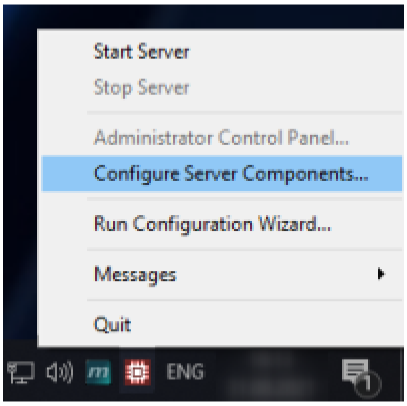
The image shows an example of possible interaction of software modules.



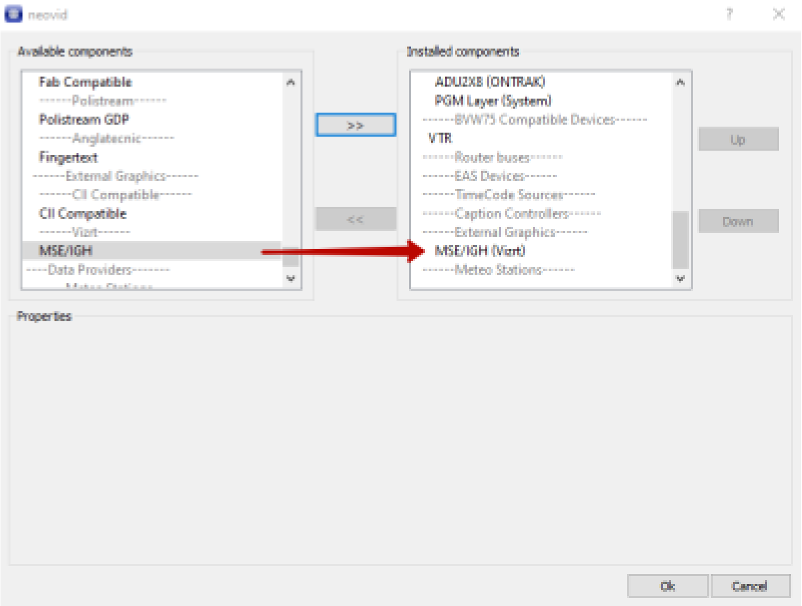
Adding a module

Before adding a new module, stop the server. To do this, right-click on the Neovid icon next to the clock, and then select `Neovid→Stop Server` in the opened window.

Open the `Neovid→Configure Server Components` module configurator window.


Start Server
Stop Server
Administrator Control Panel...
Configure Server Components...
Run Configuration Wizard...
Messages
Quit
ENG

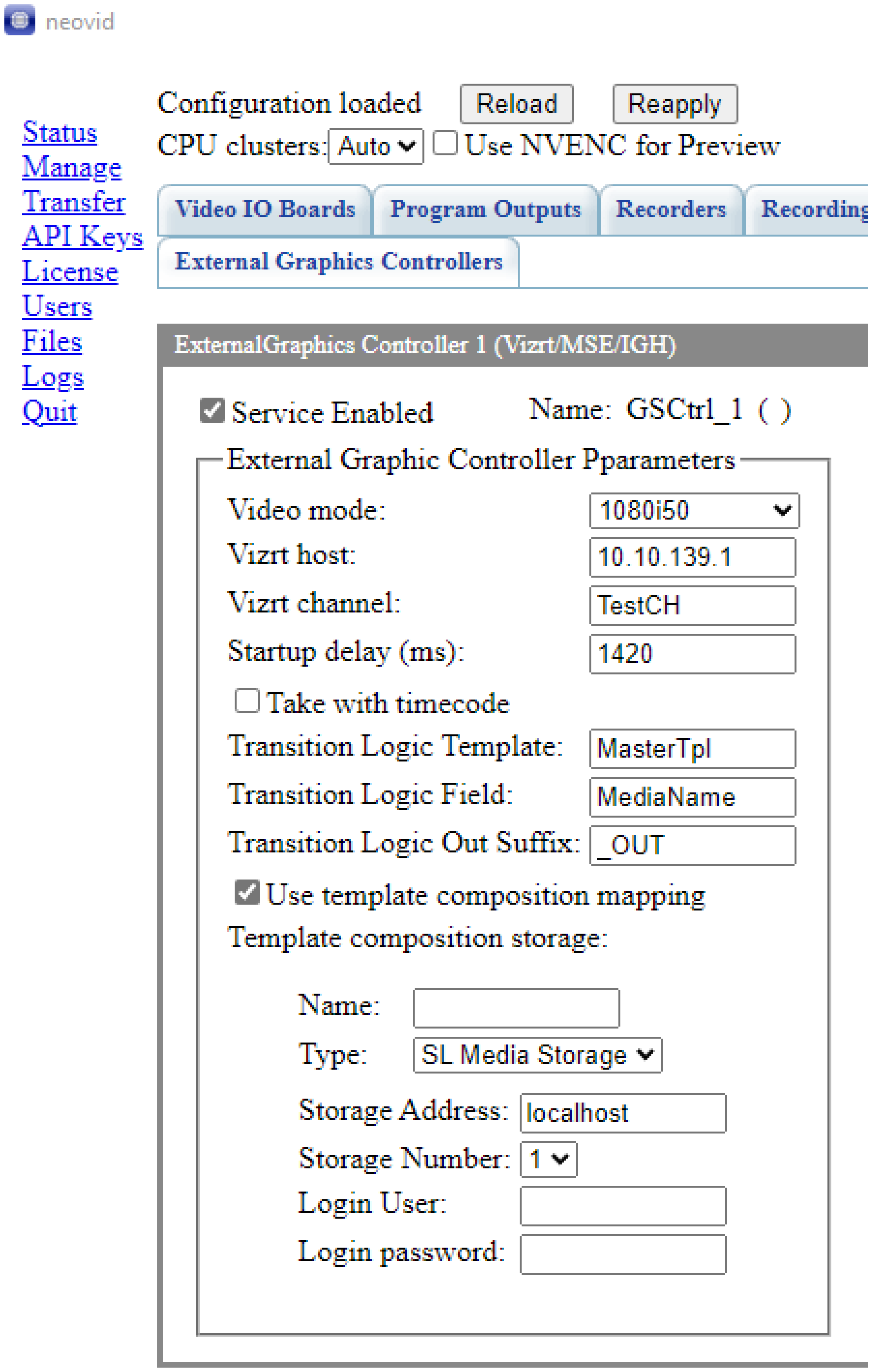
Move the `External Graphics→Vizrt→MSE/IGH` module to the right side of the window using the arrow button. If you want to manage multiple remote graphics systems, you will need to add the appropriate number of modules.



Click OK and start the server `Neovid→Start Server`.

Configuring

Start SL NEO Media Platform, open the server control panel and go to the settings of the added module: `Administrator Control Panel→Manage→External Graphics Controllers→External Graphics Controller N (Vizrt/MSE/IGH)`. Enable the module by checking the `Service Enabled` checkbox.



Configuration loaded

CPU clusters: Auto

□ Use NVENC for Preview

ExternalGraphics Controller 1 (Vizrt/MSE/IGH)

☑ Service Enabled Name: GSCtrl_1 ( )

External Graphic Controller Pparameters

Video mode: 1080i50

Vizrt host: 10.10.139.1

Vizrt channel: TestCH

Startup delay (ms): 1420

□ Take with timecode

Transition Logic Template: MasterTpl

Transition Logic Field: MediaName

Transition Logic Out Suffix: _OUT

☑ Use template composition mapping

Template composition storage:

Name:

Type: SL Media Storage

Storage Address: localhost

Storage Number: 1

Login User:

Login password:

Parameter	Description
Service Enabled	Checkbox to activate component
Name	Service Name
Video mode	Select the video mode to be used NTSC, PAL, 1080i50, etc.
Vizrt host	The address of Vizrt (Media Sequencer)
Vizrt channel	The name of the channel that is set on the Vizrt (Media Sequencer) server

Parameter	Description
Startup delay (ms)	Compensate for the delay in milliseconds. If the value is not set, events will be triggered immediately by “Take”. If the parameter is set, Startup Delay is added to the secondary event generation time and this time is entered into StartTime in the Vizrt playlist, and time-based event triggering must be configured in Vizrt. Time-based triggering allows you to achieve higher accuracy in linking the output of Vizrt events and the main events of the automation playlist. When using the “Play with params” command, the exact time of secondary event generation can be passed from the automation using parameter 5 and by passing the variable “$(event_anchor_gmt)” from the automation playlist.
Take with timecode	When this option is selected, the timecode parameter will be sent to Vizrt instead of the begin_at parameter. To activate this mode in Vizrt it is necessary to switch the mode to “External triggering”.
Transition Logic Template	Specify in this field the name of the universal template used on the Vizrt side. This is necessary when the same template is always used on the Vizrt side with all possible parameters for all possible combinations of elements that have been programmed. The required action is set by the value in the control field. Leave the field empty if you do not want to use a single template for all events. In this case, the name of the template will be taken from the ID passed to the action. In this case, if “Uset template composition mapping” is enabled, a template with the name of the graphic composition referenced by this Text will be used for the Text type ID. If “Transition logic template” or “Transition logic field” is not filled in, the events will not be merged by time, and each event will have a corresponding group and staree created under it.
Transition Logic Field	Specifies the name of the field that is used in the Vizrt side scripts as the “control” field. The Skylark system will send tokens (text commands) to this field that will perform pre-defined actions on the Vizrt side, e.g. turn logo on/off etc.
Transition Logic Out Suffix	Specifies the name of the suffix that will be added to the value passed to the “Transition Logic Field” when the duration of the show expires. The suffix is added when you want to remove a graphic element from the screen. The logic of suffix processing and its hiding is implemented on the Vizrt side.
Use template composition mapping	The parameter enables the mode in which the ID passed to the “Play/Play with params” commands will be searched in the media database and, if a text param object with such ID is found, the template name that is used as a template in the text object will be used as a Vizrt template. Also, the default parameter values that were marked as final when the text object was created will be taken from this text parameterization object.
Template composition storage	Settings for connection to the media database where text parameterization objects and graphic templates will be searched. Used when Use template composition mapping is enabled.
Name	Additional database name
Type	Base type - File Folder or SL Media Storage
Settings for File Folder	
Disk Path	The path to the folder in the file system
Settings for SL Media Storage	
Storage Address	The address of the server running the media storage service
Storage Number	The number of the media database service
Login User	The user name to connect to the media database
Login password	The user's password to connect to the media database

Actions

The customized GsCtrl module is controlled through actions.

In the Program module, you need to add calls to actions of the GsCtrl module: * in Playlist Events, we need to add the generation of a "Play" call on a secondary event for which "Vizrt" or other suitable meaning is specified in Player, * in General Events we need to add a "Stop" call on stopping the main playlist playback.

Play

The "Play" ("Play with params") command performs the following operations on Vizrt:

1. A template with the specified name is requested
2. A page is created from the template with filled in parameters by sequence number. The name for the page is obtained by adding date and time to the template name
3. A group is created in the Vizrt playlist, the name of the group is obtained by adding the date and time to the template name
4. A secondary event referring to the created page is added to the group
5. If the "Startup Delay (ms)" parameter is not set in the module configuration, the created secondary event is given the "take" command.
6. If the "Startup Delay (ms)" parameter is set in the module configuration, then Startup Delay is added to the time of secondary event generation and this time is entered into StartTime in the Vizrt playlist, at the same time, Vizrt must be configured to start events by time. Time-based triggering allows you to achieve higher accuracy in linking the output of Vizrt events and the main events of the automation playlist. When using the "Play with params" command, the exact time of secondary event generation can be passed from the automation using parameter 5 and passing the variable "$(event_anchor_gmt)" from the automation playlist

Stop

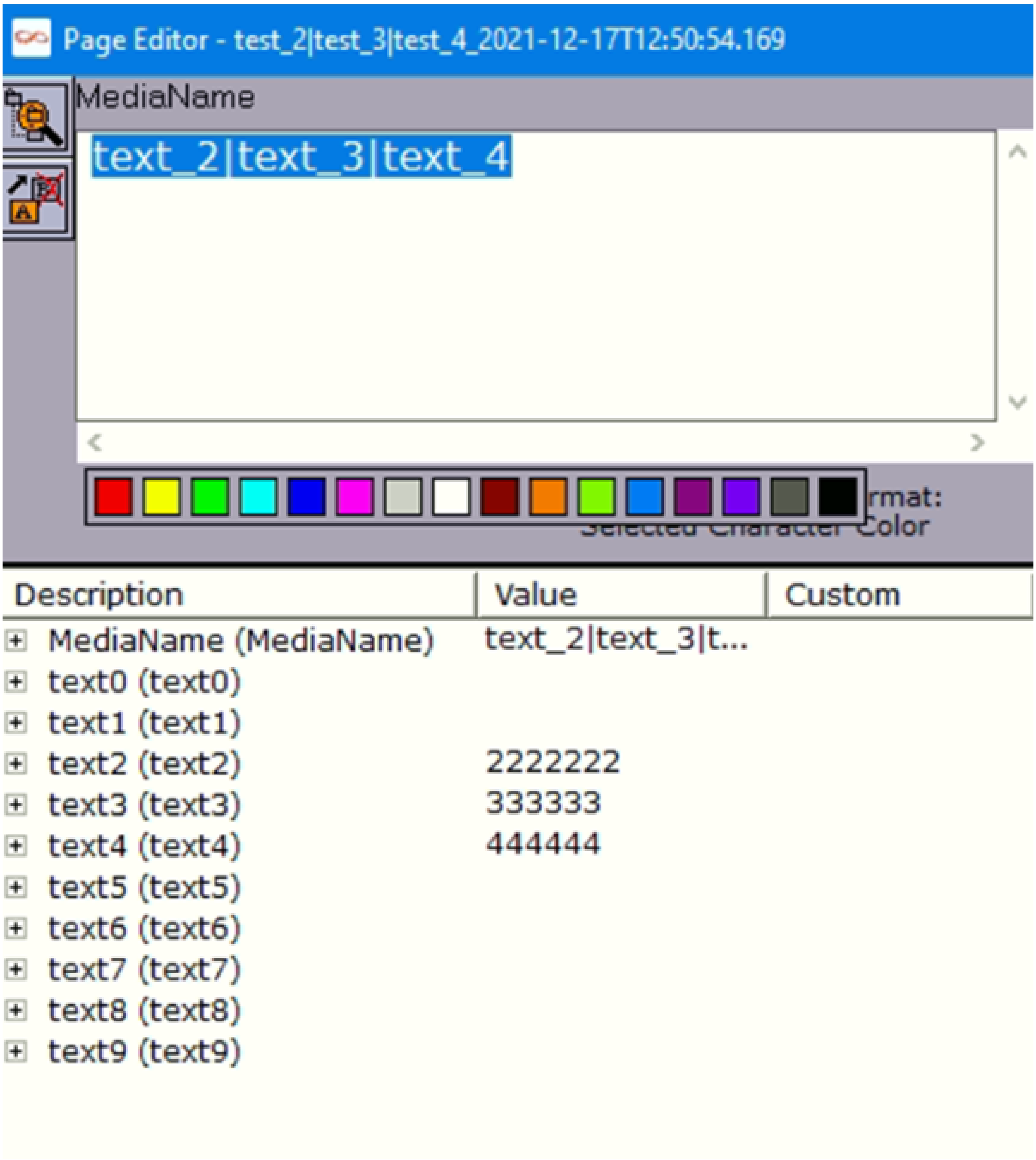
The "Stop" command performs the following operations on Vizrt:

1. VizRt playlist is queried
2. Found events are deleted
3. The pages referenced by the deleted events are deleted.

Working principle

Creating Vizrt template

A Vizrt template with a set of control fields is created on the Vizrt side.



MediaName

text_2|text_3|text_4

Description	Value	Custom
⊞ MediaName (MediaName)	text_2|text_3|t...	
⊞ text0 (text0)		
⊞ text1 (text1)		
⊞ text2 (text2)	2222222	
⊞ text3 (text3)	333333	
⊞ text4 (text4)	444444	
⊞ text5 (text5)		
⊞ text6 (text6)		
⊞ text7 (text7)		
⊞ text8 (text8)		
⊞ text9 (text9)		

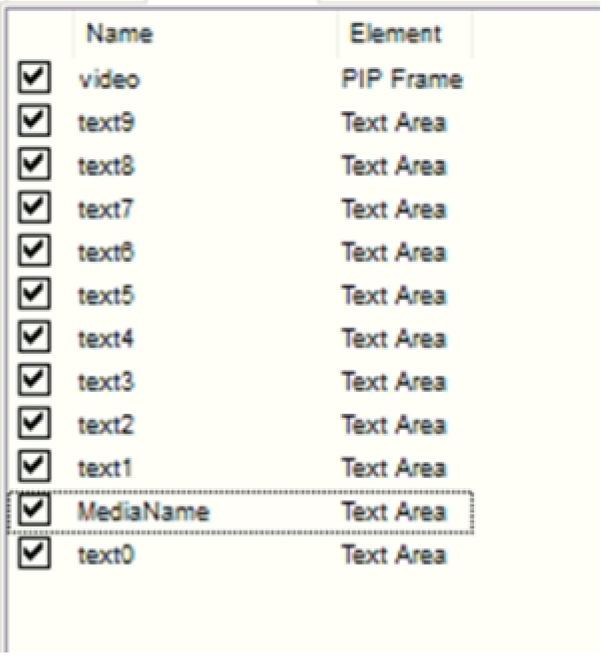
The main control field has a name: MediaName. This field describes the type of graphic being output:

- MediaName=Anons - triggers animation for graphics of the announcement type.
- MediaName=Logo_1 - launches graphics with logo #1.

This field can contain a set of different types of graphics at the same time, separated by “|”. For example: MediaName= Anons|Logo_1. In this case, 2 types of graphics will be aired at the same time.

Creating a graphic composition

The designer creates a set of graphic compositions, which will have the same parameterization fields as those needed to parameterize the corresponding part of the Vizrt scene. Additional data can be added to the templates, such as default parameter values and values for the marker field for the scene combination. Two types of objects are supported: “Text Object” and ‘PIP Frame’.

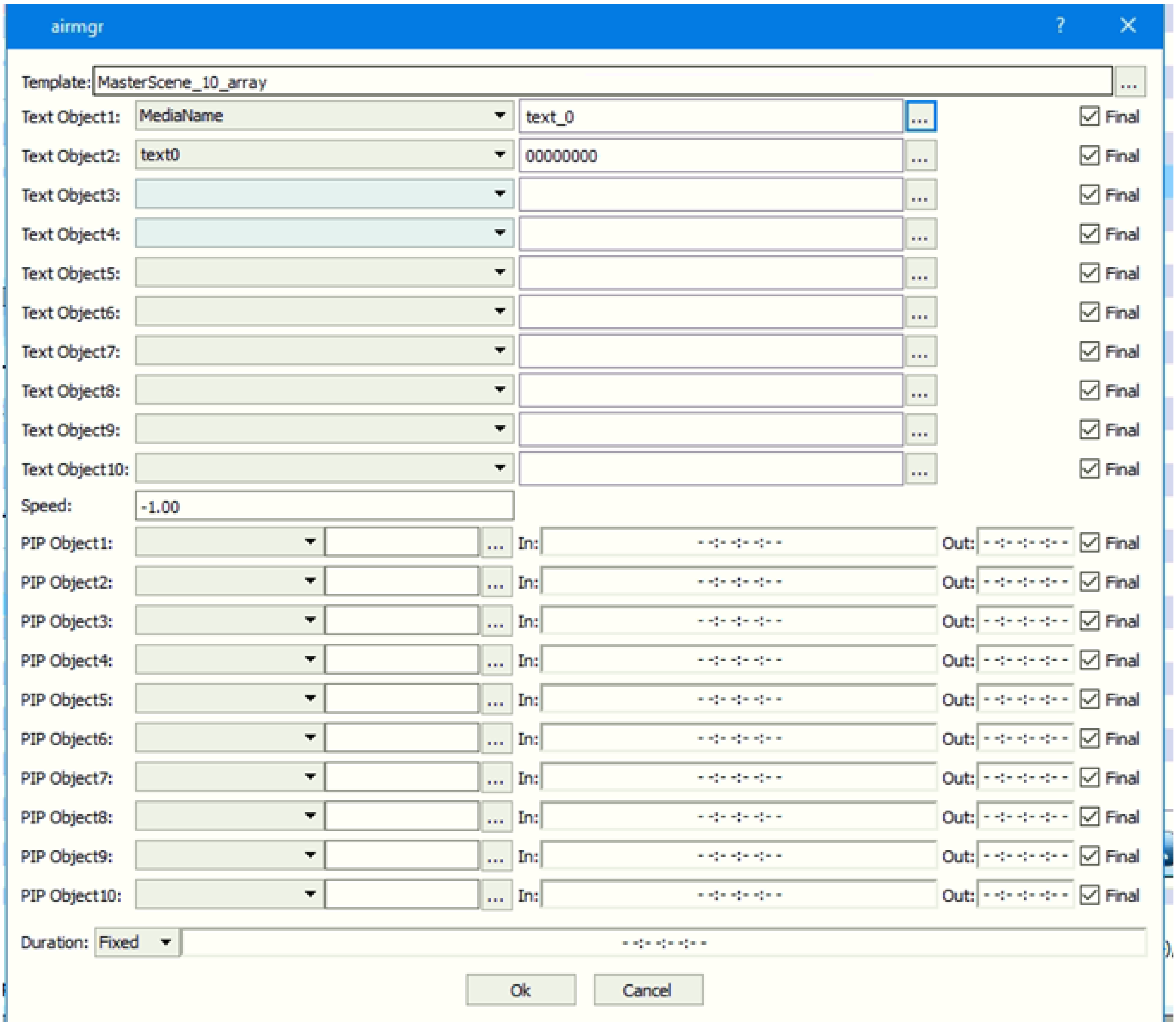


	Name	Element
☑	video	PIP Frame
☑	text9	Text Area
☑	text8	Text Area
☑	text7	Text Area
☑	text6	Text Area
☑	text5	Text Area
☑	text4	Text Area
☑	text3	Text Area
☑	text2	Text Area
☑	text1	Text Area
☑	MediaName	Text Area
☑	text0	Text Area

With the final parameterization of the “PIP Frame”, the operator will be able to select files from the file folder (via SMB/CIFS) and check dynamically for their presence to indicate “NoRef” status in the playlist. In addition, the operator will be able to see a preview of the song in File Monitor with appropriate text and parameterized graphics. The designer does not have to draw the graphic composition exactly as in the Vizrt editor, but can make it similar or not depending on what they want the operator to see.

Text Parameterization Template

This template references the created graphic composition. It also specifies values for the fields to be parameterized. The fields can have a default value. Each field can be made modifiable for the on-air operator, or you can mark that its value is not available to the operator by setting the Final option.



airmgr

Template: MasterScene_10_array ...

Text Object1:	MediaName ▾	text_0	...	☑ Final
Text Object2:	text0 ▾	00000000	...	☑ Final
Text Object3:	▾		...	☑ Final
Text Object4:	▾		...	☑ Final
Text Object5:	▾		...	☑ Final
Text Object6:	▾		...	☑ Final
Text Object7:	▾		...	☑ Final
Text Object8:	▾		...	☑ Final
Text Object9:	▾		...	☑ Final
Text Object10:	▾		...	☑ Final

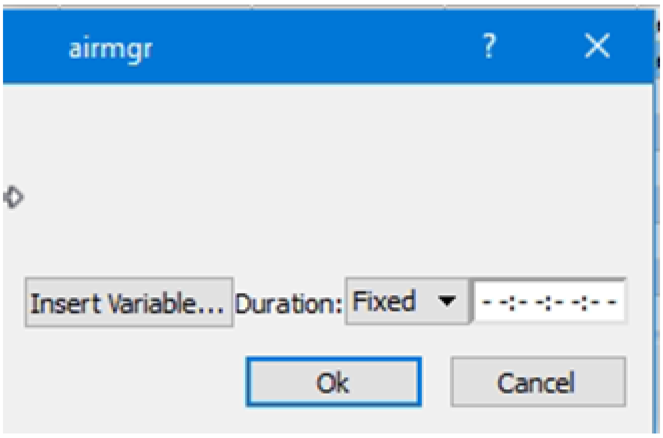
Speed: -1.00

PIP Object1:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object2:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object3:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object4:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object5:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object6:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object7:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object8:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object9:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final
PIP Object10:	▾	...	In: --:--:--:--	Out: --:--:--:--	☑ Final

Duration: Fixed ▾ --:--:--:--

Ok Cancel

When attempting to edit such an object, the ether operator will not see the fields available for editing, as they are all marked Final.

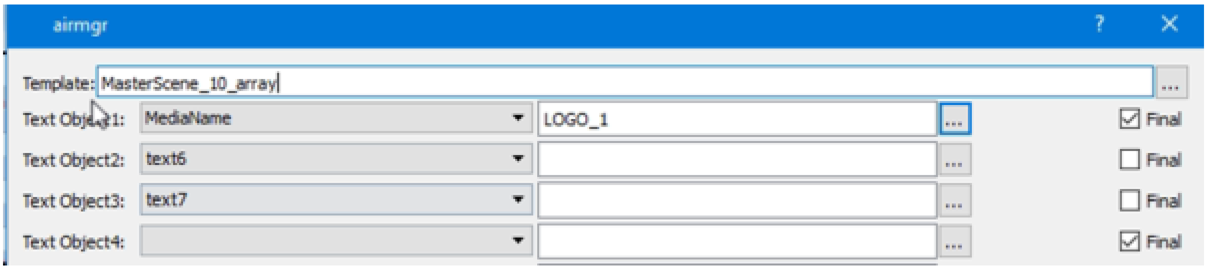


airmgr

Insert Variable... Duration: Fixed ▾ --:--:--:--

Ok Cancel

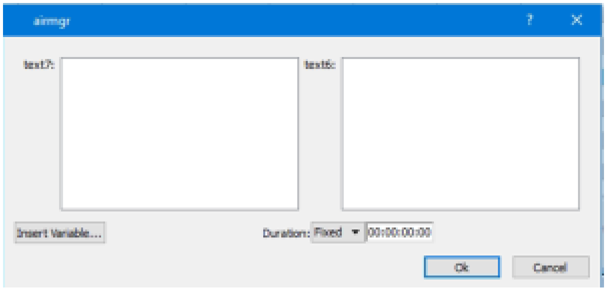
Example of filling in the template:



Template: MasterScene_10_array

Text Object1:	MediaName	LOGO_1	☑ Final
Text Object2:	text6		□ Final
Text Object3:	text7		□ Final
Text Object4:			☑ Final

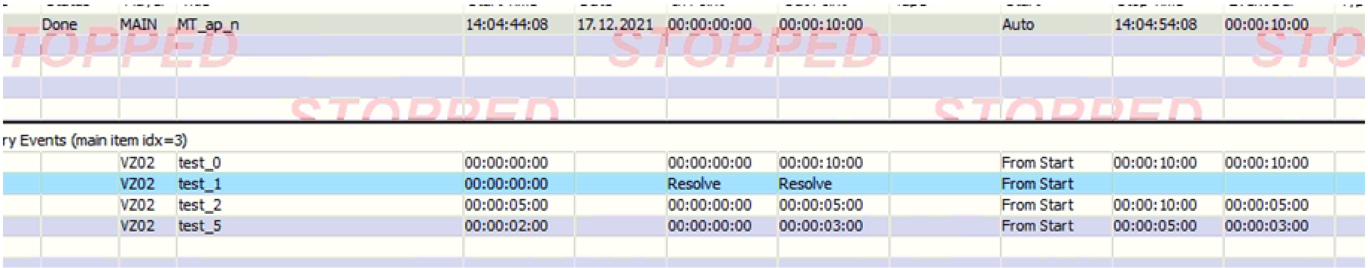
- Graphic Composition Name=MasterScene_10_array
- Control objects: MediaName, text6, text7.
- There are default values: MediaName=LOGO_1.
- The “Final” option is enabled for MediaName, so the operator will have 2 fields available for editing:



Duration: Fixed 00:00:00:00

Putting in a playlist

Either graphic songs or text parameterizations of graphic songs are put into the secondary events list of the automation playlist.



	Done	MAIN	MT_ap_n	14:04:44:08	17.12.2021	00:00:00:00	00:00:10:00		Auto	14:04:54:08	00:00:10:00	

ry Events (main item idx=3)

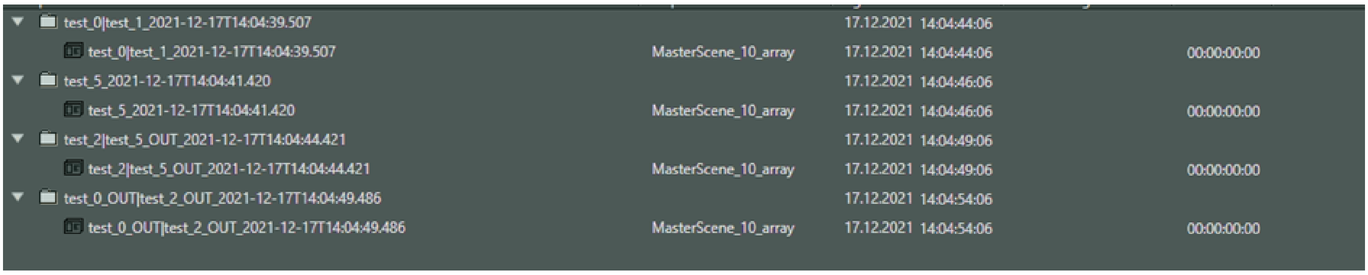
		VZ02	test_0	00:00:00:00		00:00:00:00	00:00:10:00		From Start	00:00:10:00	00:00:10:00	
		VZ02	test_1	00:00:00:00		Resolve	Resolve		From Start			
		VZ02	test_2	00:00:05:00		00:00:00:00	00:00:05:00		From Start	00:00:10:00	00:00:05:00	
		VZ02	test_5	00:00:02:00		00:00:00:00	00:00:03:00		From Start	00:00:05:00	00:00:03:00	

Each element has parameters:

- **Start Time** - determines when this element should exit relative to the start of the primary event,
- **Duration** - not required to be filled in.

When adding a secondary event to the playlist, the Vizrt MSE/IGH component creates:

- a new group in the Vizrt playlist,
- a page with parameters from the template (page),
- a page in the playlist.



test_0|test_1_2021-12-17T14:04:39.507		17.12.2021	14:04:44:06	
test_0|test_1_2021-12-17T14:04:39.507	MasterScene_10_array	17.12.2021	14:04:44:06	00:00:00:00
test_5_2021-12-17T14:04:41.420		17.12.2021	14:04:46:06	
test_5_2021-12-17T14:04:41.420	MasterScene_10_array	17.12.2021	14:04:46:06	00:00:00:00
test_2|test_5_OUT_2021-12-17T14:04:44.421		17.12.2021	14:04:49:06	
test_2|test_5_OUT_2021-12-17T14:04:44.421	MasterScene_10_array	17.12.2021	14:04:49:06	00:00:00:00
test_0_OUT|test_2_OUT_2021-12-17T14:04:49.486		17.12.2021	14:04:54:06	
test_0_OUT|test_2_OUT_2021-12-17T14:04:49.486	MasterScene_10_array	17.12.2021	14:04:54:06	00:00:00:00

Transition Logic feature

To use Vizrt's “Transition logic”, you just need to create one graphical composition with the same name as the main template in Vizrt. In this composition you need to create objects (“Text Object” and “PIP Frame”) with the same names as in the Vizrt template. This composition will be used for parameterization, but can also be used for a schematic preview of the content, as the resulting parameterization will be visible when working with the playlist in FileMonitor.

To control “Transition logic”, the graphics designer can create a set of text parameterization objects. Text parameterization allows you to use a graphic composition as a template, select fields in this composition and set their values. The field values can be marked as final or as open for changes (checkbox Final in the editor). The designer can create text parametrization objects for controlling various graphics elements - logos, tickers, etc., by setting values for control fields and marking them as final, and selecting actual fields that the operator can parametrize from the playlist. Also, he can set default values for some fields.

When setting graphics manually, the operator works with a set of text parameterization objects prepared by the designer. They can be set in the list of secondary events and parameterized in the usual way. Only fields marked as open for parameterization by the designer are available in the parameterization dialog (checkbox Final is not checked). The selection of video files to be placed in “PIP Frame” elements can be done in the usual way in the parameterization dialog via the browser. To do this, a file folder must be added to the storage list in AirMgr, to which the files available on the Vizrt server are published. These files can be selected from there. The playlist will also check if the corresponding files are available in this folder and show “NoRef” statuses if parameterized files are missing.

From the playlist, secondary events will be sent to the Vizrt control module using playlist events. For this purpose, you can select some value for the playlist field “Player” (e.g. “VZ”) and set up an event in “Playlist Events” that triggers this value and calls the “Play with params” action of the Vizrt control module. When calling this action, you need to pass the Media field from the secondary event (text parameterization name) and the list of parameters from the playlist item. The corresponding value for the Player field can be written in the Player field of the text object in the media database (this filling can be done automatically when using Active Folders or Folder Tags), then it will be transferred to the playlist automatically when dragging the secondary event there.

Checking for graphic files to parameterize songs

In Vizrt it is necessary to pass file names without paths and extensions, so if in the parameter value is added, for example, “.avi” then the status of such playlist lines will show “NoRef”. If both a file folder and a media database are added to configure GrStorages at the same time, then components are checked there and there. It is desirable to ensure that the files for Vizrt are only in the file folder, so that if there is no “NoRef” status, you can be sure that the file is in the Vizrt folder and not in some other database. If it is necessary to have files in both the folder and the media base, then this file folder can be added to the media base as Volume.

When Vizrt graphic events are released at the same time, the events are merged into a single page. TransitionLogic control field token values are combined via the “|” character. For example:

- MediaName=Logo_1, begin_time=00:00:00:00:00.
- MediaName=Anons_2, begin_time=00:00:00:00:00.

In Vizrt will be created:

- MediaName=Logo_1|Anons_2, begin_time=00:00:00:00:00.

To make this function work, you need to set the following parameters in the settings in `WebConsole→Manage→ExternalGraphicsControllers`:

- Transition Logic Template - name of the Vizrt template that is used for TransitionLogic,
- Transition Logic Field - name of the control field in the template,
- Transition Logic Out Suffix - suffix to be added to the value of the control field at the end of the duration.

Recommendations

Additionally, it is recommended to use Secondary check FS for params option (`AirMgr→Configure→Actions→Broadcast list actions→Secondary check FS for params`).

Another thing to note is that by default a module Program creates a playlist event “PlayGraphics”, which is triggered on a non-empty ‘Media’ value in the secondary event list. The second condition is that the value in the Player field is not “LOGO”. So, if a command with VZ player is sent to Vizrt and the Media field contains the ID of the template, such secondary events will generate “PlayGraphics” which will not go anywhere because a non-existent player is used there. But, they will also go to AsRun, because the command will be generated and the player will be VZ. Therefore, you need to change the second half of the condition in PlayGraphics from “NOT EQUAL” to “NOT CONTAINS” and add VZ|LOGO to the value.

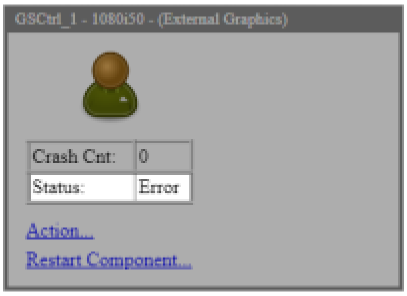
Updates

In version 2.8.17, in the Program module settings in the `Administrator Control Panel→Manage→Program Outputs→Program_N→Automation→AsRunLogs→Report Secondary Events Players` field, you can specify a list of values for the Player column for secondary events. Secondary events with the specified player names will be logged in AsRun.

Since version 2.8.72 the Vizrt control module supports signaling of the current status of exchange with the Vizrt system. The exchange occurs when new secondary events are sent to the Vizrt playlist, when song completion timers are triggered according to their duration, as well as when old items and pages in Vizrt are periodically (every 5 seconds) cleared. If any error occurs during any of these operations, the communication state of the controller with Vizrt is signaled as an error state. The error state is cleared after successful commands resume and there are no error commands for some time.

The communication status can be seen in 2 places:

- on the Status tab in the web console (Status field. When an error occurs, Status=Error)\

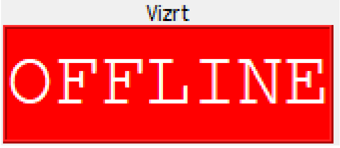


GSCtrl_1 - 1080i50 - (External Graphics)

Crash Cnt:	0
Status:	Error

Action...
Restart Component...

- In AirMgr, you can create a Marker Display that will display the communication status of the controller with Vizrt



Vizrt
OFFLINE

Time Synchronization

NTP time synchronization must be enabled on the Skylark and Vizrt servers for the systems to communicate properly.

From:
http://wiki.skylark.tv/ - **wiki.skylark.tv**

Permanent link:
http://wiki.skylark.tv/modules/components/vizrt

Last update: **2024/09/04 09:00**