

VGLIVE

СОДЕРЖАНИЕ

Концепция	2
Интерфейс	2
Основные команды	3
Избранные наборы	4
Создание и редактирование коллекции	4
Выбор набора из коллекции	4
Формирование набора сюжетов	5
Перетаскивание файлов (Drag-Drop)	5
Буфер обмена (Copy-Paste)	5
Изменение набора	5
Состояние сюжета	5
Навигация	6
Основные команды	6
Последовательное воспроизведение сюжетов	7
Видеосюжеты и картинки	7
Звуковые файлы и группы	7
Титры	7
Играть список	7
Параметры сюжета	8
Глобальные параметры	9
Параметры интерфейса	10
Параметры панели эффектов	11
Параметры таймера	11
Параметры сюжета по умолчанию	12
Внешний таймер	13
Вторичные события	13
Временные параметры событий	13
Создание и редактирование вторичных событий	13
События TCP/IP	14
Конфигурация TCP сервера	14
Создание и редактирование событий TCP/IP	15
Медиафайлы как вторичные события	15
Условные вторичные события	16
Списки вторичных событий	16
Пример	16
Настройка VGLIVE в режиме ведущий-ведомый	17
Настройка ведомого	17
Настройка ведущего	17
Настройка вторичных событий ведущего	17
Запуск сюжета	18
Управление ведомыми сюжетами	18
Метаданные шаблонов	19
Создание и редактирование метаданных	19
Файл метаданных	19
Связывание метаданных с графическими шаблонами	20
Редактор меток (триммер)	20
Запуск триммера	20
Интерфейс триммера	20

Команды триммера	20
Команды навигации внутри клипа	20
Работа с метками	21
Таймкоды	21
Титры	21
Правила оформления шаблонов	21
Заполнение текстовых полей - метаданные	22
Плейлисты в VGLIVE	23
Элементы плейлиста	24
Формирование списка	24
Изменение порядка клипов	24
Управление плейлистом	24
Таймер плейлиста	24
Взаимодействие плейлиста и основных клипов	24
Администрирование VGLIVE	25
Внешний WEB таймер	25
Вывод сюжетов с подготовкой	25
Настройка VGLIVE в режиме резервирования	26
Конфигурация сервера	27
Конфигурация клиента	27
Организация превью на клиентской станции	28
Конфигурация сервера	28
Конфигурация клиента	29
Настройка VGED в режиме резервирования	29

Концепция

VgLive -- это система выдачи видеосюжетов, графики, аудио и титров на базе движка **VgCast**. Программа позволяет оперативно формировать наборы сюжетов и выводить их в эфир. Параметры сюжетов -- способ старта и финиша, входные и выходные метки -- могут изменяться.

Интерфейс

Интерфейс программы состоит из главного меню, инструментальной панели и области для пиктограмм сюжетов. Каждый сюжет может быть видеофайлом (со звуком или без), аудиофайлом, статической картинкой и титром, сюжет в интерфейсе наглядно представлен пиктограммой, информационной строкой и индикатором уровня звука. На пиктограмме сюжета могут появляться дополнительные индикаторы состояния сюжета и/или текстовые данные.

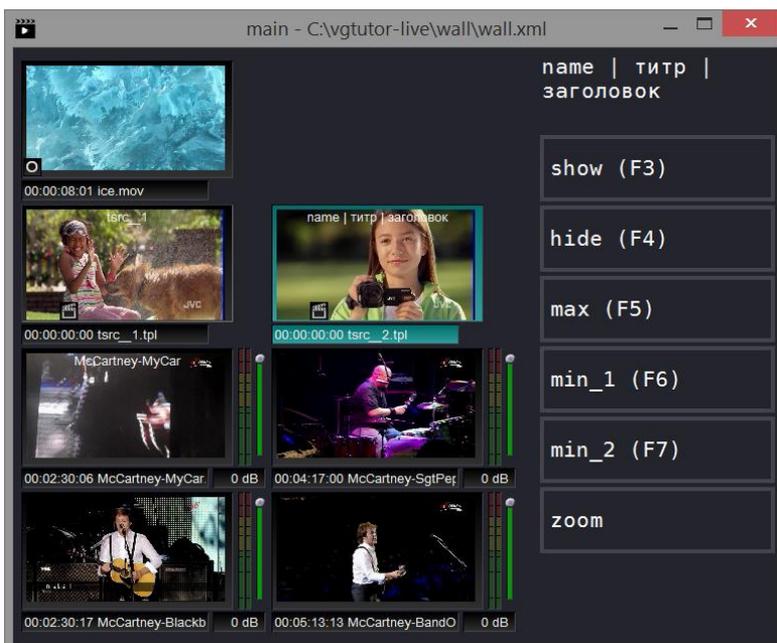


Всем командам программы могут быть назначены клавиатурные эквиваленты (шорткаты).

В описании приведены стандартные комбинации клавиш для шорткатов.

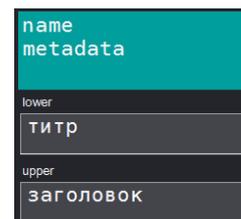
Пользователь может назначить свои комбинации клавиш в меню **Файл - Настройка клавиатуры**.

В зависимости от настроек Глобальных параметров, интерфейс программы может выглядеть по-разному. Так, например, если используется сенсорный экран и большое количество шаблонов с дополнительными эффектами и метаданными, интерфейс может иметь такой вид:



В этом режиме при выборе шаблона в панели эффектов появляются кнопки с именами эффектов и клавиатурными шорткатами, которые были заданы в шаблоне.

Над кнопками находится имя шаблона и список метаданных, которые подставляются в текстовые поля. Если нажать левую мышку на имени шаблона, появляются поля редактирования метаданных:



Основные команды

Эти команды доступны из основного меню и /или инструментальной панели.

 **Файл - Открыть набор - CTRL+O** - прочитать набор из файла.

Появляется стандартное окно выбора файлов.

 **Файл - Сохранить набор - CTRL+S** - записать набор в текущий файл

Файл - Сохранить набор как - записать набор в новый файл.

Сначала пользователю предоставляется стандартное окно выбора файлов.

 **Файл - Новый набор** - создать новый пустой набор.

 **Файл - Проводник** - открыть стандартный проводник Windows.

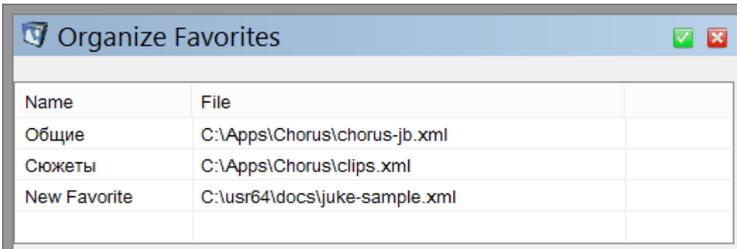
Набор запоминает положение проводника на экране и при запуске набора автоматически запускается проводник и размещается в том месте экрана, где пользователь оставил его в последний раз.

Избранные наборы

В тех случаях, когда нужно оперативно переключаться между несколькими наборами, можно сформировать и использовать коллекцию избранных наборов. В принципе, можно использовать и быстрый доступ к файлам через главное меню **Файл**, но в избранной коллекции наборы можно именовать и пользователь имеет полный контроль за содержимым коллекции.

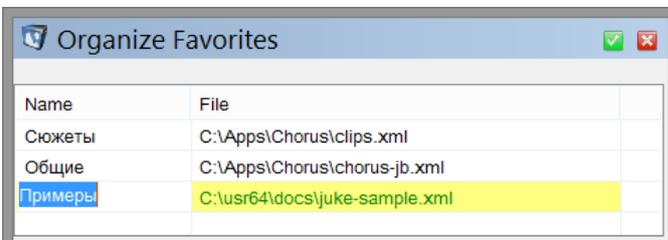
Создание и редактирование коллекции

Чтобы добавить текущий набор в коллекцию, нужно выполнить команды **Редактировать - Добавить в избранное**. Ссылка на файл текущего набора добавляется в коллекцию с именем **New Favorite**. Чтобы работать с коллекцией - редактировать имена и удалять файлы -- нужно выполнить команду **Редактировать - Редактировать избранное**.



Name	File
Общие	C:\Apps\Chorus\chorus-jb.xml
Сюжеты	C:\Apps\Chorus\clips.xml
New Favorite	C:\usr64\docs\juke-sample.xml

Чтобы поменять имя набора в коллекции нужно сделать двойной щелчок левой мышкой по ячейке из колонки **Name**, ячейка переходит в режим ввода текста и можно ввести новое имя с клавиатуры.



Name	File
Сюжеты	C:\Apps\Chorus\clips.xml
Общие	C:\Apps\Chorus\chorus-jb.xml
Примеры	C:\usr64\docs\juke-sample.xml

Правая кнопка мышки в таблице приводит к появлению контекстного меню, в котором приведены стандартные команды работы с буфером обмена и редактирования.

Поменять порядок наборов в коллекции можно перетаскиванием строк в таблице.

Выбор набора из коллекции

Для переключения наборов из коллекции нужно выполнить команду **Избранные наборы** из инструментальной панели  или главного меню **Вид**. При выполнении команды появляется выпадающее меню с перечнем всех наборов коллекции, достаточно выбрать нужный из них.

Формирование набора сюжетов

Добавлять видеоматериал в набор можно несколькими способами.

Перетаскивание файлов (Drag-Drop)

В проводнике нужно выбрать файл (один или несколько) и нажав левую кнопку мыши передвинуть курсор в область набора. При попадании курсора мыши в область набора появляется горизонтальная или вертикальная линия, которая индицирует место вставки сюжетов. Достаточно отпустить кнопку мыши -- и сюжеты будут вставлены в набор.

Буфер обмена (Copy-Paste)

В проводнике выбрать один или несколько файлов и выполнить операцию копирования либо нажав на клавиатуре комбинацию клавиш **CTRL+C** либо вызвать контекстное меню (правая кнопка мыши) и выбрать команду **Сору (Копировать)**. В области набора выбрать событие, за которым нужно вставить скопированные файлы и выполнить операцию вставки, нажав **CTRL+V** или из контекстного меню -- правая кнопка мыши, команда **Paste (Вставить)**.

Изменение набора

Перетаскивание сюжетов. Изменить расположение сюжетов в наборе можно перетаскиванием сюжетов. Для это нужно "схватить" левой кнопкой мышки сюжет за информационную строку и перетащить на новое место. При движении мышки с захваченным сюжетом появляется горизонтальная или вертикальная линия, которая индицирует место вставки сюжетов.

Контекстное меню. В контекстном меню сюжета (правая кнопка мышки на информационной строке) находятся команды буфера обмена

Вырезать (Cut) CTRL+X - сюжет копируется в буфер обмена и удаляется из набора.

Копировать (Copy) CTRL+C - сюжет копируется в буфер обмена и остается на своем текущем месте в наборе

Вставить (Paste) CTRL+V - сюжет из буфера обмена вставляется за текущим сюжетом

Удалить (Delete) DEL - сюжет из буфера обмена вставляется за текущим сюжетом

Вставить строку (Insert Row) CTRL+INS - вставляет перенос строки после текущего сюжета

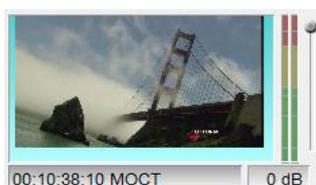
Главное меню программы. В главном меню выбрать пункт **Редактирование**, в нем находятся команды работы с буфером обмена. Команды из главного меню применяются к текущему сюжету (голубая рамка).

Состояние сюжета

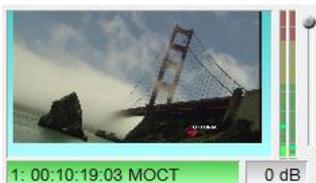
Состояние сюжета отображается различными цветами его компонент.



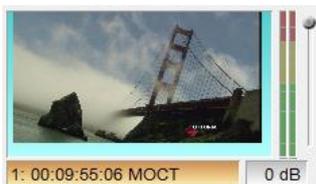
Пассивное состояние **STOP**. Сюжет не выбран, клавиатурные команды для него не работают. Информационная строка показывает длительность сюжета. В этом состоянии запустить сюжет можно только с помощью мыши, нажатием **левой** кнопки мышки на пиктограмму сюжета.



Сюжет выбран, но находится в состоянии **STOP**. Выбрать сюжет можно с помощью мышки -- нажать левую кнопки мышки на информационной строке, или с помощью команд навигации по набору (предыдущий/следующий сюжет). Рамка вокруг пиктограммы имеет ярко-голубой цвет. В этом состоянии сюжет может реагировать на клавиатурные шорткаты.



Сюжет в состоянии воспроизведения **PLAY**. Цвет информационной строки меняется на зеленый, звуковые индикаторы показывают уровень звука. Таймкод в информационной строке показывает, сколько времени осталось до конца сюжета.



Сюжет в состоянии паузы **PAUSE**. Цвет информационной строки желтый. Таймкод показывает, сколько времени осталось до конца сюжета.

Навигация

Для выбора сюжета в наборе можно использовать мышку – левый клик на информационной строке сюжета делает сюжет текущим.

Также можно использовать команды навигации из основного меню программы и их шорткаты –

Home Первый клип

End Последний клип

PgUp Предыдущий клип

PgDn Следующий клип

При выполнении команд навигации следует учитывать, что перемещение между сюжетами происходит сначала по строке, потом выполняется переход на следующую строку.

Основные команды

Главные команды работы с сюжетами -- **Воспроизвести клип (Play)**, **Пауза (Pause)** и **Остановить клип (Stop)**. Эти команды можно инициировать несколькими способами.

С помощью мышки. Нажатие мышкой на пиктограмме сюжета меняет его состояние, **левая** кнопка переключает состояние PLAY и PAUSE, **правая** кнопка выполняет команду STOP.

В глобальных параметрах можно изменить действия кнопок мыши:

Файл – Глобальные параметры – Левая мышка

Контекстное меню сюжета. Нажать правую кнопку мышки на информационной строке сюжета. Появится контекстное меню, из которого можно выбрать нужную команду.

Главное меню программы. В главном меню программы выбрать пункт **Управление**, в нем находятся команды воспроизведения и навигации. Команда применяется к **выбранному** сюжету (голубая рамка).

Комбинация клавиш на клавиатуре (шорткаты). Для каждой команды программы можно задать комбинацию кнопок клавиатуры, которая выполнит эту команду. По умолчанию установлены такие шорткаты

F3 - воспроизвести клип

F4 - остановить клип

F5 - поставить на паузу

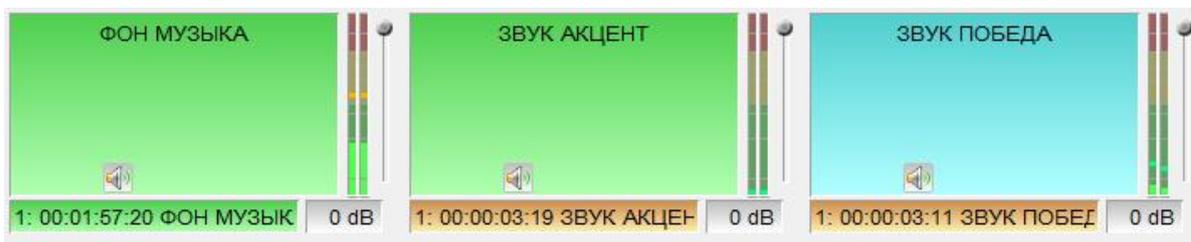
Последовательное воспроизведение сюжетов

Видеосюжеты и картинки

Если в момент выполнения команды воспроизведения какой-либо сюжет уже находится в эфире (либо в состоянии воспроизведения, либо в состоянии паузы), то перед выполнением нового клипа предыдущий снимается с эфира. Это совершенно естественное ограничение, поскольку воспроизведение всех сюжетов выполняется в одном и том же видеобуфере.

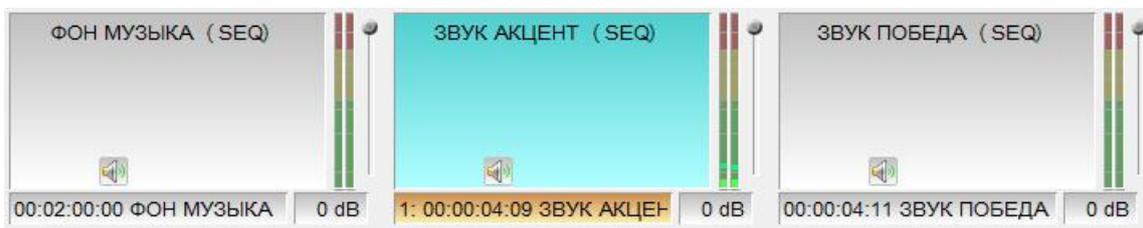
Звуковые файлы и группы

В отличие от видео и фото, запуск на воспроизведение звукового сюжета не приводит к остановке остальных сюжетов. Т.е. одновременно может выполняться видеосюжет со звуком и несколько звуковых сюжетов.



На скриншоте приведен пример одновременного воспроизведения трех звуковых сюжетов.

Однако могут быть ситуации, когда воспроизведение нового звукового клипа должно прервать воспроизведение предыдущего. Для этого звуковые сюжеты можно объединять в именованные группы. Имя группы задается в параметрах сюжета, оно отображается в имени сюжета в круглых скобках, в примере имя группы - SEQ.



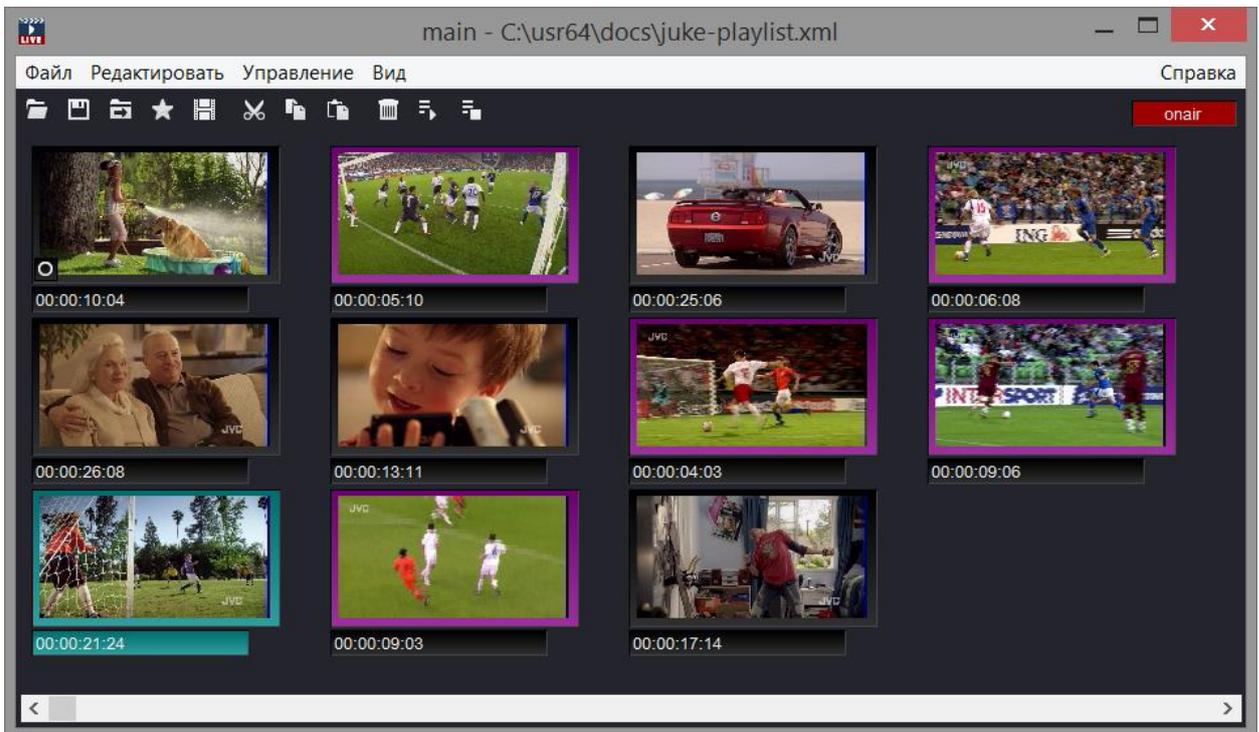
Если запустить клип "ФОН МУЗЫКА" и не дожидаясь его окончания запустить клип "ЗВУК АКЦЕНТ", то исполнение фоновой музыки прервется.

Титры

В качестве сюжетов набора можно использовать файлы графических шаблонов системы VgCast (файлы с расширением *.tpl). Выполнение титровальных сюжетов никак не влияет на уже исполняемые видео/фото/аудио сюжеты. Одновременно может быть запущено произвольное количество титров. Порядок перекрытия титров можно задавать в приоритетах слоев шаблона. Если приоритеты одинаковы, то более поздний титр перекрывает более ранний (если они вообще перекрываются).

Играть список

Кроме индивидуального воспроизведения клипов есть возможность сыграть последовательность клипов, один за другим, встык. Для этого используется команда **Играть все клипы** ▶. Предварительно нужно выбрать те клипы, которые образуют плейлист. Для выбора клипа в плейлист нужно кликнуть на информационной строке сюжета с нажатой клавишей **CTRL**. Цвет пиктограммы клипа меняется на фиолетовый.



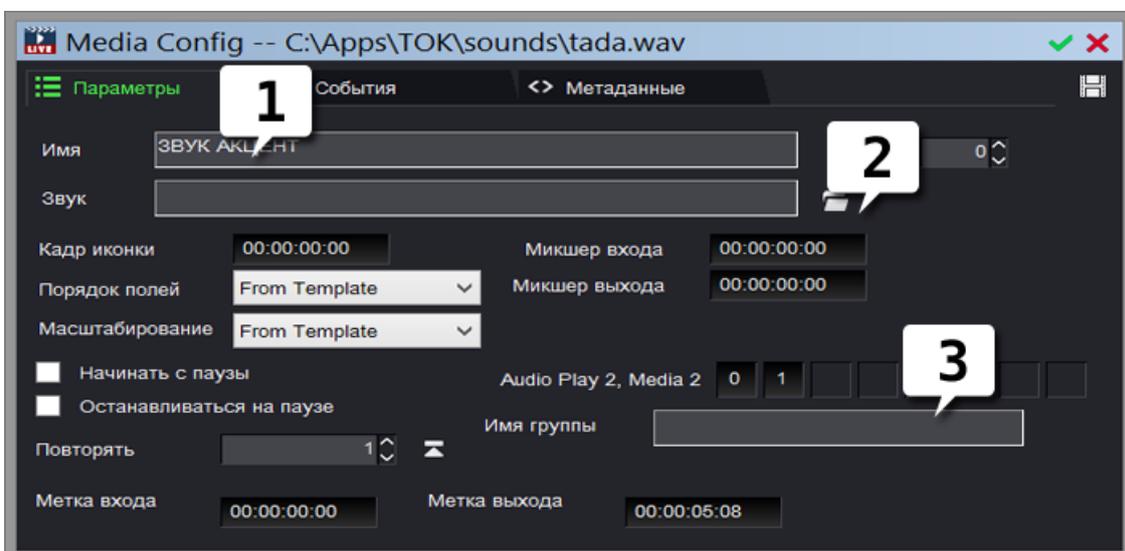
Команда начинает воспроизведение начиная с первого выбранного клипа и заканчивает воспроизведение после последнего выбранного сюжета в наборе. Напоминаю, что порядок перебора сюжетов выполняется сначала слева направо по строке, затем происходит переход на первый сюжет следующей строки, и т.д. Воспроизводятся только отмеченные клипы. Остановить воспроизведение последовательности можно командой **Остановить все (ESC)** на инструментальной панели .

Чтобы снять отметку выбора в плейлист, нужно повторно кликнуть мышкой с нажатой клавишей **CTRL** по информационной строке выбранного клипа.

Если нужно воспроизводить список выбранных клипов в цикле, есть команда и кнопка на инструментальной панели **Играть выбранные в цикле**.

Параметры сюжета

Параметры сюжета можно изменить в диалоговом окне. Для этого нужно выполнить команду контекстного меню **Параметры** (правая кнопка мышки на информационной строке или шорткат **F2**).



В поле **Имя** (1) можно ввести произвольный текст, который будет появляться в информационной строке сюжета, а для сюжетов типа **Звук** или **Титр** - и на пиктограмме сюжета.

Поле **Звук (2)** позволяет с любым сюжетом связать звуковой файл, который начнет воспроизводиться одновременно с сюжетом. При заполнении этого поля на пиктограмме сюжета появляется иконка . Это поле удобно использовать, если видеоклип организован как два файла, отдельно видео и отдельно аудио, либо если титр нужно сопроводить неким звуковым эффектом.

Кадр иконки позволяет задать номер кадра из видеоклипа, который будет использоваться для пиктограммы сюжета. По умолчанию выбирается 50й кадр, но иногда хочется выбрать кадр, который наиболее однозначно представляет собой сюжет.

Микшер входа и **Микшер выхода** позволяют задать ввод/вывод сюжета плавно, микшером за указанное время.

Порядок полей указывает, какой использовать порядок полей при выводе видео. Может принимать значения

From Template - брать порядок полей из шаблона набора (файл **vglive.tpl**)

Lower First - первым идет нижнее поле

Upper First - первым идет верхнее поле

Масштабирование удобно использовать в тех случаях, когда геометрия контента не совпадает с геометрией устройства вывода, например, когда устройство работает в формате FULL HD, а выводим контент SD. Можно задать пропорциональное масштабирование (с сохранением соотношения сторон) либо анаморфное, с искажением по горизонтали и вертикали.

Начинать с паузы - при воспроизведении такого сюжета на экран выводится первый кадр и сюжет переводится в состояние паузы. На пиктограмме сюжета появляется иконка . Обычно используется на сюжетах с альфа-каналом, например для воспроизведения анимированной заставки при переходе с рекламы на программу. Такая заставка начинается с полностью непрозрачного альфа-канала и завершается полностью прозрачным альфа-каналом.

Останавливаться на паузе - при воспроизведении такого сюжета последний кадр видеопоследовательности остается на экране (по умолчанию, после воспроизведения видео экран очищается, становится прозрачным для внешнего сигнала или DSK микшера). На пиктограмме сюжета появляется иконка . Обычно используется для сюжетов с альфа-каналом при переходе из программы на рекламу. Этот режим автоматически применяется для статических картинок или видеосюжета, состоящего из одного кадра.

Если установлены оба режима постановки на паузу, то на пиктограмме сюжета появляется иконка .

Повторять - задает количество повторений сюжета. Обычно используется для вывода зацикленных фонов. Для того, чтобы повторять сюжет "бесконечно" (до команды STOP), можно нажать кнопку .

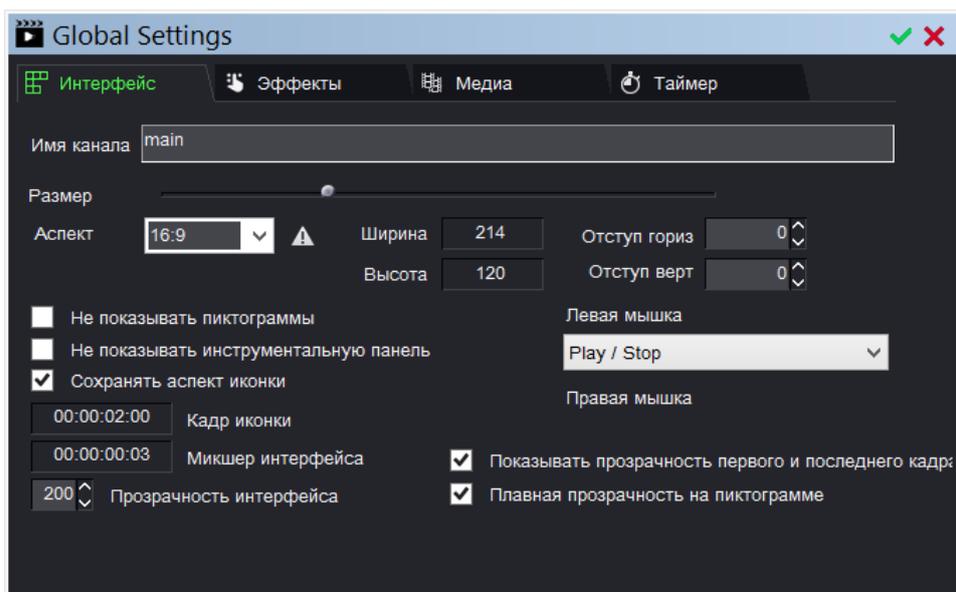
Audio (3) - задает переназначение звуковых дорожек в файле.

Метка входа и **Метка выхода** - позволяют воспроизводить не весь сюжет, а только выбранный сегмент. Обычно эти метки задаются с помощью редактора меток.

Глобальные параметры

Диалог настройки общих параметров набора можно открыть через основное меню **Файл - Глобальные параметры**. Некоторые из параметров дублируют параметры сюжета, но на самом деле они задают значения по умолчанию. Например, если в глобальных параметрах установлен порядок полей как **Lower First**, то при добавлении нового клипа в набор ему будет присвоен именно этот атрибут.

Параметры интерфейса



Имя канала - произвольный текст, который выводится в заголовок окна набора. Удобно использовать в тех случаях, когда с одной станции вещается несколько каналов.

Аспект - соотношение сторон для пиктограмм сюжетов.

Ширина, Высота или **Размер** задают размер пиктограмм сюжетов.

Помнить N последних наборов - запоминает имена последних N наборов в меню **Файл** главного меню программы.

Микшер интерфейса, Прозрачность интерфейса - всякие украшения.

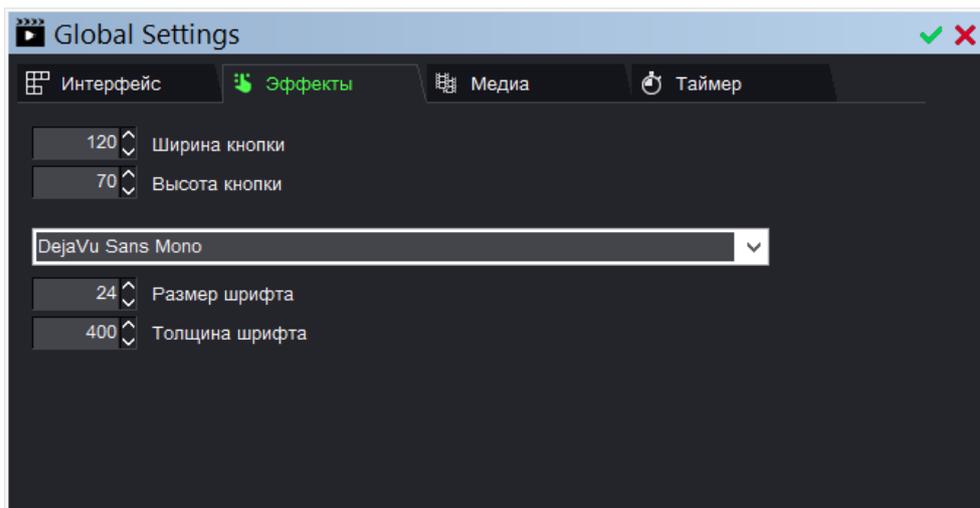
Не показывать пиктограммы – не показывает пиктограммы сюжетов. В таком режиме помещается очень много сюжетов на страницу.

Не показывать инструментальную панель – убирает кнопки инструментальной панели. Выбор этого режима дублируется в системном меню программы – **Минимальный интерфейс | Нормальный интерфейс**

Левая мышка – выбирает режим работы левой кнопки мыши при нажатии на пиктограмму сюжета. Может принимать три значения:

- **Play / Pause** – первое нажатие левой кнопки запускает воспроизведение сюжета, второе – постановку на паузу. Правая кнопки мыши останавливает сюжет.
- **Play / Stop** – первое нажатие левой кнопки запускает воспроизведение сюжета, второе – останавливает сюжет. Правая кнопка ставит сюжет на паузу. Если сюжет это шаблон, то выводится список доступных эффектов. Такой метод удобнее при работе с сенсорным экраном, если не используются шаблоны с выбором дополнительных эффектов.
- **Select** – происходит только выбор сюжета или шаблона. Правая кнопка выводит меню сюжета или список доступных эффектов. Такой режим удобен при использовании сенсорного экрана и панели эффектов.

Параметры панели эффектов

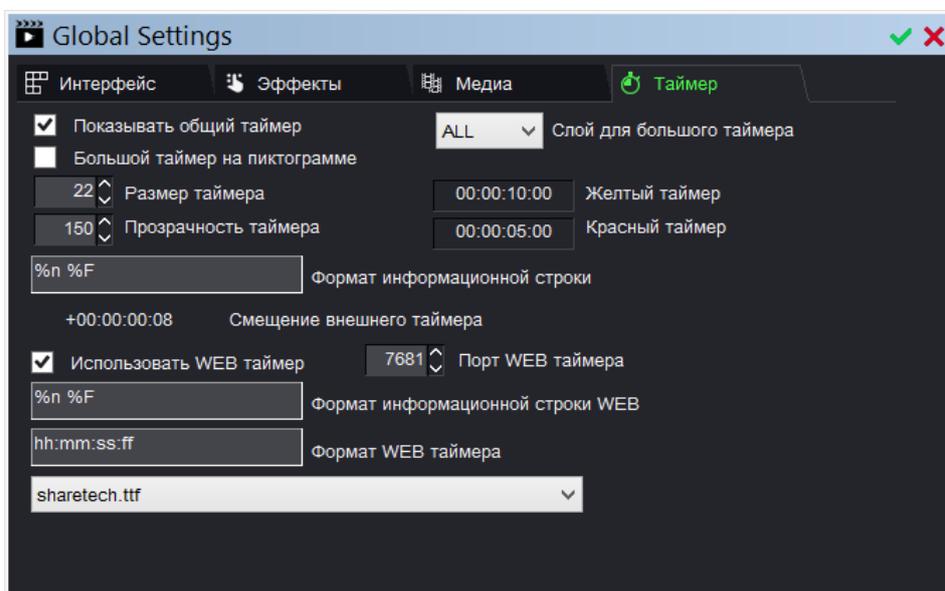


Ширина кнопки – ширина панели эффектов в пикселях.

Высота кнопки – высота каждой кнопки вызова эффекта.

Можно задавать шрифт, которым выводятся надписи на кнопках эффектов.

Параметры таймера



Показывать общий таймер - при воспроизведении видеоклипов на инструментальной панели набора показывается таймер обратного отсчета. Параллельно таймер обратного отсчета работает в информационной строке сюжета.

Большой таймер на пиктограмме - показывать большой таймер на пиктограмме (только для Windows 8).

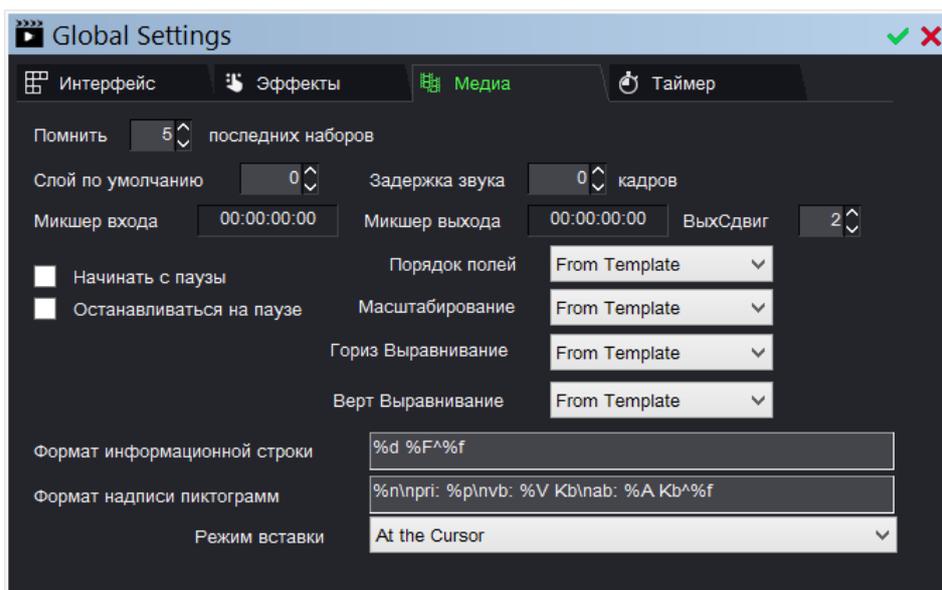
Размер таймера, Прозрачность таймера - высота цифр в таймере и их прозрачность

Желтый таймер - если до конца сюжета осталось меньше времени, чем указано в этом поле, то таймер обратного отсчета становится желтым.

Красный таймер - если до конца сюжета осталось меньше времени, чем указано в этом поле, то таймер обратного отсчета становится красным.

По цвету таймера легко контролировать последние секунды воспроизведения клипа и информировать режиссера о том, сколько времени осталось до конца сюжета.

Параметры сюжета по умолчанию



Кадр иконки указывает, какой кадр брать для пиктограммы сюжета по умолчанию.

Микшер входа, Микшер выхода, Порядок полей, Масштабирование, Начинать с паузы, Останавливаться на паузе – все эти параметры задают значения по умолчанию.

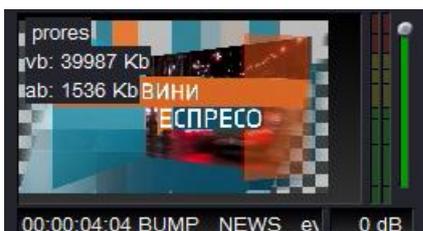
Формат информационной строки указывает, какая информация выводится в поле под пиктограммой сюжета. Формат надписи пиктограмм задает формат надписи на самой пиктограмме. Для добавления дополнительной информации в эти поля можно использовать такие метасимволы форматирования.

- %d** – длительность клипа
- %v** – количество видеопотоков
- %a** – количество аудиоканалов
- %n** – имя клипа
- %f** – полное имя файла с путем
- %F** – имя файла без пути
- %l** – номер слоя
- %V** – видеобитрейт Kbit
- %A** – аудиобитрейт Kbit
- %p** – приоритет слоя или шаблона
- \n** – переход на следующую строку
- ^** – за этим знаком следует текст всплывающей подсказки, в котором тоже можно применять форматирование, переход на новую строку в подсказке не работает.

Остальные литеры переносятся из формата в надпись пиктограммы «как есть». Например, строка

%n\nvb: %V Kb\nab: %A Kb^%f

сформирует такую надпись на пиктограмме (имя клипа **prores**):



Внешний таймер

В меню Вид можно выбрать пункт Внешний таймер, при этом открывается (или закрывается) отдельное окно, размер и положение которого можно изменять.



В этом окне отображается таймер обратного отсчета для выбранного клипа и имя клипа.

Вторичные события

С каждым клипом может быть связан список вторичных событий -- графические объекты (шаблоны) и команды для отправки в сокет TCP/IP. Для вторичных событий задается время относительно клипа (его начала или конца).

Временные параметры событий

Для каждого события имеет смысл три временных параметра -- время начала события (**start**), время окончания события (**end**) и длительность события (**dur**). Эти параметры связаны простой формулой

$$\mathbf{dur = end - start}$$

Для полного задания времени вторичного события достаточно определить любые два из этих трех параметров, оставшийся вычисляется автоматически.

Поскольку единицы времени для вторичных событий относительны, то значения времени -- это смещения относительно начала или конца первичного события, и кроме значений времени нужно задать еще тип параметра.

Значение времени вторичного события может задаваться

- Относительно начала первичного события (**start, end**)
- Относительно конца первичного события (**start, end**)
- Вычисляемое на основании остальных параметров (**start, end, dur**)
- Фиксированное (**dur**)

Значение времени вторичного события может быть как положительным, так и отрицательным.

Создание и редактирование вторичных событий

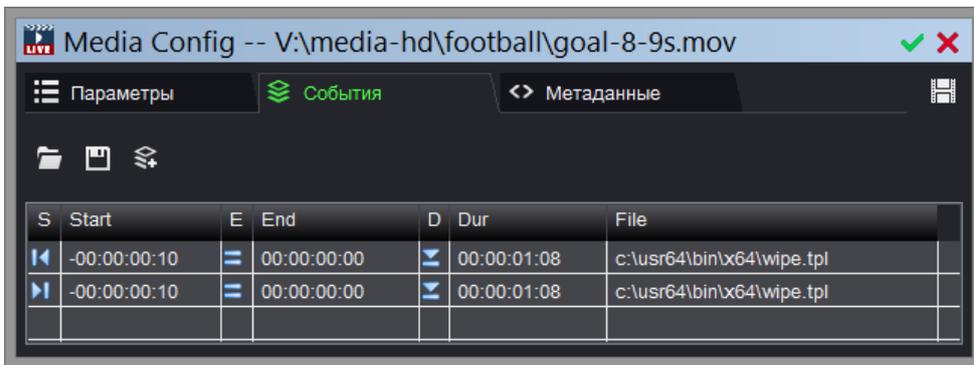
В таблице вторичных событий можно менять временные параметры событий, задавать тип начала, окончания и длительности событий, удалять и добавлять события. Колонки таблицы с заголовками S, E и D отображают тип временного параметра в виде пиктограммы.

☐ -- время относительно начала первичного события

☐ -- время относительно конца первичного события

☐ -- фиксированное время (только для длительности)

☐ -- вычисляемое время



Для создания нового вторичного события достаточно

- перетащить мышкой файл графического шаблона из проводника в таблицу
- выбрать на инструментальной панели команду "Добавить событие" и выбрать нужный файл в стандартном окне выбора файлов

Для редактирования привязки времени события нужно щелкнуть правой кнопкой мышки в соответствующей колонке (S, E или D) и из контекстного меню выбрать нужный тип временного параметра.

Для редактирования значения времени нужно ткнуть левой кнопкой мышки в поле времени и эта ячейка таблицы перейдет в режим ввода текста. Время нужно вводить в формате HH:MM:SS:FF. Нулевые старшие разряды формата можно опускать, например 3:10 будет введено как 00:00:03:10. Если в поле редактирования времени вводить число без разделителей (двоеточия) то это число интерпретируется как количество кадров, т.е. если ввести число 54, то оно будет введено как 00:00:02:04.

События TCP/IP

В качестве вторичных событий можно также использовать команды, которые отправляются в любой открытый сокет TCP/IP. Чтобы использовать события такого типа, необходимо в конфигурационном файле описать TCP-сервер, который будет принимать такие команды.

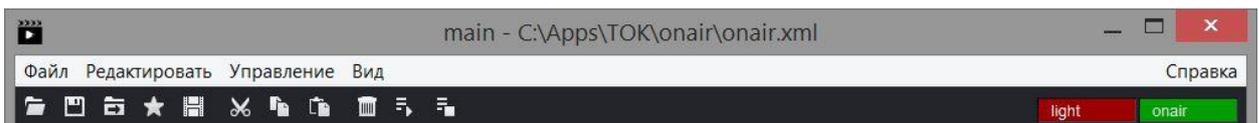
Конфигурация TCP сервера

Сервер описывается в теге `<engine>` в файле `vglive.cfx`. Рассмотрим в качестве примера сервер, который управляет осветительным оборудованием.

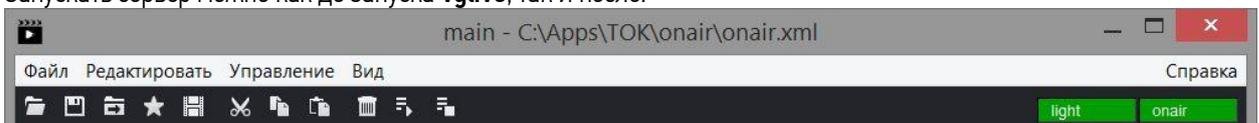
```
<engine run="" host="127.0.0.1:5002" name="light"/>
```

В параметре `host` задается IP-адрес сервера и порт, в параметре `name` -- имя сервера, на это имя нужно ссылаться во вторичных событиях. Предполагается, что команды, которые нужно отправлять в TCP-сервер -- это текстовые строки.

Индикатор состояния сервера появляется в правом верхнем углу интерфейса, красным цветом указывается, что сервер не запущен.



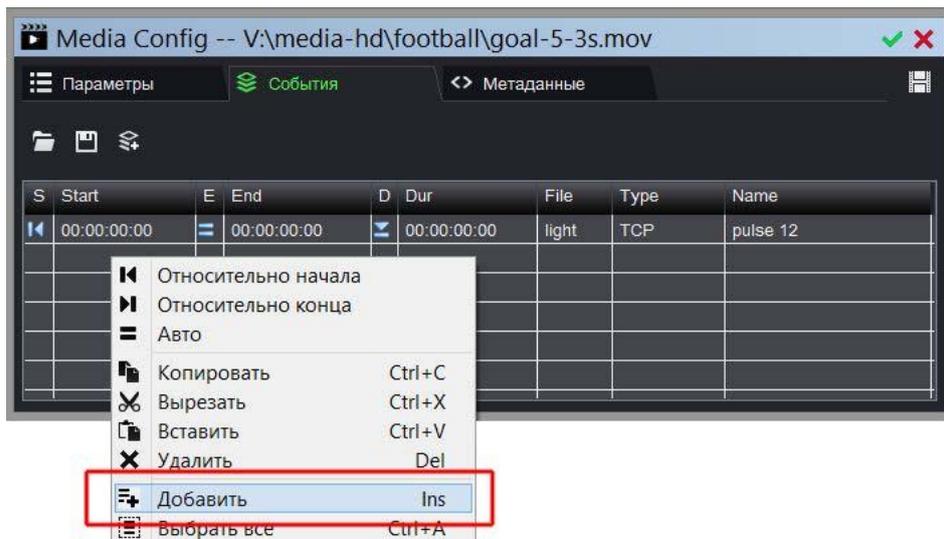
Запускать сервер можно как до запуска `vglive`, так и после.



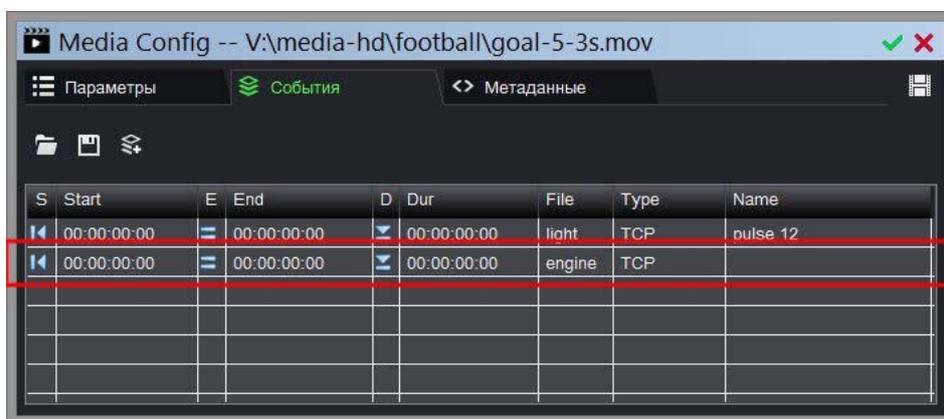
После успешного соединения с сервером индикатор меняет свой цвет на зеленый.

Создание и редактирование событий TCP/IP

Чтобы добавить событие TCP/IP в список вторичных событий, нужно открыть редактор вторичных событий через контекстное меню клипа -- правая мышка -- **Вторичные события**, и в таблице вторичных событий через контекстное меню выполнить операцию **Добавить**.



В таблицу событий добавляется новое событие

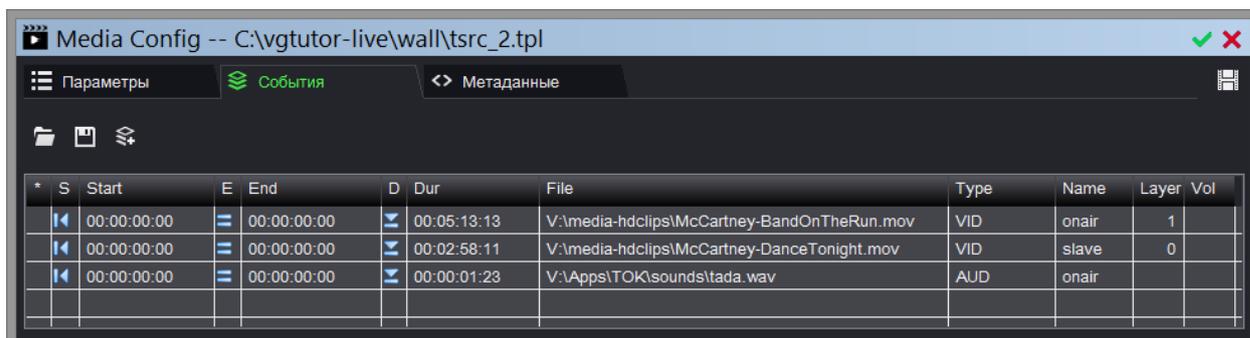


В поле **File** нужно вписать имя сервера, указанное в параметре **name** тега **<engine>** (в нашем примере сервер **light**), а в поле **Name** -- текст команды, которая будет отправлена серверу (**pulse 12** -- поднять яркость на 12м канале DMX и потом вернуть ее в 0).

Редактировать поля **File** и **Name** можно двойным щелчком мыши. Временные параметры событий TCP/IP задаются так же, как для графических событий. Длительность событий TCP/IP игнорируется.

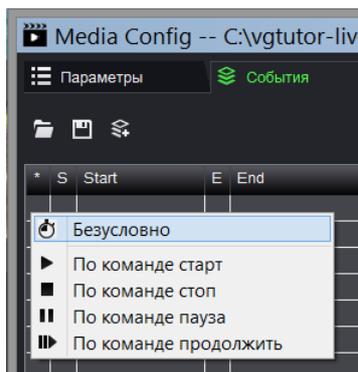
Медиафайлы как вторичные события

В таблицу вторичных событий можно перетаскивать медиафайлы. Для таких событий можно задавать имя движка, на котором будет выполняться событие (или all – выполнить на всех движках) и номер слоя (для видеофайлов).



Условные вторичные события

Кроме выполнения вторичных событий по относительно времени сюжета, можно выполнять их привязку к командам оператора.



Для этого нужно правой мышкой щелкнуть в колонке * таблицы вторичных событий и выбрать команду, к которой будет привязано событие.

По умолчанию вторичные события выполняются **Безусловно**.

Если выбран другой тип запуска, то вторичное событие выполняется точно в момент выполнения соответствующей команды: **Старт, Стоп, Пауза** или **Продолжить**. При этом учитываются положительные временные смещения, заданные в колонке **Start** таблицы.

Списки вторичных событий

Достаточно часто для графического оформления видеоматериала применяется одна и та же последовательность вторичных событий. Каждый раз формировать список вторичных событий достаточно "скучно". Чтобы избежать такого дублирования, можно сохранить имеющийся список событий в файл и затем просто использовать этот сохраненный список для оформления других первичных событий. Для сохранения списка событий нужно выбрать команду "Сохранить"  из инструментальной панели и задать имя файла для списка.

Назначить список вторичных событий первичному событию можно несколькими способами.

Перетаскивание -- из проводника захватить файл мышкой и перетащить в таблицу событий. Если открыт редактор вторичных событий, то можно перетаскивать файл списка в окно редактора вторичных событий, или использовать операции копирования-вставки   .

Пример

Очень характерно использование вторичных событий в оформлении музыкальных клипов. Как правило, сразу после начала клипа (через 1 секунду) нужно показать титр с исполнителем и названием песни (продолжительность титра 3 сек), и этот же титр показать незадолго до окончания клипа (например, он должен длиться те же 3 сек и закончиться за 1 сек до окончания клипа).



Первая строка в таблице задает графическое событие title.tpl у которого тип старта задан **относительно начала** клипа со смещением в 1 сек. Длительность титра задана как 3 сек и тип этого параметра - фиксированный. Конец титра будет вычислен как начало плюс длительность титра (4 сек). Если в качестве времени начала клипа указать **отрицательное** число (например, -00:00:03:00), то титр будет выведен за 3 секунды **до начала** клипа. Это совершенно легальный способ задания вторичных событий, но нужно учитывать, что метаданные в этот момент времени сформированы на основании **предыдущего** первичного события.

Вторая строка задает финальный титр клипа, то же графическое событие title.tpl, но для него задано время окончания титра как смещение -1 сек **относительно конца** клипа. Длительность задана как фиксированный параметр 3 сек. Начало титра будет динамически вычисляться как end - dur, т.е. титр начнется за 4 секунды до конца клипа и окончится за 1 сек до конца клипа. Время заданное относительно конца клипа может принимать **положительные** значения, и это означает, что вторичное событие будет выполняться **после** окончания клипа. Опять таки, в таких случаях нужно учитывать, что метаданные будут сформированы уже для следующего первичного события.

Третья строка в таблице задает событие, которое начинается через 4 сек 10 кадров после начала клипа и завершается за 4 сек 10 кадров до окончания клипа, т.е. длина этого события вычисляется динамически, и будет зависеть от длины клипа. Такое использование графики характерно для показа бегущих строк, чатов и пр...

Настройка VGLIVE в режиме ведущий-ведомый

Постановка задачи – есть ситуация, когда нужно синхронно выдавать разные сюжеты в разные каналы, например один сюжет выдавать на экраны в студию, а второй – в канал программы. Для реализации такой схемы работы нужно использовать два экземпляра **VgLive**, один ведущий (для экранов), второй – ведомый (для программы).

Настройка ведомого

Чтобы создать экземпляр ведомого **VgLive** нужно скопировать файл **vglive.exe** в файл с именем **vgslave.exe**, а файл с именем **vglive.cfx** – в файл **vgslave.cfx**. Если уже есть экземпляр для канала программы – можно использовать его.

Модификация файла конфигурации **vgslave.cfx**:

Открывать можно любым текстовым редактором, но желательно использовать редактор формата XML, чтобы была подсветка синтаксиса.

Ищем тег

```
<cmd_port>0</cmd_port>
```

и 0 заменяем на любой свободный номер порта TCP/IP, например 5010. При старте программы на этом порту запускается TCP-сервер, и выполняет команды, которые поступают в этот порт.

```
<cmd_port>5010</cmd_port>
```

Для удобства работы ищем тег **<channel>** и меняем в нем имя канала

```
<channel>pgm</channel>
```

Имя **pgm** будет появляться в заголовке окна программы. Собственно, на этом настройка ведомого заканчивается.

Настройка ведущего

Редактируем файл **vglive.cfx**. В него нужно добавить описание интерфейса с ведомым. Добавляем такую строку (чтобы было удобнее где-нибудь рядом с уже существующими описаниями движков):

```
<engine run="" host="127.0.0.1:5010" name="slave"/>
```

В параметре **host** должен быть указан IP-адрес и порт станции, на которой запускается ведомый, если он запускается на той-же станции, задаем адрес **127.0.0.1**. В параметре **name** можно указать любой имя.

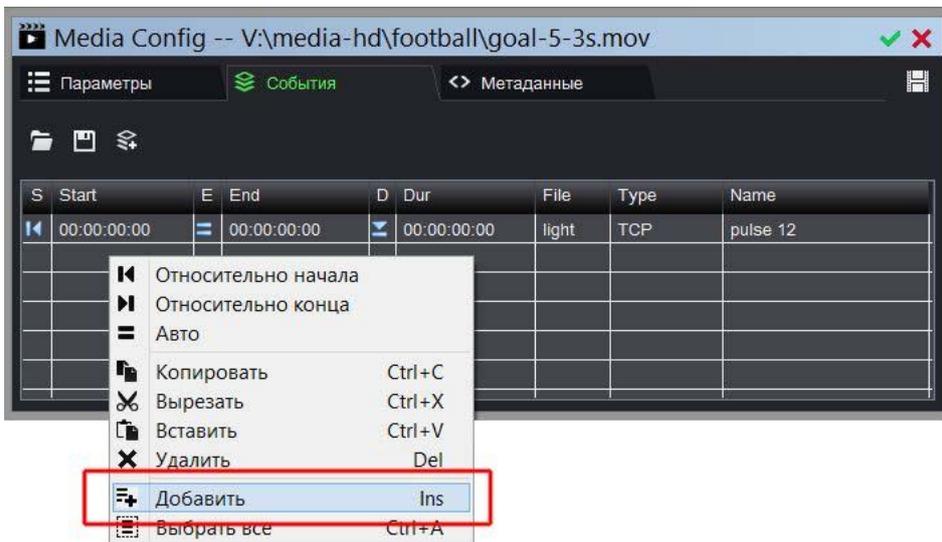
Ищем тег

```
<cmd_port>0</cmd_port>
```

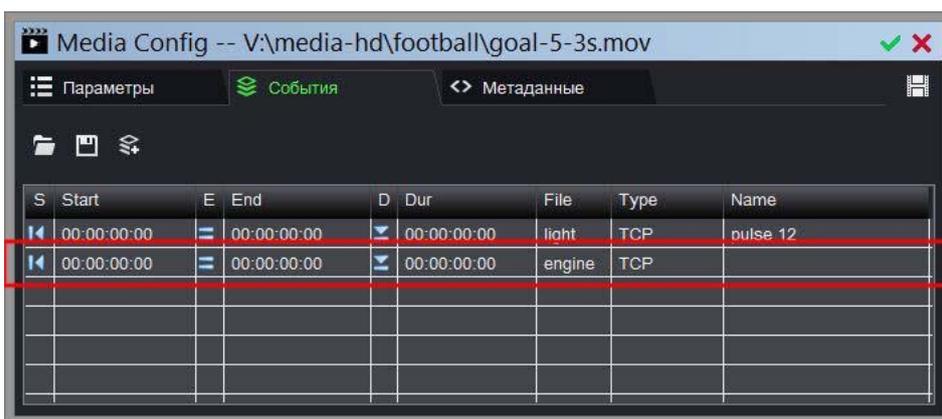
И убеждаемся, что там стоит 0 или номер порта, отличный от порта ведомого.

Настройка вторичных событий ведущего

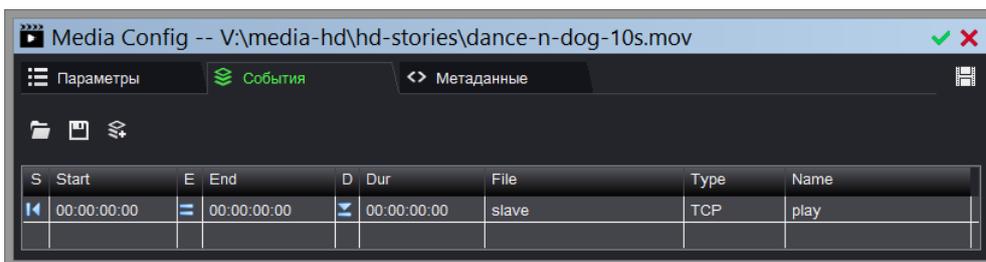
Для синхронного запуска сюжета на ведомом нужно в сюжете ведущего поставить вторичное событие. Для этого открываем редактор вторичных событий через контекстное меню клипа -- правая мышка -- **Вторичные события**, и в таблице вторичных событий через контекстное меню выполнить операцию **Добавить**.



В таблицу добавляется новое событие



В поле **File** нужно вписать имя сервера, указанное в параметре **name** тега **<engine>** (в нашем примере сервер **slave**), а в поле **Name** -- текст команды, которая будет отправлена серверу (в нашем примере команда **play**).



Редактировать поля **File** и **Name** можно двойным щелчком мыши. Временные параметры событий TCP/IP задаются так же, как для графических событий. Длительность событий TCP/IP игнорируется.

Запуск сюжета

При запуске сюжета с таким вторичным событием одновременно со стартом сюжета на ведущем будет отправлена команда **play** на ведомый. По этой команде на ведомом запустится **текущий** (выбранный) сюжет. Т.е. перед запуском нужно на ведомом **выбрать** тот сюжет, который будет запускаться синхронно с ведомым.

Управление ведомыми сюжетами

При такой схеме работы при остановке сюжета на ведущем, ведомый сюжет будет продолжать воспроизведение. Можно настроить ведущий так, чтобы команды стоп выполнялись на ведомом. Но нужно учитывать, что эти команды будут отправляться на ведомый всегда, когда на ведущем выполняется соответствующая команда. Для этого опять таки нужно отредактировать файл **vglive.cfx**.

Ищем группу тегов

```
<stop>  
  <cmd engine="onair">fo vglive.bg ~(/fade_out)</cmd>  
</stop>
```

и добавляем команду

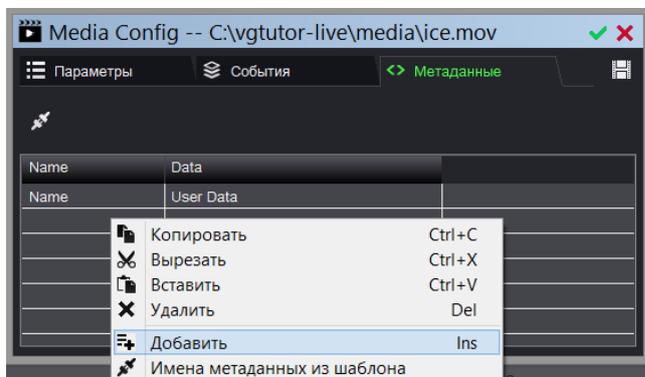
```
<stop>  
  <cmd engine="onair">fo vglive.bg ~(/fade_out)</cmd>  
  <cmd engine="slave ">stop</cmd>  
</stop>
```

Тегов, описывающих команду стоп может быть несколько – по количеству заданных слоев. Поэтому команду трансляции стопа ведомому нужно вставить в нужный слой. Или во все слои.

Метаданные шаблонов

Создание и редактирование метаданных

С каждым клипом в наборе можно связывать некоторые дополнительные данные – метаданные, и затем использовать их для оформления клипа. Для работы с метаданными нужно перейти на закладку Метаданные в параметрах клипа или выполнить команду Метаданные из всплывающего меню клипа.



Меню правой кнопки мыши в таблице метаданных содержит команду Добавить, при выполнении этой команды в таблицу добавляется новая строка с полями Name (имя метаданного) и User Data (собственно данные). Эти поля можно изменить двойным щелчком мыши.

Например, если идет работа с музыкальными клипами, то можно задать такие метаданные

Name	Data
Artist	Paul McCartney
Song	Hey Jude

Файл метаданных

Если с клипом связаны метаданные, то при старте клипа создается файл в формате XML, в который записываются эти метаданные. Для приведенного примера будет создан файл с таким содержимым

```
<?xml version="1.0" encoding="utf-8"?>  
<meta>  
  <Artist>Paul McCartney</Artist>  
  <Song>Hey Jude</Song>  
</meta>
```

Данные из этого файла можно использовать в графических шаблонах, которые заданы для клипа как вторичные события.

Расположение и имена файлов метаданных задается в конфигурационном файле **vglive.cfx** в строках с тегом **<play>**, по умолчанию для двух слоев в папке data создаются файлы:

```
<play meta="$(home)\data\vglive.meta.xml">  
  <play layer="overlay.bg" meta="$(home)\data\overlay.meta.xml">
```

Для каждого слоя задается свой файл метаданных.

Связывание метаданных с графическими шаблонами

При использовании графических шаблонов можно выполнять связывание метаданных с указанными слоями шаблона, при этом при старте шаблона подстановки метаданных в эти слои выполняются автоматически.

Редактор меток (триммер)

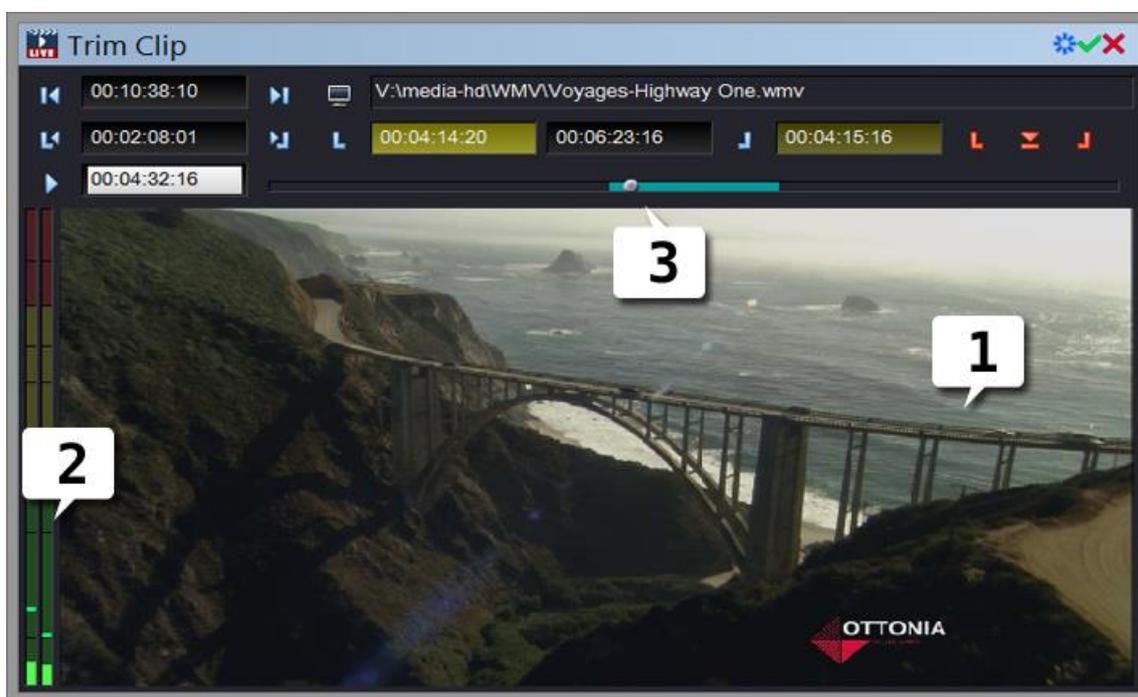
Чтобы воспроизводить не весь клип, а только некоторый его сегмент можно поставить входную и выходную метку клипа, и тогда он будет воспроизводиться от входной до выходной метки. Эти метки можно задать "вручную" в свойствах сюжета, но гораздо удобнее пользоваться **Редактором меток** - Триммер (**Trim Editor**), в котором легко искать те кадры, в которых нужно поставить метки.

Запуск триммера

Запуск триммера выполняется из контекстного меню сюжета (правая мышка в информационной строке), командой **Редактор меток (CTRL+T)**.

Интерфейс триммера

Основным элементом диалога триммера является окно просмотра клипа (1). Слева от него находится индикатор уровня звука (2), а выше окна просмотра расположен слайдер для позиционирования внутри клипа (3). Передвигая ползунок слайдера можно быстро просматривать содержимое клипа, при этом выполняется воспроизведение коротких фрагментов звуковой дорожки клипа.



Команды триммера

Команды навигации внутри клипа

Media Start -- перейти на начало клипа **⏪ (Home)** - выполняет позиционирование на первый кадр клипа.

Media End -- перейти на конец клипа **⏩ (End)** - выполняет позиционирование на последний кадр клипа.

To In -- перейти на входную метку **⏪ (SHIFT+I)** - перемещает текущую позицию на входную метку.

To Out -- перейти на выходную метку **⏩ (SHIFT+O)** - перемещает текущую позицию на выходную метку.

Play -- начать/окончить воспроизведение **▶ (Space)** - играет клип со звуком (или останавливает его).

Работа с метками

Mark In -- установить входную метку  (I) - текущая позиция в клипе выбирается как входная метка.

Mark Out -- установить выходную метку  (O) - текущая позиция в клипе выбирается как выходная метка.

Remove In -- удалить входную метку  (Alt+I) - входная метка устанавливается на первый кадр клипа.

Remove Out -- удалить выходную метку  (Alt+O) - выходная метка устанавливается на последний кадр клипа.

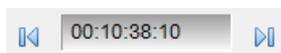
Remove InOut -- удалить обе метки  (Alt+X) - входная метка устанавливается на первый кадр клипа и выходная метка -- на последний кадр клипа.

Таймкоды

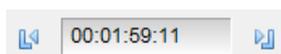
Информационные поля с таймкодами могут быть активными. В такие поля можно вводить значения вручную или менять их значение вращая колесо мышки. Активировать поле можно кликнув левой кнопкой мышки в этом поле, все поле начинает мигать, индицируя готовность поля для ввода данных.

Можно ввести нужный таймкод в активное поле набирая нужные цифры на клавиатуре, без разделителей. Так, если нужно ввести таймкод **00:00:12:20**, достаточно просто набрать последовательность цифр **1-2-2-0**.

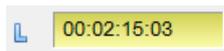
Чтобы изменить значение таймкода колесом мышки нужно вращать это колесо. Вращение от себя приводит к увеличению таймкода, вращение к себе -- к уменьшению значения таймкода. Каждый щелчок колеса меняет значение поля на один кадр. Если при вращении колеса мышки нажать кнопку **SHIFT**, то будут меняться секунды таймкода. Если нажать кнопку **CTRL**, то будут меняться разряды минут таймкода, если нажать кнопку **ALT**, то будут меняться разряды часов таймкода.



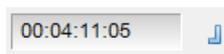
Между кнопками Media Start и Media End находится информационное поле длины клипа.



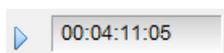
Между кнопками перехода на метки находится информационное поле с длиной текущего сегмента. Это поле является активным, изменяя значение в этом поле мы изменяем положение выходной метки.



Справа от кнопки установки входной метки находится таймкод этой метки, это активное поле. Меняя значение этого поля мы меняем положение входной метки и длину сегмента.



Слева от кнопки установки выходной метки находится таймкод этой метки. Это активное поле, изменяя его значение мы меняем положение выходной метки и длину сегмента.



Текущая позиция в клипе. Активное поле.

Такое разнообразие методов ввода позволяет быстро расставить метки наиболее удобным способом, визуально контролировать на каких кадрах находятся метки.

Титры

Сюжеты типа титры (шаблоны VgCast) довольно сильно отличаются от остальных объектов набора. Поскольку шаблоны VgCast могут включать в себя много слоев, причем каждый слой может содержать в себе видеофайлы, что позволяет формировать многослойные композиции типа "картинка в картинке". Пиктограммы титровальных сюжетов отмечены иконкой 

Правила оформления шаблонов

При использовании шаблонов в наборе нужно учитывать простое правило -- при воспроизведении шаблона из набора для визуализации шаблона используется эффект с именем **show**, при остановке шаблона -- эффект с

именем **hide**. Поскольку эти эффекты формируются автоматически при создании шаблона, в простейших случаях можно просто использовать эффекты по умолчанию. Желательно, чтобы в параметрах эффектов шаблона стояли галки **Глобальный шаблон** и **Загружать невидимым**.

Если в шаблоне заданы еще какие-либо эффекты, то их можно выполнить. Если эффектам присвоены шорткаты, то эти шорткаты доступны в программе. Эти шорткаты имеют более высокий приоритет, чем шорткаты самой программы.

Заполнение текстовых полей – метаданные

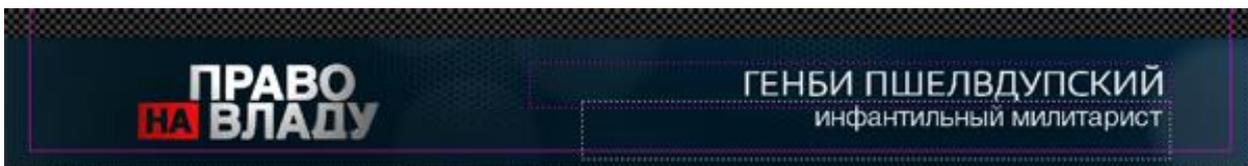
В качестве примера рассмотрим типичный титр нижней трети экрана **lower3rd.tpl**. В шаблоне описания такого титра есть три слоя с именами

name -- имя персонажа

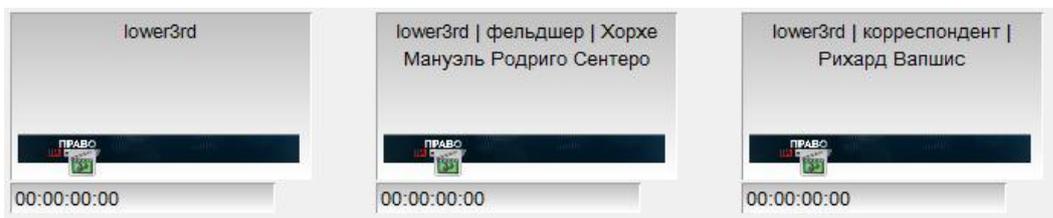
note -- дополнительная информация

bg -- анимированная подложка под титр

Если такой шаблон перетащить в набор и выполнить его, то в текстовых полях будет выведена та информация, которая была написана в шаблоне.

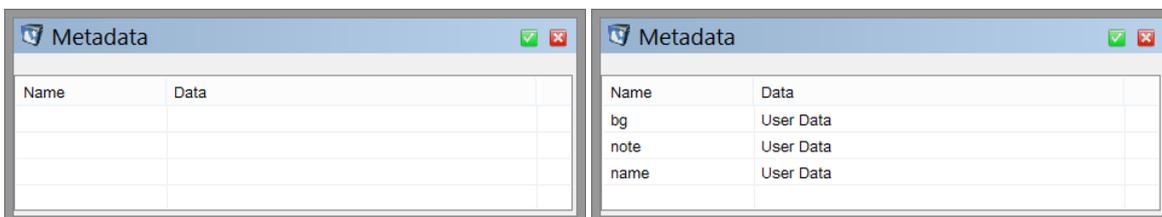


На пиктограмме будет написано только имя шаблона.



Фактически, если нужно использовать один и тот же шаблон для титровки нескольких особей, то пришлось бы создавать целую кучу шаблонов с прописанными именами и примечаниями, что конечно же крайне неудобно. Чтобы решить эту проблему в наборе есть механизм подстановки метаданных в шаблон.

Достаточно выполнить команду контекстного меню **Редактировать метаданные CTRL+M** и около пиктограммы шаблона появится окно ввода метаданных.



Изначально таблица метаданных пуста. Чтобы ее заполнить удобнее всего использовать контекстное меню метаданных (правая мышка в окне метаданных) и выполнить команду **Имена метаданных из шаблона**. После этого таблица заполнится именами слоев из шаблона. Для удобства использования можно удалить лишние строки в этой таблице -- например, строку с именем слоя **bg**, поскольку содержимое этого слоя меняться не будет. Можно сразу прописать в таблице нужные значения полей, например в поле **note** вписать слово **фельдшер**, в поле **name** - **Хорхе Мануэль Родриго Сентеро**.

Name	Data
note	фельдшер
name	Хорхе Мануэль Родриго Сентеро

Для редактирования полей в таблице нужно выполнить двойной щелчок левой мышки на ячейке таблицы. После закрытия таблицы шаблону будут назначены указанные метаданные и эти метаданные появятся на пиктограмме сюжета.

Теперь для создания титров достаточно скопировать шаблон с метаданными (**CTRL+C** затем **CTRL+V**, или выполнить перетаскивание с нажатой клавишей **SHIFT** на клавиатуре) и выполнить команду редактирования метаданных **CTRL+M**.

При выполнении шаблонов с метаданными из набора происходит следующие действия: шаблон загружается в VgCast, выполняется подстановка в поля метаданных (поскольку шаблон еще невидимый, то это действие незаметно) и только затем выполняется эффект **show**. В результате при выводе второго титра из набора получим в эфире плашку



а при выводе третьего титра



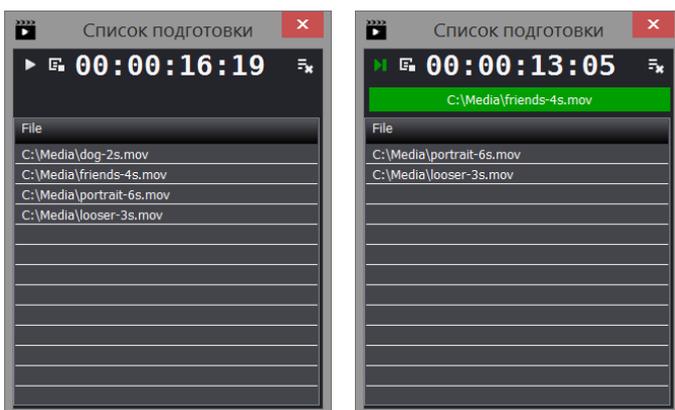
Вообще говоря, в качестве метаданных можно использовать не только текстовые слои, но и слои других типов. Добавим в шаблон lower3rd еще один слой, для фотографии, с именем **pict**. Тогда при редактировании метаданных в таблице можно оставить строку с именем **pict**:

Name	Data
note	режиссер
name	Сулим Ханмуратов
pict	C:\vgtutor-todo\dogs\Aidi.png

В поле Data строки pict можно ввести полное имя файла с нужной картинкой, либо, что существенно проще, перетянуть мышкой из проводника нужный файл и бросить его в поле Data строки pict.

Плейлисты в VGLIVE

В меню «Вид» можно выбрать пункт «Показать список подготовки» чтобы показать или убрать диалоговое окно работы с плейлистами (списками подготовки клипов).



Работа с плейлистами позволяет без подрывов воспроизводить заданную последовательность клипов за счет того, что клипы в плейлисте готовятся к воспроизведению заранее (выполняется буферизация видеок кадров из файлов).

Элементы плейлиста

В плейлист можно добавлять видеофайлы из основного интерфейса программы либо из проводника.

Формирование списка

Перетаскивание (**drag-drop**) из основного интерфейса и проводника в список CUE.

Копирование и вставка (**copy-paste**) из основного интерфейса и проводника в список CUE

Контекстное меню клипа содержит команду «**Добавить в список подготовки**», клип добавляется в **конец** плейлиста

SHIFT+левый клик на пиктограмме клипа в основном интерфейсе добавляет объект в **конец** плейлиста.

Если плейлист начинает формироваться когда исполняется файл из основного интерфейса, то плейлист активируется, и по завершению основного клипа автоматически начинается воспроизведение плейлиста.

Изменение порядка клипов

Перетаскивание и копирование-вставка внутри списка позволяет изменить порядок объектов в списке. Правая мышка в списке клипов вызывает контекстное меню с операциями редактирования.

Управление плейлистом

На инструментальной панели окна плейлиста расположены кнопки управления, имя файла в эфире и таймер плейлиста.

▶ Следующий клип

Если плейлист не выполняется, то кнопка «Следующий клип» инициирует вывод в эфир клипов из плейлиста. Пиктограмма кнопки меняется на  и ее цвет становится зеленым.

▶▶ Следующий клип

Если плейлист уже выполняется, то выполняется переход на следующий клип в списке.

⏏ Остановить плейлист

Воспроизведение плейлиста останавливается. Список файлов в плейлисте остается в текущем состоянии.

🗑 Очистить плейлист

Воспроизведение плейлиста останавливается. Список файлов в плейлисте очищается.

Таймер плейлиста

Таймер плейлиста показывает суммарное время всех клипов в плейлисте. По мере воспроизведения клипов таймер уменьшается. При этом таймер на основном интерфейсе (и внешние таймеры) показывают, сколько времени осталось до конца текущего клипа, а таймер плейлиста показывает сколько времени осталось до конца всего плейлиста.

Взаимодействие плейлиста и основных клипов

Если плейлист выполняется и пользователь выводит в эфир клип из основного интерфейса, то выполнение плейлиста прерывается, и начинает выполняться основной клип.

Если выполняется клип из основного интерфейса и пользователь начинает формировать плейлист, то плейлист активируется и начинает выполняться по завершению основного клипа.

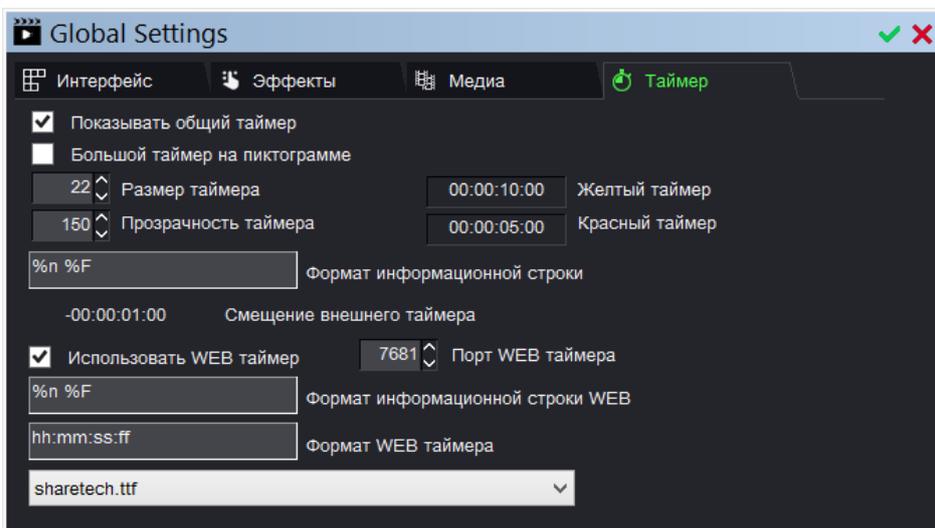
Если выполняется клип из основного интерфейса и плейлист активирован, то команда «Стоп» для основного клипа приводит к началу воспроизведения плейлиста.

Во время выполнения плейлиста можно продолжать формировать плейлист или переупорядочивать файлы в плейлисте.

Администрирование VGLIVE

Внешний WEB таймер

При выдаче сюжетов в VGLIVE в интерфейсе оператора запускается таймер обратного отсчета, который показывает, сколько времени осталось до конца сюжета. Как правило, эта информация нужна и режиссерской группе. Для этого в VGLIVE можно разрешить использование WEB-таймера, и тогда таймер обратного отсчета сюжета можно увидеть в любом браузере на любом устройстве в сети. Для этого в настройках VGLIVE нужно разрешить использование WEB-таймера:



Чтобы увидеть таймер в браузере, нужно задать URL вида <http://ip-address:7681> или <http://computer-name:7681>, например <http://192.168.1.103:7681>



В настройках WEB-таймера можно задать формат информационной строки, формат самого таймера, и шрифт, которым он будет выводиться в браузере.

Вообще говоря, насколько точно будет показывать таймер значение до конца сюжета, зависит от параметров локальной сети и ее загруженности. Поэтому можно задать смещение внешнего таймера как в положительную, так и в отрицательную сторону.

Командная строка для запуска Firefox в отдельном окне на весь экран

```
cmd>firefox.exe --kiosk -new-window http://127.0.0.1:7681  
cmd>firefox.exe -P timer -width 400 -height 200 -new-window  
http://127.0.0.1:7681
```

Как запускать Firefox в окне без заголовков

<https://superuser.com/questions/977912/firefox-hide-everything-except-content-area-of-the-browser>

Вывод сюжетов с подготовкой

При выводе сюжетов иногда возникают проблемы связанные с производительностью станции, на которой работает движок VGCAST. Если производительность процессора и/или дисковой подсистемы недостаточна для быстрого декодирования медиафайлов, то может возникать брак по звуку и/или видео. Причем часто такой эффект

возникает только при самом первом воспроизведении файла, а потом он не повторяется, поскольку выполнилось кэширование этого файла системой. Для устранения таких артефактов можно в VGLIVE настроить работу в режиме предварительной подготовки (CUE) медиафайлов. В таком режиме когда файл (пиктограмма в интерфейсе) выбирается мышкой (или командами выбора сюжета), в движок VGCAST отправляется команда подготовки файла, при этом выполняется чтение и декодирование нескольких секунд видео и аудио данных, и когда выполняется команда плей, файл уже готов для беспрепятственного воспроизведения.

В стандартной конфигурации для каждого состояния медиаданных (сюжетов) прописаны те команды, которые выполняются при переходе сюжета в соответствующее состояние - `<play>` `<stop>` `<pause>` и `<resume>`. Для работы в режиме предварительной подготовки нужно изменить команды тегов `<select>` и `<play>`.

```
<play meta="$(home)\data\vglive.meta.xml">
  <cmd engine="onair">subst vglive.bg &quot;~(//file)&quot; loop=~(//loop) field=~(//field) scale=~(//scale)
trim=~(//trim_in),~(//trim_out) map=~(//audio_map) track=~(//main_track) par=~(//par) volume=~(//vol)</cmd>
  <cmd engine="onair">fi vglive.bg ~(//fade_in)</cmd>
</play>
```

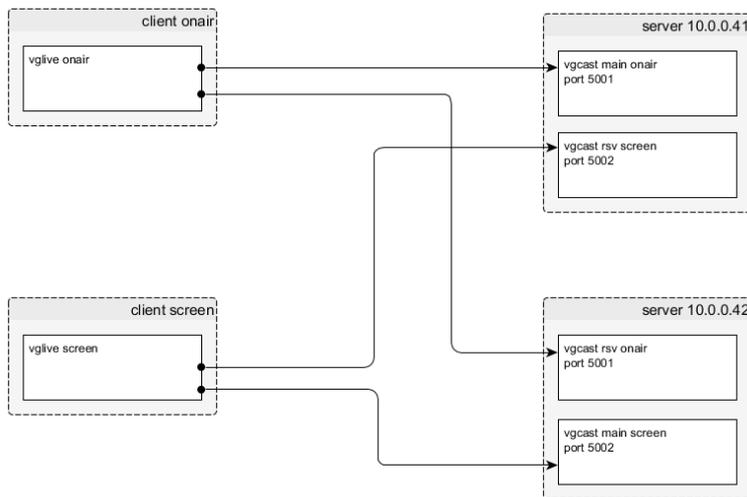
Меняем команды (а если тегов `<select>` нет в конфигурации – добавляем их):

```
<select>
  <cmd engine="onair">cue vglive.bg purge</cmd>
  <cmd engine="onair">cue vglive.bg ins &quot;~(//file)&quot; loop=~(//loop) field=~(//field) scale=~(//scale)
trim=~(//trim_in),~(//trim_out) map=~(//audio_map) track=~(//main_track) par=~(//par) volume=~(//vol)
adelay=~(//audio_delay)</cmd>
</select>
<play meta="$(home)\data\vglive.meta.xml">
  <cmd engine="onair">cue vglive.bg next</cmd>
  <cmd engine="onair">fi vglive.bg ~(//fade_in)</cmd>
</play>
```

Обратите внимание, что группы команд `<play>` и т.д. прописаны для каждого слоя (по умолчанию – два слоя). Поэтому, чтобы и на втором слое выполнялось воспроизведение с предварительной подготовкой, нужно изменения в команды `<select>` и `<play>` внести дважды.

Настройка VGLIVE в режиме резервирования

В тех случаях, когда нужно обеспечить бесперебойную выдачу сюжетов и титров можно настроить работу VGLIVE и движков VGCAST в режиме резервирования. Пусть, например, есть две станции, на которых установлены движки VGCAST и есть платы для вывода нескольких сигналов SDI. Один из выходов на каждой станции используется для формирования пары сигналов (FILL+KEY) для графического оформления эфира, а второй – для выдачи информации на экран в студии. Соответственно, есть два рабочих места оператора – одно для выдачи эфирных титров и сюжетов, второе – для выдачи сюжетов на экран в студию.



Выходы с серверов нужно подать на матрицу и обеспечить возможность оперативного переключения основного и резервного сигнала.

Клиентские и серверные компьютеры должны находиться в одной сети. Медиаданные и графические шаблоны можно размещать либо на каком-либо разделяемом носителе, либо дублировать на каждой станции в сети в одной и той-же папке и на одном и том-же диске, чтобы полный путь к файлам был одинаковым на каждой станции.

Конфигурация сервера

На серверных станциях необходимо оптимально настроить конфигурацию движков VGCAST. Предполагаем, что сервера находятся далеко от рабочих мест операторов, и не имеют своих мониторов. В таком случае можно отключить режим просмотра выходов платы на системном мониторе. Для этого нужно в конфигурационном файле устройства вывода (например, **devdecklink.dll.cfx**) изменить параметр тега <Preview>

```
<Preview use="dx" ...>... ..</Preview>
```

На

```
<Preview use="0" ...>... ..</Preview>
```

Вообще говоря, превью можно и не отключать, но такой режим позволяет существенно уменьшить нагрузку на процессор сервера.

Предполагается, что **не** используется метод формирования превью с помощью компонент **devpvw.dll**, а режим формирования превью задается непосредственно в конфигурационных файлах устройств вывода – **devdecklink**, **devnull**, **devndi** и т.д.

Конфигурация клиента

Чтобы обеспечить одновременную выдачу команд клиентской станции на основной и резервный движки VGCAST, нужно отредактировать файл конфигурации клиентской программы выдачи сюжетов **vglive.cfx**. Обычно в начале конфигурационного файла идет описание используемых движков. В стандартной конфигурации описан только один движок:

```
<engine run="" host="127.0.0.1:5001" name="onair">
  <init>load $(home)/vglive.tpl</init>
  <init>load $(home)/overlay.tpl</init>
</engine>
```

В тегах <init> указаны те команды движка VGCAST, которые выполняются при установлении соединения TCP/IP. Вообще говоря таких тегов может быть указано сколько угодно и в них можно задавать любые команды VGCAST. Для VGLIVE выполняется загрузка в движок шаблонов нулевого (**vglive.tpl**) и первого (**overlay.tpl**) слоя. В имени файла шаблона используется псевдопеременная **\$(home)** – вместо нее подставится путь к папке, из которой запущена программа VGLIVE. Для стандартной конфигурации это удобно, но для систем клиент-сервер лучше указать точный путь к файлу с шаблоном, поскольку программа VGLIVE выполняется на клиентской станции, а движок VGCAST будет загружать шаблон на серверной станции.

В нашем случае для каждого клиента VGLIVE используется 2 движка, один основной (main) и второй – резервный (rsv). Пусть шаблоны программы лежат в папке **C:\OnAir**. Описание движков для клиента выдачи в эфир должно иметь такой вид:

```

<engine run="" host="10.0.0.41:5002" name="main">
  <init>load "c:\onair\vglive.tpl"</init>
  <init>load "c:\onair\overlay.tpl"</init>
</engine>
<engine run="" host="10.0.0.42:5002" name="rsv">
  <init>load "c:/onair/vglive.tpl"</init>
  <init>load "c:/onair/vglive.tpl"</init>
</engine>

```

.Желательно имя файла заключать в двойные кавычки. Если в имени файла есть пробелы его обязательно нужно заключать в кавычки. В имени файла можно использовать как обратную косую черту, так и прямую.

Для каждого слоя описаны те команды движка, которые выполняются при изменении состояния каждого элемента VGLIVE – сюжета или шаблона. В стандартной конфигурации эти описания имеют такой вид:

```

<select>
  <cmd engine="onair"> ... </cmd>
</select>
<play meta="$(home)\data\vglive.meta.xml">
  <cmd engine="onair"> ... </cmd>
  <cmd engine="onair"> ... </cmd>
</play>
<stop>
  <cmd engine="onair"> ... </cmd>
</stop>
<pause>
  <cmd engine="onair"> ... </cmd>
</pause>
<resume>
  <cmd engine="onair"> ... </cmd>
</resume>

```

Поскольку в нашем случае используется одновременно два движка – для основного канала **main** и для резервного **rsv**, то нужно в тегах **<cmd>** перечислить все используемые движки через запятую:

```

<play meta="//share/onair\data\vglive.meta.xml">
  <cmd engine="main,rsv"> ... </cmd>
  <cmd engine="main,rsv"> ... </cmd>
</play>

```

Также нужно чтобы путь к файлу с метаданными слоя (атрибут **meta** в теге **<play>**) находился в разделяемой папке.

Организация превью на клиентской станции

Если серверное (VGCAST) и клиентское приложение (VGLIVE) выполняются на разных компьютерах, то увидеть в интерфейсе оператора превью выходных каналов невозможно, поскольку платы вывода физически расположены на другом компьютере. Однако можно организовать IP-поток для формирования картинки превью у клиента.

Конфигурация сервера

На серверной станции во всех драйверах устройств, которые нужно увидеть на клиентской стороне нужно разрешить передачу превью. В стандартной конфигурации драйверов устройств VGCAST содержатся такие настройки

```

<Stream use="0" name="proxy" port="10001">
  <ZoomOut>4</ZoomOut>
  <Format>fmt=mpegts,codec=mpeg2video,pix=yuv420p,qmin=5,qmax=10,g=1</Format>

```

</Stream>

Для того, чтобы включить формирование IP-потока с содержимым выходного буфера драйвера, нужно параметр установить **use="1"**. В потоке превью обычно нет смысла передавать видеокadres в полном разрешении, это довольно дорогостоящая операция, поскольку используется кодирование в трег-2 или h264. Обычно передается видео в более низком разрешении, а параметр **ZoomOut** указывает во сколько раз уменьшается размер кадра. Так при работе в формате HD 1080 и параметре **ZoomOut = 4**, будут передаваться кадры размером 480x270.

Конфигурация клиента

На стороне клиента необходимо создать и настроить несколько драйверов для приема IP-потоков. В стандартную инсталляцию VGCASST входит один драйвер для приема потока, **recstream.dll** и его конфигурационный файл **recstream.dll.cfx**. Необходимо создать копии этих файлов, например **recstream-1.dll**, **recstream-1.dll.cfx** и **recstream-2.dll**, **recstream-2.dll.cfx**. Далее, нужно отредактировать конфигурационные файлы **recstream-*.dll.cfx** любым текстовым редактором, а еще лучше редактором XML, чтобы был виден синтаксис и разметка текста.

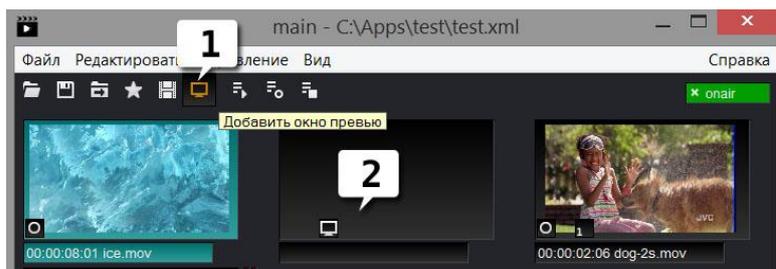
```
<Source connect="1">udp://10.0.0.41:10001</Source>
<Width>480</Width>
<Height>270</Height>
<BufWidth>480</BufWidth>
<BufHeight>270</BufHeight>
```

В примере приведены только те теги, на которые важны для организации превью.

Нужно установить параметр **connect** тега **<Source>** и 1, и указать IP-адрес и порт того сервера, превью которого нужно увидеть.

Параметры **Width** и **Height** нужно установить в соответствии с разрешением сигнала и параметром **ZoomOut**, т.е. если используется формат 1080, и **ZoomOut="4"**, то размеры кадра превью равны 480x270.

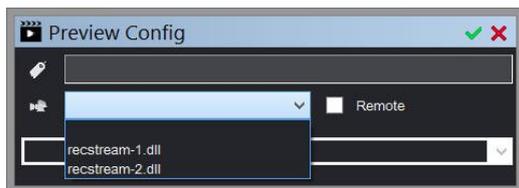
Чтобы добавить окна превью в интерфейс VGLIVE нужно выполнить команду «Добавить окно превью» через основное меню программы или кнопку (1) на инструментальной панели.



В интерфейсе VGLIVE появится пиктограмма окна превью (2), пока не привязанная к конкретному потоку.

Чтобы настроить окно превью нужно из контекстного меню объекта (правая мышка на информационной строке) выполнить команду «Параметры»

Появляется диалоговое окно для конфигурации превью, в котором нужно выбрать устройство:



Нужно поставить галку в поле "Remote" и убедиться, что в строке адреса прописан правильный IP и порт. Если все параметры заданы правильно, в пиктограмме интерфейса появится живое видео.

Настройка VGED в режиме резервирования

Если для выдачи титров в эфир используется редактор титров VGED и нужно обеспечить резервирование выдачи титров, т.е. выводить их одновременно на два канала, то нужно модифицировать конфигурационный файл редактора **vged.cfx** на клиентской машине с помощью любого текстового редактора. В этом файле нужно найти строки

```
<Port>5001</Port>  
<Host>127.0.0.1</Host>
```

и заменить их на

```
<Port>5001</Port>  
<Host>10.0.0.41</Host>  
<Aux sync="1">10.0.0.42:5001</Aux>
```

Если нужно выводить более чем на два сервера, то достаточно повторить нужное количество раз строку с тегом Aux и указать в ней правильный ip-адрес и порт.