Документация, содержащая описание функциональных характеристик программного обеспечения и информацию, необходимую для установки и эксплуатации программного обеспечения

**«Мультимедийное ядро Sentinel»**

**ТЕРМИНЫ И ОБОЗНАЧЕНИЯ**

|  |  |
| --- | --- |
| **Термин, сокращение** | **Определение** |
| **API ПО** | Программный интерфейс взаимодействия ПО с внешними системами для приема-передачи данных. |
| **Мультимедийный Источник** | Совокупность сведений об источнике трансляции, параметрах трансляции и данных о записи. Имеет уникальный идентификатор. |
| **ПО** | Программное обеспечение |
| **RTSP** | Прикладной протокол, предназначенный для использования в системах, работающих с мультимедийными данными (мультимедийным содержимым, медиасодержимым), и позволяющий удалённо управлять потоком данных с сервера, описан в RFC 2326. |
| **HTTP** | Протокол прикладного уровня передачи данных, используется для передачи произвольных данных. |

# **1 ОБЩИЕ СВЕДЕНИЯ**

## **1.1 Цели разработки ПО**

ПО является универсальной системой работы с цифровыми потоками аудио и видео информации. ПО позволяет получать мультимедийные потоки с различных средств трансляции (видеокамеры, сервисы потокового вещания и т.д.), обрабатывать, сохранять и ретранслировать во внешние системы.

## **1.2 Идея разработки ПО**

В основе ПО лежит идея о том, что потоки мультимедийных данных различного формата от различных источников возможно хранить и обрабатывать по единым принципам. Например, IP видеокамера, USB видеокамера, платформа потоковой трансляции позволяют получить данные по сильно отличающимся друг от друга протоколам и принципам. Но, аудио и видео кодек после получения и распаковки данных имеет одинаковый вид. Это позволяет записать консистентный архив и организовать передачу цельных данных во внешние системы.

ПО включает в себя следующие логические узлы:

• Рекордер

• Сервер HTTP

• Сервер приема потоковых данных

• Сервер RTSP

Управление ПО осуществляется за счет использования API библиотеки, входящей в состав Сервера HTTP. Посредством данной библиотеки создаются мультимедийные источники на стороне Рекордера, который в свою очередь отвечает за ожидание и единообразную интерпретацию поступающих потоков данных. Верифицированные потоки данных поступают в Рекордер посредством группы Серверов приема потоковых данных, которые либо инициируют самостоятельное подключение к видеокамерам, либо являются серверами ожидания общепринятых протоколов получения данных. Для единообразной ретрансляции всех данных во внешние системы используется Сервер RTSP.

## **1.3 Функции и сервисы ПО**

К функциям и сервисам ПО относятся:

- создание источников трансляций;

- запись данных в архив;

- передача архивных и потоковых данных во внешние системы

- поддержка мультиплатформенного API для подключения внешних приложений;

- возможность массового обслуживания запросов на обработку данных.

# **2 ОПИСАНИЕ ФУНКЦИОНАЛА ПО**

## **2.1 Описание подсистем ПО**

## **2.1.1 Описание подсистемы Рекордер**

Подсистема Рекордер обладает следующим функционалом:

* является аккумулирующим звеном, обеспечивающим взаимодействие внутренних подсистем ПО с внешними источниками информации;
* получает демультиплексированные данные от серверов приема потоковых данных;
* отвечает за вычисление и корректировку временных значений в потоках;
* отвечает за синхронизацию каналов (аудио и видео);
* отвечает за генерацию меток времени при их отсутствии;
* перенаправляет потоки данных во внутренние потребители (запись в архив, ретрансляторы);
* контролирует глубину записанного архива.

## **2.1.2 Описание подсистемы Сервер HTTP**

Подсистема Сервер HTTP обладает следующим функционалом:

* возможность программного создания источников для записи;
* возможность программного получения статусов источников;
* возможность получения сведений об архиве;
* возможность редактирования и управления архивом;
* возможность конфигурации ПО без перезагрузки;

Подсистема является точкой входа и выхода в процессе работы пользователя с ПО.

## **2.1.3 Описание подсистемы Сервер приема потоковых данных**

Подсистема Сервер приема потоковых данных обладает следующим функционалом:

* возможность включения сервисов приема потоковых данных (srt, rtmp и т.д.);
* возможность конфигурации портов;
* возможность конфигурации максимального количества параллельных потоков;
* Подключение и верификация потоков в Рекордере;
* Демультиплексирование и передача данных в Рекордер.

## **2.1.4 Описание подсистемы Сервер RTSP**

Подсистема Сервер RTSP обладает следующим функционалом:

* возможность подключения внешних систем для получения живых трансляций данных;
* возможность подключения внешних систем для получения архивных трансляций данных;
* конфигурация архивных трансляций (скорость, направление, цикличность и т.д.);
* возможность конфигурации портов;
* возможность конфигурации максимального количества параллельных потоков.

1. **РАБОТА С ПО**

**3.1 Создание источника (заведение потоков)**

POST http://host:port/media/create

Content-Type: application/json

{

"source\_id":"Camera\_test\_01",

"src":{

"format": "rtsp",

"url":"rtsp://admin:admin123@192.168.0.250:554/cam/realmonitor",

"transport": "tcp",

"connection\_timeout": 5000

},

"dst": [

{

"enable": true,

"format": "archive",

"container": "mp4",

"duration": 10

},

...

]

}

Ответы: 200 + тело в формате исходного JSON / 304 если нет изменений / 400 неверные параметры запроса / 500 что-то пошло не так на сервере.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Описание** | **Тип данных** | **Обяз.** | **Значение по умолчанию** |
| "source\_id" | Уникальный идентификатор видеопотока. При повторном заведении перезаписывется | String | Да | - |
| "src.format" | Тип источника данных. | String | Да | - |
| "src.url" | RTSP ссылка на поток с учетом авторизационных данных | String | Да | - |
| "src.transport" | RTSP транспорт для получения потока (“udp” | “tcp”) Не рекомендуется использовать udp – данный транспорт для rtsp источника не гарантирует ровный стабильный поток. | String | Нет | tcp |
| "src.connection\_timeout" | Таймаут в миллисекундах, после которого устройство считается оффлайн | Integer | Нет | 5000 (5 сек) |
| "dst[].format" | Формат мультиплексирования потока. | String | Да | - |

* 1. **Полное удаление источника с удалением архива**

DELETE http://host:port/media

Content-Type: application/json

{

"source\_id":"Camera\_test\_01"

}

Ответы: 200 + тело в формате исходного JSON / 400 неверные параметры запроса / 404 если источник не найден / 500 что-то пошло не так на сервере.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Параметр** | **Описание** | **Тип данных** | **Обяз.** | **Значение по умолчанию** |
| "source\_id" | Уникальный идентификатор видеопотока. | String | Да | - |

**3.3 Получение статуса потоков**

GET <http://host:port/config/show/all> [статус всех потоков]

GET <http://host:port/config/show/all?source_id=someid1&source_id=someid2> [статус потоков someid1, someid2]

GET <http://host:port/config/show/all?source_id=someid1,someid2> [статус потоков someid1, someid2]

Параметры ответа:

[

{

"source\_id": "test\_rtsp",

"status": "online",

"last\_seen": "2022-03-21T13:05:31"

},

{

"source\_id": "test\_srt",

"status": "{offline, initialization}",

"last\_seen": "2022-01-08T10:45:48"

}

]

|  |  |  |
| --- | --- | --- |
| **Параметр** | **Описание** | **Тип данных** |
| "source\_id" | Уникальный идентификатор видеопотока. | String |
| "status" | Состояние:  - “online”  - “{offline, REASON}”, где REASON – текстовое описание причины оффлайна  - “disabled” | String |
| "last\_seen" | Время последнего онлайна, в RFC3339 (если без Z – локальная таймзона), либо undefined | String |