
MicroPVR Documentation

Выпуск Last

Konstantin Shpinev

апр. 20, 2020

1	А. Описание JSON-RPC API	3
1.1	А.1 Терминология	3
1.2	А.2 Описание	3

Актуальная документация сайте Microimpuls: <https://microimpuls.com/docs/micropvr/about>

Содержание:

А. Описание JSON-RPC API

1.1 A.1 Терминология

- API - набор интерфейсов функций системы для использования во внешних приложениях.
- Channel ID — уникальный идентификатор канала.
- Stream source — URL-адрес видеопотока.
- PID — уникальный идентификатор процесса.
- Task ID — уникальный идентификатор записи, уникален для каждого файла.
- Timestamp — метка UNIX-времени.

1.2 A.2 Описание

Интерфейс MicroPVR API используется для интеграции функции «Отложенный просмотр» (Архив) сервиса IPTV с системой Middleware. Вызовы функций API выполняются через протокол JSON-RPC 2.0 поверх протокола TCP (без поддержки передачи неименованных аргументов в виде массива). По-умолчанию, MicroPVR API слушает на порту 4089.

1.2.1 add_task

Метод является устаревшим с версии 1.5.0, рекомендуется использовать `create_new_task`.
Добавляет задачу на ежесуточную запись канала.

Пример запроса:

```
{  
  "jsonrpc": "2.0",  
  "method": "add_task",
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "id": "some-id",
    "params": {
      "channel_id": 1,
      "duration": 86400,
      "lifetime": 259200,
      "gap_time": 2,
      "record_location": "/opt/storage/",
      "priority": 1,
      "repeat_period": 86400,
      "stream_source": "udp://@239.1.1.1:1234"
    }
  }
}

```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
channel_name	string	Имя канала	Нет	
duration	int	Длительность непрерывного файла архива, в секундах	Да	Не менее 120
lifetime	int	Время жизни файла после окончания записи, в секундах	Да	Не меньше 0
gap_time	int	Час, в который необходимо делать разрыв записи и начинать новый файл (по UTC)	Да	От 0 до 23
record_location	string	Полный путь до директории, в которой необходимо хранить записи	Да	
priority	int	Приоритет выдачи из данной директории (при одновременной записи на разные носители); меньше значение - выше приоритет	Да	Не меньше 0
repeat_period	int	Через какой промежуток времени возобновлять запись канала, от начала старта предыдущей	Да	<i>Должен быть равен длительности записи</i>
stream_source	string	Multicast-адрес видеопотока	Да	

Метод возвращает результат операции. Возможные варианты ответа:

- ok - задача добавлена успешно
- failed - ошибка при добавлении задачи, см. детальное описание в лог-файле

Первая запись будет начата немедленно, но все последующие будут начинаться согласно gap_time. Реальная длительность записи может отличаться от duration:

- все записи начинаются на 80 секунд раньше, чем закончится предыдущая, для избежания разрыва вещания.
- в случае ошибки записи задача будет перезапущена с сохранением времени окончания записи, причём в случае, если длительность записи меньше 80 секунд, то вместо этого сразу будет запущена следующая задача.

Задача не будет добавлена, если уже существует другая задача с теми же channel_id и record_location.

1.2.2 create_new_task

Добавляет произвольную задачу на запись.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "create_new_task",
  "id": "some-id",
  "params": {
    "channel_id": 1,
    "start_timestamp": 1420115400,
    "duration": 2700,
    "lifetime": 259200,
    "record_location": "/opt/programs/",
    "priority": 1,
    "stream_source": "udp://@239.1.1.1:1234"
  }
}
```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
channel_name	string	Имя канала	Нет	
duration	int	Длительность непрерывного файла архива, в секундах	Да	Не менее 120
lifetime	int	Время жизни файла после окончания записи, в секундах	Да	Не меньше 0
start_timestamp	int	Время начала записи, если не указано, по умолчанию время вызова метода плюс пять секунд	Нет	
record_location	string	Полный путь до директории, в которой необходимо хранить записи	Да	
priority	int	Приоритет выдачи из данной директории (при одновременной записи на разные носители); меньше значение - выше приоритет; если параметр не указан - устанавливается приоритет 50	Нет	Не меньше 0
repeat_period	int	Через какой промежуток времени возобновлять запись канала, от начала старта предыдущей, если не указан, то задача будет неперiodической	Нет	Должен быть больше или равен длительности записи
stream_source	string	Multicast-адрес видеопотока	Да	
urgent	bool	Начинает периодическую запись немедленно, для неперiodических игнорируется	Нет	

Задача будет запущена согласно start_timestamp. Если время запуска находится в прошлом или urgent равно true, то время окончания первой записи рассчитывается так, чтобы оно было минимально возможным из удовлетворяющих $\text{start_timestamp} + \text{repeat_period} * k + \text{duration}$, где k - целое число. Если расчётное время запуска этой первой задачи также находится в прошлом, то она будет запущена немедленно.

Реальная длительность записи может отличаться от duration: - все записи начинаются на 80 секунд раньше, чем закончится предыдущая, для избежания разрыва вещания (только если repeat_period

равен duration и запись не была перезапущена из-за ошибки). - в случае ошибки записи задача будет перезапущена с сохранением времени окончания записи, причём в случае, если длительность записи меньше 80 секунд, то вместо этого сразу будет запущена следующая задача.

Задача не будет добавлена, если уже существует другая задача с теми же channel_id и record_location.

Коды ошибок:

Код	Описание
100	Отсутствует объект params
101	Один из обязательных аргументов отсутствует или не соответствует допустимым значениям
140	Период повтора меньше, чем длительность задачи
141	Время завершения записи находится в прошлом (для непериодическ задач)

1.2.3 add_urgent_task

Метод является устаревшим с версии 1.5.0, рекомендуется использовать create_new_task

Добавляет задачу на периодическую запись канала

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "add_urgent_task",
  "id": "some-id",
  "params": {
    "channel_id": 1,
    "start_timestamp": 1420115400,
    "duration": 2700,
    "lifetime": 259200,
    "record_location": "/opt/programs/",
    "priority": 1,
    "stream_source": "udp://@239.1.1.1:1234"
  }
}
```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
channel_name	string	Имя канала	Нет	
duration	int	Длительность непрерывного файла архива, в секундах	Да	Не менее 120
lifetime	int	Время жизни файла после окончания записи, в секундах	Да	Не меньше 0
record_location	string	Полный путь до директории, в которой необходимо хранить записи	Да	
priority	int	Приоритет выдачи из данной директории (при одновременной записи на разные носители); меньше значение - выше приоритет	Да	Не меньше 0
repeat_period	int	Через какой промежуток времени возобновлять запись канала, от начала старта предыдущей	Нет	<i>Должен быть равен длительности записи</i>
stream_source	string	Multicast-адрес видеопотока	Да	
start_timestamp	int	Время начала передачи, когда нужно начать запись, в UNIX Timestamp	Да	Время не должно быть в прошлом на момент вызова метода

Метод возвращает результат операции. Возможные варианты ответа:

- ok - задача добавлена успешно
- failed - ошибка при добавлении задачи, см. детальное описание в лог-файле

Реальная длительность записи может отличаться от duration:

- все записи начинаются на 80 секунд раньше, чем закончится предыдущая, для избежания разрыва вещания.
- в случае ошибки записи задача будет перезапущена с сохранением времени окончания записи, причём в случае, если длительность записи меньше 80 секунд, то вместо этого сразу будет запущена следующая задача.

Задача не будет добавлена, если уже существует другая задача с теми же channel_id и record_location.

1.2.4 get_file_offset

Метод является устаревшим с версии 1.5.0, рекомендуется использовать get_file_and_offset. Метод не соответствует стандарту JSON-RPC: в возвращаемом объекте присутствуют другие поля кроме result.

Осуществляет поиск файла записи и позиции в файле по временной метке Timestamp и идентификатору канала.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_file_offset",
  "id": "some-id",
  "params": {
    "method": "get_file_offset",
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "channel_id": 1,
    "timestamp": 1420115400
  }
}

```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
timestamp	int	Временная метка в UNIX Timestamp	Да	
aid	str	Идентификатор аккаунта	Нет	

Поля возвращаемого объекта:

Название	Тип	Описание
result	str	Результат операции
offset	str	Байтовое смещение в файле по заданной временной метке, конвентированно в строку; в случае ошибки запроса равно -1
file	str	Полный путь до файла записи
timestamp	int	Временная метка найдённого смещения в UNIX Timestamp

Возможные варианты поля result:

- ok - запрос выполнен успешно, позиция найдена.
- failed - позиция не найдена или возникла ошибка при выполнении, см. детальное описание в лог-файле.

1.2.5 cancel_task

Останавливает или отменяет задачу на запись. Запрос на отмену игнорируется, если все параметры пусты или отсутствуют. Метод не удаляет файлы записи.

Пример запроса:

```

{
  "jsonrpc": "2.0",
  "method": "cancel_task",
  "id": "some-id",
  "params": {
    "channel_id": 1,
    "record_location": "/opt/programs/"
  }
}

```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
record_locationstr		Полный путь до директории, в которую производится запись	Да	

Возвращаемое значение всегда равно «ok».

1.2.6 cancel_task_by_id

Останавливает или отменяет задачу на запись по её идентификатору. Метод не удаляет файлы записи.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "cancel_task_by_id",
  "id": "some-id",
  "params": {
    "task_id": 1,
  }
}
```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
task_id	int	Идентификатор задачи	Да	Больше нуля

Возвращаемое значение всегда равно «ok».

Коды ошибок:

Код	Описание
100	отсутствует объект params
101	параметр task_id отсутствует или некорректен

1.2.7 get_file_and_offset

Останавливает или отменяет задачу на запись по её идентификатору. Метод не удаляет файлы записи.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_file_and_offset",
  "id": "some-id",
  "params": {
    "channel_id": 1,
    "timestamp": 1452124212,
  }
}
```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
timestamp	int	Временная метка в UNIX Timestamp	Да	
aid	str	Идентификатор аккаунта	Нет	

Возвращает объект со следующими полями:

Название	Тип	Описание
offset	str	Байтовое смещение в файле по заданной временной метке, конвентированно в строку; в случае ошибки запроса равно -1
file	str	Полный путь до файла записи
timestamp	int	Временная метка найдённого смещения в UNIX Timestamp

Коды ошибок:

Код	Описание
100	Отсутствует объект params
101	Отсутствует один из обязательных аргументов
200	Смещение не найдено

1.2.8 get_tasks_list

Возвращает список всех задач.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_tasks_list",
  "id": "some-id",
  "params": {}
}
```

Метод не имеет параметров. Метод возвращает массив объектов со следующими полями:

Название	Тип	Описание
task-id	int	Идентификатор задачи
channel-id	int	Идентификатор канала
channel-name	str	Имя канала
stream-source	str	Источник записи
path	str	Полный путь до директории, в которую производится запись
start-time	int	Время фактического начала записи
stop-time	int	Время фактического окончания записи
expected-duration	int	Заданная длительность записи
repeat-period	int	Период записи
lifetime	int	Разность между временем начала записи и временем её удаления, не учитывает блокировку активных записей
status	int	Статус задачи
number-of-records	int	Количество сегментов, всегда равно 1

1.2.9 get_config

Запрашивает конфигурацию.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_task",
  "id": "some-id",
  "params": {}
}
```

Метод не имеет параметров. Метод возвращает конфигурацию MicroPVR в виде JSON-объекта.

Коды ошибок:

Код	Описание
201	Общая ошибка чтения конфигурации

1.2.10 set_config

Применяет конфигурацию.

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
config	obj	Конфигурация в виде JSON-объекта	Да	

Возвращаемое значение всегда равно «ok».

Коды ошибок:

Код	Описание
100	Отсутствует объект params
101	Пропущен параметр config
220	Ошибка резервного копирования файла конфигурации
221	Ошибка записи конфигурации
222	Ошибка применения новой конфигурации, предыдущая восстановлена из резервной копии
223	Ошибка применения новой конфигурации и восстановления предыдущей из резервной копии

1.2.11 get_records

Возвращает список всех файлов записей.

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_records_list",
  "id": "some-id",
  "params": {
    "channel_id": 1
  }
}
```

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	

Метод возвращает массив объектов со следующими полями:

Название	Тип	Описание
id	int	Идентификатор задачи (записи)
channel-id	int	Идентификатор канала
channel-name	str	Имя канала
record-name	str	Имя файла
start-time	int	Время фактического начала записи
stop-time	int	Время фактического окончания записи

1.2.12 get_playlist_name

С версии 1.8.0

Запрос плейлиста

Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "add_task",
  "id": "some-id",
  "params": {
    "channel_id": 1,
    "start_timestamp": 1494511200,
    "end_timestamp": 1494514800,
  }
}
```

(continues on next page)

(продолжение с предыдущей страницы)

```

    "playlist_type": "HLS4"
  }
}

```

Параметры:

Название	Тип	Описание	Обязательный	Ограничения
channel_id	int	Уникальный идентификатор канала	Да	Больше нуля
start_timestamp	int	Время начала первого сегмента	Да	Больше нуля
end_timestamp	int	Максимальное время начала последнего сегмента в плейлисте	Да	Больше start_timestamp
playlist_type	str	Тип плейлиста	Нет	Только допустимые значения
aid	str	Идентификатор аккаунта	Нет	

При запросе может быть возвращён уже существующий плейлист с тем же start_timestamp, channel_id и playlist_type, но с большим end_timestamp. На данный момент все сегменты в возвращаемом плейлисте относятся к одной записи.

Допустимые значения playlist_type:

- HLS4 - HLS-плейлист на основе байтового диапазона (byte range);
- RMPD - MPEG-DASH-плейлист на основе байтового диапазона (byte range), функция является экспериментальной и рекомендуется к использованию только для тестирования.

Код	Описание
100	Отсутствует объект params
101	Один из параметров неверен
200	Не найдена запись, удовлетворяющая запросу

1.2.13 is_alive

Запрашивает данные о доступности и загруженности видеосервера. Пример запроса:

```

{
  "jsonrpc": "2.0",
  "method": "is_alive",
  "id": "some-id",
  "params": {}
}

```

Метод не имеет параметров. Метод возвращает объект со следующими полями:

Название	Тип	Описание
is_alive	bool	true, если значение score не превышает максимальное
score	float	Оценка загруженности сервера
session_count	int	Примерное количество активных сессий

1.2.14 get_request_counter

С версии 1.9.0

Возвращает статистику запросов. Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_request_counter",
  "id": "some-id",
  "params": {}
}
```

Метод не имеет параметров. Метод возвращает объект со следующими полями:

Название	Тип	Описание
overall	int	Общее количество запросов
near_end	int	Количество запросов не далее, чем 180 секунд от реального времени
successful	int	Количество успешных запросов

Метод возвращает статистику за последние N секунд, где N задаётся параметром конфигурации `monitor-request-lifetime`.

1.2.15 get_channel_list

С версии 1.9.0

Возвращает список записываемых каналов и статистику уникальных запросов. Пример запроса:

```
{
  "jsonrpc": "2.0",
  "method": "get_channel_list",
  "id": "some-id",
  "params": {}
}
```

Метод не имеет параметров. Метод возвращает список объектов со следующими полями:

Название	Тип	Описание
channel_id	int	Идентификатор канала
channel_names	list	Список имён каналов
channel_sources	list	Список источников
channel_counter	int	Количество клиентов, выполнявших запрос контента для данного канала

Метод возвращает статистику за последние N секунд, где N задаётся параметром конфигурации `monitor-request-lifetime`. В `channel_counter` не учитываются запросы с пустым полем `aid`.