Консольная команда

English · Руководство · Операции · Обёртки · ИИ-агент

Содержание

- Команды
- Запуск операции
- Вывод
- Итог и коды выхода
- Куда ложится результат
- Обёртки
- Очистка папок

`media_cli.cmd` в корне программы запускает то же, что кнопки окна: операцию, с теми же полями и теми же пресетами. Окно — лицо этой команды: всё, что оно запускает, оно запускает через неё, и первая строка каждого прогона в его журнале — готовая команда, которую можно повторить в консоли.

Команда нужна, когда операцию запускает не человек у окна: пакетный файл, другой инструмент, ИИ-агент, ночь кодирования.

Команды

Команда	Что делает
`describe`	Операции, их поля с вариантами и значениями по умолчанию, пресеты и раскладка окна — одной строкой JSON.
`run`	Запустить операцию.
`wrapper ИМЯ`	Запустить обёртку из `Scripts\` по имени: `media_cli.cmd wrapper ff-x264-crf14`.

Команда	Что делает
`map`	Карта программы из её данных: разделы, операции, поля, пресеты, обёртки, тексты на двух языках. `--out ФАЙЛ` пишет её в файл; `--docs` пишет справочные документы (`OPERATIONS_*`, `SCRIPTS_*` и таблицы этого руководства).
`components`	Версии yt-dlp, Deno, mpv, FFmpeg, 7-Zip и драйвера видеокарты; `--latest` добавляет свежие выпуски (GitHub спрашивается не чаще раза в сутки).
`plan --operation install_ffmpeg`	Что скачает установщик: файлы, источник, размер. Ничего не скачивается.
`version`	Номер версии программы.

`--root ПАПКА` перед командой называет другую копию программы; `--language ru` или `en` — язык журнала (по умолчанию — язык окна).

Запуск операции

```
.\media_cli.cmd run --operation storage_run
.\media_cli.cmd run --operation editing_run --set editing_profile=prores_lt_mov
.\media_cli.cmd run --operation storage_run --set storage_profile=hevc10_crf14 --set
encode_backend=cuda `
    --source "D:\Съёмка\День 1" --target "D:\Кодировано" --result
D:\Кодировано\outcome.json
.\media_cli.cmd run --operation youtube_run --set
"youtube_url=https://www.youtube.com/watch?v=..."
```

Ключ	Назначение
`--operation ID`	Идентификатор операции. Таблица ниже даёт его для каждого режима окна; `describe` — все.
`--set ИМЯ=ЗНАЧЕНИЕ`	Значение одного поля; можно повторять. Флажок — `true` или `false`, ряд флажков — значения через запятую (`--set target_codecs=x264,hevc_x265`). Поле пресета (`storage_profile`, `editing_profile` ...) ставит поля, которые оно называет, когда его назвали, остальные `--set` ложатся поверх; не названное — каждое поле остаётся при своём значении по умолчанию: пресет, который форма показывает первым, сам не применяется.

Ключ	Назначение
`--params ФАЙЛ`	JSON-файл «поле: значение». `--set` ложится поверх.
`--source ПУТЬ`	Папка (с подпапками) или один файл. Без ключа — `input` программы.
`--target ПУТЬ`	Папка. Без ключа — `output` программы.
`--only ВИДЫ`	Какие виды файлов брать из Источника: `mkv` или `mov,mxf`.
`--result ФАЙЛ`	Куда записать итог: файл `.json` вне Источника и вне `config\`. Полным путём или вызывать команду из папки программы.
`--cancel-file ФАЙЛ`	Файл остановки: когда он появляется, FFmpeg завершается, недописанный файл убирается.
`--dry-run`	Ничего не запускать: назвать команды FFmpeg или yt-dlp, которые операция запустила бы. Операция, которая не умеет только назвать их, отклоняется.

Имена полей и их вариантов — те же, что в `describe` и в списке операций. Поле, которое не задано, идёт со значением по умолчанию — как в только что открытой форме окна. Относительные `--source` и `--target` считаются от папки программы, относительные `--result`, `--params`, `--cancel-file` — от папки, из которой вызвана команда.

Что такое каждый режим окна

Раздел	Режим	Вызов
АУДИО	ИЗВЛЕЧЬ	`media_cli.cmd run --operation audio_run --set audio_workflow=extract`
АУДИО	ЗВУК В ВИДЕО	`media_cli.cmd run --operation audio_run --set audio_workflow=in_video`
АУДИО	КОНВЕРТИРОВАТЬ	`media_cli.cmd run --operation audio_run --set audio_workflow=audio_files`
КОДИРОВАНИЕ	АРХИВ	`media_cli.cmd run --operation archive_run`
КОДИРОВАНИЕ	МОНТАЖ	`media_cli.cmd run --operation editing_run`

Раздел	Режим	Вызов
КОДИРОВАНИЕ	ХРАНЕНИЕ	`media_cli.cmd run --operation storage_run`
КОДИРОВАНИЕ	ДОСТАВКА	`media_cli.cmd run --operation delivery_run`
КОДИРОВАНИЕ	RAW	`media_cli.cmd run --operation raw_develop_run`
УПАКОВКА	ВИДЕО	`media_cli.cmd run --operation remux_run --set remux_stream_mode=video`
УПАКОВКА	АУДИО	`media_cli.cmd run --operation remux_run --set remux_stream_mode=audio`
УПАКОВКА	ВИДЕО И АУДИО	`media_cli.cmd run --operation remux_run --set remux_stream_mode=video_audio`
УПАКОВКА	СУБТИТРЫ	`media_cli.cmd run --operation subtitles_extract`
НАРЕЗКА	ОСТАВИТЬ	`media_cli.cmd run --operation trim_run --set trim_action=keep`
НАРЕЗКА	ВЫРЕЗАТЬ	`media_cli.cmd run --operation trim_run --set trim_action=cut`
НАРЕЗКА	РАЗДЕЛИТЬ	`media_cli.cmd run --operation trim_run --set trim_action=split`
ЧАСТОТА	VARISPEED	`media_cli.cmd run --operation fps_run --set fps_mode=varispeed`
ЧАСТОТА	CONFORM	`media_cli.cmd run --operation fps_run --set fps_mode=conform`
ЦВЕТ	LUT	`media_cli.cmd run --operation grading_run`
СКАЧАТЬ	ЛУЧШЕЕ MP4	`media_cli.cmd run --operation youtube_run --set youtube_profile=best_mp4`
СКАЧАТЬ	НАДЁЖНОЕ MP4	`media_cli.cmd run --operation youtube_run --set youtube_profile=safe_mp4`
СКАЧАТЬ	СВОЁ ВИДЕО	`media_cli.cmd run --operation youtube_run --set youtube_profile=video_custom`

Раздел	Режим	Вызов
СКАЧАТЬ	ДЛЯ ЗАГРУЗКИ	`media_cli.cmd run --operation youtube_run --set youtube_profile=upload_ready`
СКАЧАТЬ	СВОЁ АУДИО	`media_cli.cmd run --operation youtube_run --set youtube_profile=audio_custom`
СКАЧАТЬ	СУБТИТРЫ	`media_cli.cmd run --operation youtube_run --set youtube_profile=subs_only`

СВЕДЕНИЯ, СЕРВИС и кнопки у полей:

Операция	Что это
`probe_source`	Проверить источник: Один раз запустить ffprobe по Источнику, вывести таблицу и обновить строку параметров.
`source_info`	Инфо об источниках: Прогнать ffprobe по Источнику и вывести все аудиопотоки: codec, sample_fmt, sample rate, bits, channels, bitrate, language/title.
`inventory`	Инвентаризация: Посчитать файлы в Источнике, Назначении и LUTs.
`selftest`	Самопроверка: Проверить ffmpeg, ffprobe, yt-dlp и Deno.
`profile_doctor`	Проверка профилей: Проверить операции GUI, script wrappers и общий каталог пресетов.
`hardware_capabilities`	Возможности оборудования: Короткий FFmpeg smoke для CPU, QSV/NVENC/AMF H.264 и HEVC, SVT-AV1 и dav1d.
`preset_test_matrix`	Тест матрицы пресетов: Прогнать portable smoke на 1-секундных файлах по audio, remux, FPS, LUT, архиву, монтажу, хранению и публикации.
`install_7zip`	Установить 7-Zip: Установить или обновить portable 7-Zip; нужен для FFmpeg 7z payloads.
`install_ffmpeg`	Установить FFmpeg: Установить или обновить переносную FFmpeg от Gyan (GPL) в Tools\ffmpeg по драйверу NVIDIA: самая новая стабильная, не ниже 9.0.2, без карты NVIDIA и с LTS-драйвером 610.88 и новее; ниже него 8.0.1 (570-610.87) или 7.1, чтобы работал NVENC, и доступны не все операции.
`install_ytdlp`	Установить yt-dlp: Установить или обновить portable yt-dlp и Deno в Tools.

Операция	Что это
`install_mpv`	Установить mpv: Установить или обновить portable mpv в Tools/mpv: плеер нарезки.
`install_cineraw`	Установить проявку ProRes RAW: Собрать cineraw из libcraw в Tools/cineraw из исходников: переносной MSYS2 для сборки (около 185 МБ загрузки, 5 минут), удаляется после; cineraw проявляет ProRes RAW на видеокарте с Vulkan 1.4 (NVIDIA RTX).
`youtube_formats`	Форматы ссылки: Какие форматы есть у ссылки - разрешение, кодеки, размер - списком в журнале, ничего не скачивается. Взять один из них - профилем «Своё видео».
`youtube_cookies_import`	Взять под DPAPI: Файл поля «Файл cookies» запечатывается DPAPI для этой учётной записи Windows в config\cookies.dpapi: загрузки с Cookies - «Файл» берут эту копию, и открытый файл больше не нужен.

Вывод

Строки журнала операции — как в окне: `[OK]`, `[WARN]`, `[ERR]`, `[CMD]` — команда, с которой запущен FFmpeg, `[PREFLIGHT]` — проверки до запуска, `[PROGRESS] clip.mp4 -> x264 [#####---] 42% 00:01:12/00:02:50 speed=3.4x` и под ней родная строка FFmpeg `frame= … fps= … time= … speed=`, в конце — последнее слово FFmpeg `Lsize=`. Те же строки уходят в файл журнала в `logs\`, отчёт прогона — в `report\`.

Программа, которая читает вывод трубой, получает ещё служебные строки: `::media::{"started": …}`, `{"progress": 0.42}`, `{"activity": …}`, `{"finished": {"outcome": …}}`. Консоль их не показывает.

Итог и коды выхода

Код	Итог	Что значит
0	`done`	Операция завершена.
1	`failed`	Запуск был, и что-то сломалось, или файл упал или был отклонён: журнал говорит что.

Код	Итог	Что значит
2	— / `refused`	Неверный вызов — поле, которого у операции нет, значение не из её вариантов, `--dry-run` операции, которая не умеет только назвать команды; или проверки до запуска отклонили его — в Источнике нет файлов нужного вида, нет FFmpeg. Ничего не запускалось.
130	`stopped`	Остановлено файлом остановки (`--cancel-file`) — до первого файла или посреди него.

Файл `--result`:


```
{
 "operation": "storage_run",
 "outcome": "done",
 "ok": true,
 "message": "Operation finished.",
 "data": {
  "processed": 1,
  "files": 1,
  "log_file": "...\\logs\\2026-10-15 22-34-54-804_storage_run.log",
  "batch_report_md": "...\\report\\2026-10-15 22-34-54-
804_storage_run\\batch_report.md"
 },
 "parameters": { "storage_profile": "x264_crf14", "target_codecs": ["x264"], "crf":
14, "...": "..." },
 "source": "D:\\Съёмка\\День 1",
 "target": "D:\\Кодировано",
 "seconds": 2.1
}
```


`parameters` — все значения, с которыми шла операция, включая значения по умолчанию: по ним прогон можно повторить в точности. `data` — то, что говорит операция: у медиа-операции `processed` (записано файлов), `skipped` (пропущено, причина — в журнале), `files` (файлов Источника), `failed` там, где операция идёт дальше упавшего файла, и `refused` там, где она отклонила файл; у скачивания `processed`, `downloaded` — пути записанного — и `existing`, найденное там раньше.

Куда ложится результат

В папку операции в Назначении, а в ней — в папку настроек, названную по цели и её значениям, а не по пресету: `Storage\x264_crf14_slow\clip.mp4`, `Editing\prores422\clip.mov`, `Archive\…`, `Delivery\…`, `Audio\…`, `Remux\…`, `Trim\…`, `FPS\…`, `Color\…`, `Subtitles\…`; подпапки Источника повторяются ниже. Точный путь печатает `--dry-run`. Скачанное — в `Download\<автор>\<дата> - <название> [<id>].<ext>` (плейлист — `Download\<номер> - <название> [<id>].<ext>`).

Файл, который там уже есть, не перезаписывается без `--set overwrite=true`: журнал пишет `[SKIP] ИМЯ already exists and Overwrite is off`, FFmpeg для него не запускается, и итог считает его в `skipped`, а не в `processed`.

Обёртки

95 файлов в `Scripts\` — `ff-*`, `audio-*`, `yt-*` — каждый готовая задача со своими значениями: двойным щелчком, из консоли или по имени через команду: `media_cli.cmd wrapper ff-prores-lt-mov`. У каждой в списке обёрток — тот же вызов через `run`. Обёртки нарезки берут файл и точки из переменных `AUDION_TRIM_FILE`, `AUDION_TRIM_START`, `AUDION_TRIM_END`; скачивание — ссылки после имени или `input\urls.txt`.

Меню `launcher_project.cmd`, `launcher_tools.cmd` и `cli\launcher-*.cmd` находят обёртку набором букв (FZF).

Очистка папок

Операции, которые очищают Источник, Назначение или рабочую папку, удаляют насовсем и из команды не запускаются: окно очищает их через Корзину.