

Workspace, Reports И Терминал

Source И OUT

Source - входная папка. **OUT** - папка назначения. По умолчанию это:

```
Source\  
Transcoded\
```

Поля путей read-only. Менять путь нужно через кнопки выбора рядом с полем. Это сделано специально: история, пины и safety-checks остаются в одном управляемом слое.

Рекурсивный Source

Обычные media batch-операции обходят Source рекурсивно. OUT сохраняет относительную структуру подпапок.

Правило:

```
Source\<relative>\file.ext -> OUT\<operation>\<suffix>\<relative>\file.ext
```

Это защищает от ситуации, когда несколько папок с видео превращаются в одну плоскую папку аудиоэкстрактов. Если выбран кастомный OUT, он просто заменяет **OUT** в этом правиле: внутренняя структура остаётся такой же.

Path Cache И Pins

GUI хранит историю путей и позволяет закреплять важные папки.

Использование:

- выбери папку кнопкой выбора;
- pin закрепляет путь;

- unpin снимает закрепление;
- delete удаляет путь из истории;
- dropdown показывает recent и pinned entries.

История нужна для быстрых переключений между проектами, а pin - для постоянных рабочих корней.

Cleanup-Кнопки

Есть отдельные действия:

- очистить Source;
- очистить OUT;
- очистить workspace/report;
- открыть папки и отчёты.

Cleanup внешней выбранной папки требует подтверждения. Перед подтверждением всегда проверяй полный путь, особенно если Source/OUT указывает не на проектную папку.

Probe Source

Probe Source запускает ffprobe вручную. Он не сканирует Source постоянно, потому что большие папки и сетевые диски могут быть дорогими.

Используй probe, когда нужно понять:

- контейнеры;
- видео/аудио codec;
- FPS;
- duration;
- stream layout.

Терминал Операций

Правая колонка показывает:

- preflight;
- собранные команды;

- stdout/stderr;
- FFmpeg progress;
- итоговые строки;
- пути к отчётам.

FFmpeg-прогресс выводится строками вида:

```
[PROGRESS] clip.mov [####-----] 42% 00:12/00:30
```

Верхний progress bar показывает общий ход batch-a, а terminal progress - текущий файл.

Кнопка Команда

Команда запускает dry-run текущей GUI-конфигурации. Это лучший способ проверить:

- какой FFmpeg/yt-dlp вызов будет построен;
- какой Source/OUT используется;
- какой codec/container выбран;
- включилась ли LUT;
- не включился ли overwrite.

Для новой операции нормальная последовательность:

1. выбрать профиль;
2. поправить параметры;
3. нажать **Команда** ;
4. включить **Тест: первый файл** ;
5. запустить реальную обработку.

Report

Страница **Report** показывает последние операции. Для каждой записи обычно доступны:

- открыть OUT;
- открыть report folder;
- открыть JSON;
- открыть log;

- повторить команду.

Обычный batch report:

```
batch_report.md  
batch_report.json
```

Аудио report:

```
audio_report.md  
audio_report.json
```

JSON нужен для дальнейшей автоматизации и проверки, Markdown - для чтения человеком.

Ручная Командная Строка

Нижний terminal input предназначен для служебных команд, а не для обычного запуска media-профилей.

Возможности:

- выбор Shell;
- CWD;
- история до 200 команд;
- pinned commands;
- добавление файла/папки через picker;
- многострочный ввод;
- очистка history/cache.

Если команда относится к media workflow, лучше сначала проверить, нет ли уже готовой операции в левом дереве.

Одна Операция За Раз

Один GUI-instance рассчитан на одну активную media-операцию. Это защищает:

- progress bar;
- terminal log;

- report context;
- output file naming;
- hardware resource contention.

Два независимых FFmpeg-процесса технически могут работать одновременно, но они делят CPU/GPU/диск и могут писать в конфликтующие пути. Для обычной работы запускай операции последовательно.