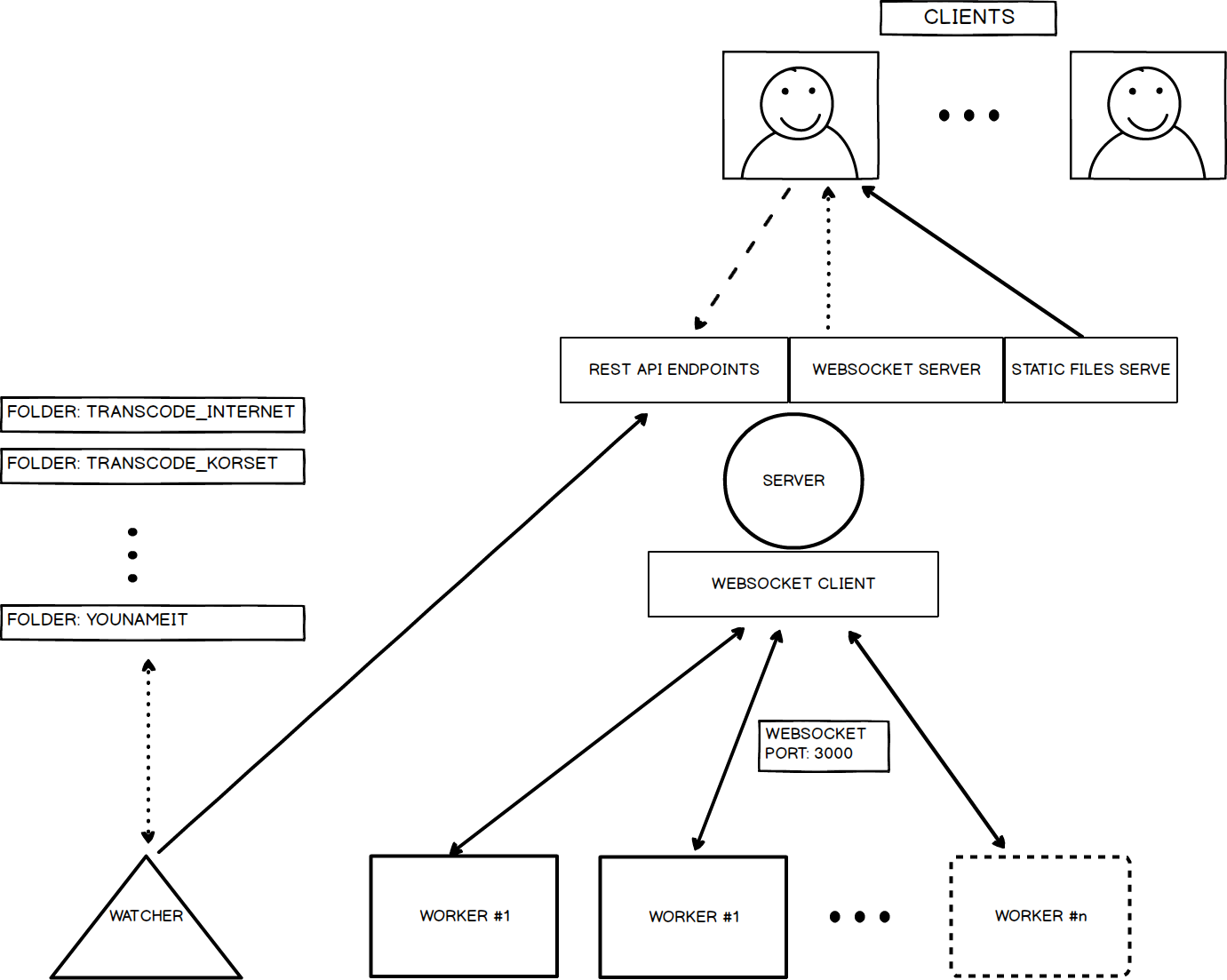
Media Converter

**Архитектура ПО**



**Назначение модулей**

ПО условно можно рассматривать как совокупность модулей, которые взаимодействуют между собой через API (Application Program Interface – программный интерфейс)

Модуль **SERVER**:

* Является веб сервером: слушает входящие соединения на 80-м порту
* Приняв новое соединение отдает клиенту статические файлы: html, css, js
* Слушает входящие соединения по протоколу WebSocket на порту 3000 для обновления в реальном времени данных о кодировании и состоянии обработчиков
* Ожидает запросы от клиентов на REST API конечных точках
* Взаимодействует с базой данных MySQL для сохранения логов и части настроек
* Координирует очередь файлов на кодировку
* Отправляет запросы на обработчики как WebSocket клиент, отслеживает их статусы и

оповещает клиентов

Модуль **WORKER**:

В основе модуля лежит ffmpeg и ffprobe

* Слушает входящие соединения по протоколу WebSocket на порту 3000
* Получив указание на обработку файла:
  + а) проверяет расширение файла
  + б) проверяет метаданные
  + в) оповещает сервер что файл принят на обработку, выслав доп. Инфу
  + б) Анализирует файл на наличие проблем и составляет стек команд для коррекции
  + Создает промежуточный, частично корректированный файл с расставленными в определенных местах ключевыми кадрами (вычисляются на основании кол-ва потоков в системе и хронометража кодируемого файла)
  + Режет промежуточный файл на части и кодирует каждую в отдельном потоке
  + Склеивает файлы и кладет итоговый в папку соответствующую категории

Модуль **WATCHER**:

Вынесен отдельно как микро сервис. Не имеет UI(user interface) интеграции

- Следит за изменениями в указанных папках

- При обнаружении нового файла отправляет запрос к серверу с данными о файле

«Модуль» **FRONTEND**:

- Представляет собой SPA(single page application). Это то что видит пользователь зайдя на веб страницу <http://vbram>

**Текущие настройки:**

Модуль SERVER: это виртуалка. Настроен на 172.26.64.30, имеет запись в DNS как vbram.tv-culture.ru. OS – Ubuntu Server 18.04. MySQL - 8.0.15. Доступ можно получить по SSH. Также можно получить доступ к файлам приложения по CIFS. Сервер настроен на автоматическое восстановление работы после перезагрузки. Приложение автоматом перезапуститься после краша. Всем этим управляет pm2 runtime(process manager – специальный third party модуль для управления nodejs приложениями).

Управление приложением:

“pm2 start” – запустить приложение

“pm2 restart **server**” – перезапустить приложение

“pm2 logs” – посмотреть логи

“pm2 monit” – показать подробный монитор приложения

“pm2 ls” – краткий отчет о том, что запущено

Модуль WORKER: настроены 4 обработчика bl-render1 – bl-render4. ОС – Windows 2012 R2. Файлы приложения лежат в C:\Program Files\MediaConverter. Сервера настроены на автоматическое восстановление работы приложения после перезагрузки и краша. Приложением управляет pm2(команды см. модуль server. Для перезапуска - “pm2 restart **worker”**). Обработчики можно добавить/удалить/отключить через web интерфейс приложения в меню «Настройки». Для нормальной работы должен быть открыт порт 3000. В случае неудачного кодирования видео: файл автоматом копируется в [\\media\v\Transcode\_Internet\_old](file:///\\media\v\Transcode_Internet_old) и [\\media\v\\_vbram\_bad\_files](file:///\\media\v\_vbram_bad_files) для дальнейшего анализа. В web интерфейсе регистрируется событие с ошибкой, но без подробностей. Для более детального лога нужно зайти на тот обработчик, который не справился с перекодированием (это можно узнать по ID в web) и посмотреть лог командой “pm2 logs”

Модуль WATCHER: Одинокий файлик лежит в C:\Program Files\FileWatcher на bl-render4. Управляется также через pm2( команды см. модуль server. Для перезапуска - “pm2 restart **watcher”**) Конфигурирование осуществляется прямо через редактирование файла: Всё что находится между комментариев // -------- CONFIG ------- и // --------------------- можно редактировать.

Пример:

**const** sources = [

["\\\\media\\v\\Transcode\_Internet", 3],

["\\\\media\\v\\Transcode\_Agentstva", 1],

["\\\\media\\v\\Transcode\_GlavnayaRol", 2],

["\\\\media\\v\\Transcode\_Kor\_Set", 4],

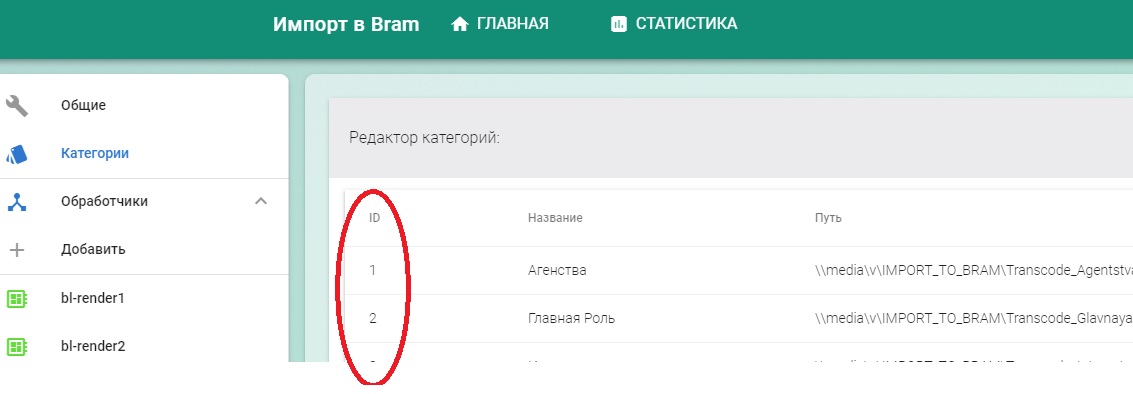
["\\\\media\\v\\Transcode\_Nabludatel", 6],

["\\\\media\\v\\Transcode\_Vesti", 7],

["\\\\media\\v\\Transcode\_Monitoring", 5]

];

[\\\\media\\v\\Transcode\_Internet](file:///\\\\media\\v\\Transcode_Internet) – это папка для отслеживания. Цифра рядом это ID категории (оно будет сопоставлено с местом сохранения файла). ID можно взять в настройках категорий:



Модуль FRONTEND: Это файлы веб-морды ☺ Через него можно закачивать файлы на обработку, конфигурировать базу данных, категории, добавлять/удалять/отключать обработчики.

Важно знать, что статические файлы откомпилированы с жестко заданным адресом SERVER-а (vbram). Если попытаться изменить DNS адрес сервера, то фронтэнд будет делать запросы к старому адресу. Выход – перекомпиляция с новыми настройками или найти в JS коде строчки с названием старого и изменить на новый.

В разделе СТАТИСТИКА показываются данные за последние 3 дня. Более поздние записи удаляются

Если был изменен адрес базы данных, то приложение SERVER нужно перезапустить – “pm2 restart server”

УСТАНОВКА:

Наброски…

Ставим Linux

Устанавливаем обновления apt-get update && apt-get upgrade –y

Проверяем чтобы время было настроено верно: date, **dpkg**-**reconfigure** tzdata, date -s '2009-02-13 11:31:30'

Создаем папку приложения /app

Создаем пользователя transcode: useradd transcode -> пароль такой же

Меняем владельца на папке chown transcode app

Создаем папку videoSource в папку /app -> /app/videoSource

Устанавливаем клиент cifs: apt-get install **cifs**-utils

Редактируем fstab чтобы виндовая шара автоматом монтировалась при рестарте:

//172.26.64.128/v/mediaConverterSrc /app/videoSource/ cifs username=mediaconverter,password=transcodeapp 0 0

Монтируем шару: mount –a

Добавляем репозиторий для установки Node.js –

curl -sL https://deb.nodesource.com/setup\_10.x | sudo bash -

Ставим Node: apt-get install -y nodejs

Ставим samba: apt-get install samba

В /etc/samba/samba\_conf добавляем следующие данные:

[global]

Dos charset = cp850

Unix charset = ISO-8859-1

Force user = transcode

[app]

Browseable = yes

Writable = yes

Path = /app

Create mask = 077

Добавляем пользователя transcode в samba db:

Useradd transcode

smbpasswd -a transcode

Рестартим сервис: sudo service smbd restart

Ставим MySQL server: sudo apt install mysql-server

По завершению установки запускаем команду настройки: mysql\_secure\_installation

Проверяем коннектом к БД: mysql –uroot –p Вводим пароль

Создаем базу данных: create database media\_converter;

Выходим из управления БД командой exit

Устанавливаем процесс менеджер для автоматического рестарта приложения в случае ошибки и авто старта после ребута: npm install –g pm2

Заходим в дирректорию /app и запускаем приложение через pm2:

pm2 start app.js --name server

Сэйвим конфигурацию: pm2 save

Добавляем в автозагрузку: pm2 startup

ВСЕ ГОТОВО! Заходим через браузер на <http://host_ip>

*[mysqld]  
wait\_timeout=300  
interactive\_timeout = 300*

show variables like "wait\_timeout";

service mysql restart