

Спутник статьи 3.6 Блока 3 (Эволюция кодеков). Печатать на A4.

Идентичность — два royalty-free кодека из покупки On2 Google

VP8	Выпущен On2 в сент. 2008; открыт в мае 2010 после покупки On2 Google за \$124,6 млн
VP9	Заморозка июнь 2013; ~40–45% ниже битрейт vs VP8; ~5–15% позади HEVC
Референс	libvpx (BSD-style) — на него линкуются и FFmpeg, и WebM Project
Контейнер	WebM (профиль Matroska) по умолчанию; оба кодека также идут в MP4

Профиль + уровень VP9 — что куда отгружать

Profile @ Level	Бит / chroma	Разрешение / fps	Где отгружается
Profile 0 @ 3.1	8-бит 4:2:0	1280 x 720 @ 30	Мобайл, WebM-в-вебе
Profile 0 @ 4.1	8-бит 4:2:0	2048 x 1080 @ 60	1080p60 SDR-стриминг
Profile 0 @ 5.0	8-бит 4:2:0	4096 x 2160 @ 30	4K30 стриминг
Profile 0 @ 5.1	8-бит 4:2:0	4096 x 2160 @ 60	ДЕФОЛТ: 4K60 SDR
Profile 2 @ 5.1	10/12-бит 4:2:0	4096 x 2160 @ 60	4K HDR (YouTube HDR)
Profile 1 @ 5.1	8-бит 4:2:2/4:4:4	4096 x 2160 @ 60	Нишевый профессиональный
Profile 3 @ 5.1	10/12-бит 4:2:2/4:4:4	4096 x 2160 @ 60	High-bit-depth архив
Profile 0 @ 6.1	8-бит 4:2:0	8192 x 4320 @ 60	8K, будущее / часть TB

Рецепты FFmpeg + libvpx-vp9, которые реально пригодятся

2-pass VOD 1080p60, цель 3 Мбит/с, quality preset:

```
ffmpeg -i in.mov -c:v libvpx-vp9 -b:v 3M -minrate 1.5M -maxrate 4.35M \
  -pass 1 -row-mt 1 -tile-columns 2 -speed 4 -auto-alt-ref 1 -an -f null /dev/null
ffmpeg -i in.mov -c:v libvpx-vp9 -b:v 3M -pass 2 -row-mt 1 -tile-columns 2 \
  -speed 1 -auto-alt-ref 1 -lag-in-frames 25 -c:a libopus -b:a 128k out.webm
```

4K HDR Profile 2 (10-бит BT.2020 PQ), constant quality:

```
ffmpeg -i in.mxf -c:v libvpx-vp9 -profile:v 2 -pix_fmt yuv420p10le \
  -crf 28 -b:v 0 -row-mt 1 -tile-columns 3 -speed 2 -auto-alt-ref 1 \
  -color_primaries bt2020 -color_trc smpte2084 -colorspace bt2020nc out.webm
```

Live-ингест, 720p30 CBR, низкая латентность:

```
ffmpeg -re -i in -c:v libvpx-vp9 -b:v 2M -minrate 2M -maxrate 2M -speed 6 \
  -row-mt 1 -error-resilient 1 -lag-in-frames 0 -g 60 -keyint_min 60 \
  -auto-alt-ref 0 -c:a libopus -b:a 96k -f webm tcp://ingest:9000
```

WebRTC и снимок браузеров / аппаратного декода в 2026

- VP8 — mandatory-to-implement видекодек WebRTC (RFC 7742, март 2016).
- Каждый Chrome, Firefox, Safari, Edge в 2026 предоставляет программный декод VP8 + VP9.
- Аппаратный VP9: универсально на Android после 2016, Smart TV после 2015, iOS 14+ (HW на A15+).
- Chrome на Linux: VP9 программно есть; аппаратно — лоскутно (зависит от VA-API).
- Аппаратный VP8 редок — сценарий WebRTC обычно справляется на CPU.

Почему VP9 на ~40–45% меньше VP8

- Superblocks 64x64 рекурсивно до 4x4 (VP8 фикс. 16x16), с прямоугольными разбиениями.
- 10 intra-модов включая TrueMotion + умнее использование соседей; прямоугольные intra-сплиты.
- Eighth-pixel motion + 8-tap luma фильтр (у VP8: quarter-pel + 6-tap); 3 рефа вкл. alt-ref.
- Трансформы до 32x32 плюс ADST для асимметричных residual (у VP8 — только 4x4).
- Tile-based параллельный декод + row-mt в libvpx; загружает 8–16 CPU-ядер на 4K60.