

# Recommended Settings for VOD

---

[developers.google.com/media/vp9/settings/vod](https://developers.google.com/media/vp9/settings/vod)

## Purpose

---

This section provides recommendations for VP9 encode settings when performing file-based encoding (i.e. not live).

These recommendations are designed for the following goals:

- A balance between quality and encoding speed
- The minimal bit rate to achieve reasonable quality
- Settings to accommodate a wide range of content types

These recommendations do not:

- Address live encoding
- Optimize for specific content types (e.g. high-motion sports footage)
- Configure settings for specific device or network requirements

## Single Resolution Encoding

---

VP9 supports a range of frame sizes, from tiny resolutions through 4K. Larger frame sizes give higher quality but they require more bandwidth to deliver, and more processing power to decode.

If you are creating a single resolution, 640x480 is a safe bet for a broad range of web and mobile devices. The following FFmpeg command-line parameters allow you to create a single-resolution file at 750kbps.

```
-vf scale=640x480 -b:v 750k -quality good -speed 0 -crf 33 \  
-c:v libvpx-vp9 -c:a libopus output.webm
```

## Multi-Resolution Encoding

---

If you plan on targeting multiple resolutions, or if your delivery network has varying bandwidth, it is recommended to create multiple resolutions. Your player can then control which resolution is being sent to the viewer.

Multi-resolution video encodes are often used in adaptive bitrate streaming, where the video player switches between resolutions in real-time based upon the user's bandwidth. For example, [Shaka Player](#) allows you to play back multi-resolution encodes, where each VP9 encode is in a separate file and a DASH manifest provides information about each encode.

For more information about packaging video in adaptive streaming formats please see the [Shaka Packager](#). This guide will focus upon the encode settings for VP9 across multiple resolutions.

## Recommended Settings

---

All settings below can be used for individual files at various resolutions, combined they provide a comprehensive set suitable for adaptive streaming. Note that the 640x480 version has two versions, one low quality (LQ) and the other medium quality (MQ).

### Bitrate

---

Constrained Quality (CQ) mode is recommended when encoding VP9 files for on-demand viewing. This encoding mode allows you to specify a target average bit rate, while controlling both the maximum quality of the video as well as minimum and maximum bitrate.

The following bitrates are suggested as baselines for web and mobile distribution. These suggestions minimize bitrate while achieving video quality suitable for consumer web and mobile distribution; think of them as a 'low' bit rate recommendation that can still achieve reasonable quality.

For the examples above, it is recommended that minimum bitrate be set at 50% of target bitrate, and maximum at 145% of target.

<b>Frame Size/Frame Rate</b>	<b>Target Bitrate (VOD, kbps)</b>	<b>Min Bitrate (50%)</b>	<b>Max Bitrate (145%)</b>
320x240p @ 24,25,30	150	75	218
640x360p @ 24,25,30	276	138	400
640x480p @ 24,25,30	512 (LQ), 750 (MQ)	256 (LQ) 375 (MQ)	742 (LQ) 1088 (MQ)
1280x720p @ 24,25,30	1024	512	1485
1280x720p @ 50,60	1800	900	2610
1920x1080p @ 24,25,30	1800	900	2610
1920x1080p @ 50,60	3000	1500	4350
2560x1440p @ 24,25,30	6000	3000	8700
2560x1440p @ 50,60	9000	4500	13050
3840x2160p @ 24,25,30	12000	6000	17400

Frame Size/Frame Rate	Target Bitrate (VOD, kbps)	Min Bitrate (50%)	Max Bitrate (145%)
3840x2160p @ 50,60	18000	9000	26100

Figure 2a: Recommended VOD bitrates

In FFmpeg, bit rate is controlled with the following commands:

### FFmpeg

`-b:v <arg>` Sets bitrate (e.g. 500k)

`-minrate <arg>` Sets minimum and maximum bitrate.  
`-maxrate <arg>`

For example, when encoding 640x480 content you could use the command-line `-b:v 750k -minrate 375 -maxrate 1088`.

### Quality

In CQ mode, you will also set the maximum quality level. The following quality levels are recommended for file-based VP9 encoding:

Frame Height	Target Quality (CQ)
240	37
360	36
480	34 (LQ) or 33 (MQ)
720	32
1080	31
1440	24
2160	15

In FFmpeg, quality is set with the `-crf` command. For example, to set the quality to 33 you would use the command `-crf 33`

### Multi-pass encoding and Encoding Speed

File-based encoding allows you flexibility in speed. You can also perform multiple passes on the same material to increase quality, and choose speeds for each one.

When encoding VP9 files in FFmpeg, it is recommend to set the `-quality` parameter to `good` and then set the speed of the first and second pass according to the table below with the `-speed` parameter. This gives a good balance between encoding time and the

quality of the output.

Frame Height	Speed (First Pass)	Speed (Second Pass)
240	4	1
360	4	1
480	4	1
720	4	2
1080	4	2
1440	4	2
2160	4	2

For example, a first pass encode string in FFmpeg might include `-quality good -speed 4`.

## Keyframe Spacing

---

It is recommended to allow up to 240 frames of video between keyframes (8 seconds for 30fps content). Keyframes are video frames which are self-sufficient; they don't rely upon any other frames to render but they tend to be larger than other frame types. For web and mobile playback, generous spacing between keyframes allows the encoder to choose the best placement of keyframes to maximize quality.

In FFmpeg keyframe spacing is controlled with the `-g` command, indicating the number of frames. For 240 frames this would be `-g 240`.

## Tiling and Threading Recommendations

---

Tiling splits the video frame into multiple columns, which slightly reduces quality but speeds up encoding performance. Tiles must be at least 256 pixels wide, so there is a limit to how many tiles can be used.

Depending upon the number of tiles and the resolution of the output frame, more CPU threads may be useful. Generally speaking, there is limited value to multiple threads when the output frame size is very small.

The following settings are recommended for tiling and threading at various resolutions.

Frame Size	Number of tile-columns	Number of threads
320x240	1 ( <code>-tile-columns 0</code> )	2
640x360	2 ( <code>-tile-columns 1</code> )	4
640x480	2 ( <code>-tile-columns 1</code> )	4

Frame Size	Number of tile-columns	Number of threads
1280x720	4 ( <code>-tile-columns 2</code> )	8
1920x1080	4 ( <code>-tile-columns 2</code> )	8
2560x1440	8 ( <code>-tile-columns 3</code> )	16
3840x2160	8 ( <code>-tile-columns 3</code> )	16

In FFmpeg, the number of tiles is controlled with the `-tile-columns` parameter and the number of threads by `-threads`. For example, a 640x480 encode would use the command-line `-tile-columns 2 -threads 4`.

## FFmpeg Command Lines

---

Bringing the above recommendations together, the following FFmpeg commands may be used to encode VP9 content.

Note that the first-pass and second-pass commands are chained together. The `-y` argument in the second-pass command answers "Yes" when FFmpeg asks to overwrite the first-pass statistics file with the output video.

Also note that a 1080p source is used for encodes targeting up to 1280x720 output. A 4k source is used for larger output.

### 320x240 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=320x240 -b:v 150k \
  -minrate 75k -maxrate 218k -tile-columns 0 -g 240 -threads 2 \
  -quality good -crf 37 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-320x240.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=320x240 -b:v 150k \
  -minrate 75k -maxrate 218k -tile-columns 0 -g 240 -threads 2 \
  -quality good -crf 37 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 1 -y tos-320x240.webm
```

- Input file: [tears\\_of\\_steel\\_1080p.webm](#) (1920x800, 544.88 MB)
- Output file: [tos-320x240.webm](#) (320x240, 21.73 MB)

### 640x360 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x360 -b:v 276k \
  -minrate 138k -maxrate 400k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 36 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-640x360.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x360 -b:v 276k \
  -minrate 138k -maxrate 400k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 36 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 4 -y tos-640x360.webm
```

- Input file: [tears\\_of\\_steel\\_1080p.webm](#) (1920x800, 544.88 MB)
- Output file: [tos-640x360.webm](#) (640x360, 31.53 MB)

## 640x480 (Low Quality, 24, 25 or 30 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x480 -b:v 512k \
  -minrate 256k -maxrate 742k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 34 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-640x360-low.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x480 -b:v 512k \
  -minrate 256k -maxrate 742k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 34 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 4 -y tos-640x480-low.webm
```

- Input file: tears\_of\_steel\_1080p.webm (1920x800, 544.88 MB)
- Output file: tos-640x480-low.webm (640x480, 51.18 MB)

## 640x480 (Medium Quality, 24, 25 or 30 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x480 -b:v 750k \
  -minrate 375k -maxrate 1088k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 33 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-640x360-medium.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=640x480 -b:v 750k \
  -minrate 375k -maxrate 1088k -tile-columns 1 -g 240 -threads 4 \
  -quality good -crf 33 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 4 -y tos-640x480-medium.webm
```

- Input file: tears\_of\_steel\_1080p.webm (1920x800, 544.88 MB)
- Output file: tos-640x480-medium.webm (640x480, 69.27 MB)

## 1280x720 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=1280x720 -b:v 1024k \
  -minrate 512k -maxrate 1485k -tile-columns 2 -g 240 -threads 8 \
  -quality good -crf 32 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-1280x720-24-30fps.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=1280x720 -b:v 1024k \
  -minrate 512k -maxrate 1485k -tile-columns 2 -g 240 -threads 8 \
  -quality good -crf 32 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 4 -y tos-1280x720-24-30fps.webm
```

- Input file: tears\_of\_steel\_1080p.webm (1920x800, 544.88 MB)
- Output file: tos-1280x720-24-30fps.webm (1280x720, 98.2 MB)

## 1280x720 (50 or 60 frames per second)

---

```
ffmpeg -i tears_of_steel_1080p.webm -vf scale=1280x720 -b:v 1800k \
  -minrate 900k -maxrate 2610k -tile-columns 2 -g 240 -threads 8 \
  -quality good -crf 32 -c:v libvpx-vp9 -c:a libopus \
  -pass 1 -speed 4 tos-1280x720-50-60fps.webm && \
ffmpeg -i tears_of_steel_1080p.webm -vf scale=1280x720 -b:v 1800k \
  -minrate 900k -maxrate 2610k -tile-columns 2 -g 240 -threads 8 \
  -quality good -crf 32 -c:v libvpx-vp9 -c:a libopus \
  -pass 2 -speed 4 -y tos-1280x720-50-60fps.webm
```

- Input file: tears\_of\_steel\_1080p.webm (1920x800, 544.88 MB)
- Output file: tos-1280x720-50-60fps.webm (1280x720, 157.1 MB)

## 1920x1080 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=1920x1080 -b:v 1800k \  
-minrate 900k -maxrate 2610k -tile-columns 2 -g 240 -threads 8 \  
-quality good -crf 31 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-1920x1080-24-30fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=1920x1080 -b:v 1800k \  
-minrate 900k -maxrate 2610k -tile-columns 3 -g 240 -threads 8 \  
-quality good -crf 31 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -speed 4 -y tos-1920x1080-24-30fps.webm
```

- Input file: [tearsofsteel\\_4k.mov](#) (3840x1714, 6.76 GB)
- Output file: [tos-1920x1080-24-30fps.webm](#) (1920x1080, 151.51 MB)

## 1920x1080 (50 or 60 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=1920x1080 -b:v 3000k \  
-minrate 1500k -maxrate 4350k -tile-columns 2 -g 240 -threads 8 \  
-quality good -crf 31 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-1920x1080-50-60fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=1920x1080 -b:v 3000k \  
-minrate 1500k -maxrate 4350k -tile-columns 3 -g 240 -threads 8 \  
-quality good -crf 31 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -speed 4 -y tos-1920x1080-50-60fps.webm
```

- Input file: [tearsofsteel\\_4k.mov](#) (3840x1714, 6.76 GB)
- Output file: [tos-1920x1080-50-60fps.webm](#) (1920x1080, 238.18 MB)

## 2560x1440 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=2560x1440 -b:v 6000k \  
-minrate 3000k -maxrate 8700k -tile-columns 3 -g 240 -threads 16 \  
-quality good -crf 24 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-2560x1440-24-30fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=2560x1440 -b:v 6000k \  
-minrate 3000k -maxrate 8700k -tile-columns 3 -g 240 -threads 16 \  
-quality good -crf 24 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -speed 4 -y tos-2560x1440-24-30fps.webm
```

- Input file: [tearsofsteel\\_4k.mov](#) (3840x1714, 6.76 GB)
- Output file: [tos-2560x1440-24-30fps.webm](#) 2560x1440, 533.54 MB)

## 2560x1440 (50 or 60 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=2560x1440 -b:v 9000k \  
-minrate 4500k -maxrate 13050k -tile-columns 3 -g 240 -threads 16 \  
-quality good -crf 24 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-2560x1440-50-60fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=2560x1440 -b:v 9000k \  
-minrate 4500k -maxrate 13050k -tile-columns 3 -g 240 -threads 16 \  
-quality good -crf 24 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -speed 4 -y tos-2560x1440-50-60fps.webm
```

- Input file: [tearsofsteel\\_4k.mov](#) (3840x1714, 6.76 GB)
- Output file: [tos-2560x1440-50-60fps.webm](#) 2560x1440, 664.04 MB)

## 3840x2160 (24, 25 or 30 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=3840x2160 -b:v 12000k \  
-minrate 6000k -maxrate 17400k -tile-columns 3 -g 240 -threads 24 \  
-quality good -crf 15 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-3840x2160-24-30fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=3840x2160 -b:v 12000k \  
-minrate 6000k -maxrate 17400k -tile-columns 3 -g 240 -threads 24 \  
-quality good -crf 15 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -speed 4 -y tos-3840x2160-24-30fps.webm
```

- Input file: tearsofsteel\_4k.mov (3840x1714, 6.76 GB)
- Output file: tos-3840x2160-24-30fps.webm 3840x2160, 1.03 GB)

## 3840x2160 (50 or 60 frames per second)

---

```
ffmpeg -i tearsofsteel_4k.mov -vf scale=3840x2160 -b:v 18000k \  
-minrate 9000k -maxrate 26100k -tile-columns 3 -g 240 -threads 24 \  
-quality good -crf 15 -c:v libvpx-vp9 -c:a libopus \  
-pass 1 -speed 4 tos-3840x2160-50-60fps.webm && \  
ffmpeg -i tearsofsteel_4k.mov -vf scale=3840x2160 -b:v 18000k \  
-minrate 9000k -maxrate 26100k -tile-columns 3 -g 240 -threads 24 \  
-quality good -speed 4 -crf 15 -c:v libvpx-vp9 -c:a libopus \  
-pass 2 -y tos-3840x2160-50-60fps.webm
```

- Input file: tearsofsteel\_4k.mov (3840x1714, 6.76 GB)
- Output file: tos-3840x2160-50-60fps.webm 3840x2160, 1.56 GB)

Rate and review