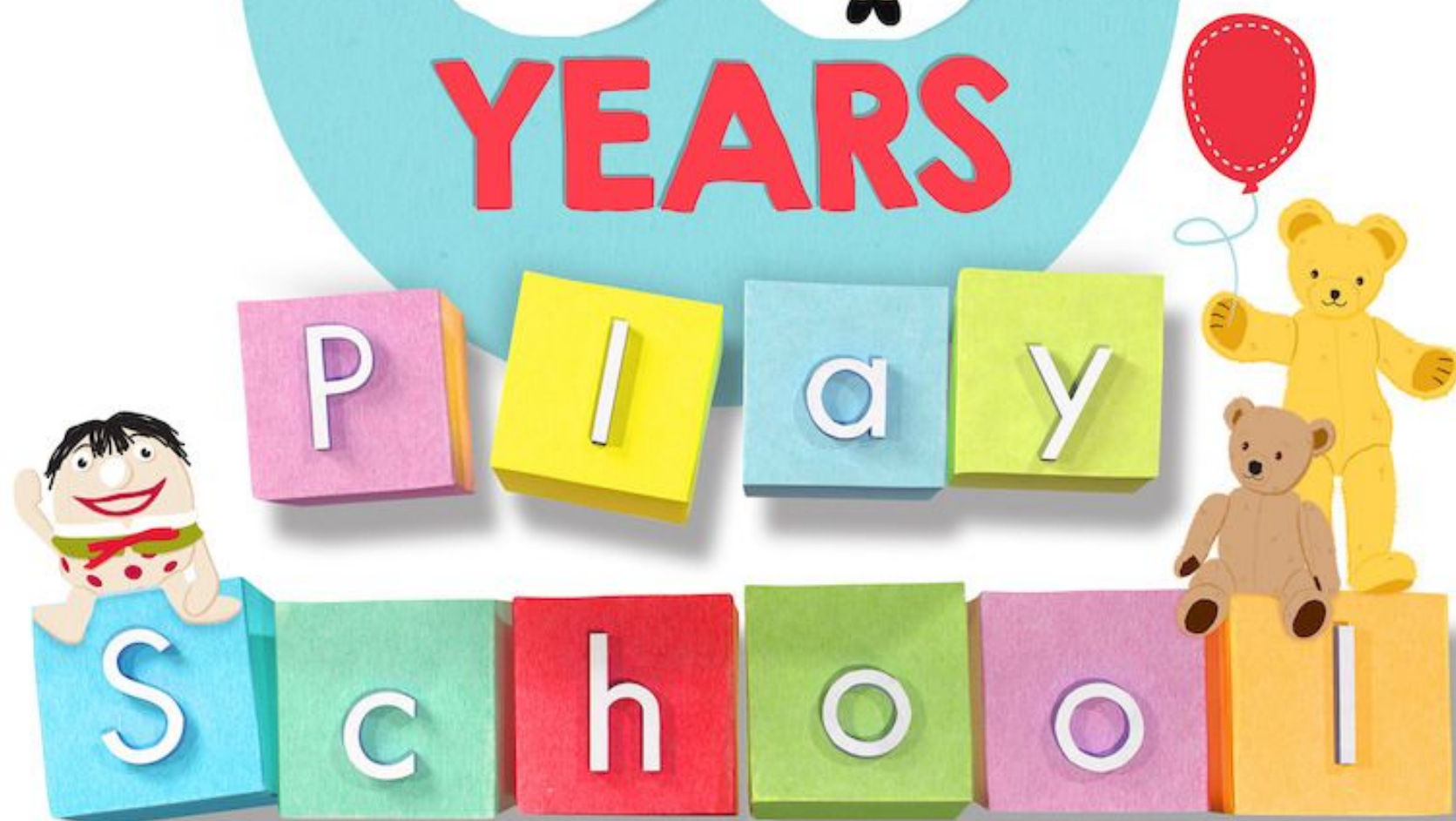


Video Transcoding at the ABC with microservices

Daphne Chong
@daphnechong



ABC





ABC
iVIEW  

media
transcoder

1. What is Transcoding?
2. Architecture
3. Autoscaling transcoders
4. Costs
5. Future roadmap

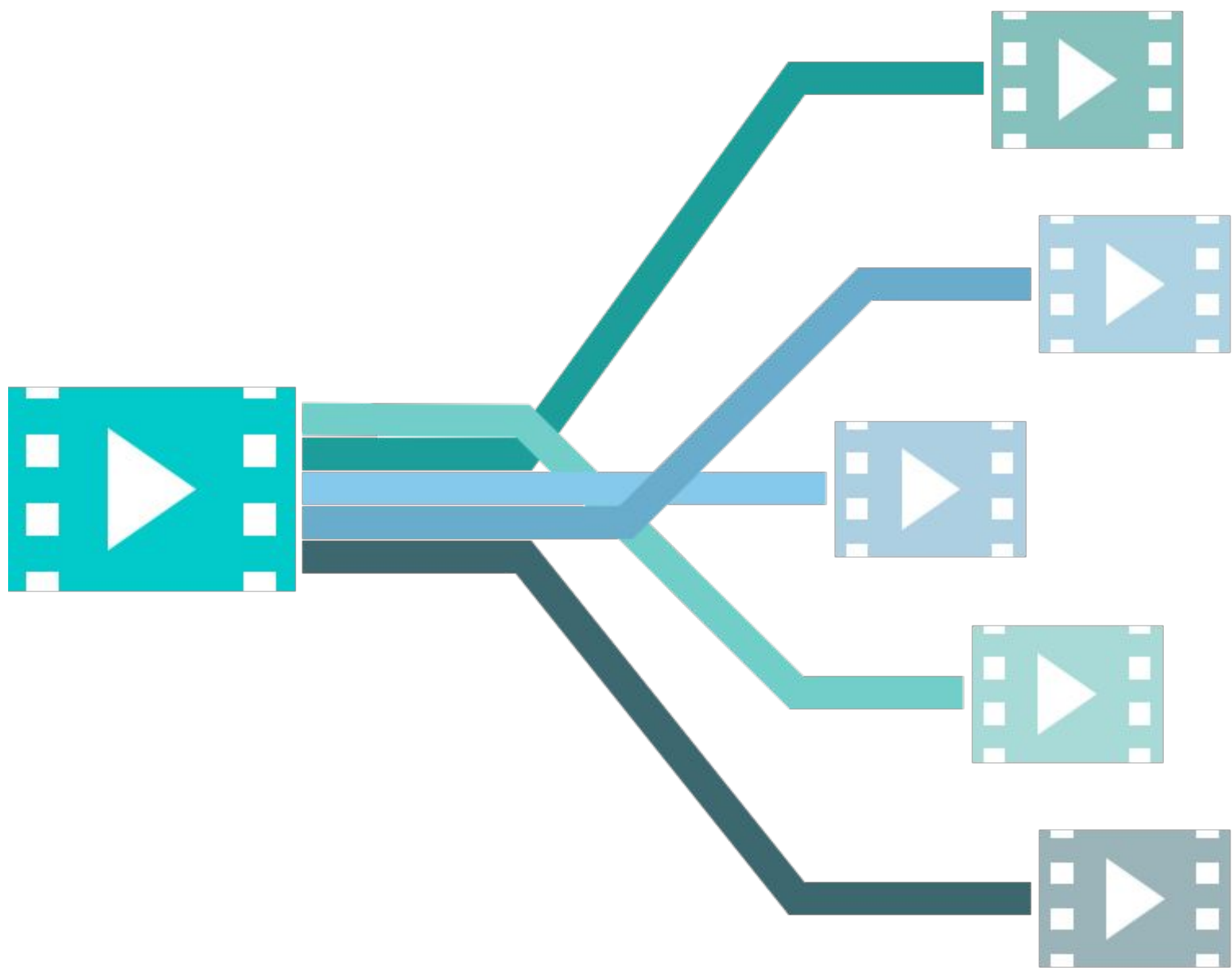
1. What is Transcoding?

2. Architecture

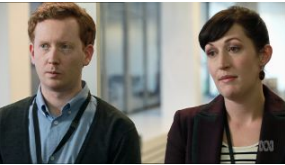
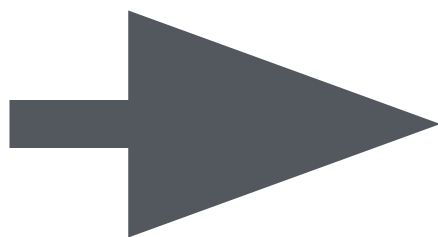
3. Autoscaling transcoders

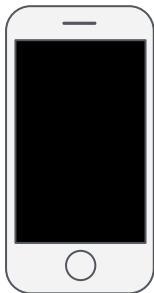
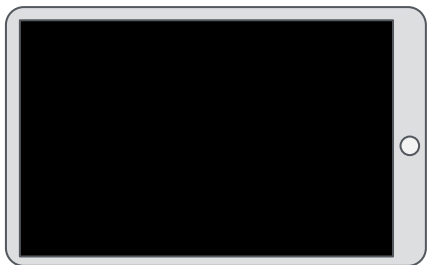
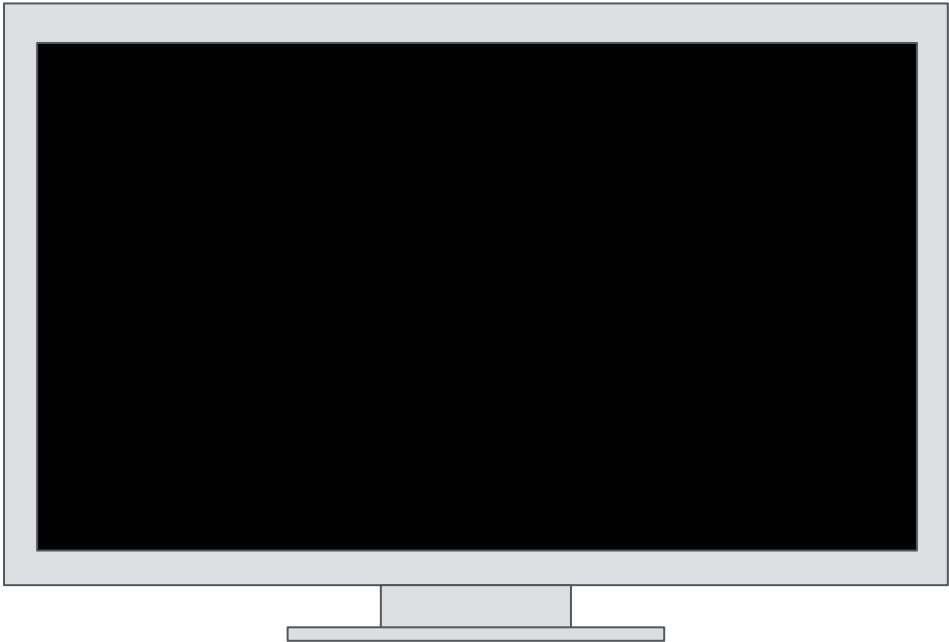
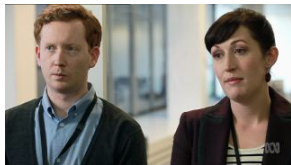
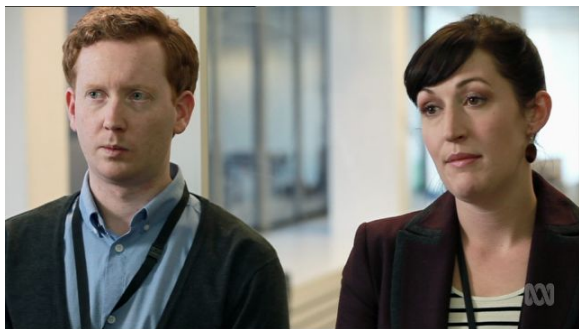
4. Costs

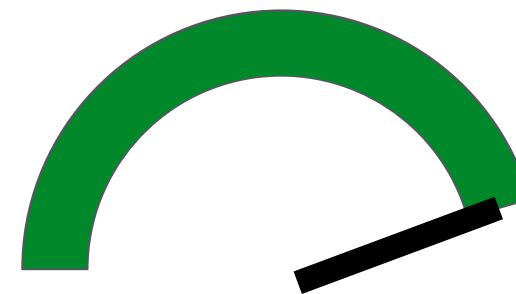
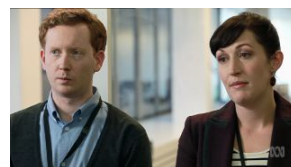
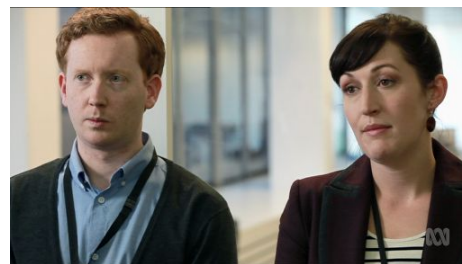
5. Future roadmap



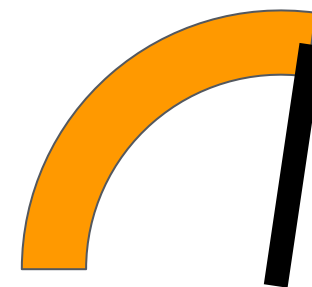
What changes
are we making?



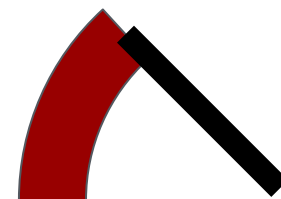




High speed



Medium speed



Low speed

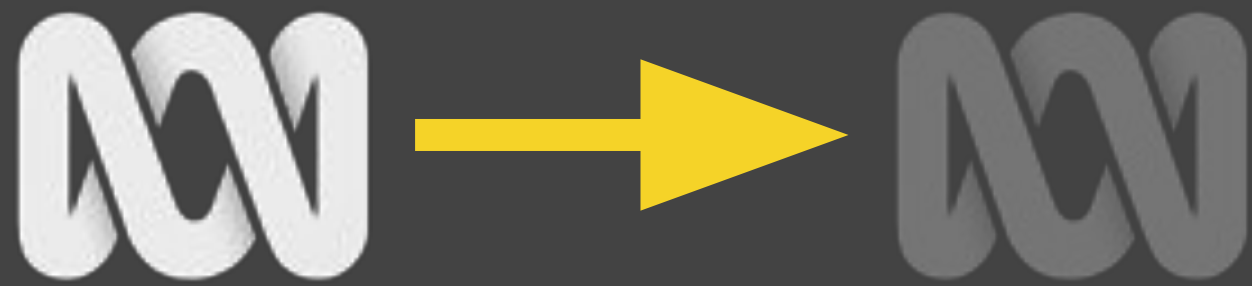
Standard Encoding

H.264 MP4

Transcoding Example



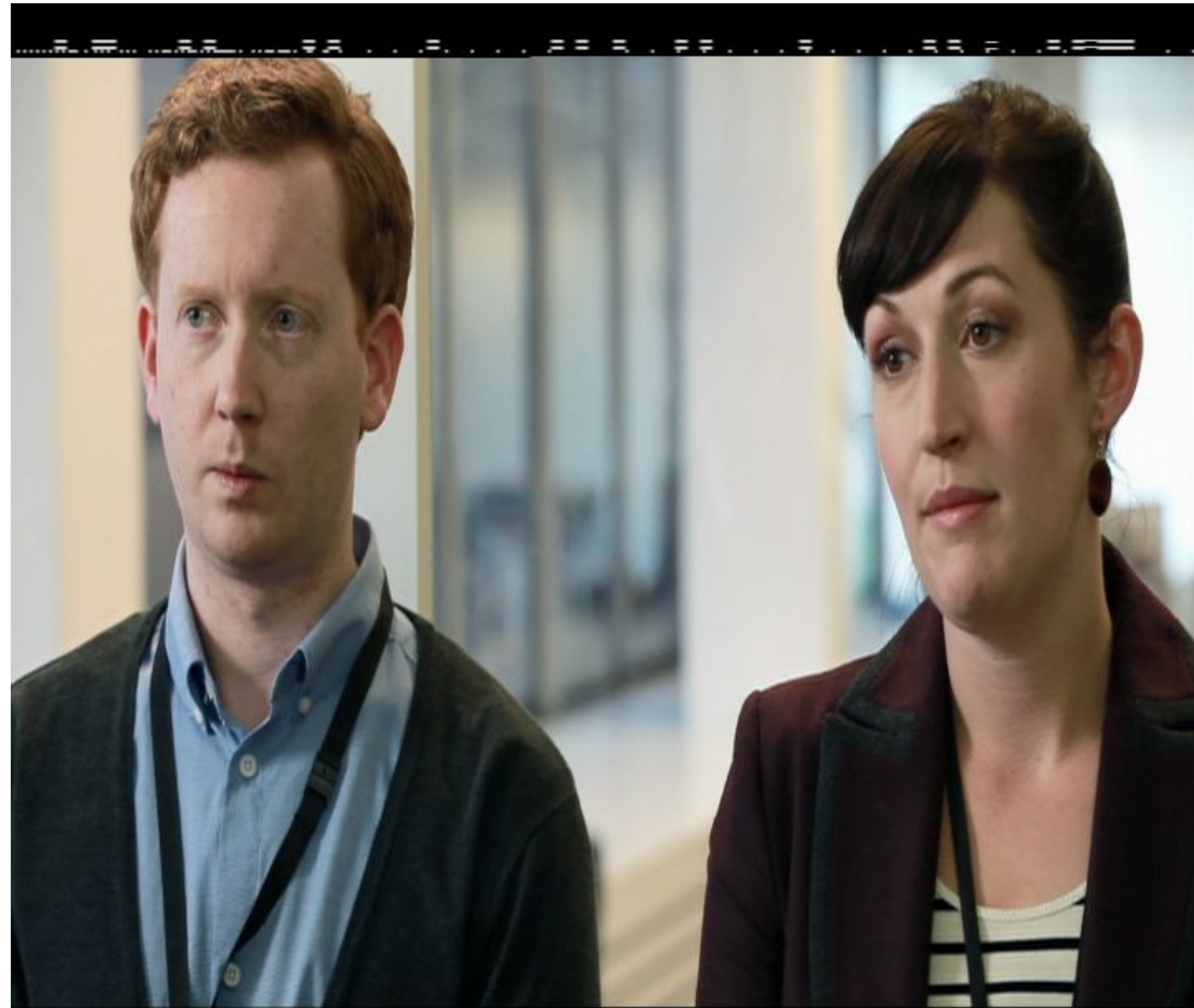
Step 1: Prepare watermark



Step 1: Prepare watermark



Step 1: Prepare watermark



Step 2: Input video (GXF)



Step 3: Asymmetrical cropping



Step 4: Scale and set aspect ratio



Step 5: Apply watermark



Step 6: Audio Normalization



Original file audio track



After audio normalization



Step 7: Combine changes



Step 8: Split into segments

That's a lot of
changes...

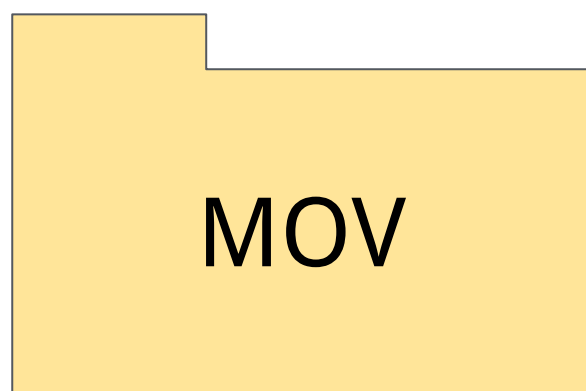
Transcoding Profiles



Profile:

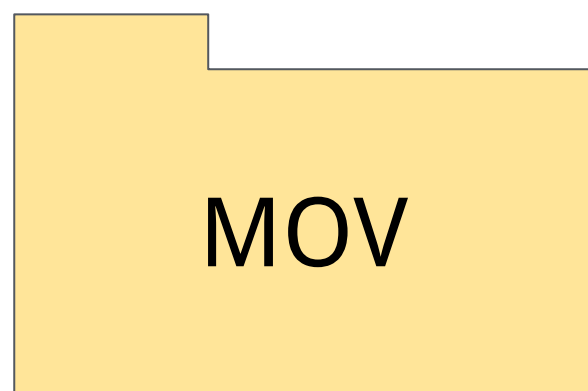
- Renditions (6)
- Crop
- Normalize audio
- Adjust ratio
- Add watermark
- GXF to MP4





Profile:

- Renditions (4)
- Adjust ratio
- ProRes MOV to MP4





FFmpeg

```
./ffmpeg -i input.gxf output.mp4
```

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovr1],  
[main][ovr1]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[main  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovr1],  
[main][ovr1]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovr1],  
[main][ovr1]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

```
./ffmpeg -i INPUT -c:v libx264 -filter_complex  
"[0:v]crop=720:576:0:32,yadif,scale=1024x576,setdar=16/9[ma  
in], [1:v]scale=54:-1[ovrl],  
[main][ovrl]overlay=main_w-overlay_w-47:main_h-overlay_h-44  
" -crf 22 -maxrate 1404k -bufsize 1500k -pix_fmt yuv420p  
-profile:v high -level 4.0 -refs 3 -r 25 -g 50 -c:a  
libfdk_aac -b:a 96k -filter_complex "[0:a:0][0:a:1]  
amerge=inputs=2,  
dynaudnorm=f=500:g=31:p=0.82:m=30.0:r=0.0:s=0.0 [aout]" -ar  
44100 -ac 2 -map 0:0 -map "[aout]" -movflags +faststart  
-threads 0 -y OUTPUT
```

In reality, it looks like this

1. What is Transcoding?

2. Architecture

3. Autoscaling transcoders

4. Costs

5. Future roadmap

1. What is Transcoding?

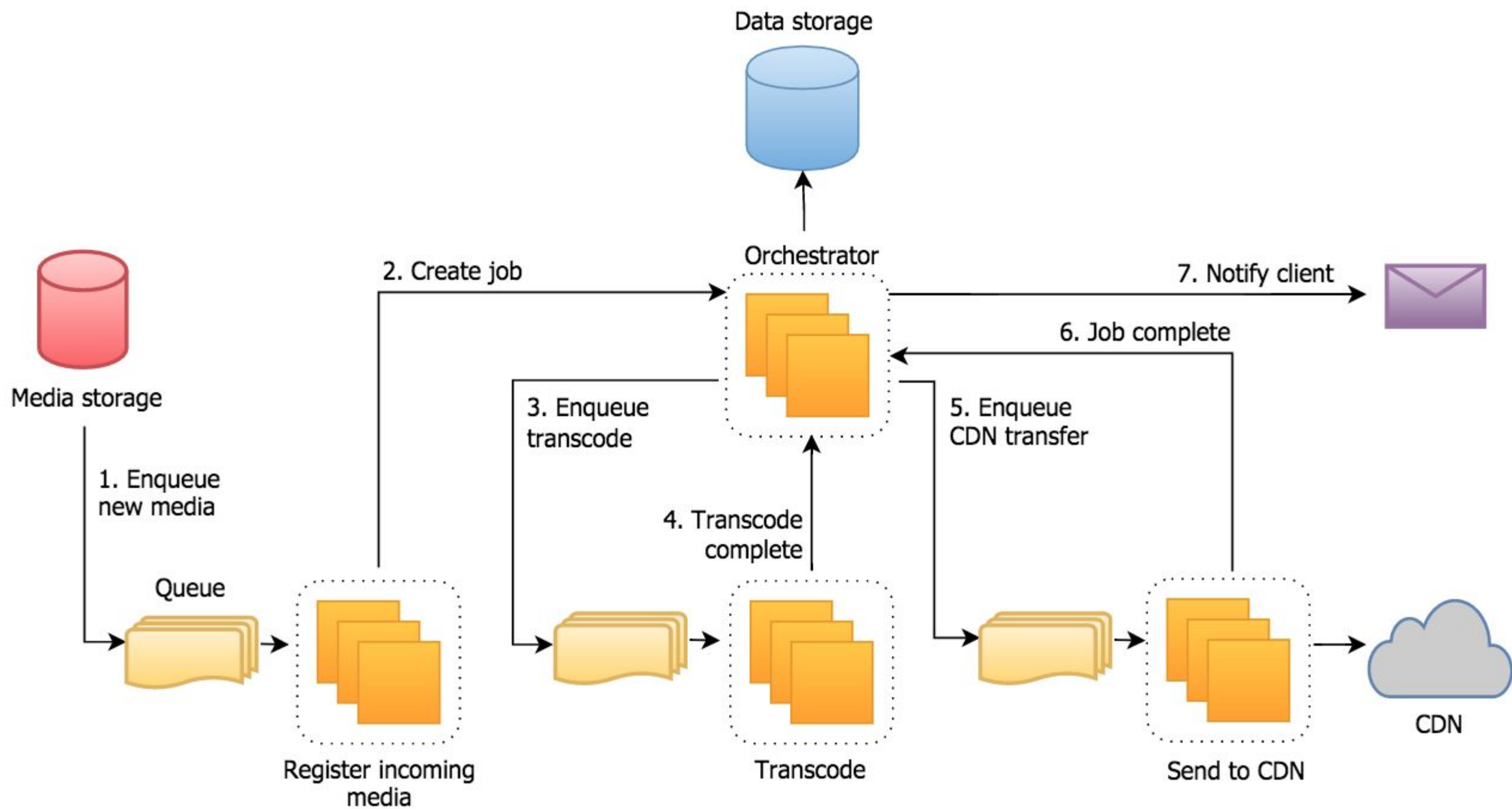
2. Architecture

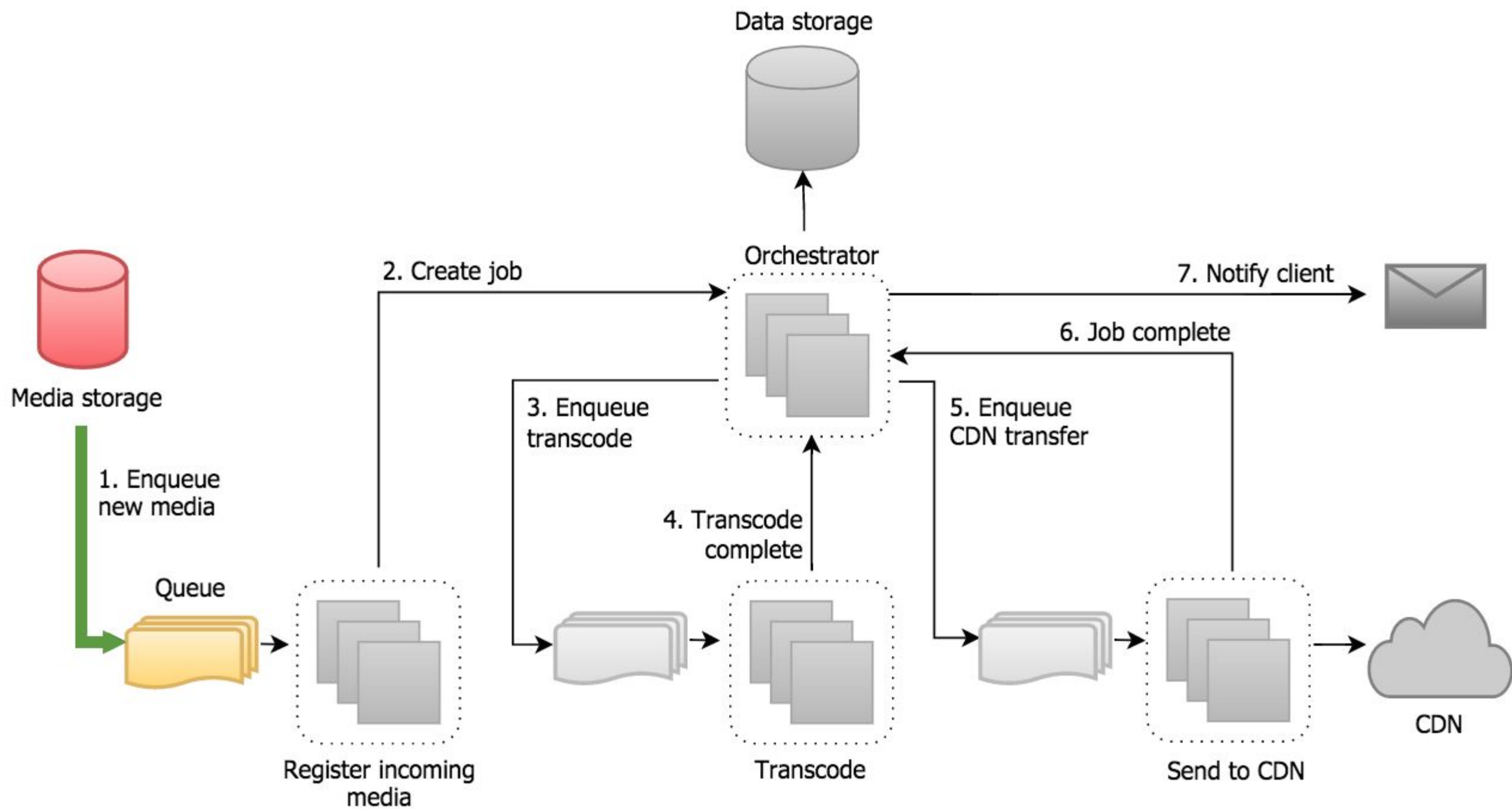
3. Autoscaling transcoders

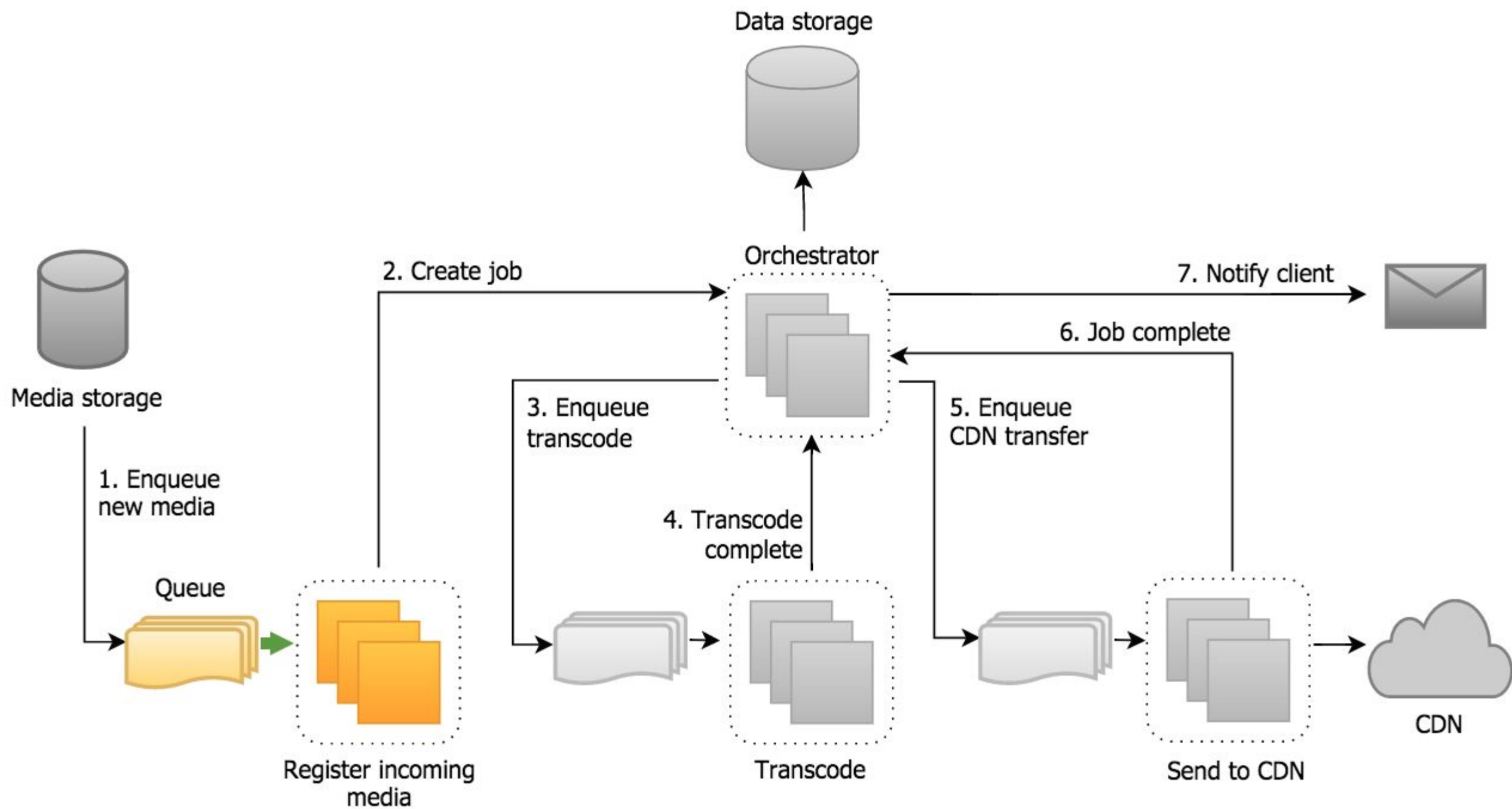
4. Costs

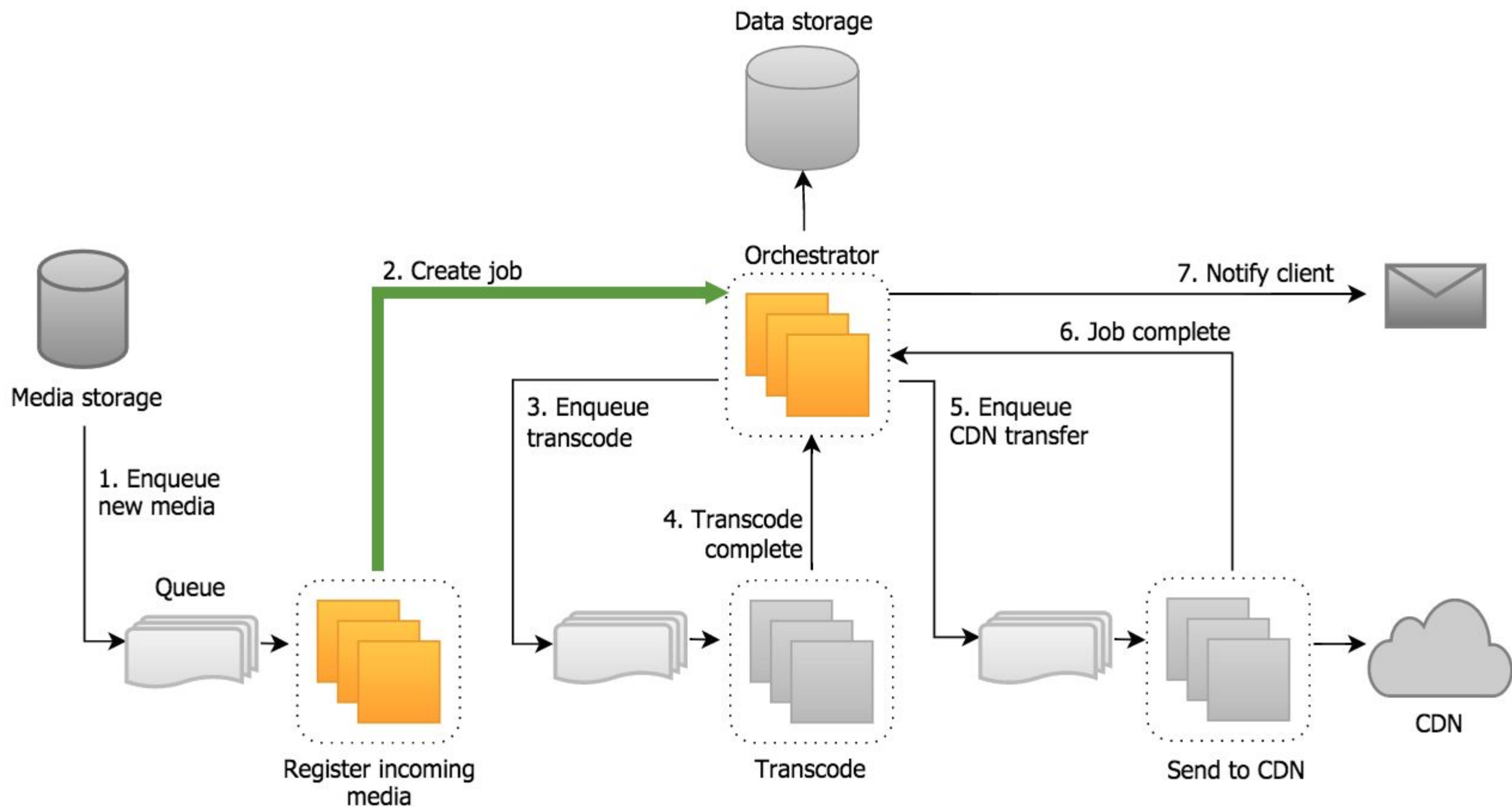
5. Future roadmap

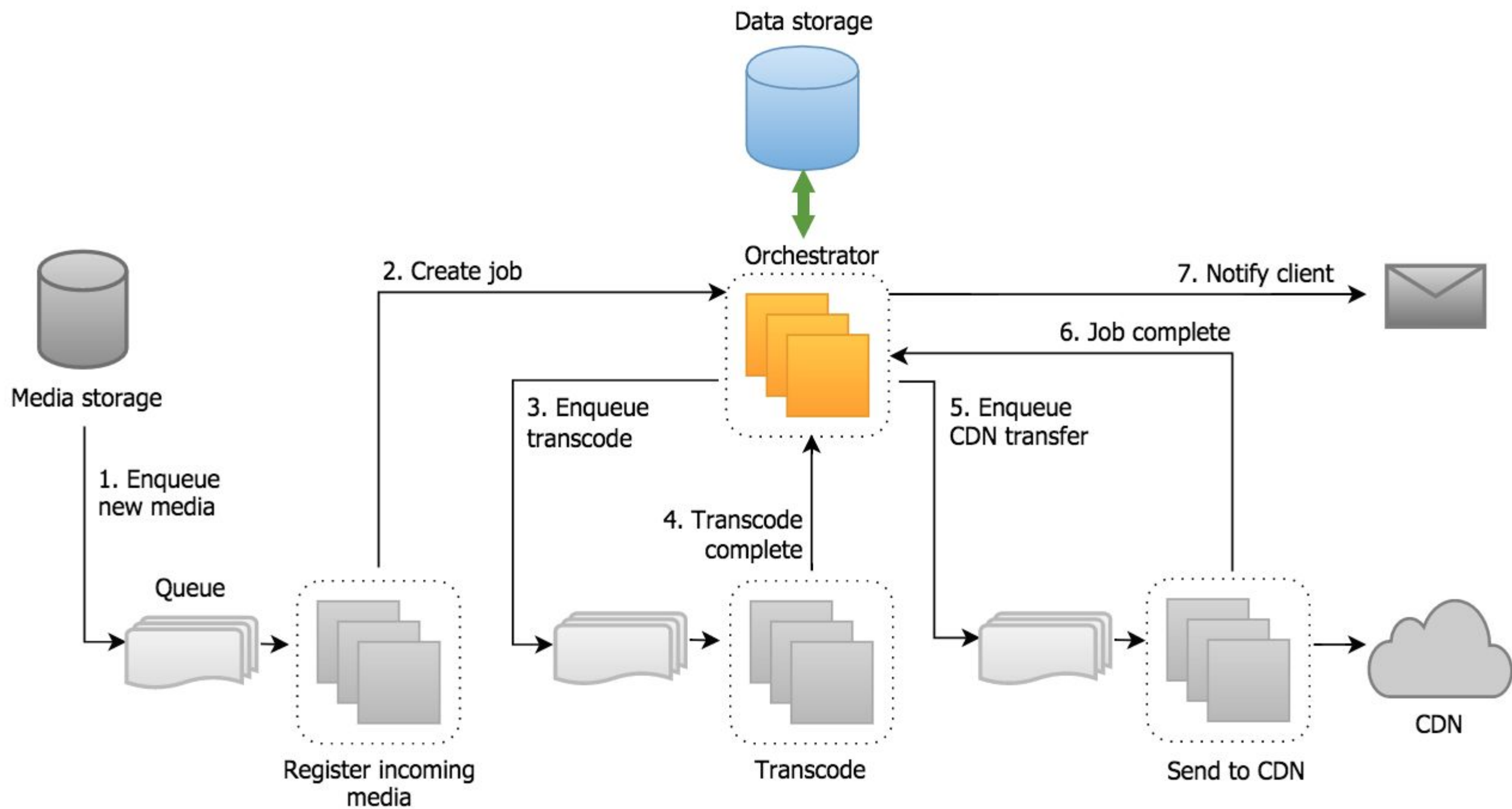
Workflow

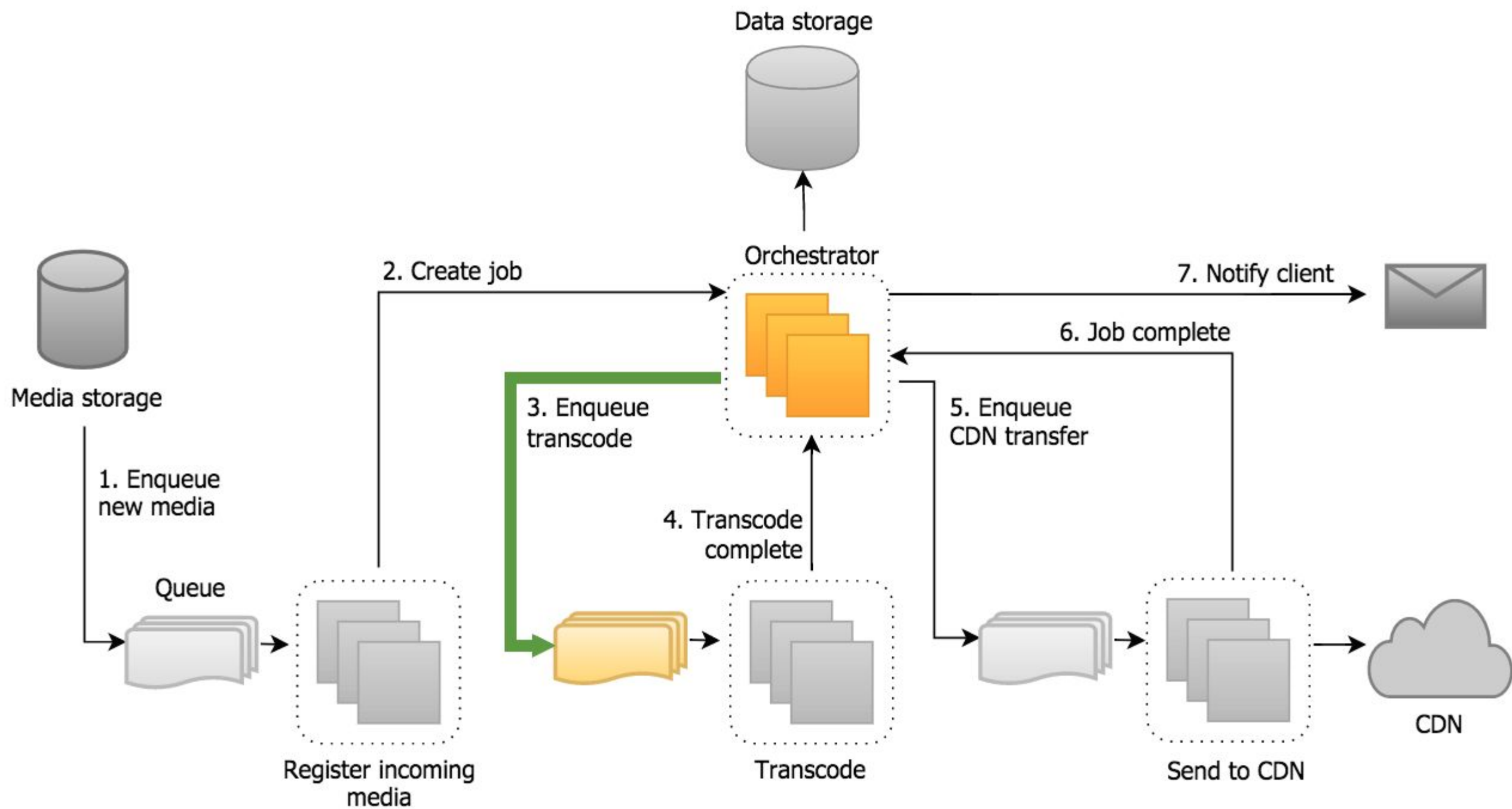


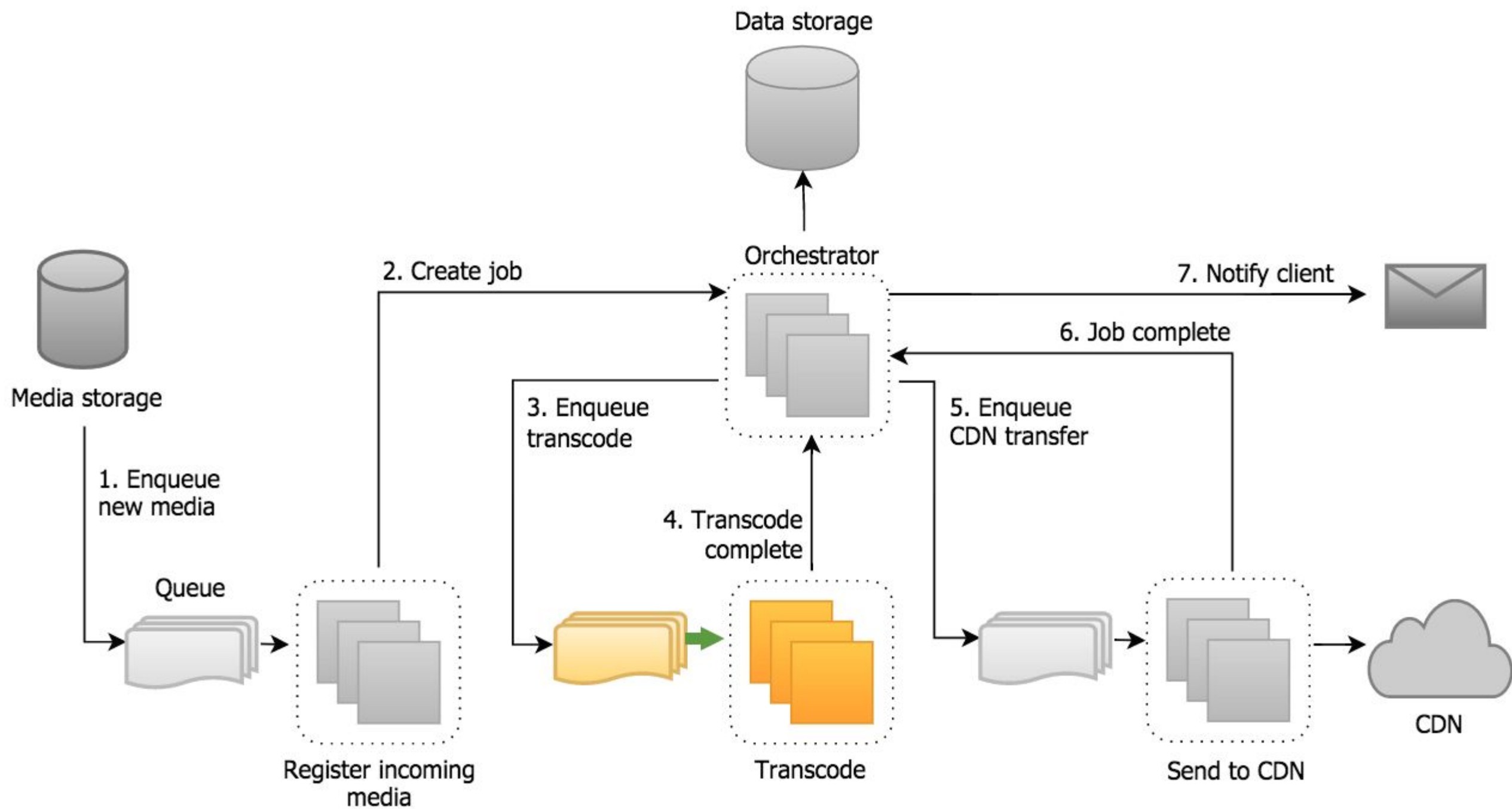


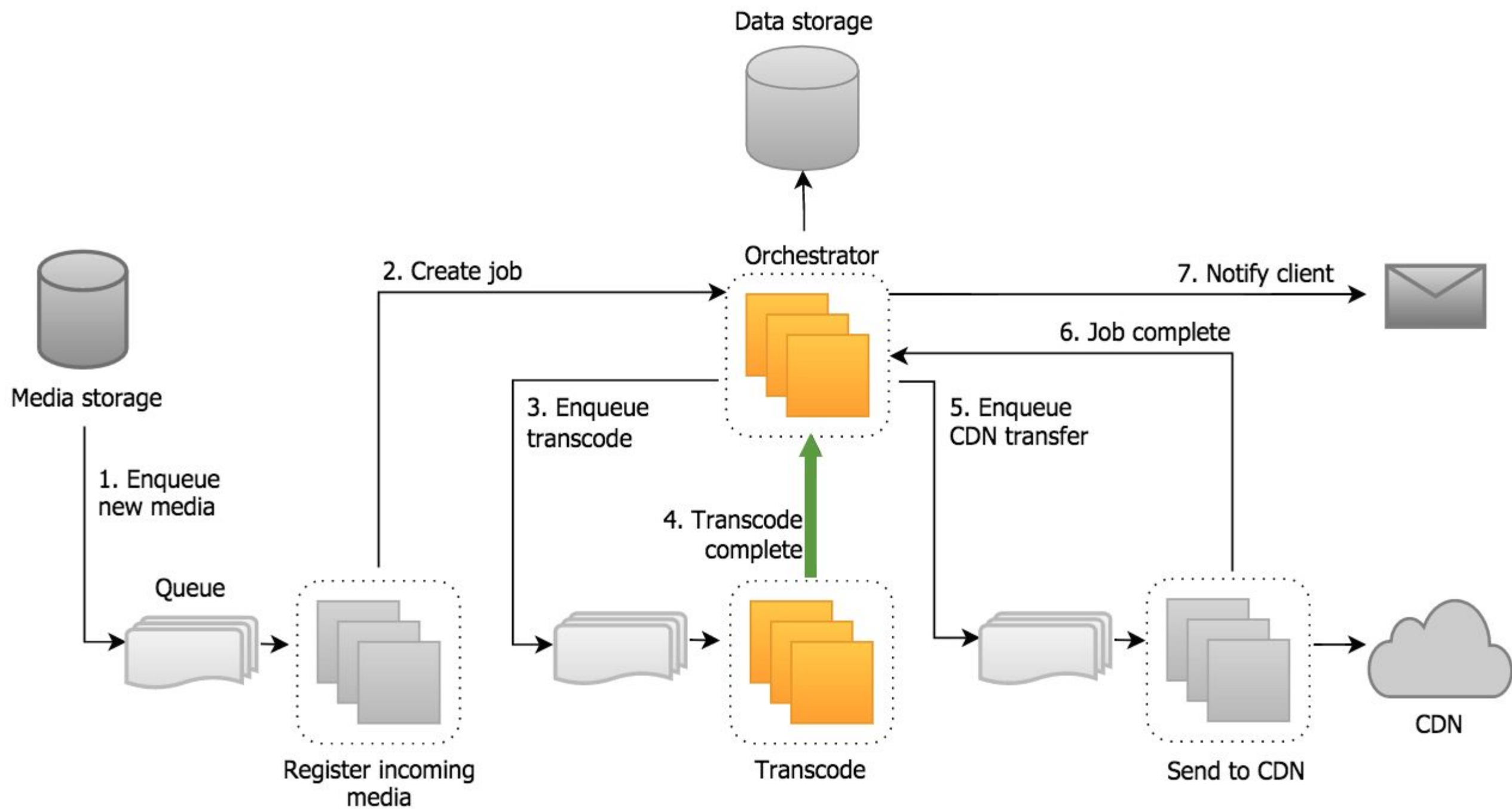


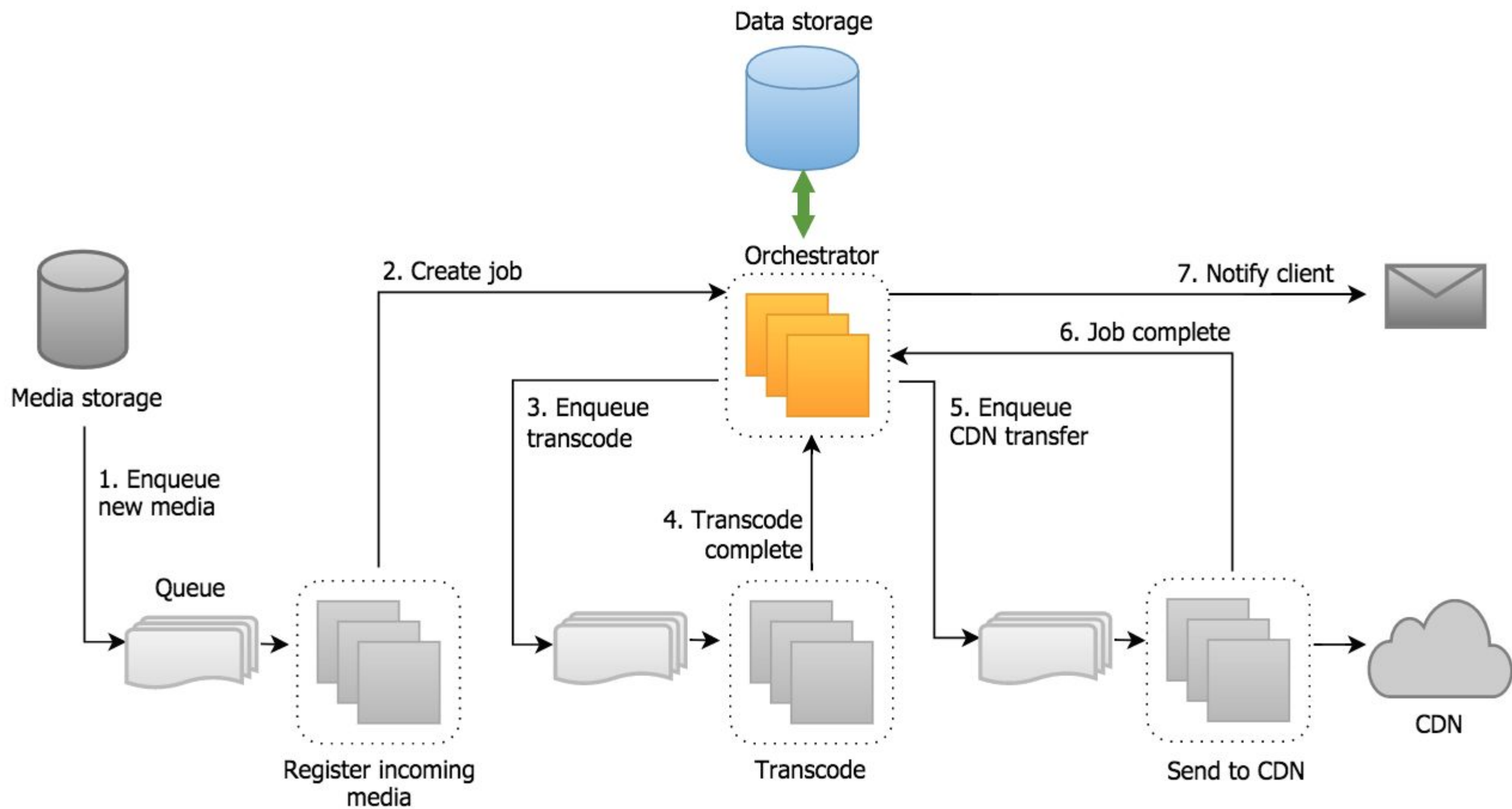


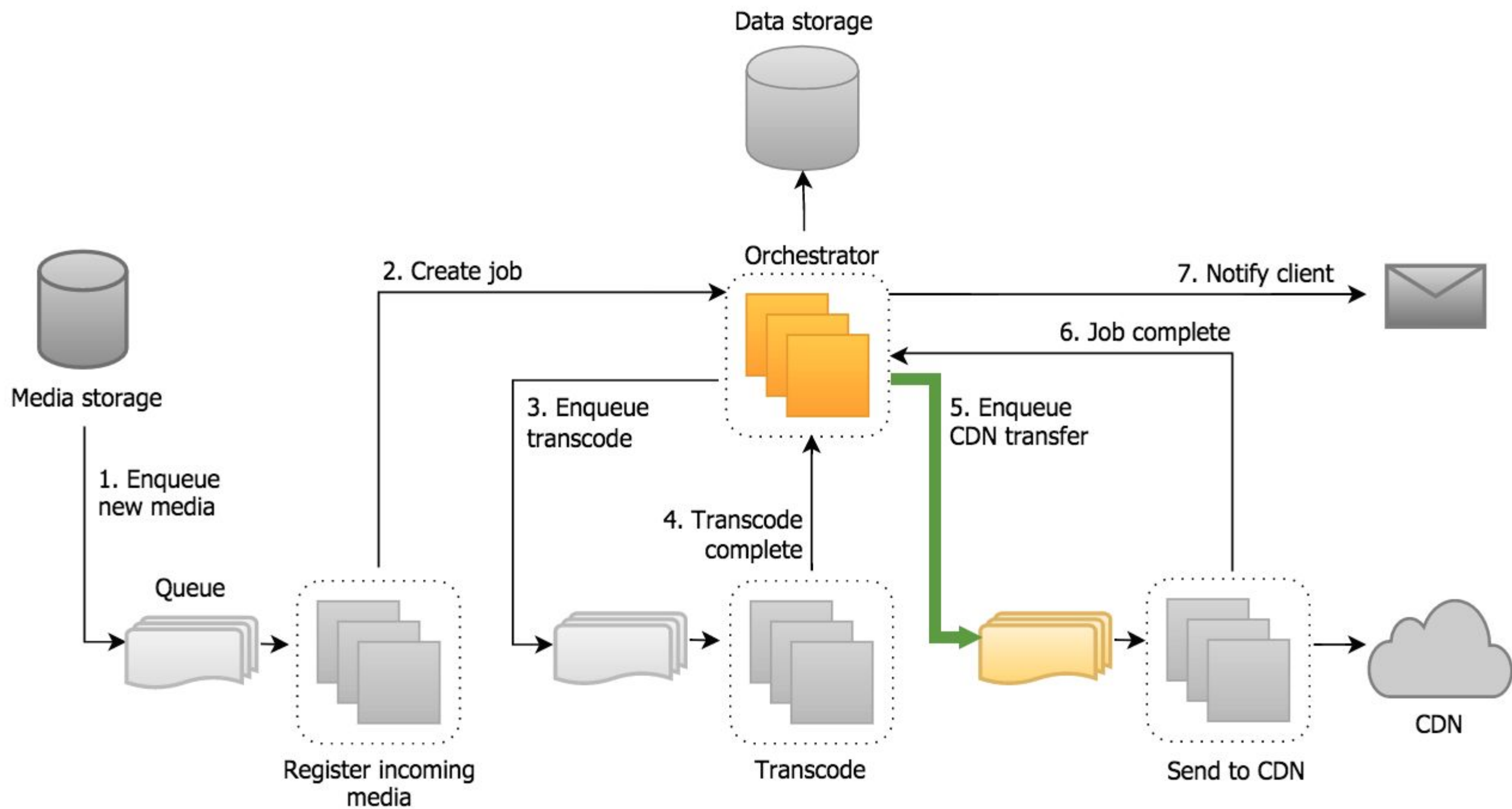


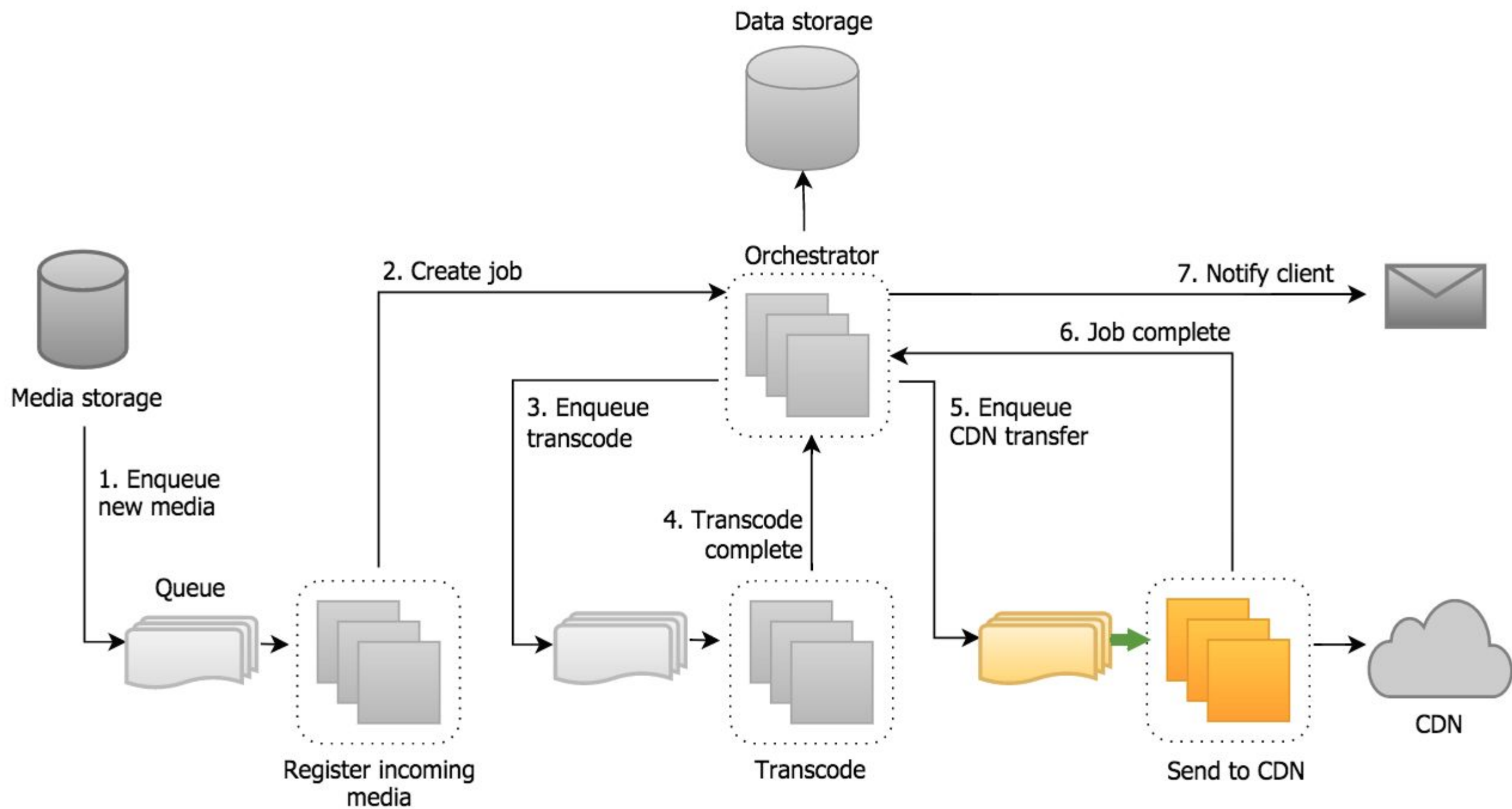


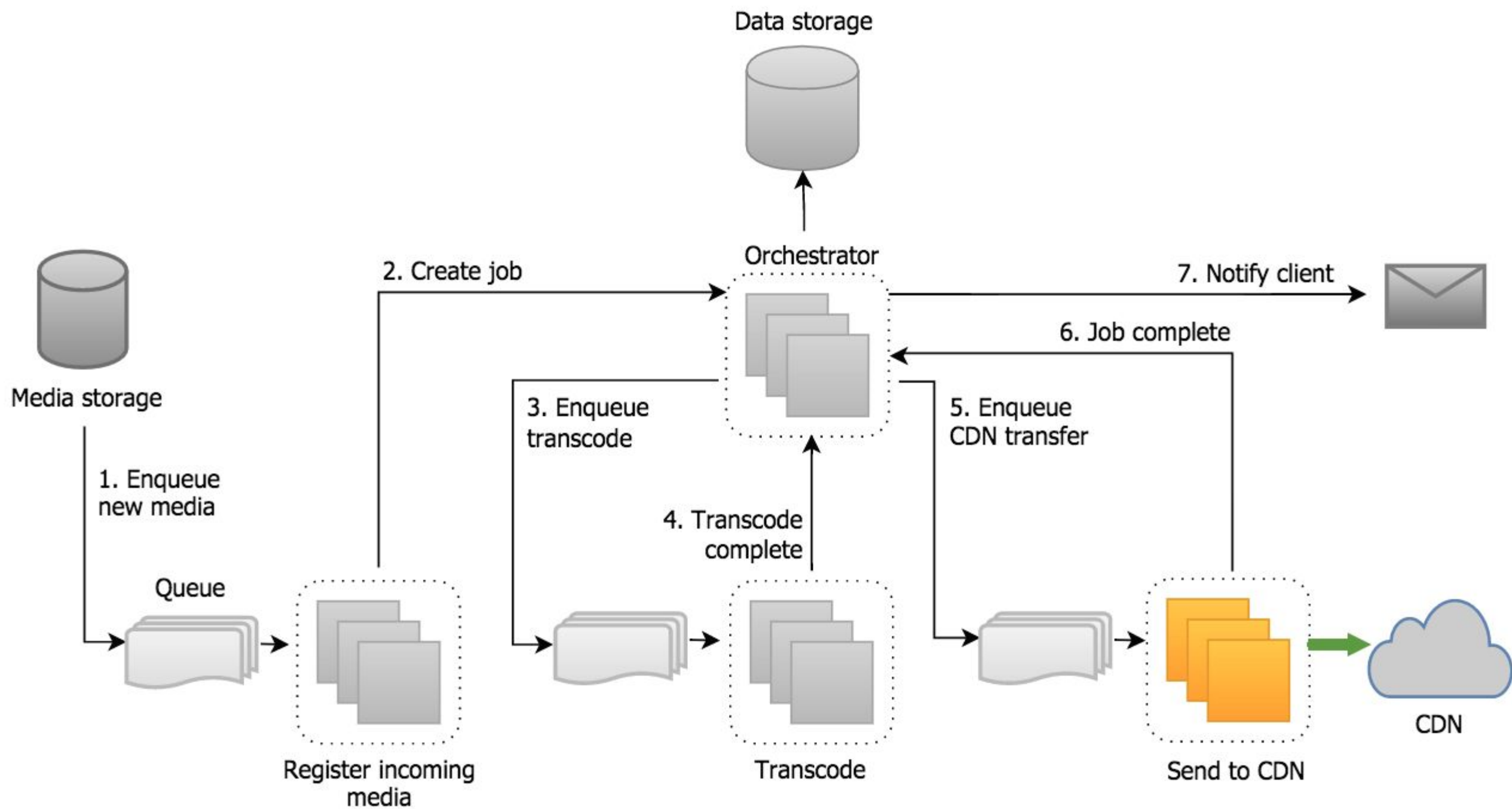


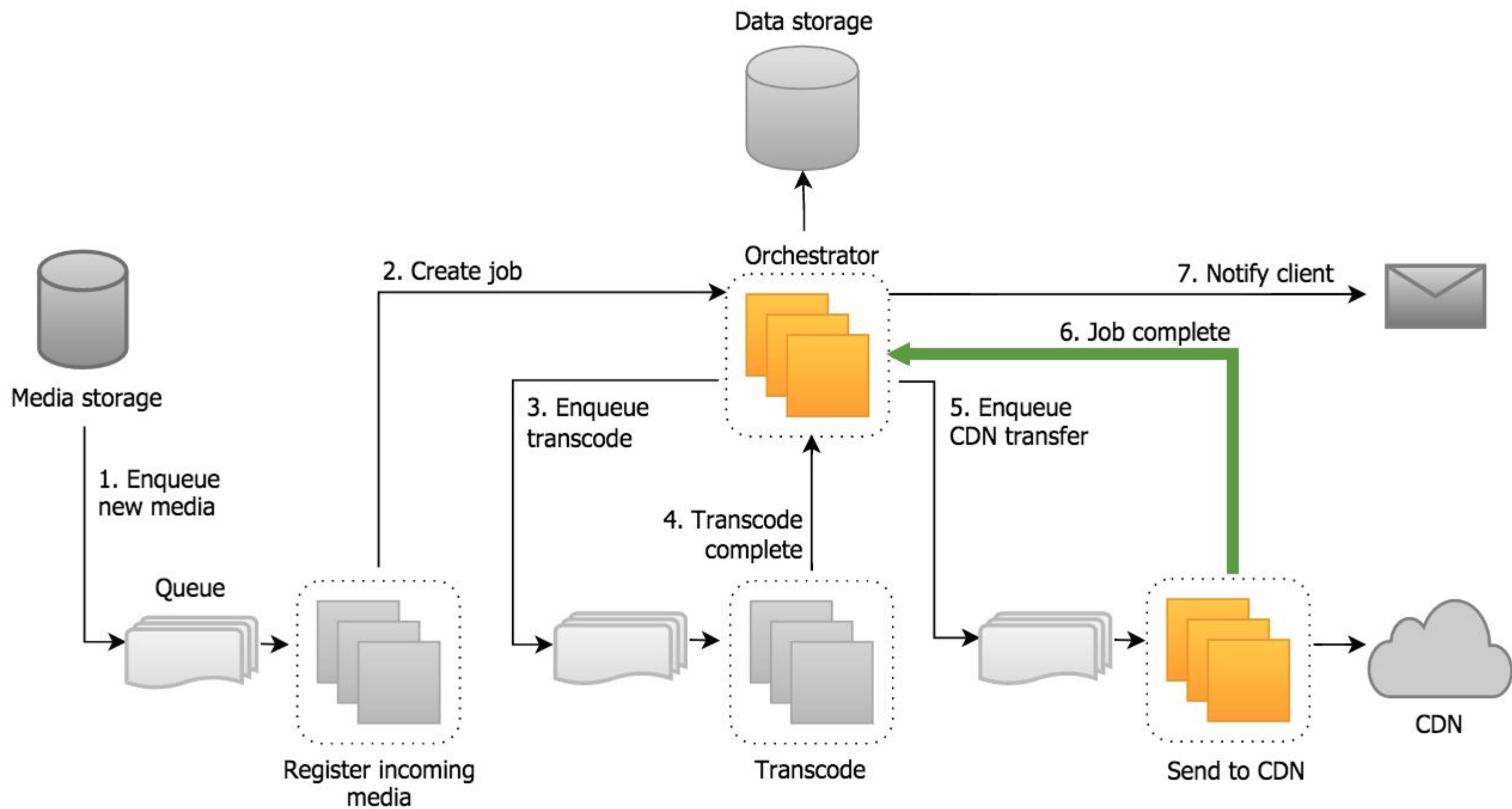


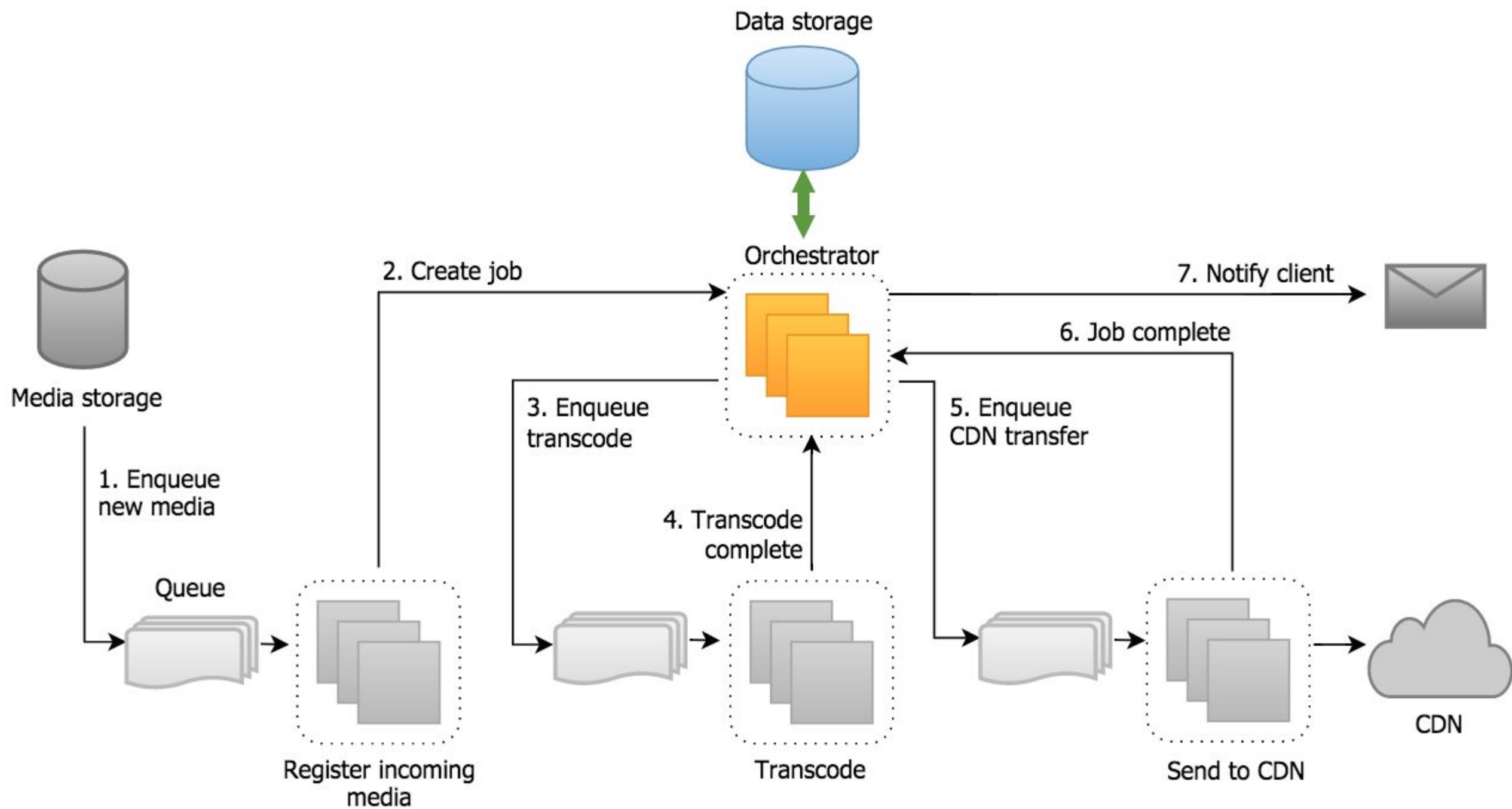


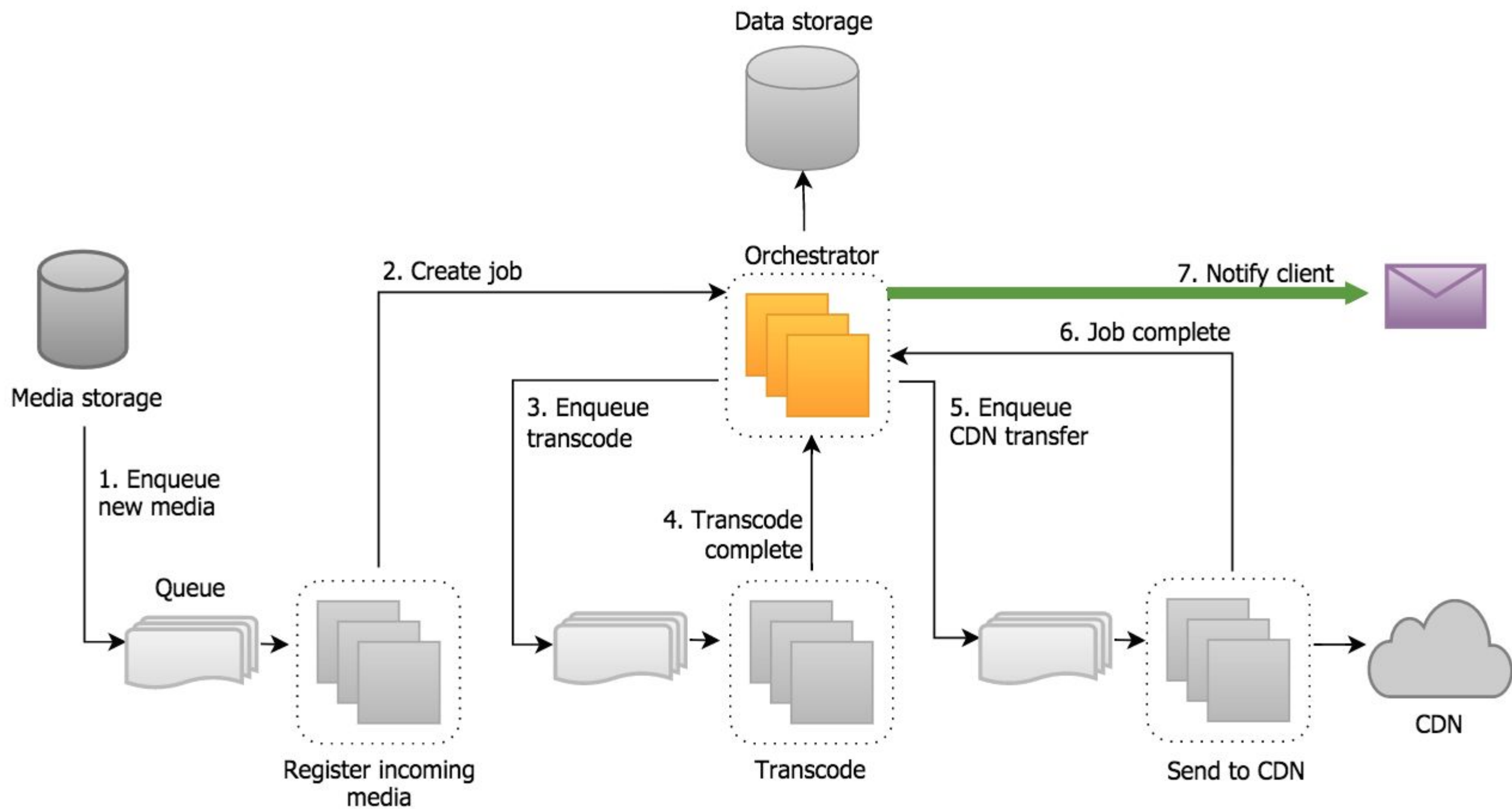








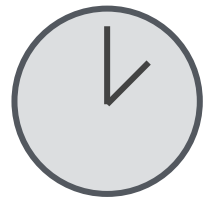




8 minutes 27 seconds

Time to produce rendition

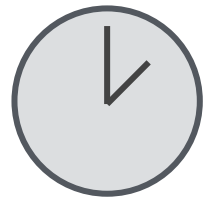
Original
length



60 min



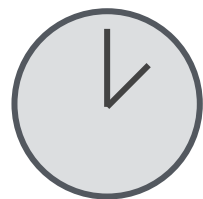
Lowest
Quality



7 min



Highest
Quality



42 min

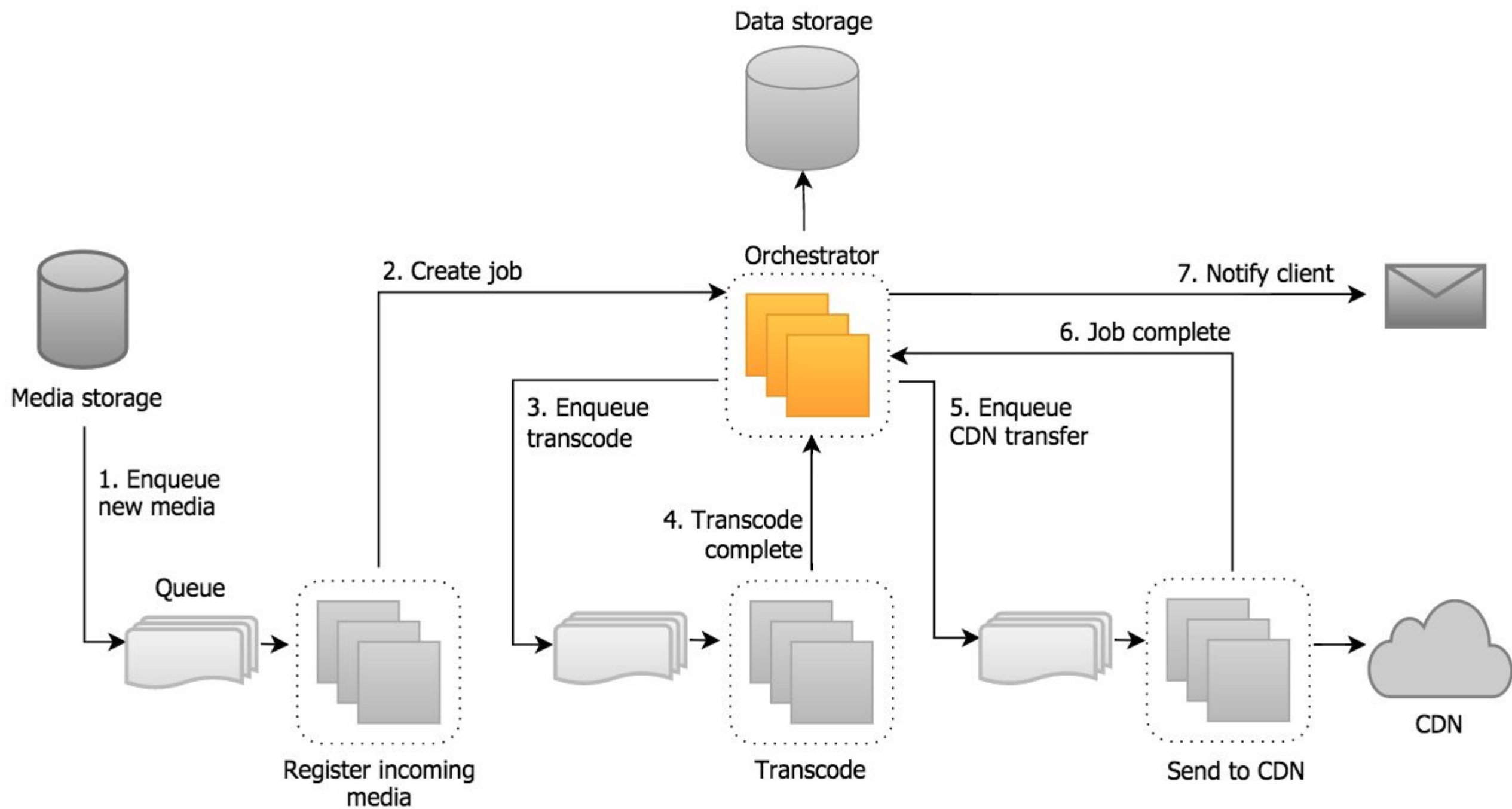


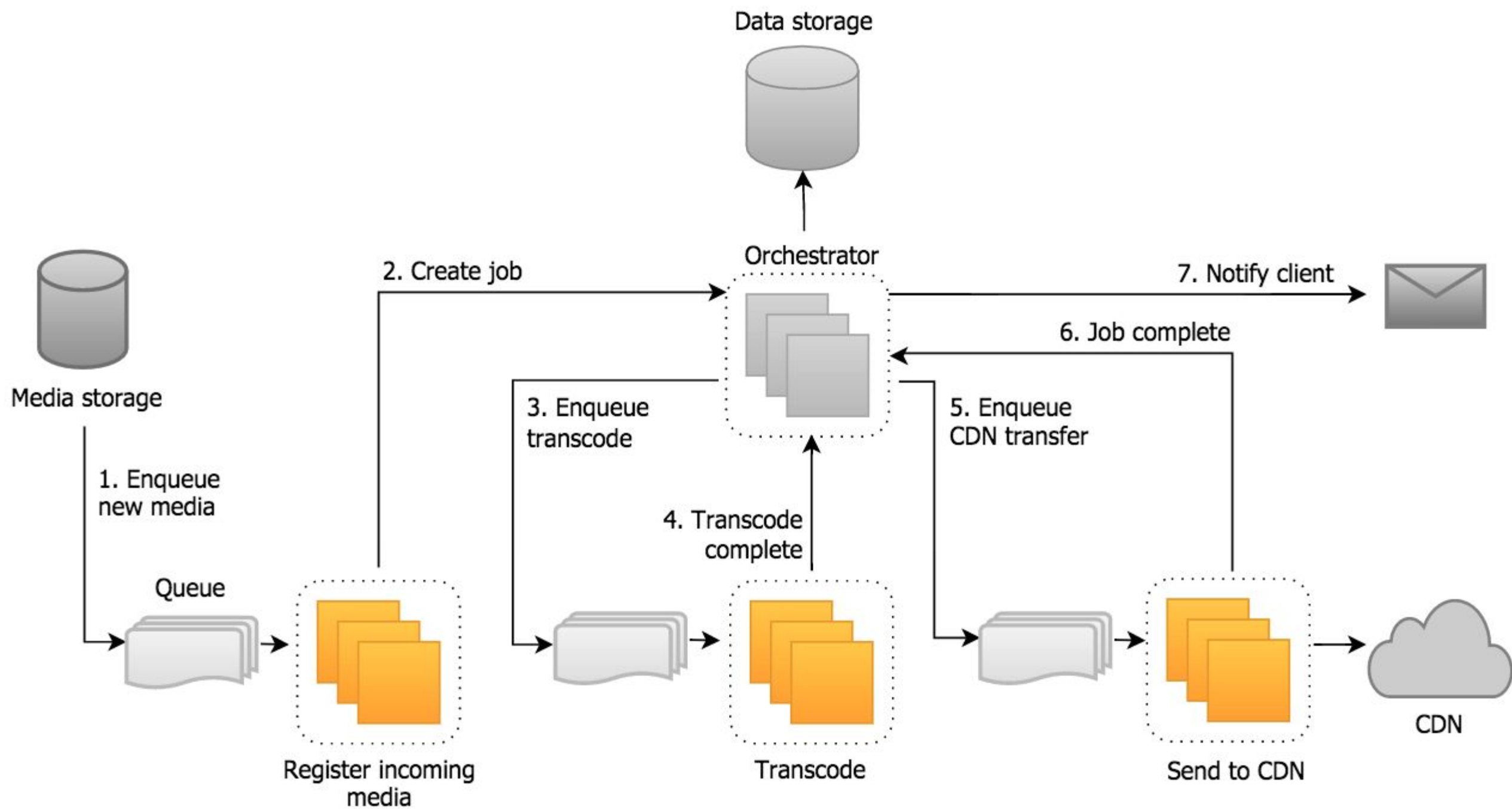
AWS & Languages

Languages:

Node.js

Go





AWS Infrastructure:

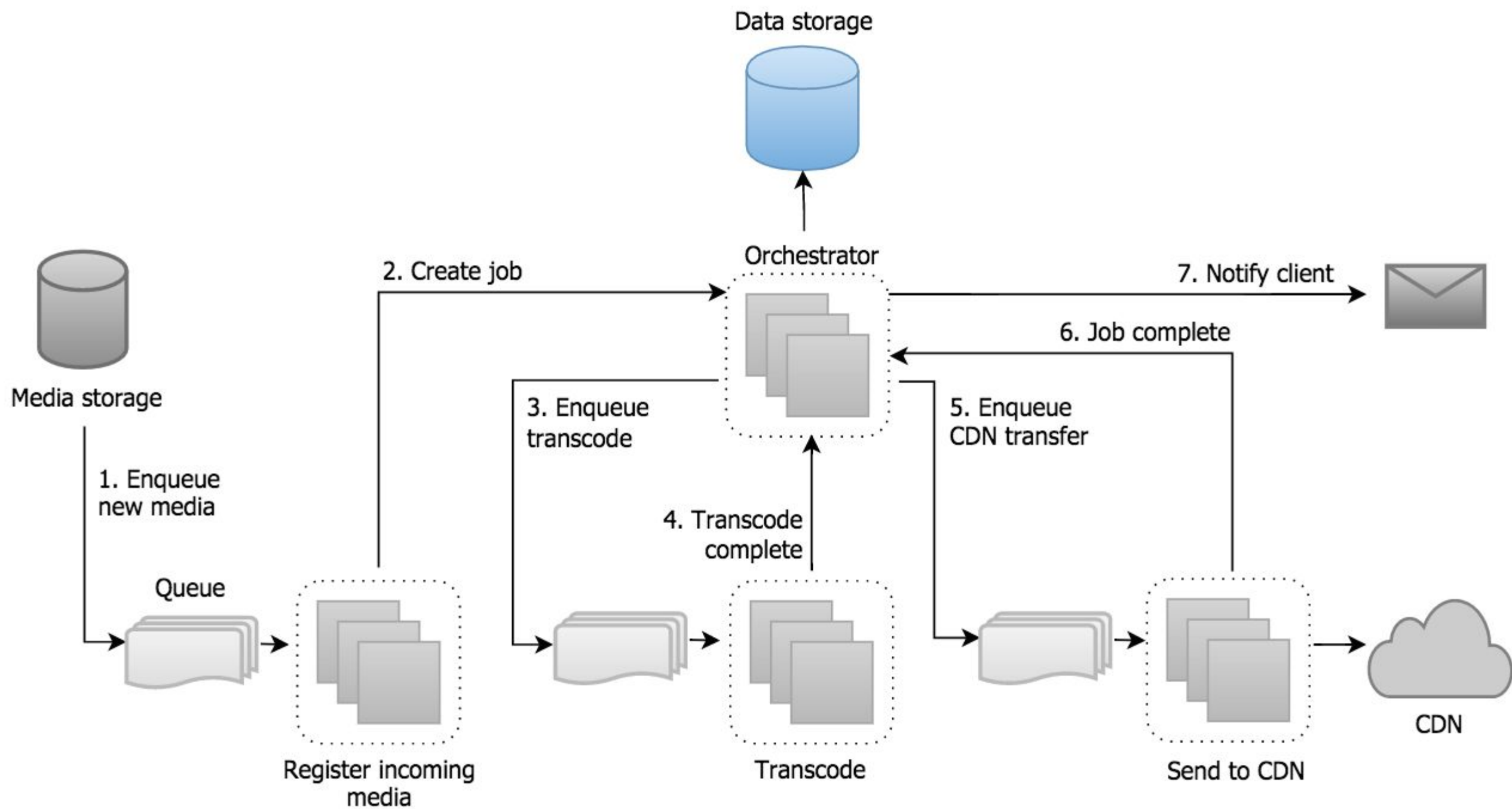
RDS - state storage

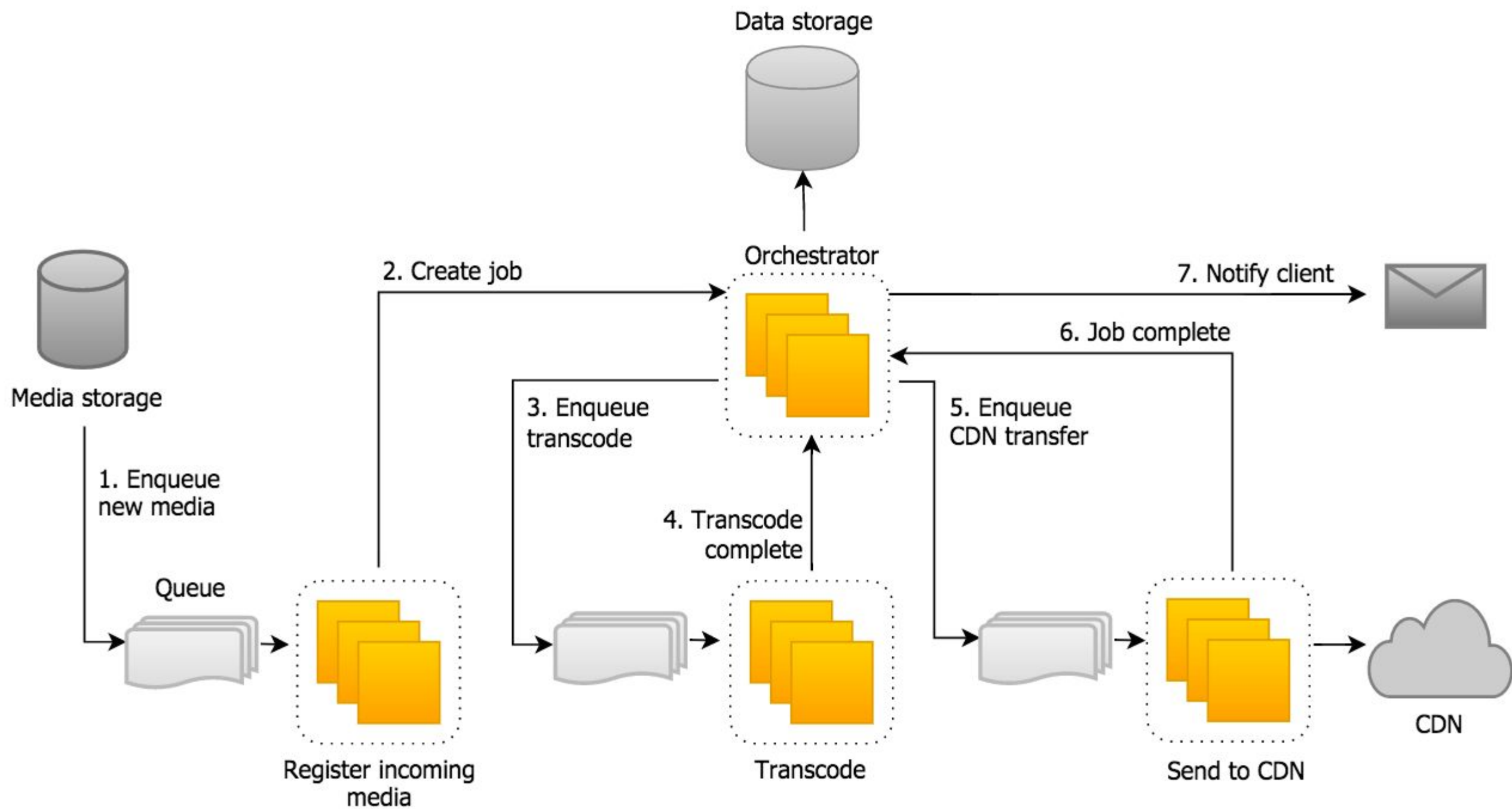
EC2 - instances

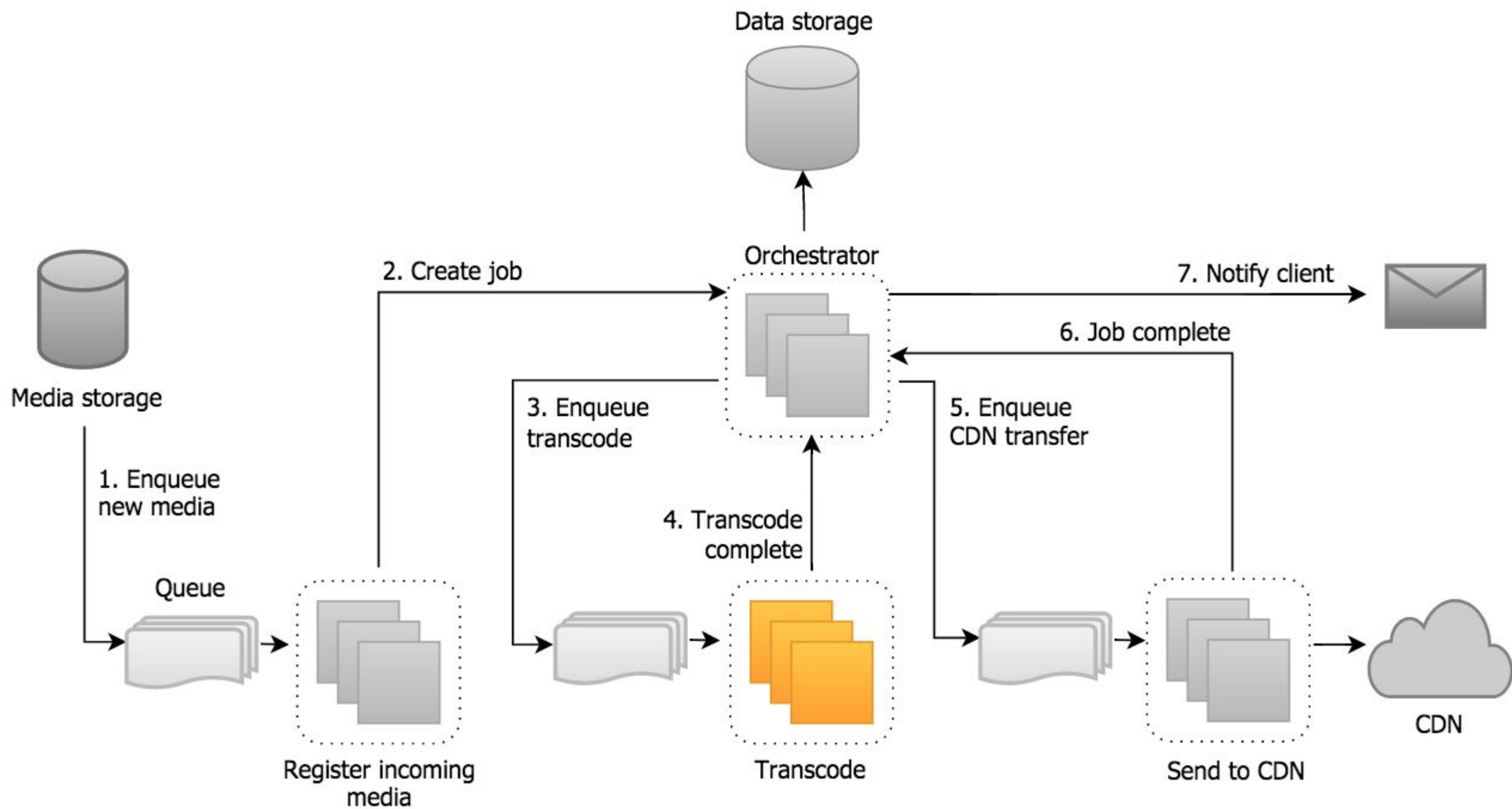
SQS - messaging

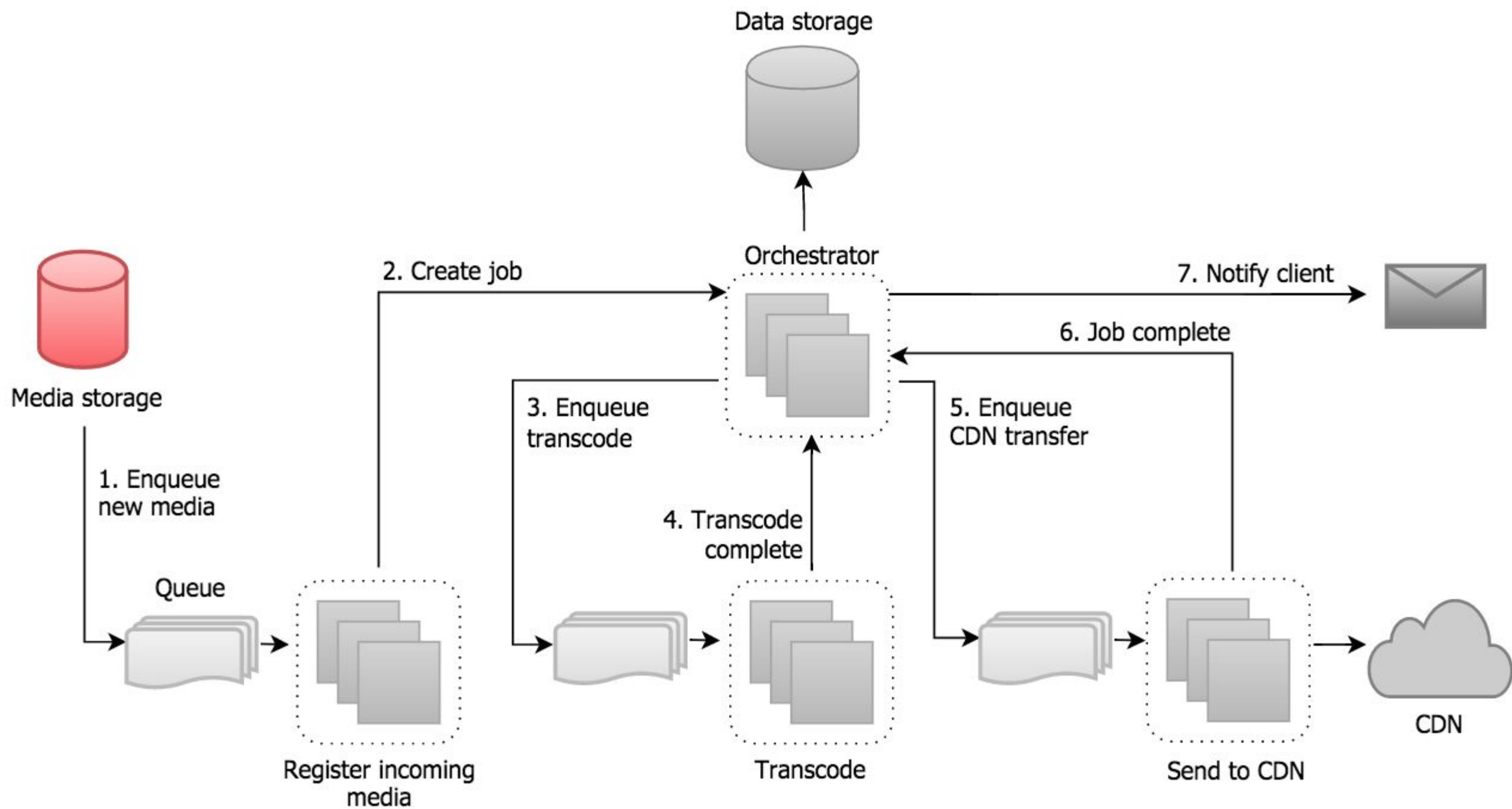
S3 - file storage

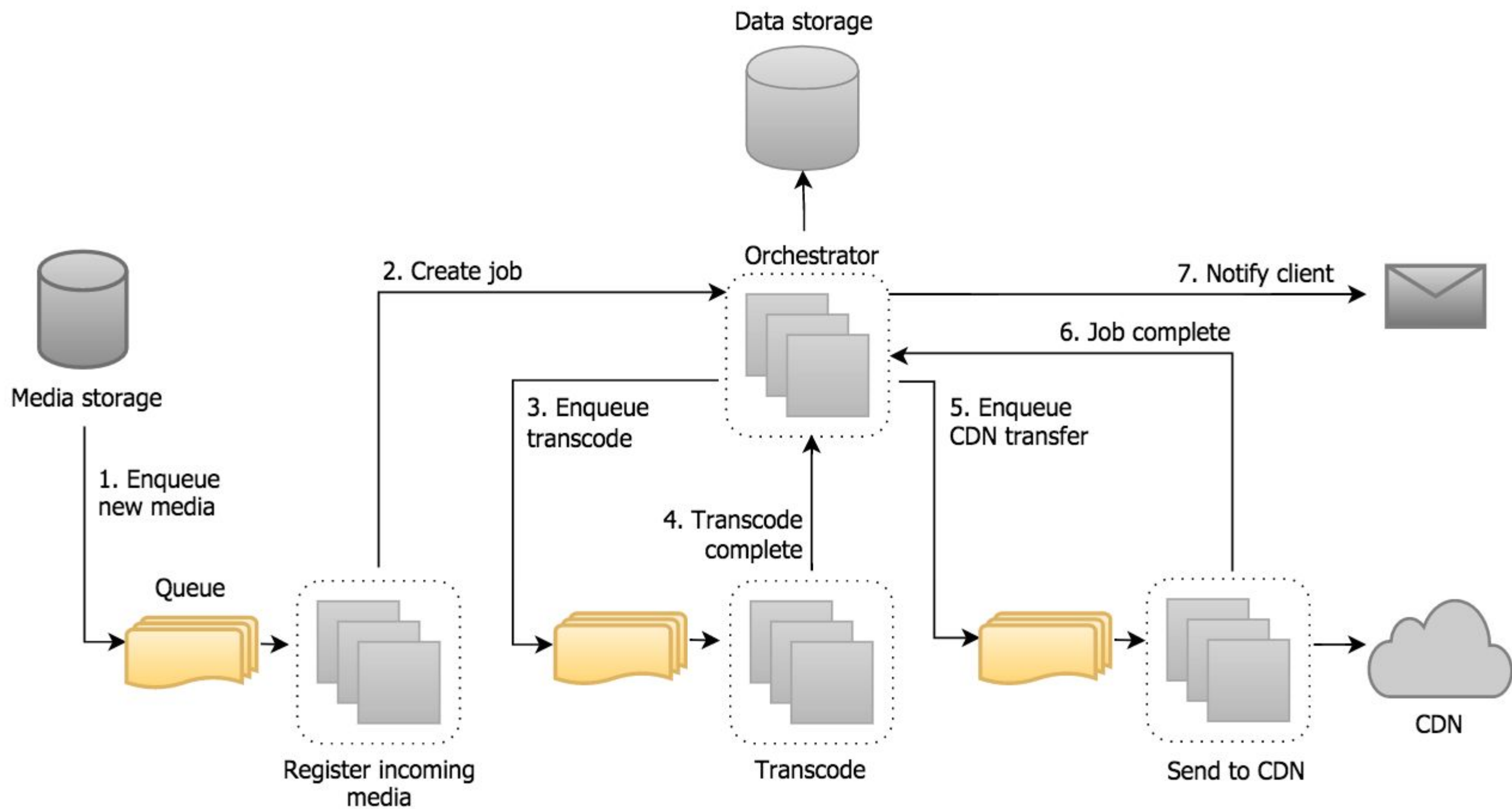
SNS - notifications











Advantages

Deployments

Independent scaling

Really easy to expand

Clear responsibilities

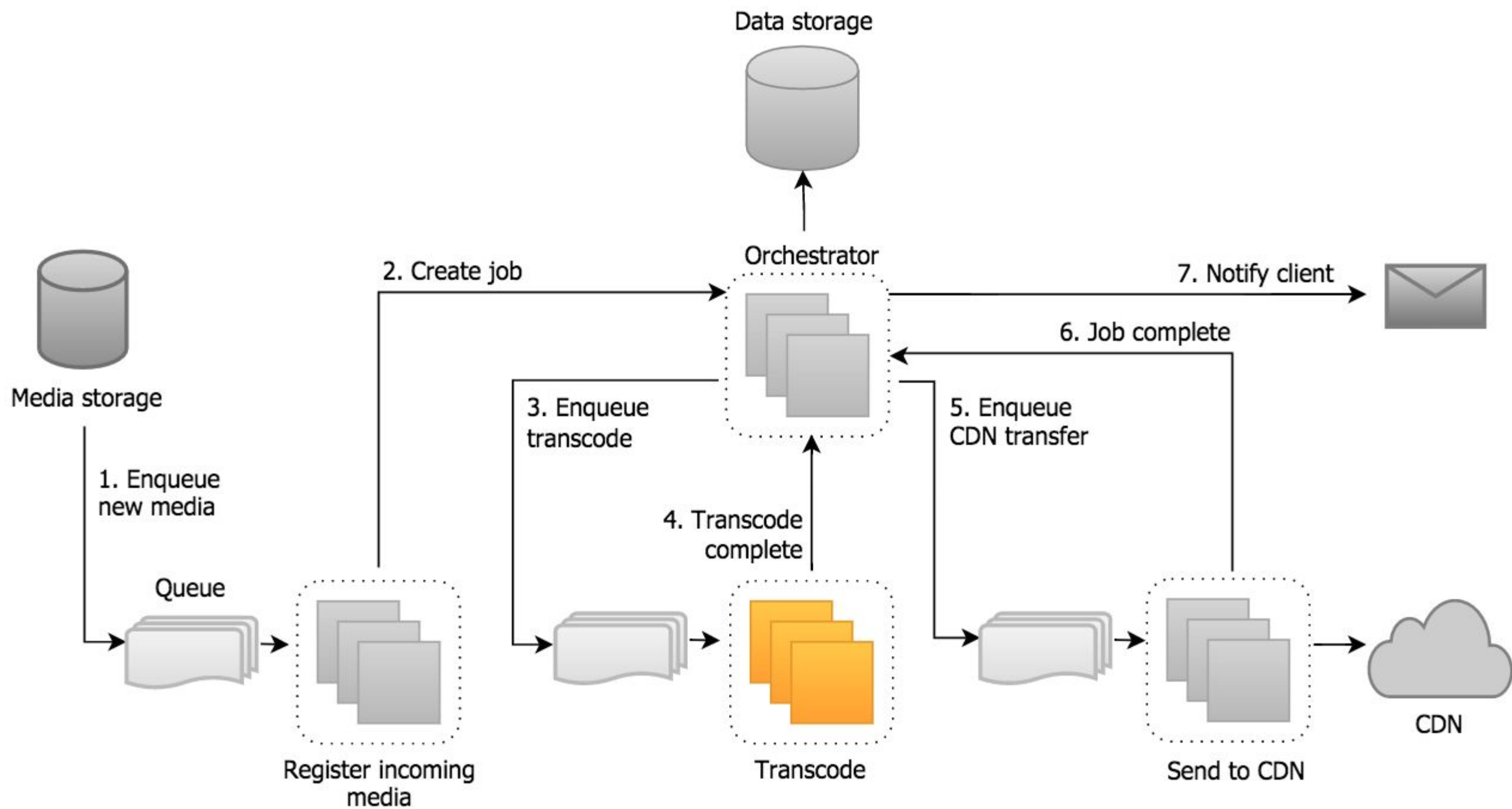
Challenges

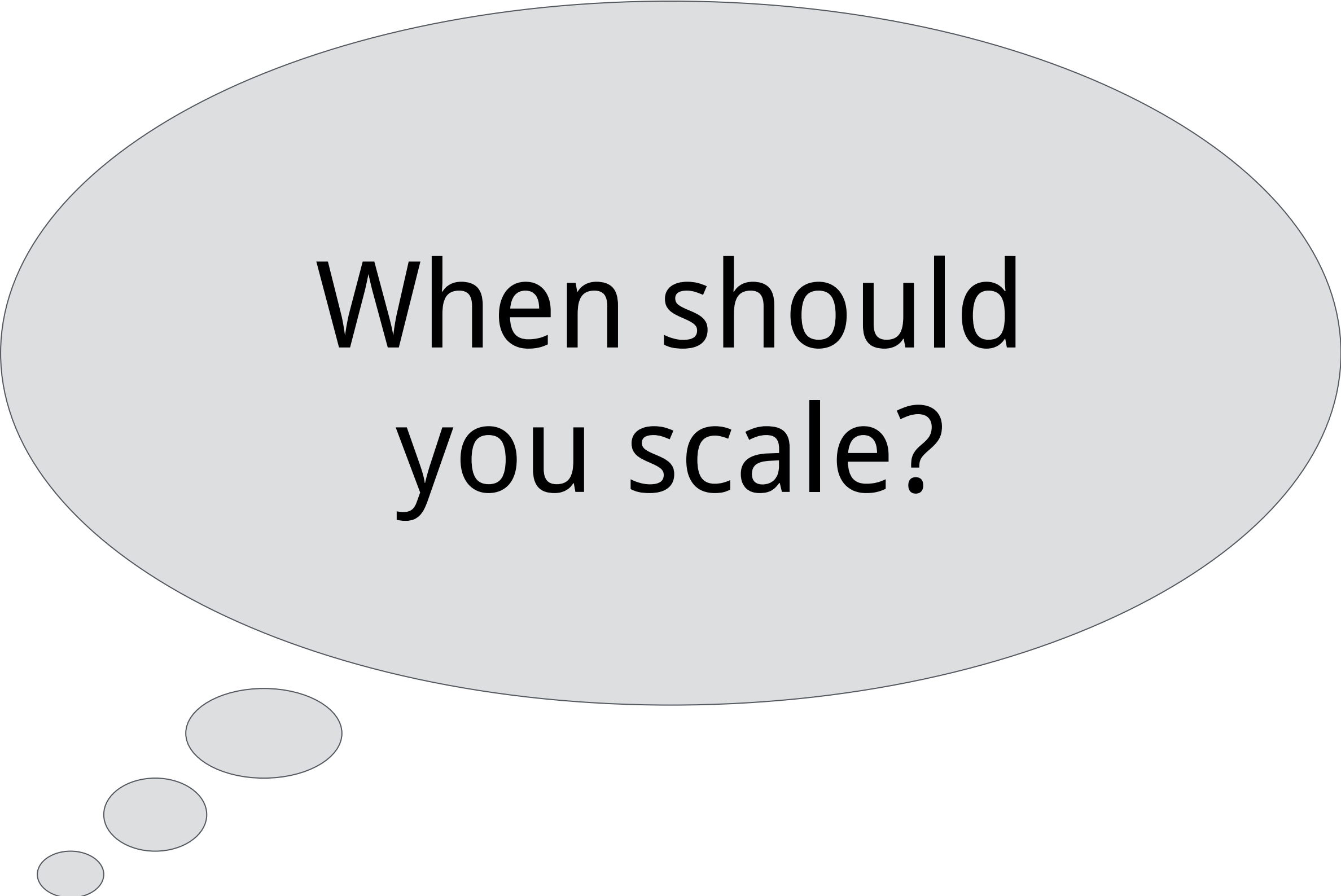
Deployments

General overhead

1. What is Transcoding?
- 2. Architecture**
3. Autoscaling transcoders
4. Costs
5. Future optimizations

1. What is Transcoding?
2. Architecture
- 3. Autoscaling transcoders**
4. Costs
5. Future optimizations





When should
you scale?

Scaling Considerations:

Content priority

Ad-hoc requests

Varied content length

AWS Considerations:

Hourly billing

Cost vs Speed

Fleet reaction time

Scaling down

MVP:

Don't make things wait
too long

MVP:

Supply enough
capacity at all times



Profile:

- Renditions (6)
- Crop
- Normalize audio
- Adjust ratio
- Add watermark
- GXF to MP4











```
var autoscaling = new AWS.AutoScaling(...);
```

```
autoscaling.setDesiredCapacity(params, ...);
```

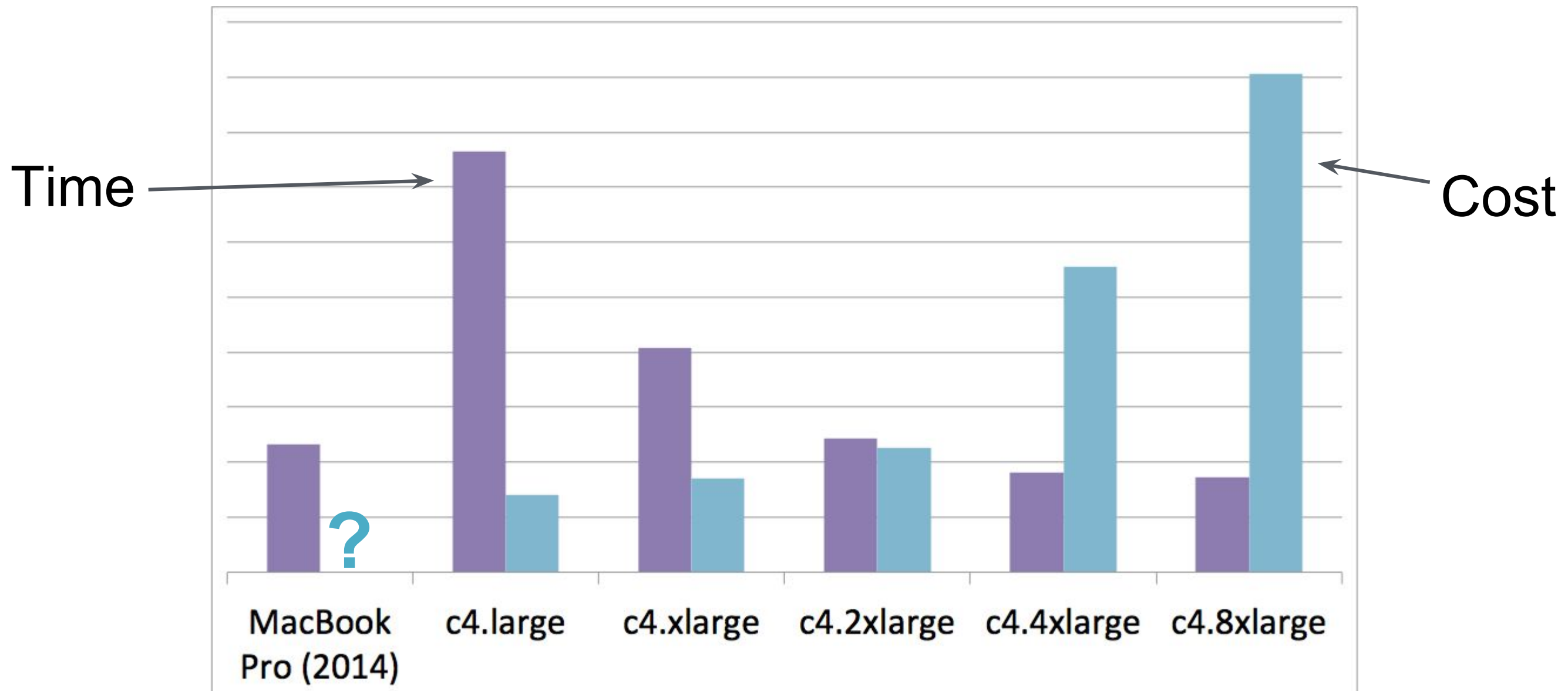
```
var autoscaling = new AWS.AutoScaling(...);
```

```
var params = {  
    AutoScalingGroupName: groupName,  
    DesiredCapacity: capacity  
};
```

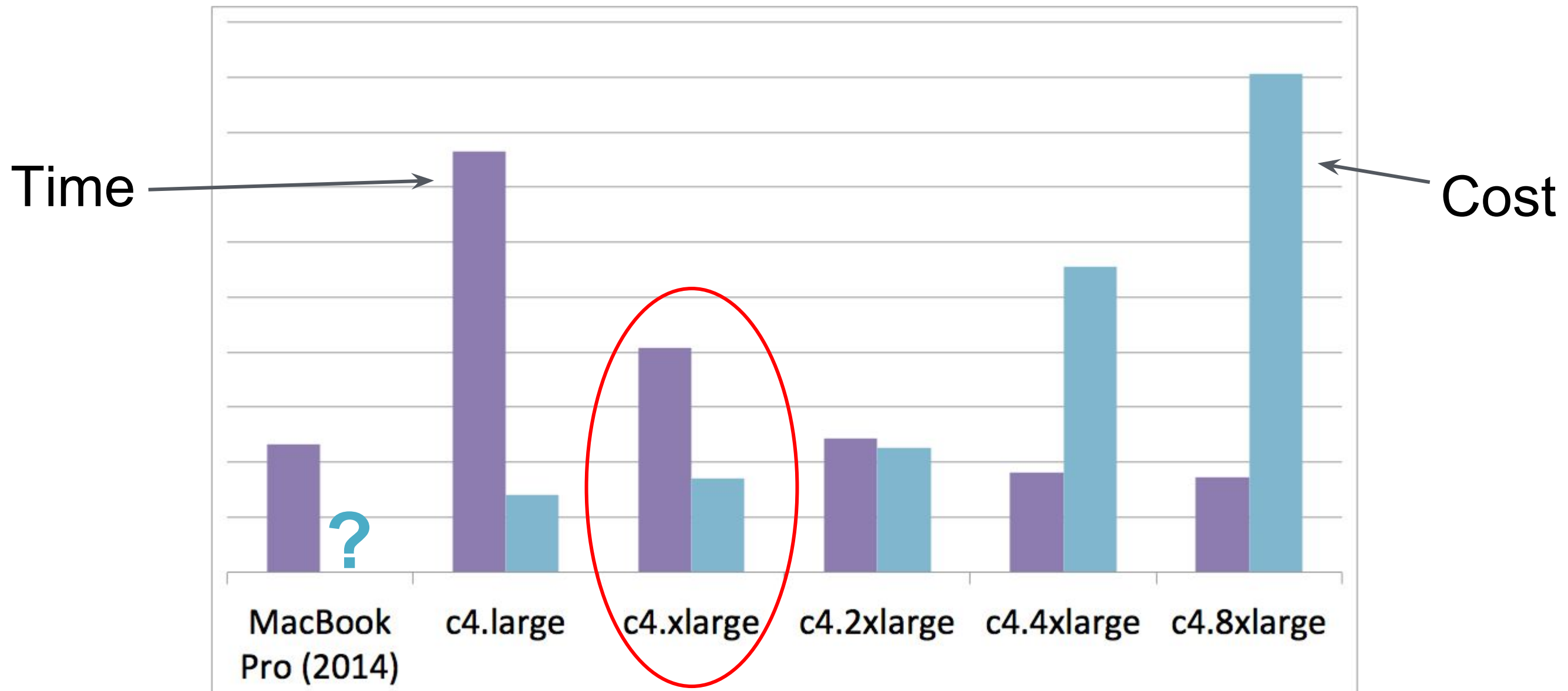
```
autoscaling.setDesiredCapacity(params, ...);
```

Which instance size?

Time vs Cost for 1 hour of content



Time vs Cost for 1 hour of content



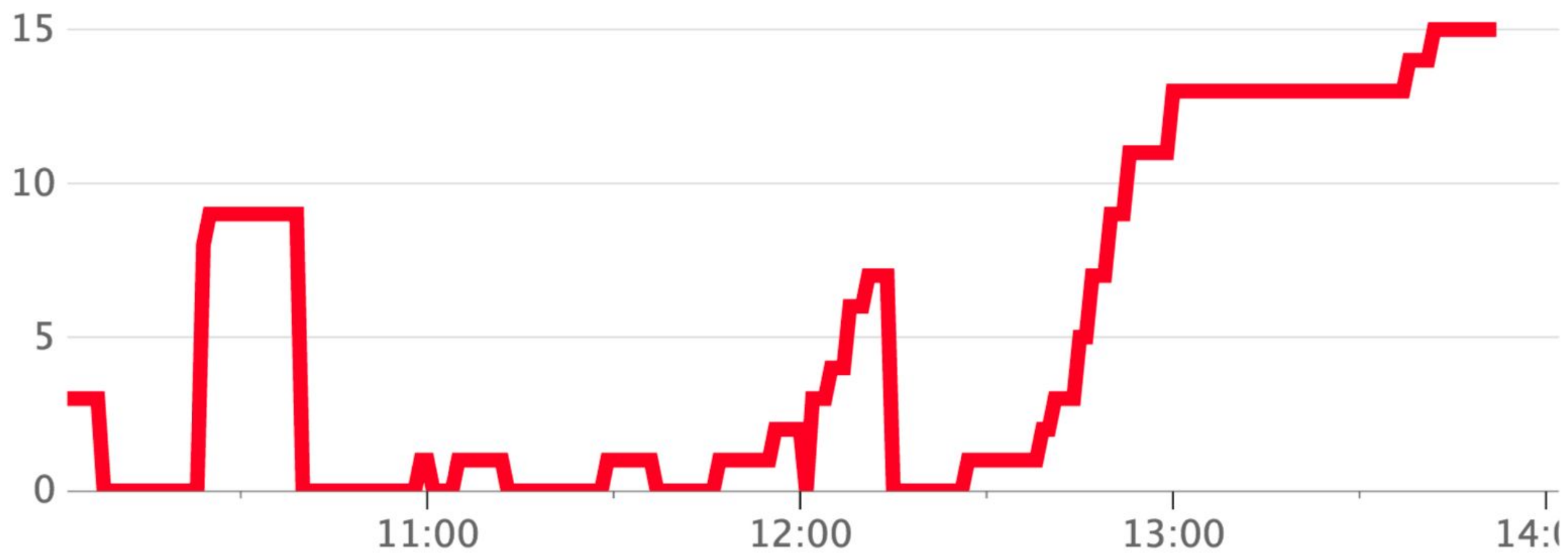
Winner:

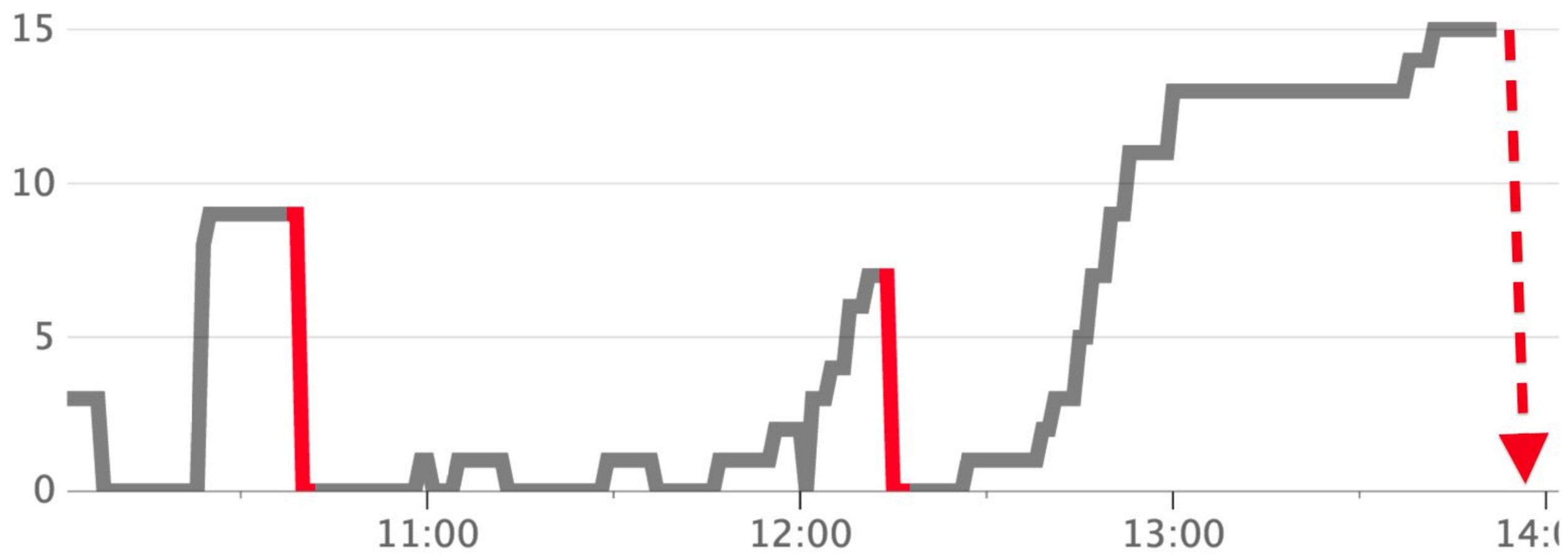
c4.xlarge

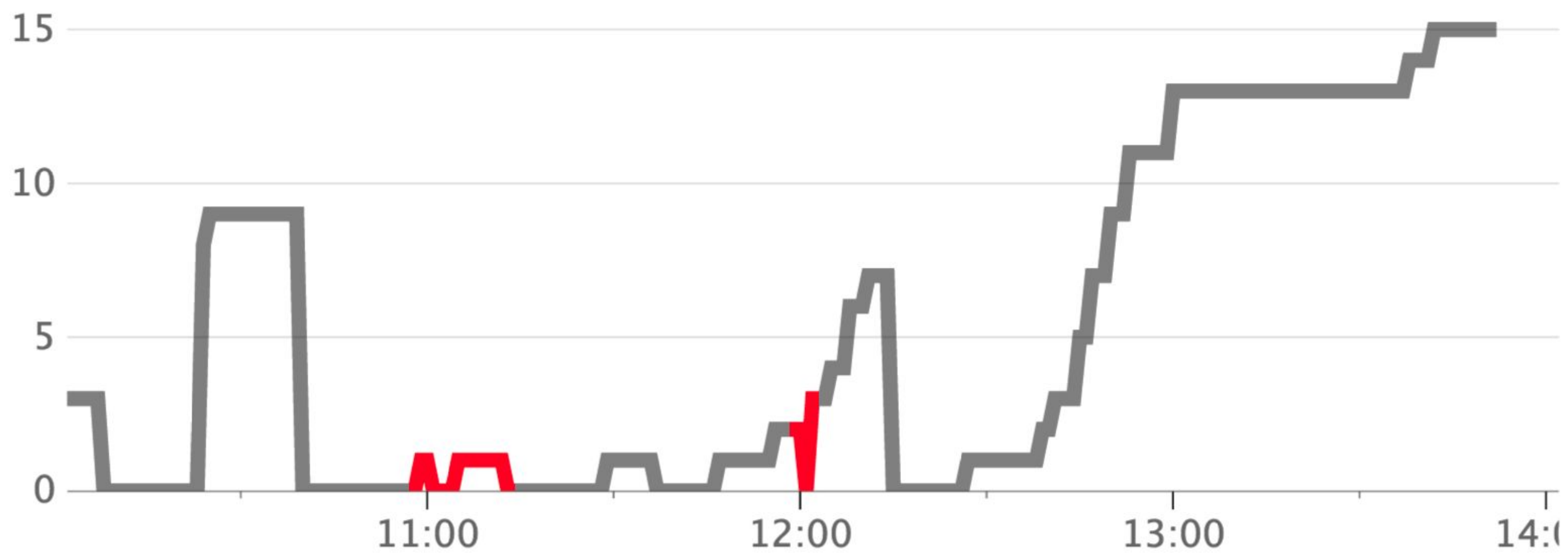
When to scale down?

Up: step scaling

Down: stop everything

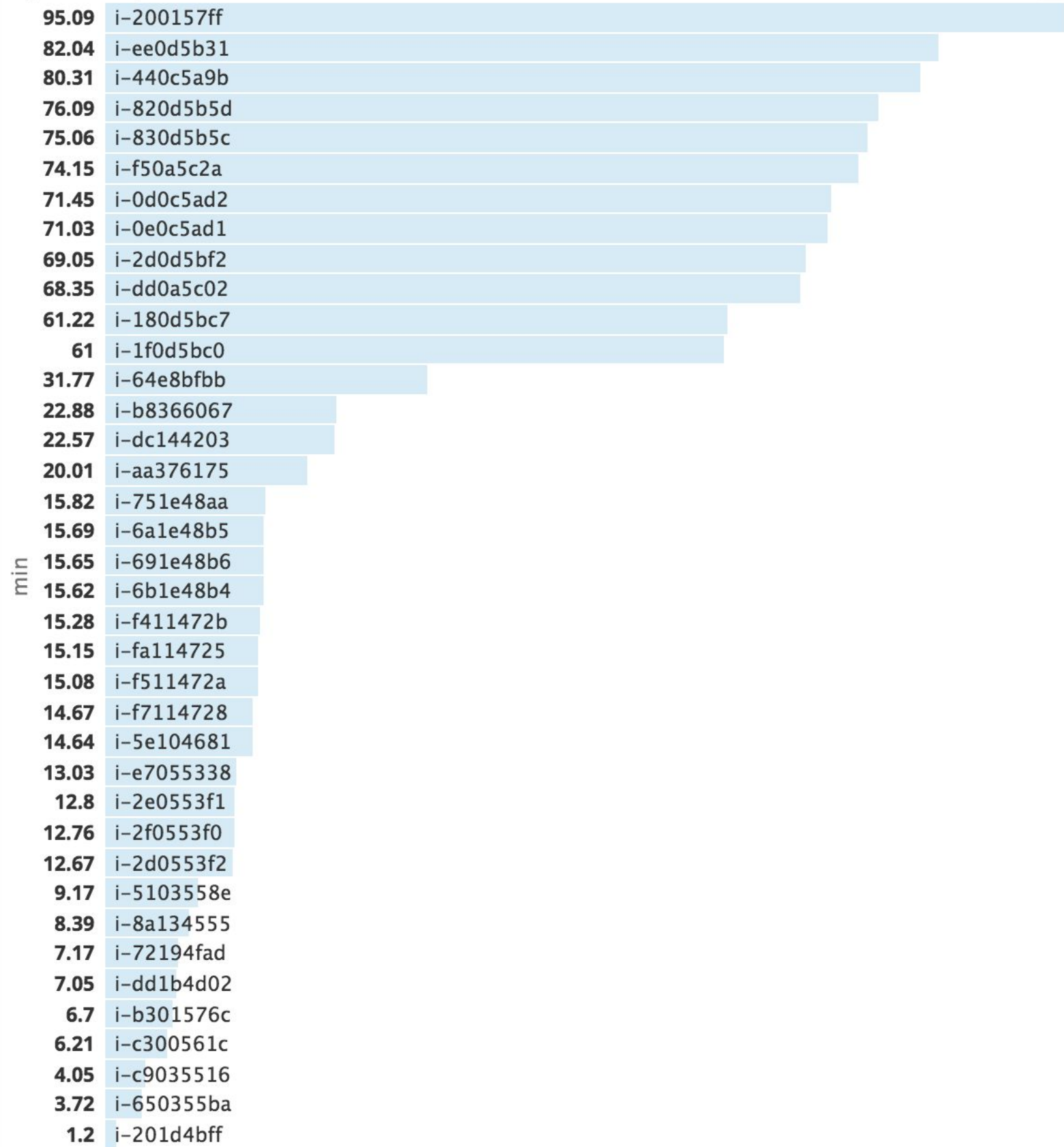






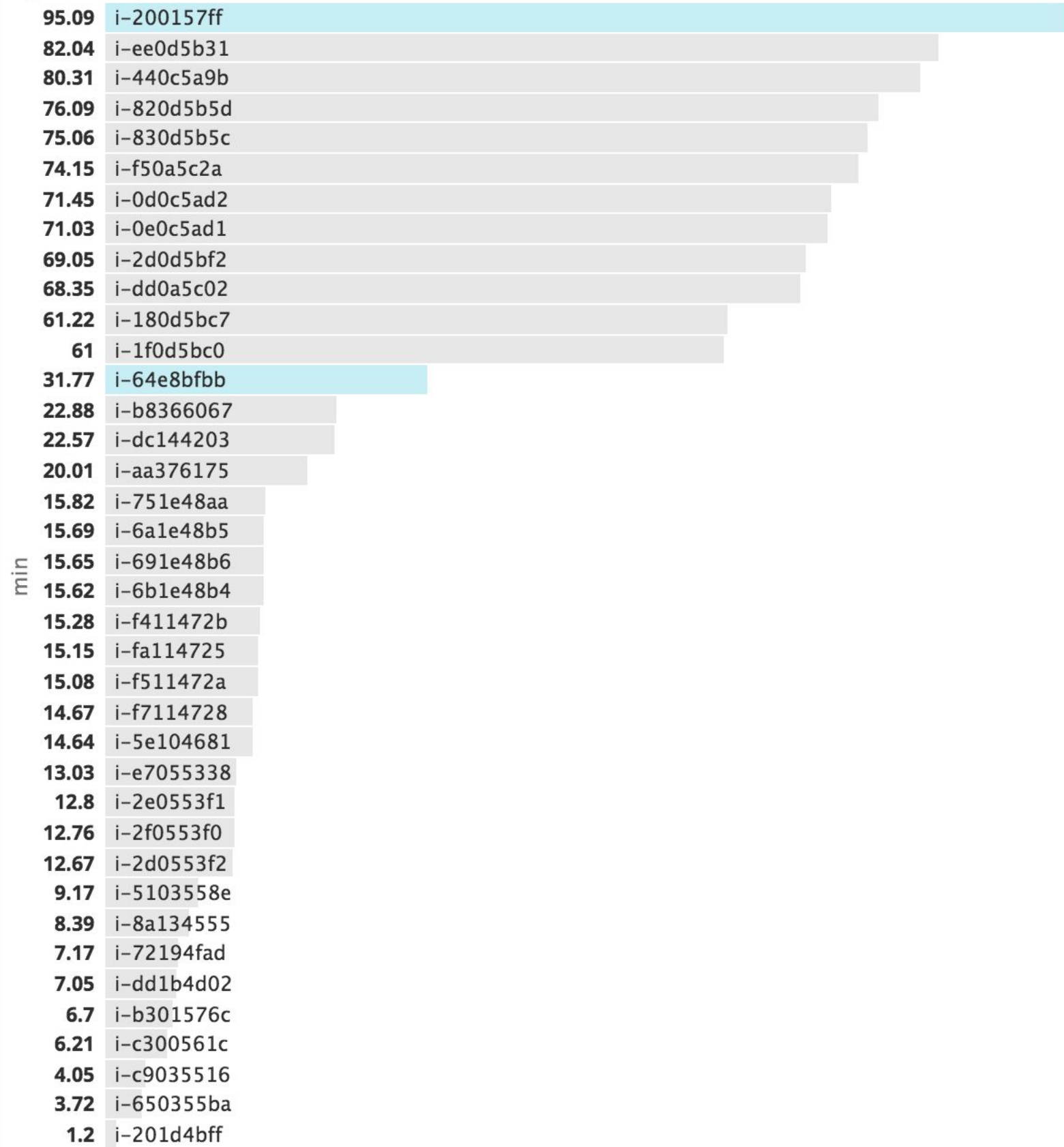
system.uptime

4h



system.uptime

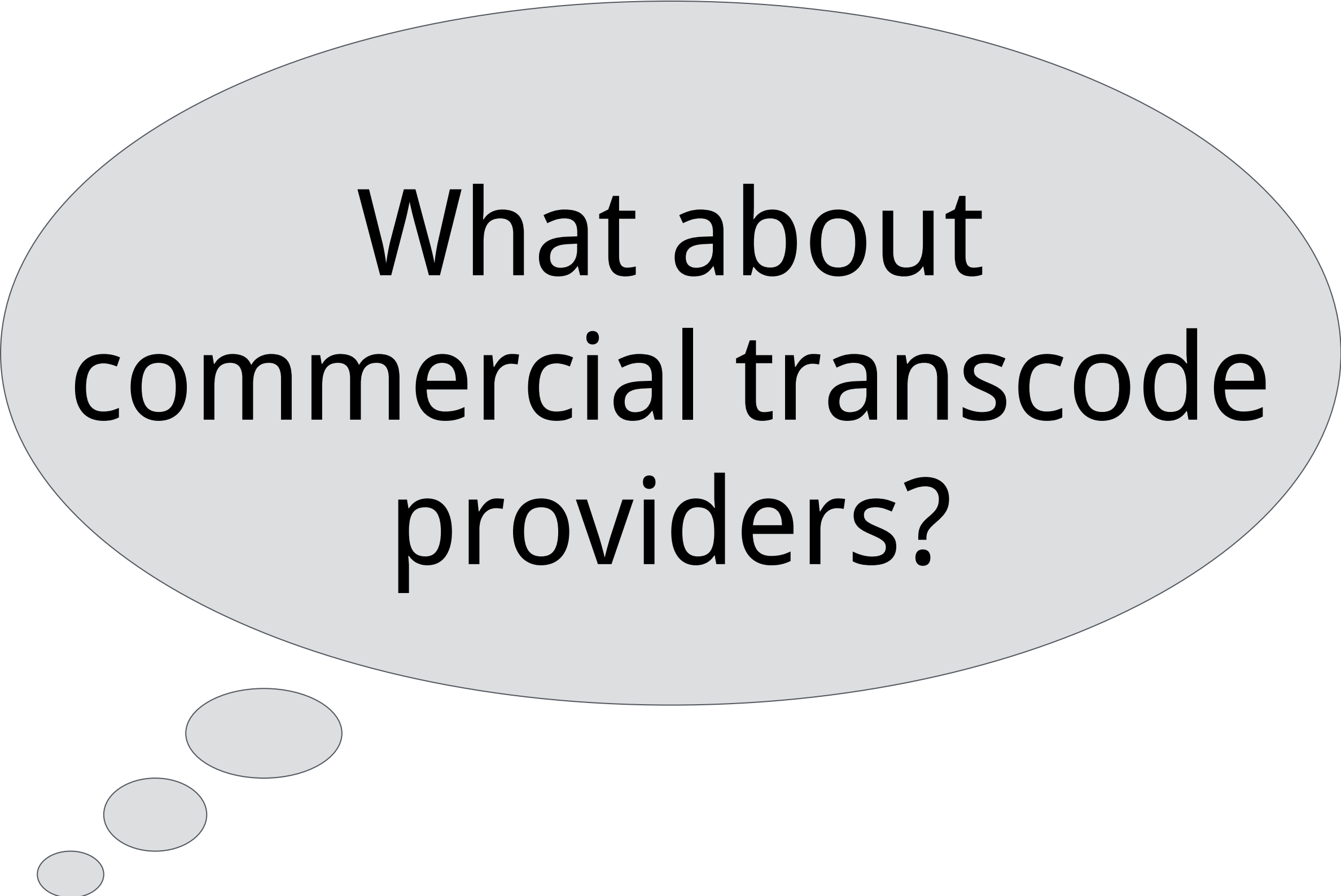
4h



An interesting puzzle :)

1. What is Transcoding?
2. Architecture
- 3. Autoscaling transcoders**
4. Costs
5. Future optimizations

1. What is Transcoding?
2. Architecture
3. Autoscaling transcoders
- 4. Costs**
5. Future optimizations

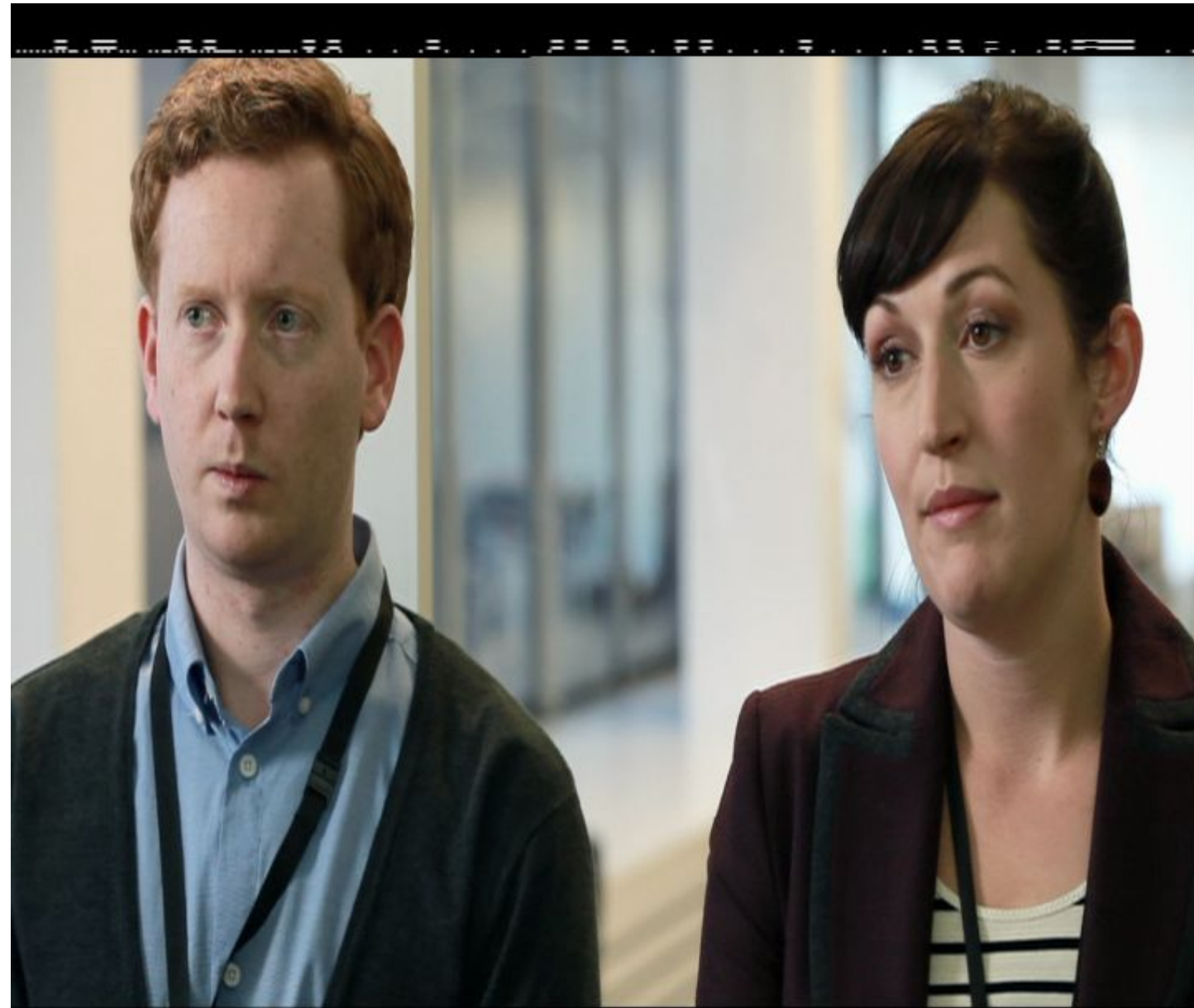


What about
commercial transcode
providers?

Full-Service Transcoding

\$(a lot)

DIY on-demand

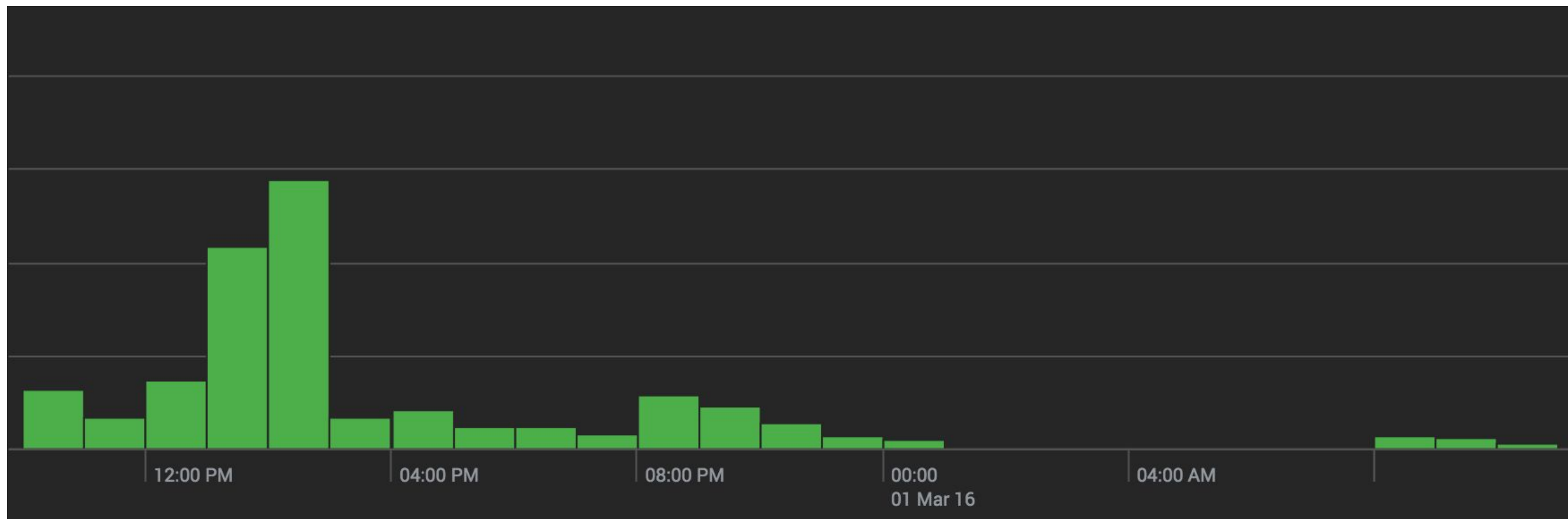


\$45k

Metro's costs?

\$1 1.2k

Some daily examples

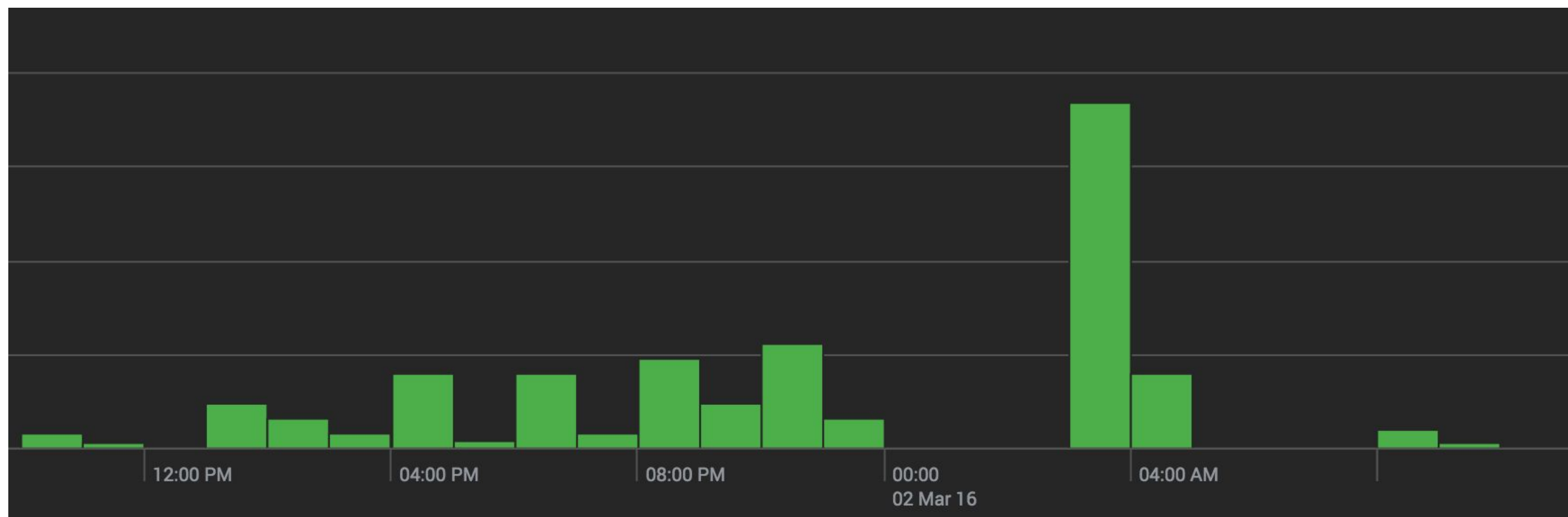


(10am - 10am)

\$42.86

Transcoding costs: 24 hrs

EC2 Instances:	\$34.31
EC2 (other):	\$ 4.24
RDS:	\$ 2.66
S3:	\$ 1.65
SQS:	\$ 0.00
Total (USD):	\$42.86



Next 24 hours

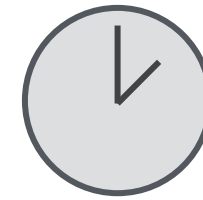
\$25.93



1. What is Transcoding?
2. Architecture
3. Autoscaling transcoders
- 4. Costs**
5. Future optimizations

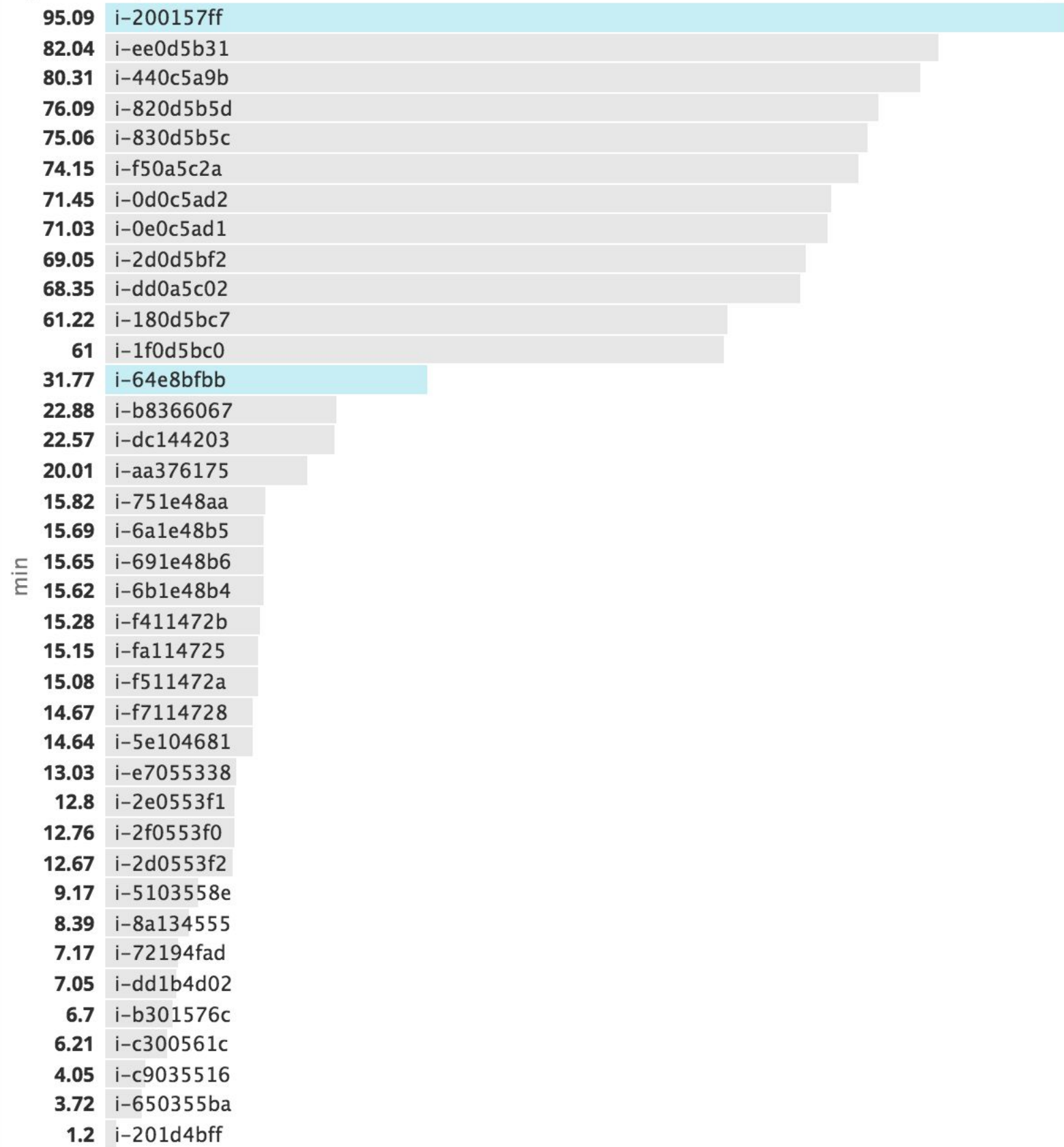
1. What is Transcoding?
2. Architecture
3. Autoscaling transcoders
4. Costs
- 5. Future optimizations**

Throughput optimizations



system.uptime

4h



Transcoding in chunks/segments



Introspection & Validation



Profile:

- Crop
- Normalize audio
- Adjust ratio
- Add watermark
- GXF to MP4

ffprobe

```
ffprobe version 2.7.1 Copyright (c) 2007-2015 the FFmpeg developers
  built with llvm-gcc 4.2.1 (LLVM build 2336.11.00)
  configuration: --prefix=/Volumes/Ramdisk/sw --enable-gpl --enable-pthreads --enable-version3 --enable-libspeex --enable-libvpx --disable-decoder=libvpx --enable-libmp3lame --enable-libtheora --enable-libvorbis --enable-libx264 --enable-avfilter --enable-libopencore_amrwb --enable-libopencore_amrnb --enable-filters --enable-libgsm --enable-libvidstab --enable-libx265 --disable-doc --arch=x86_64 --enable-runtime-cpudetect
  libavutil      54. 27.100 / 54. 27.100
  libavcodec     56. 41.100 / 56. 41.100
  libavformat    56. 36.100 / 56. 36.100
  libavdevice    56.  4.100 / 56.  4.100
  libavfilter    5. 16.101 /  5. 16.101
  libswscale     3.  1.101 /  3.  1.101
  libswresample  1.  2.100 /  1.  2.100
  libpostproc   53.  3.100 / 53.  3.100
[glfw @ 0x7f83bc000000] Could not find codec parameters for stream 5 (Unknown: none): unknown codec
Consider increasing the value for the 'analyzeduration' and 'probesize' options
Input #0, glfw, from 'utopia_01_07.glfw':
  Metadata:
    timecode           : 00:00:00:00
    timecode_at_mark_in: 00:00:00:00
    timecode_at_mark_out: 00:26:25:24
  Duration: 00:26:26.00, start: 0.000000, bitrate: 26126 kb/s
    Stream #0:0: Video: mpeg2video (4:2:2), yuv422p(tv), 720x608 [SAR 152:135 DAR 4:3], 25000 kb/s, 50 fps, 25 tbr, 50 tbn, 50 tbc
    Stream #0:1: Audio: pcm_s16le, 48000 Hz, mono, s16, 768 kb/s
    Stream #0:2: Audio: pcm_s16le, 48000 Hz, mono, s16, 768 kb/s
    Stream #0:3: Audio: pcm_s16le, 48000 Hz, mono, s16, 768 kb/s
    Stream #0:4: Audio: pcm_s16le, 48000 Hz, mono, s16, 768 kb/s
    Stream #0:5: Unknown: none
    Stream #0:6: Data: none
  Unsupported codec with id 0 for input stream 5
  Unsupported codec with id 0 for input stream 6
```

26126 kb/s

(tv), 720x608 [SAR 1

s16, 768 kb/s

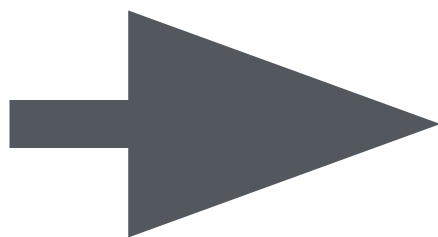
s16, 768 kb/s

kb/s, 50 fps, 25 tbr,

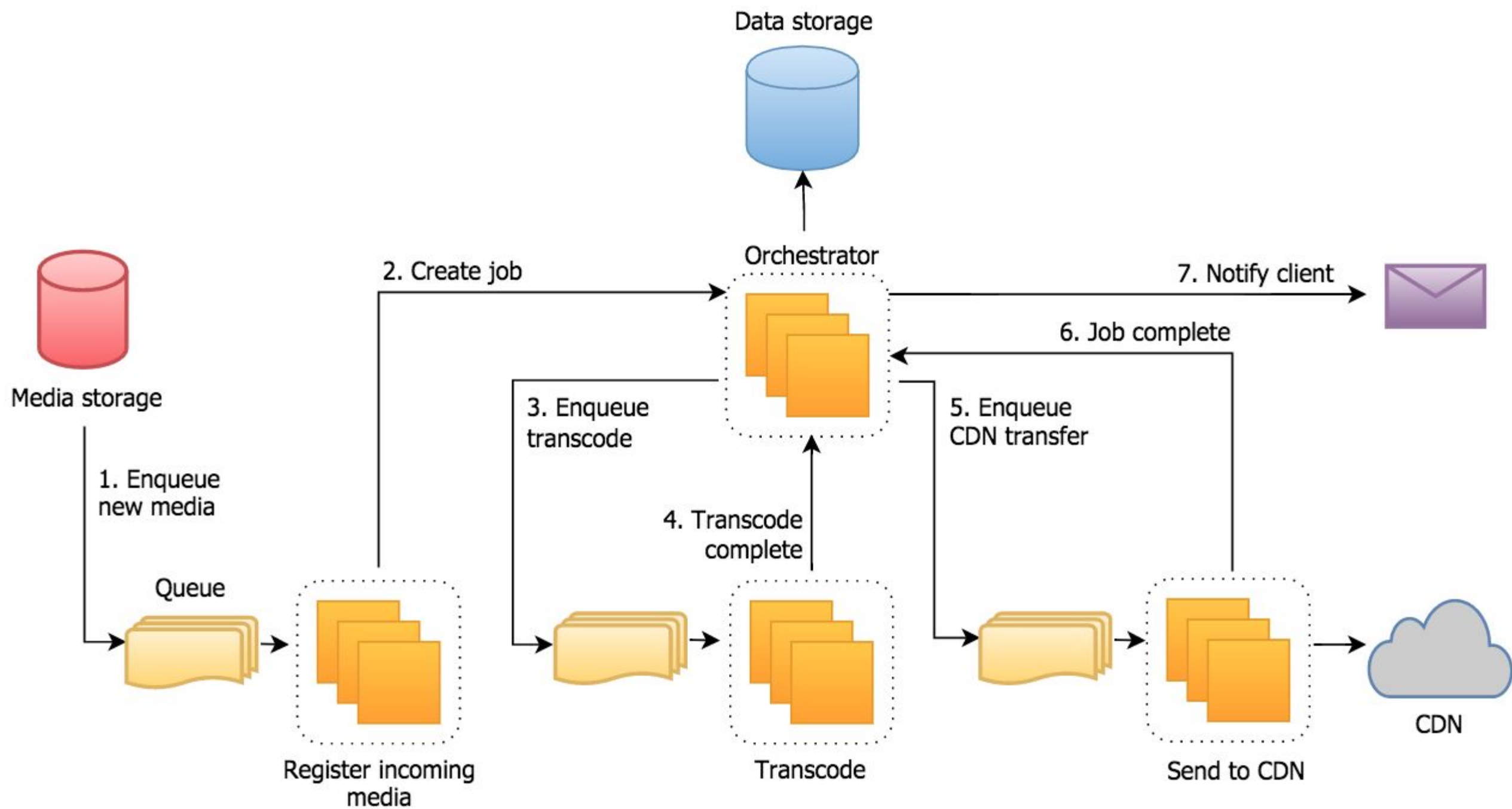
```
Stream #0:3: Audio: pcm_s16le  
Stream #0:4: Audio: pcm_s16le  
Stream #0:5: Unknown: none  
Stream #0:6: Data: none  
Unsupported codec with id 0 for i
```

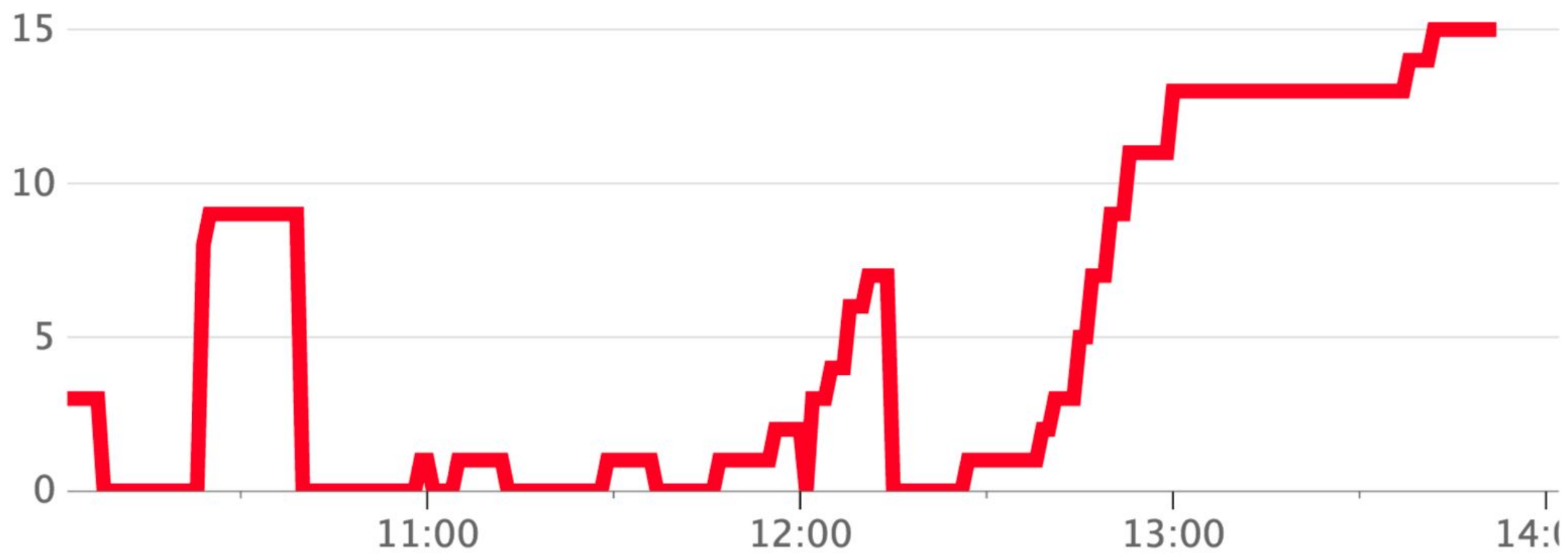
1. What is Transcoding?
2. Architecture
3. Autoscaling transcoders
4. Costs
- 5. Future optimizations**

Summary



ABC
iVIEW  





8 minutes 27 seconds

\$1 1.2k



Thanks!



@daphnechong

goto;
conference



**Click 'Rate Session'
to rate session
and ask questions.**