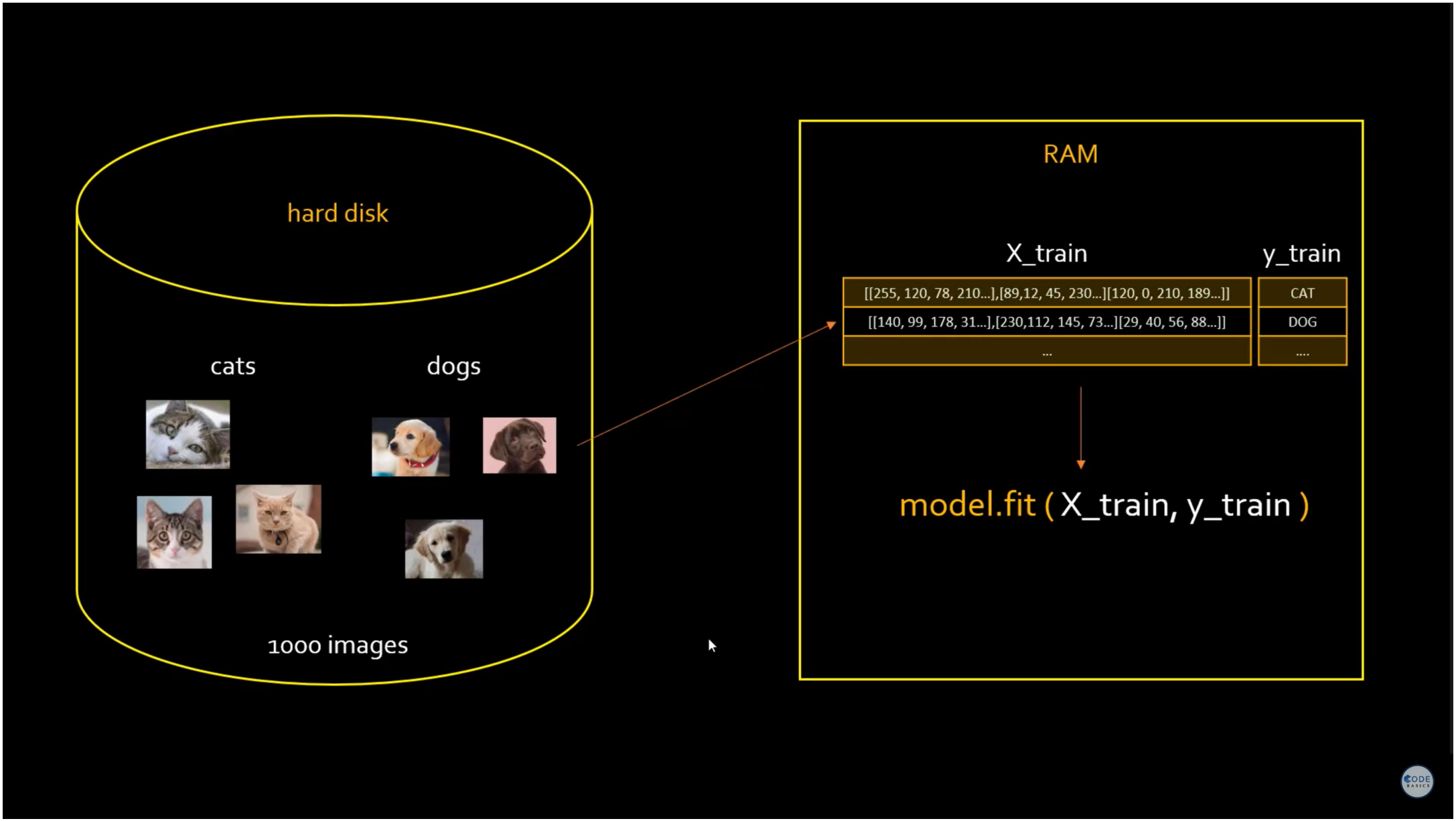
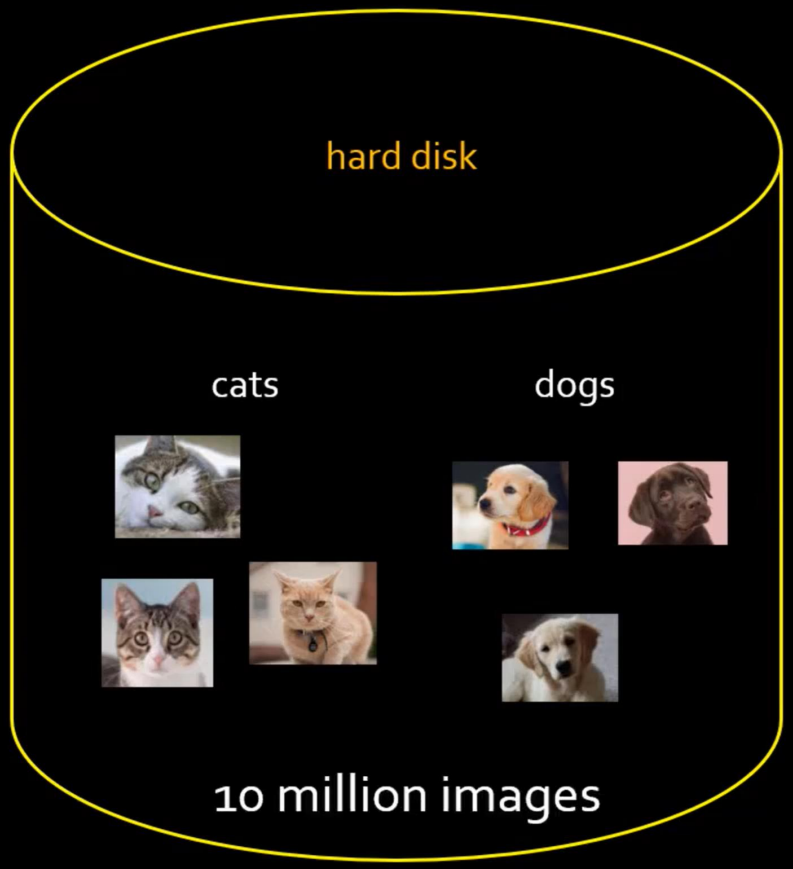
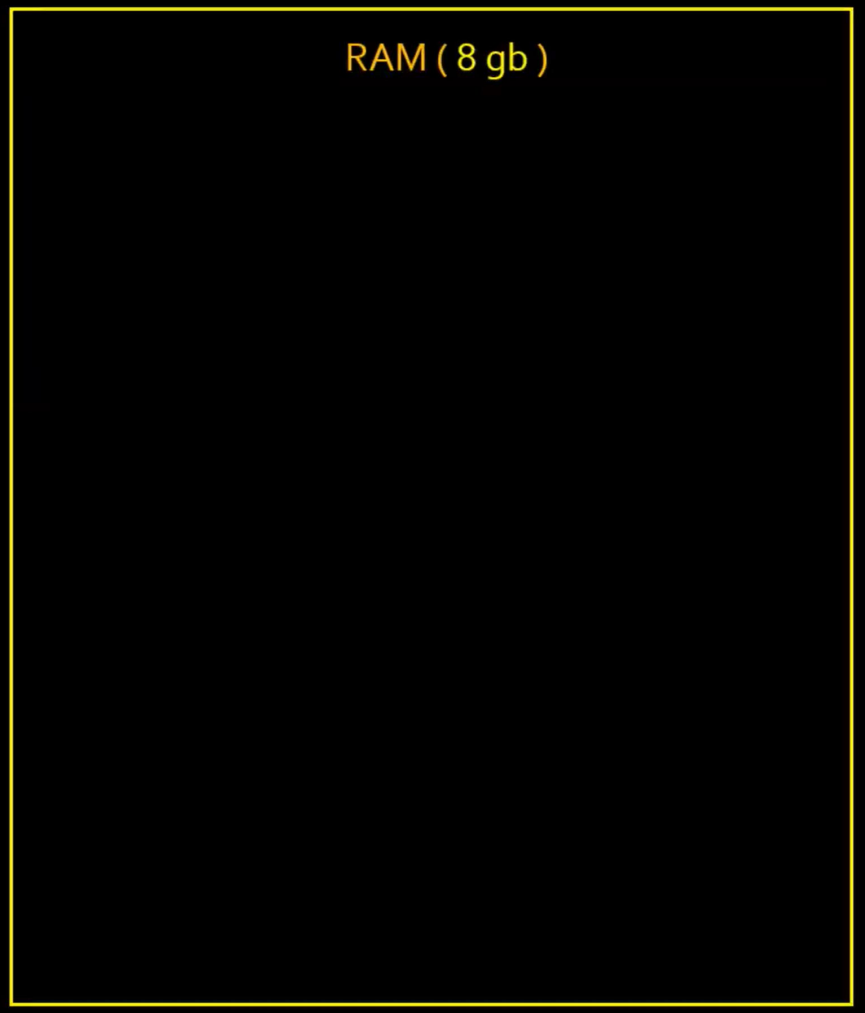
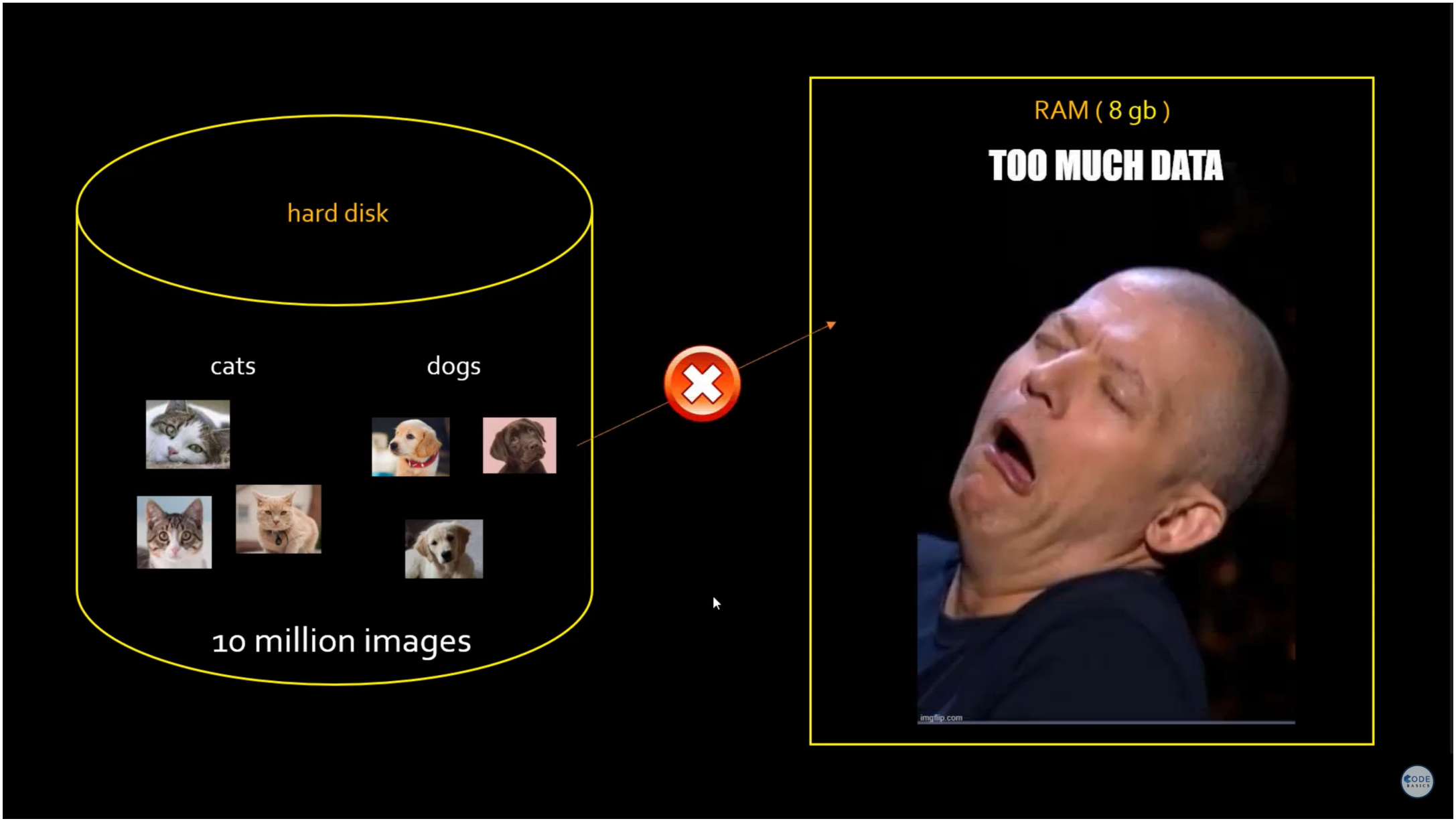
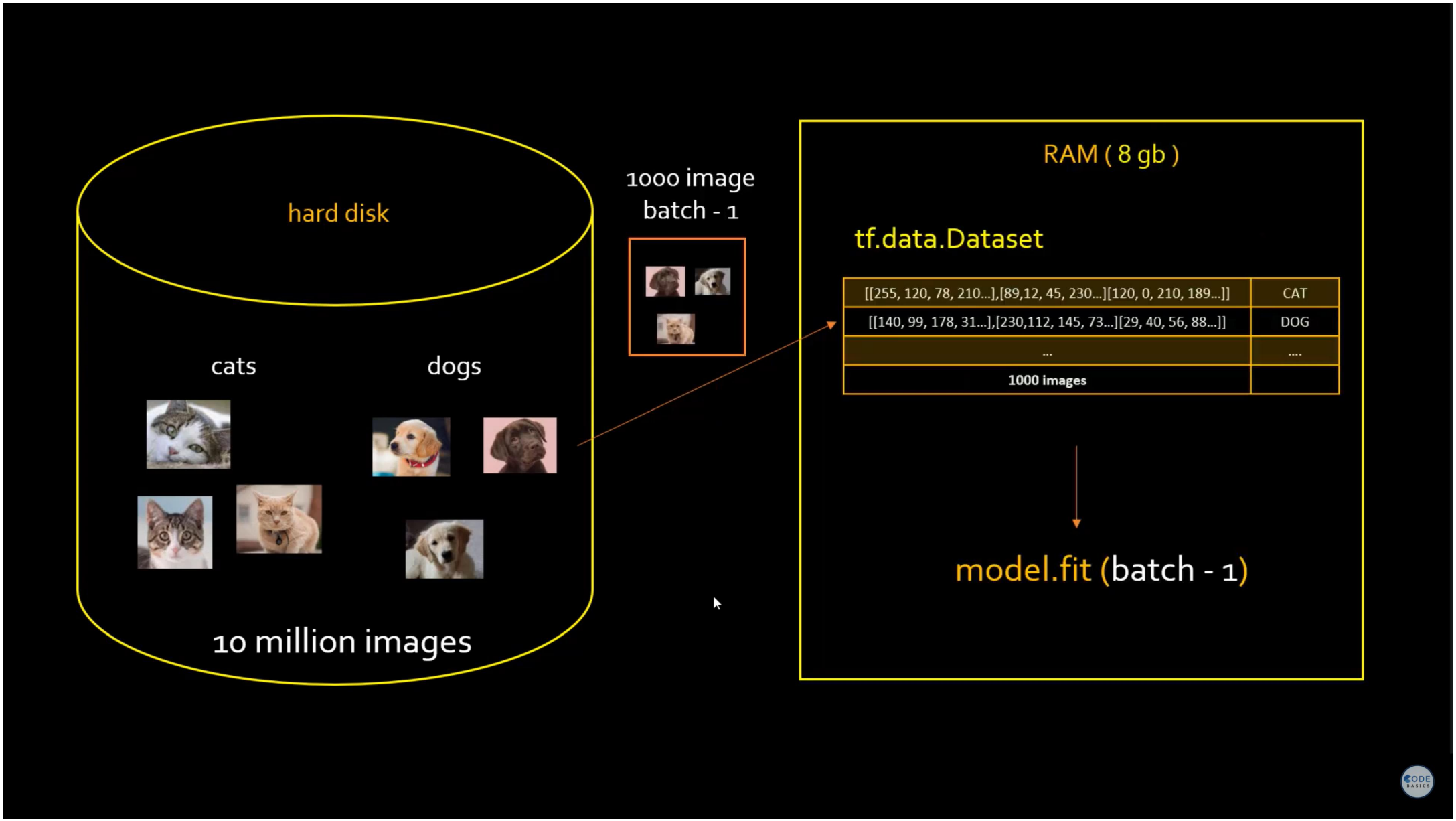
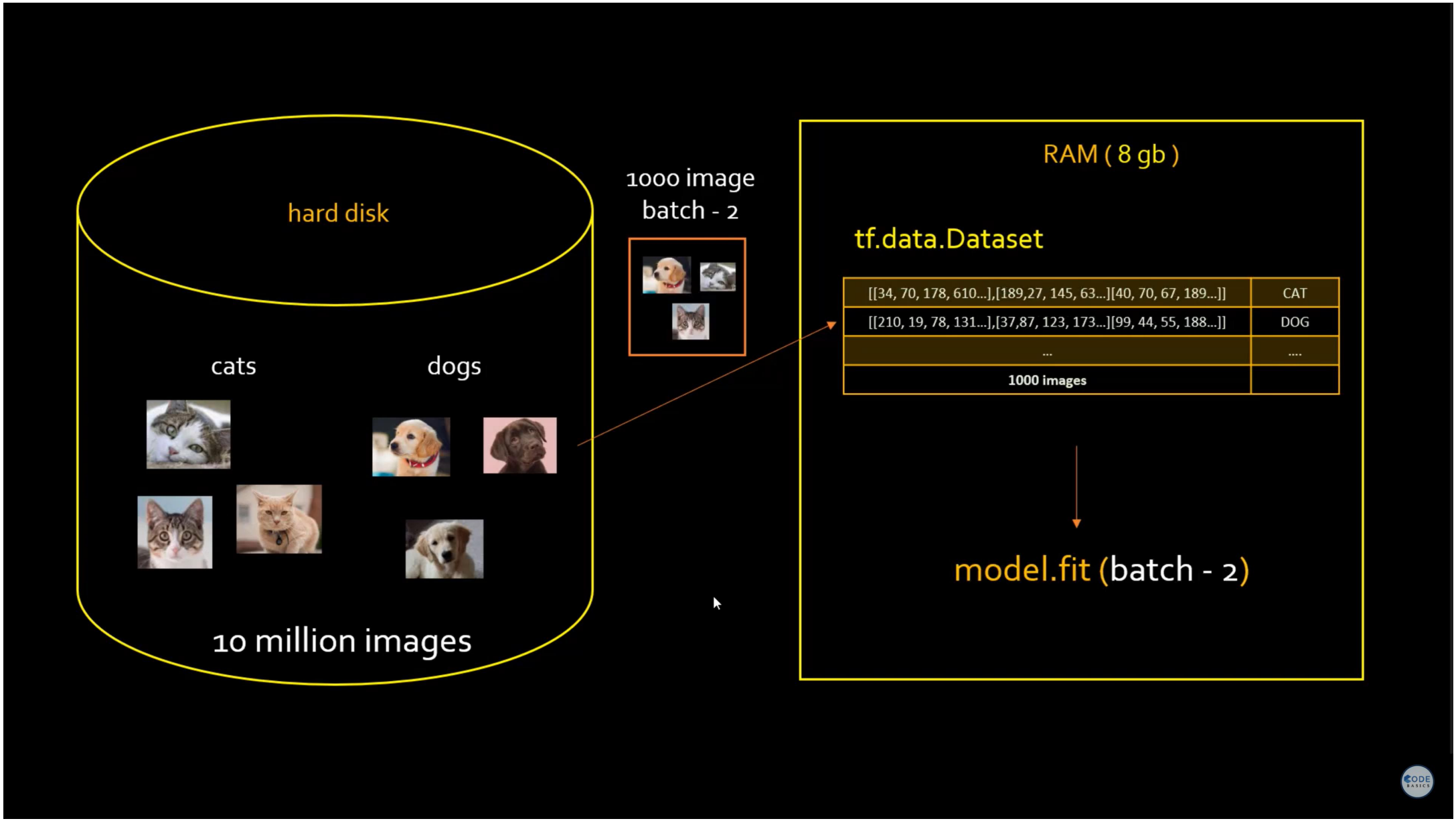

hard disk
cats
dogs
1000 images
RAM
X_train
y_train
[[255, 120, 78, 210...],[89,12, 45, 230...][120, 0, 210, 189...]]
CAT
[[140, 99, 178, 31...],[230,112, 145, 73...][29, 40, 56, 88...]]
DOG
...
....
model.fit (X_train, y_train)

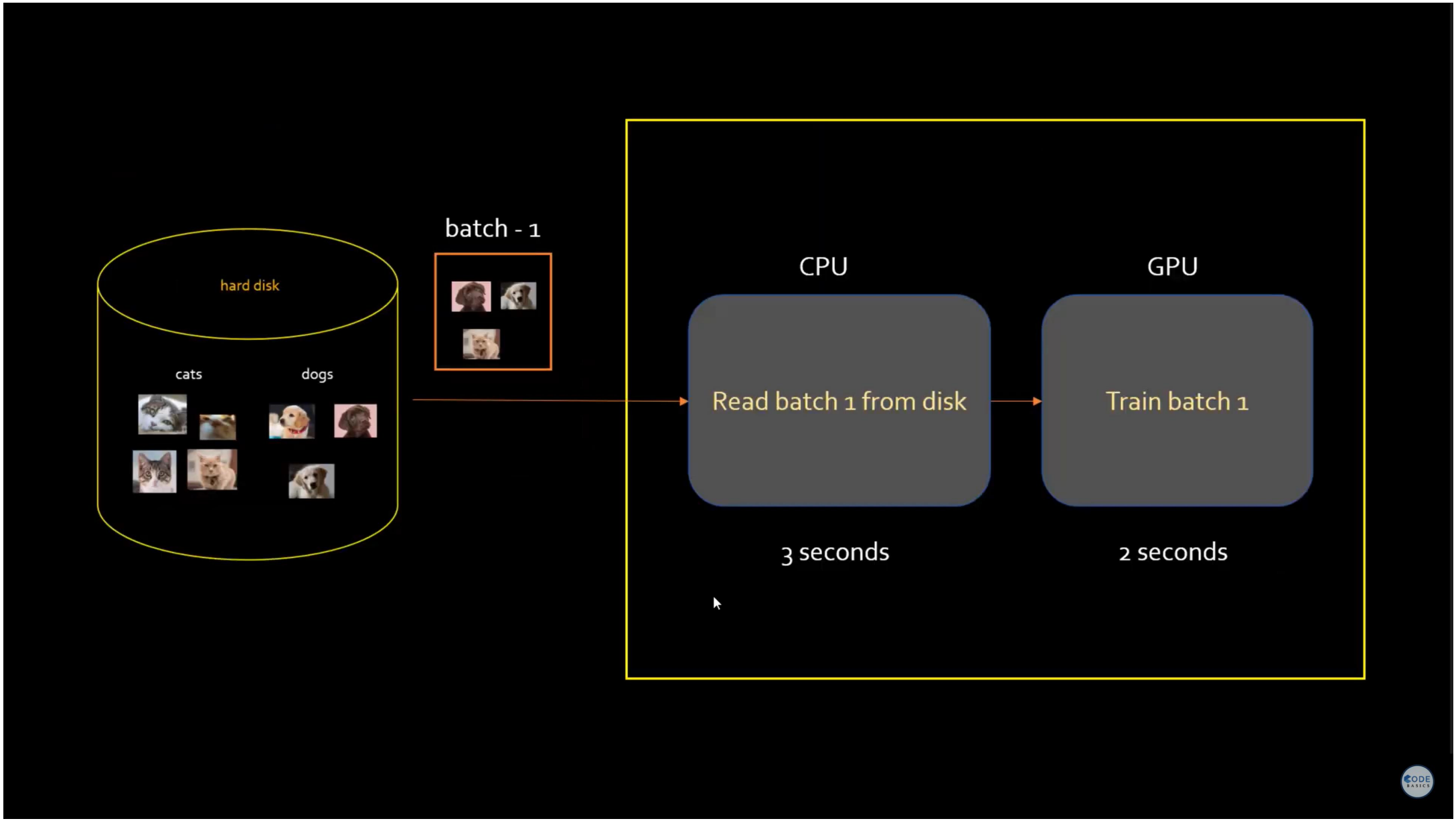

hard disk
cats
dogs
10 million images

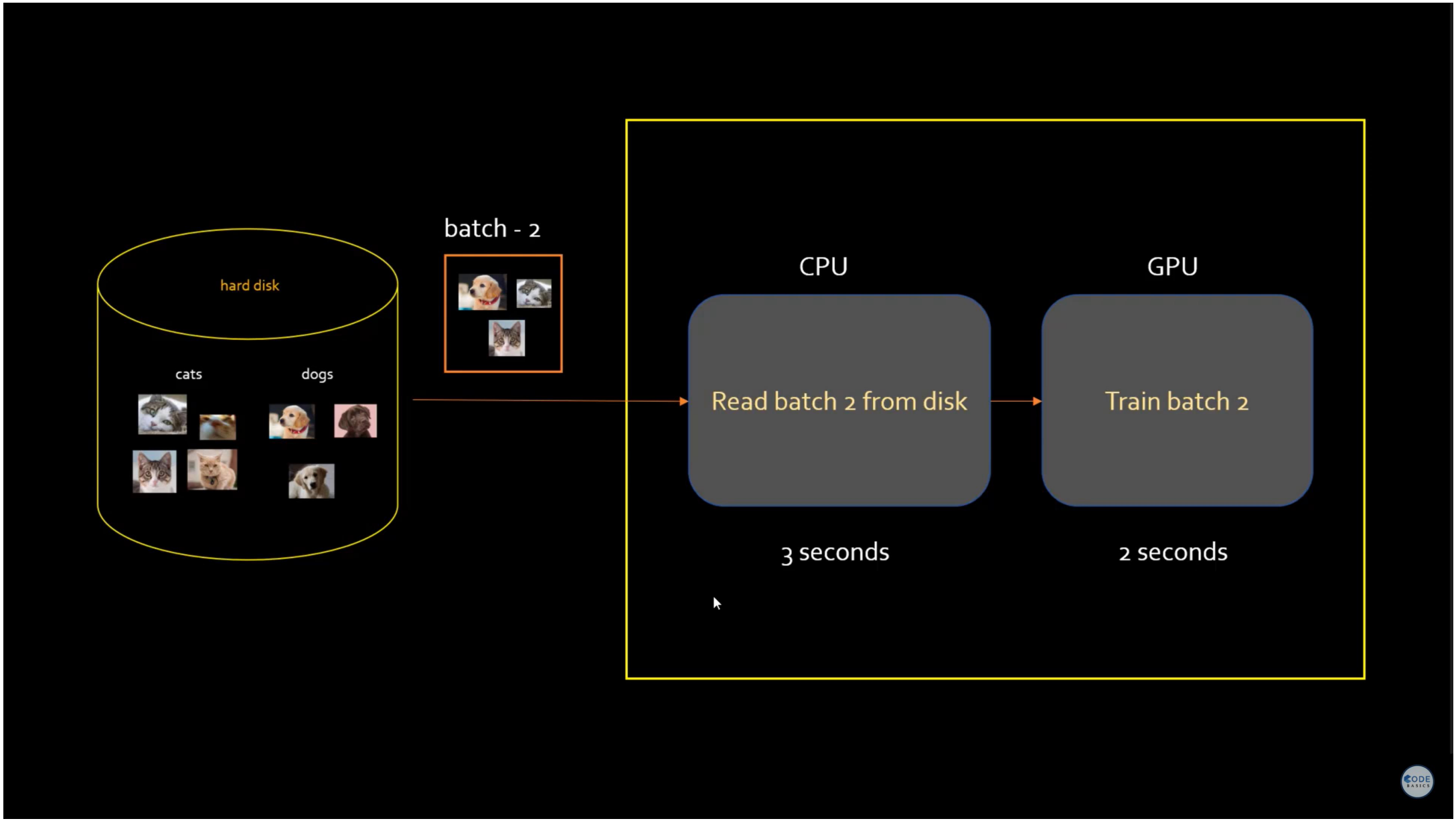

RAM (8 gb)

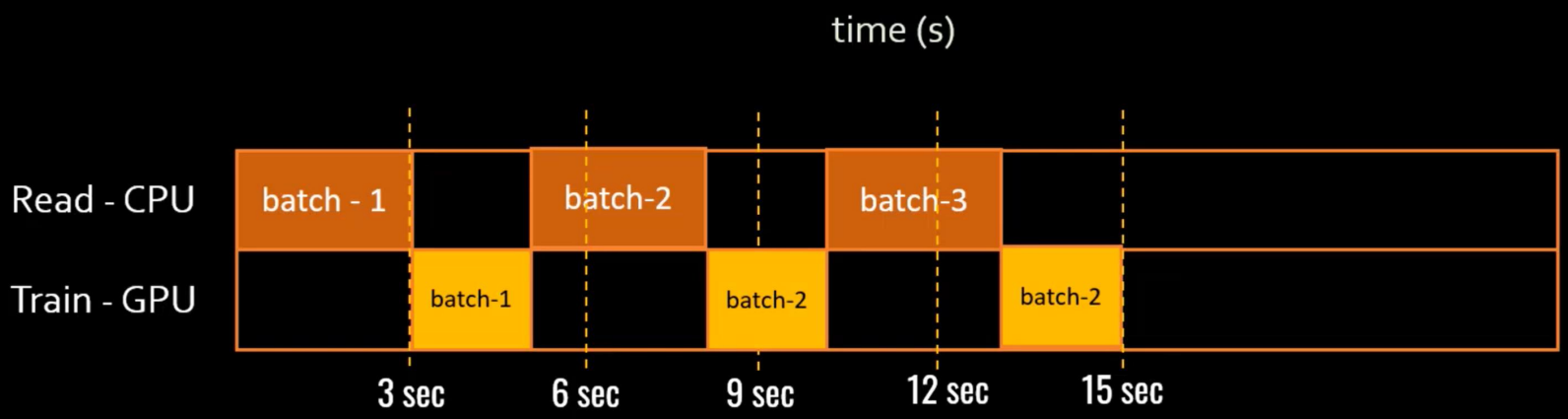

hard disk
cats
dogs
10 million images
RAM (8 gb)
TOO MUCH DATA
imgflip.com

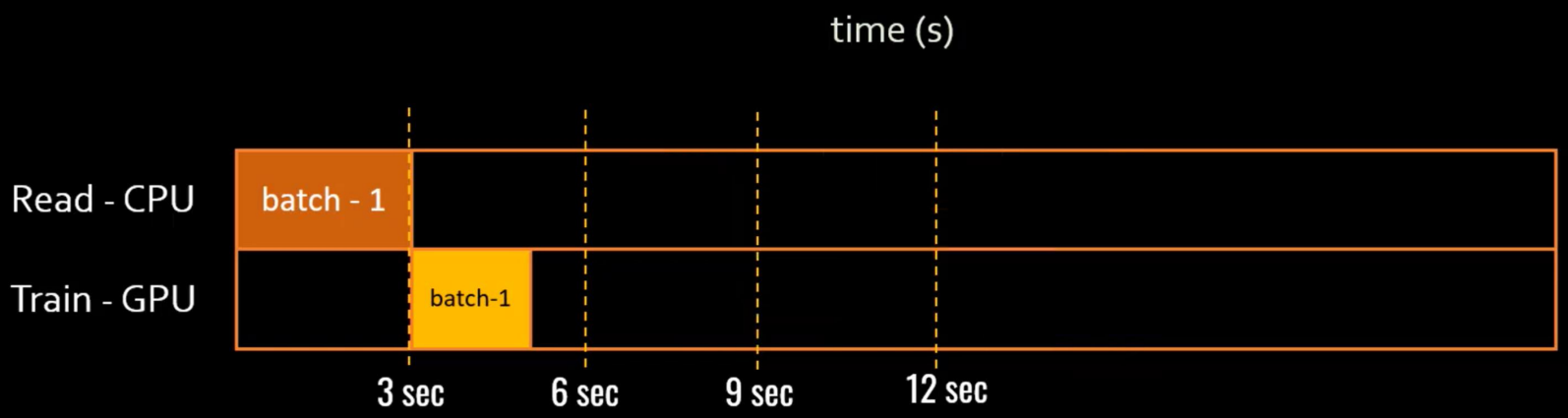

hard disk
cats
dogs
10 million images
1000 image
batch - 1
RAM (8 gb)
tf.data.Dataset
[[255, 120, 78, 210...],[89,12, 45, 230...][120, 0, 210, 189...]]
CAT
[[140, 99, 178, 31...],[230,112, 145, 73...][29, 40, 56, 88...]]
DOG
...
....
1000 images
model.fit (batch - 1)

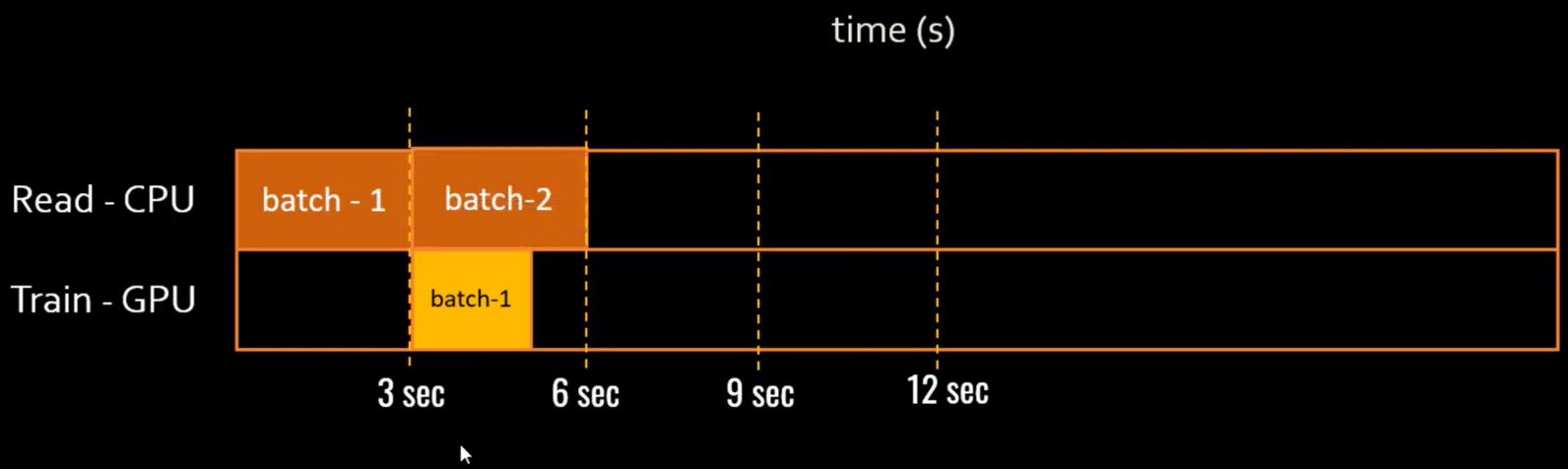

hard disk
cats
dogs
10 million images
1000 image
batch - 2
RAM (8 gb)
tf.data.Dataset
[[34, 70, 178, 610...],[189,27, 145, 63...][40, 70, 67, 189...]]
CAT
[[210, 19, 78, 131...],[37,87, 123, 173...][99, 44, 55, 188...]]
DOG
...
....
1000 images
model.fit (batch - 2)

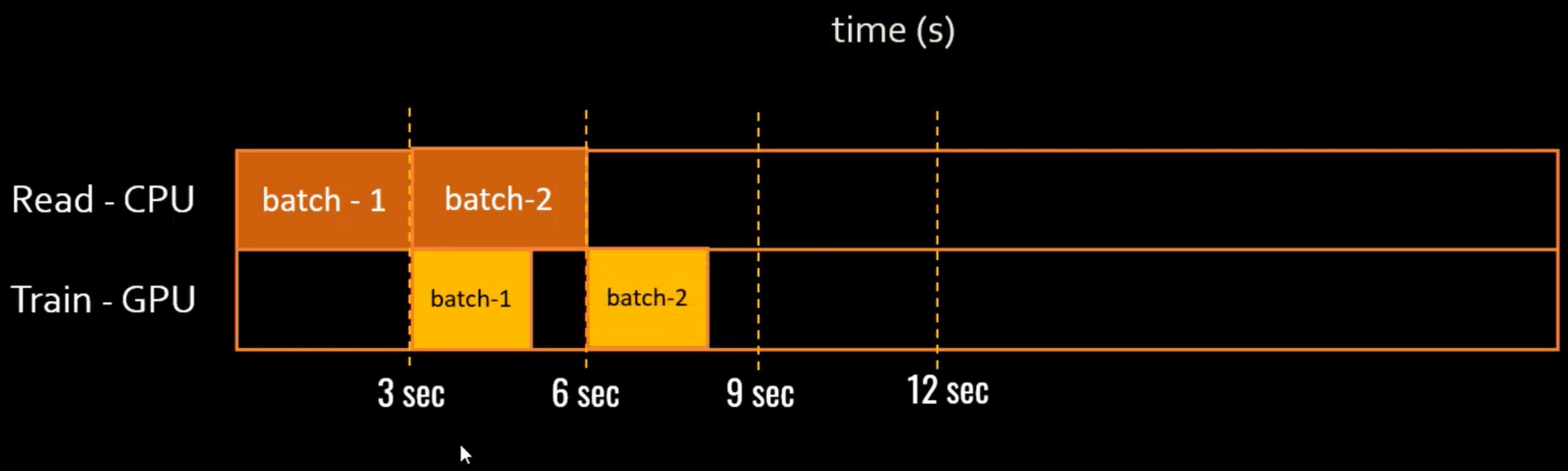

hard disk
cats
dogs
batch - 1
CPU
GPU
Read batch 1 from disk
Train batch 1
3 seconds
2 seconds

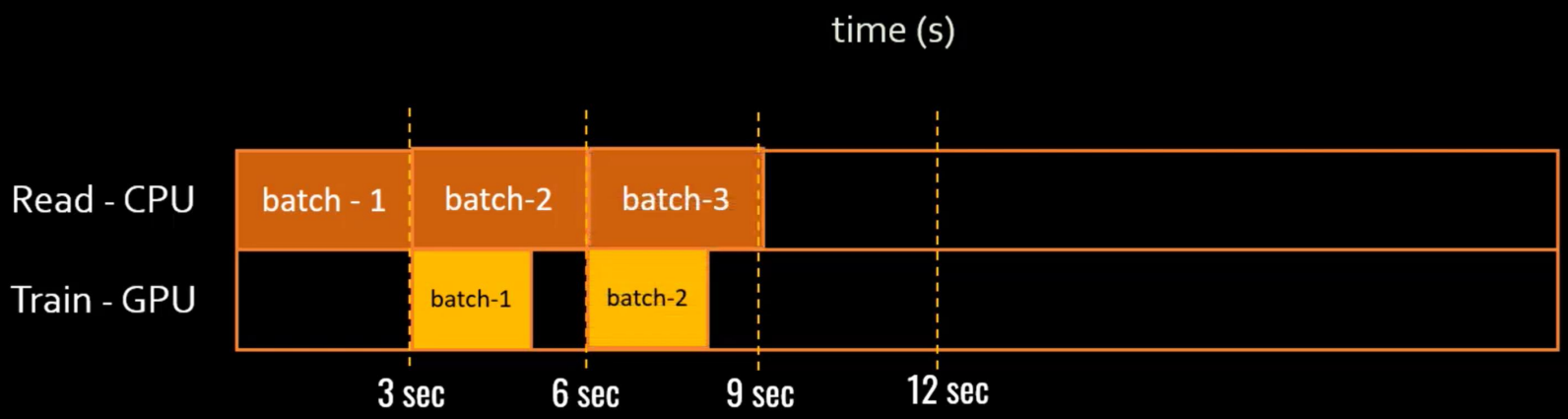

hard disk
cats
dogs
batch - 2
CPU
GPU
Read batch 2 from disk
Train batch 2
3 seconds
2 seconds

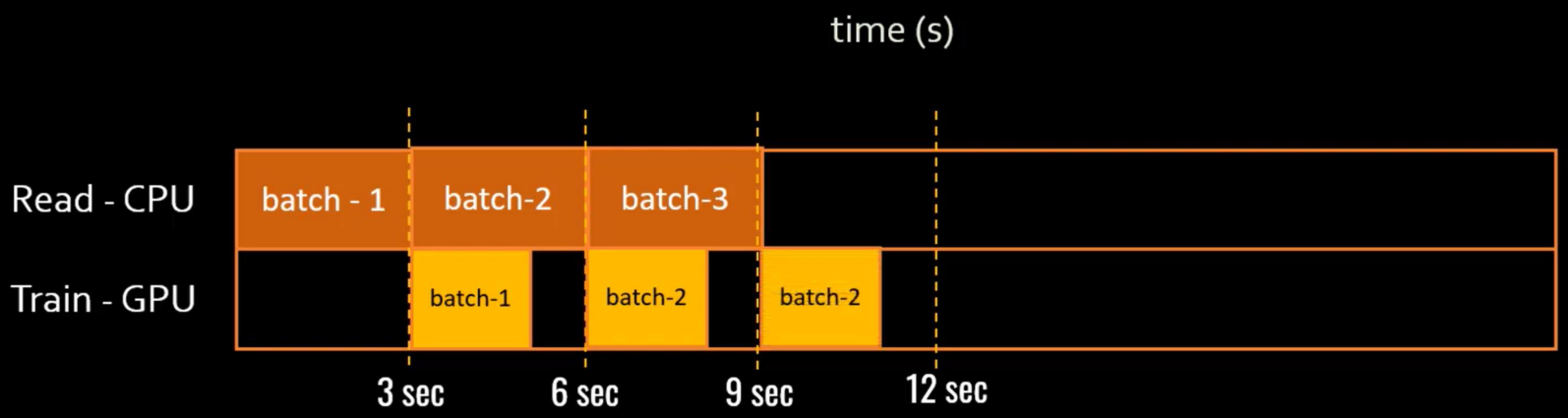

time (s)
Read - CPU
Train - GPU
batch - 1
batch-2
batch-3
batch-1
batch-2
batch-2
3 sec
6 sec
9 sec
12 sec
15 sec

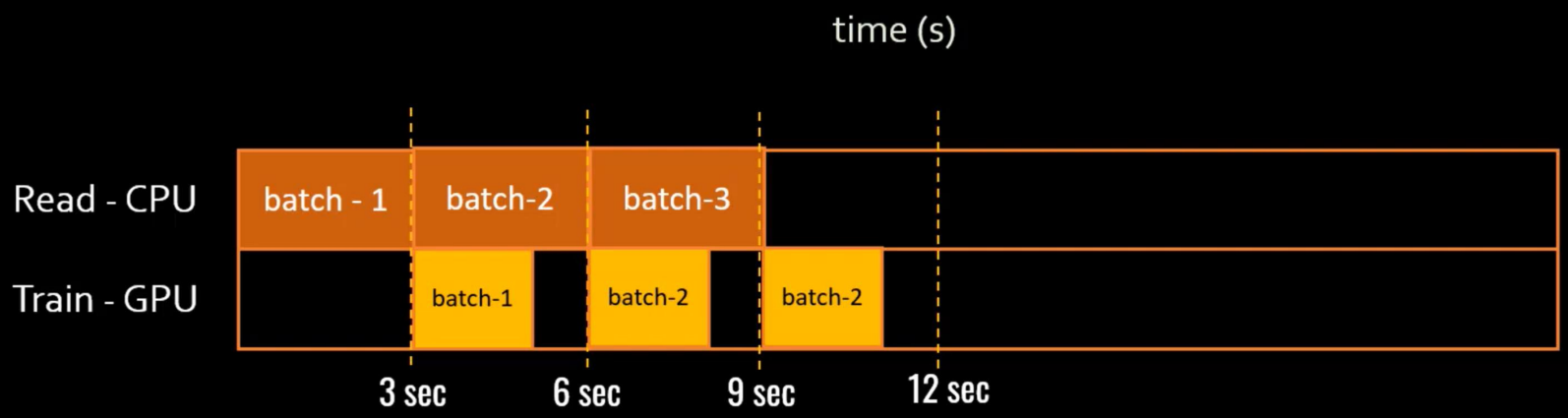

time (s)
Read - CPU
batch - 1
Train - GPU
batch-1
3 sec
6 sec
9 sec
12 sec


time (s)
Read - CPU
batch - 1
batch-2
Train - GPU
batch-1
3 sec
6 sec
9 sec
12 sec

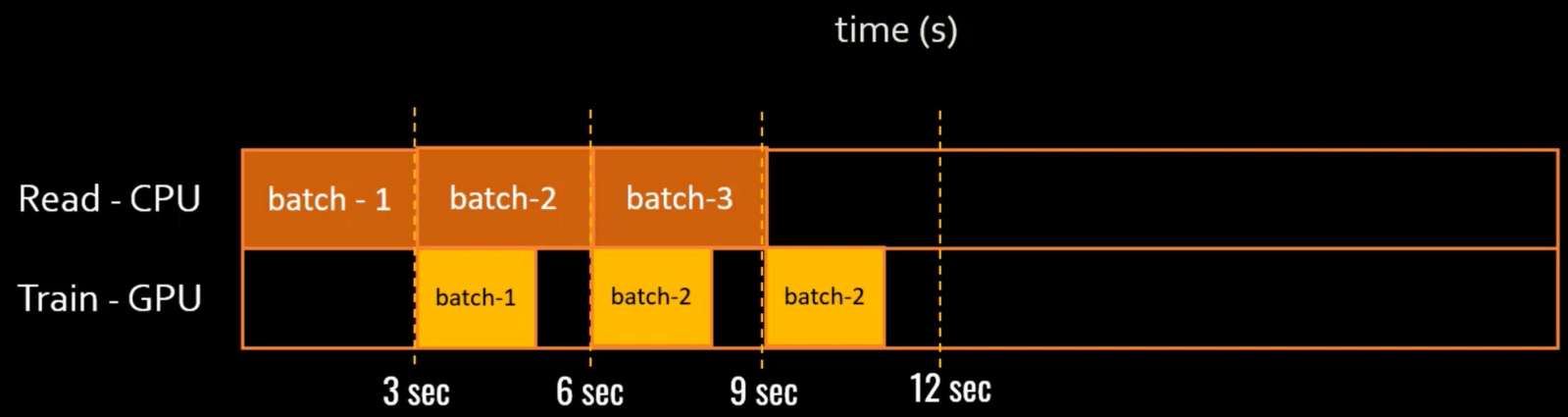

time (s)
Read - CPU
batch - 1
batch-2
Train - GPU
batch-1
batch-2
3 sec
6 sec
9 sec
12 sec


time (s)
Read - CPU
batch - 1
batch-2
batch-3
Train - GPU
batch-1
batch-2
3 sec
6 sec
9 sec
12 sec


time (s)
Read - CPU
batch - 1
batch-2
batch-3
Train - GPU
batch-1
batch-2
batch-2
3 sec
6 sec
9 sec
12 sec



tf.data.Dataset.prefetch(1)

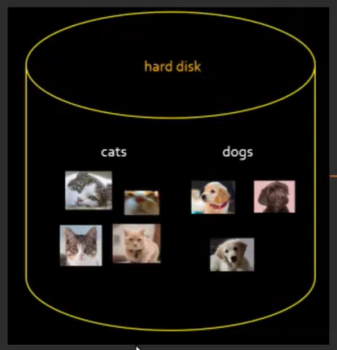


tf.data.Dataset.prefetch(1)

tf.data.Dataset.prefetch(AUTOTUNE)

Step 1: Build Data Pipeline

tf.data.Dataset.list_files('images/*').map(process_img).filter(filter_func).map (lambda x: x/255).prefetch(AUTOTUNE)



Load images from images folder

Convert image content to numpy array. Extract label from folder

Filter Blurred Images

Scaling

prefetch

Step 2: Train the model

```
model.fit(tf_dataset)
```

Image credit: tensorflow official documentation

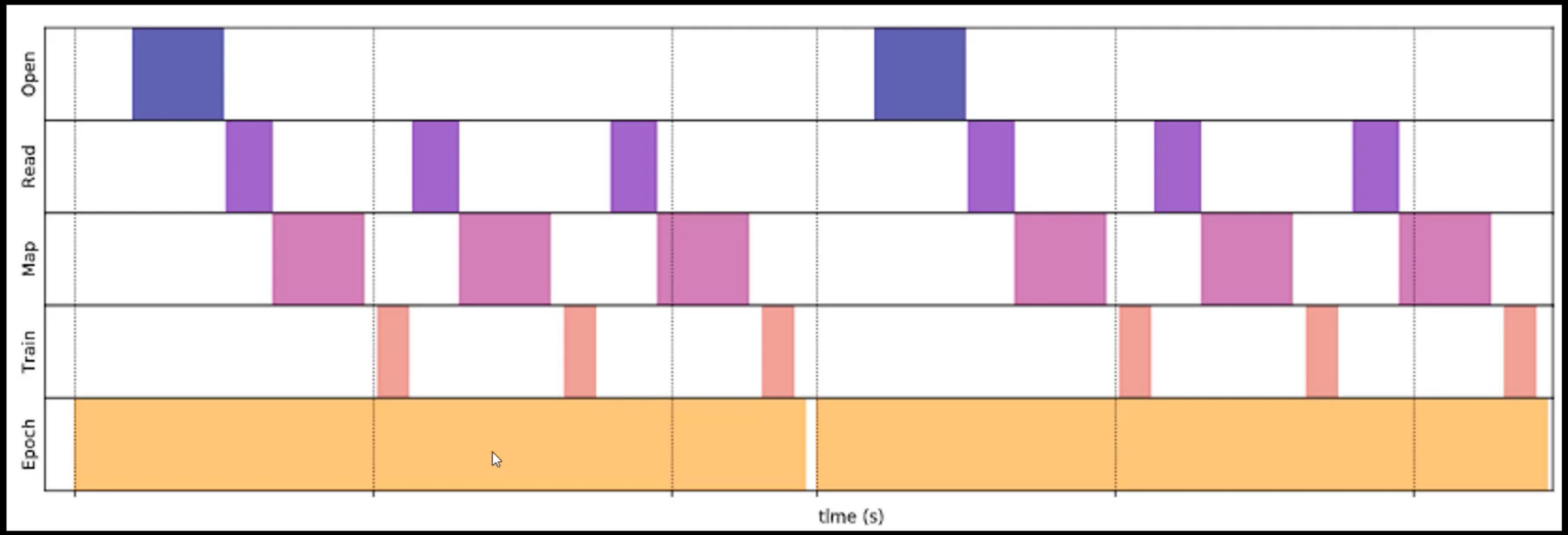


Image credit: tensorflow official documentation

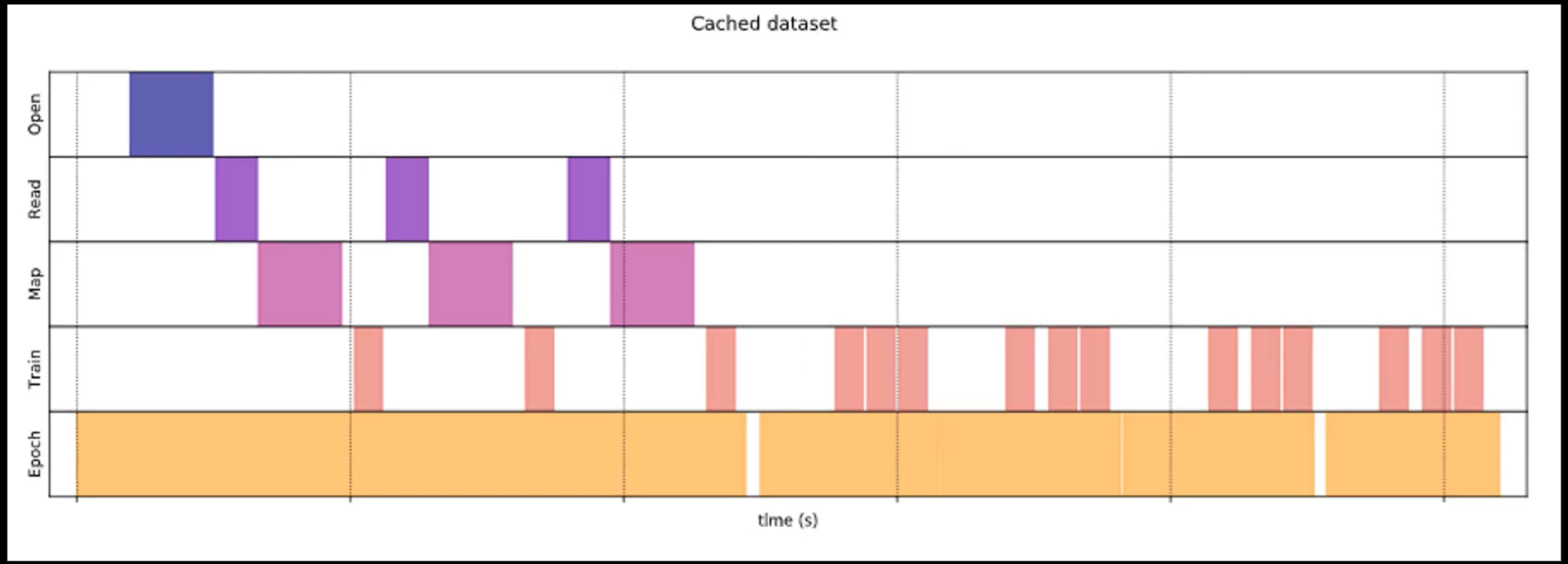
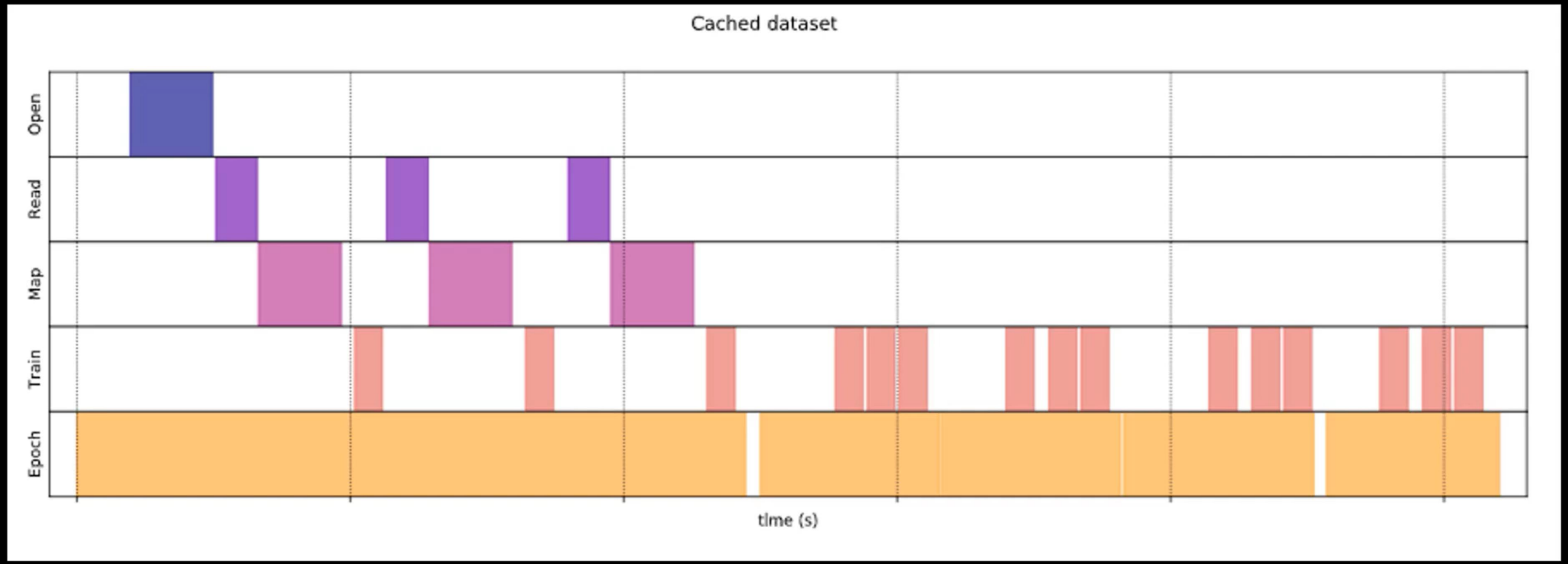


Image credit: tensorflow official documentation



tf.data.Dataset.cache()