

# Thumbnail Generation

## VM Sequential Processing

- Sequentially generate Thumbnails
- 50, 100, 150, 200, 250 images

## VM Multiprocessing

- Generate Thumbnails with the Python Multiprocessing Library
- 50, 100, 150, 200, 250 images

## Lithops Processing

- Generate Thumbnails with the Lithops API
- Supports spawning of hundreds of instances on Cloud Code Engine
- 50, 100, 150, 200, 250 images

**62s**

**Lithops**

**298s**

**VSI**

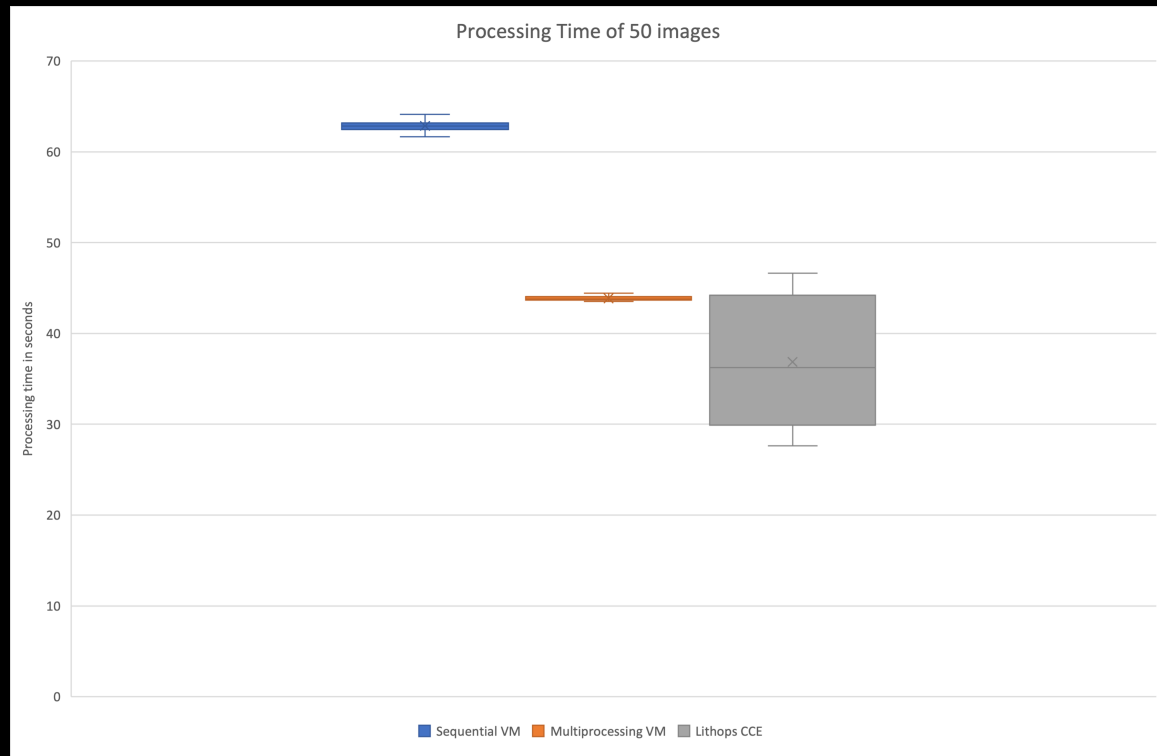
# Processing time of 50 images compared

## VM

- Significantly higher processing time when generating the images sequentially
- Low variance

## Lithops

- Slightly low processing time compared to the VM
- High variance due to slow spawning instances



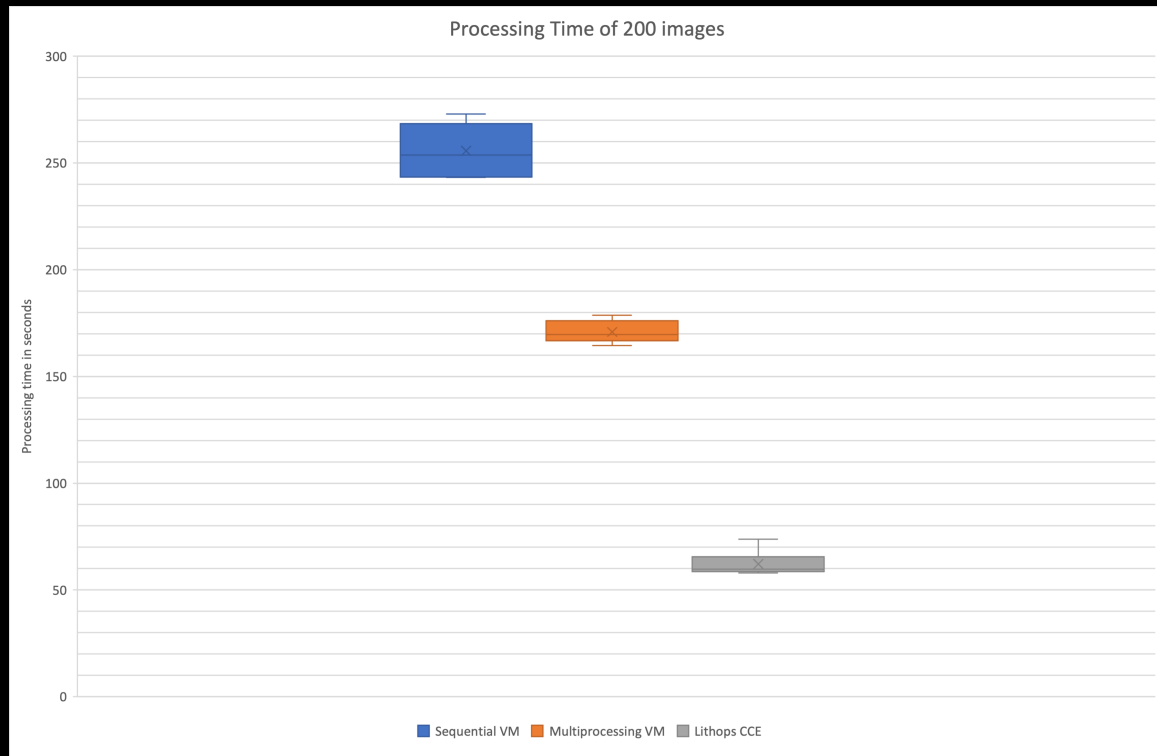
# Processing time of 200 images compared

## VM

- Significantly higher processing time when generating the images sequentially

## Lithops

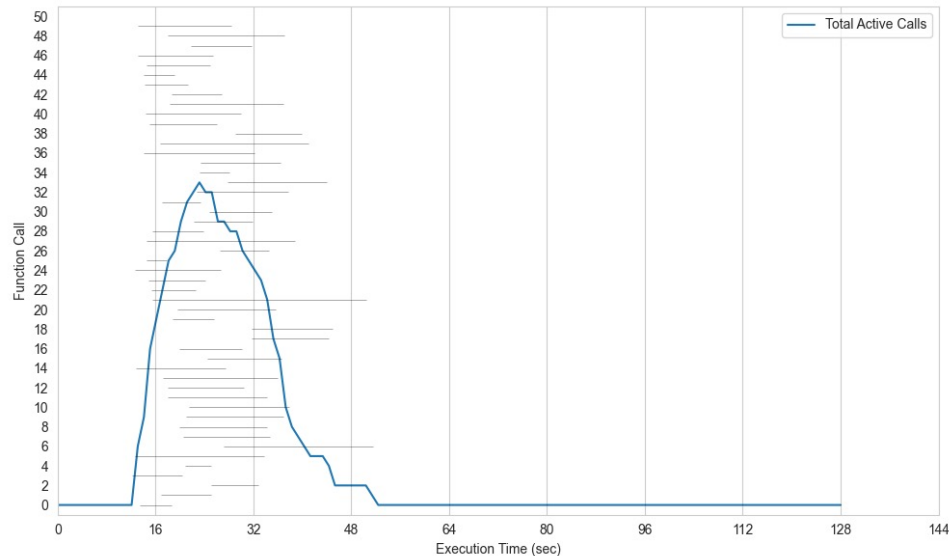
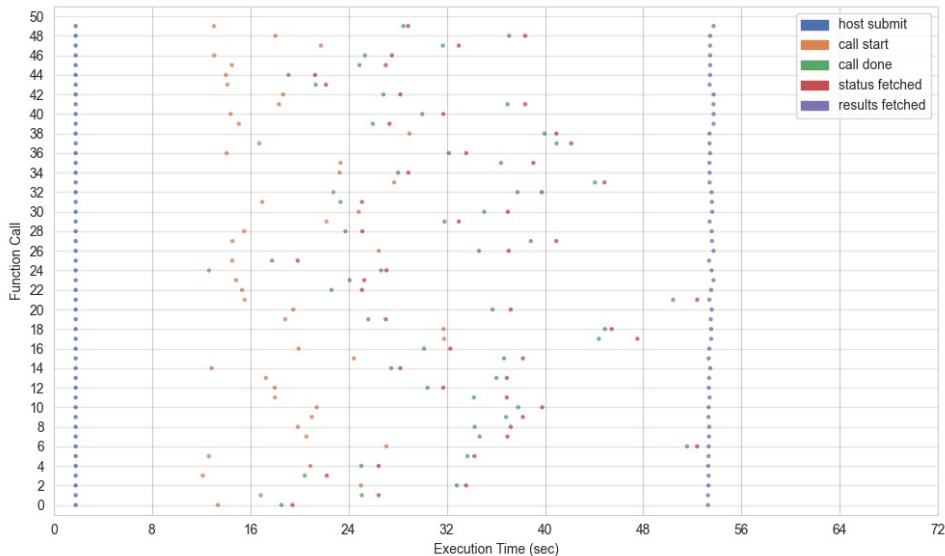
- Significantly lower processing time compared to the VM
- Slightly lower variance



# Example Lithops batch job invocation for 50 images

- Processing time depends on the slowest instance

- Large gap after host submission to cloud code engine



# Scalability compared

## VM

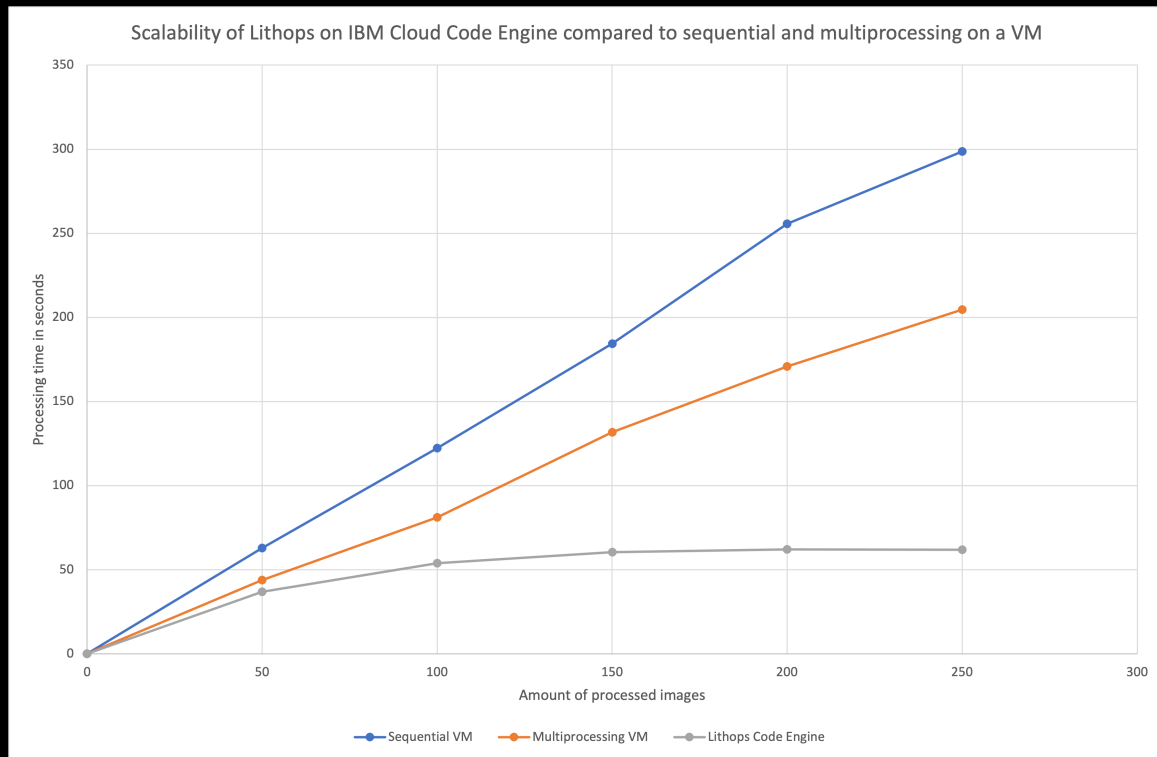
- Linear growth of execution time

## Lithops

- Logarithmic time complexity
- Asymptotically approaching **62 seconds** with:

*$f(x)$  describes the time needed to generate  $x$  thumbnails in seconds*

$$\lim_{x \rightarrow \infty} f(x) \approx 62$$



# Conclusion – What have we learned

CCE performs better at handling compute intensive API requests compared to VSI with same configuration



CCE handles the network load better without needing prior configuration compared to VSI



Using the Lithops API drastically reduces the learning curve of running compute intensive tasks as a serverless batch job



Running compute intensive batch jobs on CCE results in logarithmic growth of processing time per processed item

