CS 340
Computer Systems

#20: Cloud Architectures and Caching

October 31, 2023 · G Carl Evans

Cloud Architectures

Q: What are cloud architectures?

Monolithic Software Architecture:

Characteristics:

A (Small) Monolithic Example: Illinois Open Source Queue
Endpoint: https://queue.illinois.edu/
Source Code: https://github.com/illinois/queue

Microservice Software Architecture:

Characteristics:

Microservice Example:
PiggyMetrics by Alexander Lukyanchikov
GitHub: https://github.com/sqshq/PiggyMetrics

Serverless Software Architecture:

Characteristics:

Serverless Examples:

adoptable-pet-bot/serverless.yml
https://github.com/lynnaloo/adoptable-pet-bot/blob/master/serverless.yml

```
provider:
  name: aws
  runtime: nodejs4.3
  region: us-east-1
…
functions:
  tweetPet:
    handler: handlers/tweetPet.tweetPet
    description: Tweets Adoptable Pets on a Schedule
    memorySize: 512
    timeout: 10
    events:
      - schedule: rate(6 hours)
```

emojibot/serverless.yaml
https://github.com/markhobson/emojibot/blob/master/serverless.yml

```
provider:
  name: aws
  region: eu-west-1
  stage: dev
  runtime: nodejs14.x
  memorySize: 128
…
functions:
  event:
    handler: src/handler.event
    events:
      - http: POST /event
  explain:
    handler: src/handler.explain
    events:
      - http: POST /explain
```

serverless-image-labeller/serverless.yml
https://github.com/nileshprasad137/serverless-image-labeller/blob/master/serverless.yml

```
provider:
  name: aws
  logs:
    restApi: true
  runtime: python3.7
…
functions:
  labelOnS3Upload:
    handler: handlers/S3UploadHandler.labelOnS3Upload
    events:
      - s3:
          bucket: ${self:provider.environment.SERVERLESS_IMAGE_LABELLING_BUCKET}
          event: s3:ObjectCreated:*
          existing: true
  getImagesByLabel:
    handler: handlers/getImagesByLabelHandler.getImagesByLabel
    events:
      - http:
          path: getImagesByLabel
          method: post
          cors: true
```

Q: What is caching?

...have we talked about caching before?

Q: Why is caching important on a cloud-scale?

Caching can occur at multiple different places – and we have different strategies and techniques for each "layer" of the full caching strategy.

[Users] **[Origin]**

[1]:

[2]:

[3]:

Caching Technique #1: Age-Based Caching

Idea + Purpose of Age-Based Caching:

Age-Based Caching Use Case:

- The CS 340 course websites the Bootstrap CSS library from stylizing the front-end content.
- When we load the page from a browser that has never visited the CS 340 website, we see the following requests:

GET https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/js/bootstrap.min.js

```
GET /bootstrap/4.2.1/js/bootstrap.min.js HTTP/1.1\r\n
[...]
```

HTTP Response:

```
1 HTTP/1.1 200 OK\r\n
2 date: Wed, 02 Nov 2022 18:37:00 GMT\r\n
3 age: 6156514\r\n
4 cache-control: public, max-age=31919000\r\n
5 cdn-cache: HIT\r\n
6 cdn-cachedat: 2022-03-12 17:14:47\r\n
… [...]
```

[Line 3]: The `age` header:

[Line 4]: The `cache-control["max-age"]` header:

Q: What happens when we request a page using this resource again before the content expires?

....how much traffic is saved?

...how much time is saved?

...what are disadvantages of age-based caching?