

## What benefits/problems come from deploying services in Serverless functions?

Benefits:

- **Improve develop efficiency:** Serverless computing does not mean developers work without any server. It instead allows them not to focus on issues related to server-based architectures. Therefore, developers can pay more attention to the logic for processing client requests, which can improve efficiency of software development.
- **Decrease develop cost:** Past IDC and modern cloud architecture, which normally uses a monthly charging mode, can result in the situation that they still charge even if no current users or no applications are deployed. Contrastingly, Serverless architecture often uses a Pay-as-you-go mode, which allows only paying infrastructure provider for resources they actually used, and not pay for idle capacity. For example, with AWS<sup>1</sup> developers pay only for the individual services, for as long as using periods, and without requiring long-term contracts or complex licensing, which have a potential savings range from 99.8% to 99.95%[1].
- **Reduction in lead time:** Packaging and deploying a Serverless function are simple compared to deploying an entire server. All server-related works that used to be allocated to architects are handed to infrastructure provider, which is fast, and saves workforce. A fully Serverless solution requires zero system administration. For example, no Puppet/Chef, no start/stop shell scripts, and no decisions about whether to deploy one or many containers on a machine[2]. Easier operational management makes it use less time to market.

Problems:

- **Startup latency:** Empirical evidence suggests Serverless architecture usually do long time preparation before going into runtime. The latency introduced by multiple hops, and tail-effects also need to be taken into consideration[1], as rapidly increasing number of request will result in time out. AWS has improved this area over time, but there are still significant concerns.
- **Hard to debug:** It is difficult to troubleshoot problems because implementation logic is almost everywhere and a simple operation may cause thousands of Lambdas to execute. There has been some progress, mostly related to running functions locally, or in line with the testing updates. AWS X-Ray<sup>2</sup> and CloudWatch<sup>3</sup> can help developers monitoring and managing.
- **Restrictions from vendors:** Serverless increase market monopoly that was already severe in Cloud architecture. As our Serverless features are provided by third-party vendors, restrictions from vendors are inevitable. For example, lack of system control may manifest as system downtime, unexpected limits, cost changes, loss of functionality, forced API upgrades, etc.[2]

We can learn from a case study from MindMup (a commercial online mind-mapping application) to see the cost saving benefits of Serverless. Between February 2016 (running on Heroku) and February 2017 (running on Lambda), the number of active users supported by the platform grew by more than 50%, but the hosting costs dropped by slightly less than 50%[1].

## References

- [1] A. Colyer, **Serverless computing: economic and architectural impact** <https://blog.acolyer.org/2017/10/19/Serverless-computing-economic-and-architectural-impact/> Date Accessed: 21 February 2019
- [2] M. Roberts, **Serverless Architectures** <https://martinfowler.com/articles/Serverless.html> Date Accessed: 22 February 2019

---

<sup>1</sup><https://aws.amazon.com/pricing/>

<sup>2</sup><https://aws.amazon.com/xray/>

<sup>3</sup><https://aws.amazon.com/cloudwatch/>

Part II:

Download .csv

	User	Access key ID	Secret access key	Email login instructions
▶	serverless-admin	AKIAJTNA7ZW4ONI3JLIA	6wNgIX9VWoHRehTqqNATH Cc9Ca7vCba6VIfkuYsK Hide	Send email ↗

this is the image that we get the secret key and access ID

```
(Math) jinweizhang@dyn3160-134:~/Desktop/475-software-engineering/topic6/lambda-test2$ claudia create --name lambda-test2 --region us-east-2 --api-module server
packaging files npm install -q --no-audit --production
npm WARN lambda-test@1.0.0 No description
npm WARN lambda-test@1.0.0 No repository field.

saving configuration
{
  "lambda": {
    "role": "lambda-test2-executor",
    "name": "lambda-test2",
    "region": "us-east-2"
  },
  "api": {
    "id": "iaz6gqhqr",
    "module": "server",
    "url": "https://iaz6gqhqr.execute-api.us-east-2.amazonaws.com/latest"
  }
}
```

this is the json file that shows up when we succeed in deploying the application onto amazon aws

You are not subscribed to this object.

Edit User Subscription

Callback URL

https://iaz6gqhqr.execute-api.us-east-2.amazonaws.com/latest

Verify Token

dhaifofho38hoj

☐ No

 Include Values

Cancel

Verify and Save

this is the page callback URL and token

Token Generation

Page token is required to start using the APIs. This page token will have all messenger permissions even if your app is not approved to use them yet, though in this case you will be able to message only app admins. You can also generate page tokens for the pages you don't own using Facebook Login.

Page

Jinwei Zhang

Page Access Token

EAADWg9wJZCWABAMktfudUvNCjoYvvo5jyX6Yt6C4ZAKvI4ZBWympTZBoKMig8IXcZCNZBAhkuDaA4LdxhQ7Gs3J9Z

MSC in Advanced Computing - SprL...

493 (1 unread)

pengzhe-bot-test - Dashboard - Fa...

(1) Jinwei Zhang - Home

IAM Management Console

Hello World AWS Lambda function

claudiajs/claudia: C

"475 software engineering for industry is so good"

a simple sentence that called from server.