.NET Core 2 In AWS Lambda

Drew Carpenter

History of .Net Core

- November 2014 .NET Core is open sourced as .NET Core 5
- November 2015 .NET Core 5 RC 1
- January 2016 .NET Core 5 renamed to .NET Core 1.0
- **June 2016** .NET Core 1.0 released
- September 2016 .NET Standard released
- November 2016 .NET Core 1.1 released
- **December 2016** .NET Core 1.0 released to AWS Lambda
- July 2017 .NET Standard 2.0 released
- **August 2017** .NET Core 2.0
- **January 2018** .NET Core 2.0 released to AWS lambda

Cold starts

.Net Core 1.0 on Lambda had an issue with cold start times

.Net Core 2.0 on Lambda reduces cold start times but they are still excessive

Increasing Memory limit will has a inverse correlation to startup time.

http://theburningmonk.com/2017/06/aws-lambda-compare-coldstart-time-with-different-languages-memory-and-code-sizes/

https://medium.com/@mendoza.paul/cold-start-example-for-my-net-core-2-0-aws-lambda-function-f03ab17250fd

The dreaded .Net Standard 1.6.1

- .Net Standard >=2.0 no longer requires a netstandardlibrary package.
- Only supports Nuget package Microsoft.AspNetCore.All Version 2.0.3
- AWS has a history of long lag between an LTS and supporting the new language version.

COSTS

- Lambda
  - First 1,000,000 requests free
  - $0.20 per 1,000,000 requests
  - 400,000 GB seconds of compute time free
  - $0.00001667 per GB Second
- API Gateway
  - $3.50 per million API calls received, plus the cost of data transfer out, in gigabytes.
  - $0.09/GB for the first 10 TB

Where is the SDK?

- Visual studio https://aws.amazon.com/visualstudio/
- .NET Core CLI `dotnet new -i Amazon.Lambda.Templates::*`
- Nuget Tools
    - Amazon.Lambda.Tools version 2.1.1
    - Amazon.Lambda.AspNetCoreServer version 2.0.1

DEMO

Knock on wood

https://github.com/Blue-speed/exif.viewer