caffeinated by sonatype

NOVEMBER 10, 2022

Performance tuning an Azure App Service with K6

Jose Luis Latorre, Swiss Life

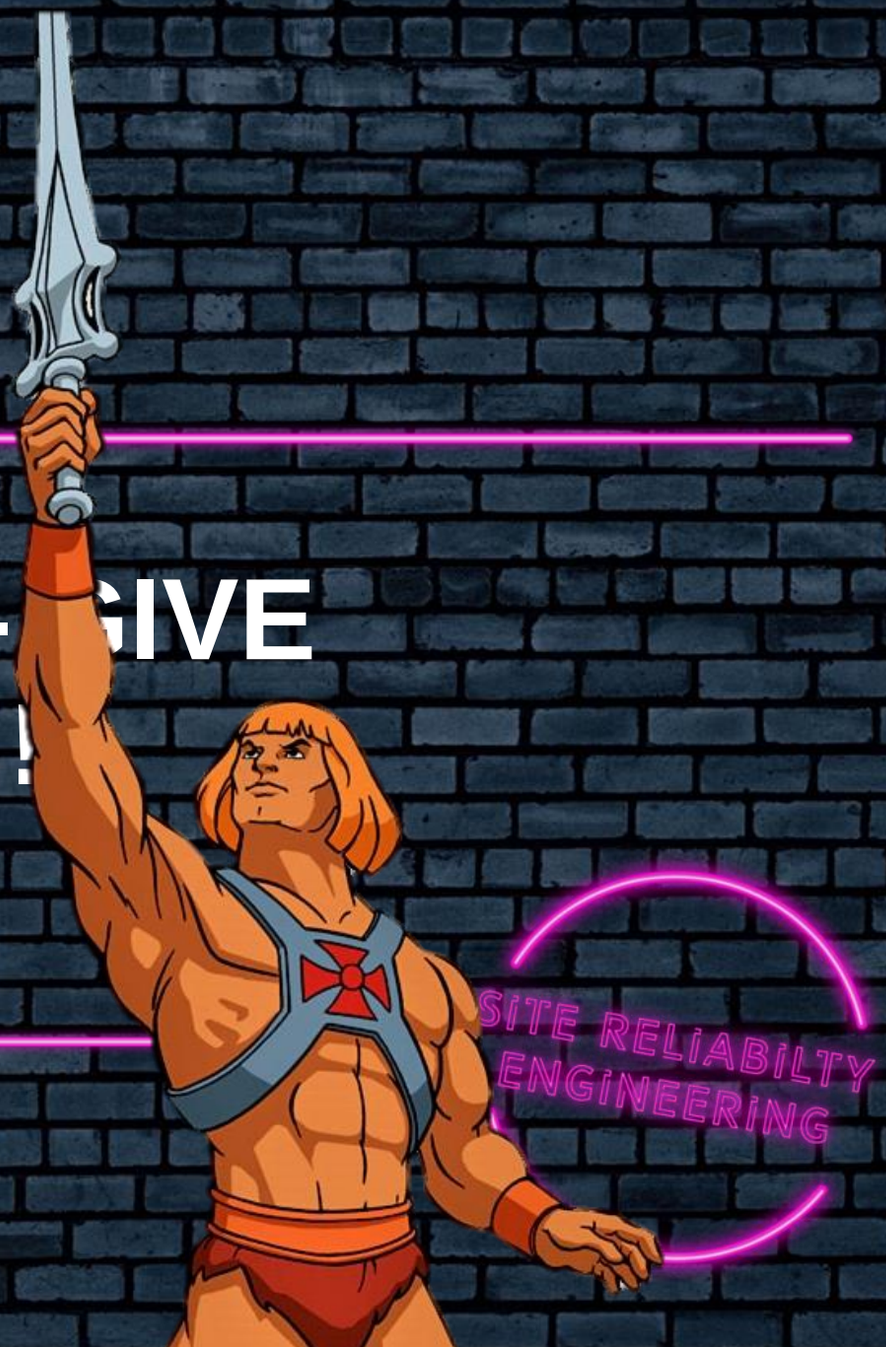
ALL DAY DEVOPS

caffeinated by sonatype

NOVEMBER 10, 2022

I WANT THE POWER – GIVE ME THE POWERRRR!!!

With Azure App Service & K6



Just Jose ;)

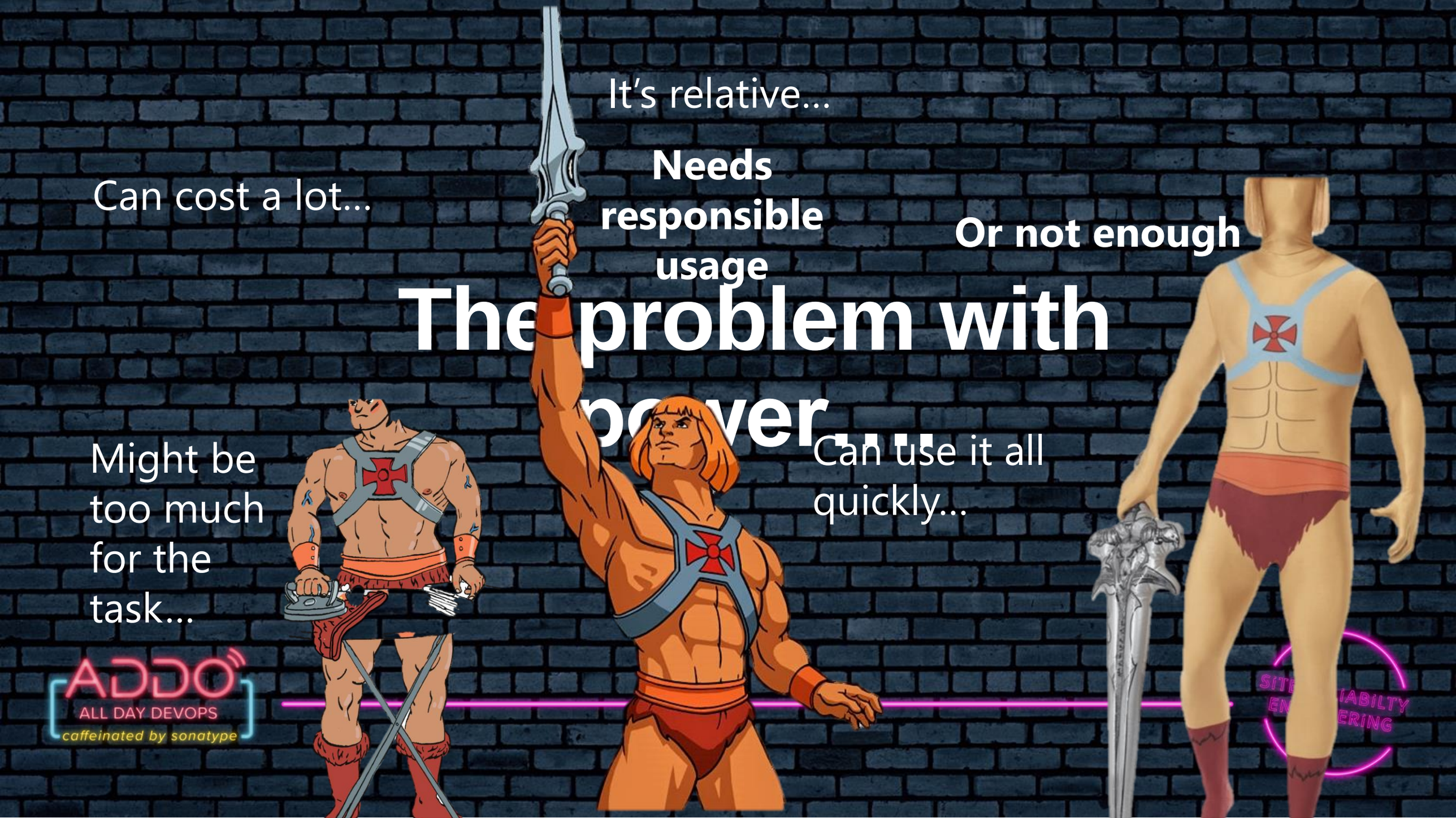
Jose Luis Latorre
@joslat - https://github.com/joslat
Software architect & DEV Community Lead
OSS, Sharing knowledge & Passion for tech

MVP Microsoft Most Valuable Professional

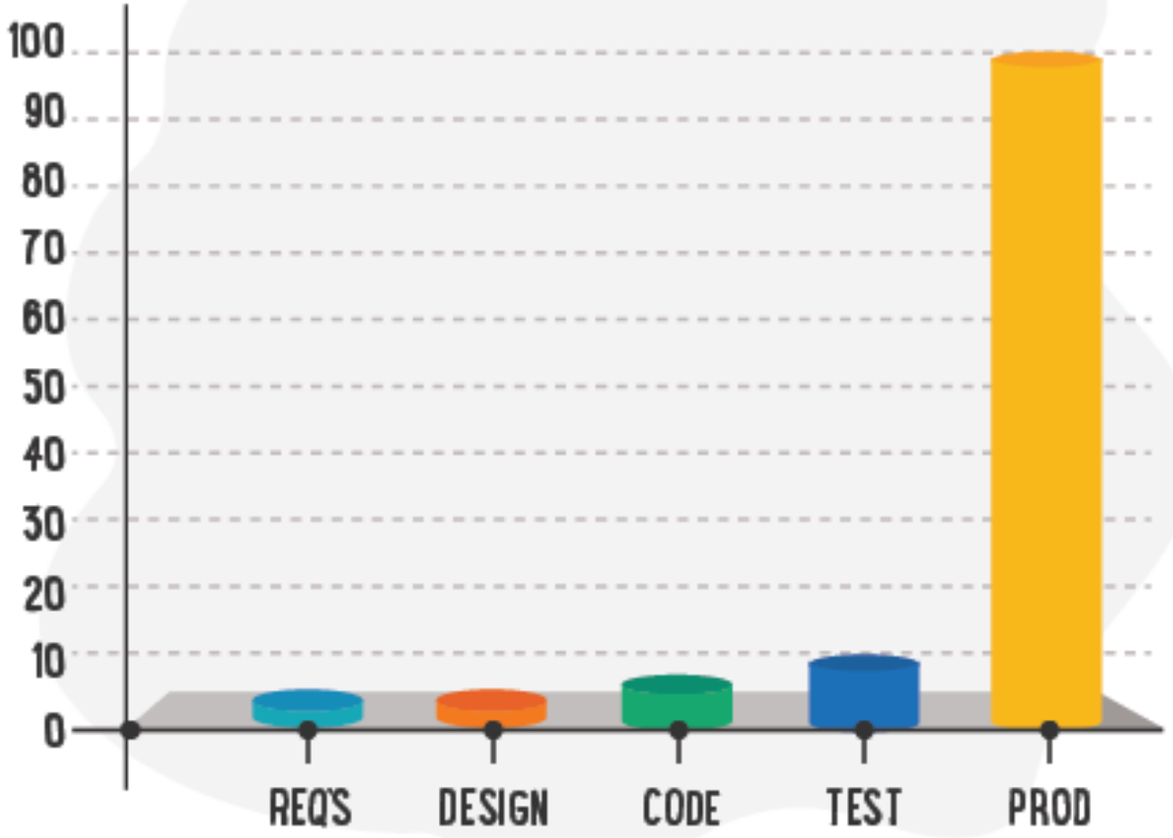
.NET User Group Zürich
Lead

ADDO ALL DAY DEVOPS caffeinated by sonatype

SITE RELIABILTY ENGINEERING


It's relative...
Needs responsible usage
Can cost a lot...
Or not enough
The problem with power...
Might be too much for the task...
Can use it all quickly...
ADDO
ALL DAY DEVOPS
caffeinated by sonatype

THE RELATIVE COST OF FIXING DEFECTS



ILLUSTRATED BY SEGUE TECHNOLOGIES

Agenda

- Understanding power
- Measuring w K6
- The fastest API on earth
- Measuring it
- The Lifecycle!
- Lessons learned and “The Power” Handover

Understanding power

- How fast can it go?
- What are the SLA?
- Set up the SLA!
- How much memory? How much data? CPU?
- (continued…) networking, variability, where are they…
- How many variability in users?
- It’s too complex! Let’s not do it…

Understanding power

How fast…?

Response time

Where are they?

How Strong?

How many users?

How big are the request and response payloads?

How complex is the process?

How many at the same time…?

What users?

What variability? And spikes?

Understanding power needs

GIVE ME THE SLA

Mean speed for a number of users and its variability

Understanding power needs

Usually “the usual”

They don’t know, so is not set

But they will come and complain anyway

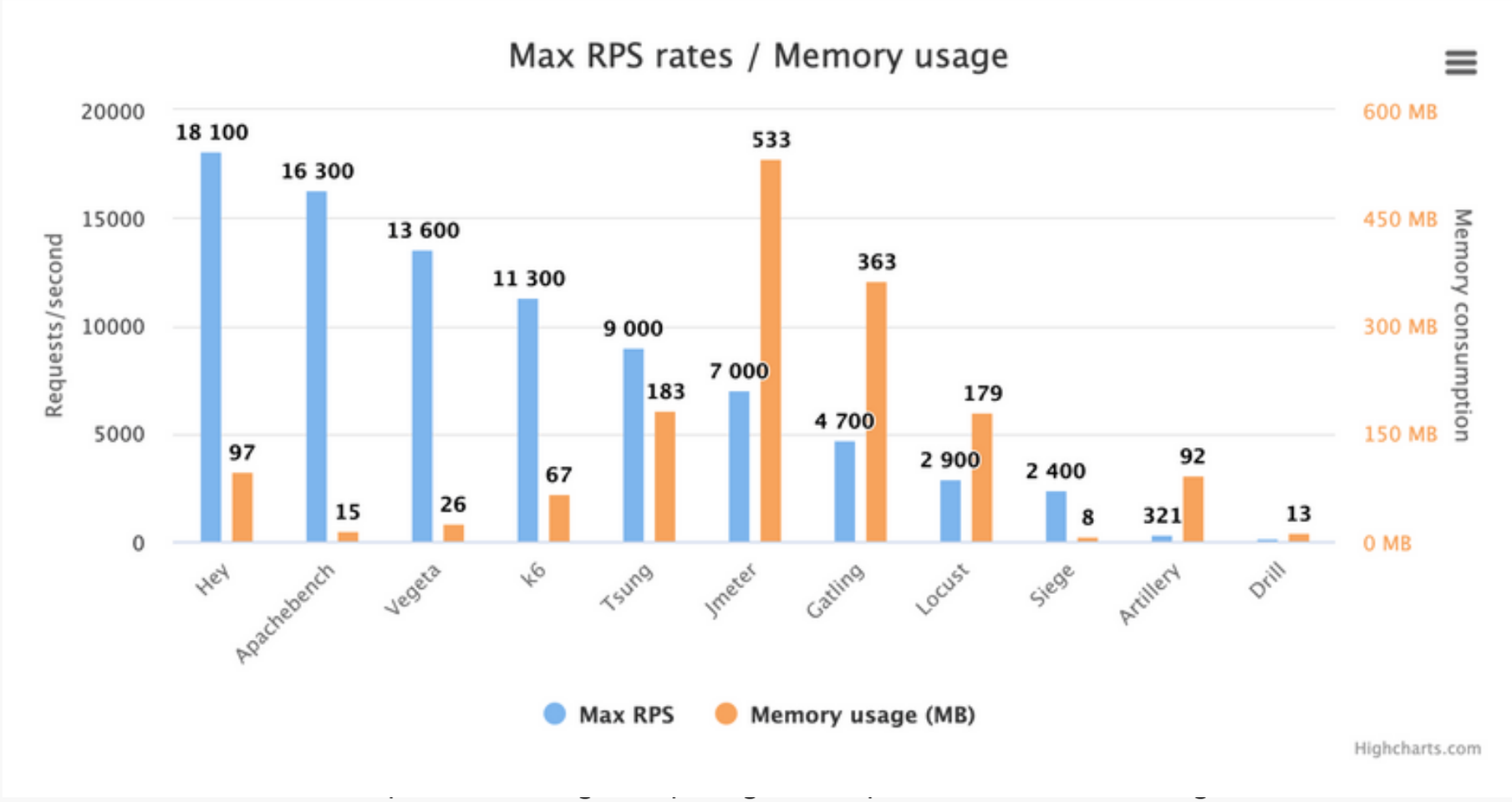
Solution: You set up the SLA!

According to budget and infrastructure, you push back and state the rules. If they want more, more time, budget will be needed and maybe compromises will need to be made.

Under 200-800 ms x web API, 50 internal API, <10ms critical API

Web MUST load in 1 to 3 seconds max!

Why K6? Performance & features!!



https://k6.io/blog/comparing-best-open-source-load-testing-tools/

Features!

- Code Based! Great for devs (like me)
- Performant! (important in a perf tool...)
- Open Source
- You can define any load, how you want
- Cloud & DevOps Friendly
- Extensible

ADDO
ALL DAY DEVOPS
caffeinated by sonatype

SITE RELIABILTY ENGINEERING

Where to start?

http://K6.io or watch one of my introduction to k6 talks ;)

ADDO
ALL DAY DEVOPS
caffeinated by sonatype

SITE RELIABILTY ENGINEERING

ALL DAY DEVOPS

caffeinated by sonatype

NOVEMBER 10, 2022

The API & The Test & Environment

I did it! Fastest ever!

- Did I? really?
- Simple, no issues, get request and respond “something” ;)
- A simple API wo problems, based on `https://docs.microsoft.com/en-us/azure/app-service/app-service-web-tutorial-rest-api`
- Stepping back…

Not too much to optimize right?

```
namespace WebApi01.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    0 references
    public class simpeApiCall : ControllerBase
    {
        [HttpGet]
        0 references
        public string Get()
        {
            return "You got this!";
        }
    }
}
```

Let’s see it…

…in Local

How fast? Let’s test it!

The Environment

Somewhere, in a Swiss Data Center...

Virtual Network

NSG

Bastion

Test VM

Private Endpoint

App Service (API)

DNS

ADDO ALL DAY DEVOPS caffeinated by sonatype

SITE RELIABILTY ENGINEERING

K6 Cloud
Tracking, baseline & more…

Fine Tuning

Code

- Make it Asynchronous
- Optimize
- Cache all-that-you-can / Preload
- Use the right .NET stack ;)
- Profile!
- Set up Azure Application Insights

App Service

- Avoid latency (batching, proximity, CDN, traffic manager, use Redis cache, etc…)
- http://azurespeedtest.azurewebsites.net/
- Use “Always On” to avoid app recycle
- Use app service advisor
- Http 2.0
- Turn off App Request Routing Cookie

How fast? Let's test it!
(in the right environment)

Take away & conclusions

- Use the right environment (latency, cache, optimize)
- Follow the flow – be water!
  - Measure
  - Baseline
  - Fine Tune
  - Repeat!
- Use the right tool K6 or the venom of your choice ;)
- I am thinking of doing a more detailed session and a variation with Containers, let me know if you'd like that!
- Code & slides: github.com/joslat/AzureAPiPerformanceTesting

SCAN ME

ADDO
ALL DAY DEVOPS
caffeinated by sonatype

SITE RELIABILTY ENGINEERING


ADDO
ALL DAY DEVOPS
caffeinated by sonatype