Open in app

Antony Melvin

Follow 102 Followers About

Starting AWS architecture: auto-scaling

Antony Melvin May 2, 2019 · 3 min read

Auto-scaling is a key AWS Cloud (or any Cloud) architecture principle. It is the process of elastically & dynamically changing the number of servers available for an application based on the users demands. I'm relatively new to cloud architecture, so I'd be delighted with any feedback on whether I have got things right (or wrong for that matter!).

This is a sanitized version of a proposal we've made to improve the user experience at a client while simultaneously lowering costs by using auto-scaling.

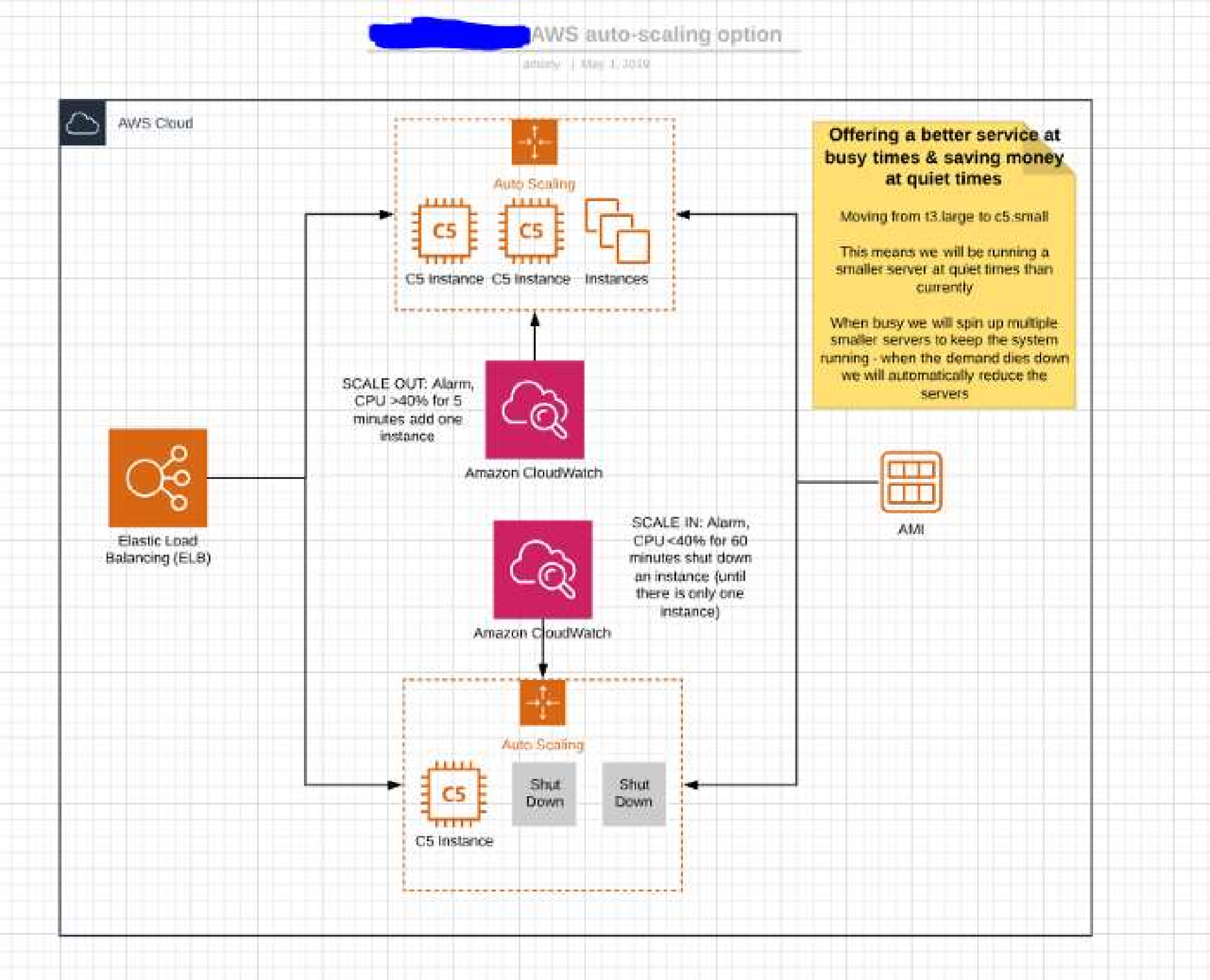
The client as part of their operations have teams in different parts of the world. As a result there are some servers that are heavily utilized for different core hour sets.

We proposed switching from a large server instance to a fleet of smaller instances & utilizing load balancing & auto scaling to optimize both user experience & cost.

This should greatly improve user experience — as the large server was grinding to a halt at times — but we're unsure until we test usage patterns if this will result in any more than a modest nett fall in AWS billing. The load balancer ($30) will be a new billable item, AWS auto-scaling is a free item.

The basis of the proposal was to make the architecture more complex (a single server being used is pretty simple!) but no more costly & to deliver greatly enhanced user experience. The client had previously scaled up servers (moved to bigger ones) to try to alleviate the usage problems. But in tech terms demand expands to fill the available supply & all that.

Cloudwatch to monitor for occasions when CPU maxes out over a prolonged period & spin up new instances (based on pre-built Amazon Machine Instances (AMIs)) when the alarms are raised. Before shutting down server instances to save money when demand falls.



Offering a better service at busy times & saving money at quiet times

Moving from t3.large to c5.small

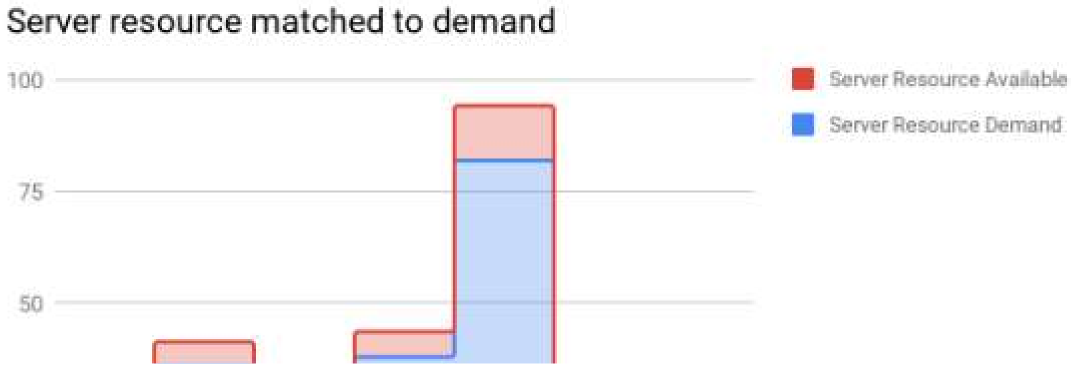
This means we will be running a smaller server at quiet times than currently

When busy we will spin up multiple smaller servers to keep the system running - when the demand dies down we will automatically reduce the servers

SCALE OUT: Alarm, CPU >40% for 5 minutes add one instance

SCALE IN: Alarm, CPU <40% for 60 minutes shut down an instance (until there is only one instance)

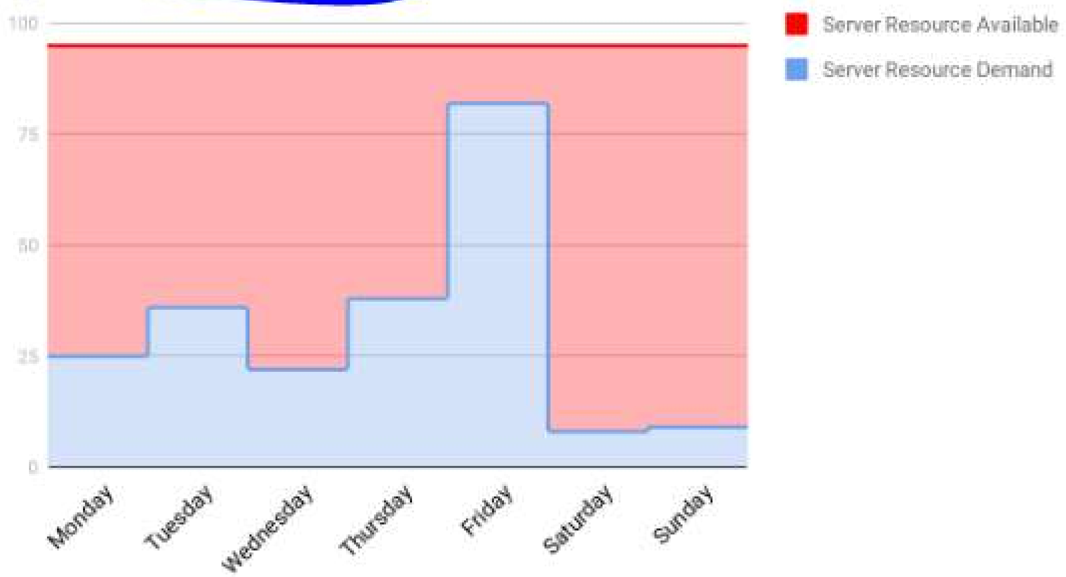
Using auto-scaling will allow us to use smaller server instances so that at quiet times there will be a lower bill & at busy times there will be better user experience.



CPU Utilization when auto-scaling to one or more small servers (made up numbers!)

This contrasts with the standard scale up approach where a server instance keeps getting bigger in an attempt to provide service at all times, but at quiet times the server basically idles along.

[redacted]



Single large server CPU utilization (made up numbers!)

Thanks for reading this small foray into AWS architecture.

I am the APN Alliance lead for Agidea (www.agidea.uk), a technology consultancy based in Manchester with several years of experience developing .Net applications on AWS. I've been tasked with getting Agidea promoted to the Select Partner Tier over the next three month & we're on target!

I have some AWS certifications & accreditations & I'm working towards a Solutions Architect certification:

AWS Certified Cloud Practitioner
AWS Accredited Business Professional

This was a sanitized version of a proposal to improve the user experience at a client while simultaneously lowering costs by using auto-scaling. Hope it helps.

AWS Aws Auto Scaling Autoscaling Load Balancing