

Remote Build Execution

How custom would you like it?

Build Meetup Munich, 2025-04-07

Sara Adams, Founding Engineer @ [EngFlow](#)



Raise your hand
if you **agree** with the following statement

The **more** configurable the **better**

Upsides of configurability **for the user**

Allows you to:

- have more control over the behavior & avoid surprises
- foster familiarity and skill with used tools

→ get **exactly** what you want

Downsides of configurability **for the user**

Need to:

- understand each option
- understand the interplay between each option
- keep each option up-to-date

→ steep learning curve & high maintenance cost

→ worse experience for the user

Downsides of configurability **for the service**

Need to:

- implement & maintain configurable features
- avoid conflicts between configurations
- support snowflakes (each customer is unique)

→ slower development & harder debugging

→ worse experience for the user



Raise your hand
if you **agree** with the following statement

The **less** configurable the **better**

but of course...
It depends.

Configurability

Automation

Quotas

How many builds per time unit?

Worker Scaling

How many workers?

Priorities

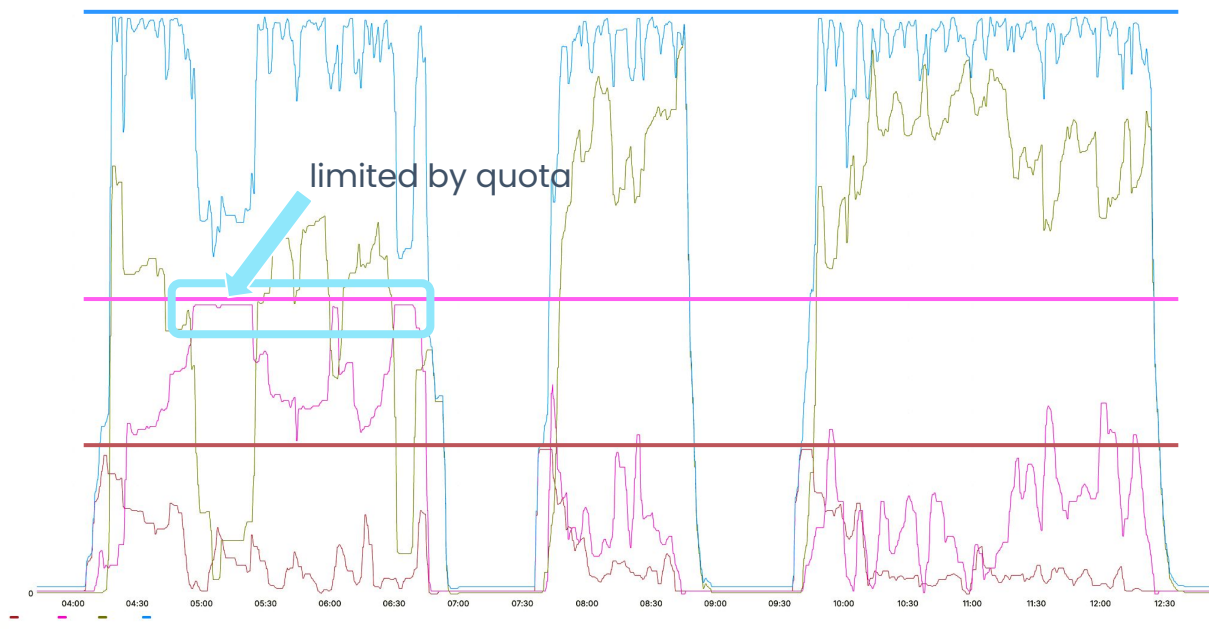
Which builds first?

Pool Selection

What kind of workers?

Quotas

Used Cores



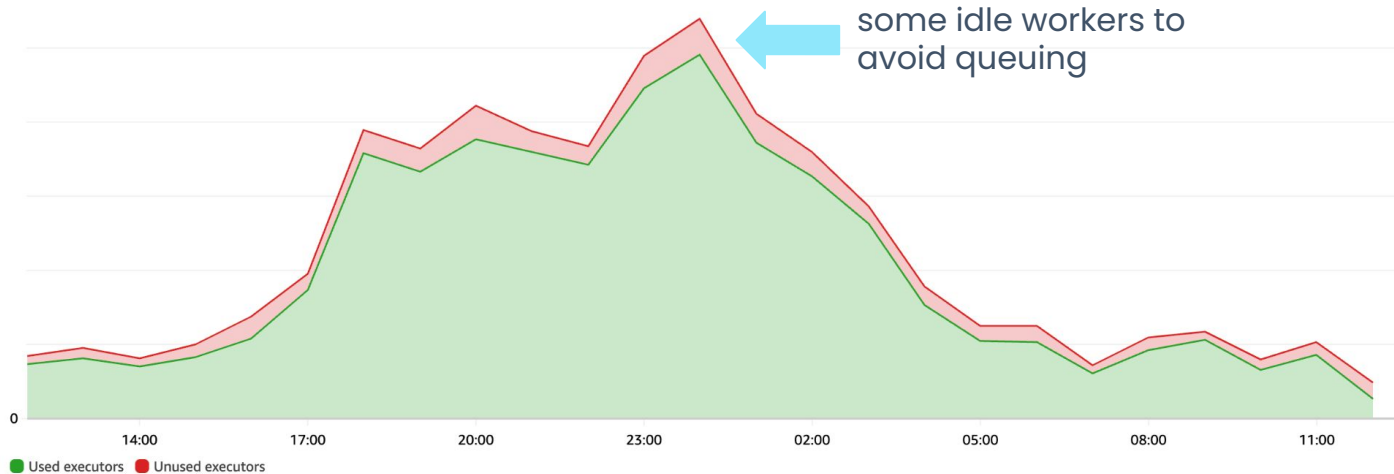
Priorities



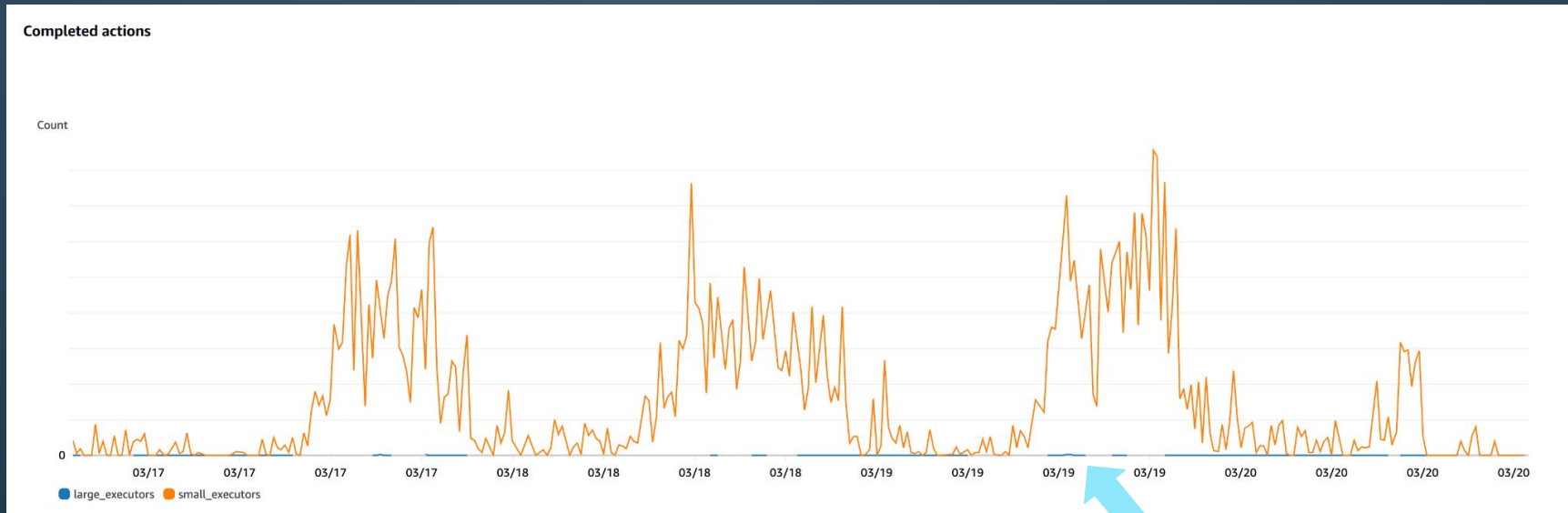
Worker Scaling

Executor Utilization

Number of executors



Pool Selection



only 0.1%
of all executed actions
need the large workers

Thanks for those four examples,
but...

Should feature x be highly configurable?

How often does it change?



How specific is it to the build?





Raise your hand
if you have a question