Using Platform Engineering to Raise the Complexity Limit

ANDREW BOYAGI | SENIOR EVANGELIST | LINKEDIN.COM/IN/ANDREWBOYAGI/

Deliver high-quality software, fast.

What's driving this diversity in performance?

Stage 1
Let's play a game

Memory game

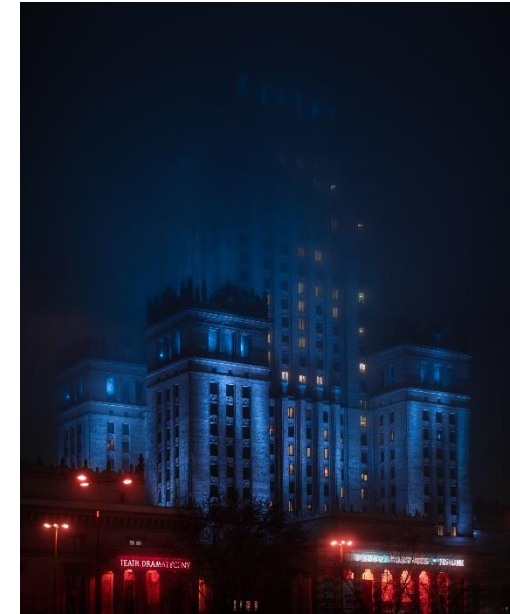| Who	What	Where
Batman	Superhero	Gotham

Who

Andrei Bocan

Mike Cannon-Brookes

What

Principal engineer

Co-CEO

Where

San Francisco

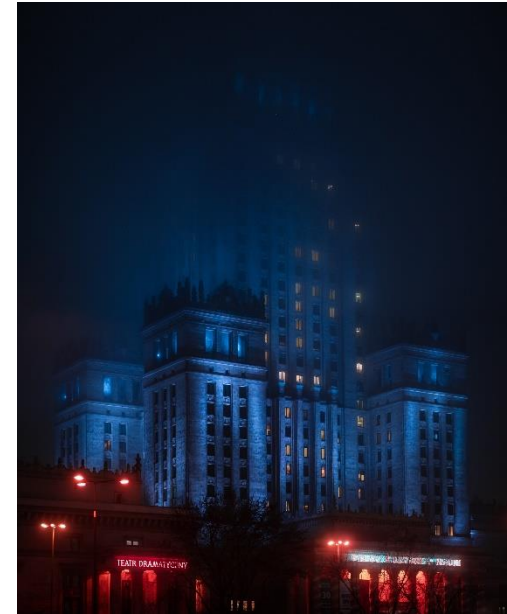Sydney

Stage 1

Where does Batman live?

Who

Batman

What

Superhero

Where

Gotham

Stage 2

Who

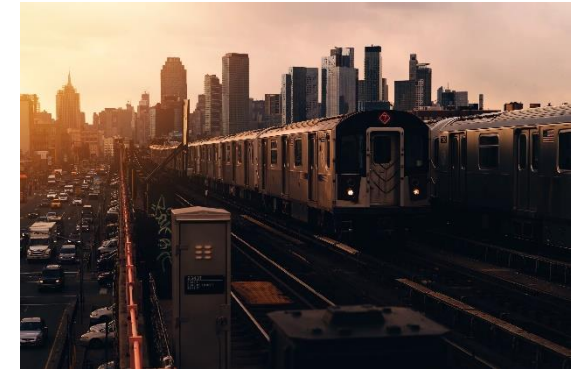## Spiderman

Buzz Lightyear

What

Student

Toy

Where

Queens

Andy's House

Who

Grogu

Mickey

What

Unknown

Wizard

Where

The Razor Crest

Disneyland

Stage 2

What color was the symbol on Mike Cannon-Brookes' hat?

Mike Cannon-Brookes

Co-CEO

Sydney

Stage 3

257895

928736

456852

Stage 3

Which two fictional characters appeared in another person's row?

Who

Andrei Bocan

Grogu

What

Principal engineer

Unknown

Where

Springfield

The Razor Crest
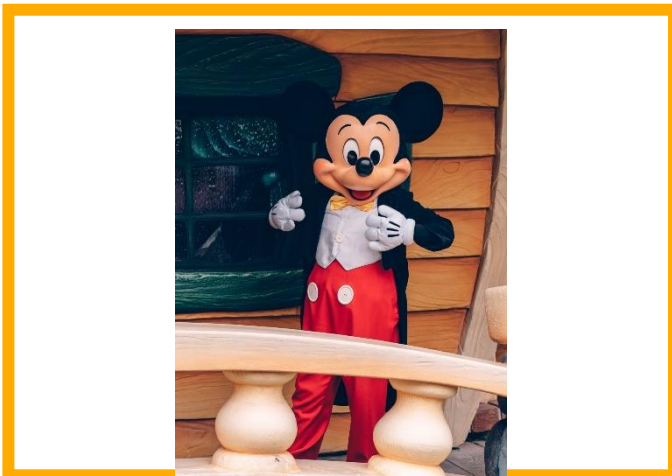
Who	What	Where
Mike Cannon-Brookes	Co-CEO	Sydney
Mickey	Wizard	Disneyland

Complexity in software delivery

The Atlassian microservices journey

Software team effectiveness

Delivering faster, with higher quality

Time

DORA* metrics

Quality

Security and compliance

Developer satisfaction

Memory game - Stage 1

DORA* - DevOps Research and Assessment

The Atlassian microservices journey

Software sprawl begins to take hold

Software team effectiveness

Time

→ Change velocity

↑ Software components

↑ Cognitive load

↓ Developer satisfaction
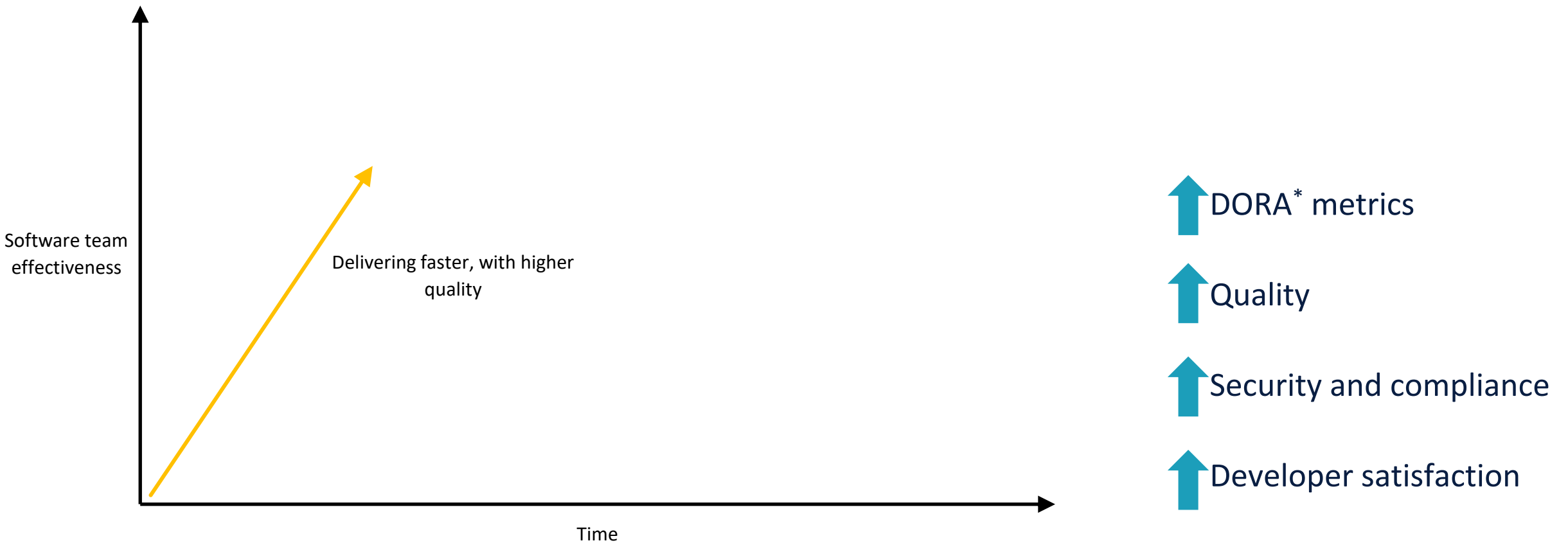
Memory game - Stage 2

The Atlassian microservices journey
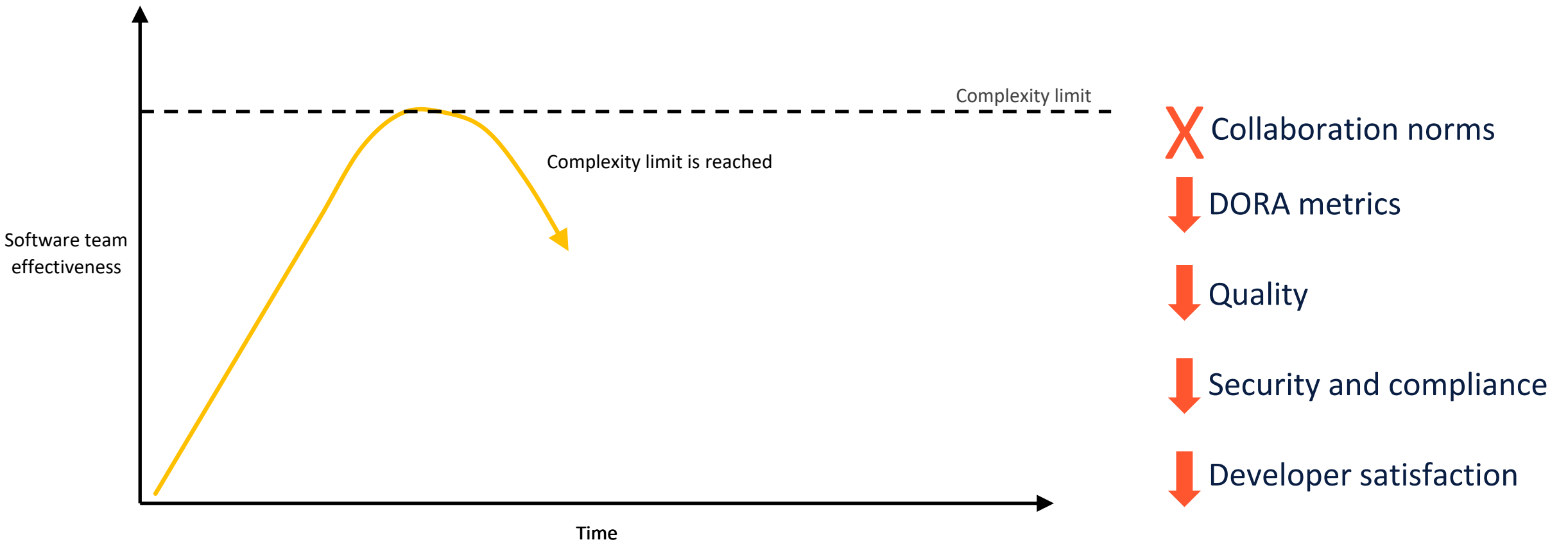
Memory game - Stage 3

Simplification is a losing strategy

- Complexity will grow faster than you can simplify.

- Simplification "projects" operate in the same complex environments that they aim to simplify.

What if we increase the complexity limit for software teams?

The Atlassian microservices journey
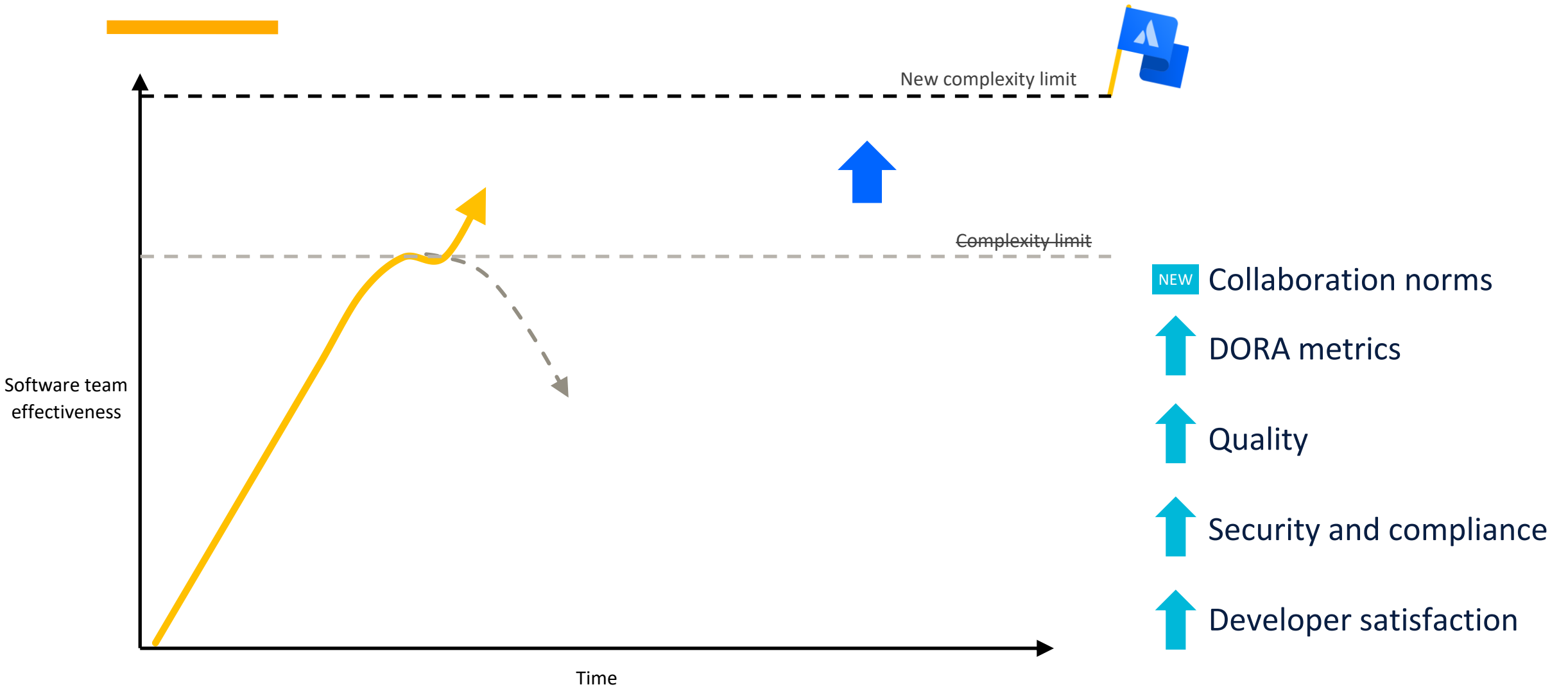
Reduce the need to remember things with a developer experience platform

Problem #1 – Who, What, Where

- Who owns this thing, and what does it do?

- Sprawling microservices + complex dependencies = cognitive load

Component Catalog - Who, what, where?

Teams no longer need to remember which team to engage, what the thousands of components do, or where to go for information.
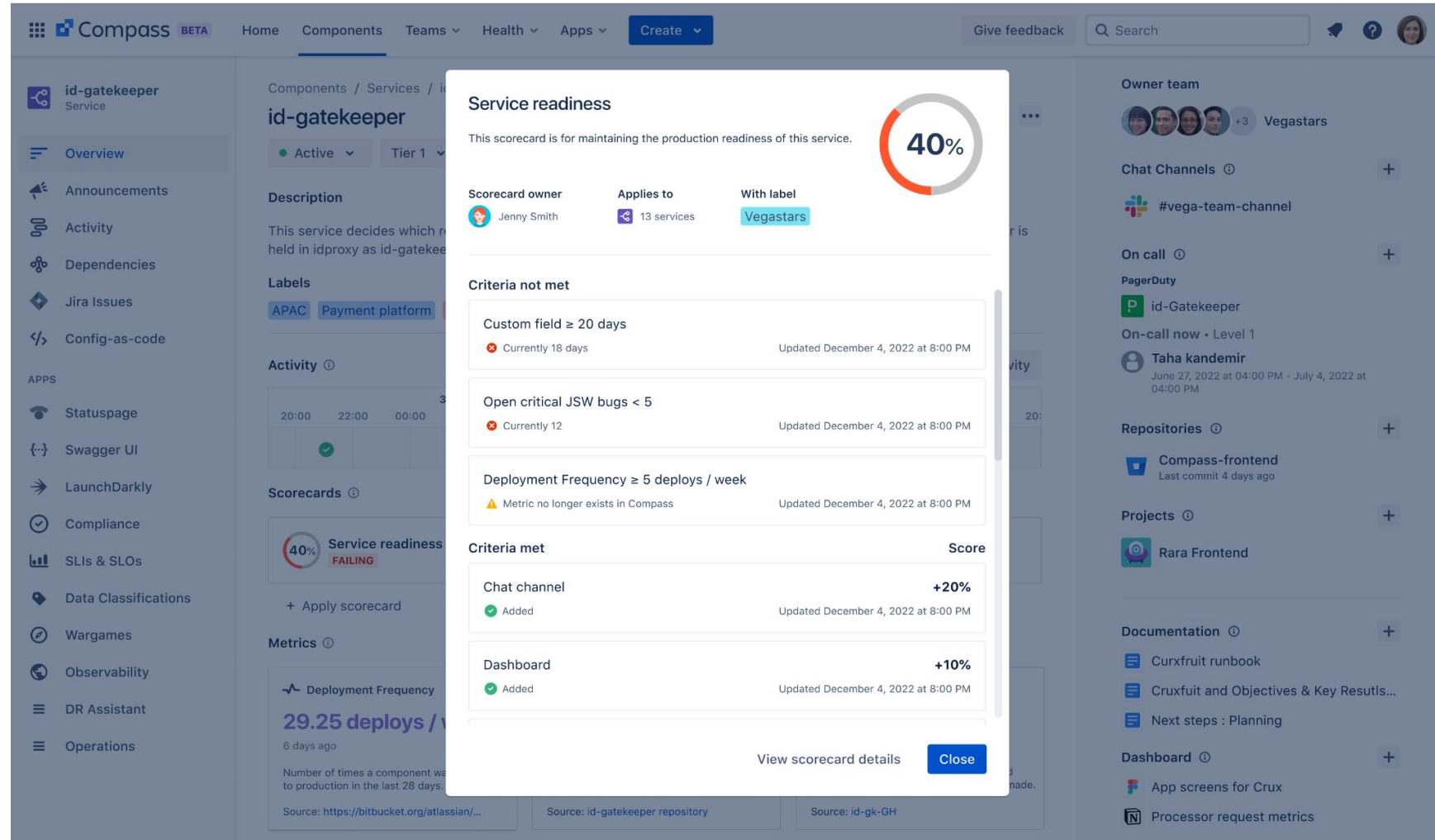
Problem #2 – Quality standards and rework

- How do we communicate security standards quickly, across distributed teams?

- Not meeting a standard = rework for software teams

Scorecards – Standards at scale

Teams know what they need to do upfront, and how they're tracking

Problem #3 – What about me?

- Software teams have diverse, specific needs

- Platform teams become the bottleneck if trying to meet all the needs

Extensibility – self-serve platform

Teams can build their own apps and add them to the platform themselves

Wargames app →

Getting started with Platform Engineering

Tips to accelerate your platform engineering journey

Talk to your developers

Developers are the customer, understand their pain points and what they need.

Vision: Improve Dev Exp.

Define a vision that focuses on improving things for developers. Prioritize what developers need to be productive.

Don't build from scratch

There are similar challenges across most organizations, accelerate your journey by leveraging what's already available.

Final Question

What colour was the number 5?

4568652
257895

Thank you!