



BUILDING SYSTEMS THAT PIVOT

Zhamak Dehghani
@zhamakd

Sam Newman
@samnewman

DO YOU HAVE AN ARCHITECTURE PROBLEM?



DO YOU HAVE AN ARCHITECTURE PROBLEM?



Overhead of releasing a large software



DO YOU HAVE AN ARCHITECTURE PROBLEM?



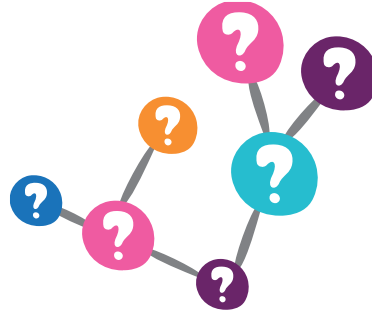
Infrequent releases when orchestrating many dependent teams



Large software
release



DO YOU HAVE AN ARCHITECTURE PROBLEM?



High risk release with untested dependencies



Multiple teams
orchestration



Large software
release



DO YOU HAVE AN ARCHITECTURE PROBLEM?



Untested system dependencies



Multiple teams orchestration



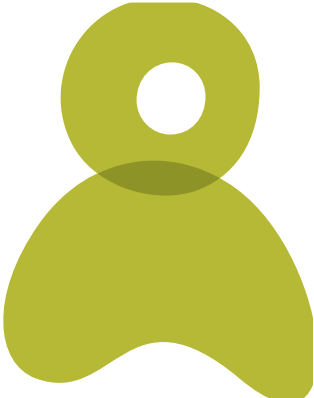
Large software release



A LONG AND BROKEN DIALOGUE



A LONG AND BROKEN DIALOGUE



>50%

A LONG AND BROKEN DIALOGUE



<30%

>50%

AGENDA

- What is **Microservices** Architecture
 - *benefits*
- How to **implement** a Microservices architecture
- How to **operate** a Microservices architecture

CONTINUOUS DELIVERY AS AN ENABLER



CONTINUOUS DELIVERY AS AN ENABLER

- *automated build*



CONTINUOUS DELIVERY AS AN ENABLER

- *automated build*
- *automated deployment*



CONTINUOUS DELIVERY AS AN ENABLER

- *automated build*
- *automated deployment*
- *automated environment provisioning*



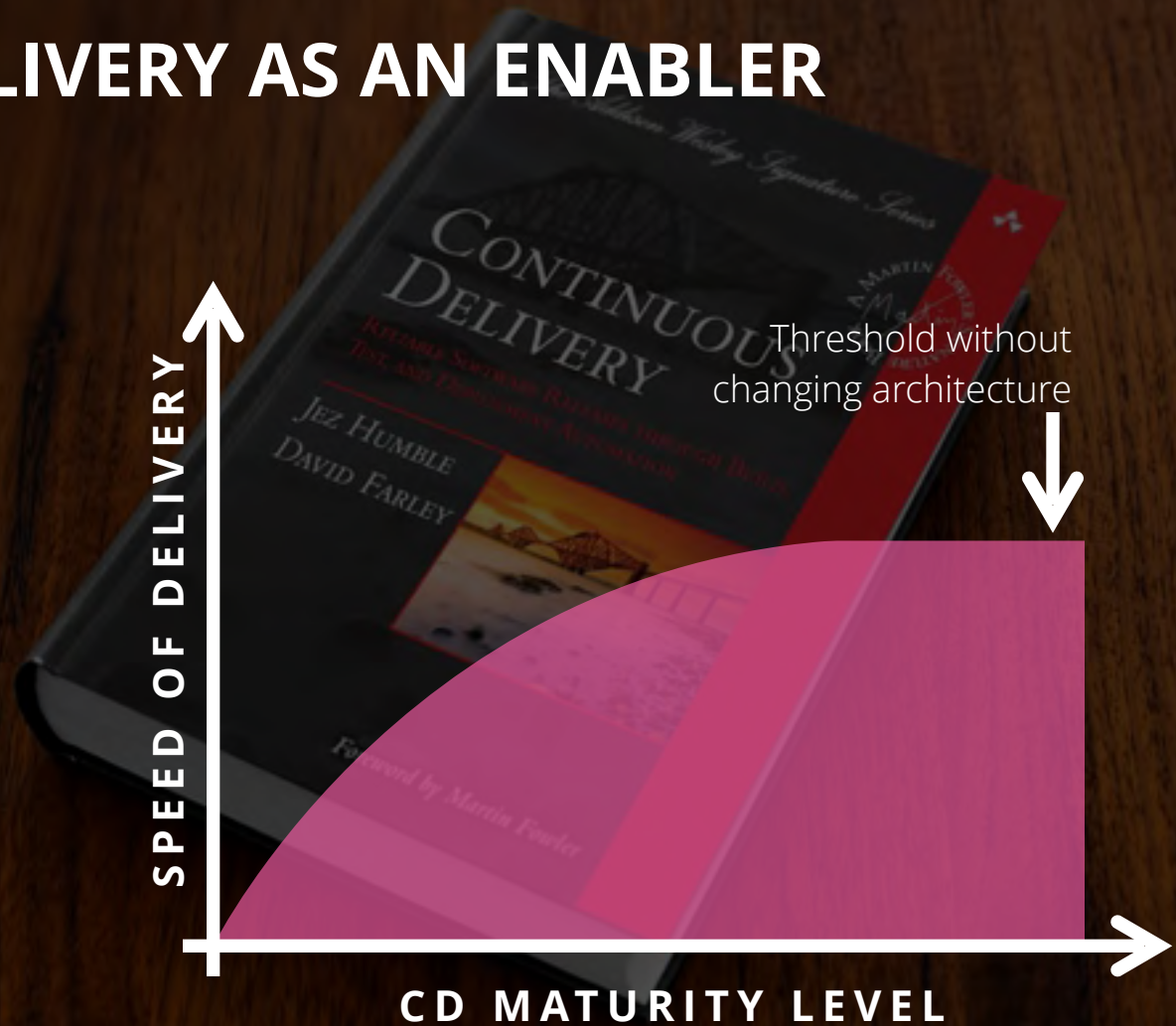
CONTINUOUS DELIVERY AS AN ENABLER

- *automated build*
- *automated deployment*
- *automated environment provisioning*
- *automated testing*



CONTINUOUS DELIVERY AS AN ENABLER

- *automated build*
- *automated deployment*
- *automated environment provisioning*
- *automated testing*



ARCHITECTURE



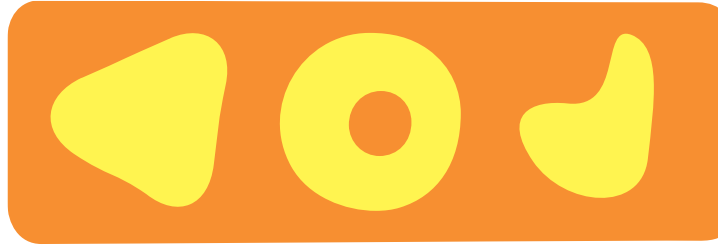
MONOLITHIC SYSTEM



MONOLITHIC SYSTEM



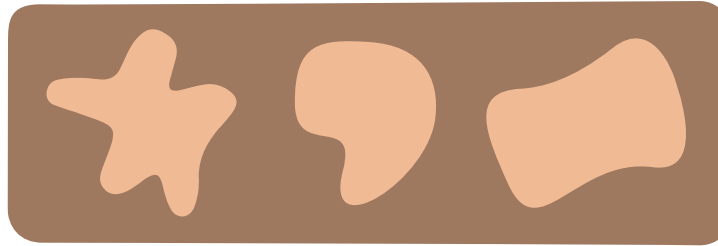
LAYERED/TIERED ARCHITECTURE



User Interface

Channels

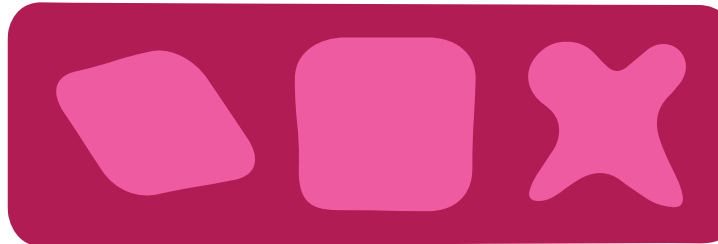
Application



Business Logic & Rules

Middleware

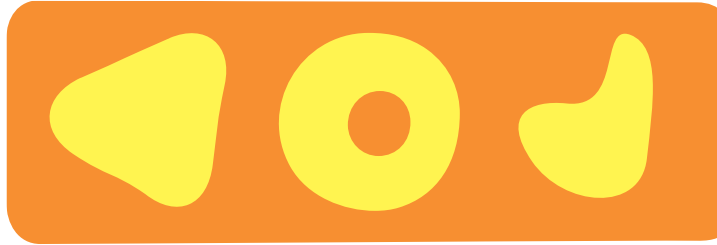
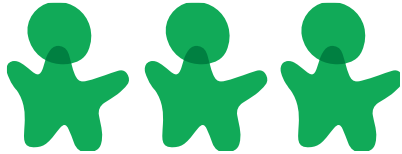
Services platform



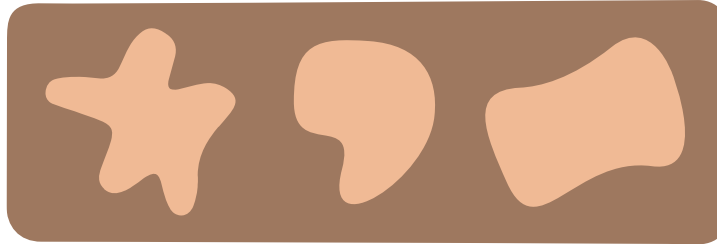
Database

Systems of Record

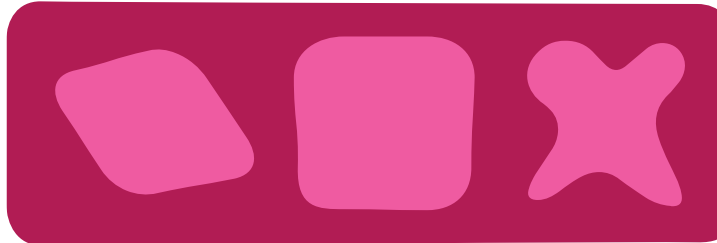
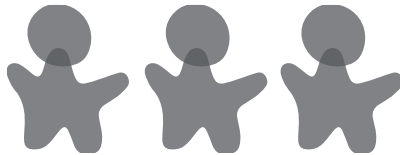
LAYERED/TIERED ARCHITECTURE



User Interface
Channels
Application

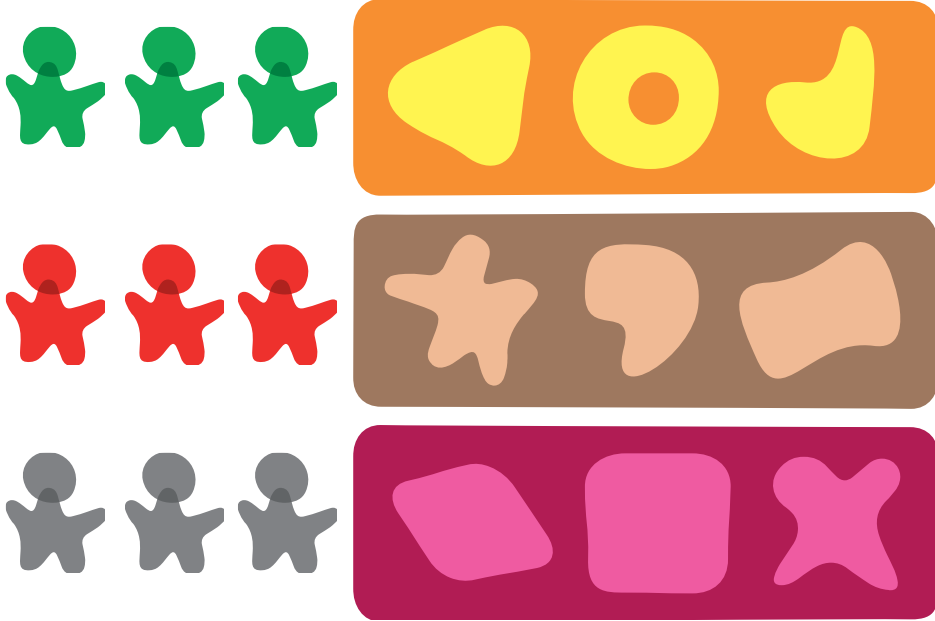


Business Logic & Rules
Middleware
Services platform

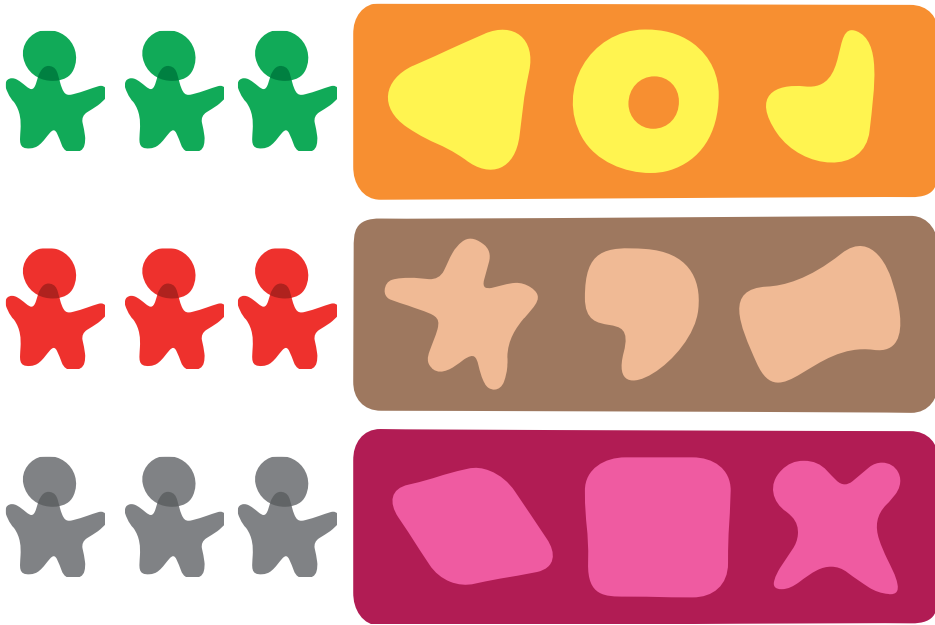


Database
Systems of Record

CHALLENGES OF THE TIERED ARCHITECTURE

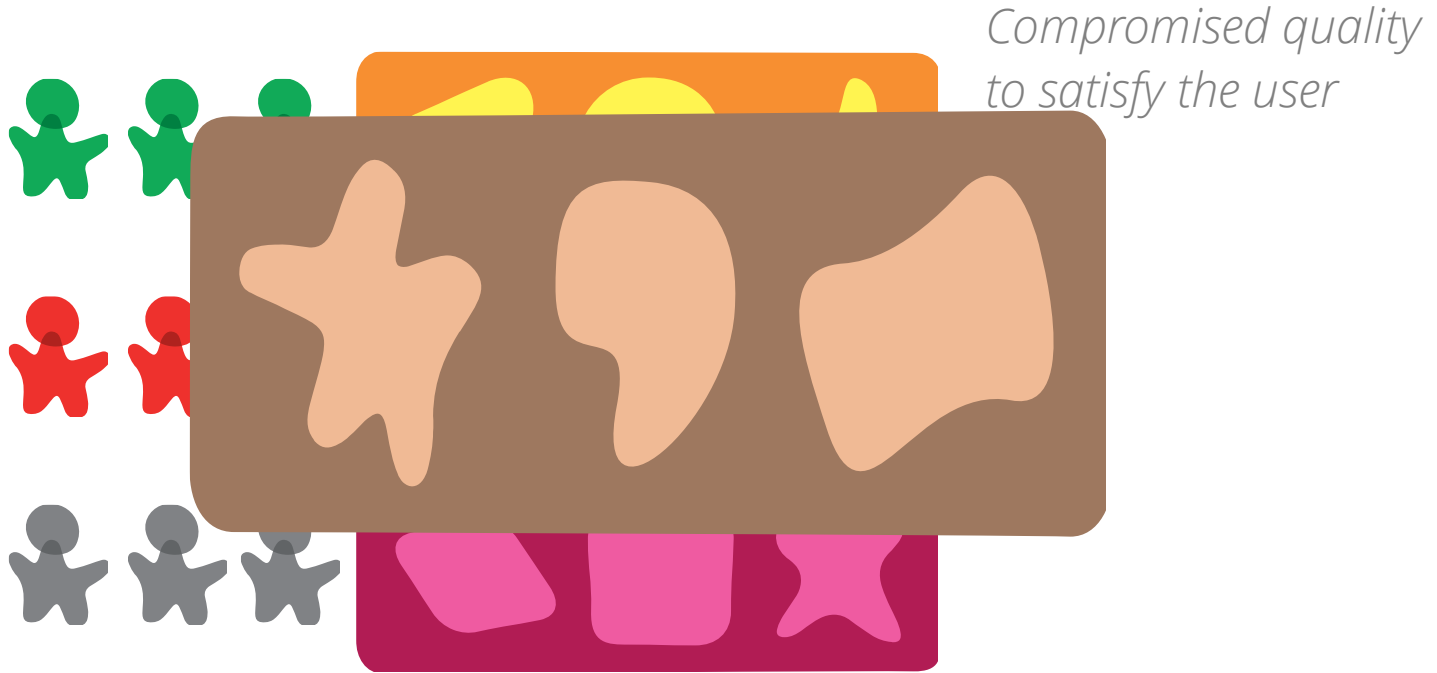


CHALLENGES OF THE TIERED ARCHITECTURE

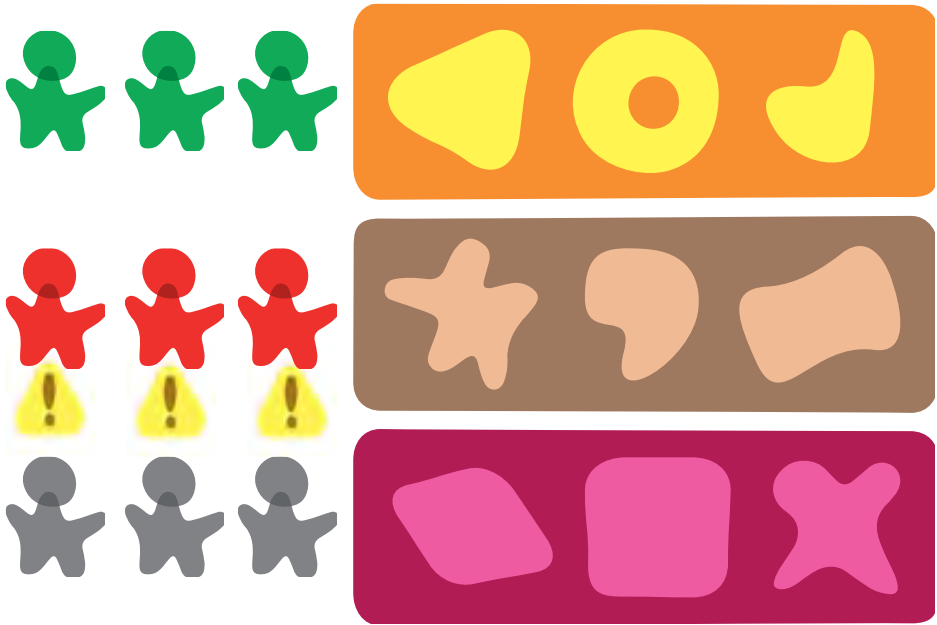


*Compromised quality
to satisfy the user*

CHALLENGES OF THE TIERED ARCHITECTURE

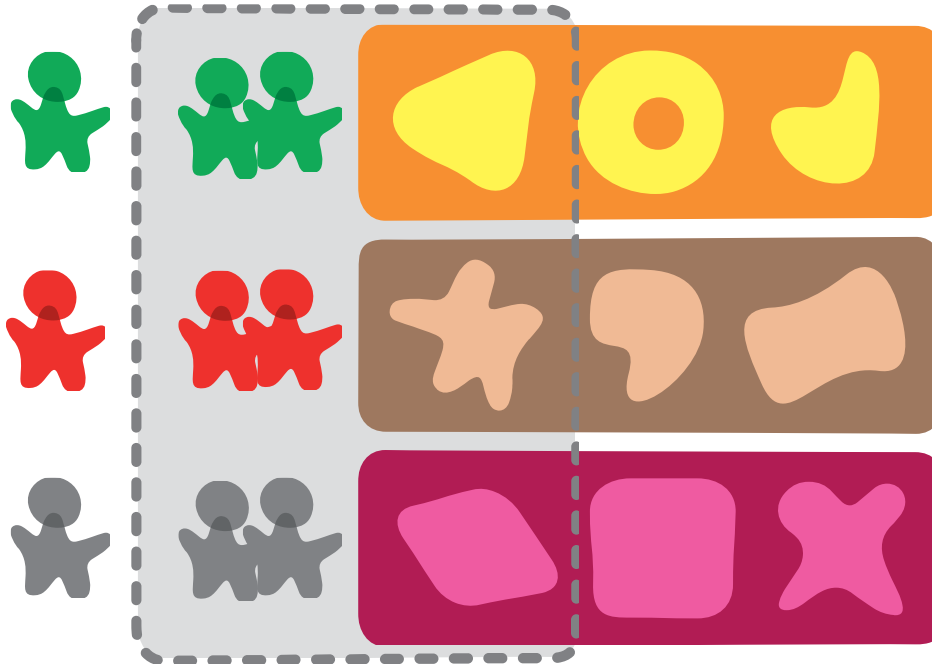


CHALLENGES OF THE TIERED ARCHITECTURE



*Compromised quality
to satisfy the user*

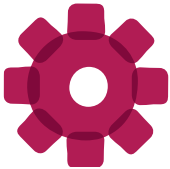
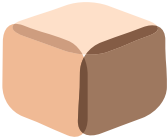
CHALLENGES OF THE TIERED ARCHITECTURE



*Compromised quality
to satisfy the user*

Longer release cycle

FINE-GRAINED ARCHITECTURE



FINE-GRAINED ARCHITECTURE



*Broken down based
on vertical
business functions*

FINE-GRAINED ARCHITECTURE



*Technology layers
are encapsulated
Inside each service*

MICROSERVICES

MICRO

SERVICES

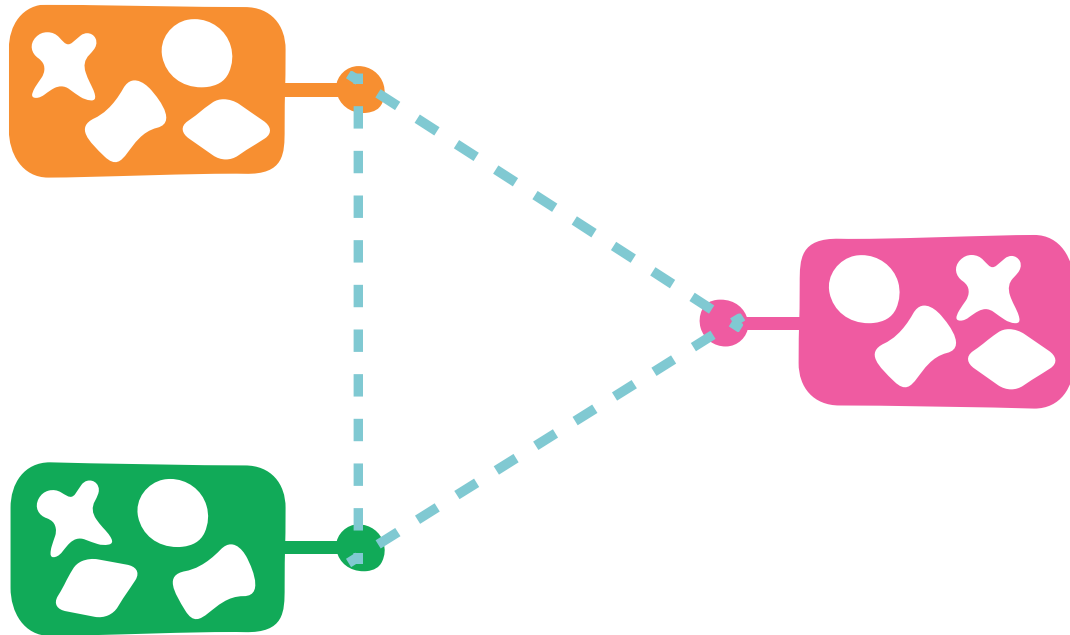
- *Independently releasable*
- *Business capability*
- *Language agnostic API*

MICRO

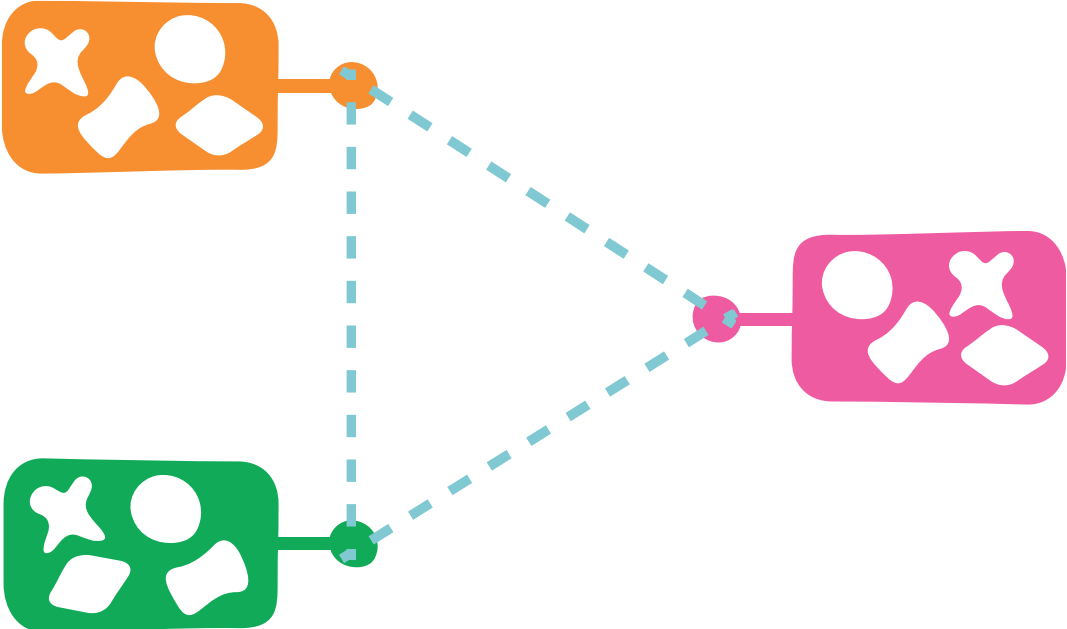
- *Small*
- *100s, not 1000s*

SERVICES

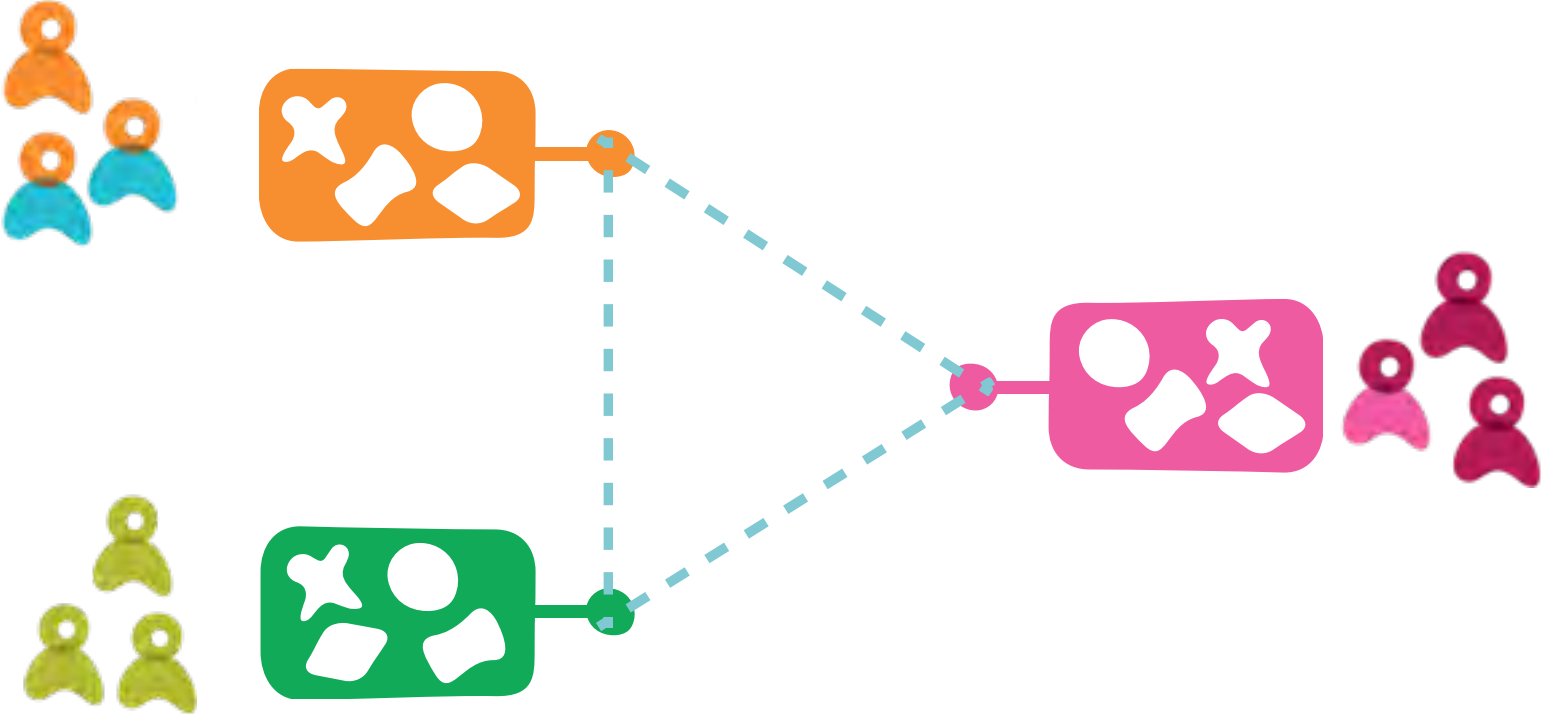
- *Independently releasable*
- *Business capability*
- *Language agnostic API*



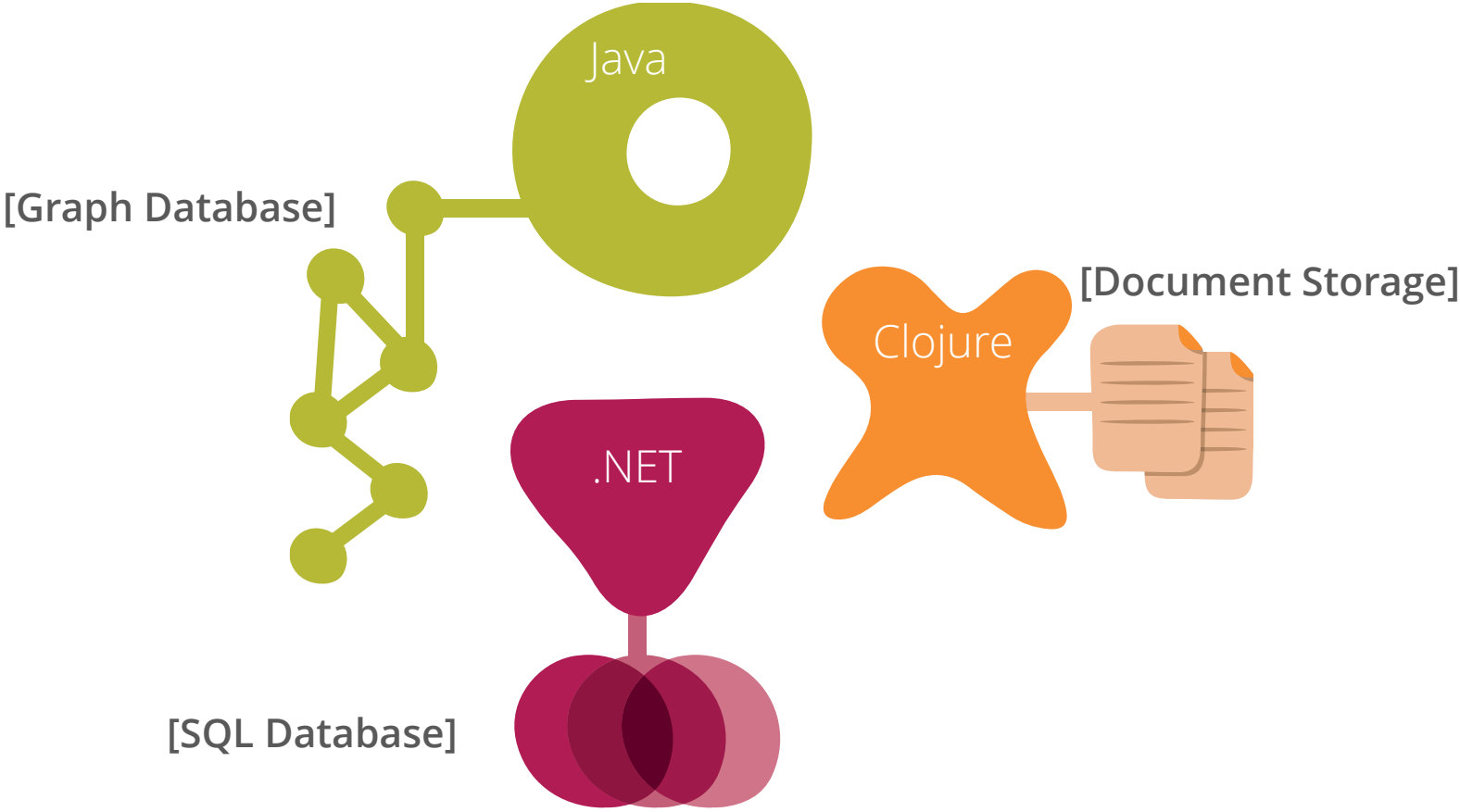
AUTONOMY



AUTONOMY



ABSORBS TECHNOLOGY FASTER



COMPOSE-ABILITY



COMPOSE-ABILITY



COMPOSE-ABILITY



COMPOSE-ABILITY



COMPOSE-ABILITY



REPLACABILITY

HOW TO IMPLEMENT MICROSERVICES

BOUNDARIES



SPLITTING



SPLITTING



Speed of change



SPLITTING



Security

Speed of change



A large green rounded rectangle containing seven icons arranged in three rows. The top row has two circular icons: an orange person silhouette and a green checkmark. The middle row has three square icons: a brown cardboard box, a green shopping cart, and a red and pink pie chart. The bottom row has two square icons: a purple truck and a tan document with lines of text.

SPLITTING



Security



Speed of change



Team Structure



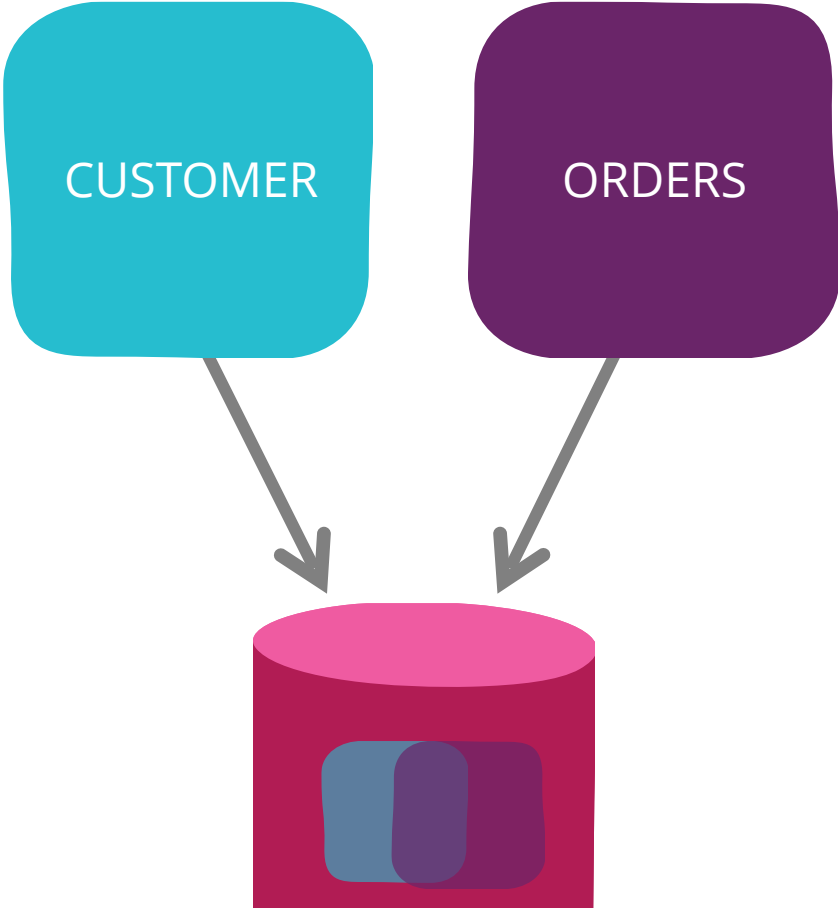


CAN I **RELEASE** THEM **SEPARATELY**?

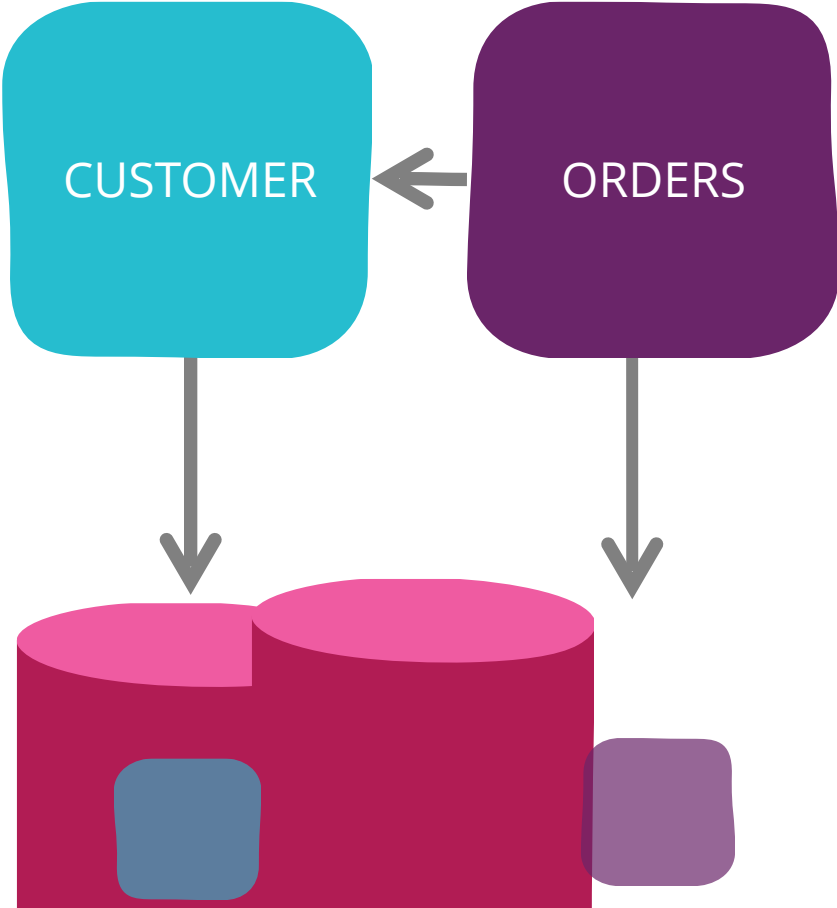
COUPLING THROUGH DATABASE



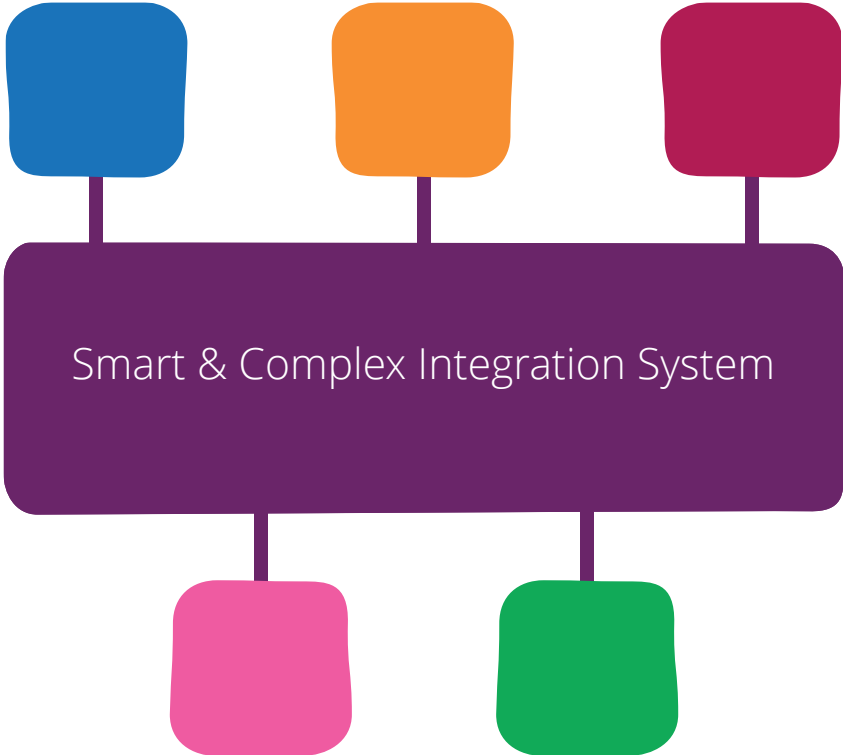
COUPLING THROUGH DATABASE



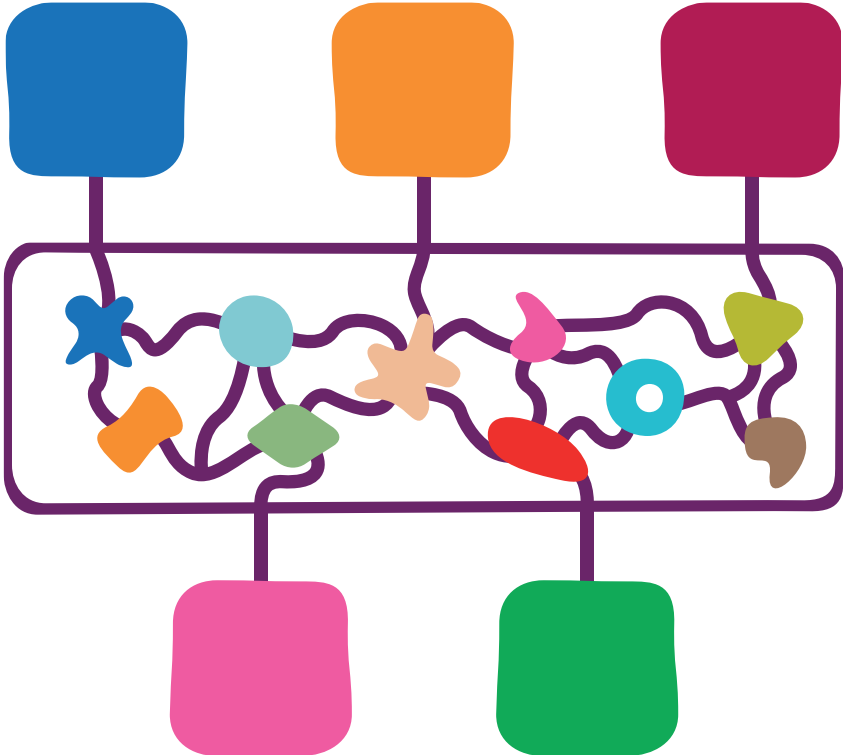
COUPLING THROUGH DATABASE



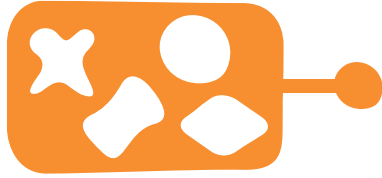
COUPLING THROUGH ENTERPRISE SERVICE BUS



COUPLING THROUGH ENTERPRISE SERVICE BUS



MICROSERVICES INTEGRATION

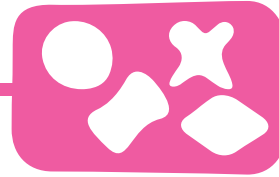


MICROSERVICES INTEGRATION

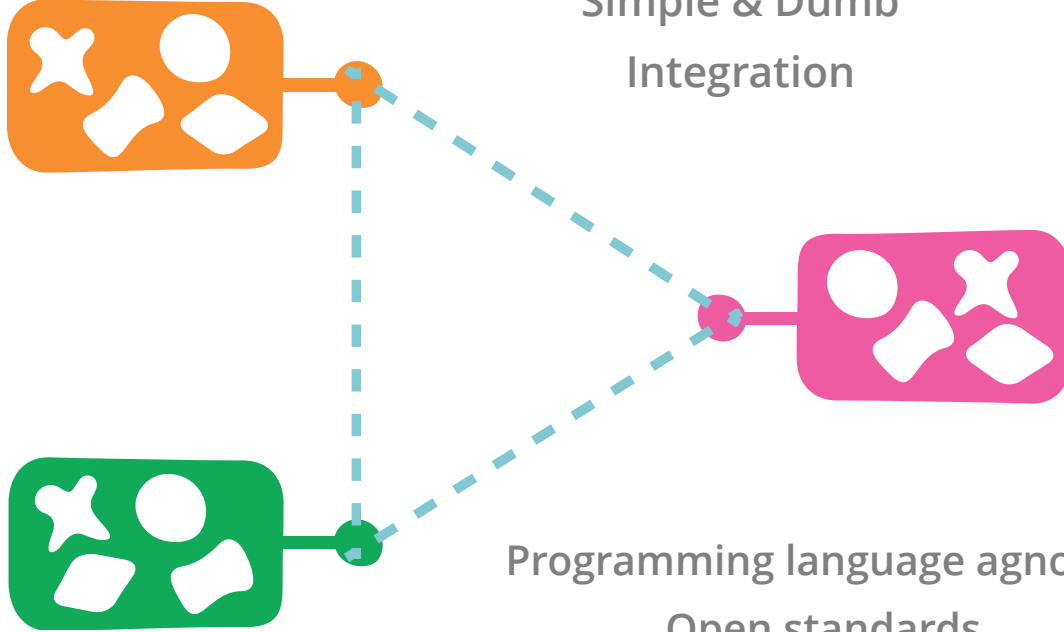
Smart Service &
Endpoint



Simple & Dumb
Integration



Programming language agnostic
Open standards
HTTP/SOAP



WHAT ABOUT **SOA**?

OR **OSGI?**

HOW TO OPERATE MICROSERVICES

AUTOMATE!

AUTOMATE!

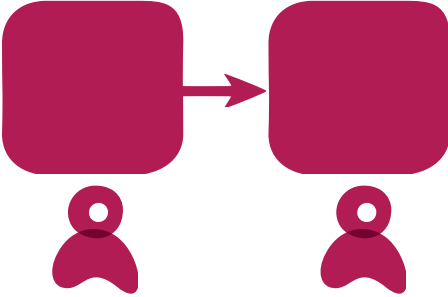
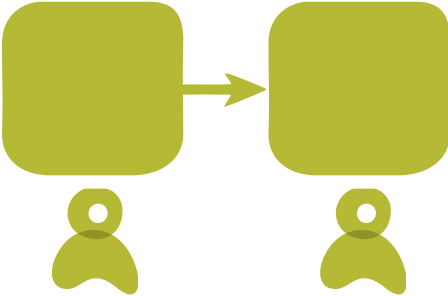
AUTOMATE!

AUTOMATION

AUTOMATION

Build

Test

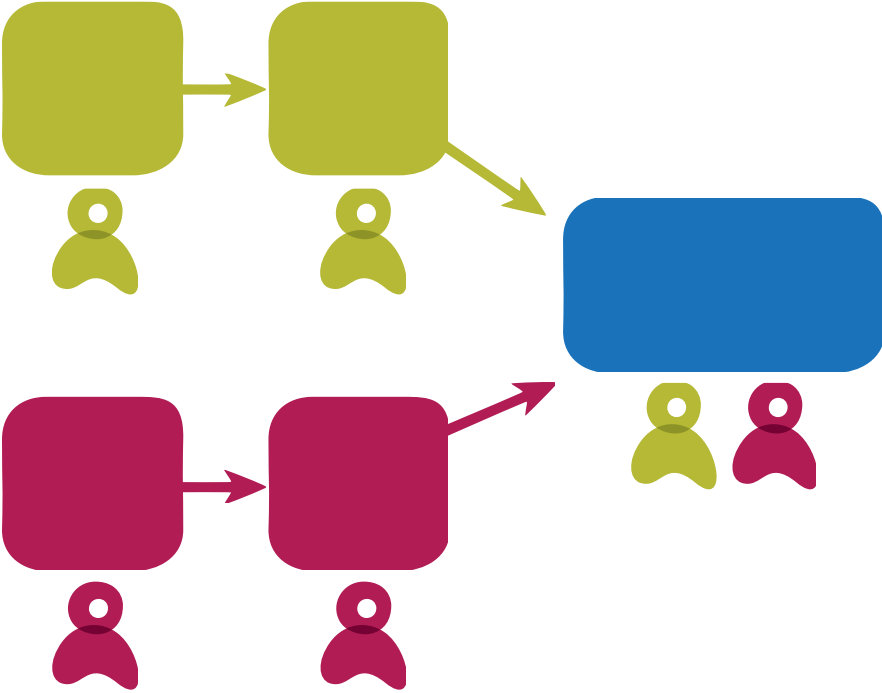


AUTOMATION

Build

Test

Integration



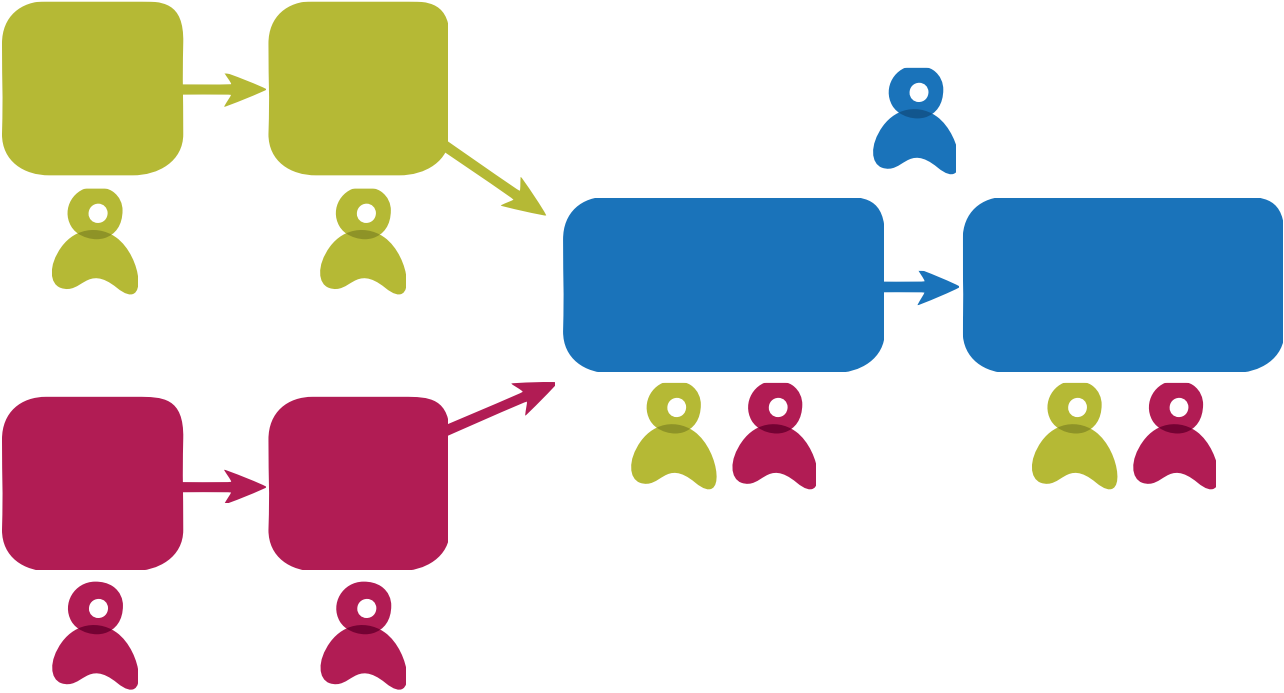
AUTOMATION

Build

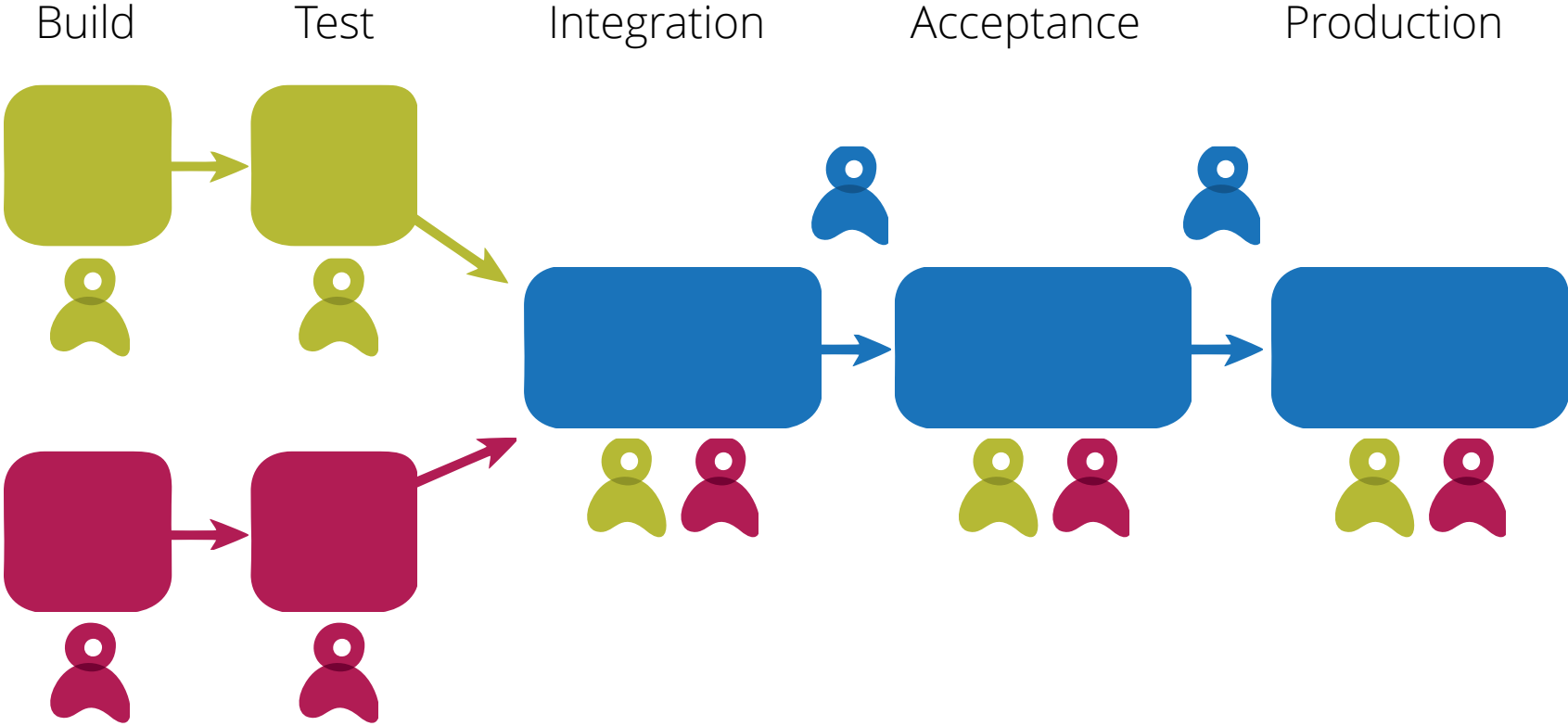
Test

Integration

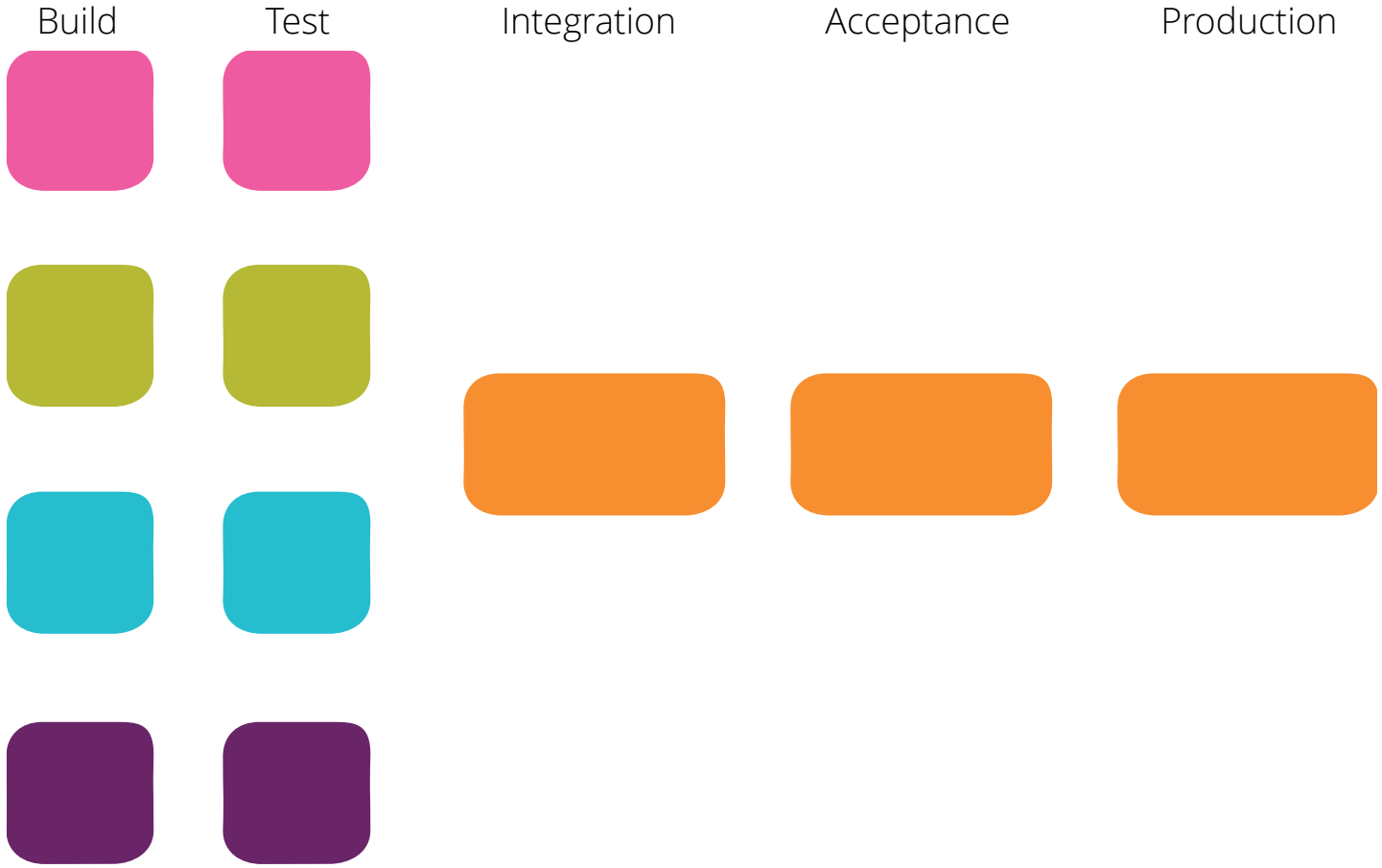
Acceptance



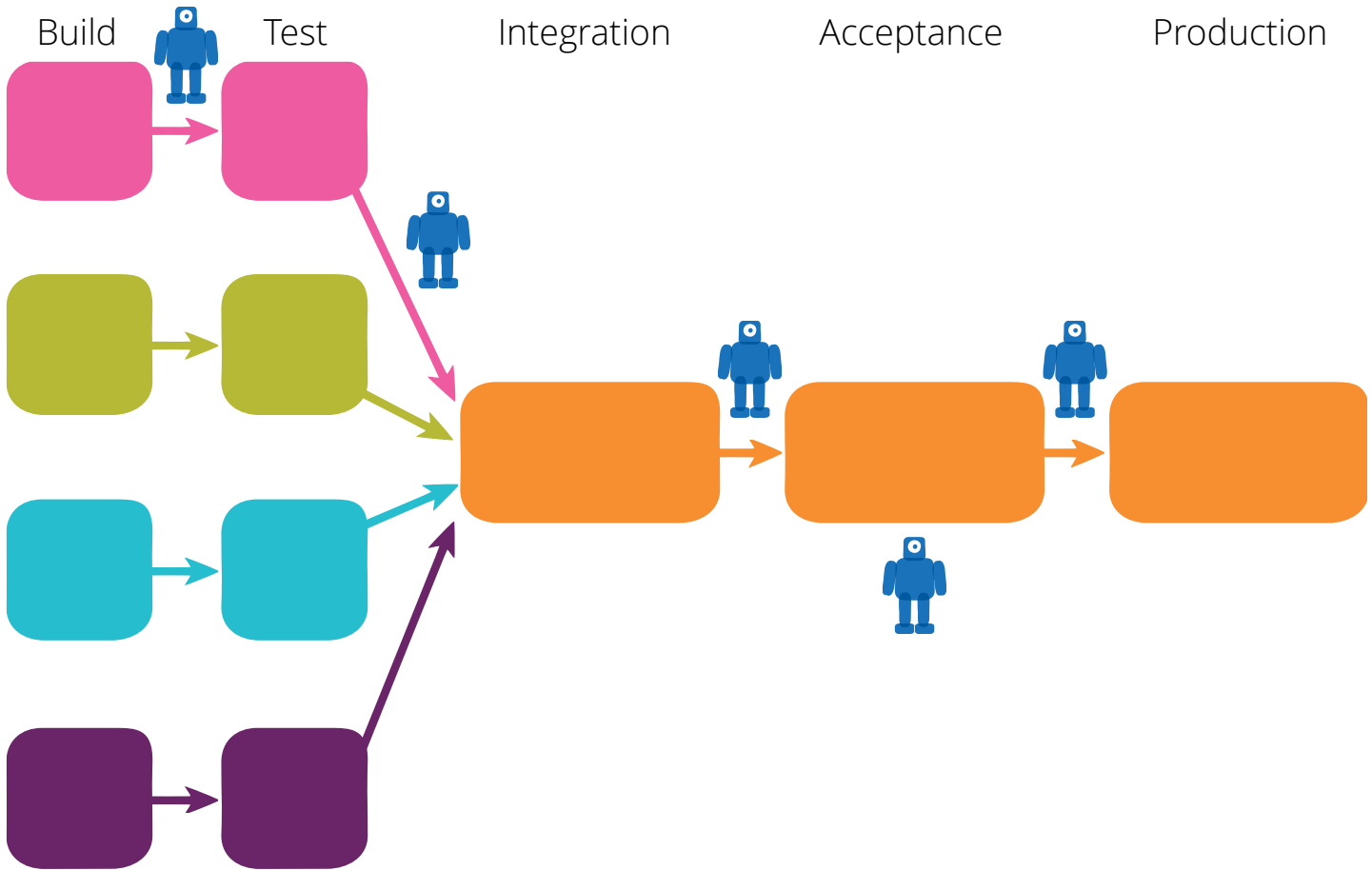
AUTOMATION



MICROSERVICES AUTOMATION



MICROSERVICES AUTOMATION



CONSUMER-DRIVEN CONTRACTS

Consumer



Purchase
Confirm.
Email



Provider



Purchase
Service

CONSUMER-DRIVEN CONTRACTS

Consumer



Purchase
Confirm.
Email



Purchase
Service
Mock



Provider



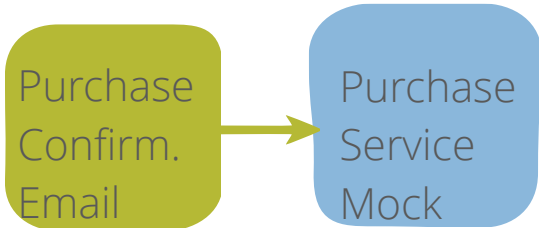
Purchase
Service
API Test



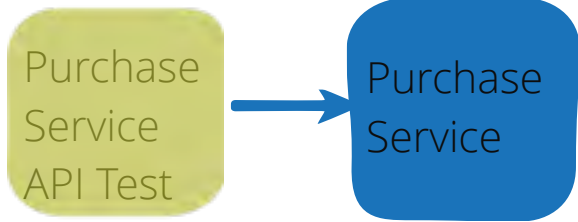
Purchase
Service

CONSUMER-DRIVEN CONTRACTS

Consumer

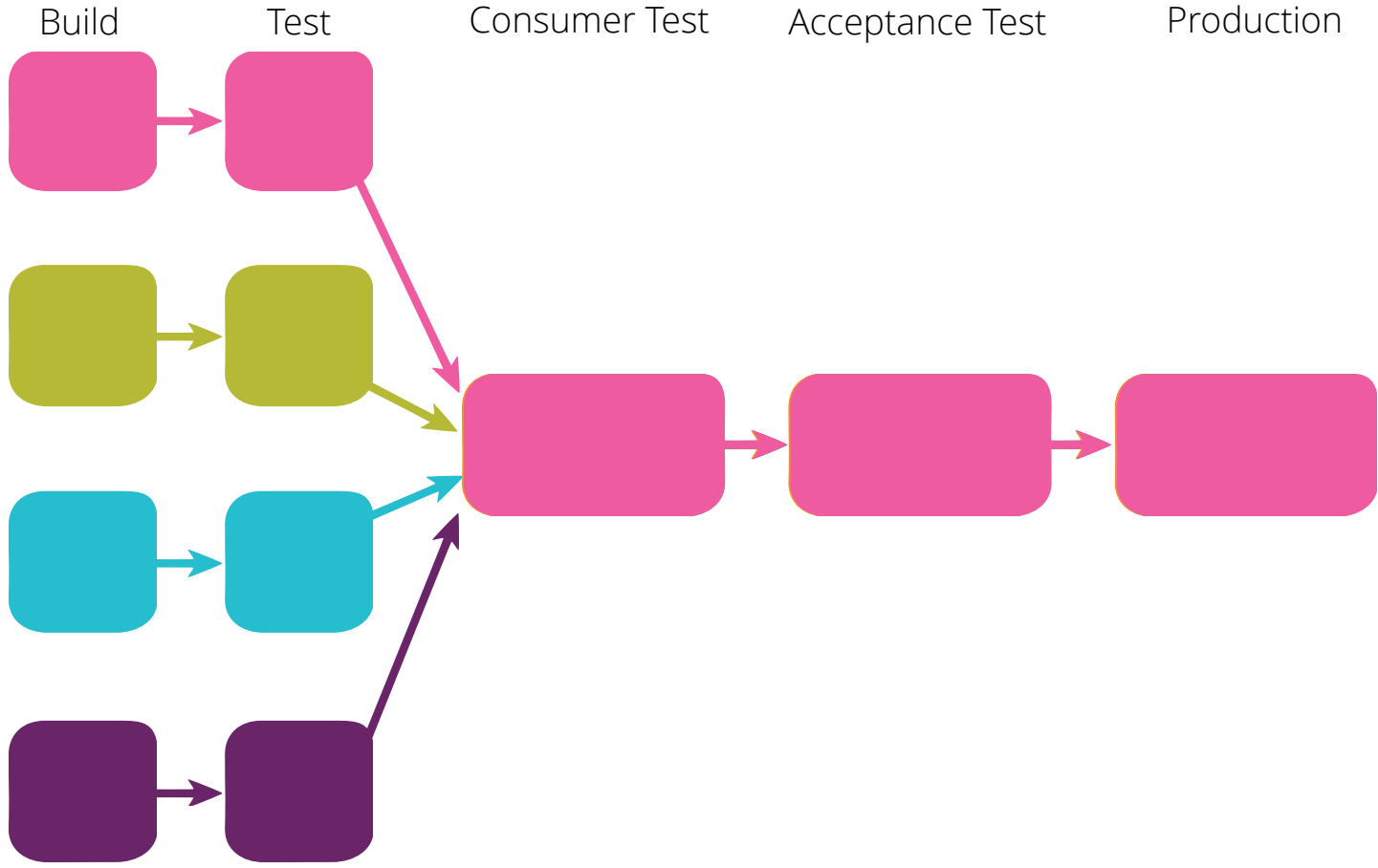


Provider

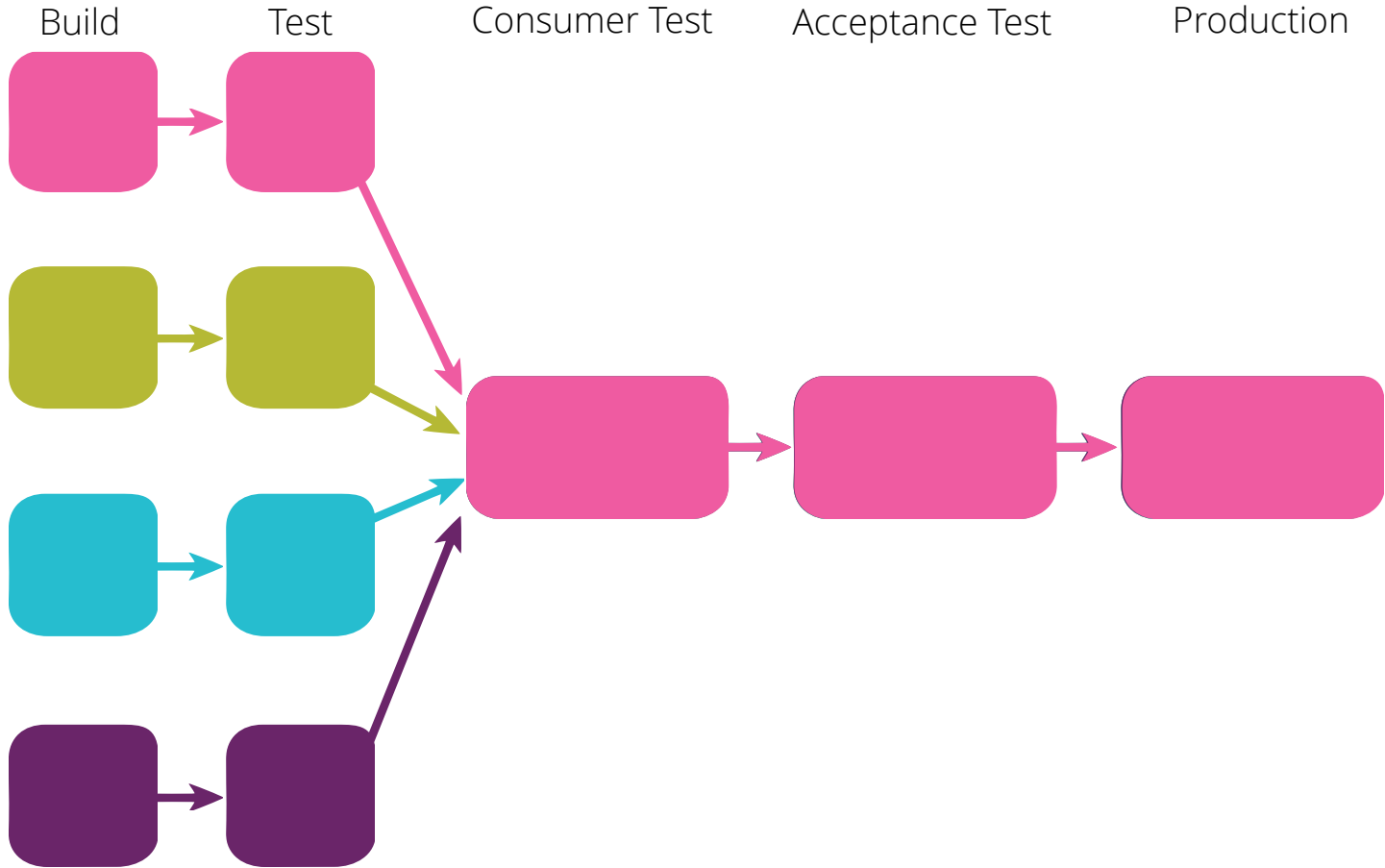


Consumer contract tests

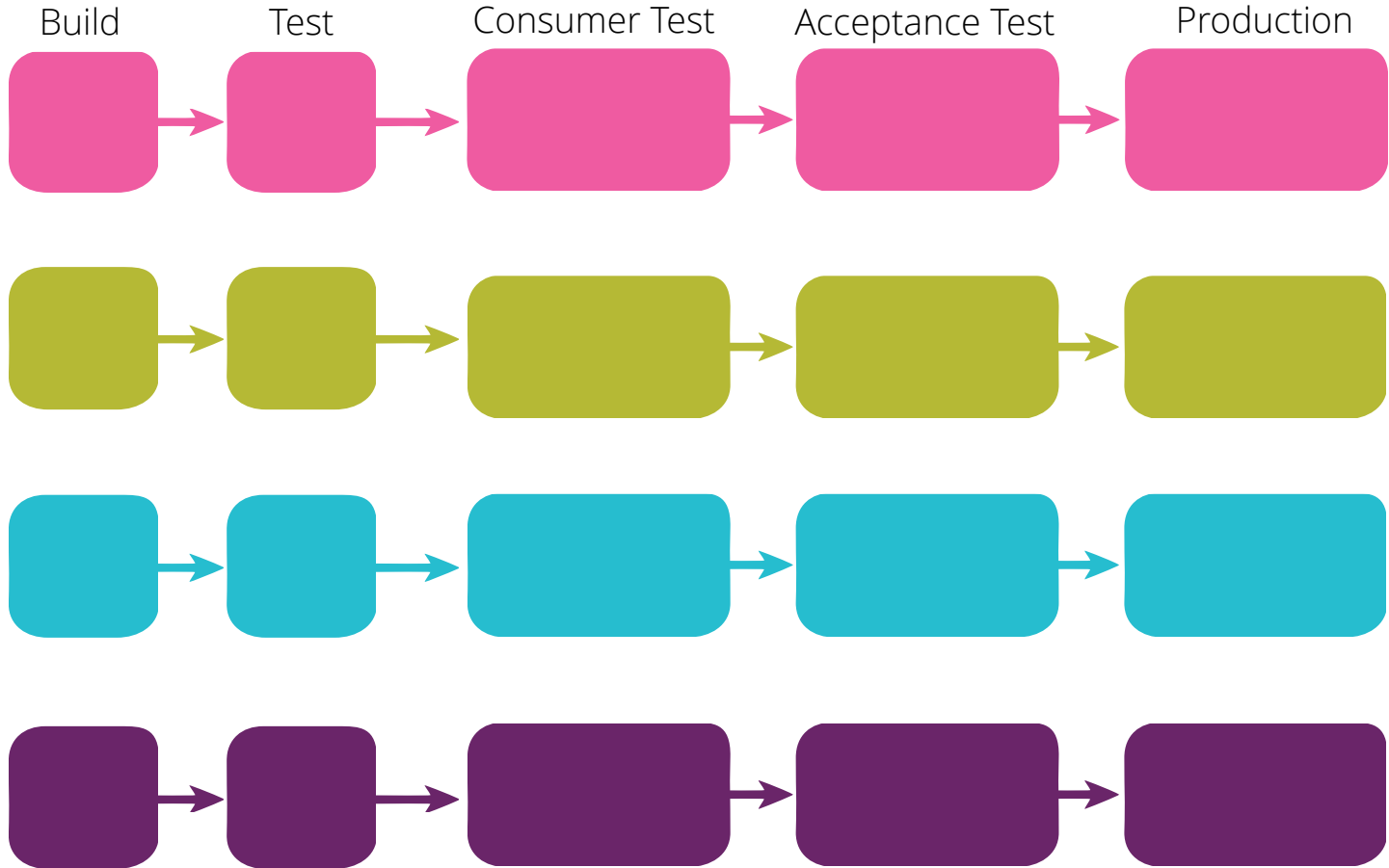
MICROSERVICES' PATH TO PRODUCTION



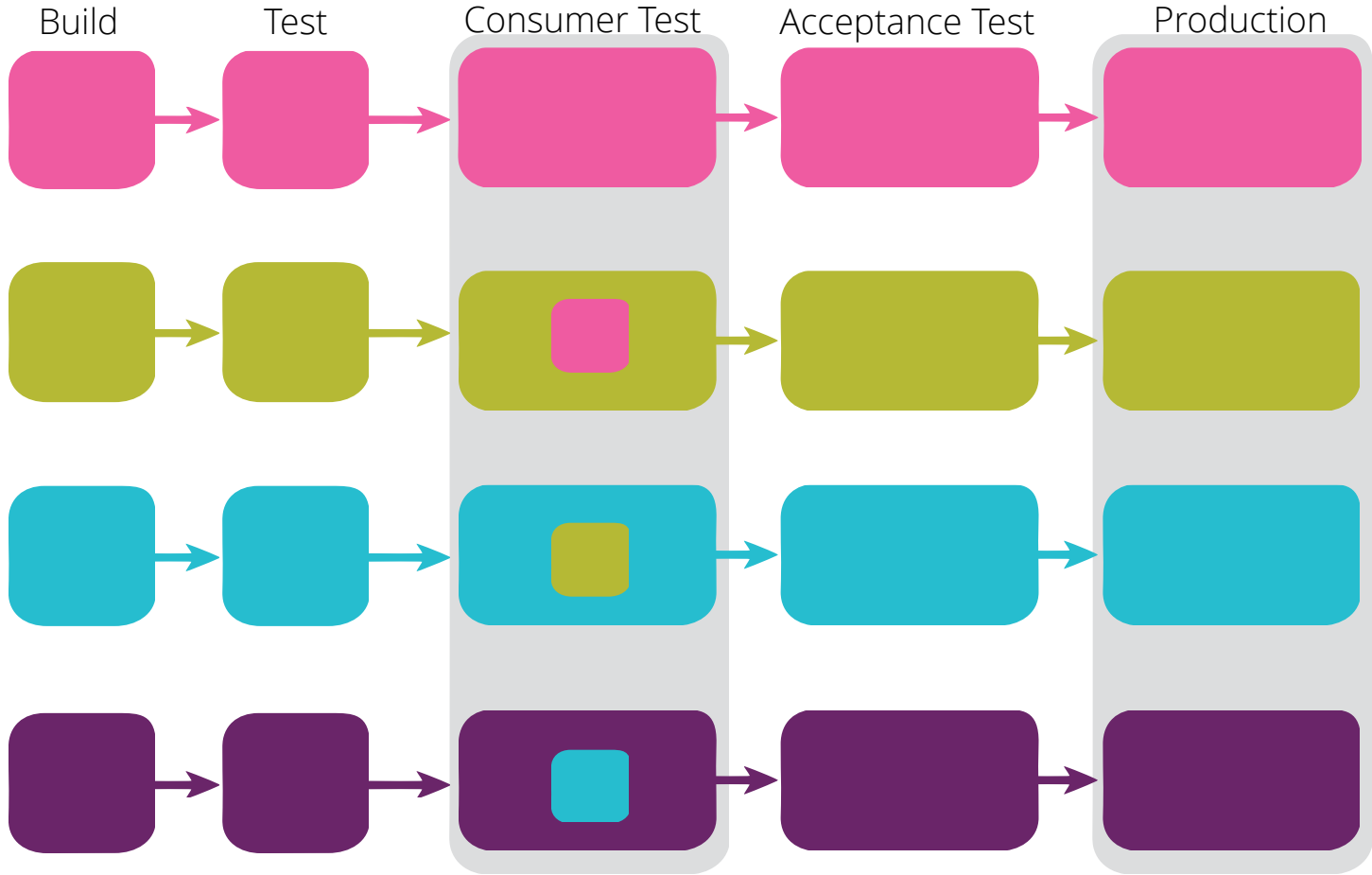
MICROSERVICES' PATH TO PRODUCTION



MICROSERVICES' PATH TO PRODUCTION



MICROSERVICES' PATH TO PRODUCTION



MONITORING



MONITORING



STATUS

- Up?

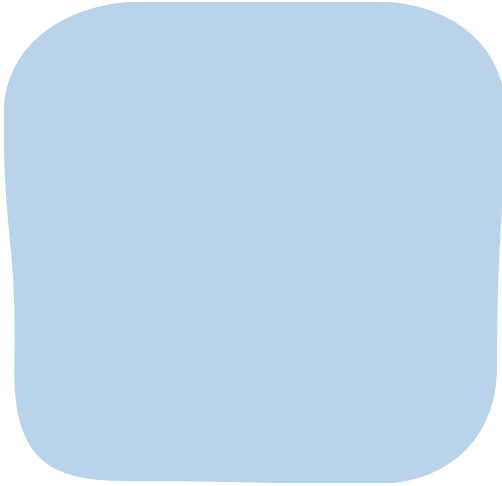
MONITORING



STATUS

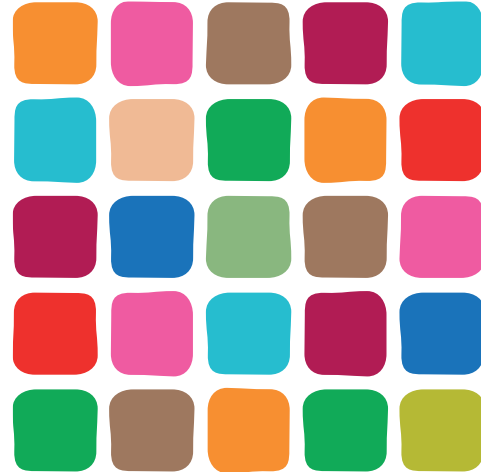
- Up?
- Down?

MONITORING



STATUS

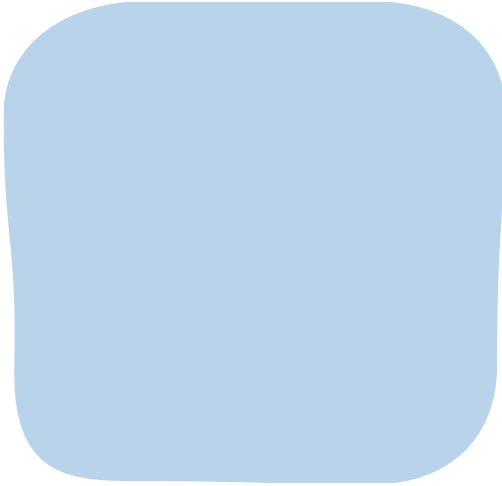
- Up?
- Down?



STATUS

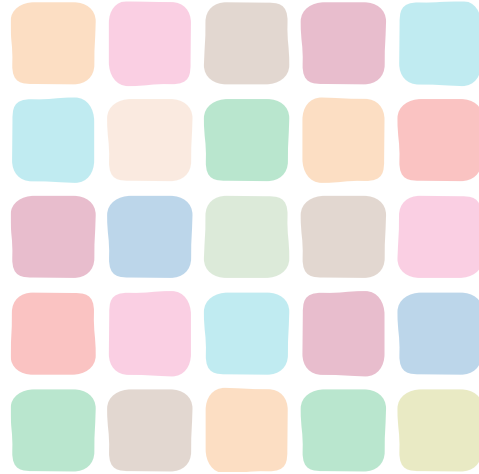
- All up?

MONITORING



STATUS

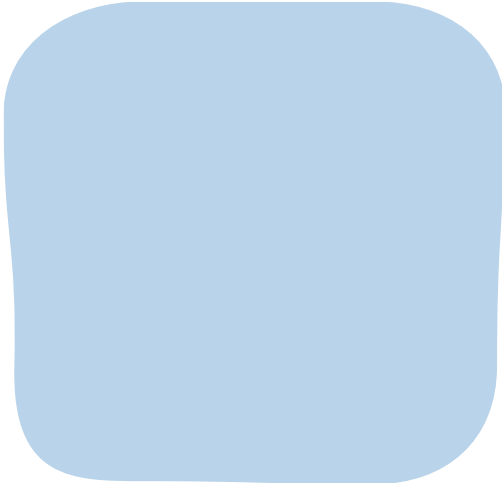
- Up?
- Down?



STATUS

- All up?
- All down?

MONITORING



STATUS

- Up?
- Down?



STATUS

- All up?
- All down?
- Mostly up?

MONITORING



STATUS

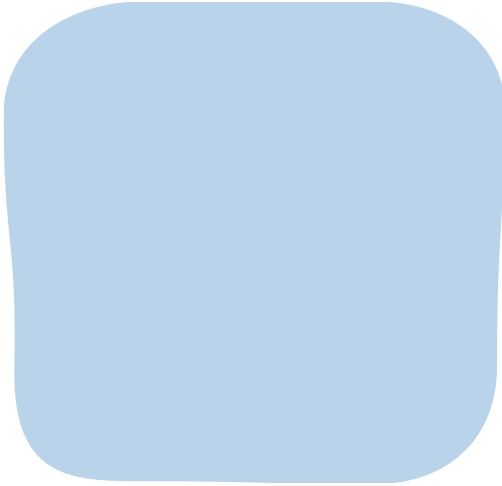
- Up?
- Down?



STATUS

- All up?
- All down?
- Mostly up?
- Up apart from 1?

MONITORING



STATUS

- Up?
- Down?



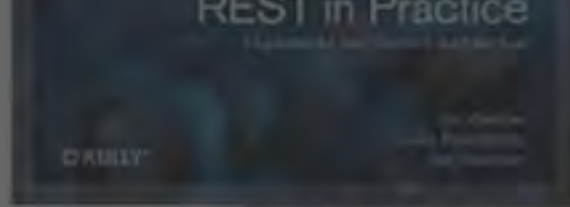
STATUS

- All up?
- All down?
- Mostly up?
- Up apart from 1?
- All of the important bits are up?
- ...

AMAZON PRODUCT PAGE

The screenshot shows the top portion of an Amazon product page. At the top left is the Amazon logo with a 'Try Prime' tag. To the right are links for 'Andy's Amazon.com', 'Today's Deals', and 'Gift'. Below the logo is a 'Shop by Department' section with a search bar containing the text 'All rest'. Navigation links include 'Buy a Kindle', 'Kindle eBooks', 'Advanced Search', and 'Daily Deals'. A yellow banner contains the text: 'Great news! You can now shop for digital content on Amazon.com.au. Have you moved recently? You may change your country or region.' Below this is a promotional text: 'Start reading REST in Practice: Hypermedia and RESTful Architecture'. The main product image shows the book cover with a 'Click to LOOK INSIDE!' callout. The cover features a blue background with several colorful, rounded objects. The title 'REST in Practice' is at the bottom of the cover, with 'Hypermedia and RESTful Architecture' in smaller text below it. The author's name 'Jan Robinson' is visible. Below the cover is a 'kindle edition' label and a 'Share your own customer images' link. To the right of the cover, the title 'REST in F Architecture' is partially visible, along with the author's name 'Jan Robinson (Author)'. Below the author name are links for 'Digital List Price', 'Print List Price', 'Rent From', and 'Save up to'. Further down, the 'Print ISBN-10' and 'Print ISBN-13' are listed, along with 'Edition: 1' and a bullet point indicating 'Length: 448'. A 'Formats' section is partially visible, showing 'Kindle Ec' and 'Paperba'. At the bottom of the page, there is a section titled 'Shop the new tech.book(store)' with a small icon and text: 'New! Introducing the tech.book(store), a hub for Software De other technology professionals to find highly-rated and highly- data, or read this week's blog posts by authors and thought-le'. Below this is a section titled 'Book Description' with the text: 'Why don't typical enterprise projects go as smoothly as pr architectural style really present a viable alternative for bu applications?'.

AMAZON PRODUCT PAGE



kindle edition



[Share your own customer images](#)

Print ISBN-13

Edition: 1

- Length: 448

Formats

Kindle Ec

Paperba



Shop the new tech.book(store)

New! Introducing the [tech.book\(store\)](#), a hub for Software De
other technology professionals to find highly-rated and highly
[data](#), or read this week's blog posts by authors and thought-le

Book Description

AMAZON PRODUCT PAGE



kindle edition



[Share your own customer images](#)

Print ISBN-13

Edition: 1

- Length: 448

Formats

Kindle Ec

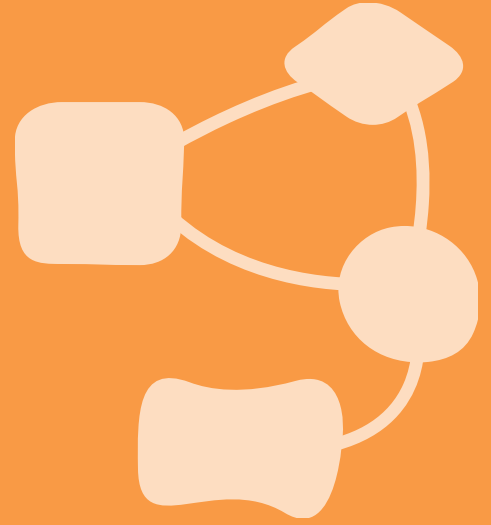
Paperba

DOESN'T APPEAR... DO WE CARE?

Book Description

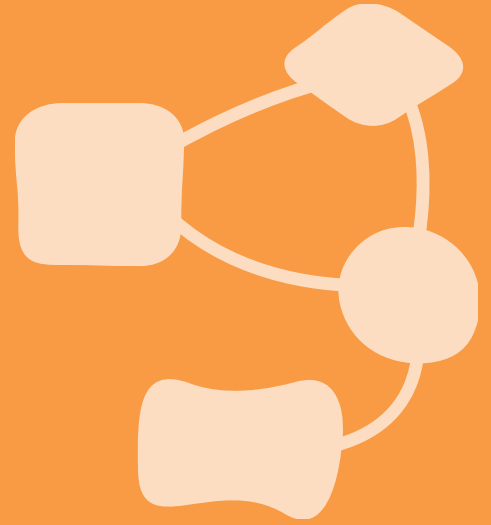
SUMMARY

RECOMMENDATIONS



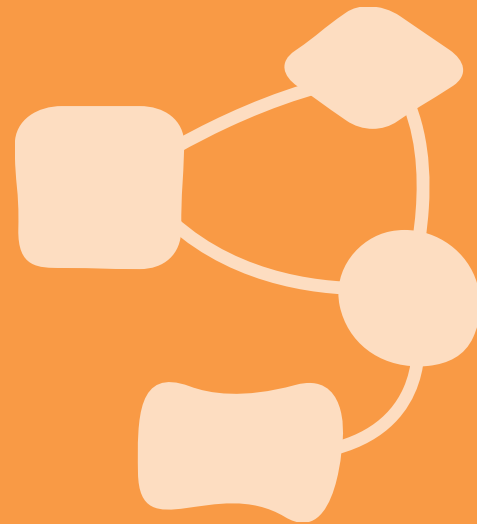
RECOMMENDATIONS

- *Build independently releasable microservices*
 - *align microservice with business capability*



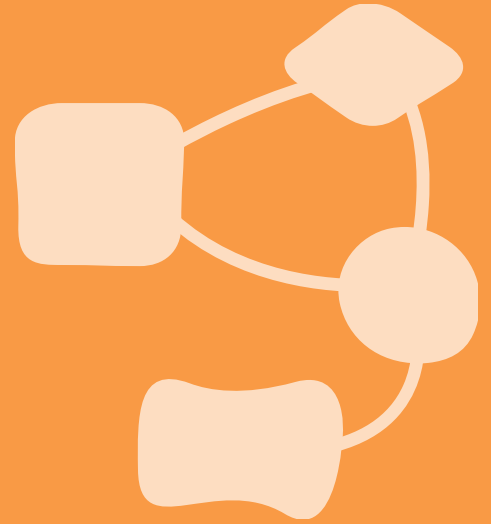
RECOMMENDATIONS

- *Build independently releasable microservices*
 - *align microservice with business capability*
- *Integrate microservices with*
 - *dumb, open language agnostic*
 - *outside database*



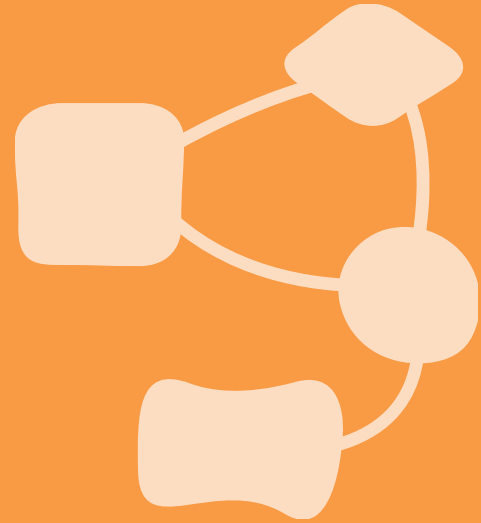
RECOMMENDATIONS

- *Build independently releasable microservices*
 - *align microservice with business capability*
- *Integrate microservices with*
 - *dumb, open language agnostic*
 - *outside database*
- *Operate microservices architecture by*
 - *decentralizing and automating all aspects of build, test and deployment*



RECOMMENDATIONS

- *Build independently releasable microservices*
 - *align microservice with business capability*
- *Integrate microservices with*
 - *dumb, open language agnostic*
 - *outside database*
- *Operate microservices architecture by*
 - *decentralizing and automating all aspects of build, test and deployment*



The image features a solid orange background. In the top-left and bottom-right corners, there are abstract, overlapping shapes in red, green, and black. The word "CONCLUSION" is centered in the middle of the page in a bold, white, sans-serif font.

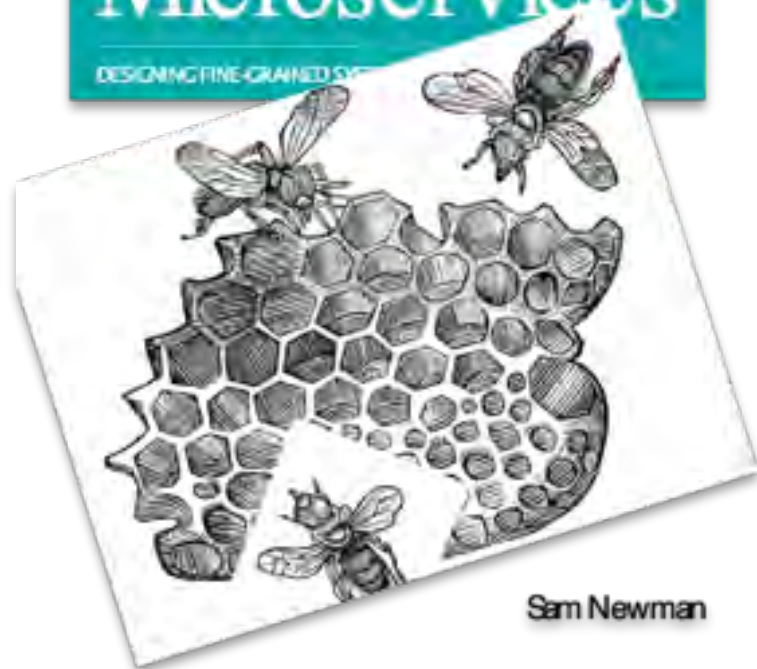
CONCLUSION

ThoughtWorks®

O'REILLY

Building Microservices

DESIGNING FINE-GRAINED SYSTEMS



Sam Newman

THANK YOU

Zhamak Dehghani

zdehghan@thoughtworks.com @zhamakd

Sam Newman

snewman@thoughtworks.com @samnewman