



Builders Online Series

Tools for building your MVP on AWS

Rob De Feo
Startup Evangelist
Amazon Web Services



@robdefeo

What is an

MVP?

“If you aren't embarrassed by the first version of your product, you shipped too late.”

Reid Hoffman
Co-founder LinkedIn

Minimum {x} product



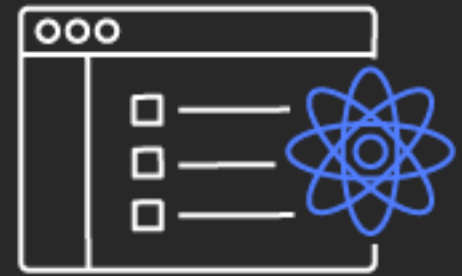
Viable



Usable

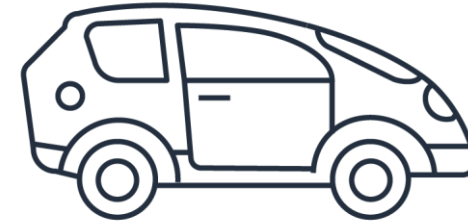
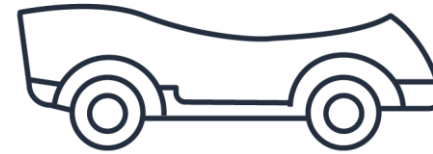


Lovable

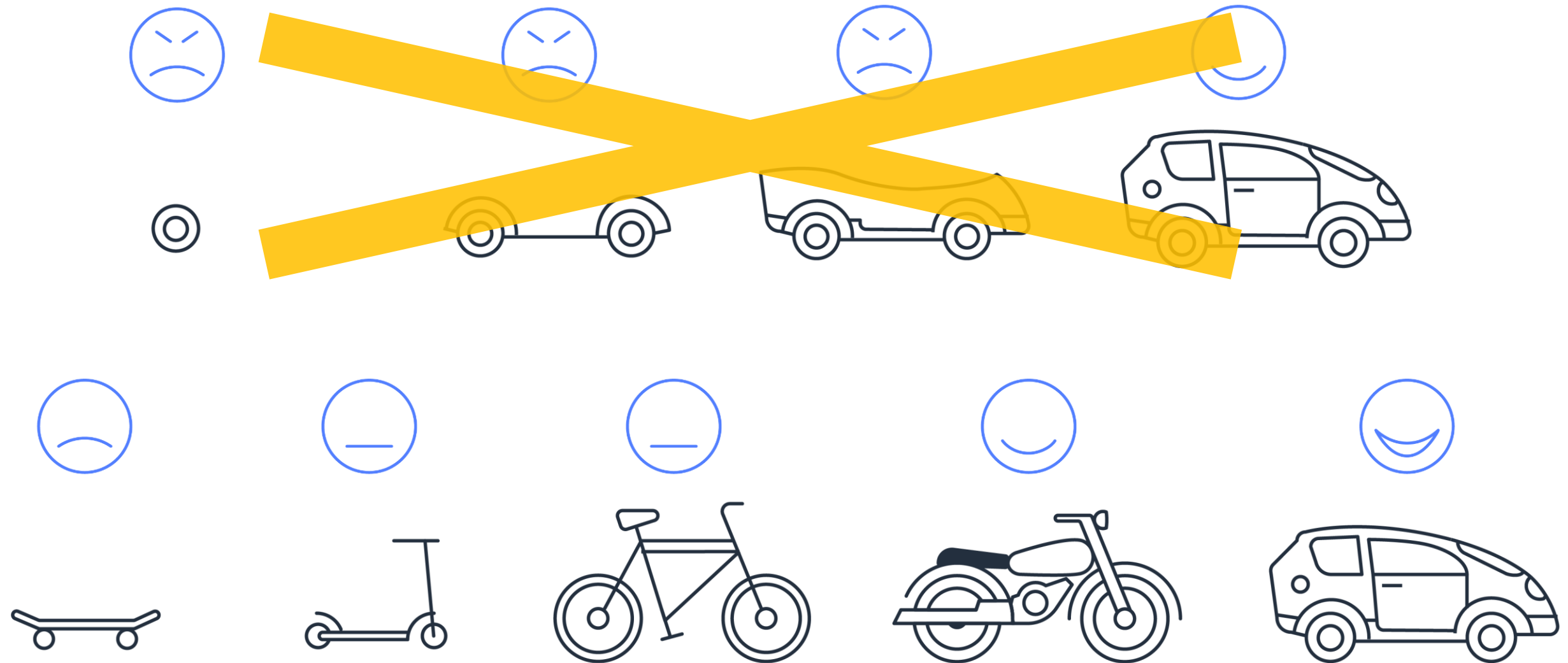


Testable

Product



Product



What we'll cover



Development process



Anti-patterns

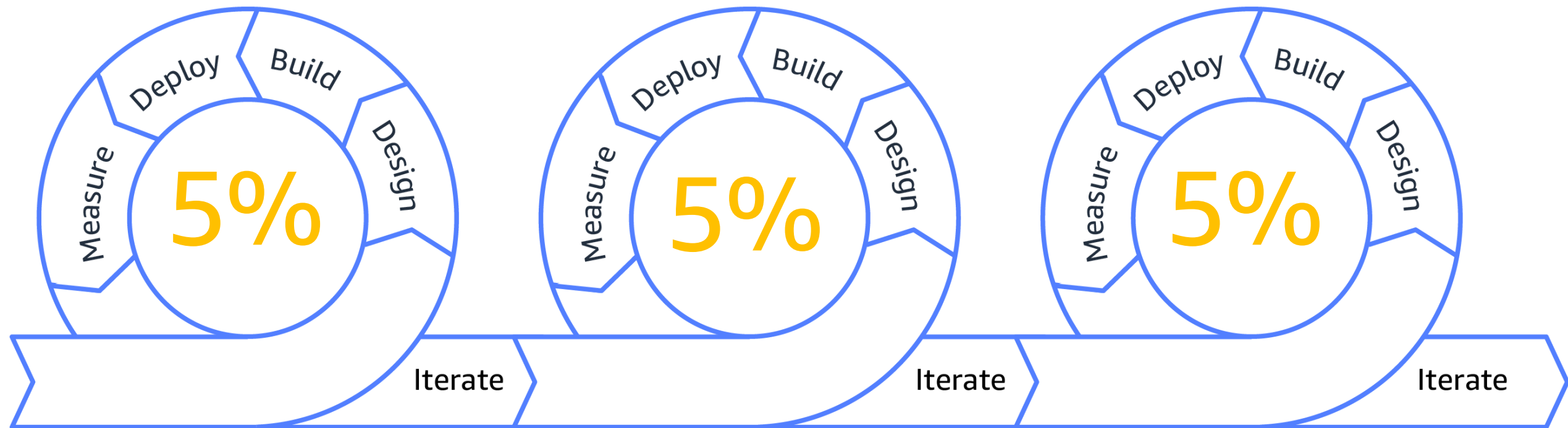


Architectures



Core foundations

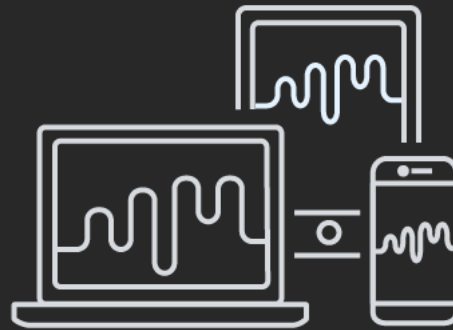
Development iterations



Sprints



Complete

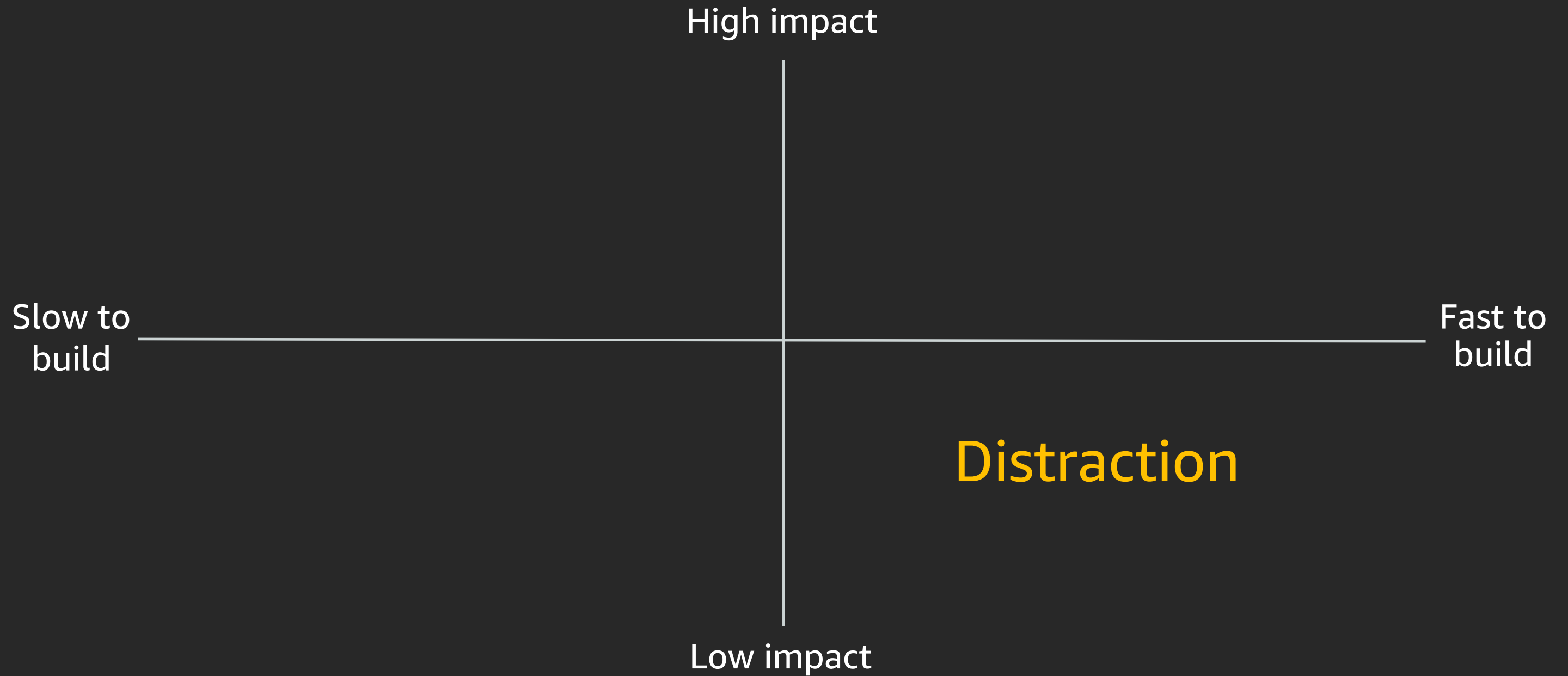


Uninterrupted

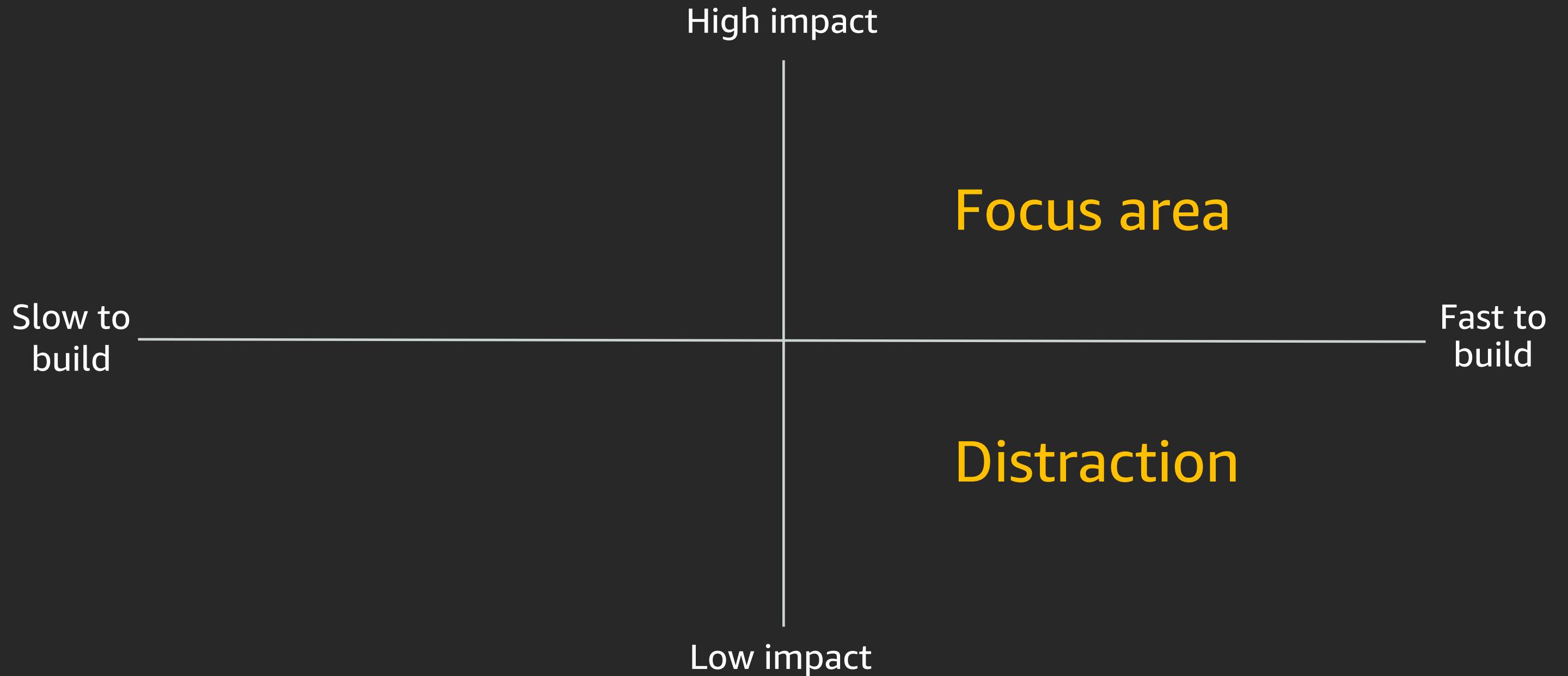


Short

Planning



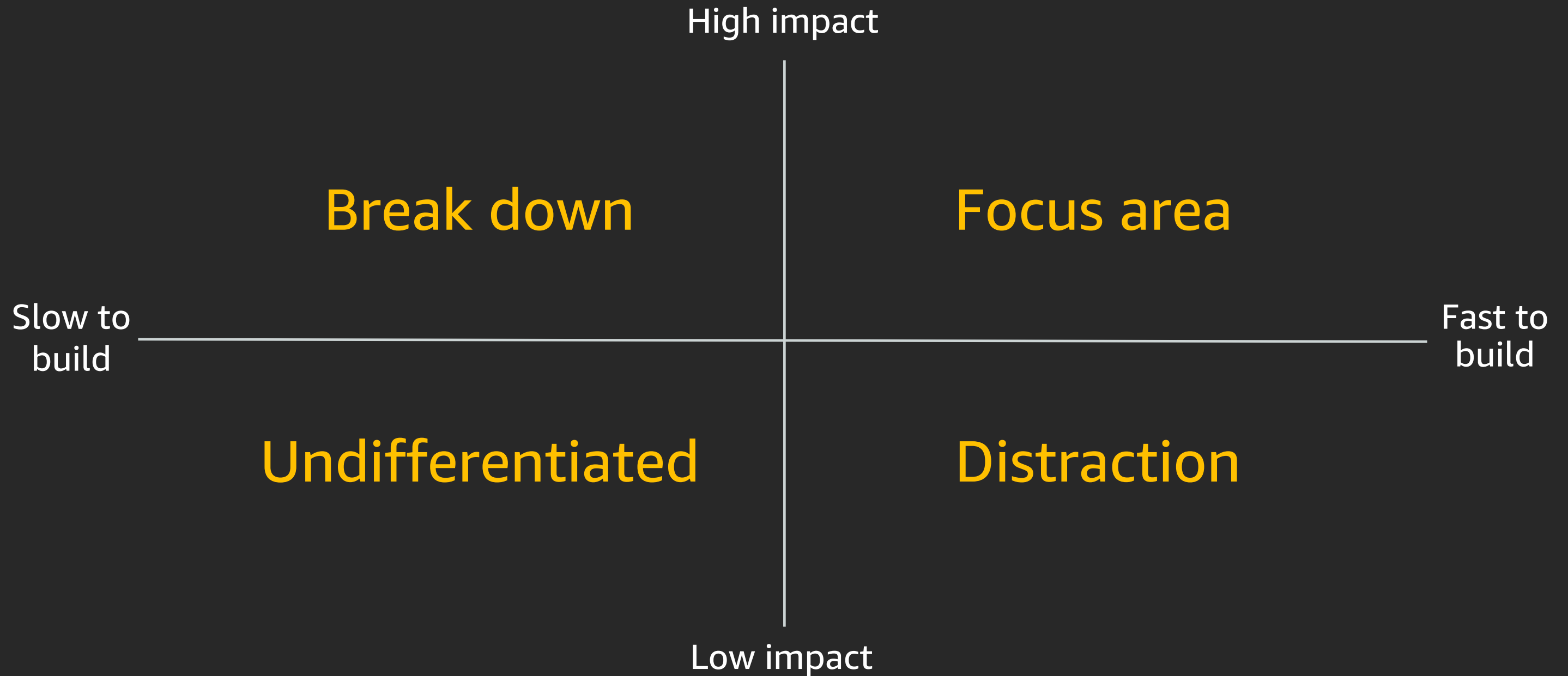
Planning



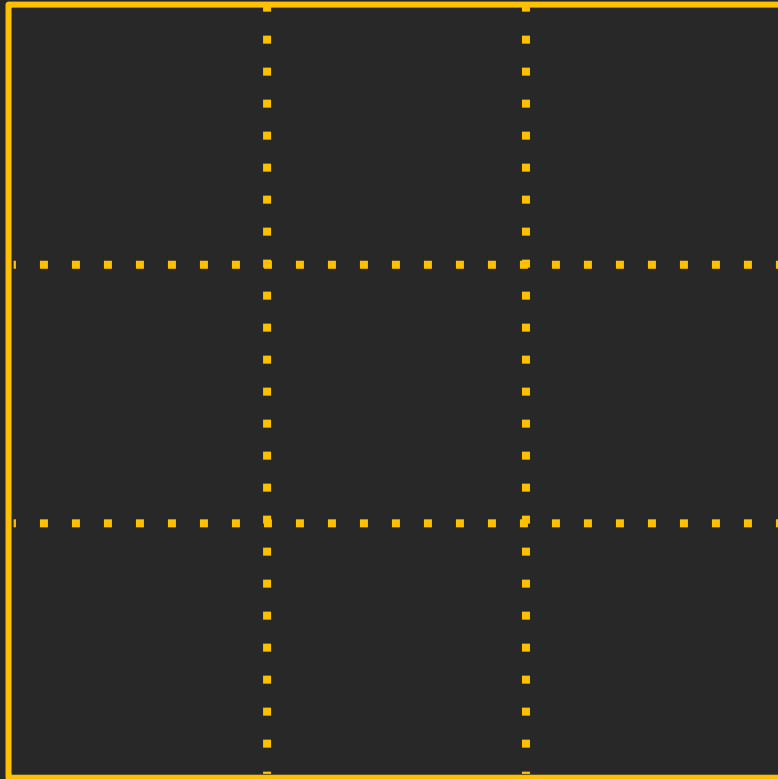
Planning



Planning



Minimize task size



One engineer can handle

Done in a single sprint

Adds incremental value

Easier to prioritize

Reducing scope removes ambiguity

Standups

Every day

First thing in the morning

Everyone

Tech, product, marketing, business

1 Minute

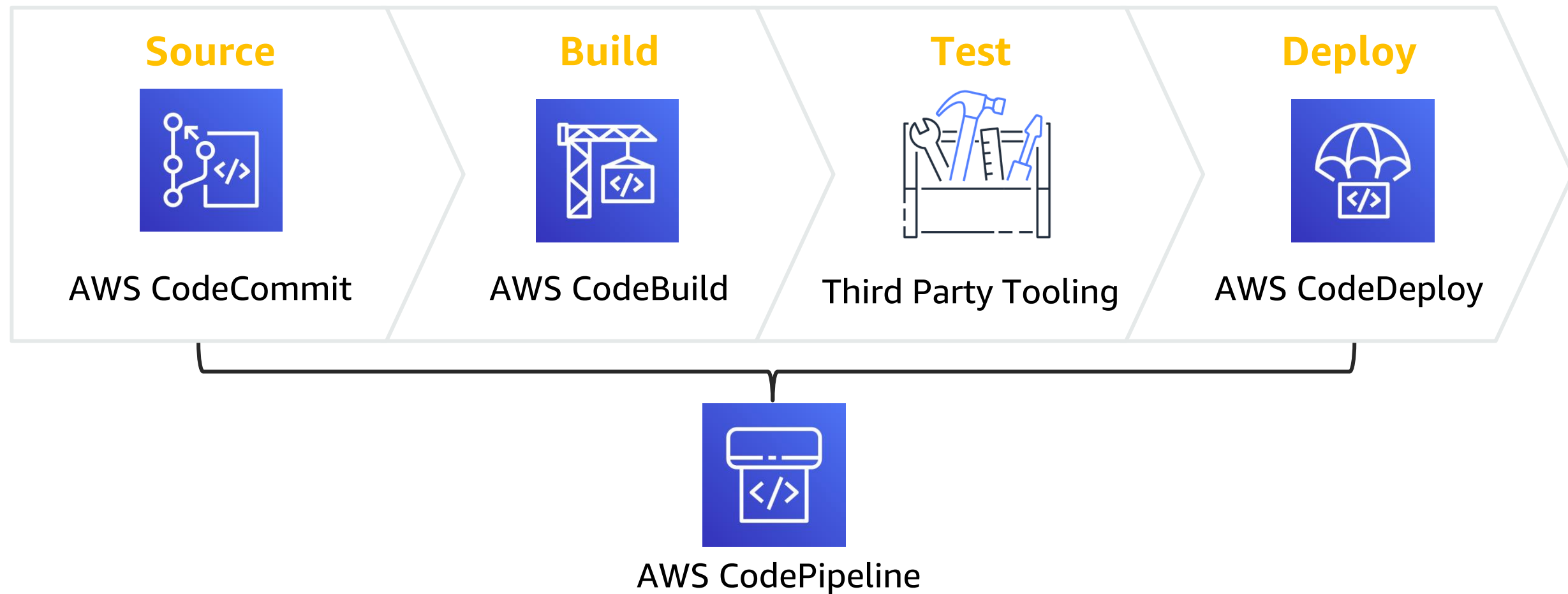
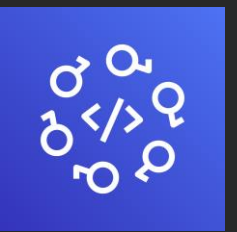
Per person max

15 Minutes

Max

Continuously ship releases

AWS CodeStar



What we'll cover



Development process



Anti-patterns

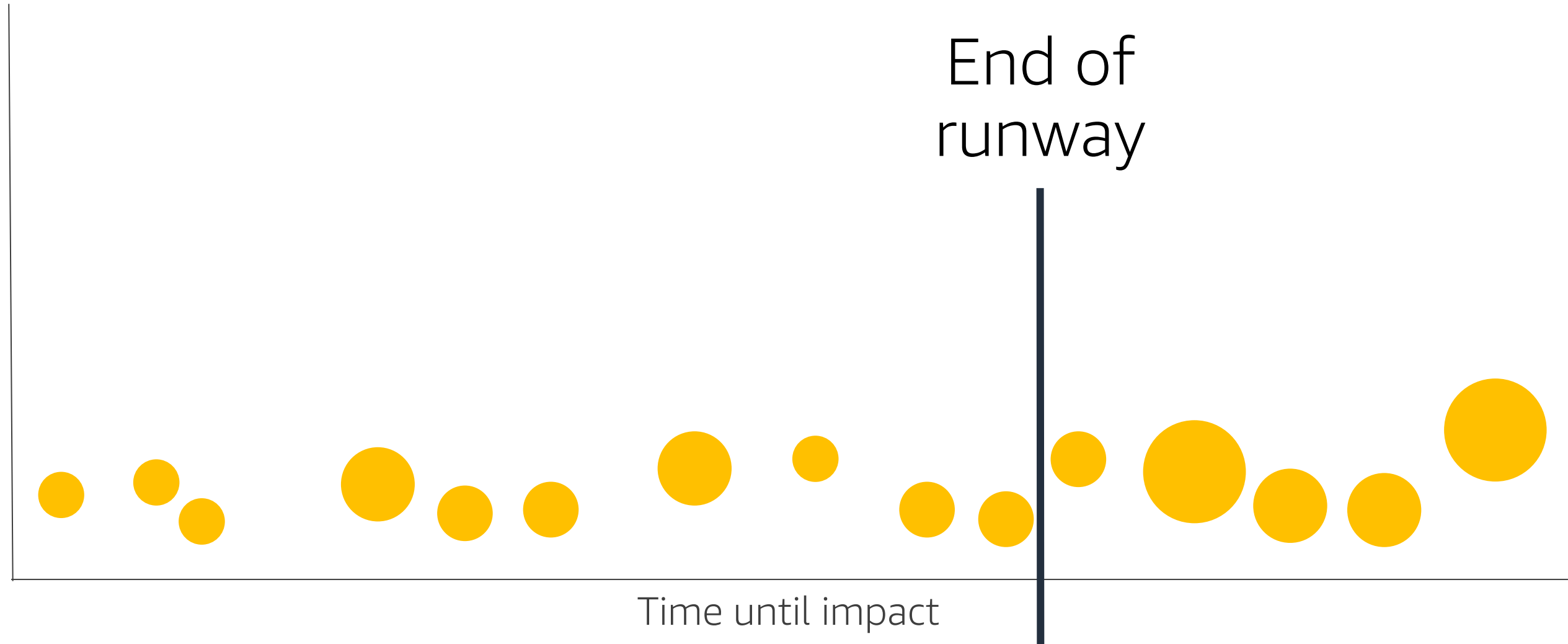


Architectures

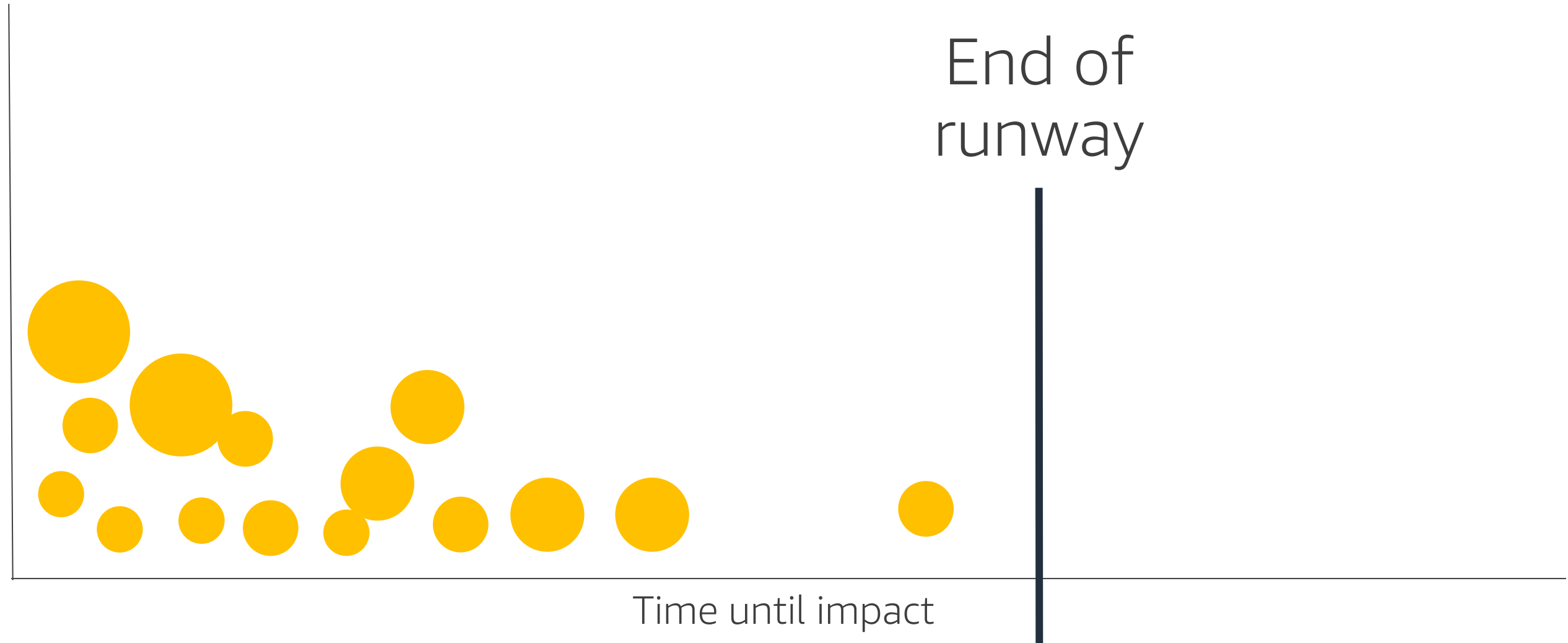


Core foundations

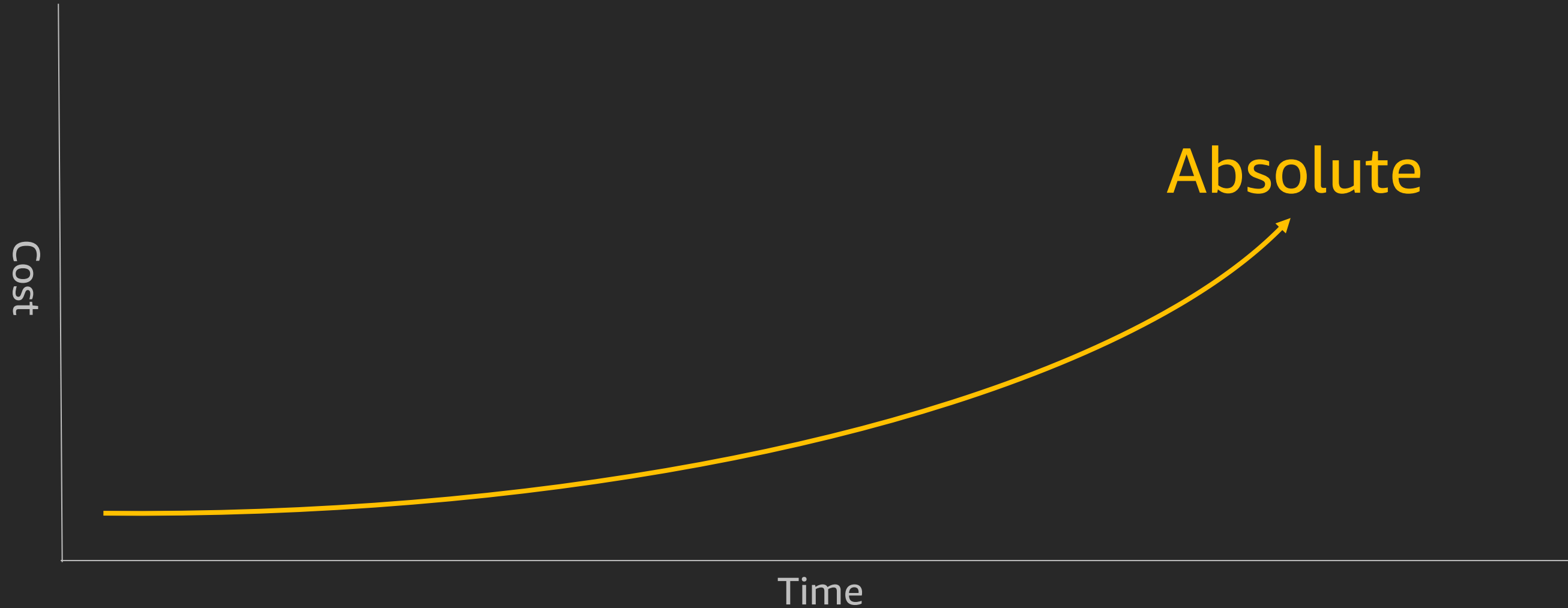
Roadmap prioritization



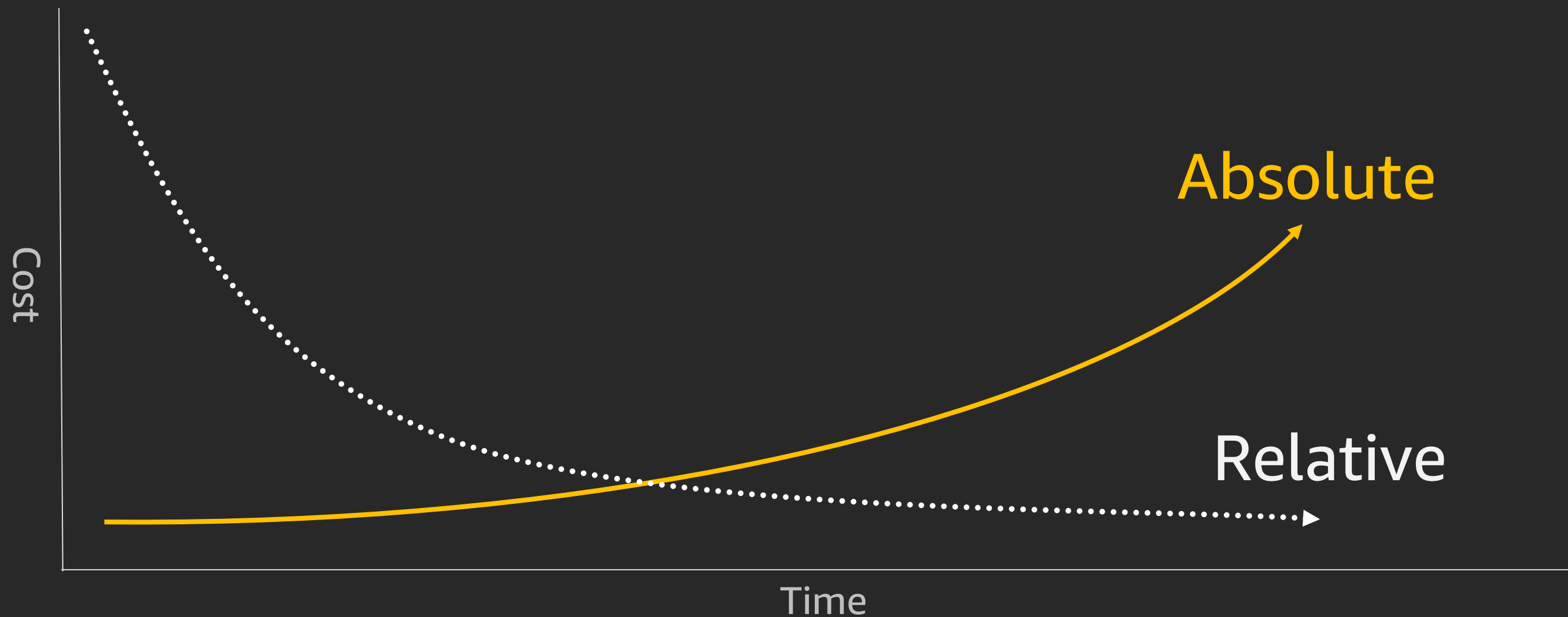
Roadmap prioritization



Technical debt

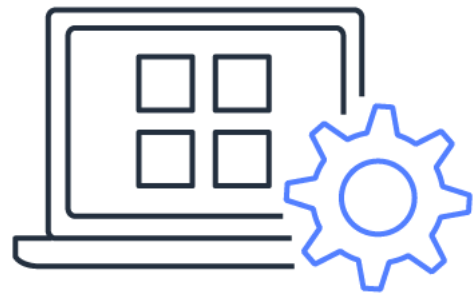


Technical debt



Abstractions

Easy



1 System

Hard

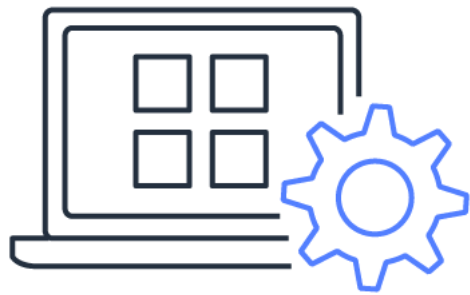


N Systems

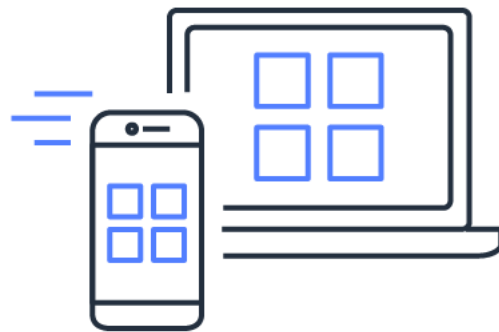
Abstractions

Easy

Hard



1 System

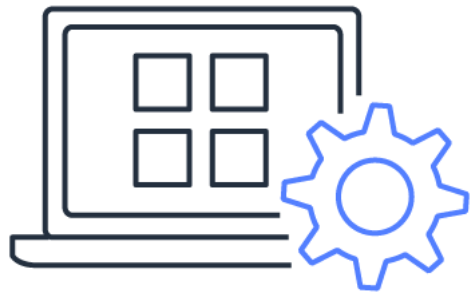


2 Systems



N Systems

Abstractions



1 System



2 Systems



N Systems

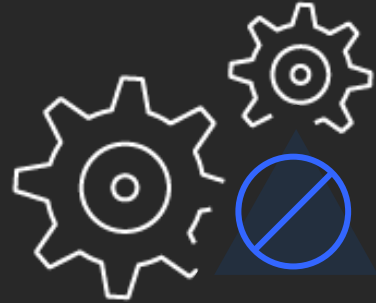
Rigid

Flexible

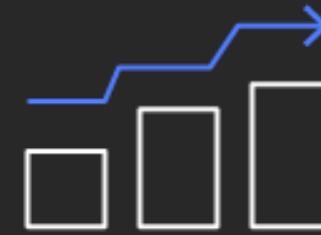
If you build it...

You must maintain it

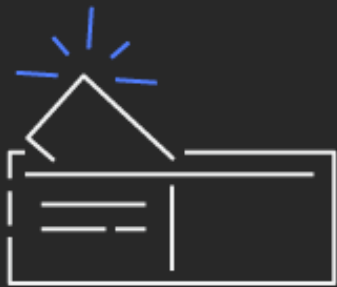
Build less, deliver more



No infrastructure provisioning,
no management



Scale automatically

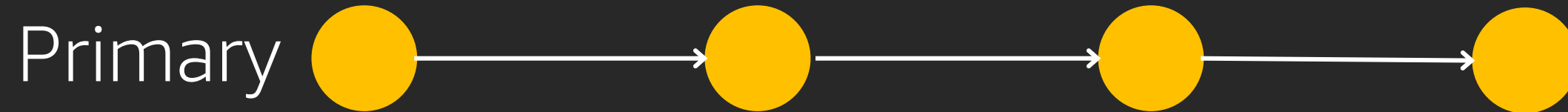


Pay for value

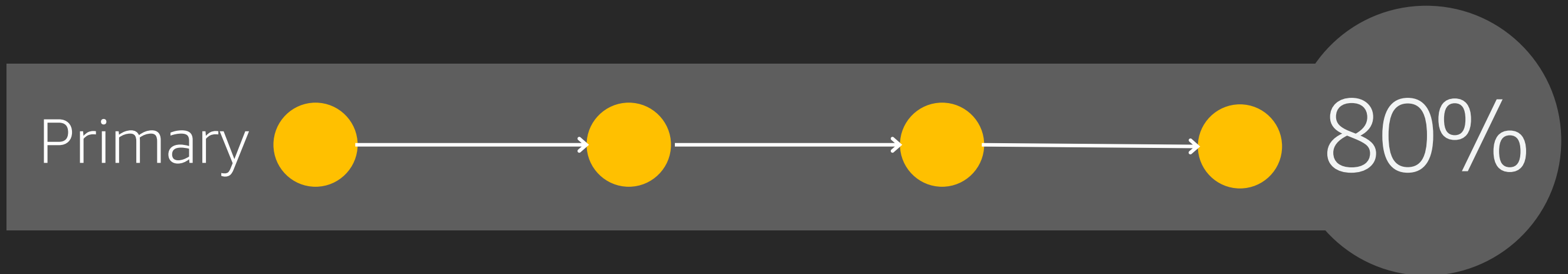


Highly available and secure

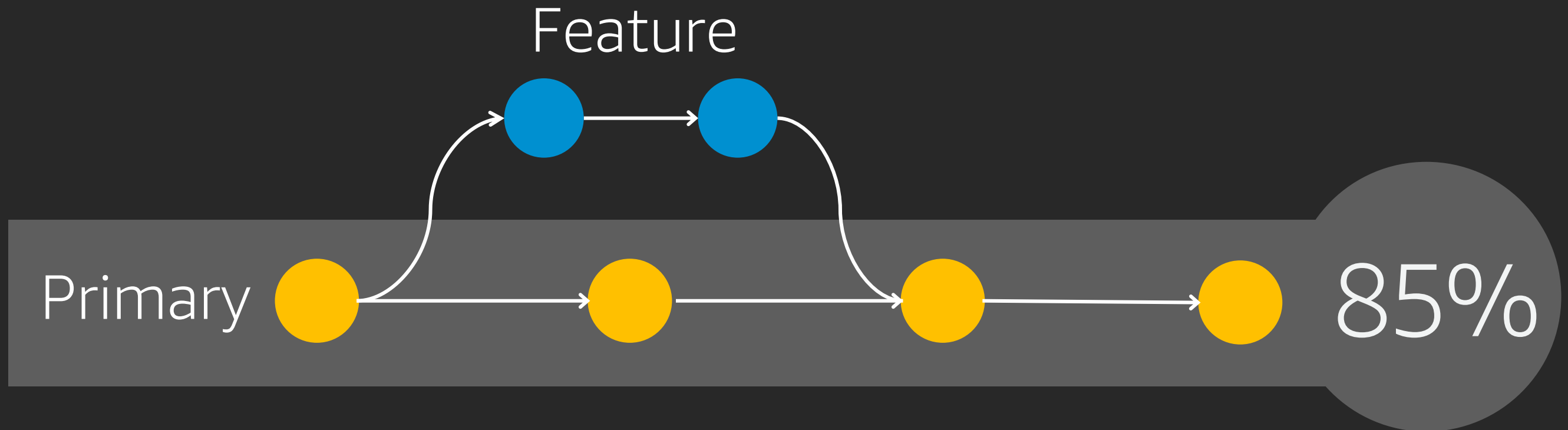
Tracer code



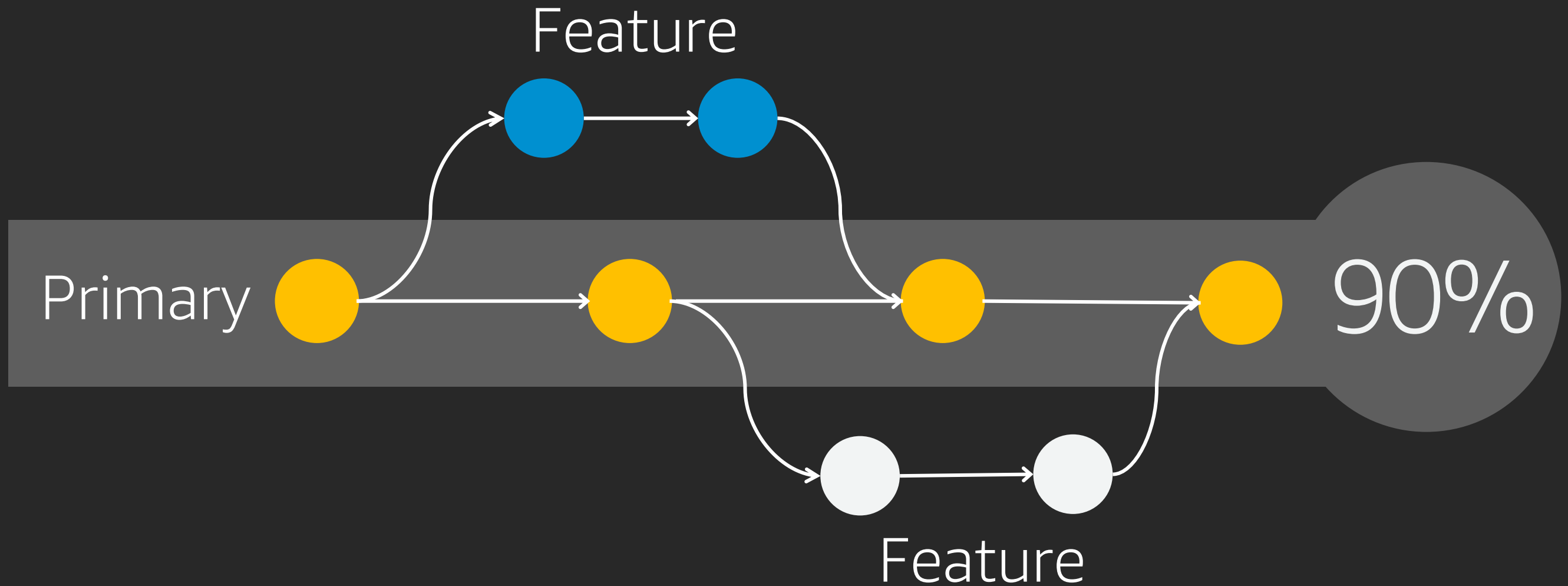
Tracer code



Tracer code



Tracer code



Optimize for learning



Learn



Experiment



Prototype

What we'll cover



Development process



Anti-patterns



Architectures



Core foundations

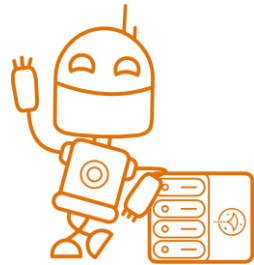
Monoliths are OK



Monolith
Does everything

Easiest way to get started

Amazon Lightsail



Computing
power



DNS
management



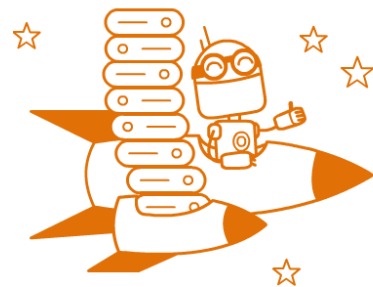
One static
IP/instance



Networking &
data transfer



Create larger
instances



Add attached
block storage



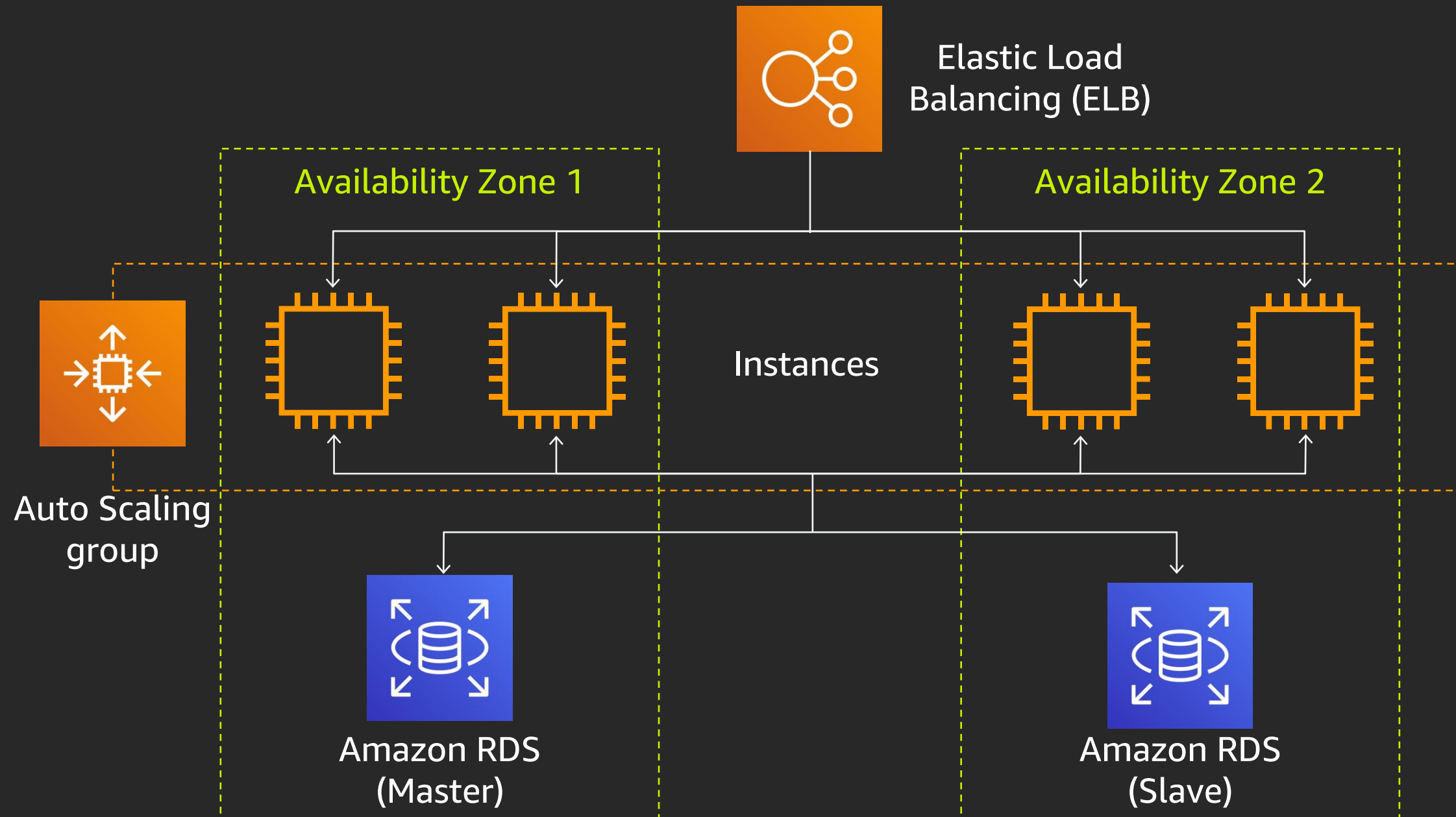
Load balance
your application



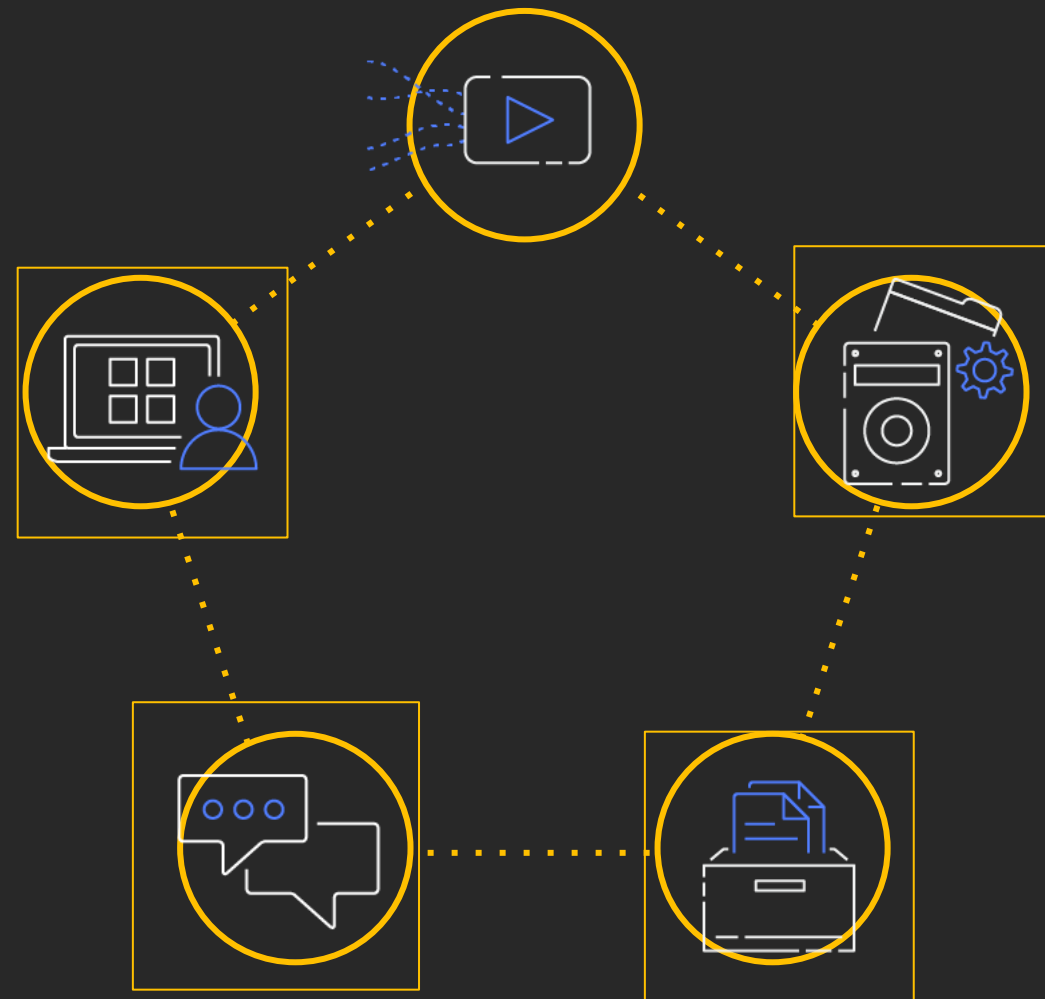
Connect to
AWS services

Deploy and scale monolith

AWS Elastic Beanstalk

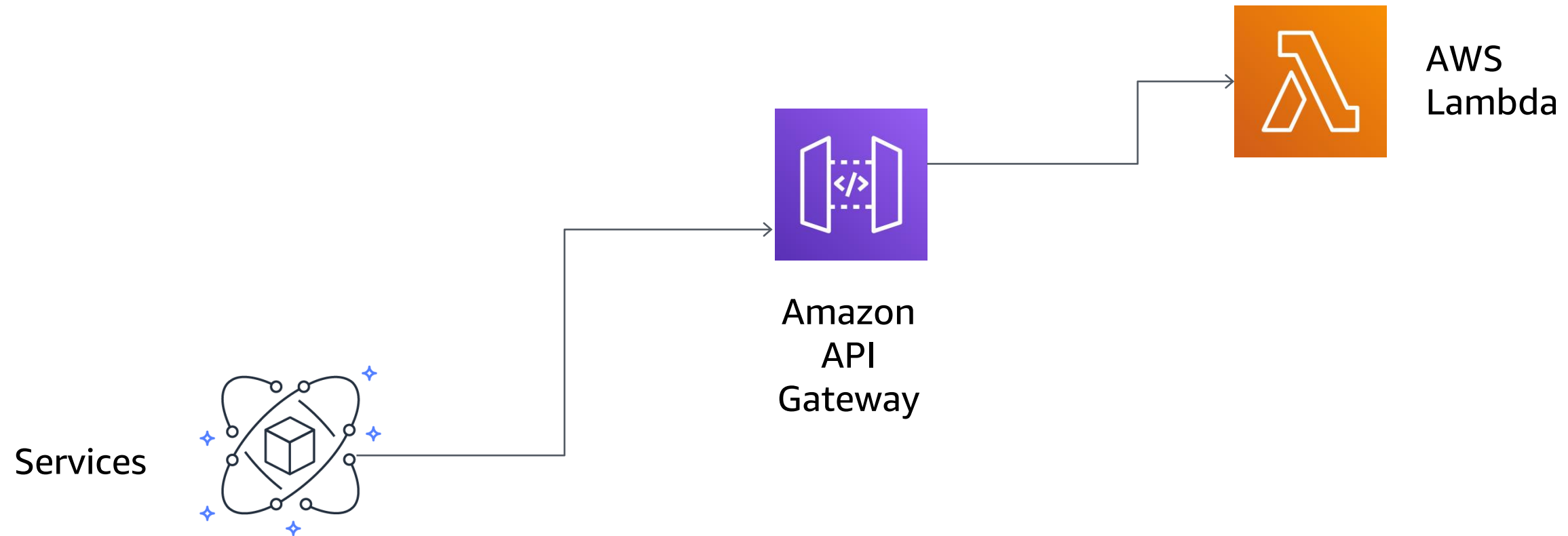


Microservices

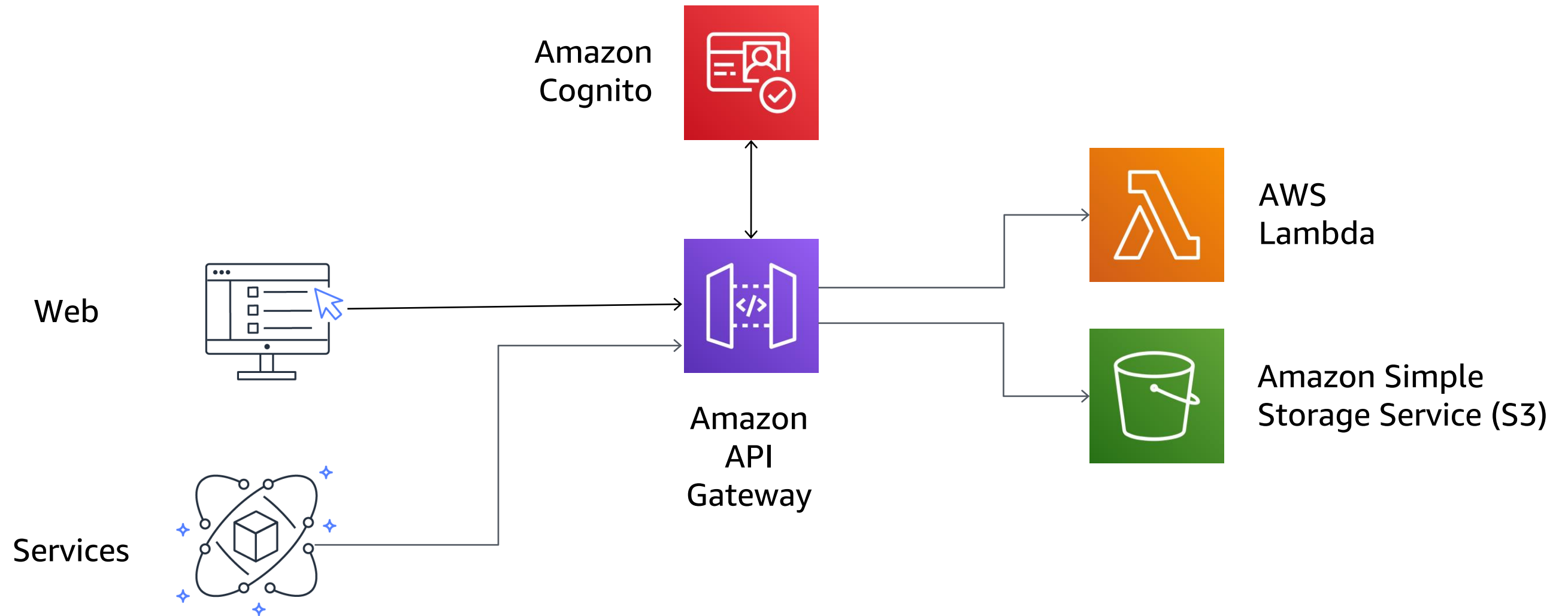


Microservices
Does one thing

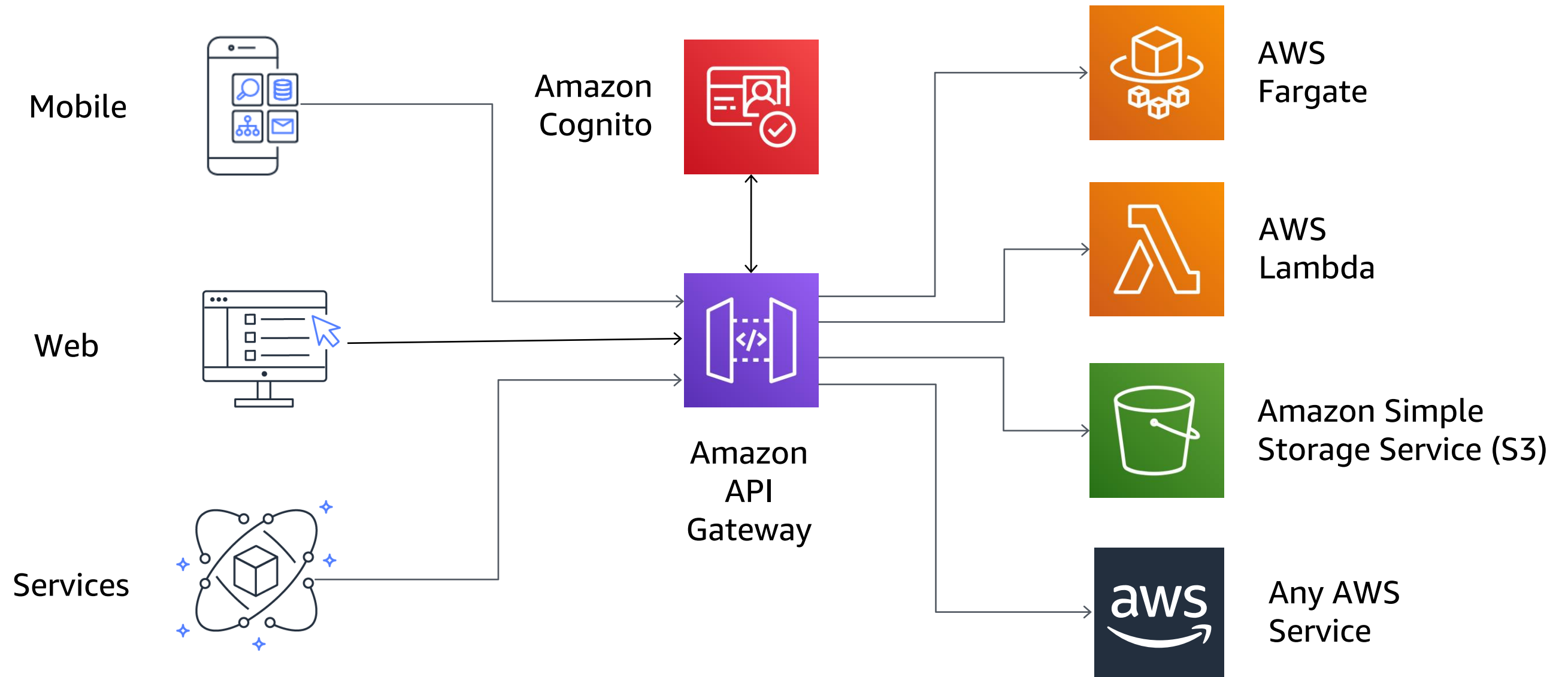
Building your API on AWS

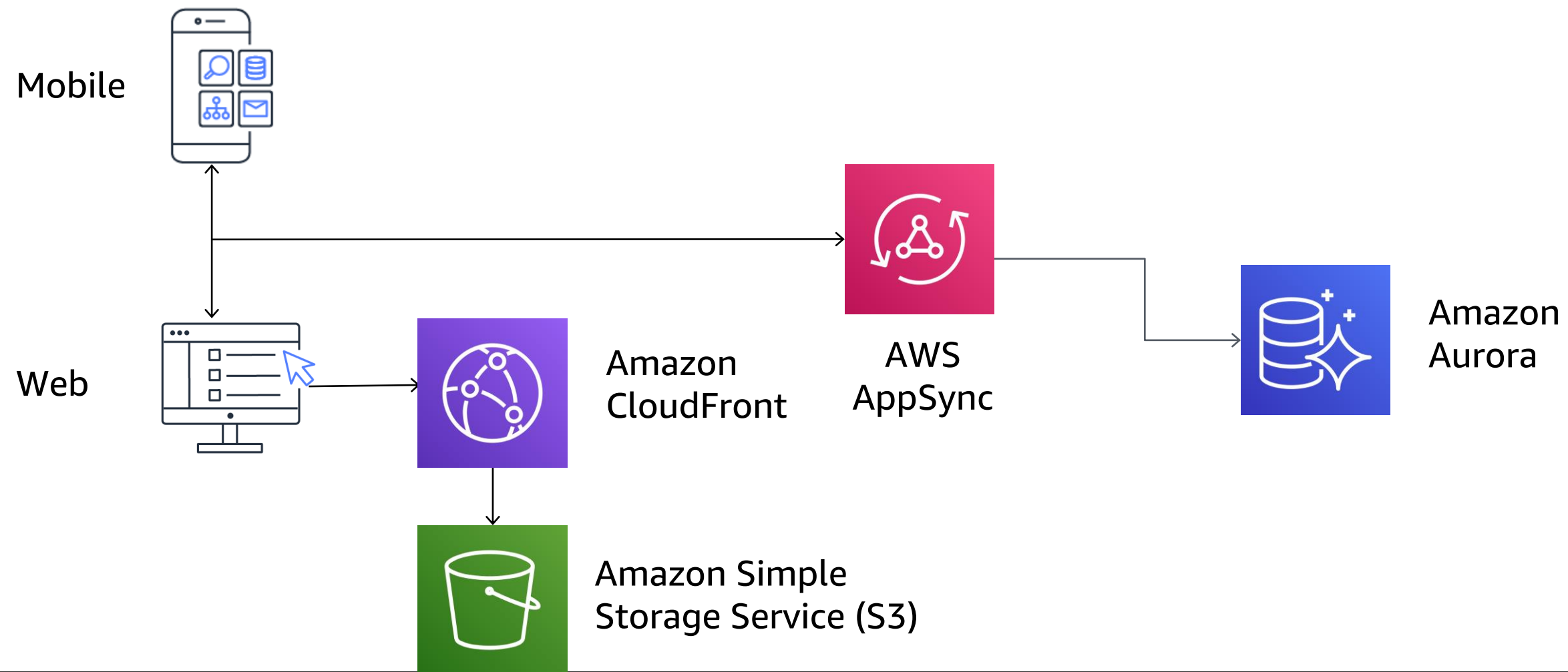


Building your API on AWS



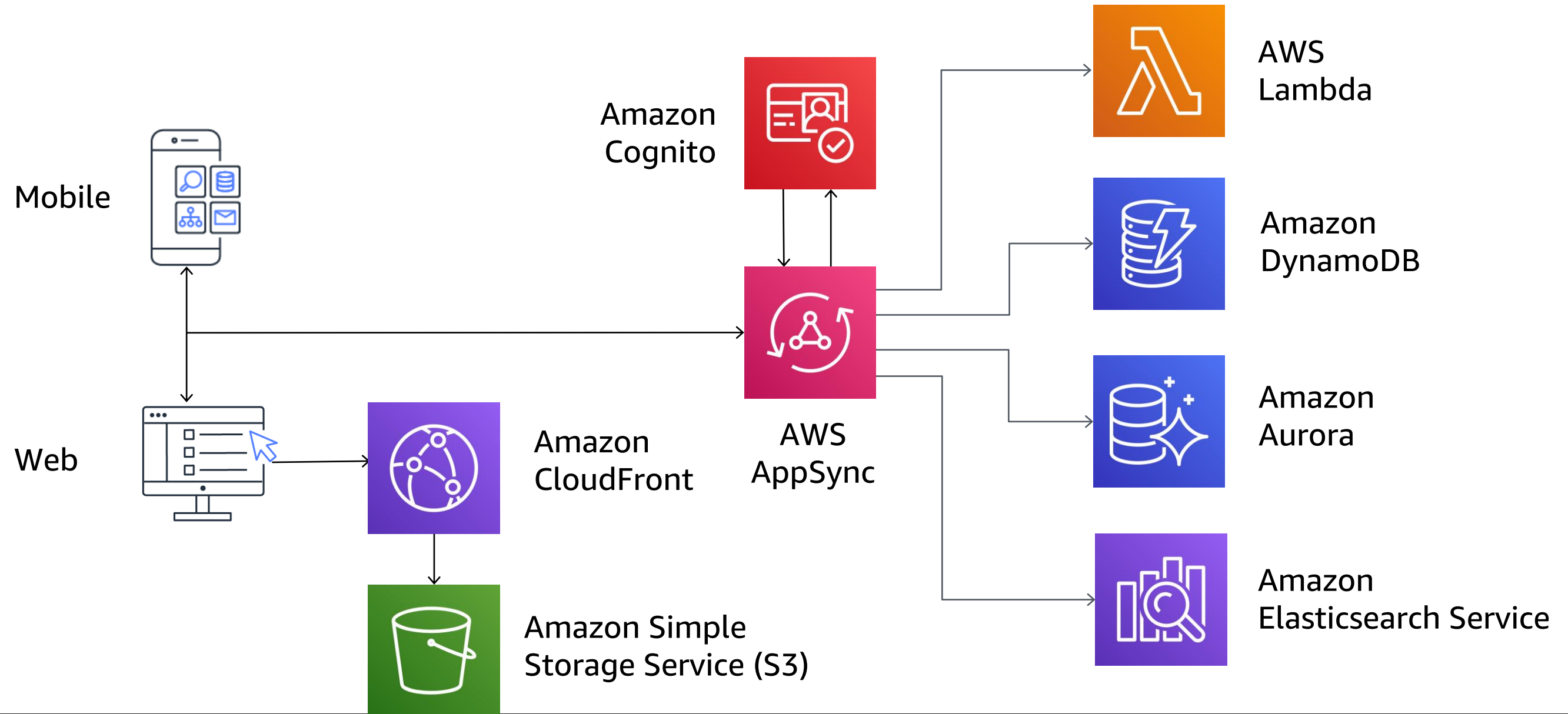
Building your API on AWS



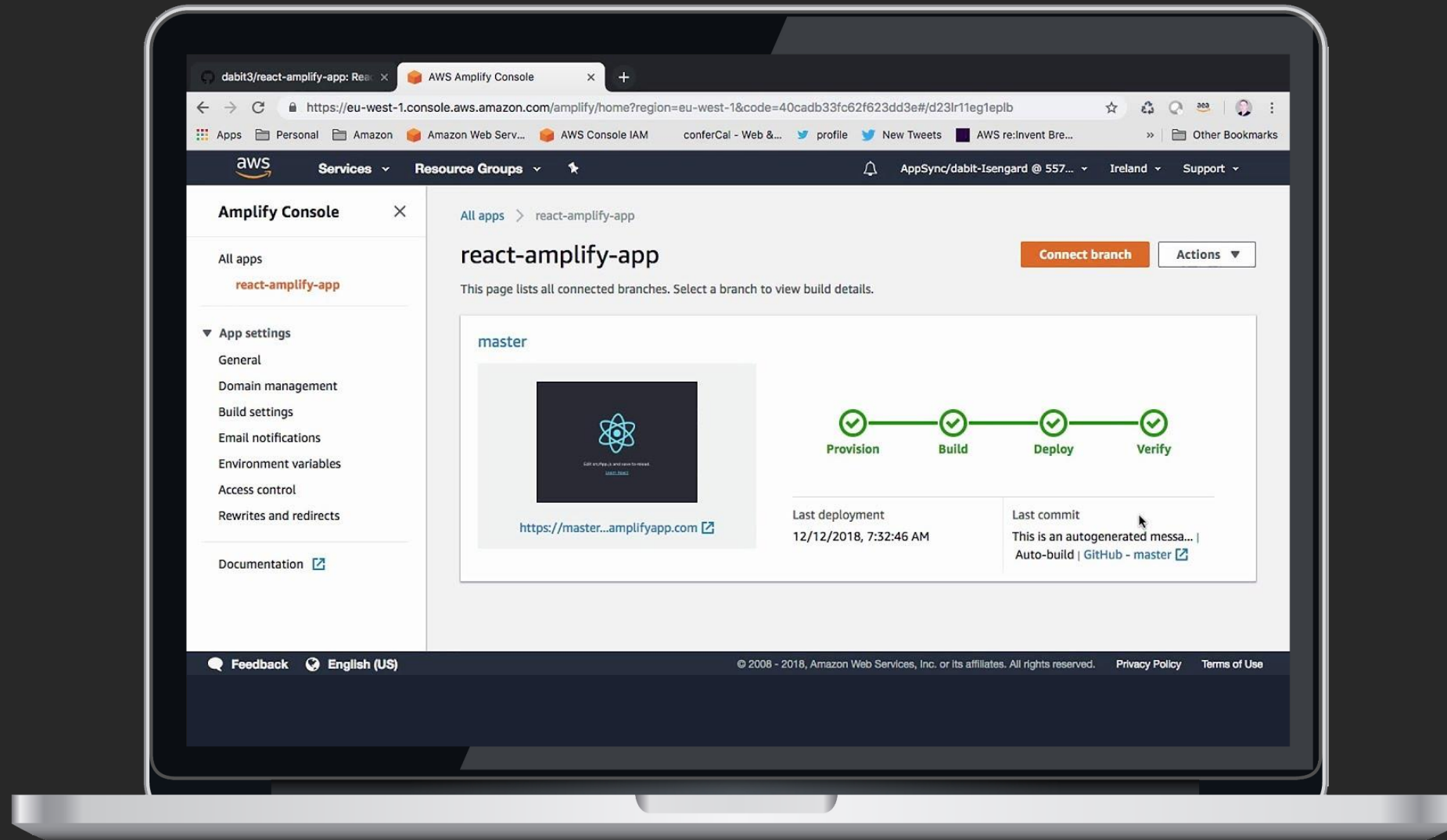


Mobile

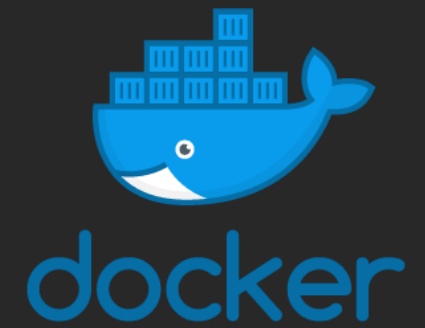
AWS Amplify



Static / single page web



Containers



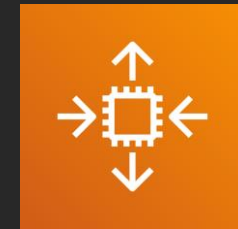
AWS Elastic
Beanstalk



Amazon Elastic
Container Service



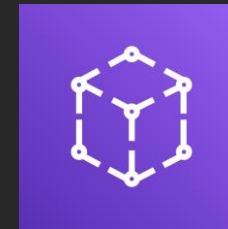
Amazon Elastic Container
Service for Kubernetes



AWS Fargate



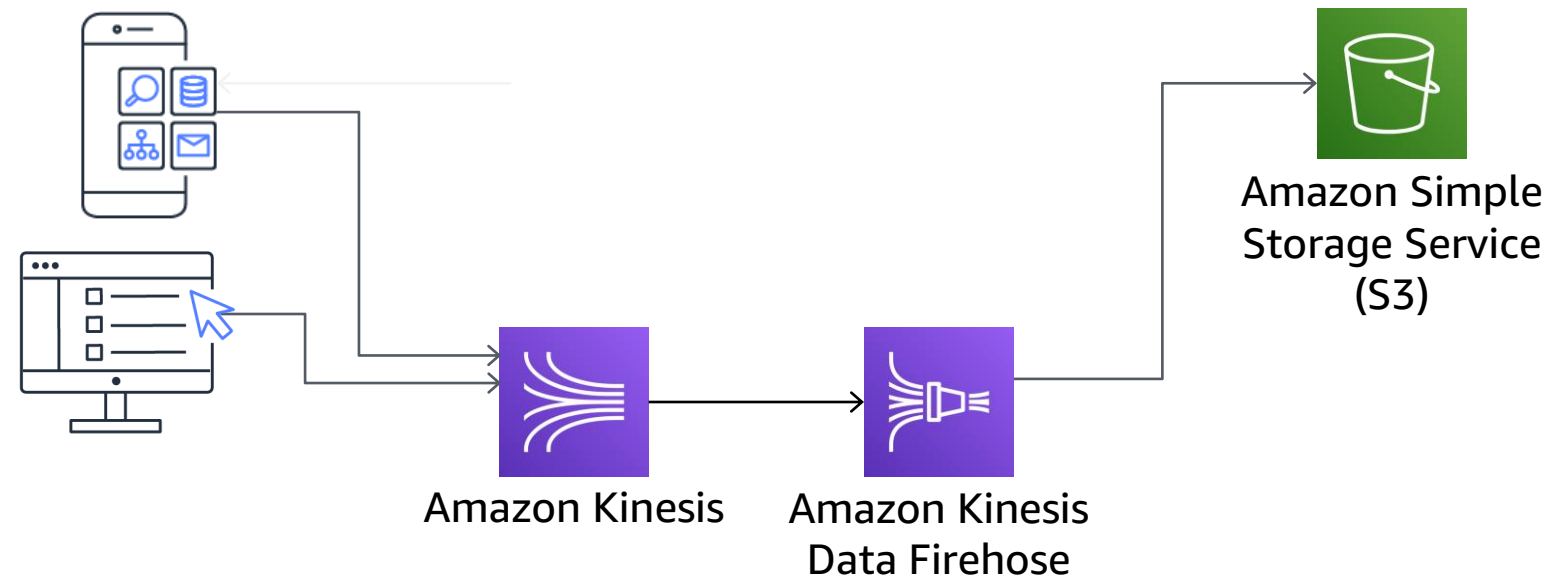
Amazon EC2
Container Registry



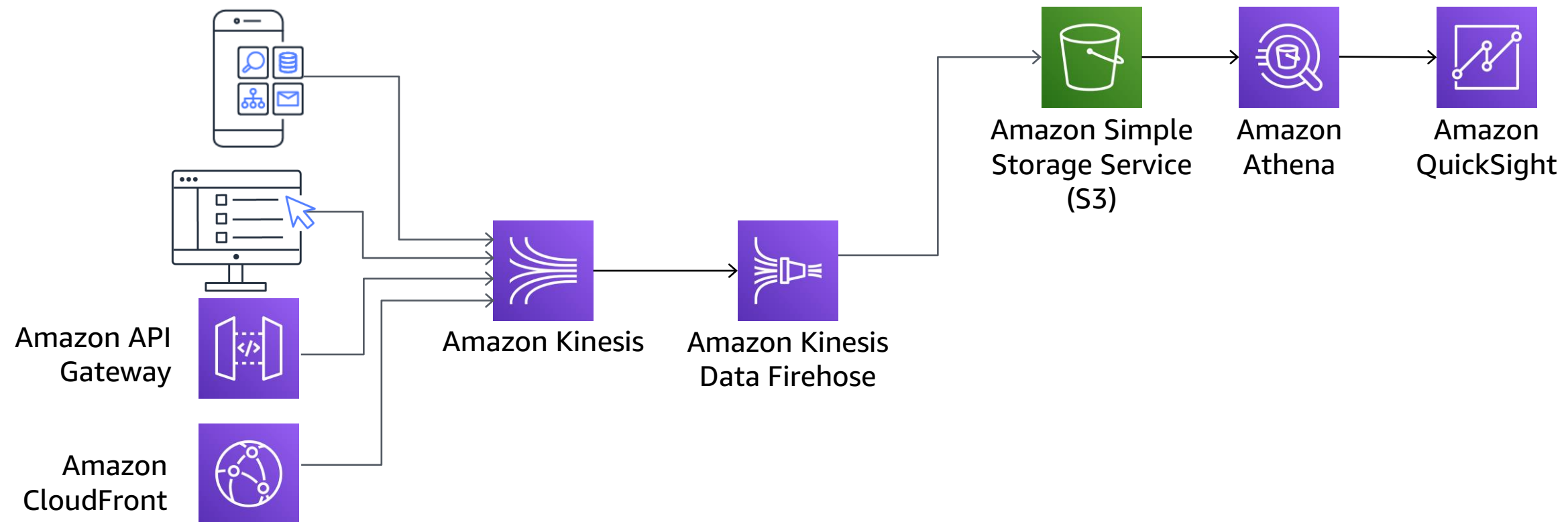
AWS App
Mesh



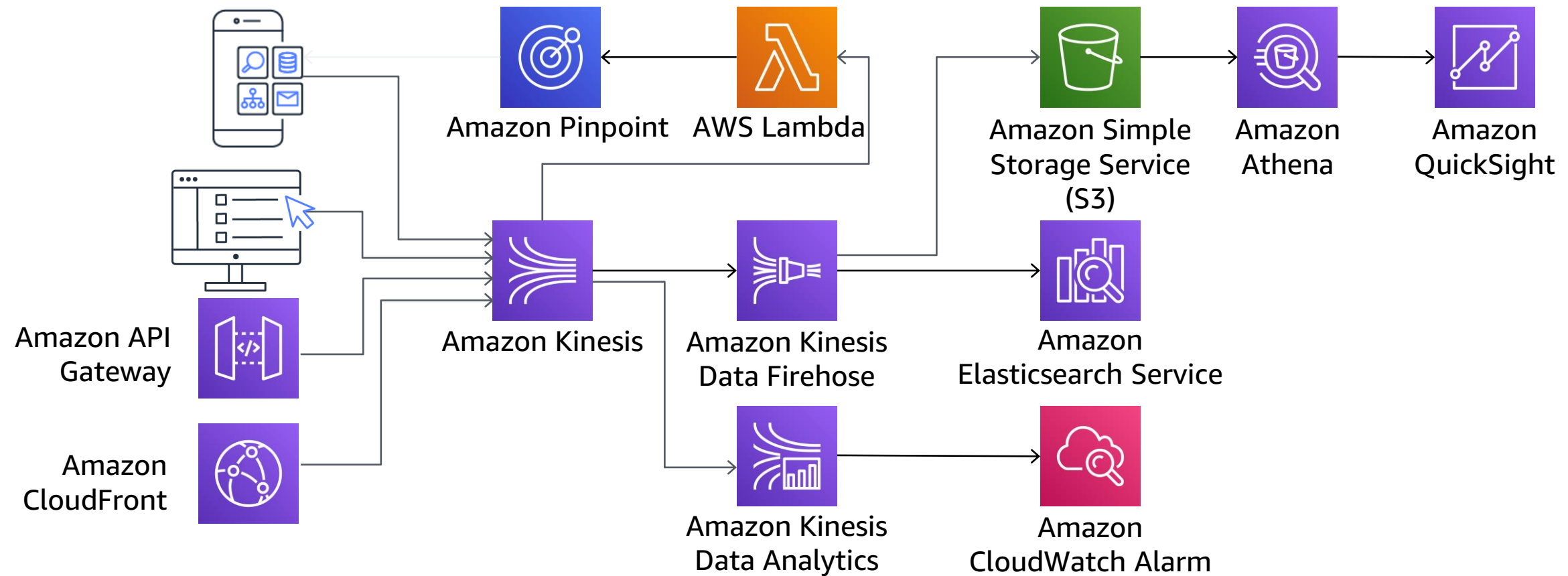
Analytics



Analytics



Analytics



What we'll cover



Development process



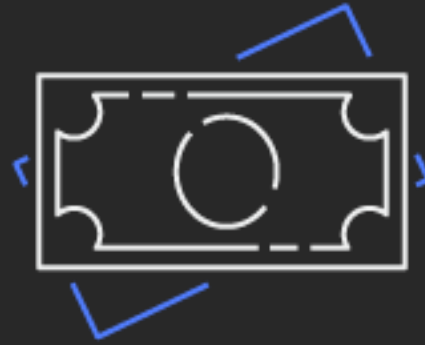
Anti-patterns



Architectures



Core foundations

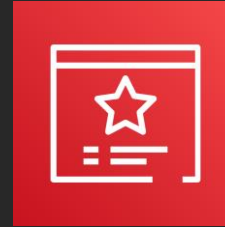


Over invest where
it is difficult to change

Security



Amazon Cognito



AWS Certificate
Manager



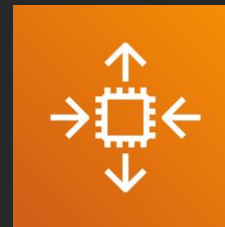
Amazon
CloudFront



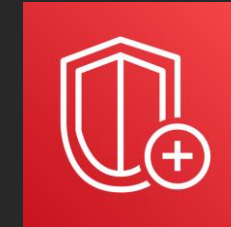
AWS Identity and
Access Management



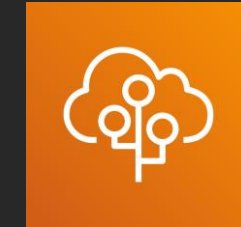
MFA token



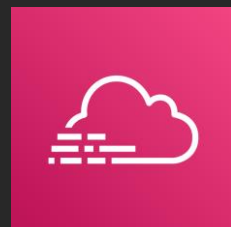
AWS Secrets Manager
(Parameter Store)



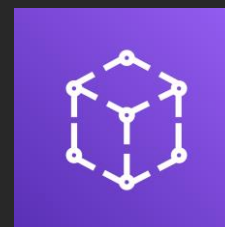
AWS Shield



AWS Systems Manager
(Session Manager)



AWS
CloudTrail

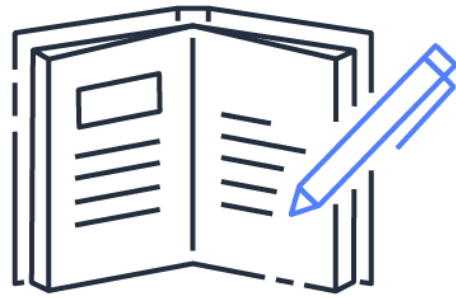


AWS Key
Management Service



AWS
Organizations

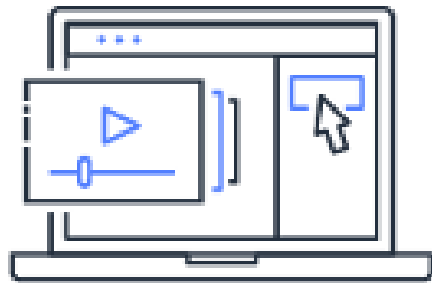
Integration points



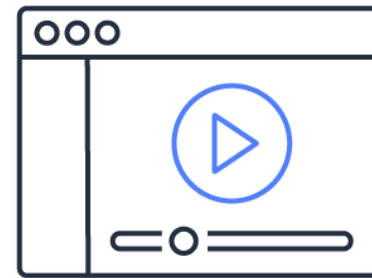
Documentation



Specification



Tutorials



Videos

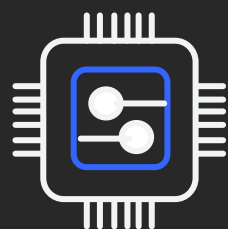


Quick Starts

Data model and store



Relational

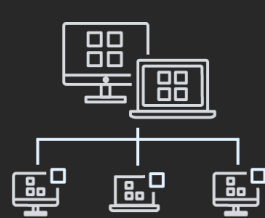


In-memory

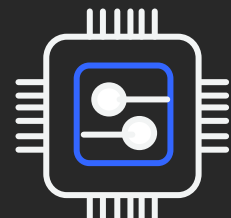
	Referential integrity, ACID transactions, schema-on-write	Query by key with microsecond latency					
Common Use Cases	Lift and shift, ERP, CRM, finance	Leaderboards, real-time analytics, caching					
AWS Service(s)	Amazon Aurora, Amazon RDS	Amazon ElastiCache					



Data model and store



Relational



In-memory



Key-value



Document

Referential integrity, ACID transactions, schema-on-write

Query by key with microsecond latency

High throughput, low-latency reads and writes, endless scale

Store documents and quickly access querying on any attribute

Common Use Cases

Lift and shift, ERP, CRM, finance

Leaderboards, real-time analytics, caching

Real-time bidding, shopping cart, social, product catalog, customer preferences

Content management, personalization, mobile

AWS Service(s)

Amazon Aurora, Amazon RDS

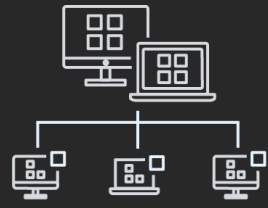
Amazon ElastiCache

Amazon DynamoDB

Amazon DocumentDB



Data model and store

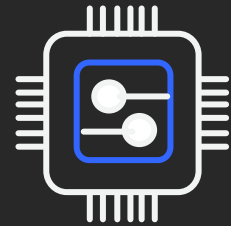


Relational

Referential integrity, ACID transactions, schema-on-write

Lift and shift, ERP, CRM, finance

Amazon Aurora, Amazon RDS



In-memory

Query by key with microsecond latency

Leaderboards, real-time analytics, caching

Amazon ElastiCache

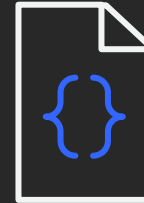


Key-value

High throughput, low-latency reads and writes, endless scale

Real-time bidding, shopping cart, social, product catalog, customer preferences

Amazon DynamoDB



Document

Store documents and quickly access querying on any attribute

Content management, personalization, mobile

Amazon DocumentDB



Graph

Quickly and easily create and navigate relationships between data

Fraud detection, social networking, recommendation engine

Amazon Neptune



Time-series

Collect, store, and process data sequenced by time

IoT applications, event tracking

Amazon Timestream



Ledger

Complete, immutable, and verifiable history of all changes to application data

Systems of record, supply chain, health care, registrations, financial

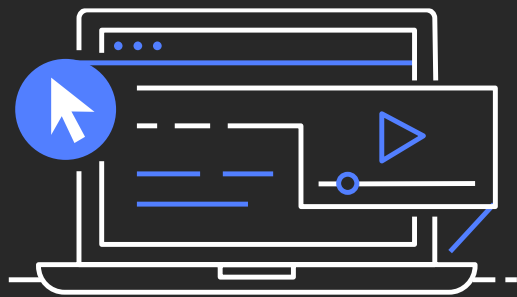
Amazon QLDB

Common Use Cases

AWS Service(s)



AWS Digital Training



Flexibility to Learn Your Way

Build cloud skills with 550+ free digital training courses, or dive deep with classroom training

Featured Courses

- [AWS Cloud Practitioner Essentials \(Second Edition\)](#)
Learn the fundamentals of the AWS Cloud and prepare for the AWS Certified Cloud Practitioner exam.
- [Amazon DynamoDB for Serverless Architectures](#)
An introduction to Amazon DynamoDB and how it's leveraged in building a serverless architecture.
- [AWS Security Fundamentals](#)
Learn fundamental cloud computing and AWS security concepts, including AWS access control and management, governance, logging, and encryption methods.
- [Getting Started with Amazon Simple Storage Service \(Amazon S3\)](#)
The course provides you with the knowledge to determine when to use Amazon S3 by reviewing typical use cases and understanding how the service provides object storage for your applications.

Thank you for attending AWS Builders Online Series

We hope you found it interesting! A kind reminder to **complete the survey**.
Let us know what you thought of today's event and how we can improve the event experience for you in the future.



aws-apac-marketing@amazon.com



twitter.com/AWSCloud



facebook.com/AmazonWebServices



youtube.com/user/AmazonWebServices



slideshare.net/AmazonWebServices



twitch.tv/aws



Builders Online Series

Thank you

Rob De Feo
Startup Evangelist
Amazon Web Services



@robdefeo