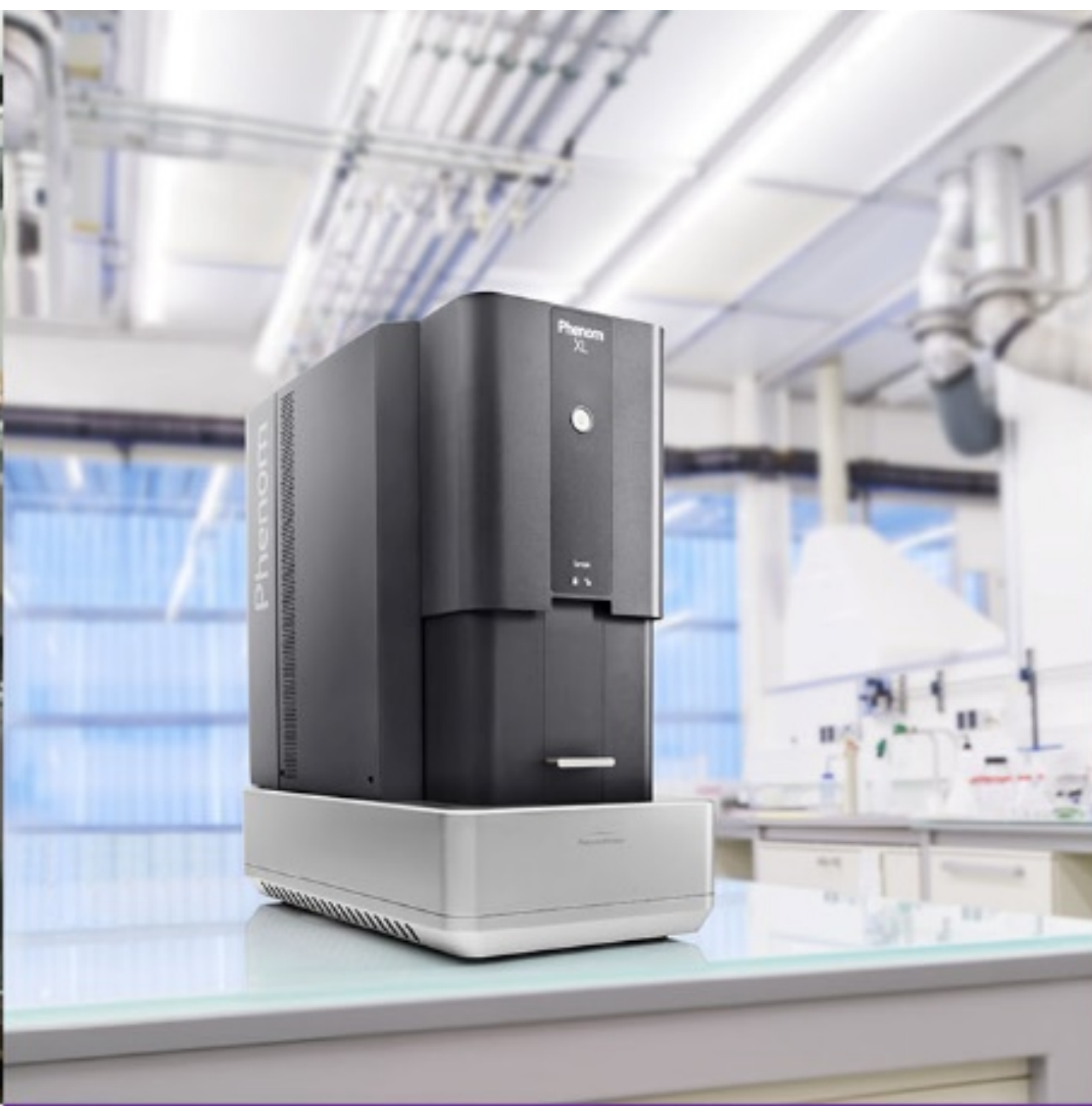
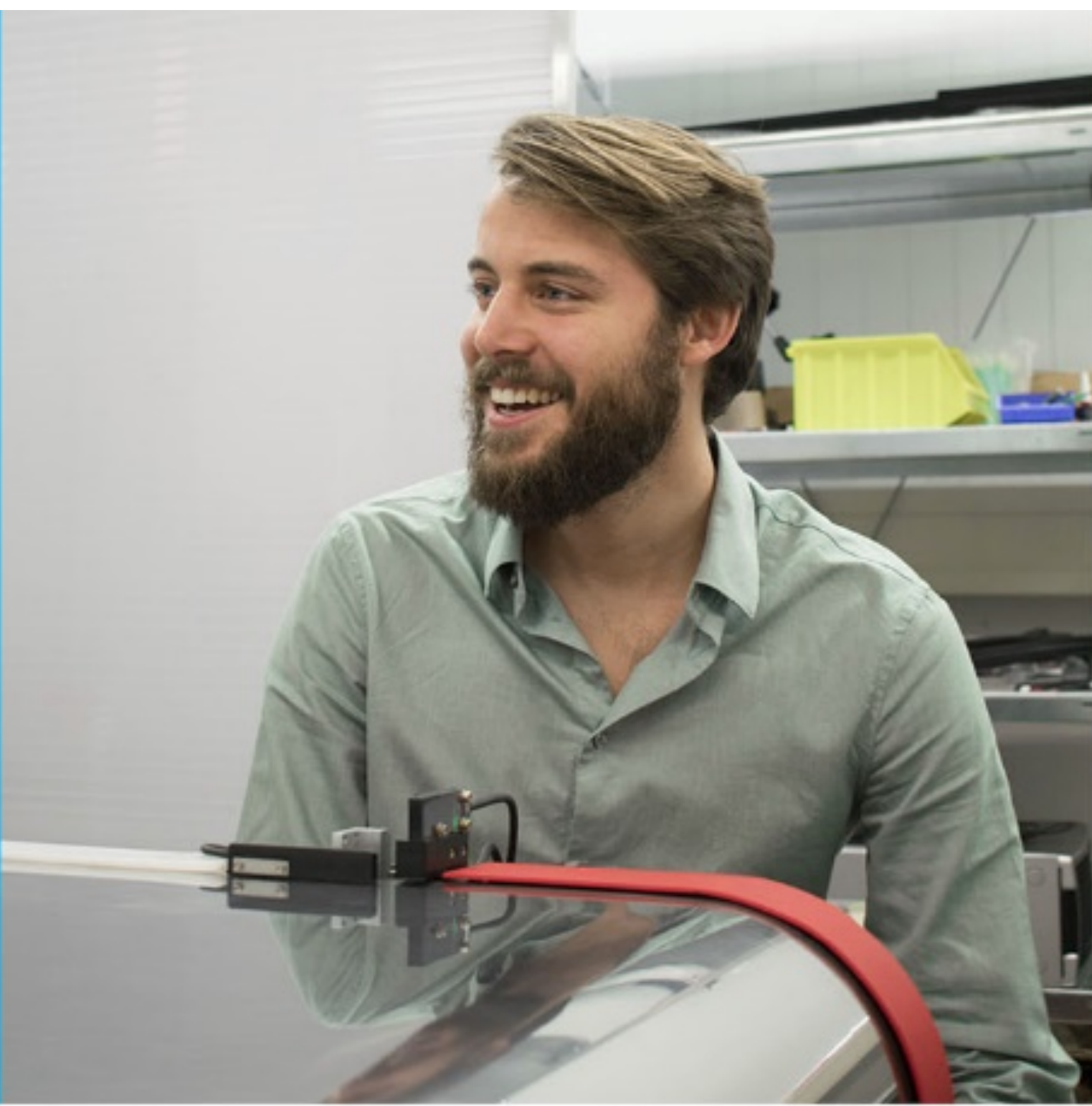





MECHATRONICS



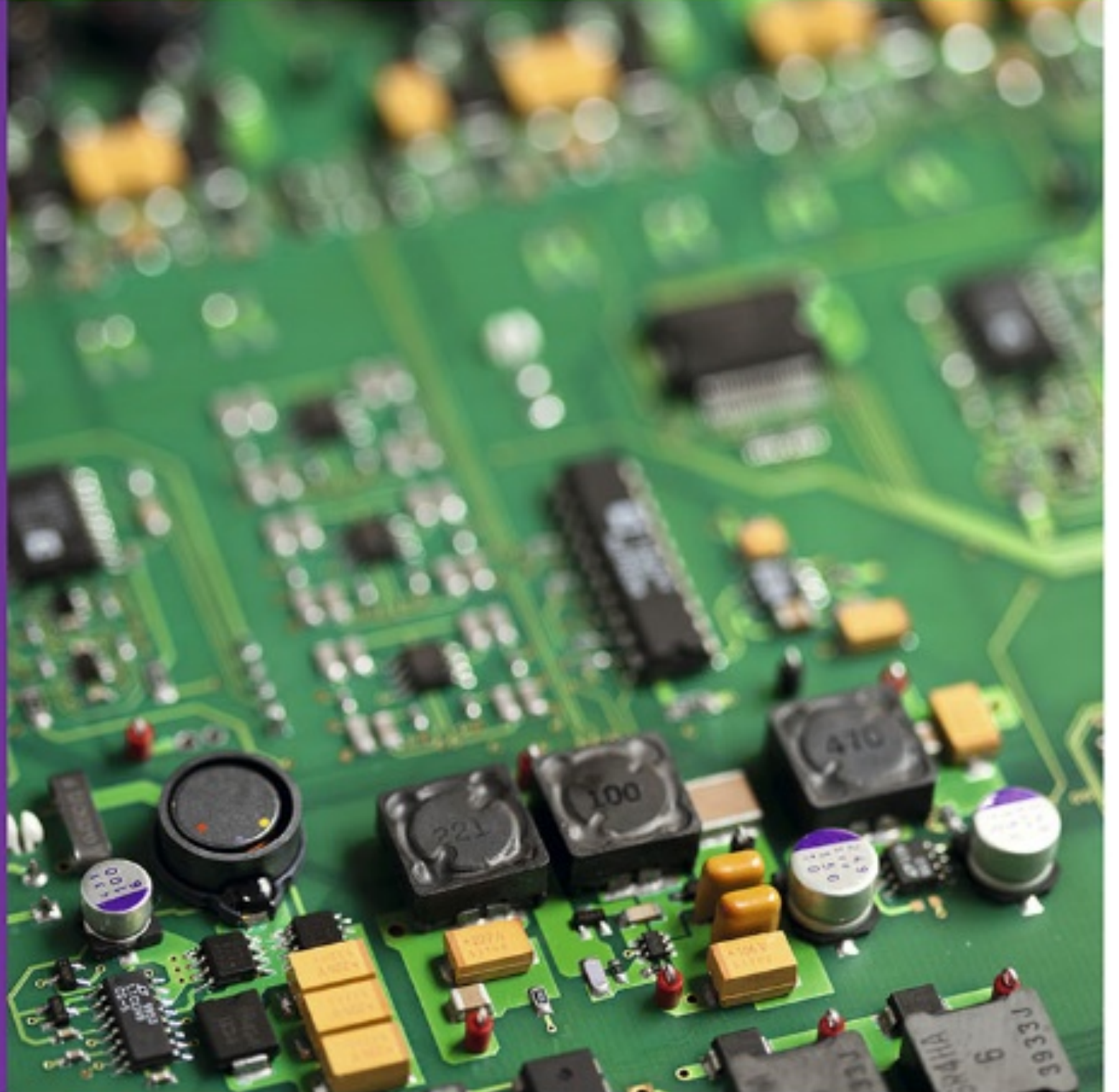
MATHWARE

ASSEMBLY

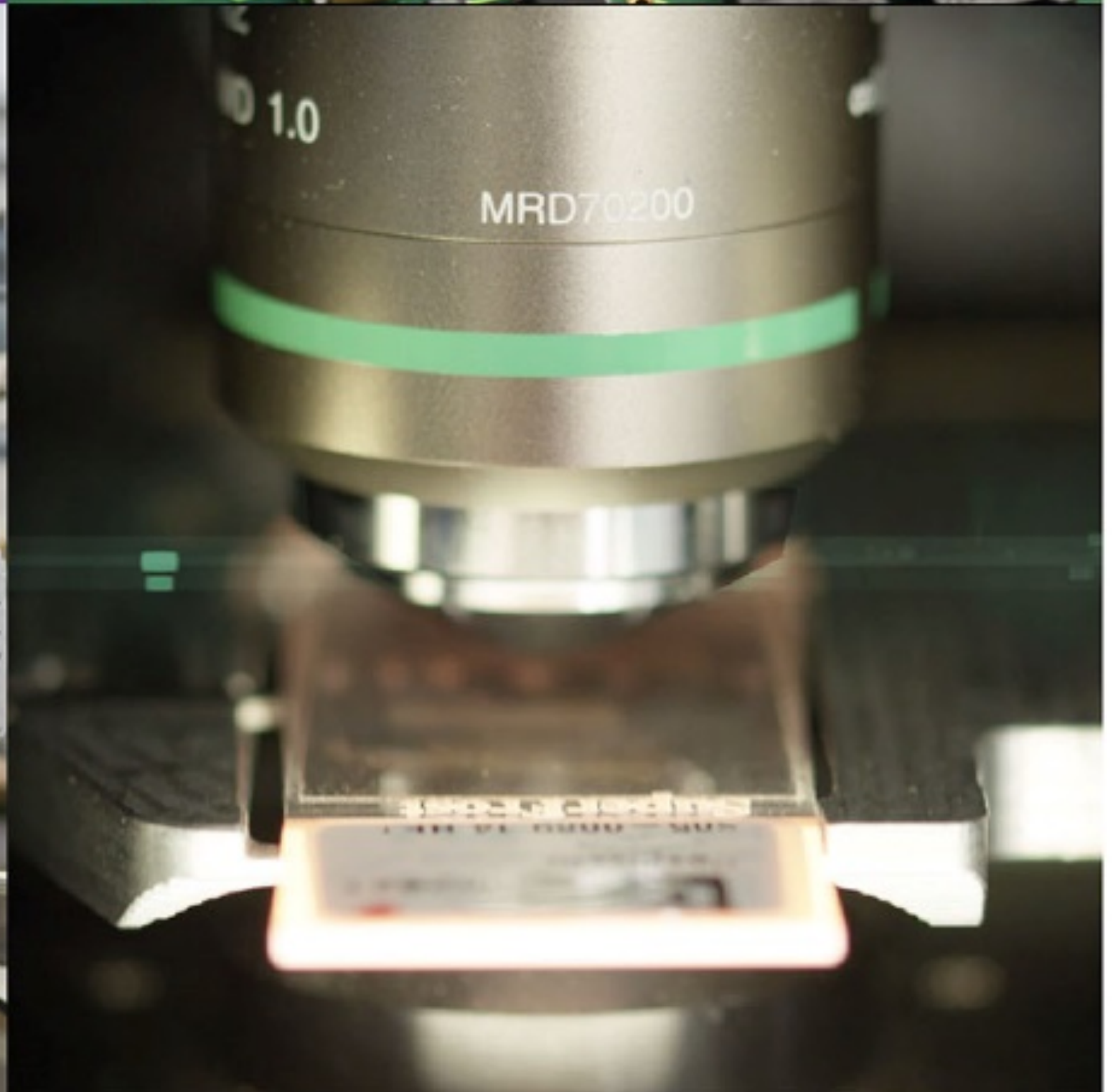
SIoux
TECHNOLOGIES



ELECTRONICS



SOFTWARE



We bring high-tech to life

Hot or Not

Trends in Cloud technology



WELCOME TO SIOUX LABS



MEET

Meet, organize and relax outside the office. Sioux Labs is the livingroom for your network!



EAT

A gym for the mind means food for thought! Sioux Labs feeds the brain and the belly 😊

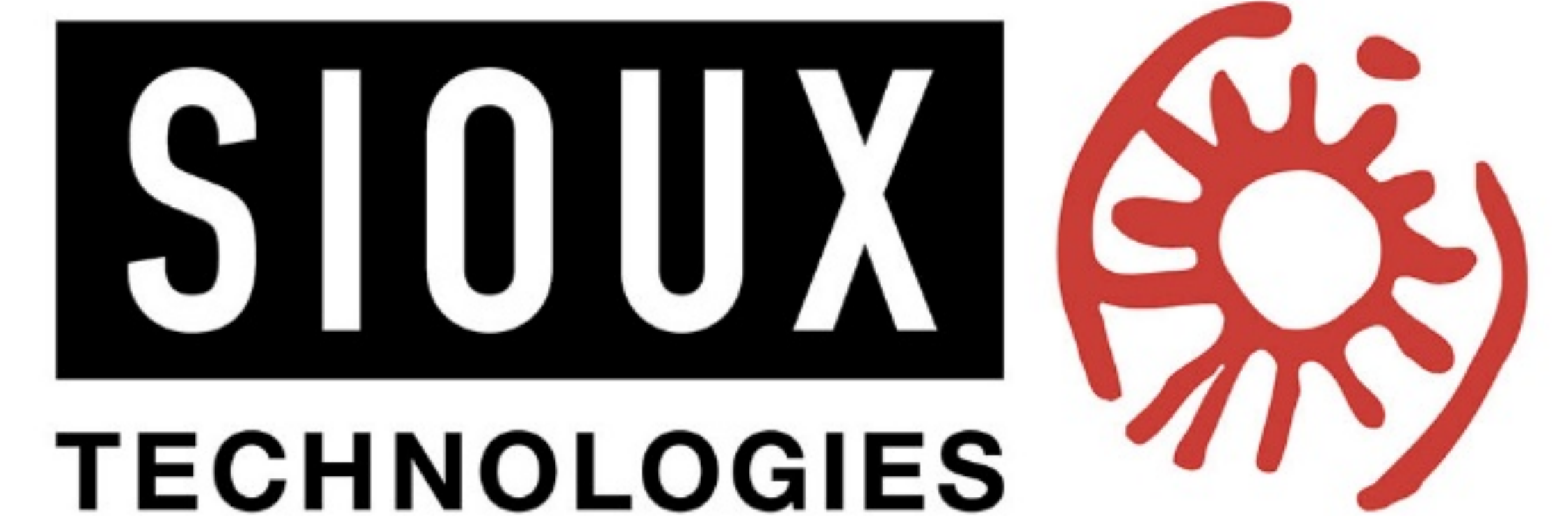


AND HAVE FUN

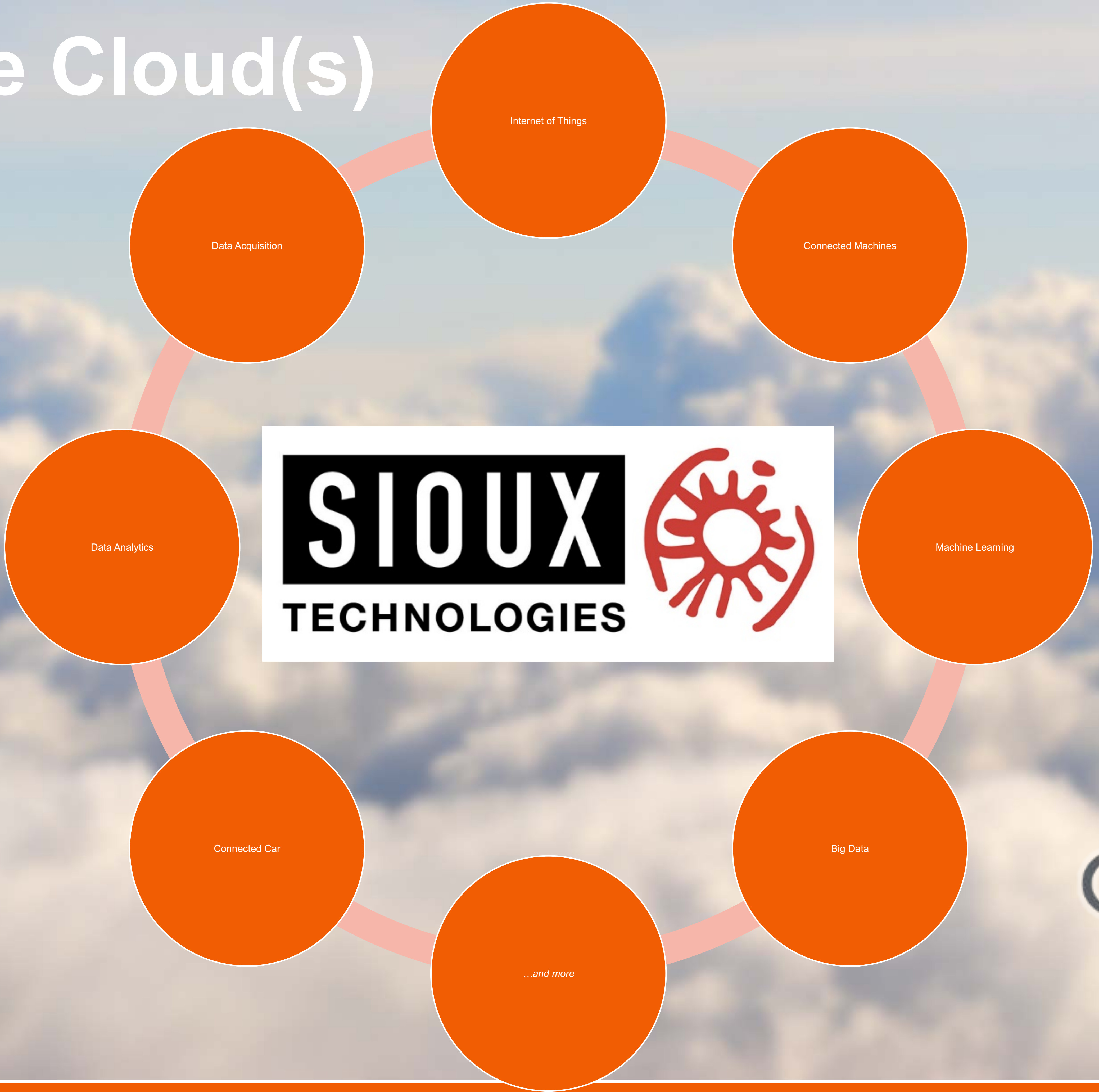
Get to know each other, come together and have fun! Gaming, sports, drinks, go out, go to Sioux Labs!

Introduction

- Michael van Kampen
- Cloud specialist at Sioux
 - Cloud enthusiast
 - Currently working at VECOS
 - Cloud competence lead at Sioux



Sioux in the Cloud(s)





Hot-or-Not: Trends in Cloud technology

Speaker

- Steven Bryen
- Senior Technical Evangelist
at Amazon Web Services





Trends in Cloud

Steve Bryen | AWS Technical & Developer Evangelism

 @steven_bryen

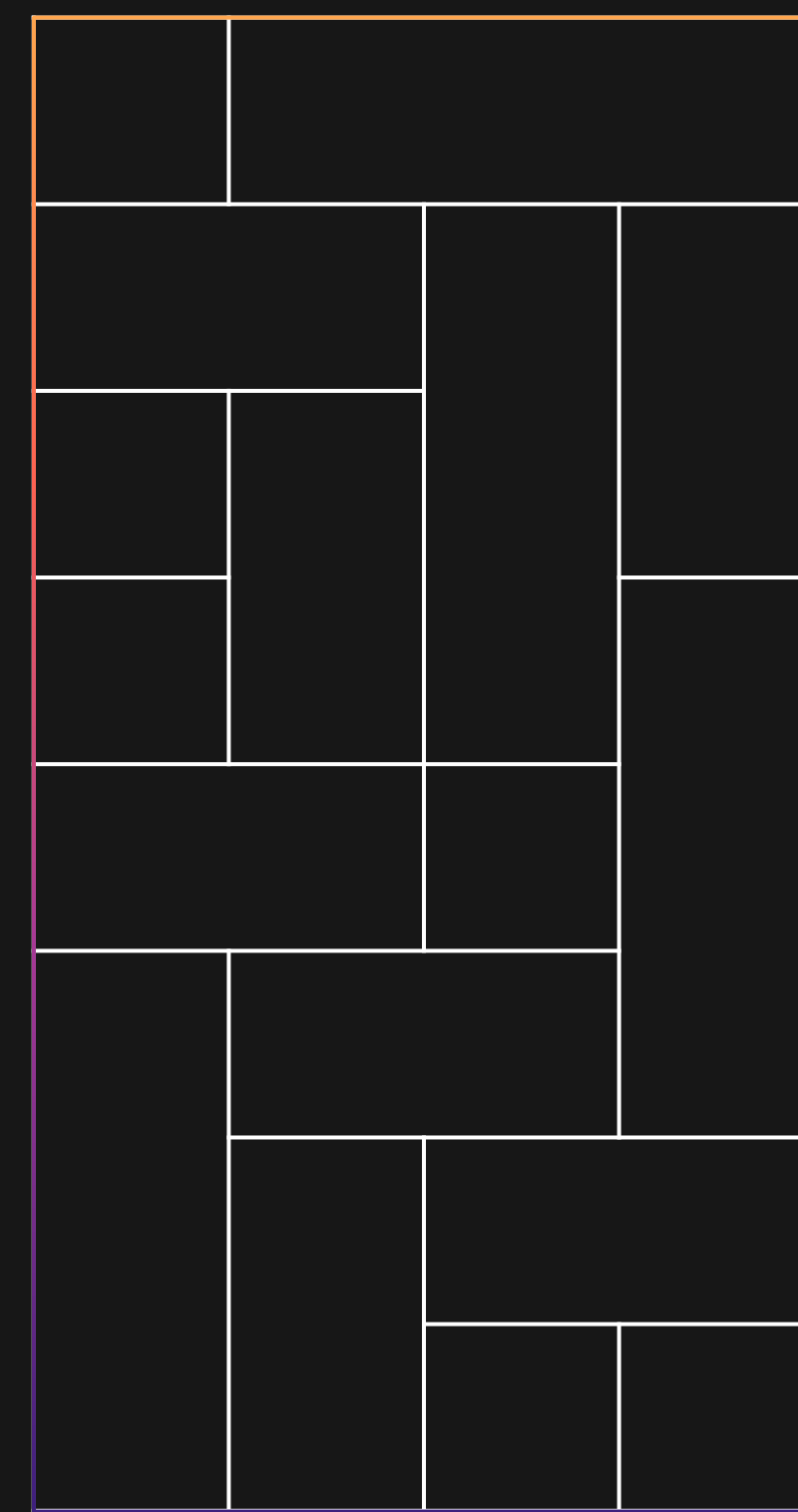


Modern applications

Development transformation at Amazon: 2001–2002

LESSON LEARNED: **DECOMPOSE FOR AGILITY**

2001



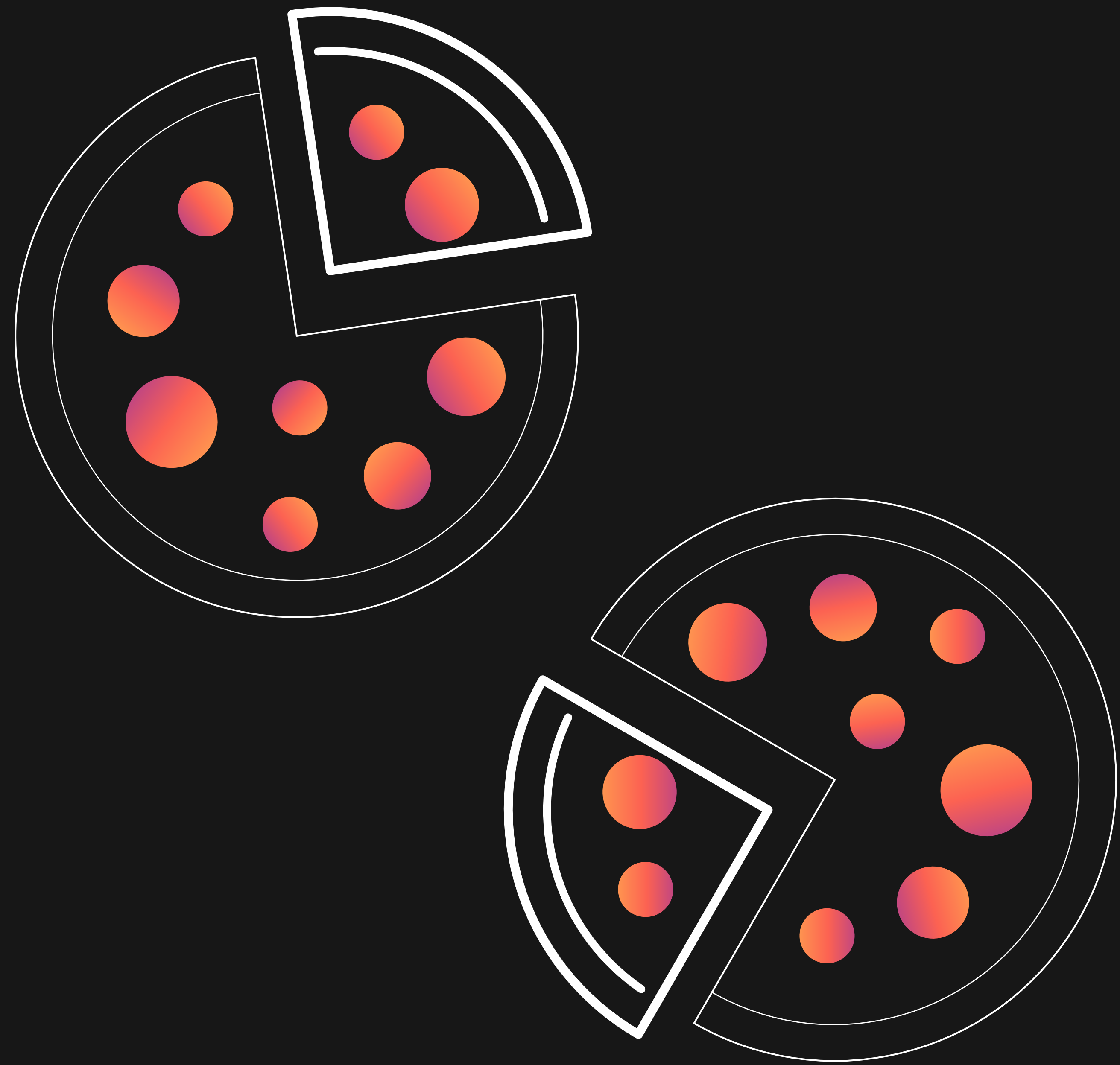
monolithic application +
teams

2002



microservices
+ 2 pizza teams

Two-pizza teams



Full ownership

Full accountability

“DevOps”

Focused innovation

What changes
have to be made
in this new world?

Architectural patterns

Operational model

Software delivery



Changes to the architectural patterns

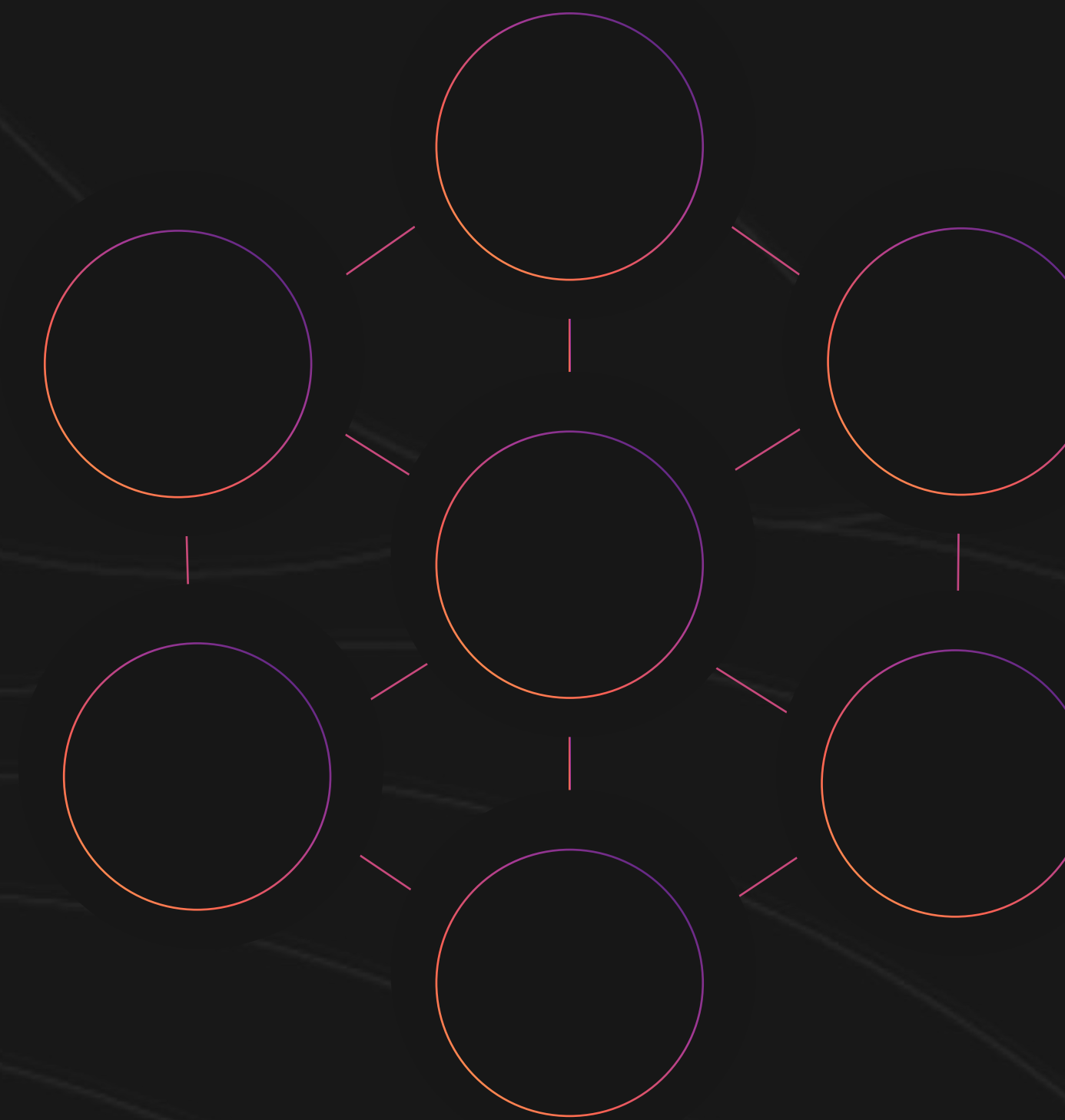
When the impact of change is small, release velocity can increase

MONOLITH



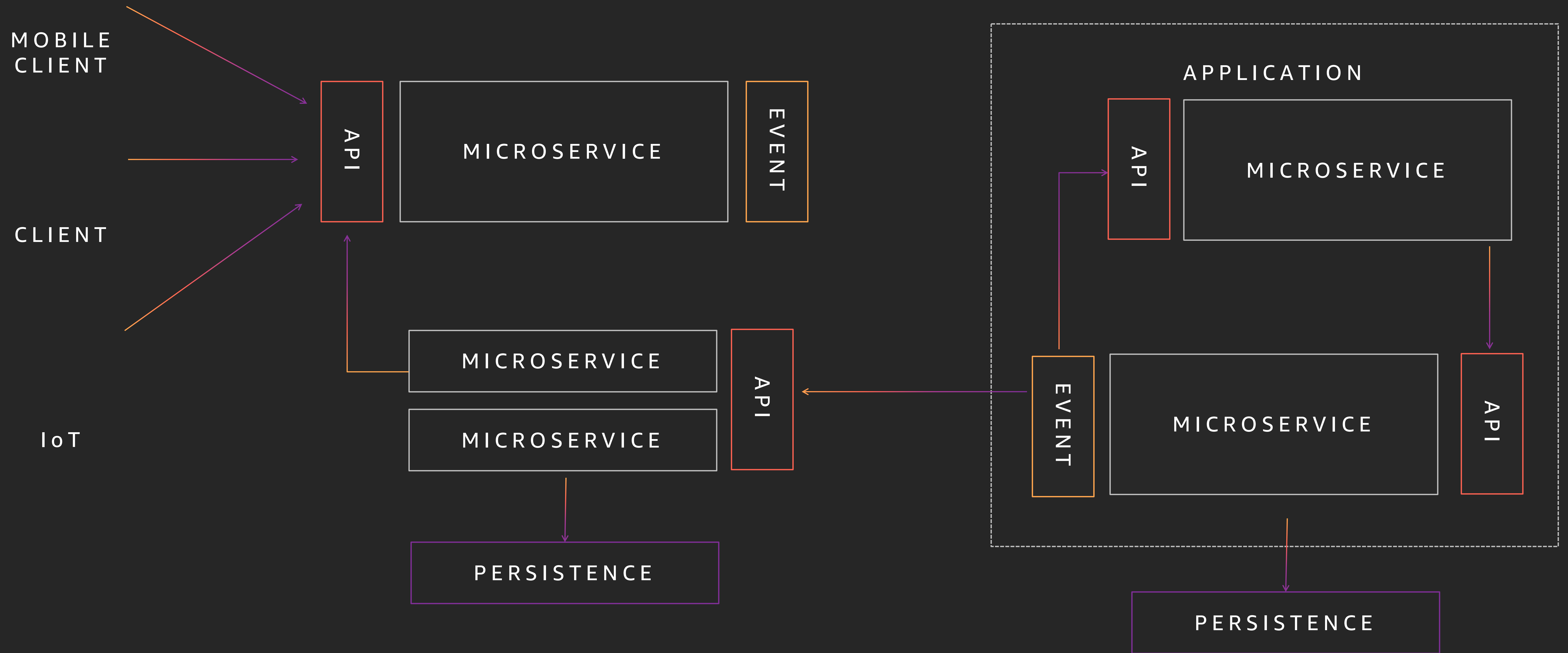
Does everything

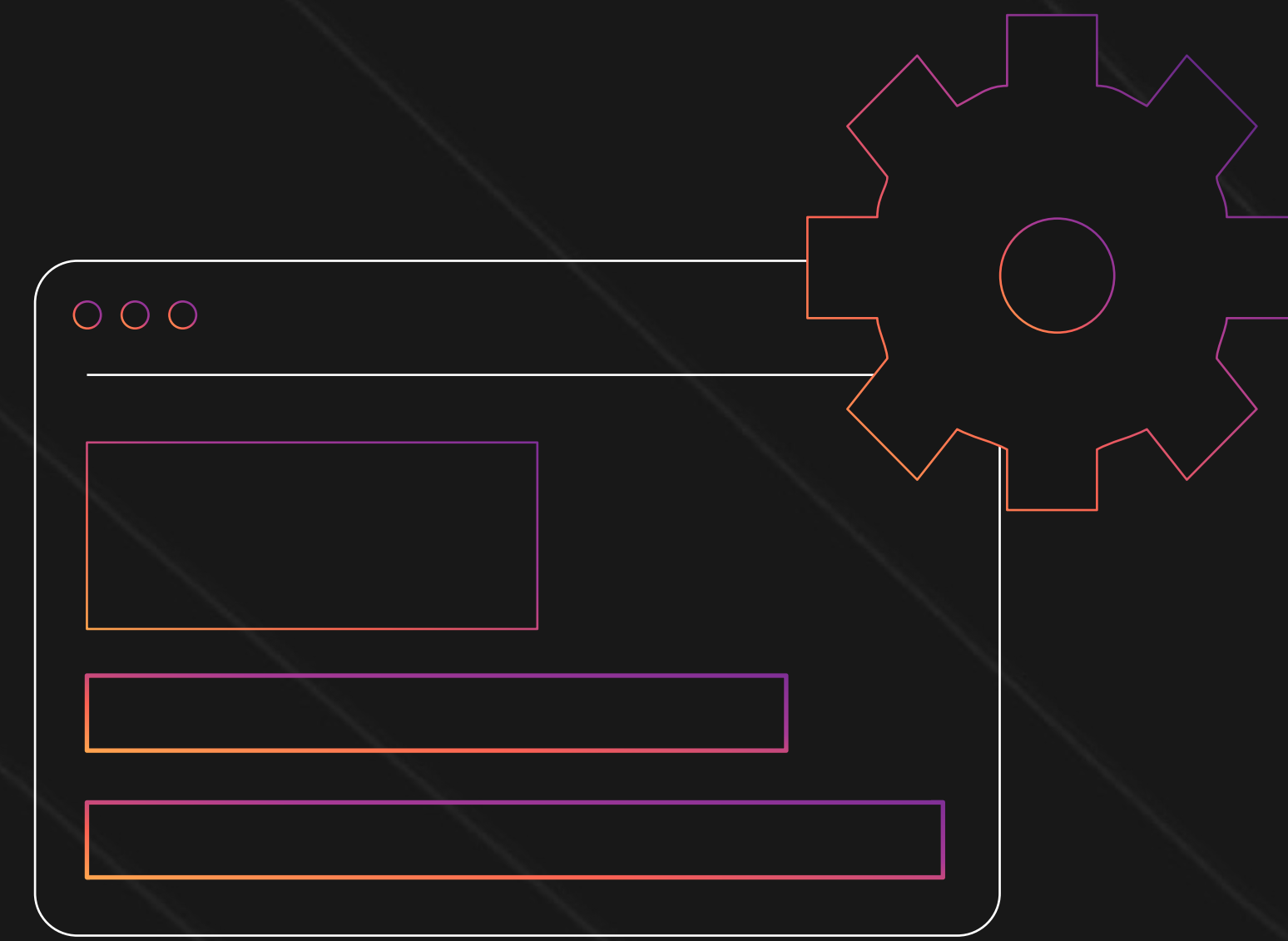
MICROSERVICES



Do one thing

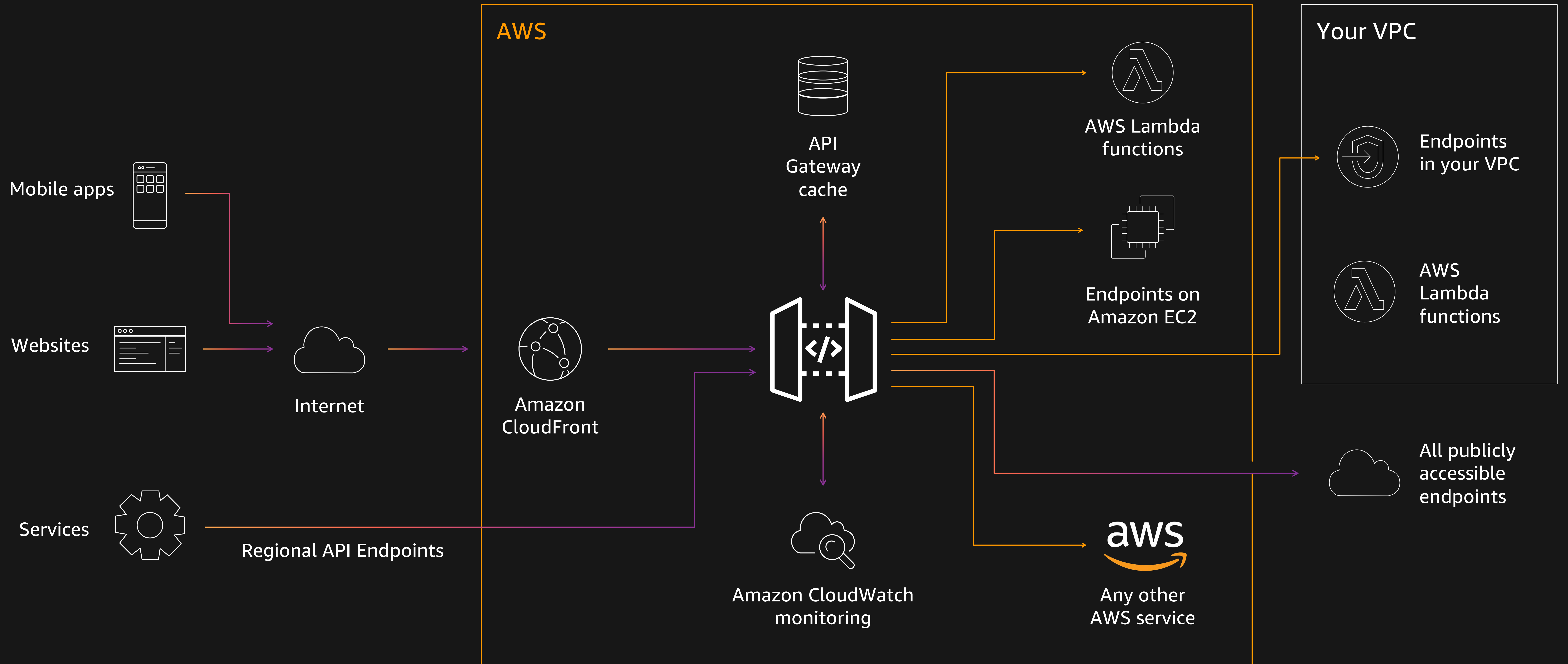
Microservices architectures

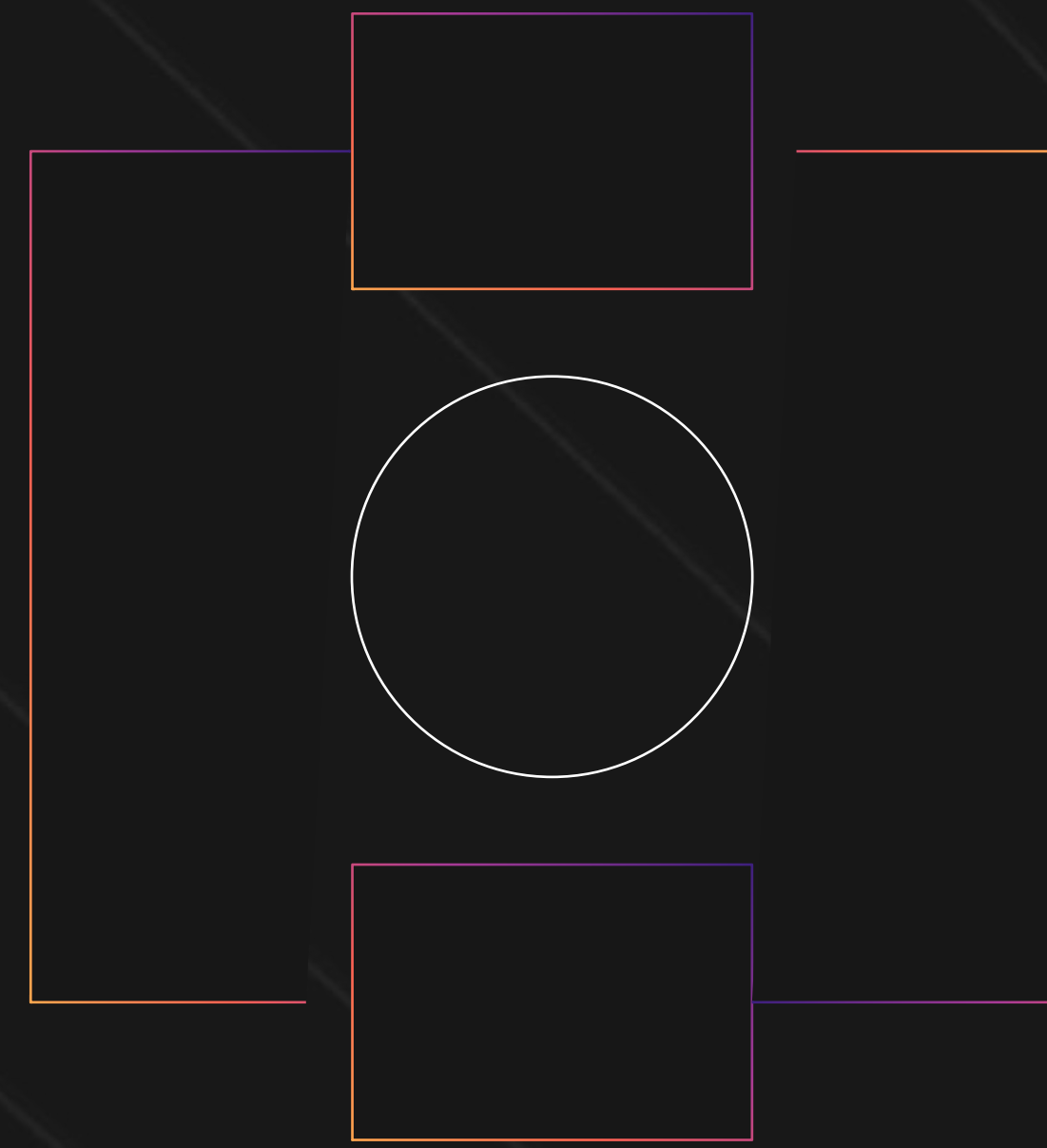




APIs are the front door of microservices

Manage APIs with API Gateway





Event-driven architectures

Decouple state from code using messaging

MESSAGING



Amazon Simple Queue Service

QUEUES

Simple
Fully-managed
Any volume



Amazon Simple Notification Service

PUB/SUB

Simple
Fully-managed
Flexible



Amazon CloudWatch Events

SYNCHRONIZATION

Rapid
Fully-managed
Real-time

And data streams

DATA STREAM CAPTURE



Amazon Kinesis Data Streams

INGEST

Data streams
Data processing
Real-time



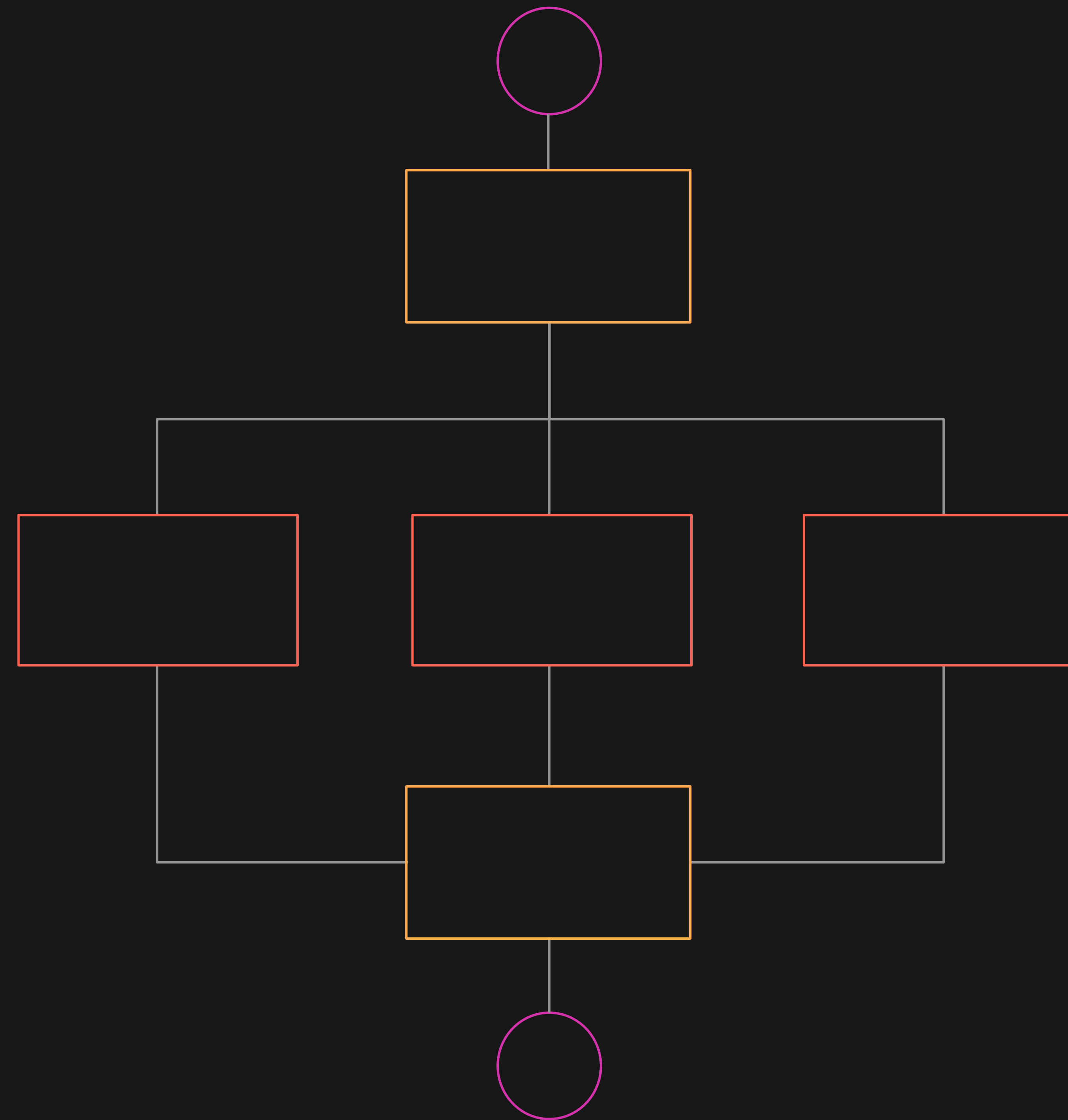
Amazon Dynamo DB

DATA STORE

Microservices
Performance at scale
Fast and Flexible

Build workflows to orchestrate everything

TRACK STATUS
OF DATA
AND EXECUTION

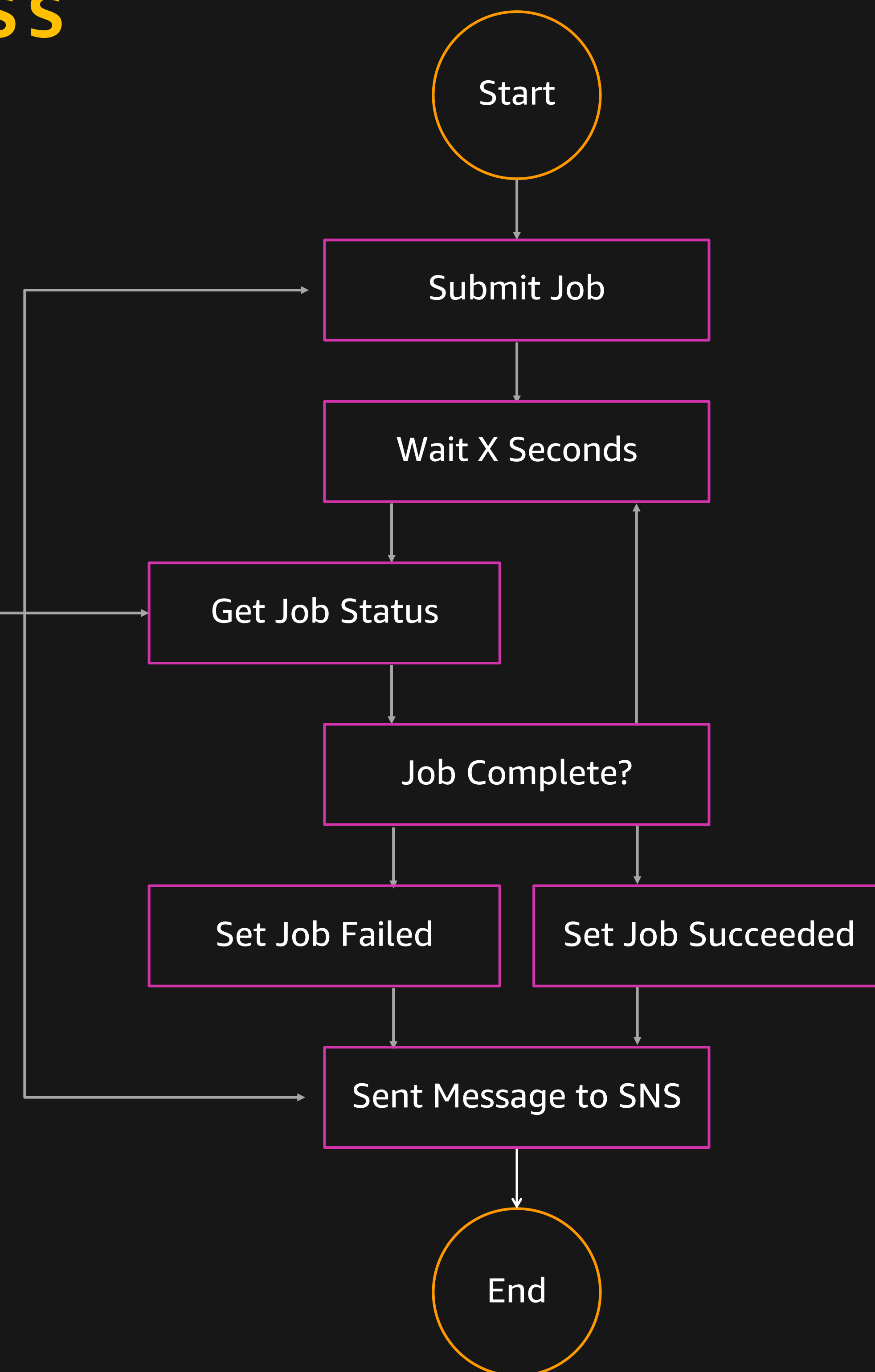


REMOVE
REDUNDANT
CODE

Simpler integration, less code

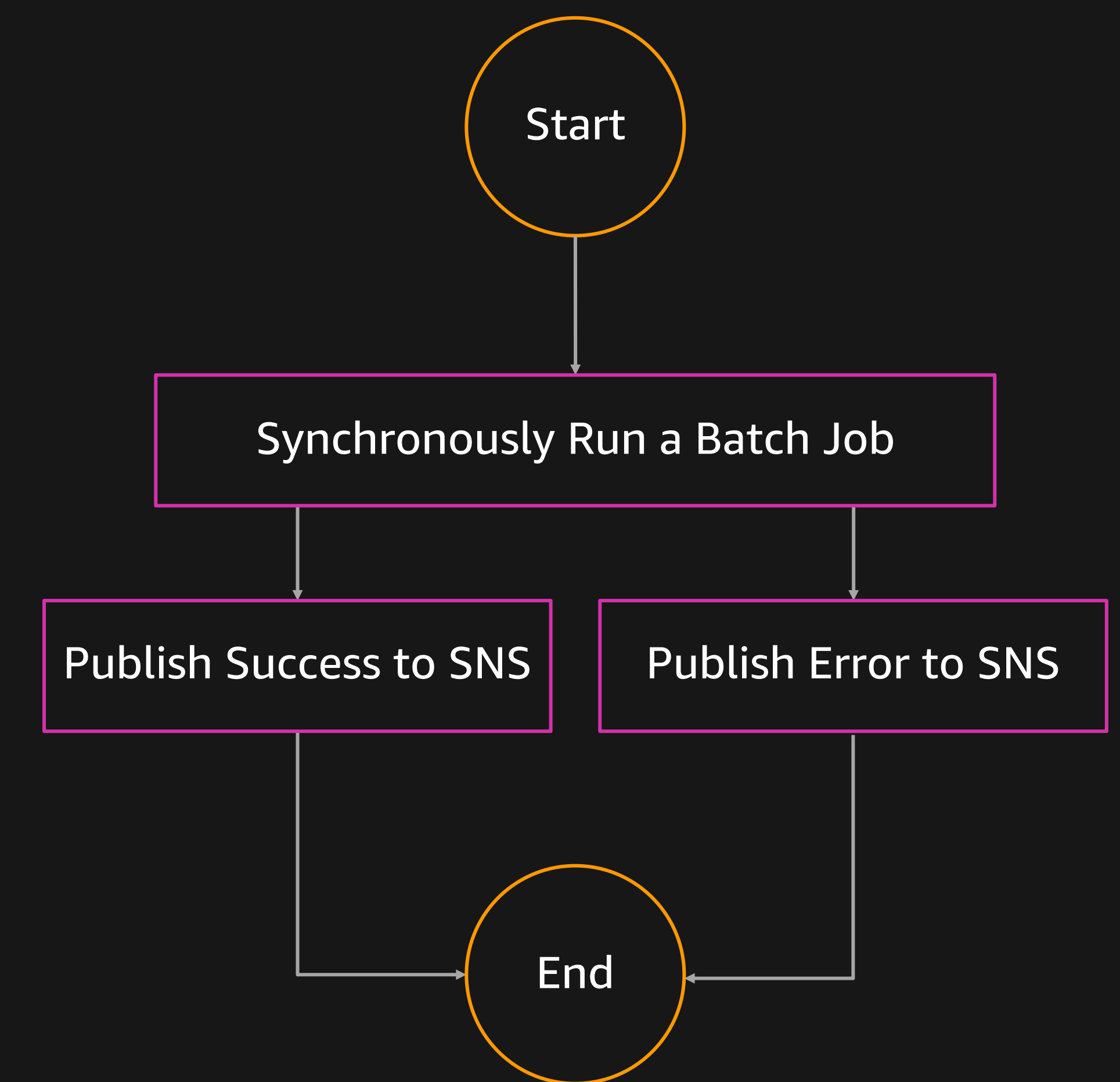
WITH SERVERLESS POLLING

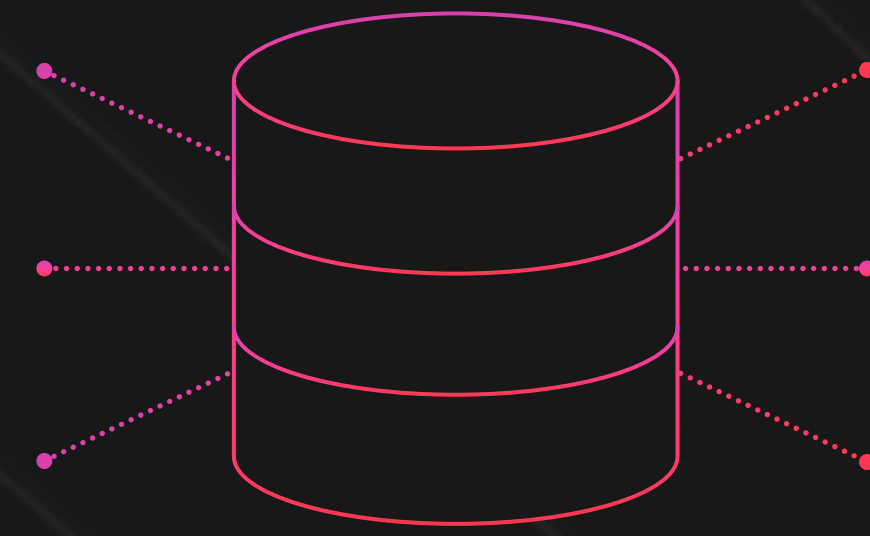
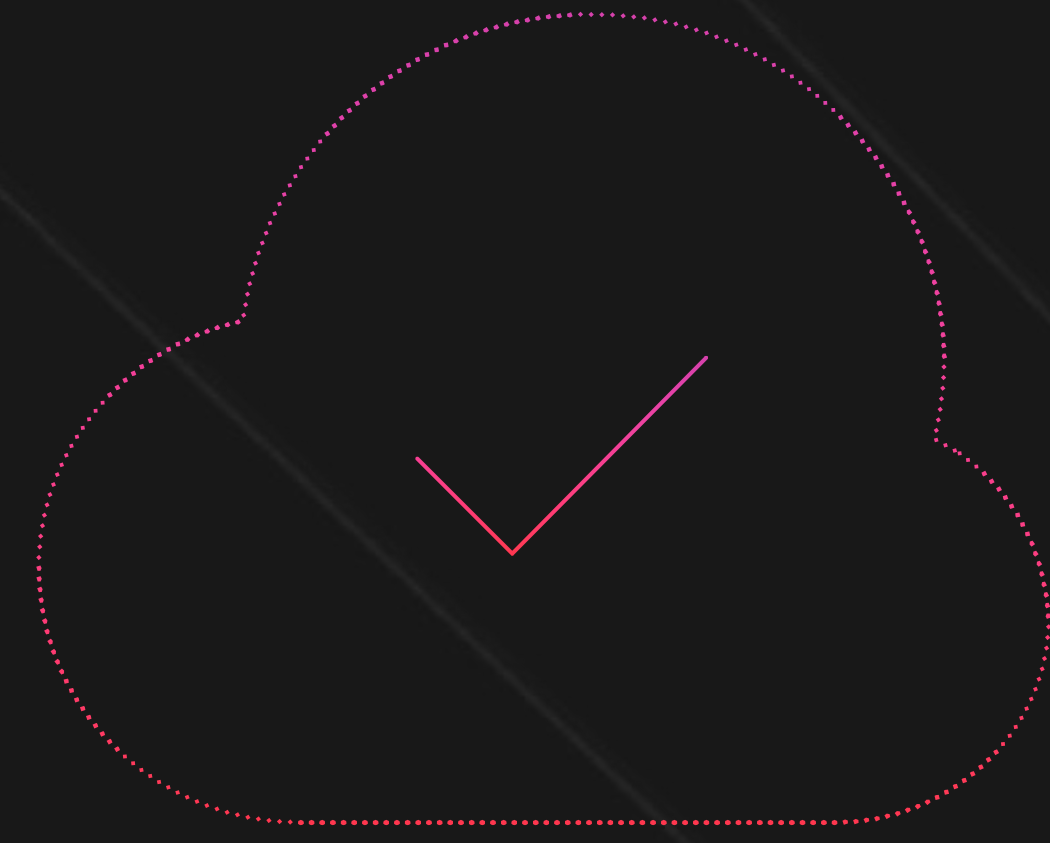
AWS LAMBDA FUNCTIONS



WITH NEW SERVICE INTEGRATION

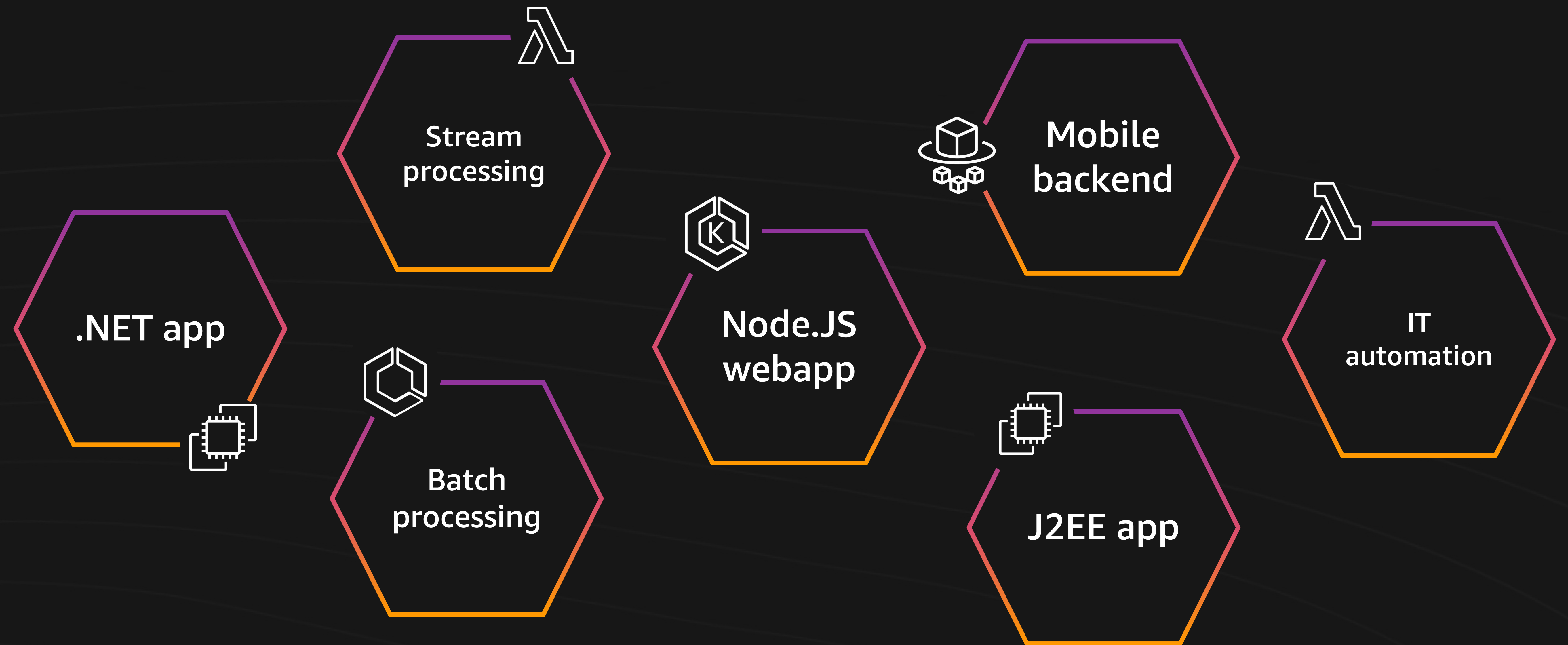
NO LAMBDA FUNCTIONS



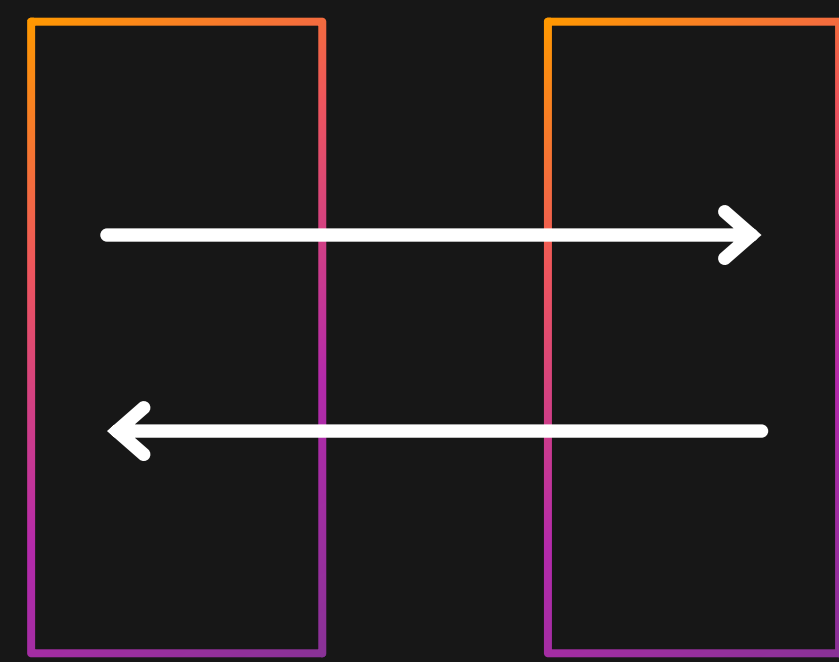


Cloud-native architectures are small pieces, loosely joined

How do all our applications communicate with each other?



What is needed



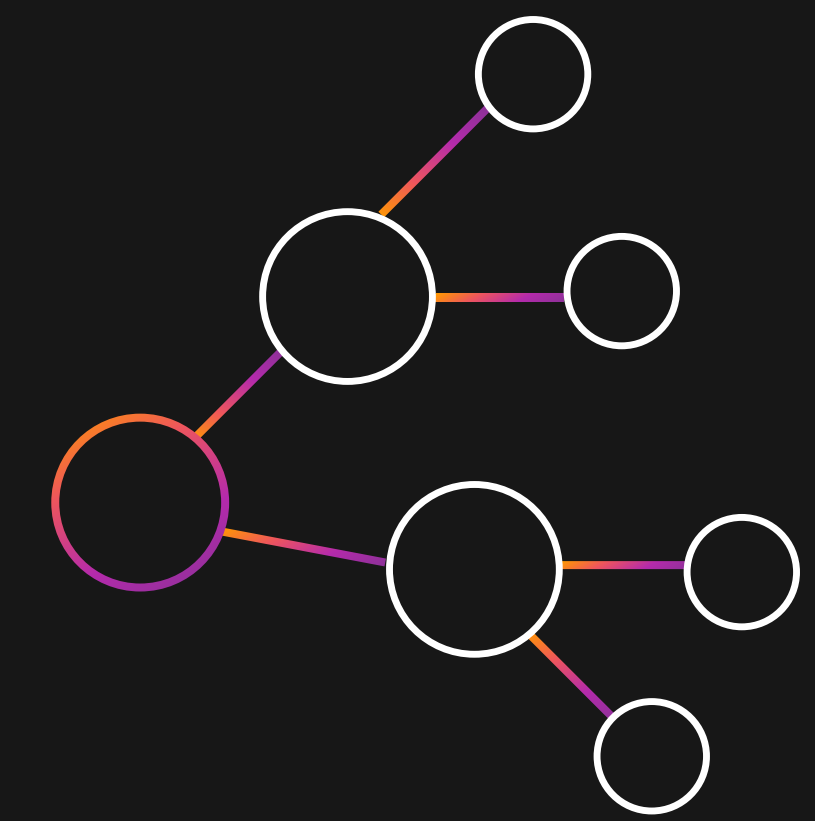
Consistent
communications
management



Complete visibility



Failure isolation
and protection

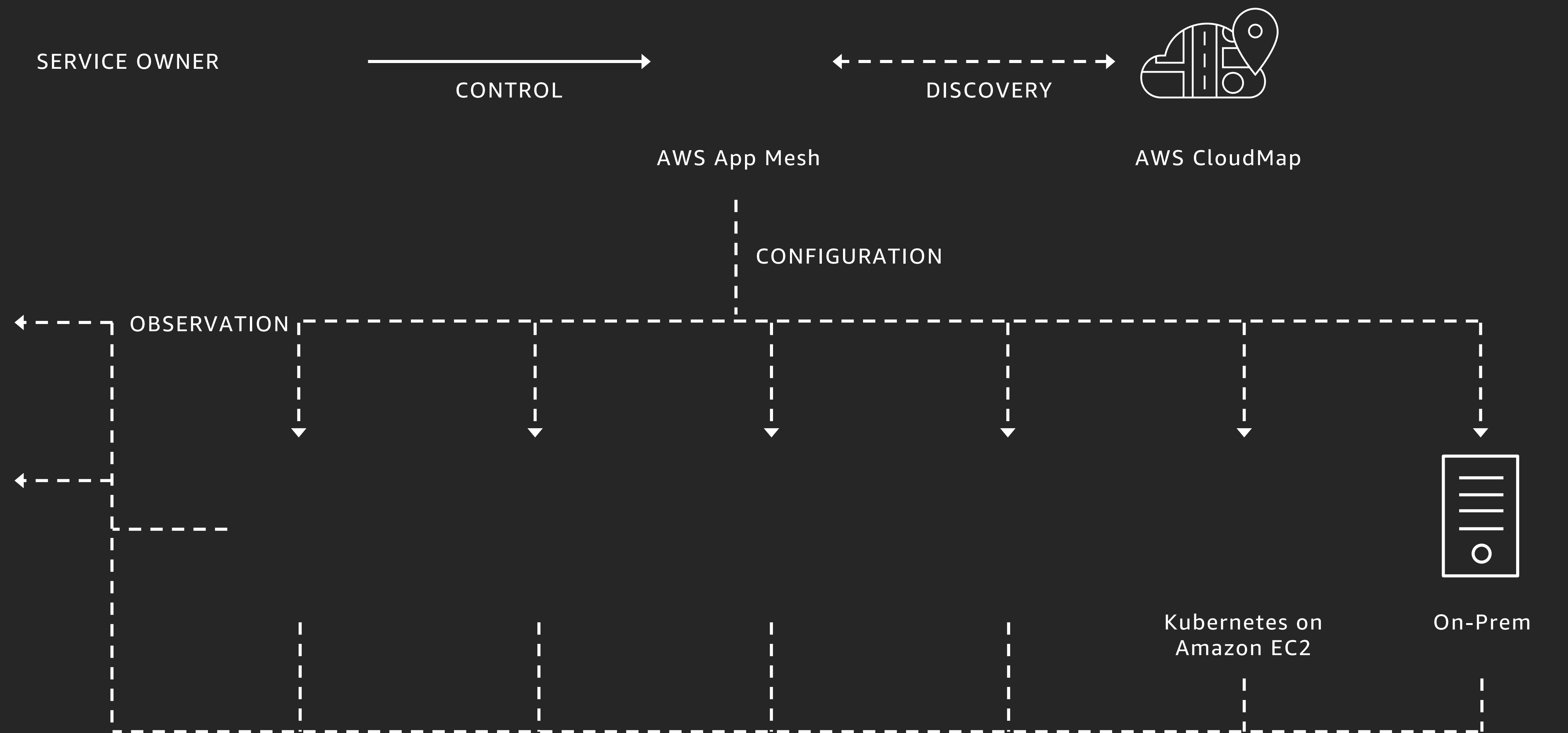


Fine-grained
deployment
controls

GENERALLY
AVAILABLE

AWS App Mesh

Application-level networking
for all your resources



NEW!



Amazon EventBridge

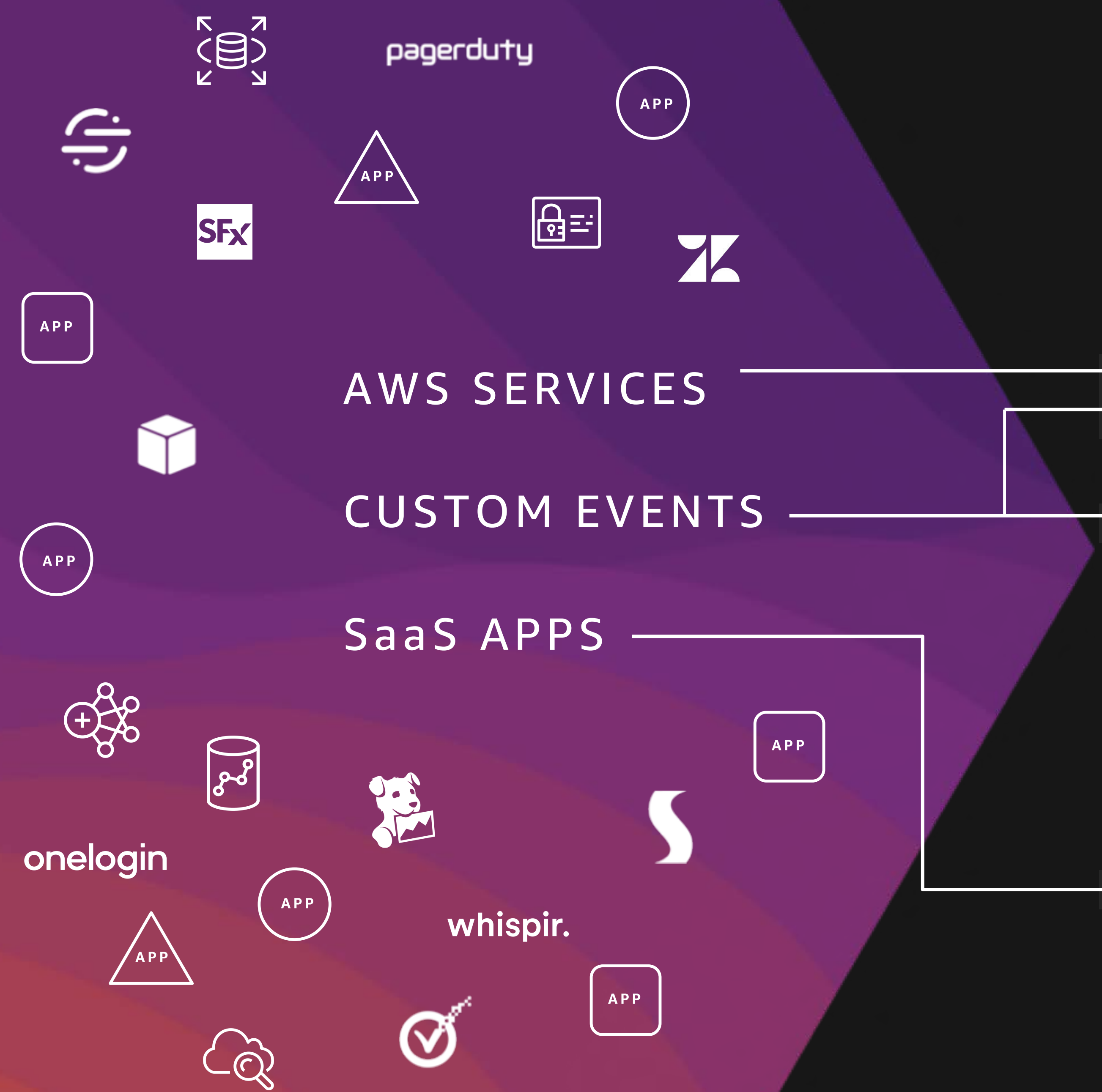
Serverless event bus for ingesting and processing data across AWS services and SaaS applications

Removes friction of writing “point-to-point” integrations

Works across dozens of AWS and SaaS applications

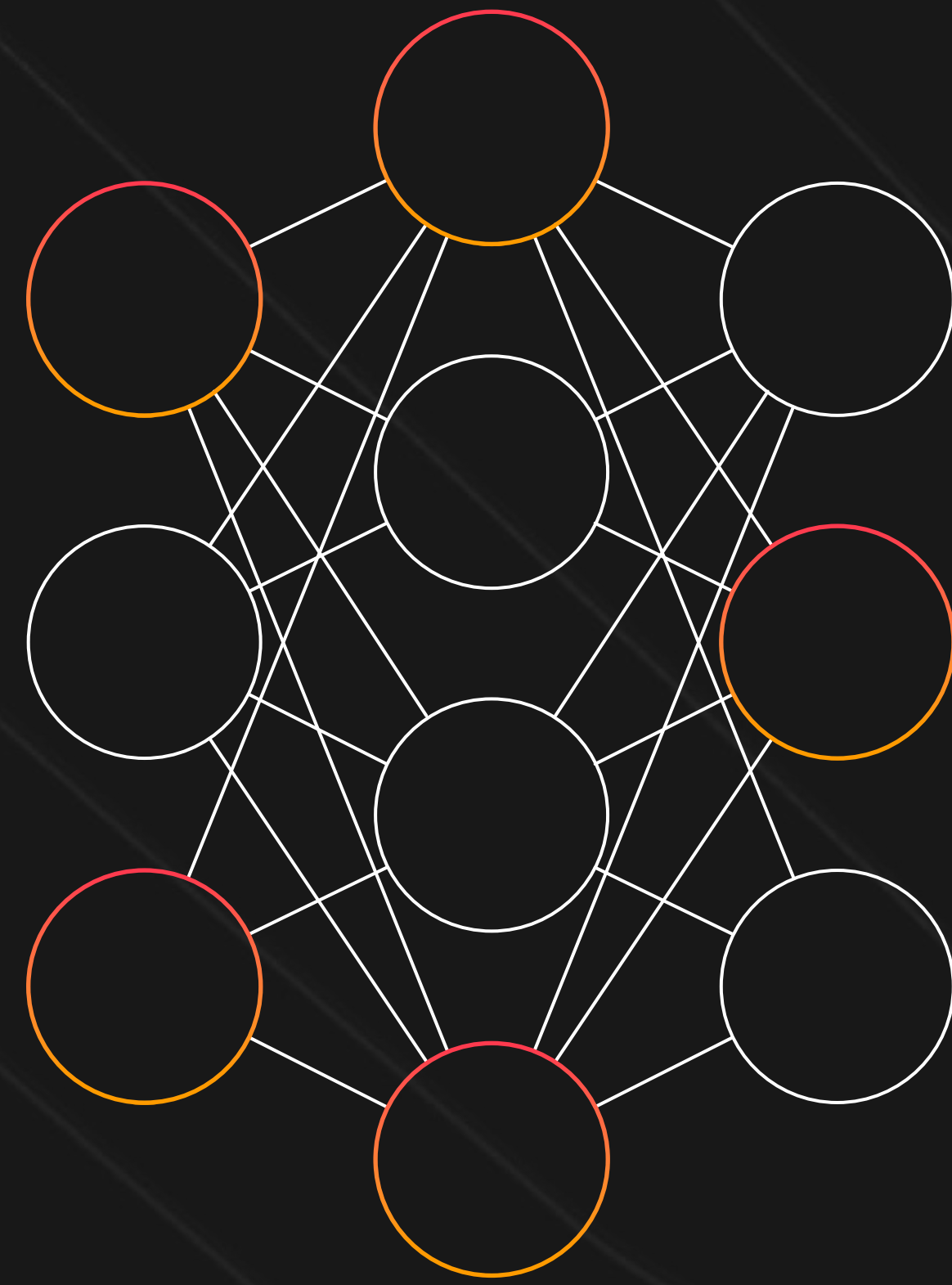
Provides simple programming model

Fully managed; pay-as-you-go



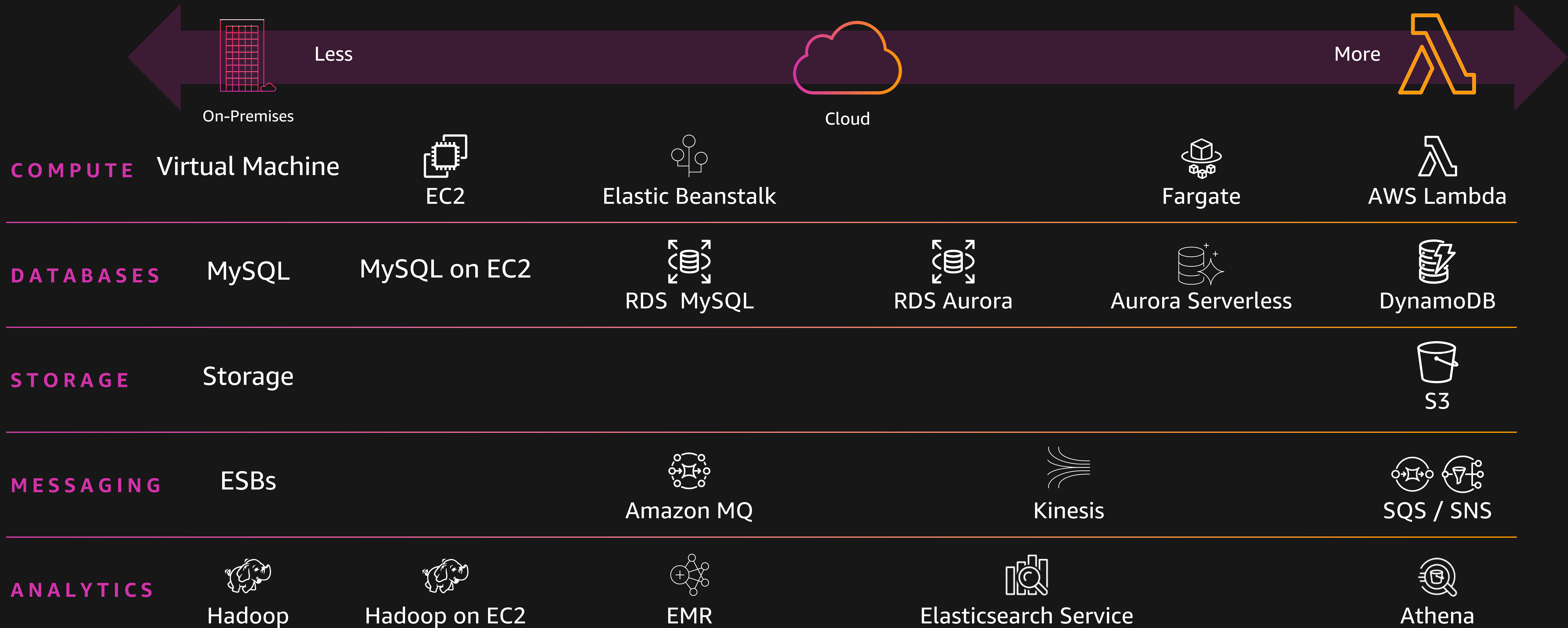
- TARGETS**
Route events to a variety of targets for processing
- AWS Lambda**
 - Amazon Kinesis Data Firehose**
 - Amazon SNS**
 - Additional targets

Changes to the operational model



Isn't all of this very hard now that we have lots of pieces to operate?

AWS operational responsibility models





What is serverless?

No infrastructure provisioning, no management

Automatic scaling

Pay for value

Highly available and secure

Serverless is an operational model that spans many different categories of services

COMPUTE



AWS
Lambda



AWS
Fargate

DATA STORES



Amazon
S3



Amazon Aurora
Serverless



Amazon
DynamoDB

INTEGRATION



Amazon
API Gateway



Amazon
SQS



Amazon
SNS



AWS
Step Functions



AWS
AppSync

Let's focus on compute for now



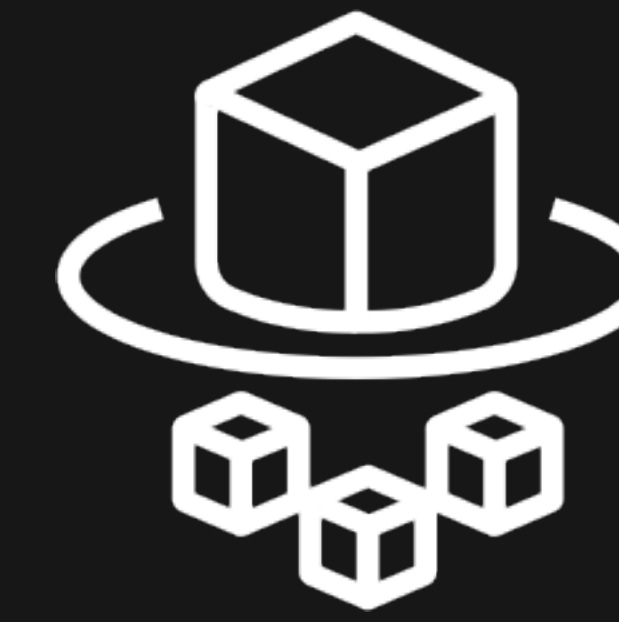
AWS LAMBDA

**Serverless event-driven
code execution**

Short-lived

All language runtimes

Data source integrations



AWS FARGATE

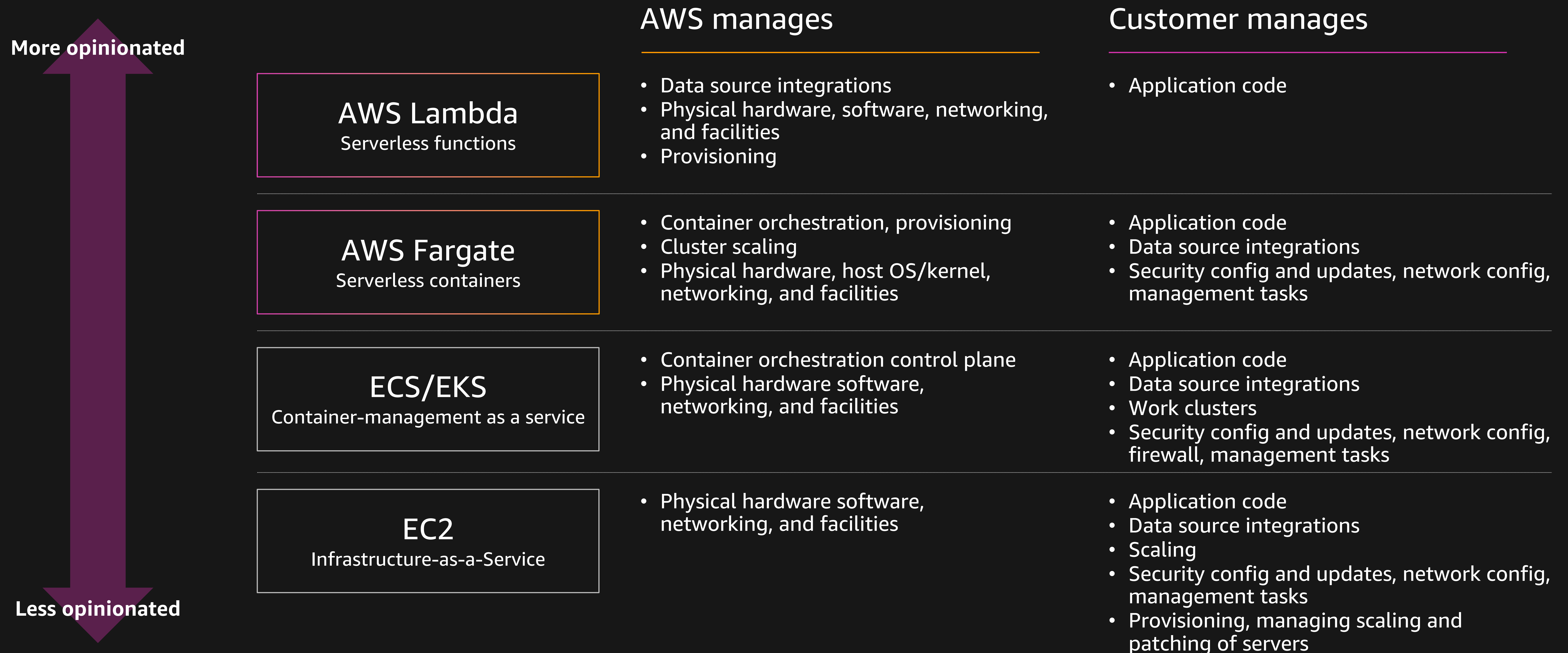
**Serverless compute engine
for containers**

Long-running

Bring existing code

Fully-managed orchestration

Comparison of operational responsibility



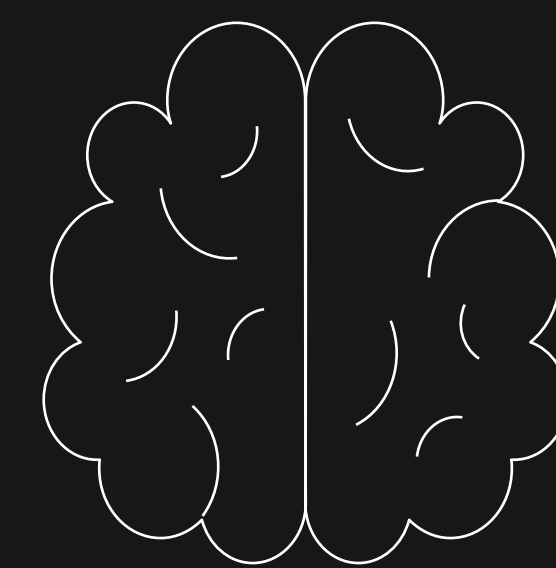
Making development easier with AWS Lambda



Accessible for all developers

Support for all runtimes with Lambda Layers and Runtime API

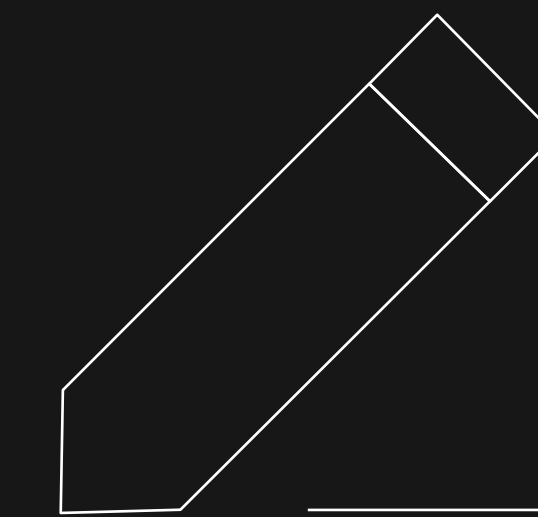
ISO, PCI, HIPAA, SOC, GDPR, and FedRamp compliances



Greater productivity

Toolkits for popular IDEs: VSCode, IntelliJ, and PyCharm

Simplified deployment with nested apps



Enable new application patterns

15 minute functions

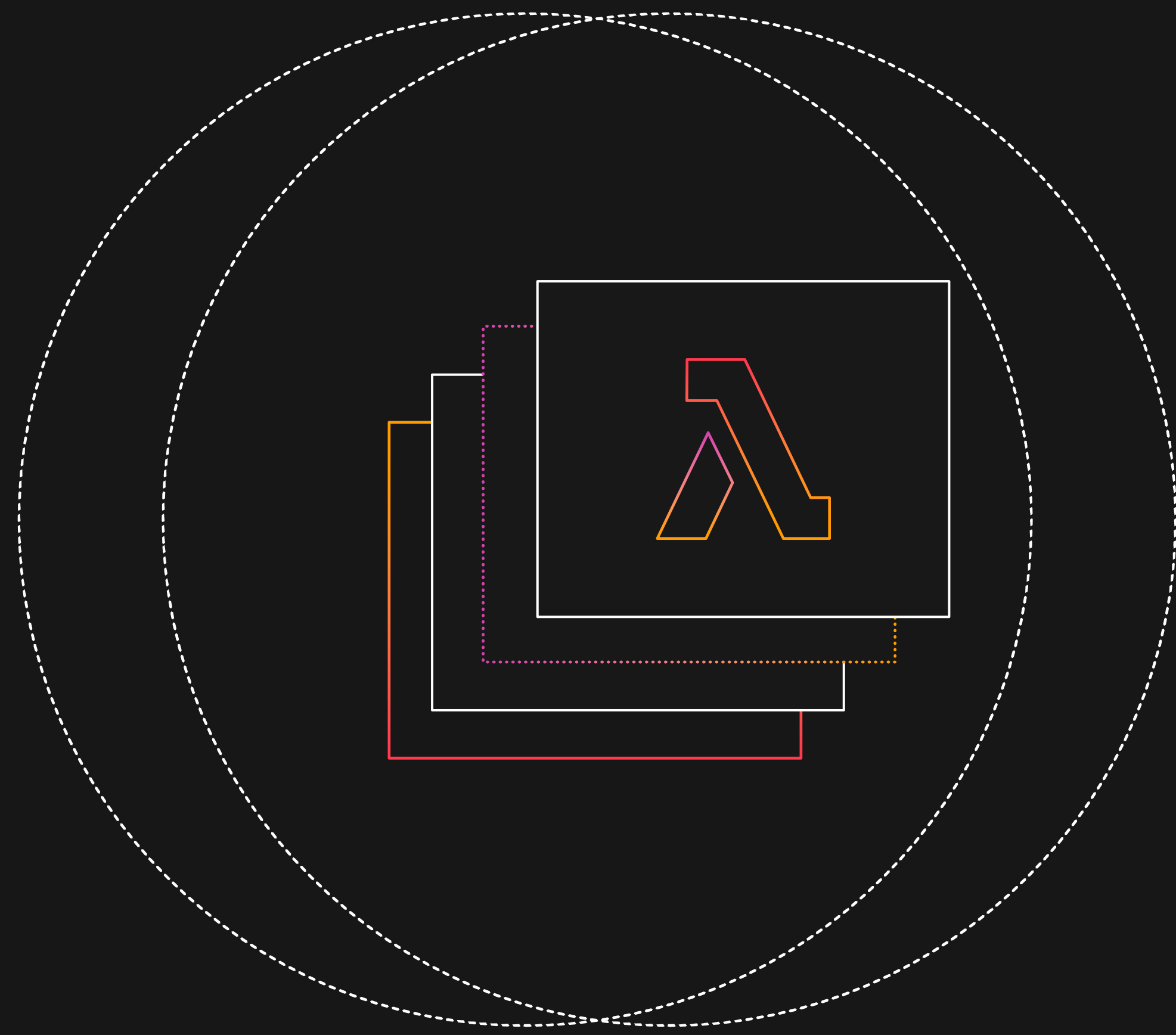
SQS for Lambda

Automatic Load Balancing for Lambda

Support for Kinesis Data Streams Enhanced Fan-Out and HTTP/2

Trillions of executions every month for hundreds of thousands of active customers

Lambda Layers

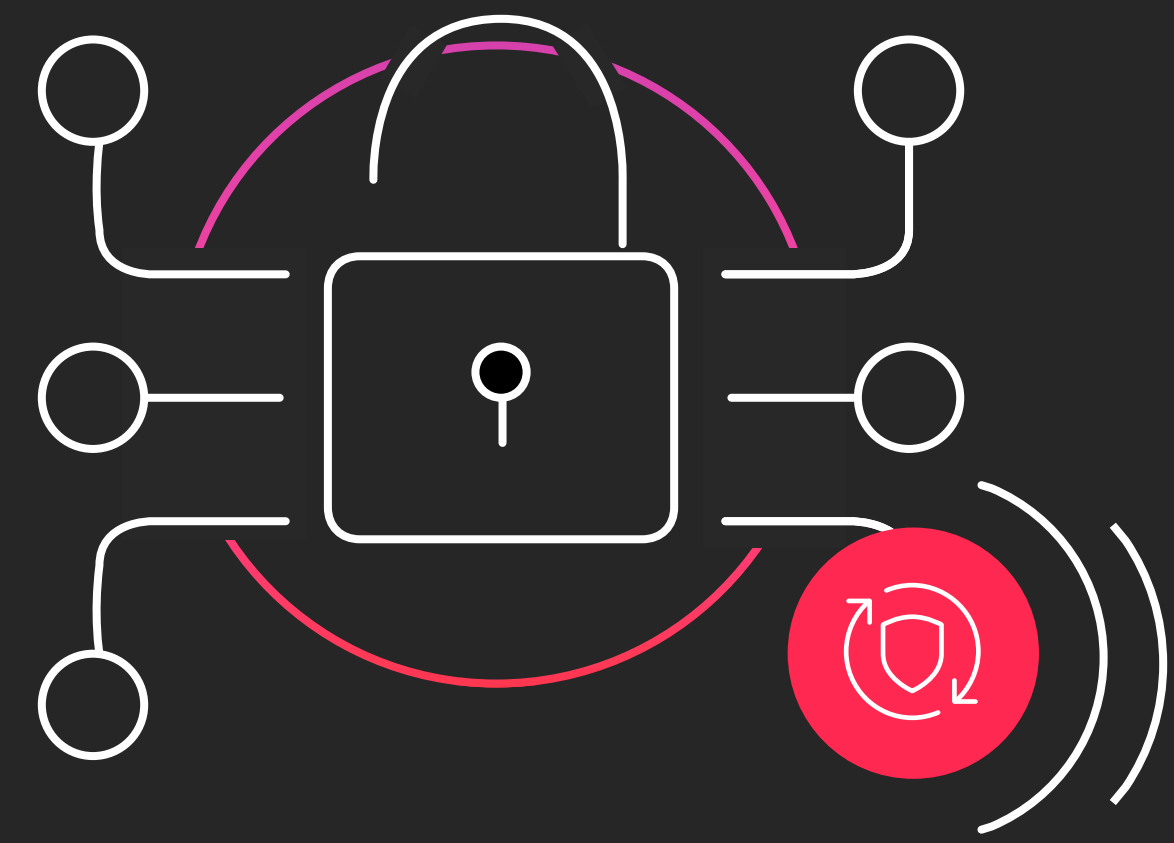


Lets functions easily share code: Upload layer once, reference within any function

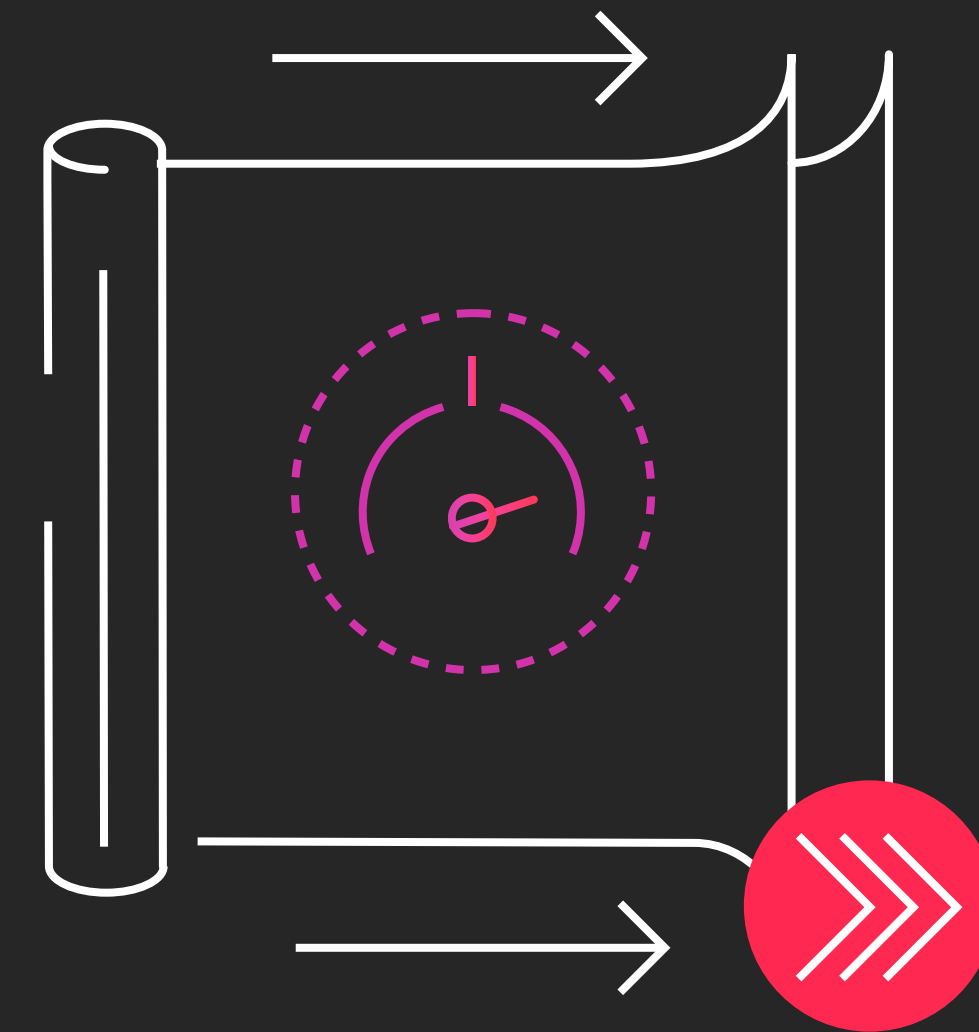
Promote separation of responsibilities, lets developers iterate faster on writing business logic

Built in support for secure sharing by ecosystem

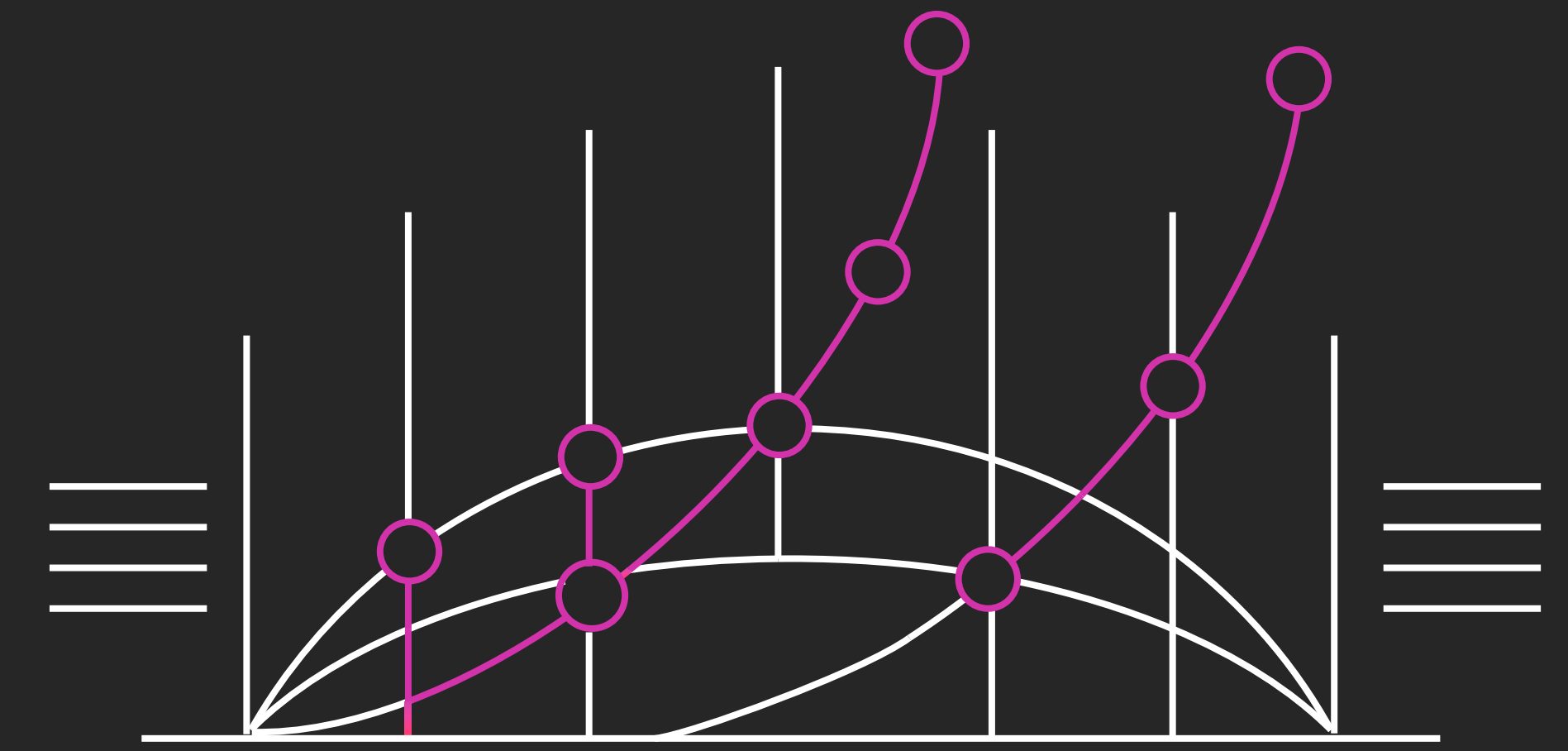
Accelerating Fargate and Lambda with Firecracker



Security



Speed by design



Scale and efficiency

Firecracker is open sourced to enable broad access and innovation

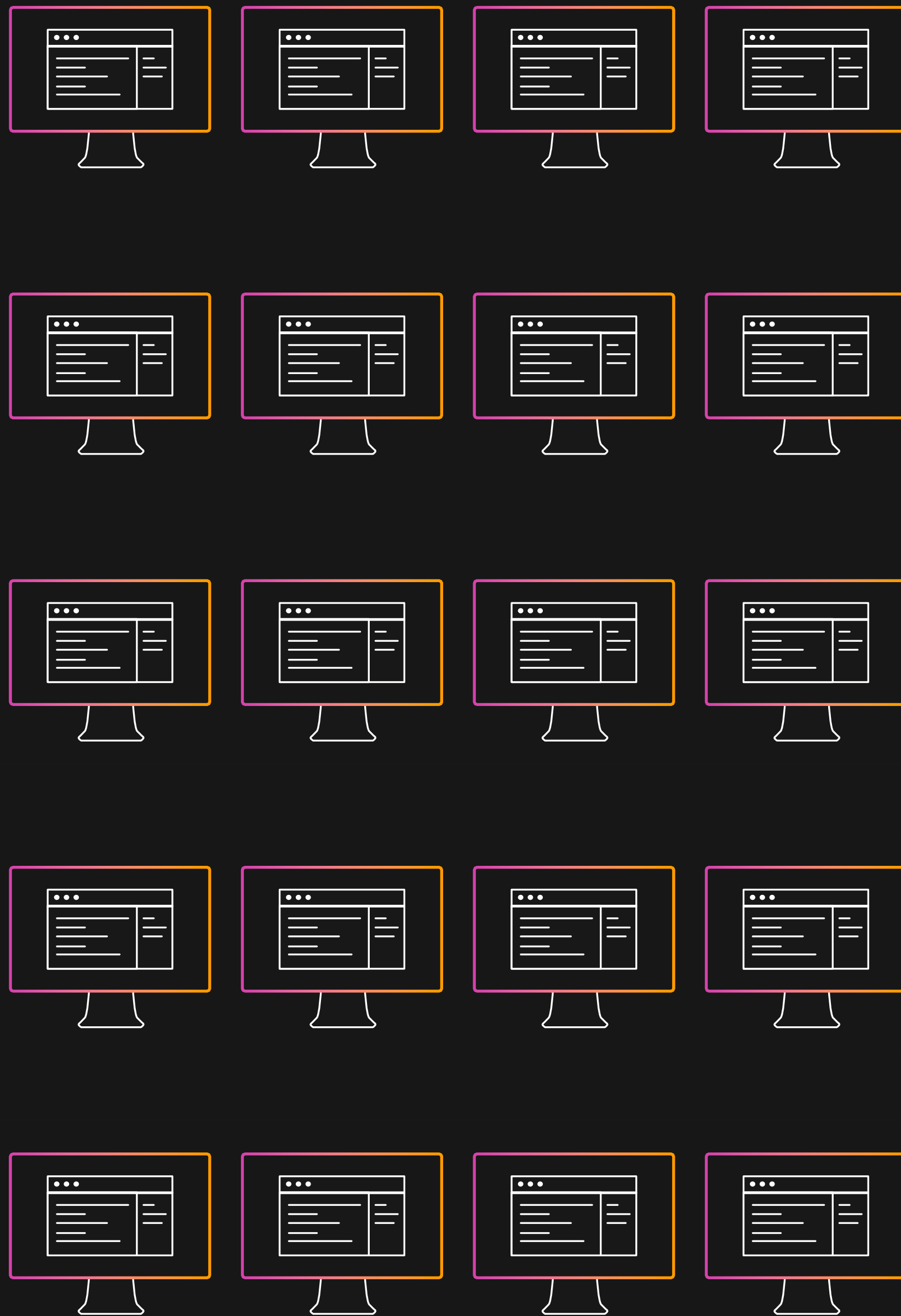
Changes to the delivery of software



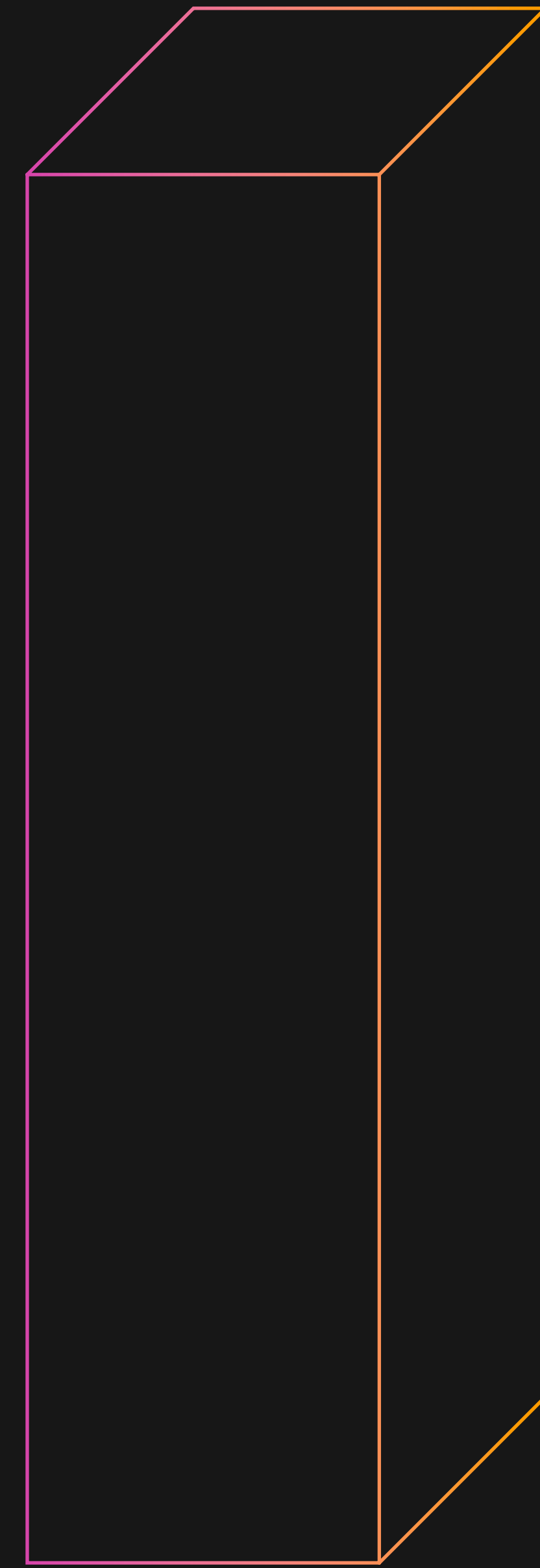
How do I develop and deploy code in a serverless
microservices architecture?

Monolith development lifecycle

DEVELOPERS



SERVICES



DELIVERY PIPELINES

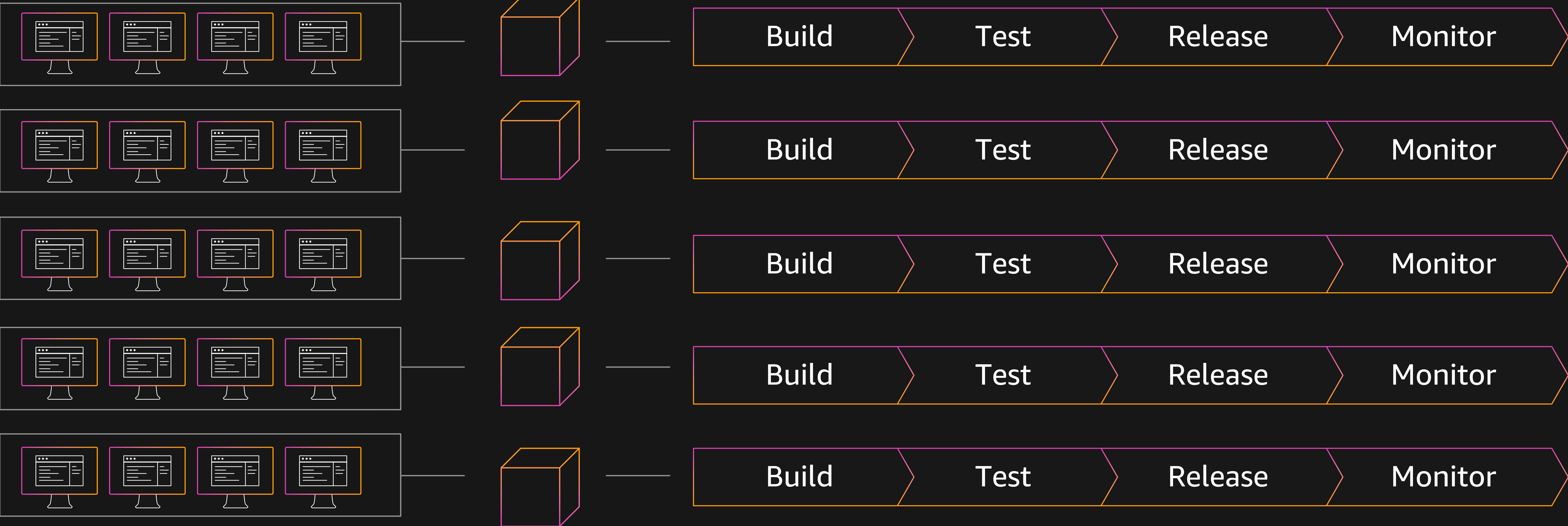


Microservice development lifecycle

DEVELOPERS

SERVICES

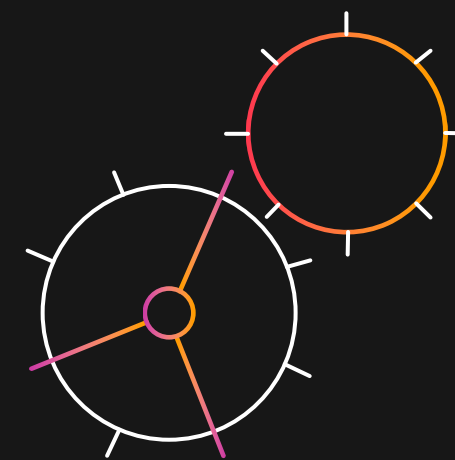
DELIVERY PIPELINES



Best practices



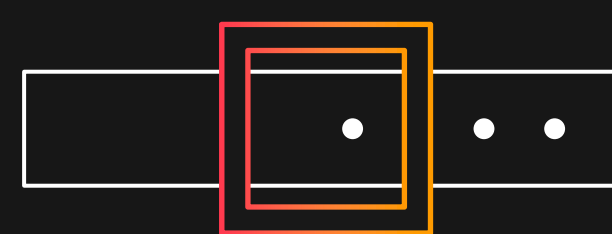
Decompose for agility
(microservices, 2 pizza teams)



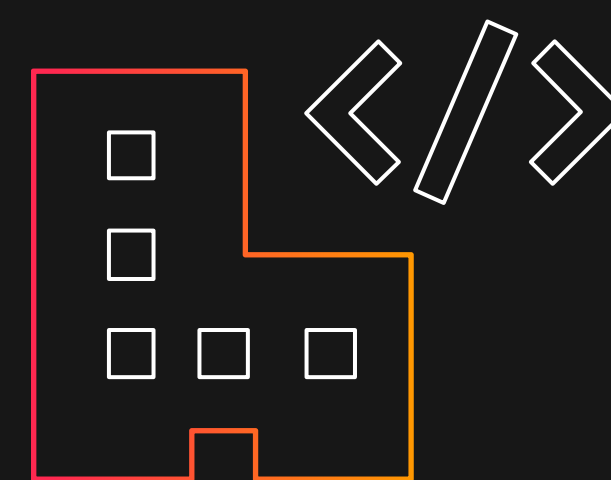
Automate everything



Standardized tools

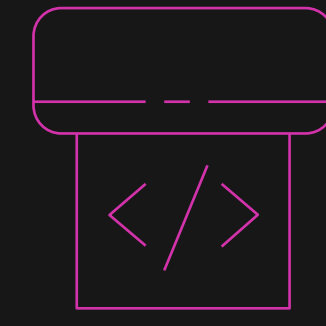


Belts and suspenders
(governance, templates)

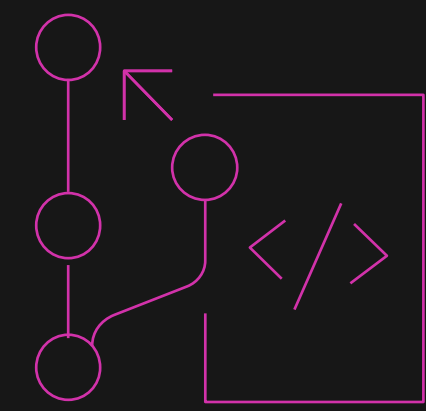
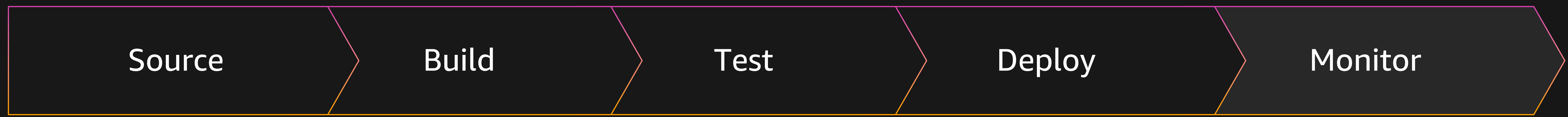


Infrastructure as code

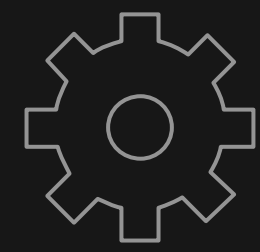
AWS Developer Tools for CI/CD



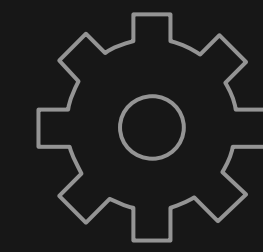
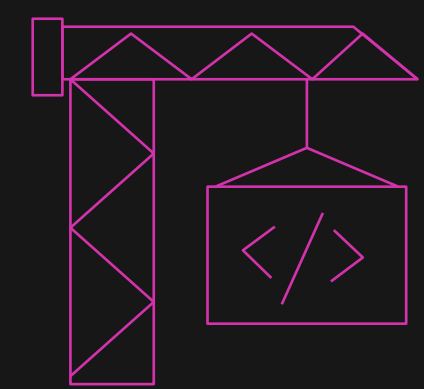
AWS CodePipeline



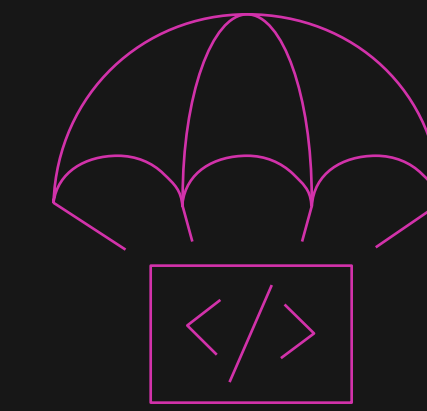
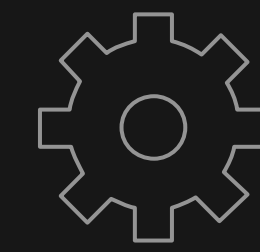
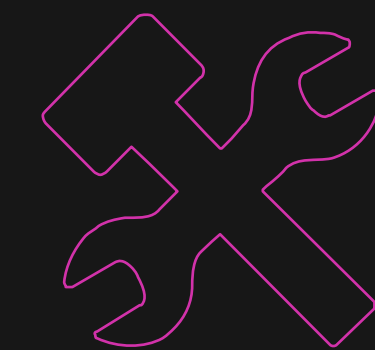
AWS
CodeCommit



AWS
CodeBuild



AWS CodeBuild +
Third Party



AWS
CodeDeploy

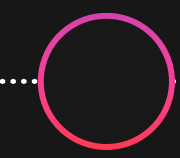


AWS
X-Ray

AWS Developer Tools are focused on supporting containers and Lambda

2016

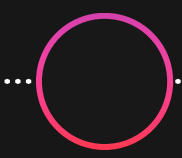
NOV



Support for Lambda deployment with CodePipeline and CloudFormation

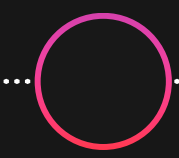
2017

NOV



Support for rolling and blue/green Lambda deployments with CodeDeploy

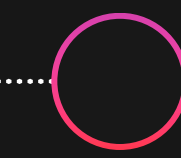
DEC



Support for Fargate and ECS deployments in CodePipeline

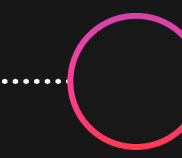
2018

OCT



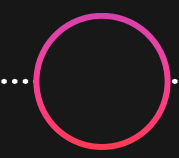
CodePipeline supports Config for improved governance

NOV



CodePipeline supports ECR as a source

NOV



Support for blue/green deployments for Fargate and ECS with CodeDeploy

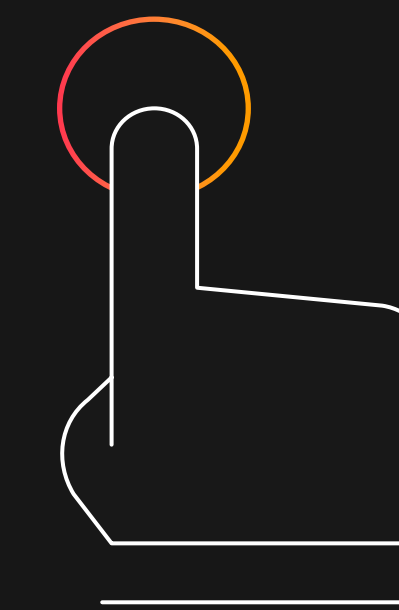
AWS X-Ray is Built for Modern Applications



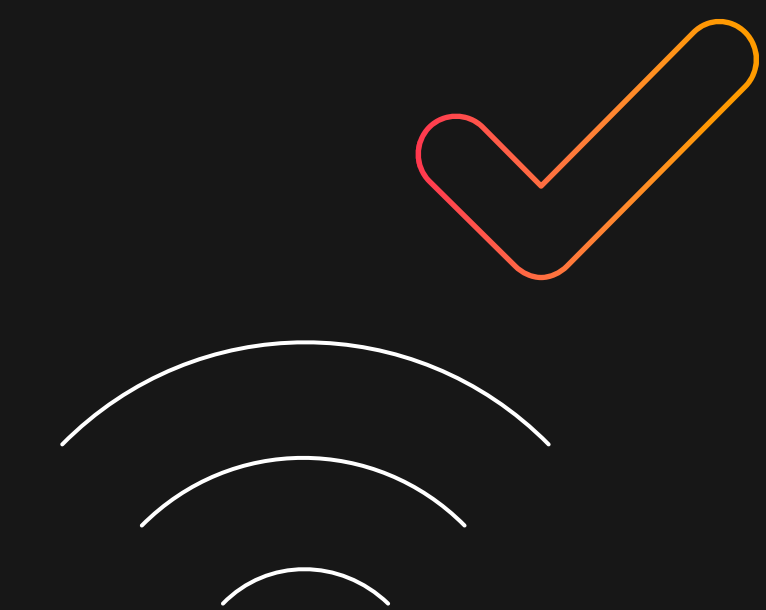
Analyze and debug
issues quickly



End-to-end view of
individual services



Identify customer Support for Serverless
impact



How do I edit and debug my serverless application code?



Author and debug Lambda applications on AWS using your favorite IDEs

DEVELOPER PREVIEW

DEVELOPER PREVIEW



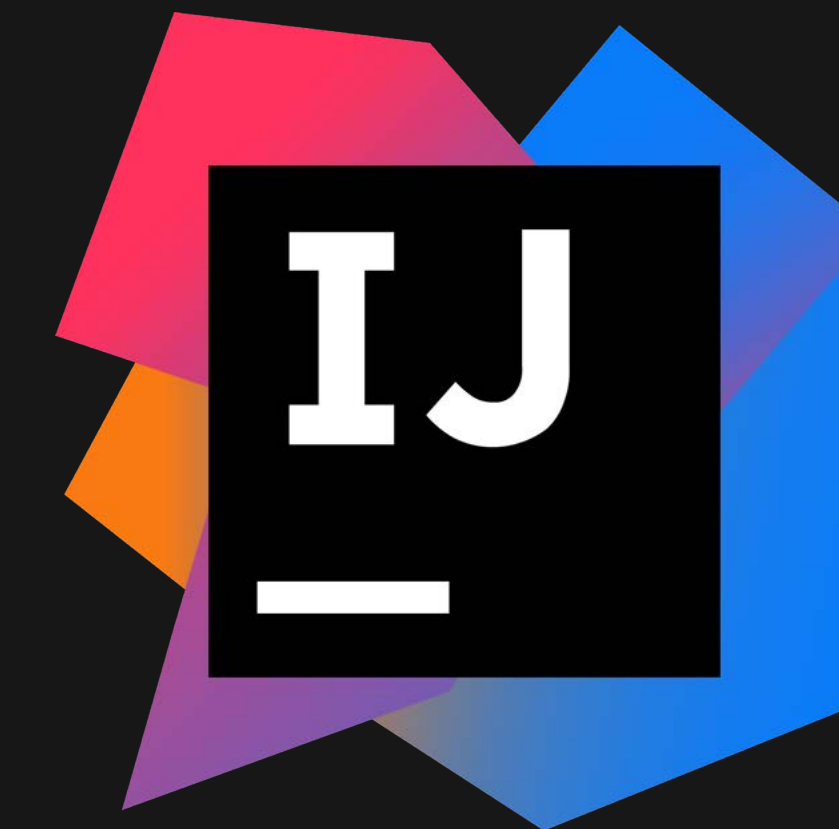
**AWS
Cloud9**

Python, Node



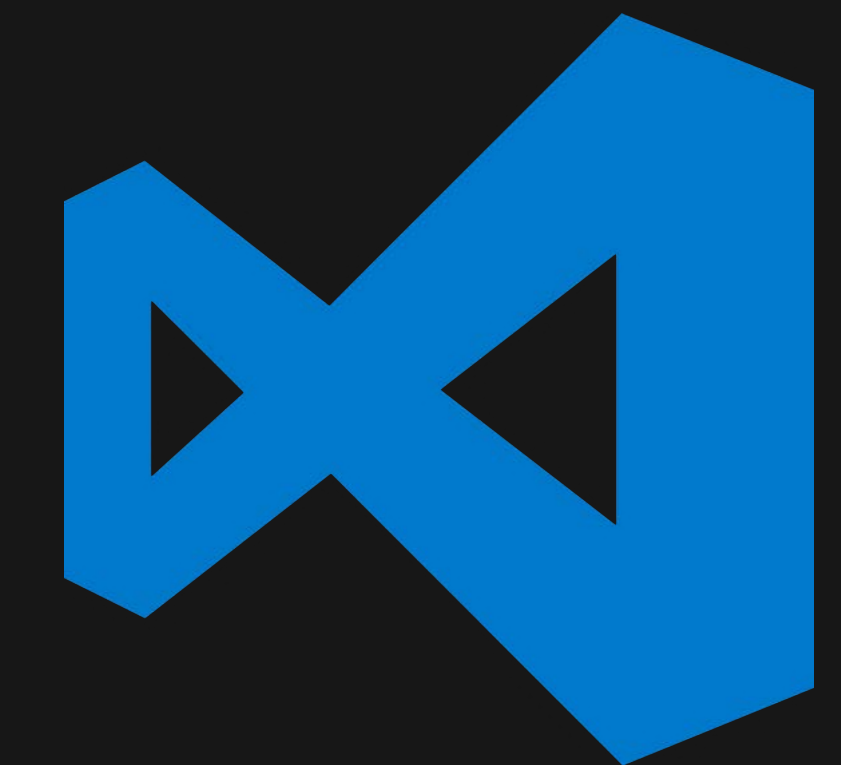
**AWS Toolkit
for PyCharm**

Python



**AWS Toolkit
for IntelliJ**

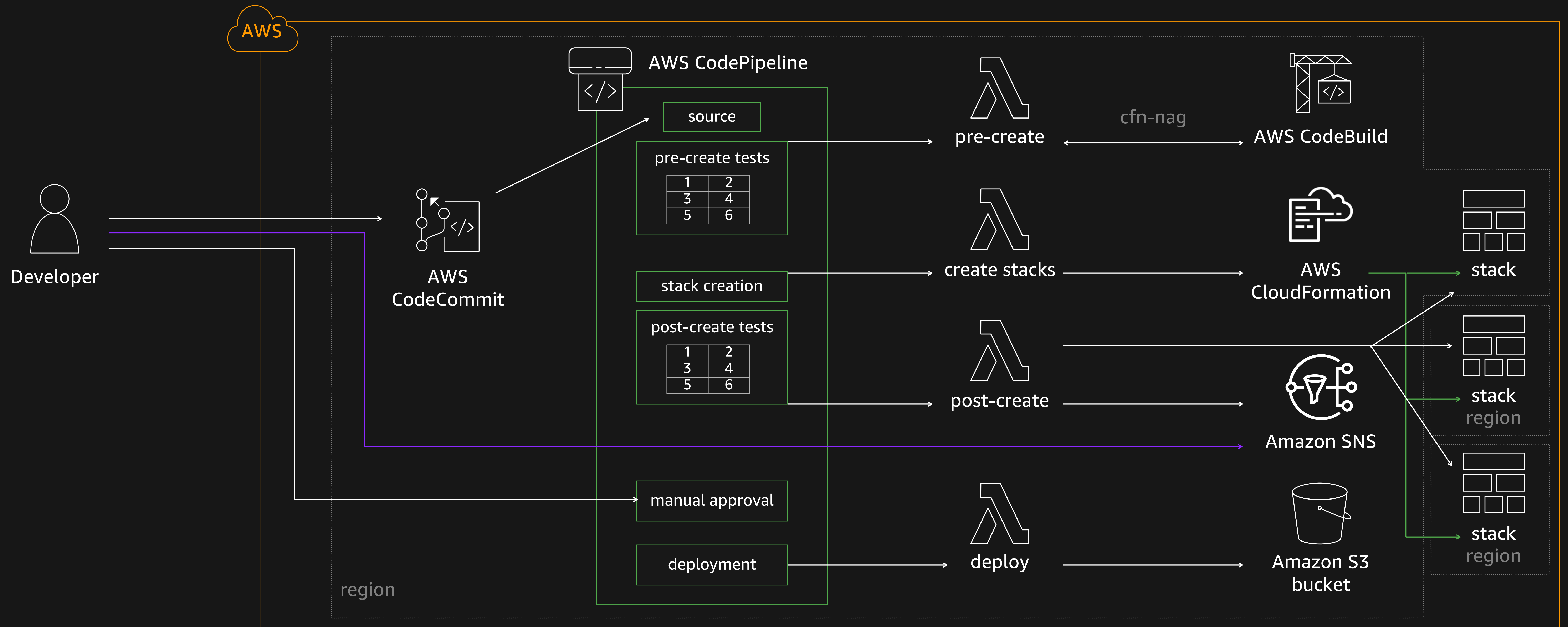
Java, Python



**AWS Toolkit for
Visual Studio Code**

.NET, Node

How can we best model and provision our infrastructure?



Our infrastructure management
journey

Manual

Run a command Actions

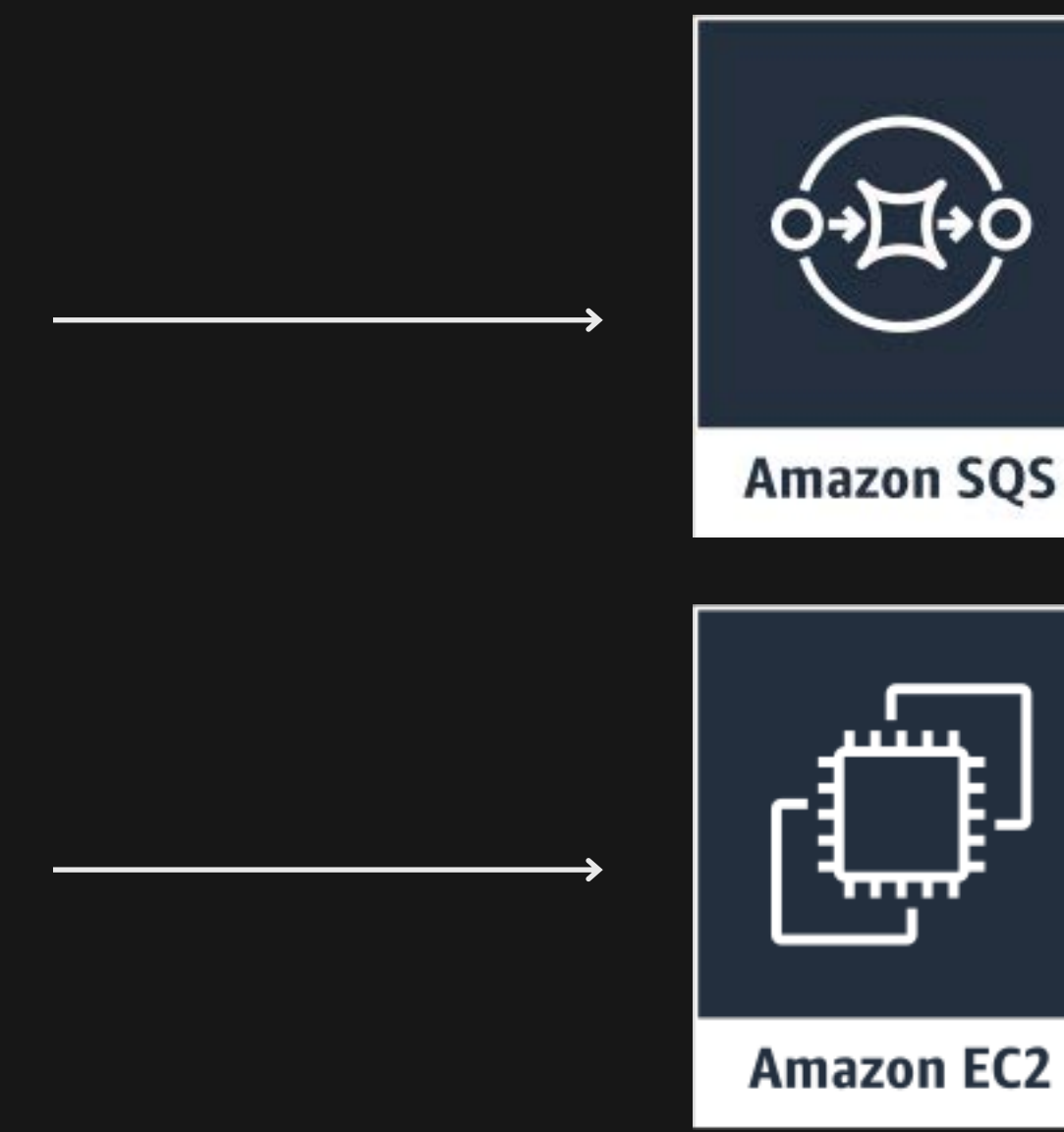
Filter by attributes

Command ID	Instance ID	Document name	Status	Requested date	Comment
65555b90-ee60-45...	i-8fd6aa30	AWS-RunPowerSh...	Success	October 21, 2015 at...	Listing services
65555b90-ee60-45...	i-d583f76a	AWS-RunPowerSh...	Success	October 21, 2015 at...	Listing services
65555b90-ee60-45...	i-8ed6aa31	AWS-RunPowerSh...	Success	October 21, 2015 at...	Listing services
ca4b10c6-cee1-437...	i-d583f76a	AWS-RunPowerSh...	Success	October 20, 2015 at...	getting list of pro
561e5f4a-27d2-419...	i-d583f76a	AWS-RunPowerSh...	Success	October 20, 2015 at...	ipconfig on the t

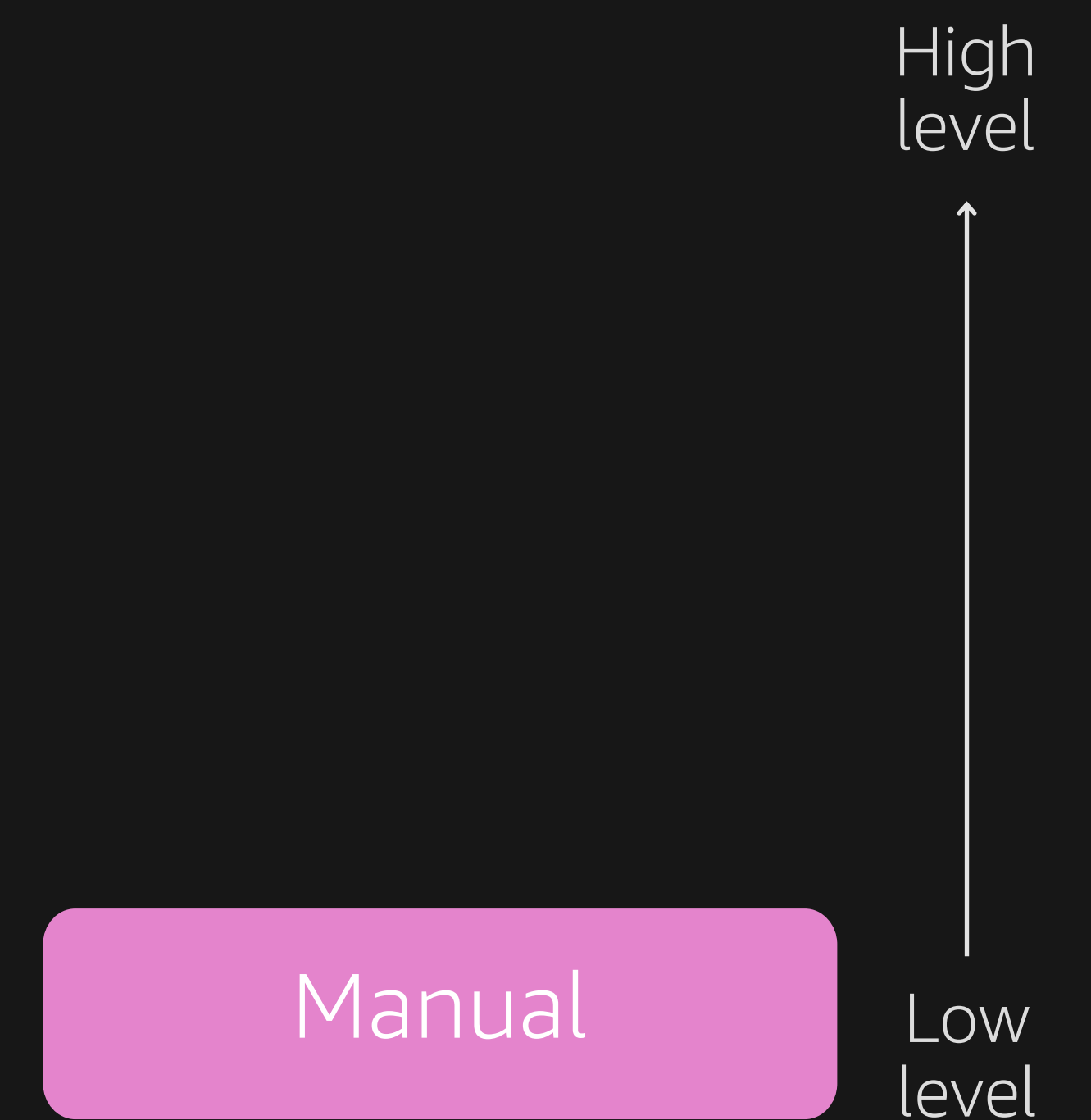
Command ID: 65555b90-ee60-4520-9dc3-e42e94445469 Instance ID: i-8fd6aa30

Description Output

Command ID	65555b90-ee60-4520-9dc3-e42e94445469	Instance ID	i-8fd6aa30
Document name	AWS-RunPowerShellScript	Status	Success
Date requested	October 21, 2015 at 3:56:59 PM UTC-7	Comment	Listing services
Output S3 bucket	run-command-test	Document parameters	



- 👍 Easy to get started
- 🤔 Not reproducible
- 🤔 Error prone
- 🤔 Time consuming

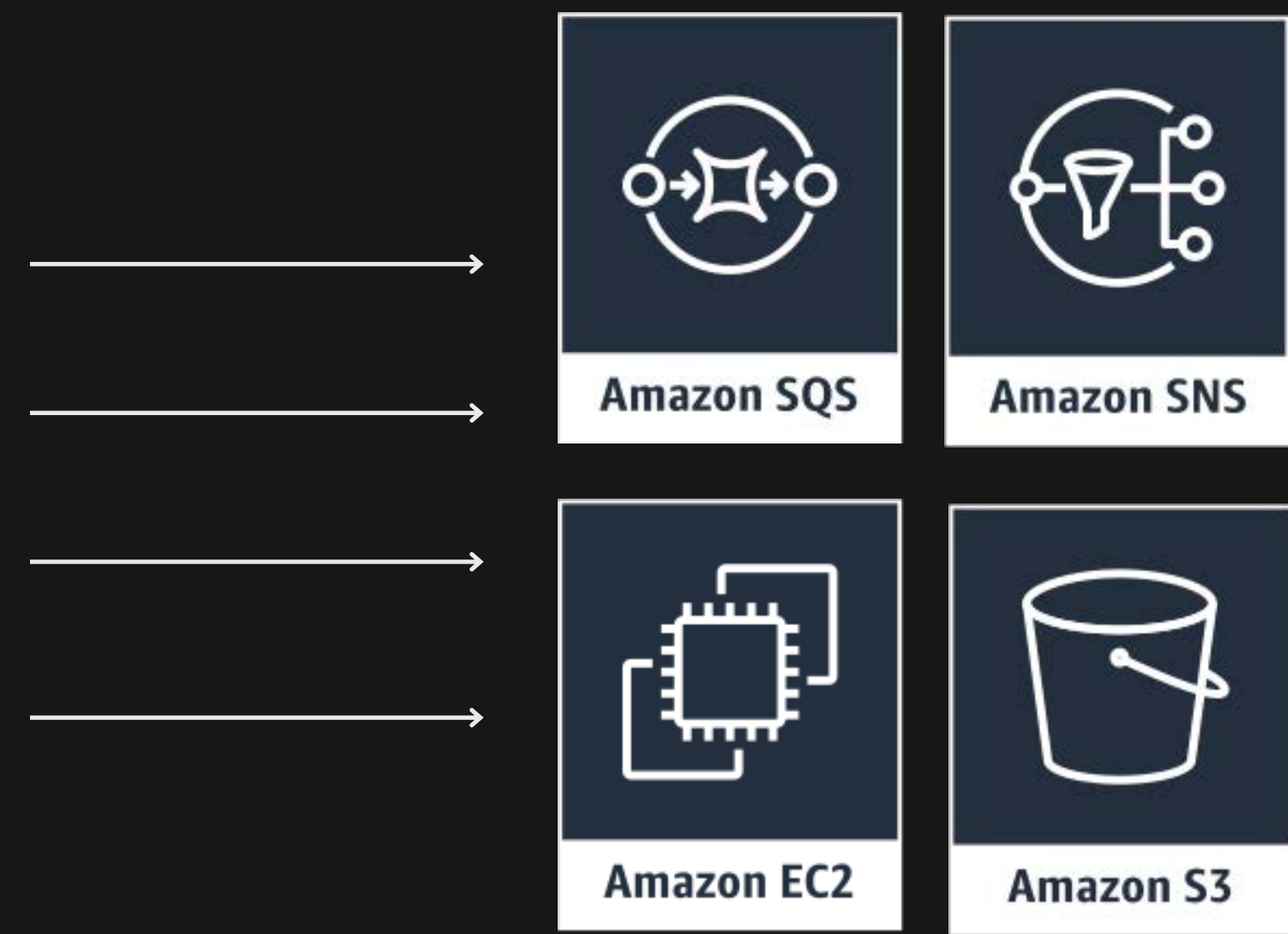


Scripted

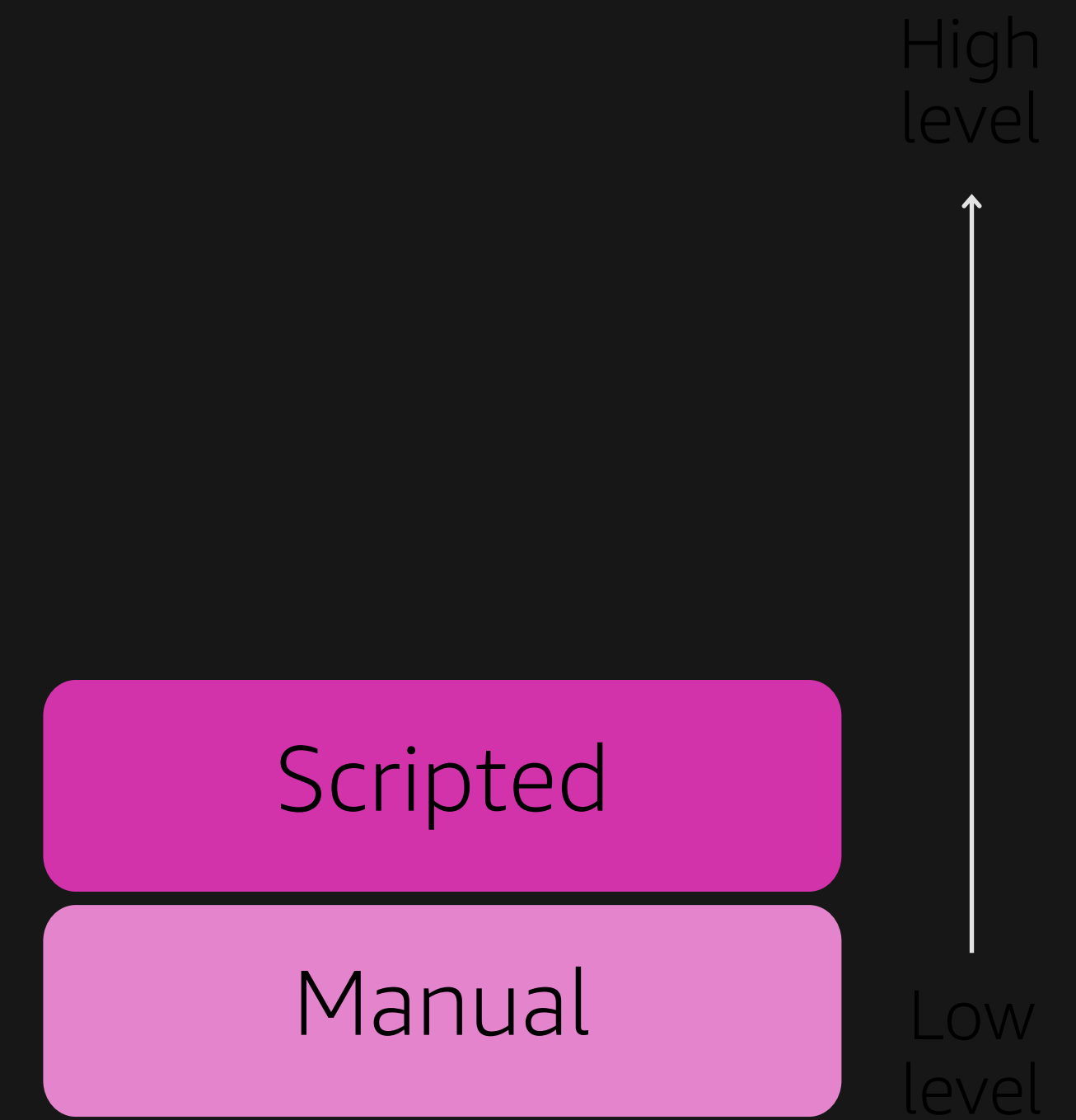
```
require 'aws-sdk-ec2'

ec2 = Aws::EC2::Resource.new(region: 'us-west-2')

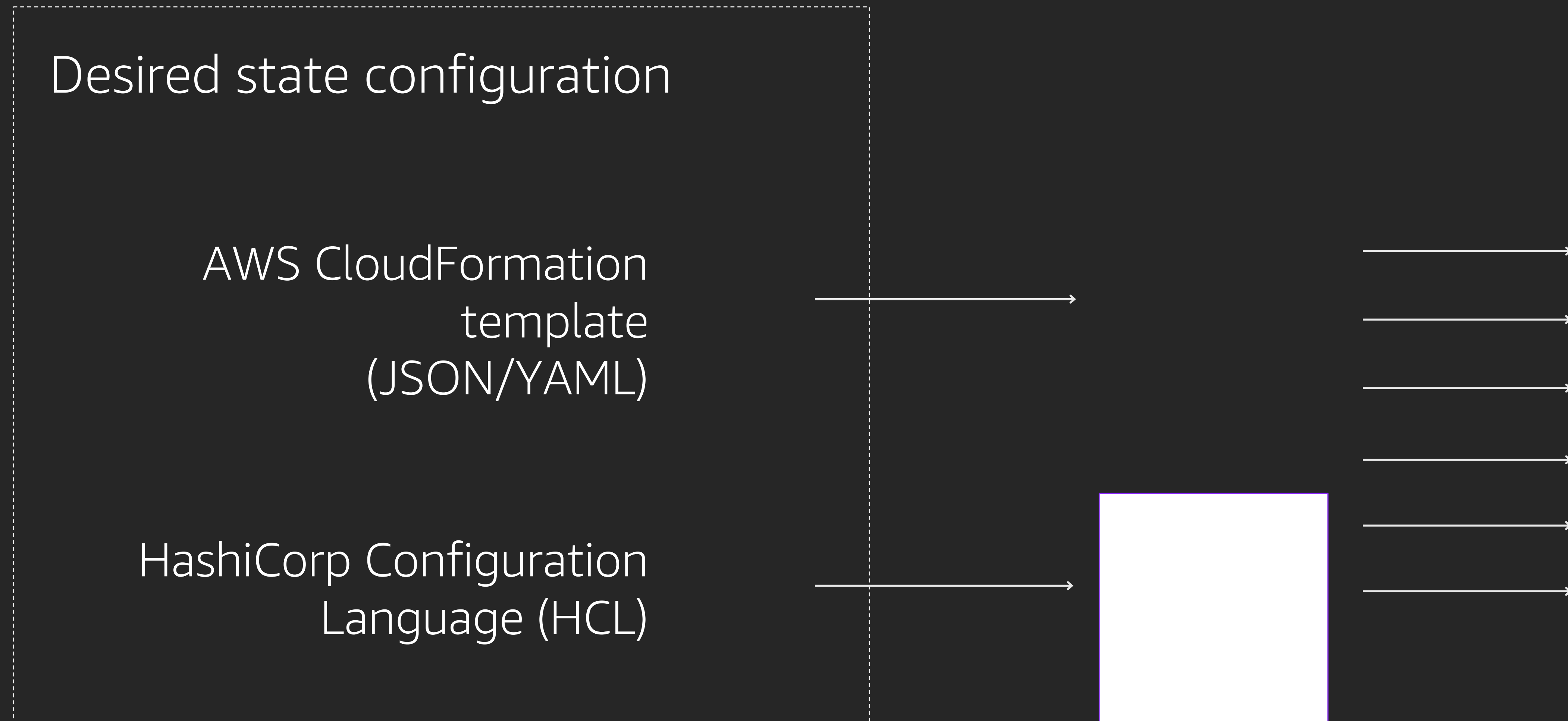
instance = ec2.create_instances({
  image_id: 'IMAGE_ID',
  min_count: 1,
  max_count: 1,
  key_name: 'MyGroovyKeyPair',
  security_group_ids: ['SECURITY_GROUP_ID'],
  instance_type: 't2.micro',
  placement: {
    availability_zone: 'us-west-2a'
  },
  subnet_id: 'SUBNET_ID',
  iam_instance_profile: {
    arn: 'arn:aws:iam::' + 'ACCOUNT_ID' + ':instance-profile/aws-opsworks-ec2-role'
  }
})
```



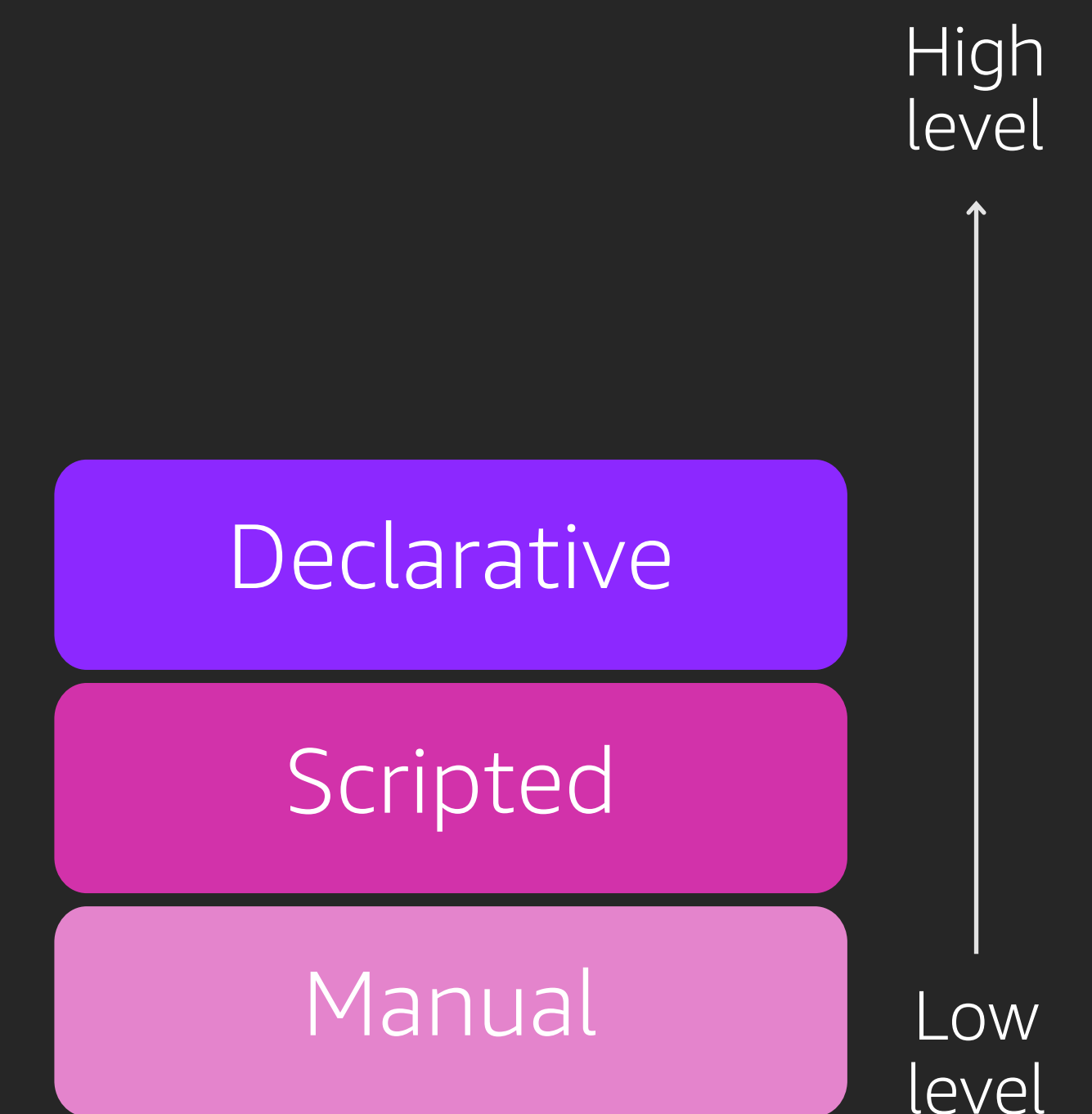
- 🤔 What happens if an API call fails?
- 🤔 How do I make updates?
- 🤔 How do I know a resource is ready?
- 🤔 How do I roll back?



Resource provisioning engines



- 👍 Easy to automate
- 👍 Reproducible
- 😞 Configuration syntax
- 😞 No abstraction, lots of details



Document Object Models (DOMs)

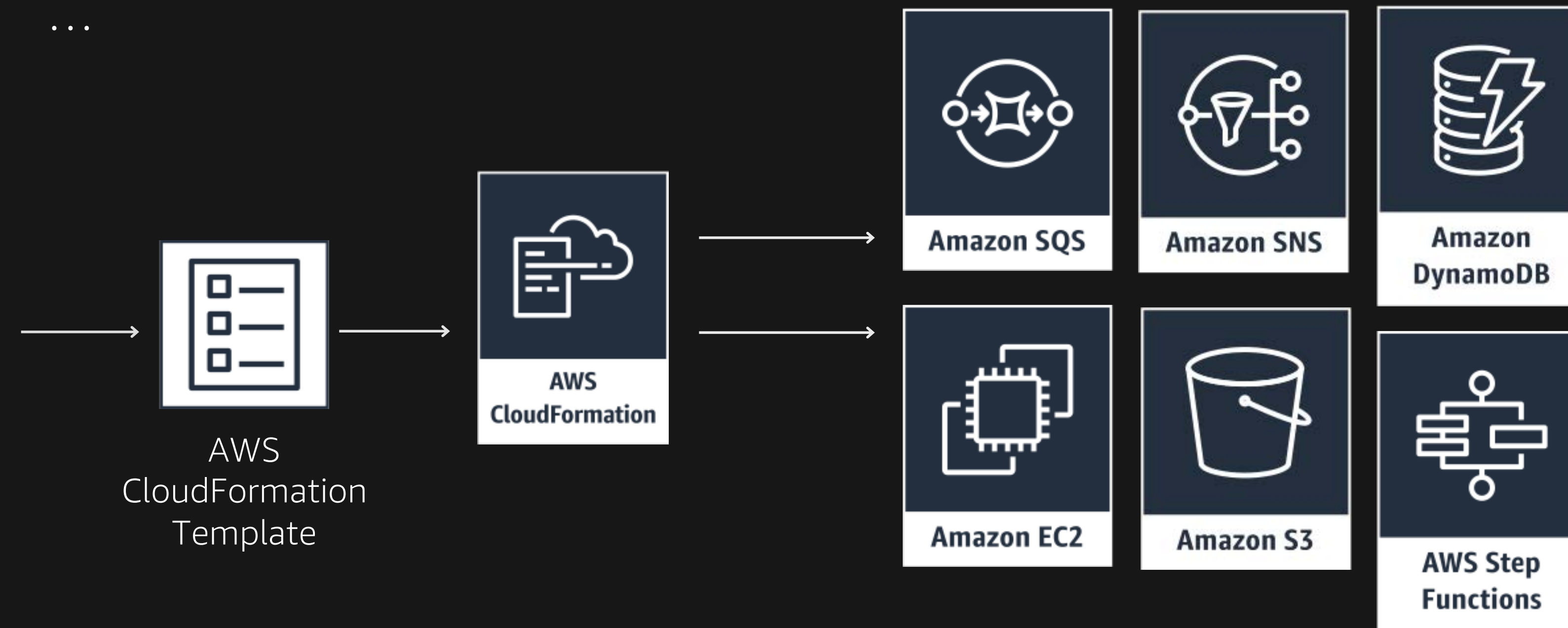
```
from troposphere import Template
from troposphere.ec2 import VPC, Subnet, InternetGateway

t = Template()

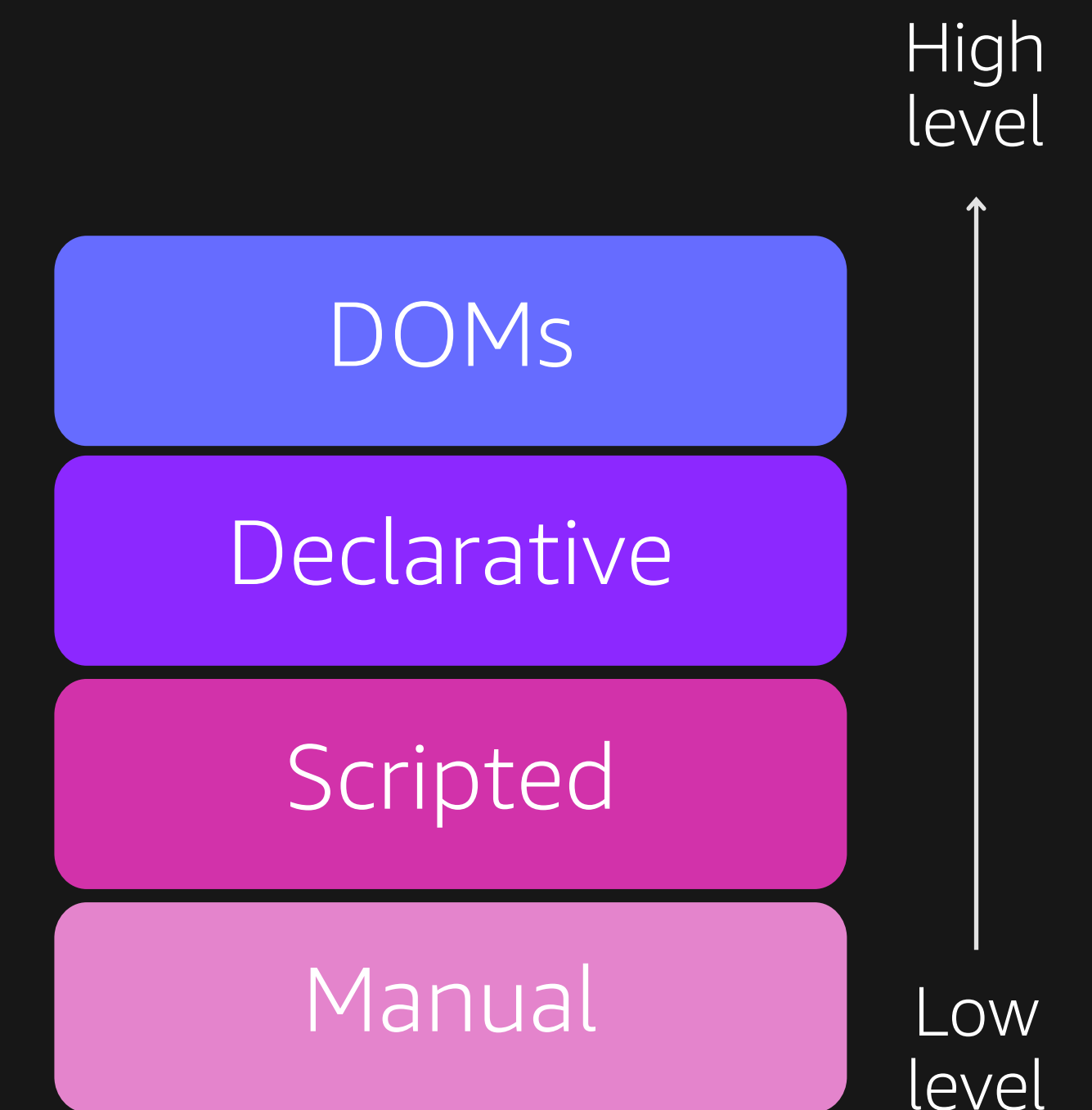
VPC = t.add_resource(
    VPC(
        'VPC',
        CidrBlock='10.0.0.0/16',
        Tags=Tags(
            Application=ref_stack_id)))

subnet = t.add_resource(
    Subnet(
        'Subnet',
        CidrBlock='10.0.0.0/24',
        VpcId=Ref(VPC),
        Tags=Tags(
            Application=ref_stack_id)))
```

Troposphere Python
SparkleFormation Ruby
GoFormation Go
...



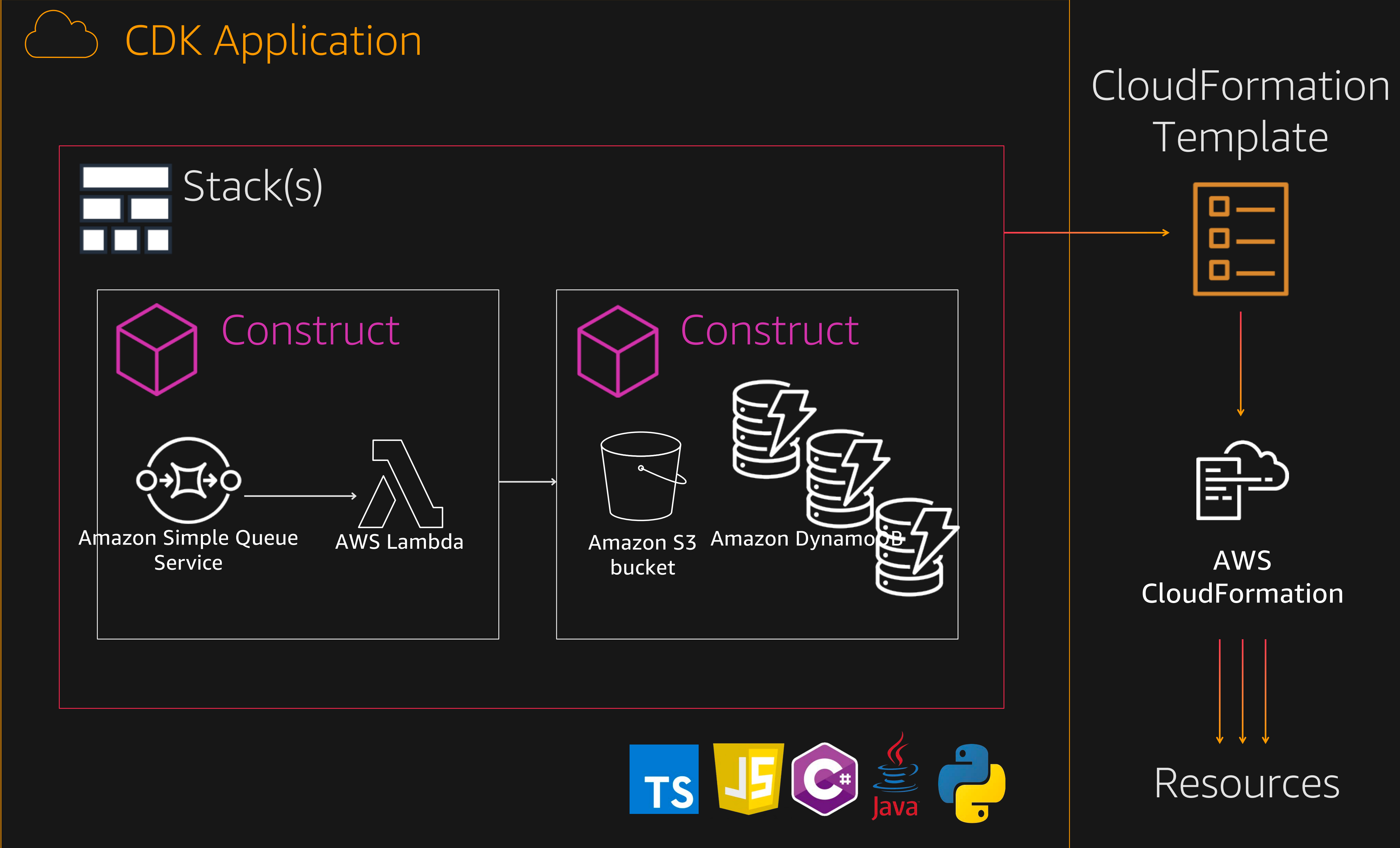
- 👍 Real code ❤️ if statements, for loops, IDE benefits
- 👍 Desired state
- 😞 Abstraction is not built-in Ex: 218 lines of Troposphere for a VPC



AWS Cloud Development Kit (AWS CDK)



AWS Cloud Development Kit





Demo Time

Application models
simplify building
serverless applications



AWS SAM

AWS customers are pioneering modern applications

FICOTM

————— reduced overall compute costs **by 95%**

Coca-Cola

————— cut processing time **from 36 hours to 10 seconds**

FINRA 

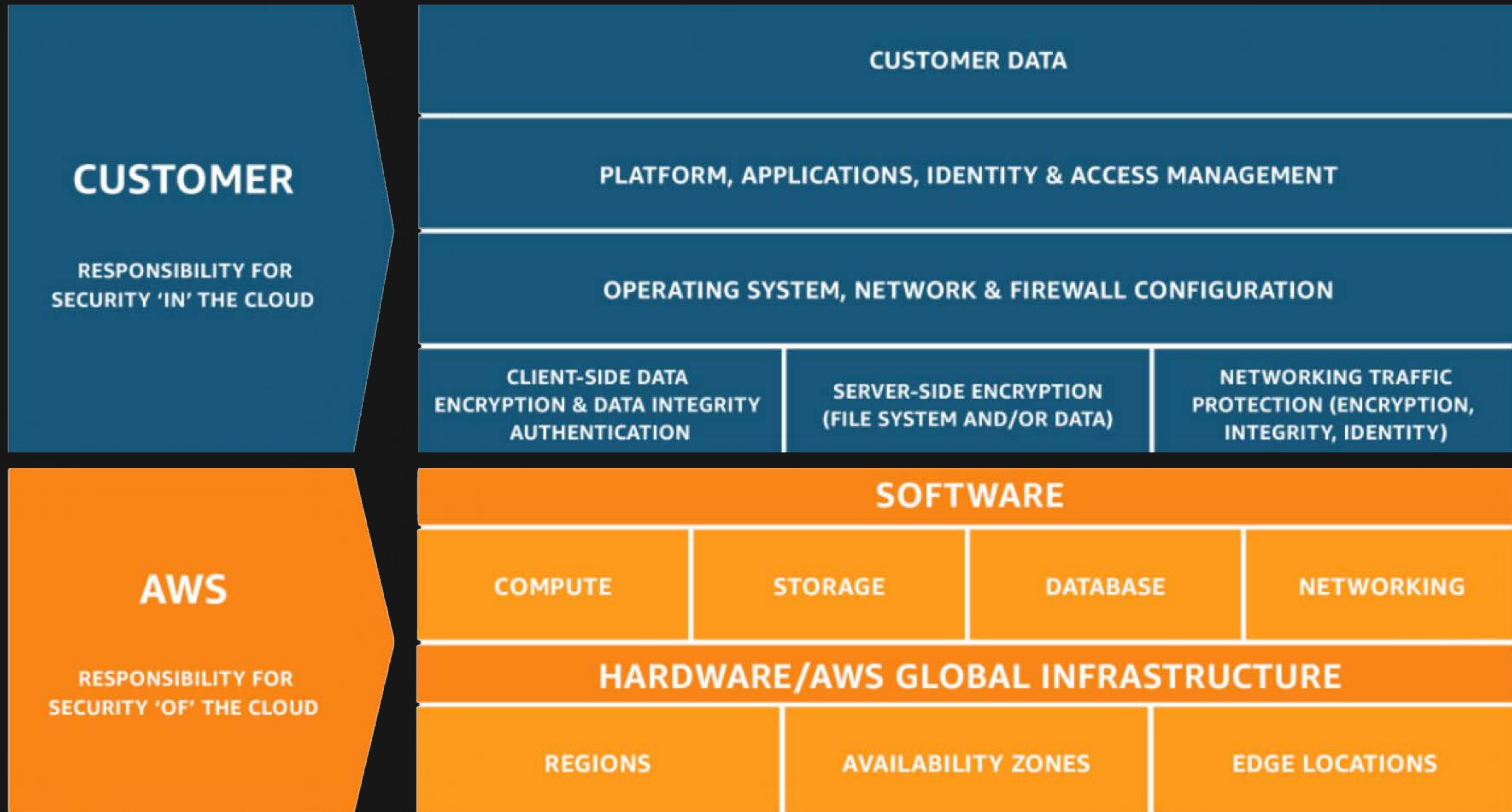
————— created a stock trade validation system in **3 months**

The Washington Post

————— releases **over 50+ deployments per hour**

How do we implement *security* at scale?

Security is a shared responsibility



```
import re  
re.search('([Dd]ev[Ss]ec|[Ss]ec[Dd]ev|[Rr]ugged\s[Dd]ev)[Oo]ps')
```

=

Security Automation

```
import re
re.search('([Dd]ev[ss]ec|[Ss]ec[Dd]ev|[Rr]ugged\s[Dd]ev)[Oo]ps')
```

=

Security **Automation**

Pace of Innovation... meets pace of Protection

Why?

Who?

Where?

When?

What?

Why?

Who?

Where?

When?

What?

Security is everyone's job

Security is a service team, not a blocker
Protect and Serve

Allow flexibility and freedom
but control the flow and result.

Meet the new security team

Operations

Engineering

Application Security

Compliance

Meet the new security team

Operations

Engineering

DEVELOPMENT

Application Security

Compliance



Why?

Who?

Where?

When?

What?

Continuous Integration / Continuous Deployment

1. Security **of** the CI/CD Pipeline

Access roles

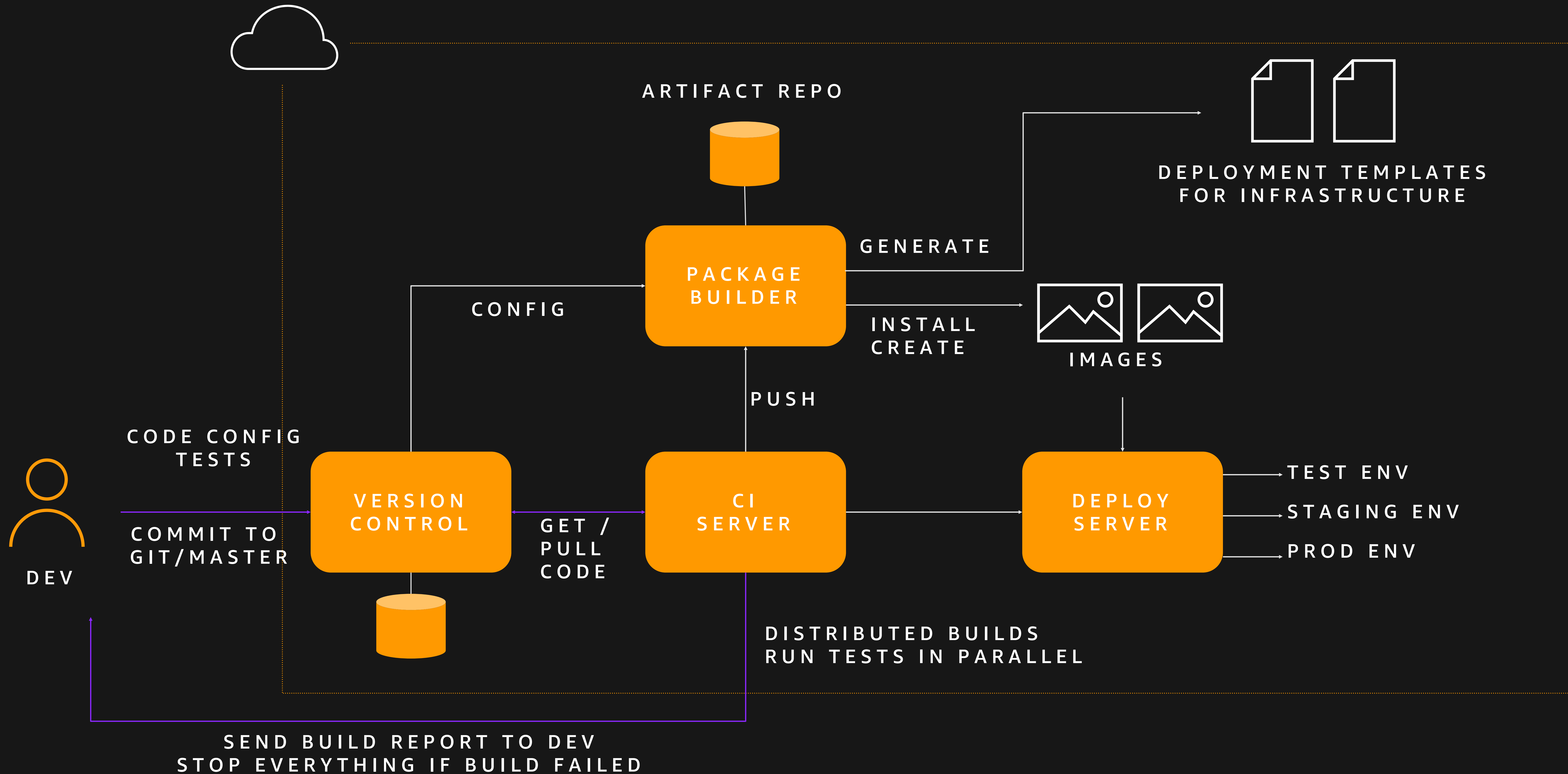
Hardening build servers/nodes

2. Security **in** the CI/CD Pipeline

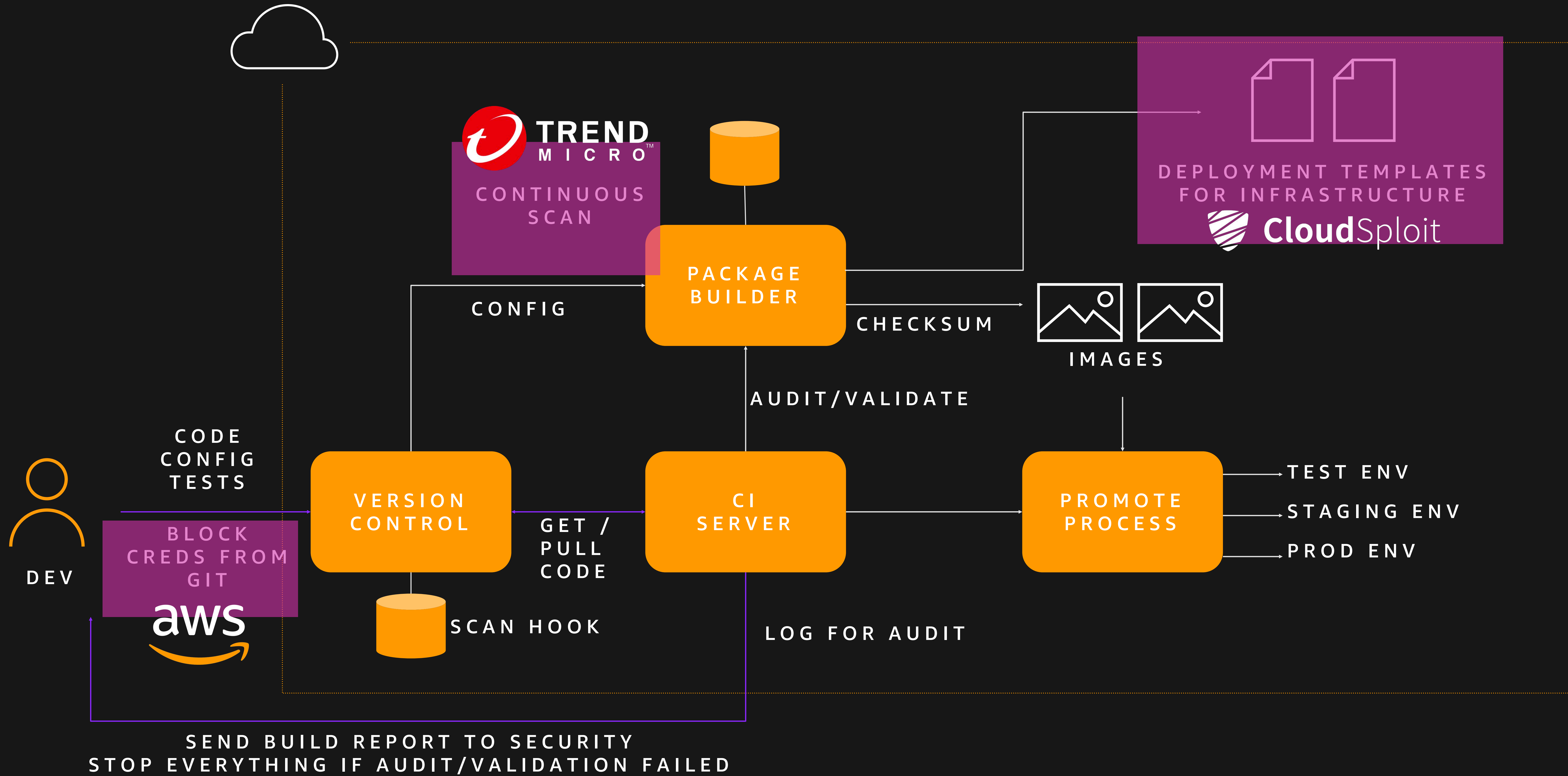
Artifact validation

Static code analysis

CI/CD for DevOps



CI/CD for DevSecOps



3. Cloud scale security

a.k.a. all the other stuff people are really talking about

Infrastructure as code

Base requirement!

Split ownership

Pre-deploy validation

Run time security

Tag-based targeting

Rip-n-replace

Continuous pen testing

Elastic security automation

API driven

Auto Scaling groups – hooks

Execution layer scales with targets

Immutable infrastructure

Validation and enforcement

Integrate with managed services

Why?

Who?

Where?

When?

What?

When? *Easy*. All the time!

When – Control and Validate

Pre-event - When possible

Store infrastructure in code repository

Validate each push (git hooks)

Use managed microservices as execution engine

Scan cloud infrastructure templates for unwanted/risk valued configurations

Validate container definitions

Validate system code early on

Find unwanted libraries, etc.

Force infrastructure changes through templates

Block if needed/unsure

When – Control and Validate

Post-event - **Always**

- Follow-up on sensitive APIs

 - IAM, security groups/firewall, encryption keys, logging, etc.

 - Alert/inform

- Use source of truth

 - Locked to execution function (read only)

- Validate source**

 - Human or machine/CICD

- Decide on remediation

Why?

Who?

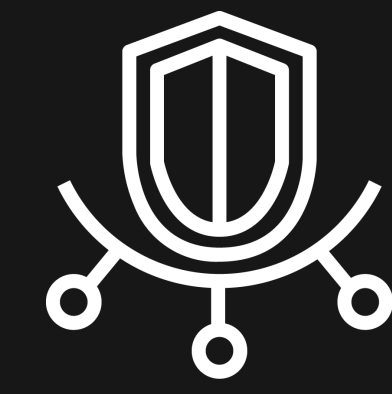
Where?

When?

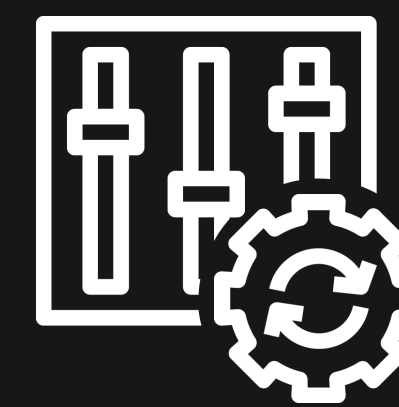
What?

What?

+ Some real world examples



AWS Trusted Advisor



AWS Config



Amazon Inspector



Amazon CloudWatch

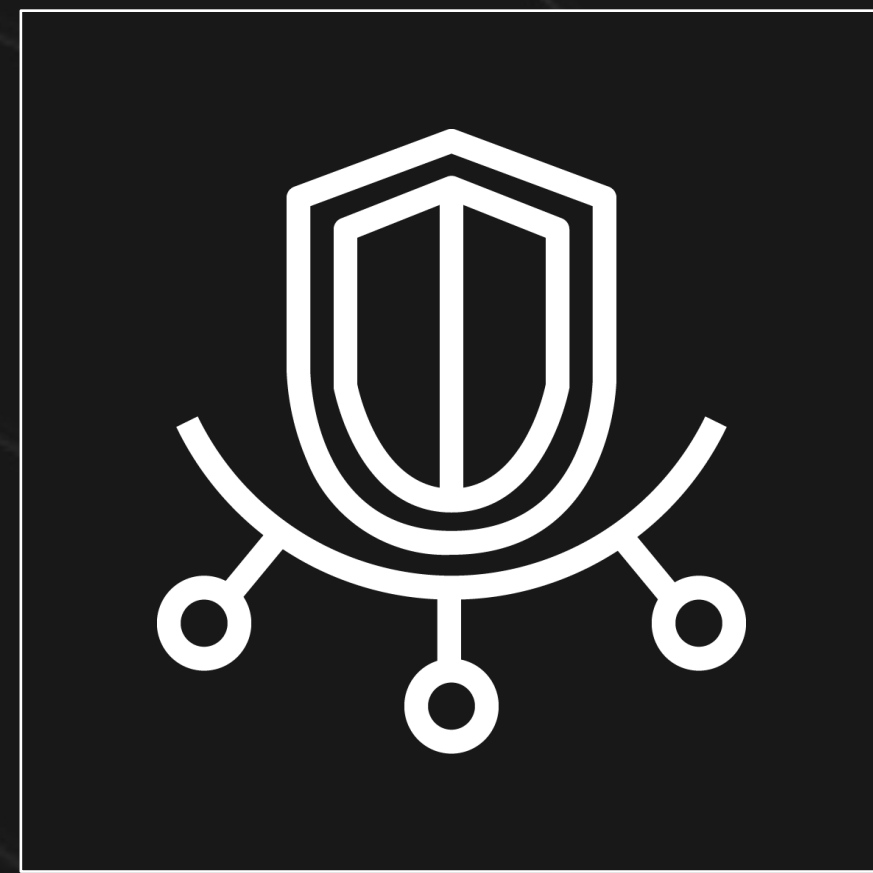


AWS CloudTrail



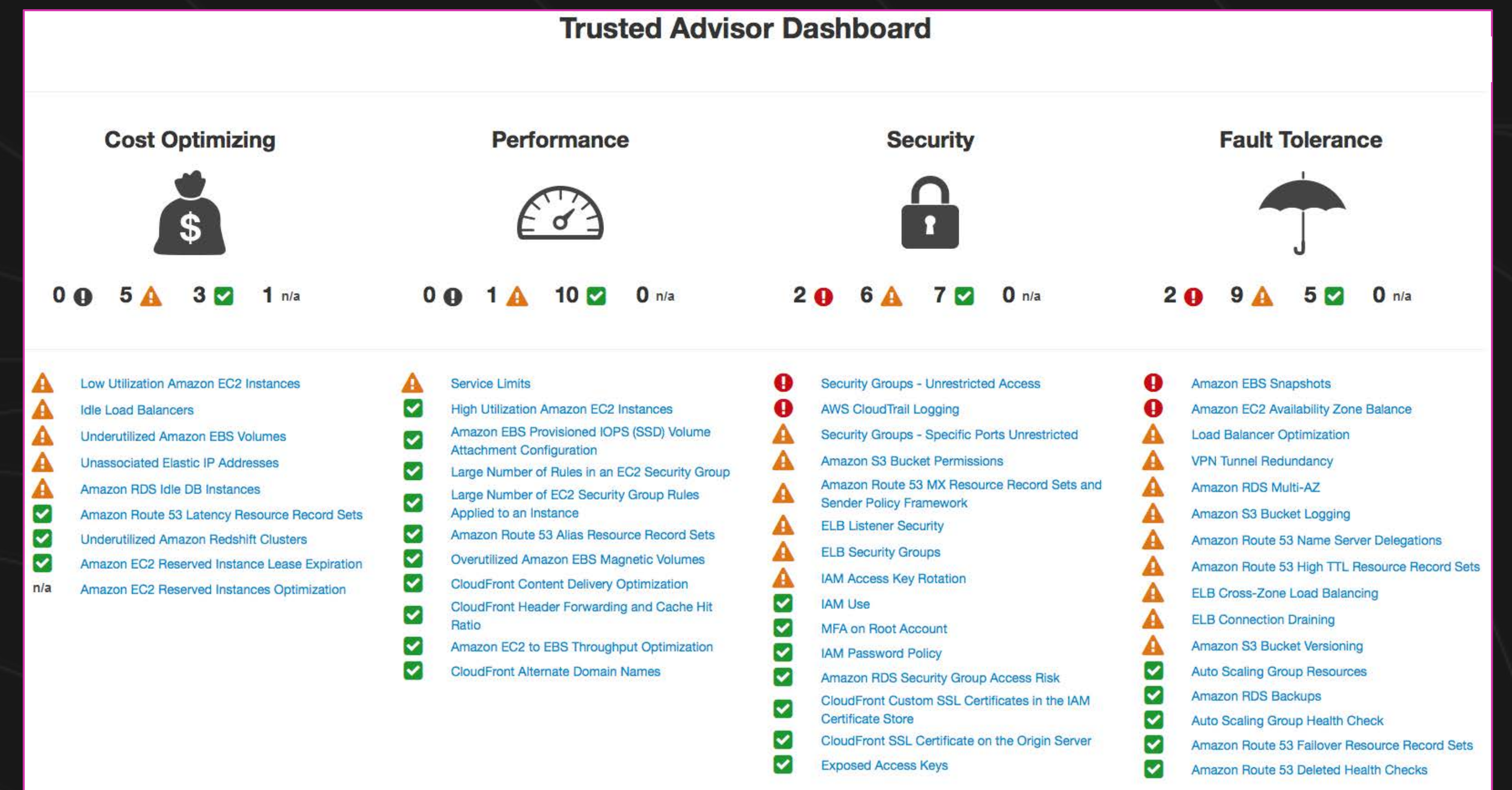
Amazon Macie

AWS Trusted Advisor – Real time guidance

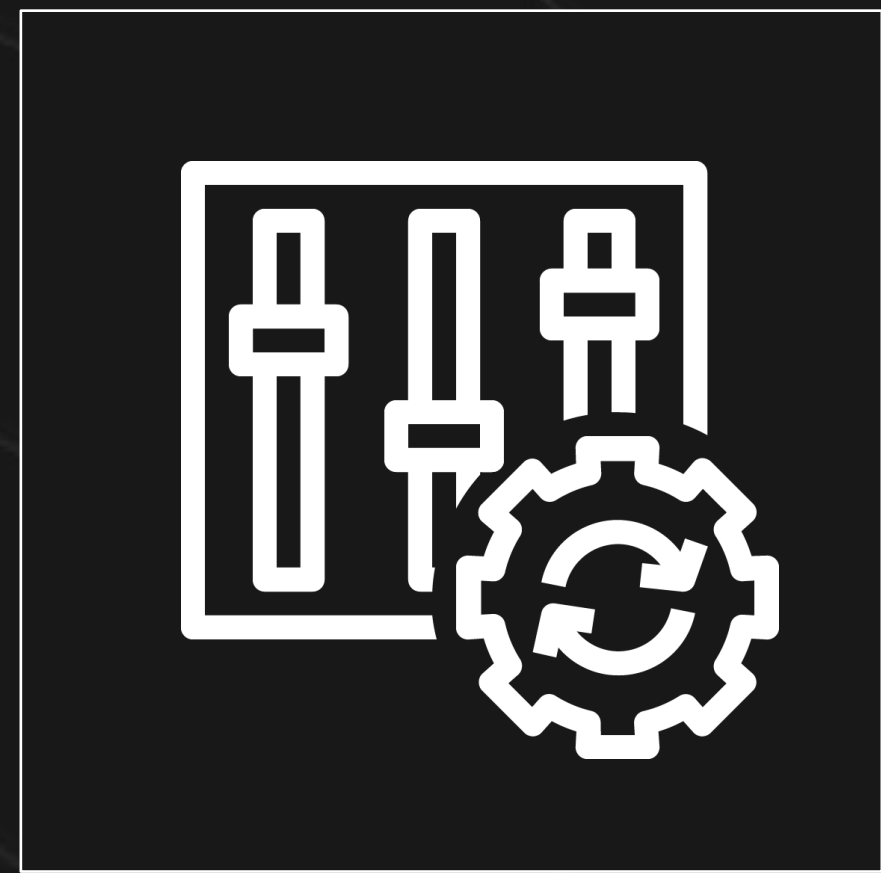


Security configuration checks of your AWS environment:

- Open ports
- Unrestricted access
- CloudTrail Logging
- S3 Bucket Permissions
- Multi-factor auth
- Password Policy
- DB Access Risk
- DNS Records
- Load Balancer config



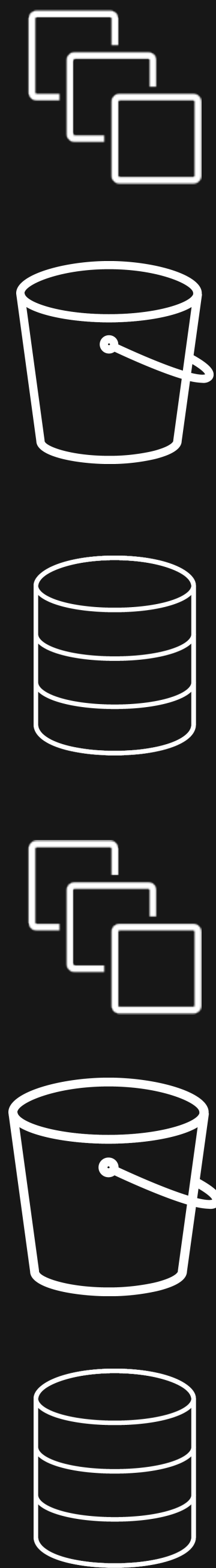
AWS Config – Configuration monitoring



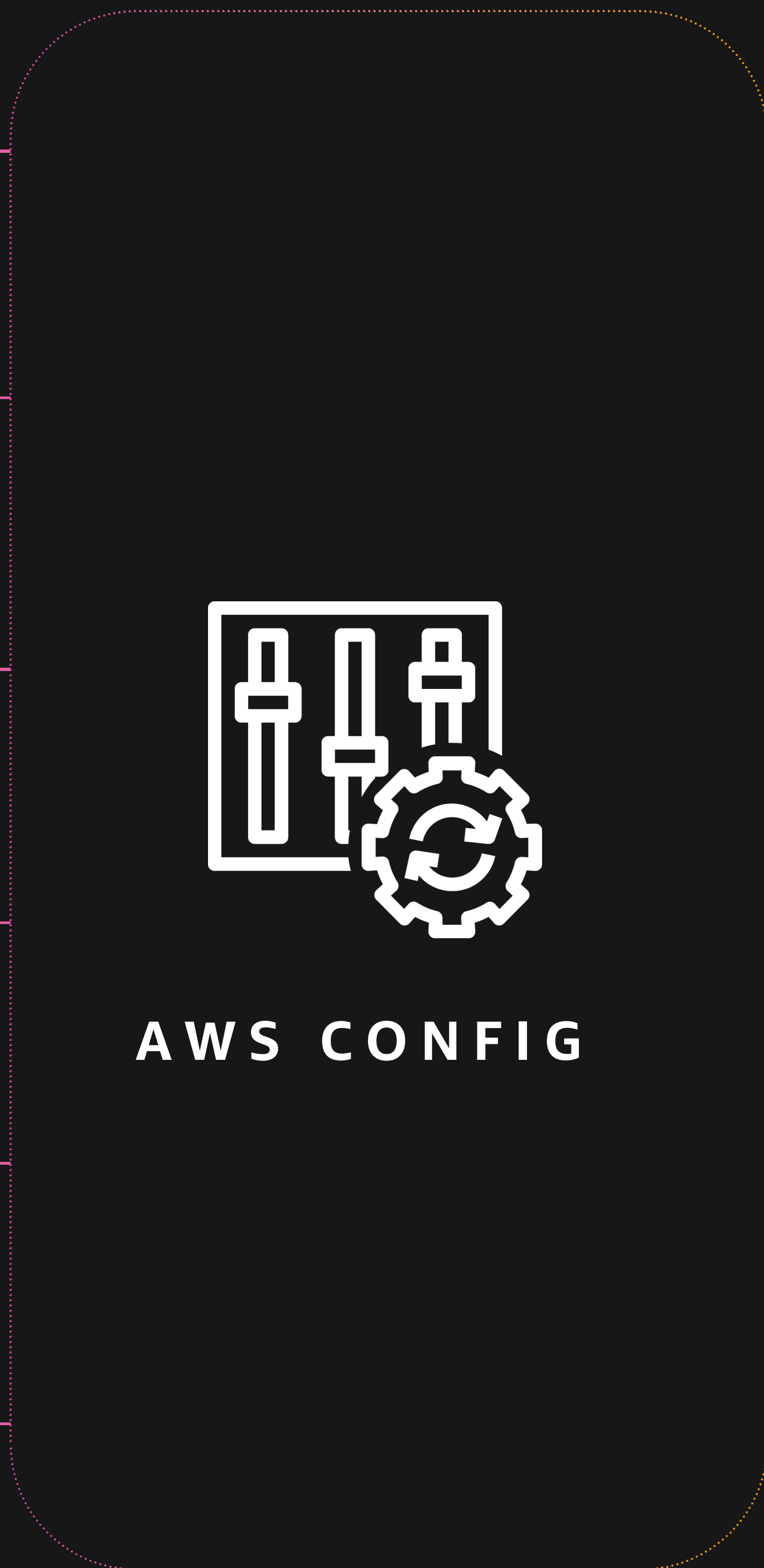
AWS Config is a fully managed service that provides you with an inventory of your AWS resources, lets you audit the resource configuration history and notifies you of resource configuration changes.

AWS Config Rules

CHANGING RESOURCES



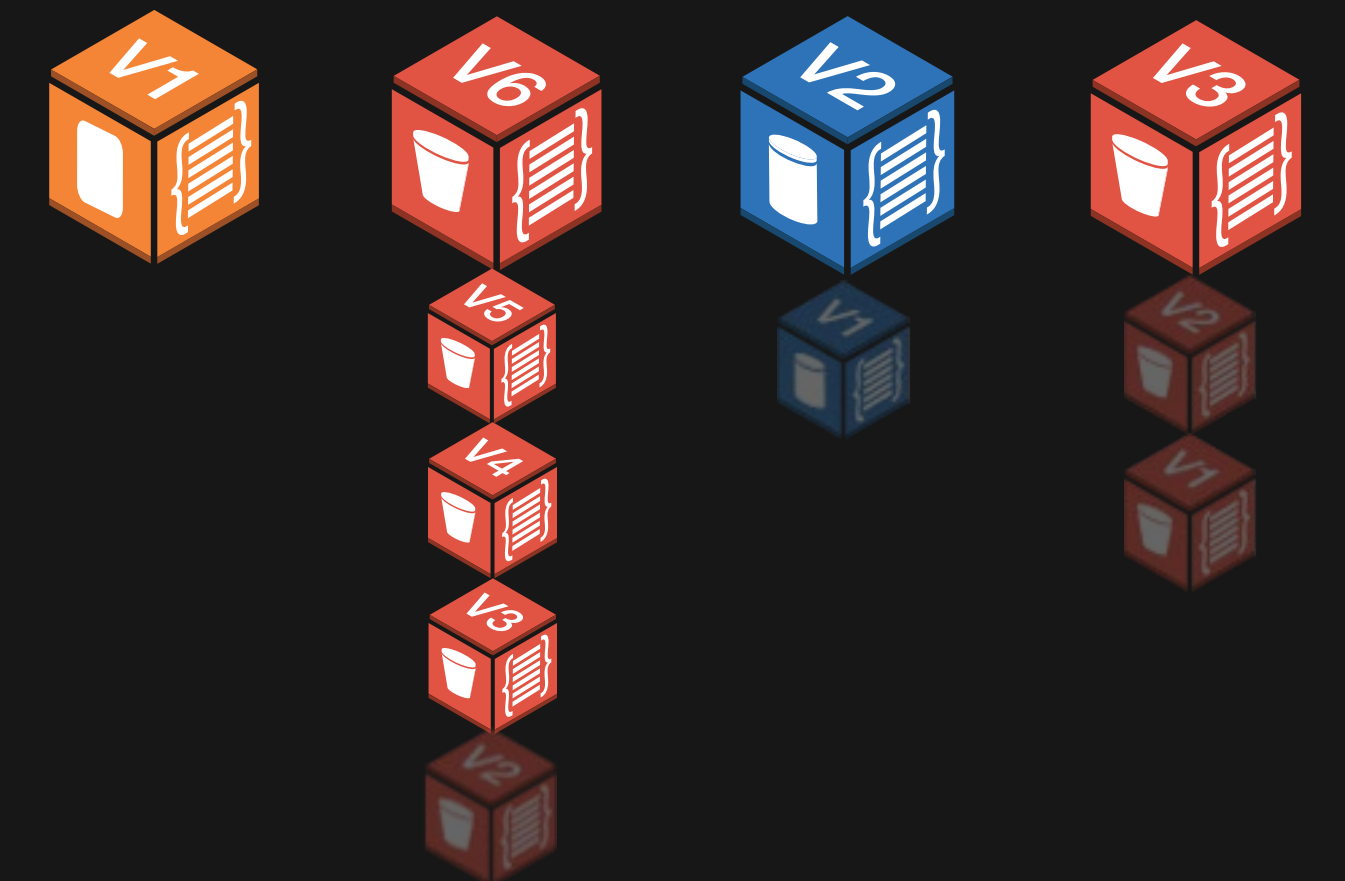
RECORDING



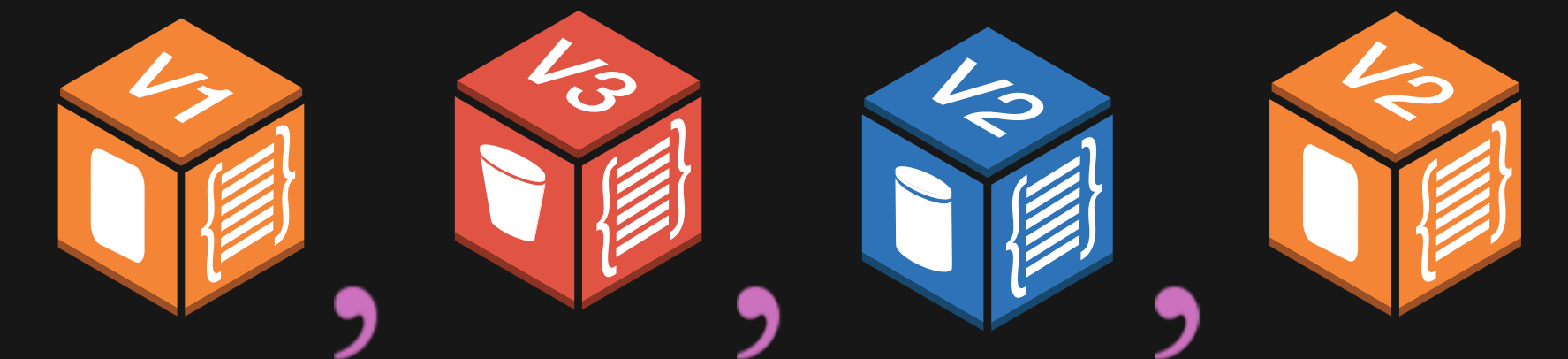
CONTINUOUS CHANGE



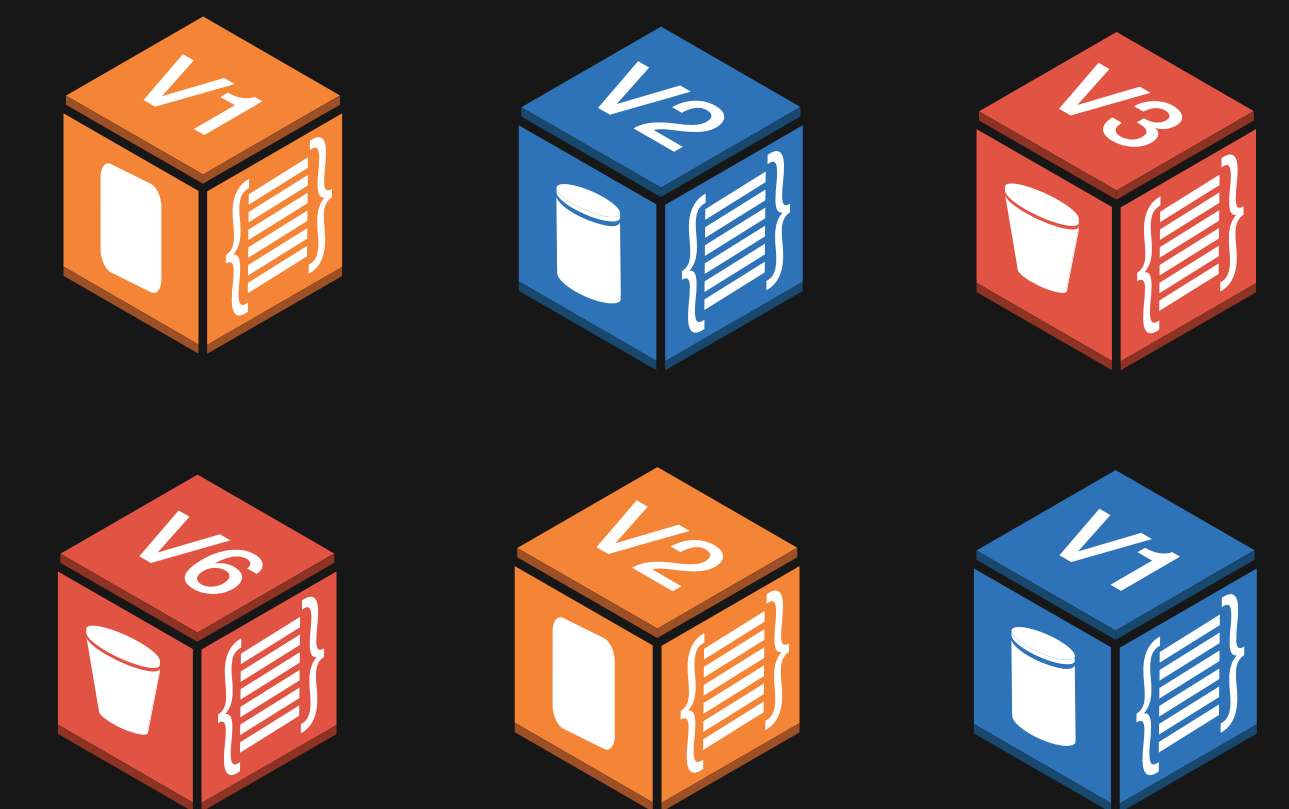
HISTORY



STREAM



SNAPSHOT (EX. 2014-11-05)



AWS CloudTrail – “Cloud” usage logging



You are making API calls...

On a growing set of services around the world...

AWS CloudTrail is continuously recording API calls...

And delivering log files to you



Amazon Macie

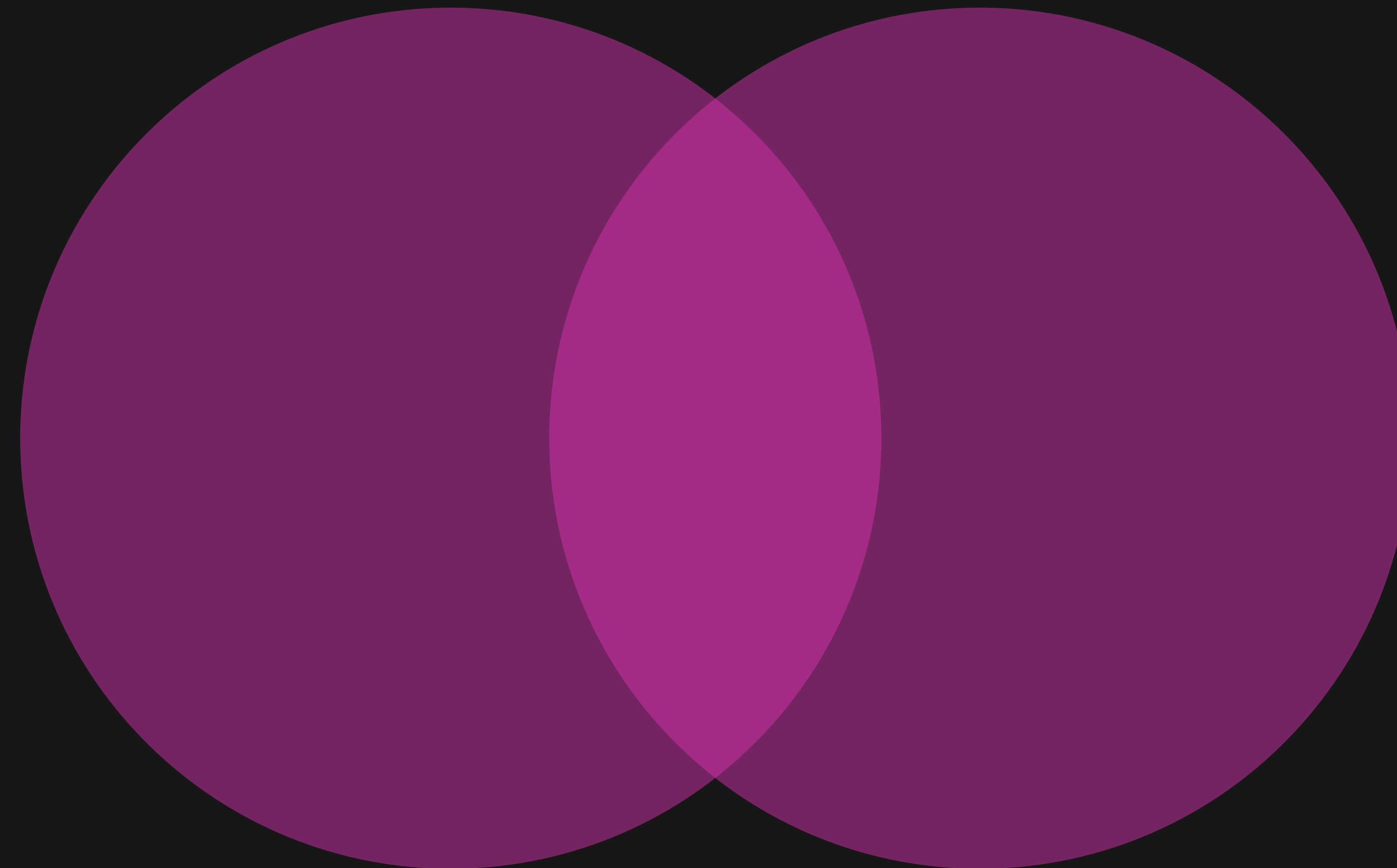
AI-powered visibility service identifies sensitive information to HELP automate security and compliance

Data Management Challenges

- What data do I have in the cloud?
- Where is it located?
- How is data being shared and stored?
- How can I classify data in near-real time?
- What PII/PHI is possibly exposed?
- How do I build workflow remediation for my security and compliance needs?

Apply Machine Learning

Understand Your Data
Natural Language
Processing (NLP)



Understand Data Access
Predictive User
Behavior Analytics (UBA)

Machine Learning Goals for Security

Categorize new or unknown threads based on known and theorized examples

High coverage (volume of true positives)

High accuracy (few false positives)

Adaptive

Machine Learning Challenges for Security

Every customer is different

Threats are ever changing

Penalty for error is high

Flood of data

How Does Amazon Macie use Machine Learning

Understand behavioral analytics to baseline normal behavior

Train and develop contextualized alerts by understanding
the value of data being accessed

Context for content

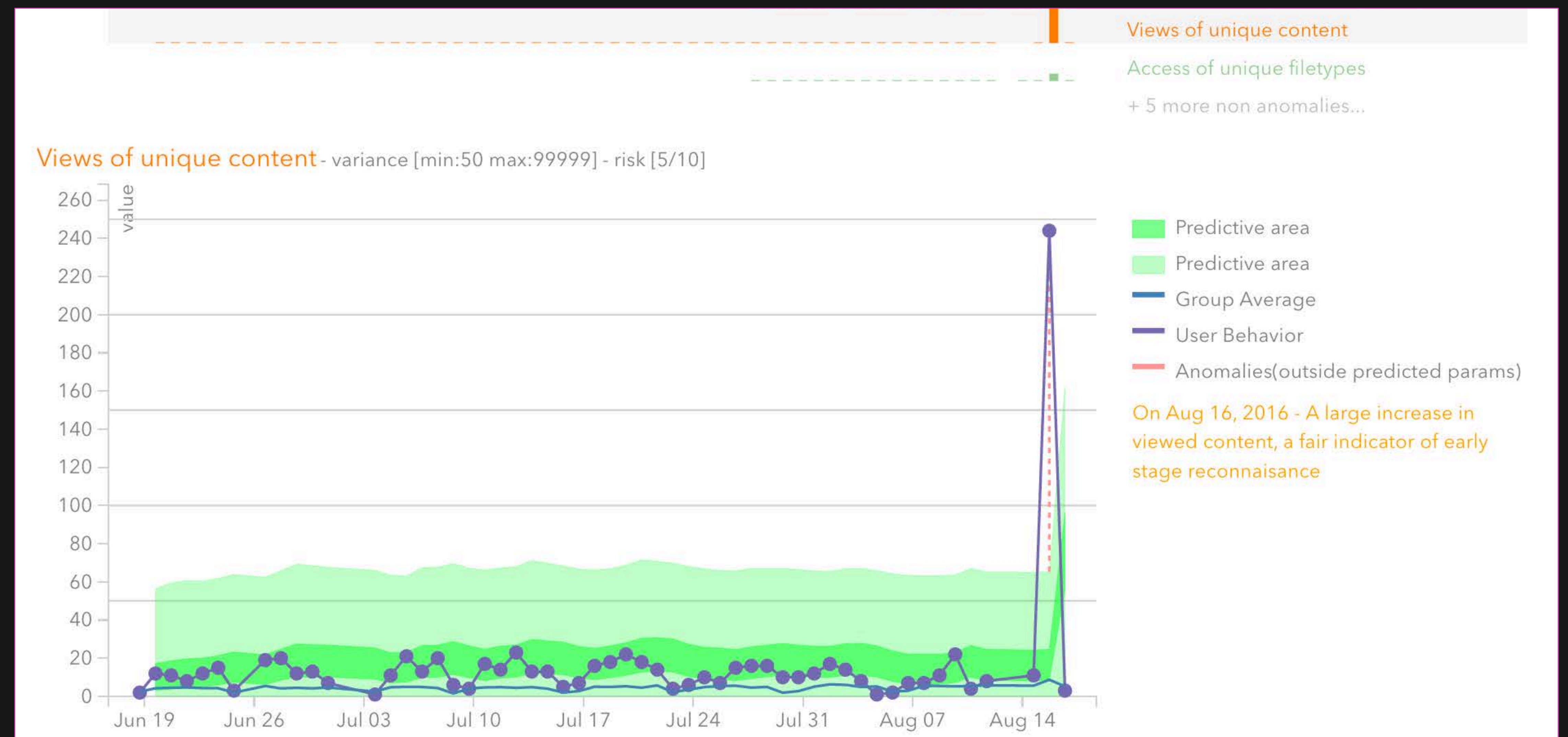
Amazon Macie Behavior Analytics

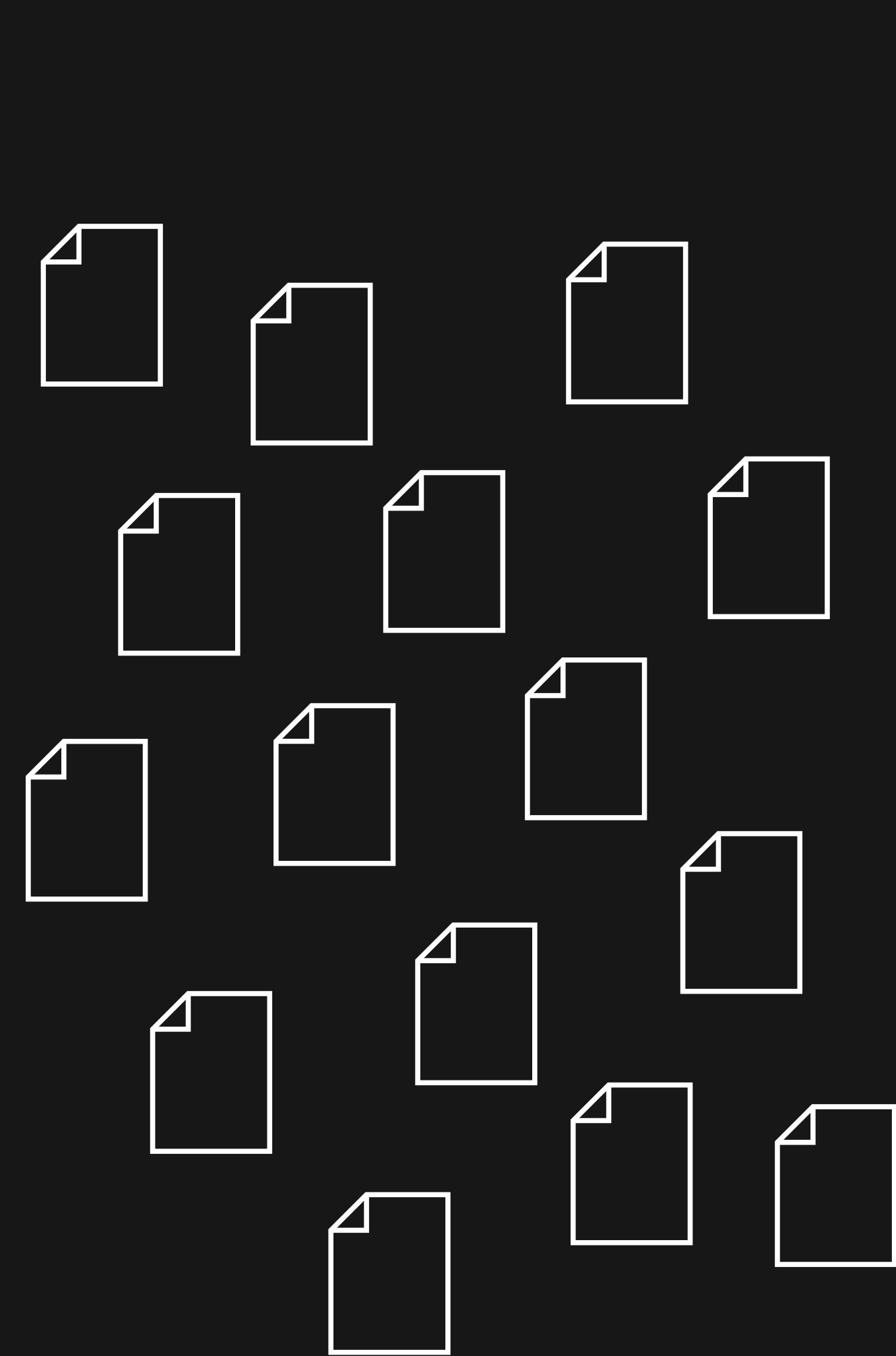
We use behavioral analytics to baseline normal behavior patterns.

Contextualize by value of data being accessed.

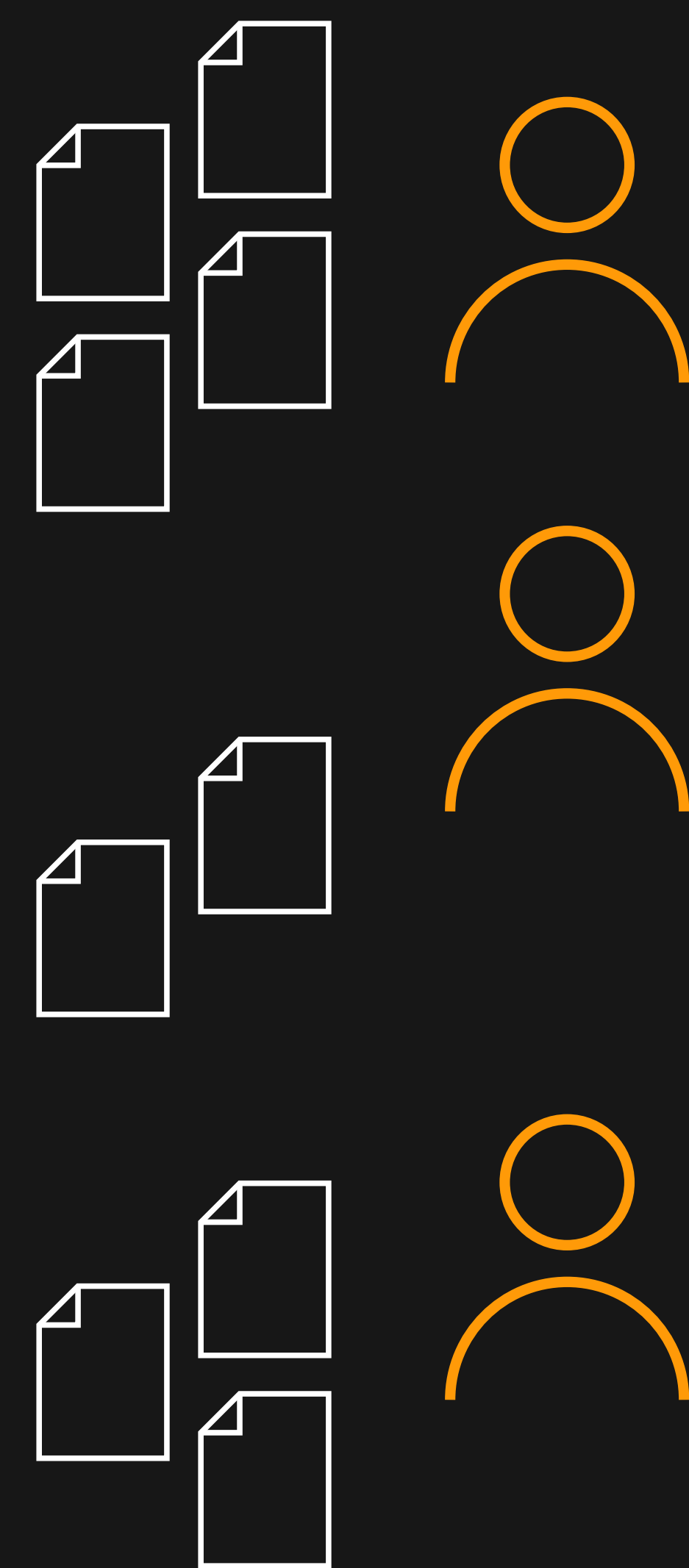
Goals:

- Go to crazy lengths to avoid false positives
- Features, features
- Compare peers
- Tell a narrative



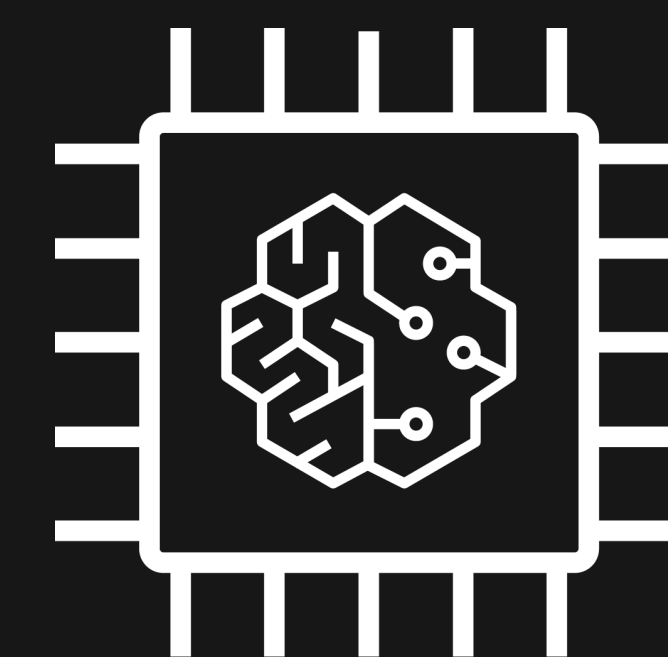
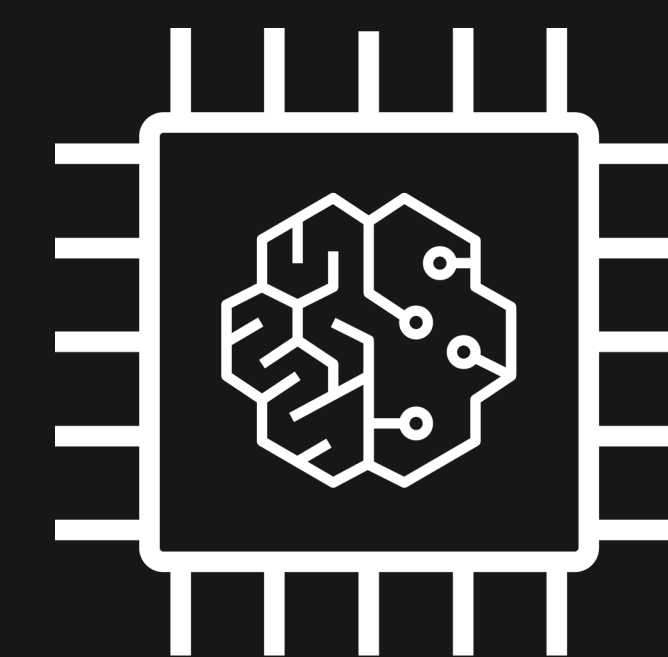
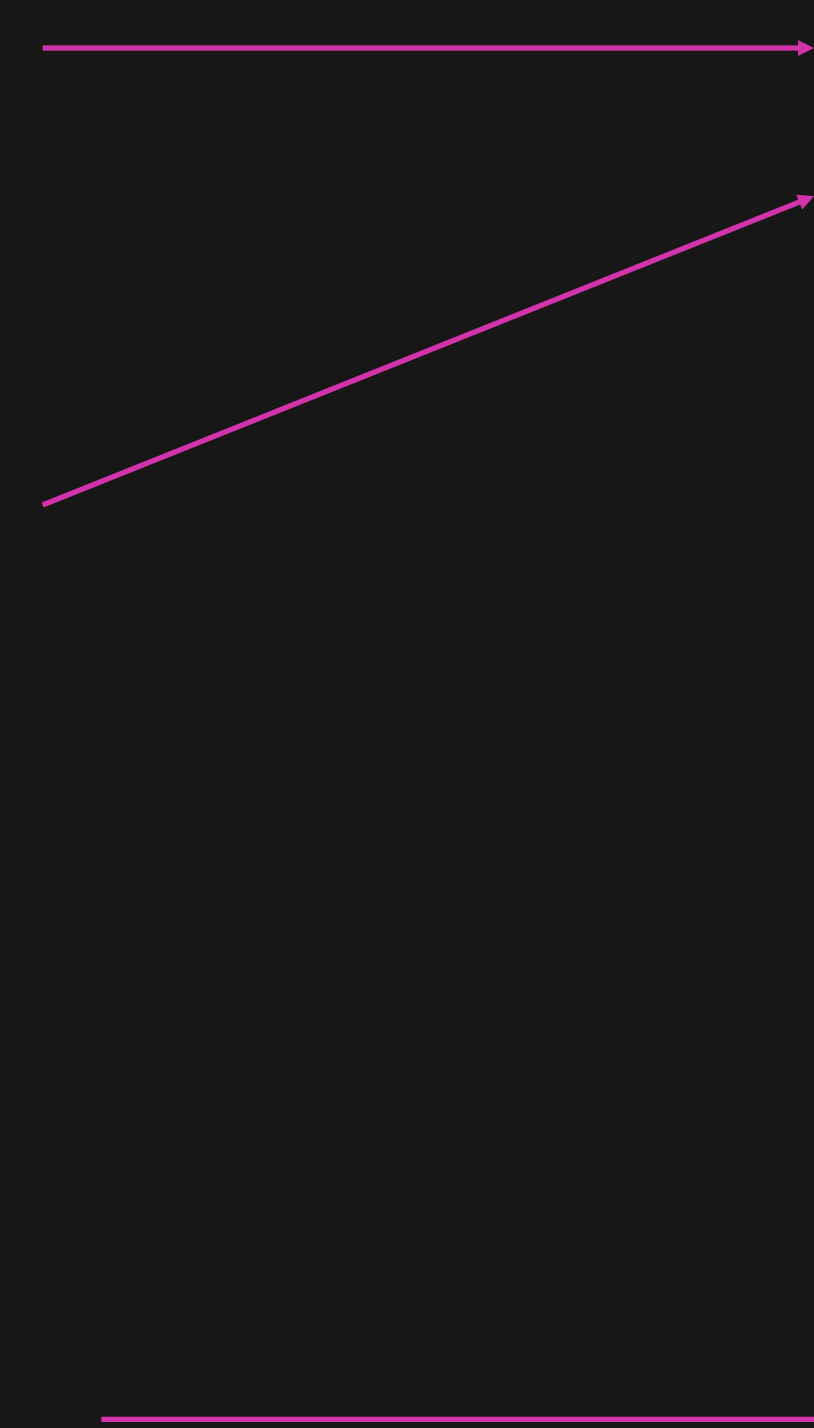


0. Feature extraction from event data.



1. Map Into user time-series.

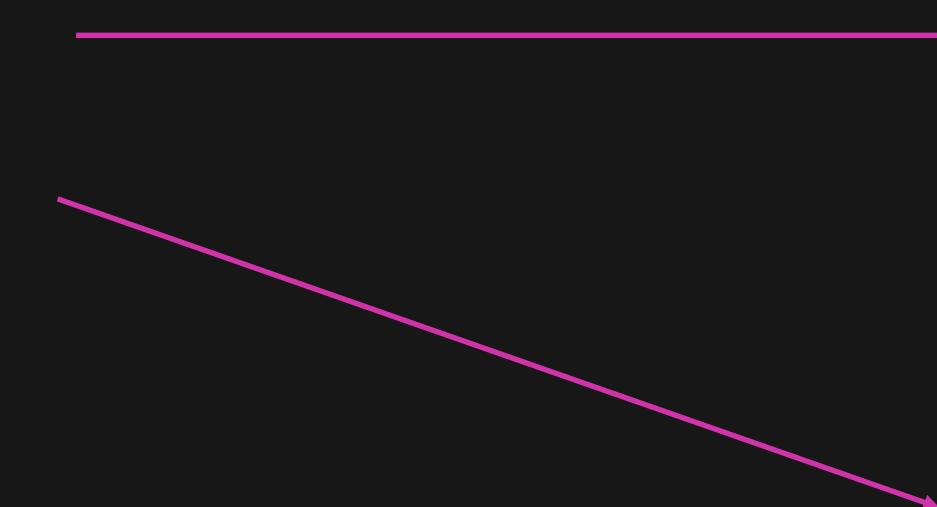
2. Cluster peer groups.



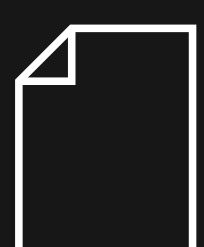
3. Predict user activity. Update models.

4. Identify Anomalies.

5. Attempt to explain statistically.



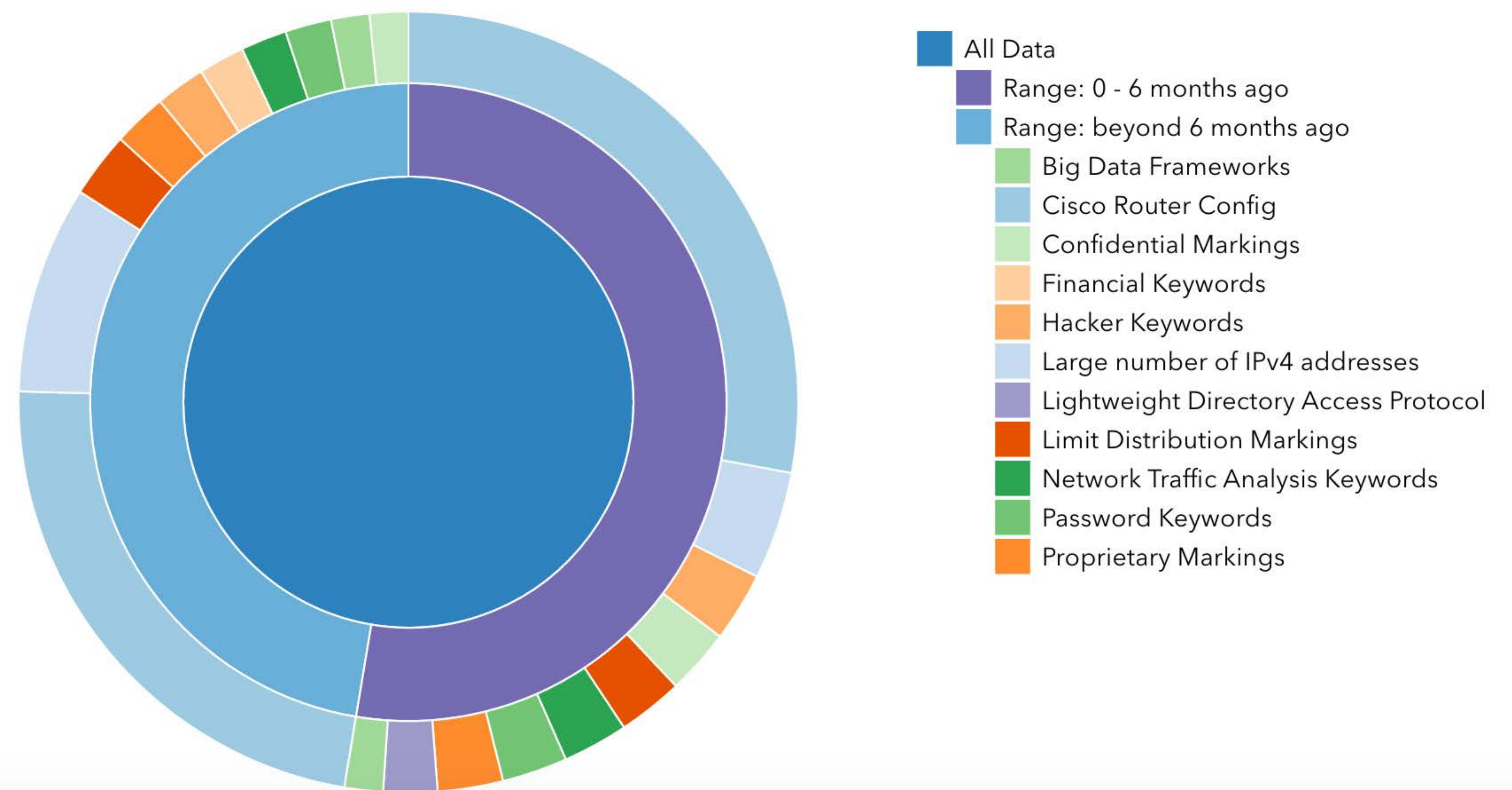
7. Alert and narrative explanation created.

 Normal accesses

Amazon Macie Content Classification

- PII and personal data
- Source code
- SSL certificates, private keys
- iOS and Android app signing keys
- Database backups
- OAuth and Cloud SAAS API Keys

Amazon S3 Overview by DLP Theme - minRisk: (7)



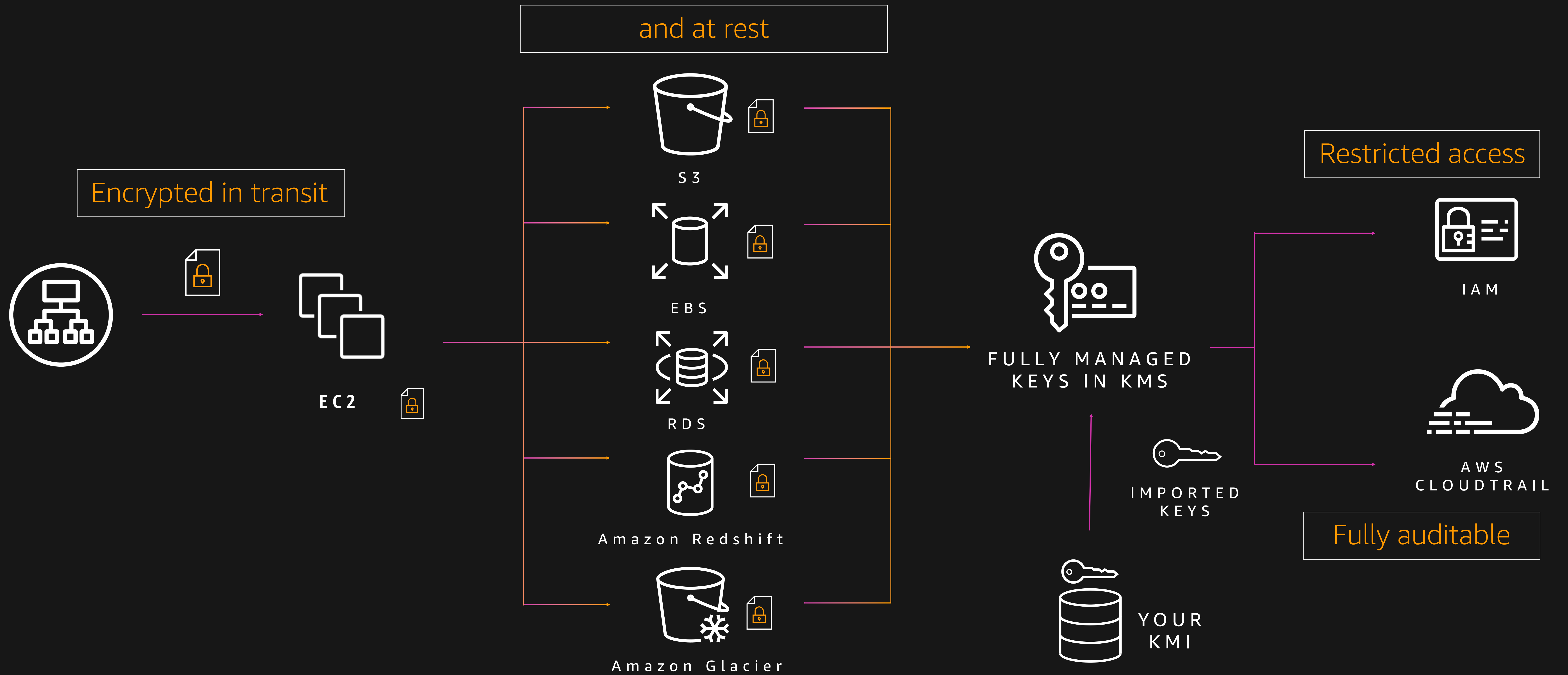
Machine Learning for Protection

For PII-types like names, addresses, user names and passwords, a regex-based approach isn't possible

Dance
like no one
is watching

**Encrypt like
everyone is**

Ubiquitous encryption



Making TLS work better in your apps



“signal to noise”

- A TLS library designed by AWS to help your developers implement transport security with faster performance
- Avoids implementing rarely used TLS options and extensions; ~6,000 lines of code

<https://github.com/awslabs/s2n>



AWS Certificate Manager (ACM)

Provision trusted SSL/TLS certificates from AWS for use with AWS resources:

- Elastic Load Balancing
- Amazon CloudFront distributions

AWS handles the muck

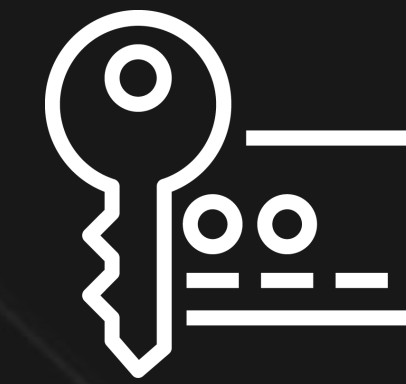
- Key pair and CSR generation
- Managed renewal and deployment

Domain validation (DV) through email

Available through AWS Management Console, AWS Command Line Interface (AWS CLI), or API

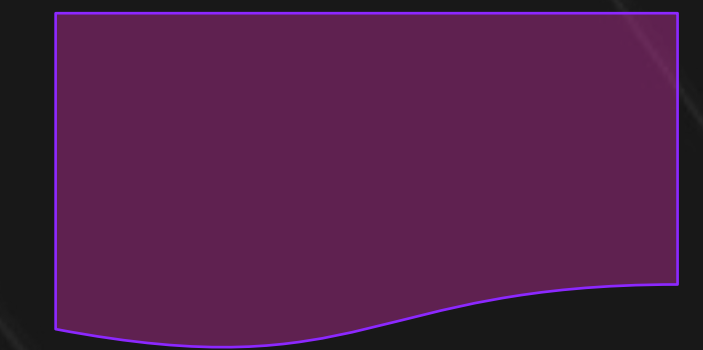
Bring your own key

Create customer master key (CMK) container



KMS

Creates



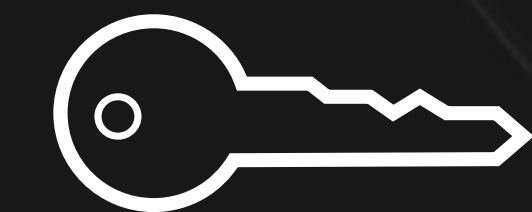
EMPTY CMK CONTAINER WITH UNIQUE KEY ID

Download a public wrapping key



KMS

Downloads



RSA PUBLIC KEY

Export your key material encrypted under the public wrapping key



YOUR KEY MANAGEMENT INFRASTRUCTURE

Exports



YOUR 256-BIT KEY MATERIAL ENCRYPTED UNDER KMS PUBLIC KEY

Import encrypted key material under the KMS CMK key ID; set optional expiration period



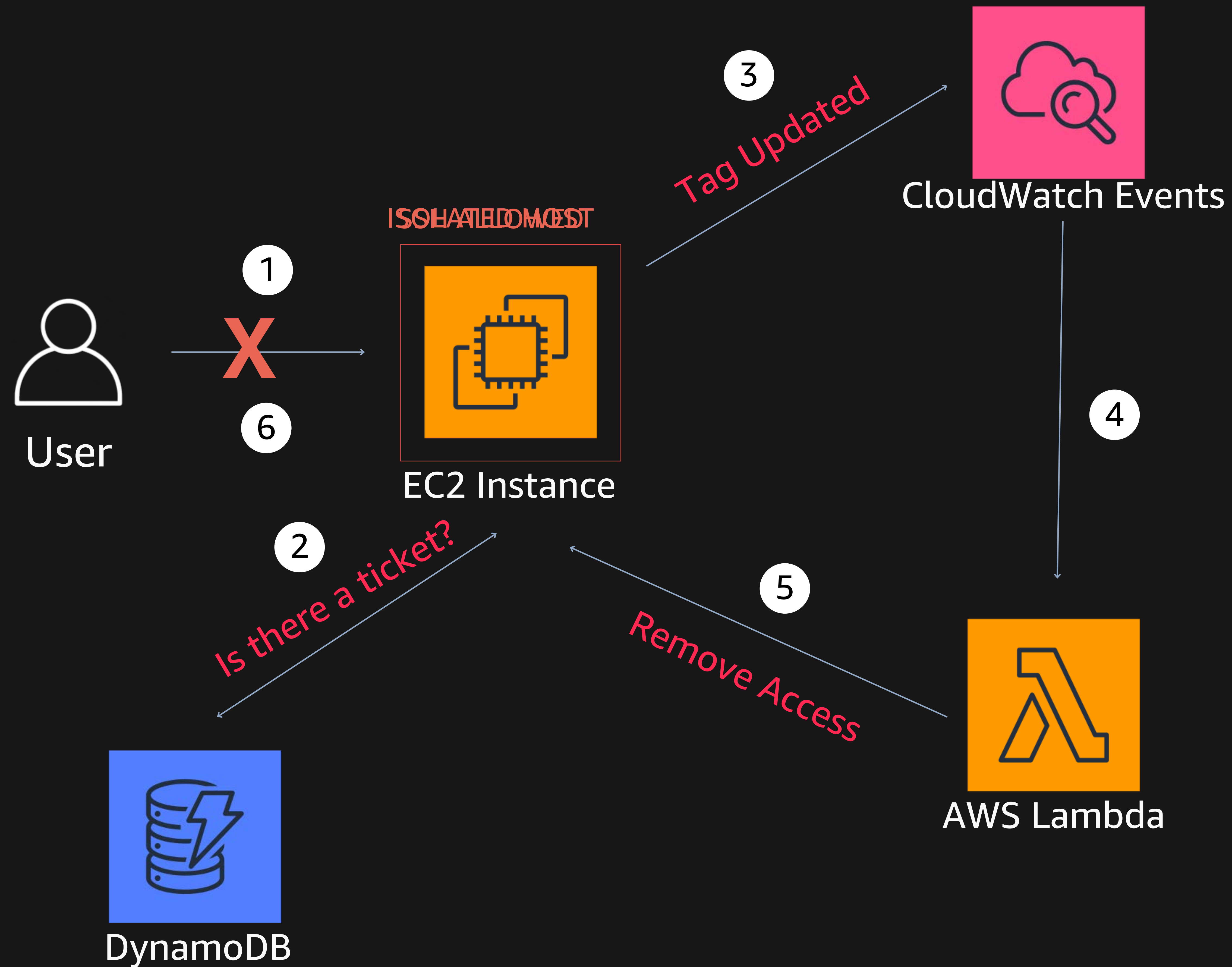
Imports



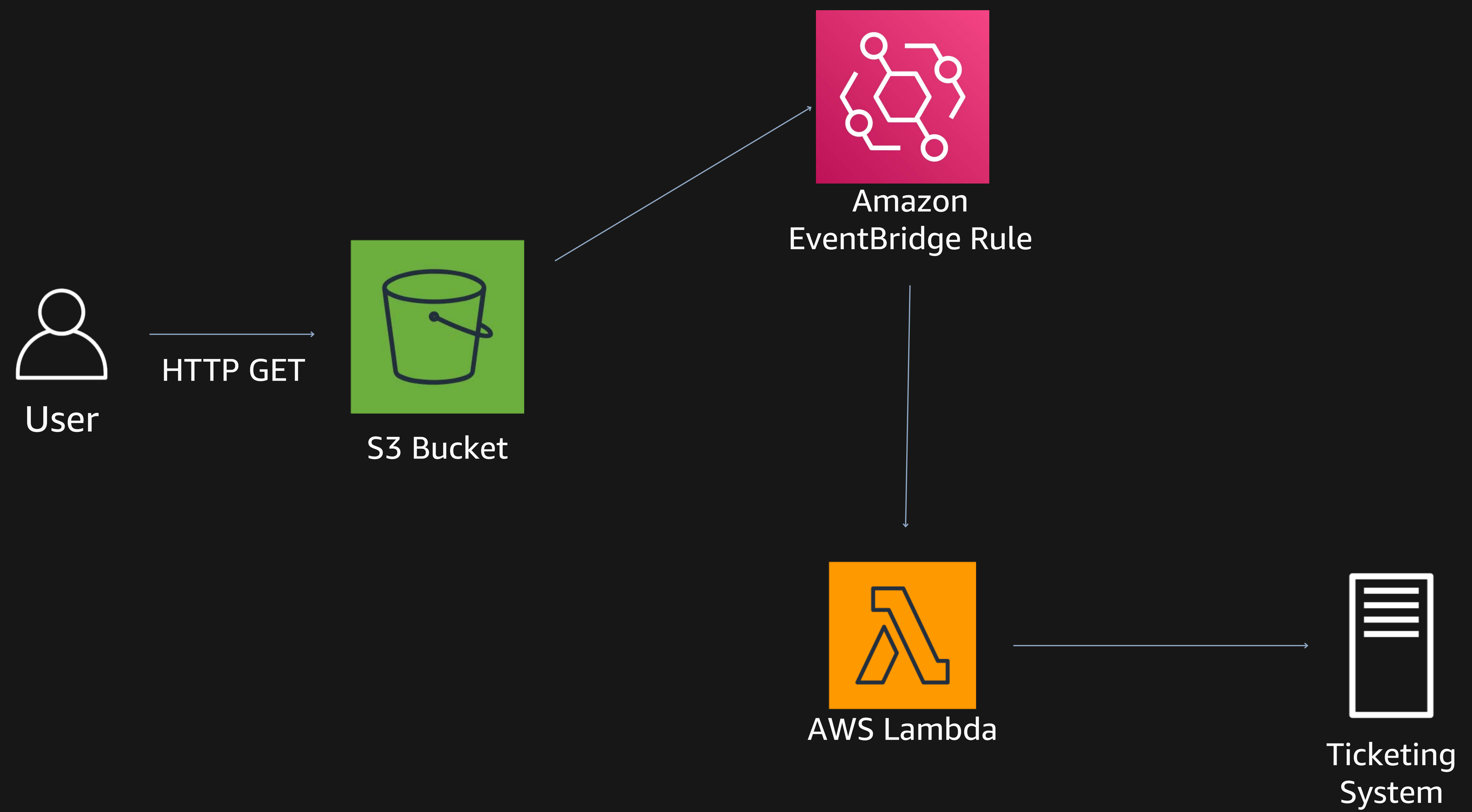
YOUR KEY MATERIAL PROTECTED IN KMS

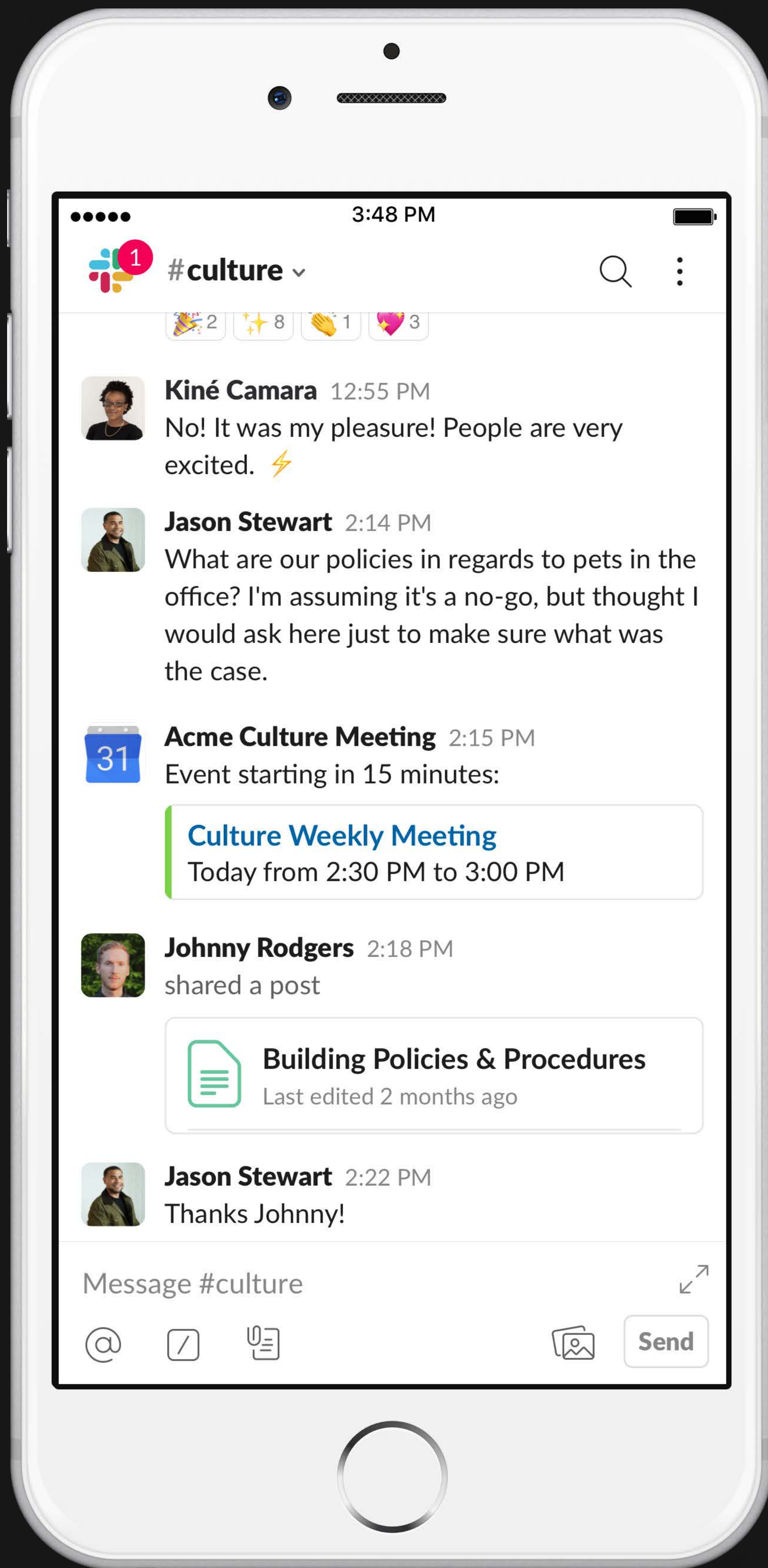
So... those examples?

Example – Auto isolation – Host meets Cloud



Example – Raise Ticket based on activity





"The fact that we can rely on the AWS security posture to boost our own security is really important for our business. AWS does a much better job at security than we could ever do running a cage in a data center."

Richard Crowley Director of Operations, Slack



“I have been in IT for 25 years, responsible for many data centers. I have to say that I never had such a secure data center as I have today with AWS”

Vladimir Klein, CIO



 **MONETA** | MONEY BANK



Making the most of Data

To Become a Leader, Data is Your Differentiator

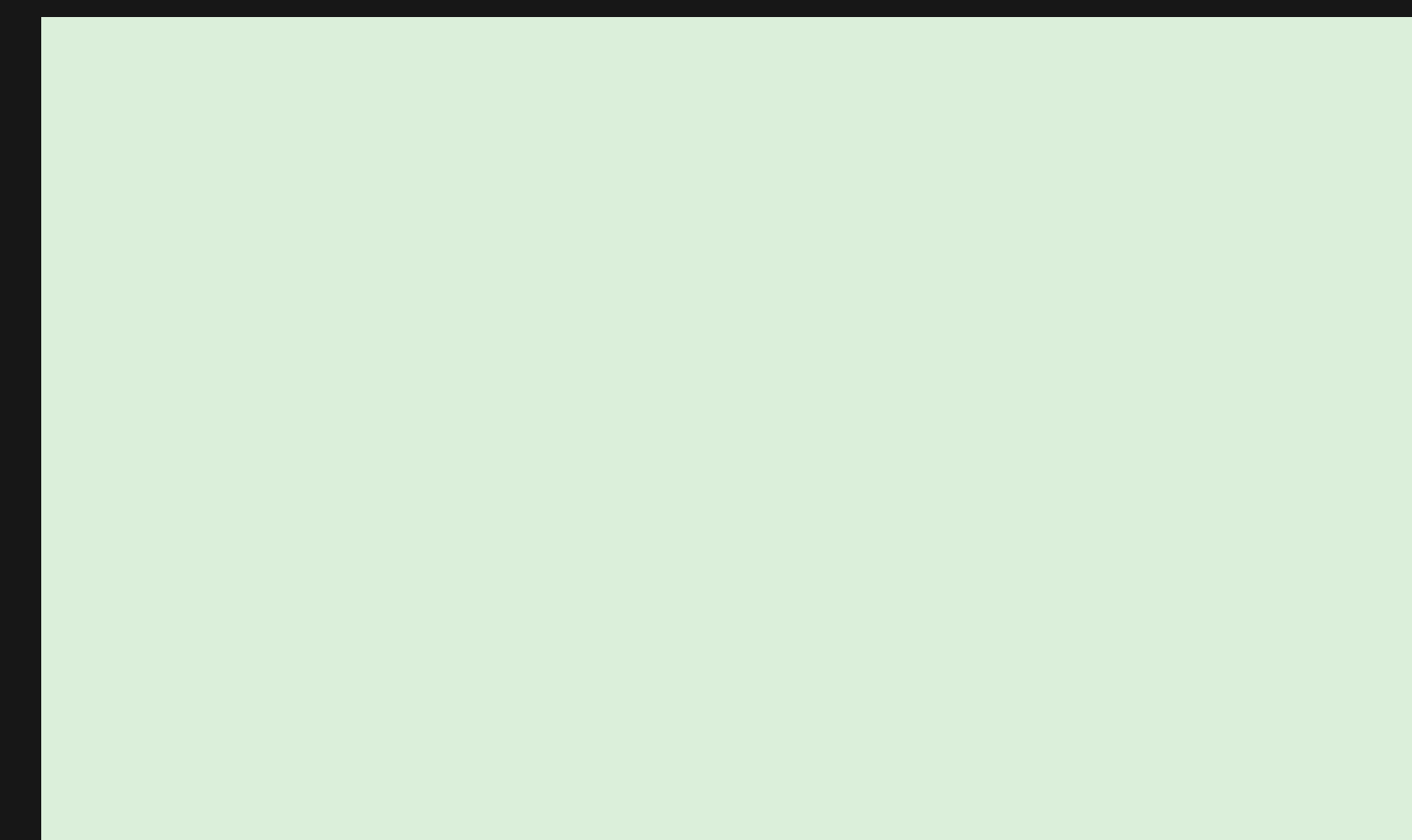
Organic revenue growth

24%



Leaders

15%



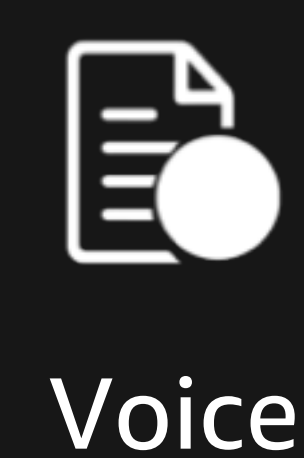
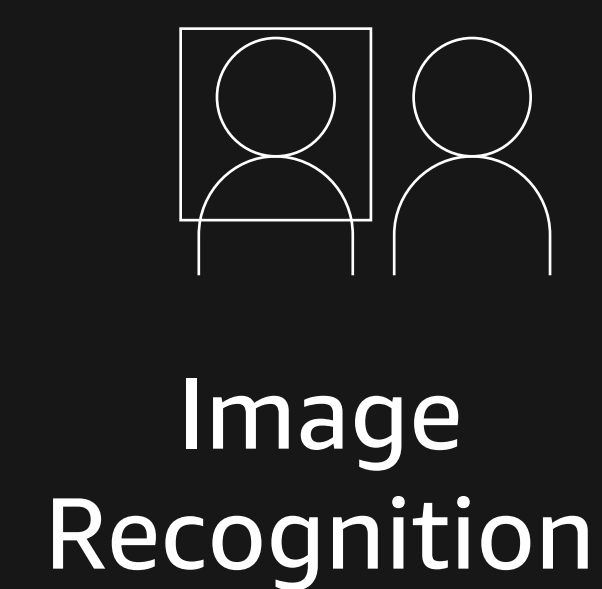
Followers

Organizations that successfully generate business value from their data, will outperform their peers. An Aberdeen survey saw organizations who implemented a Data Lake outperforming similar companies by 9% in organic revenue growth.*

*Aberdeen: Angling for Insight in Today's Data Lake, Michael Lock, SVP Analytics and Business Intelligence

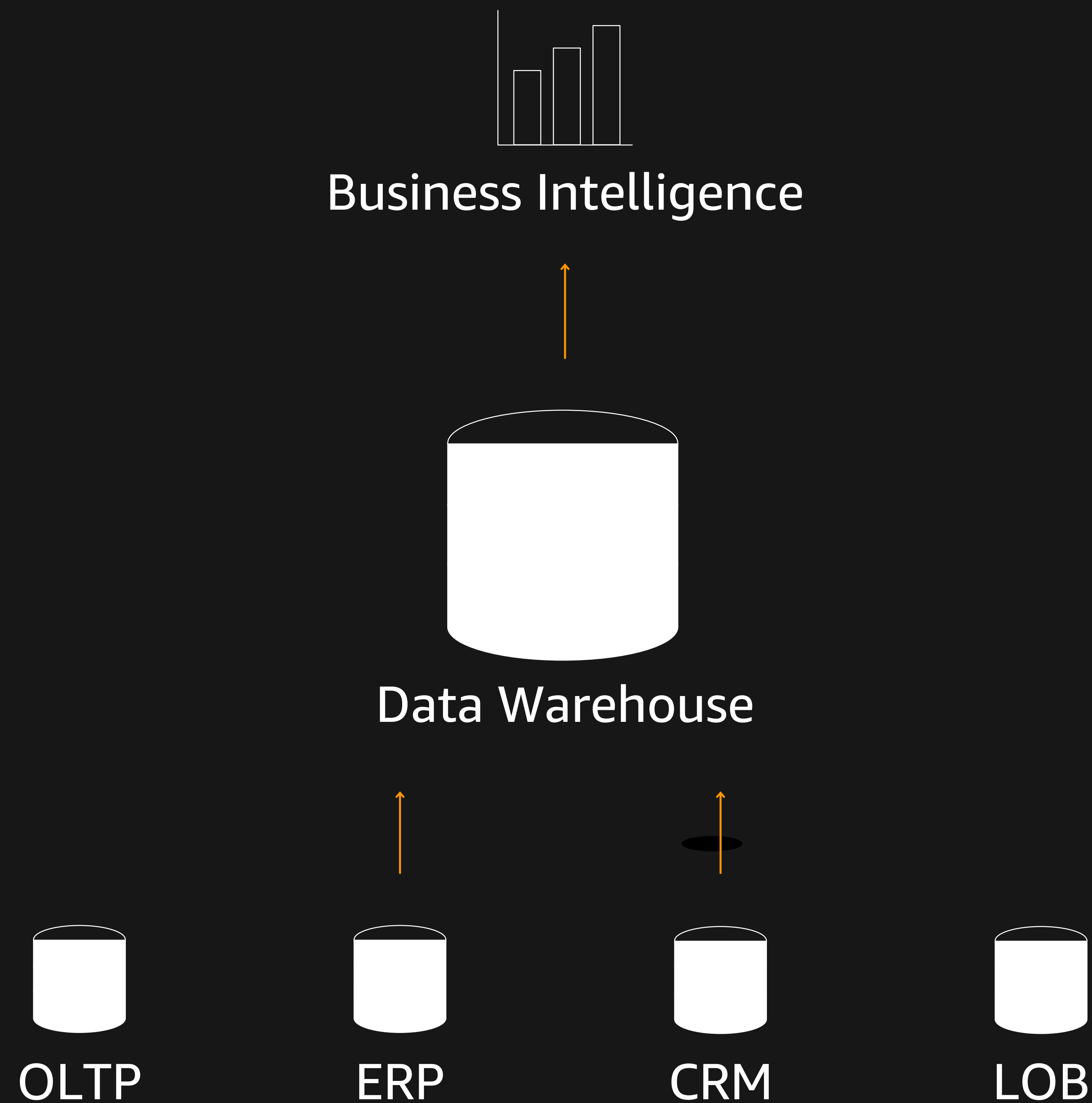
For Data to Be a Differentiator, Customers Need to Be Able to...

New types of analytics



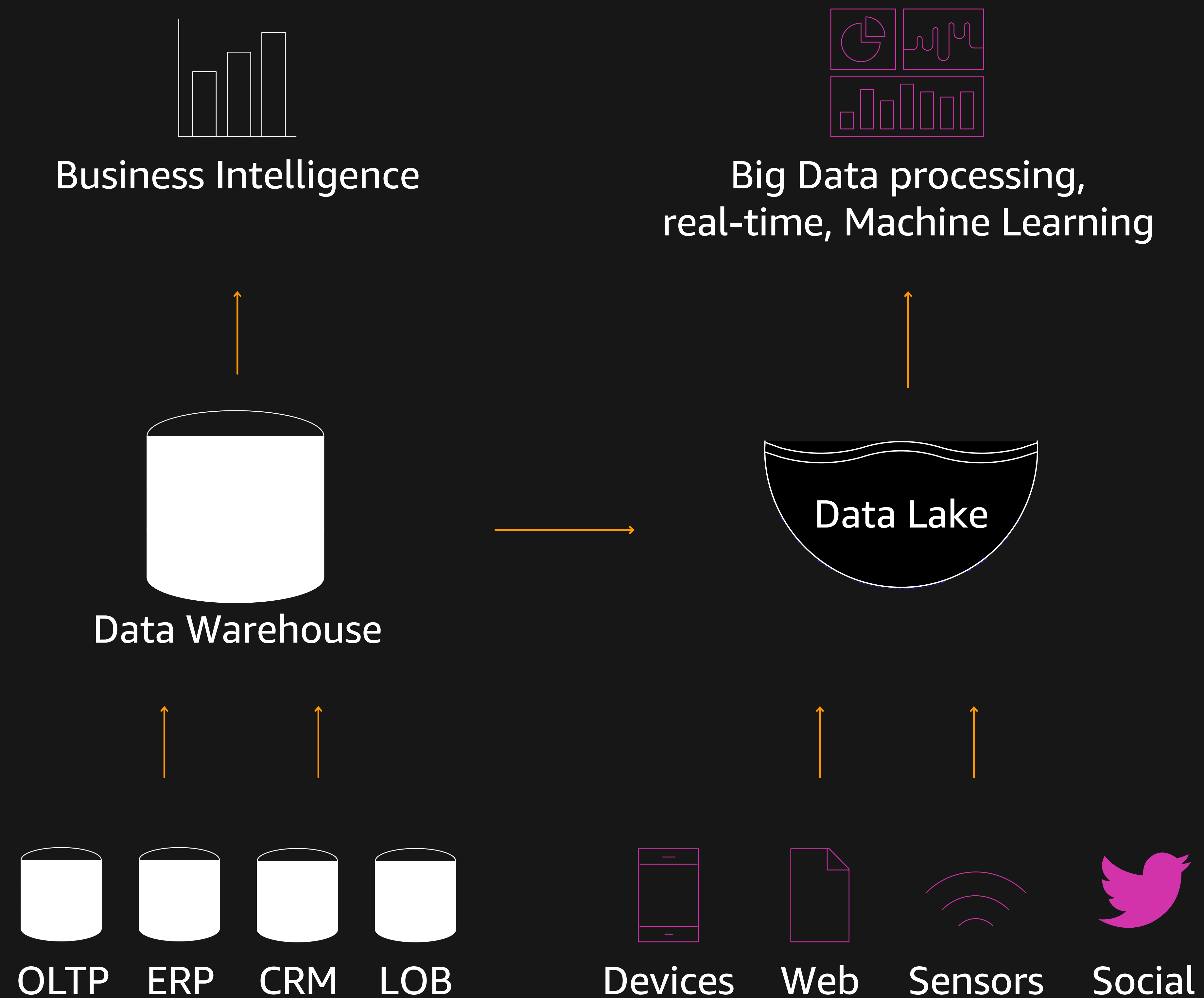
- Capture and store new non-relational data at PB-EB scale in real time
- New type of analytics that go beyond batch reporting to incorporate real-time, predictive, voice, and image recognition
- Democratize access to data in a secure and governed way

Traditionally, Analytics Used to Look Like This



- Relational data
- TBs–PBs scale
- Schema defined prior to data load
- Operational reporting and ad hoc
- Large initial CAPEX + \$10K–\$50K/TB/Year

Data Lakes Extend the Traditional Approach



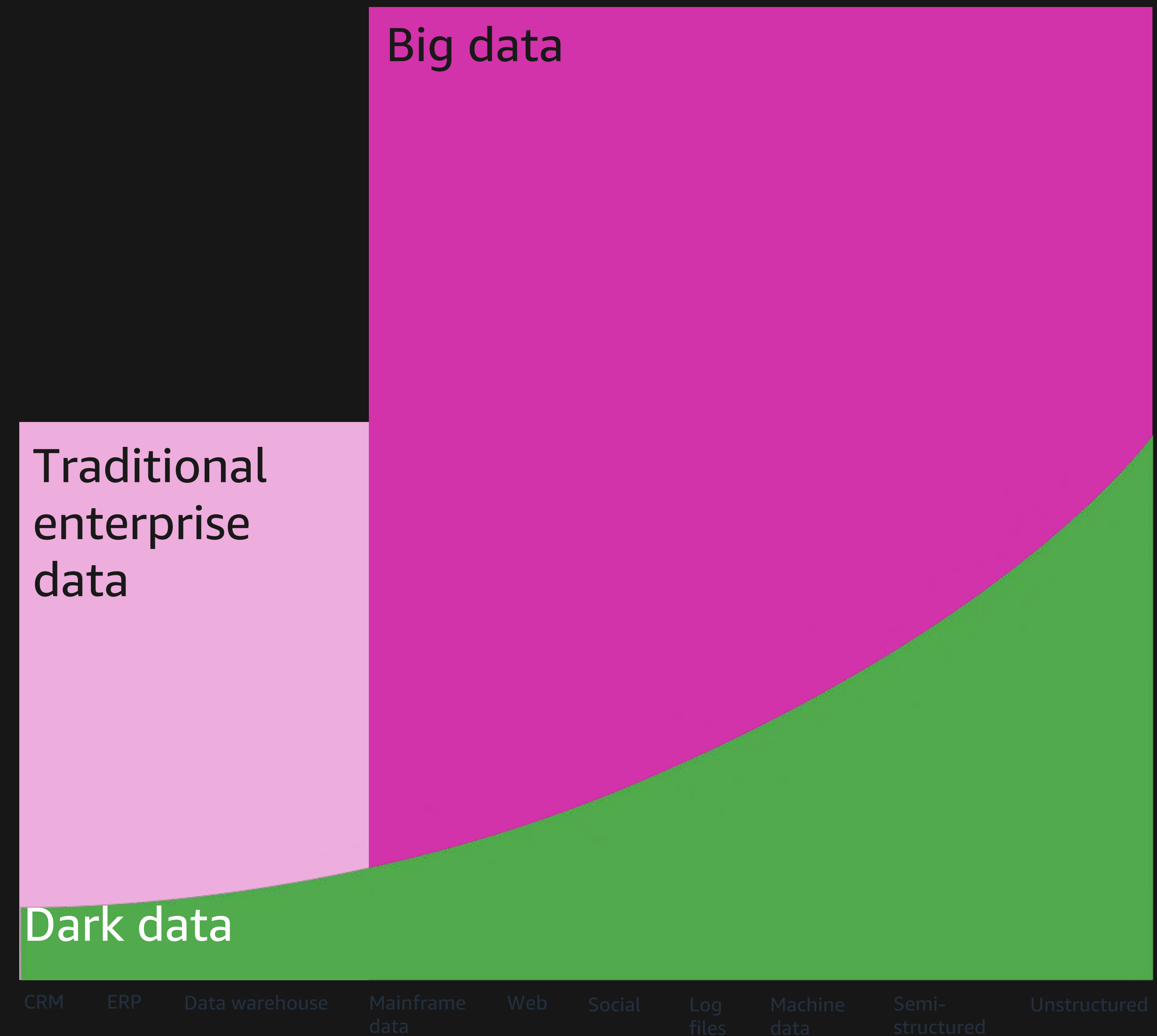
- Relational and non-relational data
- TBs–EBs scale
- Diverse analytical engines
- Low-cost storage & analytics

A data lake is a **centralized repository that allows you to store all your structured and unstructured** data at any scale

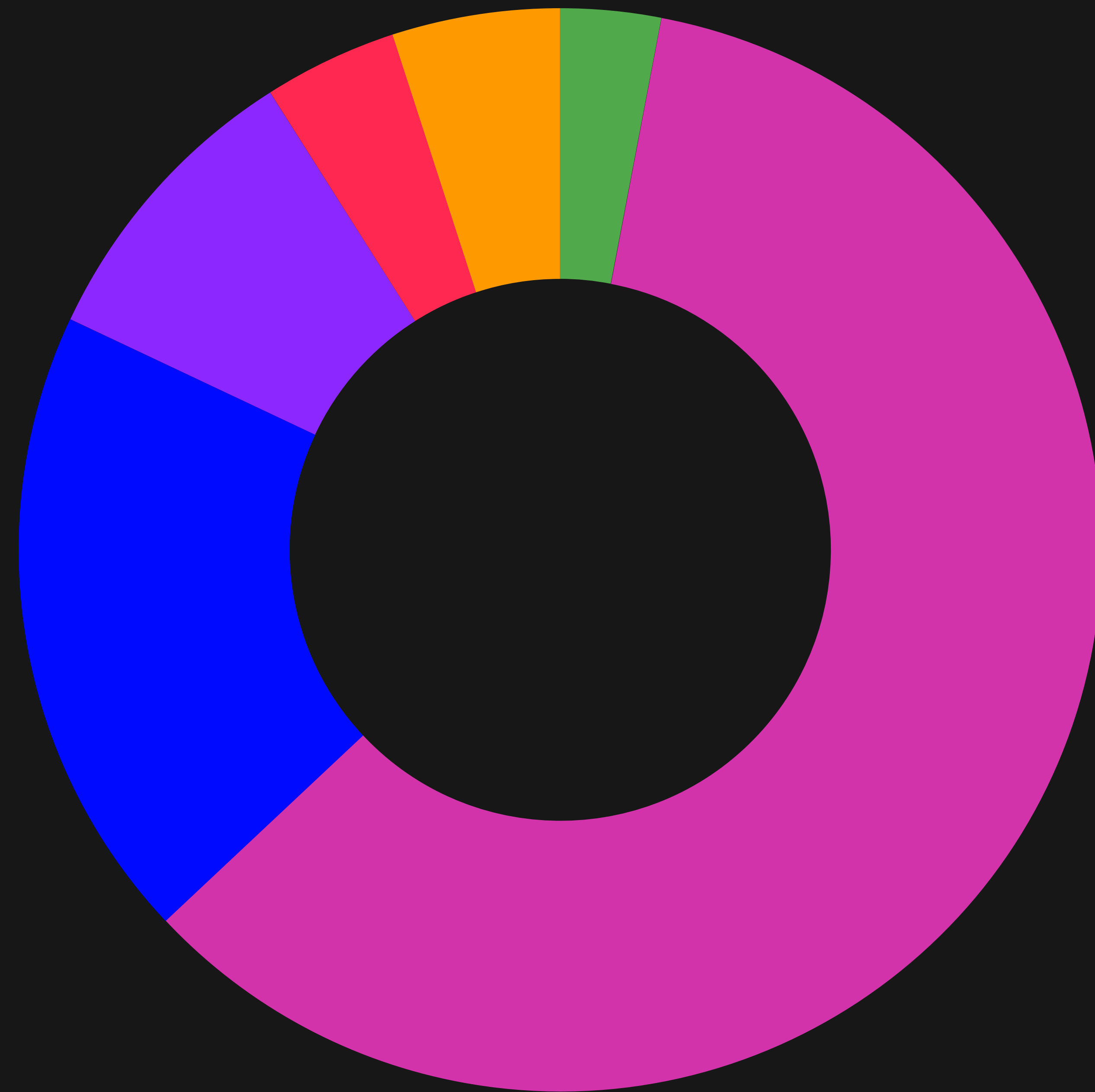
Storing is Not Enough, Data Needs to Be Discoverable

“ Dark data are the information assets organizations collect, process, and store during regular business activities, but generally fail to use for other purposes (for example, analytics, business relationships and direct monetizing). ”

Gartner IT Glossary, 2018
<https://www.gartner.com/it-glossary/dark-data>



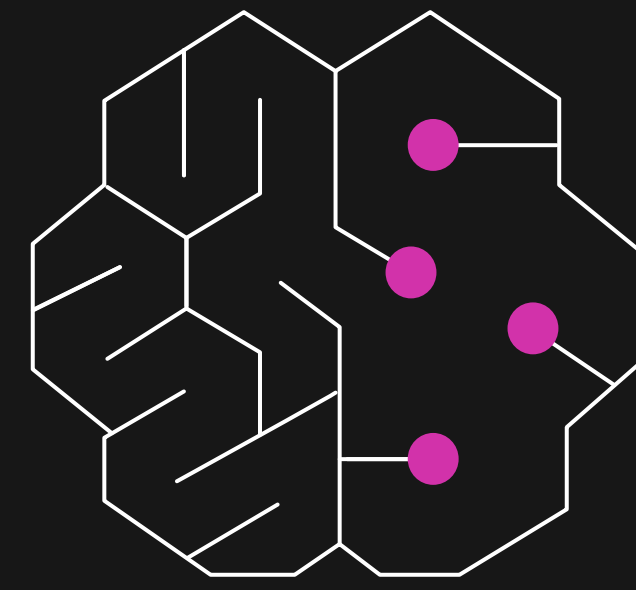
Data Preparation Accounts for ~80% of the Work



- Building training sets
- Cleaning and organizing data
- Collecting data sets
- Mining data for patterns
- Refining algorithms
- Other

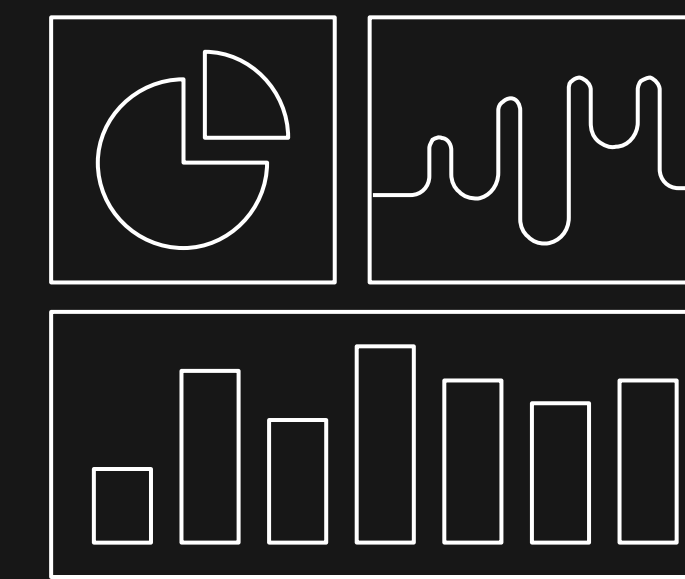
Data Lakes, Analytics, and ML Portfolio from AWS

Broadest, deepest set of analytic services



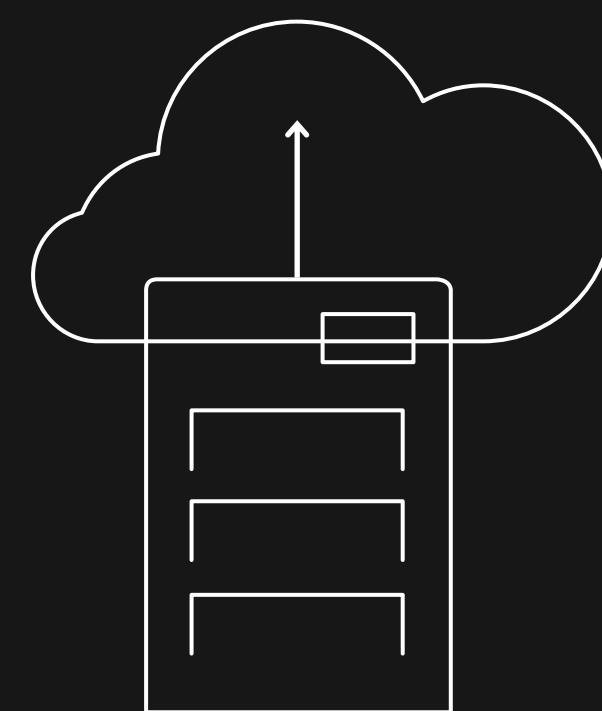
Machine Learning

- Amazon SageMaker
- AWS Deep Learning AMIs
- Amazon Rekognition
- Amazon Lex
- AWS DeepLens
- Amazon Comprehend
- Amazon Translate
- Amazon Transcribe
- Amazon Polly



Analytics

- Amazon Athena
- Amazon EMR
- Amazon Redshift
- Amazon Elasticsearch service
- Amazon Kinesis
- Amazon QuickSight



On-premises Data Movement

- AWS Direct Connect
- AWS Snowball
- AWS Snowmobile
- AWS Database Migration Service
- AWS Storage Gateway



Real-time Data Movement

- AWS IoT Core
- Amazon Kinesis Data Firehose
- Amazon Kinesis Data Streams
- Amazon Kinesis Video Streams

More **data** than ever, but how do you use it?



If I asked you...

"How important is it for your company to invest in Machine Learning and AI?"

What we often hear:

"We're not a Machine Learning company."

"They're not central to our business."

"We have more important topics to address."

Now, if I asked you...

"How important is it for your organization to build better products and deliver a better customer experience?"

We always hear:

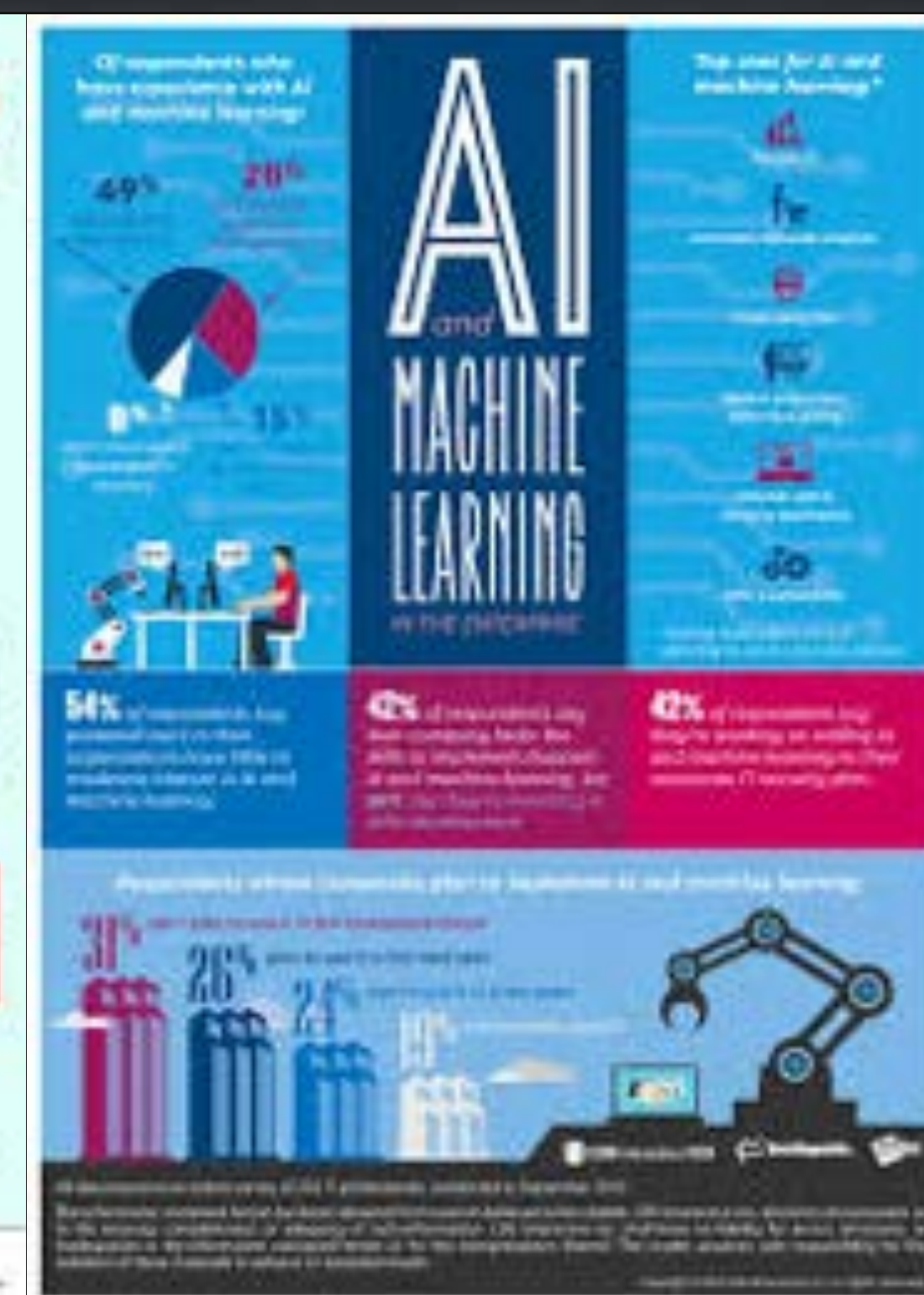
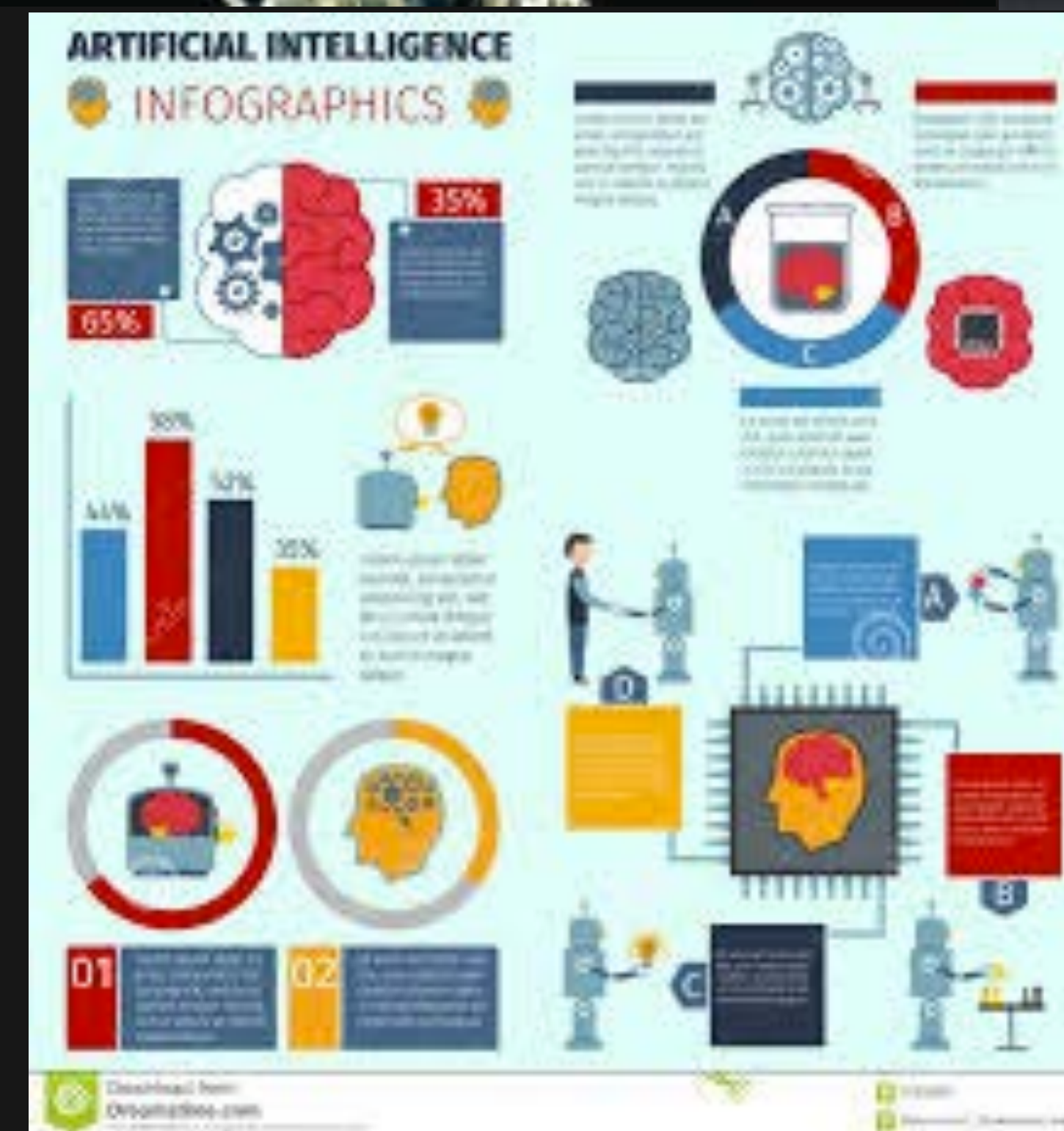
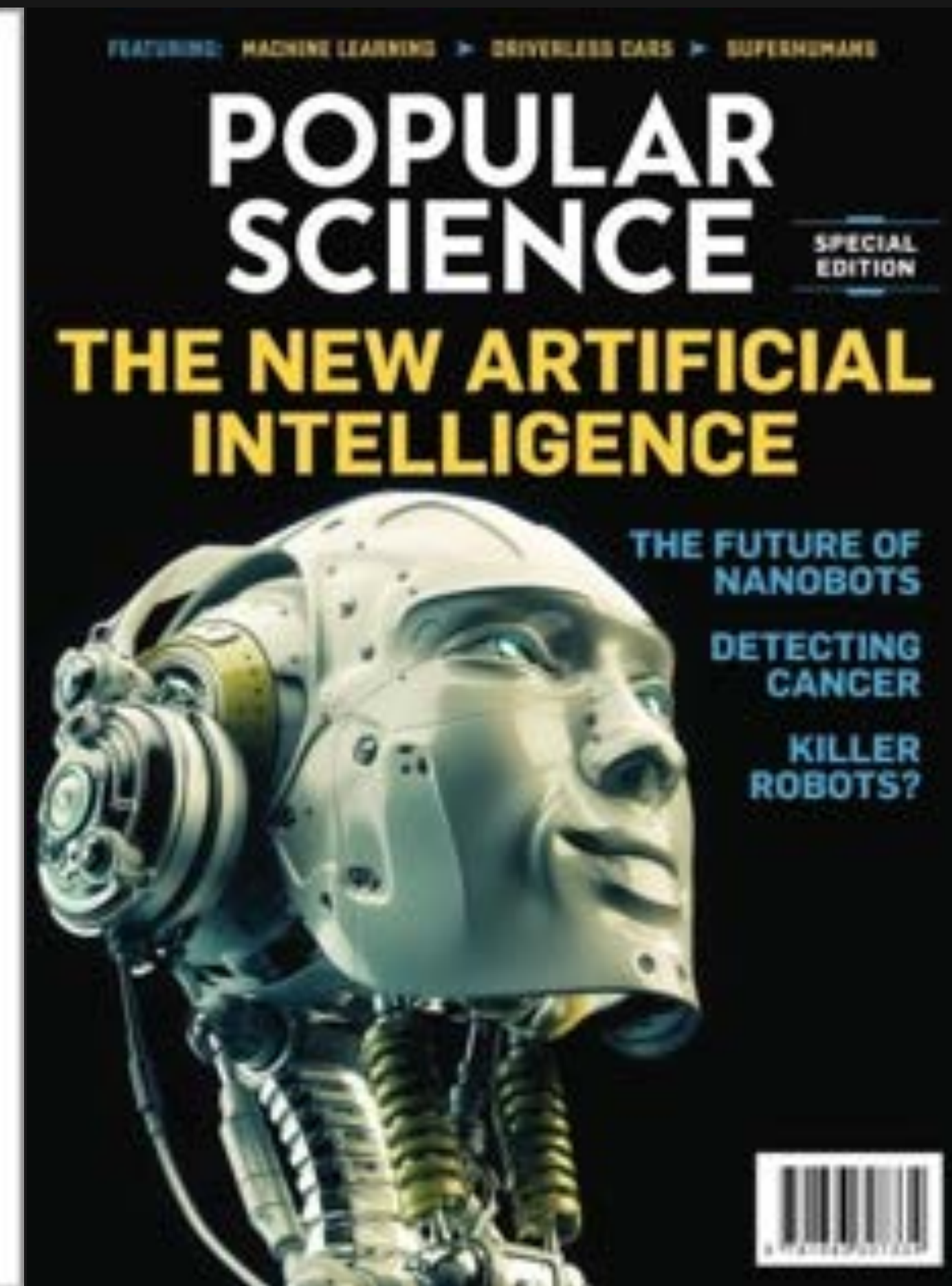
"This is our number one priority."

But isn't this the same question, really?

What is Artificial Intelligence?

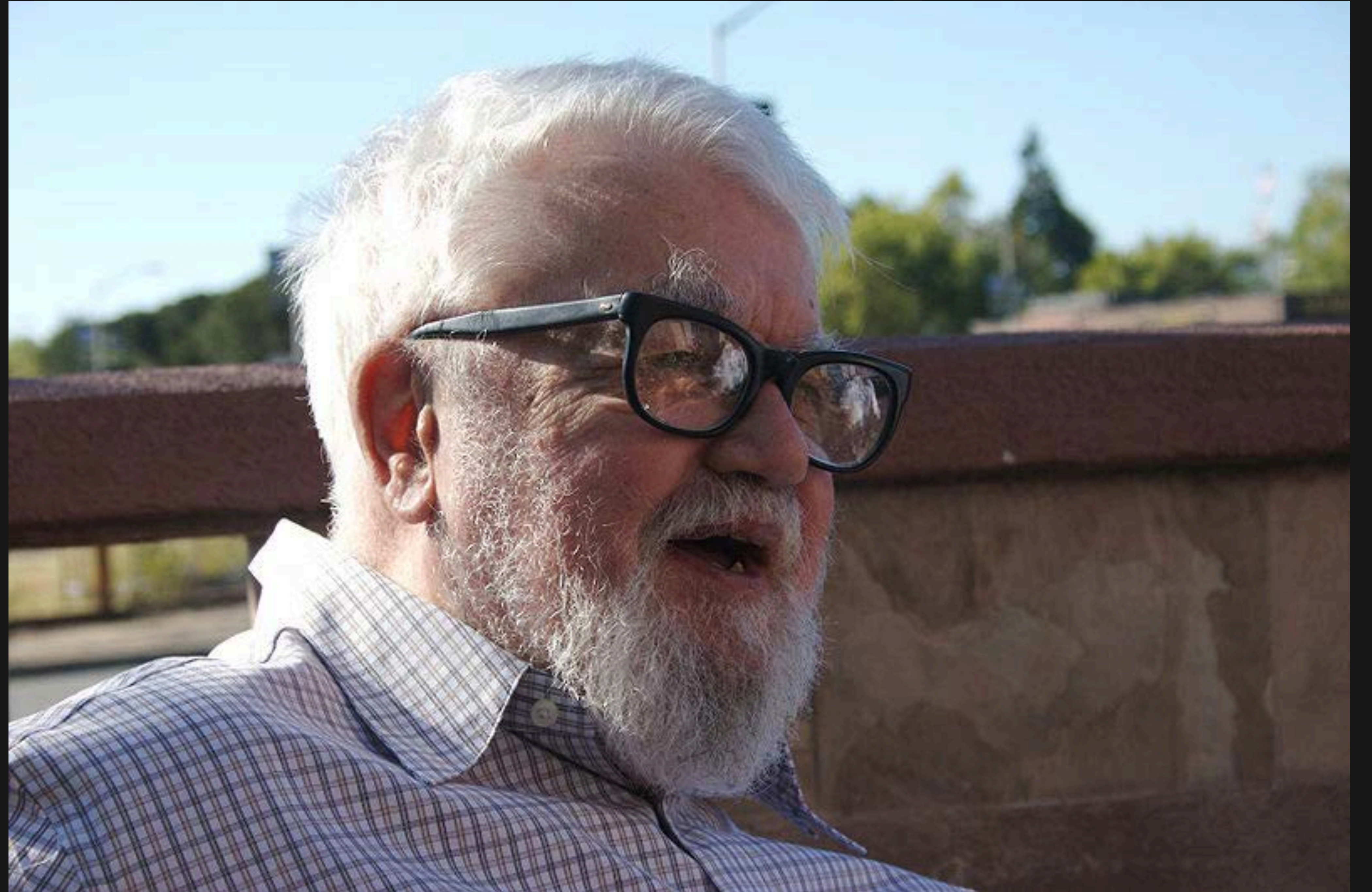
What is Machine Learning?

Myth - AI is the flavour of the month

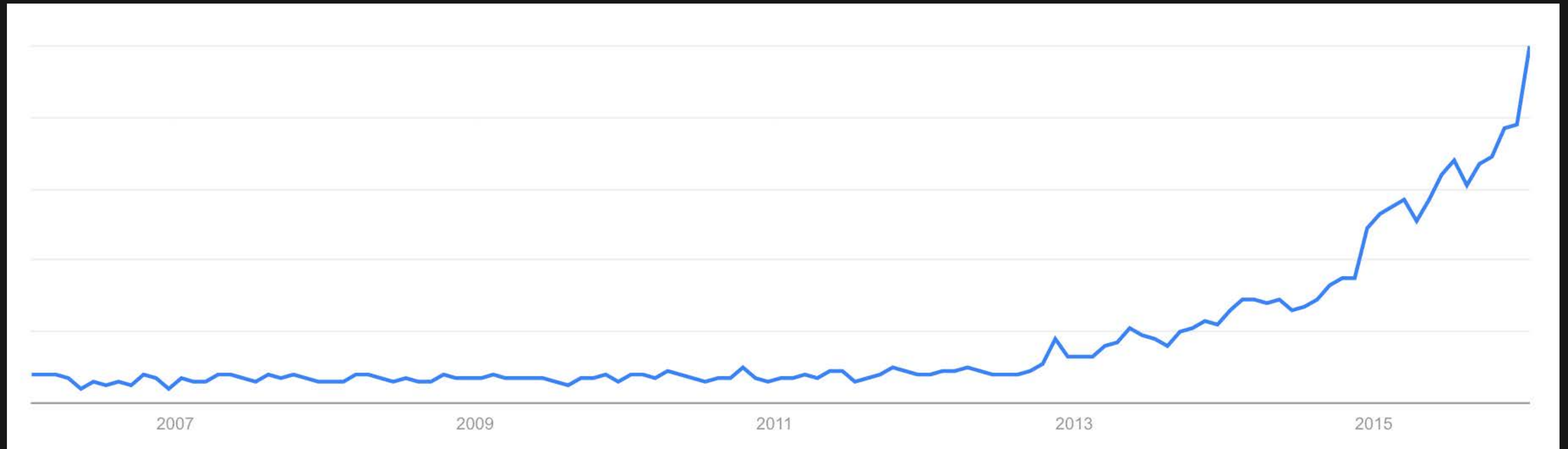


One of the "Founding Fathers" of AI

John McCarthy
1955

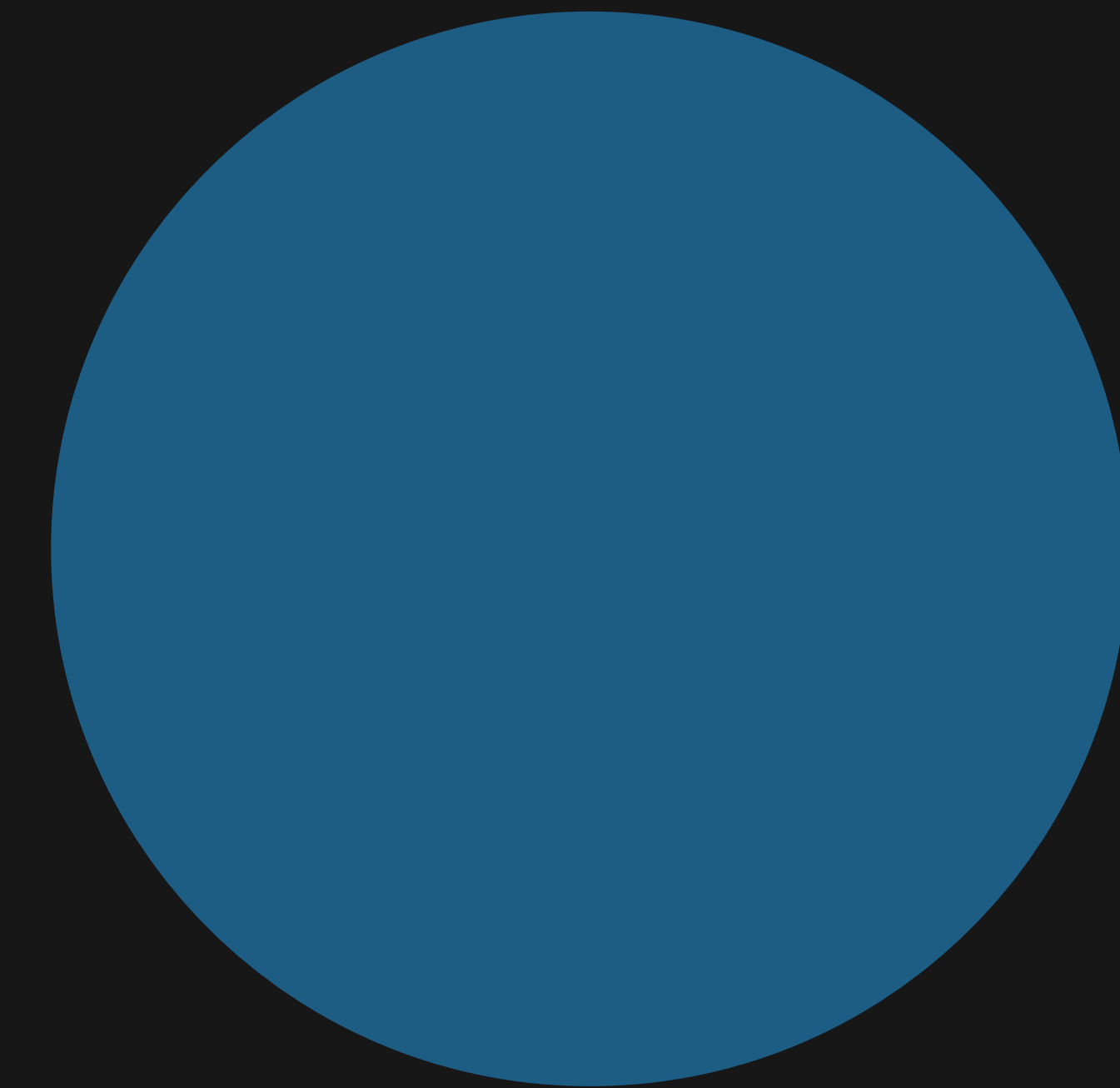


The (60+ years) rise of Artificial Intelligence

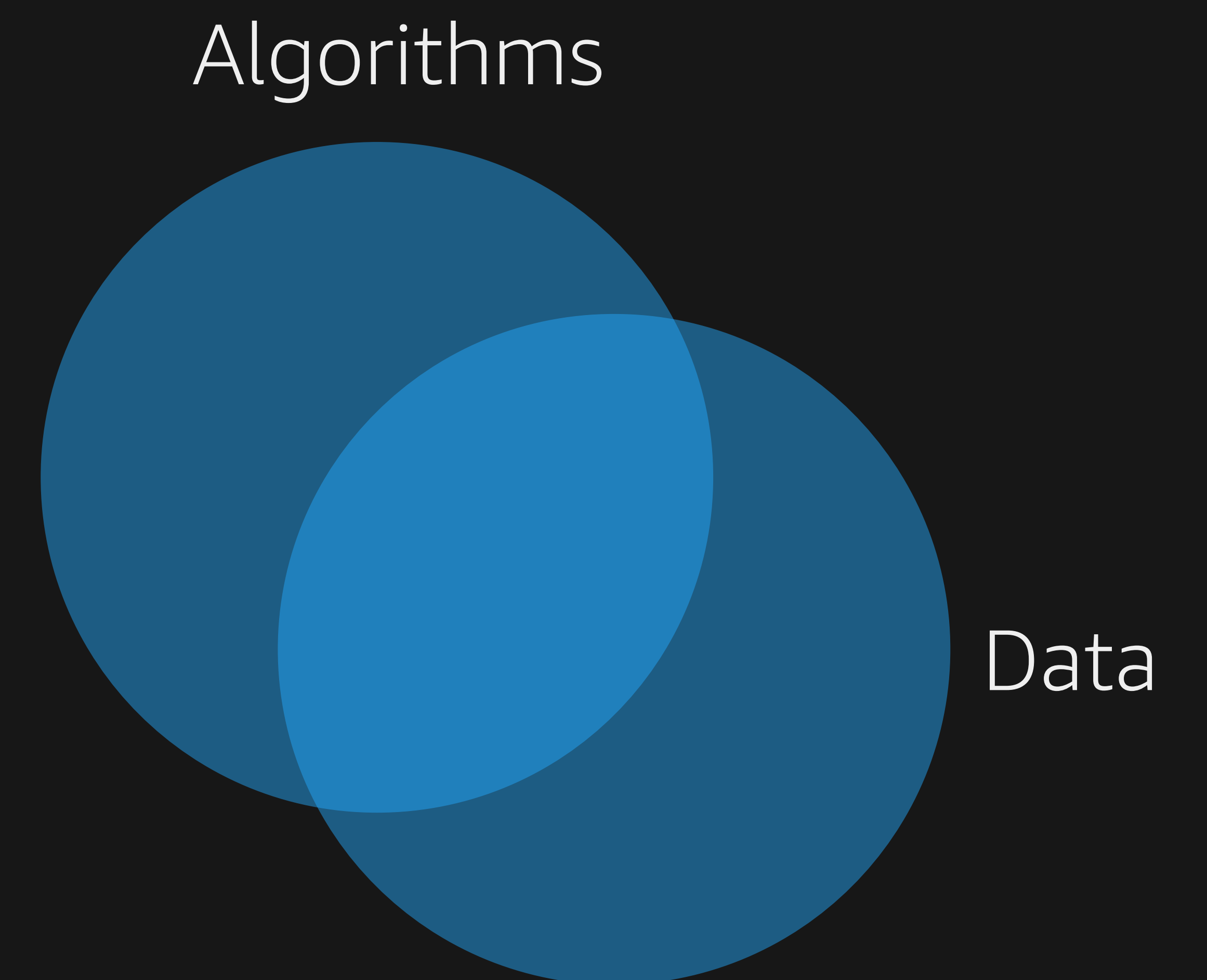


The Advent of AI

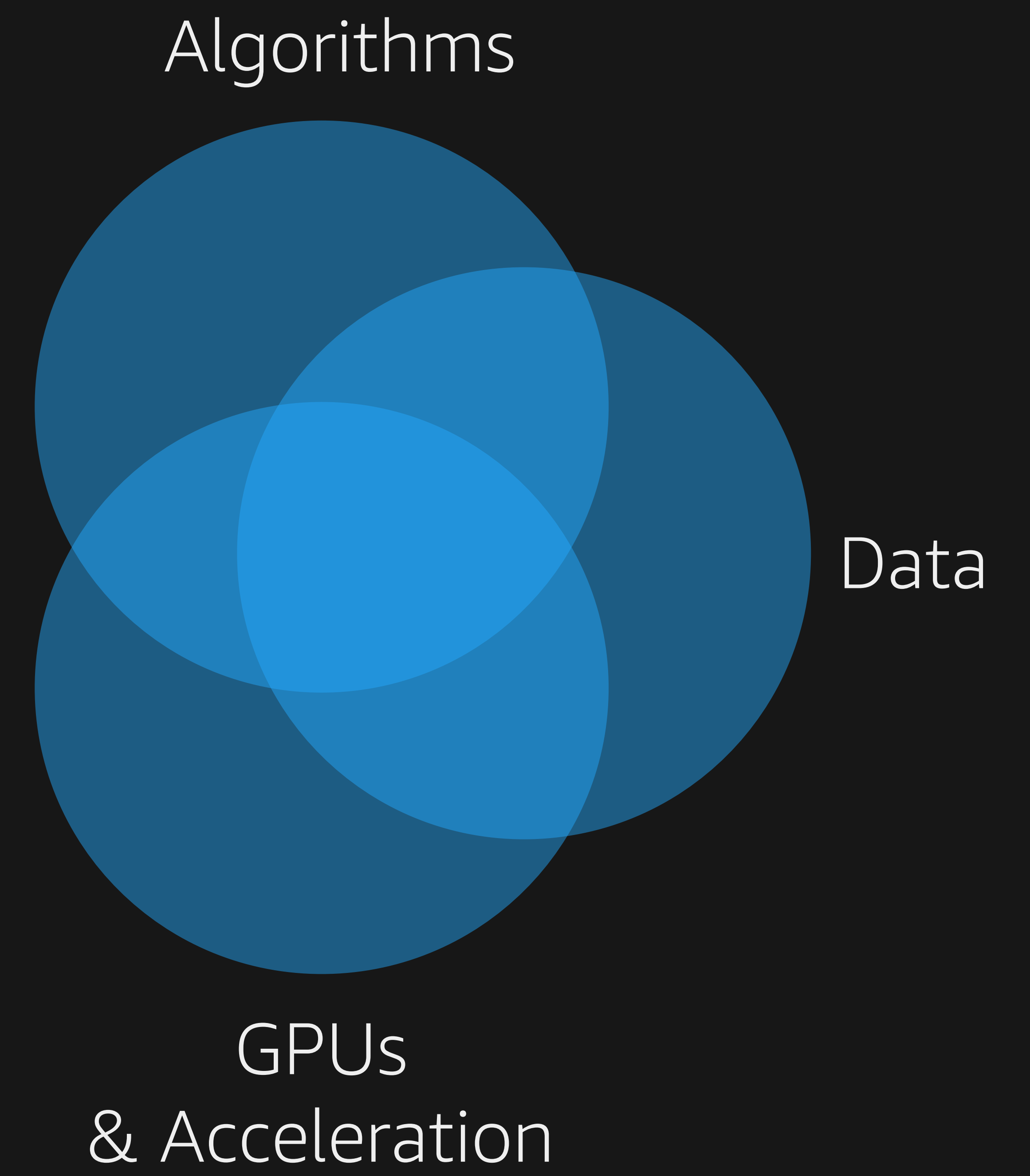
Algorithms



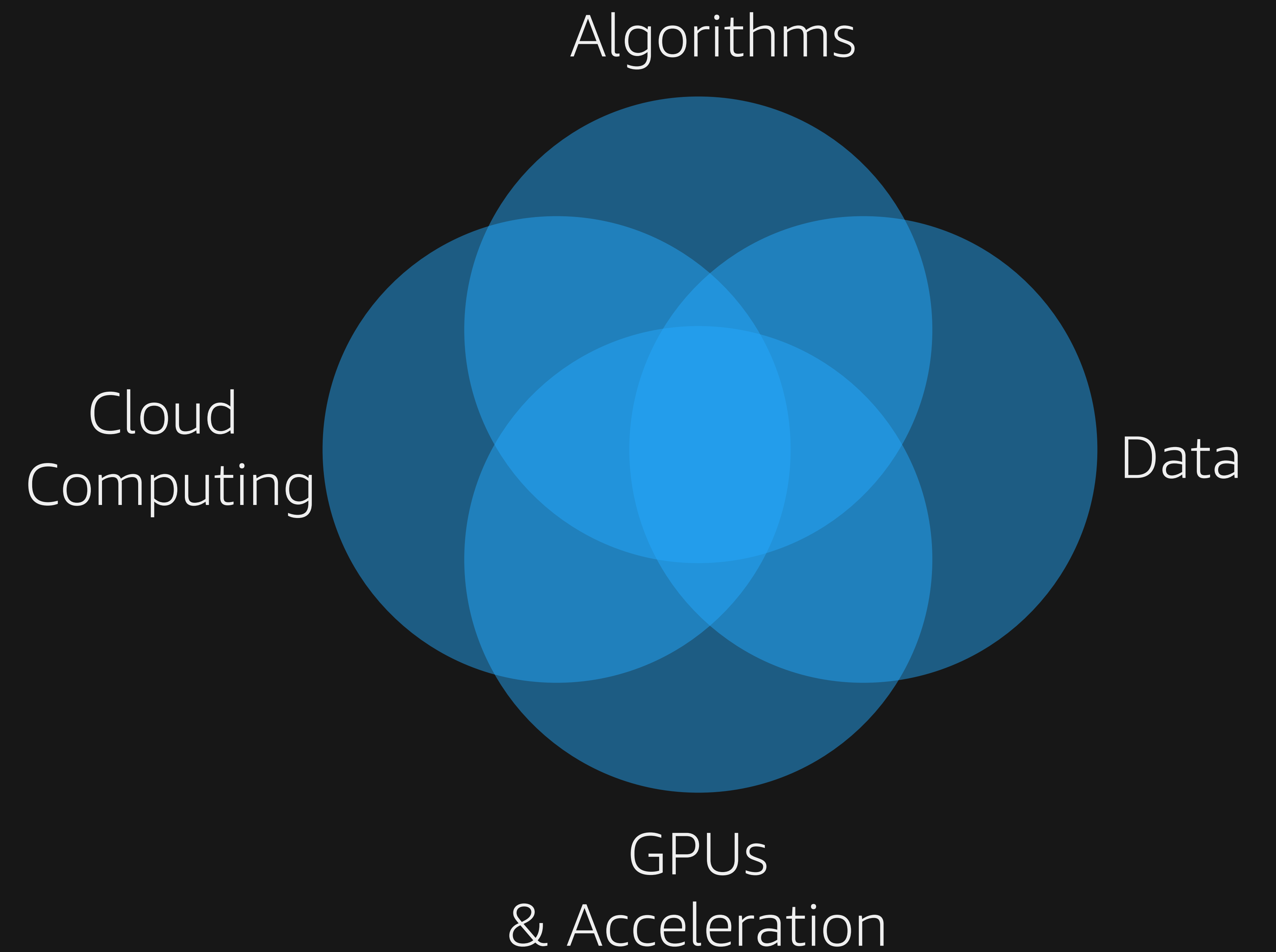
The Advent of AI

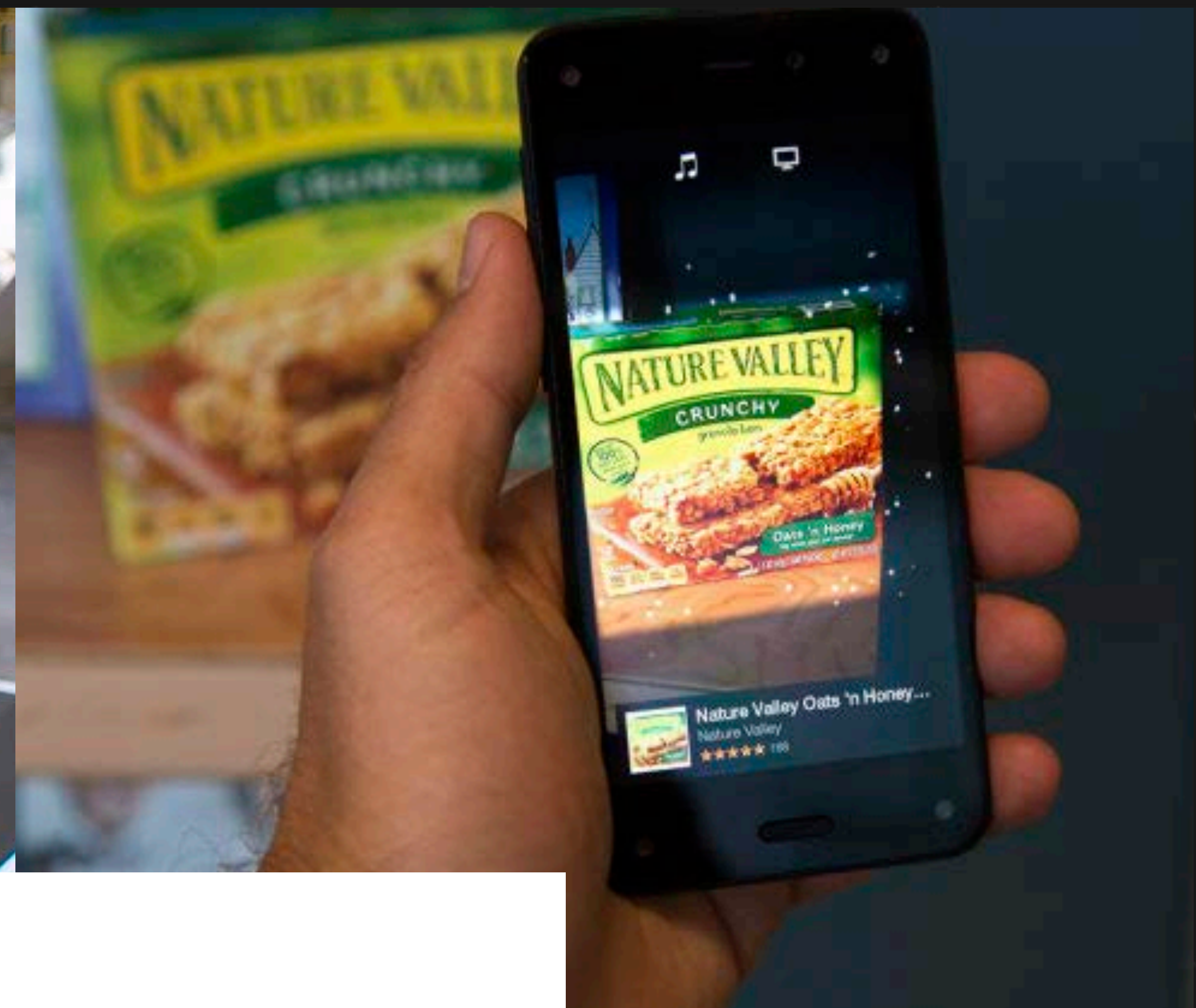
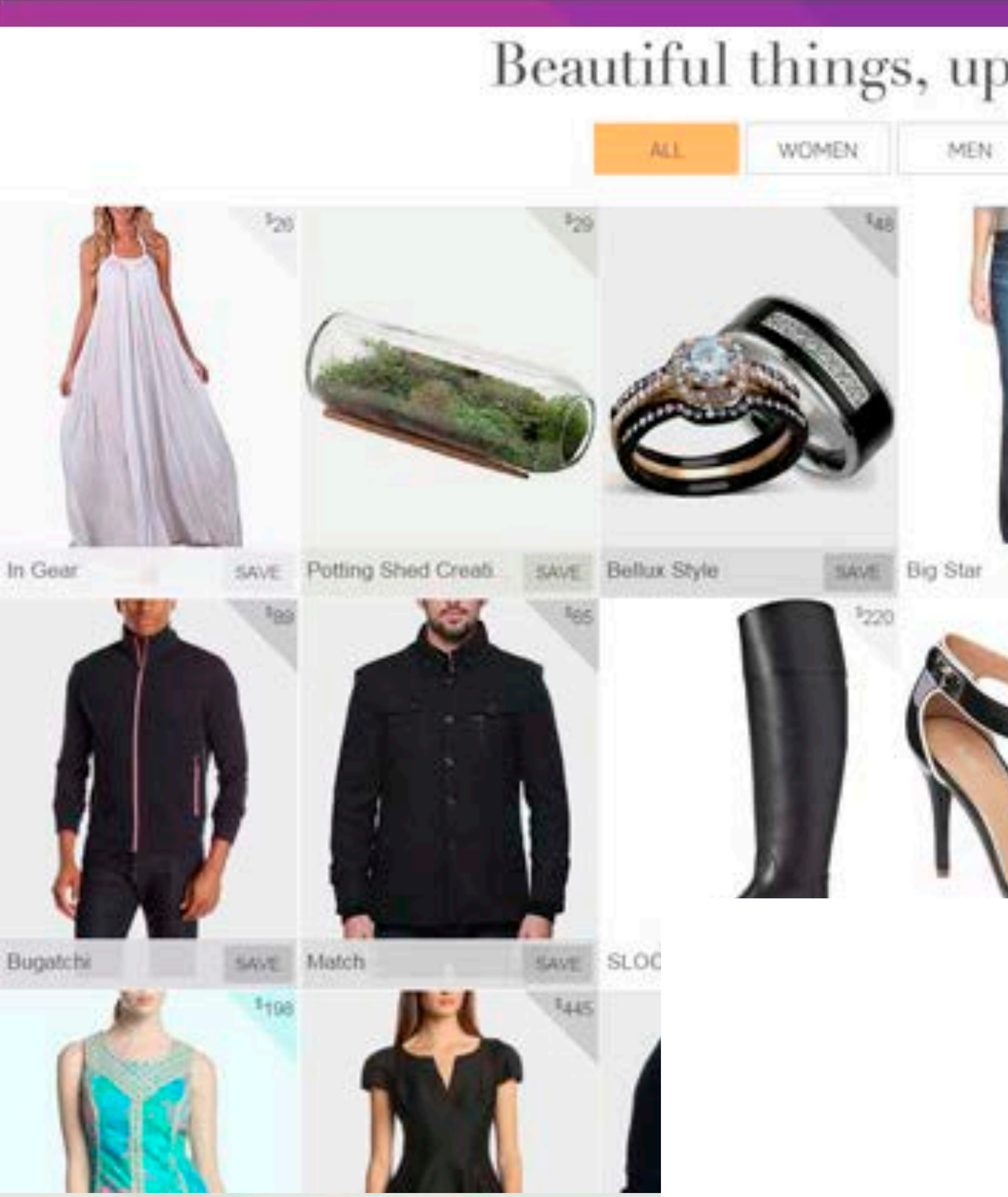


The Advent of AI



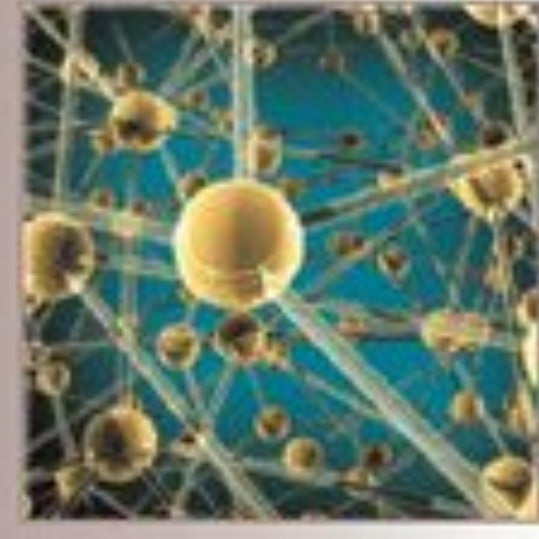
The Advent of AI





Amazon AI





Two Decades of Recommender Systems at Amazon.com

Amazon is well-known for personalization and recommendations, which help customers discover items they might otherwise not have found. In this update to our original article, we discuss some of the changes as Amazon has grown.

Brent Smith
Amazon.com

Greg Linden
Microsoft

For two decades now,¹ Amazon.com has been building a store for every customer. Each person who comes to Amazon.com sees it differently, because it's individually personalized based on their interests. It's as if you walked into a store and the shelves started rearranging themselves, with what you might want moving to the front, and what you're unlikely to be interested in shuffling further away.

From a catalog of hundreds of millions of items, Amazon.com's recommendations pick a small number of items you might enjoy based on your current context and your past behavior. The algorithms aren't magic; they simply share with you what other people have already discovered. The algorithm does all the work. It's computers helping people help other people, implicitly and anonymously.

Amazon.com launched item-based collaborative filtering in 1998, enabling recommendations at a previously unseen scale for millions of customers and a catalog of millions of items. Since we wrote about the algorithm in *IEEE Internet Computing* in 2003,² it has seen widespread use across the Web, including YouTube, Netflix, and many others. The algorithm's success has been from its simplicity, scalability, and often surprising and useful

recommendations, as well as desirable properties such as updating immediately based on new information about a customer and being able to explain why it recommended something in a way that's easily understandable.

What was described in our 2003 *IEEE Internet Computing* article has faced many challenges and seen much development over the years. Here, we describe some of the updates, improvements, and adaptations for item-based collaborative filtering, and offer our view on what the future holds for collaborative filtering, recommender systems, and personalization.

The Algorithm

As we described it in 2003, the item-based collaborative filtering algorithm is straightforward. In the mid-1990s, collaborative filtering was generally user-based, meaning the first step of the algorithm was to search across other users to find people with similar interests (such as similar purchase patterns), then look at what items those similar users found that you haven't found yet. Instead, our algorithm begins by finding related items for each item in the catalog. The term "related" could have several meanings here, but at this point,

Standing the Test of Time

As part of recognizing *IEEE Internet Computing* for its 20 years in publication, I recommended to the editorial board that we pick one of our magazine articles that, over the past 20 years, has withstood the "test of time." In selecting an article, we evaluated the ideas in more than 20 candidate articles that reported on "evergreen" research areas over the past two decades and then assessed these articles based on downloads from IEEE Xplore, citations, and mentions of the work in popular press. This information was presented to a committee consisting of previous Editors in Chief for the magazine. I would like to thank the selection committee from the editorial board — led by Arun Iyengar, and including Fred Douglass, Robert Filman, Michael Huhns, Charles Petrie, Michael Rabinovich, and Munindar Singh. This committee deliberated on the top three articles by evaluating each work's previous importance within the context of its sustained importance in the future.

It's my pleasure to recognize the committee's official "Test of Time" winner: an industry article titled "Amazon.com Recommendations: Item-to-Item Collaborative Filtering" by Greg Linden,

Brent Smith, and Jeremy York, from the January/February 2003 issue of *IC* (see doi:10.1109/MIC.2003.1167344). Fourteen years after the publication of this article, it shows 125 downloads from IEEE Xplore in one month, with more than 12,754 downloads since January 2011. The article currently shows 4,258 citations in Google Scholar. I'm delighted that the selection committee recommended an industry article, as it aligns with the magazine's focus of accessibility in academic, research, and industrial populations.

In addition to recognizing the article, we asked the authors to create this retrospective piece discussing research and insights that have transpired since publishing their winning "Test of Time" article, while projecting into the future.

Going forward, the magazine hopes to celebrate a "Test of Time" article every 2–3 years. I hope that you enjoy this retrospective article, and please take a moment to congratulate Greg Linden, Brent Smith, and Jeremy York.

— M. Brian Blake

Editor-in-Chief, *IEEE Internet Computing*
Provost and Distinguished Professor, Drexel University

let's loosely define it as "people who buy one item are unusually likely to buy the other." So, for every item i_1 , we want every item i_2 that was purchased with unusually high frequency by people who bought i_1 .

Once this related items table is built, we can generate recommendations quickly as a series of lookups. For each item that's part of this customer's current context and previous interests, we look up the related items, combine them to yield the most likely items of interest, filter out items already seen or purchased, and then we are left with the items to recommend.

This algorithm has many advantages over the older user-based collaborative filtering. Most importantly, the majority of the computation is done offline — a batch build of the related items — and the computation of the recommendations can be done in real time as a series of lookups. The recommendations are high quality and useful, especially given enough data, and remain competitive in perceived quality even with the newer algorithms created over the last two decades. The algorithm scales to hundreds of millions of users and tens of millions of items without sampling or other techniques that can reduce the quality of the recommendations. The algorithm updates immediately on new information about a person's interests. Finally, the recommendations can be explained in an

intuitive way as arising from a list of items the customer remembers purchasing.

In 2003: Amazon.com, Netflix, YouTube, and More

By the time we published in *IEEE* in 2003, item-based collaborative filtering was widely deployed across Amazon.com. The homepage prominently featured recommendations based on your past purchases and items browsed in the store. Search result pages recommended items related to your search. The shopping cart recommended other items to add to your cart, perhaps impulse buys to bundle in at the last minute, or perhaps complements to what you were already considering. At the end of your order, more recommendations appeared, suggesting items to order later. Using e-mails, browse pages, product detail pages, and more, many pages on Amazon.com had at least some recommended content, starting to approach a store for every customer.

Others have reported using the algorithm, too. In 2010, YouTube reported using it for recommending videos.³ Many open source and third-party vendors included the algorithm, and it showed up widely in online retail, travel, news, advertising, and more. In the years following, the recommendations were used so extensively by Amazon.com that a Microsoft Research report estimated 30 percent of Amazon.com's page

Arizona State University & Amazon bring voice-technology programme to campus





amazon

WALK OUT
WITH NO CHECKOUT
**JUST
WALK
OUT**
SHOPPING
AMAZON

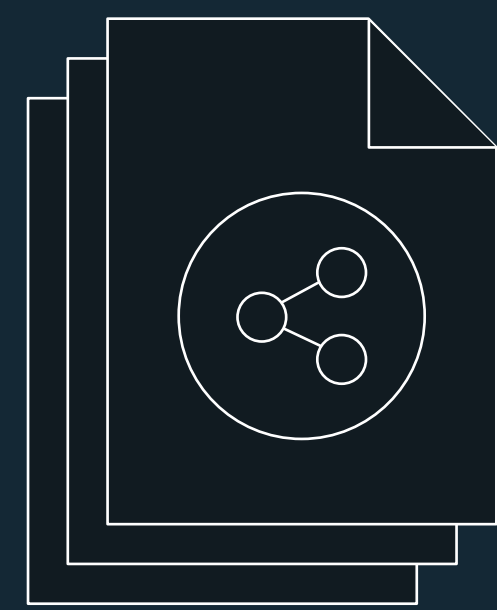


...but how does this help me?

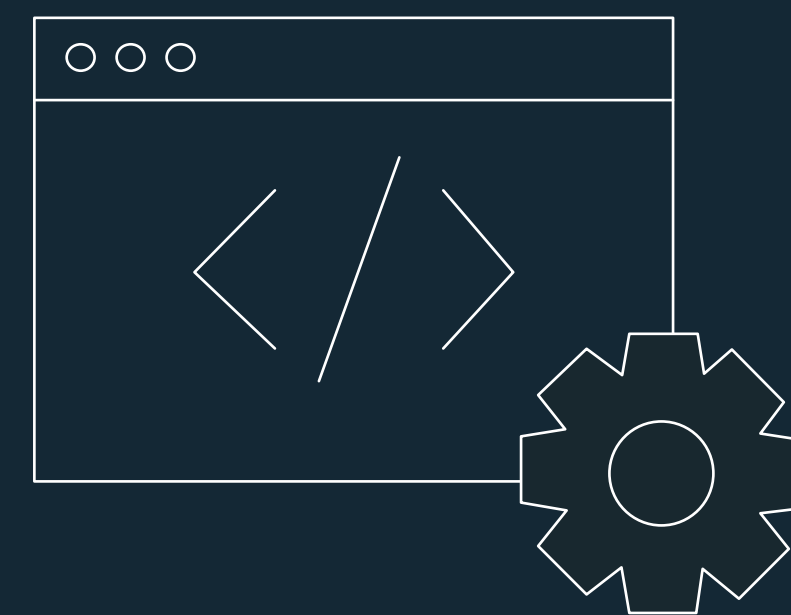
Our mission at 

Put machine learning in the
hands of every developer

Amazon SageMaker



Build, Train and Deploy Machine Learning Models at Scale

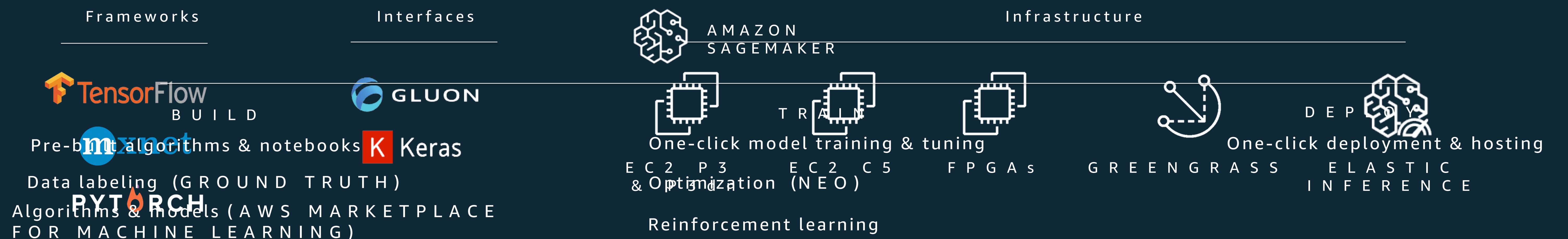


Pre Built Developer Notebooks, Popular Machine Learning algorithms, Automatic Model Tuning



10,000+ Customers using SageMaker today

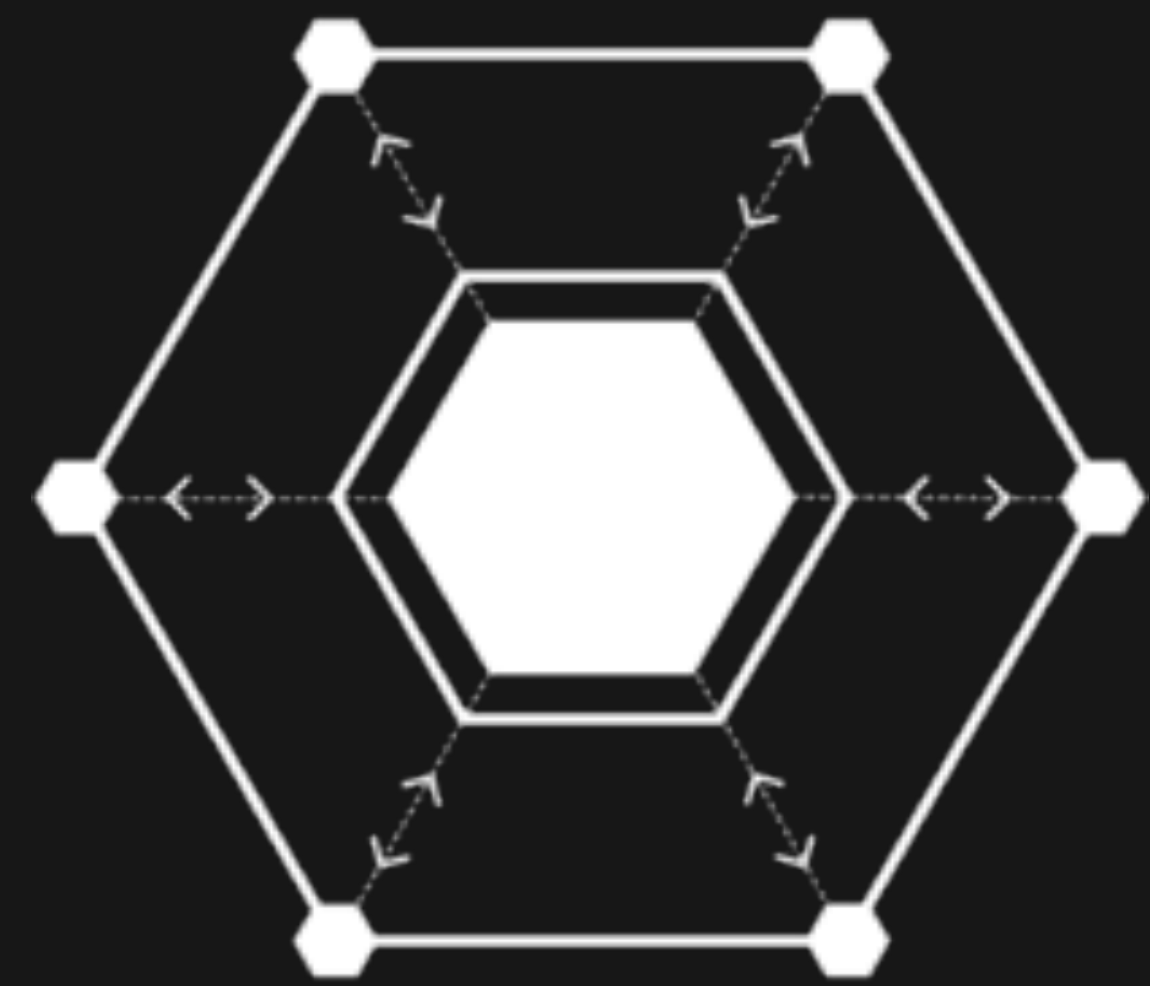
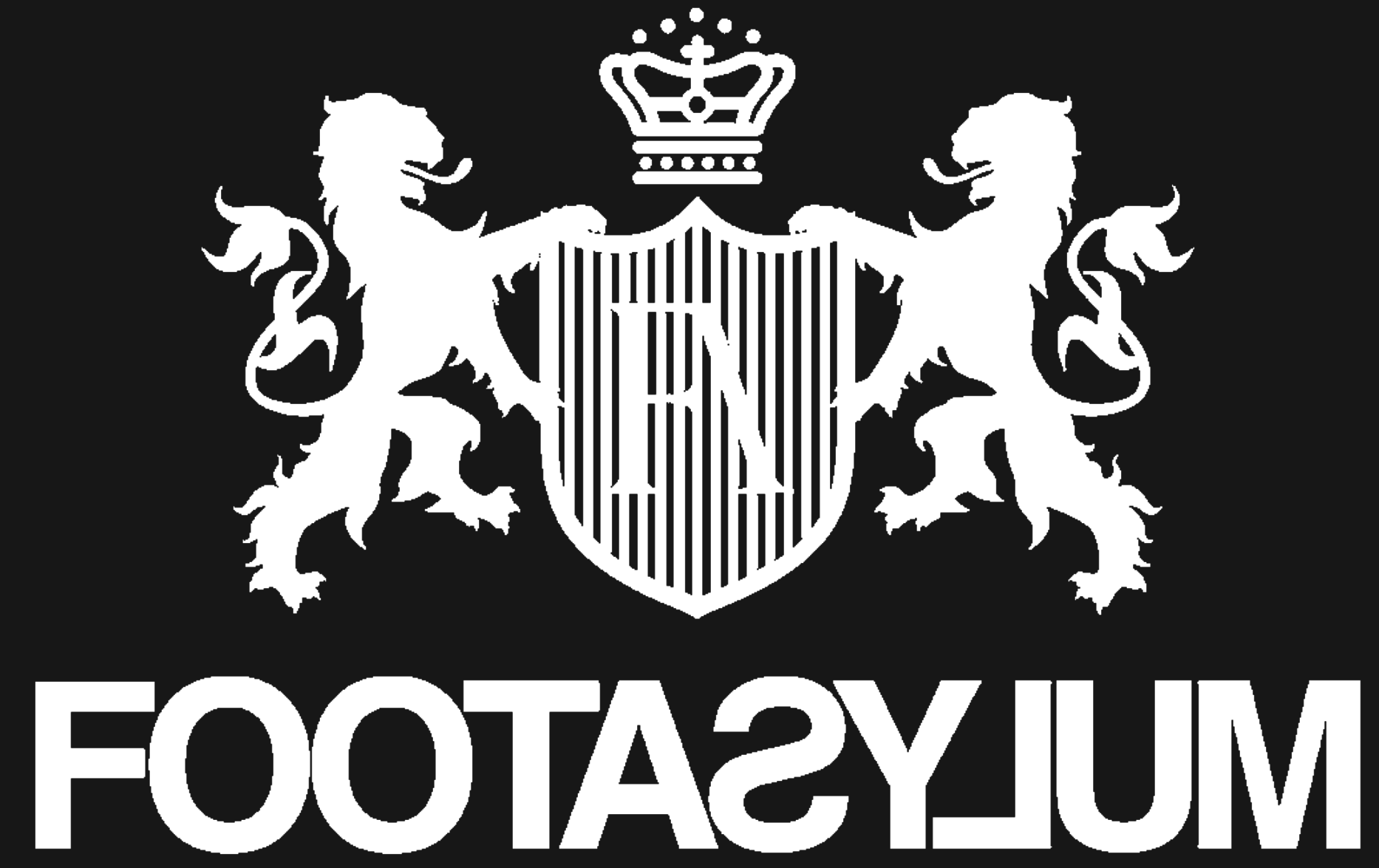
ML FRAMEWORKS & ML SERVICES INFRASTRUCTURE





FOOTASYLUM

**28% UPLIFT IN REVENUE FROM MARKETING
COMMUNICATIONS**



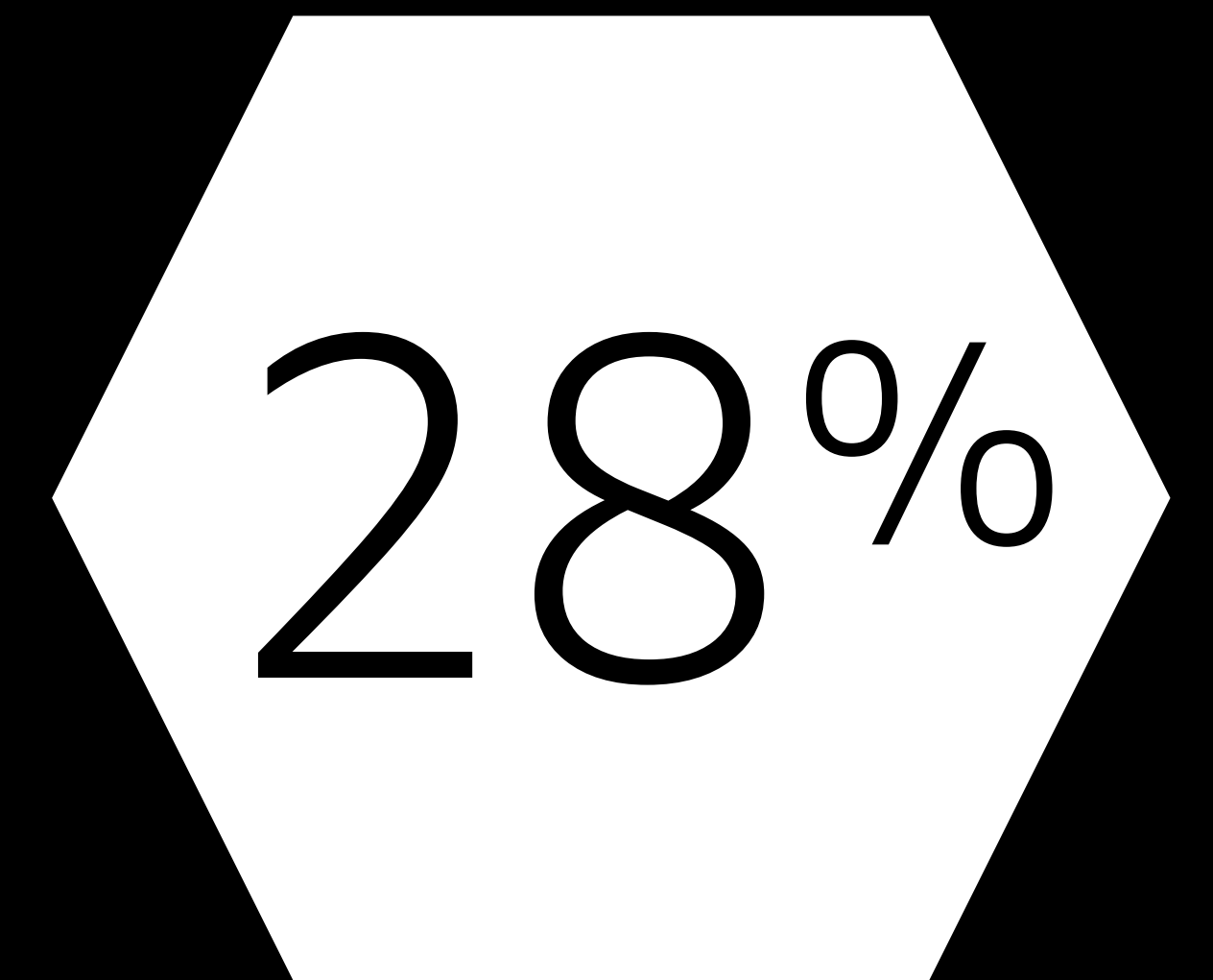
AI System



Machine Learning Algorithms



Segment-based Factors



Uplift in Revenue ↑

Highly-targeted, hyper-personalised marketing.

Immediate results, with a 28% uplift in revenue per email sent.

Using Amazon SageMaker - powered by Intel C5 Compute Instances

AI Services

AI SERVICES

Vision



REKOGNITION
IMAGE



REKOGNITION
VIDEO



TEXTRACT

Speech



POLLY



TRANSCRIBE

Language



TRANSLATE



COMPREHEND
& COMPREHEND
MEDICAL

Chatbots



LEX

Forecasting

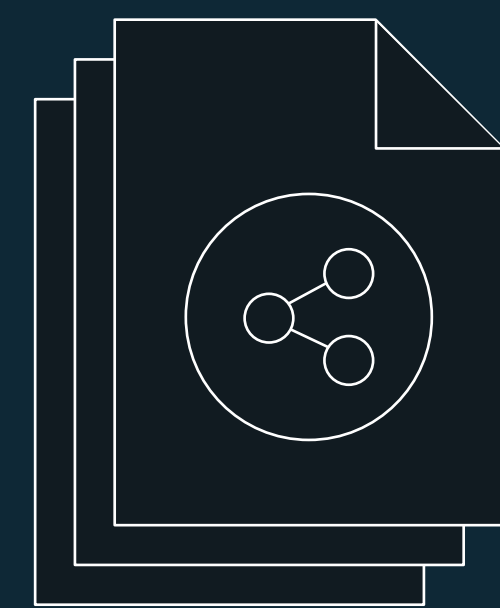


FORECAST

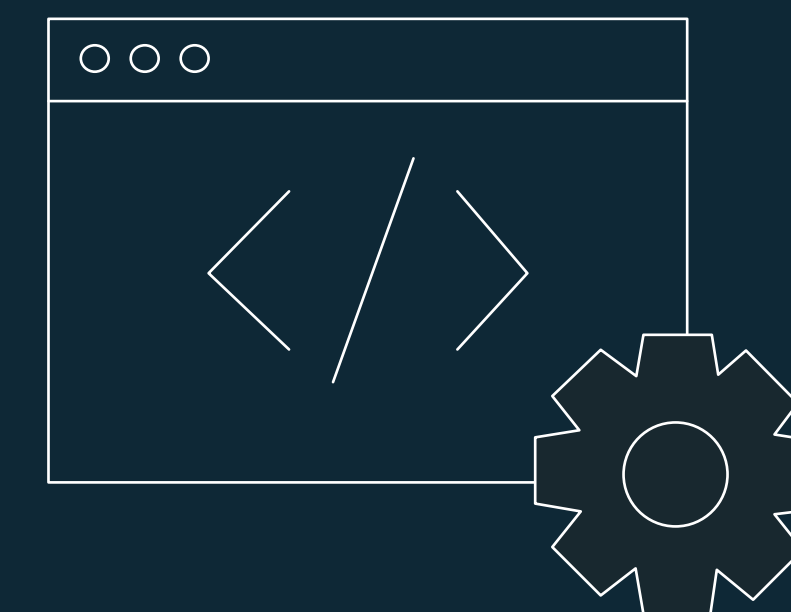
Recommendations



PERSONALIZE



Pre-trained AI services that require
no ML skills or training



Easily add intelligence to your existing
apps and workflows



Quality and accuracy from
continuously-learning APIs

Amazon Lex

BookHotel



Intents

An intent performs an action in response to natural language user input

Utterances

Spoken or typed phrases that invoke your intent

Slots

Slots are input data required to fulfill the intent

Fulfillment

Fulfillment mechanism for your intent

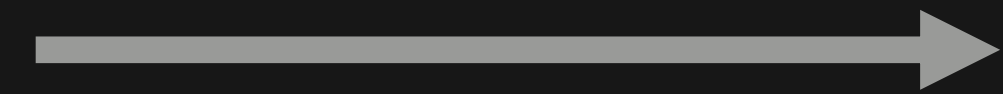


NHS

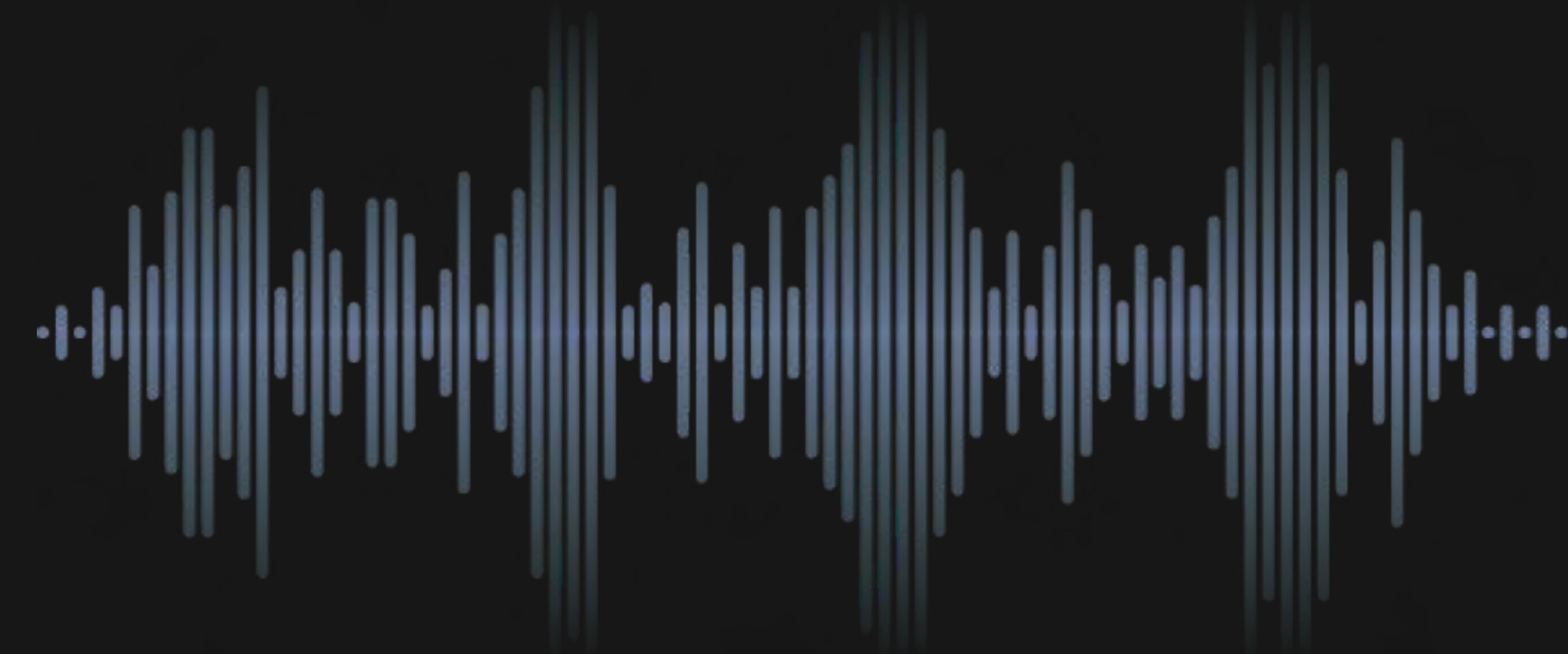
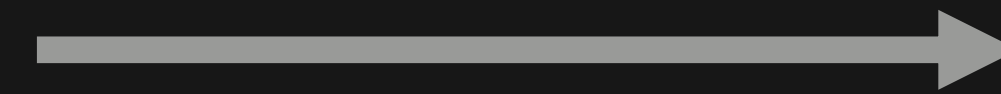
Amazon Lex was able to handle 40% of the incoming call volume.

Amazon Polly: Text In, Life-like Speech Out

"The weather
today in Seattle,
WA is 65°F"

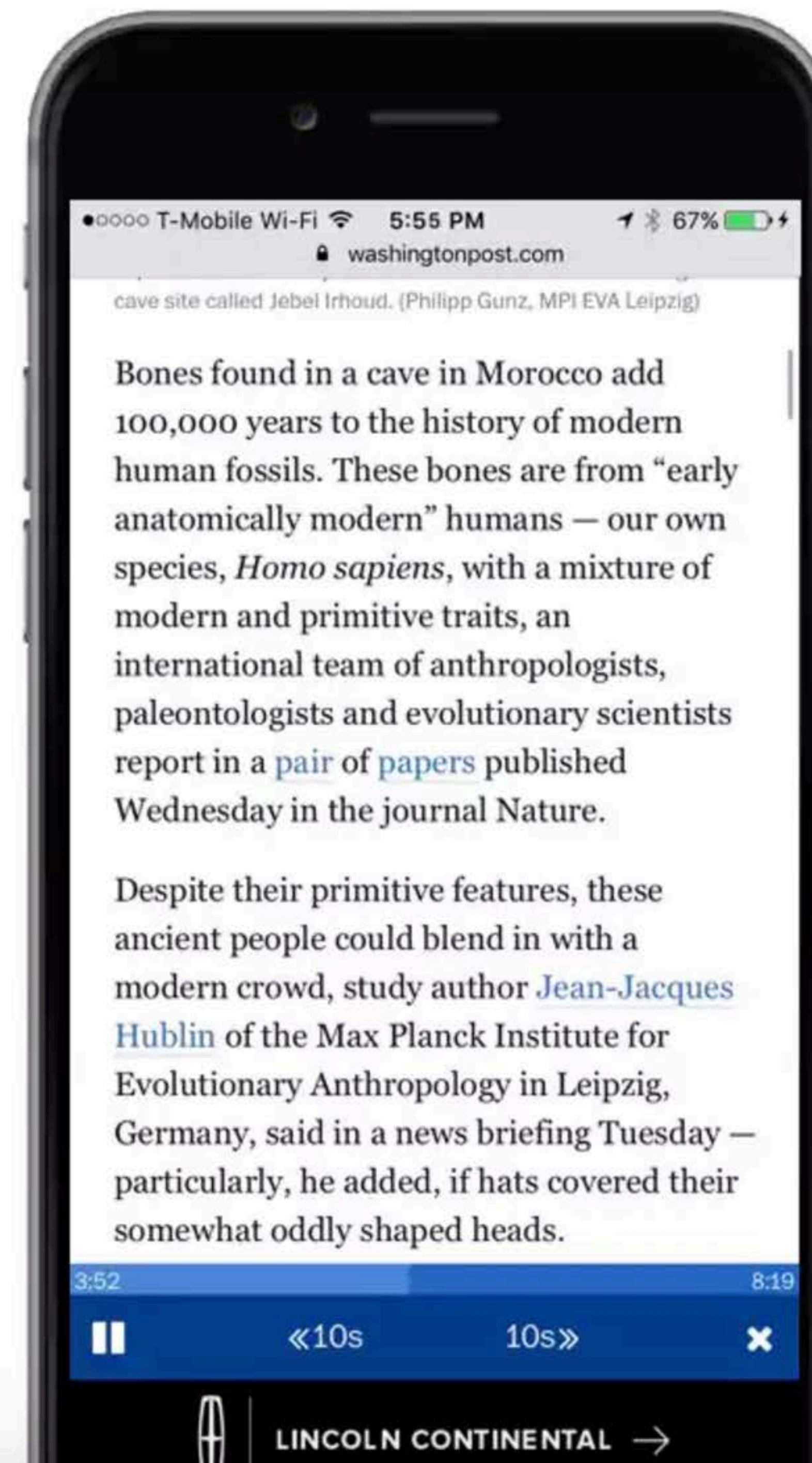
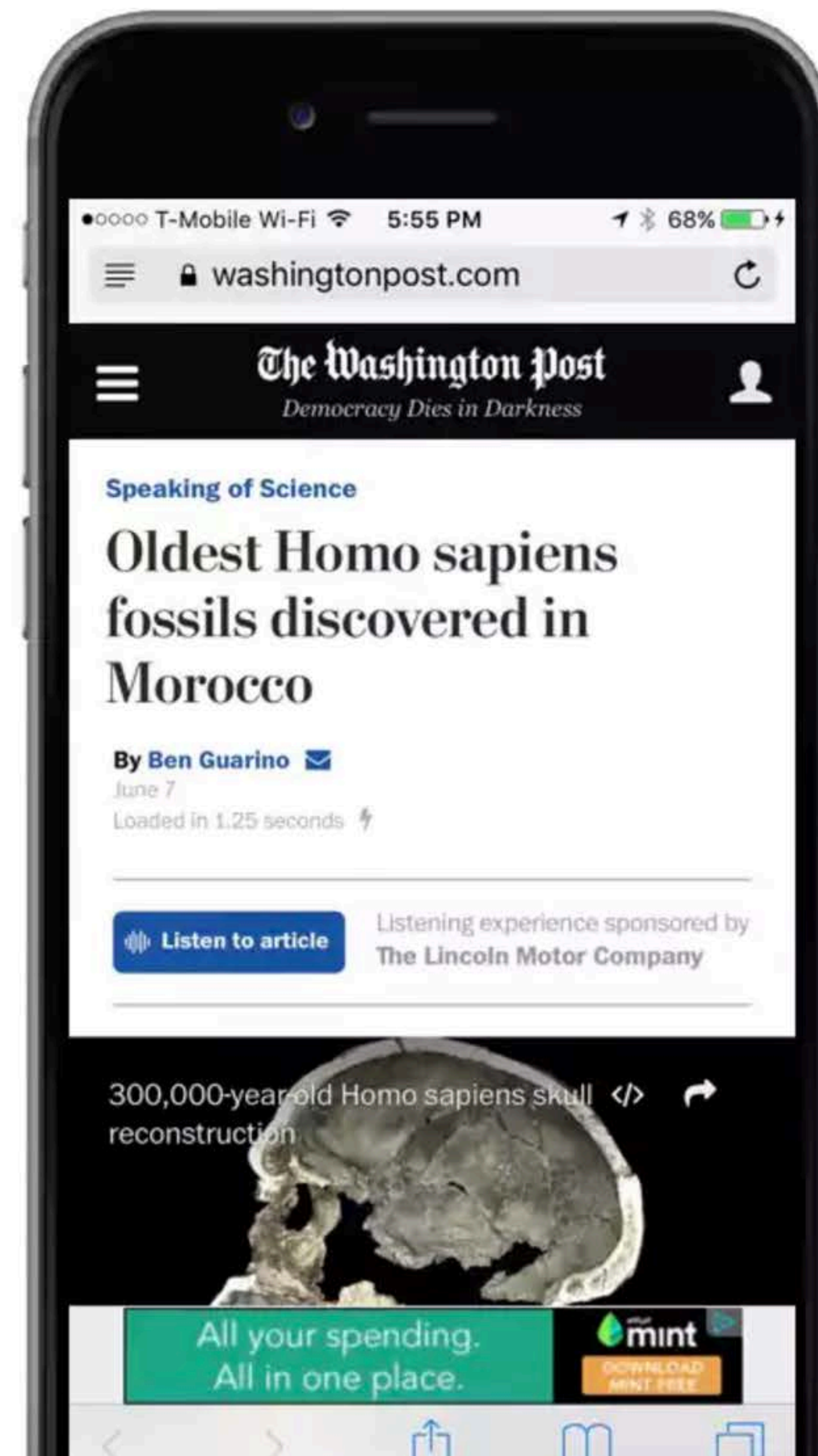


Amazon Polly



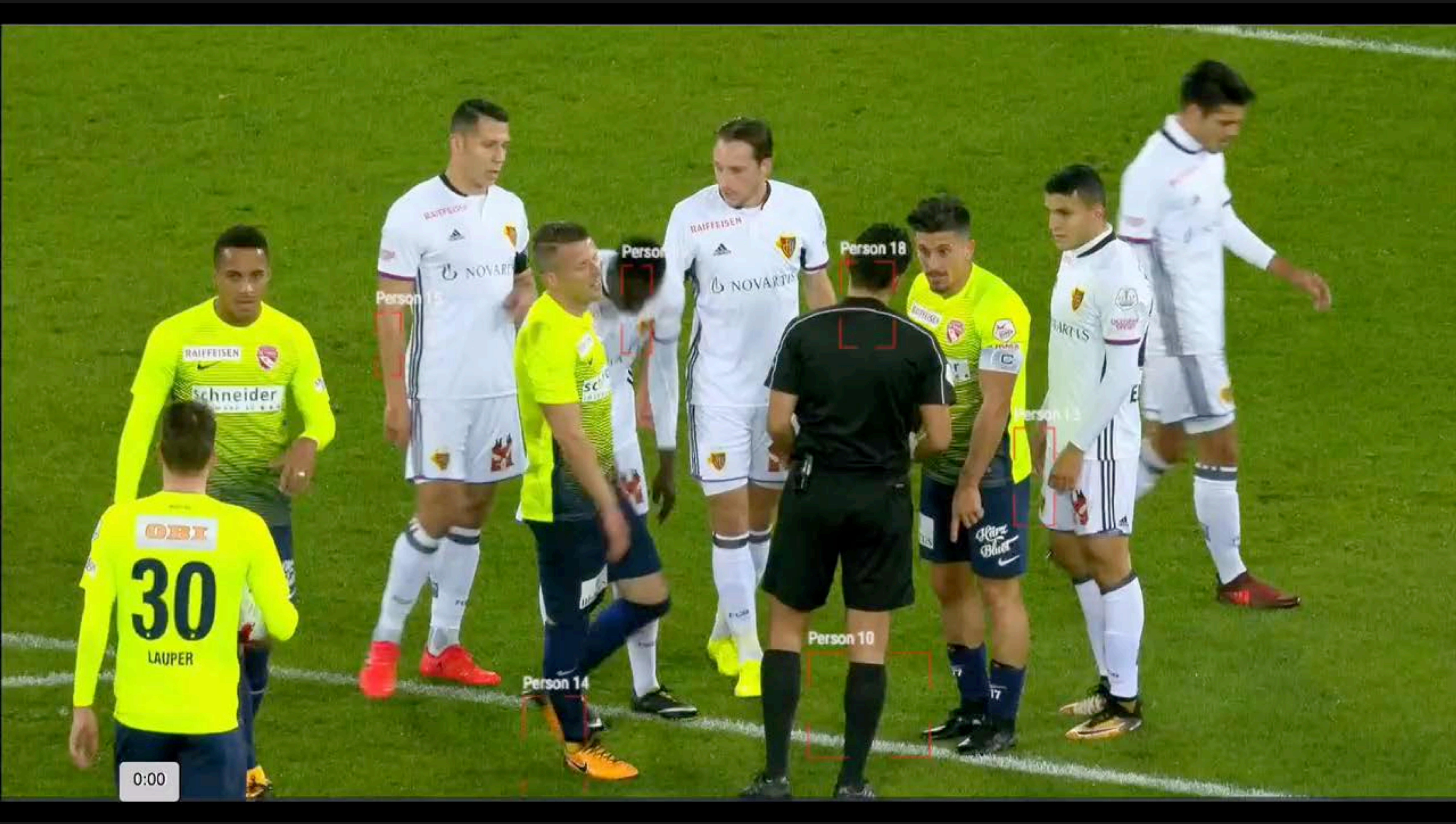
The Washington Post to start experimenting with audio articles using Amazon Polly

By WashPostPR June 9



Amazon Rekognition – Image and Video Analysis





Person 15

Person 15

Person 18

Person 13

Person 14

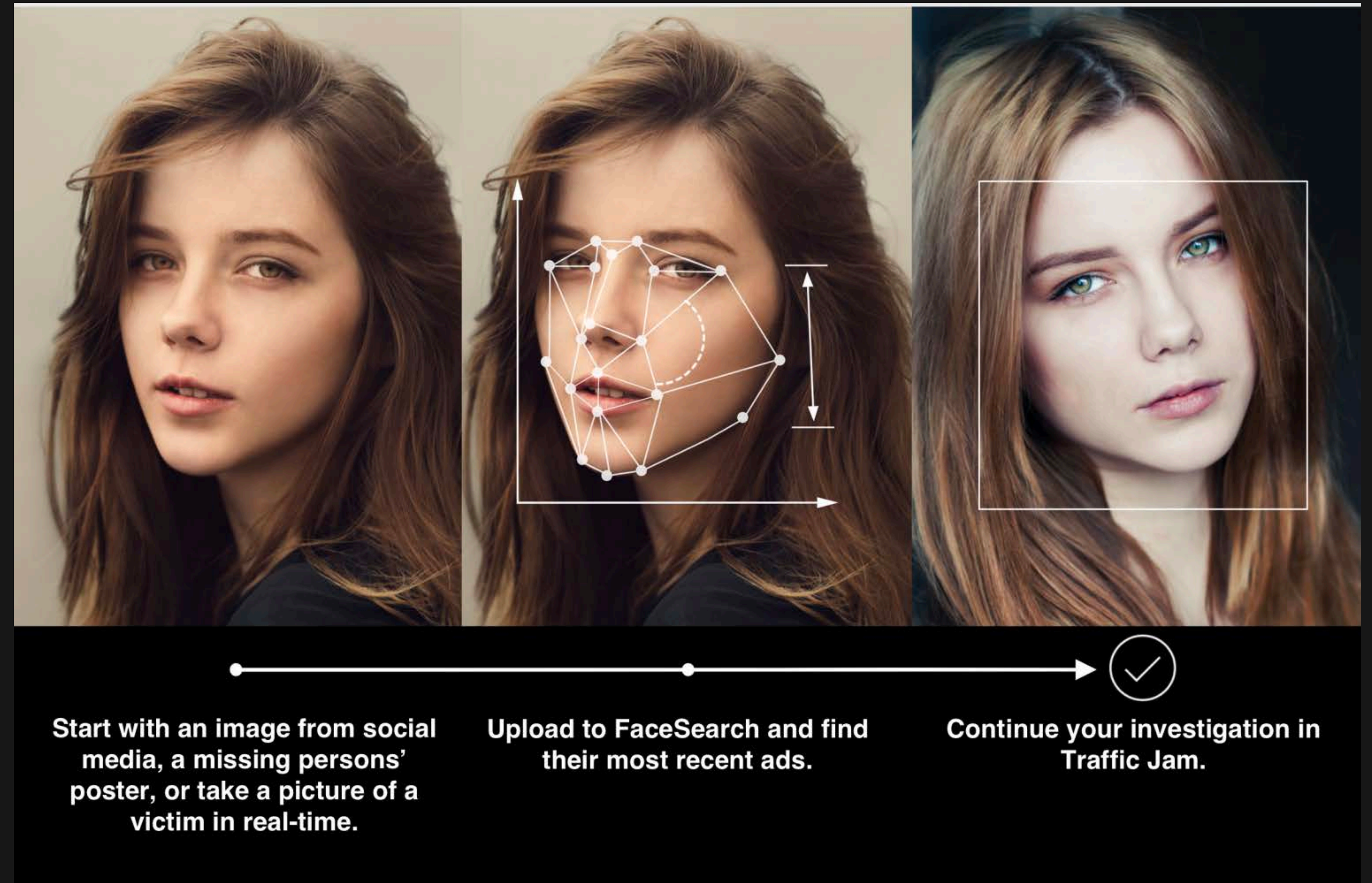
Person 10

0:00

Marinus Analytics - TrafficJam

FaceSearch

Helping to fight Human Trafficking with Amazon Rekognition.





Demo Time

Beat the Bot!

Dean



Steve

We are building a **cloud** that best supports your modern application development needs, and we are innovating across the **entire stack**: from the hypervisor layer to the application construction layer.

Go **Build!**

 @steven_bryen

Verdict

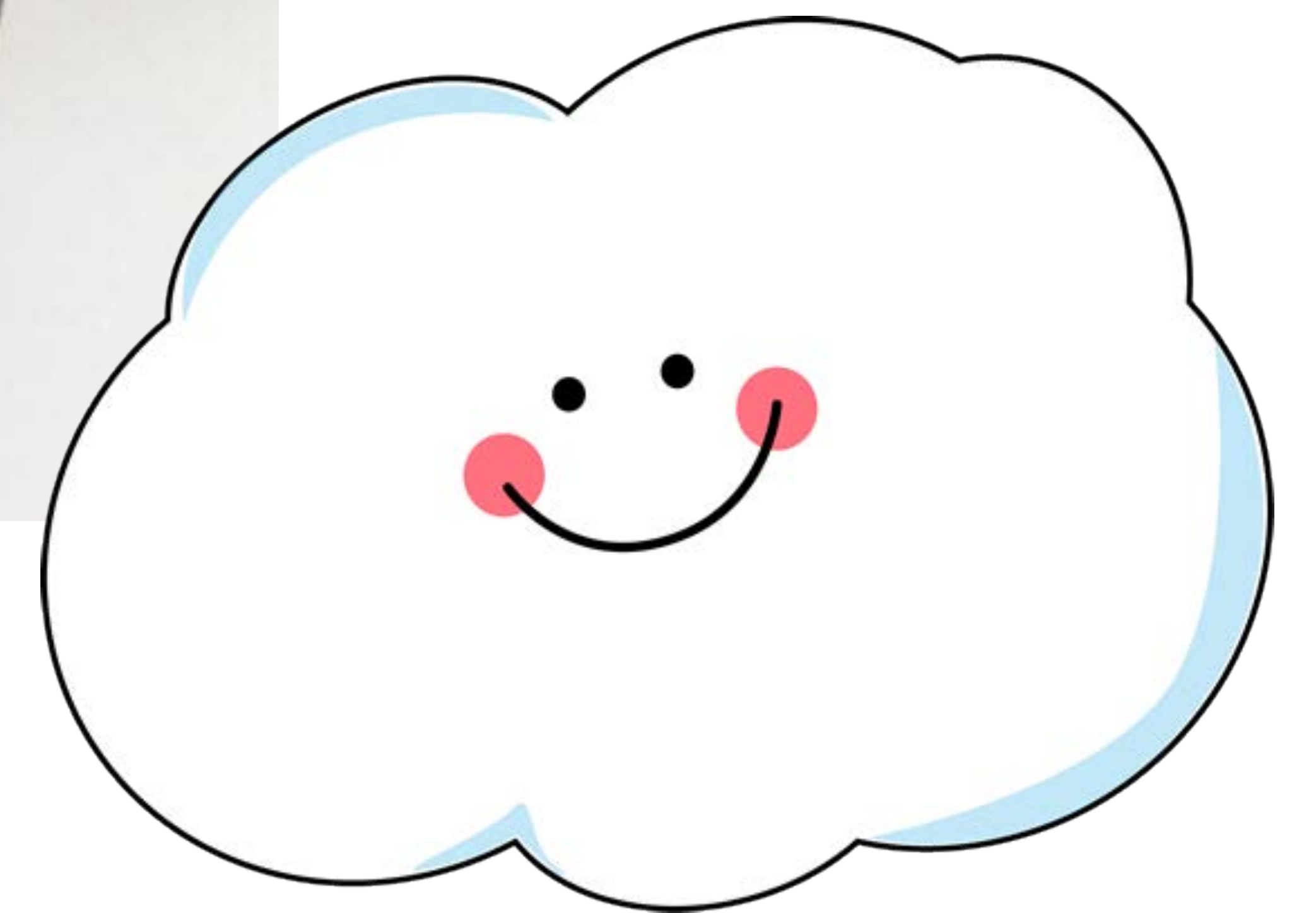
- Cloud is HOT
- Trends in Cloud technology
 - **HOT** or **NOT**?



Thanks AWS Speaker!



Google Home Mini
Hands-free smart speaker



Upcoming Sioux events

- **15 October**

Hot-or-Not: The Next Generation

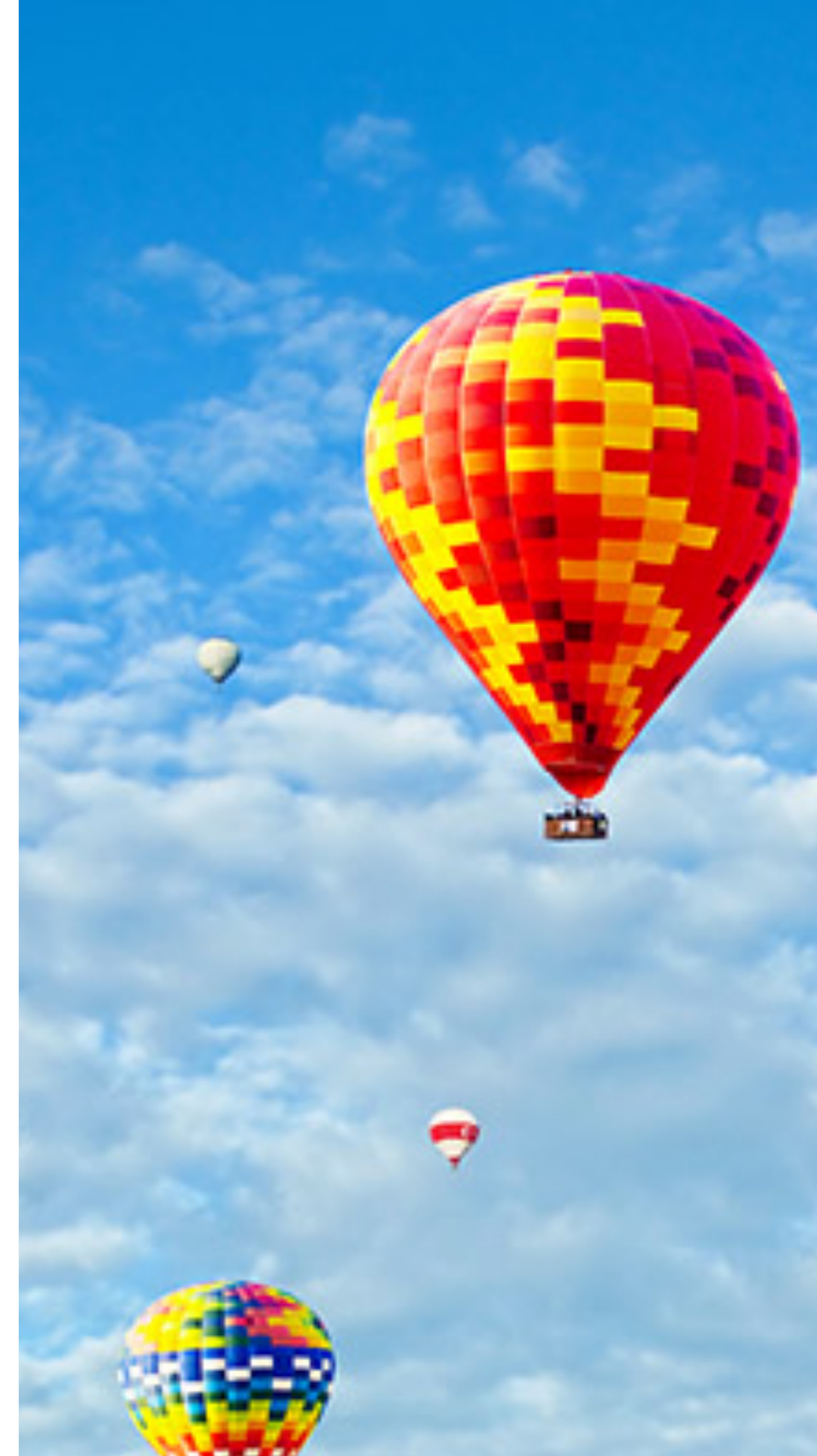
- **19 November**

Sioux Seats to Meet - Eindhoven



Wrap Up

- Thanks for attending
- Drinks
- Demos
- Siou(x) next time!



Source of your technology