INNOVATE

MODERN APPLICATIONS EDITION

20 October, 2022

A developer's guide to modern application development

Matthew Coles
Principal Engineer
Amazon Web Services

Agenda

- Examining current developer challenges
- Overview of AWS developer landscape
- AWS developer tooling
- Golden pathways

Examining current developer challenges

Examining current developer challenges

Over 700 languages

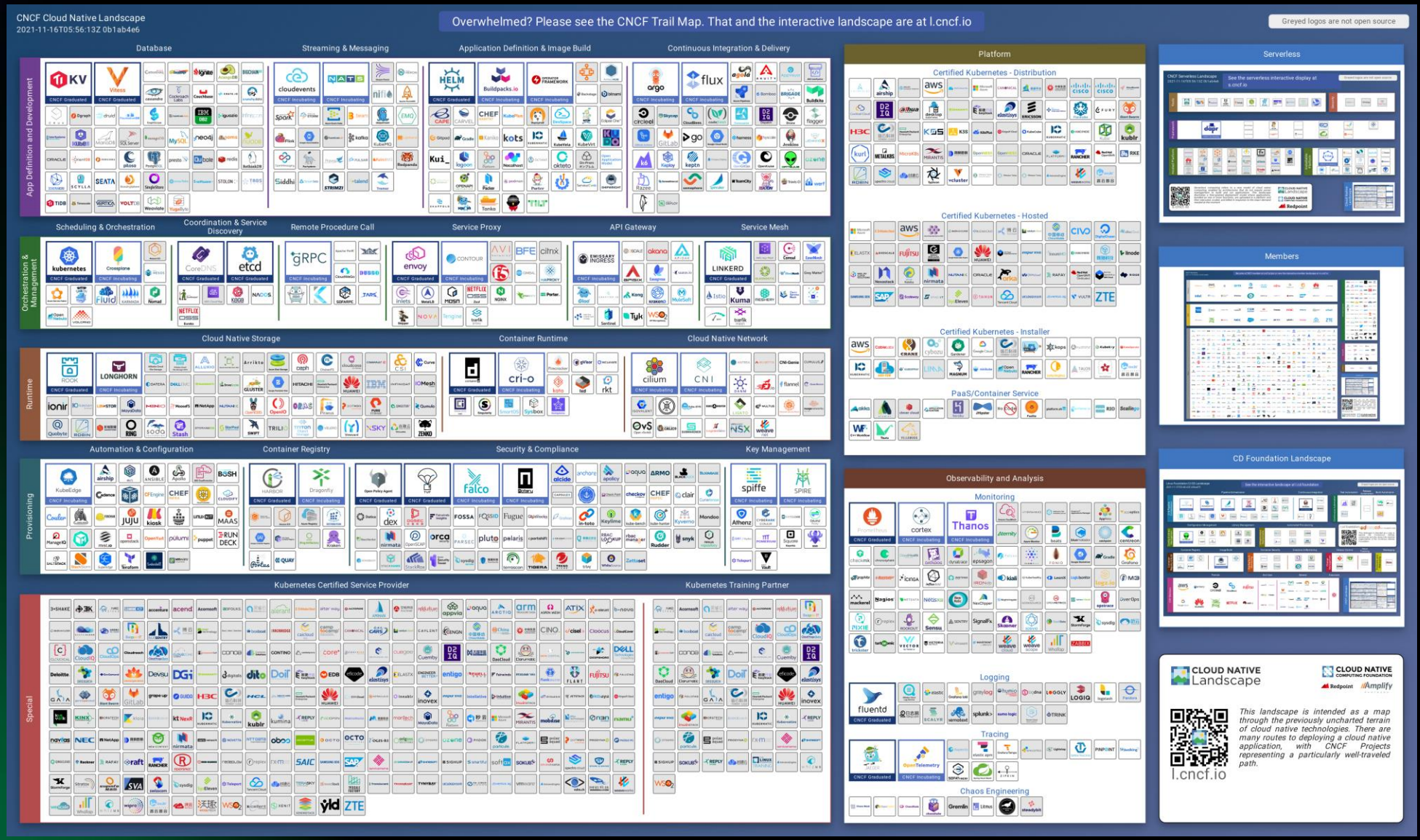
ABAP	Boo	D	**Go**	Ladder Logic	Oz	Ring	Turing
ABC	Bourne shell	Dart	Groovy	Lasso	Paradox	RPG	**TypeScript**
ActionScript	**C**	DCL	Hack	Lisp	Pascal	**Ruby**	Vala
Ada	C shell	Delphi	Harbour	LiveCode	**Perl**	**Rust**	VB.NET
Alice	**C#**	Dylan	Haskell	Logo	**PHP**	S	VBScript
Apex	**C++**	Eiffel	HCL	Lua	Pike	SAS	Verilog
APL	Ceylon	Elixir	HTML	Maple	PL/I	Scala	VHDL
Arc	Clipper	Emacs Lisp	Icon	MATLAB	PL/SQL	Scheme	Visual Basic
ASP	Clojure	Erlang	IDL	Mercury	PostScript	Scratch	Visual FoxPro
AspectJ	**COBOL**	Euphoria	Inform	ML	PowerShell	Shell	Whitespace
Assembly	CoffeeScript	Euphoria	Io	MQL4	Prolog	Smalltalk	xBase
AutoLISP	ColdFusion	F#	**Java**	NATURAL	**Python**	SPARK	XML
AWK	Common Lisp	Factor	**JavaScript**	NXT-G	Q	**SQL**	Xquery
Bash	Crystal	Forth	Julia	Objective-C	R	Stata	Yorick
Basic	CSS	**Fortran**	Korn shell	OCaml	Racket	Swift	Z shell
bc	cT	Gambas	**Kotlin**	OpenCL	REBOL	Tcl	Zig
BCPL	Curl	Genie	LabVIEW	OpenEdge ABL	Red	Transact-SQL	

Examining current developer challenges

Javascript

Adonis	Cheerio	Dotenv	Handlebars	Luxon	Nodemon	**React Native**	Swagger
Angular	Clean-CSS	EJS	Hapi	Marked	NoSQL	Reactstrap.	Tailwind
Angular	Cloudinary	**Electron**	Helmet	Materialize	**npm**	Redux	tslib
Ant Design	CodePen	**Ember**	HTML-Minifier	**MEAN**	**Nuxt**	Request	TypeScript
Async	Commander	ESLint	Inquirer	**MERN**	ORMs	Restify	UglifyJS2
Axios	Concurrently	**Express**	**Jamstack**	Meteor	**Parcel**	Rollup	Underscore
Babel	Config	Faker	Jasmine	Mocha	Passport	**RxJS**	Uuid
Backbone	**Cordova**	Flow	Jest	Moment	Pluralize	Sails	Validator
Bcrypt	core-js	Formik	Joi	Mongoose	PM2	**Scully**	**Vue**
Body-parser	Cors	Foundation	jQuery	Morgan	**Polymer**	Sequelize	**Webpack**
Bootstrap	**Crank**	Fs-extra	JSDoc	Multer	Preact	Sharp	**Websockets**
Bower	CSS-in-JS	**Gatsby**	JSFiddle	Mustache	Prettier	ShellJS	Winston
Browserify	CSV	GM	JSHint	**Next**	Protractor	Shortid	WS
Brunch	Cypress	**GraphQL**	JSONWebToken	Node-cache	Puppeteer	**Socket.io**	**Yarn**
Bulma	DayJS	**Gridsome**	Koa	Node-dir	QUnit	SQL	Yeoman
Chai	Debug	**Grunt**	local storage	**Node.js**	Randomcolor	**Stencil**	YUIDoc
Chalk	Docco	**Gulp**	**Lodash**	Nodemailer	**React**	**Svelte**	**Yup**

Examining current developer challenges



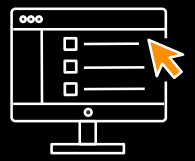
Examining current developer challenges

Development complexity

Multiple programming languages

Multiple frameworks and libraries

Numerous cloud services and APIs



Identifying relevant best practices

Business needs

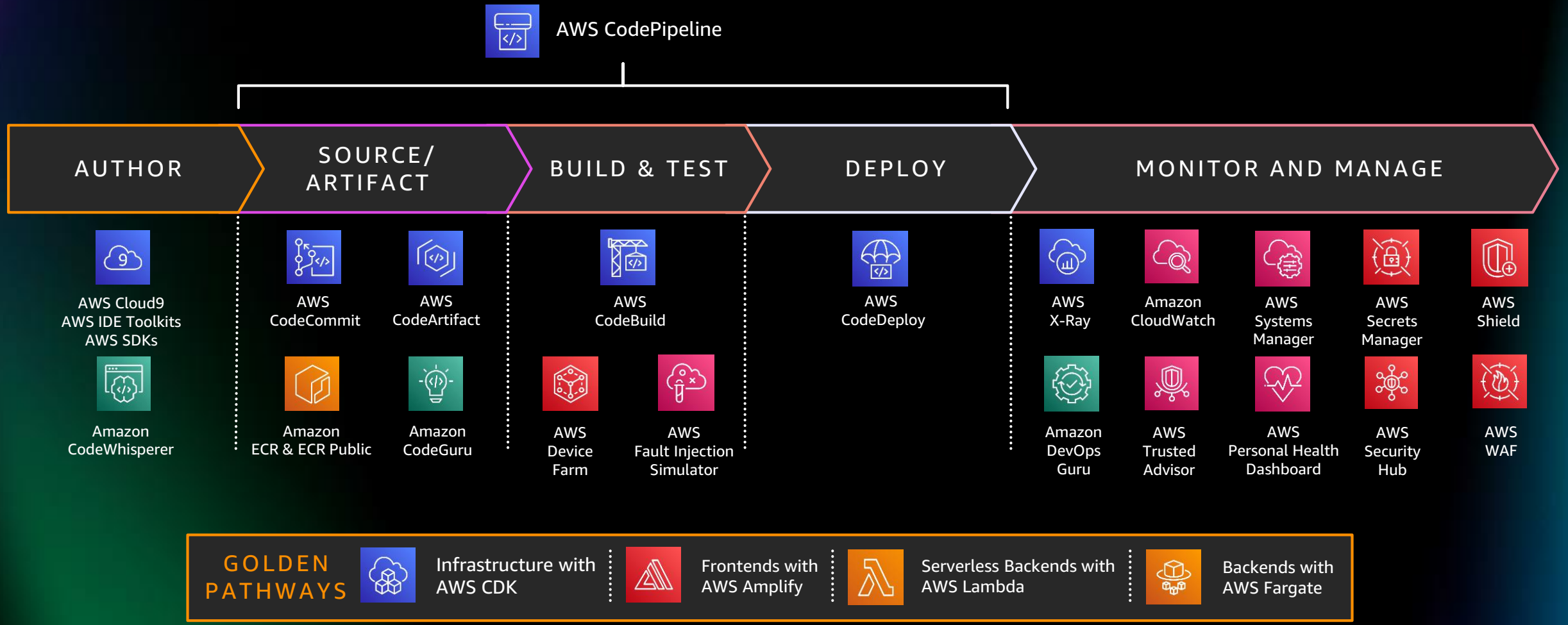
Fast Delivery

Scalable and Production Grade

Innovate and Differentiate

No Downtime

Overview of AWS developer landscape



AWS Developer Tooling

Authoring

Work with many languages with AWS SDK's

Backend


JavaScript


Python


PHP


.NET


Node.js


Embedded C


C++


Ruby

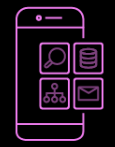

Java


Go


Arduino Yún

Frontend


iOS


Android


Javascript


React Native


Ionic


Flutter


React


Angular


Vue

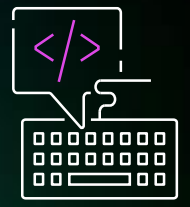
Work with your favourite IDE's with AWS Toolkits

Microsoft
Visual Studio

Microsoft
Visual Studio Code

Pair Programming with Amazon CodeWhisperer

Accelerate frontend and backend development by empowering developers with automatic code recommendations



Languages: Python, Java, JavaScript

IDEs: JetBrains, Visual Studio Code, AWS Cloud9, AWS Lambda console

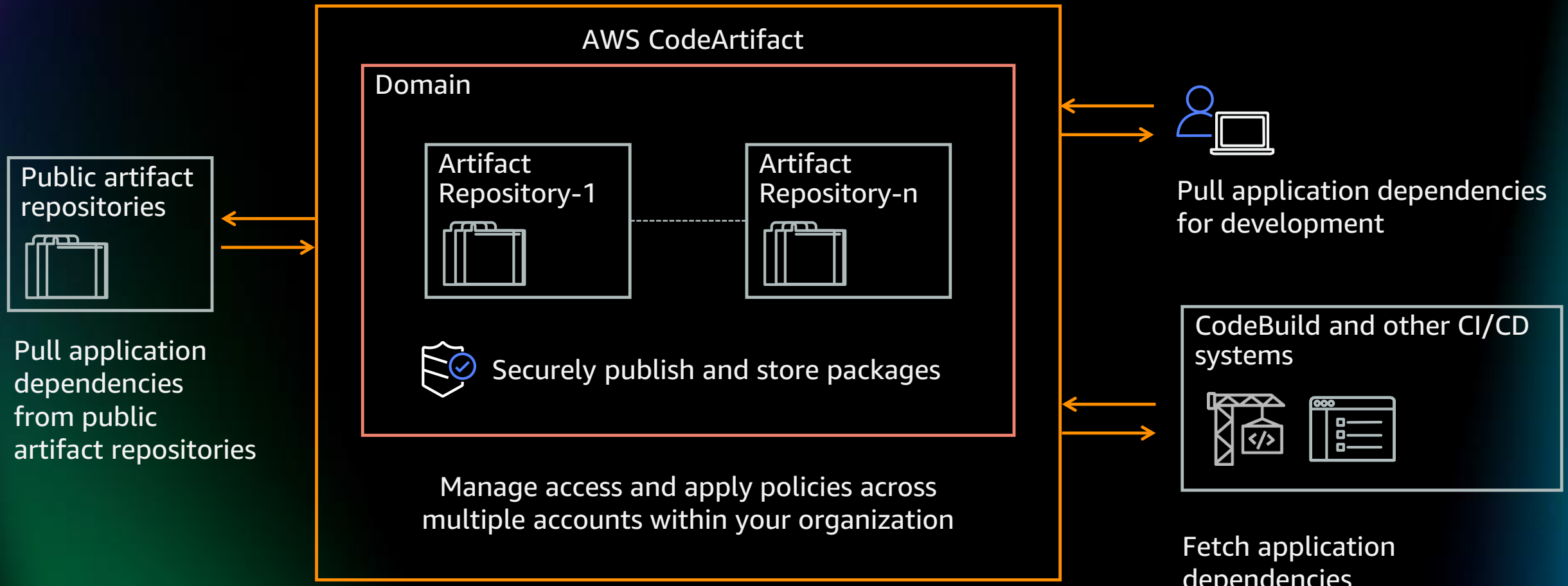
First-class support for AWS APIs

```python
# Write a function to upload a file to S3.
def upload_file_to_s3(file_name, bucket_name, object_name):
    """
    Uploads a file to an S3 bucket

    :param file_name: File to upload
    :param bucket_name: Bucket to upload to
    :param object_name: S3 object name. If none then file_name is used
    :return: True if file was uploaded, else False
    """

    # Upload the file
    s3_client = boto3.client('s3',
                             aws_access_key_id=AWS_ACCESS_KEY_ID,
                             aws_secret_access_key=AWS_SECRET_ACCESS_KEY,
                             region_name=AWS_REGION_NAME)

    try:
        s3_client.upload_file(file_name, bucket_name, object_name)
        print(f'File {file_name} uploaded to S3 bucket {bucket_name} as {object_name}')
        return True
    except FileNotFoundError:
        print(f'File {file_name} not found')
```

AWS Developer Tooling

Source / Artifact

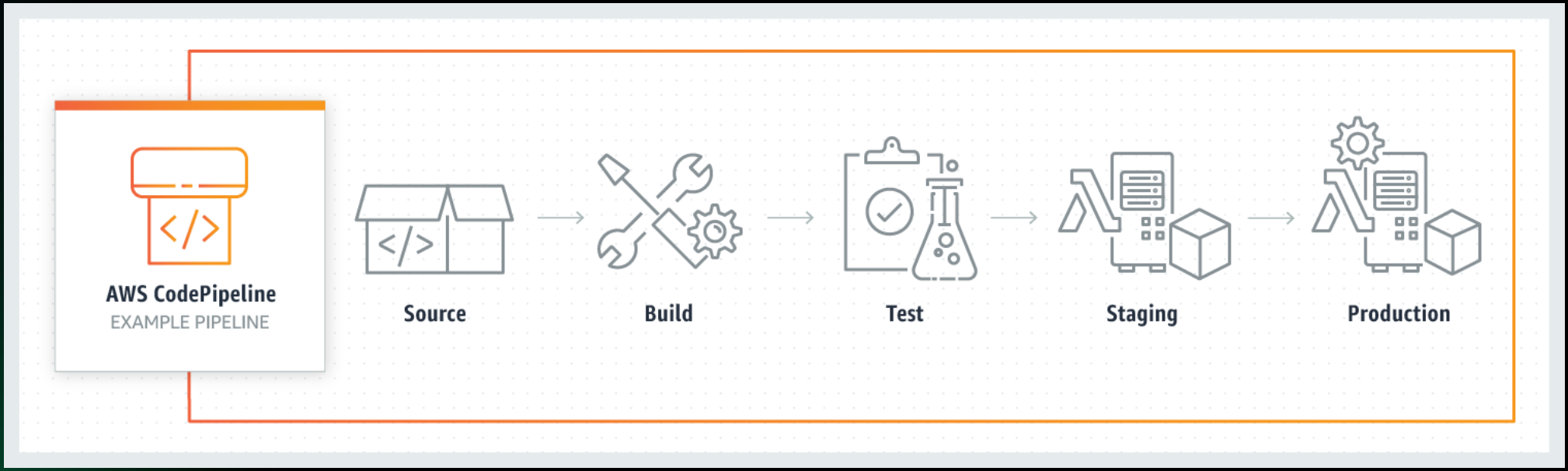
Managing code assets with AWS CodeArtifact



AWS Developer Tooling

Build, Test and Deploy

CI / CD with AWS Code Suite



ML-driven code reviews with Amazon CodeGuru

CodeGuru Reviewer

CodeGuru Profiler

Write and review

Build and test

Deploy

Measure

Improve

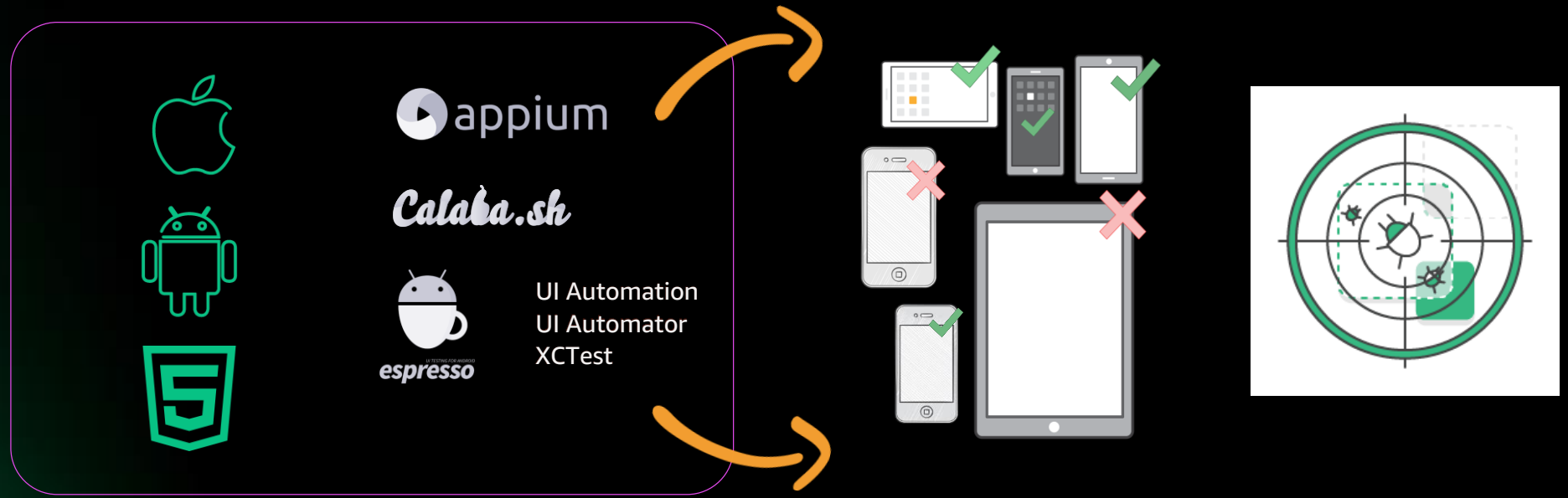
Built-in **code reviews** with actionable recommendations

Detect and **optimize** the expensive lines of code

Easily identify **performance and cost improvements** in production environment

aws

Testing with AWS Device Farm



Your App + Popular Test Frameworks

AWS Device Farm

Detailed Test Reports

AWS Developer Tooling

Monitor and Manage

Automated monitoring with AWS DevOps Guru

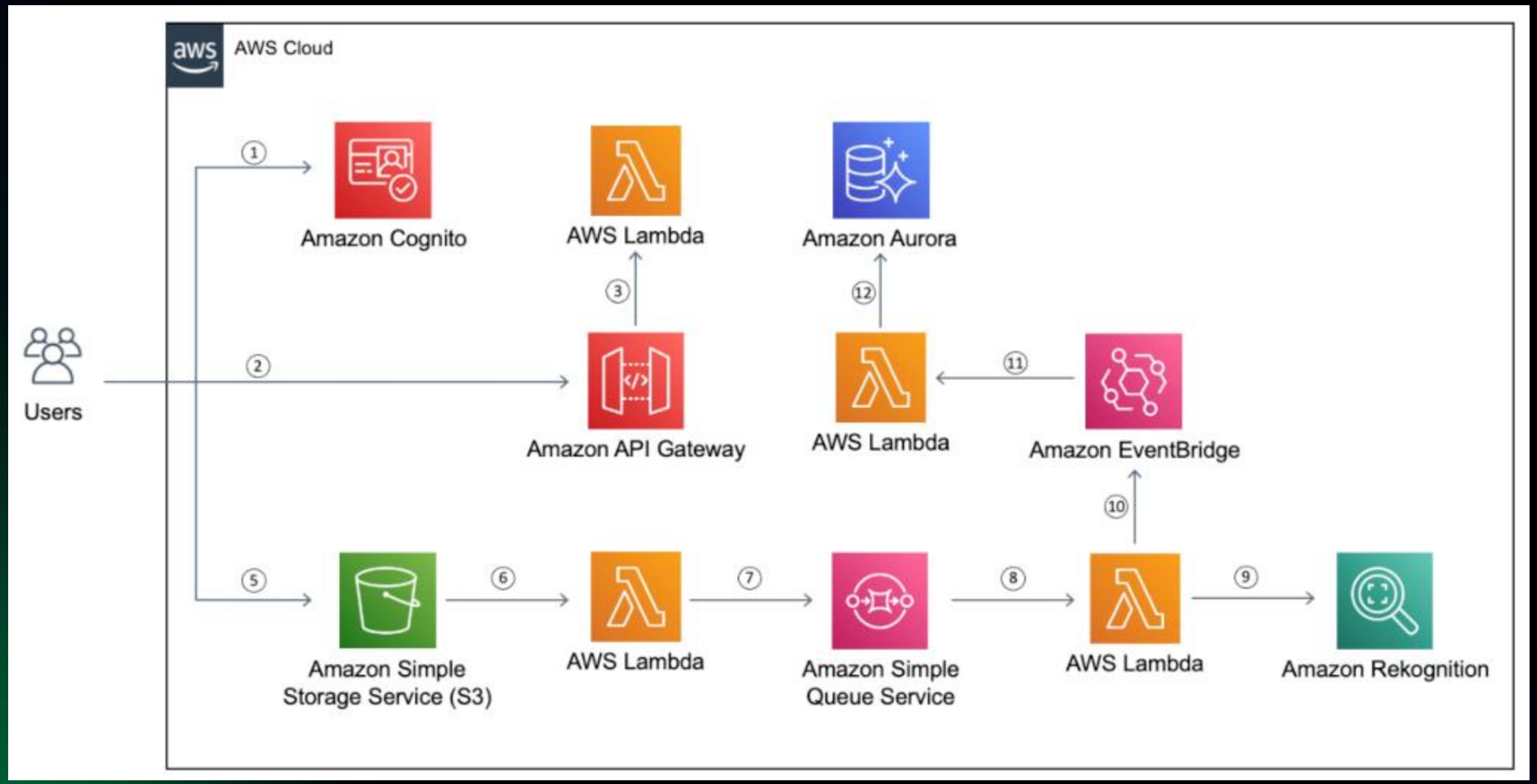
AWS DevOps Guru is an ML-powered service that makes it easy for developers and operators to automatically detect issues to improve application availability and reduce expensive downtime—no machine learning experience required.

Golden Pathways

Golden Pathways

Infrastructure

Building infrastructure with AWS Cloud Development Kit (AWS CDK)



Building infrastructure with AWS CDK

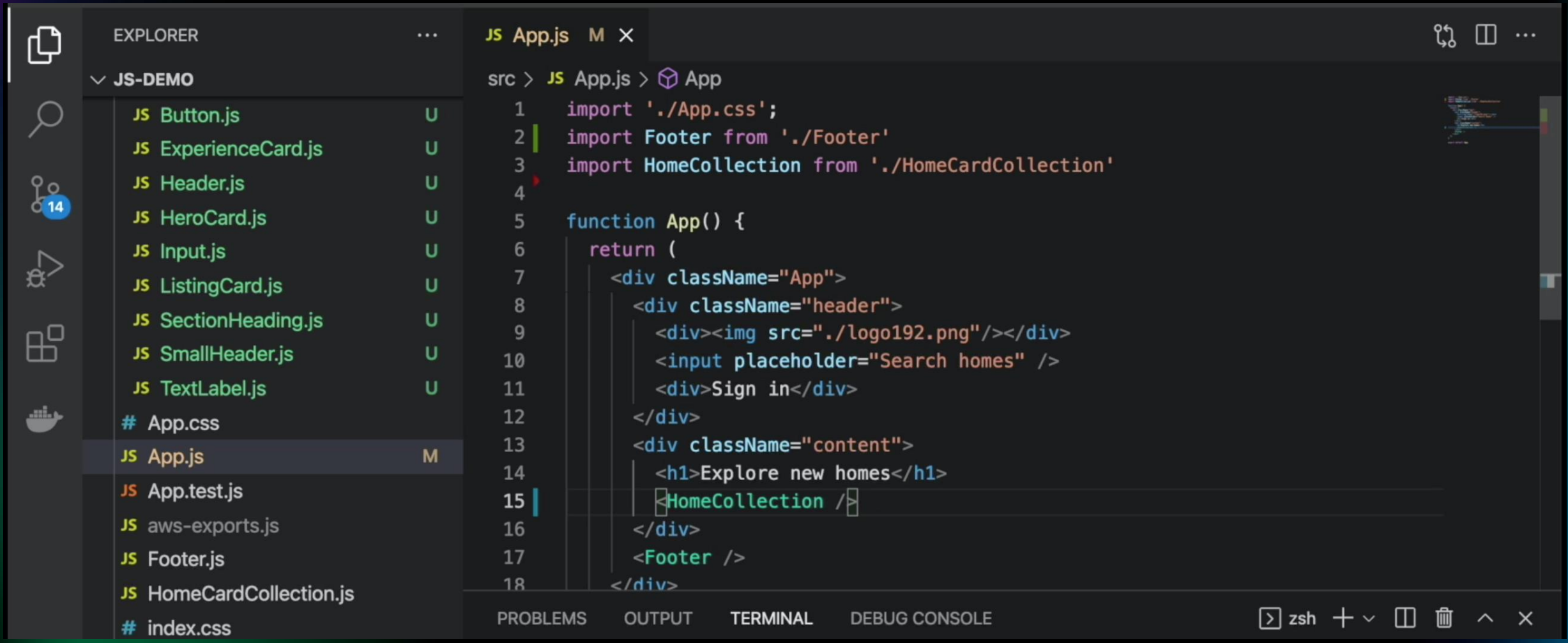
TypeScript | JavaScript | Python | Java | C#

```typescript
export class MyEcsConstructStack extends core.Stack {
  constructor(scope: core.App, id: string, props?: core.StackProps) {
    super(scope, id, props);

    const vpc = new ec2.Vpc(this, "MyVpc", {
      maxAzs: 3 // Default is all AZs in region
    });

    const cluster = new ecs.Cluster(this, "MyCluster", {
      vpc: vpc
    });

    // Create a load-balanced Fargate service and make it public
    new ecs_patterns.ApplicationLoadBalancedFargateService(this, "MyFargateService", {
      cluster: cluster, // Required
      cpu: 512, // Default is 256
```

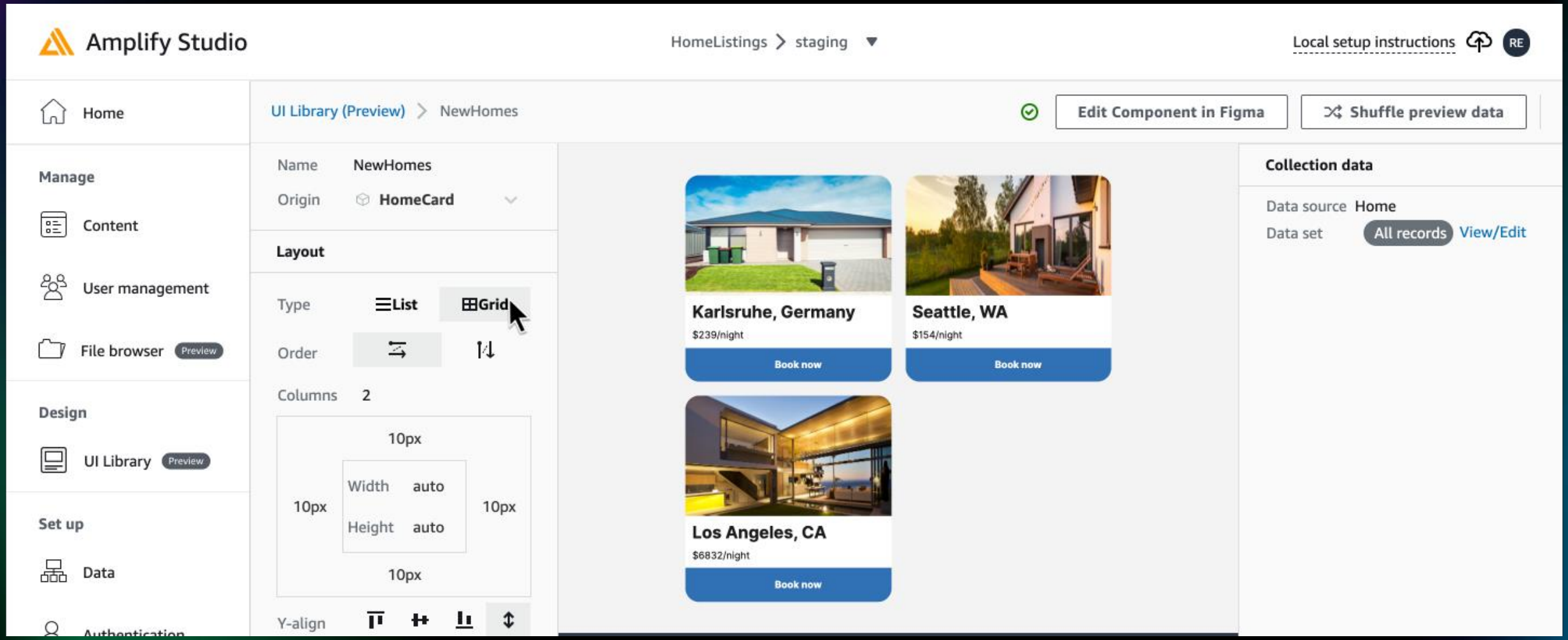
Golden Pathways

Frontend

Frontend Apps with AWS Amplify



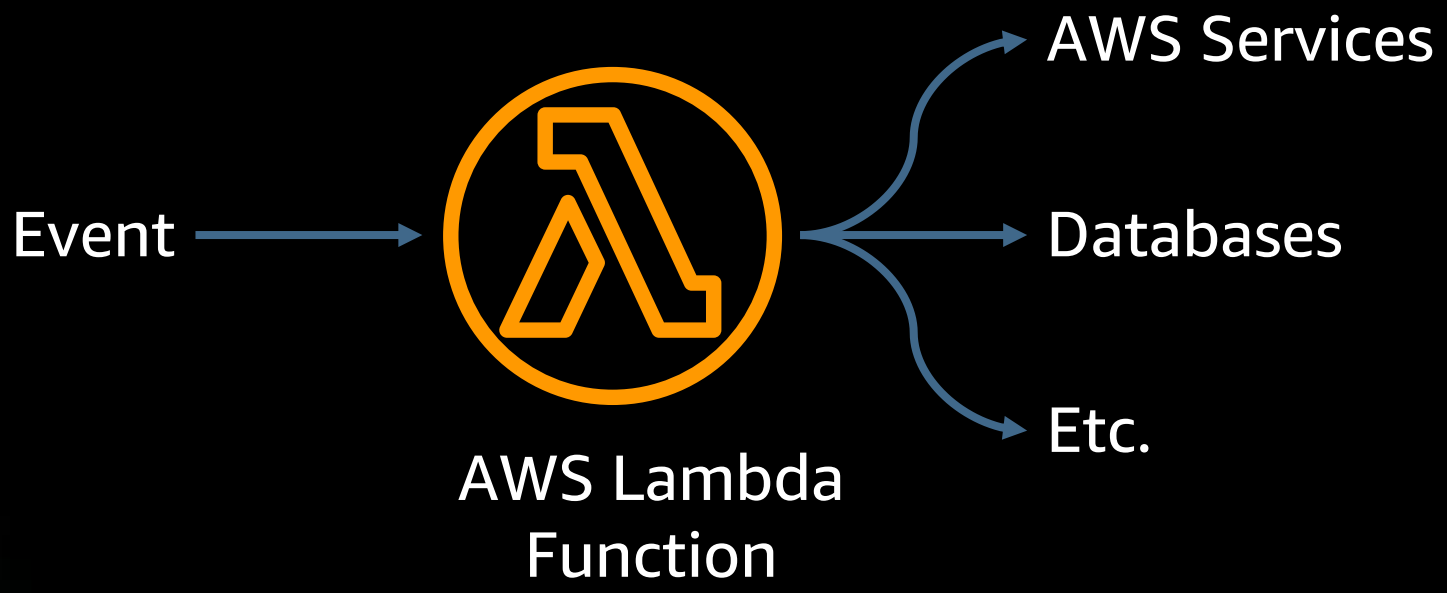
Frontend Apps with AWS Amplify



Golden Pathways

Serverless Backend

Serverless backends with AWS Lambda



Python
Javascript
Java
Golang
C#
BYOL
Container images

Made easy with AWS SAM

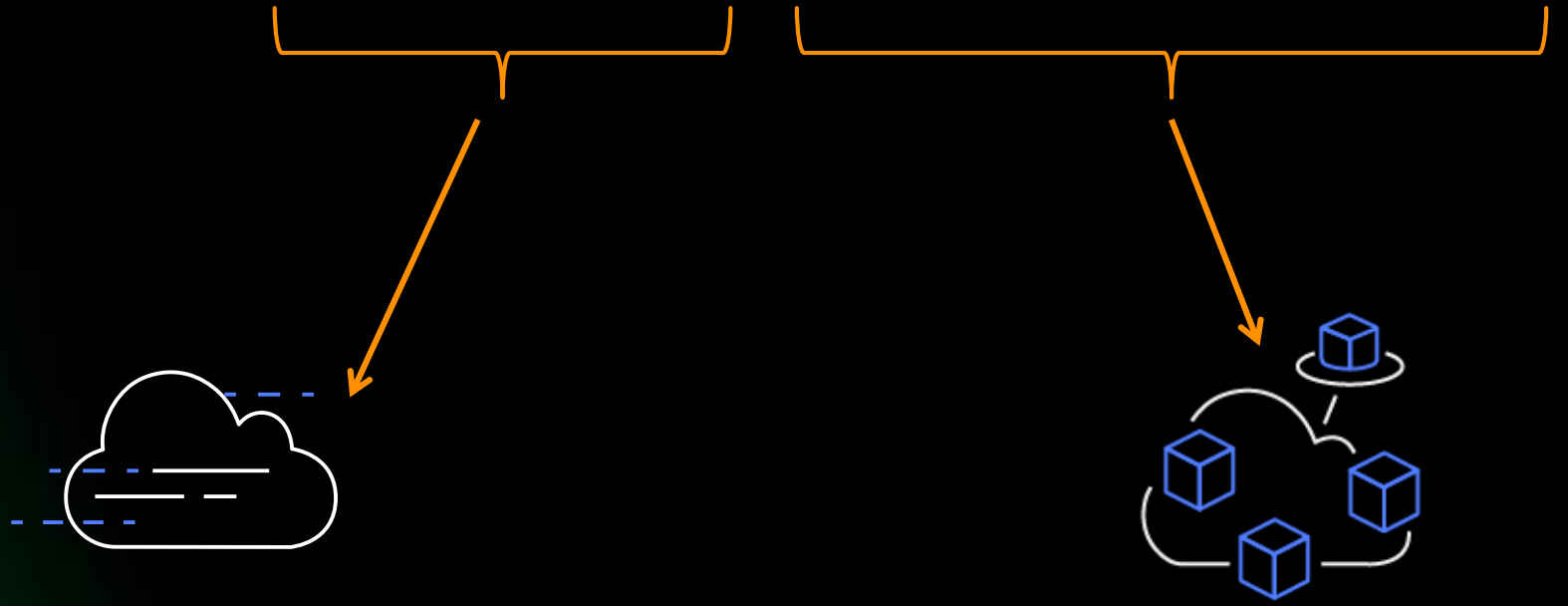
AWS SAM is an open-source framework for building serverless applications. It provides shorthand syntax to express functions, APIs, databases, and event source mappings.

Golden Pathways

Backend

Backend applications with AWS Fargate

Fargate is a serverless compute engine for containers



Managed by AWS, no AMIs to maintain, no EC2 instances to provision, scale or manage, pay for what you need

AWS Fargate is where your containers will run. Customers don’t interact with it directly

Made easy with AWS App Runner

Customer applications

 Amazon ECR public

 GitHub

 Amazon ECR private

 **AWS App Runner**

Frontend API

Orchestration Workflow

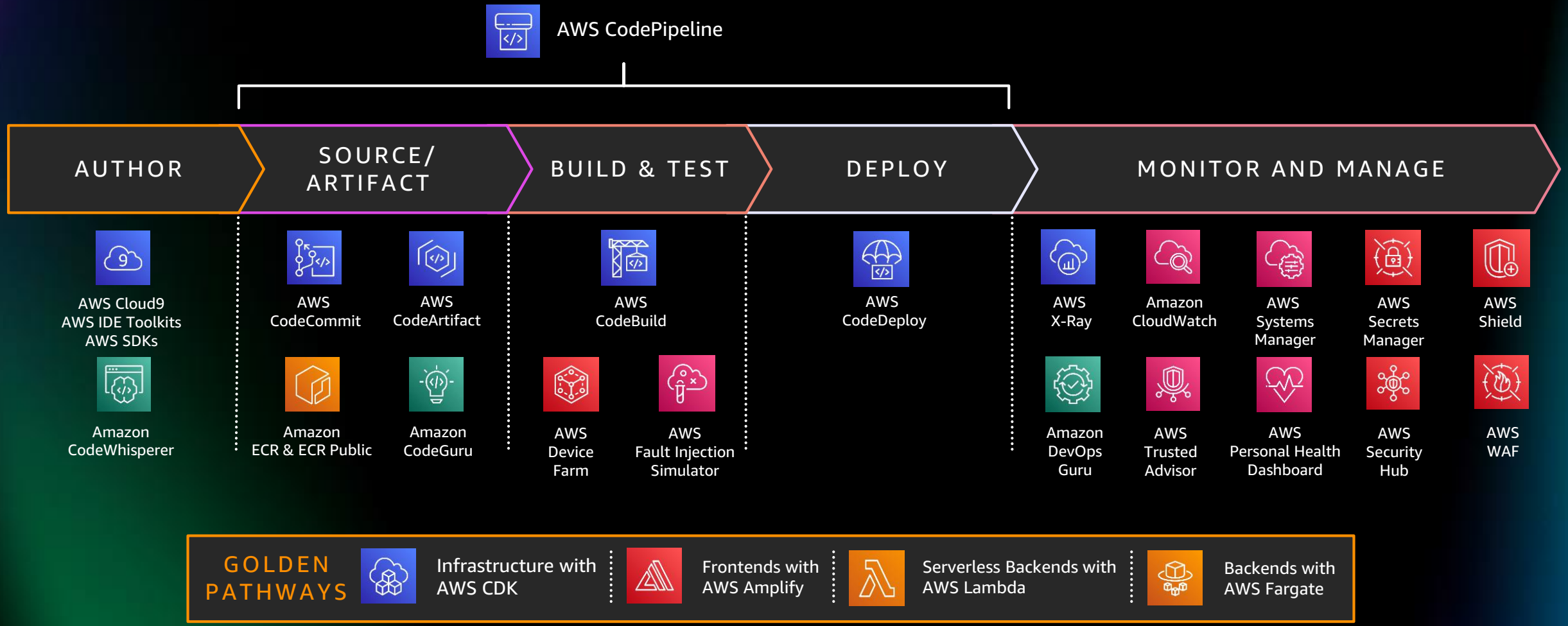
With App Runner, customers don't need to manage…

Build (CB)	Hosting (AWS Fargate, VPC)	Load Balancing & Health Check (Envoy)	Custom Domain (NLB)
Auto Scaling (Request-based)	CI / CD (Webhook, CW Events)	Safe Deployment (Blue-green)	Runtime Patching (Amazon ECR)
Host Patching (AWS Fargate)	Deployment & Application Logs (CW Logs)	Service & Instance Metrics (CW Metrics)	Encryption (AWS KMS)

aws

In Summary

In Summary



Visit the Modern Applications resource hub

Dive deeper with these resources to help you develop an effective plan for your modernization journey.

- Build modern applications on AWS
- Business value of cloud modernization
- An introduction to event-driven architectures
- Accelerate full-stack web and mobile app development
- Determining the total cost of ownership: Comparing serverless and server-based technologies
- Building event-driven architectures with AWS
- Continuous learning, continuous modernization

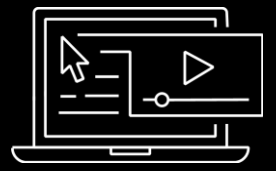
https://tinyurl.com/modern-apps-aws

Visit resource hub

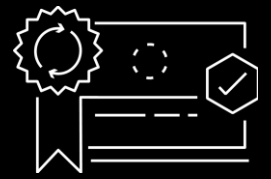
AWS Training and Certification

Get started with Free Digital Training for you and your team today

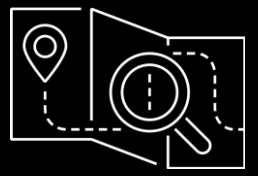
Achieve key milestones and plan your next steps with the AWS Modern Application skills training



Access 500+ free digital courses with AWS Skill Builder



Earn an industry-recognized credential:
AWS Certified Developer – Associate
AWS Certified DevOps – Professional



Create a self-paced learning roadmap
AWS ramp-up guide - Developer
AWS ramp-up guide - DevOps

aws

Thank you for attending AWS Innovate Modern Applications Edition

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-apj-marketing@amazon.com


twitter.com/AWSCloud


facebook.com/AmazonWebServices


youtube.com/user/AmazonWebServices


slideshare.net/AmazonWebServices


twitch.tv/aws

Thank you!