

Continuous Deployment with Amazon Web Services



Julien Simon

Principal Technical Evangelist, AWS

julsimon@amazon.fr

@julsimon



Agenda

- A few definitions
- The DevOps story at Amazon.com
- The Code* services
- Demo #1: CI/CD on C library
- Demo #2: CI/CD on Java web application



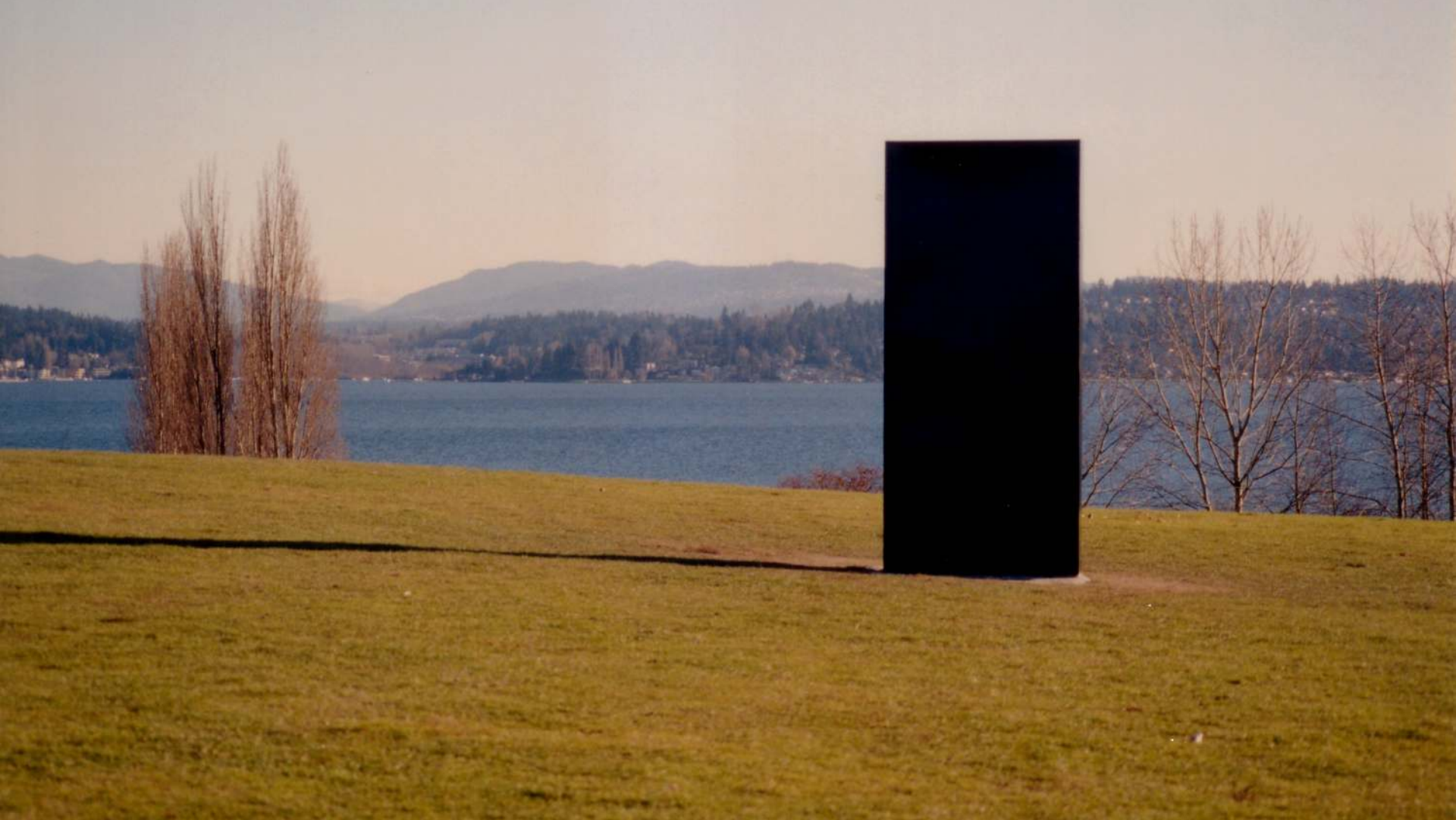
CI, CD, DevOps

Continuous Integration: breaking down system integration into small steps by regularly merging code into a shared mainline and fully testing the updated code automatically

Continuous delivery: a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time.

Continuous deployment: every change is automatically deployed to production

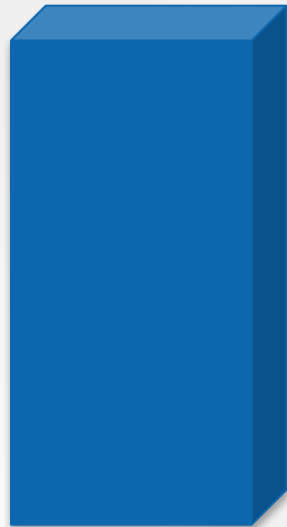
DevOps: a set of practices that emphasize the collaboration and communication of both software and operations while automating the process of software delivery and infrastructure changes.



Monolith development lifecycle



developers



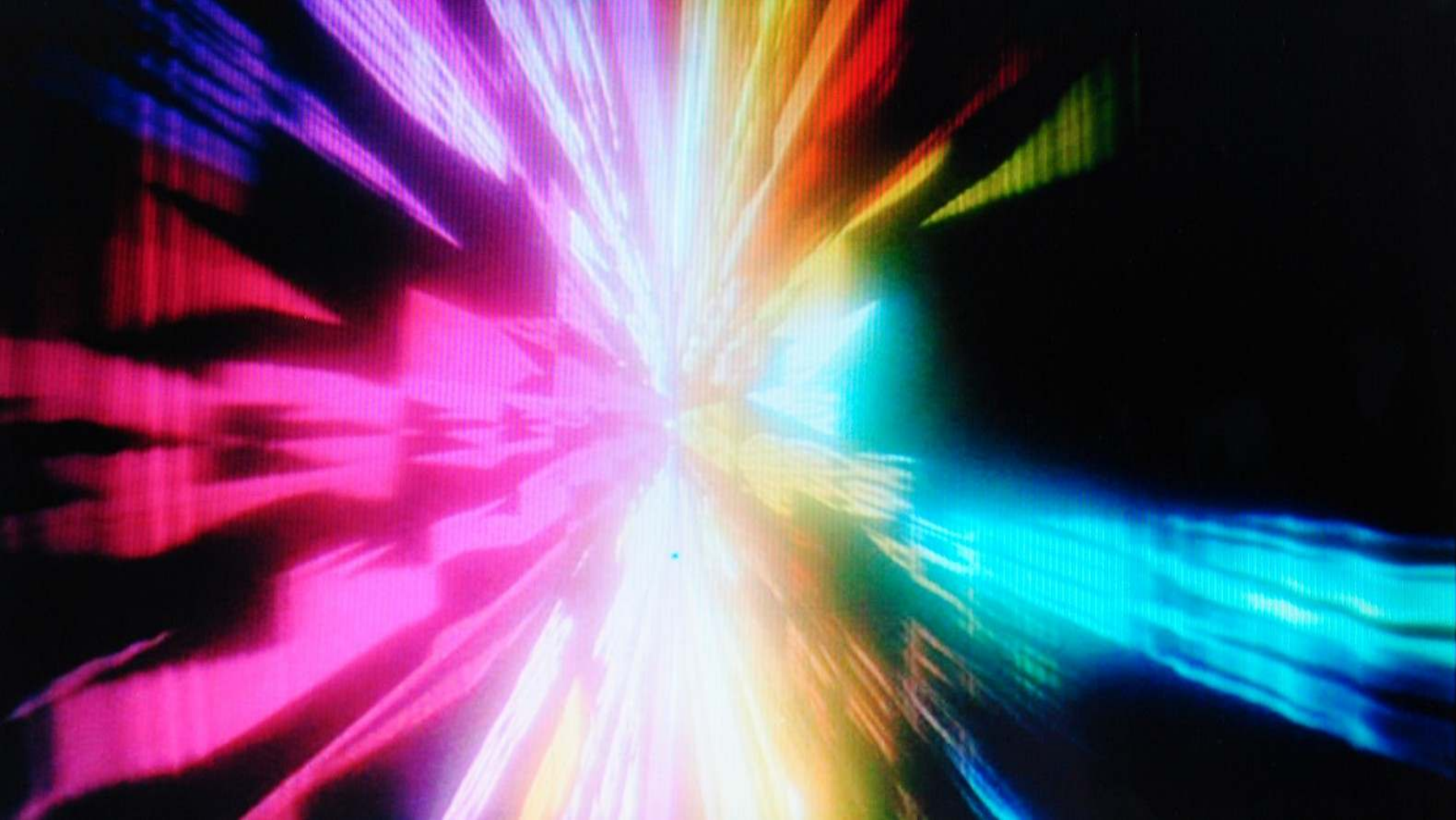
app

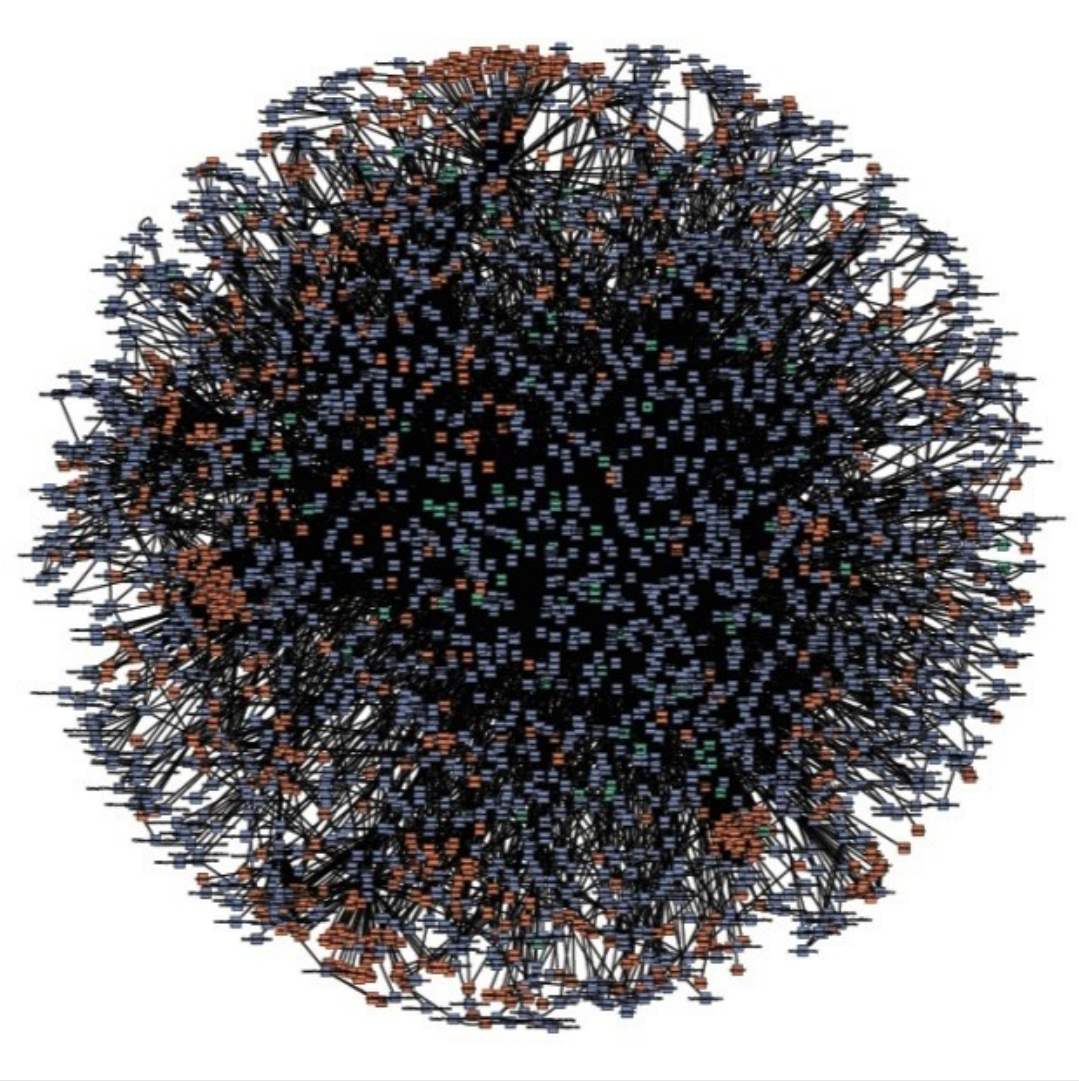


delivery pipeline



IWORM





Service-Oriented
Architecture (SOA)

Single-purpose

Connected through APIs

Highly decoupled

“Microservices”



Two-pizza teams

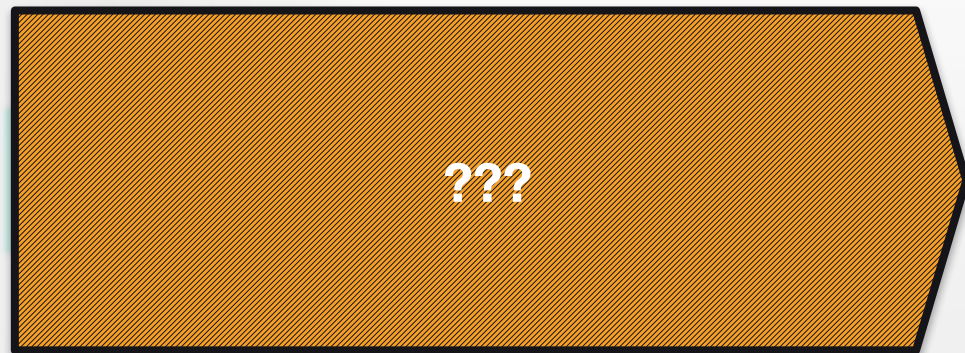
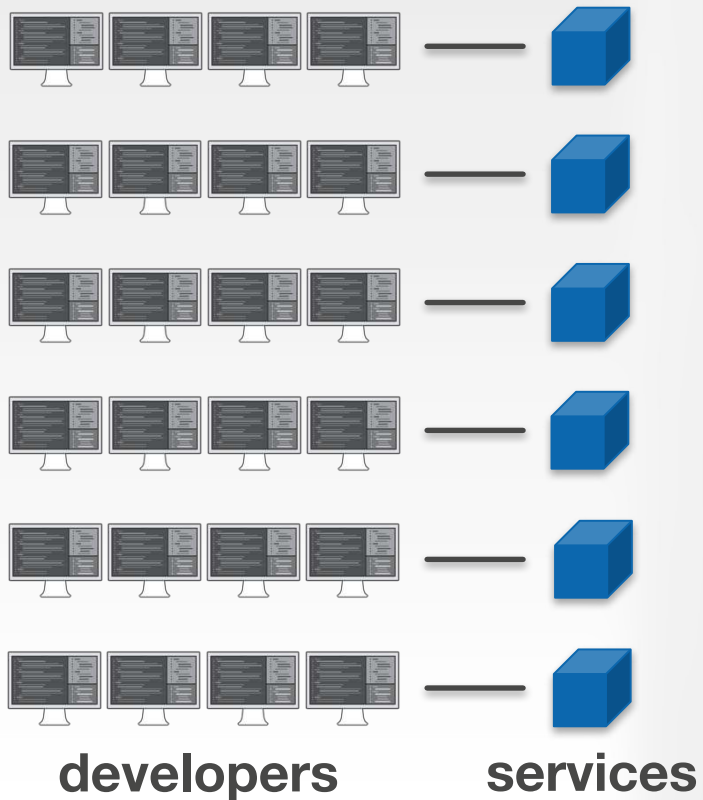
Full ownership

Full accountability

Aligned incentives

“DevOps”

Missing tools



delivery pipeline



Self-service

Technology-agnostic

Encourage best practices

Single-purpose services



Deployment service

No downtime

Health tracking

Versioned artifacts

Rollbacks



Pipelines

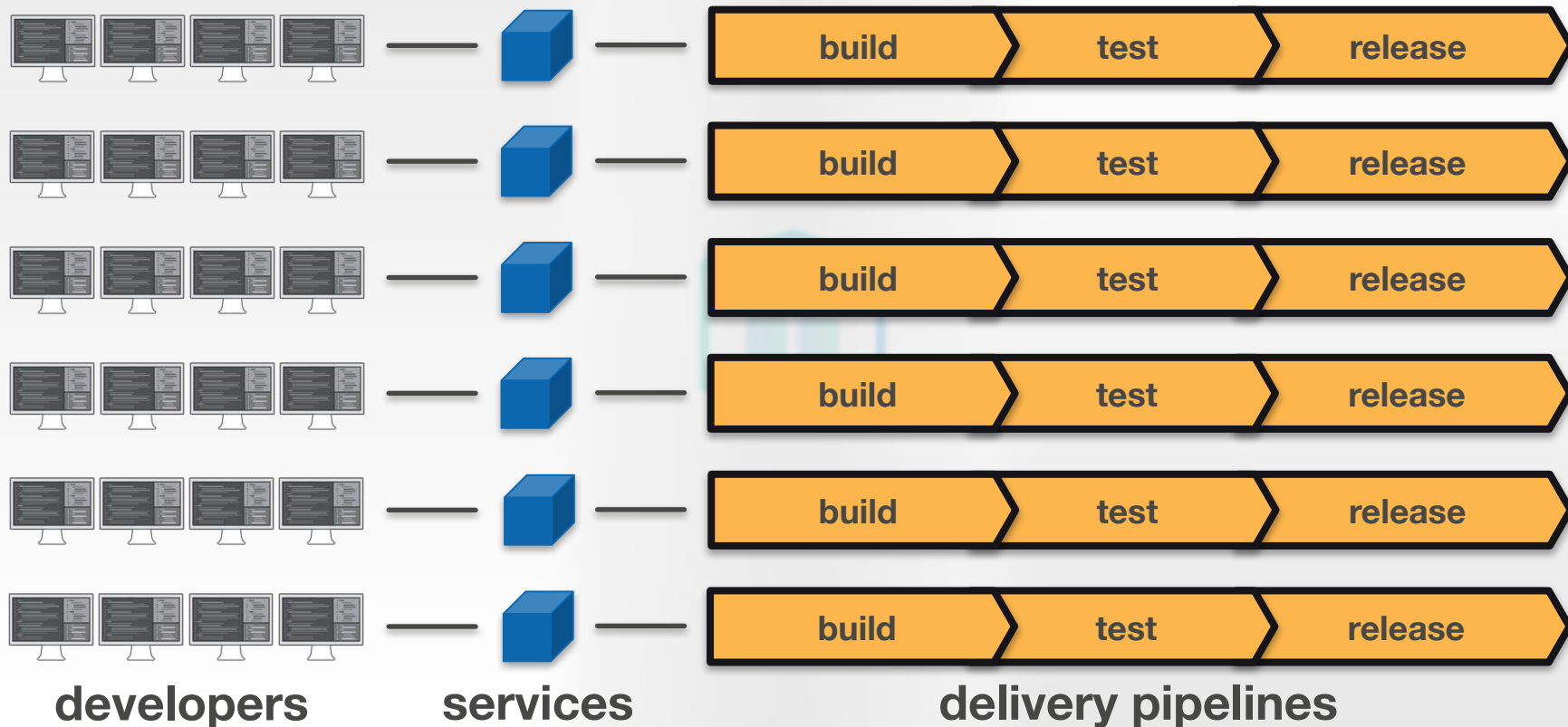
Continuous delivery

Automated release
process

Faster and more reliable
releases

Used by >90% of teams

DevOps development lifecycle





Thousands of teams
× Microservice architecture
× Continuous delivery
× Multiple environments

= 50 million deployments a year
(1.5 deployment every second)



How can we help others do this?



Setting up a delivery pipeline

AWS

release

Source



AWS
Code
Commit

Build



AWS
Code
Build

Testing



Staging



Production



AWS

AWS Code* partners

GitHub

Atlassian

circleci

CHEF™

Jenkins

CloudBees

Travis CI

puppet
labs

Solano Labs

CODESHIP

A

ANSIBLE

Apica

Runscope

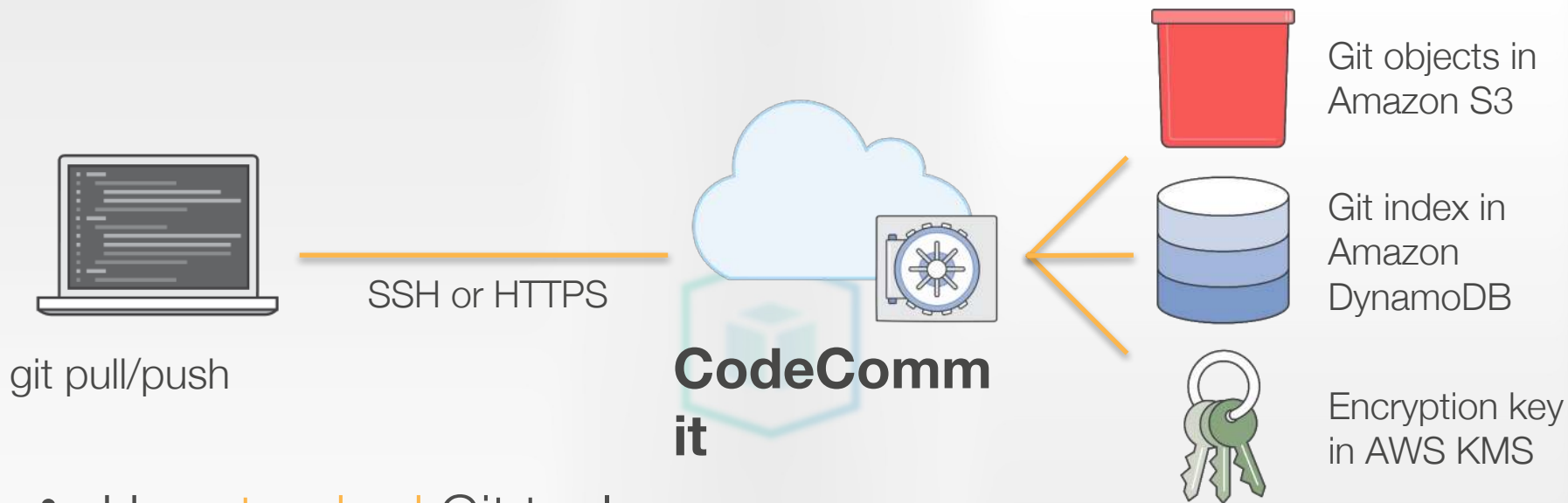
XebiaLabs
Deliver Faster

BlazeMeter

Ghost Inspector

SALTSTACK

AWS CodeCommit

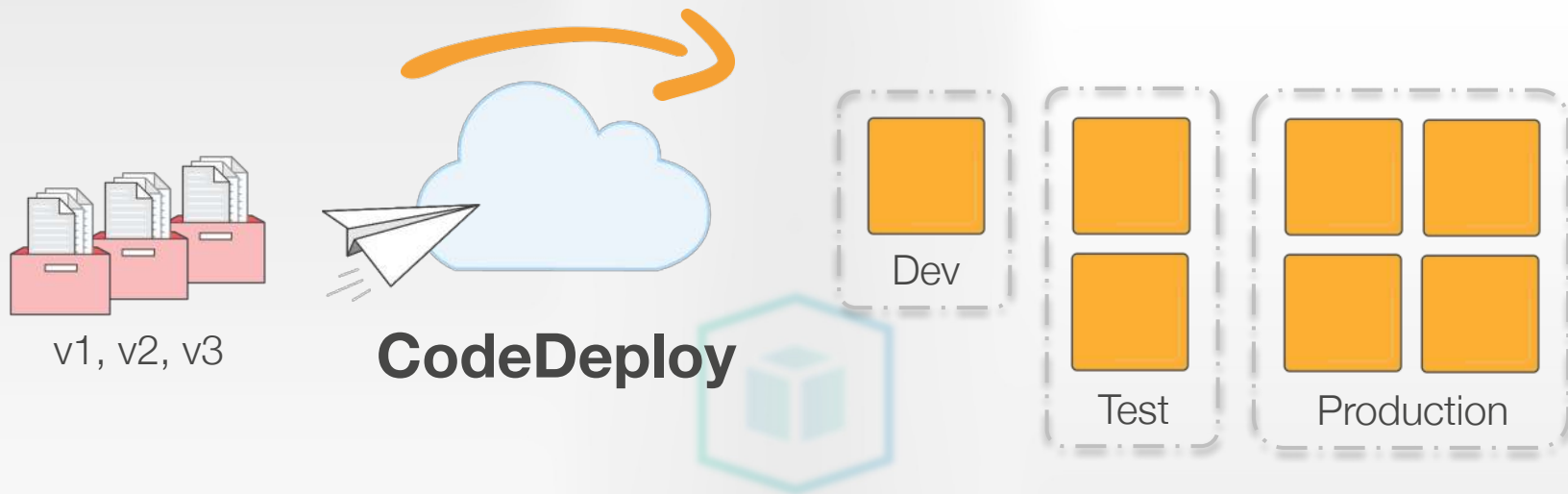


- Use **standard** Git tools
- Scalability, availability and durability of **Amazon S3**
- **Encryption** at rest with customer-specific keys
- Pricing: first 5 users free, then \$1 / user / month

AWS CodeBuild

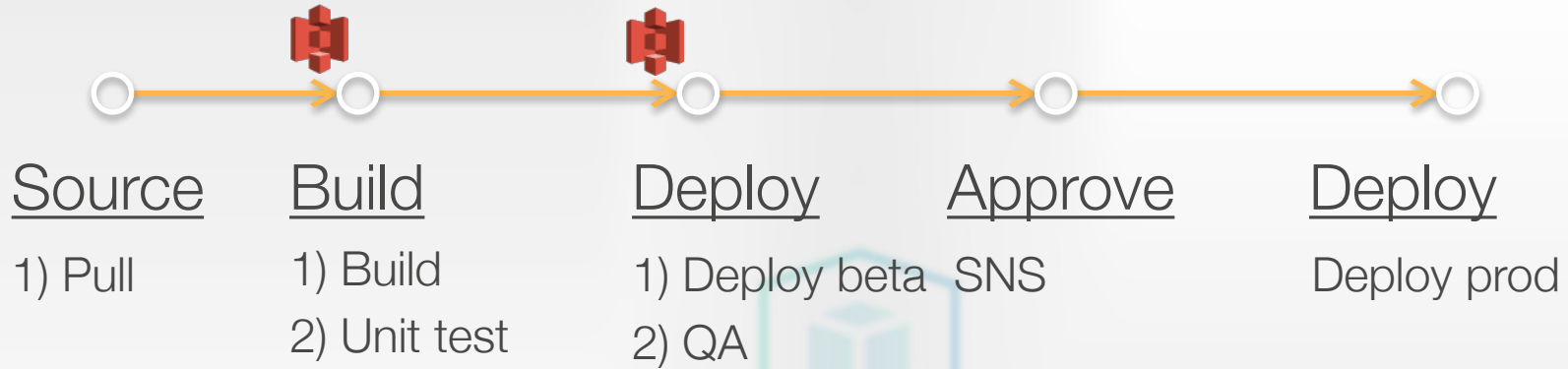
- New service launched at re:Invent 2016
- Managed build environments (Linux only for now)
- Pull sources from Github, S3 or CodeCommit
- Build on an AWS-provided image or your Docker container
- Supported environments : “base”, Android, Java, Go, Python, Ruby, Go, Docker
- Build commands: inline or in buildspec.yml file
- Pricing starts at \$0.005 per minute (free tier available)

AWS CodeDeploy



- **Easy** and **reliable** deployments: zero downtime, rollbacks
- Deploy to **any** server with agent: Linux / Windows, EC2 / on-premise
- **Scale** with ease: Auto Scaling groups supported
- **Green-blue deployment** supported
- Pricing : no extra charge for EC2

AWS CodePipeline



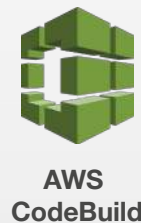
- Define **stages**: Source, Build, Test, Deploy, Invoke, Approve
- Connect to **best-of-breed** tools
- Build a **fast, consistent and traceable** release process
- S3 is used to store source and build **artefacts**
- Pricing: \$1 / active pipeline / month



Demo #1

The Pipeline

- **Source:** CodeCommit repository
 - C sources (library for generic data structures)
 - `buildspec.yml`: config file for CodeBuild
 - `appspec.yml`: config file for CodeDeploy
 - Also here: <https://github.com/juliensimon/data-structures>
- **Build:** CodeBuild 'base' environment
 - Install dependencies
 - Build libraries
 - Run unit tests (Cunit)
- **Deploy:** CodeDeploy
 - One deployment group with a single instance
 - Copy libraries to `/usr/local/lib`



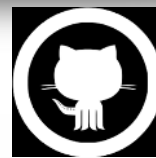
Demo #2



Expanded version of <http://blogs.aws.amazon.com/application-management/post/Tx2CIB02ZO05ZII/Explore-Continuous-Delivery-in-AWS-with-the-Pipeline-Starter-Kit>

AWS Code* demo

Source (GitHub) → Build (Jenkins) → Deploy Dev (CodeDeploy) → Approve (SNS Email) → Deploy Prod (CodeDeploy)



Code
+ appspec.yml
+ scripts



CloudFormation



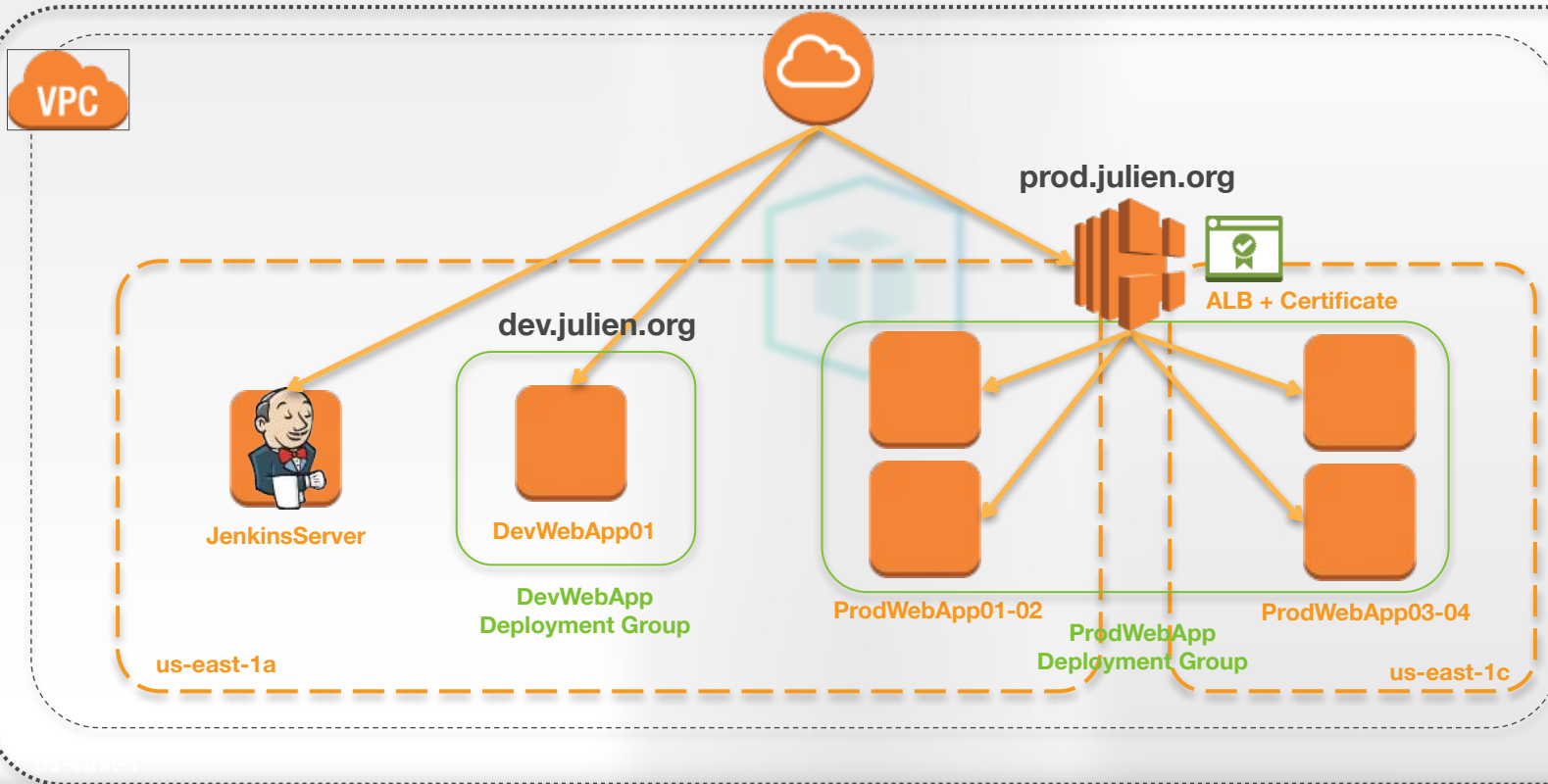
CodePipeline



CodeDeploy



SNS



APPROVAL NEEDED: AWS CodePipeline app-name-Pipeline for action My_Approval



✉ Email JS

11:15



Hello,

The following Approval action is waiting for your response:

--Pipeline Details--

Pipeline name: app-name-Pipeline

Stage name: Approval

Action name: My_Approval

Region: us-east-1

--Approval Details--

Content to review: <http://dev.julien.org>

Approve or reject: https://console.aws.amazon.com/codepipeline/home?region=us-east-1#/view/app-name-Pipeline/Approval/My_Approval/approve/0bba2e19-4c19-4a74-87de-cdc1f9612613

Additional information: Please review this deployment

Deadline: This review request will expire on 2016-10-12T09:15Z

Sincerely,

Amazon Web Services

Typical issues you may encounter 😊

- EC2 instances are not allowed to access build artefacts in S3 bucket
→ check IAM role
- CodeDeploy agent is either not installed or not running
→ install it or start it
- appspec.yml and scripts are not included in build artefact
→ check that they're included in buildspec.yml
- Not enough EC2 instances are running to satisfy health constraints of the deployment configuration
→ Add more instances or relax health constraints
- How to debug:
 - Check artefacts in S3 bucket (file names, directory structure, etc.)
 - Read deployment logs stored in CloudWatch Logs
 - <https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting.html>

Closing words

- Automation is a **key factor** in technical & business agility
- You can use the same tools as **Amazon.com!**
- **Zero** dev infrastructure to purchase & manage
- **Minimal** cost
- Compatible with your **existing** CI/CD tools

Next steps

Learn more

<http://aws.amazon.com/awsgcode>

<http://blogs.aws.amazon.com/application-management>

<https://aws.amazon.com/fr/new/#dev-tools>

Get started

<http://aws.amazon.com/free>

<http://console.aws.amazon.com>

AWS User Groups



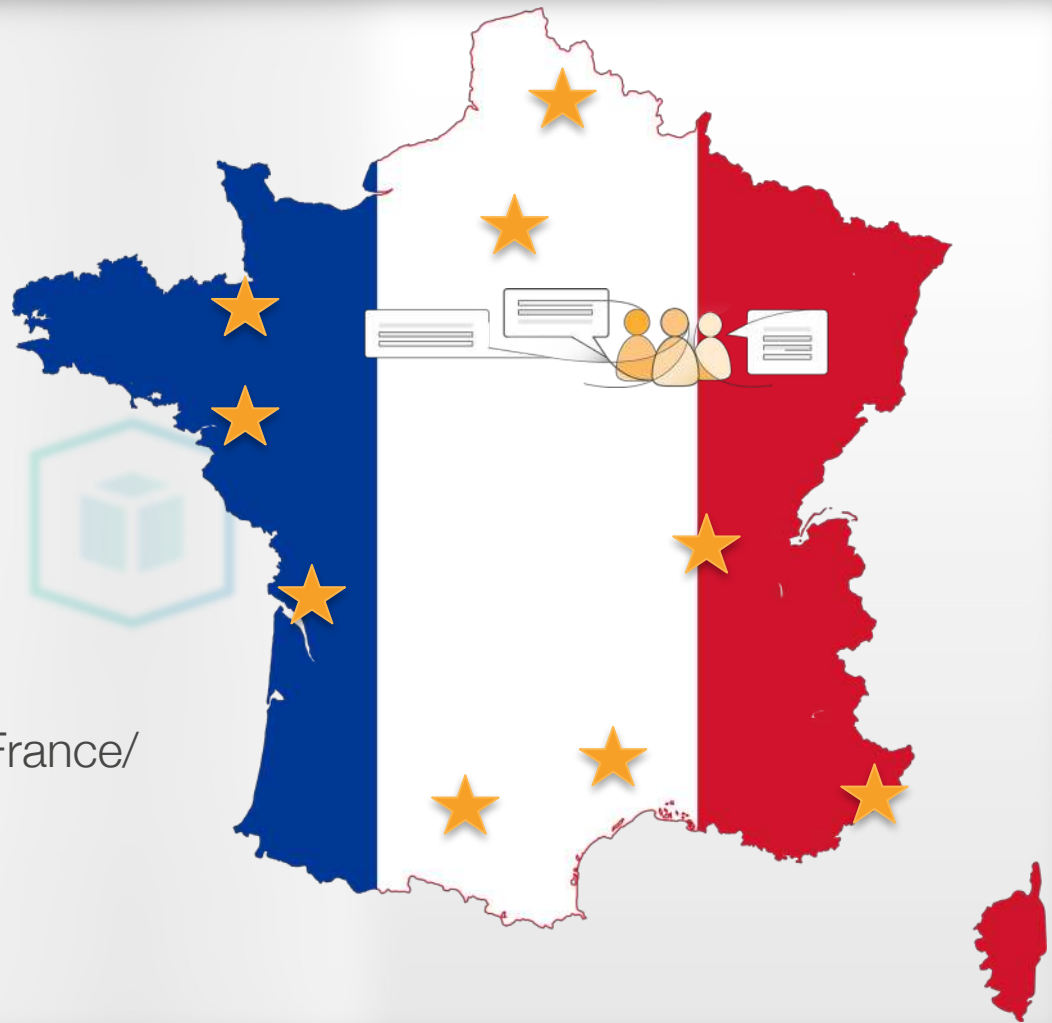
Lille
Paris
Rennes
Nantes
Bordeaux
Lyon
Montpellier
Toulouse
Côte d'Azur (new!)



facebook.com/groups/AWSFrance/



[@aws_actus](https://twitter.com/aws_actus)



<https://aws.amazon.com/fr/events/webinaires/>

Mars	Avril
Mardi 28 mars - 15h30	Mardi 25 avril - 15h30
EC2, pas juste des instances (en savoir plus)	Les bases de données relationnelles (en savoir plus)

Merci !



Julien Simon

Principal Technical Evangelist, AWS

julsimon@amazon.fr

@julsimon