DCD-T5

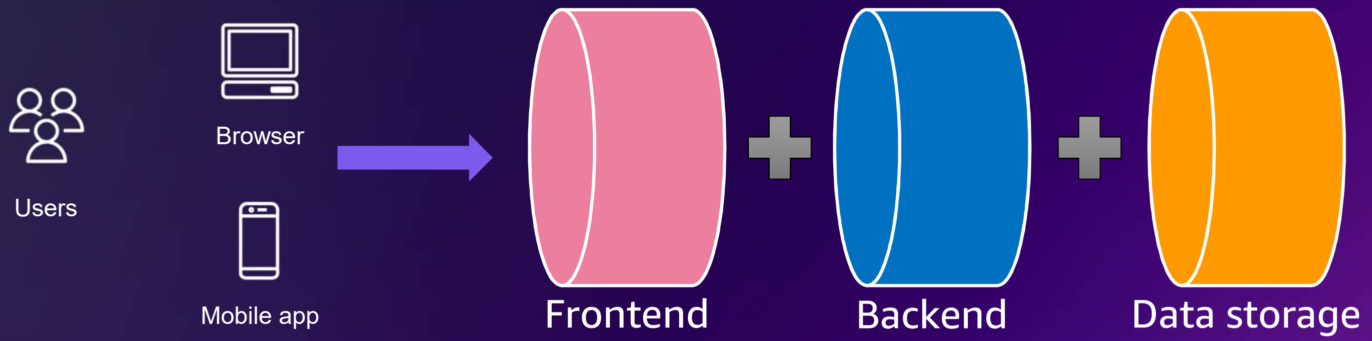
Scaling up to your first 10 million users

Damian Niezgoda
Senior Solutions Architect
Amazon Web Services

Fabrizio Fortunato
Head of Frontend
Ryanair Labs

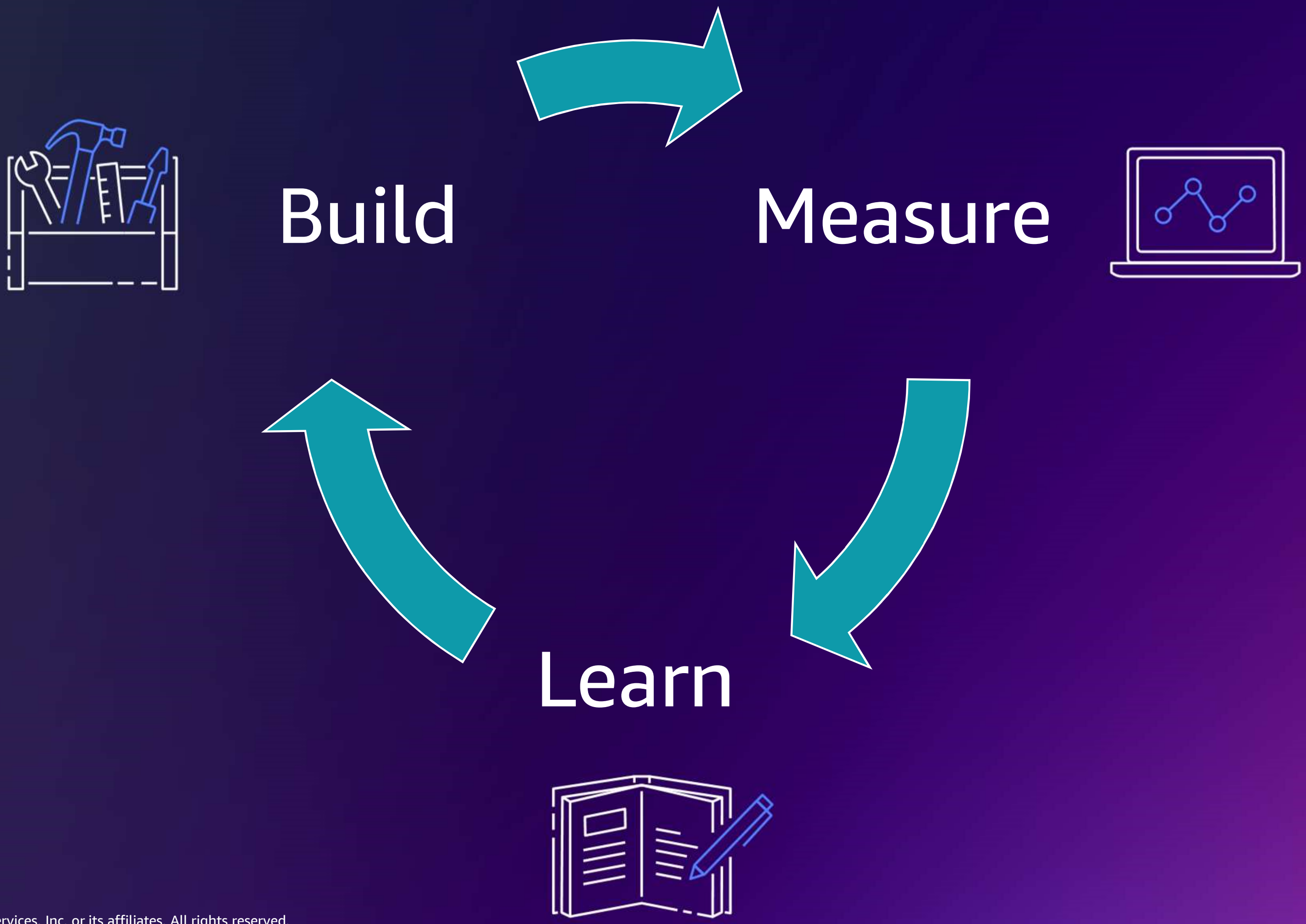
What do we mean by an "app"

- ➢ Could be the entirety of a startup's product
- ➢ Could be one of many products in a larger company
- ➢ For today: App == user-interfacing layer + business logic layer + data storage



How do I … at scale ?

- Manage users
- Maintain performance (incl multiple geos)
- Detect and respond to incidents
- Maintain business continuity
- Manage security and compliance
- Develop and test
- Manage change
- Track and manage costs
- Optimize for cost
- Minimize my carbon footprint

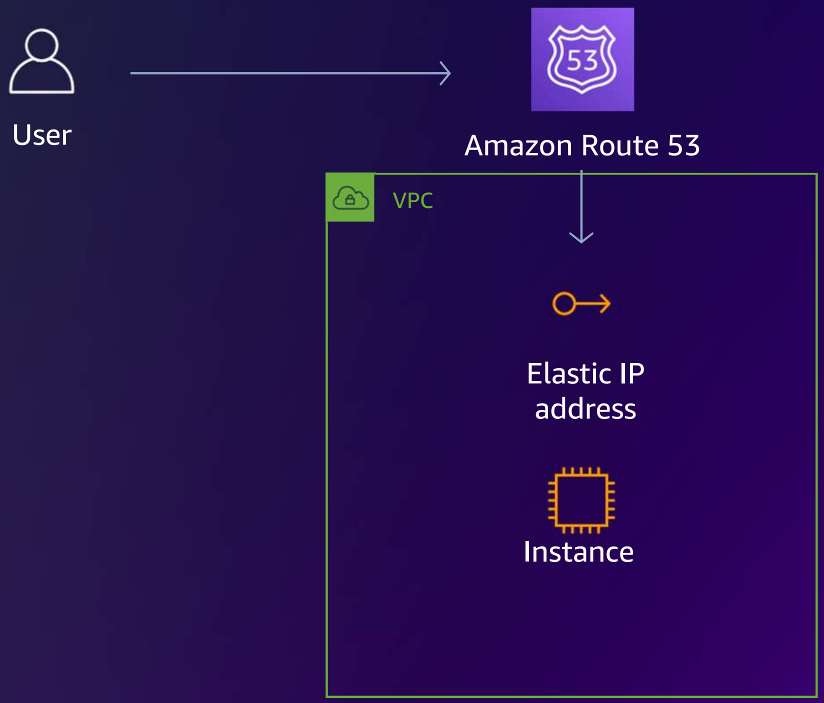


Build

Measure

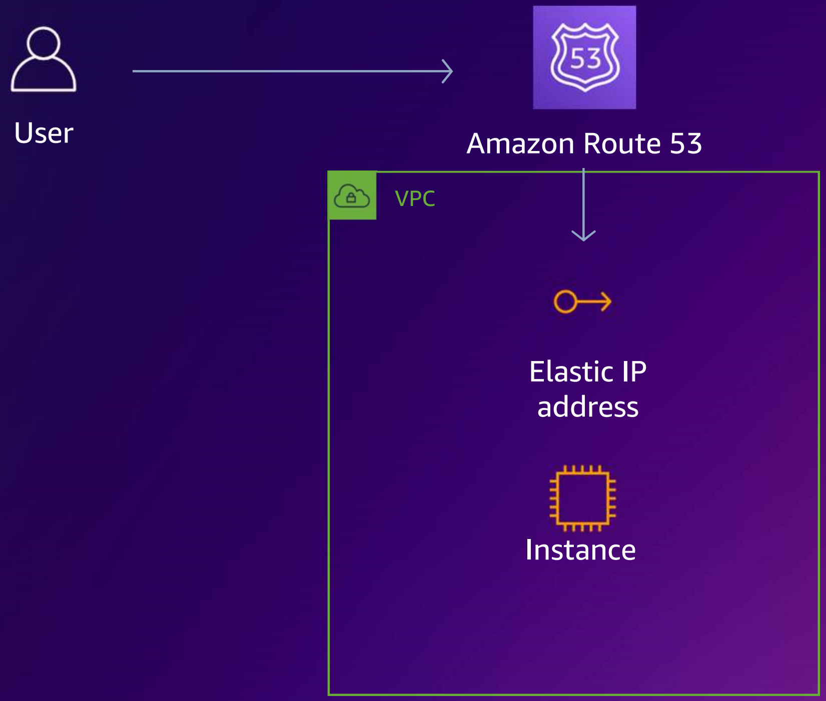
Learn

Day 1 / User 1 : Developer / PoC



Single instance

- No **failover**
- No **redundancy**
- Can't **scale individual components** independently
- Constrained on **technology choices** for individual components



AWS's guidance: make use of managed compute for your backend and managed databases for your data tier

Evaluating managed compute on AWS

More opinionated

	AWS manages	Customer manages
AWS Lambda Serverless functions	• Data source integrations • Physical hardware, software, networking, and facilities • Provisioning	• Application code
AWS Fargate Serverless containers	• Container orchestration, provisioning • Cluster scaling • Physical hardware, host OS/kernel, networking, and facilities	• Application code • Data source integrations • Security config and updates, network config, management tasks
ECS/EKS Container-management as a service	• Container orchestration control plane • Physical hardware software, networking, and facilities	• Application code • Data source integrations • Work clusters • Security config and updates, network config, firewall, management tasks
EC2 Infrastructure-as-a-Service	• Physical hardware software, networking, and facilities	• Application code • Data source integrations • Scaling • Security config and updates, network config, management tasks • Provisioning, managing scaling, and patching of servers

Less opinionated

Exposing business logic to the frontend

THREE OPTIONS FOR EXPOSING AN API

Amazon API Gateway

Application Load Balancer

AWS AppSync

To NoSQL, or not to NoSQL?

Why start with SQL?

- Established and well-known technology
- Lots of existing code, communities, books, and tools
- You aren't going to break SQL databases with your first millions of users.
  No, really, you won't.*
- Clear patterns to scalability

*Unless…

Why else might you need NoSQL?

- Super low-latency applications
- Metadata-driven data sets
- Highly nonrelational data
- Need schema-less data constructs
- Rapid ingestion of data
- Massive amounts of data

Database options

Self-managed

Amazon EC2

Fully managed

Amazon RDS

Amazon DynamoDB

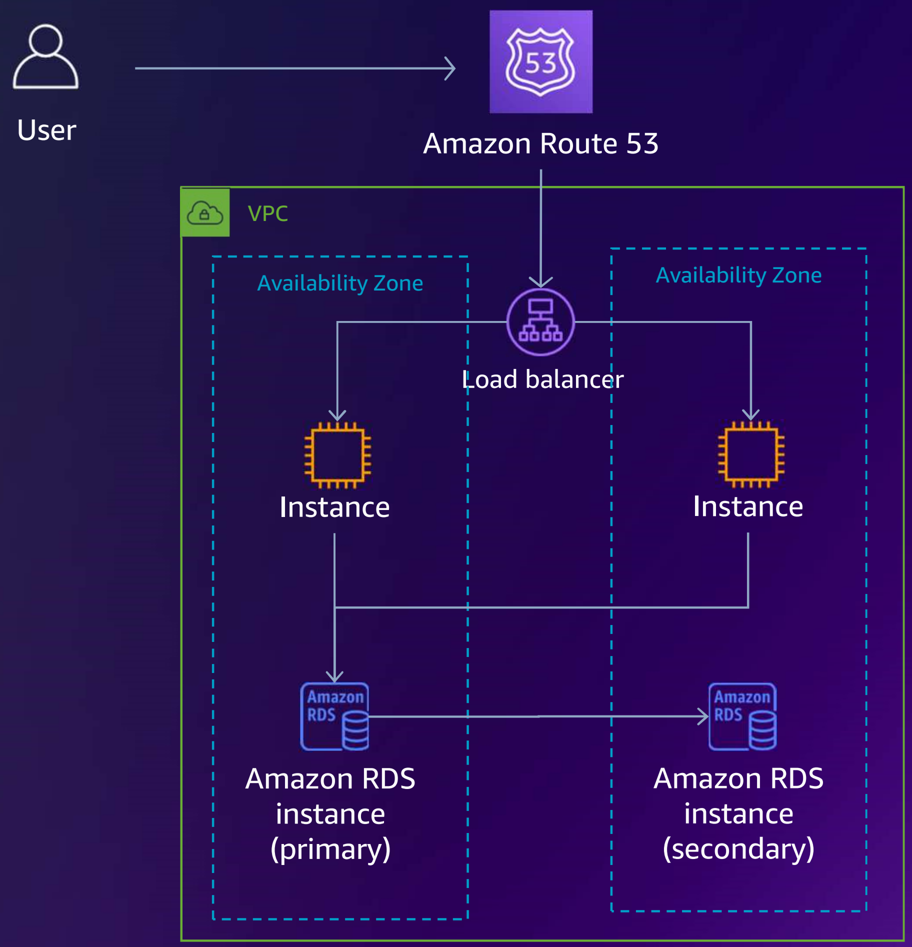
Amazon Neptune

Amazon Aurora

Amazon Timestream

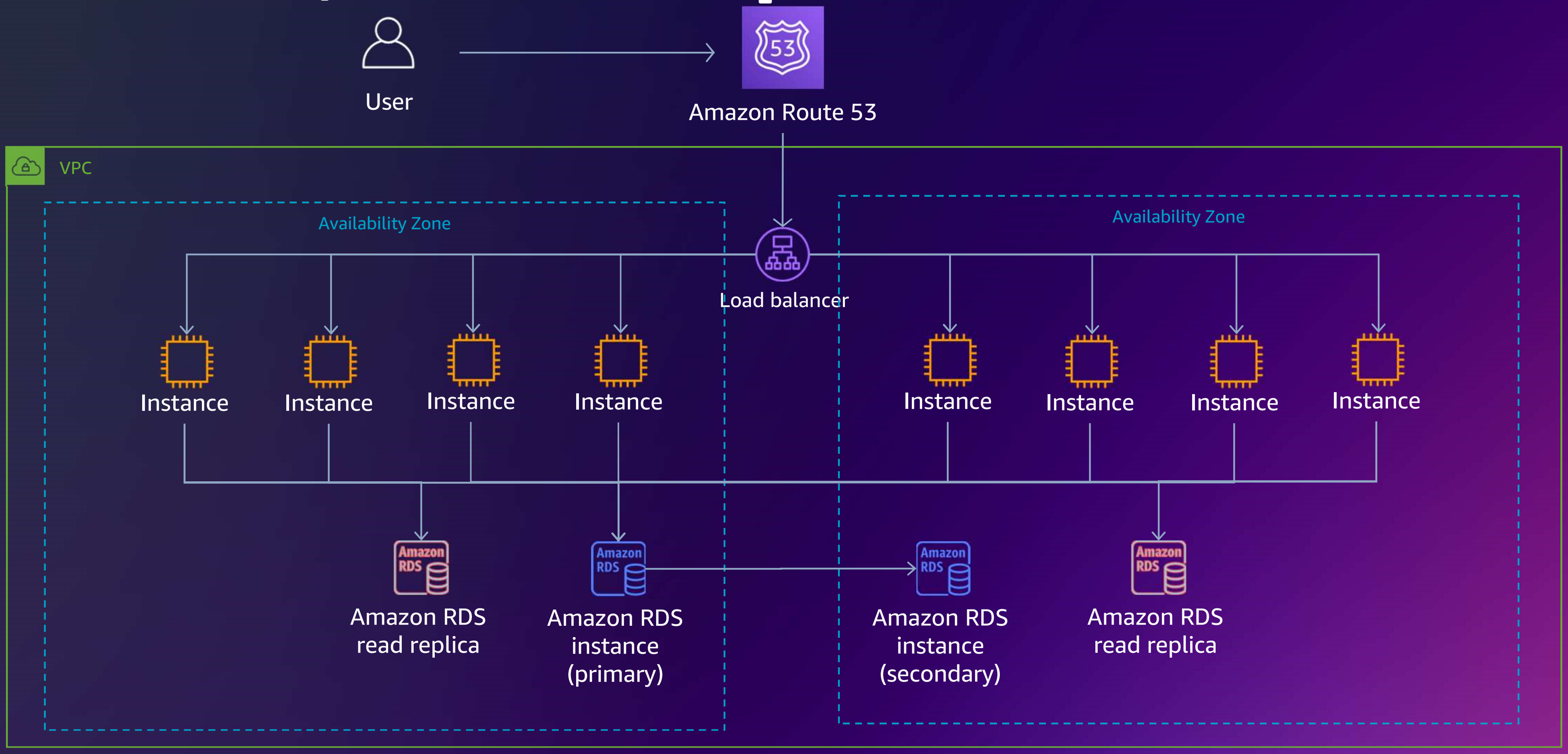
and more…

1 < Users < 1000

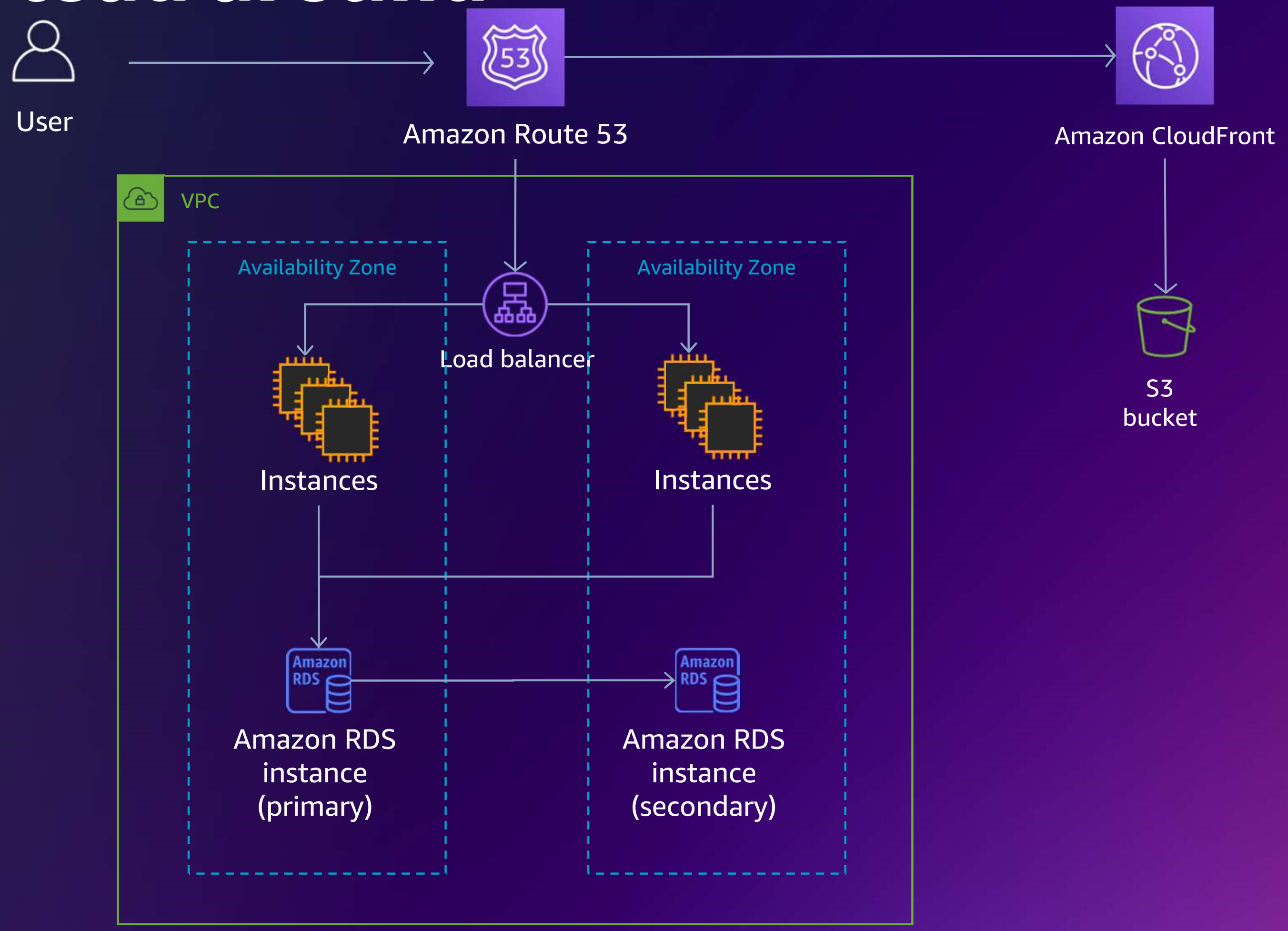

User
Amazon Route 53
VPC
Availability Zone
Availability Zone
Load balancer
Instance
Instance
Amazon RDS
Amazon RDS
Amazon RDS instance (primary)
Amazon RDS instance (secondary)

Users >10,000

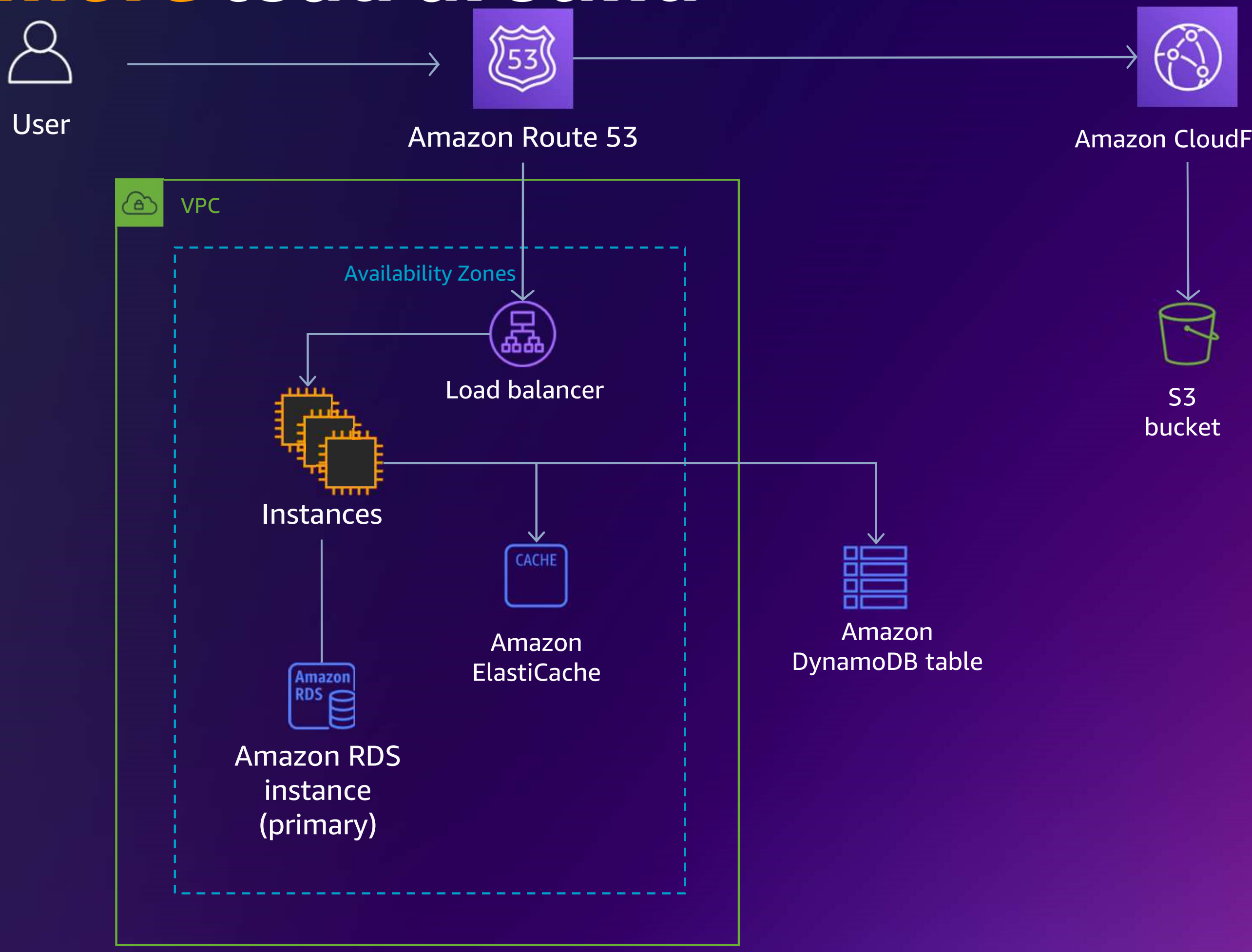
Users >10,000 : Read replicas



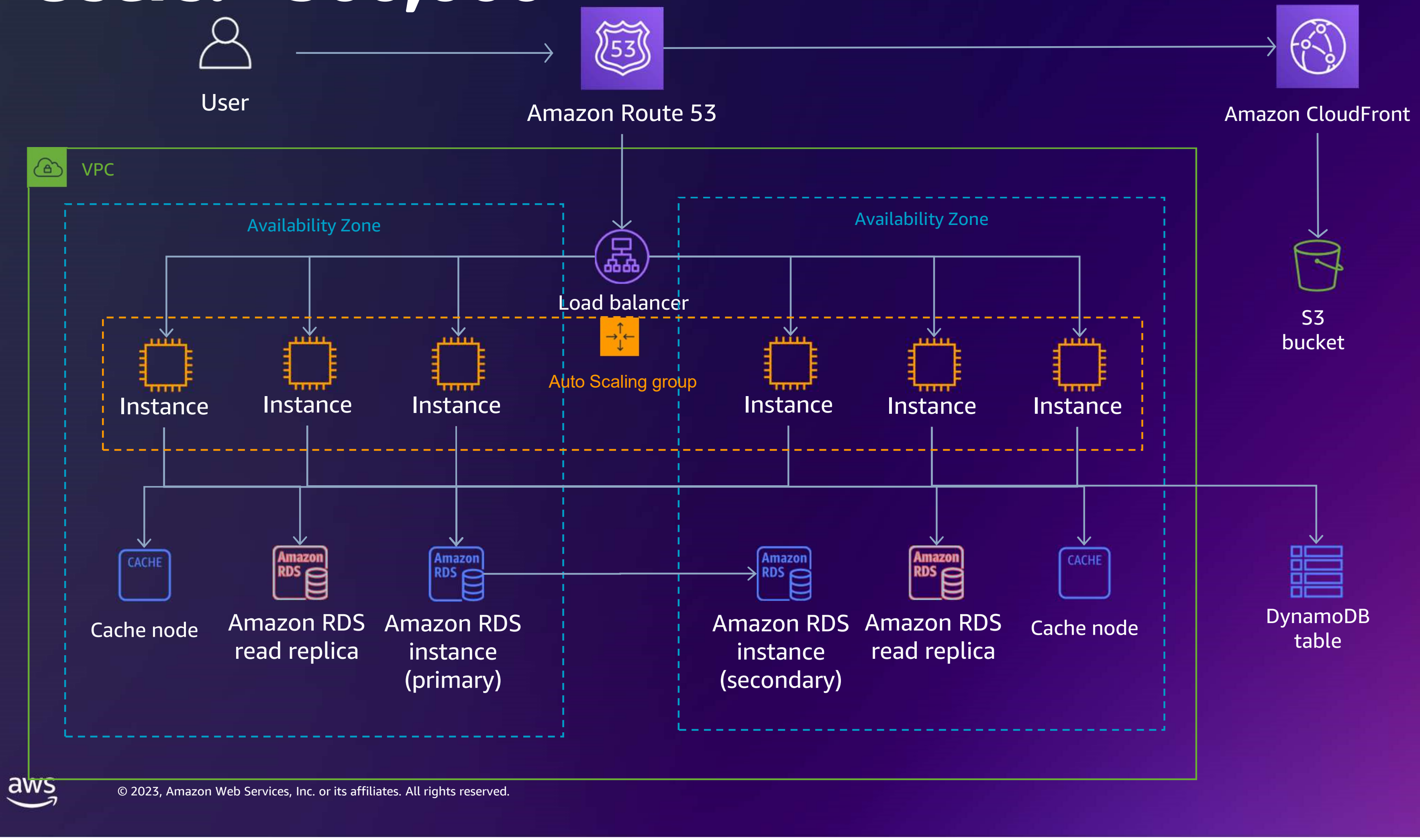
Shift some load around



Shift *even more* load around



Users: >500,000



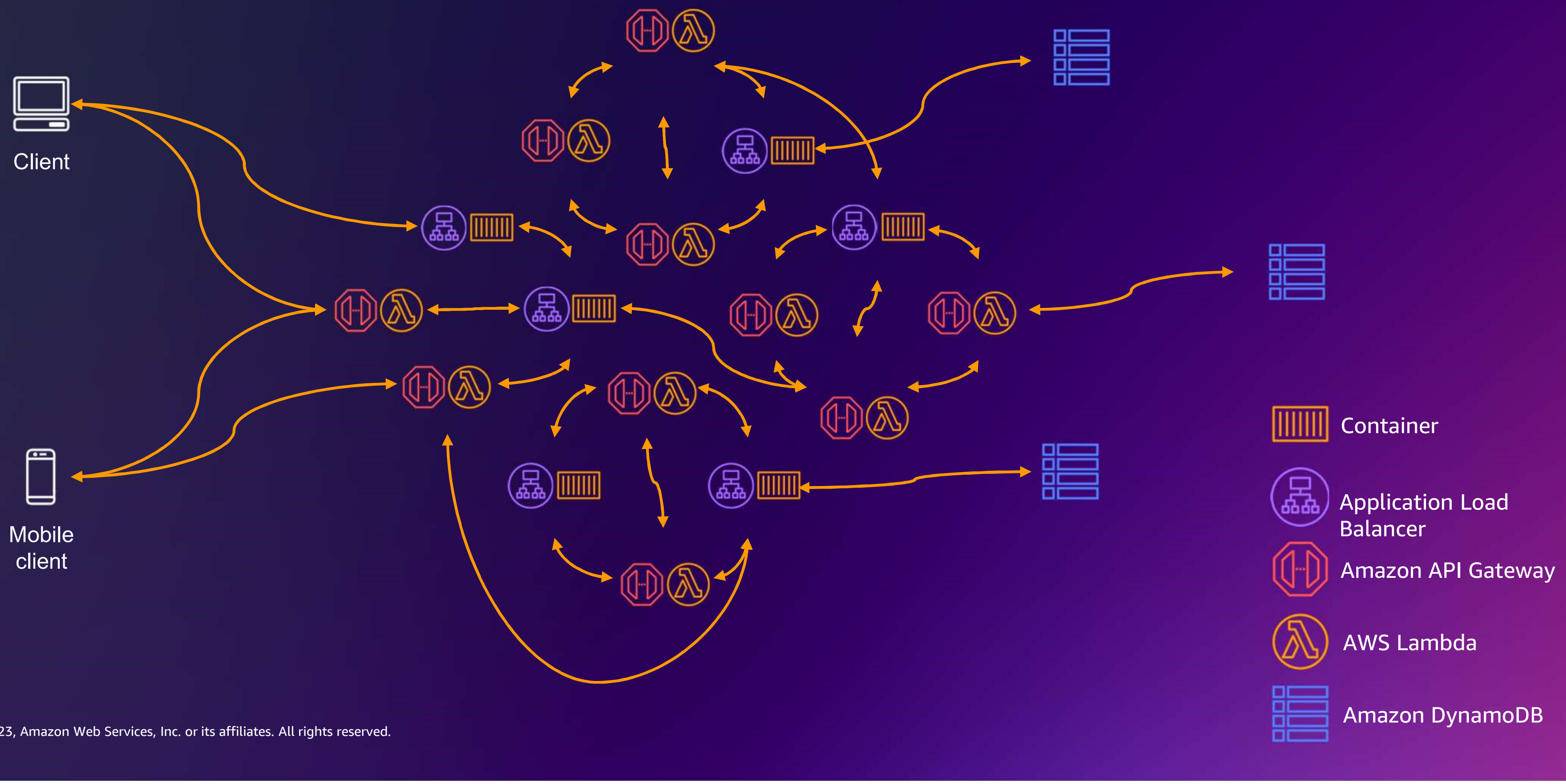
Beyond 1 million

Going the microservices route

Moving to a service-oriented or microservices based architecture is a refactor that requires deep planning across all layers.

- Start with with the easiest to cut away features/capabilities that don't involve too many cross-function ties
  - Data domain mapping
  - Business function mapping
- Good time to evaluate other compute technologies for specific needs
- Will need to think about how to "glue" everything together

Microservices architecture on AWS



To infinity . . .

Head of Frontend at Ryanair Labs

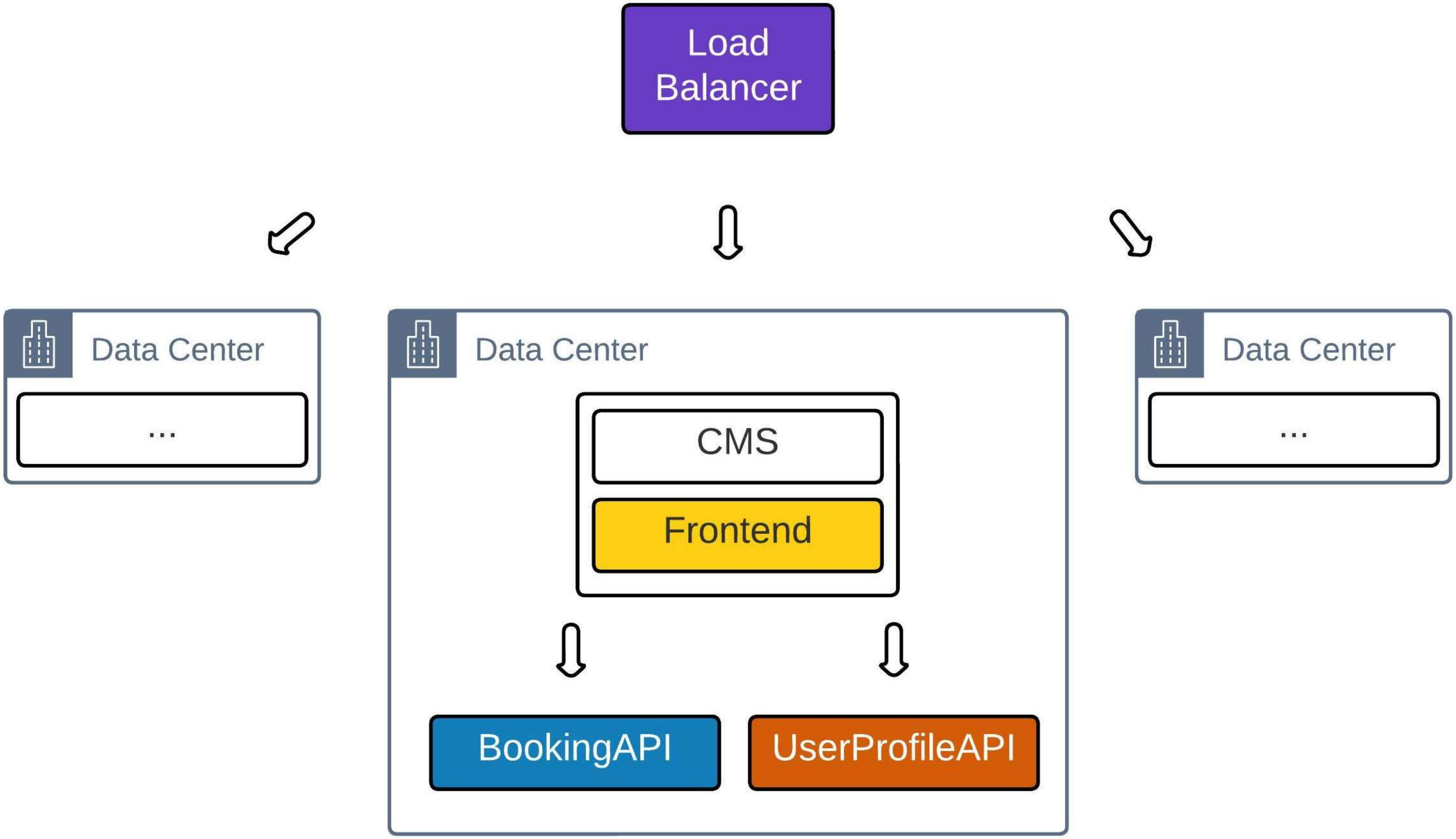
@izifortune

https://izifortune.com

RYANAIR LABS

DUBLIN · WROCŁAW · MADRID · LISBON

RYANAIR



Once upon a datacenter

Frontend/Devops

- DevOps infrastructure and pipeline responsibilities
- Frontend limited responsibilities

RYANAIR

Monolith SPA

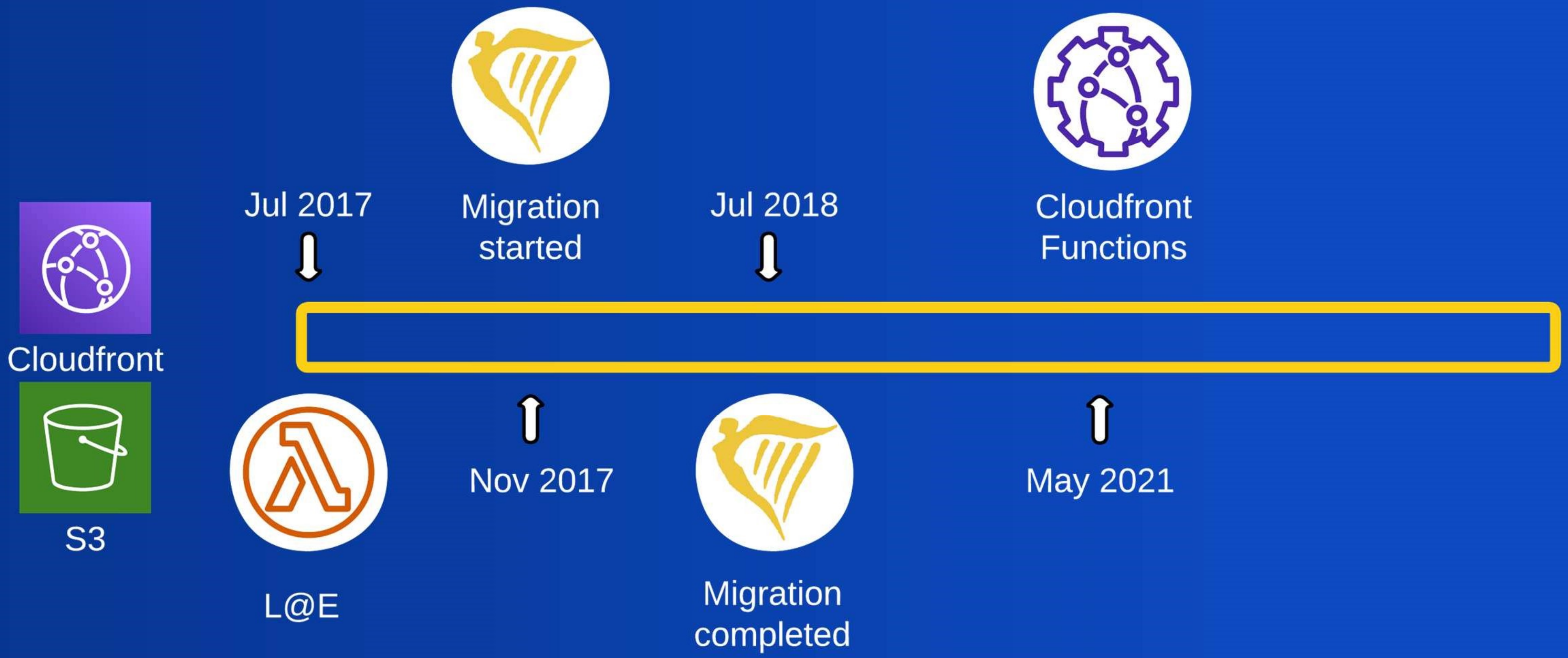
- Application vs website
- Dynamically rewrite current webpage
- 75+ pages

RYANAIR

Challenges

- Limited infrastructure visibility
- No segregation across pages
- Release downtimes
- Performance improvements

RYANAIR



Cloudfront

S3

Jul 2017

L@E

Migration started

Nov 2017

Jul 2018

Migration completed

May 2021

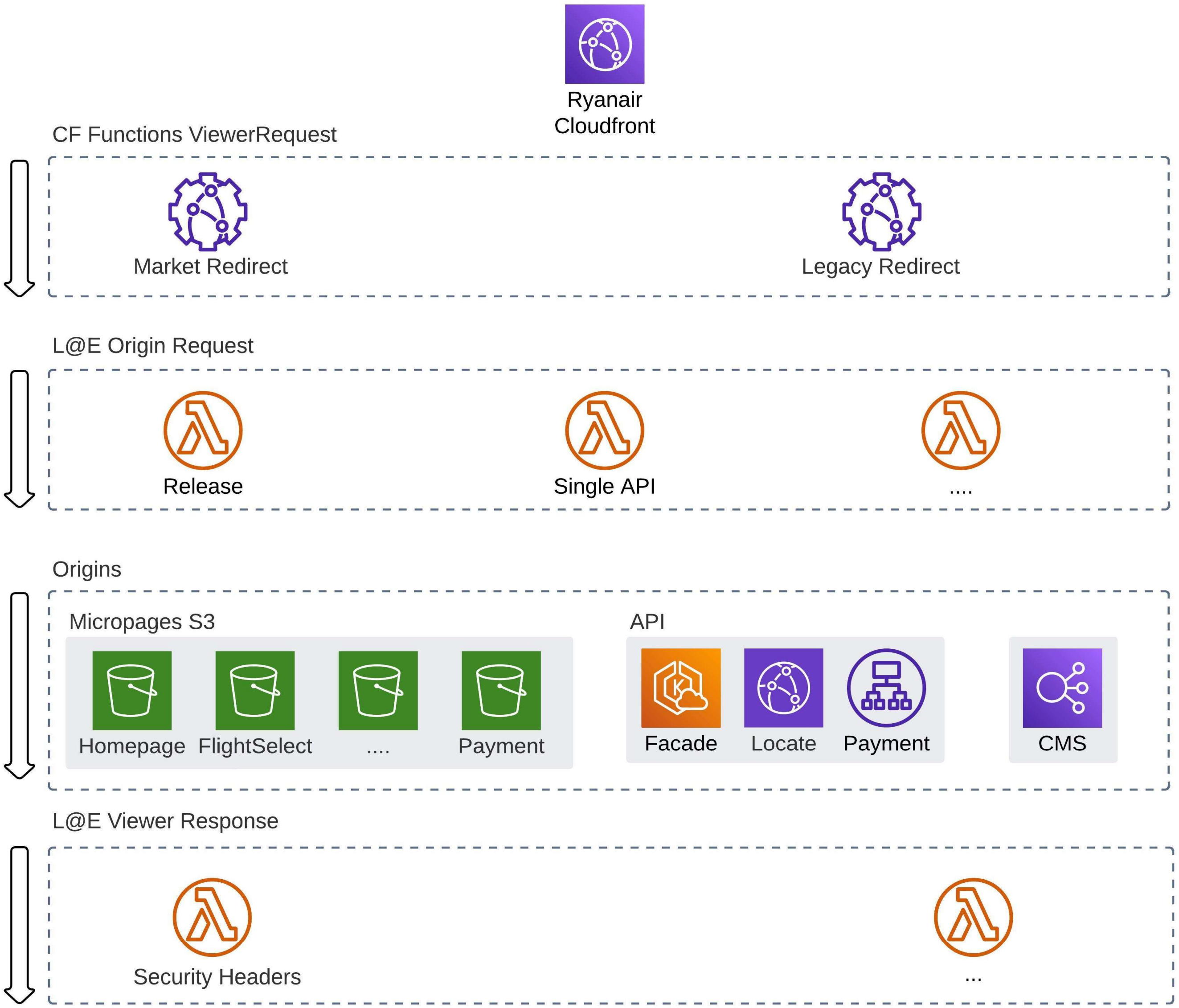
Cloudfront Functions

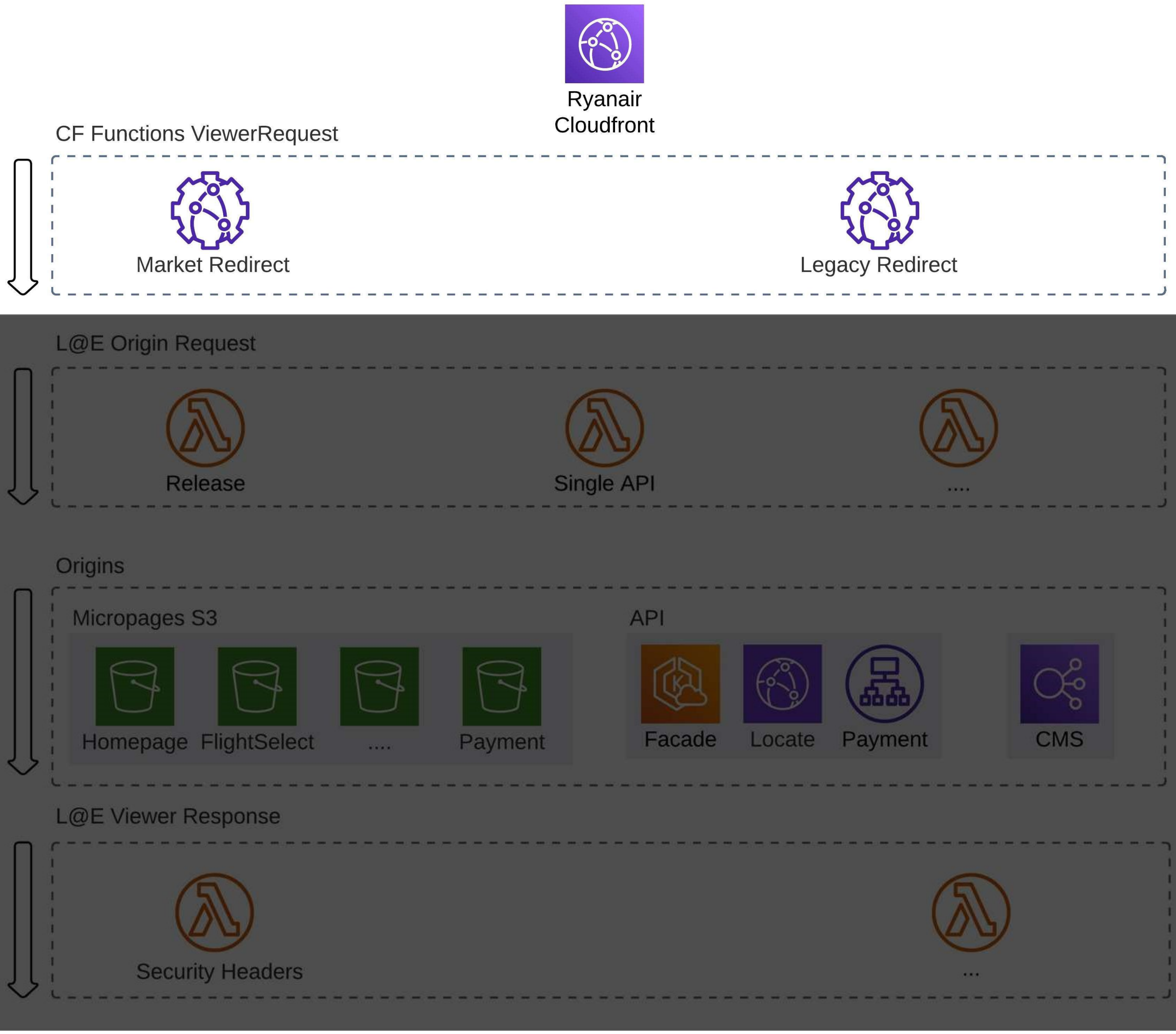
Cloud Migration

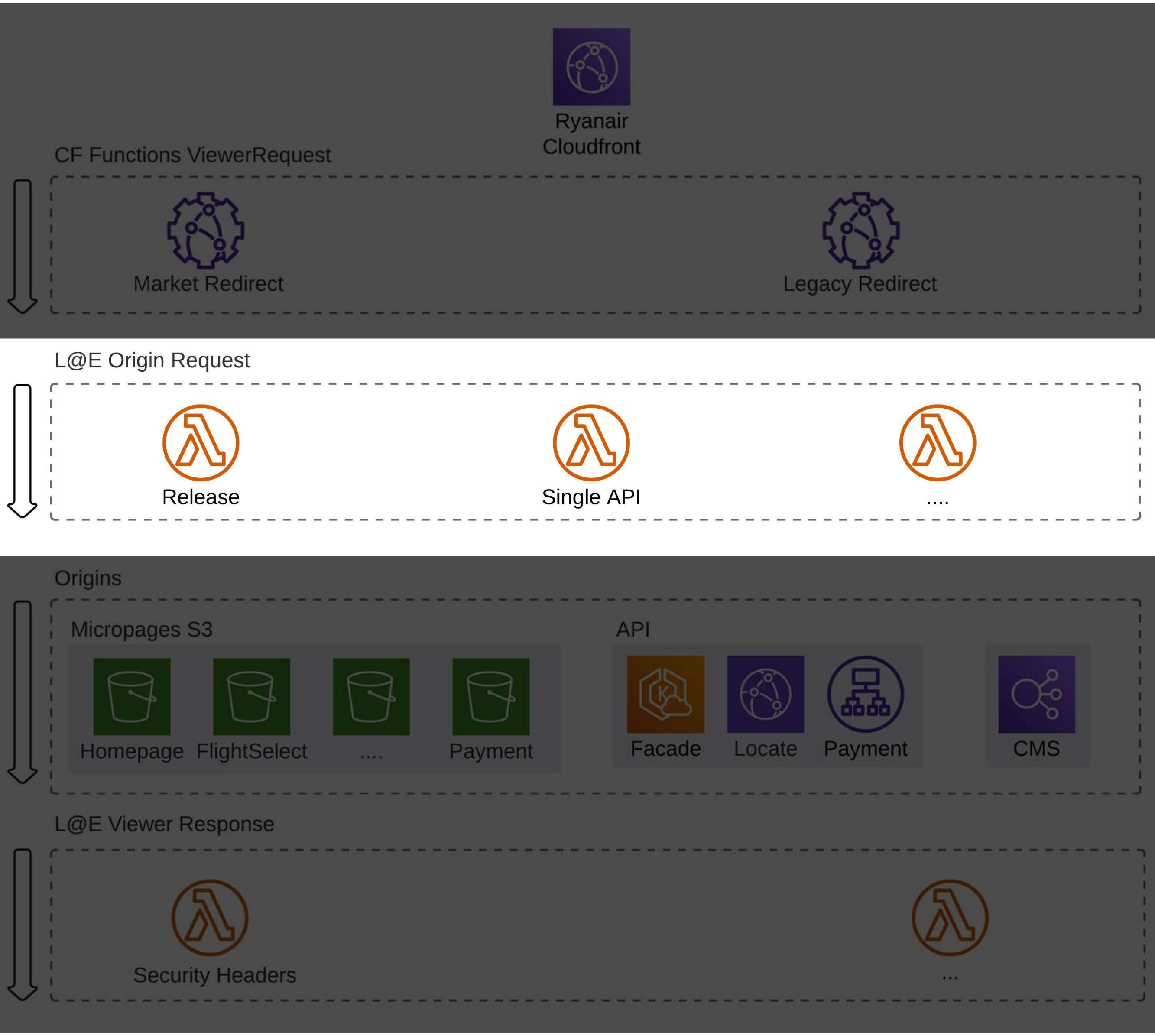
RYANAIR

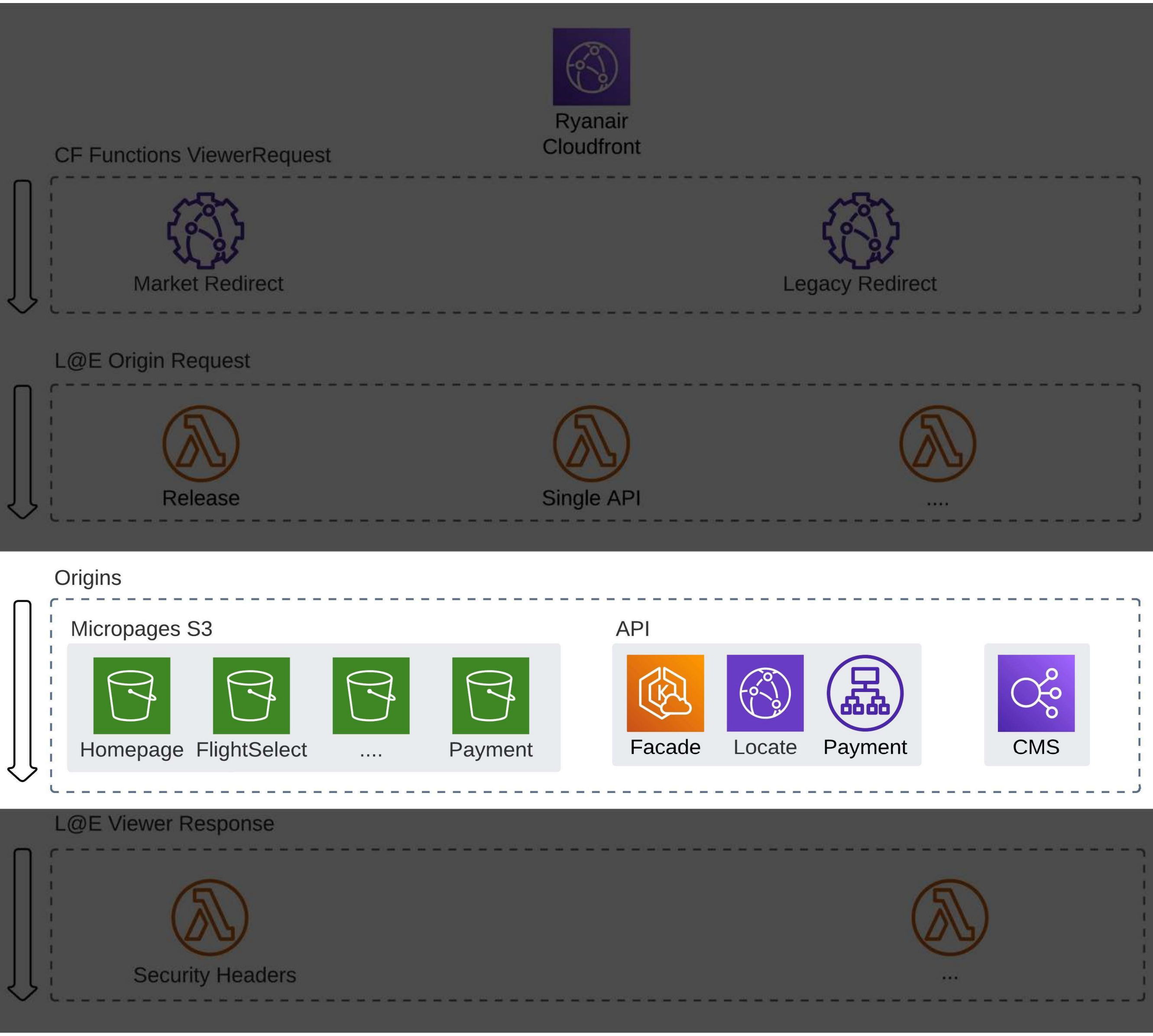
DON’T TRY THIS AT HOME

RYANAIR


Ryanair Cloudfront
CF Functions ViewerRequest
Market Redirect
Legacy Redirect
L@E Origin Request
Release
Single API
....
Origins
Micropages S3
Homepage
FlightSelect
....
Payment
API
Facade
Locate
Payment
CMS
L@E Viewer Response
Security Headers
...


Ryanair Cloudfront
CF Functions ViewerRequest
Market Redirect
Legacy Redirect
L@E Origin Request
Release
Single API
....
Origins
Micropages S3
Homepage
FlightSelect
....
Payment
API
Facade
Locate
Payment
CMS
L@E Viewer Response
Security Headers
...


Ryanair Cloudfront
CF Functions ViewerRequest
Market Redirect
Legacy Redirect
L@E Origin Request
Release
Single API
....
Origins
Micropages S3
Homepage
FlightSelect
....
Payment
API
Facade
Locate
Payment
CMS
L@E Viewer Response
Security Headers
...



Ryanair Cloudfront

CF Functions ViewerRequest

Market Redirect

Legacy Redirect

L@E Origin Request

Release

Single API

....

Origins

Micropages S3

Homepage

FlightSelect

....

Payment

API

Facade

Locate

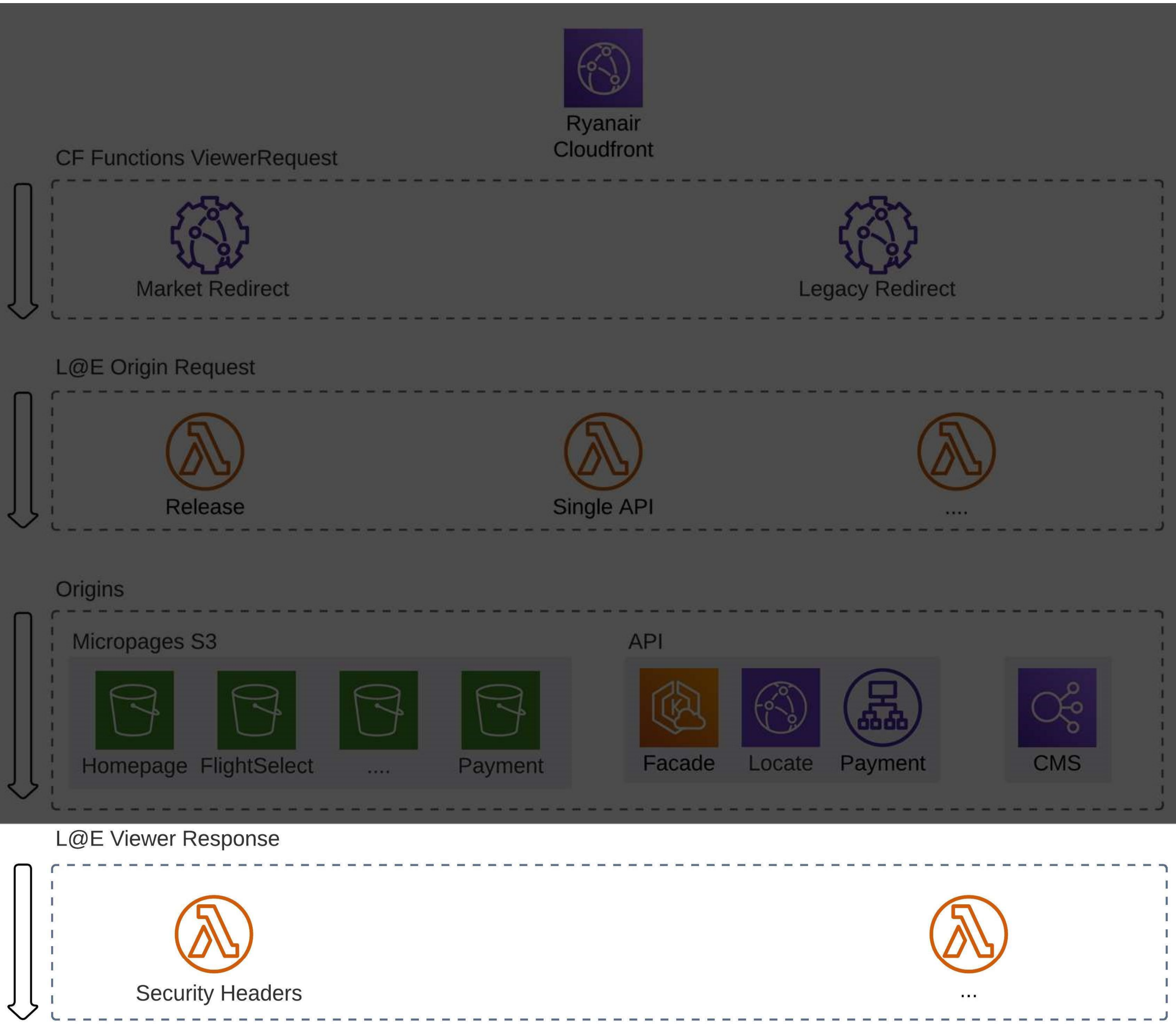
Payment

CMS

L@E Viewer Response

Security Headers

...



Ryanair Cloudfront

CF Functions ViewerRequest

Market Redirect

Legacy Redirect

L@E Origin Request

Release

Single API

....

Origins

Micropages S3

Homepage

FlightSelect

....

Payment

API

Facade

Locate

Payment

CMS

L@E Viewer Response

Security Headers

...

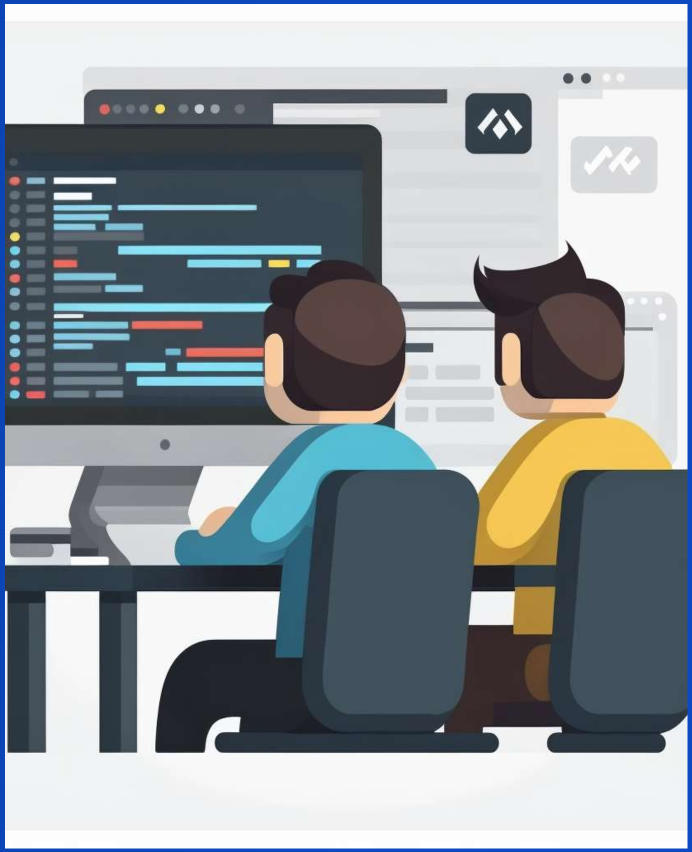
FrontendInfra

- Micropages isolation
- Performance fine tuning
- Zero downtime
- Plug & Play functions

RYANAIR

FrontendOps

- Team growth
- Infrastructure ownership with Infrastructure as code and Serverless



TAILOR INFRASTRUCTURE TO YOUR **BUSINESS NEEDS** AND YOUR **TEAM SKILLS**

RYANAIR