Cloud Native with Spring Boot and Kubernetes

From development to production

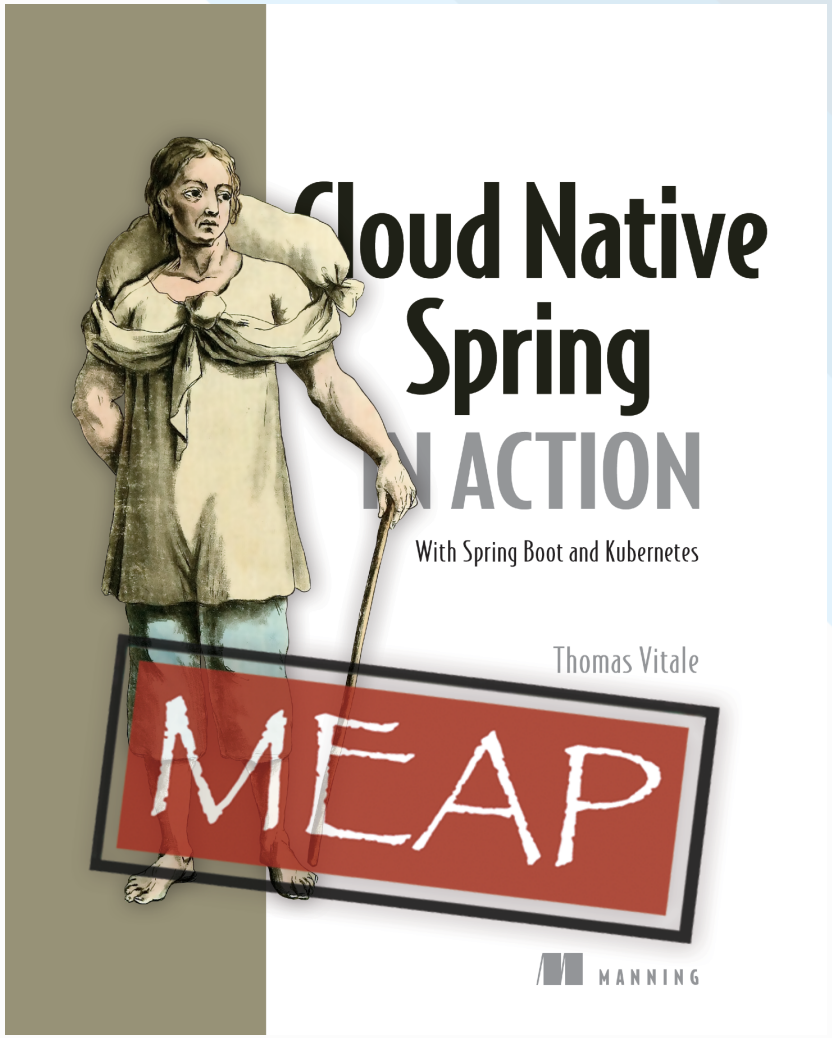
Thomas Vitale
GOTO Aarhus
Jun 10th, 2021

@vitalethomas

About Me

Thomas Vitale

- Senior Software Engineer at Systematic, Denmark.
- Spring, Cloud Native, DevOps, Kubernetes, Application Security.
- Author of “Cloud Native Spring in Action” (Manning).



Cloud Native

thomasvitale.com

@vitalethomas

Why Cloud Native?

Speed

Faster and flexible delivery

Resilience

Availability and stability

Scale

Elasticity and dynamic scaling

Cost

Efficiency and cost optimisation

The Three P's of Cloud Native Applications

Place	Properties	Practices
Private Cloud Public Cloud Hybrid Cloud	Scalability Loose Coupling Resilience Manageability Observability Security	Automation Continuous Delivery DevOps

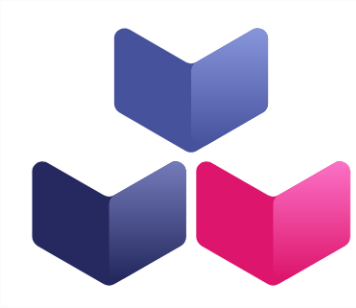
From Development to Production

Cloud native journey in less than 45 minutes

Development

Spring Boot

Containerization



Cloud Native Buildpacks

Deployment

Kubernetes

Cloud Native Development

thomasvitale.com

@vitalethomas

Cloud Native Development

Development principles with Spring Boot

- **Self-contained application**
  - Embedded server
  - No external dependencies
  - JAR packaging (“fat-JAR”)
- **Externalized configuration**
  - Property files for default values
  - JVM system variables
  - Environment variables

Containerization

thomasvitale.com

@vitalethomas

1. Don’t use fat JARs
2. Optimize build/runtime performance
3. Don’t run as *root* or include secrets

Containerize Spring Boot Applications

Dockerfiles

Jib

Cloud Native Buildpacks

thomasvitale.com

@vitalethomas

Spring Boot on Kubernetes

thomasvitale.com

@vitalethomas

Spring Boot on Kubernetes

Kubernetes manifests for deploying applications

- ***Deployment*** -> deploy the application (with replicas)
- ***Service*** -> expose the application to the inside of the cluster
- ***Ingress*** -> expose the application to the Internet

Deploying Spring Boot Applications

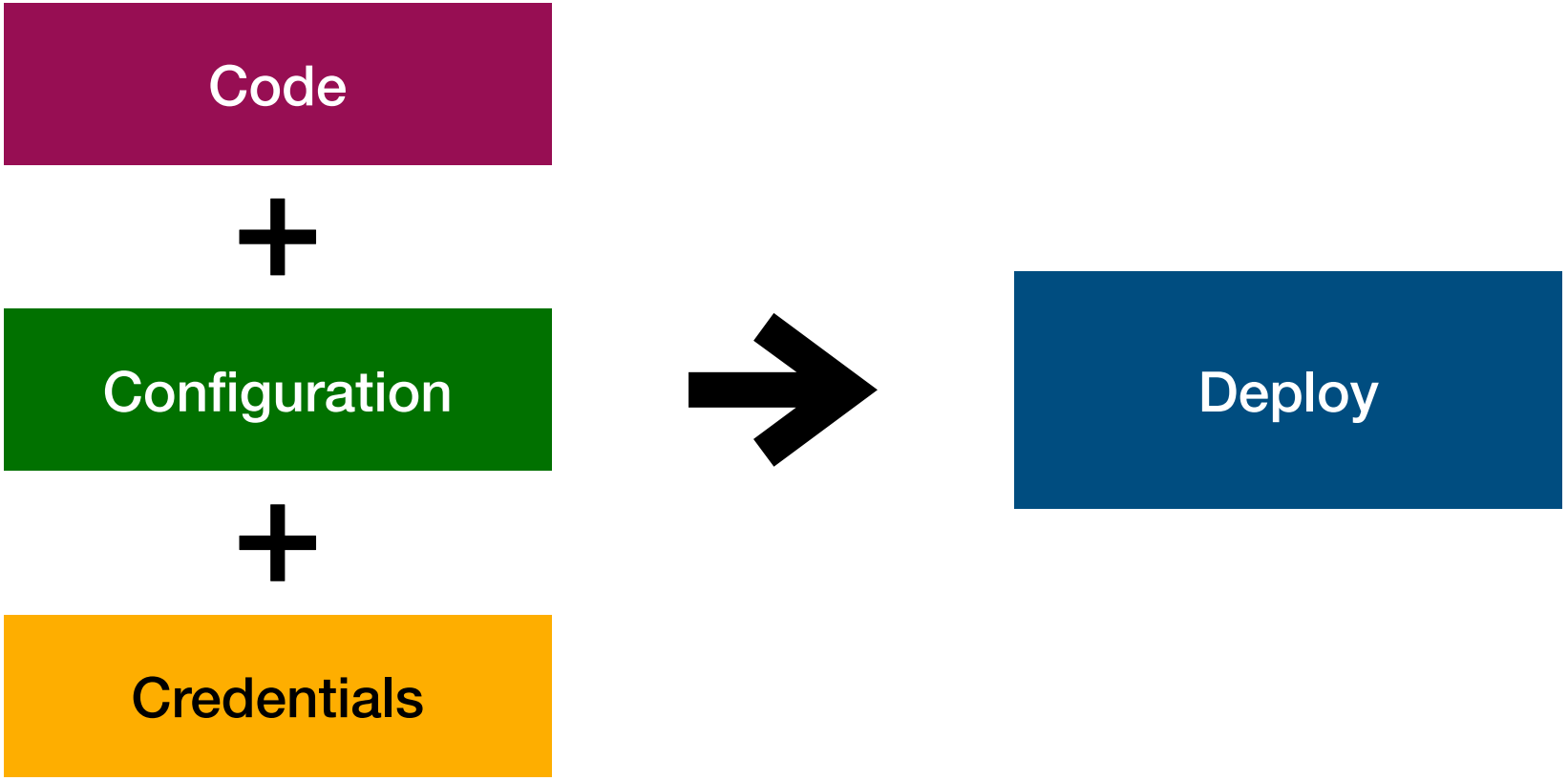
Computational resource configuration

- ***Requests*** are the resources guaranteed to the application container.
- ***Limits*** define the maximum resources an application container can get.

- **CPU** is *compressible*.
  - When limit hit: throttle.
  - For JVM containers, no limit for startup boost.
- **Memory** is *non-compressible*.
  - When limit hit: OOMKilled
  - For JMV containers, same value for *requests* and *limits*.

Externalized Configuration

Code, Configuration, Credentials



ConfigMaps and Secrets

Configuration and credentials

- **ConfigMaps**
  - Environment variables
  - Volume mounts
- **Secrets**
  - Environment variables
  - Consider a backend like Vault for actual encryption or Sealed Secrets

Graceful Shutdown

thomasvitale.com

@vitalethomas

Graceful shutdown

Spring Boot and Kubernetes

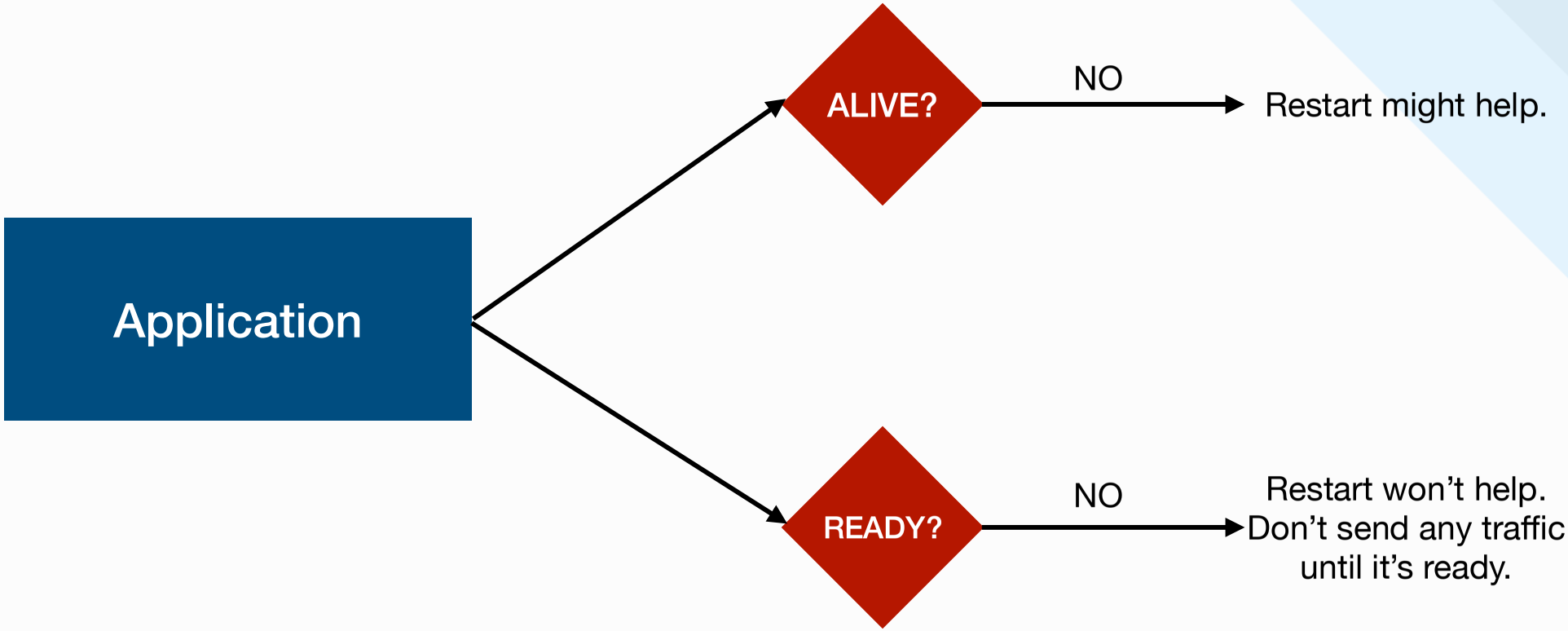
- **Spring Boot**
  - Enable graceful shutdown
  - Define a grace period
- **Kubernetes**
  - Add pre-stop hook
  - Define a grace period

Health Probes

thomasvitale.com

@vitalethomas

Liveness and Readiness Probes



Health Probes

Liveness and readiness

- **Spring Boot**
  - Use Spring Boot Actuator
  - Liveness and readiness health endpoints are automatically exposed when Kubernetes is detected.
- **Kubernetes**
  - Configure liveness probe
  - Configure readiness probe

Spring Native

thomasvitale.com

@vitalethomas

Spring Native

Native executables with GraalVM

- **Benefits**
  - Instant startup
  - Instant peak performance
  - Reduced memory consumption
- **Tradeoffs**
  - Slower and heavier build process
  - Fewer runtime optimizations

Service discovery

thomasvitale.com

@vitalethomas

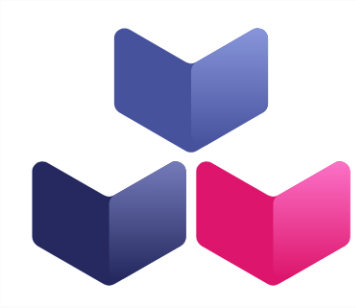
From Development to Production

Cloud native journey in less than 45 minutes

Development

Spring Boot

Containerization



Cloud Native Buildpacks

Deployment

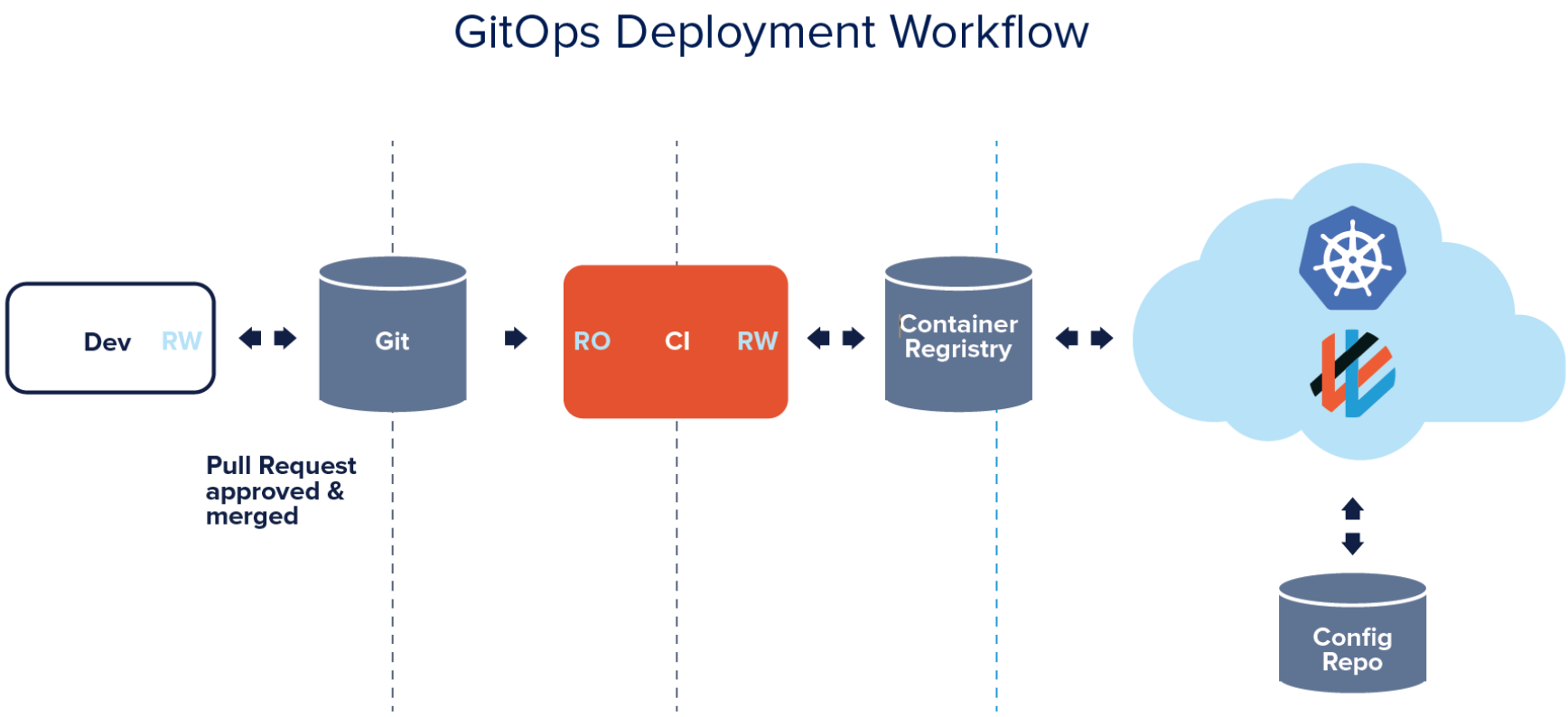
Kubernetes

GitOps & Kubernetes

thomasvitale.com

@vitalethomas

GitOps - Cloud Native Operations



https://www.weave.works/technologies/gitops/

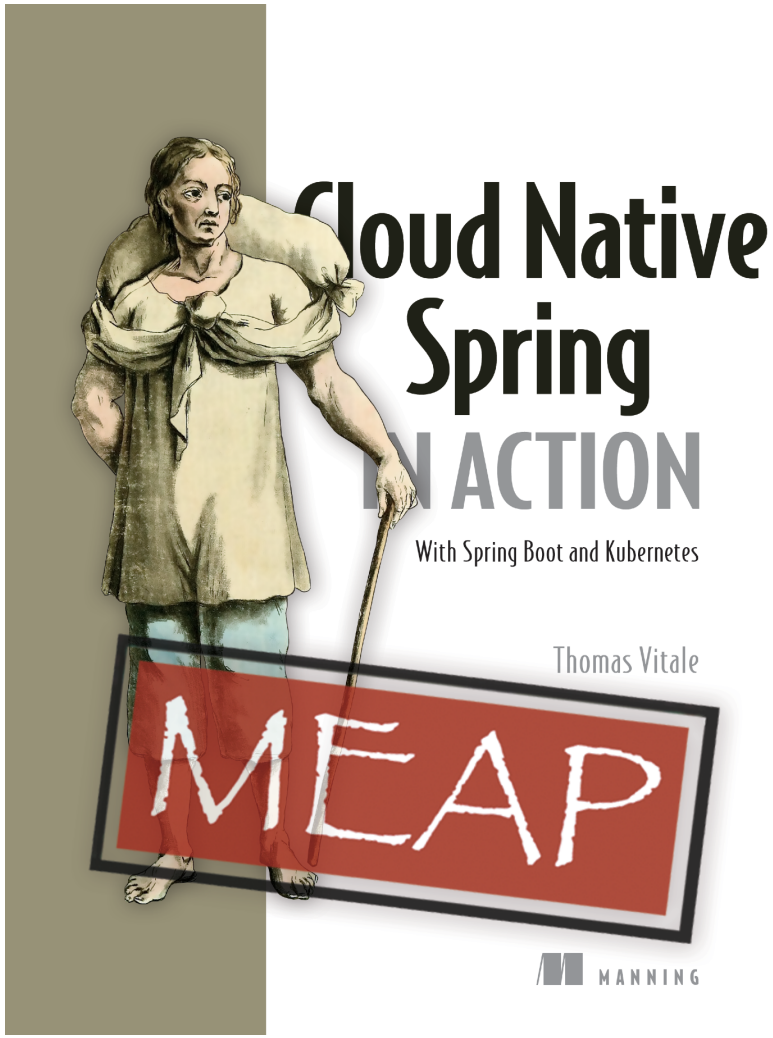
GitOps - Configuration

```
helm upgrade -i flux fluxcd/flux \
  --set git.user=$GITHUB_USER \
  --set git.email=$GITHUB_USER@users.noreply.github.com \
  --set git.url=git@github.com:$GITHUB_USER/spring-boot-
kubernetes-goto-2021 \
  --set git.path="deployment" \
  --set git.branch="main" \
  --namespace flux
```

Cloud Native Spring in Action

With Spring Boot and Kubernetes

- 35% discount code, valid for all products in all format
  - `ctwgotoaar21`
  - manning.com



www.thomasvitale.com @vitalethomas