

The background of the slide features a faded image of a multi-story brick building on the left and a church with a tall, ornate stone tower on the right. A large, white, stylized letter 'S' is superimposed over the right side of the image.

Tag 3: GitOps, Docker in der Entwicklung und Deployment-Strategien

19.06.2024, Daniel Krämer & Malte Fischer

© Copyright 2024 anderScore GmbH

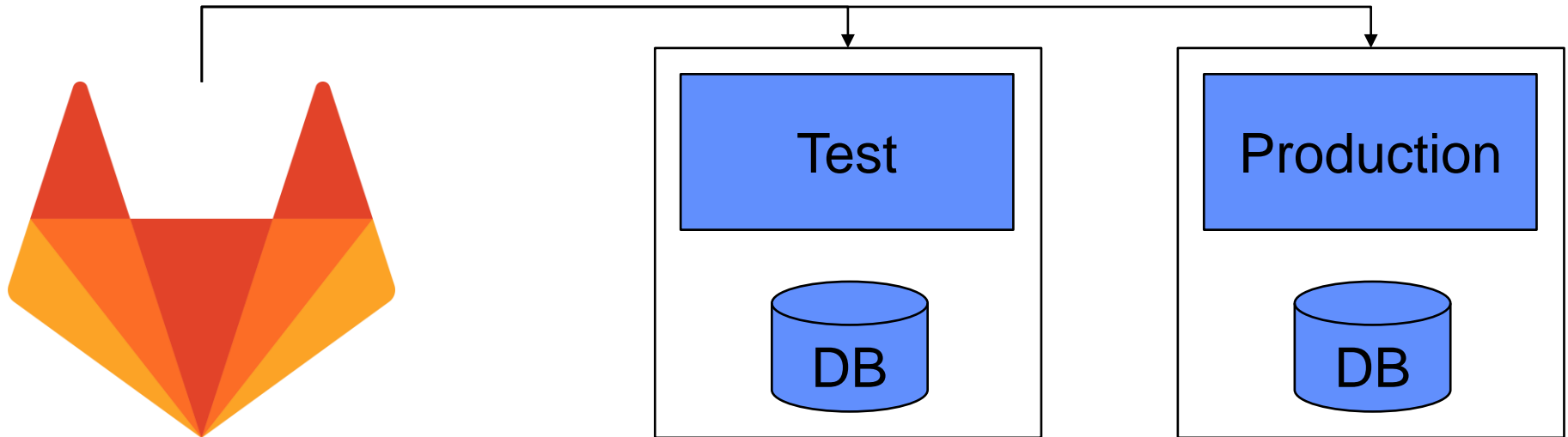
- **Tag 1 – Einführung in Git und GitLab, Git-Workflow im Team**
 - Einführung & Kursüberblick
 - Grundlagen von Git
 - Git Rebase und Merge-Strategien
 - Git Remote
 - Grundlagen von GitLab
 - Git-Workflow im Team
- **Tag 2 – Vertiefung Git-Workflow, CI/CD & GitLab CI**
 - Gitflow-Workflow
 - Tags, Releases & deren Verwaltung
 - GitLab-Runner
 - Einführung in GitLab CI/CD & gitlab.yml
- **Tag 3 – GitOps, Docker in der Entwicklung und Deployment-Strategien**
 - GitOps Grundlagen
 - Lokale Entwicklung mit Docker
 - Container/Docker-Registry
 - Erstellen von Release- und Tagged-Images
 - Möglichkeiten des Deployments & Verwaltung von Konfiguration
 - Abschlussübung & Diskussion

- **Tag 1 – Einführung in Git und GitLab, Git-Workflow im Team**
 - Einführung & Kursüberblick
 - Grundlagen von Git
 - Git Rebase und Merge-Strategien
 - Git Remote
 - Grundlagen von GitLab
 - Git-Workflow im Team
- **Tag 2 – Vertiefung Git-Workflow, CI/CD & GitLab CI**
 - Gitflow-Workflow
 - Tags, Releases & deren Verwaltung
 - GitLab-Runner
 - Einführung in GitLab CI/CD & gitlab.yml
- **Tag 3 – GitOps, Docker in der Entwicklung und Deployment-Strategien**
 - GitOps Grundlagen
 - Lokale Entwicklung mit Docker
 - Container/Docker-Registry
 - Erstellen von Release- und Tagged-Images
 - Möglichkeiten des Deployments & Verwaltung von Konfiguration
 - Abschlussübung & Diskussion

Möglichkeiten des **DEPLOYMENT**

Deployment mit Gitlabs

- Sehr Flexibel
- Viele Strategien möglich
- Fast jeder Workflow

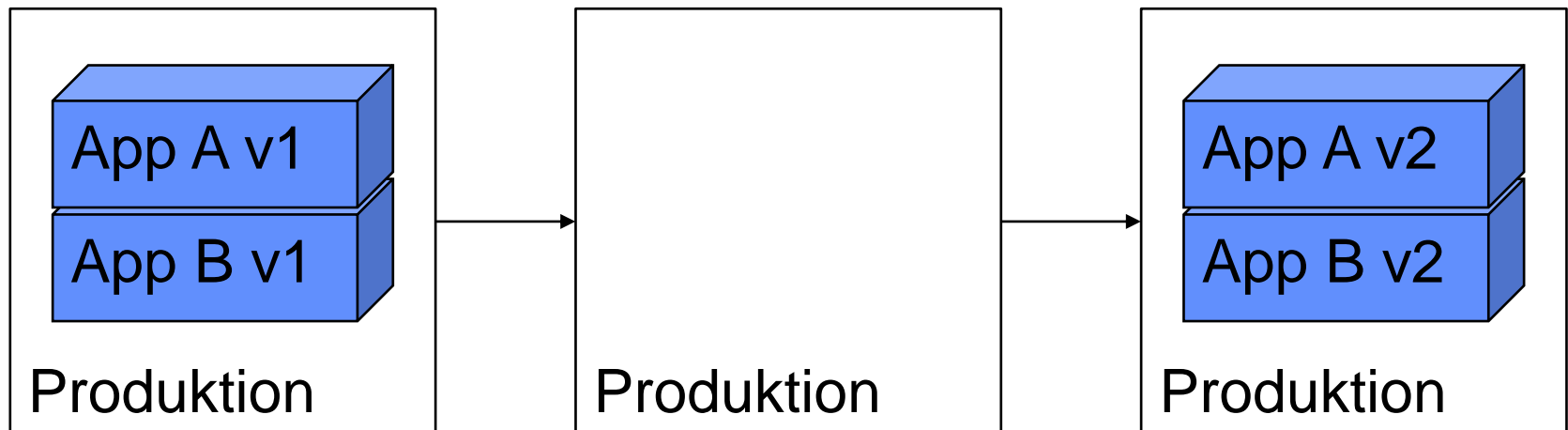


Verschiedene Deployment Strategien

- Recreate Deployment
- Blue-Green Deployment
- Canary Deployment
- Shadow Deployment

Recreate Deployment

- Altsystem wird offline genommen
- Neusystem wird danach deployed und gestartet



Vorteile

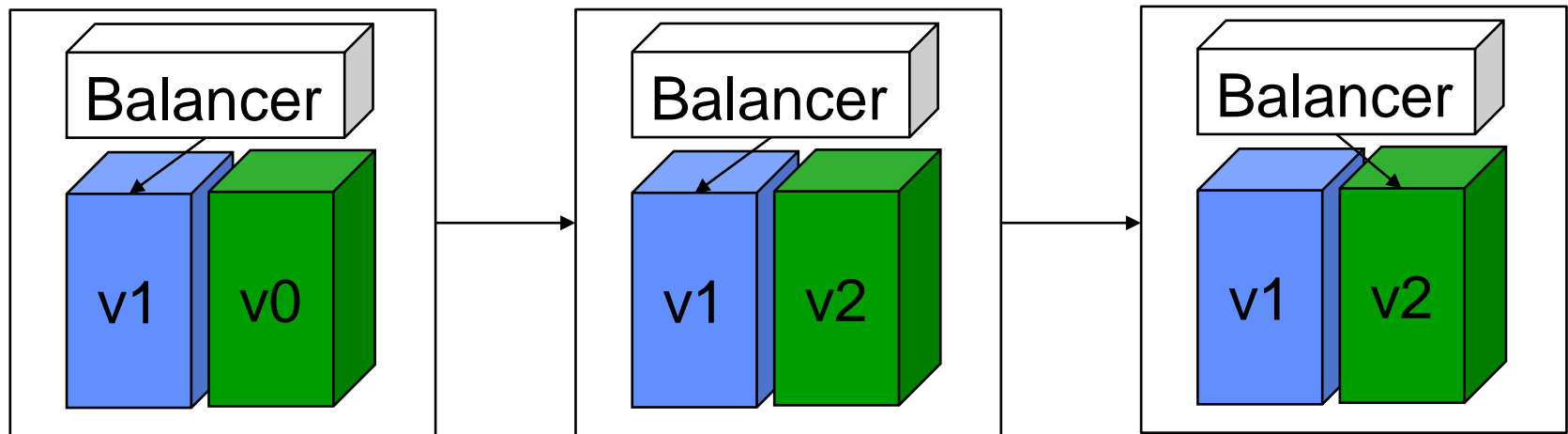
- Einfache Implementierung
- Günstig

Nachteile

- Downtime
- Aufwändiges Rollback
- Kein partieller Deploy möglich

Blue-Green Deployment

- Alt und Neu werden parallel gehostet
- Anfragen gehen erst nur teilweise aufs neue System
- Alt System für Rollback und weitere Updates nutzen



Vorteile

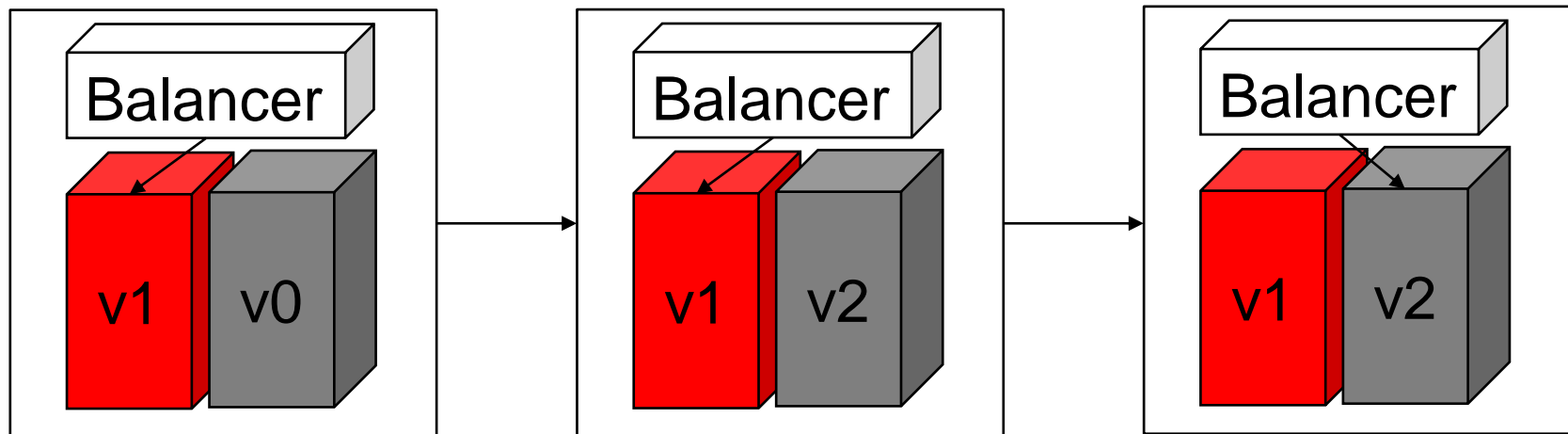
- Keine Downtime
- Einfaches Rollback
- Partieller Deploy

Nachteile

- Aufwändige Implementierung
- Zufällige Zuweisung zu neuem System
- Teuer

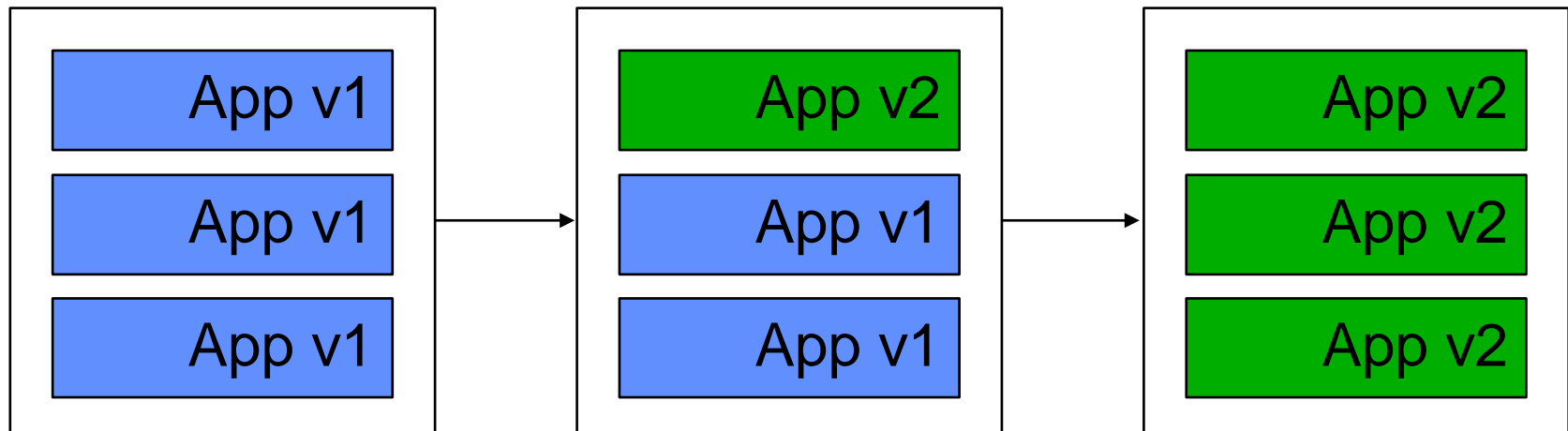
Red-Black Deployment

- Sehr ähnlich zu Blue-Green
- Kompletter Wechsel zu Neusystem



Canary Deployment

- Neue Version nur für bestimmten Teil der User released
- Test der Version mit Testgruppe



Vorteile

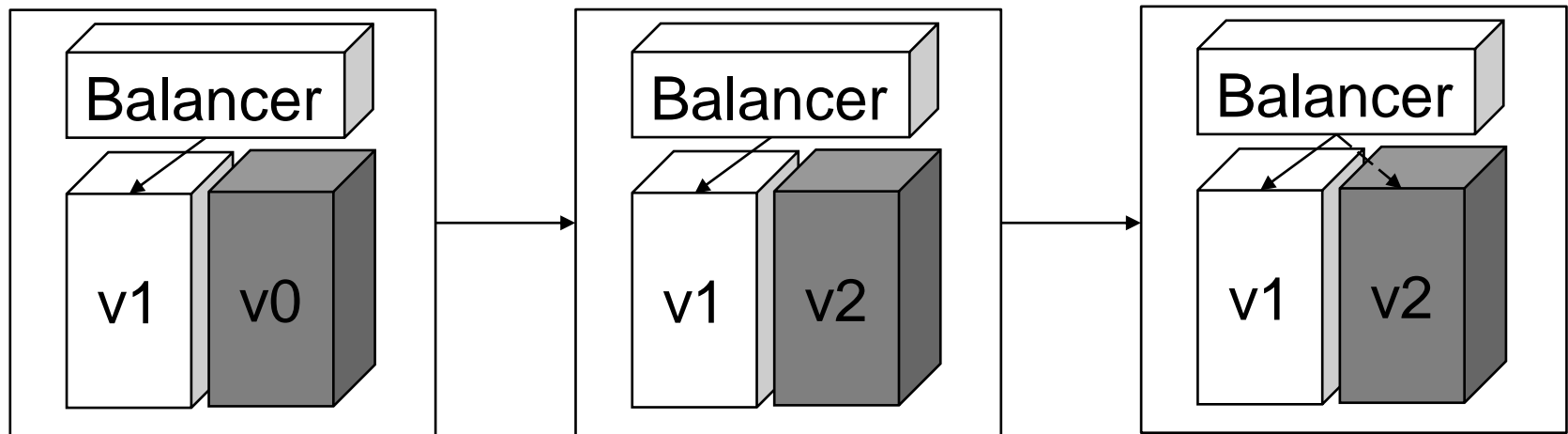
- Keine Downtime
- Partieller Deploy
- Gezielte Testgruppe

Nachteile

- Aufwändige Implementierung
- Teuer
- Testgruppe nötig

Shadow Deployment

- Neusystem wird parallel zu Altsystem gehosted
- Anfragen werden vom Altsystem bearbeitet
- Anfragen werden zum Neusystem gespiegelt



Vorteile

- Keine Downtime
- Kein Rollback nötig
- Test unter Volllast

Nachteile

- Aufwändige Implementierung
- Teuer