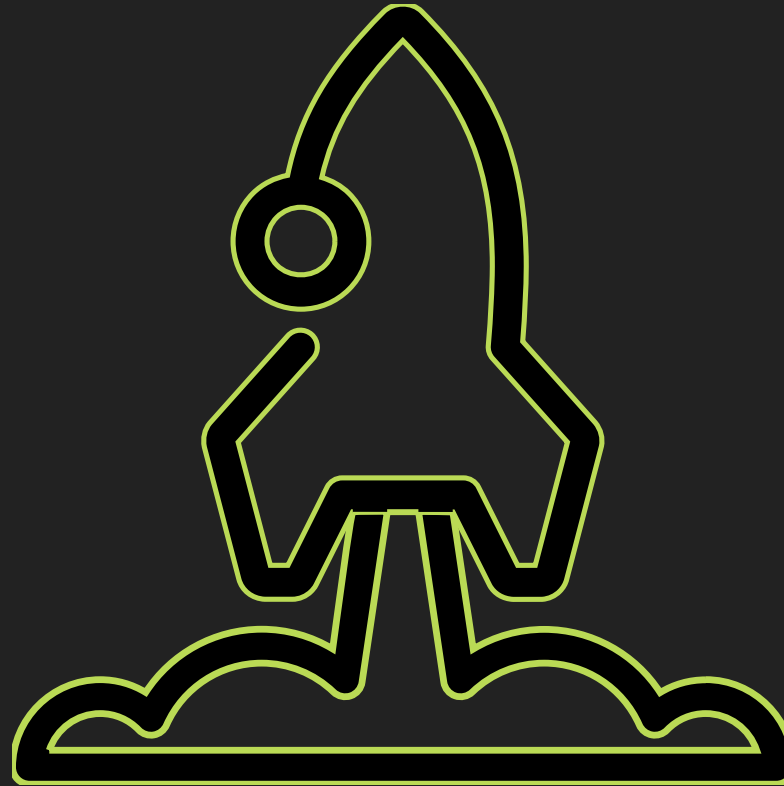


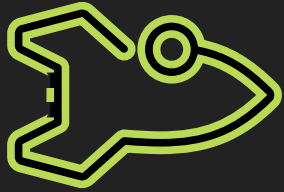
CONTINUOUS



DEPLOYMENT

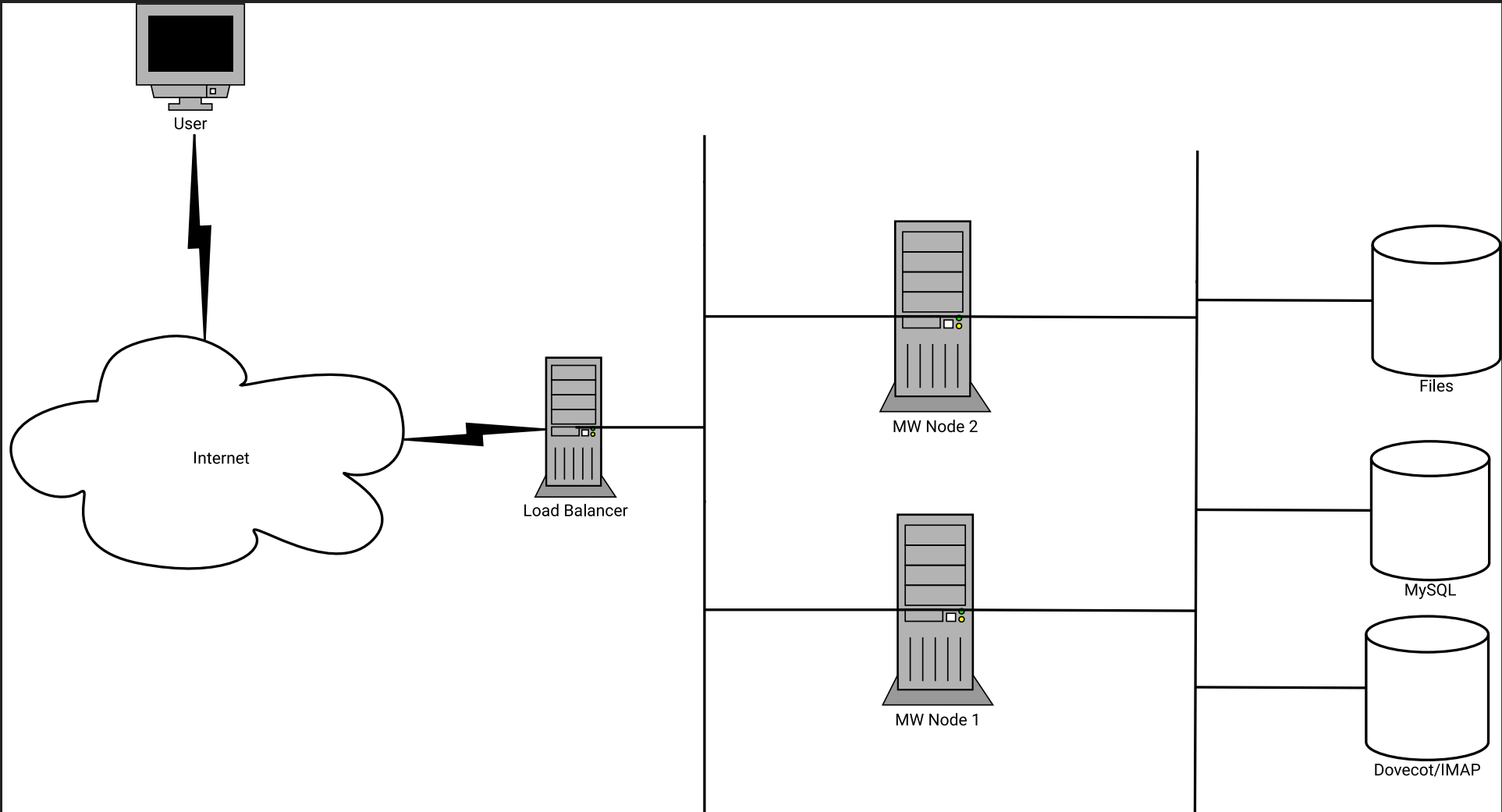
Rocket is CC-BY-SA @gitlab (AFAIK)

CONTINUOUS



DEPLOYMENT

APP SUITE ARCHITEKTUR



SZENARIO

- Middleware
- UI
- Storage services
- Environment: Stable
- Environment: Current
- Environment: Preview

SZENARIO

- Middleware
- UI
 - Storage services
- Environment: Stable
- Environment: Current
- Environment: Preview

SZENARIO

- Middleware
- UI
 - Storage services
- Environment: Stable
- Environment: Current
- Environment: Preview

LÖSUNG

*Just push your code and GitLab
takes care of everything else.*

ALTERNATIVER ANSATZ

*Nur wirklich benötigte
Komponenten bauen und
verdrahten. Alles automatisieren.*

DOCKER IMAGES

- base images
- Entwicklungsumgebung
- Schnelles Deployment einzelner Komponenten in andere Umgebungen

DOCKER IMAGES

```
build image:
  stage: package
  image: docker:latest
  script:
    - docker build --pull \
      -t $CI_REGISTRY_IMAGE:$CI_COMMIT_REF_SLUG .

    - docker login -u gitlab-ci-token \
      -p $CI_BUILD_TOKEN $CI_REGISTRY

    - docker push $CI_REGISTRY_IMAGE:$CI_COMMIT_REF_SLUG
```

END-TO-END UMGEBUNG

```
deploy e2e container:  
  stage: deploy  
  image: docker:latest  
  variables:  
    DOCKER_HOST: docker.ui.cloud.open-xchange.com  
    DOCKER_IMAGE: $CI_REGISTRY_IMAGE:$CI_COMMIT_REF_SLUG  
    CONTAINER_NAME: e2e_${CI_PIPELINE_ID}  
  script:  
    - docker pull $CI_REGISTRY_IMAGE:$CI_COMMIT_REF_SLUG  
    - docker run --rm --name $CONTAINER_NAME $DOCKER_IMAGE
```

TRÆFIK

FROM THE INTERNET

TO YOUR INFRASTRUCTURE

API.DOMAIN.COM

DOMAIN.COM/WEB

BACKOFFICE.DOMAIN.COM



HTTPS



traefik

METRICS
TRACING
LOGS

AUTOMATIC & DYNAMIC
ROUTING

AUTO DISCOVERY

API
SERVER

KUBERNETES

WEB
SERVER

MESOS

PRIVATE
SERVER

BACKOFFICE
SERVER - 1

BACKOFFICE
SERVER - 2

DOCKER

LOAD
BALANCING

TRÆFIK

DEMO

CONCLUSION