

Automatisiertes Aufsetzen eines Kubernetes-Clusters auf Raspberry Pis mithilfe von Ansible-Playbooks

KL

Seminararbeit (Bachelor)

Betreuer: RH, HB

Agenda

- Einleitung
- Docker, Kubernetes, Raspberry Pi, Ansible
- Umsetzung
- Live-Demo + Austausch
- Fazit

Einleitung

- Kubernetes ermöglicht skalierbare, verteilte Applikationen
- Komplizierte Einrichtung mit Ansible automatisieren

Docker

- Container-virtualisierung
- Isolation von Ressourcen
- Dockerfiles

```
FROM ubuntu:18.04
```

```
RUN apt-get update && \
    apt-get install \
    redis-server
```

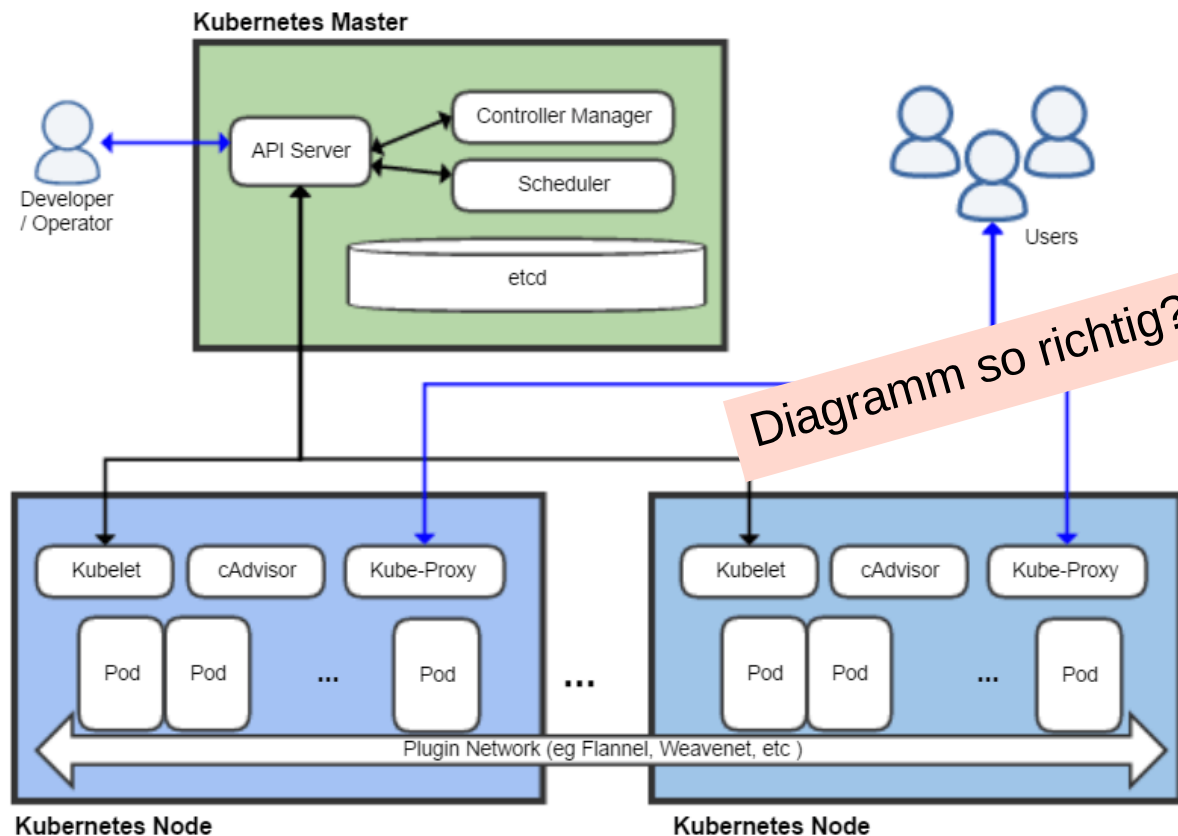
Hier könnte Ihr
nginx-Dockerfile stehen

```
CMD ["redis-server",  
    "--protected-mode no"]
```

<https://linuxize.com/post/how-to-build-docker-images-with-dockerfile/>

Kubernetes

- Skalierung und Verwaltung von Containern
- Verteilte Hosts/Cloud



Von Khtan66 - Eigenes Werk, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=53571935>

Raspberry Pi

- Einplatinencomputer
- ARM-Architektur
- Günstig und sparsam



Ansible

- Automatisierung von Software-Installationen
- Infrastructure as code
- YAML-Playbooks
- Module

```
---  
- hosts: all  
  become: yes  
  tasks:  
    - name: Set master flag  
      set_fact: master=yes  
      when: ansible_host == masterIp  
  
    - name: Set timezone  
      timezone:  
        name: "{{ timezone }}"  
  
    - name: Upgrade apt packages  
      apt:  
        upgrade: yes  
        update_cache: yes
```

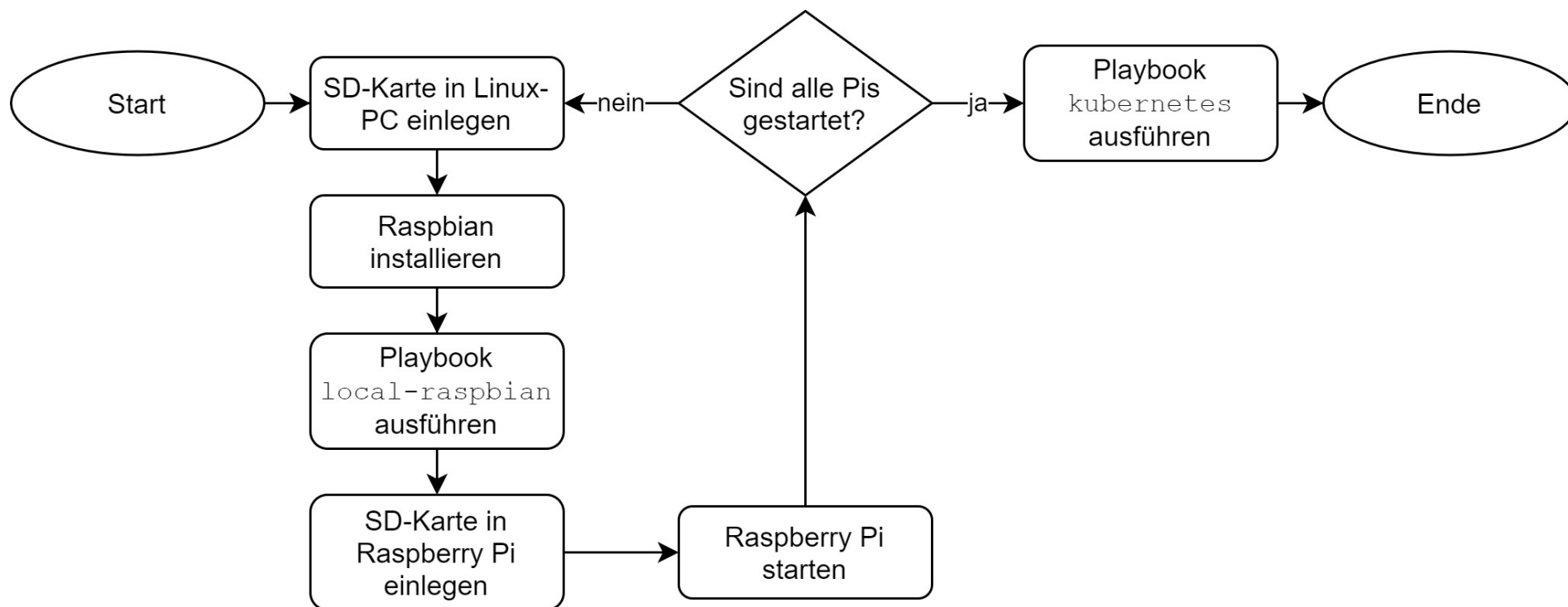
Aufgaben

- Raspbian installieren
- WiFi einrichten (Zugangsdaten, statische IP)
- Swapfile deaktivieren
- Control Groups aktivieren
- SSH-Keys hinterlegen
- Softwarepakete aktualisieren
- Docker installieren
- Kubernetes installieren
- Master Node initialisieren
- Weitere Nodes zum Cluster hinzufügen
- Virtuelles Netzwerk installieren

`local-raspbian.yaml`

`kubernetes.yaml`

Ablauf der Einrichtung



„Live“-Demo

Fazit

- Kubernetes-Cluster in 35 Minuten
- Raspbian-Update nicht reibungslos

Fragen?

Danke!