



**resinOS**

**IoT Devices unter eigener Kontrolle**

**Pi and More 9 1/2**

**Nico Maas**



**Nico Maas**

**IT Systemelektroniker**

**Bachelor of Science**

**mail@nico-maas.de**

**www.nico-maas.de**

**@nmaas87**



- **I. Einführung**
  - Problemstellung
  - Mögliche Lösungen?!
  - Neue Probleme?!
- **II. resinOS**
  - Grundlagen
  - Docker
  - Resin.io / resinOS
  - Live Demo
- **III. Abschluss**
  - Fragen
  - Quellenangaben

# Problemstellung



## Einführung

- Problem
- Lösungen?!
- Neue Probleme?!

resinOS

Abschluss



# Problemstellung

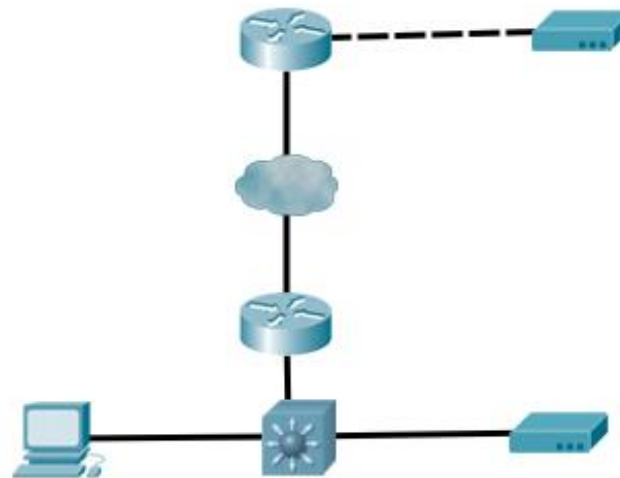


## Einführung

- Problem
- Lösungen?!
- Neue Probleme?!

resinOS

Abschluss



# Problemstellung

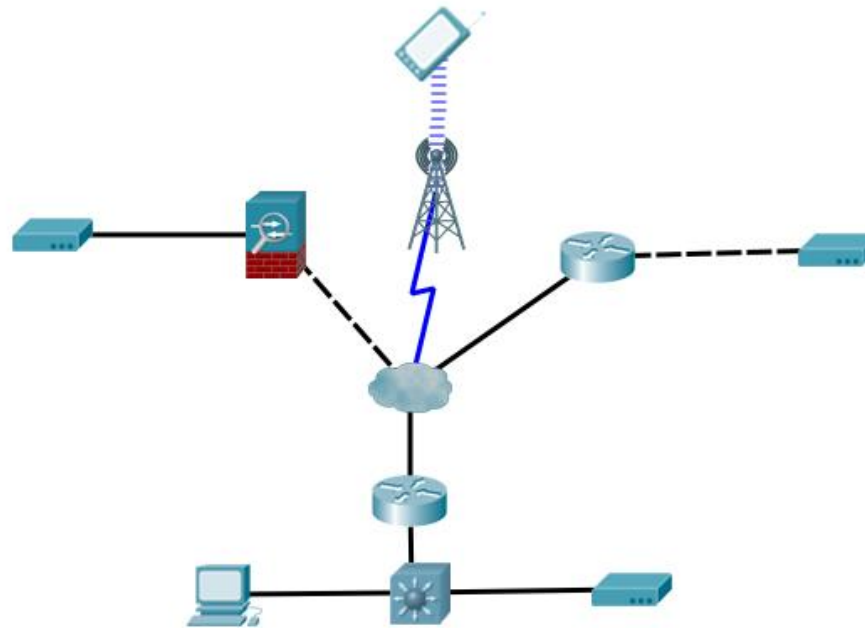


## Einführung

- Problem
- Lösungen?!
- Neue Probleme?!

resinOS

Abschluss



# Mögliche Lösungen?!



## Einführung

- Problem
- Lösungen?!
- Neue Probleme?!

resinOS

Abschluss

- ftp
- reverse ssh
- cronjob Download von binärem Blob
- cronjob Download von github

# Neue Probleme?!



## Einführung

- Problem
- Lösungen?!
- **Neue Probleme?**

resinOS

Abschluss

- Router/Firewall verbietet Zugriff auf Geräte
- Verbindungsabbrüche
- Übertragungsfehler
- Updatescript wird beim "Self-Update" beschädigt / enthält Fehler
- App enthält Fehler und legt das Gerät lahm
- Stromausfall beschädigt die SD Karte





PRODUCT

PRICING

CUSTOMERS

DOCS

COMMUNITY

COMPANY

SIGN UP | LOGIN

Modern development and deployment workflow for the Internet of Things

# Update Your Linux Device Fleet Over The Air



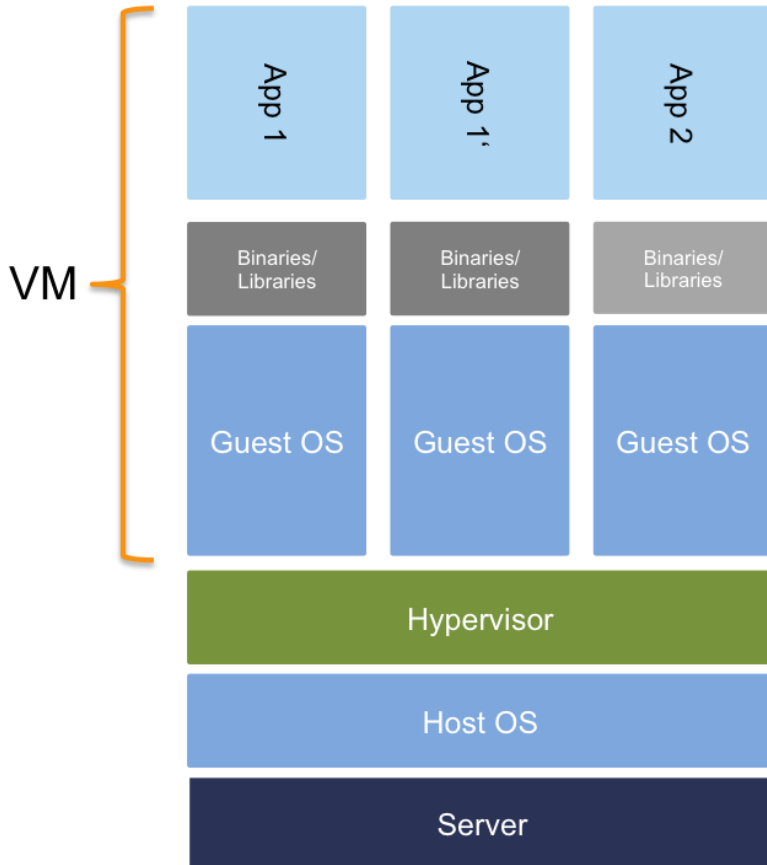
RESIN.IO IN 2 MINUTES

DEPLOY YOUR FIRST APPLICATION

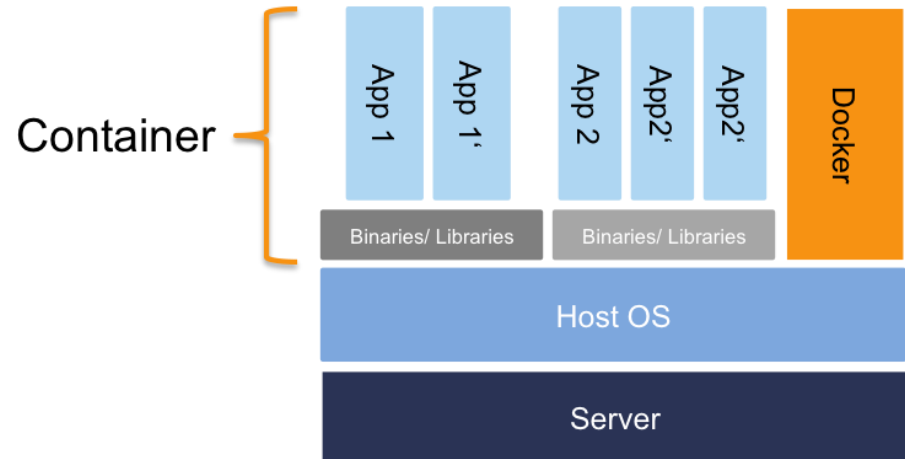
# Docker: VMs vs Container



## Virtualisierung: Virtuelle Maschinen vs. Docker-Container



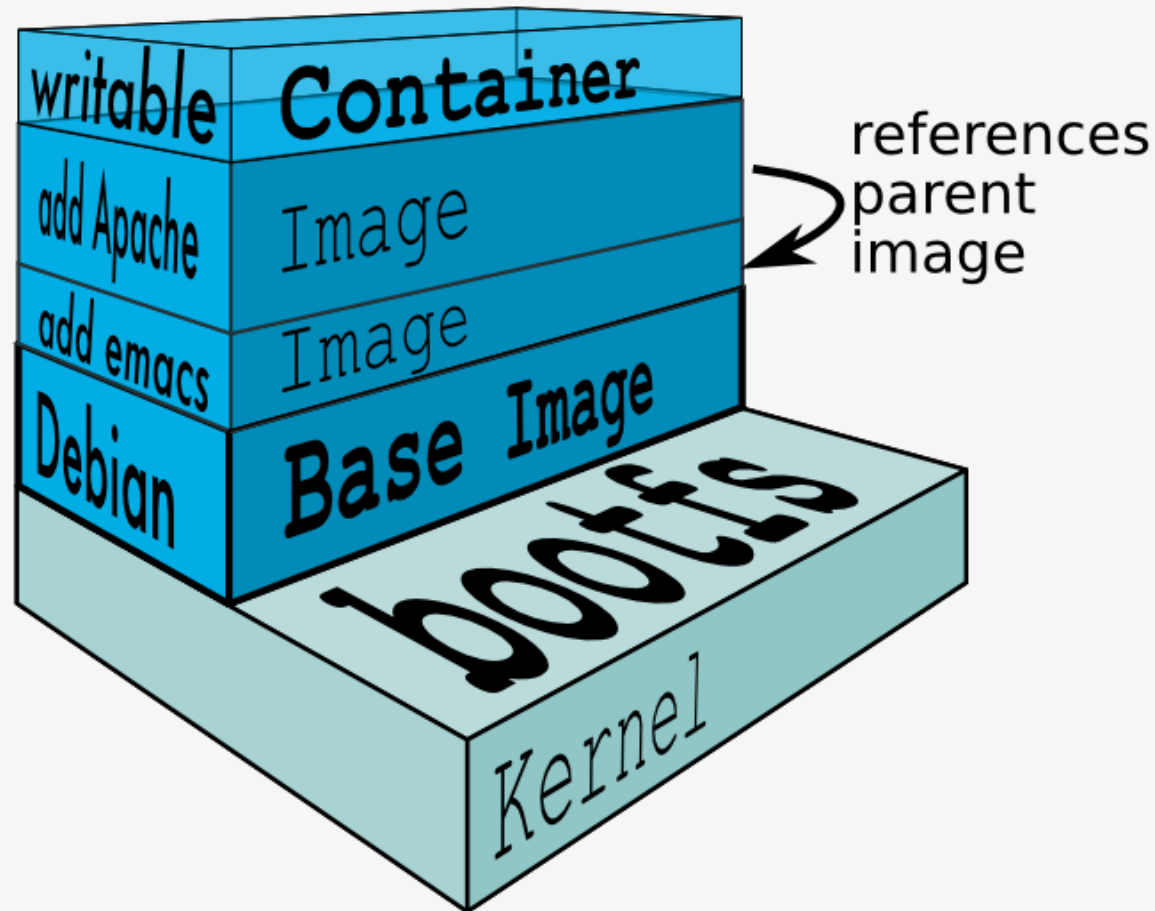
Type 2 Hypervisor



Kernel/ namespaces  
Virtualisierung

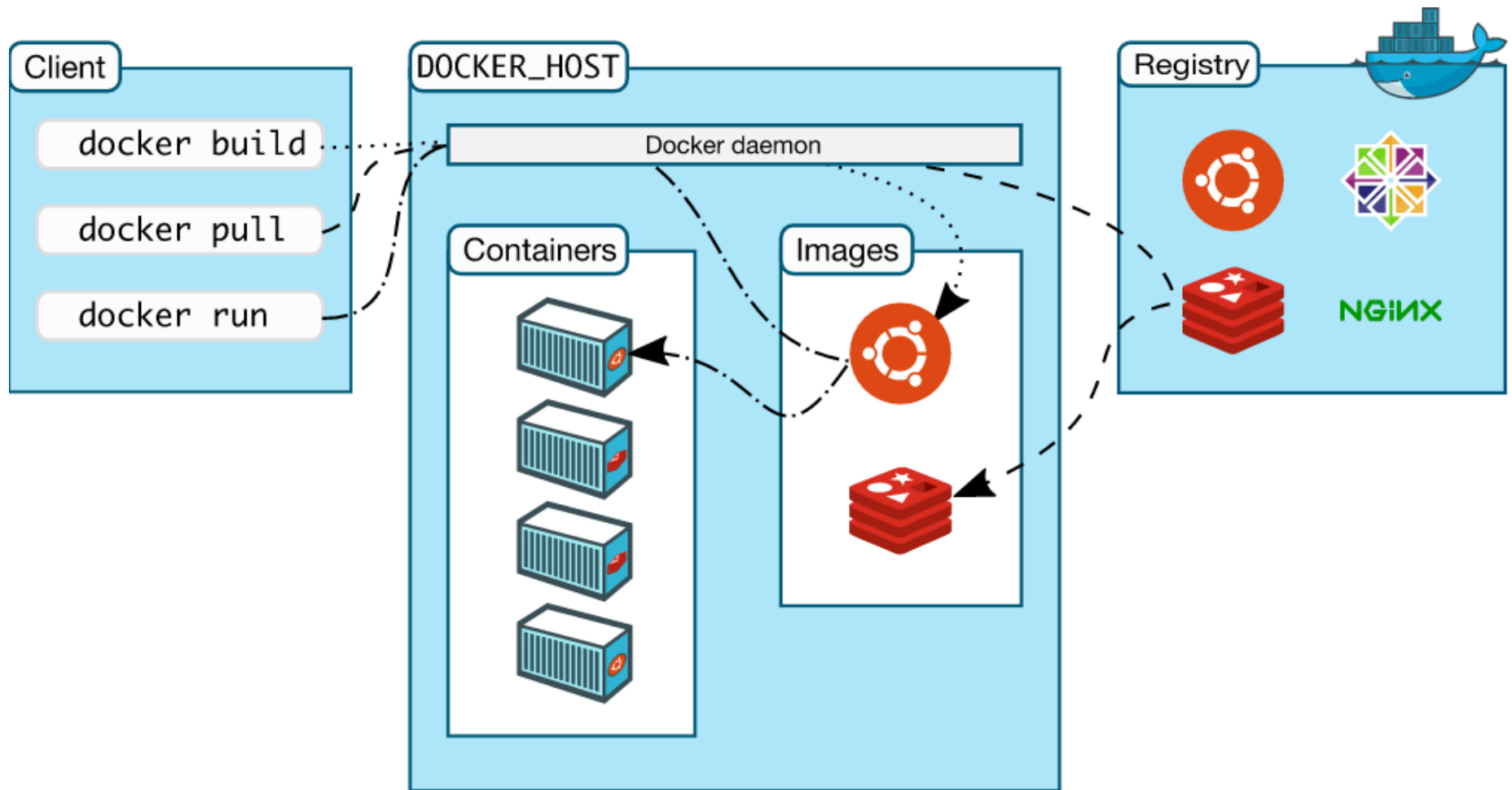
Quelle: Docker, Crisp Research, 2014

# Docker: Container / Images / Layers



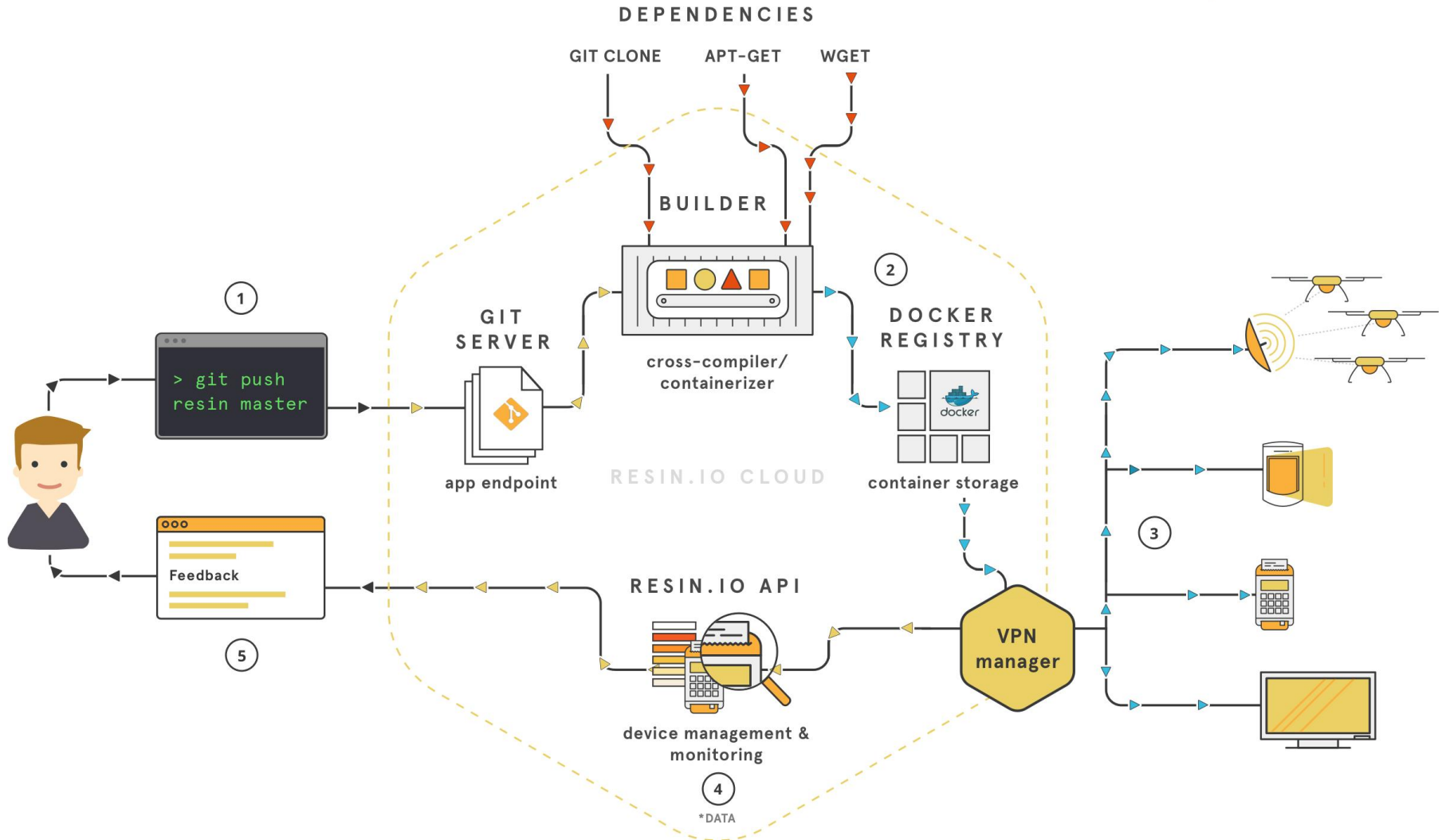
<http://stackoverflow.com/questions/24702233/docker-container-and-memory-consumption>

# Docker: Architektur

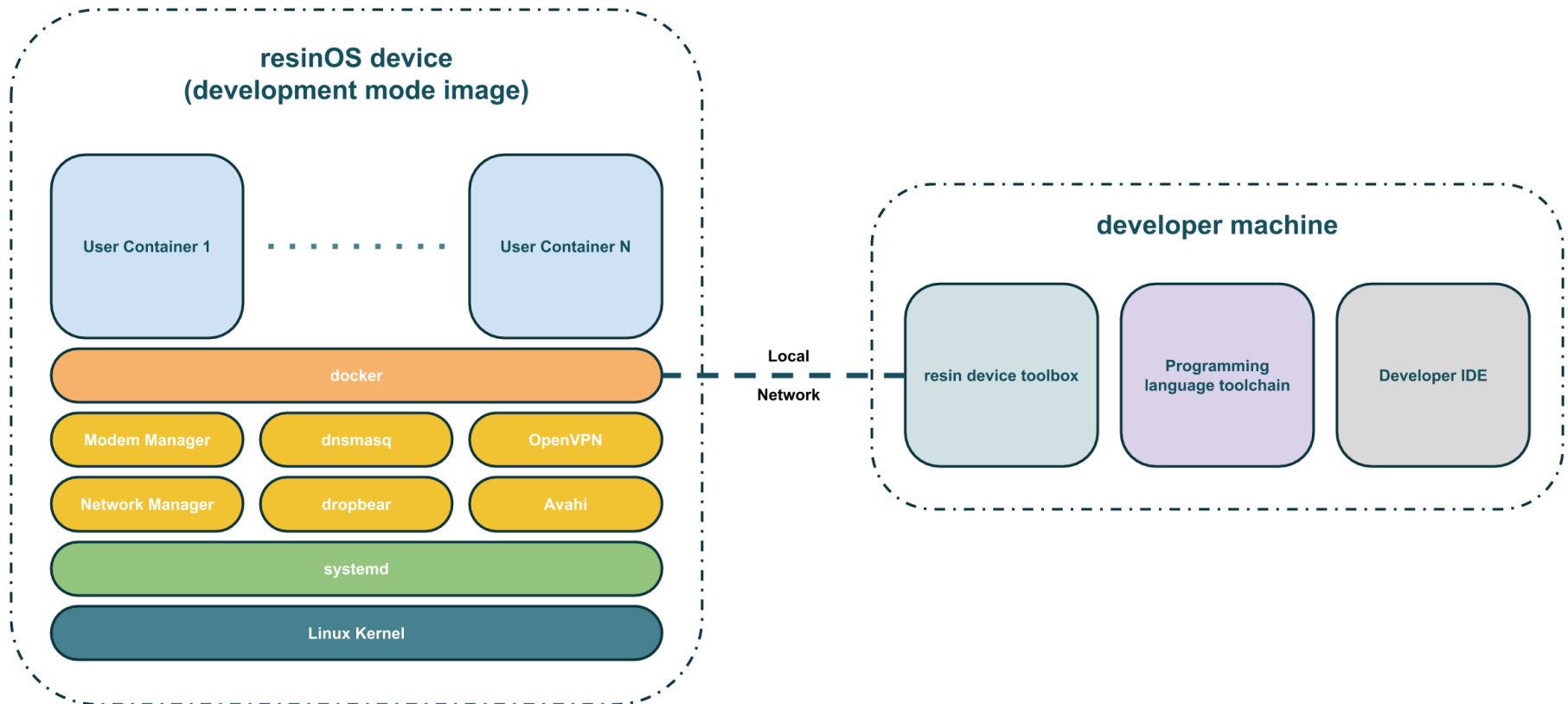


[ Docker ]

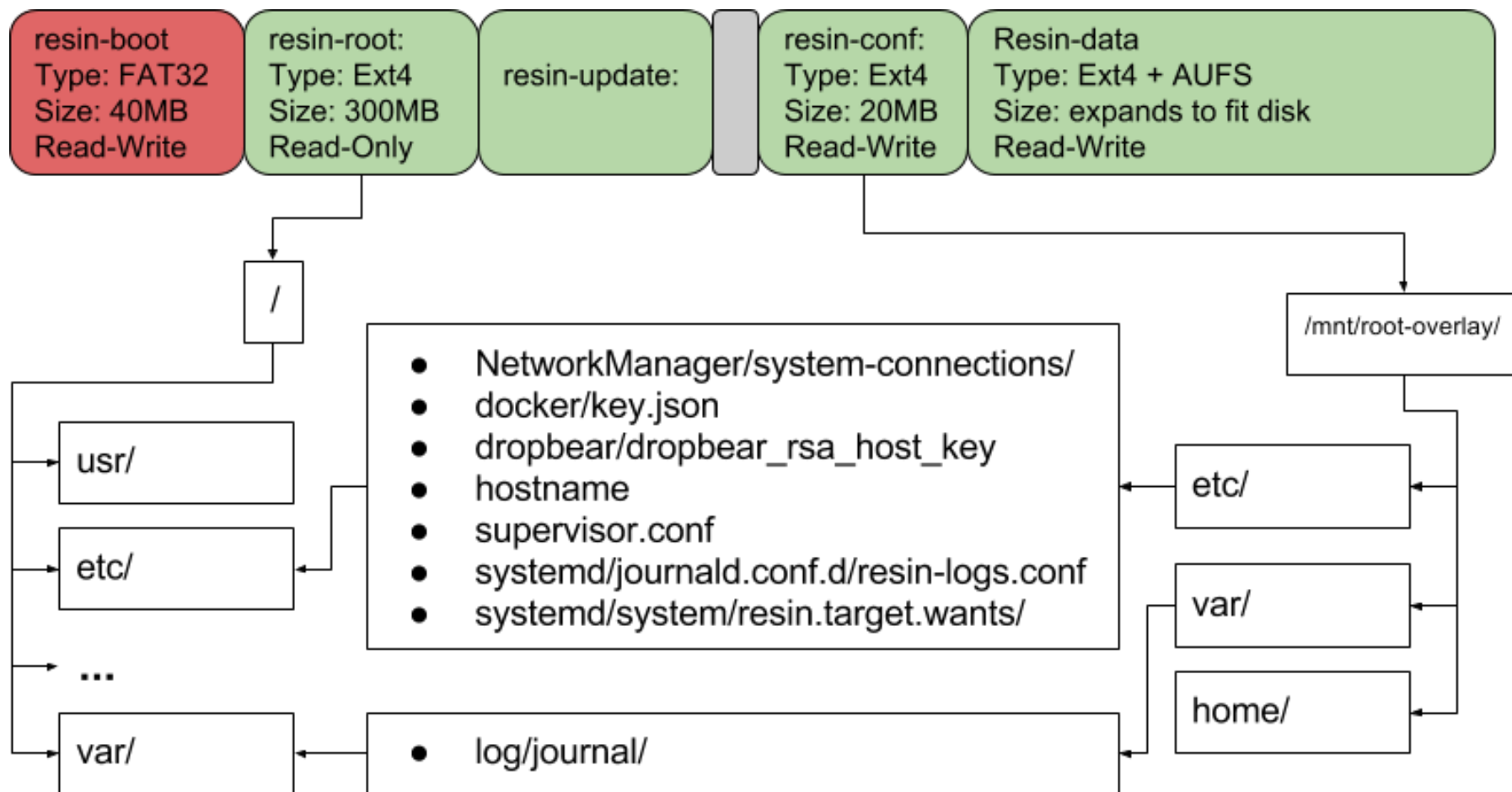
# Resin.io Architektur



# resinOS Architektur



# resinOS Architektur



# How To



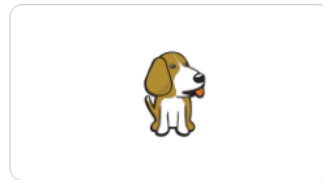
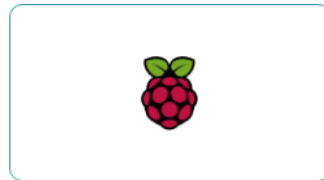
## Einführung

### resinOS

- Grundlagen
- Docker
- **resinOS**
- Live Demo

## Abschluss

## Image für das Gerät herunterladen



Raspberry Pi		
MODEL	DEVELOPMENT RELEASE	PRODUCTION RELEASE
Raspberry Pi 1	<a href="#">resinOS 2.0.0-beta.1</a>	Coming Soon
Raspberry Pi 2	<a href="#">resinOS 2.0.0-beta.1</a>	Coming Soon
Raspberry Pi 3	<a href="#">resinOS 2.0.0-beta.1</a>	Coming Soon



<https://resinos.io/#downloads-raspberrypi>



# How To



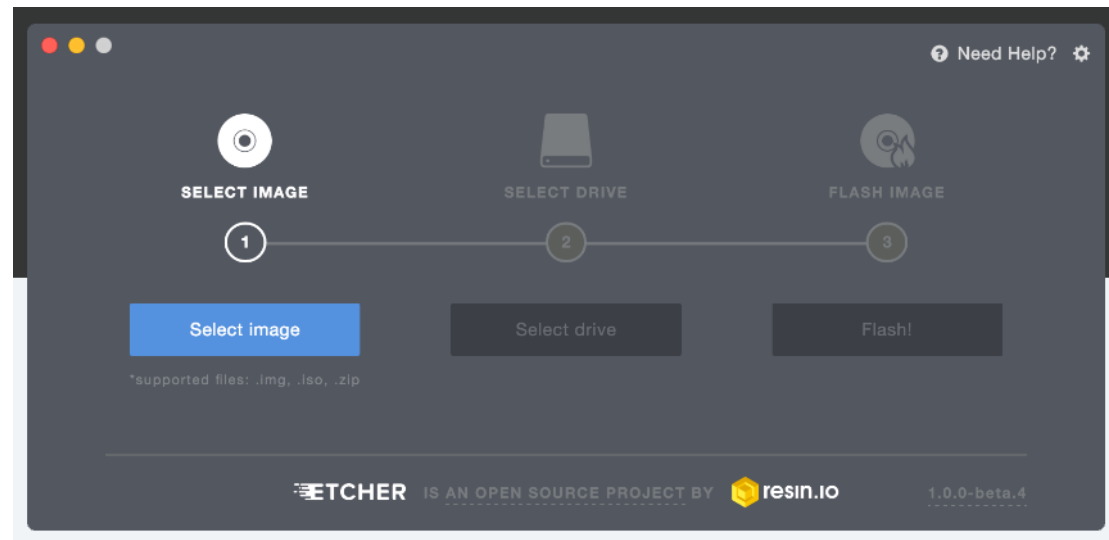
## Einführung

### resinOS

- Grundlagen
- Docker
- **resinOS**
- Live Demo

## Abschluss

Image auf SD Karte schreiben, z.B. mit SDFormatter und Win32DiskImager oder etcher.io



# How To



## Einführung

### resinOS

- Grundlagen
- Docker
- resinOS
- Live Demo

## Abschluss

Auf PC Resin Device Toolbox installieren

- NodeJS (6.x) installieren
- npm installieren
- rsync
- ssh
- Resin Device Toolbox

Ubuntu:

```
cd ~
```

```
curl -sL https://deb.nodesource.com/setup_6.x -o nodesetup.sh
```

```
sudo bash nodesetup.sh
```

```
sudo apt-get install nodejs git ssh rsync
```

```
sudo npm install --upgrade -g npm
```

```
sudo npm install -g resin-device-toolbox
```

# How To



## Einführung

### resinOS

- Grundlagen
- Docker
- **resinOS**
- Live Demo

## Abschluss

### rdt

```
Usage: rdt [COMMAND] [OPTIONS]
```

If you need help, or just want to say hi, don't hesitate in reaching out at:

```
GitHub: https://github.com/resin-os/resin-device-toolbox/issues/new  
Gitter: https://gitter.im/resin-io/chat
```

Primary commands:

```
ssh [deviceIp]           Get a shell into a resinOS device  
push [deviceIp]         Push your changes to a container on local resinOS device  
logs [deviceIp]        Get or attach to logs of a running container on a resinOS device  
scan                    Scan for resinOS devices in your local network  
configure <target>     (Re)configure a resinOS drive or image  
flash <image>          Flash an image to a drive  
version                 Output the version number  
help [command...]      Show help
```

Global Options:

```
--help, -h
```

# How To



Einführung

resinOS

- Grundlagen
- Docker
- **resinOS**
- Live Demo

Abschluss

rdt scan

```
Reporting scan results
-
host:          resin.local
address:       192.168.2.68
dockerInfo:
  Containers:   1
  ContainersRunning: 1
  ContainersPaused: 0
  ContainersStopped: 0
  Images:       6
  Driver:       aufs
  SystemTime:   2016-12-18T18:27:49.036607336Z
  KernelVersion: 4.1.21
  OperatingSystem: Resin OS 2.0.0-beta.1
  Architecture: armv7l
dockerVersion:
  Version:      1.10.3
  ApiVersion:   1.22
```

# How To



## Einführung

### resinOS

- Grundlagen
- Docker
- **resinOS**
- Live Demo

## Abschluss

Beispiel Repo klonen:

```
git clone https://github.com/nmaas87/resin-openwrt-flask-app.git
```

Ins Verzeichnis wechseln:

```
cd resin-openwrt-flask-app
```

Repo als Container auf den RPi im Netzwerk pushen

```
rdt push [IP] -s . -n <name>
```

(IP erhalten wir durch rdt scan)

Container Logs betrachten

```
rdt logs [IP]
```

Per SSH in Container hängen

```
rdt ssh [IP]
```

Der RPi kann über SSH an Port 22222 / User root erreicht werden

# How To



Einführung

**resinOS**

- Grundlagen
- Docker
- resinOS
- **Live Demo**

Abschluss

## Live Demo!

# Fragen?



Vielen Dank für Ihre Aufmerksamkeit!

# Quellenangaben



- Bilder, soweit nicht anders angegeben, von resin.io