Ansible für Java Entwickler

JUG Essen, 30.05.2017, Sandra Parsick

# Ansible

* Konfigurationsmanagement
* Software-Verteilung
* ad-hoc Commands ausführen
* *nicht speziell für Java, sondern allgemein*
* *schneller Einstieg, 0,5-1 PT*

# Konfigurationsmanagement

* Software
* Hardware
* Services (z.B. externe Dienstleiter)
* System (Infrastructure as Code) 🡪 für Ansible relevant

# Idee/Ansatz

Ansible-Skripte (YAML, alternative JSON) definieren Bereitstellung von Server und Deployment von Artakten.

Ausführungsmaschine: Ansible, Skripte

PUSH ↓

Zielmaschine: SSH-Login, Python 2.x

# Konzepte

* Inventories: Maschinen-Definitionen
* Templates: Jinja2 (<http://jinja.pocoo.org/docs/2.9/>)
* Rollen: je für ein System definierbar (Snippets), z.B. docker, git, postgres, …
* Includes (playbooks rufen playbooks auf)
* ad-hoc Commands, um remote auf mehreren Maschinen gleichzeitig Services zu starten/stoppen

# Szenario

* Ansible-Skripte im VCS
* CI triggert Ansible
* Ansible Tower: Bezahlvariante

# IDE-Support

* yEdit Plugin für YAML-Support verfügbar
* IntelliJ: YAML-Support von Haus aus, zusätzlich Ansible-Plugin verfügbar

# Features

* Vault
* Module für Docker
* Dynamische Inventories: AWS API nach Server abfragen
* Es gibt Ansible Container (s. hub.docker.com)

# Tests

* --check (Trockenlauf)
* --syntax-check
* Jenkins + Vagrant (Vagrant verwaltet VMs, z.B. VirtualBox)
* ServerSpec (testet Zielbedingungen: Systeme wie gewünscht installiert?)

# Ressourcen

<http://docs.ansible.com> (gute Doku)

<https://hub.docker.com/>: Suche nach Ansible

Buch: Ansible für DevOps (Leanpub): regelmäßig gepflegt, teilweise Aktionen für 2€

<https://github.com/sparsick/ansible-talk>

# Sonstiges

Rundeck.org, Einstieg über <https://github.com/rundeck/anvils-demo>