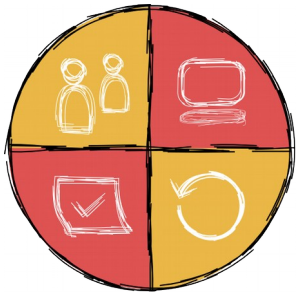
Frankfurter Entwicklertage 2016

Ansible für Entwickler

Sandra Parsick
info@sandra-parsick.de
@SandraParsick

Zur meiner Person

- Freiberufliche Softwareentwickler und Consultant im Java-Umfeld
- Schwerpunkte:
  - Java Enterprise Anwendungen
  - Agile Methoden
  - Software Craftmanship
  - Automatisierung von Entwicklungsprozessen
- Softwerkskammer Dortmund



Agenda

1. Ansible – Was ist das?
2. Warum ist es für Entwickler interessant?
3. Einführung in Ansible
4. Wie unterscheidet sich Ansible zur seiner Konkurrenz?
5. Weitere Einsatzszenarien aus Entwicklersicht

Ansible
Was ist das?

Ansible

- Software für
  - Konfigurationsmanagement,
  - Softwareverteilung und
  - Ad-hoc-Kommando-Ausführung

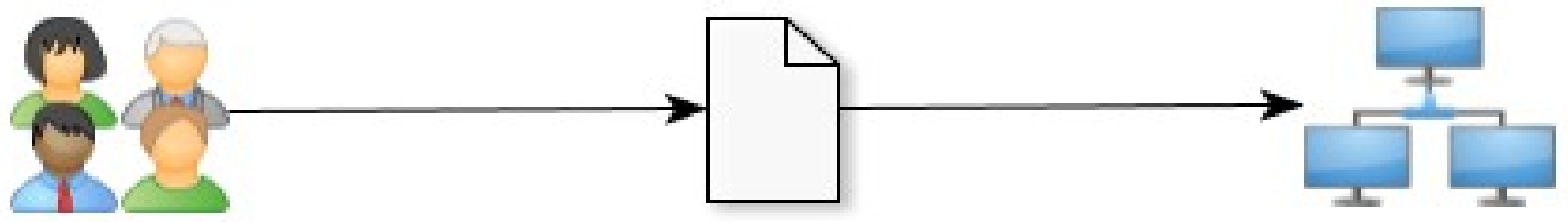
Konfigurationsmanagement (KM)

„Das KM umfasst alle technischen, organisatorischen und beschlussfassenden Maßnahmen und Strukturen, die sich mit der Konfiguration (Spezifikation) eines Produkts befassen.“
https://www.projektmagazin.de/glossarterm/konfigurationsmanagement

Konfigurationsmanagement (KM)

- Softwarekonfiguration
- Hardwarekonfiguration
- Dienstleistungskonfiguration
- Systemkonfiguration

Systemkonfiguration - „Infrastructure As Code“



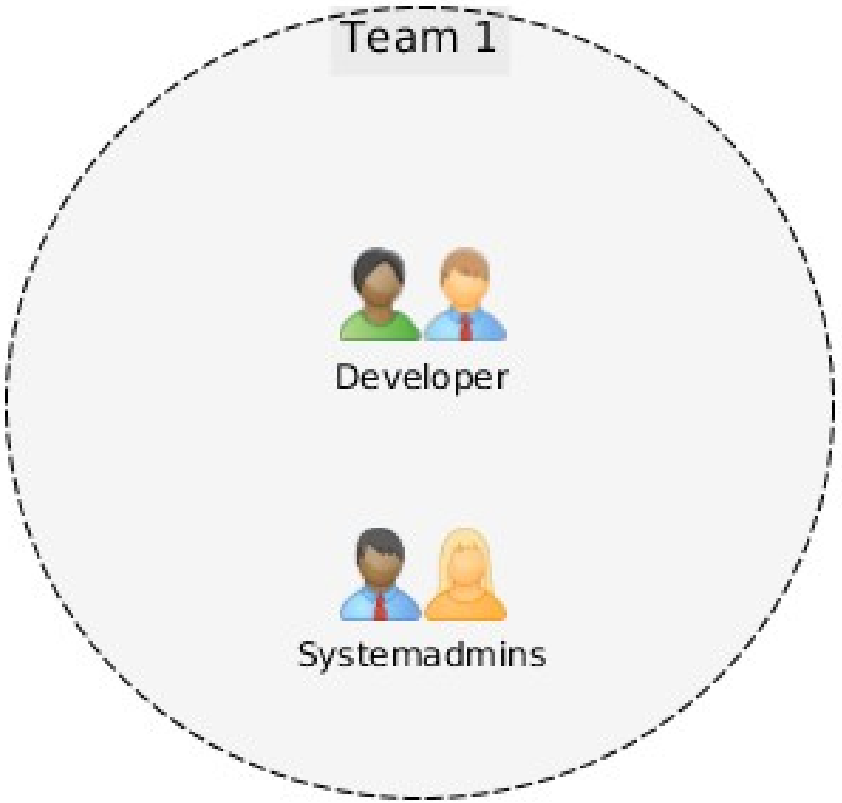
Systemkonfiguration - „Infrastructure As Code“

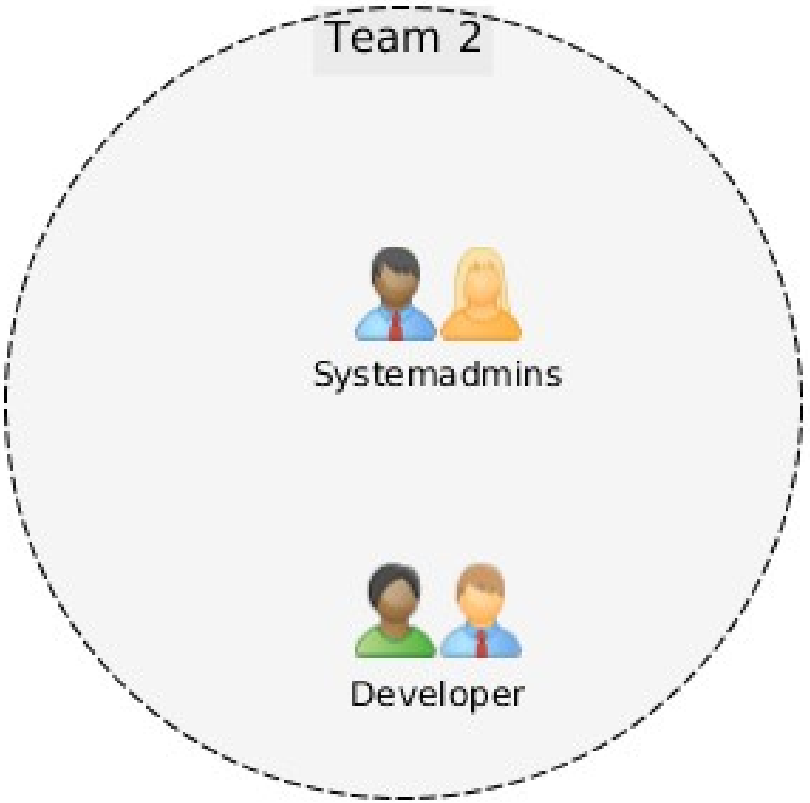
Ansible

Warum ist es für Entwickler interessant?

Systemkonfiguration für Entwickler

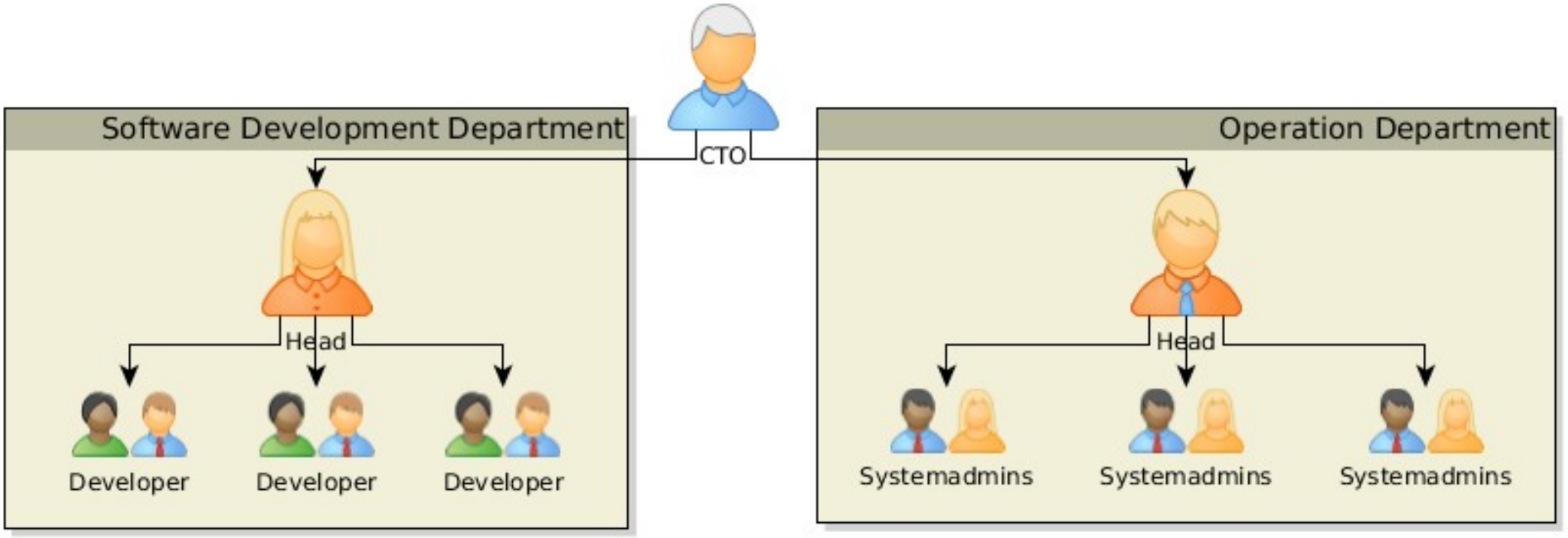
Organisatorische Ausgangslage
Wunsch





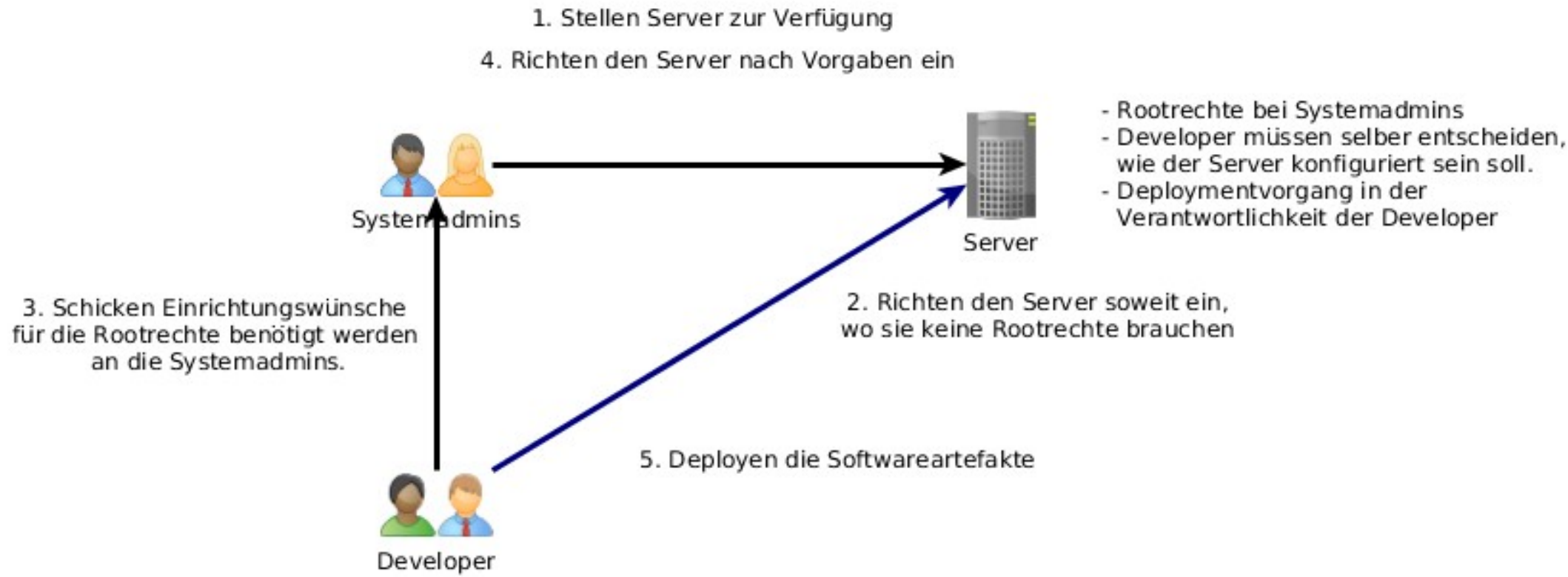
Systemkonfiguration für Entwickler

Organisatorische Ausgangslage
Realität



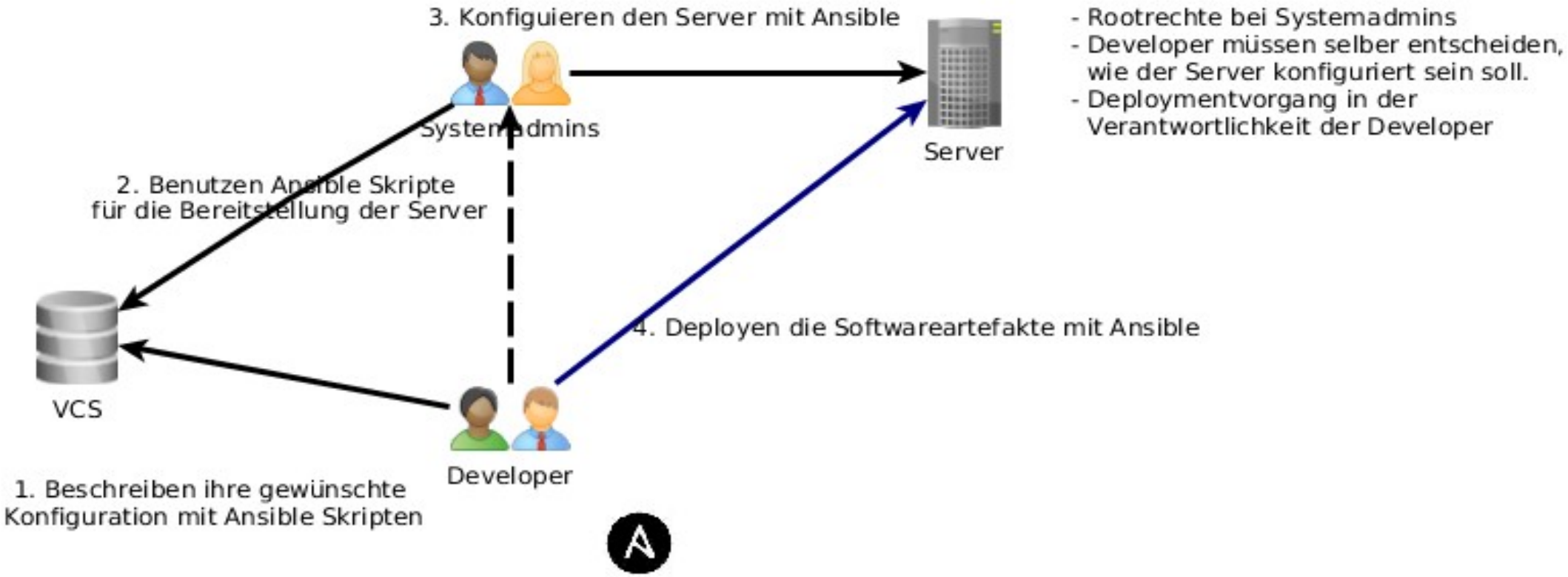
Systemkonfiguration für Entwickler

Prozess zwischen Development und Operation



Systemkonfiguration für Entwickler

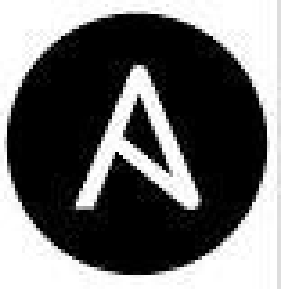
Lösungidee mit Ansible



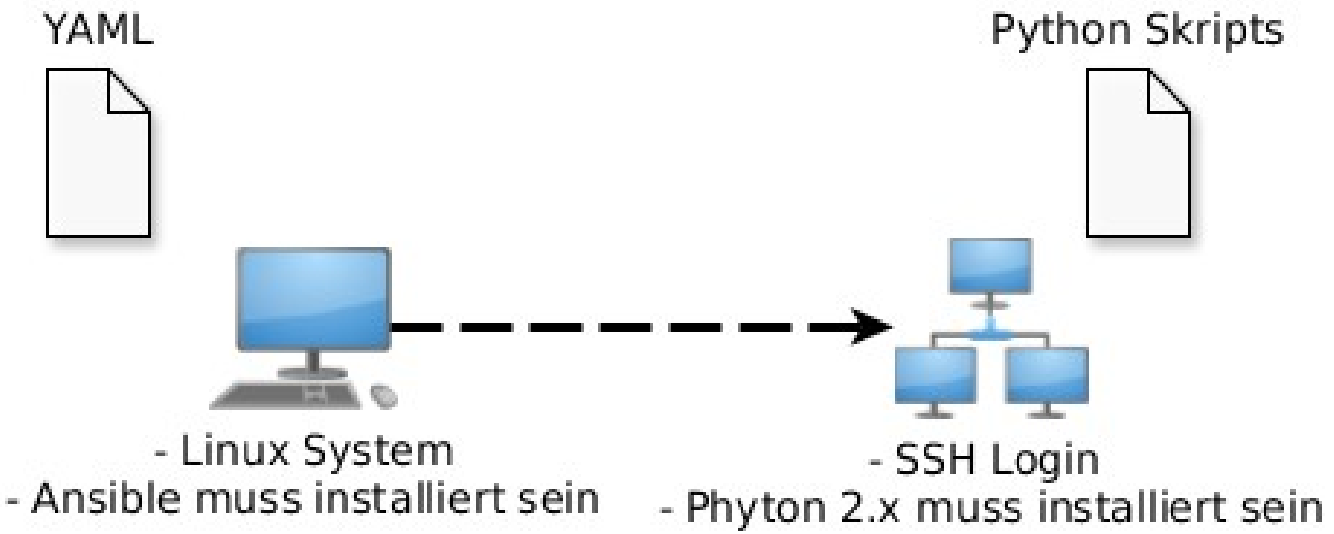
Einführung in Ansible

Ansible

- Software für
  - Konfigurationsmanagement,
  - Softwareverteilung und
  - Ad-hoc-Kommando-Ausführung
- Sprache: Python
- Ansible Skripte: YAML



Funktionsweise



Exkurs: YAML

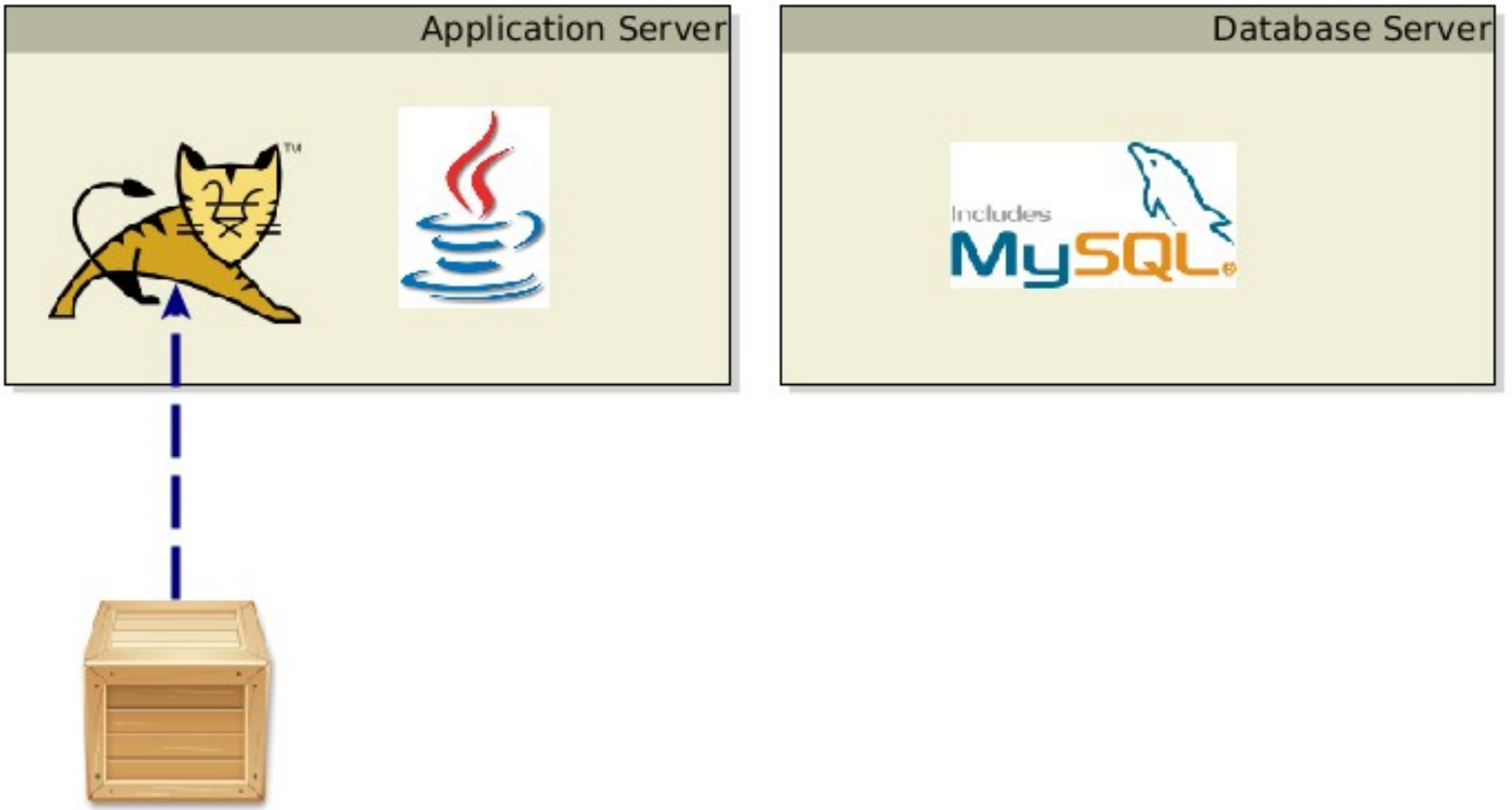
YAML

```yaml
---
foo: "bar"
baz:
  - "qux"
  - "quxx"
corge: null
grault: 1
garply: true
waldo: "false"
fred: "undefined"
emptyArray: []
emptyObject: {}
emptyString: ""
```

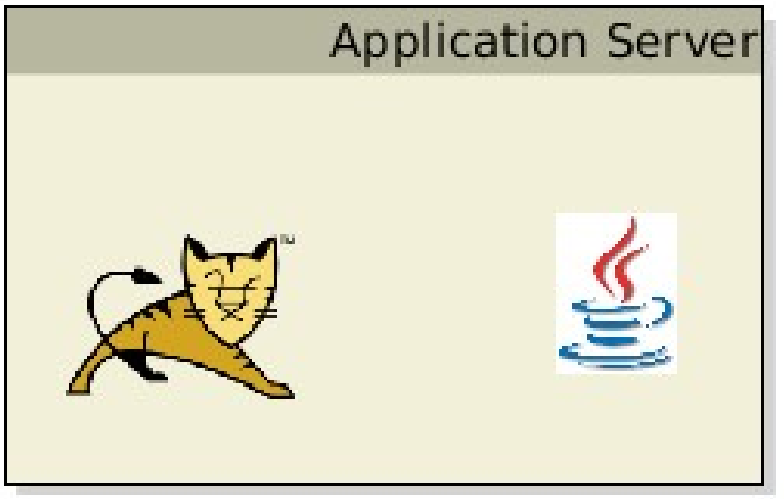
JSON

```json
{
  "foo": "bar",
  "baz": [
    "qux",
    "quxx"
  ],
  "corge": null,
  "grault": 1,
  "garply": true,
  "waldo": "false",
  "fred": "undefined",
  "emptyArray": [],
  "emptyObject": {},
  "emptyString": ""
}
```

Ansible Beispiel



Setup Application Server Playbook



```yaml
- hosts: application-server
  vars:
    tomcat_version: 8.0.24
    tomcat_base_name: apache-tomcat-{{ tomcat_version }}
    #catalina_opts: "-Dkey=value"

  tasks:
    - name: install java
      apt: name=openjdk-7-jdk state=present
      become: yes
      become_method: sudo

    - name: Download current Tomcat 8 version
      local_action: get_url url="http://archive.apache.org/dist/tomcat/tomcat-8/v{{ tomcat_version }}/bin/{{ tomcat_base_name }}.tar.gz" dest=/tmp

    - name:
      file: name=/opt mode=777
      become: yes
      become_method: sudo

    - name: Install Tomcat 8
      unarchive: src=/tmp/{{ tomcat_base_name }}.tar.gz dest=/opt creates=/opt/{{ tomcat_base_name }} owner=vagrant group=vagrant

    - name: Set link to tomcat 8
      file: src=/opt/{{ tomcat_base_name }} dest=/opt/tomcat  state=link force=yes

    - name: setup setenv.sh
      template: dest="/opt/{{ tomcat_base_name }}/bin/setenv.sh" src="roles/tomcat8/templates/setenv.sh.j2" mode=755
      when: catalina_opts is defined

    - shell: ls /opt/{{ tomcat_base_name }}/bin/*.sh
      register: tomcat_scripts
      ignore_errors: yes
```

Inventories

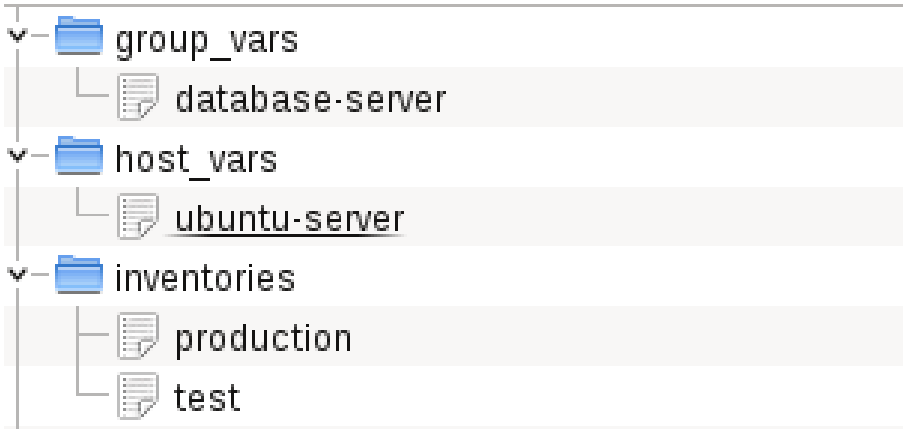
Production

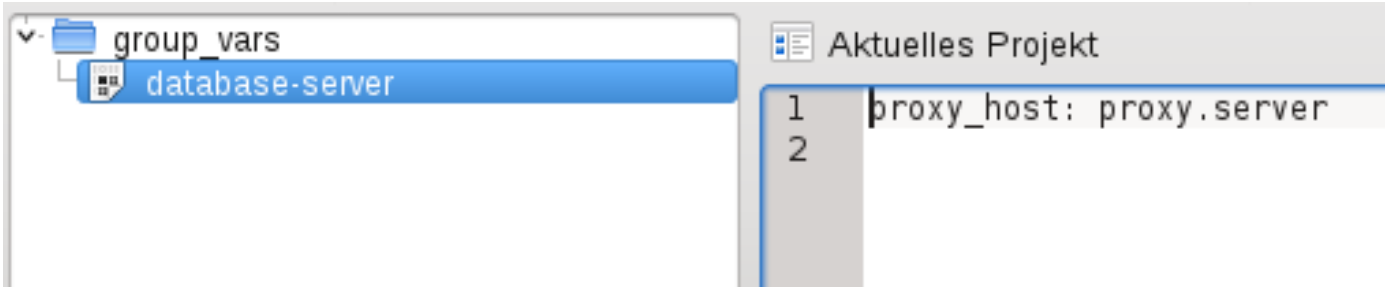
```
[application-server]
192.168.33.10
ubuntu-server db_host=mysql01

[mysql-db-server]
mysql[01:10]

[oracle-db-server]
db-[a:f].oracle.company.com

[database-server:children]
mysql-db-server
oracle-db-server

[application-server:vars]
message="Welcome"

[database-server:vars]
message="Hello World!"
```

Test

```
[application-server]
192.168.33.10

[database-server]
192.168.33.10

```

Inventories





```
12
13      - name: Download current Tomcat 8 version
14        local_action: get_url url="http://archive.apache.org/dist/tomcat/tomcat-8/v{{ tomcat_version
          }}/bin/{{ tomcat_base_name }}.tar.gz" dest=/tmp
15
16      - name:
17        file: name=/opt mode=777
18        become: yes
19        become_method: sudo
20
21      - name: Install Tomcat 8
22        unarchive: src=/tmp/{{ tomcat_base_name }}.tar.gz dest=/opt creates=/opt/{{ tomcat_base_name
          }} owner=vagrant group=vagrant
23
24      - name: Set link to tomcat 8
25        file: src=/opt/{{ tomcat_base_name }} dest=/opt/tomcat  state=link force=yes
26
27      - name: setup setenv.sh
28        template: dest="/opt/{{ tomcat_base_name }}/bin/setenv.sh"
          src="roles/tomcat8/templates/setenv.sh.j2" mode=755
29        when: catalina_opts is defined
30
31      - shell: ls /opt/{{ tomcat_base_name }}/bin/*.sh
32        register: tomcat_scripts
33        ignore_errors: yes
34
35      - name: ensure tomcat scripts are executable
36        file: name={{item}} mode=755
37        with_items: tomcat_scripts.stdout_lines
38
39      - name: install init.d script for tomcat
40        copy: src=roles/tomcat8/files/init.d/tomcat dest=/etc/init.d/tomcat owner=vagrant
          group=vagrant mode=755
41        become: yes
42        become_method: sudo
```

Templates

- setenv.sh.j2

```
1    CATALINA_OPTS="{{ catalina_opts }}"
```

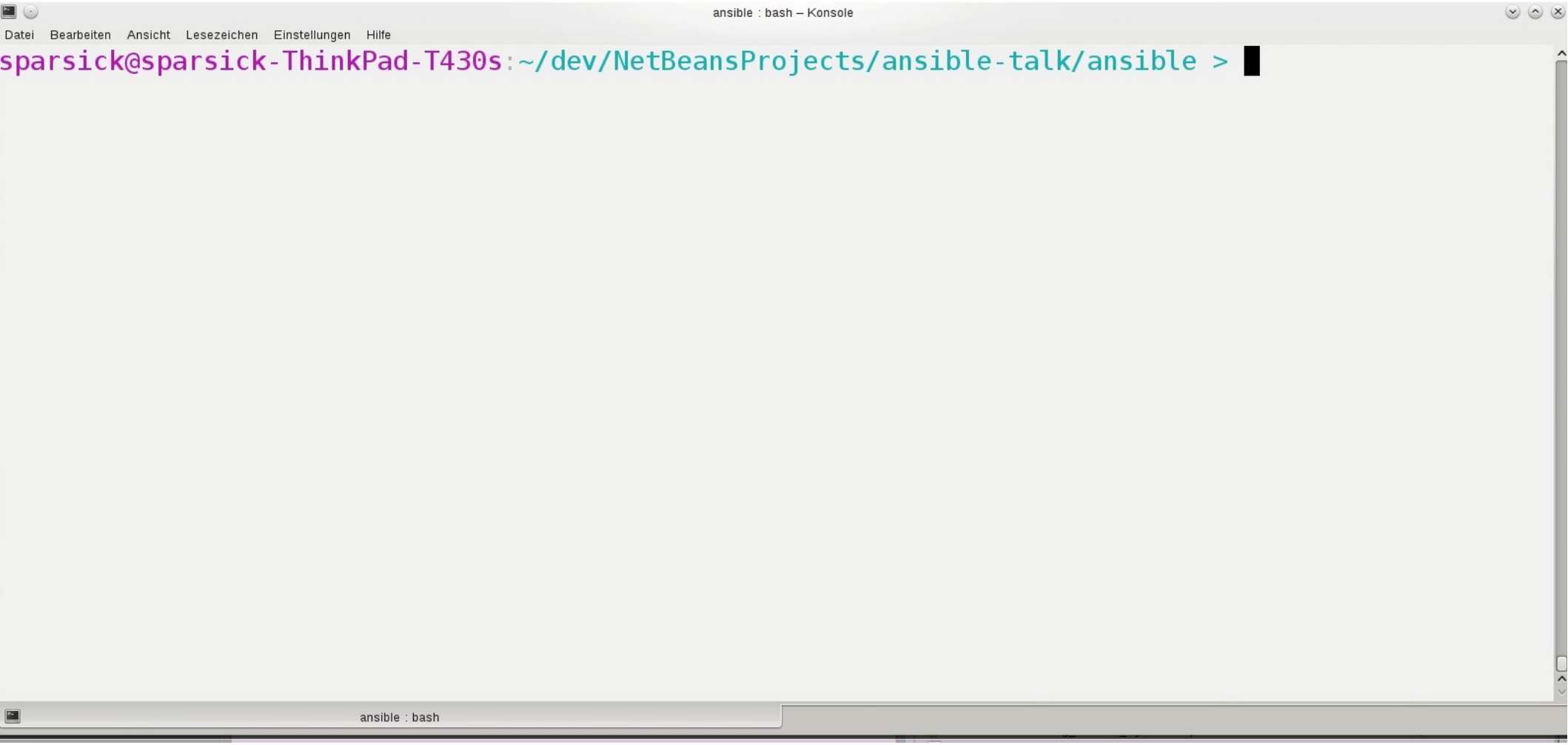
Templates - Jinja2

- Templating engine für Python

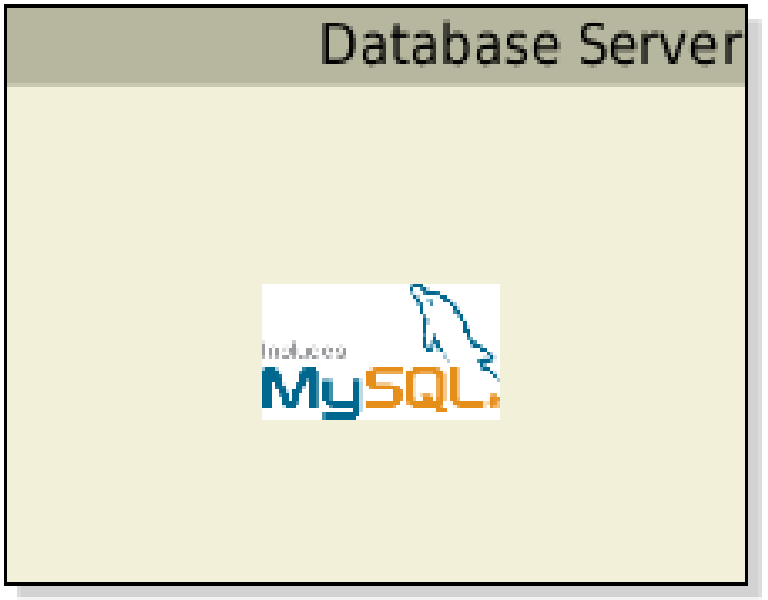
```
<title>{% block title %}{% endblock %}</title>
<ul>
{% for user in users %}
  <li><a href="{{ user.url }}">{{ user.username }}</a></li>
{% endfor %}
</ul>
```

- Mehr Information unter http://jinja.pocoo.org/docs/dev/

Setup Application Server Playbook



Setup Database Server Playbook



```yaml
- hosts: database-server
  become: yes
  become_method: sudo

  tasks:
    - name: install mysql db
      apt: name=mysql-server state=present

    - name: installs python-mysqldb
      apt: name=python-mysqldb state=present

    - name: start mysql
      service: name=mysql state=started

    - name: set bind address
      lineinfile: dest=/etc/mysql/my.cnf
        line='bind-address = 0.0.0.0'
        state=present
        regexp=^bind-address(.*)
      notify: restart mysql

    - name: creates db user dba
      mysql_user: name=dba password=g3h31m priv=*.*:ALL,GRANT state=present host=%

  handlers:
     - name: restart mysql
       service: name=mysql state=restarted

```

Setup Database Server Playbook

Setup Database Server Playbook

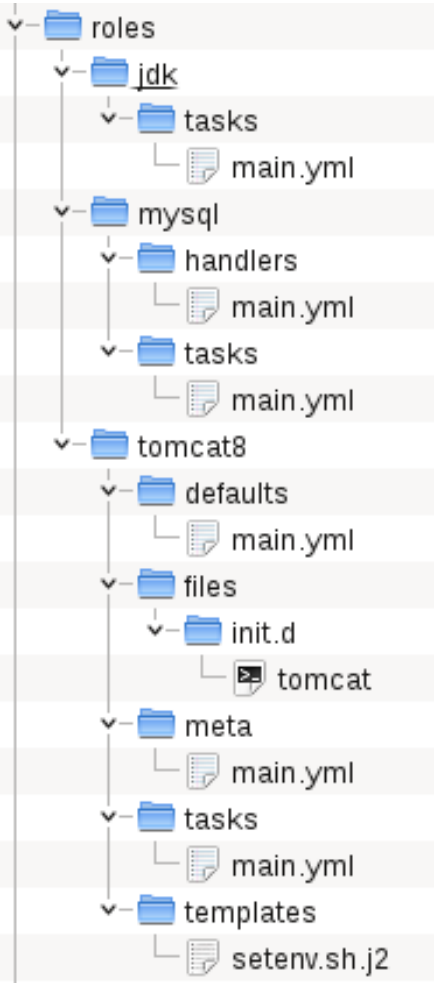
```yaml
- hosts: application-server
  vars:
    tomcat_version: 8.0.24
    tomcat_base_name: apache-tomcat-{{ tomcat_version }}
    #catalina_opts: "-Dkey=value"

  tasks:
    - name: install java
      apt: name=openjdk-7-jdk state=present
      become: yes
      become_method: sudo

    - name: Download current Tomcat 8 version
      local_action: get_url url="http://archive.apache.org/dist/tomcat/tomcat-8/v{{ tomcat_version }}/bin/{{ tomcat_base_name }}.tar.gz" dest=/tmp

    - name:
      file: name=/opt mode=777
      become: yes
      become_method: sudo

    - name: Install Tomcat 8
      unarchive: src=/tmp/{{ tomcat_base_name }}.tar.gz dest=/opt creates=/opt/{{ tomcat_base_name }} owner=vagrant group=vagrant

    - name: Set link to tomcat 8
      file: src=/opt/{{ tomcat_base_name }} dest=/opt/tomcat  state=link force=yes

    - name: setup setenv.sh
      template: dest="/opt/{{ tomcat_base_name }}/bin/setenv.sh" src="roles/tomcat8/templates/setenv.sh.j2" mode=755
      when: catalina_opts is defined

    - shell: ls /opt/{{ tomcat_base_name }}/bin/*.sh
      register: tomcat_scripts
      ignore_errors: yes
```

```
12
13      - name: Download current Tomcat 8 version
14        local_action: get_url url="http://archive.apache.org/dist/tomcat/tomcat-8/v{{ tomcat_version
          }}/bin/{{ tomcat_base_name }}.tar.gz" dest=/tmp
15
16      - name:
17        file: name=/opt mode=777
18        become: yes
19        become_method: sudo
20
21      - name: Install Tomcat 8
22        unarchive: src=/tmp/{{ tomcat_base_name }}.tar.gz dest=/opt creates=/opt/{{ tomcat_base_name
          }} owner=vagrant group=vagrant
23
24      - name: Set link to tomcat 8
25        file: src=/opt/{{ tomcat_base_name }} dest=/opt/tomcat  state=link force=yes
26
27      - name: setup setenv.sh
28        template: dest="/opt/{{ tomcat_base_name }}/bin/setenv.sh"
          src="roles/tomcat8/templates/setenv.sh.j2" mode=755
29        when: catalina_opts is defined
30
31      - shell: ls /opt/{{ tomcat_base_name }}/bin/*.sh
32        register: tomcat_scripts
33        ignore_errors: yes
34
35      - name: ensure tomcat scripts are executable
36        file: name={{item}} mode=755
37        with_items: tomcat_scripts.stdout_lines
38
39      - name: install init.d script for tomcat
40        copy: src=roles/tomcat8/files/init.d/tomcat dest=/etc/init.d/tomcat owner=vagrant
          group=vagrant mode=755
41        become: yes
42        become_method: sudo
```

```yaml
- hosts: database-server
  become: yes
  become_method: sudo

  tasks:
    - name: install mysql db
      apt: name=mysql-server state=present

    - name: installs python-mysqldb
      apt: name=python-mysqldb state=present

    - name: start mysql
      service: name=mysql state=started

    - name: set bind address
      lineinfile: dest=/etc/mysql/my.cnf
        line='bind-address = 0.0.0.0'
        state=present
        regexp=^bind-address(.*)
      notify: restart mysql

    - name: creates db user dba
      mysql_user: name=dba password=g3h31m priv=*.*:ALL,GRANT state=present host=%

  handlers:
     - name: restart mysql
       service: name=mysql state=restarted

```

Roles

```
roles/
    common/
      files/
      templates/
      tasks/
      handlers/
      vars/
      defaults/
      meta/
```



Setup Playbooks mit Roles

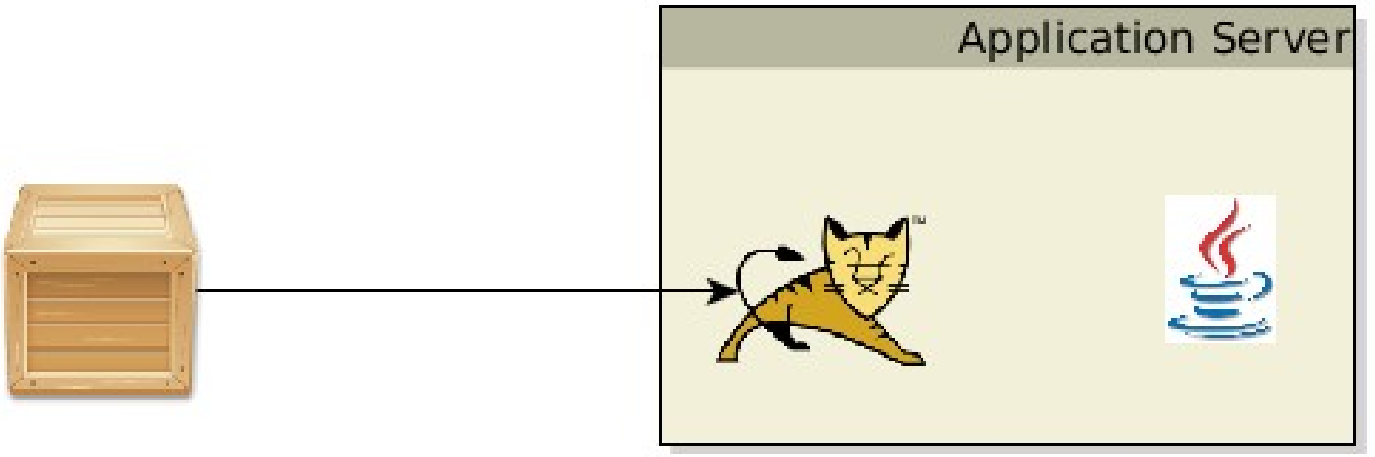
- Setup Application Server

```
1  - hosts: application-server
2    roles:
3      - jdk
4      - { role: tomcat8, tomcat_version: 8.0.30 }
5
6
7
```

- Setup Database Server

```
1  - hosts: database-server
2    roles:
3      - mysql
4
5
6
```

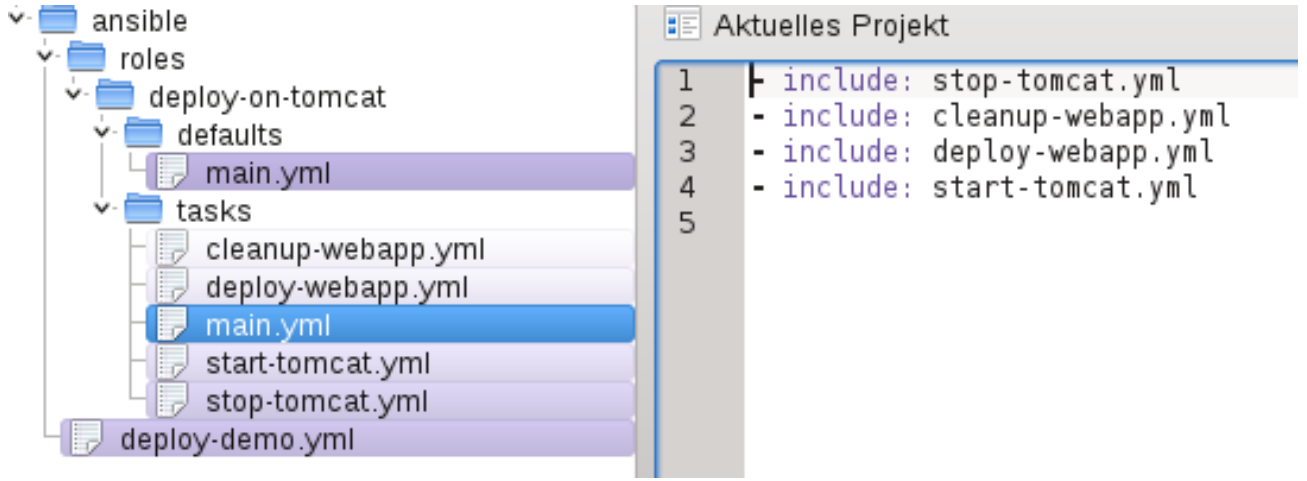
Java Webapplikation Deployment



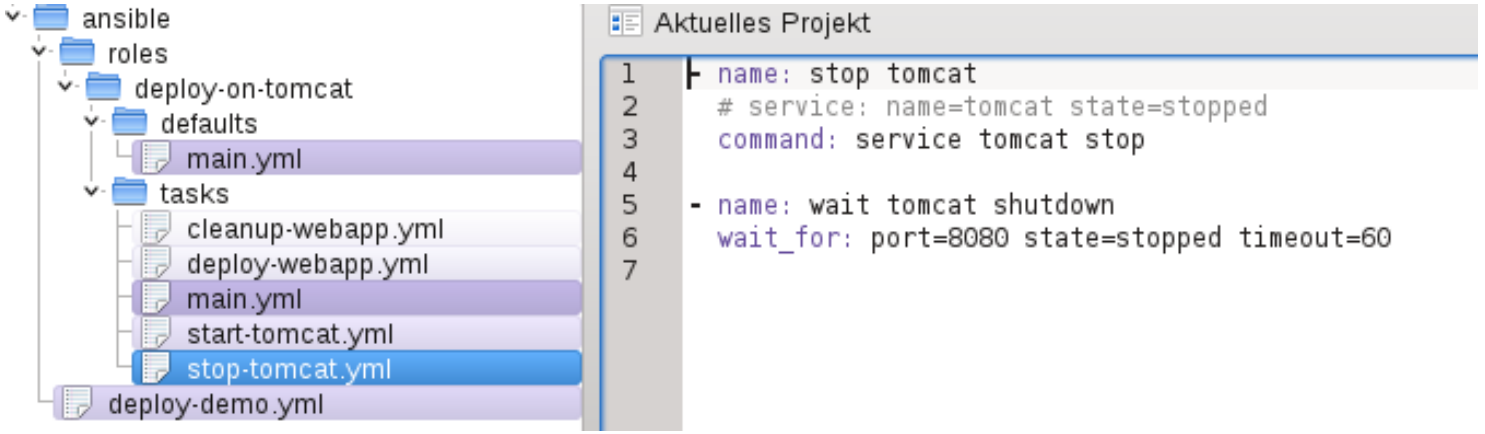
Deploy Application Playbook

```
1  - hosts: application-server
2    roles:
3      - {role: deploy-on-tomcat, webapp_source_path: ./demo-app-ansible-deploy-1.0-
4      SNAPSHOT.war, webapp_target_name: demo }
```

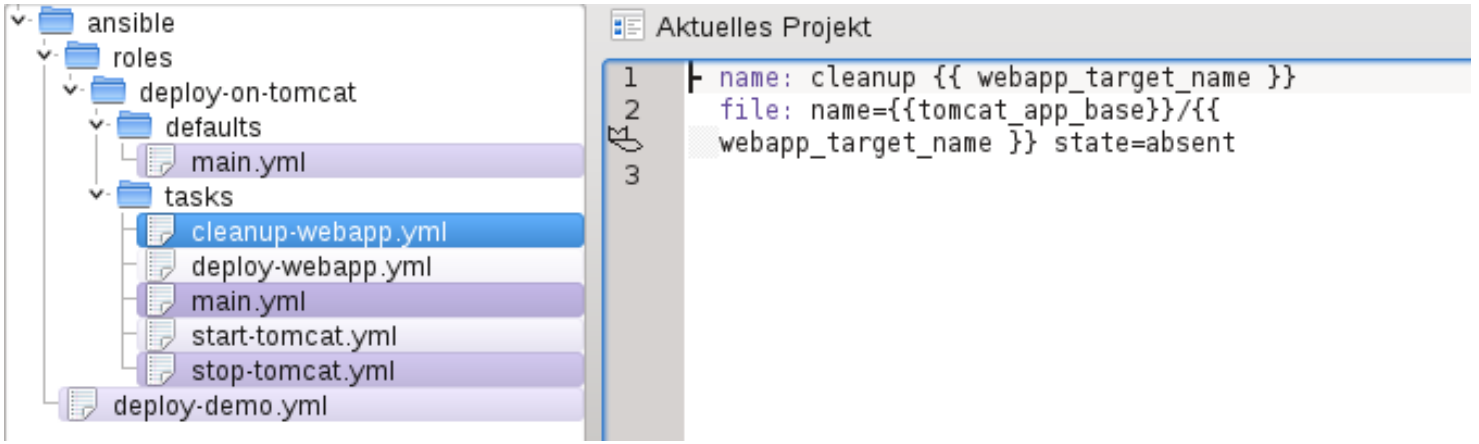
deploy-on-tomcat Role



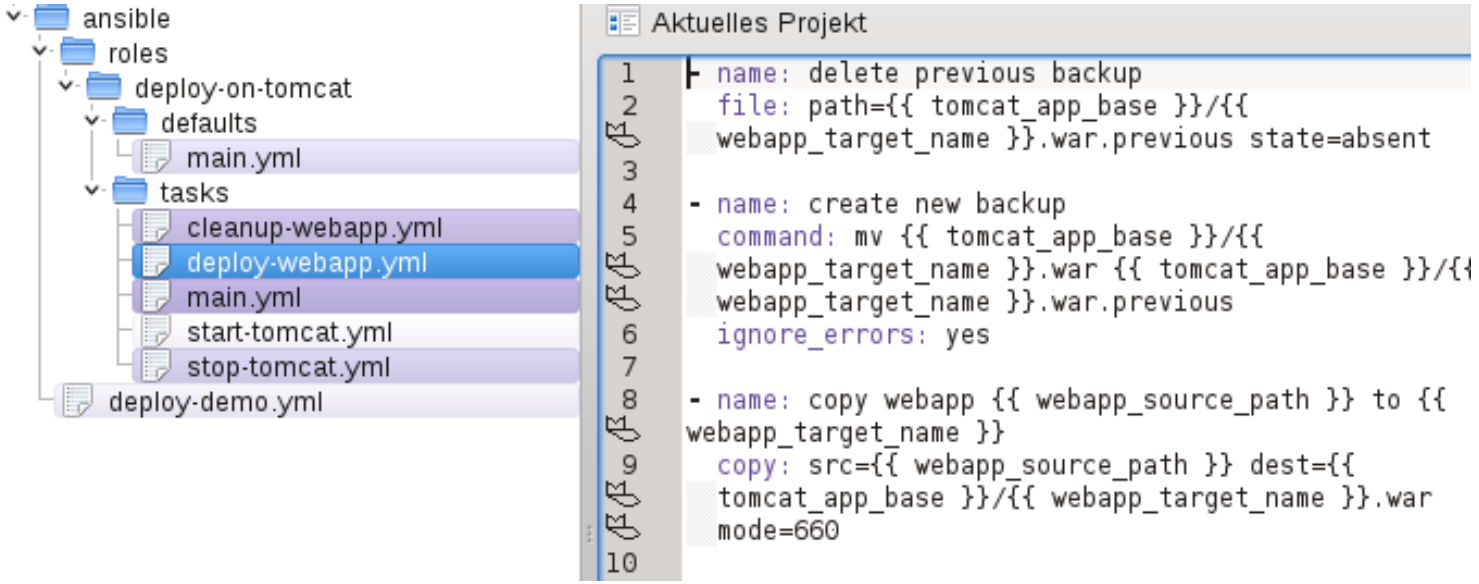
deploy-on-tomcat Role



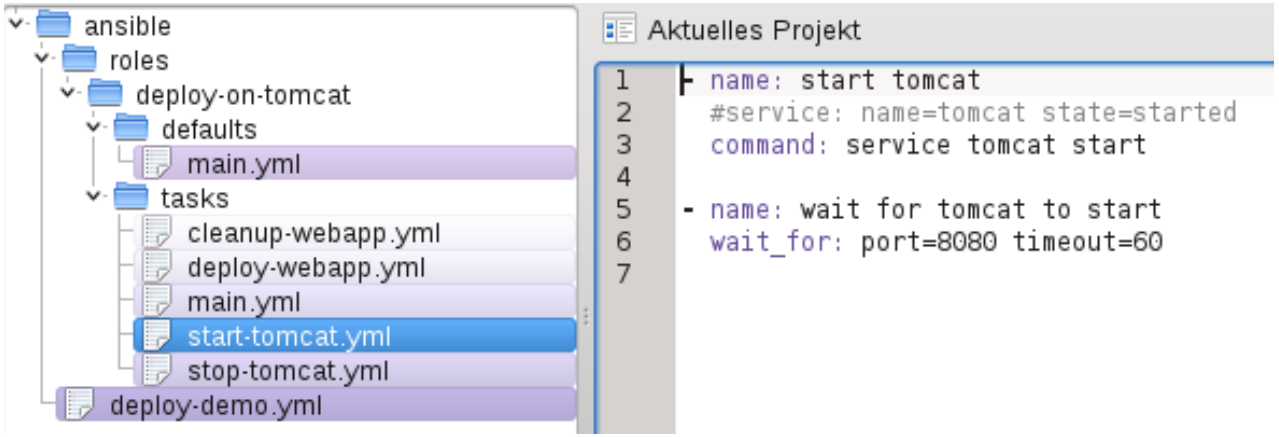
deploy-on-tomcat Role



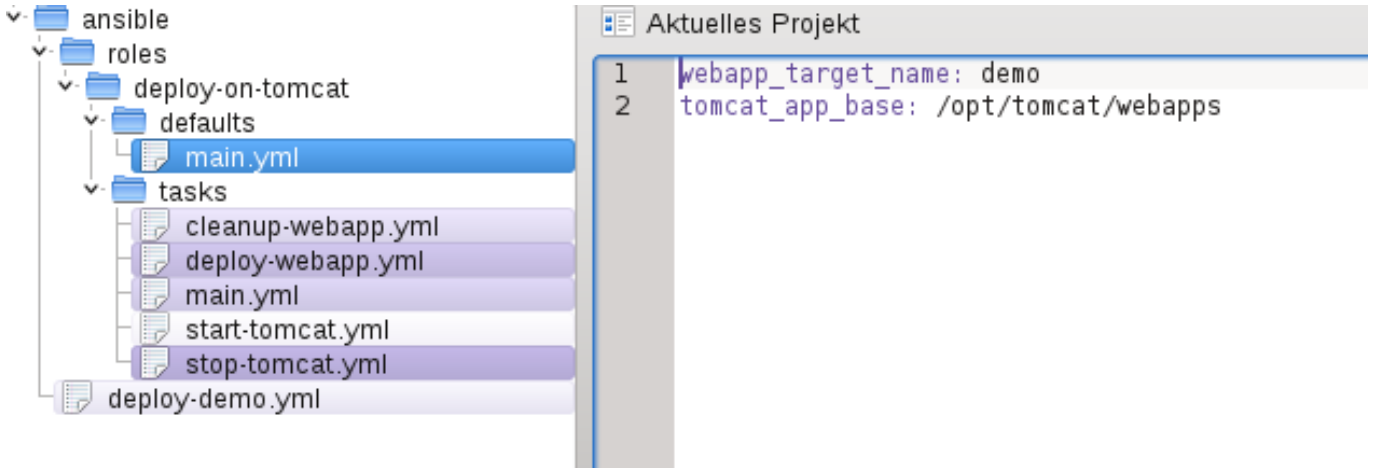
deploy-on-tomcat Role



deploy-on-tomcat Role



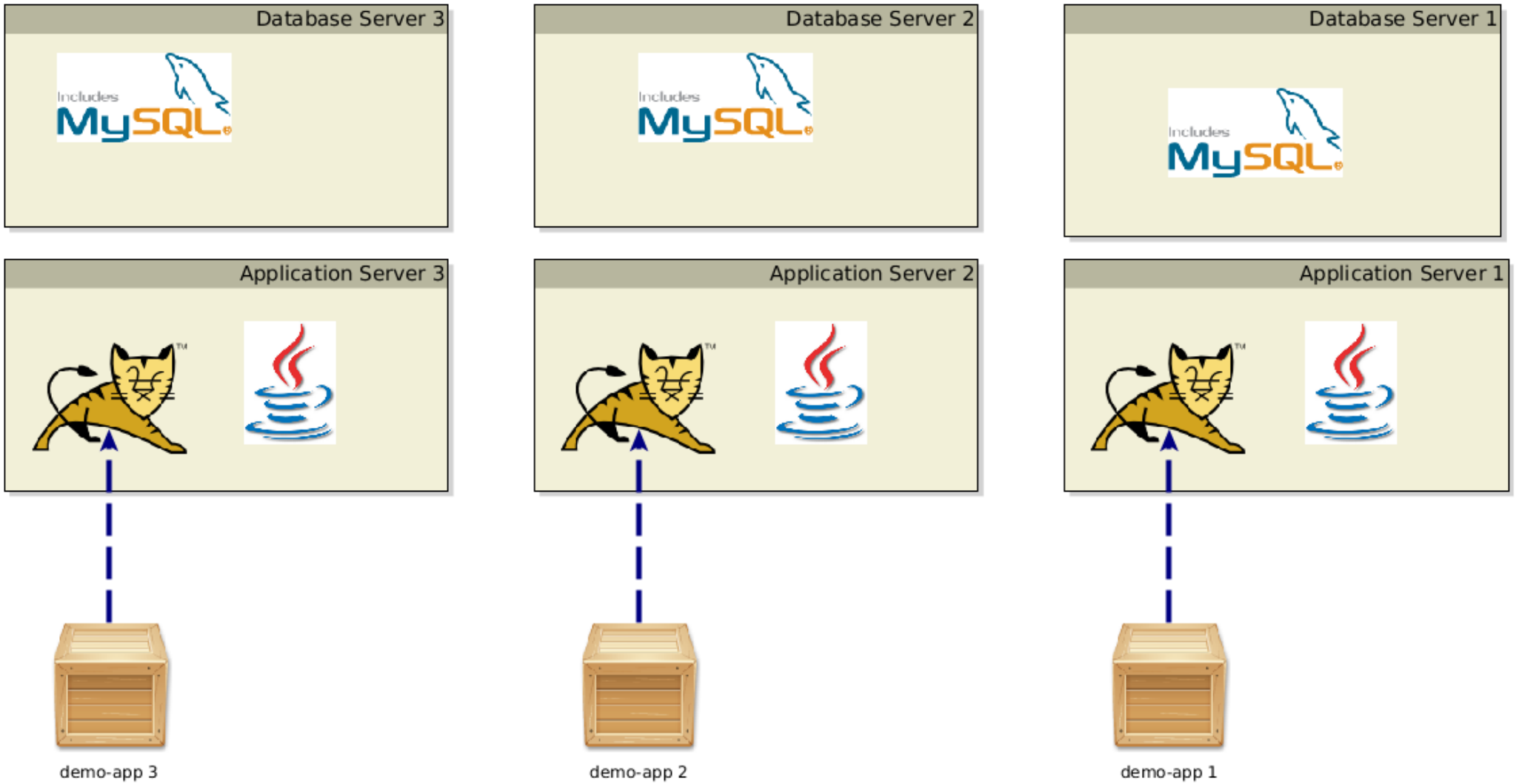
deploy-on-tomcat Role



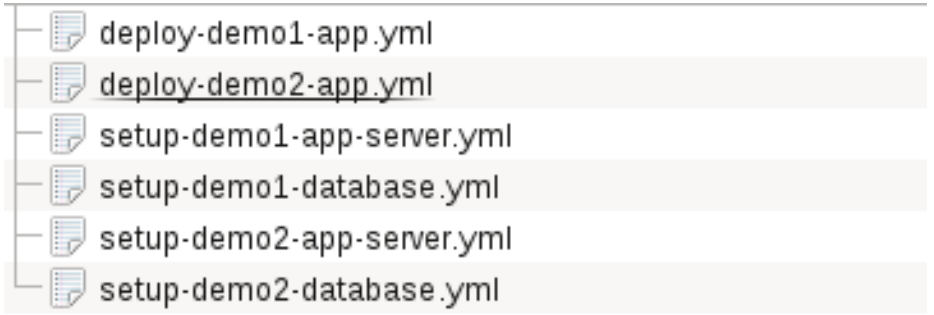
Deploy Application Playbook

Ad-hoc-Kommando

Warum Roles?


Database Server 3
Includes MySQL
Database Server 2
Includes MySQL
Database Server 1
Includes MySQL
Application Server 3
Application Server 2
Application Server 1
demo-app 3
demo-app 2
demo-app 1

Warum Roles?

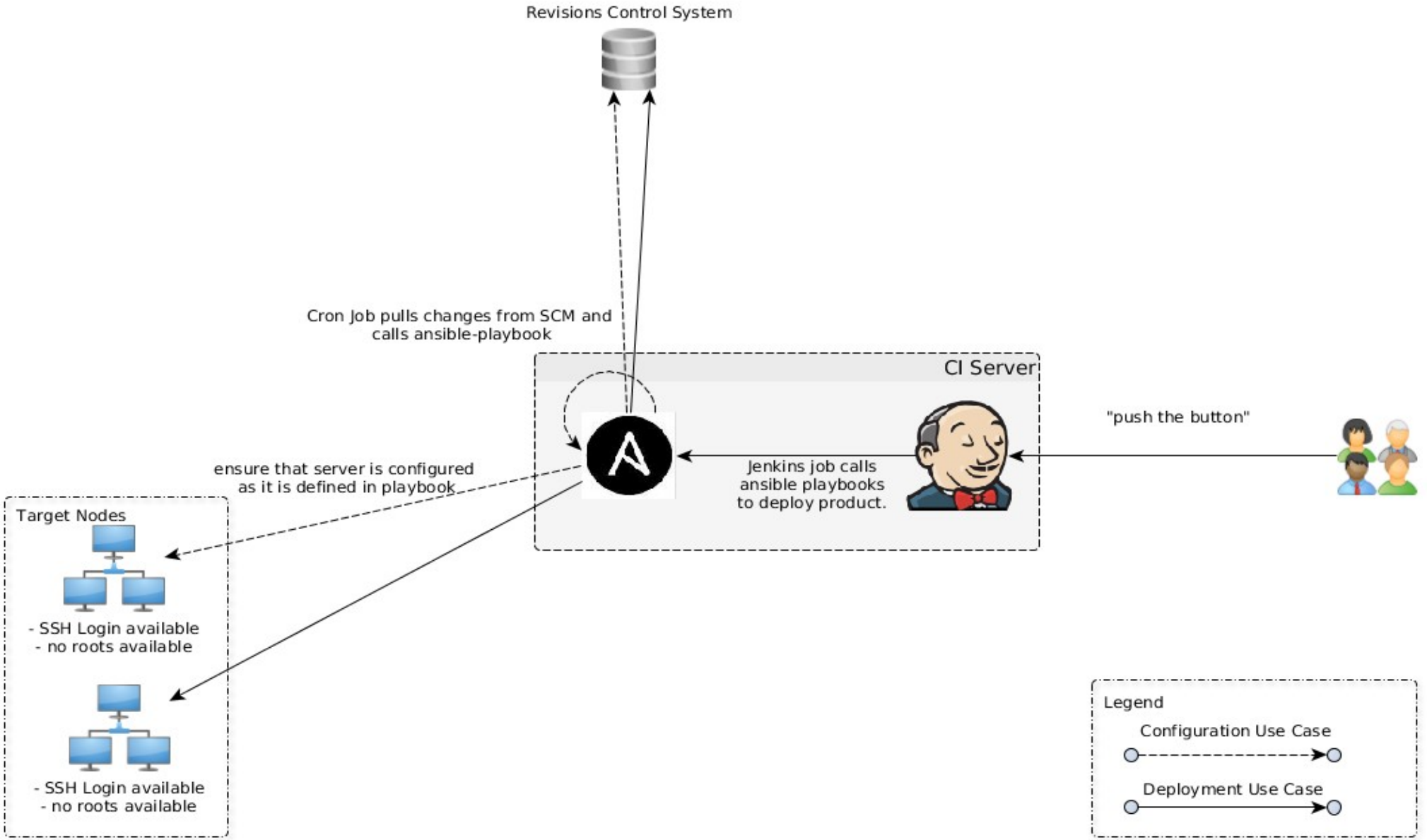

deploy-demo1-app.yml
deploy-demo2-app.yml
setup-demo1-app-server.yml
setup-demo1-database.yml
setup-demo2-app-server.yml
setup-demo2-database.yml

Warum Roles?

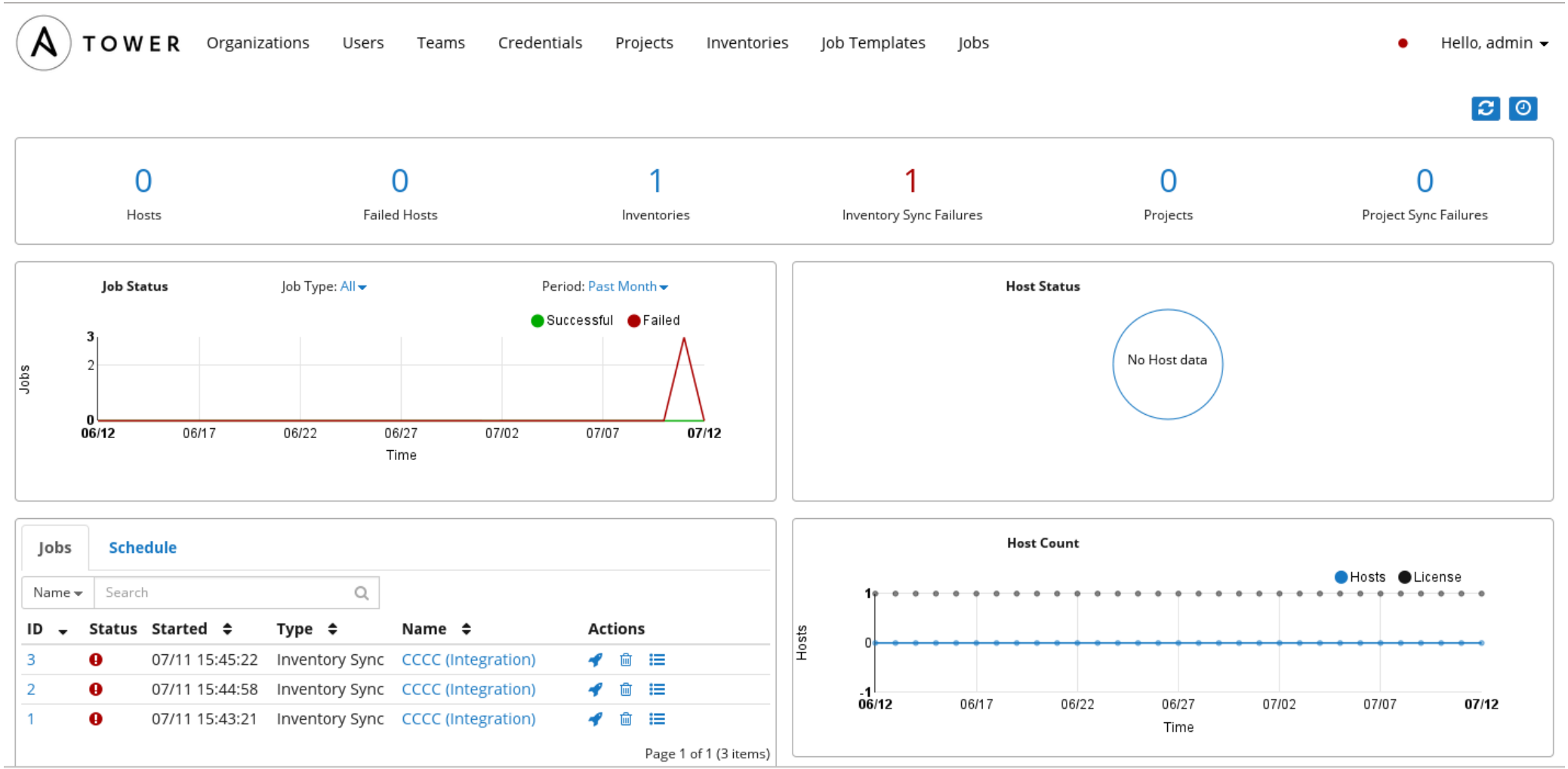
```
1  - hosts: demo1-application-server
2    roles:
3      - {role: deploy-on-tomcat, webapp_source_path: ./demo1-1.0-
       SNAPSHOT.war, webapp_target_name: demo1 }
4
```

```
1  - hosts: demo2-application-server
2    roles:
3      - {role: deploy-on-tomcat, webapp_source_path: ./demo2-1.0-
       SNAPSHOT.war, webapp_target_name: demo2 }
4
5
```

Ansible Infrastruktur



Ansible Tower



Wie werden Ansible Skripte getestet?

- `ansible-playbook --check`
- `ansible-playbook --syntax-check`
- Jenkins + Vagrant
- Rspec tests

ServerSpec Tests

```ruby
require 'spec_helper'

describe package('mysql-server') do
  it { should be_installed }
end

describe service('mysql') do
  it { should be_enabled   }
  it { should be_running   }
end

describe 'MySQL config parameters' do
  context mysql_config('bind-address') do
    its(:value) { should eq '0.0.0.0' }
  end
end

```

```ruby
require 'spec_helper'

describe package('openjdk-7-jdk') do
  it { should be_installed }
end

describe command('ls /etc/init.d/tomcat') do
  its(:exit_status) { should eq 0 }
end

describe command('ls /opt/tomcat') do
  its(:exit_status) { should eq 0 }
end

```

ServerSpec Tests

Wie unterscheidet sich Ansible zu seiner Konkurrenz?

Vergleich

Ansible

- Orchestrierung über SSH
- Benötigt keine Rootrechte auf Zielsystem
- Konfigurationsmgmt + Applikationsdeployment
- Monitoringtool nur in der Enterprise Variante
- Skripte mehr imperativ
- Windows-Support rudimentär
- Skripte OS- bzw. Distrobutions-spezifisch

Puppet

- Client-Server Architektur
- Für komfortables Arbeiten benötigt es Rootrechte
- Konfigurationsmgmt
- Monitoringtools Open Source
- Skripte mehr deklarativ
- Windows-Support
- Skripte können OS-unspezifisch sein

Vergleich

Puppet

```
class nodejs {

  class { 'apt':
  }

  exec { 'apt-get-update':
    command     => '/usr/bin/apt-get update',
  }

  package {'software-properties-common' :
    ensure=> installed,
    require => Exec['apt-get-update'],
  }

  apt::ppa {'ppa:chris-lea/node.js' :}

  package { 'nodejs' :
    ensure => installed,
    require => Apt::Ppa ['ppa:chris-lea/node.js'],

  }
```

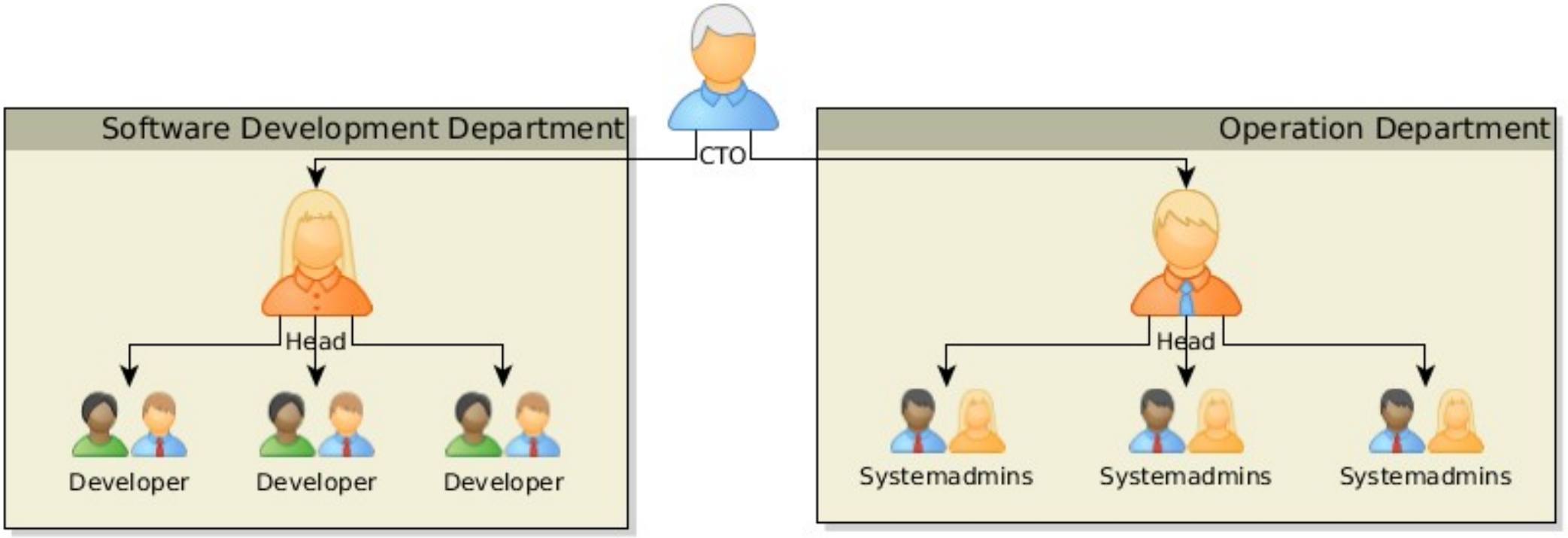
Ansible

```
- name: add nodejs ppa
  apt_repository: repo='ppa:chris-lea/node.js'

- name: install nodejs package
  apt: name=nodejs update-cache=yes
```

Weitere Einsatzszenarien aus Entwicklersicht

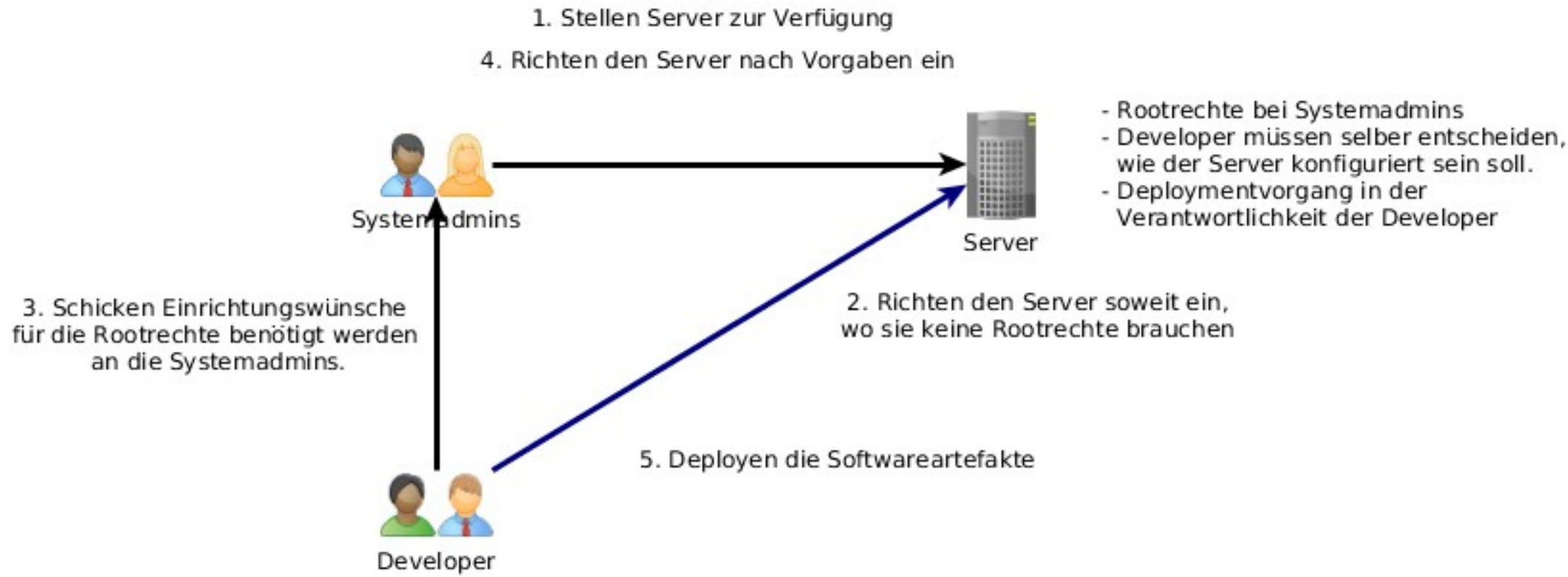
Systemkonfiguration für Entwickler

Organisatorische Ausgangslage
Realität



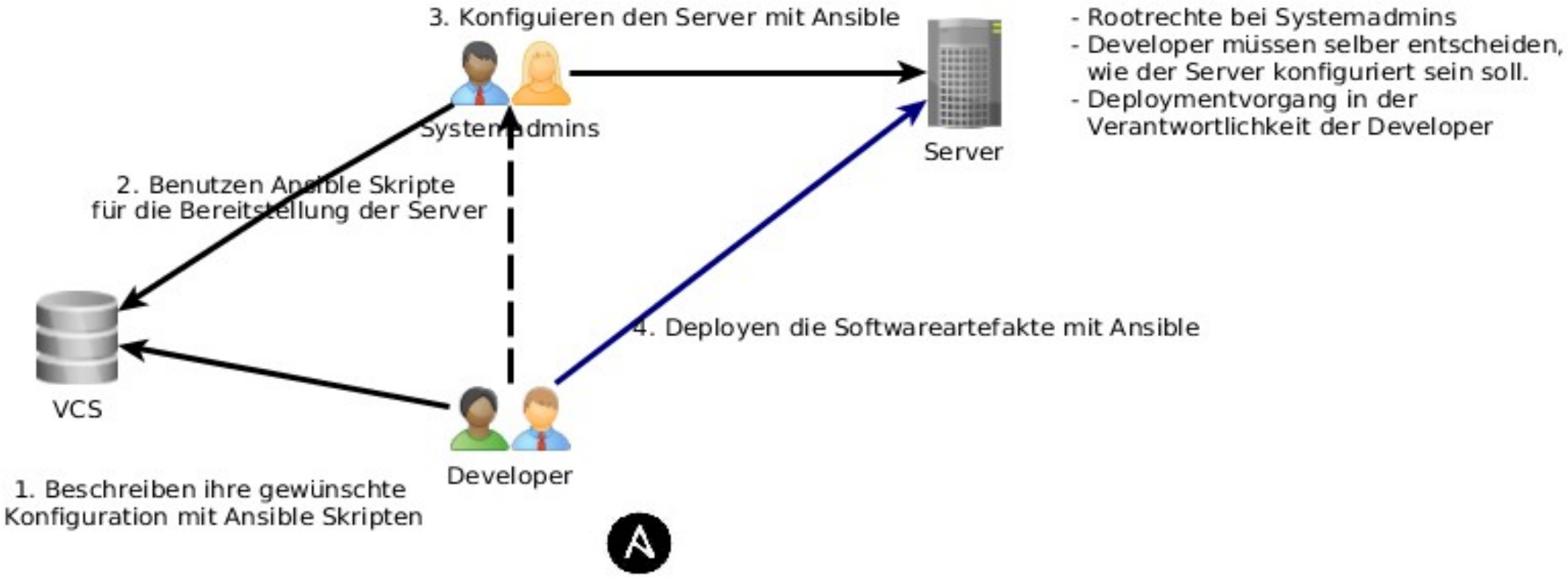
Systemkonfiguration für Entwickler

Prozess zwischen Development und Operation



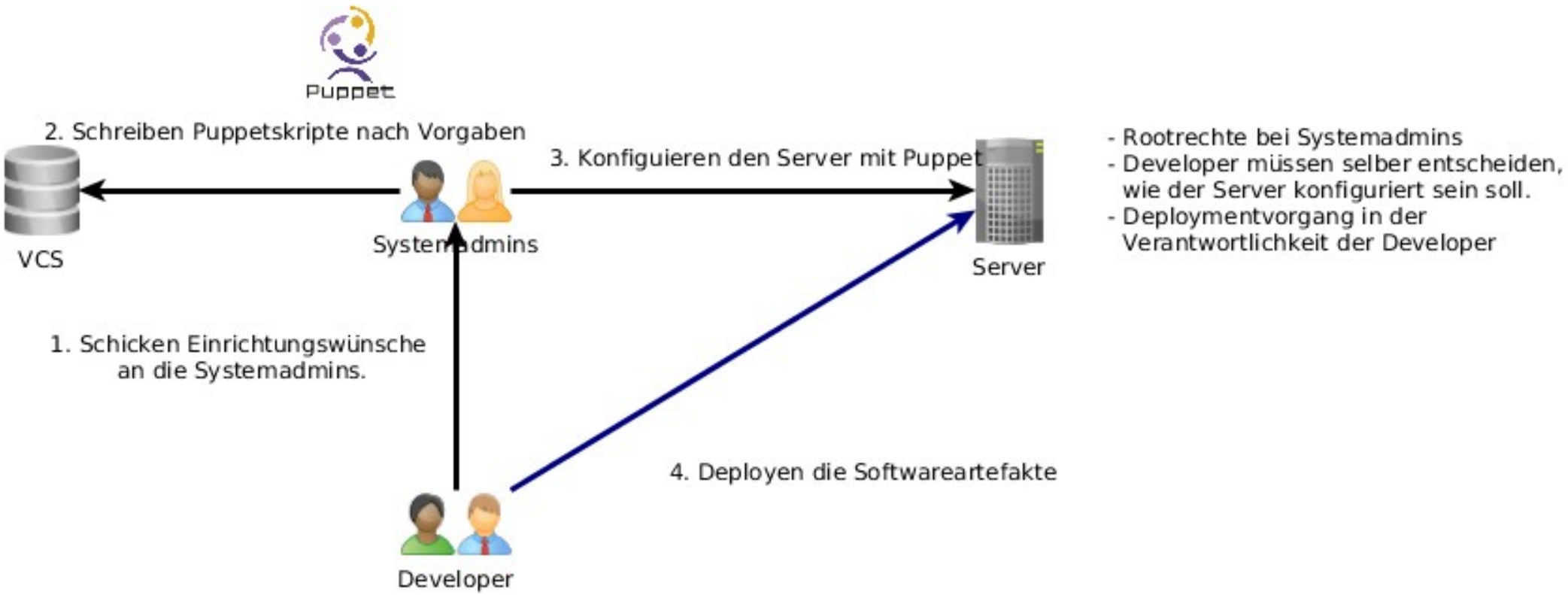
Systemkonfiguration für Entwickler

Lösungidee mit Ansible



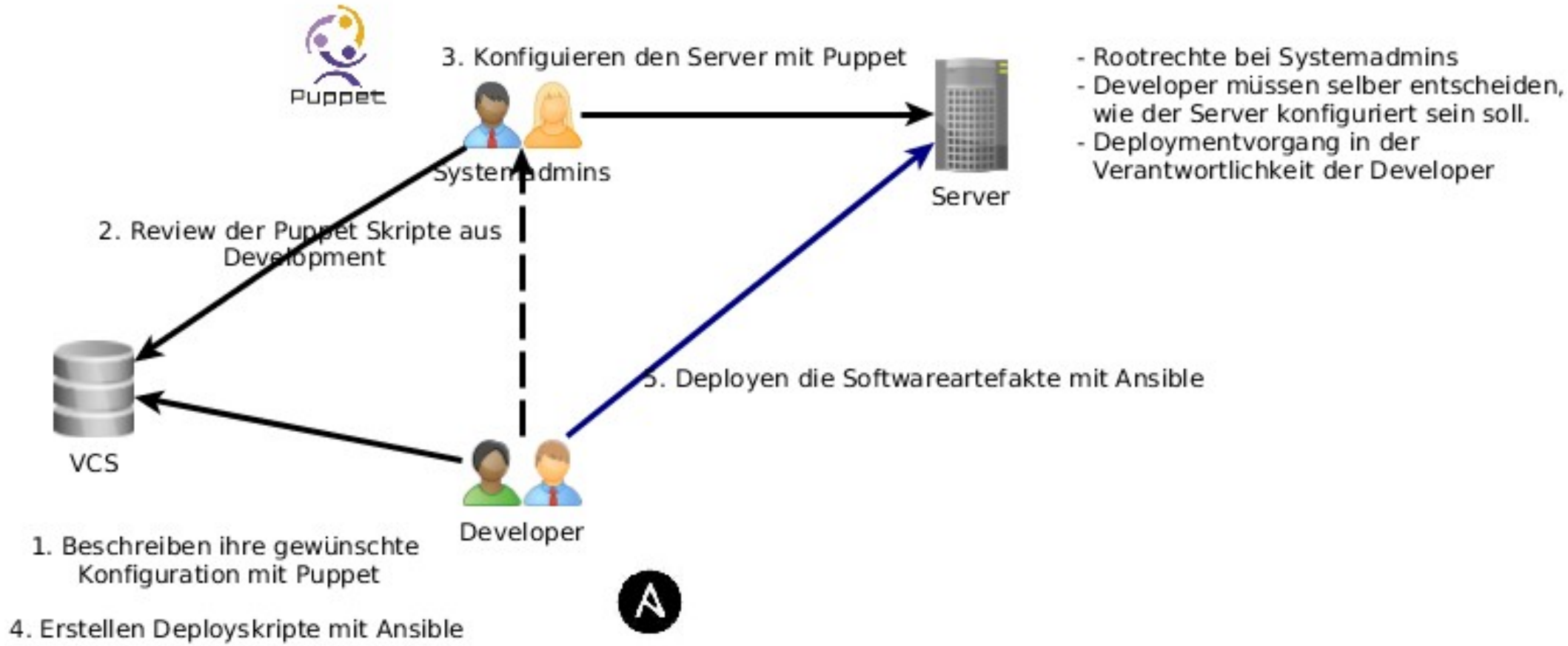
Systemkonfiguration für Entwickler

Variante - Prozess zwischen Development und Operation



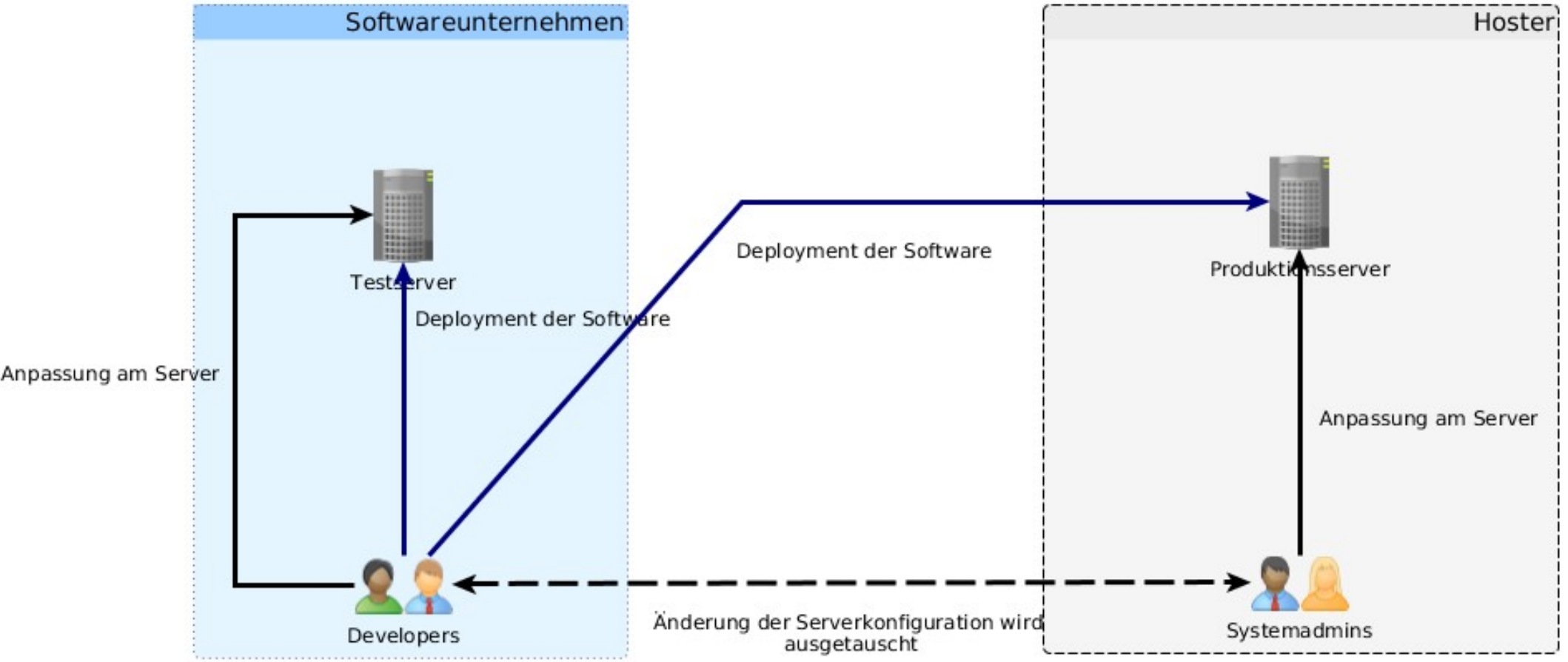
Systemkonfiguration für Entwickler

Lösungsvariante



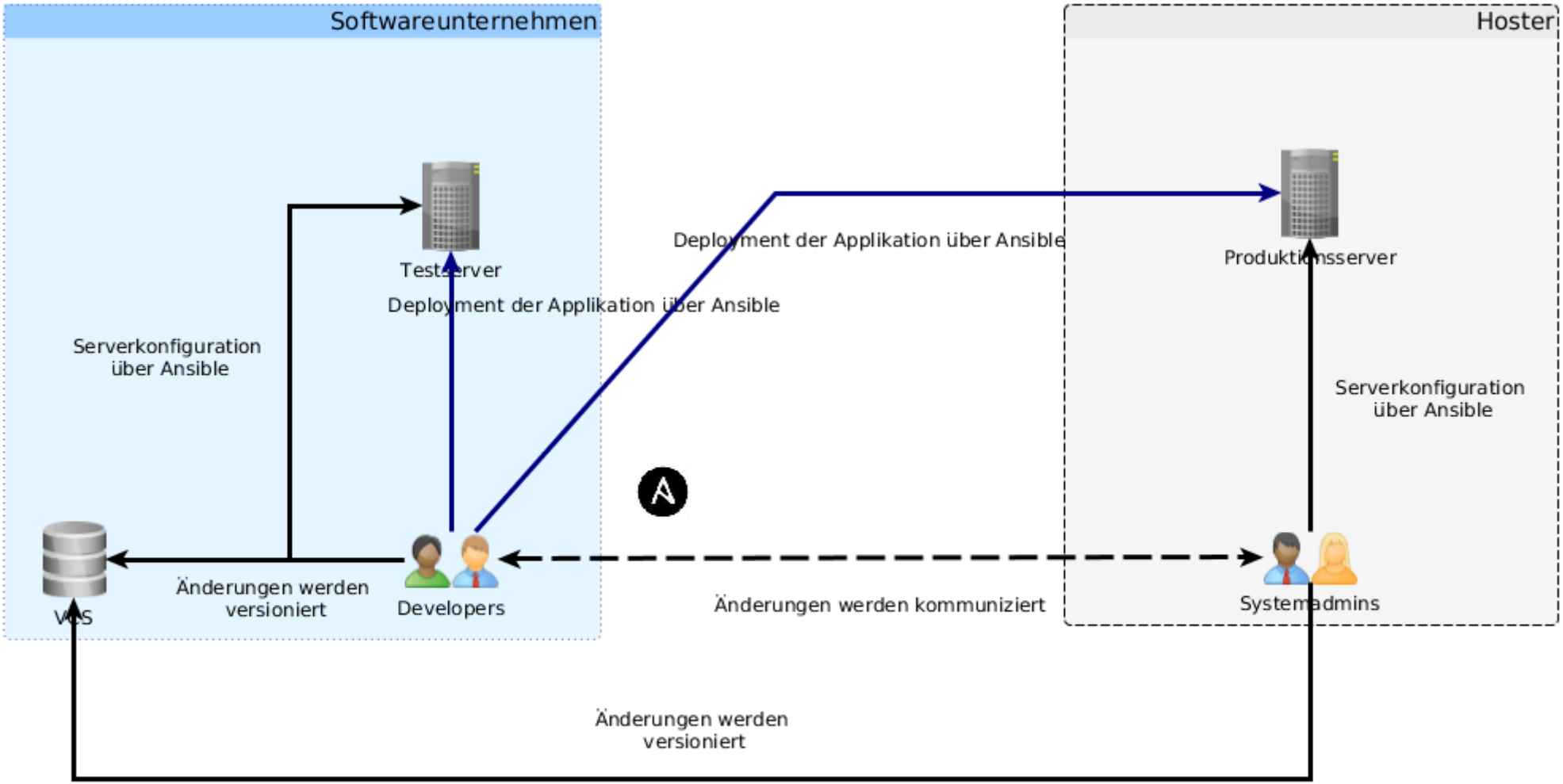
Systemkonfiguration für Entwickler

Produktionsserver sind beim externen Hoster



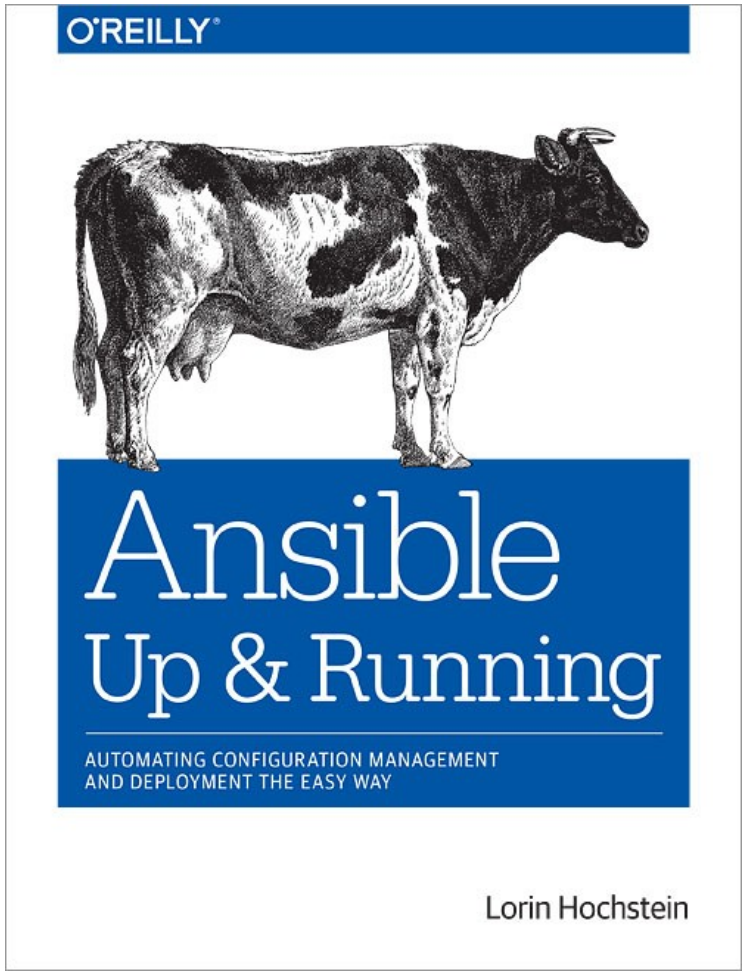
Systemkonfiguration für Entwickler

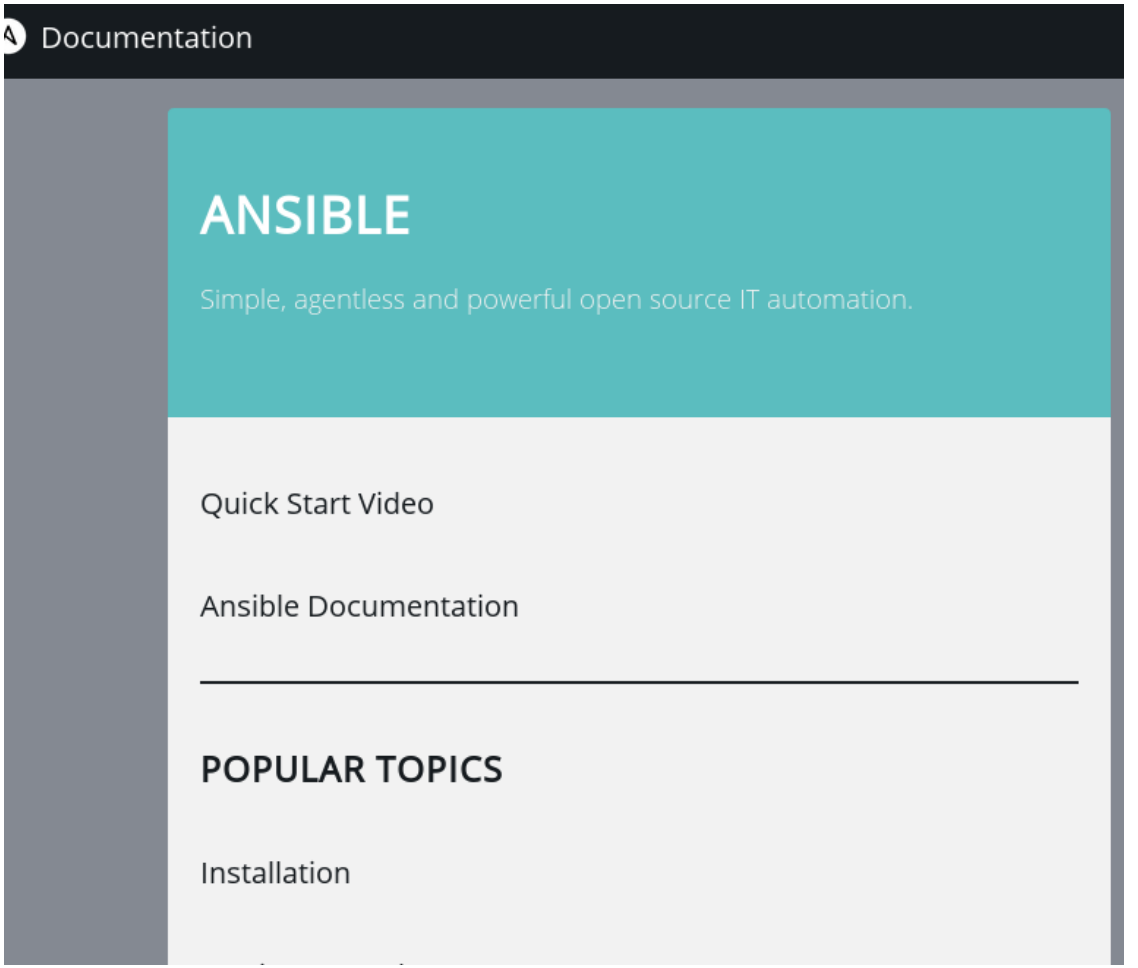
Lösungsidee



- Hoster verantwortlich für die Systemkonfiguration
- Softwareunternehmen verantwortlich für das Deployment
- Synchronisation zwischen Testserver und Produktionsserver wird vereinfacht

Weitere Informationen





http://docs.ansible.com/

Fragen?

@SandraParsick
info@sandra-parsick.de