

Next Level Ansible

FrOSCon 2018



Manuel Bonk

25. August 2018

- ▶ consultant at ATIX AG
- ▶ mainly Ansible, Saltstack, Kafka, Foreman
- ▶ training Ansible
- ▶ contributing to the Foreman Project (foreman-ansible-modules, nailgun)

- ▶ configuration management and orchestration tool
- ▶ masterless and daemonless
- ▶ idempotence
- ▶ minimal node requirements: sshd and python
- ▶ YAML
- ▶ jinja2 templating engine

- ▶ ansible-pull with cron module
- ▶ JUnit Callback Plugin
- ▶ omit
- ▶ tags behavior during import_tasks vs. include_tasks
- ▶ zip_longest
- ▶ merge dicts with default values
- ▶ dict rotation

- ▶ pull playbooks from a vcs repository and execute them
- ▶ executed as a cronjob...
- ▶ puppet-like behaviour in an ansible environment

- ▶ playbook output as a JUnit test case result file
- ▶ can be used in automated testing e.g. visualization in Jenkins

- ▶ module thinks parameter isn't set
- ▶ possible use case: testing task with different parameters (using `include_tasks`)

- ▶ tags are used to execute a subset of tasks
- ▶ `import_tasks` is rendered statically → tags of imported tasks are ignored
- ▶ `include_tasks` is rendered dynamically → tags of included tasks are checked

- ▶ zipping lists together
- ▶ use 'fillvalue' for lists of unequal lengths
- ▶ can be used to build cli parameter strings

- ▶ one dict with default settings
- ▶ only non-default values have to be written down
- ▶ use case: configuration files for microservices

from this...

```
ostable:
```

```
OS1:
```

```
  provisioning_templates:
```

```
    - PT1
```

```
    - PT2
```

```
OS2:
```

```
  provisioning_templates:
```

```
    - PT2
```

```
    - PT3
```

...into this

```
template_dict:
```

```
PT1:
```

```
  - OS1
```

```
PT2:
```

```
  - OS1
```

```
  - OS2
```

```
PT3:
```

```
  - OS2
```