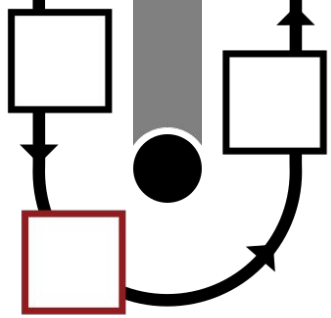
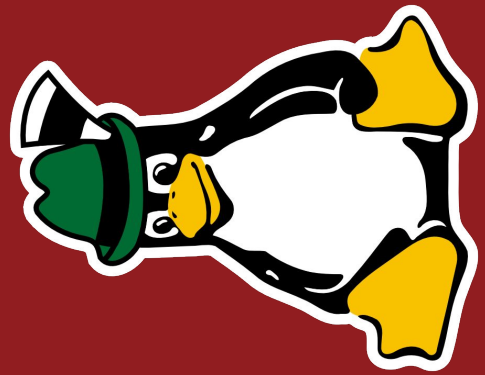


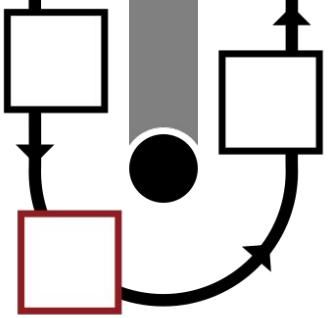
Paternoster

Admin-Scripts mit Ansible

https://github.com/uberspace/paternoster

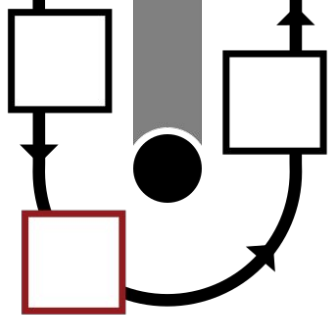


Shellscripts



```
$ cat account-create.sh
#!/bin/sh
USERNAME=$1  ## please don’t do this.

## add system account
/usr/sbin/useradd ${USERNAME};
## create log dir
mkdir /readonly/${USERNAME}
chown root:${USERNAME} /readonly/${USERNAME}
chmod 750 /readonly/${USERNAME}

echo Account erstellt. Alles gut.
```

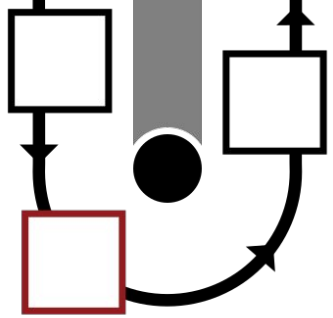
Ansible-Script



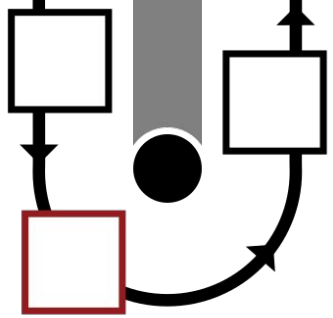
```
$ cat account-create.yml
#!/bin/env ansible-playbook
- name: create unix account and setup SSH keys
  hosts: localhost
  tasks:
    - user: name={{ username }}
    - file:
        dest: /readonly/{{ username }}
        owner: root
        group: "{{ username }}"
        mode: 0750
    - debug: msg="Account erstellt. Alles gut."
```

Aufruf

```
$ account-create.sh luto
Account erstellt. Alles gut.

$ account-create.yml --extra-vars username=luto
Account erstellt. Alles gut.
```

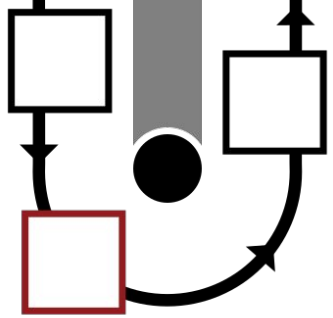


Aufruf



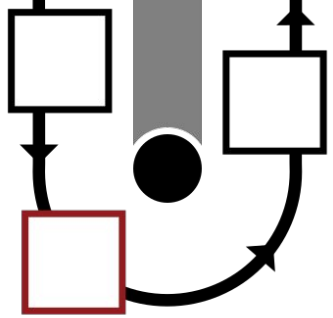
```
$ account-create.sh luto
Account erstellt. Alles gut.

$ account-create.yml --extra-vars username=luto
Account erstellt. Alles gut.

$ account-create.yml --username luto
Account erstellt. Alles gut.
```

Paternoster-Header



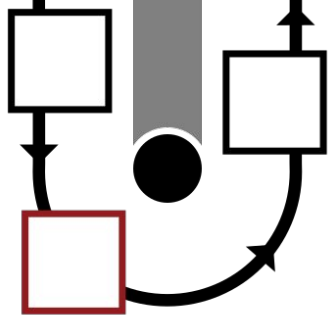
```
#!/usr/bin/env paternoster
- hosts: paternoster
  vars:
    parameters:
      - name: username
        short: u
        help: "name of the user to create"
        required: yes
        type: paternoster.types.restricted_str
        type_params:
          allowed_chars: 'a-z'
```

Paternoster-Script



```
(... header ...)

- name: create unix account and setup SSH keys
  hosts: localhost
  tasks:
    - user: name={{ username }}
    - file:
        dest: "/readonly/{{ username }}"
        state: directory
        group: "{{ username }}"
        mode: 0750
    - debug: msg="Account erstellt. Alles gut."
```

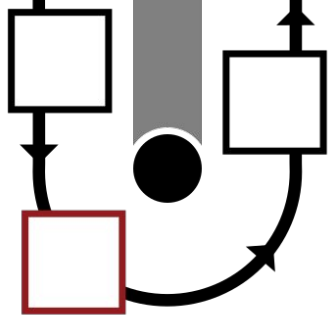
Paternoster-Script



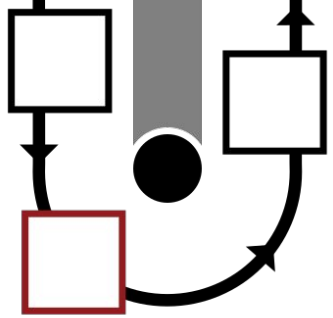
```
(... header ...)

- name: create unix account and setup SSH keys
  hosts: localhost
  tasks:
    - user: name={{ param_username }}
    - file:
        dest: "/readonly/{{ param_username }}"
        state: directory
        group: "{{ param_username }}"
        mode: 0750
    - debug: msg="Account erstellt. Alles gut."
```

Aufruf

```
$ account-create.yml --username luto
Account erstellt. Alles gut.
```

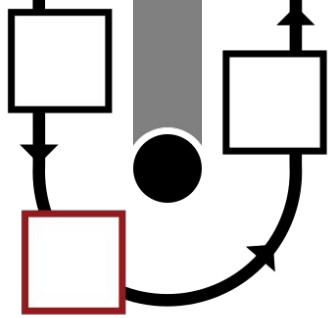


Aufruf - Hilfe



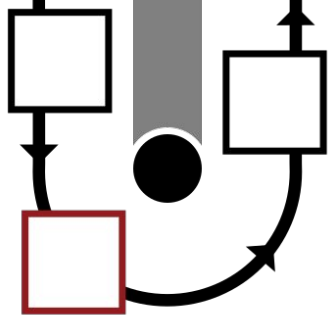
```
$ account-create.yml --help
usage: account-create.yml [-h] -u USERNAME [-v]

required arguments:
  -u USERNAME, --username USERNAME
                        name of the user to create

optional arguments:
  -h, --help            show this help message and exit
  -v, --verbose         run with a lot of debugging output
```

Aufruf - Debug



```
$ account-create.yml -u luto -vvv
(...)
PLAY [paternoster] ****************************************
skipping: no hosts matched

PLAY [create unix account and setup SSH keys] **************
TASK [Gathering Facts] ************************************
ok: [localhost]

TASK [user] ***********************************************
changed: [localhost] => {"changed": true, "comment": "",
"createhome": true, "group": 1005, "home": "/home/luto", "name":
"luto", "shell": "/bin/bash", "state": "present", "system": false,
"uid: 1001}
```

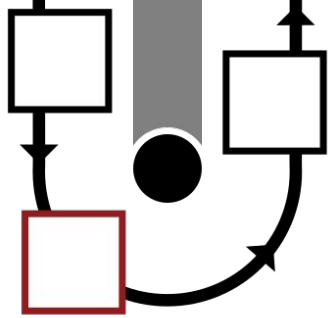
Parameter



```
#!/usr/bin/env paternoster
- hosts: paternoster
  vars:
    parameters:
      - name: domain
        (...)
      - name: mail
        short: m
        help: "use the given domain for the mailserver"
        action: store_true
```

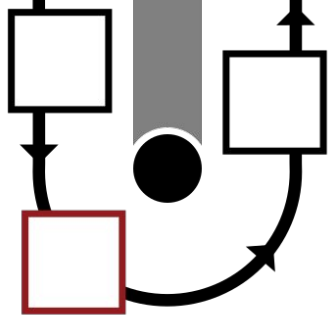
https://docs.python.org/2/library/argparse.html

Parametertypen



```
#!/usr/bin/env paternoster
- hosts: paternoster
  vars:
    parameters:
      - name: username
        short: u
        help: "name of the user to create"
        required: yes
        type: paternoster.types.restricted_str
        type_params:
          allowed_chars: 'a-z'
```

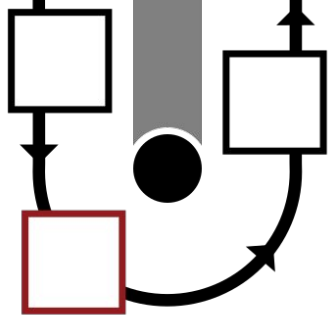
Parametertypen - String / Integer



```
type: paternoster.types.restricted_str
type_params:
  regex: "^[a-z][a-z0-9]+$"
  minlen: 5
  maxlen: 30

type: paternoster.types.restricted_int
type_params:
  min: 0
  max: 100
```

Parametertypen - Domain

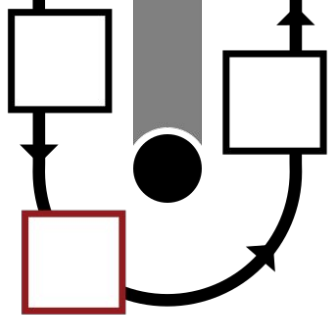
```
type: paternoster.types.domain
type_params:
  wildcard: true
```



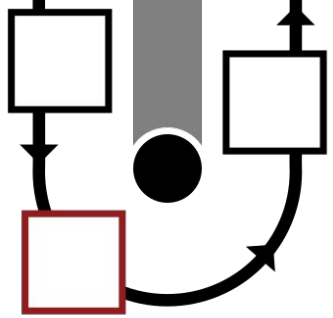
- Validierung von Länge, Format, Anzahl der Komponenten,...
- Validierung der TLD
- Optional "*.google.com" auch möglich

Privilege Escalation

```
#!/usr/bin/env paternoster
- hosts: paternoster
  vars:
    become_user: userfacts
    parameters:
      - name: domain
        short: d
        help: "your new domain"
        required: yes
        type: paternoster.types.domain
```



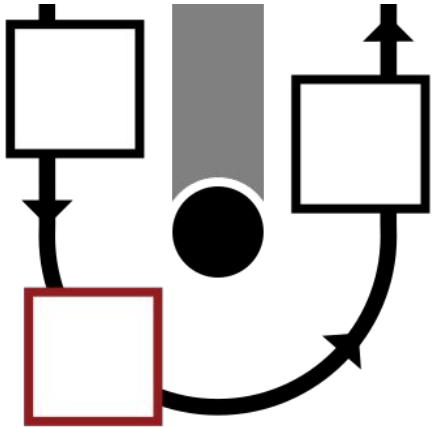
Parameter Dependencies



```
#!/usr/bin/env paternoster
- hosts: paternoster
  vars:
    parameters:
      - name: mail
        action: store_true
      - name: mailbox
        depends_on: mail
        type: paternoster.types.restricted_str
        type_params:
          allowed_chars: "a-z"
```

Danke! - Fragen?

https://github.com/uberspace/paternoster



Kontakt:

luto
@luutoo auf Twitter
m@luto.at

bei uberspace.de :)