

Jinja2

- Simple file templating with loops
- Simple file templating with if/else
- Even use variables for file names!
- Iterate through items, globs, and hashes

```
- name: Install nodejs and python-pip Debian family
  apt: pkg={{ item }} state=present
  when: ansible_os_family == "Debian"
  with_items:
    - nodejs
    - python-pip
  tags:
    - packages
```

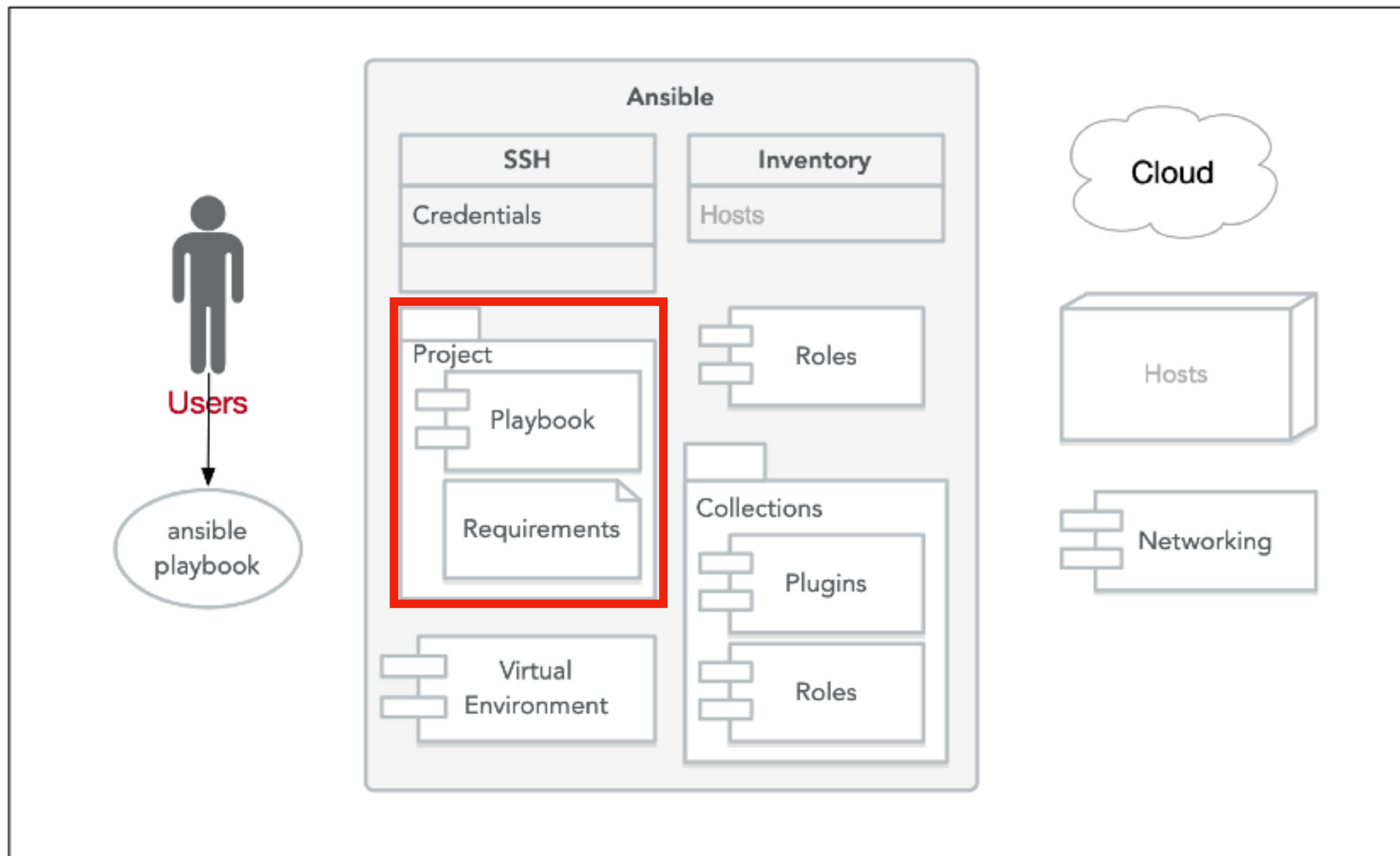
```
# Include on this server, the variables about the server we are targeting.
- include_vars: "../../private_vars/{{ inventory_hostname }}.yaml"

# Include cause we push out the authkeys for cm server here.
- include_vars: "../../private_vars/{{ private_vars_dir }}/cm.yaml"
```

```
# Custom settings
myhostname = {{ my.server.name }}
myorigin = wpengine.com
{% if cluster.use_mailgun is defined %}
relayhost = [{{ creds.mailgun.relay_address }}]:{{ creds.mailgun.relay_port }}
{% else %}
relayhost = [{{ creds.postfix.relay_address }}]:{{ creds.postfix.relay_port }}
{% endif %}
```

```
options timeout:1
options attempts:1
{% for dns_host in dns_servers %}
nameserver {{ dns_host }}
{% endfor %}
```

How to work with **Projects**?



A **Project** can be used to organize content

A Project lists its dependencies as Requirements

Specific behaviour is de-coupled in **Roles** or **Collections**

A Project is stored in a Git repository