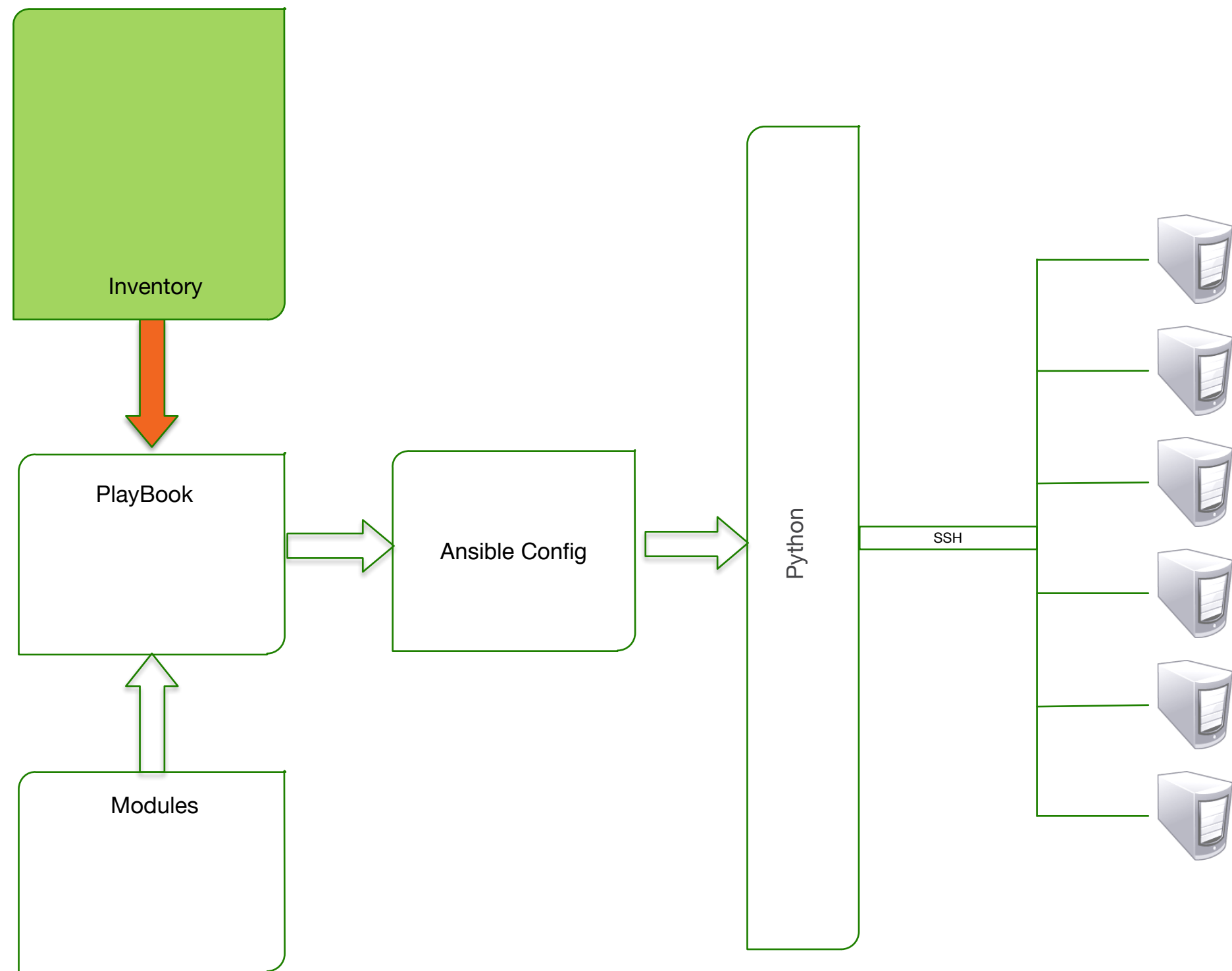


Ansible Inventory and Configuration



Aaron Paxson

<http://myteneo.net> | @Neelixx | +Aaron Paxson



Inventory Features

Behavioral
Parameters

Groups

Groups of Groups

Assign Variables

Scaling out using
multiple files

Static/Dynamic

Inventory File

[db]

db1.company.com

db2.company.com

Inventory File

```
[ db ]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com
```

Inventory File

```
[db]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com ansible_python_interpreter=/usr/bin/python
```

Inventory File

```
[ db ]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com ansible_python_interpreter=/usr/bin/python
```

```
[ datacenter-west:children ]
```

```
db
```

Inventory File

```
[ db ]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com ansible_python_interpreter=/usr/bin/python
```

```
[ datacenter-west ]
```

```
db
```


Inventory File

```
[ db ]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com ansible_python_interpreter=/usr/bin/python
```

```
[ datacenter-west:children ]
```

```
db
```

Inventory File

```
[db]
```

```
db1.company.com ansible_ssh_user=aaron ansible_ssh_pass=123
```

```
db2.company.com ansible_python_interpreter=/usr/bin/python
```

```
[datacenter-west:children]
```

```
db
```

```
[datacenter-west:vars]
```

```
ansible_ssh_user=ansible_user
```

```
ansible_ssh_pass=#45e!@Gh
```

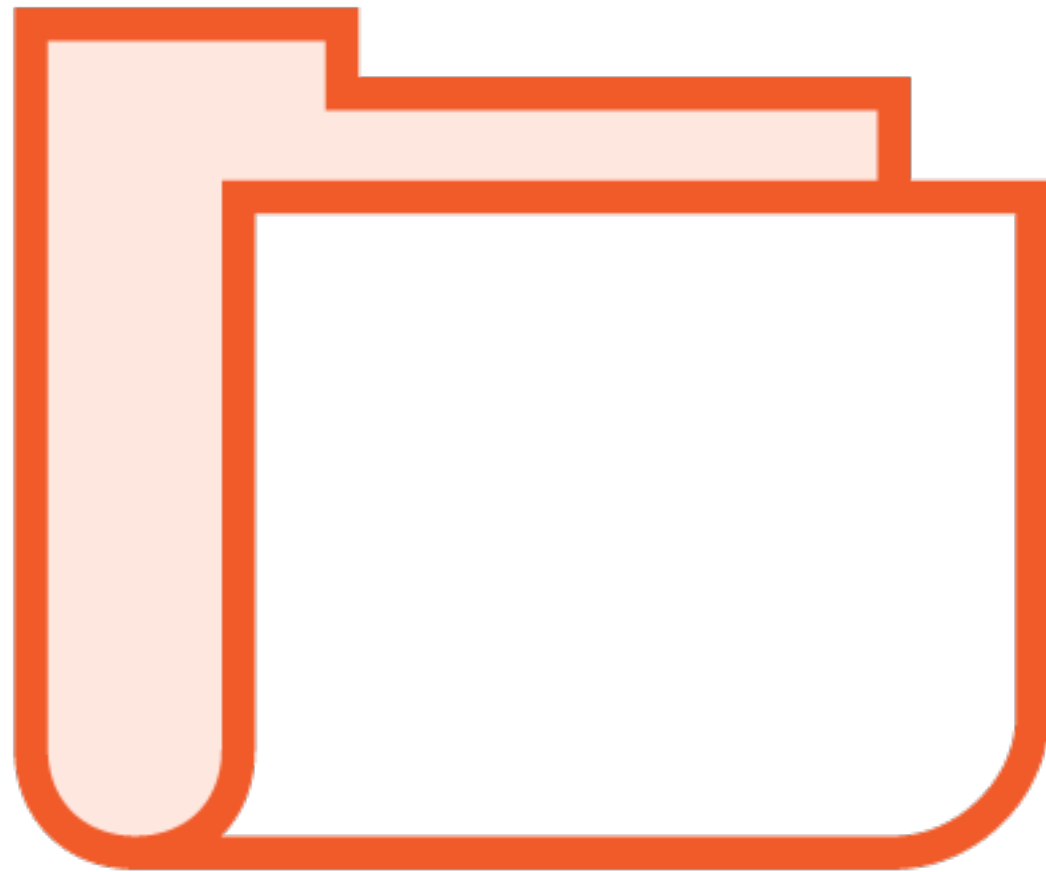
```
ntp-server=5.6.7.8
```

Creating our Inventory File

- Add Behavioral Parameters
- Create host-based variables
- Create a Group
- Create group-based variables



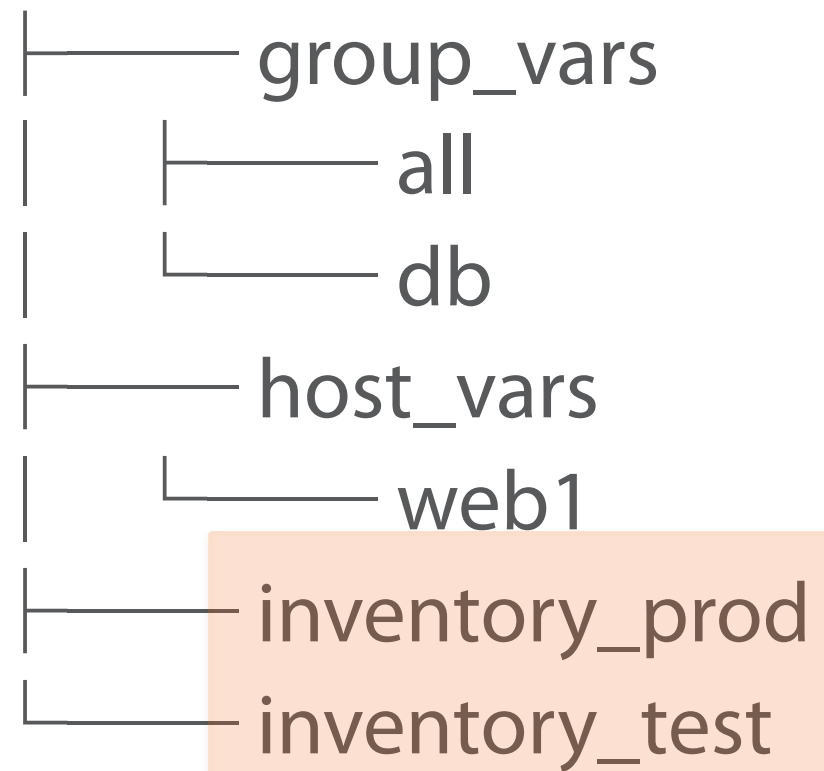
Scaling-out Inventory Files



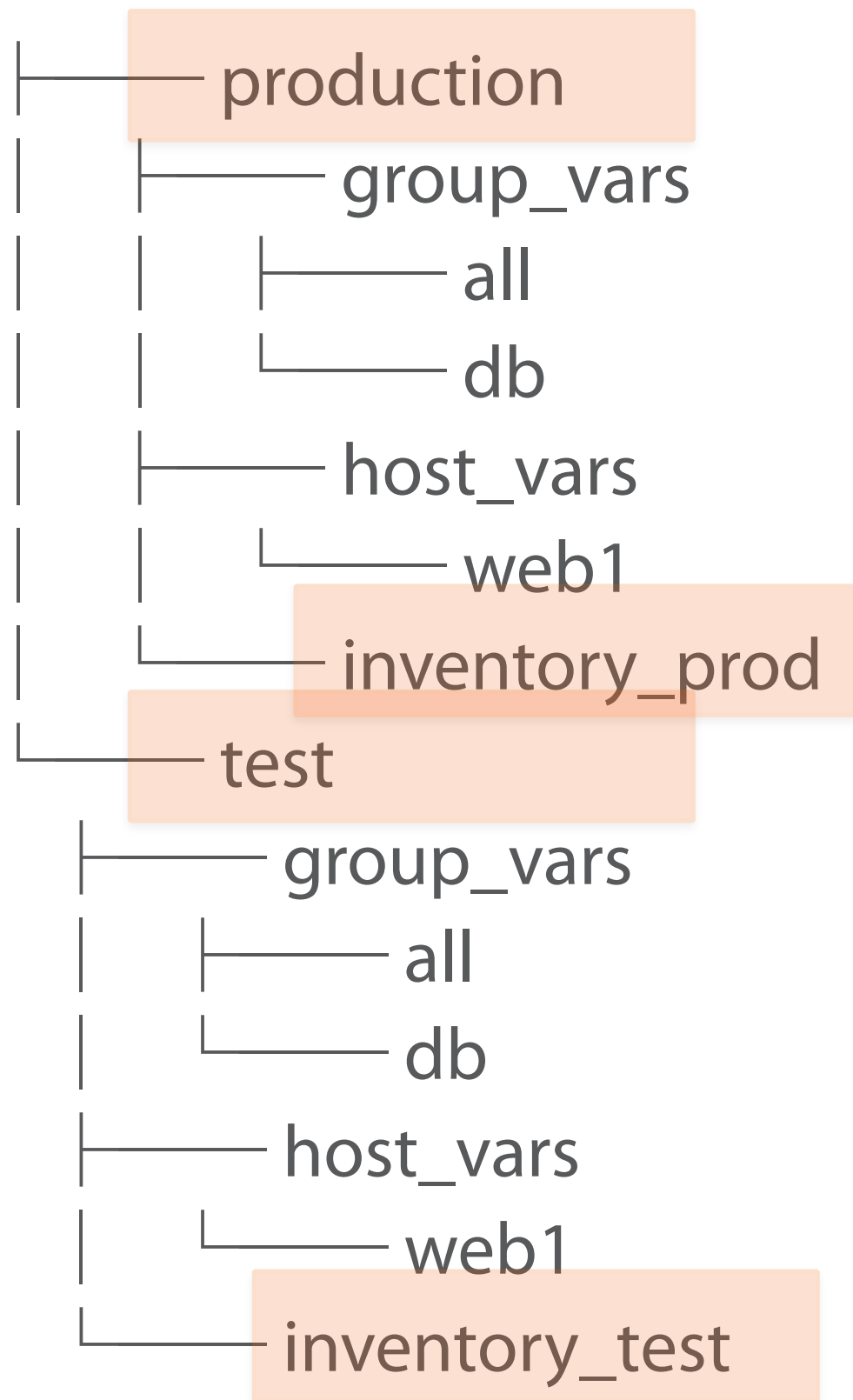
Using Directories

Can use to break-out long-running inventory files.

Very useful when dealing with large environments.



Basic Directory Structure



Multi-Environment Directory Structure

Order-of-Operations (Precedence)

- 1). (Group_Vars) All
- 2). (Group_Vars) GroupName
- 3). (Host_Vars) HostName

Variable File Example

file: group_vars/dc1-west

ntp: ntp-west.company.com

syslog: logger-west.company.com

Variable files are written in YAML

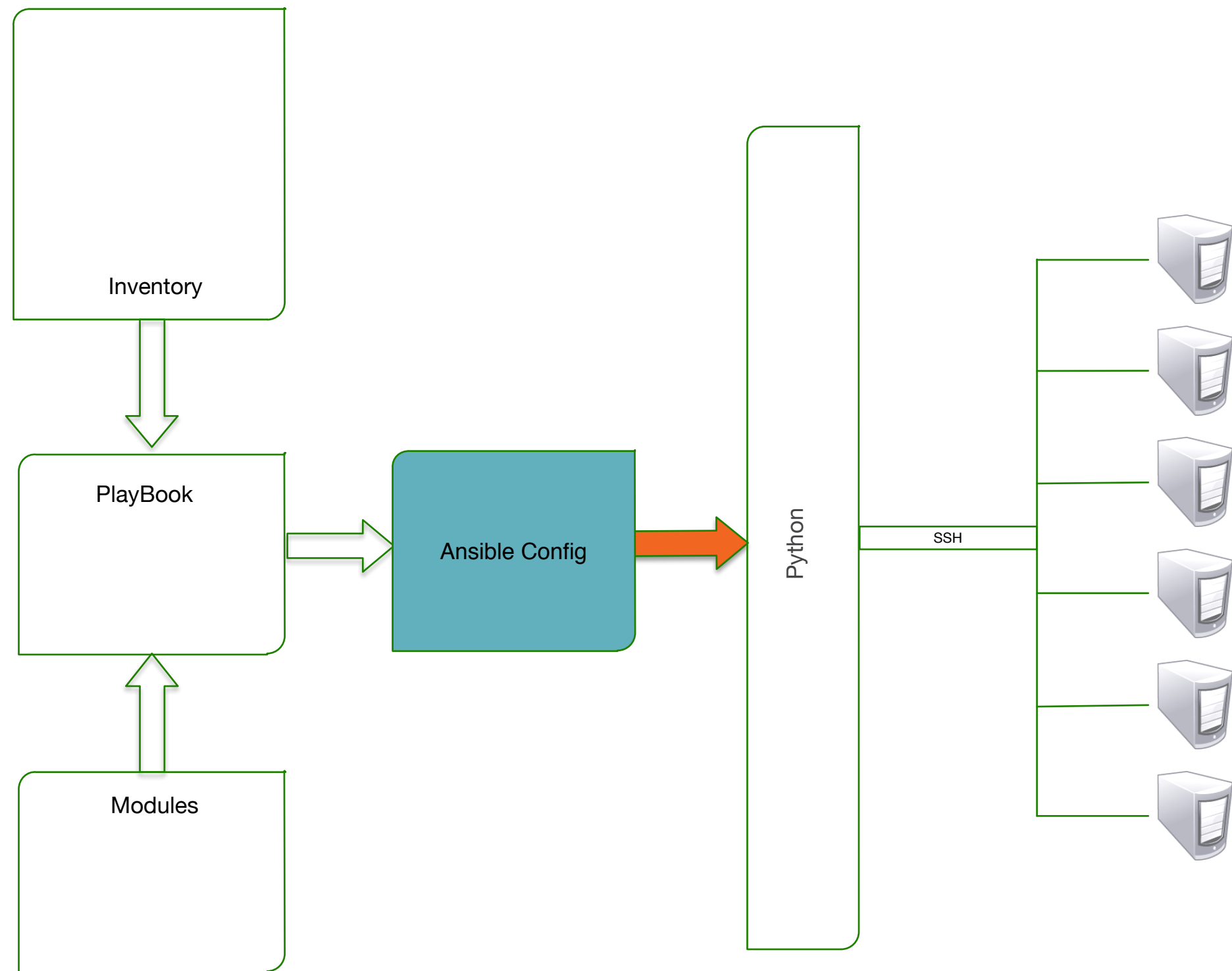
Comments use #

Key : Value pairs

Scaling Variable Files

- Create Group Variables in separate file
- Show Order-of-Precedence





Configuration Settings Order-of-Operations

1

`$ANSIBLE_CONFIG`

2

`./ansible.cfg`

3

`~/.ansible.cfg`

4

`/etc/ansible/ansible.cfg`

Configuration files are not merged.
First one wins!

Environment Variable Overrides

`$ANSIBLE_<configsetting>`

Override specific settings by
prefixing `ANSIBLE_` to the
name

`# export ANSIBLE_FORKS=10`

Great way to override specific
settings on-the fly!

[defaults] forks

Default set to 5

Total number of parallel operations Ansible executes

Production Recommendation: 20

Start at 20, and go up or down depending on performance

[defaults]
host_key_checking

Default set to True

For Production environments, do not change

Development Environment: set to False

Due to the dynamic environment of Dev, keeps it easy

[defaults]
log_path

Default set to Null

Write information on Ansible executions

Set path to log file

Make sure all users of Ansible has permissions to write

Editing Configuration

- Define settings in configuration file
- Override setting in environment variable

