



puppet



ANSIBLE

BY

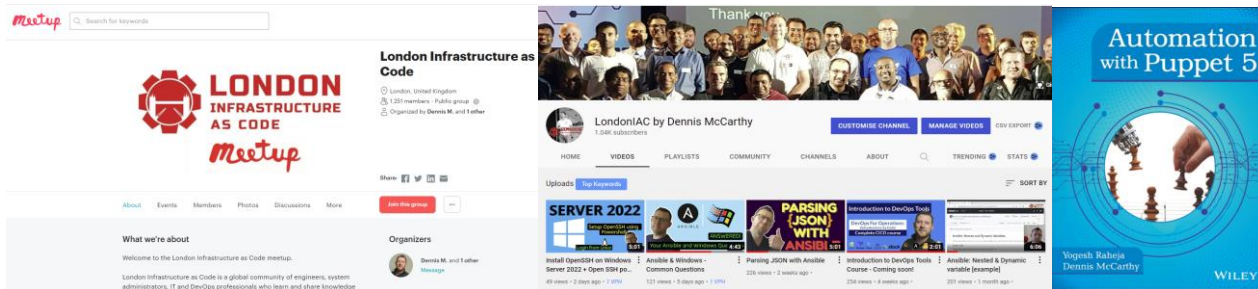
DENNIS MCCARTHY

APRIL 2022

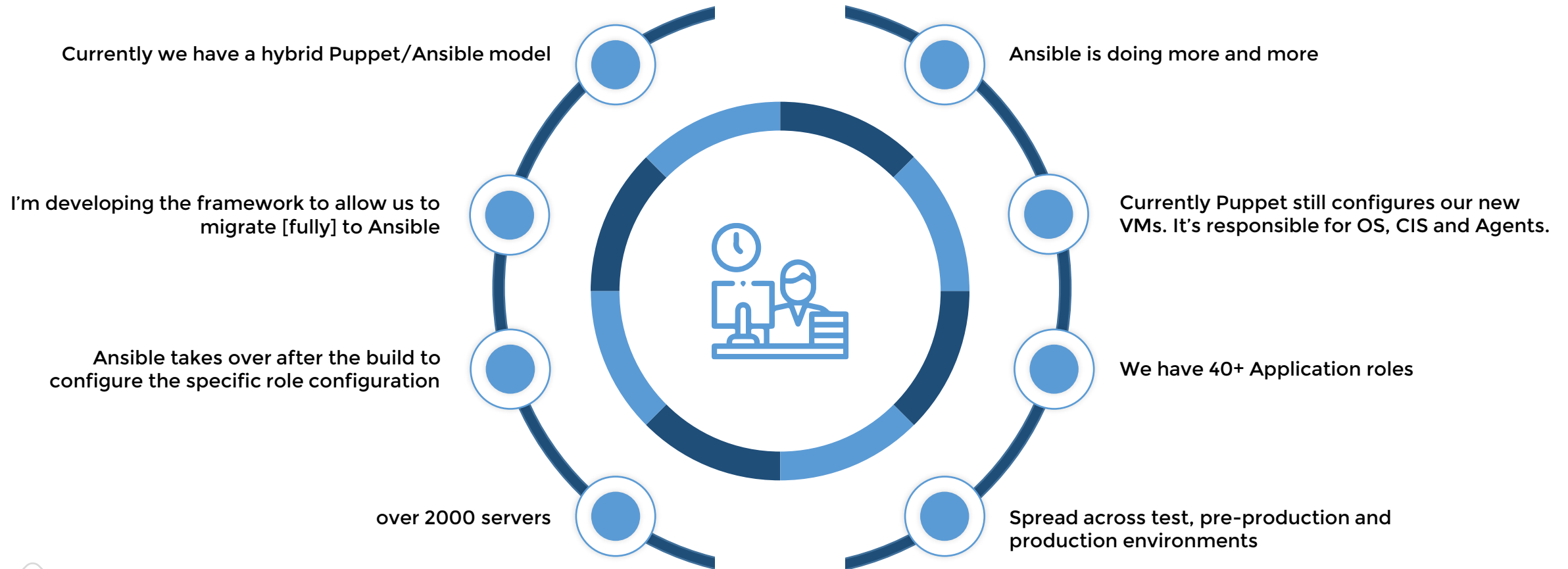


ABOUT ME

- I love automation!
- I've been working with Ansible for 5 years.
- I founded the LondonIAC meetup and its YouTube Channel
- Used to work for Red Hat.
- Published author... (wiley) "Automation with puppet 5".
- I now work at WorldPay.
- My roles is 80% writing ansible (updating or writing new).
- I like to keep it simple.



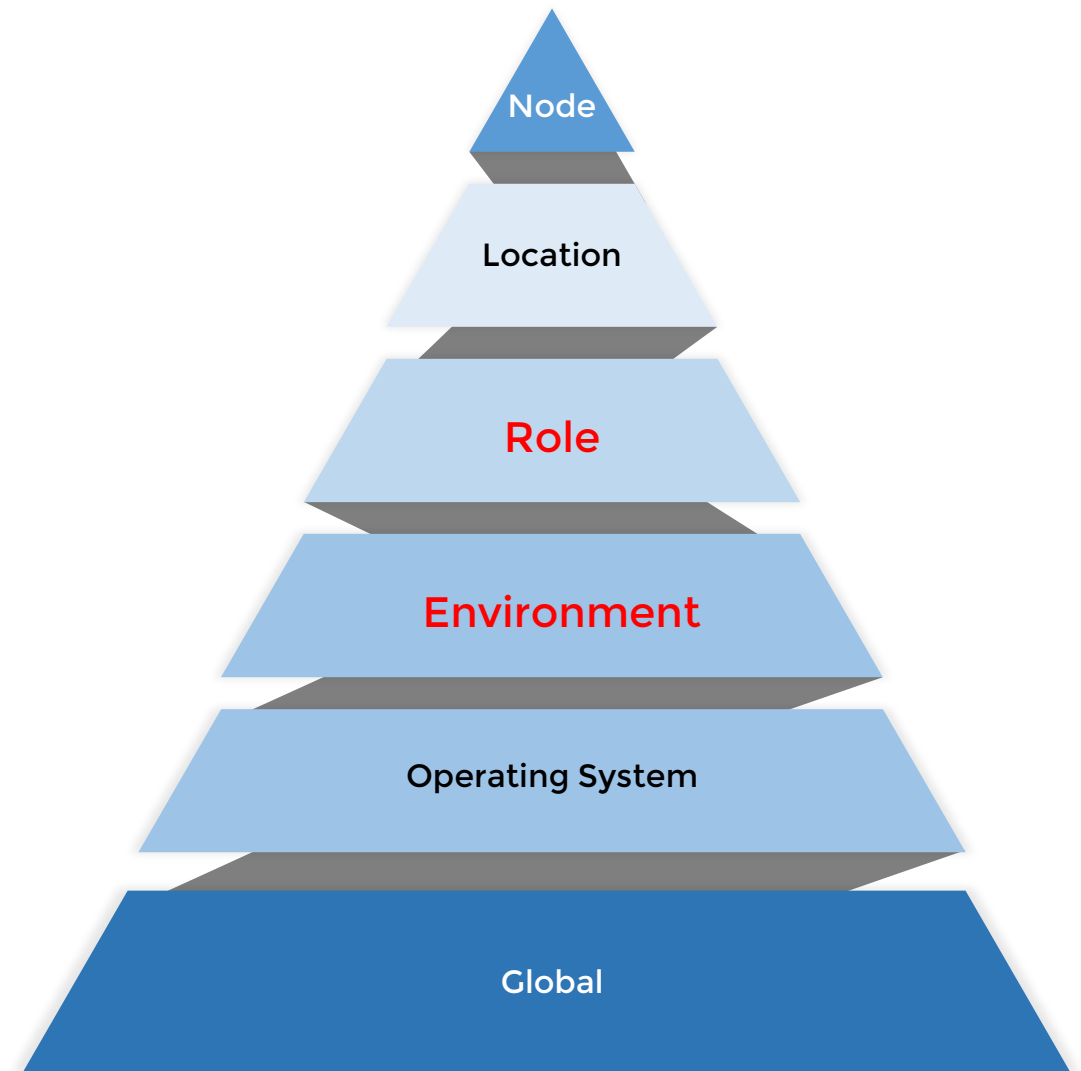
MY ENVIRONMENT



PUPPET AND HIERA

WHAT IS HEIRA?

Hierarchy is a way to structure an organization **using different levels of authority and a vertical link, or chain of command, between superior and subordinate levels of the organization.** Higher levels control lower levels of the hierarchy. You can think of an organizational hierarchy as a pyramid.



CONVERTING TO ANSIBLE



Looking at the Puppet hiera structure, I needed a way to replicate the levels of hierarchy

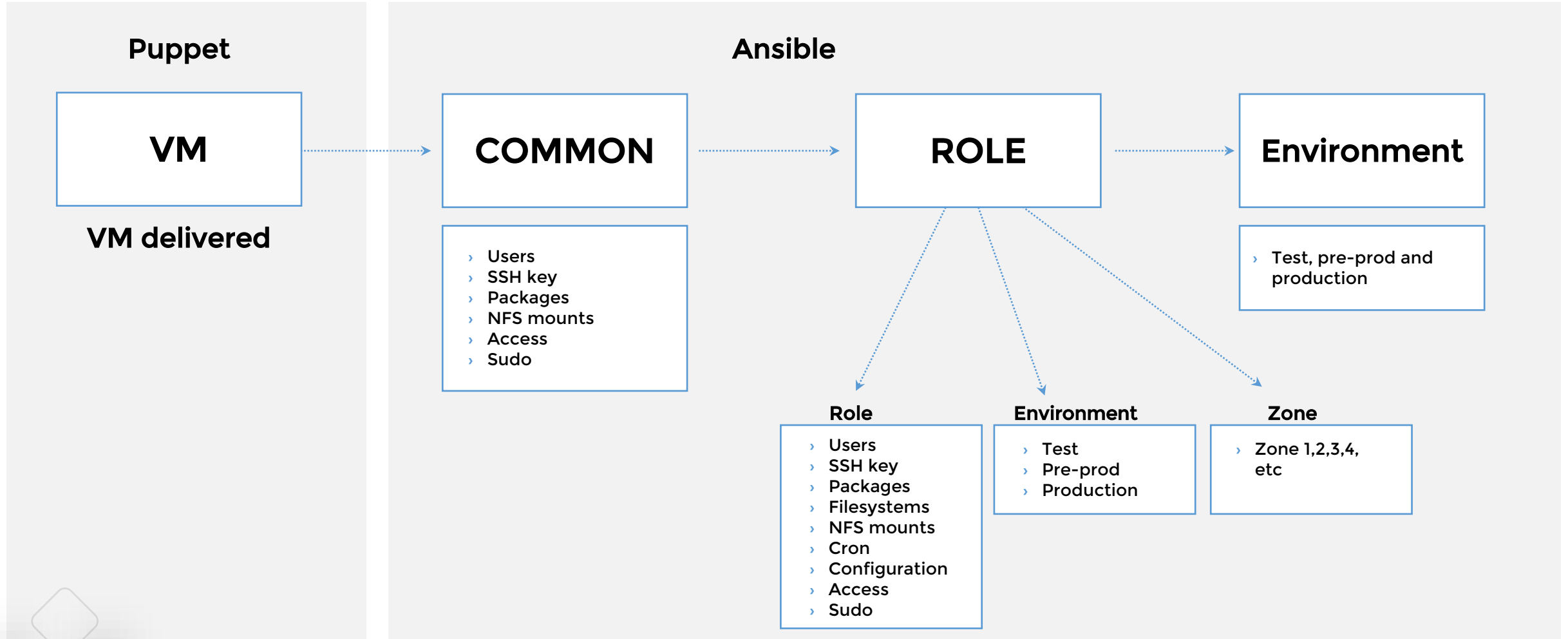


The three main factors for me were Role, Environment and zone.

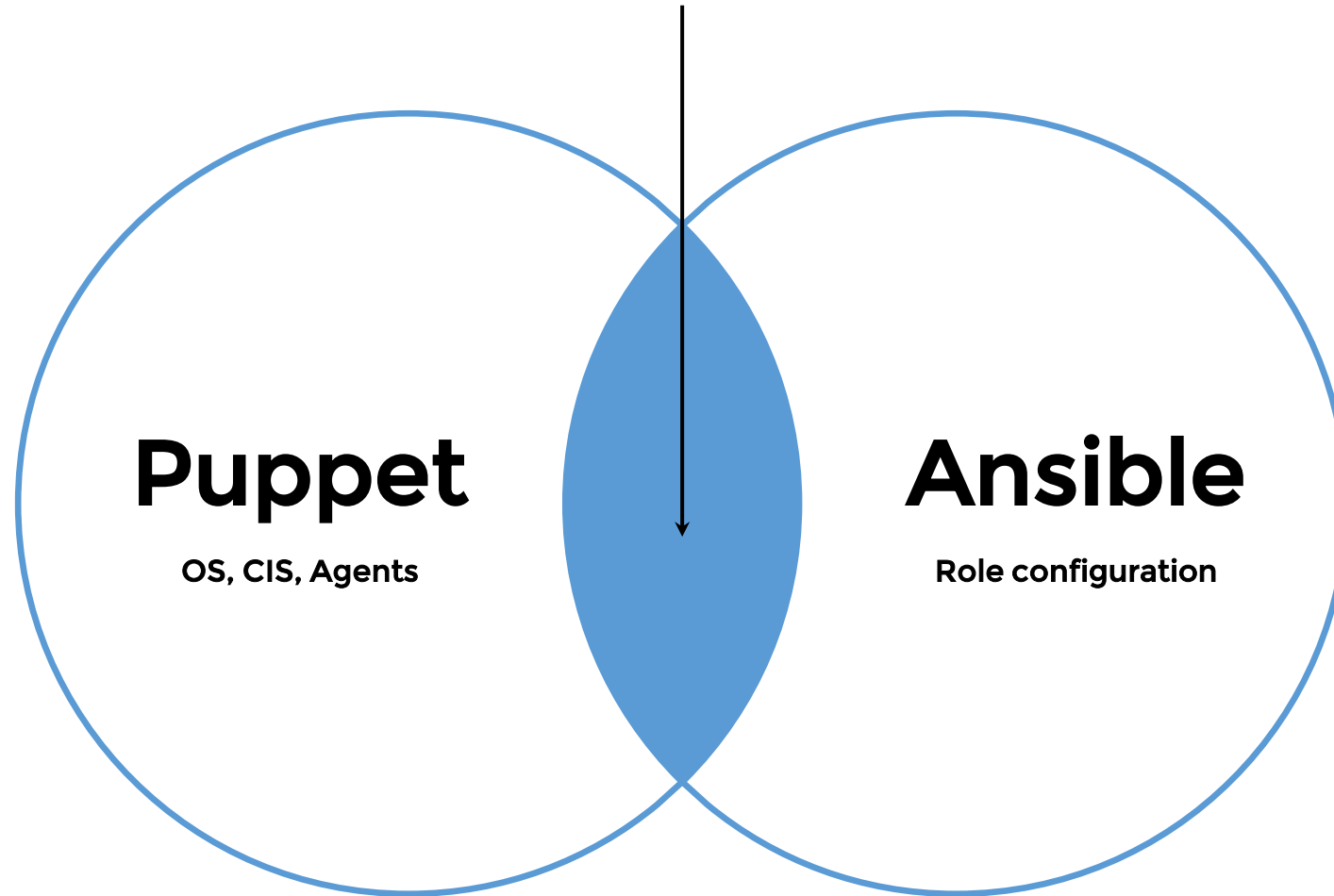


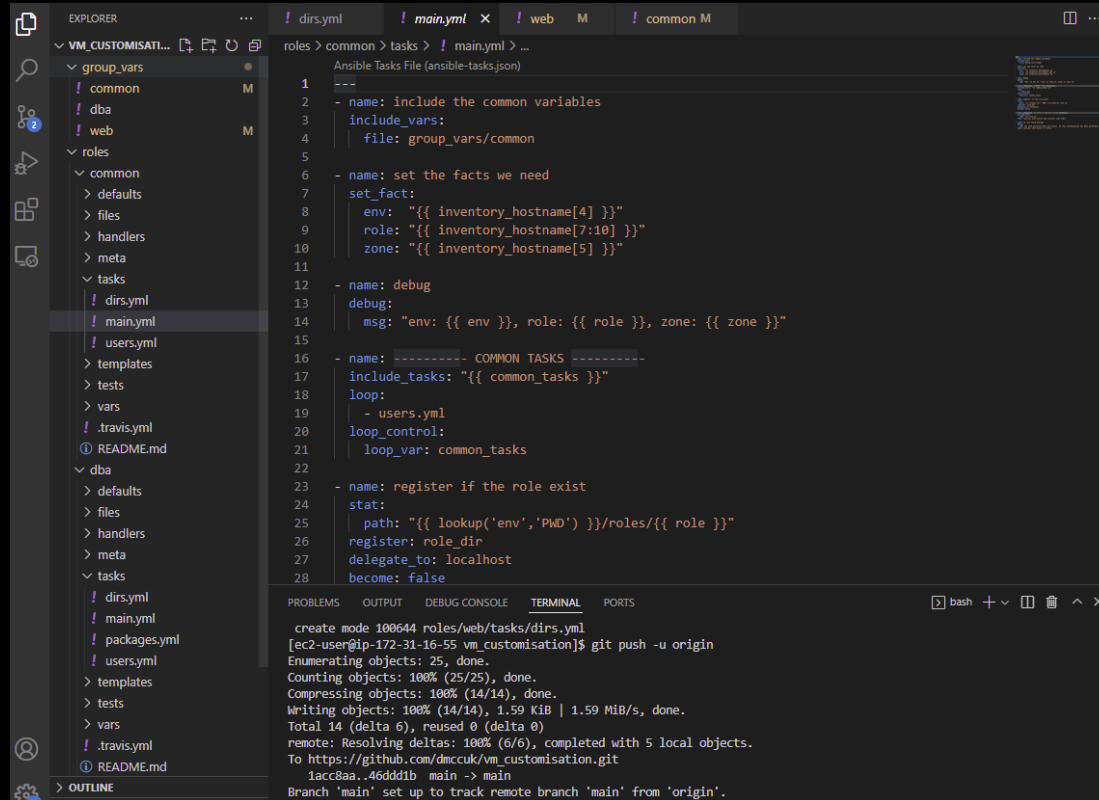
So this is the workflow I came up with

THE WORKFLOW



CROSSOVER FOR ACCESS AND SOME CIS RULES





The image shows a laptop screen with an IDE. The left sidebar shows a file explorer with a tree structure: `VM_CUSTOMISATI...`, `group_vars` (containing `common`, `dba`, `web`), `roles` (containing `common`, `defaults`, `files`, `handlers`, `meta`, `tasks`), `templates`, `tests`, `vars`, `.travis.yml`, and `README.md`. The main editor shows the `roles > common > tasks > ! main.yml` file. The code is an Ansible tasks file with the following content:

```
1 ---
2 - name: include the common variables
3   include_vars:
4     file: group_vars/common
5
6 - name: set the facts we need
7   set_fact:
8     env: "{{ inventory_hostname[4] }}"
9     role: "{{ inventory_hostname[7:10] }}"
10    zone: "{{ inventory_hostname[5] }}"
11
12 - name: debug
13   debug:
14     msg: "env: {{ env }}, role: {{ role }}, zone: {{ zone }}"
15
16 - name: ----- COMMON TASKS -----
17   include_tasks: "{{ common_tasks }}"
18   loop:
19     - users.yml
20   loop_control:
21     loop_var: common_tasks
22
23 - name: register if the role exist
24   stat:
25     path: "{{ lookup('env','PWD') }}/roles/{{ role }}"
26   register: role_dir
27   delegate_to: localhost
28   become: false
```

The bottom terminal window shows the output of a git push command:

```
create mode 100644 roles/web/tasks/dirs.yml
[ec2-user@ip-172-31-16-55 vm_customisation]$ git push -u origin
Enumerating objects: 25, done.
Counting objects: 100% (25/25), done.
Compressing objects: 100% (14/14), done.
Writing objects: 100% (14/14), 1.59 KiB | 1.59 MiB/s, done.
Total 14 (delta 6), reused 0 (delta 0)
remote: Resolving deltas: 100% (6/6), completed with 5 local objects.
To https://github.com/dmccuk/vm_customisation.git
 1acc8aa..46ddd1b main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

CODE OVERVIEW

https://github.com/dmccuk/vm_customisation

DEMO

```
ansible]# ansible-playbook check_disk_space.yml
```

```
*****
```

```
y Facts] *****
```

```
2.239]
```

```
3.165]
```

```
/boot usage on Hosts] *****
```

```
32.202.239]
```

```
32.115.165]
```

```
*****
```

```
3.165] => {
```

```
"": "Filesystem      Type  Size  Used Avail Use% Mounted on\n/dev/vda1          ext4   20G   2.
```

```
2.239] => {
```

```
"": "Filesystem      Type  Size  Used Avail Use% Mounted on\n/dev/vda1          ext4   20G   4.
```

```
*****
```

```
: ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0
```

```
: ok=3    changed=1    unreachable=0    failed=0    skipped=0    rescued=0
```