

Writing Ansible Playbooks



JP Toto

INFRASTRUCTURE DEVELOPER

@jptoto <http://jptoto.jp>



Ansible Is Powerful



`ansible webservers -m win_ping`



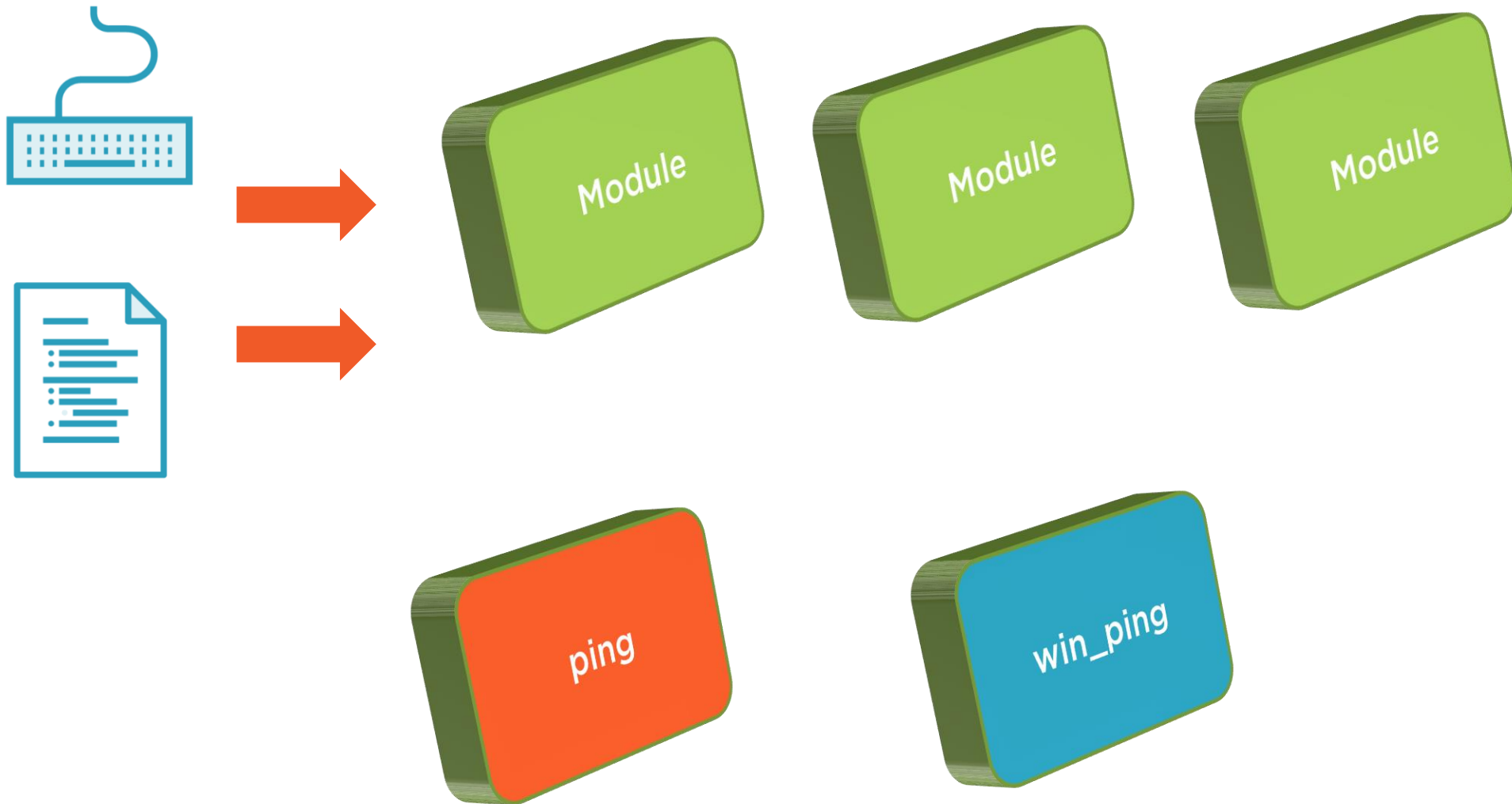
`/etc/ansible/hosts`



That's all helpful and necessary, but not to other teams



Modules



Ansible Comes with Many Modules



Copy files



Install software



Create files from templates



Operating system updates



Many many more...



Modules Are Idempotent



Running same Ansible commands...



“Desired State”



All the Modules are Listed



Windows modules start with win_



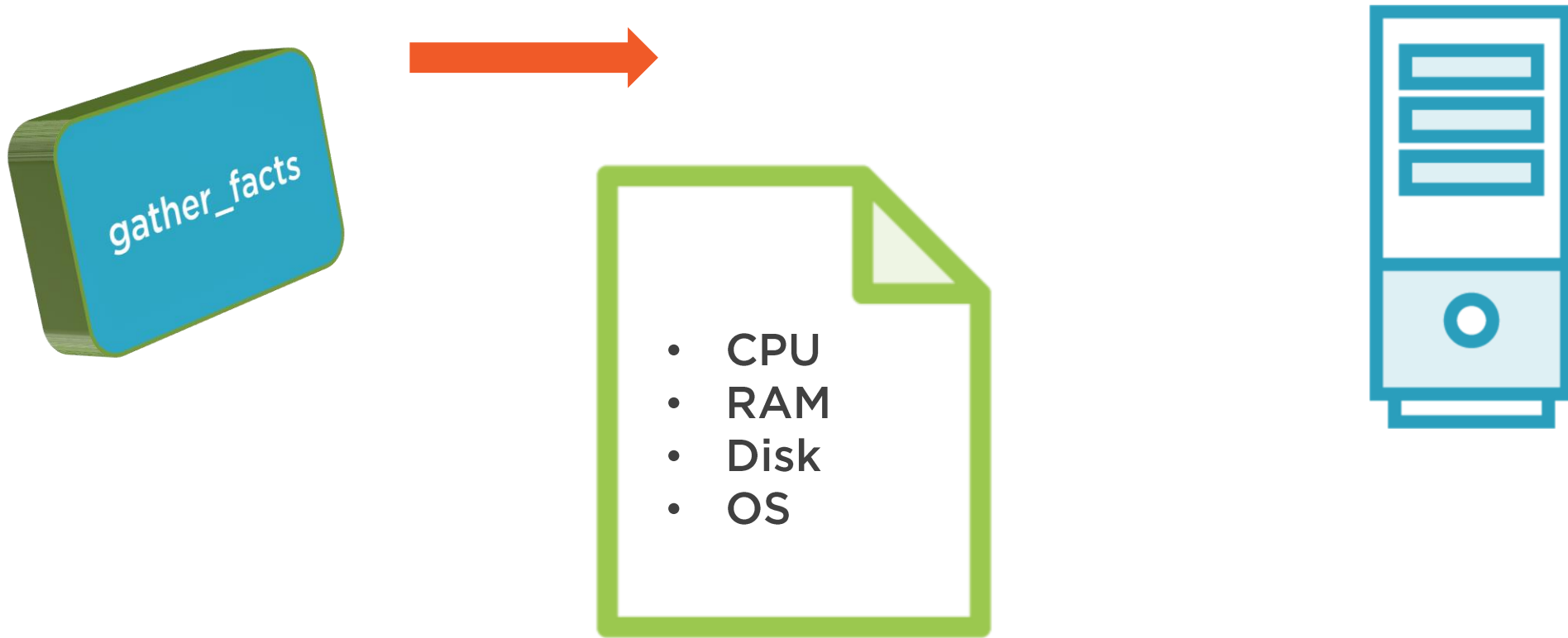
Linux modules have no prefix



You can perform just about any operation you can imagine with the included modules



The gather_facts Module



Demo



Let's check out the `gather_facts` module in action

Information gleaned from `gather_facts` can be used in future actions



Let's Talk About Playbooks



Playbooks are groups of tasks that usually call modules



Playbooks are written in YAML



May contain variables, lists of hosts, groups, and tasks



Playbooks Are Often Stored in Source Control



The Play

```
---  
- hosts: linux_servers  
  vars:  
    foo: "bar"  
  tasks:  
  
- name: Copy a file  
  copy:  
    src: "/tmp/config.json"  
    dest: "/etc/config.json"
```

This is one play

Hosts

playbook.yml

```
---  
- hosts: linux_servers ←  
  vars:  
    foo: "bar" ←  
  tasks:  
  
  - name: Copy a file  
    copy:  
      src: "/tmp/config.json"  
      dest: "/etc/config.json"
```

Variables

Scope of "version" variable

playbook.yml

```
---
- hosts: linux_servers
  vars:
    foo: "bar"
  tasks:
    - name: Copy a file
      vars:
        version: "beta"
      copy:
        src: "/tmp/config.json"
        dest: "/etc/config.json"
```

Variable

Variable

Scope of "foo" variable

Tasks

playbook.yml

```
- hosts: linux_servers
  vars:
    foo: "bar"
  tasks:
```

```
#1 - name: Copy a file
    copy:
      src: "/tmp/config.json"
      dest: "/etc/config.json"
```

← Task

← Module

} Arguments


```
#2 - name: Start Apache
    service:
      name: httpd
      state: started
```

← Task

← Module

} Arguments

Tasks

playbook.yml

```
---
- hosts: linux_servers
  vars:
    foo: "bar"
  tasks:
    - name: Copy a file
      copy:
        src: "/tmp/config.json"
        dest: "/etc/config.json"
```

← Module

} Arguments

Demo



Let's write a simple Ansible playbook

We'll just copy a file but we get to see all the important parts of a playbook



Demo



Let's write a playbook that configures something

We'll install Apache on Linux



Demo



We came here to learn Ansible with Windows

We'll install and configure IIS and set the proper parameters



Demo



Read Ansible Facts to set conditionals

Use the hostname to decide which application file to copy



Summary



Wrote our first Ansible playbooks

Graduated from basic file copy to complicated IIS and application configuration

Learned how conditionals can save time and gain efficiency

Next: Ansible Roles

