



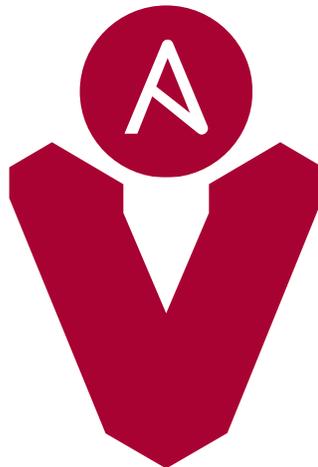
Grado en Ingeniería Informática - Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

Práctica 6 Máquinas virtuales y aprovisionamiento

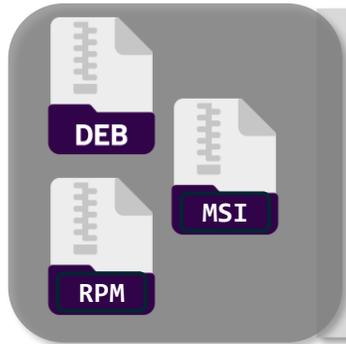


- 1. Introducción a Vagrant**
 - 2. Imágenes**
 - 3. Aprovisionamiento**
 - 4. Y yo, ¿qué puedo hacer en mi proyecto?**
 - 5. Tutorial: campo de entrenamiento de Vagrant**
 - 6. Tutorial: configurando Vagrant para una aplicación**
 - 7. Ejercicio práctico: Vagrant y uvlhub**
- 

- 1. Introducción a Vagrant**
2. Imágenes
3. Aprovisionamiento
4. Y yo, ¿qué puedo hacer en mi proyecto?
5. Tutorial: campo de entrenamiento de Vagrant
6. Tutorial: configurando Vagrant para una aplicación
7. Ejercicio práctico: Vagrant y uvlhub

1. Introducción a Vagrant

Pequeño recordatorio



Distintas instalaciones de módulos y paquetes de python de forma simultánea



Aislar dependencias más allá de python



Aislar todas las dependencias del sistema

Overhead y aislamiento

1. Introducción a Vagrant

¿Qué es?

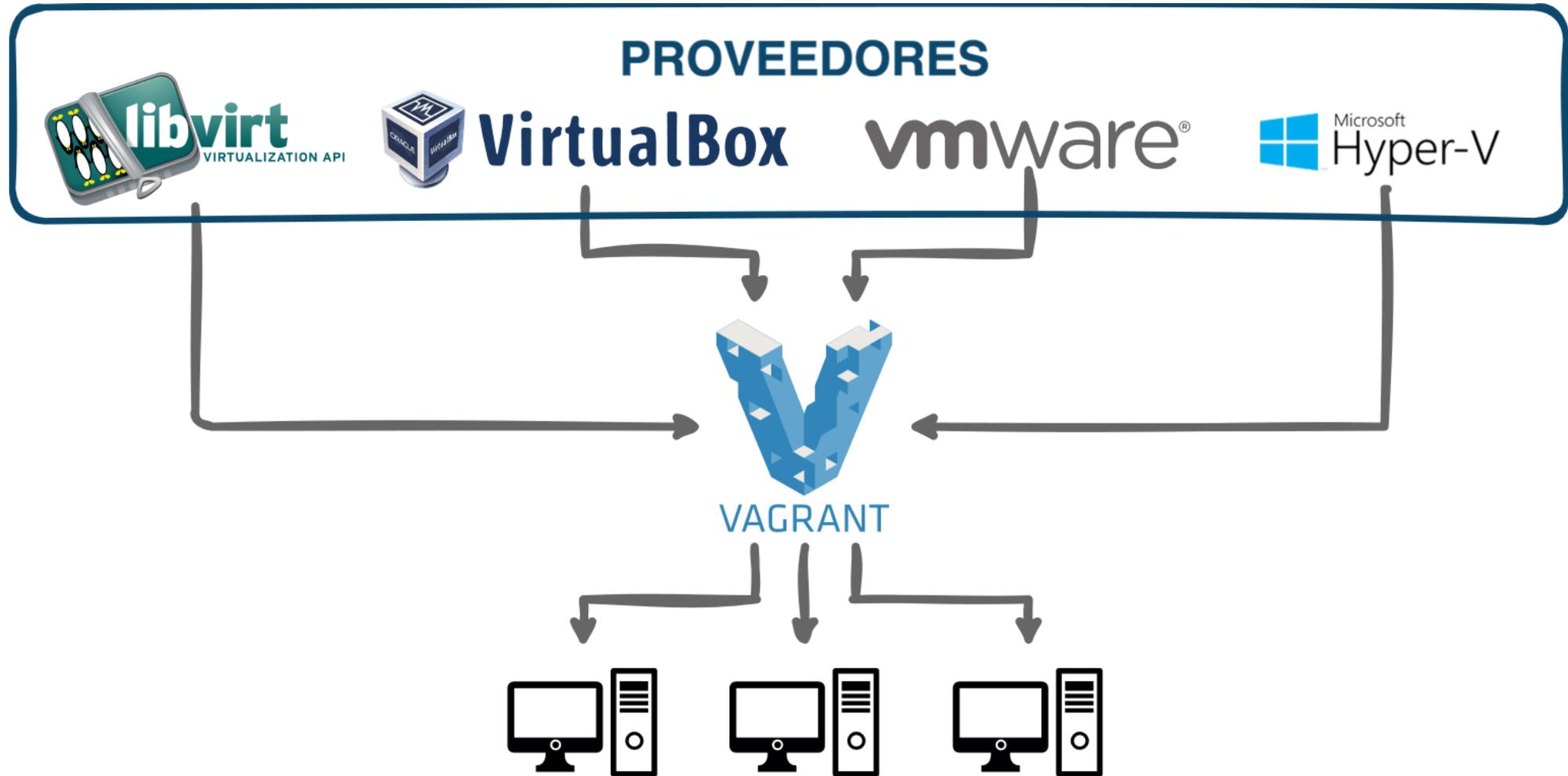


VAGRANT

- Herramienta que permite **crear y gestionar entornos virtuales** de desarrollo de **forma automatizada**
- Utiliza máquinas virtuales o contenedores configurados mediante un **Vagrantfile**

1. Introducción a Vagrant

Proveedores



1. **Introducción a Vagrant**
2. **Imágenes**
3. **Aprovisionamiento**
4. **Y yo, ¿qué puedo hacer en mi proyecto?**
5. **Tutorial: campo de entrenamiento de Vagrant**
6. **Tutorial: configurando Vagrant para una aplicación**
7. **Ejercicio práctico: Vagrant y uvlhub**

2. Imágenes

```
Vagrant.configure("2") do |config|
```

```
# Choose a base box
```

```
config.vm.box = "ubuntu/jammy64"
```

```
# Network configuration
```

```
config.vm.network "forwarded_port", guest: 5000, host: 5000
```

```
config.vm.network "forwarded_port", guest: 8089, host: 8089
```

```
# Synced folders
```

```
config.vm.synced_folder "../", "/vagrant"
```

```
# Provider configuration
```

```
config.vm.provider "virtualbox" do |vb|
```

```
  vb.memory = "2048"
```

```
  vb.cpus = 4
```

```
end
```

```
end
```

Imagen base

```
graph TD; IB[Imagen base] --> C1[config.vm.box]; P[proveedor] --> C2[config.vm.provider]; RP[Redirección de puertos] --> C3[config.vm.network]; V["volumen"]; C1 --- C2 --- C3 --- V;
```

Redirección de puertos

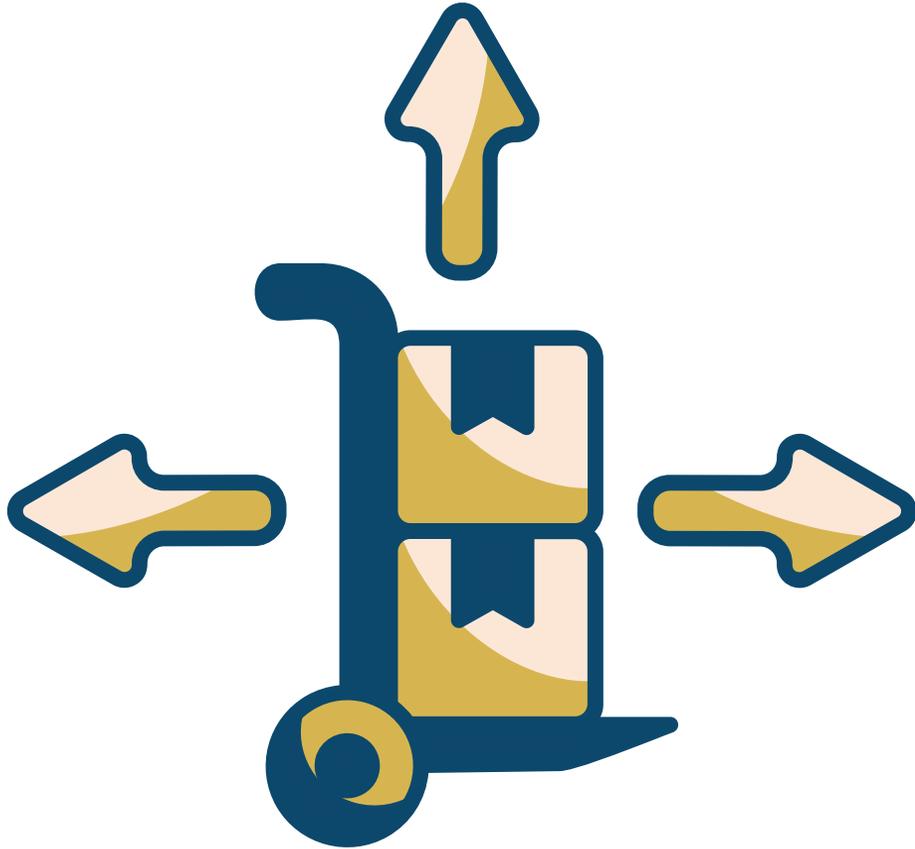
"volumen"

proveedor

1. Introducción a Vagrant
2. Imágenes
3. **Aprovisionamiento**
4. Y yo, ¿qué puedo hacer en mi proyecto?
5. Tutorial: campo de entrenamiento de Vagrant
6. Tutorial: configurando Vagrant para una aplicación
7. Ejercicio práctico: Vagrant y uvlhub

3. Aprovisionamiento

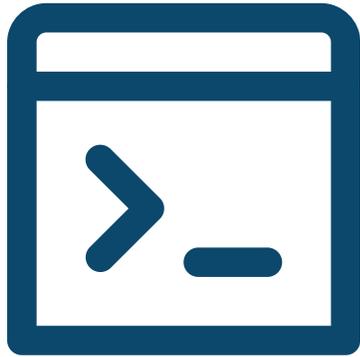
¿Qué es?



- **Proceso de configurar automáticamente** un entorno virtualizado
- Instala software, aplica configuraciones y prepara el sistema **según unas especificaciones**

3. Aprovisionamiento

Formas de aprovisionamiento



**Mediante shell
scripts**



ANSIBLE



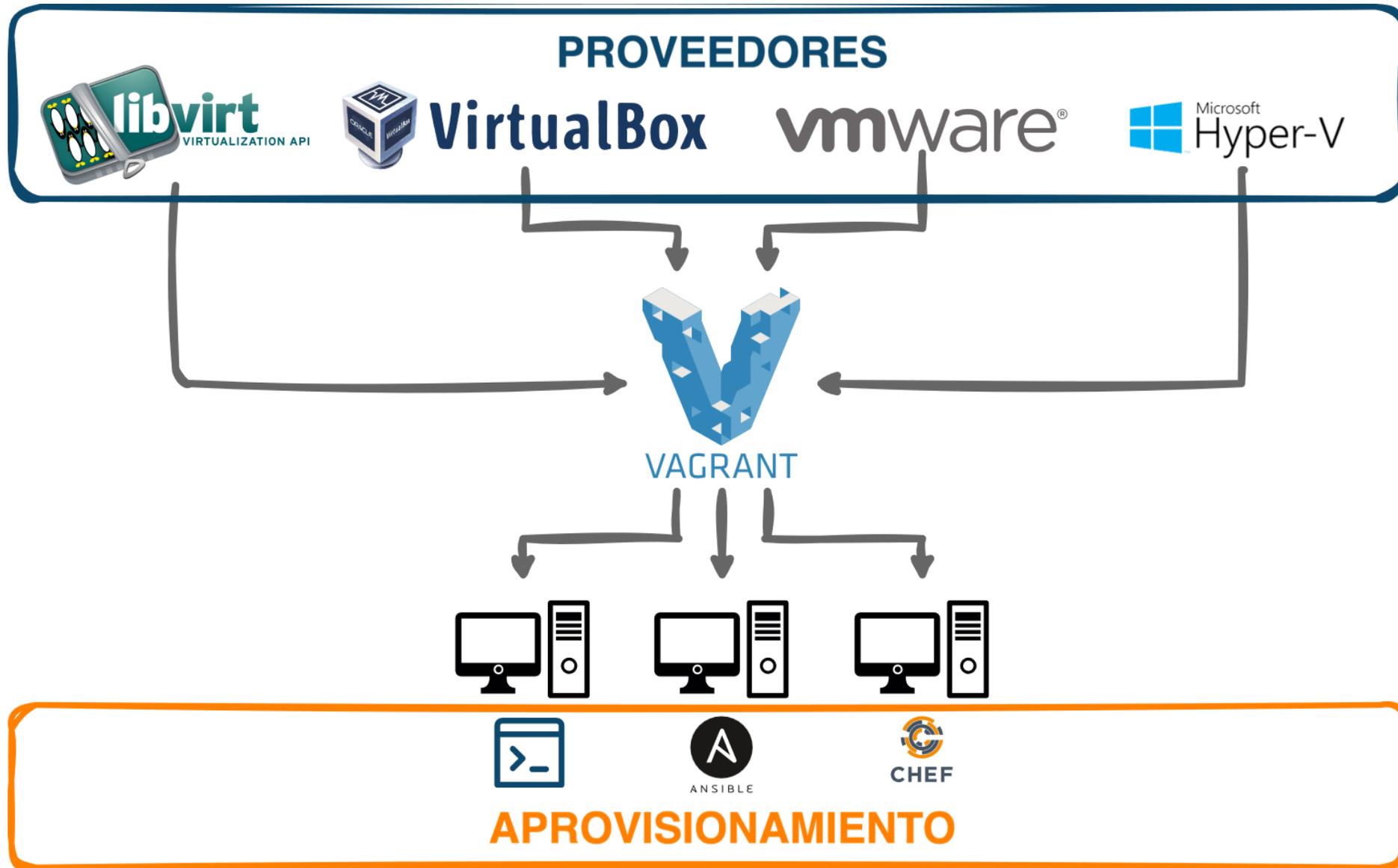
puppet



CHEF

**Mediante herramienta
de configuración**

3. Aprovisionamiento



3. Aprovechamiento

Ansible



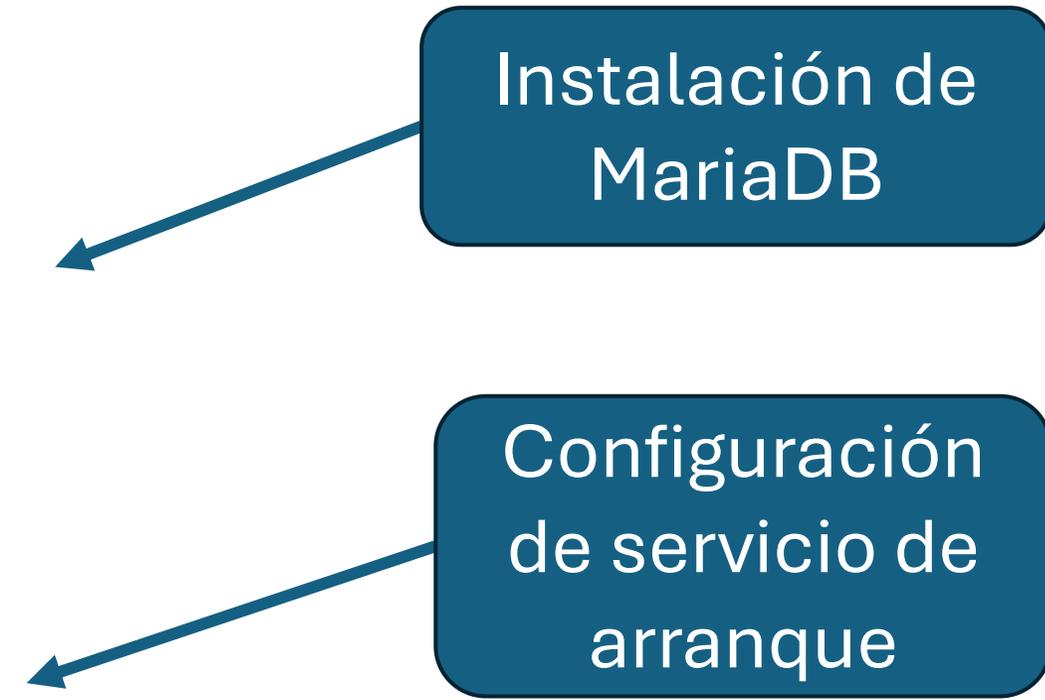
A N S I B L E

- **Herramienta de automatización** que permite gestionar configuraciones, implementar aplicaciones y orquestrar tareas en múltiples sistemas de manera sencilla
- Utiliza archivos YAML llamados **playbooks** (recetas)

3. Aprovisionamiento

Ejemplo de *playbook* (uvlhub / vagrant / 02_install_mariadb.yml)

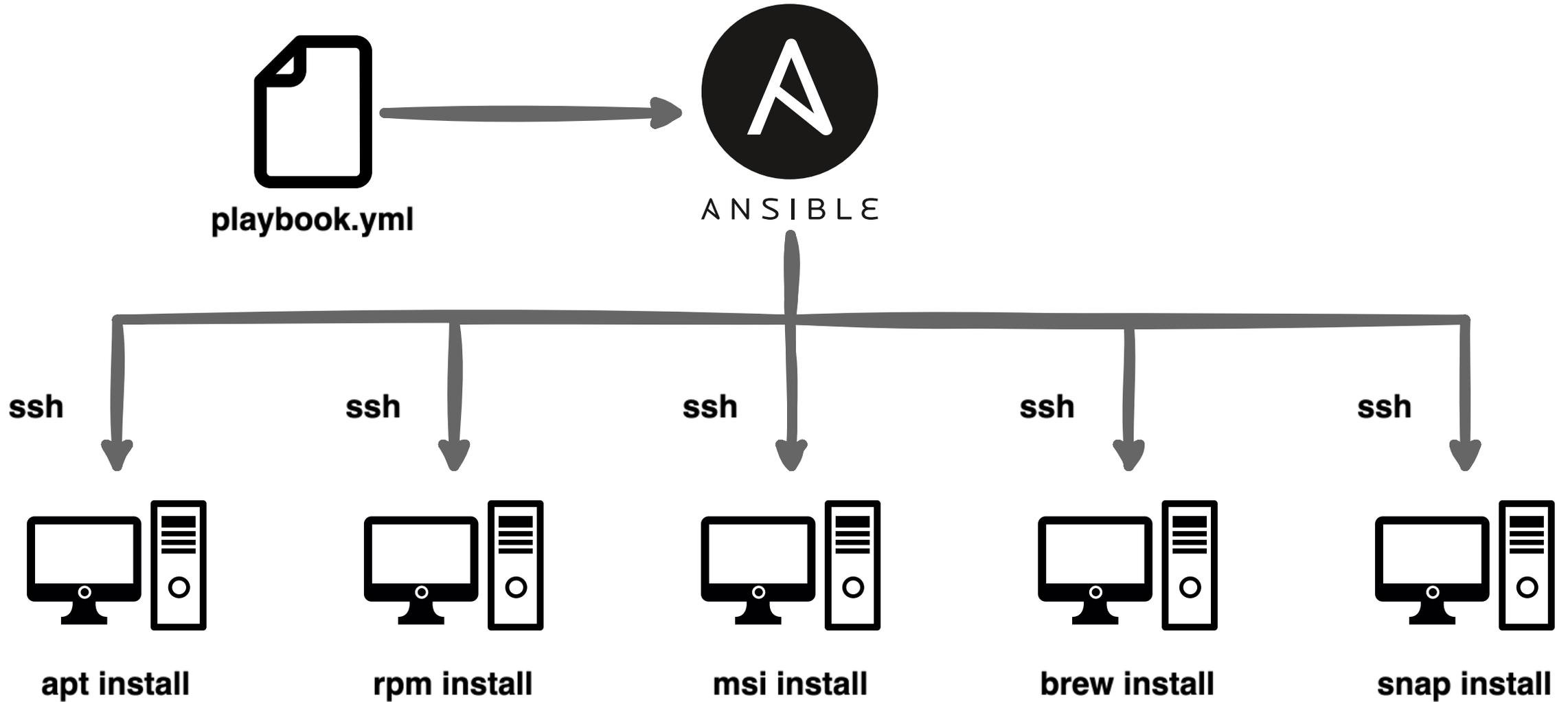
```
---  
- hosts: all  
  become: true  
  
  tasks:  
    - name: Install MariaDB Server  
      apt:  
        name:  
        - mariadb-server  
        - python3-pymysql  
        update_cache: yes  
        become: yes  
  
    - name: Start and enable MariaDB service  
      systemd:  
        name: mariadb  
        state: started  
        enabled: yes
```



Instalación de
MariaDB

Configuración
de servicio de
arranque

3. Aproveccionamiento



3. Aprovechamiento

Ventajas de Ansible



A N S I B L E

- Facilidad de uso
- Escalabilidad
- Portabilidad
- Velocidad y eficiencia
- Extensibilidad
- Automatización y reutilización
- Comunidad y documentación
- Seguridad
- Integración DevOps
- Bajo Coste

1. **Introducción a Vagrant**
2. **Imágenes**
3. **Aprovisionamiento**
4. **Y yo, ¿qué puedo hacer en mi proyecto?**
5. **Tutorial: campo de entrenamiento de Vagrant**
6. **Tutorial: configurando Vagrant para una aplicación**
7. **Ejercicio práctico: Vagrant y uvlhub**

4. Y yo, ¿qué puedo hacer en mi proyecto?



**¡Diseña tu propio
aprovisionamiento en
Vagrant!**

Modifica y/o crea
playbooks

Diseña *playbooks* para
desarrollo y producción

Usa **otra** herramienta de
aprovisionamiento

1. Introducción a Vagrant
2. Imágenes
3. Aprovisionamiento
4. Y yo, ¿qué puedo hacer en mi proyecto?
- 5. Tutorial: campo de entrenamiento de Vagrant**
6. Tutorial: configurando Vagrant para una aplicación
7. Ejercicio práctico: Vagrant y uvlhub

5. Tutorial: campo de entrenamiento de Vagrant



https://1984.lsi.us.es/wiki-egc/index.php/Tutorial_Campo_de_entrenamiento_de_Vagrant

1. Introducción a Vagrant
 2. Imágenes
 3. Aprovisionamiento
 4. Y yo, ¿qué puedo hacer en mi proyecto?
 5. Tutorial: campo de entrenamiento de Vagrant
 - 6. Tutorial: configurando Vagrant para una aplicación**
 7. Ejercicio práctico: Vagrant y uvlhub
- 

6. Tutorial: configurando Vagrant para una aplicación



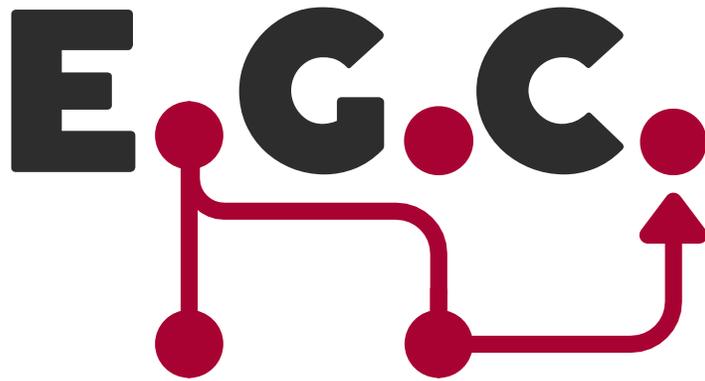
https://1984.lsi.us.es/wiki-egc/index.php/Tutorial_configurando_vagrant_para_una_aplicaci%C3%B3n

1. **Introducción a Vagrant**
2. **Imágenes**
3. **Aprovisionamiento**
4. **Y yo, ¿qué puedo hacer en mi proyecto?**
5. **Tutorial: campo de entrenamiento de Vagrant**
6. **Tutorial: configurando Vagrant para una aplicación**
7. **Ejercicio práctico: Vagrant y uvlhub**

7. Ejercicio práctico: Vagrant y uvlhub



docs.uvlhub.io/installation/installation_with_vagrant



Grado en Ingeniería Informática - Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

¡Gracias!

*“El trabajo se expande hasta
llenar el tiempo disponible para
su realización.”*

- Cyril Northcote Parkinson