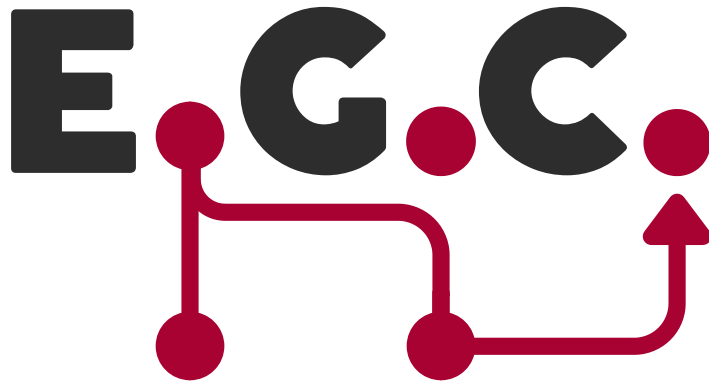




Grado en Ingeniería Informática – Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

Procesamiento y compilación aisladas





Receta
(la pipeline)

**Mismo
plato**

Cocina
(el entorno)

Objetivo principal: buscar el aislamiento y la reproducibilidad de los artefactos software, intentando resolver el problema de la dependencia del entorno ("*en mi máquina funciona*") y justificar el adecuado uso de tecnologías como Docker, Vagrant y Dev Containers.



- 1. De la automatización a la fiabilidad**
 - 2. Aislamiento y reproducibilidad**
 - 3. El entorno del desarrollador**
 - 4. ¿Y cómo afecta la IA a todo esto?**
 - 5. Resumen**
- 

1. De la automatización a la fiabilidad |

El entorno IMPORTA

Automatización del **QUÉ**



Compilar



Testear



Desplegar



Entregas rápidas



Entregas repetibles



1. De la automatización a la fiabilidad |

El entorno IMPORTA

Automatización del **QUÉ**



Compilar



Testear



Desplegar



Entregas rápidas



Entregas repetibles



¿Qué pasa con la FIABILIDAD?



1. De la automatización a la fiabilidad |

El entorno IMPORTA

DÓNDE

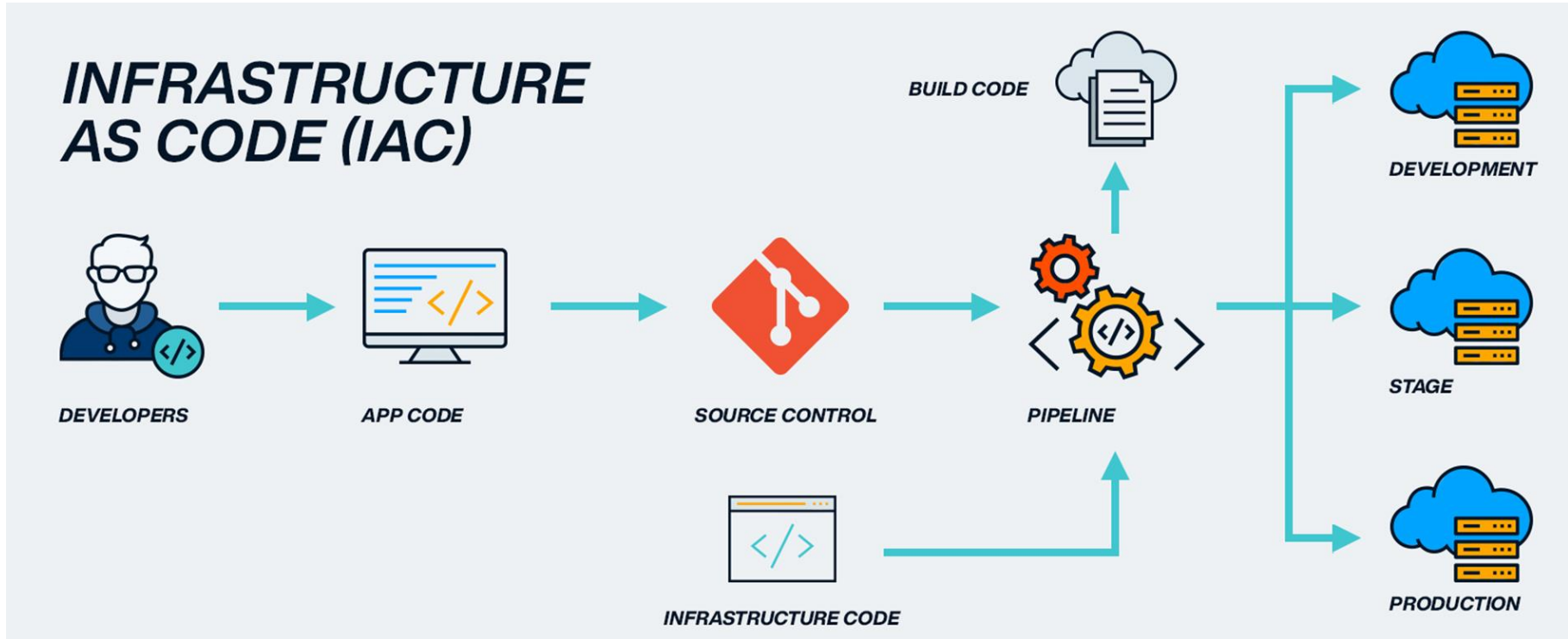


CÓMO



1. De la automatización a la fiabilidad |

El entorno IMPORTA



- Autocontenido
- Declarativo
- Desechable
- Idéntico
- Predecible
- Transparente
- Versionable

El entorno puede ser gestionado como el código lo que permitiría que fuera:

1. De la automatización a la fiabilidad |

El entorno IMPORTA

martinFowler.com

Snowflake Server

It can be finicky business to keep a production server running. You have to ensure the operating system and any other dependent software is properly patched to keep it up to date. Hosted applications need to be upgraded regularly. Configuration changes are regularly needed to tweak the environment so that it runs efficiently and communicates properly with other systems. This requires some mix of command-line invocations, jumping between GUI screens, and editing text files.

The result is a unique snowflake – good for a ski resort, bad for a data center.

VS

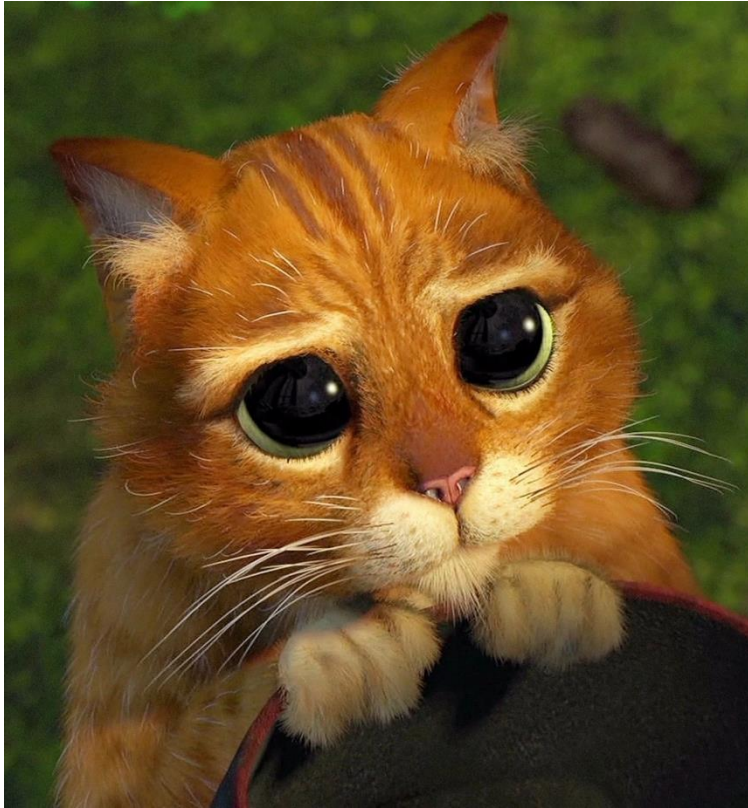
Ficheros versionados
dependencias · Dockerfile
infraestructura como código

reconstruir

Entorno
reproducible

1. De la automatización a la fiabilidad |

El entorno IMPORTA

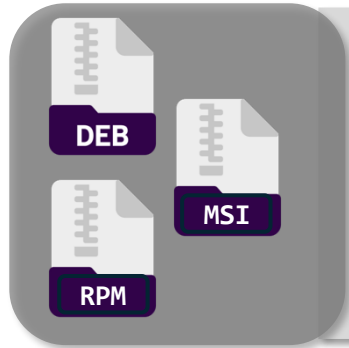


VS



- 1. De la automatización a la fiabilidad**
 - 2. Aislamiento y reproducibilidad**
 - 3. El entorno del desarrollador**
 - 4. ¿Y cómo afecta la IA a todo esto?**
 - 5. Resumen**
- 

2. Aislamiento y reproducibilidad



Distintas instalaciones de módulos y paquetes de python de forma simultánea



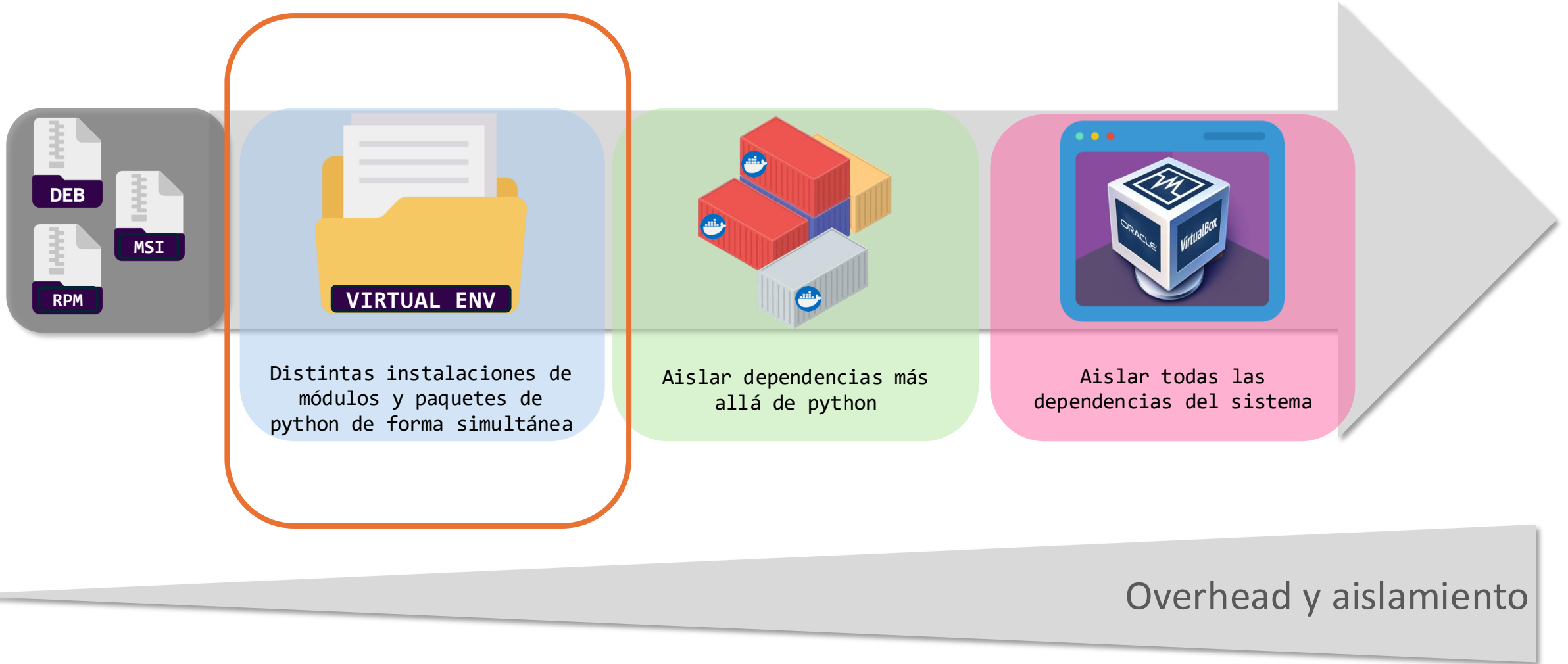
Aislar dependencias más allá de Python



Aislar todas las dependencias del sistema

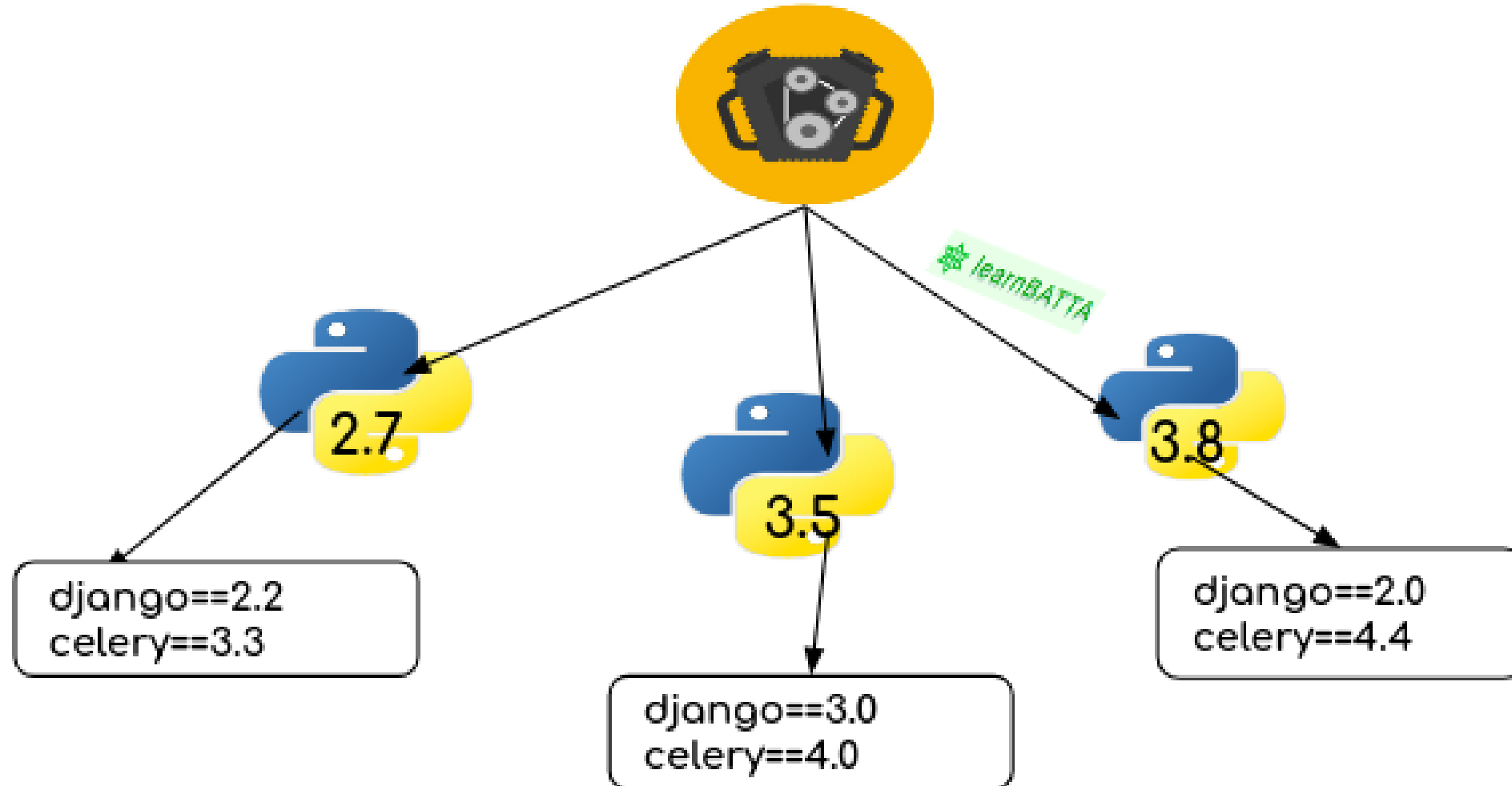
Overhead y aislamiento

2. Aislamiento y reproducibilidad

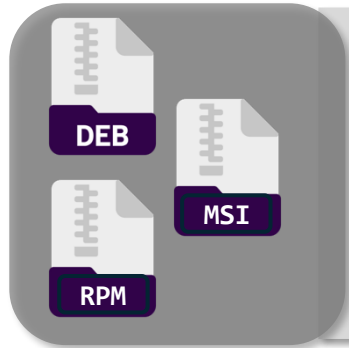


2. Aislamiento y reproducibilidad

Virtualenv



2. Aislamiento y reproducibilidad



Distintas instalaciones de módulos y paquetes de python de forma simultánea



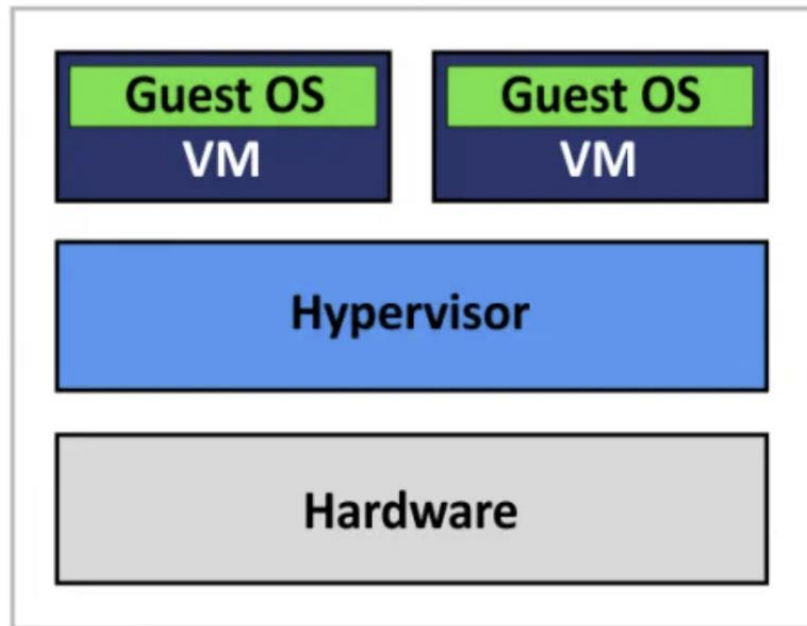
Aislar dependencias más allá de python



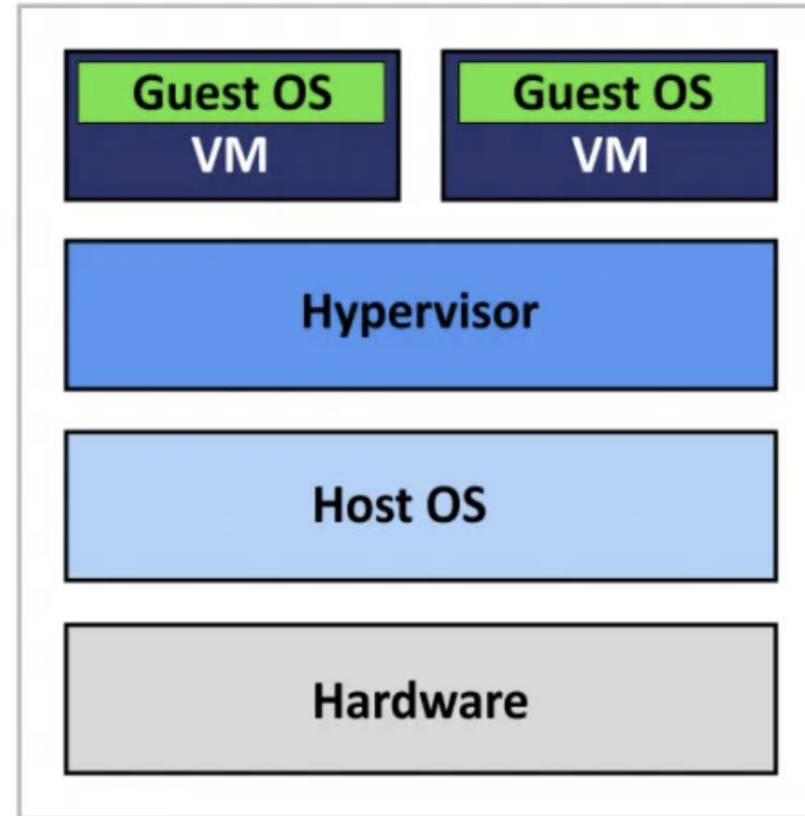
Aislar todas las dependencias del sistema

Overhead y aislamiento

2. Aislamiento y reproducibilidad | Virtualización



Type 1 Hypervisor
(Bare-Metal Architecture)



Type 2 Hypervisor
(Hosted Architecture)

2. Aislamiento y reproducibilidad | Vagrant

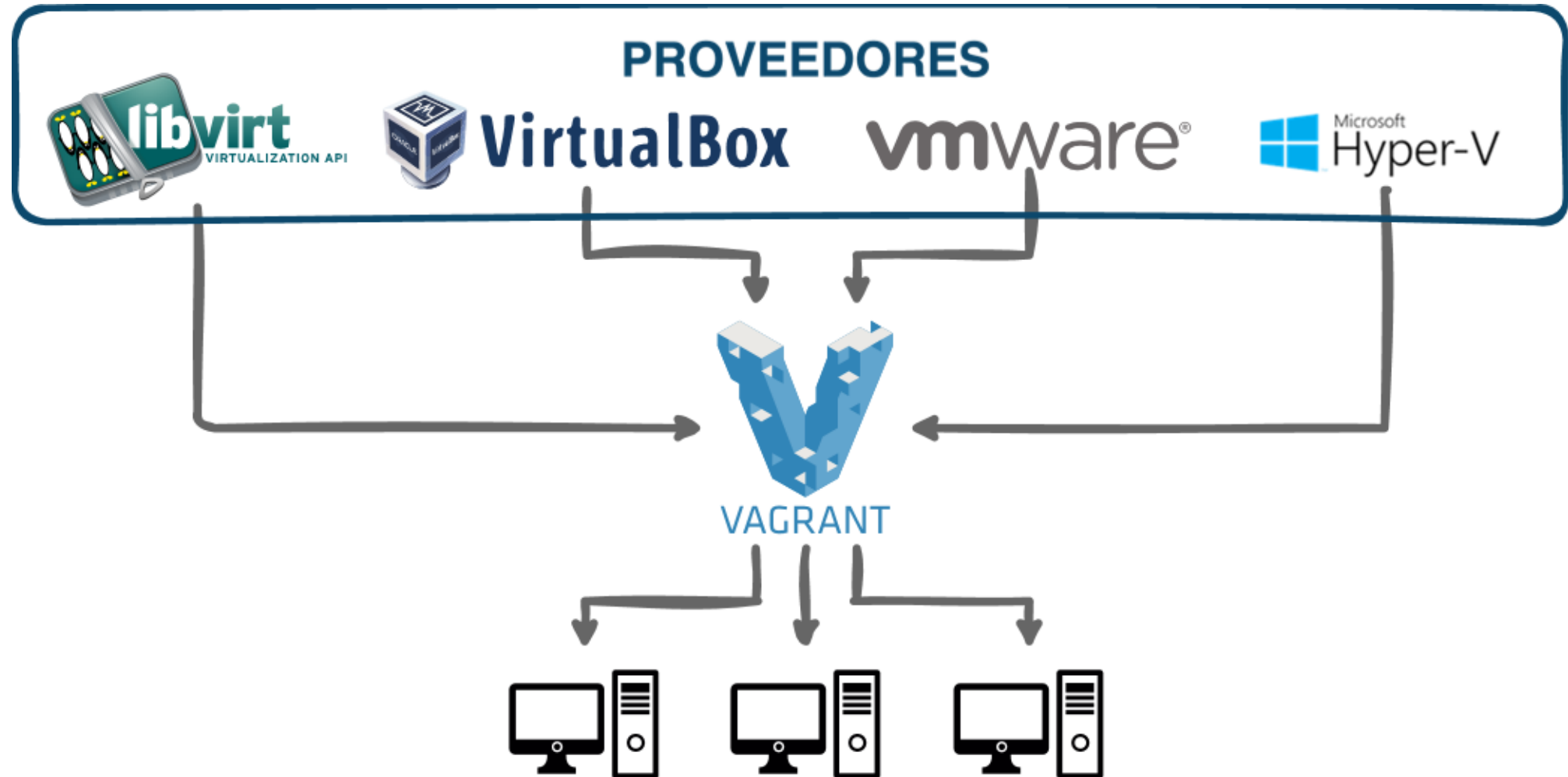
¿Qué es?



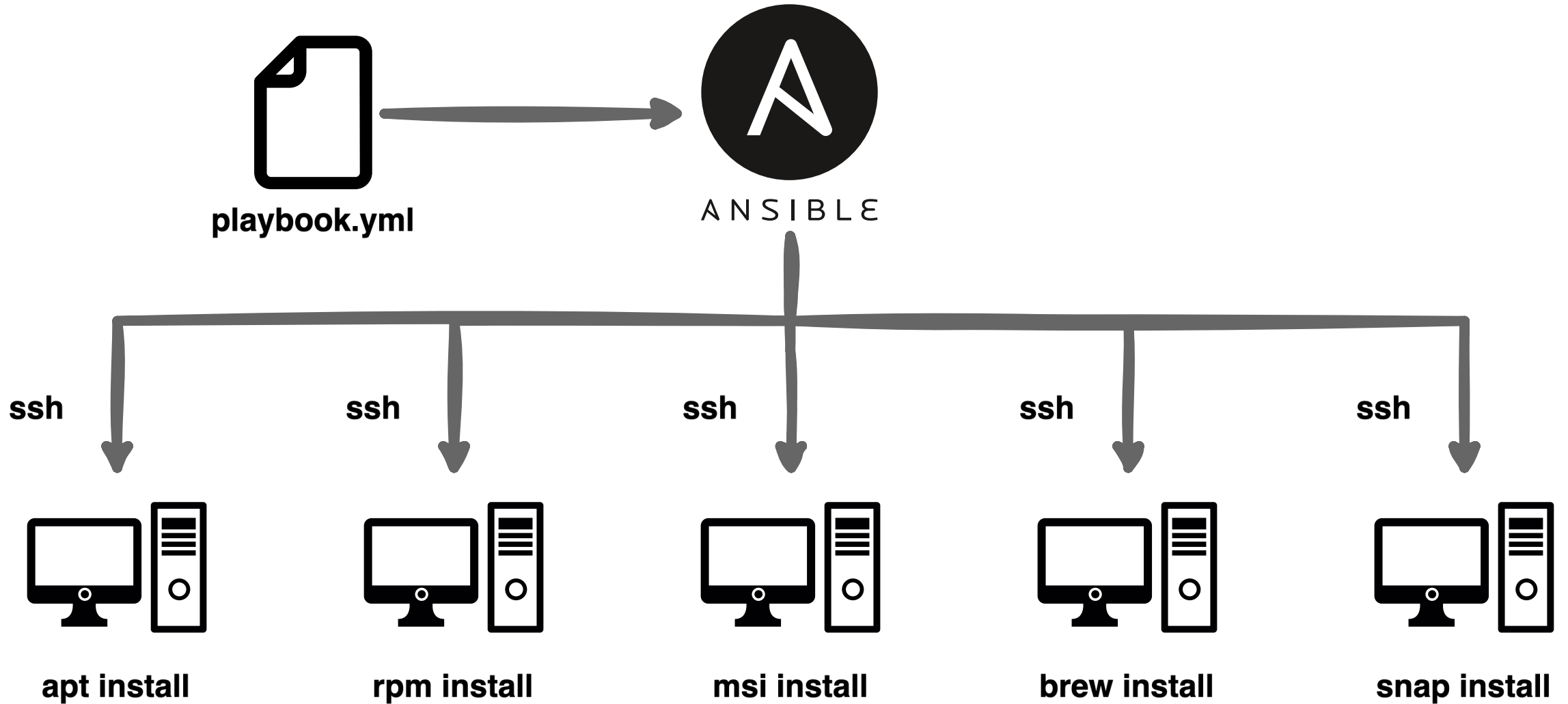
- Herramienta que permite **crear y gestionar entornos virtuales** (generalmente de desarrollo) de **forma automatizada**
- Utiliza máquinas virtuales o contenedores configurados mediante un **Vagrantfile**

2. Aislamiento y reproducibilidad | Vagrant

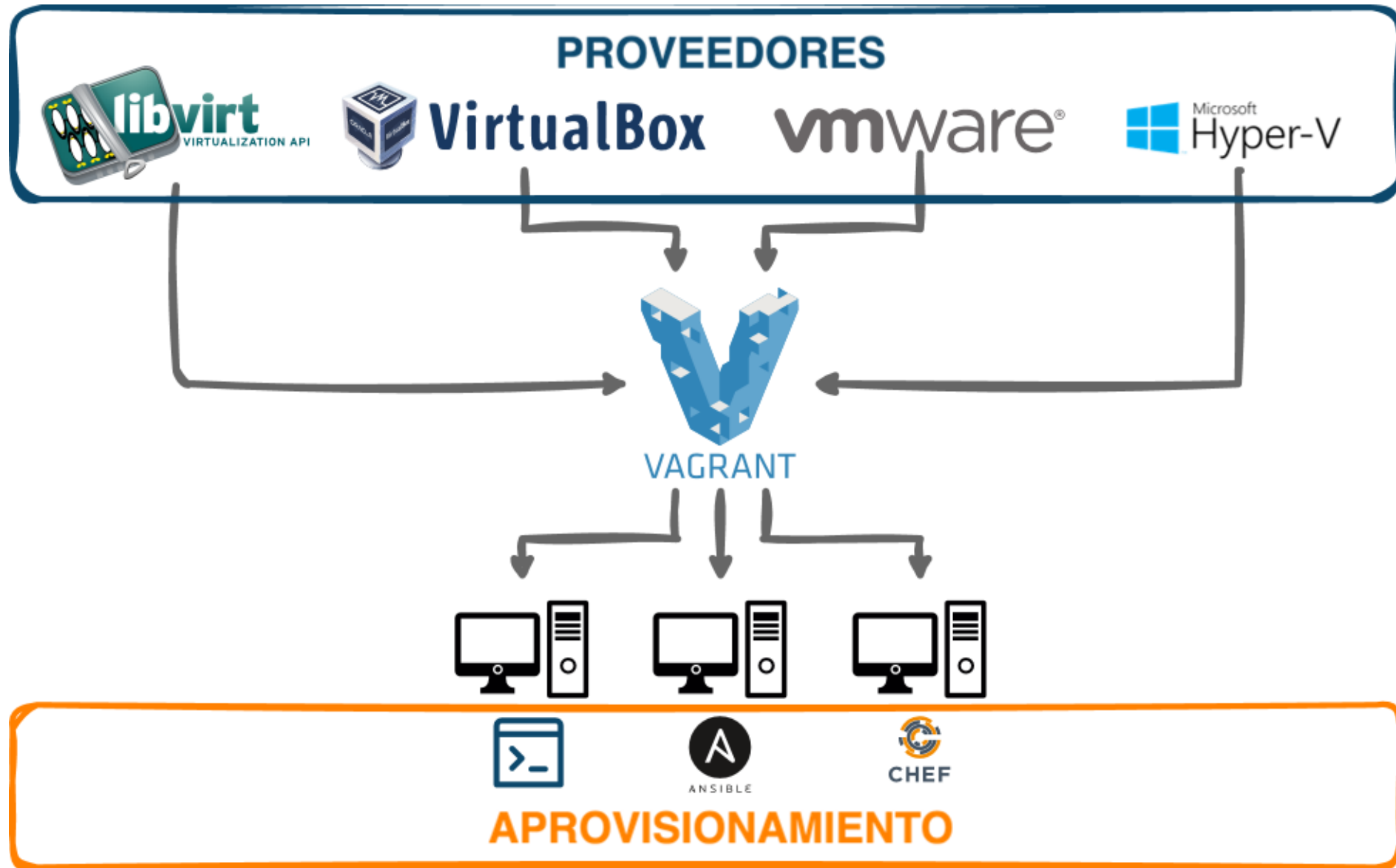
Proveedores



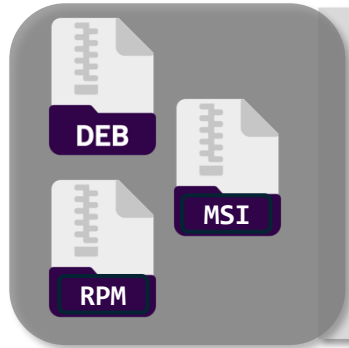
2. Aislamiento y reproducibilidad | Ansible



2. Aislamiento y reproducibilidad | Vagrant



2. Aislamiento y reproducibilidad



VIRTUAL ENV

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

2. Aislamiento y reproducibilidad | Procesamiento aislado

El **procesamiento aislado** es la ejecución de cualquier tarea (compilar, testear, empaquetar, etc.) dentro de un espacio de *nombres limitado* (namespaces) o un *sandbox*.



**Hemerticity =
capacidad de
estar hermético**



2. Aislamiento y reproducibilidad | Procesamiento aislado

¿Qué son los namespaces?



2. Aislamiento y reproducibilidad | Procesamiento aislado

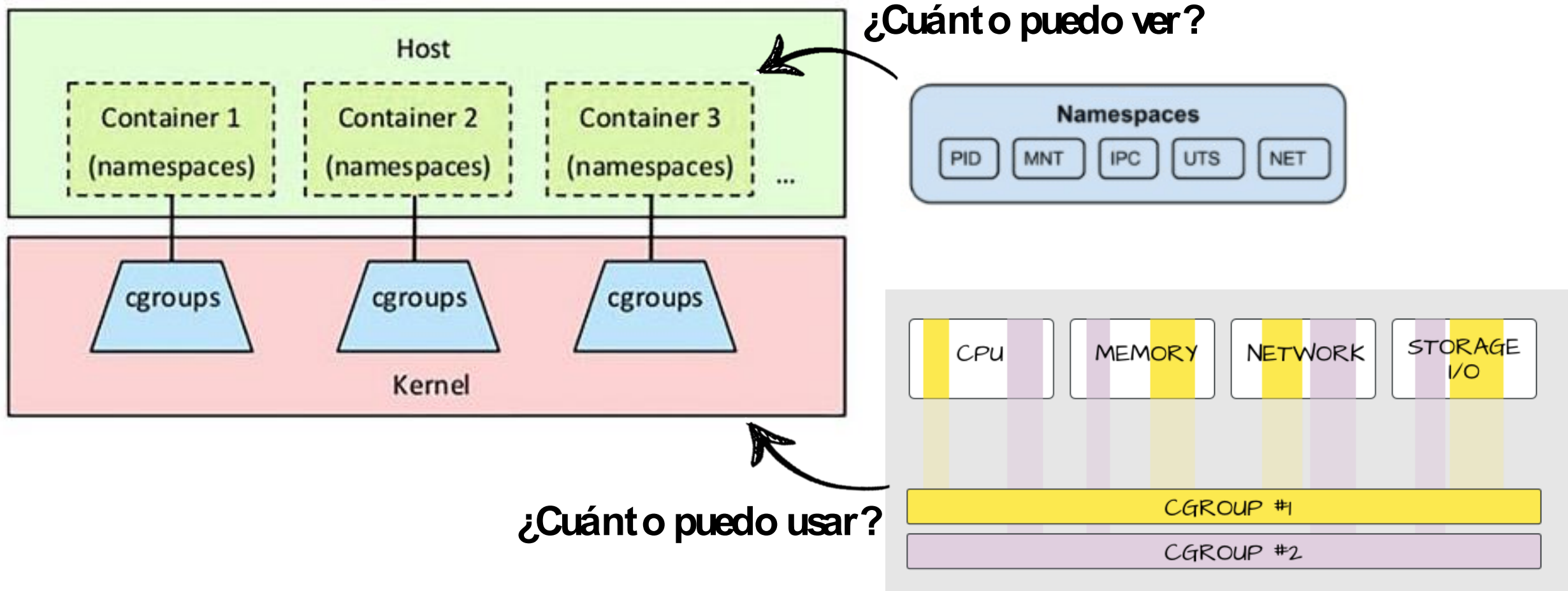
¿Qué son los namespaces?

¿Y los cgroups?



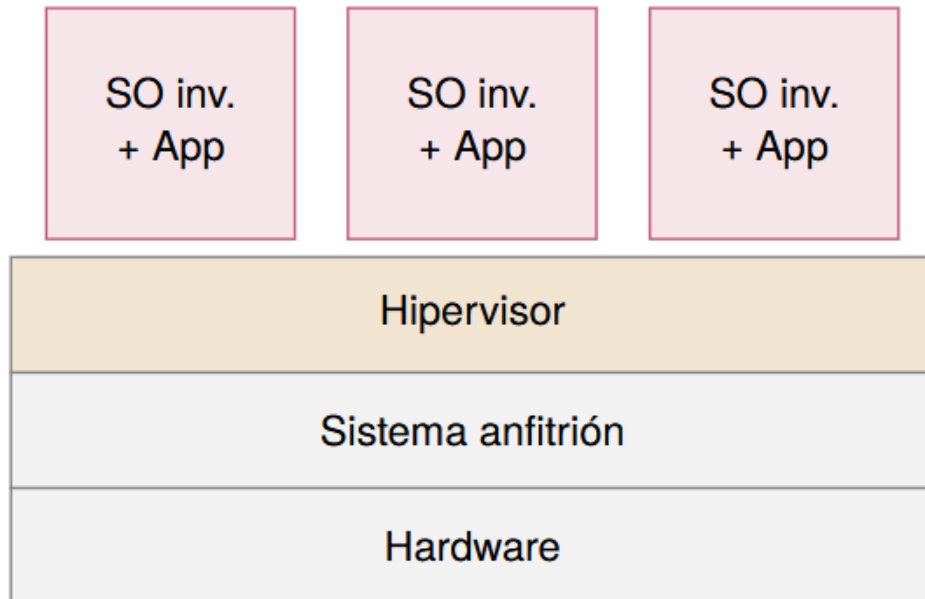
2. Aislamiento y reproducibilidad | Procesamiento aislado

Aislamiento lógico

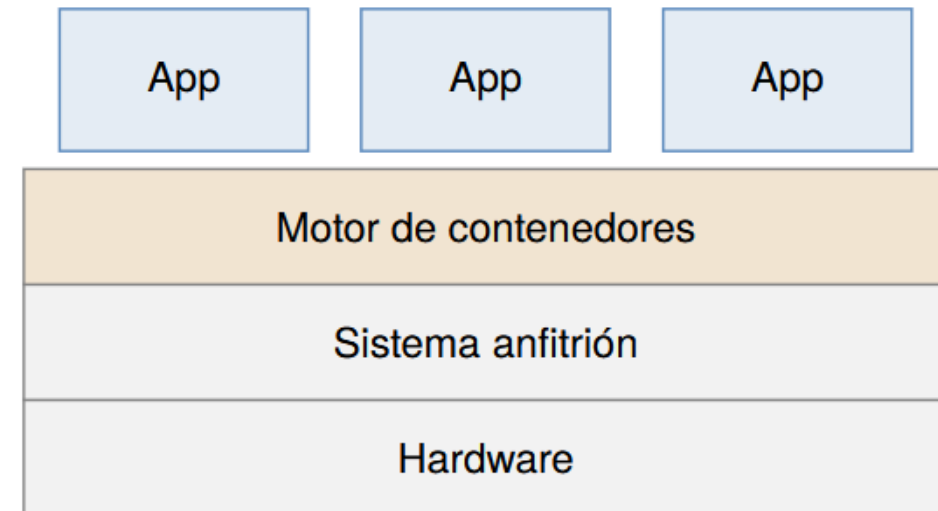


2. Aislamiento y reproducibilidad | Virtualización y contenerización

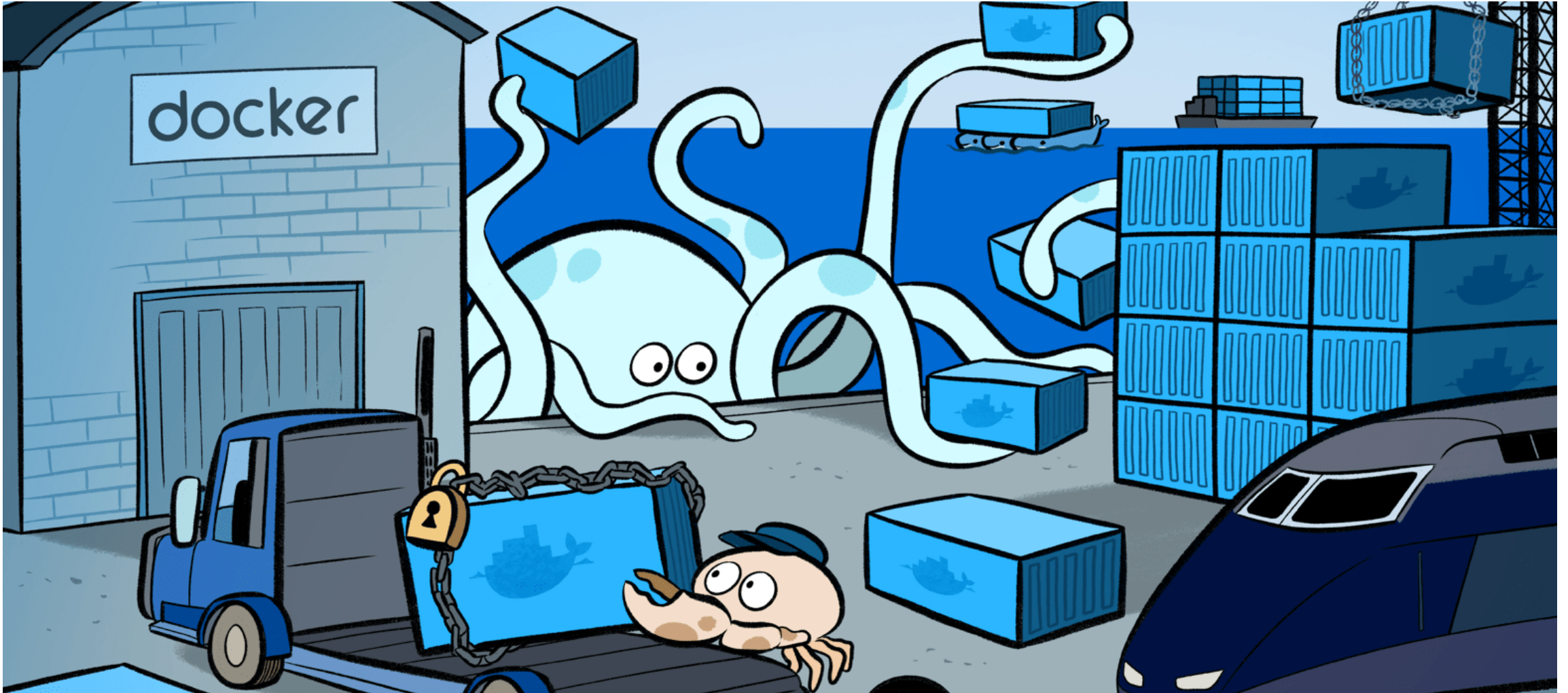
Máquinas virtuales



Contenedores



2. Aislamiento y reproducibilidad | Docker



2. Aislamiento y reproducibilidad + Docker

+ Dockerfile



2. Aislamiento y reproducibilidad | Docker

```
FROM registry.access.redhat.com/ubi8/nodejs-16:latest

# Copy package.json and package-lock.json
COPY package*.json ./

# Install npm production packages
RUN npm install --production

COPY . /opt/app-root/src

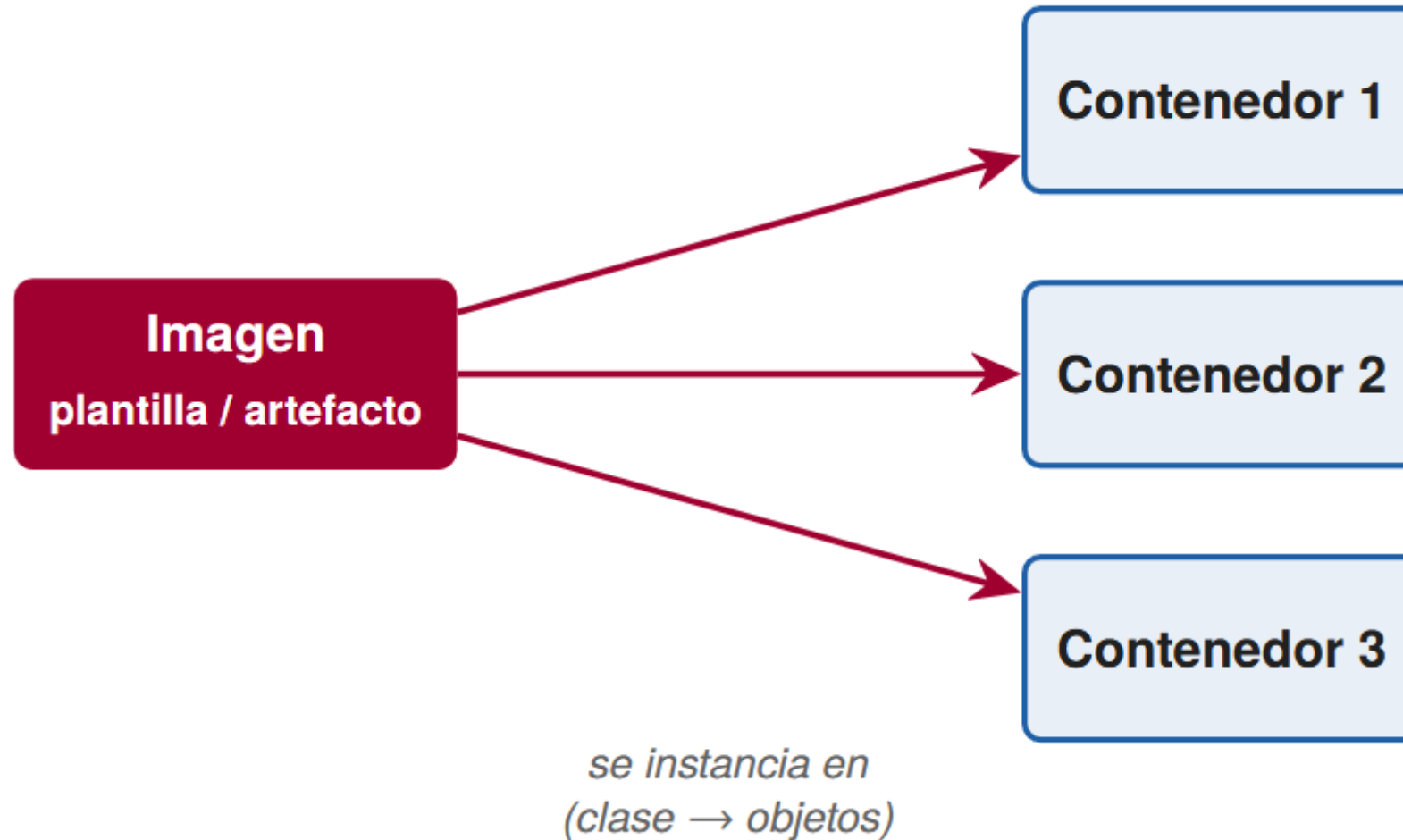
ENV NODE_ENV production
ENV PORT 3000

EXPOSE 3000

CMD ["npm", "start"]
```

<https://github.com/nodeshift/docker/blob/main/Dockerfile>

2. Aislamiento y reproducibilidad | Docker



2. Aislamiento y reproducibilidad | Docker compose

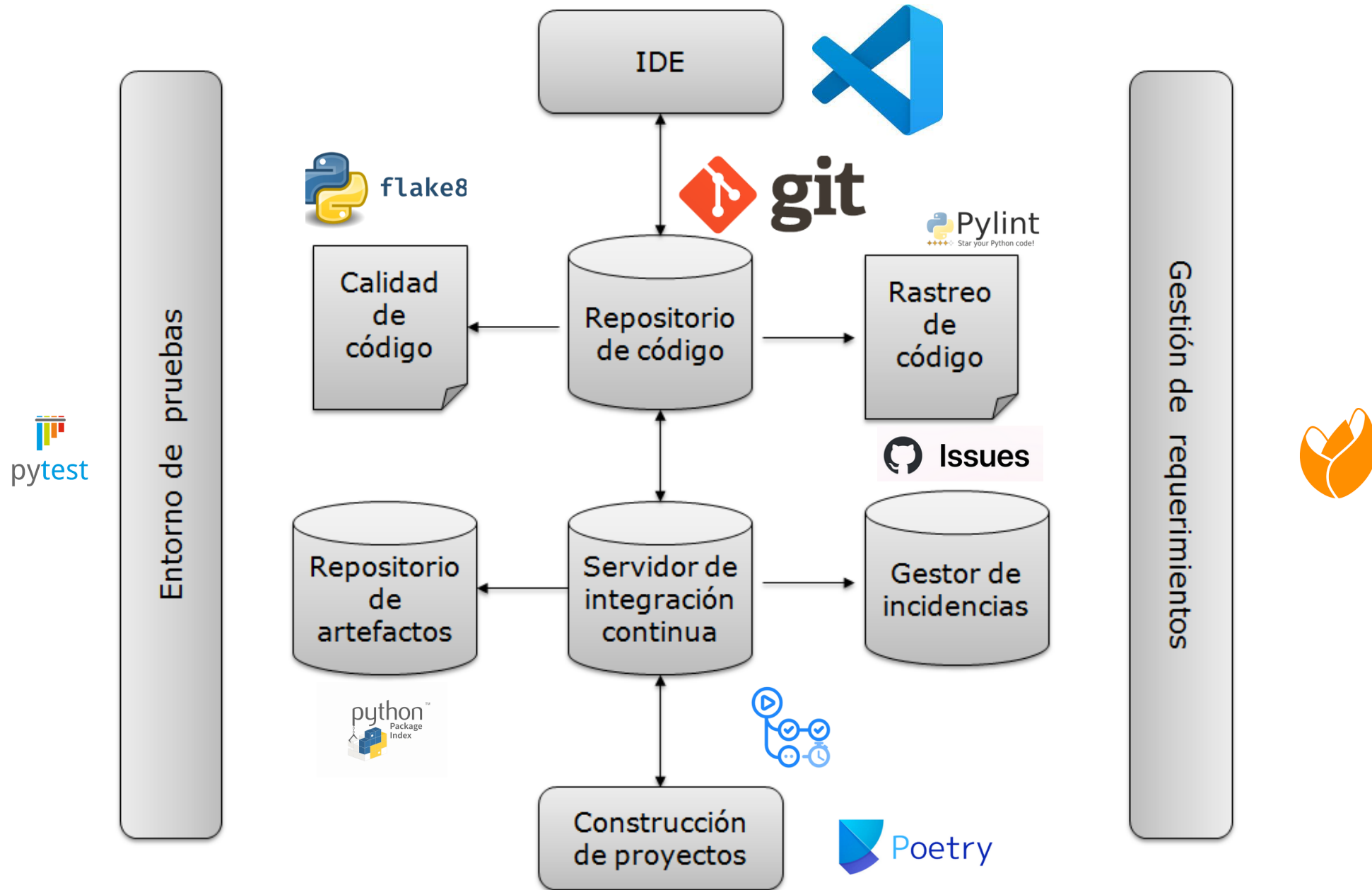
```
services:
  db:
    # We use a mariadb image which supports both amd64 & arm64 architecture
    image: mariadb:10.6.4-focal
    # If you really want to use MySQL, uncomment the following line
    #image: mysql:8.0.27
    command: '--default-authentication-plugin=mysql_native_password'
    volumes:
      - db_data:/var/lib/mysql
    restart: always
    environment:
      - MYSQL_ROOT_PASSWORD=somewordpress
      - MYSQL_DATABASE=wordpress
      - MYSQL_USER=wordpress
      - MYSQL_PASSWORD=wordpress
    expose:
      - 3306
      - 33060
```

```
wordpress:
  image: wordpress:latest
  ports:
    - 80:80
  restart: always
  environment:
    - WORDPRESS_DB_HOST=db
    - WORDPRESS_DB_USER=wordpress
    - WORDPRESS_DB_PASSWORD=wordpress
    - WORDPRESS_DB_NAME=wordpress
  volumes:
    db_data:
```

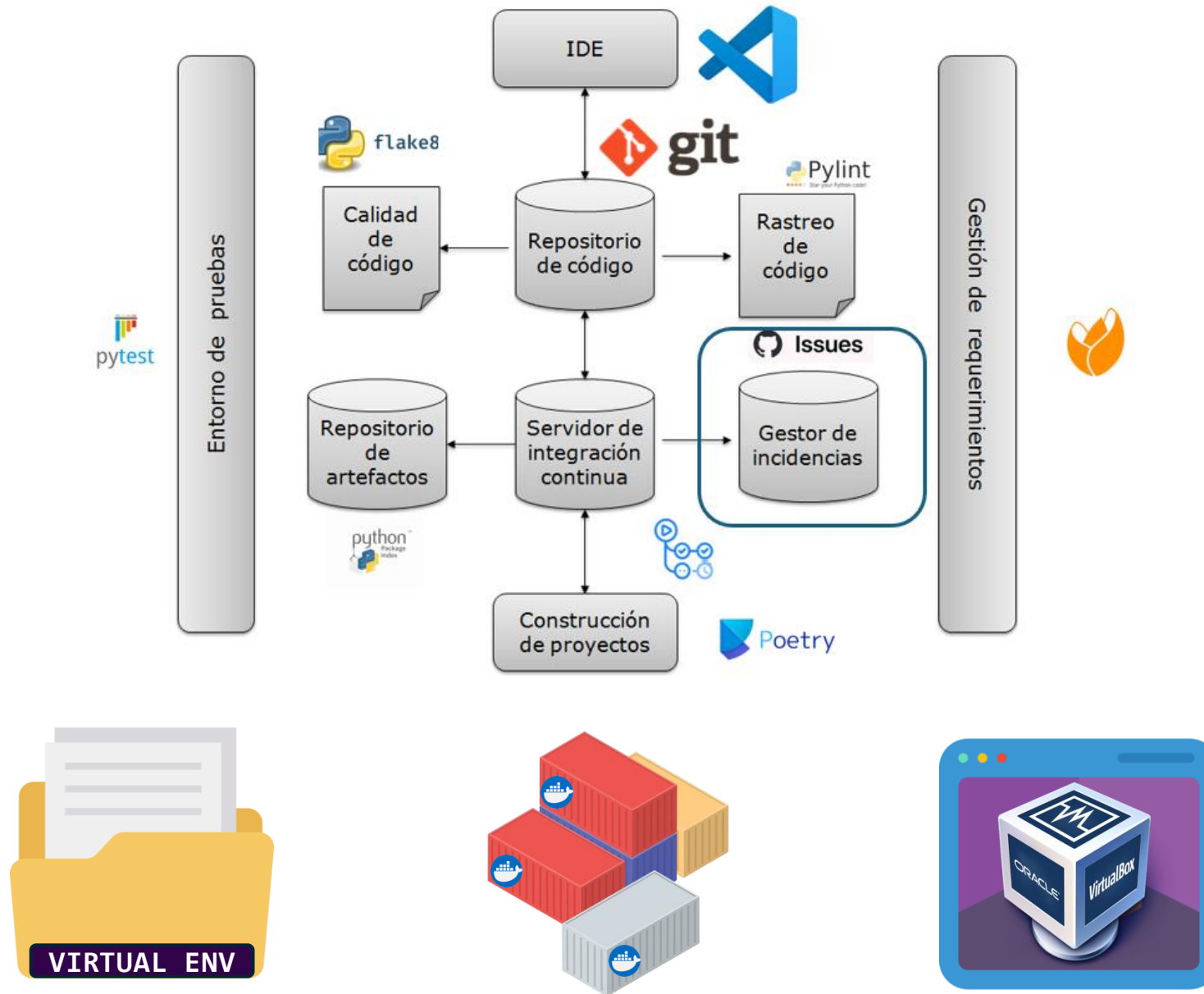
<https://github.com/docker/awesome-compose/blob/master/wordpress-mysql/compose.yaml>

- 1. De la automatización a la fiabilidad**
 - 2. Aislamiento y reproducibilidad**
 - 3. El entorno del desarrollador**
 - 4. ¿Y cómo afecta la IA a todo esto?**
 - 5. Resumen**
- 

3. El entorno de desarrollo | ecosistemas



3. El entorno de desarrollo | ecosistemas



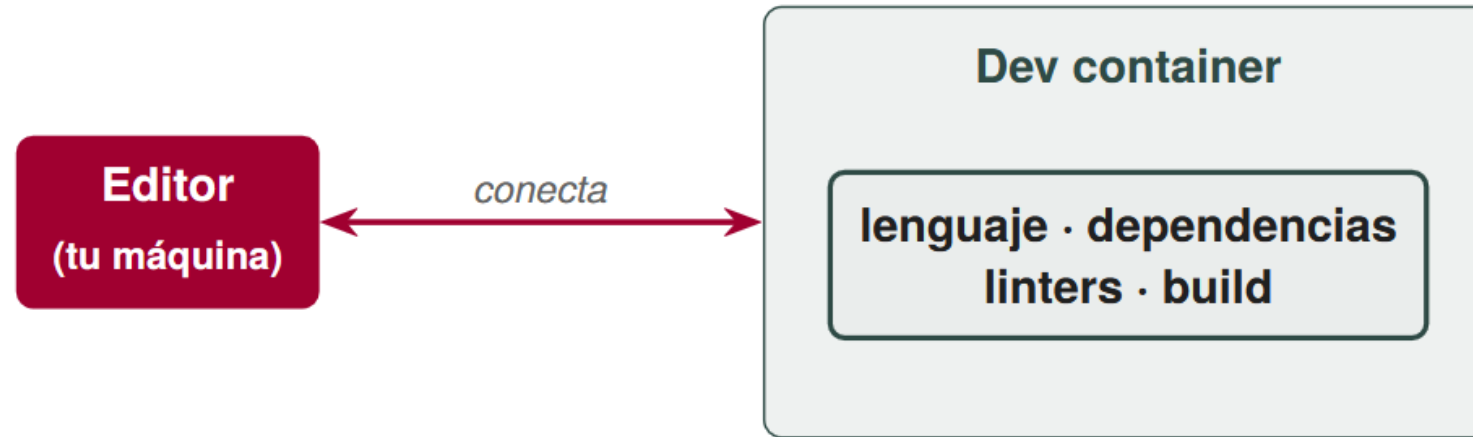
3. El entorno del desarrollador

Si la CI y la puesta en producción se han aislado
¿Por qué el desarrollo sigue haciéndose en un entorno
“sucio”?



Posible solución: devcontainers

3. El entorno del desarrollador | devcontainers

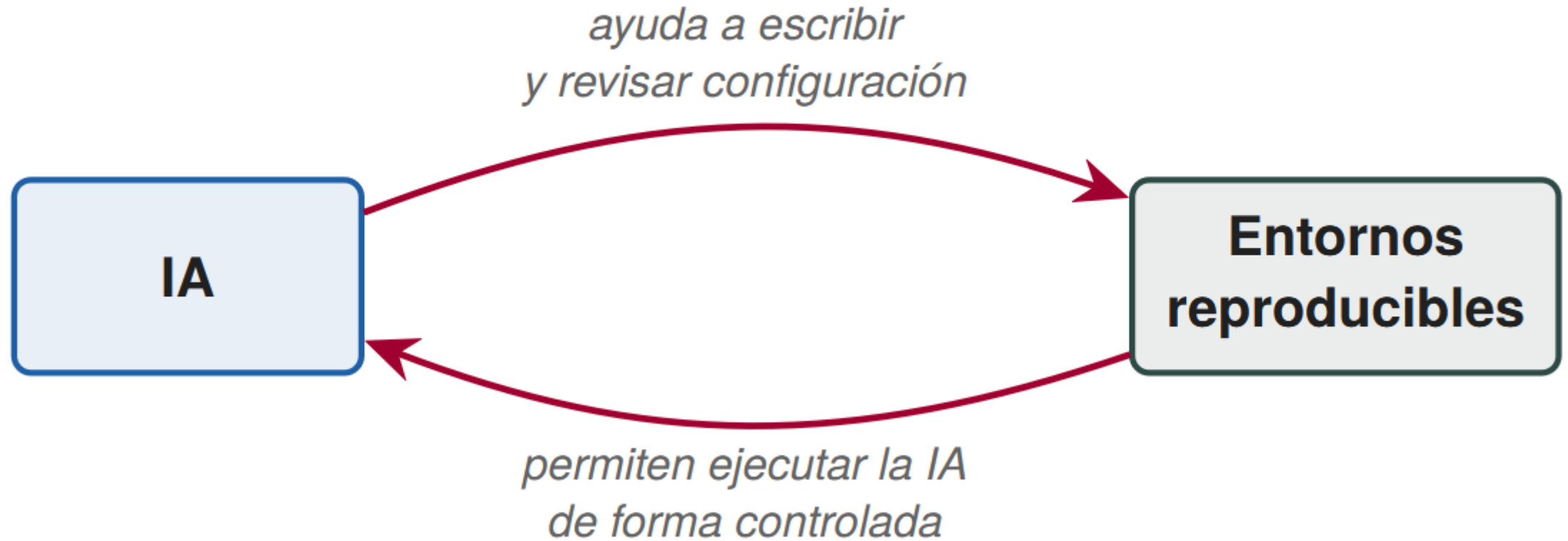


Install the Dev Containers extension

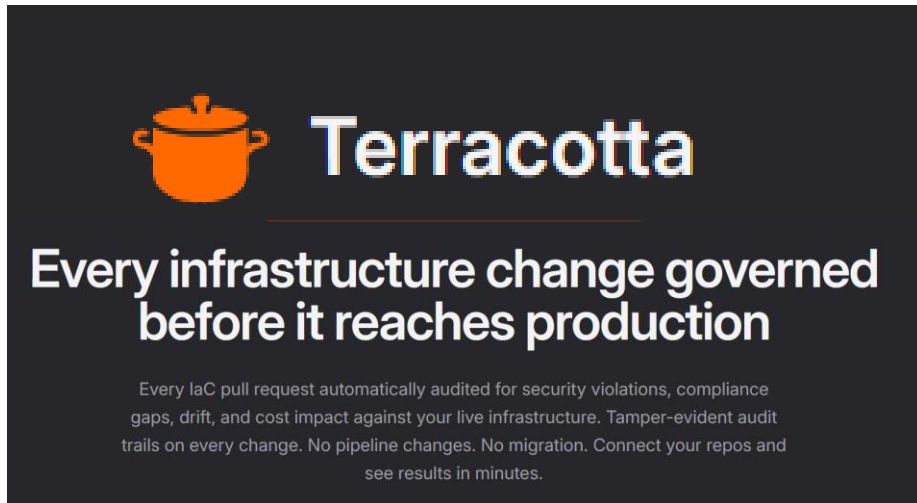
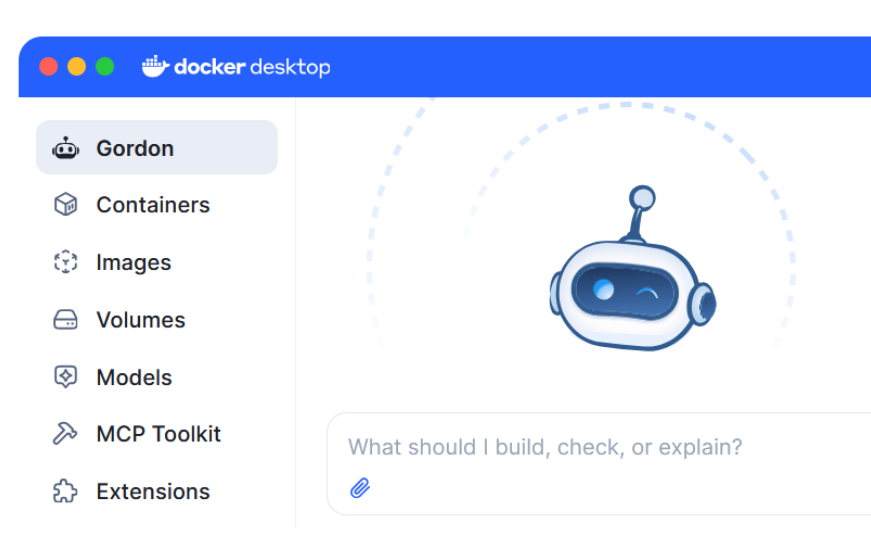
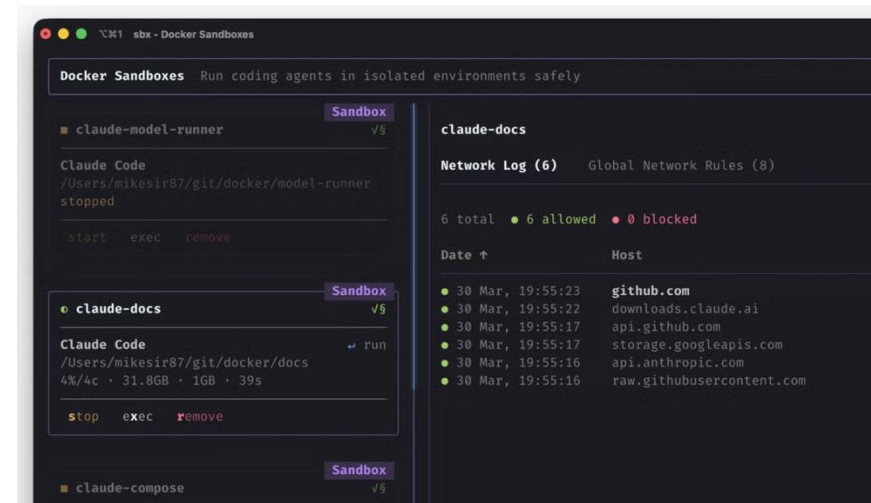
The screenshot shows the Visual Studio Marketplace page for the "Dev Containers" extension by Microsoft. On the left is the extension's icon, a blue circle with a white cube. To the right, the title "Dev Containers" is displayed with the version "v0.251.0" and a "Preview" badge. Below the title, it shows the Microsoft logo, a download icon with "14,826,246" downloads, and a five-star rating with "(37)" reviews. The description reads "Open any folder or repository inside a Docker container". At the bottom, there are buttons for "Disable", "Uninstall", and "Switch to Pre-Release Version", along with a settings gear icon. A status message at the very bottom states "This extension is enabled globally."

1. De la automatización a la fiabilidad
 2. Aislamiento y reproducibilidad
 3. El entorno del desarrollador
 4. ¿Y cómo afecta la IA a todo esto?
 5. Resumen
- 

3. ¿Y cómo afecta la IA a todo esto?



3. ¿Y cómo afecta la IA a todo esto?



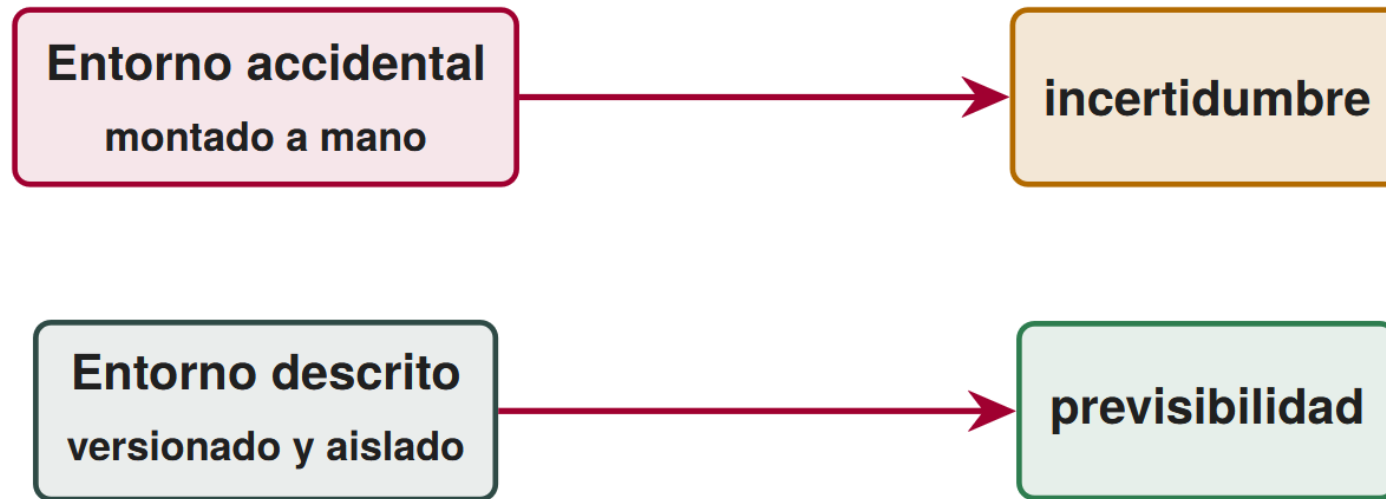
- 1. De la automatización a la fiabilidad**
 - 2. Aislamiento y reproducibilidad**
 - 3. El entorno del desarrollador**
 - 4. ¿Y cómo afecta la IA a todo esto?**
 - 5. Resumen**
- 

5. Resumen

- 🌐 El problema del entorno exige aislamiento → **hermeticidad**
- 🌐 El aislamiento se basa en **inmutabilidad y reproducibilidad**
- 🌐 Las tecnologías son herramientas para implementar estos principios.
- 🌐 El procesamiento aislado es un principio fundamental de la **gestión de la configuración**.
- 🌐 Nuestro CI/CD es una fábrica y el aislamiento garantiza que la línea de montaje sea **idéntica** siempre.

5. Resumen

- 🌐 El problema del entorno exige aislamiento → **hermeticidad**
- 🌐 El aislamiento se basa en **inmutabilidad y reproducibilidad**
- 🌐 Las tecnologías son herramientas para implementar estos principios.
- 🌐 El procesamiento aislado es un principio fundamental de la **gestión de la configuración**.
- 🌐 Nuestro CI/CD es una fábrica y el aislamiento garantiza que la línea de montaje sea **idéntica** siempre.



5. Resumen

“DEVOPS IS RUINING MY CRAFT”

BY ALEX TATIYANTS    

February 22nd, 2012

Below is an excerpt from an interview with a disenfranchised systems administrator. All identifies have been kept anonymous to protect the innocent:

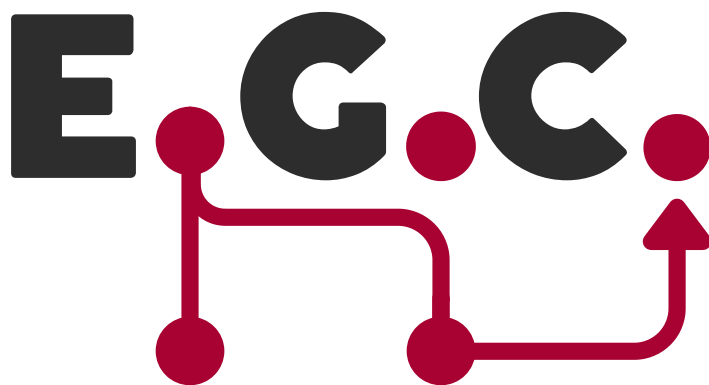
“I come from a long line of sysadmins. My father was a sysadmin, as was his father before him. I have apprenticed at the feet of some of the greatest sysadmins American BankCorp has ever seen.

Some say that I just build servers. Well, I respectfully disagree. Did Stadivarius just make violins? Did Henry Winston just make jewelry? Did Louis Cartier just make watches? Nay, these men made art, first and foremost.

I too consider myself an artist and a craftsman of server building. With each click of a mouse, I create a work of art. With every option I select, every module I install, every registry tweak I make, every configuration file I edit, I create a unique, one of a kind masterpiece.

I have taken years to develop and refine my craft, culminating in what I consider to be my greatest

<http://tatiyants.com/devops-is-ruining-my-craft/>



Grado en Ingeniería Informática – Ingeniería del Software

Evolución y Gestión de la Configuración



Escuela Técnica Superior de
Ingeniería Informática

¡Gracias!