A Strategy for Container Lifecycle Management

Federico Aguirre, Alfredo Edye, Edgardo Hames

September 7, 2017

46JAIIO Jornadas Argentinas de Informática

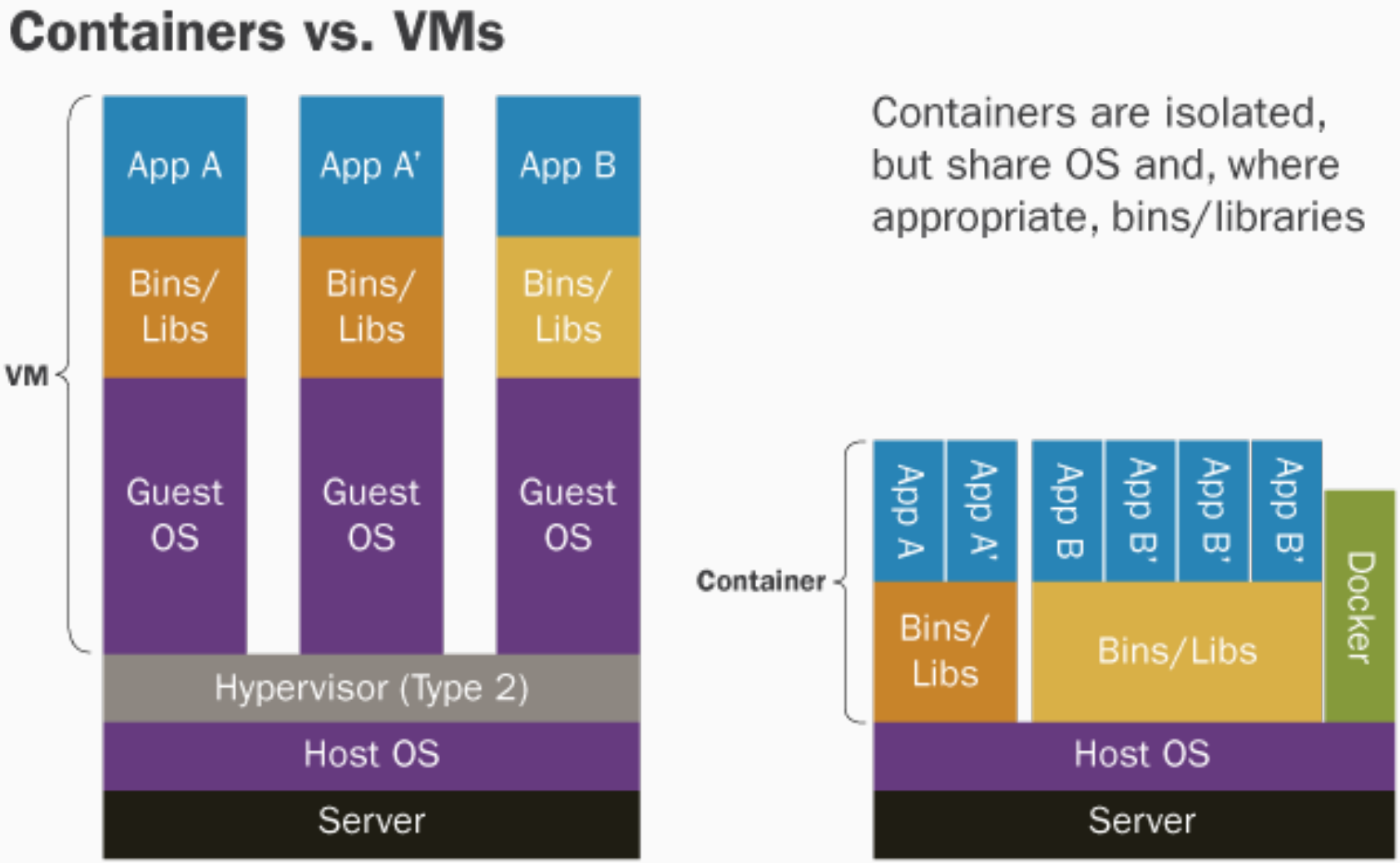
Table of contents

Introducción

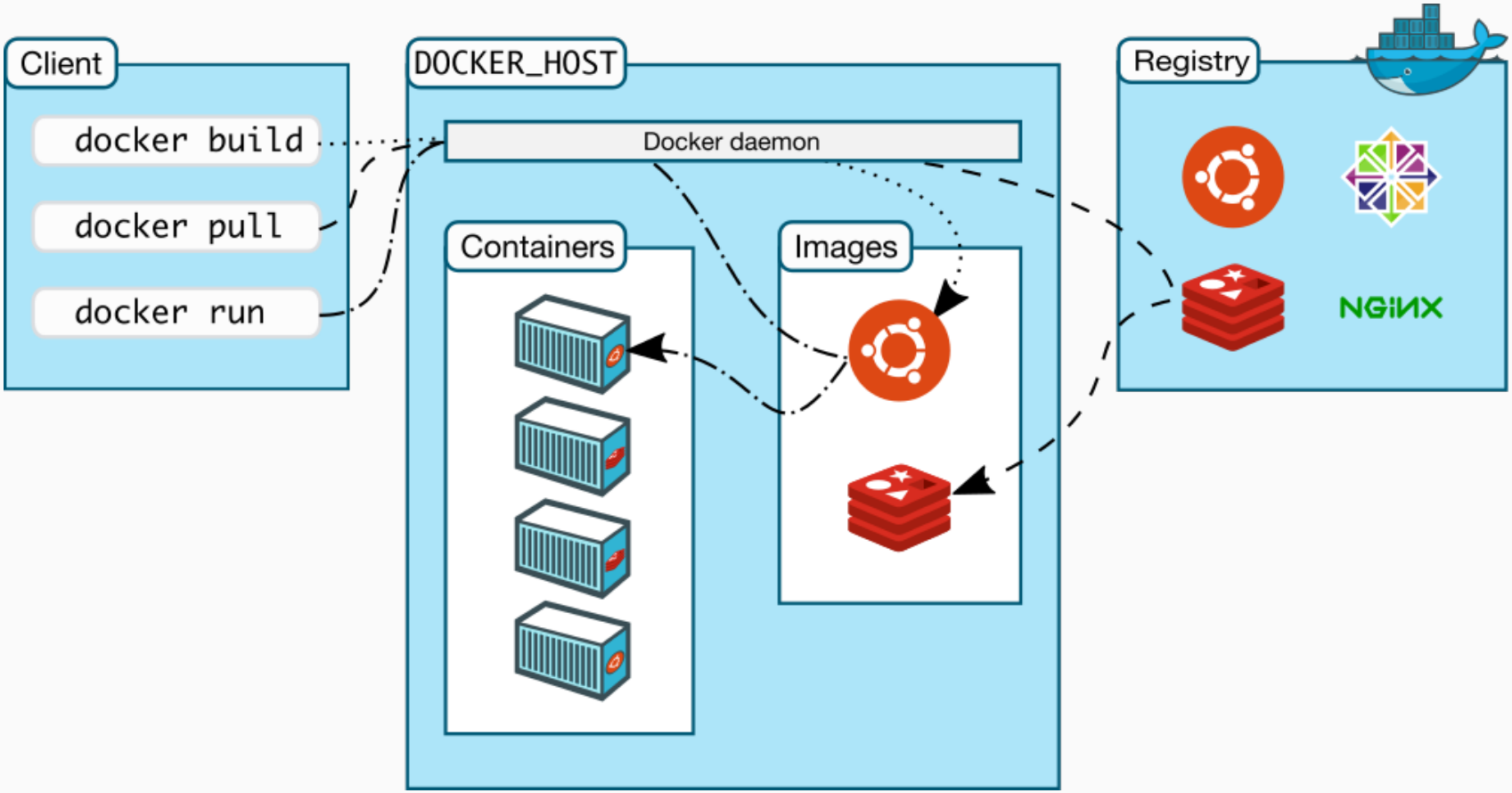
Reseña histórica

1956	Memoria Virtual, Fritz-Rudolf Güntsch
1959	Tiempo Compartido, John McCarthy
1964	Hypervisor (14 VMs concurrentes), IBM
1979	*chroot*, Unix
2000	Virtualización OS, Virtuozzo
2000	*jails*, FreeBSD
2004	*zones*, Solaris
2008	*namespaces*, Linux
2013	Docker

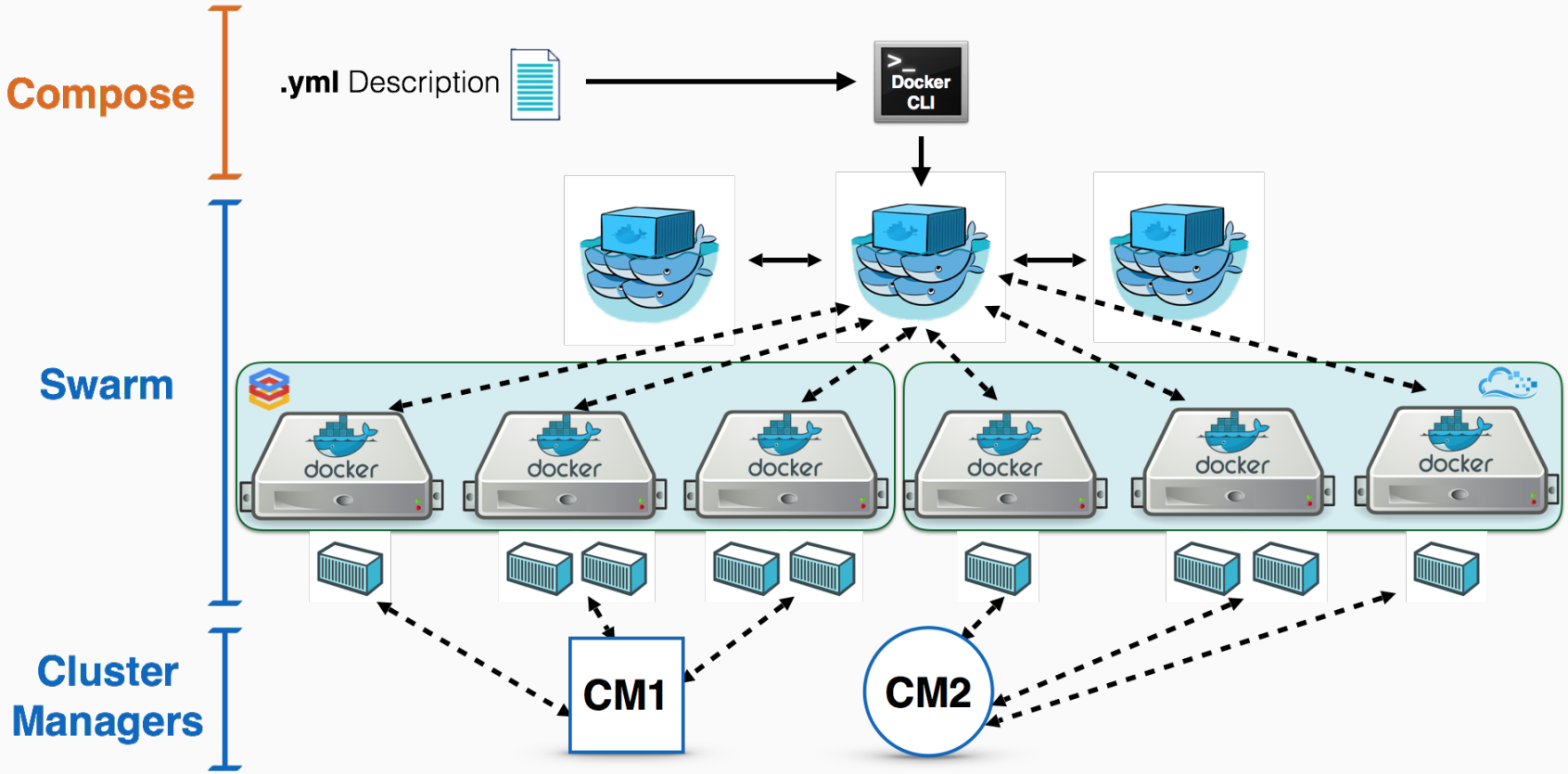
Virtualización



Arquitectura de Docker



Compose & Swarm



12-Factor Apps

Presentación

- Uso de **formatos declarativos** para configuraciones

Presentación

- Uso de **formatos declarativos** para configuraciones
- Existencia de un **contrato claro** entre las aplicaciones y el sistema operativo para mayor portabilidad

Presentación

- Uso de **formatos declarativos** para configuraciones
- Existencia de un **contrato claro** entre las aplicaciones y el sistema operativo para mayor portabilidad
- **Simplificación de despliegue** en plataformas *cloud*

Presentación

- Uso de **formatos declarativos** para configuraciones
- Existencia de un **contrato claro** entre las aplicaciones y el sistema operativo para mayor portabilidad
- **Simplificación de despliegue** en plataformas *cloud*
- **Integración continua** para mayor agilidad

Presentación

- Uso de **formatos declarativos** para configuraciones
- Existencia de un **contrato claro** entre las aplicaciones y el sistema operativo para mayor portabilidad
- **Simplificación de despliegue** en plataformas *cloud*
- **Integración continua** para mayor agilidad
- **Escalabilidad** sin cambios significativos en la arquitectura, herramientas y prácticas de desarrollo

Procesos administrativos

El código de administración y despliegue debe

- entregarse junto con el de la aplicación para evitar inconsistencias

Procesos administrativos

El código de administración y despliegue debe

- entregarse junto con el de la aplicación para evitar inconsistencias
- ser autocontenido y no depender de la existencias de herramientas o bibliotecas en el sistema

Problema

¿Cómo distribuimos los scripts de mantenimiento y sus configuraciones?

Bootstrap

Bootstrap

Un nuevo contenedor denominado *Bootstrap* encargado de

- distribución de scripts de mantenimiento y configuraciones

Bootstrap

Un nuevo contenedor denominado *Bootstrap* encargado de

- distribución de scripts de mantenimiento y configuraciones
- descarga de imágenes del sistema

Bootstrap

Un nuevo contenedor denominado *Bootstrap* encargado de

- distribución de scripts de mantenimiento y configuraciones
- descarga de imágenes del sistema
- gestión del ciclo de vida (inicio, detención, estado, etc)

Bootstrap

Un nuevo contenedor denominado *Bootstrap* encargado de

- distribución de scripts de mantenimiento y configuraciones
- descarga de imágenes del sistema
- gestión del ciclo de vida (inicio, detención, estado, etc)
- tareas de mantenimiento (*upgrade*, *downgrade*, migración, etc)

Uso

Para descargar las imágenes de los servicios

```
$ docker run --rm bootstrap pull
```

Para desplegar los contenedores

```
$ docker run --rm bootstrap up
```

Para detener los contenedores

```
$ docker run --rm bootstrap stop
```

Implementación (script)

```
#!/bin/bash
command=$1
case "$command" in
  "pull")
     docker-compose pull
     ;;
  "up")
     docker-compose up
     ;;
  "stop")
     # TODO: Pedir al usuario que confirme
     docker-compose stop
     ;;
esac
```

Implementación (Dockerfile)

```
FROM docker/compose:1.12.0
COPY docker-compose.yml /opt/bitlogic
COPY startup.sh /opt/bitlogic
RUN chmod 755 /opt/bitlogic/startup.sh
ENTRYPOINT /opt/bitlogic/startup.sh
```

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker
- Puede usarse con cualquier motor de integración y despliegue continuos

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker
- Puede usarse con cualquier motor de integración y despliegue continuos
- Facilita el despliegue del producto en el entorno de desarrollo

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker
- Puede usarse con cualquier motor de integración y despliegue continuos
- Facilita el despliegue del producto en el entorno de desarrollo
- Permite desplegar remotamente usando la API TCP de Docker

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker
- Puede usarse con cualquier motor de integración y despliegue continuos
- Facilita el despliegue del producto en el entorno de desarrollo
- Permite desplegar remotamente usando la API TCP de Docker
- Garantiza la compatibilidad con cada versión del producto

Ventajas

- Aplica a cualquier tipo de proyecto que use Docker
- Puede usarse con cualquier motor de integración y despliegue continuos
- Facilita el despliegue del producto en el entorno de desarrollo
- Permite desplegar remotamente usando la API TCP de Docker
- Garantiza la compatibilidad con cada versión del producto
- Es agnóstico de la plataforma subyacente (Swarm, Kubernetes, etc)

Comparación con Puppet, Chef, Salt, Ansible, etc

- Tiene un propósito muy específico

Comparación con Puppet, Chef, Salt, Ansible, etc

- Tiene un propósito muy específico
- No requiere agentes adicionales (solo Docker)

Comparación con Puppet, Chef, Salt, Ansible, etc

- Tiene un propósito muy específico
- No requiere agentes adicionales (solo Docker)
- Puede ser implementado por el equipo de desarrollo del producto

Conclusión

Resumen

- Los contenedores permiten consolidar la infraestructura.

Resumen

- Los contenedores permiten consolidar la infraestructura.
- Docker ha bajado la barrera de acceso a contenedores.

Resumen

- Los contenedores permiten consolidar la infraestructura.
- Docker ha bajado la barrera de acceso a contenedores.
- Bootstrap resuelve el problema de la distribución de configuraciones.

Resumen

- Los contenedores permiten consolidar la infraestructura.
- Docker ha bajado la barrera de acceso a contenedores.
- Bootstrap resuelve el problema de la distribución de configuraciones.
- Bootstrap permite unificar la interfaz de operación.

¿Preguntas?