

☁ Configuration Management – Cheatsheet

Curso: Configuration Management and the Cloud
Módulo 1: Automating with Configuration Management

📖 Introducción

- **Configuration Management (CM):** práctica de automatizar la configuración, despliegue y mantenimiento de sistemas de manera consistente.
- **Objetivo:** reducir errores humanos, ahorrar tiempo y asegurar consistencia en entornos grandes.

💠 Conceptos Clave

- **Idempotencia:** aplicar configuraciones repetidamente sin efectos adversos
- **Declarative vs Imperative:**
 - Declarative: defines el estado deseado y la herramienta lo alcanza
 - Imperative: defines paso a paso cómo alcanzar el estado
- **Infraestructura como Código (IaC):** definir configuración y despliegue mediante código en lugar de procesos manuales

💠 Herramientas Comunes

Herramienta	Uso
Puppet	Gestión de configuraciones automatizada
Ansible	Automatización y despliegue de tareas
Chef	Automatización de infraestructura
Terraform	IaC y provisión de recursos en la nube

💠 Buenas Prácticas

- Mantener código de configuración versionado en repositorios (Git)
- Usar **modularidad** y **roles** para separar configuraciones por función
- Probar cambios en entornos de desarrollo antes de producción
- Documentar configuraciones y dependencias


💠 Ejemplo Básico de Declarative CM

```
# Instala y asegura que Apache esté activo
package { 'httpd':
  ensure => installed,
}

service { 'httpd':
  ensure => running,
  enable => true,
}
```

- La herramienta verifica el estado y aplica solo los cambios necesarios
 - Evita errores al ejecutar scripts manuales repetidamente
-

Referencias

-  [Puppet Docs – Getting Started](#)
-  [Ansible Docs – Introduction](#)
-  [Idempotency Concept](#)