

DevOPS as Code

Carlos González Peñalba
Gerente de Arquitectura Tecnológica

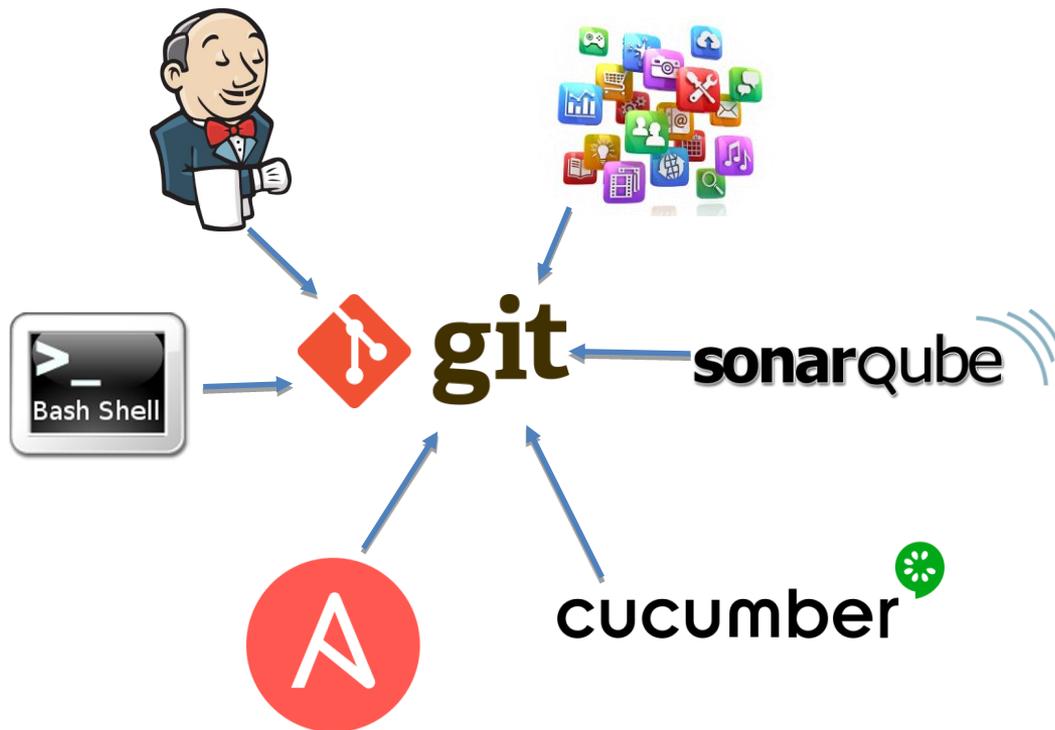
Moisés Rivera
Cloud Lead Red Hat Iberia

OPEN
24/7

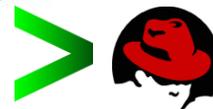
¿Quieres entrar?

¿Qué es DevOps-as-Code?

DevOPS Best Practices



Implica el uso de un repositorio SCM para toda la lógica de la construcción automatizada de aplicaciones, pruebas, despliegues y gestión de la infraestructura.



¿Qué ganamos con ello?



@my-company ▾

These are all projects from My Company Inc

🔔 Global ▾

Projects Subgroups

Filter by name...

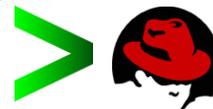
Last updated ▾

New Project

	web-front Website from My Company. Angular powered.	🔒
	jenkins Jenkins files to handle logic of Pipelines as Code	🔒
	scripts Script used for testing and build application.	🔒
	ansible-site Ansible playbooks for configuration management and automated deploys	🔒

Estandarización

- La lógica de los componentes se almacena como código fuente en el repositorio SCM.
- Todos los cambios se hacen en un único sitio.
- Uso de building-blocks dinámicos para gobernar la lógica de las construcciones.
- Generación automatizada de construcciones (p.ej: microservicios) de forma rápida y sencilla.

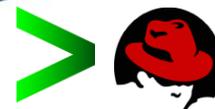


Seguridad



Michał Golebiowski	cddeb7c	Build: Updating the 2.1-stable version to 2.1.4-pre.	18 Dec 2014
Michał Golebiowski	8f2a9d9	2.1.3	18 Dec 2014
Michał Golebiowski	998556f	Build: Smoke test on Node with jsdom	11 Sep 2014
Michał Golebiowski	685fae9	Ajax: Don't use bare location, it breaks Node with jsdom Tests not included; the proper fix is coming to the m...	17 Dec 2014
Timmy Willison	6bc0e50	Build: Sizzle version labels must start with a dash	17 Dec 2014
Timmy Willison	d6c97ab	Build: Sizzle version labels must start with a dash	17 Dec 2014
Dave Methvin	e8ef663	Build: Updating the 1.11-stable version to 1.11.3-pre.	17 Dec 2014
Timmy Willison	dda65fb	Build: update node dependencies	17 Dec 2014
Timmy Willison	ac70dd0	Build: account for version labels in Sizzle versions Fixes gh-1939	17 Dec 2014
Timmy Willison	9101704	Build: update node dependencies	17 Dec 2014
Timmy Willison	78ac753	Build: account for version labels in Sizzle versions Fixes gh-1939	17 Dec 2014
Dave Methvin	9690801	1.11.2	17 Dec 2014
Dave Methvin	ce9869f	Build: Update commit/please to 2.0.0	17 Dec 2014
Dave Methvin	56fb758	Build: update commit/please to 2.0.0	17 Dec 2014
Dave Methvin	5352fc8	Build: Updating the 2.1-stable version to 2.1.3-pre.	17 Dec 2014
Dave Methvin	a04f5ff	2.1.2	17 Dec 2014
Dave Methvin	0b4c81e	Revert "Release: ensure Sizzle is the latest tag" This reverts commit 523c87138afc23acd7d0817504d254bd...	16 Dec 2014
Dave Methvin	37da931	Revert "Release: ensure Sizzle is the latest tag" This reverts commit 1d931068d30b6a516b883cf7f420606de...	16 Dec 2014

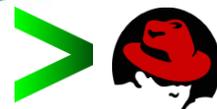
- Trazabilidad absoluta sobre los cambios que se han realizado.
- Usando un gestor de repositorios colaborativo, como GitLab, GitHub, Bitbucket, etc, podemos ver los cambios fácilmente.
- El uso de frameworks y building-blocks es la industrialización de los propios procedimientos DevOPS, por lo que se reducen las manualidades, y se minimiza el error humano.



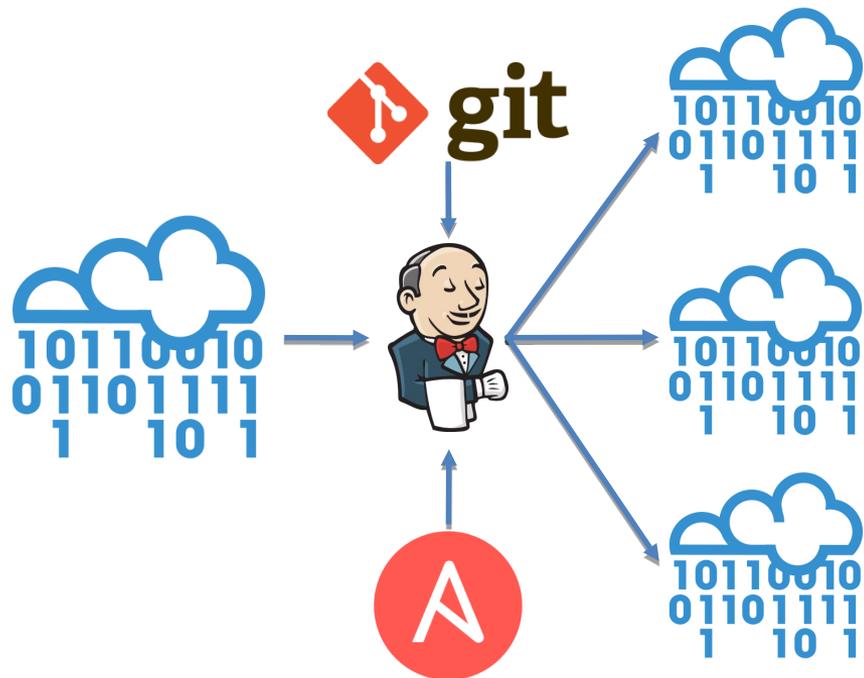


Quick-Recovery

- La adopción de métodos IaC (infraestructura como código) habilita la rápida recuperación de los sistemas en caso de desastre.
- La lógica de las construcciones está en el repositorio SCM, por lo que no existe pérdida de datos en caso de desastre con los servidores CI.
- Al estar versionados todos los cambios, se puede volver a una versión anterior al instante en caso de necesidad.



Entornos bajo demanda



- Los entornos de aplicaciones se pueden replicar de forma rápida y sencilla en la nube.
- Al estar toda la lógica de creación, configuración y construcción en un repositorio, se puede utilizar para crear tantos entornos de pruebas como se necesite.
- Facilita la generación de “sandboxes” para que los desarrolladores puedan realizar pruebas de su trabajo sin afectar a los entornos previos compartidos.



¿Cómo se hace?



Pipelines como Código

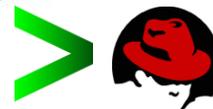
The screenshot shows the Jenkins web interface. At the top, there's a navigation bar with 'Jenkins', 'Open Blue Ocean', and a search bar. Below the navigation bar, there's a breadcrumb trail: 'Jenkins > MyCompany > MyCompany-Front'. On the left side, there's a sidebar with various actions like 'Up', 'Status', 'Configure', 'Scan Multibranch Pipeline Now', etc. The main content area shows the configuration for 'MyCompany-Front' with a 'Branches (4)' table.

S	W	Name ↓	Último Éxito
		develop	47 Min - #13
		feature/ajustes-en-test-automatizados	48 Min - #12
		feature/pruebas-de-color	34 Min - #4
		master	1 Hor 19 Min - #4

Icono: [S](#) [M](#) [L](#) [Guía de iconos](#)

Trabajos en la cola: No hay trabajos en la cola

- Utilizando Jenkins, se realizan las construcciones mediante los “MultiBranch Pipelines”
- Se vinculan a un repositorio GIT, del que escaneará de forma automática las ramas creadas, y generará un job por cada una.
- Se puede configurar para que se lance automáticamente el job de construcción ante cambios en el repositorio.
- Con un solo JOB en Jenkins, se puede utilizar estrategias diferentes para cada rama ó tipo de ramas.





Infraestructura como Código

The screenshot shows the Ansible Tower web interface. At the top, there are navigation tabs: TOWER, PROJECTS, INVENTORIES, TEMPLATES, and JOBS. The 'JOBS' tab is selected. Below the navigation, there are buttons for 'JOBS' and 'SCHEDULES', and a search bar. A table lists several jobs, all named 'MyCompany Front Deploy'. The table has columns for ID, NAME, and a timestamp. A modal window is open over the table, showing the details of a job. The modal has a 'DETAILS' tab selected. It shows the job's status as 'Successful', the start and end times, the template used, the job type, the inventory, the project, the revision, the playbook, the machine, the credentials, the hosts, the verbosity, and the extra variables. The 'PLAYBOOK' section shows the following content:

```
Identity address (/tmp/ansible_tower_inventory/(/tmp/ansible_tower_inventory/
PLAY [webserver] .....
TASK [include] .....
TASK [deploy_mycompany_front : Include environment vars] .....
ok: [webserver-mycompany.com] .....
TASK [deploy_mycompany_front : Clean remote temp directory if exists] .....
```

- Utilizando Ansible Tower, se puede centralizar toda la lógica de despliegues automatizados de aplicaciones.
- Jenkins sirve de orquestador para realizar los despliegues, apoyándose en Tower.
- Mediante Ansible Tower, se pueden provisionar de forma automática instancias en AWS, OpenStack, Azure, etc.
- Se puede tener toda la infraestructura codificada en un “SITE” de Ansible, de forma que con un solo click se pueda regenerar por completo.





Demo Funcional



DEMO



Manual vs Automatizado



Construcción, pruebas y empaquetado manual

El entregable se envía a sistemas para su implantación.

El responsable verifica que el entorno de pruebas funciona bien

Desarrollo reempaqueta el entregable de producción

El entregable se envía a sistemas para su implantación.

T-0

1 día

5 días

10 días



sonarqube

cucumber



APACHE HTTP SERVER

T-0

5 min

10 min

15 min



DevOps as Code



accenturetechnology

Carlos González Peñalba

Email: c.gonzalez.penalba@accenture.com

Linkedin: [linkedin.com/in/carlos-gonzalez-penalba/](https://www.linkedin.com/in/carlos-gonzalez-penalba/)



Moisés Rivera

Email: moy@redhat.com

<https://www.linkedin.com/in/mois-es-rivera/>