

Meetup DevOps CI/CD

Primeros pasos con Jenkins y Kubernetes



Jenkins



kubernetes

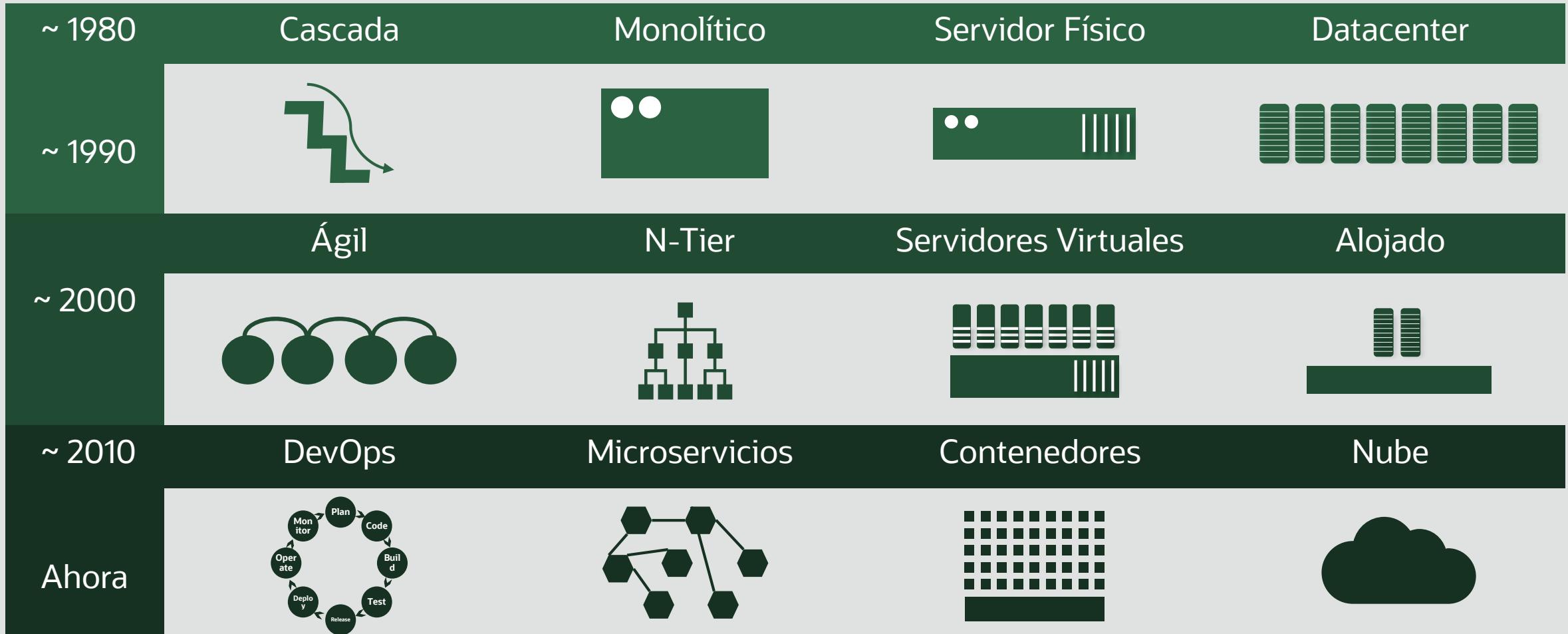
Programa de la agenda

- 1 Conceptos y Siglas
- 2 Softwares Open Source
- 3 Aplicación Monolítica y Microservicios
- 4 Cloud Native Computing Foundation
- 5 Demo: Jenkins y Kubernetes

Programa de la agenda

- 1 Conceptos y Siglas**
- 2 Softwares Open Source
- 3 Aplicación Monolítica y Microservicios
- 4 Cloud Native Computing Foundation
- 5 Demo: Jenkins y Kubernetes

La tecnología ha llegado muy lejos...



¿Lo que efectivamente las empresas buscan?



Los conceptos de moda y algunas siglas

GPL

Gnu Program License

Open Source

Private Cloud

DevSecOps

CI / CD

Continuos Integration / Continuous Delivery

ITIL

SDN

Software-Defined Network

IOT

Internet of Thing

Lean

SCRUM

AR

Augmented Reality

Public Cloud

Hybrid Cloud

Multi Cloud

Agile

Containers

VR

Virtual Reality

CNCF

Cloud Native Computing Foundation

DevOps

SRE

Site Reliability Engineering

Microservices

Blockchain

Serverless

Transformación Digital

SDS

Software-Defined Storage

NFV

Network Function Virtualization

Machine Learning

AI

Artificial Inteligence

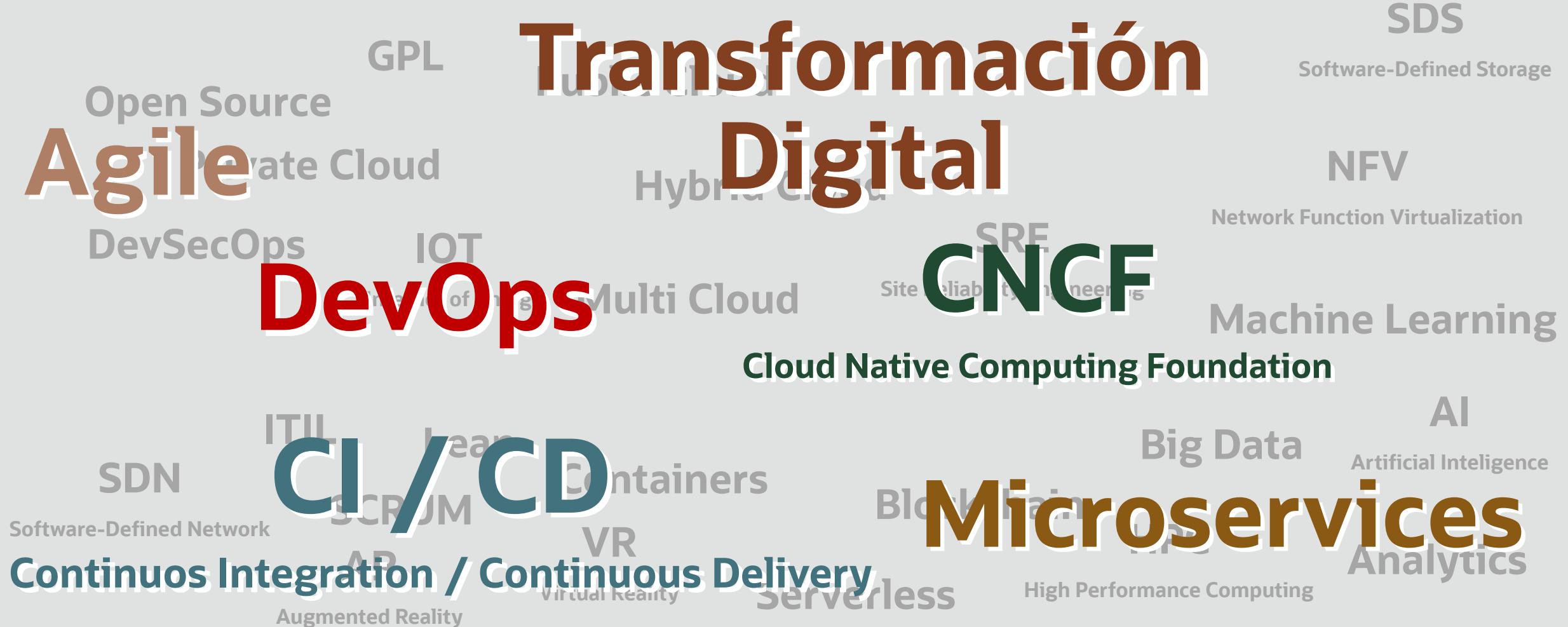
Big Data

Analytics

HPC

High Performance Computing

Los conceptos de moda y algunas siglas



Transformación Digital

Transformación Digital es el proceso de exploración de las tecnologías digitales y soporte de las capacidades para crear un modelo de negocios digital nuevo y robusto.

-Gartner

El concepto de Agile

Agile, ahora conocido por algunos de sus autores de manifiestos como agilidad, se centra en eliminar las barreras del proceso y permitir que las partes interesadas clave, personas como desarrolladores y clientes, colaboren más estrechamente para acelerar la entrega. **Agile** destaca la constancia del cambio y reconoce que, como productores de software, a menudo no sabemos todo lo que necesitamos para concebir, desarrollar y entregar con éxito software de alta calidad en ciclos de vida monolíticos.

Entonces, aunque **Agile** ha llegado a significar cosas diferentes en las últimas dos décadas, sus fundamentos permanecen: eliminar las barreras de proceso que empoderan a las personas, producir software de trabajo rápidamente, colaborar estrechamente con los clientes y responder (en lugar de resistirse) al cambio.



Manifiesto **Agile**

Seguimos estos principios:

- Nuestra máxima prioridad es satisfacer al cliente a través de la entrega temprana y continua de software valioso.
- La gente de negocios y los desarrolladores deben trabajar juntos a diario durante todo el proyecto.
- El software de trabajo es la medida principal del progreso.
- Los procesos ágiles promueven el desarrollo sostenible.
- El patrocinador, los desarrolladores y los usuarios deberían poder mantener un ritmo constante indefinidamente.

DevOps es la práctica de los ingenieros de operaciones y desarrollo que participan juntos en todo el ciclo de vida del servicio, desde el diseño hasta el proceso de desarrollo y el soporte de producción.

-The Agile Admin blog



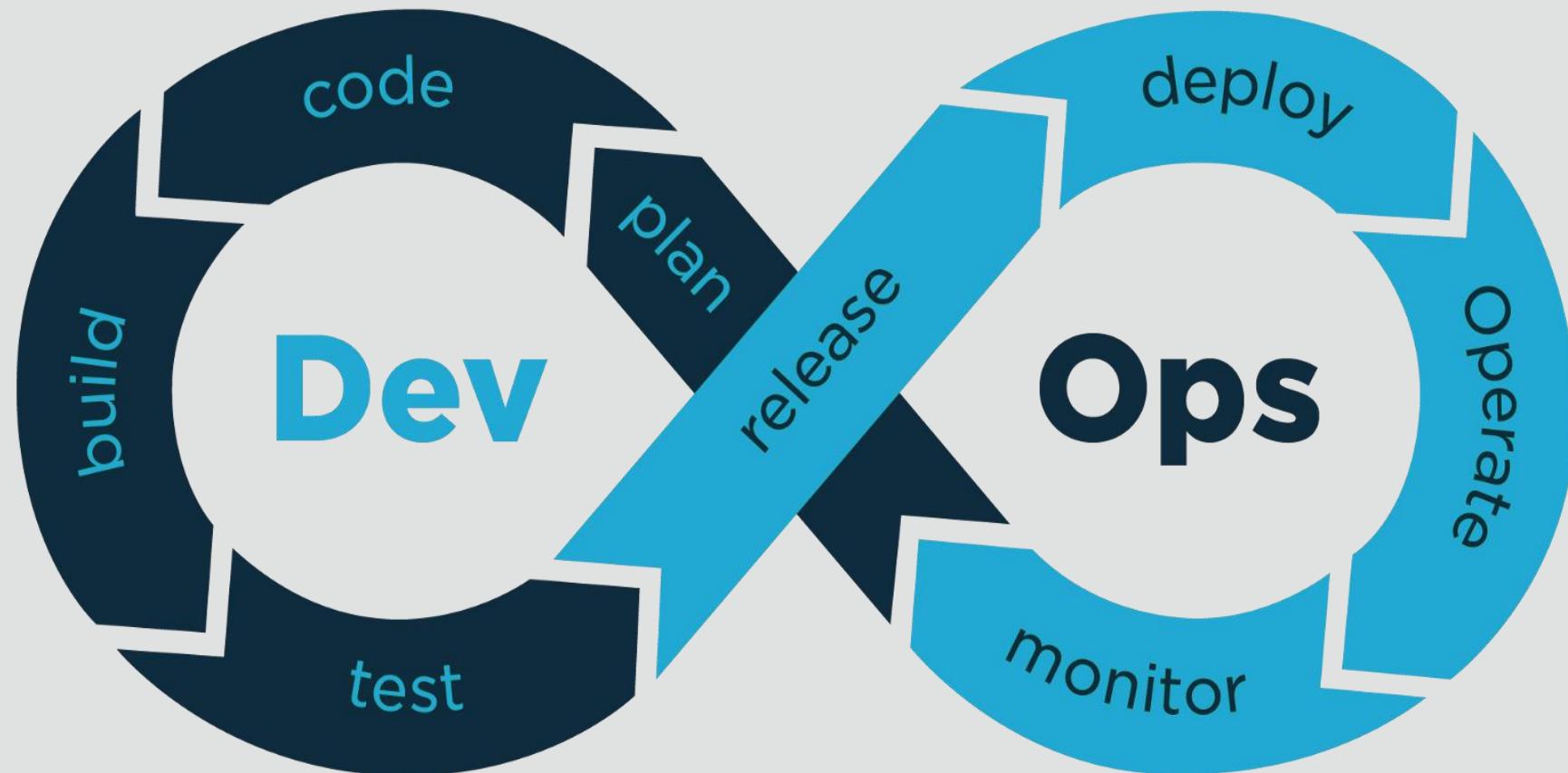
- Generalmente basado en principios de CALMS
- Basado en las ideas de Automatización, Medición, Compartir
- Énfasis en una cultura colaborativa en las organizaciones.
- Operaciones de desplazamiento más a la izquierda
 - Los equipos de operaciones hacen más que solo "administración de servidores"
 - Utiliza las mismas técnicas que los desarrolladores para el trabajo de sistemas.

CALMS

CALMS es un marco conceptual para la integración de equipos, funciones y sistemas de desarrollo y operaciones (DevOps) dentro de una organización. El **marco CALMS** a menudo se usa como un **modelo de madurez**, lo que **ayuda a los gerentes a evaluar si su organización está lista o no para DevOps y, si no, qué necesita cambiar**. El acrónimo CALMS está acreditado a Jez Humble, coautor del "Manual de DevOps".

Culture
Automation
Lean
Measurement
Sharing

DevOps Logo



Fundamentos DevOps

Primera
Metodología

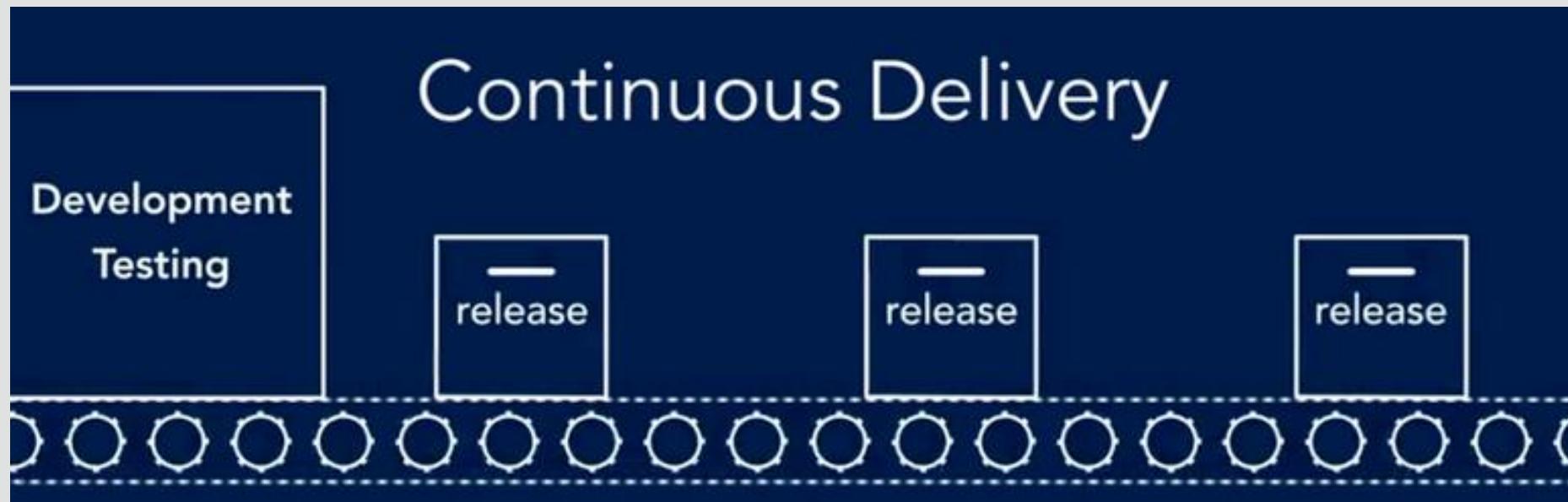
Personas

Procesos

Herramientas

Fundamentos DevOps

Segunda Metodologia



Fundamentos DevOps

Tercera Metodología

Gestión eficiente

Trabajar en lotes pequeños.

Límites de trabajo en progreso.

Circuitos de retroalimentacion.

Visualización.

Fundamentos DevOps

Cuarta Metodología

Control de cambio

Eliminar artefactos frágiles.

Crear un proceso de compilación repetible.

Administrar dependencias.

Crear un entorno de mejora continua.

Fundamentos DevOps

Quinta Metodología

Infraestructura como Código

Sistemas tratados como código.

Comprobado en control de fuente.

Revisado, construido y probado.

Gestionado programáticamente.

Fundamentos DevOps

- 1 Personas sobre proceso sobre herramientas**
- 2 Entrega continua**
- 3 Gestión eficiente**
- 4 Control de cambio de operaciones visibles**
- 5 Infraestructura como código**

**WORKED FINE IN
DEV**

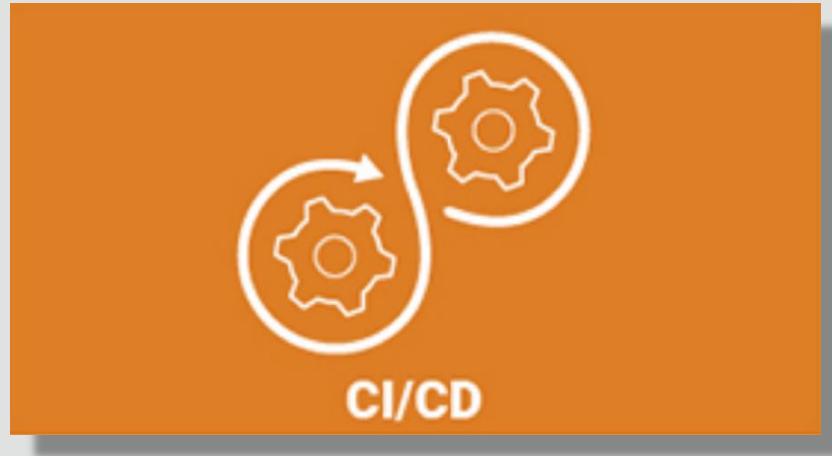
OPS PROBLEM NOW

memegenerator.net

CI/CD

(Continuous Integration / Continuous Delivery)

Orígenes en el segmento de automatización de DevOps.



Integración Continua

- El commit del código es frecuente en el control de código fuente.
- Ejecuta compilaciones automatizadas en cada commit de código.
- Resultado = Detección de errores más rápido

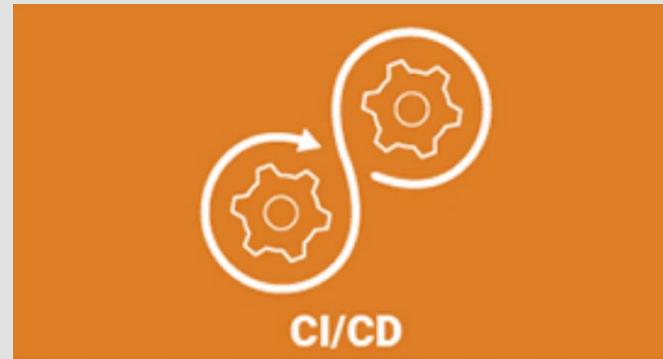
Entrega Continua

- Paso 1: Integración Continua
- Liberar compilaciones de código para usuarios finales.
- Resultado = Enviar software más rápido a los usuarios finales.

Diferencia entre Agile, CI/CD y DevOps



Agile



CI/CD



DevOps

se centra en
procesos

resaltando
cambio

mientras acelera
entrega

se centra en
ciclos de vida definidos por software

resaltando
herramientas

que enfatizan
automatización

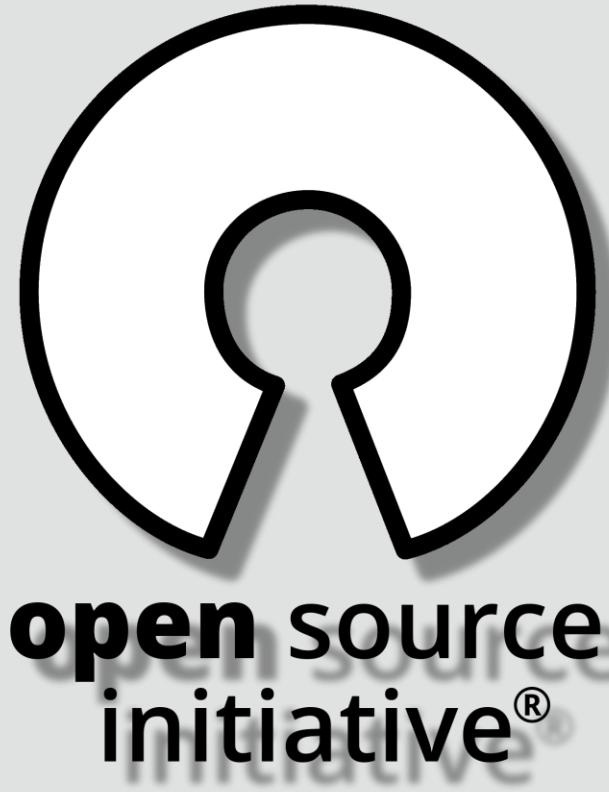
se centra en
cultura

resaltando
roles

que enfatizan
sensibilidad

Programa de la agenda

- 1 Conceptos y Siglas
- 2 **Softwares Open Source**
- 3 Aplicación Monolítica y Microservicios
- 4 Cloud Native Computing Foundation
- 5 Demo: Jenkins y Kubernetes



El **código abierto** es un modelo de desarrollo de software basado en la colaboración abierta. Se enfoca más en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre. Para muchos el término “libre” hace referencia al hecho de adquirir un software de manera gratuita.



La **GNU General Public License** es una licencia de derecho de autor ampliamente usada en el mundo del software libre y código abierto.

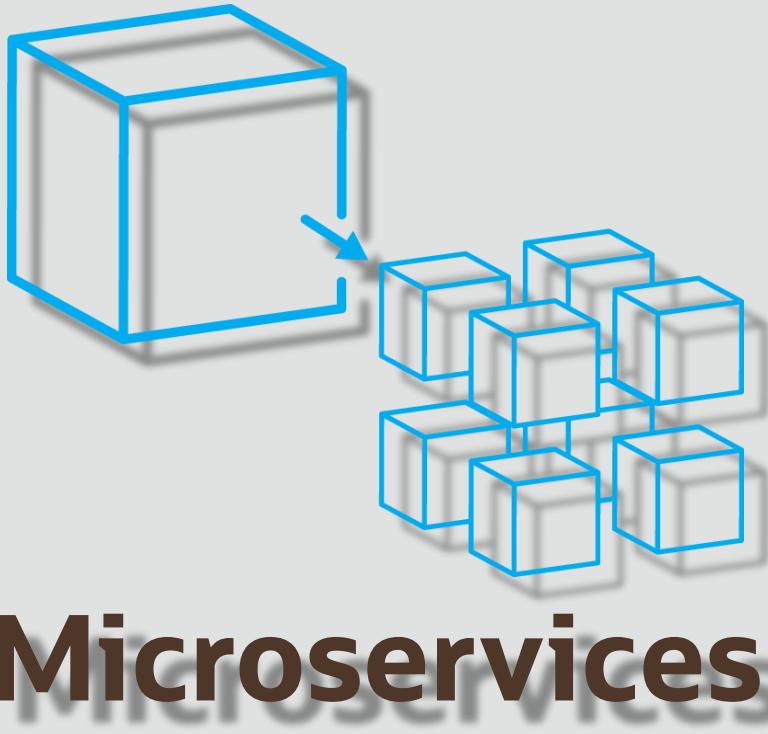
Software Open Source, algunos pocos ejemplos

Linux	PostgreSQL	Kata	C++	Prometheus
KDE	Git	Kubernetes	Jenkins	MongoDB
OpenStack	Go	Docker	LXC	Fluentd
Gnome	Vagrant	Python	MySQL	KVM
FreelIPA	OpenLDAP	Terraform	Containerd	Grafana
Firefox	Keepalive	Postfix	Rundek	XEN
Ruby	HAProxy	Wireshark	Elastic	oVirt
	Corosync	Ansible	JavaSC	Thunderbird
	Gimp	Sendmail	Virtualbox	Spacewalk
	LibreOffice	Ceph	Gluster	

Software Open Source, algunos pocos ejemplos

Programa de la agenda

- 1 Conceptos y Siglas
- 2 Softwares Open Source
- 3 **Aplicación Monolítica y Microservicios**
- 4 Cloud Native Computing Foundation
- 5 Demo: Jenkins y Kubernetes

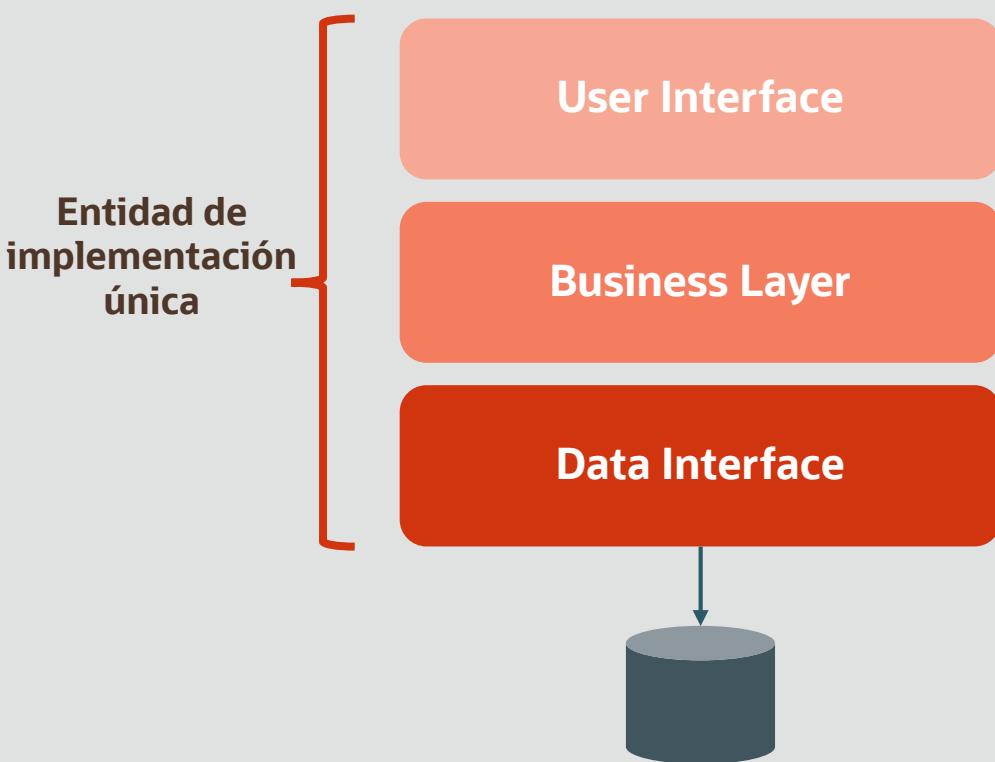


En una arquitectura de **microservicios**, cada microservicio es propietario de una tarea simple y se comunica con los clientes o con otros microservicios mediante mecanismos de comunicación sencillos, como las solicitudes de la API de REST.

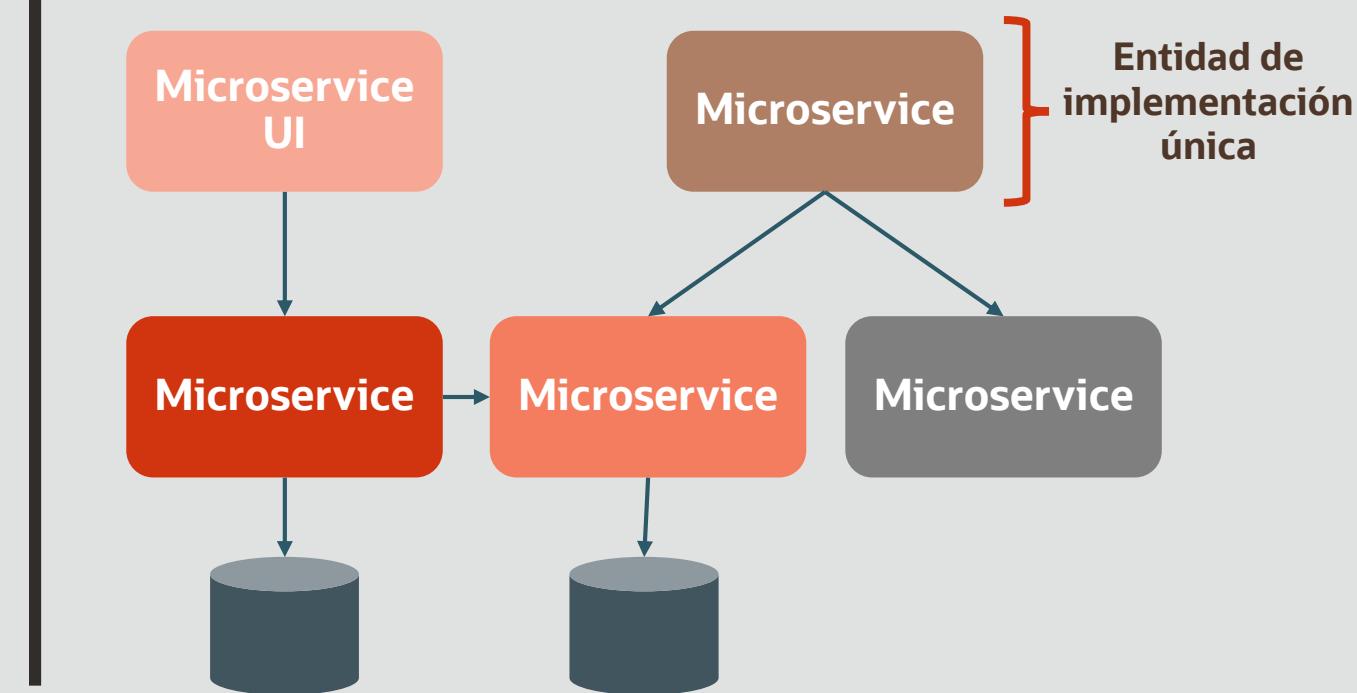
-Karthik Gaekwad

Comparación de arquitectura de aplicaciones y microservicios

Arquitectura monolítica tradicional



Arquitectura de microservicios



Servicios independientes con comunicación utilizando API REST / HTTP

Componentes principales que permiten microservicios

DevOps / CI / CD		
Service Registry	API Management	Data Storage
State	Polyglot	Configuration
Serverless	Messaging/Eventing	Logging
Security	Monitoring	Scaling
Docker / Container Infrastructure		

Fundamentalmente, los microservicios son una compensación

Quieres...

Desarrollo de App tradicionales Microservicios

Waterfall Lifecycle

- Entregado como monolito, menos partes móviles, más fácil probar
- La implementación / prueba se centra en un componente grande
- Más difícil de escalar
- Menos ágil

Complejidad pero flexible

- Entregado como componentes individuales
- La implementación / prueba se centra en muchos servicios y en diferentes combinaciones de versiones
- Fácil de escalar
- Más ágil

Casos de uso comunes de adopción de microservicios



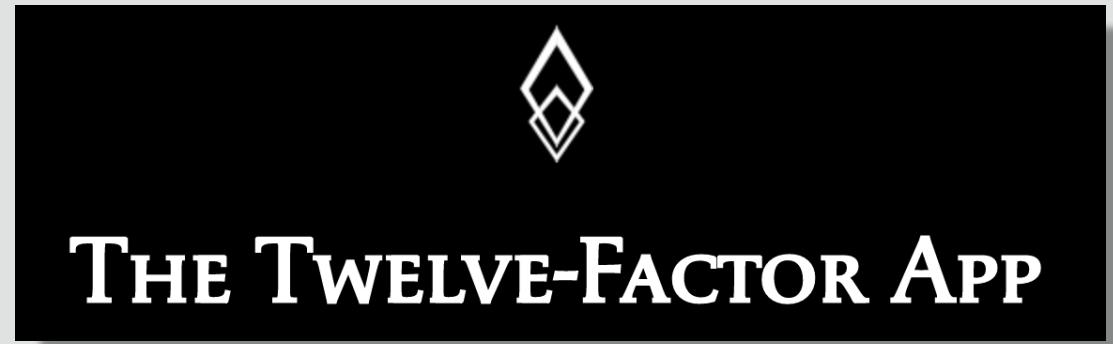
Quiero ampliar mi aplicación monolítica existente agregando microservicios en la periferia.

Quiero construir una
**nueva
aplicación** de estilo
microservicios desde cero.

Quiero
descomponer una
aplicación modular existente en
una aplicación de estilo
microservicios

En estos tiempos, el software se está distribuyendo como un servicio: se le denomina web apps, o software as a service (SaaS).

“**The twelve-factor app**” es una metodología para construir aplicaciones SaaS.



TWELVE FACTORS

I. Código base (Codebase)

Un código base sobre el que hacer el control de versiones y multiples despliegues

II. Dependencias

Declarar y aislar explícitamente las dependencias

III. Configuraciones

Guardar la configuración en el entorno

IV. Backing services

Tratar a los “Backing services” como recursos conectables

V. Construir, desplegar, ejecutar

Separar completamente la etapa de construcción de la etapa de ejecución

VI. Procesos

Ejecutar la aplicación como uno o más procesos sin estado

VII. Asignación de puertos

Publicar servicios mediante asignación de puertos

VIII. Concurrencia

Escalar mediante el modelo de procesos

IX. Desechabilidad

Hacer el sistema más robusto intentando conseguir inicios rápidos y finalizaciones seguras

X. Paridad en desarrollo y producción

Mantener desarrollo, preproducción y producción tan parecidos como sea posible

XI. Historiales

Tratar los historiales como una transmisión de eventos

XII. Administración de procesos

Ejecutar las tareas de gestión/administración como procesos que solo se ejecutan una vez

Programa de la agenda

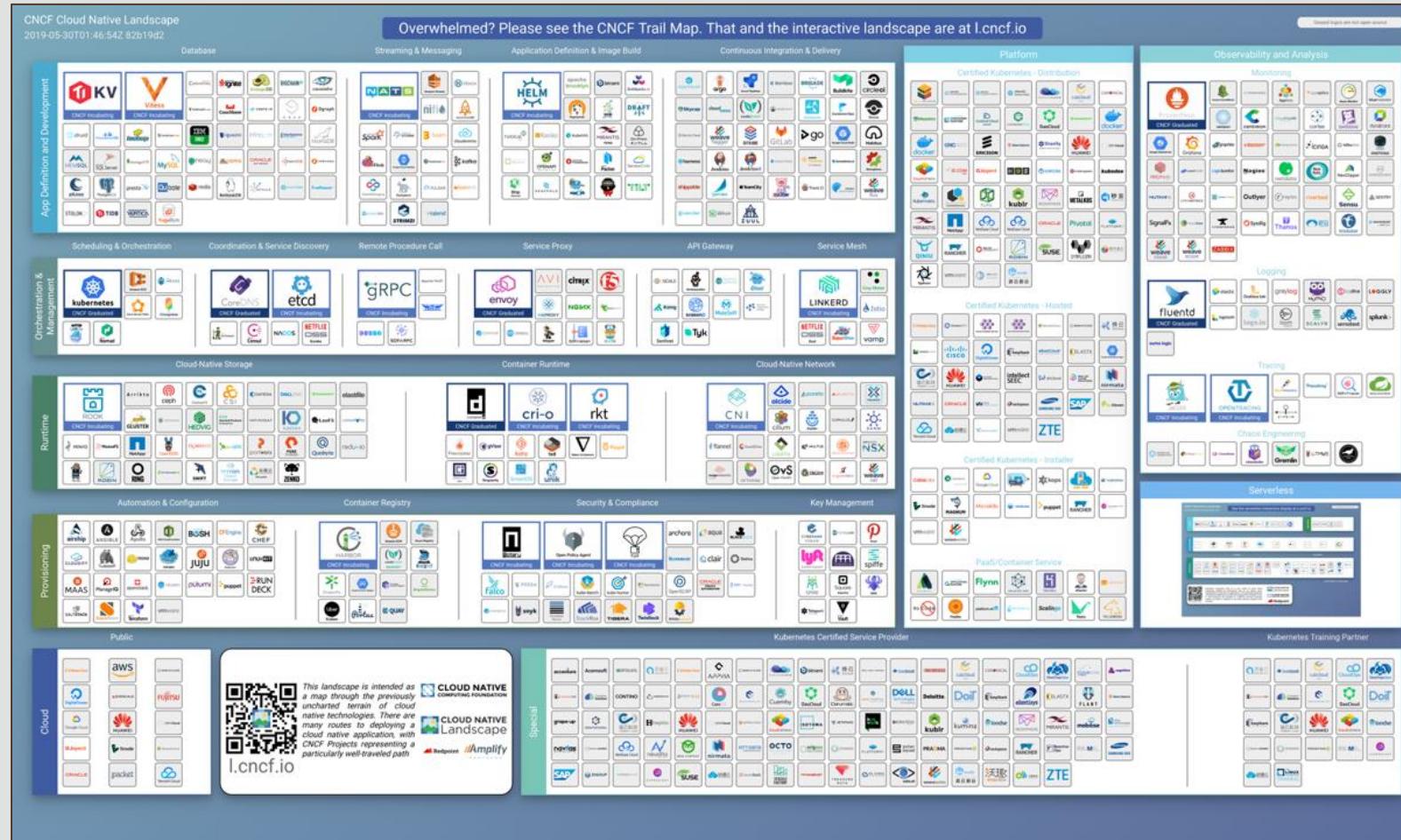
- 1 Conceptos y Siglas
- 2 Softwares Open Source
- 3 Aplicación Monolítica y Microservicios
- 4 **Cloud Native Computing Foundation**
- 5 Demo: Jenkins y Kubernetes



La **Cloud Native Computing Foundation** utiliza una pila de software de código abierto para implementar aplicaciones como microservicios, empacando cada parte en su propio contenedor y orquestando dinámicamente esos contenedores para optimizar la utilización de los recursos, además de generar estándares para el Mercado.

-CNCF

Un panorama tecnológico complicado, en rápida evolución



El Mapa del Sendero hacia Cloud Native

- Más fácil de digerir
- Código abierto, pero necesita asegurarse de que las herramientas funcionen en su propio ecosistema
- Todo bricolaje (DIY)
- Desafíos de nubes múltiples/nube híbrida





kubernetes

Kubernetes es un sistema de código libre para la automatización del despliegue, ajuste de escala y manejo de aplicaciones en contenedores que fue originalmente diseñado por Google y donado a la Cloud Native Computing Foundation. Soporta diferentes entornos para la ejecución de contenedores, incluido Docker.

-Wikipedia



kubernetes

Pods

Replica Sets

Services

Volumes

Namespace

ConfigMaps and Secrets



Prometheus

Prometheus es una aplicación de software de código abierto que se utiliza para la supervisión y alerta de eventos. Registra métricas en tiempo real en una base de datos de series de tiempo construida utilizando un modelo de extracción HTTP, con consultas flexibles y alertas en tiempo real.

-*Wikipedia*



Grafana

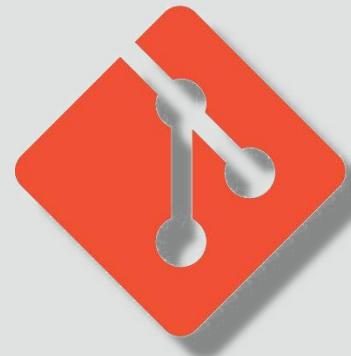
Grafana Grafana es una suite de visualización y análisis de métricas de código abierto. Se usa más comúnmente para visualizar datos de series temporales para análisis de infraestructura y aplicaciones, pero muchos lo usan en otros dominios, incluidos sensores industriales, domótica, clima y control de procesos.

-*Grafana..com*



Jenkins es un servidor de automatización open source escrito en Java. Está basado en el proyecto Hudson y es, dependiendo de la visión, un fork del proyecto o simplemente un cambio de nombre.

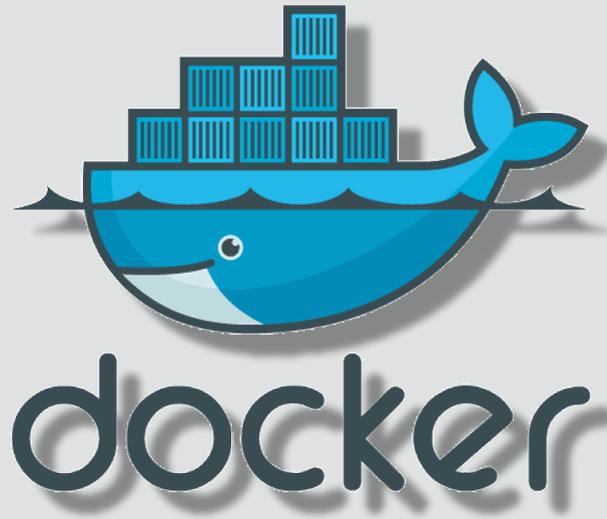
-Wikipedia



git

Git es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código Fuente.

-*Wikipedia*



Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.

-Wikipedia



VirtualBox

Oracle VM VirtualBox es un software de virtualización para arquitecturas x86/amd64. Actualmente es desarrollado por Oracle Corporation como parte de su familia de productos de virtualización.

-*Wikipedia*



Vagrant es una herramienta para la creación y configuración de entornos de desarrollo virtualizados. Originalmente se desarrolló para VirtualBox y sistemas de configuración tales como Chef, Salt y Puppet.

-Wikipedia



Oracle Linux es el sistema operativo Linux de Oracle diseñado para la nube abierta. Está disponible gratuitamente bajo el Licencia Pública General de GNU (GPLv2). Es 100% compatible con Red Hat e incluye las herramientas estandares de CNCF (Kubernetes, Docker, Ceph, Gluster ...).

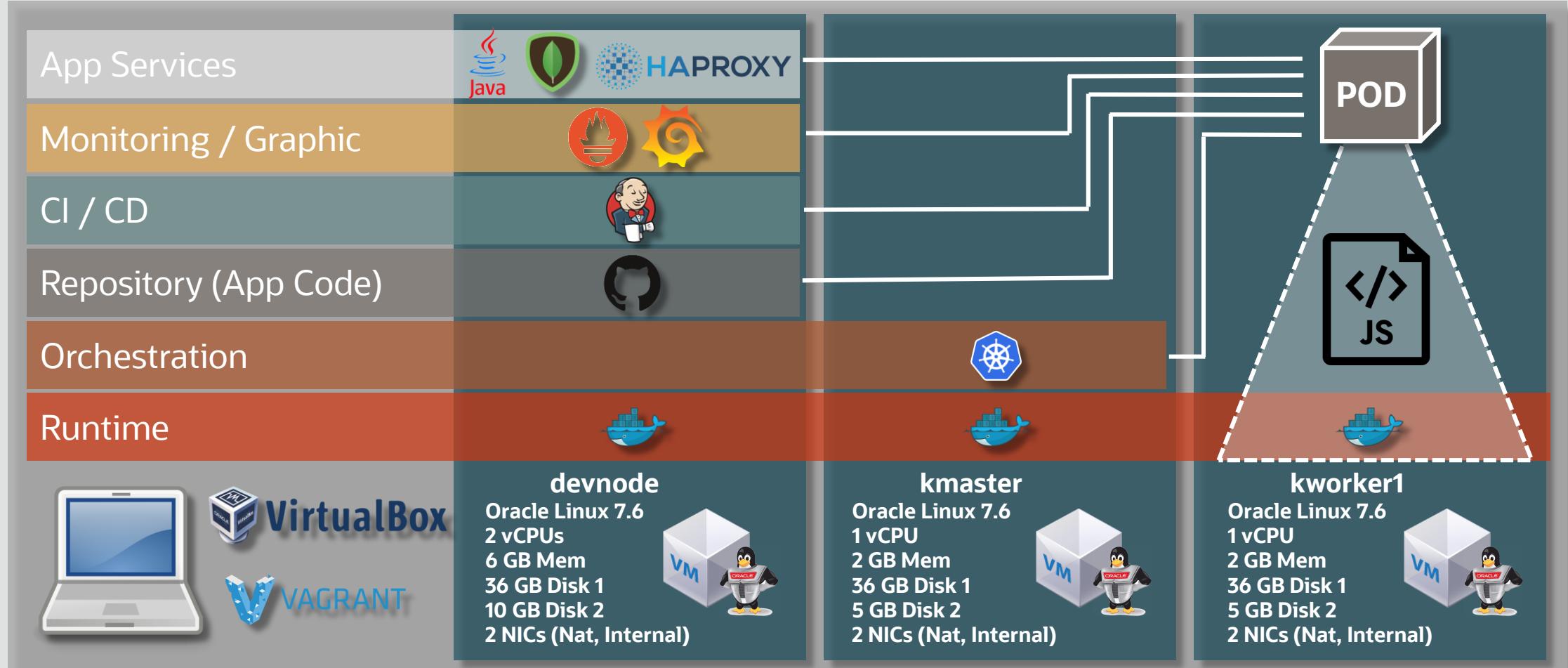
-Oracle

**La demo a la continuación está construida
100% con software libre ocupando como
Sistema Operativo base el Oracle Linux,
disponible para download gratis con
repositorios publicos yum.oracle.com**

Programa de la agenda

- 1 Conceptos y Acrónimos
- 2 Softwares Open Source
- 3 Aplicación Monolítica y Microservicios
- 4 Cloud Natives Computing Foundation
- 5 **Demo: Jenkins y Kubernetes**

Arquitectura de la demo



[1] "All other trademarks are the property of their respective owners."

Jenkins – Primeros pasos

Getting Started

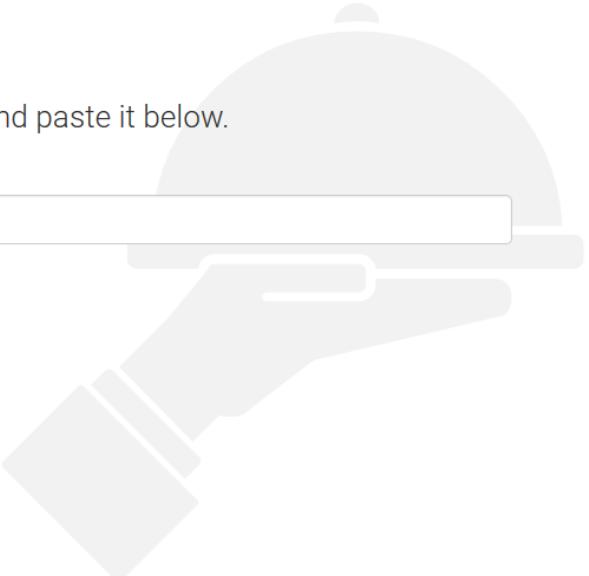
Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/home/demo/.jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password



Continue

Jenkins – Primeros pasos

Getting Started

Customize Jenkins

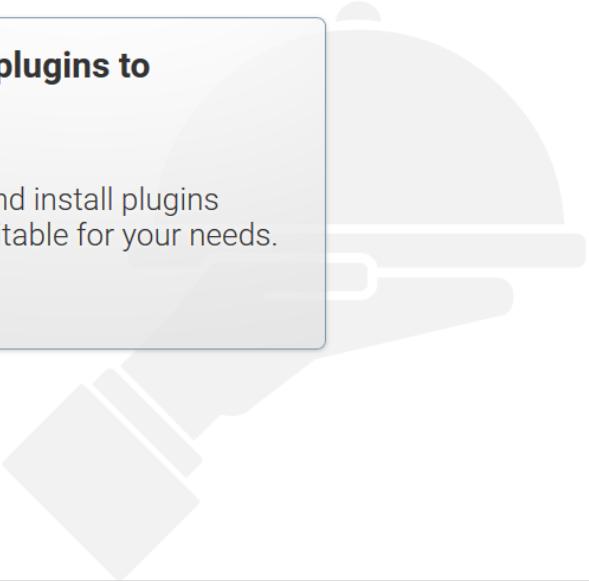
Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.



Jenkins 2.176.3

Jenkins – Primeros pasos

Getting Started

Getting Started

✓ Folders	OWASP Markup Formatter	Build Timeout	Credentials Binding	** JAXB Folders ** Oracle Java SE Development Kit Installer
Timestamper	Workspace Cleanup	Ant	Gradle	
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline: Stage View	
Git	Subversion	SSH Slaves	Matrix Authorization Strategy	
○ PAM Authentication	LDAP	Email Extension	Mailer	

** – required dependency

Jenkins 2.176.3

Jenkins – Primeros pasos

Getting Started

Create First Admin User

Username:

Password:

Confirm password:

Full name:

E-mail address:

Jenkins 2.176.3

Continue as admin

Save and Continue

Jenkins – Primeros pasos

The screenshot shows the Jenkins 'Instance Configuration' page under the 'Getting Started' tab. The main title is 'Instance Configuration'. A 'Jenkins URL' input field contains the value 'http://devnode:4000/'. Below it, a descriptive text explains that the Jenkins URL is used for absolute links to various resources and is required for proper operation of features like email notifications and PR status updates. It also notes that the value is 'not saved yet' and should be set to the URL users expect. At the bottom, there are 'Not now' and 'Save and Finish' buttons.

Getting Started

Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.176.3

Not now Save and Finish

Jenkins – Primeros pasos

The screenshot shows the Jenkins 'Getting Started' page. At the top left, there is a 'Getting Started' link. The main content area features a large heading 'Jenkins is ready!' followed by the subtext 'Your Jenkins setup is complete.' Below this, there is a blue button labeled 'Start using Jenkins'. At the bottom of the page, the text 'Jenkins 2.176.3' is displayed.

Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

Jenkins 2.176.3

Jenkins – Primeros pasos

The screenshot shows the Jenkins dashboard with the following elements:

- Header:** Jenkins logo, search bar, user name "Andre da Silva", and log out link.
- Left Sidebar:** Navigation menu with links: New Item, People, Build History, Manage Jenkins, My Views, Credentials, Lockable Resources, and New View.
- Middle Content:**
 - Welcome to Jenkins!** Main heading.
 - A message: "Please [create new jobs](#) to get started."
 - Build Queue:** Sub-section showing "No builds in the queue."
 - Build Executor Status:** Sub-section showing "1 Idle" and "2 Idle".
- Page Footer:** Page generated: Sep 27, 2019, 1:35:30 PM UTC, REST API, Jenkins ver. 2.176.3.

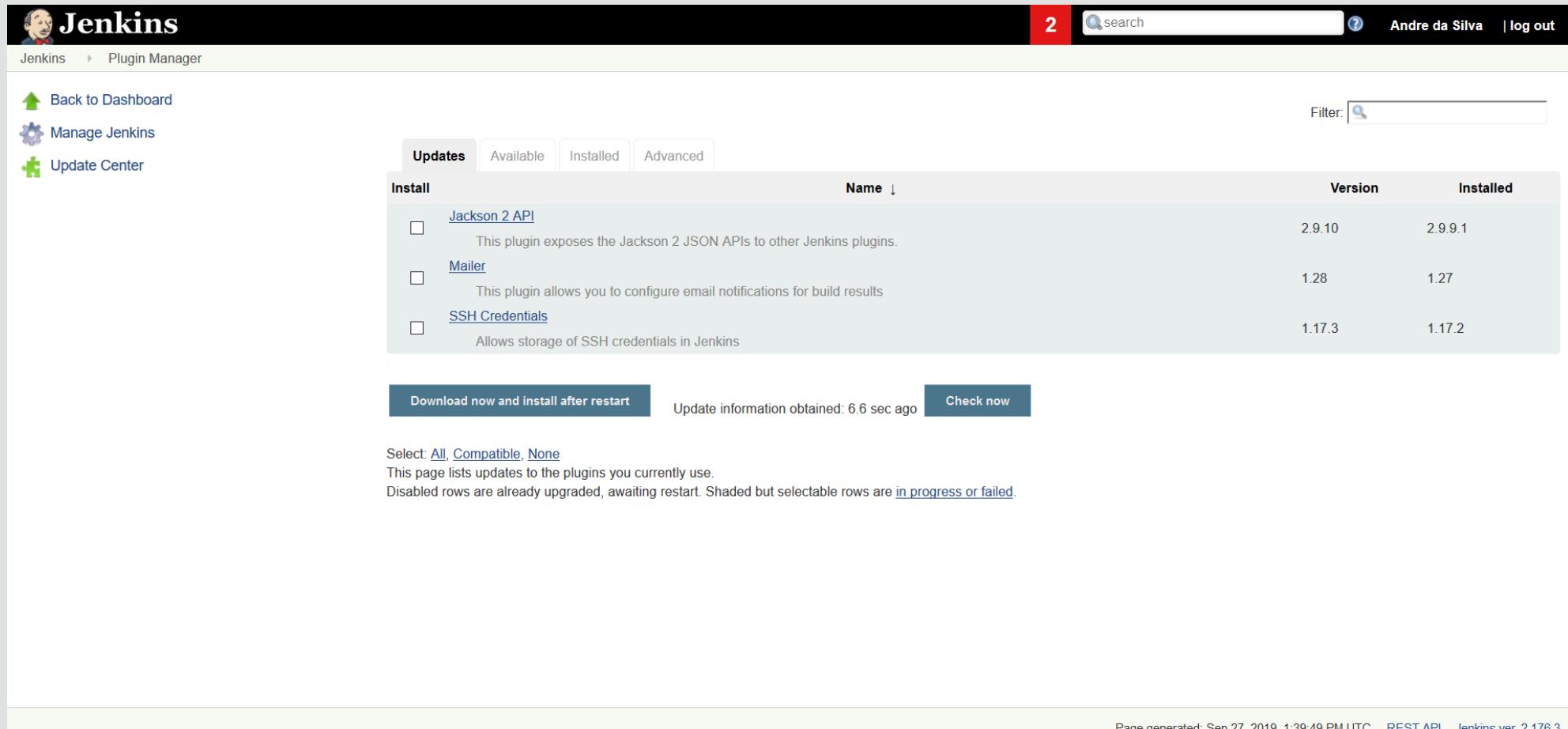
Jenkins – Primeros pasos

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links like New Item, People, Build History, Manage Jenkins (which is selected), My Views, Credentials, Lockable Resources, and New View. Below this are two collapsed sections: Build Queue (No builds in the queue) and Build Executor Status (1 Idle, 2 Idle). The main content area is titled "Manage Jenkins" and lists several configuration options:

- Configure System**: Configure global settings and paths.
- Configure Global Security**: Secure Jenkins; define who is allowed to access/use the system.
- Configure Credentials**: Configure the credential providers and types.
- Global Tool Configuration**: Configure tools, their locations and automatic installers.
- Reload Configuration from Disk**: Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.
- Manage Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins. A "Manage Plugins" button is shown below the link.
- System Information**: Displays various environmental information to assist trouble-shooting.
- System Log**: System log captures output from `java.util.logging` output related to Jenkins.

At the bottom left, it says "devnode:4000/pluginManager". The top right shows the user "Andre da Silva" and a "log out" link, along with a "search" bar and a "ENABLE AUTO REFRESH" checkbox.

Jenkins – Primeros pasos



The screenshot shows the Jenkins Plugin Manager interface. At the top, there is a navigation bar with the Jenkins logo, a search bar, and a user account for "Andre da Silva". A red box highlights the number "2" in the top right corner, indicating available updates. The main content area has a sidebar with links to Back to Dashboard, Manage Jenkins, and Update Center. The "Updates" tab is selected, showing a list of available plugins:

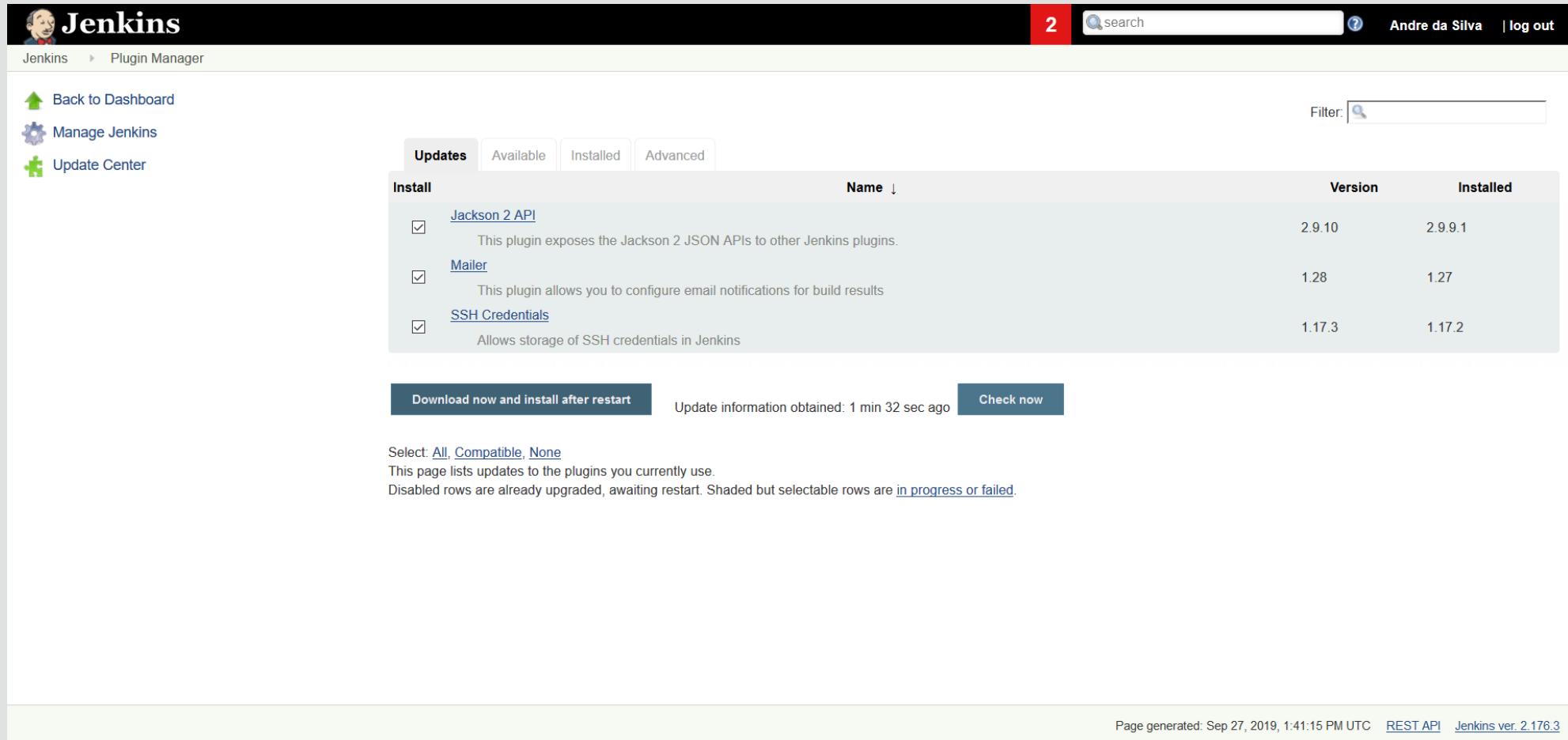
Install	Name	Version	Installed
<input type="checkbox"/>	Jackson 2 API This plugin exposes the Jackson 2 JSON APIs to other Jenkins plugins.	2.9.10	2.9.9.1
<input type="checkbox"/>	Mailer This plugin allows you to configure email notifications for build results	1.28	1.27
<input type="checkbox"/>	SSH Credentials Allows storage of SSH credentials in Jenkins	1.17.3	1.17.2

At the bottom of the list, there are buttons for "Download now and install after restart" and "Check now". Below the table, there is a note about selecting filter options: "Select: All, Compatible, None". It also states that the page lists updates for currently used plugins and that disabled rows are awaiting restart.

Jenkins – Primeros pasos

The screenshot shows the Jenkins Update Center interface. At the top, there is a navigation bar with the Jenkins logo, a search bar, and user information for "Andre da Silva". A red notification badge with the number "2" is visible. Below the navigation bar, the main content area has a title "Installing Plugins/Upc" and a sub-section "Preparation". On the left, there is a sidebar with links: "Back to Dashboard", "Manage Jenkins", and "Manage Plugins". The main content area lists several components under "Preparation": "JAXB", "Folders", "Oracle Java SE Development Kit Installer", and "Script Security". A message at the top right indicates a new version (2.190.1) is available for download or upgrade. A warning message below states: "Warnings have been published for the following currently installed components." It specifically mentions "Jenkins 2.176.3 core and libraries" and links to a changelog for "Multiple security vulnerabilities in Jenkins 2.196 and earlier, and LTS 2.176.3 and earlier". A "Configure which of these warnings are shown" link is also present. At the bottom right, there is a "Manage Jenkins" link and a "Success" status indicator.

Jenkins – Primeros pasos



The screenshot shows the Jenkins Plugin Manager interface. At the top, there is a navigation bar with the Jenkins logo, a search bar, and a user account for "Andre da Silva". A red notification badge with the number "2" is visible. Below the navigation bar, the page title is "Plugin Manager". On the left, there is a sidebar with links: "Back to Dashboard", "Manage Jenkins", and "Update Center". The main content area has tabs for "Updates", "Available", "Installed", and "Advanced", with "Updates" selected. A table lists the available updates:

Install	Name ↓	Version	Installed
<input checked="" type="checkbox"/>	Jackson 2 API This plugin exposes the Jackson 2 JSON APIs to other Jenkins plugins.	2.9.10	2.9.9.1
<input checked="" type="checkbox"/>	Mailer This plugin allows you to configure email notifications for build results	1.28	1.27
<input checked="" type="checkbox"/>	SSH Credentials Allows storage of SSH credentials in Jenkins	1.17.3	1.17.2

At the bottom of the table, there are two buttons: "Download now and install after restart" and "Check now". A message indicates that update information was obtained 1 minute 32 seconds ago. Below the table, there is a note about selecting plugin compatibility levels: "Select: All, Compatible, None". It also states that the page lists updates for currently used plugins and that disabled rows are awaiting restart.

Jenkins – Primeros pasos

Mailer	● Success
Mailer	● Downloaded Successfully. Will be activated during the next boot
Jackson 2 API	● Downloaded Successfully. Will be activated during the next boot
SSH Credentials	● Downloaded Successfully. Will be activated during the next boot

 [Go back to the top page](#)
(you can start using the installed plugins right away)

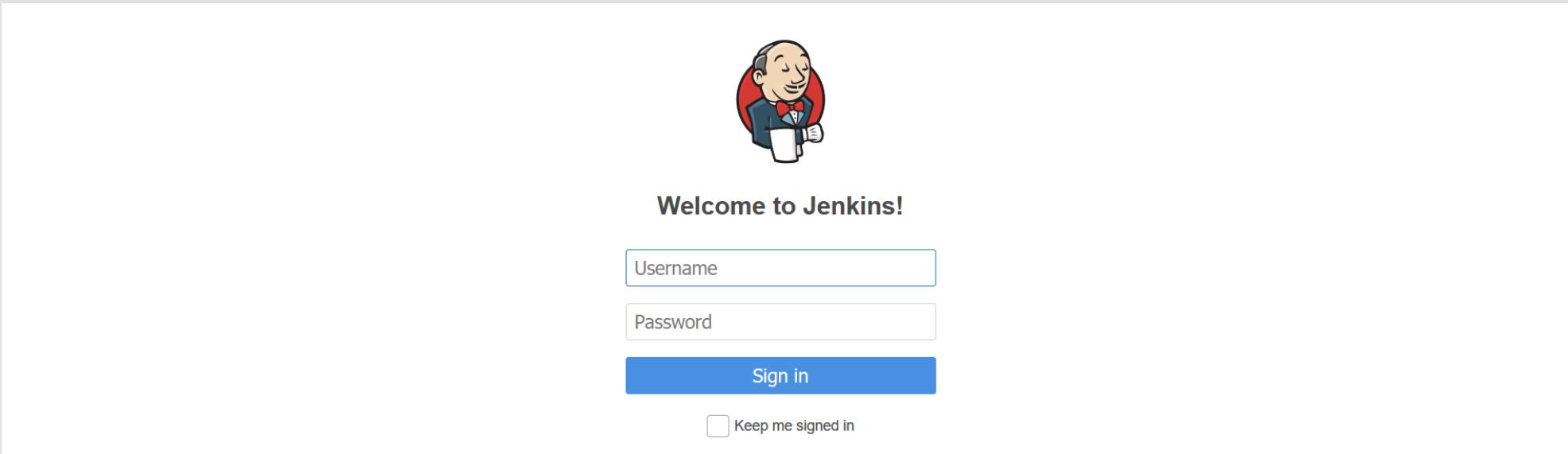
 Restart Jenkins when installation is complete and no jobs are running

Mailer	● Success
Mailer	● Downloaded Successfully. Will be activated during the next boot
Jackson 2 API	● Downloaded Successfully. Will be activated during the next boot
SSH Credentials	● Downloaded Successfully. Will be activated during the next boot
Restarting Jenkins	● Running

 [Go back to the top page](#)
(you can start using the installed plugins right away)

 Restart Jenkins when installation is complete and no jobs are running

Jenkins – Primeros pasos



Jenkins – Primeros pasos

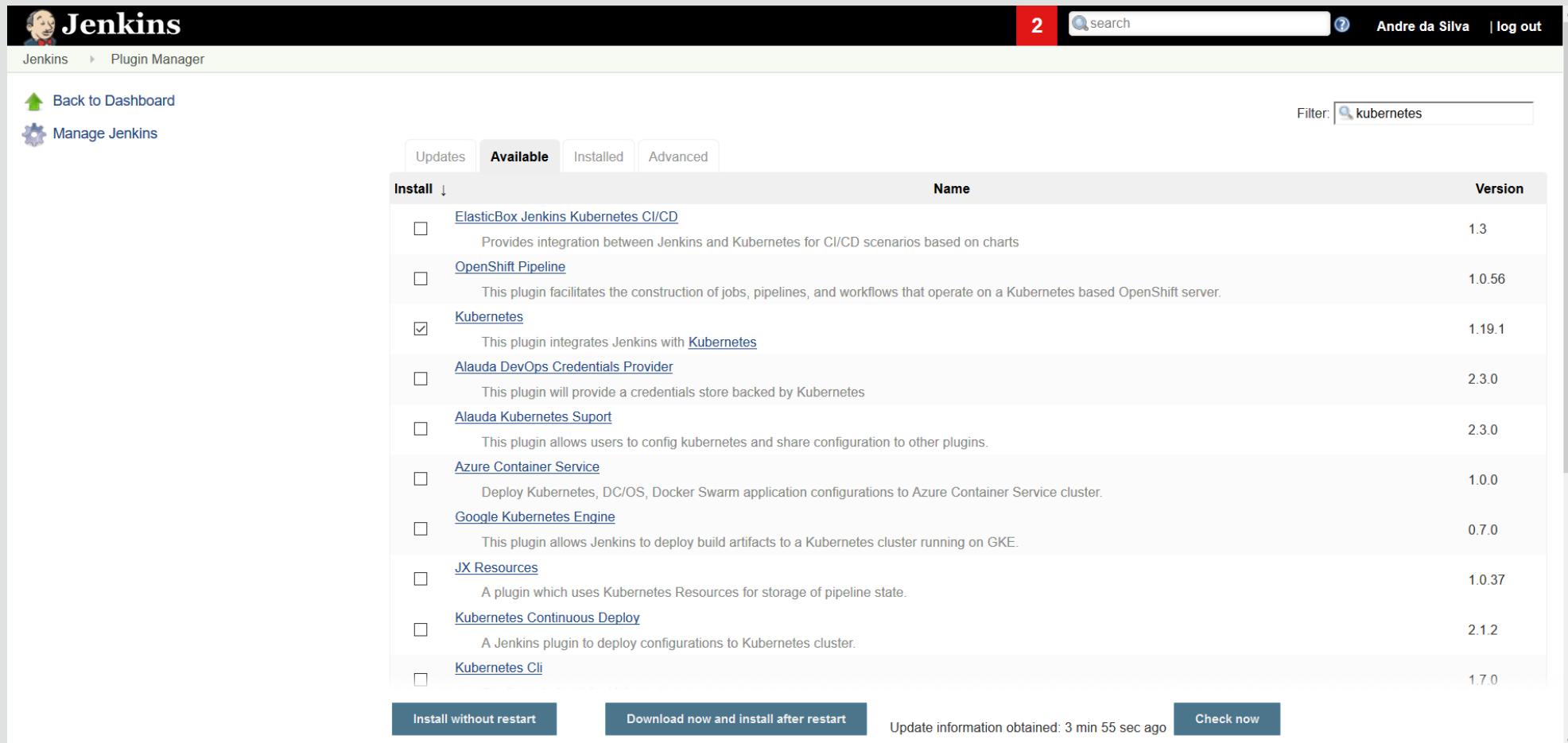
The screenshot shows the Jenkins Plugin Manager page. At the top, there is a navigation bar with the Jenkins logo, a search bar, and a user account for "Andre da Silva". A red notification badge with the number "2" is visible. Below the header, the page title is "Plugin Manager" under the "Jenkins" section. There are two main links: "Back to Dashboard" and "Manage Jenkins". A filter search bar is located on the right side of the header.

The main content area has a table with the following columns: "Install", "Name", "Version", and "Installed". The table displays the message "No updates".

Below the table, there is a status message: "Update information obtained: 7.5 sec ago" and a "Check now" button. A note below the message says: "Select: All, Compatible, None" and "This page lists updates to the plugins you currently use."

At the bottom of the page, there is a footer with the text: "Page generated: Sep 27, 2019, 1:48:25 PM UTC REST API Jenkins ver. 2.176.3".

Jenkins – Primeros pasos



The screenshot shows the Jenkins Plugin Manager interface. At the top, there is a navigation bar with the Jenkins logo, a user icon for 'Andre da Silva', and a 'log out' link. A red notification badge with the number '2' is visible. Below the navigation bar, the page title is 'Plugin Manager'.

On the left side, there are two links: 'Back to Dashboard' and 'Manage Jenkins'. Below these, there are four tabs: 'Updates' (disabled), 'Available' (selected), 'Installed', and 'Advanced'. A search bar at the top right contains the text 'Filter: kubernetes'.

The main content area displays a table of available Jenkins plugins related to Kubernetes. The columns are 'Name', 'Version', and 'Install'. The 'Install' column contains checkboxes for each plugin. The table rows are:

Install	Name	Version
<input type="checkbox"/>	ElasticBox Jenkins Kubernetes CI/CD Provides integration between Jenkins and Kubernetes for CI/CD scenarios based on charts	1.3
<input type="checkbox"/>	OpenShift Pipeline This plugin facilitates the construction of jobs, pipelines, and workflows that operate on a Kubernetes based OpenShift server.	1.0.56
<input checked="" type="checkbox"/>	Kubernetes This plugin integrates Jenkins with Kubernetes	1.19.1
<input type="checkbox"/>	Alauda DevOps Credentials Provider This plugin will provide a credentials store backed by Kubernetes	2.3.0
<input type="checkbox"/>	Alauda Kubernetes Suport This plugin allows users to config kubernetes and share configuration to other plugins.	2.3.0
<input type="checkbox"/>	Azure Container Service Deploy Kubernetes, DC/OS, Docker Swarm application configurations to Azure Container Service cluster.	1.0.0
<input type="checkbox"/>	Google Kubernetes Engine This plugin allows Jenkins to deploy build artifacts to a Kubernetes cluster running on GKE.	0.7.0
<input type="checkbox"/>	JX Resources A plugin which uses Kubernetes Resources for storage of pipeline state.	1.0.37
<input type="checkbox"/>	Kubernetes Continuous Deploy A Jenkins plugin to deploy configurations to Kubernetes cluster.	2.1.2
<input type="checkbox"/>	Kubernetes Cli	1.7.0

At the bottom of the page, there are three buttons: 'Install without restart', 'Download now and install after restart', and 'Check now'. A status message indicates 'Update information obtained: 3 min 55 sec ago'.

Jenkins – Primeros pasos

Screenshot of the Jenkins Update Center interface showing the "Installing Plugins/Upgrades" page.

The page displays the following information:

- Preparation:**
 - Checking internet connectivity
 - Checking update center connectivity
 - Success
- Kubernetes Credentials:** Success (blue circle)
- Variant:** Success (blue circle)
- Kubernetes Client API:** Installing (grey circle with progress bar)
- Kubernetes:** Pending (grey circle)

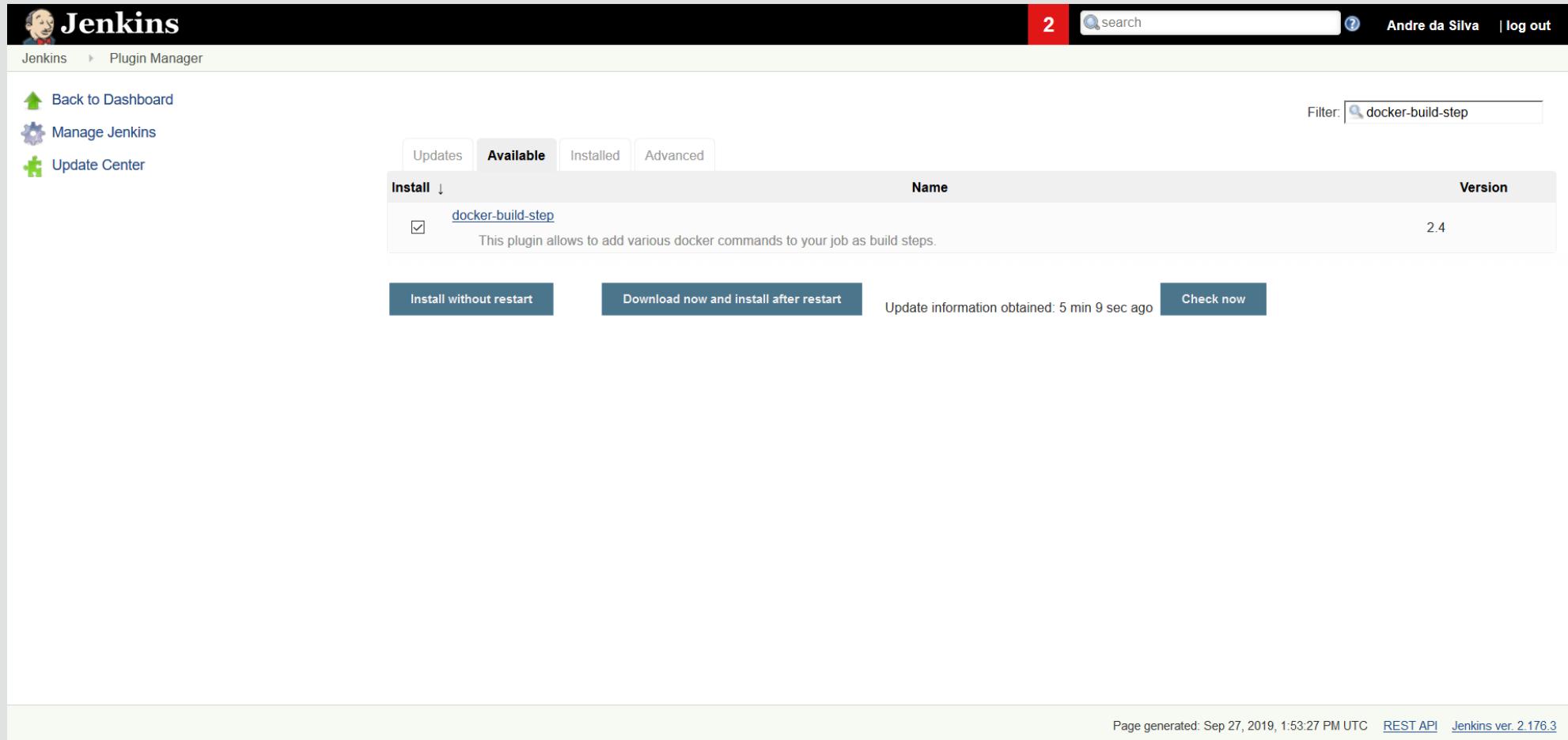
Buttons at the bottom:

- [Go back to the top page](#) (you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running

Page footer:

Page generated: Sep 27, 2019, 1:52:57 PM UTC [REST API](#) Jenkins ver. 2.176.3

Jenkins – Primeros pasos



The screenshot shows the Jenkins Plugin Manager interface. At the top, there is a navigation bar with the Jenkins logo, a search bar, and user information (Andre da Silva | log out). A red notification badge with the number '2' is visible. Below the navigation bar, the page title is 'Plugin Manager'.

On the left side, there is a sidebar with links: 'Back to Dashboard', 'Manage Jenkins', and 'Update Center'. The 'Available' tab is selected in the top navigation bar. A search bar at the top right contains the text 'docker-build-step'.

The main content area displays a table for the 'docker-build-step' plugin. The table has columns for 'Name' and 'Version'. One row is shown, with the name 'docker-build-step' and version '2.4'. A note below the table states: 'This plugin allows to add various docker commands to your job as build steps.' There are three buttons at the bottom of this section: 'Install without restart', 'Download now and install after restart', and 'Check now'. A status message 'Update information obtained: 5 min 9 sec ago' is also present.

At the bottom of the page, there is a footer with the text 'Page generated: Sep 27, 2019, 1:53:27 PM UTC REST API Jenkins ver. 2.176.3'.

Jenkins – Primeros pasos

Screenshot of the Jenkins "Installing Plugins/Upgrades" page.

The page shows the progress of installing several plugins:

Plugin	Status
Preparation	Success
Kubernetes Credentials	Success
Variant	Success
Kubernetes Client API	Success
Kubernetes	Success
docker-build-step	Installing

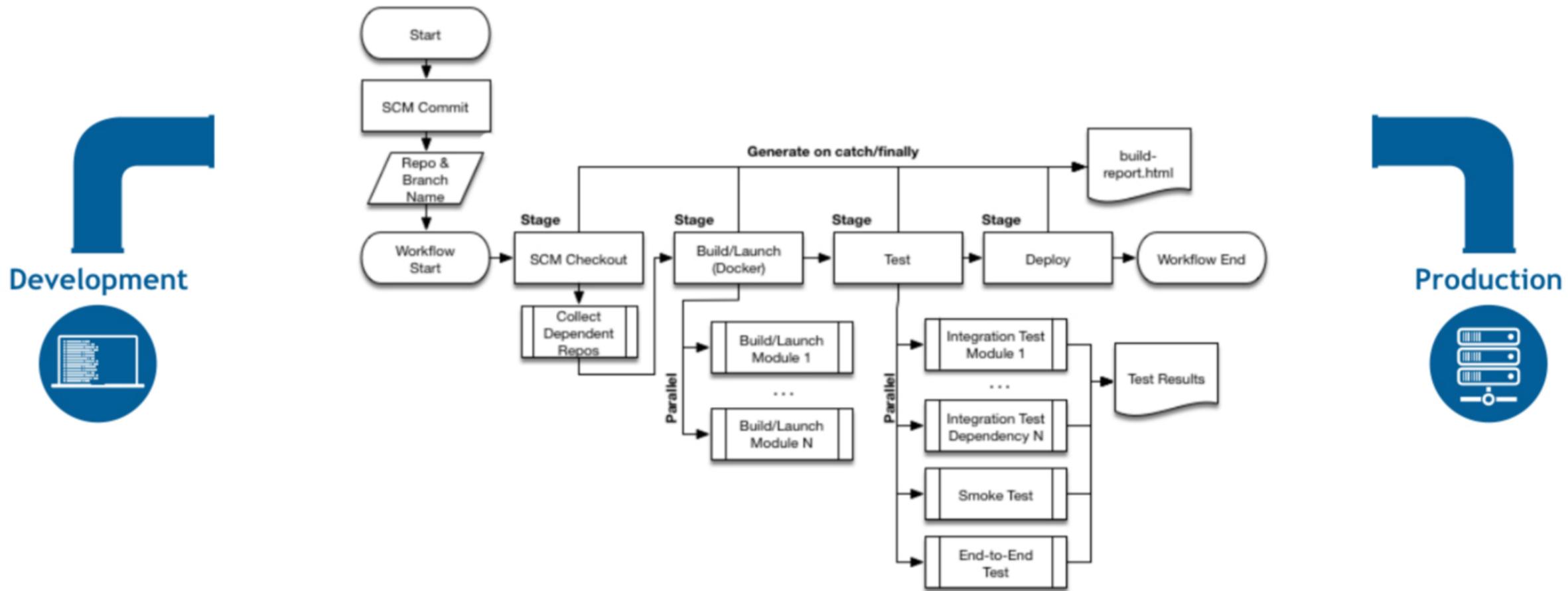
Below the table, there are two links:

- [Go back to the top page](#) (you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running

Page footer: Page generated: Sep 27, 2019, 1:54:07 PM UTC REST API Jenkins ver. 2.176.3

Jenkins – Primeros pasos

Jenkins Workflow



Demo

Enlaces útiles para referencia

GRATIS - DEVOPS Essentials Professional Certificate Training Material

<https://www.ennovate.cl/wp-content/uploads/2019/01/Material-Trainer-DEPC-V1-parte1.pdf>

CNCF - Cloud Native Computing Foundation Home Page

<https://www.cncf.io/>

DevOps Institute Home Page

<https://devopsinstitute.com/>

DevOpsDays Home Page

<https://devopsdays.org>

DevOps Roadmap

<https://roadmap.sh/devops>

StackShare

<https://stackshare.io>

StackOverflow – for developers, by developers

<https://stackoverflow.com/>

Kubernetes

<https://kubernetes.io/es/>

Jenkins

<https://jenkins.io/>

Enlaces útiles para referencia

Oracle Linux Cloud Native Environment

https://www.oracle.com/a/ocom/docs/oracle-linux-cloud-native-environment-ds.pdf?elq_mid=129432&sh=26132309220818150526292227322215&cmid=CNVG180718P00078

Oracle Linux Cloud Native Environment Training (Free Videos Trainings) Docker, Kubernetes, Kata Containers, Istio, Helm, Prometheus

https://apexapps.oracle.com/pls/apex/f?p=44785:141:28164444451620::::P141_PAGE_ID,P141_SECTION_ID:544,3742

Official Docker builds of Oracle Linux

https://hub.docker.com/_/oraclelinux/

Pre-Built Developer VMs (for Oracle VM VirtualBox)

<https://www.oracle.com/downloads/developer-vm/community-downloads.html>

Oracle Linux Vagrant boxes

<https://yum.oracle.com/boxes>

Oracle Linux Yum Server

<http://yum.oracle.com/>

Oracle Container Registry

<https://container-registry.oracle.com/pls/apex/f?p=113:10:::::>



TIPS

Maillist

devopsweekly.com

Twiter

Gareth Rushgrove
@garethr

devops.com
@devopsdotcom

Alan Shiel
@ashimmy

Podcast
devopscafe.org

John Willis
@botchgalupe

Damon Edwards
@damonedwards

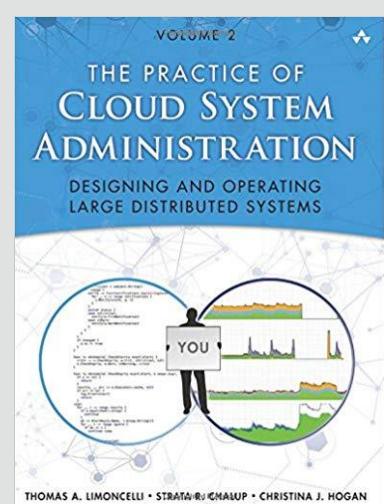
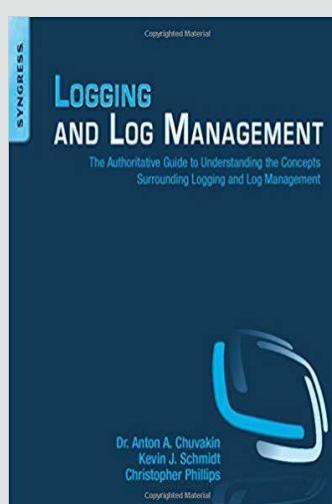
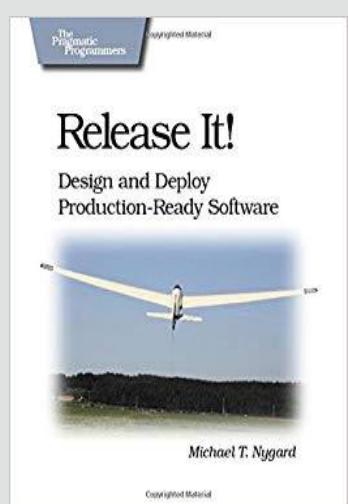
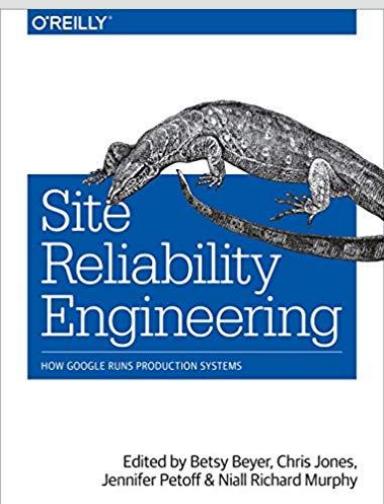
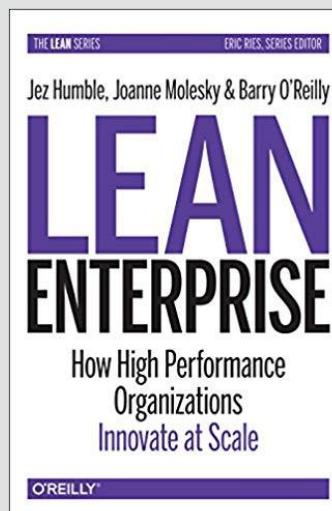
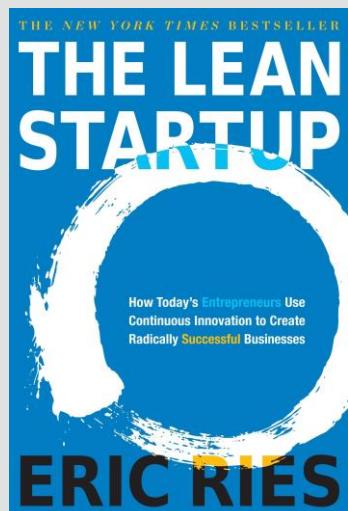
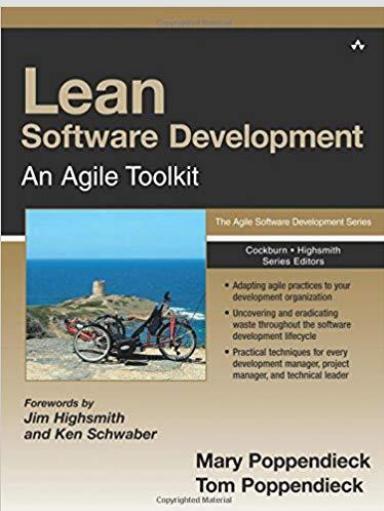
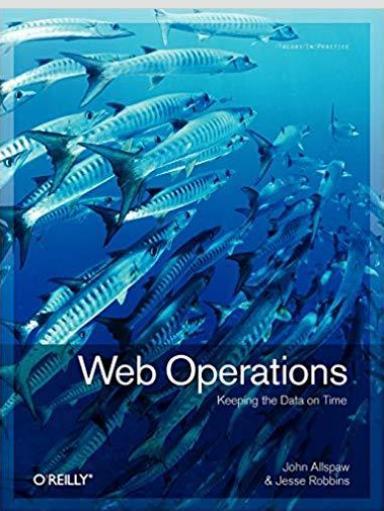
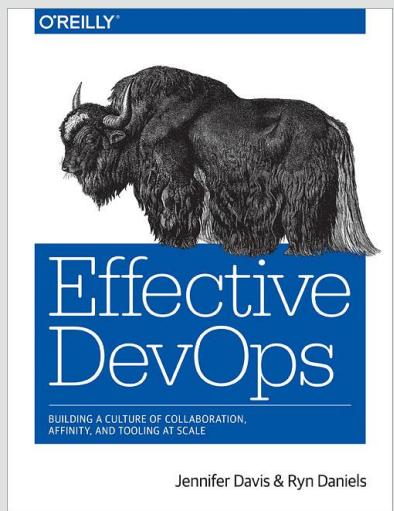
theshipshow.com

arresteddevops.com

devopsmastery.com/podcasts

Brian Wagner
@devopsmaster

Sugestión de libros



Sugestión de libros

