

Red Hat  
**Summit**

**Connect**

# Automatización en organizaciones complejas

Caso de Estudio

**Juanjo Giménez**

CEO SeidorOpentrends



# Grupo SEIDOR

Somos una consultora tecnológica que impulsa la competitividad y la transformación de las organizaciones desde las oportunidades que presentan las tecnologías y el conocimiento de negocio, con foco siempre en el valor de lo humano y comprometidos con el talento y el desarrollo social.



**40 años**

en el mercado



**45 países**

83 Oficinas en Europa, LATAM, EEUU, Oriente Medio, África y Asia



**TOP-10**

6ª empresa de servicios TI en España\*  
1ª de capital español 100% privado



**765 M€**

x2 en los últimos 5 años



**+8.000 personas**

50% millennials



**+8.500 clientes**

Gran empresa, PYME y Sector Público

# SEIDOR OPENTRENDS

## **Humanizamos la tecnología**

Somos digitalizadores. Un grupo transdisciplinar que crea experiencias tecnológicas únicas diseñadas por y para personas. A través de la colaboración, la empatía y la multiculturalidad como formas de crear un impacto duradero en un mundo mejor, más sostenible e inclusivo.

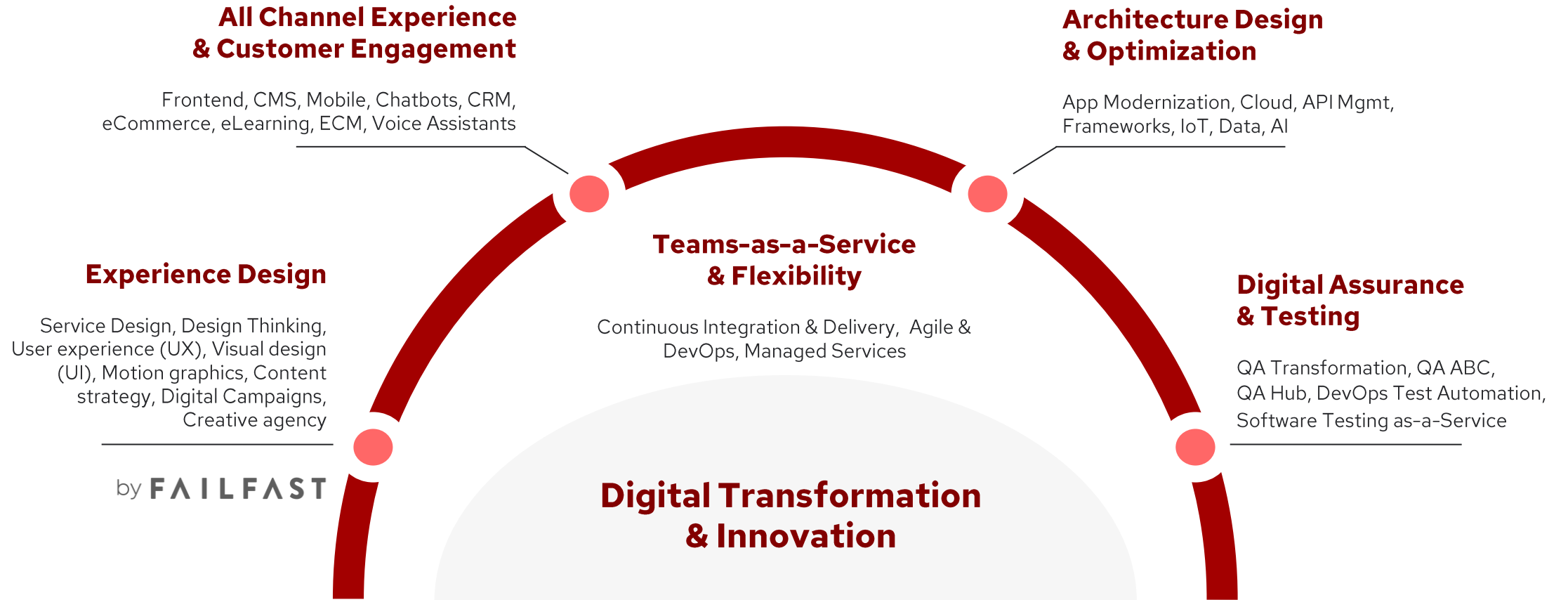
## **Impulsamos el New Digital**

Derribamos fronteras espacio-temporales explorando lo desconocido. Vivimos en una era que se mueve entre lo virtual y lo real. El New Digital va más allá de la tecnología y entiende la innovación como una oportunidad para crear experiencias inmersivas y ultrapersonalizadas.

## **Construimos Soluciones Digitales. De extremo a extremo**

Probamos, descartamos y adaptamos las nuevas tecnologías a tus necesidades desde el punto de vista tanto tecnológico como creativo. Lo que permite centralizar el proyecto en un solo proveedor. Por este motivo, hemos creado nuestra propia agencia digital, Fail Fast.

# Construimos soluciones digitales. Extremo a extremo



# Nuestros clientes



# Caso de Estudio

# Algo de (pre)Historia

## Mi primer Docker

En el equipo de Desarrollo y Arquitectura del **Institut Municipal d'Informàtica del Ayuntamiento de Barcelona** desde hace +15 años

Hace 6-7 años se parte de un escenario on-prem con virtualización de servidores y gestionado con enfoque ITIL que se quiere evolucionar hacia despliegue en contenedores y enfoque DevOps, con el objetivo principal de mejorar la escalabilidad de las aplicaciones

Ya se automatizaba en el desarrollo, principalmente la compilación, testeo y despliegue en entornos no productivos.

De hecho los servicios ALM (Git, Jenkins, Sonar...) son los primeros en dockerizarse y desplegarse con Rancher. Primeros proyectos piloto con despliegue manual

# Algo de Historia

## Contenedores + Kubernetes

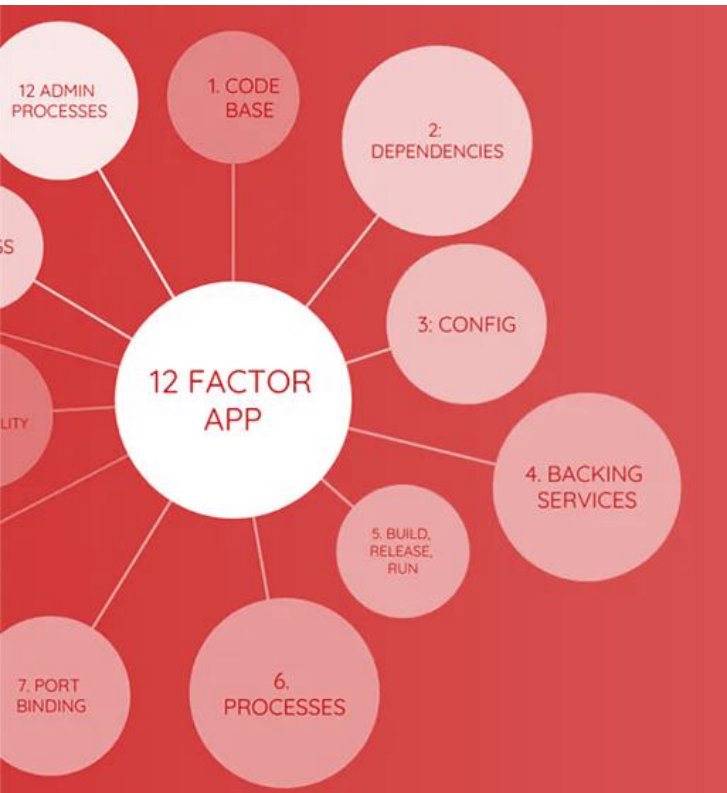
Una vez evaluadas las tecnologías se ve enseguida que:

- ▶ Hace falta un entorno de contenedores más robusto, se pasa a Kubernetes con IBM Cloud Private
- ▶ No se pueden provisionar los entornos de contenedores con la escalabilidad deseada con operación manual o semimanual, se incorpora el concepto de infraestructura como código (IaC) y se automatiza su despliegue con herramientas como Helm, Ansible y Terraform
- ▶ En un sitio como el ayuntamiento no se puede dejar sólo a los desarrolladores la responsabilidad de definir y desplegar esa infraestructura en los sistemas corporativos.



# Primera versión funcional

ICP (Contenedores + Kubernetes) + Custom ALM + 12 Factors



Desde la oficina de arquitectura se definen entornos predefinidos y preparados para que las apps desarrolladas en los diferentes stacks tecnológicos puedan desplegarse correctamente en la arquitectura empresa del ayuntamiento (reglas de seguridad, conectividad entre zonas, FWs, registro de imágenes único, API manager, monitorización, Backup, DR ...)

Se desarrolla a medida un orquestador de todos estos procesos y herramientas que automatiza toda la construcción y despliegue según las normas y decisiones tomadas desde las áreas de arquitectura, seguridad y sistemas. **Primera adaptación a la realidad tecnológica y organizativa**

Para los nostálgicos, siguiendo la metodología Twelve Factors: código fuente único, dependencias declaradas, configuraciones incluidas,... lo que nos facilitaba el camino posterior a DevOps y todavía más tarde a GitOps.

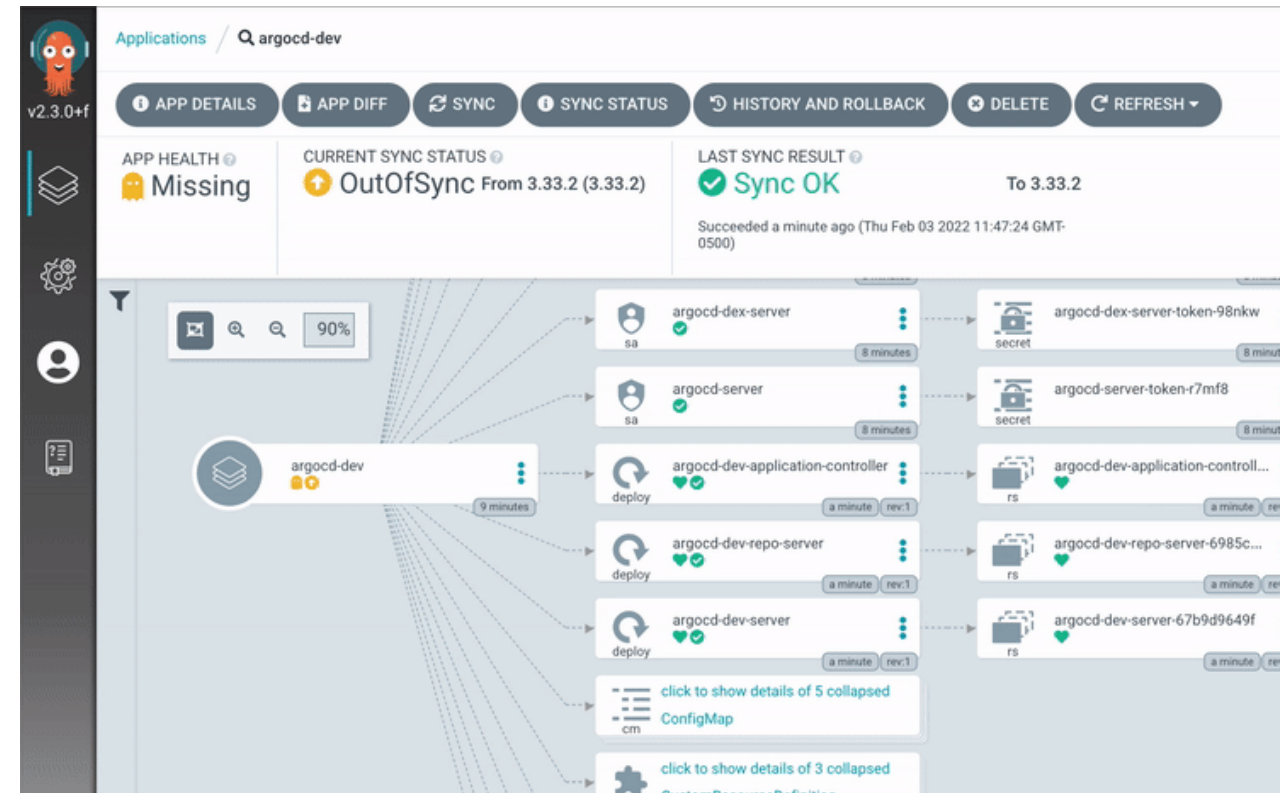
# Actualidad

## OpenShift 4.x (Contenedores + Kubernetes) + Custom ALM + DevOps > GitOps

Con aplicaciones compiladas desde el código con Integración continua (CI/CD), con sus configuraciones y entornos construidos y desplegados de manera declarativa (IaC), el siguiente paso lógico era integrar toda esa información en el control de versiones (GIT), nos movemos a un modelo GitOps

Se evalúa incorporar Argo CD diseñado específicamente para desplegar en kubernetes con enfoque GitOps, complementando al veterano Jenkins

Una vez más, se consensua la adaptación de la herramienta al ALM y procesos de la casa, desde las áreas de arquitectura, sistemas y seguridad. Y una vez más es la parte más complicada.

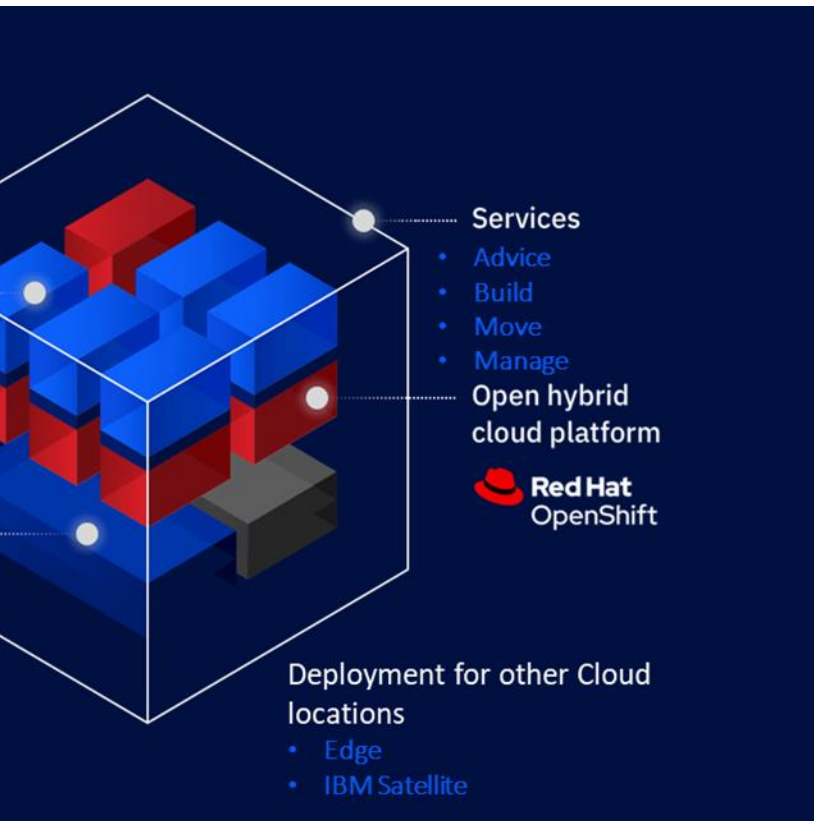


# Evolución

## Journey to Cloud (Híbrido)

Igual que en otras organizaciones con predominancia de tecnología VMWare, se plantea hacer un primer salto a la nube con soluciones como VMWare Cloud para facilitar el movimiento de cargas de trabajo y servidores entre los 2 mundos (cloud, on-prem), organizaciones que han dado el paso hacia la contenedorización lo hacen montando en la nube entornos kubernetes propios. En este caso, con IBM Cloud Pak sobre Azure RedHat Openshift (ARO)

Si bien levantar contenedores a demanda desde imágenes comunes en el cloud o on-prem, nos aleja del concepto lift & shift de cuando migramos VMs a la nube, con aplicaciones monolíticas, estas aplicaciones todavía no aprovechan todas las posibilidades de un enfoque cloud-native más allá de su entorno kubernetes (no PaaS, no Serverless)



# Evolución (II)

## Site Reliability Engineering (SRE) ? AIOps ?

Nada para y continuamente hay nuevas tecnologías y enfoques a evaluar para ver que nos pueden aportar

SRE comparte muchas similitudes con el enfoque DevOps, como la necesidad de colaborar entre los diferentes equipos, la automatización de los procesos, la implementación de cambios graduales, el uso de métricas y monitorización... si bien su foco es más en el ciclo de vida de la operación que en el de las aplicaciones.

Por su parte AIOps, aplica ML a toda esa información de monitorización (actual e histórica) para detectar y prever incidentes, determinar las acciones correctivas o respuestas más adecuadas y aplicarlas, ¿cómo? Con la automatización, claro

Otra vez habrá que evaluar soluciones e integrarlas en el ecosistema técnico y organizativo

The screenshot displays the IBM Automation interface for a specific incident. The title is "Robot-info-service internal Server Error + 30 alerts, 2". The interface is divided into several sections:

- Overview:** Shows a list of "Probable cause alerts" with three items:
  - IO latency high for storage controller (Runbooks)
  - Average database transaction duration high (Runbooks)
  - Memory utilization high on the front end service (Runbooks)
- Alert topology:** A diagram showing the relationship between alerts. It includes a "Distributed stor..." alert and a "Storage controller" resource. An arrow labeled "Uses volume from" points from the storage controller to the distributed storage alert.
- Resources:** Lists "Pod 1:Fuel tracker".
- Recommended runbooks:** A search bar with the text "Filter for recommended runbook associated with this story".

# Lecciones Aprendidas

## Algunas cifras

Con más de :

+300 aplicaciones y plataformas

+900 servidores virtuales

+1000 elementos desplegables

+10 stacks tecnológicos

+130 proyectos activos

+20 proveedores IT externos

Los cambios tecnológicos se ha de introducir paulatinamente, de manera segura y continuada, sin afectar los servicios y procesos que soportan.

Mas allá del impacto de la tecnología, tanto DevOps, como GitOps o SRE, balancean y modifican tanto roles como responsabilidades en los nuevos procesos a implantar, haciendo el impacto en la organización todavía mayor

## Lecciones Aprendidas (II)

En arquitectura tenemos pocos quickwins y pocas oportunidades de empezar de cero

A los nuevos modelos objetivos se llega después de largas transiciones desde sistemas legacy y operativas preexistentes. Es más fácil desplegar nuevas herramientas y tecnologías que cambiar operativas y procesos

Los beneficios se constatan a medida que se completa la transición. La escalabilidad es evidente a medida que tenemos nuevas aplicaciones en esos entornos, en cambio el ahorro de costes se ve impactado por la velocidad a la que podemos decomisionar los sistemas legacy. Es más fácil añadir que quitar

En todo caso, es imposible abordar esta transición sin oficinas y equipos de arquitectura, sistemas, operación, cloud, seguridad... que trabajen coordinadamente, incorporando conocimiento interno de la organización y externo de las nuevas tecnología y tendencias del mercado

Red Hat  
**Summit**

**Connect**

Thank you



[linkedin.com/company/red-hat](https://www.linkedin.com/company/red-hat)



[facebook.com/redhatinc](https://www.facebook.com/redhatinc)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[twitter.com/RedHat](https://twitter.com/RedHat)