

FACULTAD DE INFORMÁTICA

ADRIÁN GARCÍA GARCÍA

Arquitecturas Cloud y microservicios

April 28, 2017



This work is licensed under a [CC-BY-SA 4.0 License](https://creativecommons.org/licenses/by-sa/4.0/).

Índice

Introducción, 2

Abstract

Este trabajo tiene como objetivo exponer y profundizar en los conceptos expuestos en una conferencia optativa que tuvo lugar durante la semana de la informática de 2017. Los ponentes fueron dos ingenieros de la empresa GMV (Ricardo de Castro y Roberto Galán). En concreto, el nombre de la conferencia era el siguiente: Despliegue automático de arquitecturas escalables basadas en microservicios sobre el Cloud de Google (23 de Febrero, 11-14 horas). Conviene puntualizar que al final no usaron el Cloud de Google, sino que se basaron en Amazon Web Services y Docker Swarm para desplegar una aplicación web que se basaba en el uso de microservicios para su funcionamiento.

1 Introducción

Durante la conferencia, se hizo especial hincapié en las necesidades de los clientes y las características que estos exigen en una aplicación cloud. Los requisitos más destacados son los siguientes:

- **Fiabilidad.** El funcionamiento de las aplicaciones web está íntimamente relacionado con la imagen que las grandes empresas tienen en la sociedad, si una tienda de comercio electrónico se cae durante una hora puede suponer pérdidas millonarias para la empresa y acarrea el riesgo de que sus clientes dejen de confiar en ella y pasarse a la competencia. Debido a esto, los clientes esperan que su aplicación tenga el mínimo número de errores posibles. Uno de los aspectos más importantes en este sentido es que se haga un proceso meticuloso de *testing* y validación antes de publicar una aplicación, priorizando las partes más críticas del sistema y que pueden suponer un cuello de botella para la misma.
- **Recuperación instantánea ante fallos** (*Zero Down Time*[2]). En caso de que la aplicación llegue a fallar, lo que es imposible de evitar al completo, es necesario minimizar el tiempo en el que no se provee un servicio. Asegurar un tiempo mínimo de recuperación supone un desafío importante cuando una máquina física actúa como el punto de fallo único de la aplicación. Sin embargo, el hecho de tener una aplicación desplegada en contenedores virtuales, nos facilita en gran medida asegurar un sistema tolerante a fallos. Este sistema debe constar de ciertas características, entre las que se destacan:

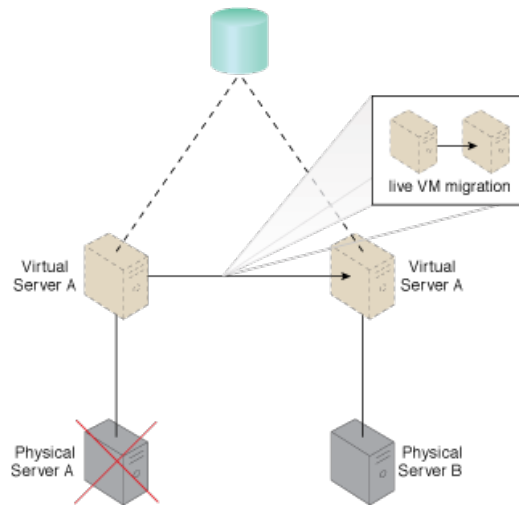


Figure 2: Esquema de recuperación ante fallos.

- Disponer de un cluster de máquinas configurado.
- Aplicar técnicas de replicación y balanceo de carga.
- Tener montado un volumen compartido de datos que permita a diferentes máquinas acceder a las imágenes que utilizan las máquinas virtuales.

Cómo se puede observar en la Figura 2, cuando el servidor físico A falla, se desencadena una migración del contenedor virtual al servidor B. Lo que debería ocurrir sin problema en caso de que el volumen de datos compartido por red siga disponible y solo haya ocurrido un problema aislado en la máquina A.

- **Tiempo rápido de despliegue.** Uno de los problemas más extendidos en el ámbito de las aplicaciones empresariales de elevada complejidad es que la aplicación puede tener muchas dependencias entre diferentes componentes, lo que hace difícil realizar un despliegue limpio y sin errores. Se ponía como ejemplo el clásico problema de que en el entorno de desarrollo funciona todo perfectamente, pero cuando se traslada a producción deja de hacerlo y cada equipo le echa la culpa a otro.
- **Escalabilidad.** Los servicios pueden recibir grandes picos de tráfico y deben ser capaces de responder ante los mismos sin errores. Es evidente

que la optimización de cada componente de la aplicación es un aspecto importante, pero además se debe diseñar un sistema capaz de balancear la carga y aumentar la dedicación de recursos replicando sus unidades de cómputo para responder ante los incrementos en demanda. Esto es relativamente fácil de hacer en entornos cloud o con soluciones de virtualización como Docker.

References

- [1] Fundamentals of Cloud Service Reliability, Microsoft Secure Blog. [Enlace](#).
- [2] Zero Downtime, CloudPatterns.org. [Enlace](#)