

# MICROSERVICIOS EN LA NUBE: CONSECUENCIAS Y RETOS

1<sup>st</sup> Hazel David Pinzón Uribe  
Ingeniería De Sistemas  
Uis  
Bucaramanga - Santander  
hazelpinzon13@gmail.com

2<sup>nd</sup> Andrea Jimena Morales Osma  
Ingeniería De Sistemas  
Uis  
Bucaramanga - Santander  
andreaajm13@hotmail.com

3<sup>rd</sup> Sergio Andres Dulcey Lizarazo  
Ingeniería De Sistemas  
Uis  
Bucaramanga - Santander  
serduli18@gmail.com

## ***Resumen***

Los servicios en la nube han sido aplicaciones monolíticas durante mucho tiempo, actualmente se están reemplazando a una arquitectura de aplicaciones de micro servicios las cuales tienen ventajas ya que estos facilitan la prueba y el despliegue de cambios. Debido a que cada uno está separado de los otros, se mejora el aislamiento de fallas. En este artículo se profundizará sobre el funcionamiento, retos y consecuencias que tienen estos en los servicios más utilizados actualmente tales como aplicaciones representativas de extremo a extremo que se hayan creado con micro servicios. Analizando si esta arquitectura ha resultado más útil y fácil de usar que las aplicaciones monolíticas, y como es la relación de las microarquitecturas.

## ***Abstract***

Cloud services have been monolithic applications for a long time, they are currently replacing an architecture of microservices applications which have advantages as these facilitate the testing and deployment of changes. Because each one is separated from the others, fault isolation is improved. This article will elaborate on the functioning, challenges and consequences that these have in the most commonly used services, stories as representative end-to-end applications that have been created with microservices. Analyzing if this architecture has a more useful and easy to use result than monolithic applications, and how is the relation of microarchitectures.

## ***I. Introducción***

Los microservicios presentan oportunidades y desafíos al optimizar la calidad del servicio (QoS) y la utilización de la nube. A través de éste documento, que será entregado para la asignatura Arquitectura de Computadores de Ingeniería de Sistemas impartida en la Universidad Industrial de Santander, se plantea como tema las implicaciones arquitecturas de microservicios en la nube además de mostrar y explicar qué es lo que se busca con los mismos y cuáles son los propósitos de estos sistemas en la actualidad mediante la definición de conceptos y desarrollando la evolución y los alcances de dicha tecnología hasta nuestros días.

En un esfuerzo para satisfacer estos, a menudo chocando requisitos, aplicaciones de nube han pasado por redefiniciones extensas. Esto incluye un turno reciente de servicios monolíticos que abarca la funcionalidad del servicio entero en un solo binario, a centenares o miles de pequeños

Los microservicios están apelando por varias razones. Primero, ellos simplifican y aceleran el despliegue a través de modularidad (La modularidad del software indica la cantidad de módulos que la aplicación es capaz de servir a un dominio comercial específico), como cada microservice es responsable para un subconjunto pequeño de la funcionalidad de la aplicación entera. los microservicios puede aprovechar el lenguaje y la programación heterogénea de marco, desde entonces sólo requieren una cruz común-aplicación API, típicamente sobre llamadas de procedimiento remoto (RPC).

## ***II. Marco teórico***

En el documento relacionado se estudian las implicaciones de los microservicios en la nube, para lo que es necesario definir algunos conceptos básicos que nos lleven a un mejor entendimiento del tema, para iniciar, cuando hablamos de un servicio en la nube hacemos referencia a servicios que se utilizan a través de Internet.

Antes del concepto en la nube, todo lo referente al procesamiento y almacenamiento de datos se hacía desde el propio ordenador. Con los servicios en la nube conseguimos utilizar aplicaciones o sistemas de gestión de manera remota gracias a Internet, sin necesidad de tenerlos instalados [1], por ende, deberíamos hablar de sistemas monolíticos como aquellos en los que su centro es un grupo de estructuras fijas, las cuales funcionan entre sí. Entendemos también que una arquitectura de microservicios consta de una colección de servicios autónomos y pequeños. Los servicios son independientes entre sí y cada uno debe implementar una funcionalidad individual [2]. Los microservicios es un componente fundamental de la optimización del desarrollo de aplicaciones hacia un modelo nativo de la nube, los microservicios son tanto un estilo de arquitectura como un modo de programar software. Con los microservicios, las aplicaciones dividen sus componentes más pequeños e independientes entre sí. A diferencia del enfoque tradicional de las aplicaciones, en el que todo se integra en una única pieza, los microservicios son independientes y funcionan en

conjunto para llevar a cabo las mismas tareas. Cada uno de estos elementos o procesos es un microservicio. Este enfoque sobre el desarrollo de software privilegia el nivel de detalle, la sencillez y la capacidad de compartir procesos similares en varias aplicaciones.[4]

El enfoque de este artículo es explorar el cambio de sistema monolíticos a microservicios, este se da debido a que migrar a la nube es un paso importante del proceso de modernización, pero estos beneficios no se dan de manera automática por el simple hecho de estar en la nube.

Cumplir con la promesa de un servicio más eficiente requiere de un ajuste paralelo en la arquitectura de su aplicación. Las arquitecturas monolíticas se convierten fácilmente en un cuello de botella para sistemas complejos y a gran escala. los microservicios son un elemento esencial de una estrategia de adopción de prácticas cloud para la transformación digital [3], pero el cambio a microservicios trae también consecuencias, estos afectan la relación entre computación y comunicación, a medida que la comunicación domina y la cantidad de computación por microservicio disminuye [5], he de tener en cuenta igualmente los servidores utilizados. Para poder analizar estos necesitamos aplicaciones representativas de extremo a extremo que se hayan creado con microservicios.

Algunas paginas conocidas que son pioneros en el estilo arquitectónico incluyen Amazon, Netflix, The Guardian , el Servicio Digital del Gobierno del Reino Unido , realestate.com.au , Forward y comparethemarket.com [6], todas han tenido experiencias positivas, Estudios respecto al tema han hecho Clousuite, Tailbench o sirius con servicios interactivos en la nube [5].

### **III. Estado Del Arte**

Los microservicios están de moda porque muchas compañías grandes y tecnológicamente sofisticadas lo usan como son Amazon, eBay, Twitter, Netflix, PayPal, que son algunos ejemplos exitosos de microservicios.

El artículo Yu Gan and Christina Delimitrou habla sobre las implicaciones arquitectónicas de los microservicios en la nube y es un referente para basarnos en este artículo.

Cloudsuite incluye servicios por lotes e interactivos, y se ha utilizado para estudiar las implicaciones arquitectónicas de los puntos de referencia de la nube [11].

### **IV. Contenido**

#### **A. Aplicaciones Monolíticas**

Las aplicaciones monolíticas se refiera a una única aplicación de software en la que el interfaz de usuario y el código de acceso a datos se combinan en un solo programa.

Cuando empezó la época de los computadores personales era común encontrarse con esta clase de aplicaciones, cuando una aplicación era almacenada en una única máquina, las

aplicaciones monolíticas contenían toda la funcionalidad de la aplicación en una gran y frecuentemente difícilmente mantenible pieza de software.

Sin embargo, cualquier cambio requerido a cualquier parte de la funcionalidad podría potencialmente afectar otras partes. Porque la presentación, negocio, y la lógica de acceso a datos están localizados dentro de la misma pieza de código de aplicación, la recopilación de varias partes del código podría ser necesaria, incrementando la sobrecarga de nueva o cambios de funcionalidad.

Incluso, los cambios en partes del código podría introducir bugs no intencionales en otras partes aparentemente no relacionadas.

Una aplicación monolítica es autónoma, independiente y de las aplicaciones computacionales. La filosofía del diseño es que la aplicación es responsable no sólo de una tarea determinada, pero puede realizar todos los pasos necesarios para completar una función particular. Hoy en día, algunas aplicaciones de finanzas personales son monolíticas en el sentido de que ayudan al usuario realizar una tarea completa, de extremo a extremo, y son “datos privados de los silos” en lugar de partes de un sistema más amplio de aplicaciones que funcionan juntos. Procesadores de texto son un ejemplo de una aplicación monolítica. Estas aplicaciones se asocian a veces con mainframe ordenadores.

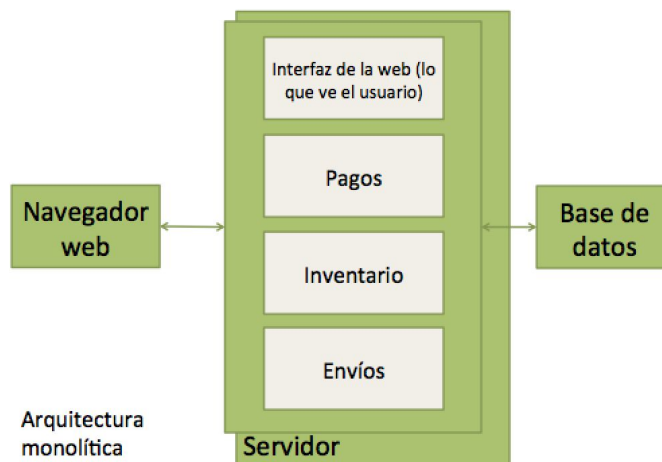


Figura 1. Aplicaciones Monolíticas

### B. MicroServicios

El concepto de microservicios lo definió Martin Fowler en 2014 como un enfoque técnico para desarrollar aplicaciones como un conjunto de pequeños servicios. Se trata de unidades funcionales independientes que se ejecutan de manera autónoma en su propio proceso y que se comunican entre sí a través de APIs. Es una alternativa a las soluciones monolíticas más tradicionales, donde todos los elementos funcionales están unidos en un programa o aplicación[7].

Realmente no existe un estándar de cómo debe ser una arquitectura basada en microservicios: ni en el tamaño de los mismos, ni siquiera en el lenguaje de programación.

Cada pequeño servicio que compone la aplicación puede estar escrito en un lenguaje de programación diferente, ya que se comunica con los otros a través de llamadas de API. Estos servicios se desarrollan alrededor de capacidades de negocio y se pueden implementar de forma independiente y completamente automatizada. Existe un mínimo de administración centralizada de estos servicios, que puede escribirse en diferentes lenguajes de programación y utilizando diferentes tecnologías de almacenamiento de datos. Los microservicios se adaptan perfectamente a los requerimientos de agilidad, escalabilidad y confiabilidad de las aplicaciones modernas en la nube.

#### Ventajas que ofrecen los microservicios

- Escalabilidad: permiten escalar cada servicio de manera independiente para adaptarse a los picos de tráfico puntuales o estacionales.
- Productividad: la menor cantidad de código en comparación con un monolito permite entender claramente la funcionalidad del microservicio y facilita su ciclo de vida (desarrollo, mantenimiento, pruebas, implantación).
- Diseño: los microservicios obligan a mantener una adecuada separación física de funcionalidades en nuestra arquitectura. Esto promueve una filosofía UNIX al diseñar grandes aplicaciones (un componente hace una sólo cosa y la hace bien).
- Innovación más ágil: la independencia de los microservicios favorece su innovación y mejora.
- Mantenimiento más rápido y sencillo.
- Posibilidad de aislar errores.
- Despliegue progresivo.

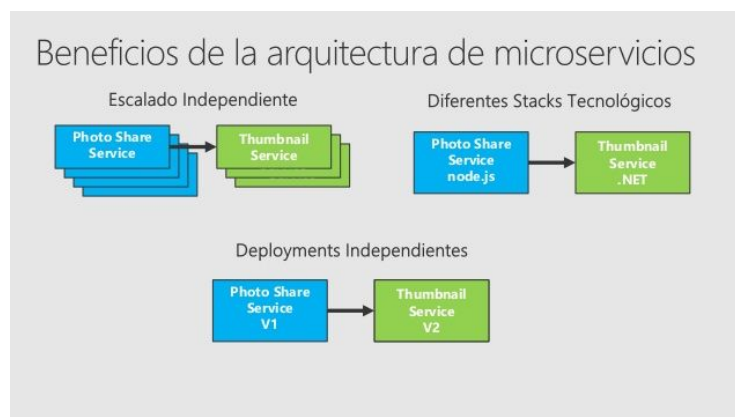


Figura 2. Beneficios Microarquitecturas

### C. Aplicaciones monolíticas vs MicroServicios

Microservicios Está apelando para varias razones. Primero simplifica y acelera el despliegue a través de modalidad (La modularidad del software que indica la cantidad de módulos de aplicación), como cada microservicio es responsable para un subconjunto pequeño de la funcionalidad de la aplicación entera. Los microservicios puede aprovechar el lenguaje de programación y heterogeneidad de marco en la programación, desde entonces sólo requieren una API de aplicación cruzada común, típicamente sobre llamadas de procedimiento remoto (RPC) o una API RESTful. En contraste, los monolitos hacen las actualizaciones frecuentes pesadas y son expuestas al error, y limita el conjunto de lenguajes de programación que puede ser utilizado para desarrollo.

Entonces, los microservicios simplifican la exactitud y la actuación de depuración, como los bugs que pueden ser aislados a componentes concretos, a diferencia de monolitos, donde los troubleshooting (solución de problemas) a menudo implica el End-to-end (Extremo a extremo). Finalmente, los microservicios cabe amablemente en el modelo de centro de datos, donde cada microservicio se sitúa en un contenedor. Un número de aumentó proveedores de servicio de la nube, incluyendo Twitter, Netflix, EN&T, Amazon, y eBay ha adoptado este modelo de aplicación.

A pesar de sus ventajas, los microservicios cambia varias conjeturas que se ha usado por mucho tiempo para diseñar y dirigir sistemas de nube. Por ejemplo, afectan la computación a proporción de la comunicación, sabiendo que la comunicación está dominando poco a poco, y la cantidad de computación por microservicios disminuye a la vez. De modo parecido, los microservicios requiere conocer qué tipo de servidores grandes o pequeños, son los que necesita, cuantificando el i-cache con presión de sus huellas de código y determinando las fuentes de rendimiento a través del servicio End-to-end. Donde se necesita aplicaciones extremo a extremo que se construye con los microservicios para contestar las preguntas sobre su funcionamiento y conocer sobre estos servicios.

Los monolíticos hacen las actualizaciones frecuentes pesadas y expuestas al error, donde limita el conjunto de lenguajes de programación que puede ser utilizado para desarrollo.

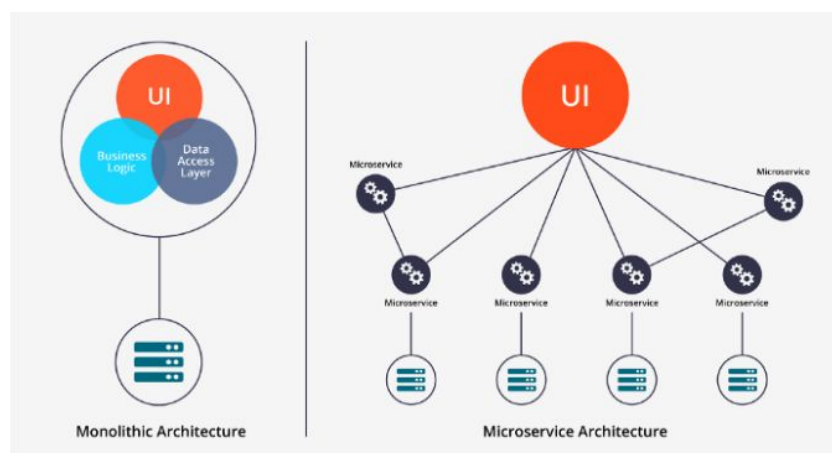


Figura 3. Aplicaciones Monolíticas vs Microservicios.

#### *D. Aplicaciones basadas en Microservicios*

Los microservicios sintetizan la rectificación y la depuración del rendimiento, puesto que los errores se aíslan en elementos específicos, a diferencia de los monolíticos, donde la solución a esos problemas con frecuencia involucra el servicio de End-to-end.

El principio End-to-end, conocido también como E2E, es un framework de diseño en las redes de computadores, donde su característica de la aplicación se establece en los nodos finales de comunicación en la red, en vez de nodos intermedios, como routers y los gateways. Este principio se soporta en el manejo de lo que se llama internet, ya que depende de dos capas conocidas como layers de procesamiento, donde sus funciones se basan en distribuir mensajes e interpretarlos.

La primera capa se llama capa de red, su función es procesar los datos según se desplace por medio de la red, mientras tanto la segunda capa llamada capa de aplicación es la encargada de recibir los flujos de bit y procesarlos para modificarlos en correos electrónicos, páginas web, entre otros.

La compañía VSN Video Stream Networks, una empresa española de alta tecnología, que se especializa en aportar soluciones avanzadas a la industria broadcast y media dar solución al menester de los entornos de creación, gestión y distribución de contenidos audiovisuales en diversos organismos; VSN dió a conocer la primera solución end to end en la nube para los microservicios. Esta solución se basa en permitir que las industrias audiovisuales les permita gestionar y reproducir su contenido en cualquier lugar, además de ofrecer ventajas como la escalabilidad que la tecnología de la nube aporta a las instalaciones de la industria y un flujo de trabajo optimizado para ser más eficiente en la nube. [10]

En el artículo de Yu Gan and Christina Delimitrou estudian: EL SERVICIO DE STREAMING DE PELÍCULAS DE EXTREMO A EXTREMO que nombraremos a continuación, El servicio de extremo a extremo se crea utilizando microservicios populares de código abierto, incluidos nginx, memcached, MongoDB, Xapian y node.js para garantizar la representatividad. Estos microservicios se conectan entre sí utilizando el marco RPC de Apache Thrift [1] para proporcionar la funcionalidad de servicio de extremo a extremo, que incluye mostrar información de películas, revisar, alquilar y transmitir películas y recibir Recomendaciones de publicidad y películas. A menos que se indique lo contrario, todos los microservicios se ejecutan en contenedores Docker para simplificar la instalación y la implementación.[5]

Un Caso específico es: Netflix

En el pasado, cuando todavía no era un servicio de streaming, sino que enviaba por correo películas en formato DVD, Netflix se basaba, como la mayoría de empresas, en un sistema monolítico, hasta que en 2008 un error en una base de datos provocó una interrupción del servicio durante cuatro días. A partir de este momento se decidió desintegrar el antiguo

sistema y dividirlo en microservicios. Con esto se logró que la empresa pudiera lanzar los cambios mucho más rápidamente y que las reparaciones también pudieran llevarse a cabo con mucha más agilidad. Como el sistema de Netflix es muy extenso, la empresa desarrolló un programa propio para coordinar los diferentes microservicios entre sí: este se conoce como Conductor.

Conductor permite que Netflix pueda gestionar los microservicios de forma central (pausar o reiniciar) o escalarlos. En el núcleo del programa trabaja un servicio de nombre Decider que puede planificar los procesos de forma automatizada y reaccionar a eventos en el workflow. Otros programas desarrollados por Netflix para trabajar eficazmente con microservicios son Mantis (Stream processing), Dymomite (datastore) y Vizceral (traffic intuition).[12]

### *E. Retos de los Microservicios*

Aunque traen muchas ventajas, los microservicios son un concepto relativamente nuevo y, por lo cual, presentan bastante retos:

- En primer lugar, la complejidad: Una aplicación basada en microservicios es más compleja que un monolito, ya que está compuesta por muchos servicios distintos e independientes. Se necesita por lo cual de una política de gobernanza adecuada.
- Además, manejar los fallos es más complicado, ya que se necesita monitorizar distintas piezas para detectar los posibles problemas.
- Finalmente, no todos los profesionales de IT poseen los conocimientos necesarios para desarrollar y gestionar correctamente una arquitectura de microservicios.

Hablando de los desafíos que implica una arquitectura basada en microservicios, Martin Fowler destaca:

-Distribución: los sistemas distribuidos son más complejos de programar ya que las llamadas remotas son lentas y están siempre en riesgo de fallo.

-Consistencia: mantener una fuerte consistencia es extremadamente difícil para un sistema distribuido, lo que significa que tiene que lidiar con consistencia eventual.

-Complejidad operacional: requiere un equipo de operaciones maduro y entornos con una gran capacidad de automatización y monitorización para gestionar muchos servicios que son desplegados regularmente.

### *F. Relación con las microarquitecturas*

En computación, una microarquitectura es una descripción del circuito eléctrico de una computadora, de la unidad central de proceso o del procesador digital de señal, que es suficiente para describir completamente el funcionamiento del hardware.[9]



El concepto de Microservicios tiene que ver con la separación de las preocupaciones y toma mucho de la filosofía de Unix de hacer una cosa y hacerlo bien. En parte por esa razón, creemos que debe haber una separación lógica y arquitectónica entre los servicios con diferentes responsabilidades.

Los microservicios deben ir asociados a microarquitecturas hechas para soportar la ejecución de cada servicio que en este caso se realizan por partes para finalizar en la convergencia de estos. Los microservicios se mantienen aislados los unos de los otros y se ejecutan en su propio entorno.

### ***V. Conclusiones***

Lo expuesto a lo largo de este artículo permite llegar a las siguientes conclusiones:

Aunque los microservicios es la nueva arquitectura utilizada para las aplicaciones en la nube, hay que tener en cuenta que para que el funcionamiento de estos sea el esperado, debe haber un correcto diseño de cada servicio y una arquitectura que los soporte. En algunos casos aún es funcional las aplicaciones monolíticas.

Las mayor parte de las aplicaciones en la nube se pueden mudar a microservicios, es más, lo están haciendo, pero deben tener en cuenta como hacer de manera correcta para que sea un caso de éxito, tal como pudo lograrlo netflix u otras, además por supuesto toma bastante tiempo lograr un buen funcionamiento.

Aunque los microservicios en su mayor parte trae beneficios, es necesario tener muy claro no solo las ventajas, sino también los retos que plantea una arquitectura basada en Microservicios, ya sea a nivel de costo por el cambio de arquitectura o por el hecho de sean globalmente separados.

Una aplicación con microservicios debe estar compactada con un hardware que soporte el funcionamiento de estos.

Las aplicaciones monolíticas apunta a una única aplicación de software en niveles en los que la interfaz de usuario y código de acceso a datos se combinan en un solo programa de una plataforma única, en cambio los microservicios es una visión para desarrollar aplicaciones como un conjunto de pequeños servicios, donde cada uno se ejecuta de manera autónoma y contacto entre sí.

## VI. Referencias

- [1] [https://www.palbin.com/customer/es\\_es/portal/articles/1507535-%C2%BFqu%C3%A9-es-un-servicio-en-la-nube-](https://www.palbin.com/customer/es_es/portal/articles/1507535-%C2%BFqu%C3%A9-es-un-servicio-en-la-nube-), 2014.
- [2] Microsoft Azure, Estilo de arquitectura de microservicios, <https://docs.microsoft.com/es-es/azure/architecture/guide/architecture-styles/microservices>, 2018,
- [3] De sistemas monolíticos a microservicios, <https://www.aplyca.com/es/blog/aplicaciones-monoliticas-o-microservicios>
- [4] Microservicios, <https://www.redhat.com/es/topics/microservices/what-are-microservices>
- [5] Yu Gan and Christina Delimitrou, The Architectural Implications of Microservices in the Cloud , <https://arxiv.org/abs/1805.10351>, 2018
- [6] James Lewis, Microservicios, <https://martinfowler.com/articles/microservices.html>, 2014
- [7] Martin Fowler, Microservicios, <https://martinfowler.com/articles/microservices.html>, 2014
- [8] MediaCloud, ¿Qué son los microservicios? Definición, características y retos, <https://blog.mdcloud.es/que-son-los-microservicios-definicion-caracteristicas-y-retos>
- [9] Definición de microarquitectura <http://www.alegsa.com.ar/Dic/microarquitectura.php>
- [10] VSN presenta la primera solución End-to-End en la nube <https://www.vsn-tv.com/es/end-to-end-en-la-nube-ibc-2017>
- [11] M. Ferdman, A. Adileh, and et al., "Clearing the clouds: A study of emerging scale-out workloads on modern hardware," in *ASPLOS*. 2012
- [12] <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/los-microservicios-en-el-desarrollo-de-aplicaciones/>