Microservicios

BBVA

¿Cúal es el propósito del desarrollo de software?

BBVA

¿Cúal es el propósito del desarrollo de software?

Crear impacto en el negocio

BBVA

El código no es un activo...

- Escribir código es un coste.
- Esperar por el código es un coste.
- Cambiar el código es un coste.
- Entender el código es un coste.

el código es el coste!

BBVA

Definición

El estilo de arquitectura de microservicios, es un enfoque para desarrollar una sola aplicación como un conjunto de pequeños servicios, donde cada uno de ellos se ejecuta en su propio proceso y se comunican mediante mecanismos ligeros, frecuentemente un API de recursos HTTP

Martin Fowler & James Lewis

BBVA

Que no es ...

- Monolítica
  - Toda la aplicación corre en un sólo gran proceso que soporta todas sus funcionalidades.
- Por capas y cross-functional
  - La aplicación se divide en capas técnicas con diferentes funcionalidades en cada una de ellas.
- Inteligentemente integrada
  - La complejidad se mueve a mecanismo de integración (ESB's).
- Integrada centralmente
  - Se integra usando una base de datos central.

BBVA

1 Características de este estilo de arquitectura

BBVA

Servicios como componentes

- Componente
  - Una unidad de software que es independientemente reemplazable y actualizable.
- Servicio
  - Proceso independiente que presenta un API para comunicarse con el.
  - Se despliega y escala de forma independiente.
  - Puede mapearse a un proceso en ejecución o a múltiples procesos que pueden generalmente ser desarrollados y desplegados de forma conjunta.

BBVA

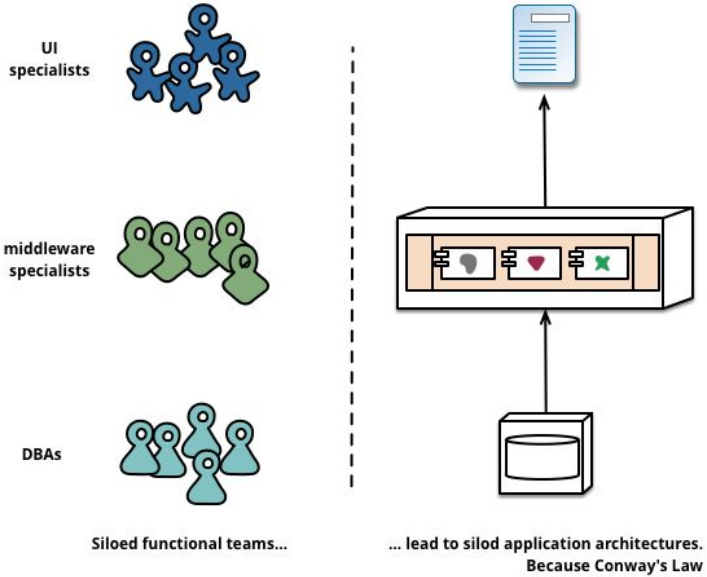
Cualquier organización que diseña un sistema (definido en términos generales) producirá un diseño cuya estructura es una copia de la estructura de comunicación de la organización

-- Melvyn Conway, 1967

BBVA

Organización.Monolito

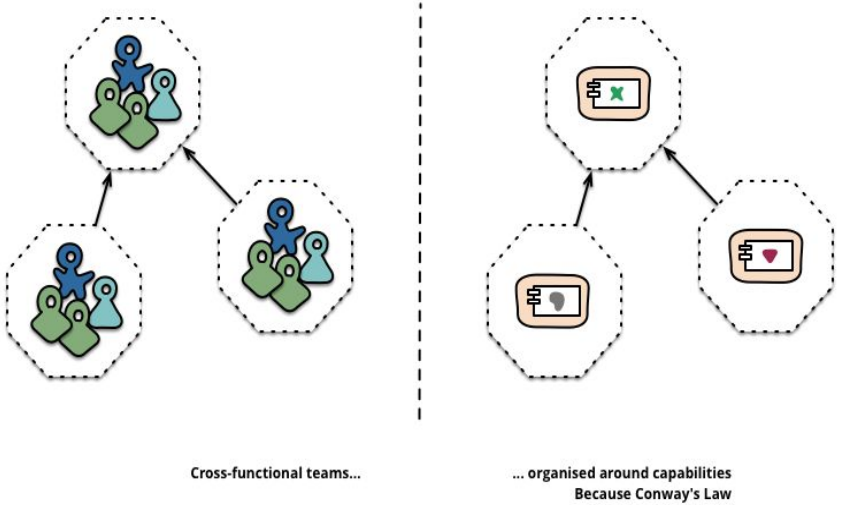
- Las aplicación se divide dependiendo de la tecnología, con equipos de UI, servidor equipos de BBDD, etc.
- Cambios simples involucran a varios equipos llevando tiempo y aprobaciones presupuestarias.
- La lógica de negocio se disemina entre varias capas.



BBVA

Organización.Microservicios

- Los servicios se organizan alrededor de capacidades de negocio.
- Los equipos son multidisciplinares.
- Los servicios toman una implementación completa del software para esa área de negocio, incluyendo UI, persistencia y cualquier colaboración externa.
- Ley de Conway inversa.
- Organizados alrededor de los Bounded Contexts. DDD.



BBVA

Productos no Proyectos

- En lugar de entender el software como un conjunto de funcionalidades, la pregunta es cómo el software puede asistir a los usuarios para mejorar las capacidades de negocio.
- El equipo de desarrollo tiene responsabilidad absoluta sobre el software en producción, es dueño del producto durante todo su ciclo de vida.

BBVA

Mecanismos de Integración

- Las aplicaciones construidas en microservicios buscan ser todo lo desacopladas y cohesivas como sea posible.
- Los servicios se integran:
  - Utilizando sencillos protocolos basados en REST.
  - Usando un bus de mensajes ligero. La infraestructura elegida es típicamente tonta, en el sentido que sólo hace de enrutador.
- La inteligencia vive en los en los servicios que consumen y producen mensajes.
- Se buscan distintas maneras distintas de gestionar esos contratos. Patrones como el Tolerant Reader y el Consumer-Driven Contracts se aplican a menudo a microservicios.

BBVA

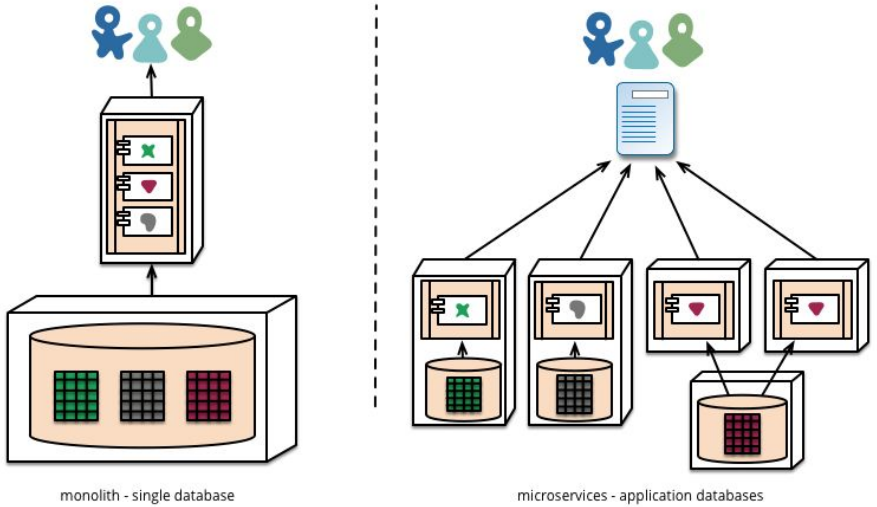
Gobierno Descentralizado

- Un gobierno centralizado tiende a estandarizar en plataformas tecnológicas únicas. Este enfoque es restrictivo.
- En una arquitectura de microservicios se puede usar la herramienta adecuada para cada trabajo.
- El hecho de que puedas hacer algo, no significa que debas hacerlo.
- Se evitan el tipo de normas rígidas establecidas aplicadas por grupos de arquitectura, se usan de estándares abiertos.
- Los estándares surgen del desarrollo una vez probados, se comparten con un grupo más amplio utilizando modelos de open source interno.

BBVA

Gestión de datos descentralizada

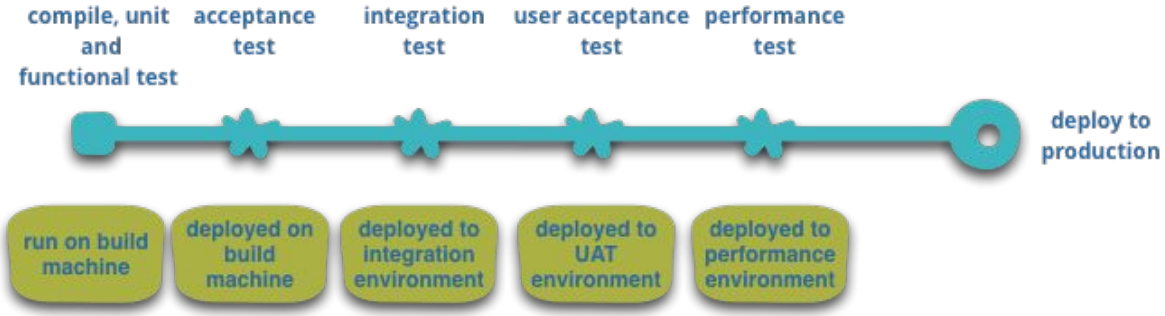
- Se descentralizan las decisiones sobre los almacenes de datos.
- Cada servicio gestionar su propia base de datos. Persistencia Políglota.
- Las transacciones distribuidas son notoriamente difíciles de implementar y como consecuencia las arquitecturas de microservicios enfatizan la coordinación no transaccional entre servicios.
- La consistencia entre microservicios puede ser sólo consistencia final y los problemas son tratados con operaciones de compensación.



BBVA

Automatización de infraestructura

- Los equipos que desarrollan microservicios hacen un amplio uso de las técnicas de automatización de la infraestructura.
- Continuous delivery, consiste en automatizar todo el proceso de despliegue de una aplicación.
- El objetivo de continuous delivery es hacer el proceso el despliegue aburrido. Cada incremento de código puede llegar potencialmente a producción, siguiente un proceso definido.
- Se usan test automatizados para tener la mayor confianza como sea posible de que nuestro software funciona.
- Infraestructura como código.



BBVA

Diseñar para fallar

RESILIENCE

STATUS

- Up
- Down

STATUS

- All up
- All down
- Mostly up
- Up apart from 1

BBVA

Diseñar para fallar

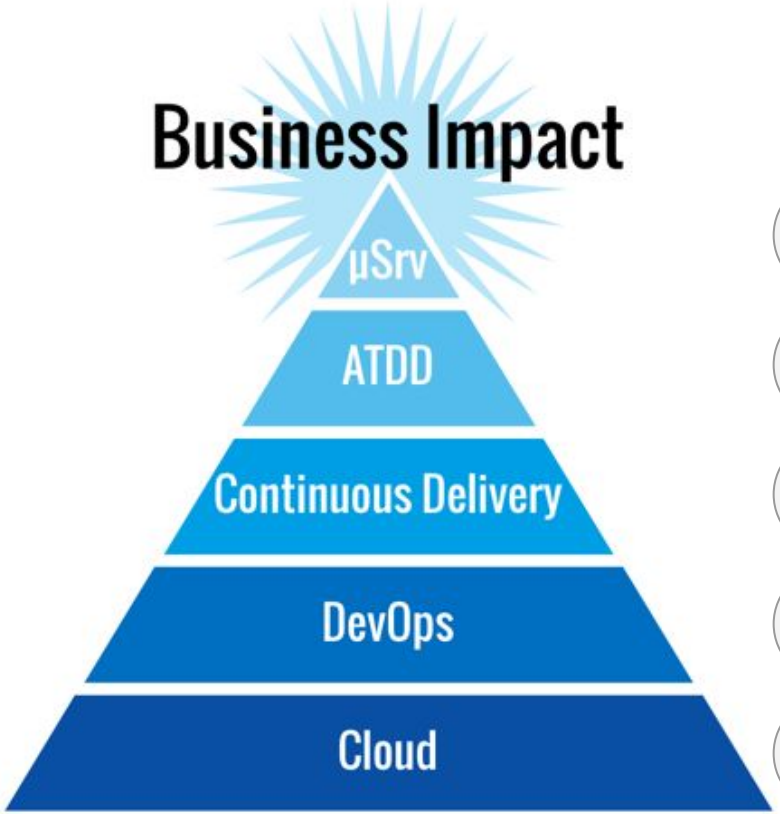
- Las aplicaciones de necesitan ser diseñadas para poder tolerar el fallo de servicios.
- Es importante ser capaz de detectar lo fallos rápido y, si es posible, automáticamente restaurar el servicio
- La monitorización es vital para detectar un mal comportamiento emergente rápidamente para que pueda ser arreglado.
- Debido a la cantidad de procesos un sistema de log centralizado que permita poder consultar fácilmente esa información es imprescindible.
- El Semantic monitoring puede proporcionar un sistema de alerta temprana de que algo va mal que dispara a los equipos de desarrollo a seguir e investigar.

BBVA

Diseño evolutivo

- La descomposición es una herramienta que permite controlar el cambio en una aplicación sin ralentizar el mismo.
- Se pueden realizar cambios frecuentes, rápidos y bien controlados en el software.
- La propiedad clave de un componente es la noción de reemplazado independiente y capacidad de actualización.
- Muchos servicios serán desechados y no evolucionarán a largo plazo.
- Las cosas que cambian al mismo tiempo deben estar en el mismo módulo.

BBVA



What?

- Small, independently deployable and scalable services
- Integrated via lightweight mechanisms
- Responsible for a specific business domain

- Collaboration.
- Specification by example.
- Automated regression testing & living documentation

- No manual intervention for safer automated deployments.
- Faster SW lifecycle.
- Version control everything.

- You build it, you run it.
- Build quality in.
- Design for operation

- Self service infrastructure.
- Heterogeneous and decoupled Tennant.
- Infrastructure as code via APIs

2 Cuando deberían usarse

BBVA

Proporcionan beneficios...

- Límites de módulo fuertes
  - Los microservicios refuerzan la estructura modular, lo que es particularmente importante para equipos más grandes.
- Despliegue independiente
  - Los servicios simples son sencillos de desplegar, y como son autónomos, es menos probable que causen que el sistema vaya mal.
- Diversidad tecnológica
  - Con los microservicios puedes mezclar múltiples lenguajes, frameworks de despliegue y tecnologías de almacenamiento de datos.

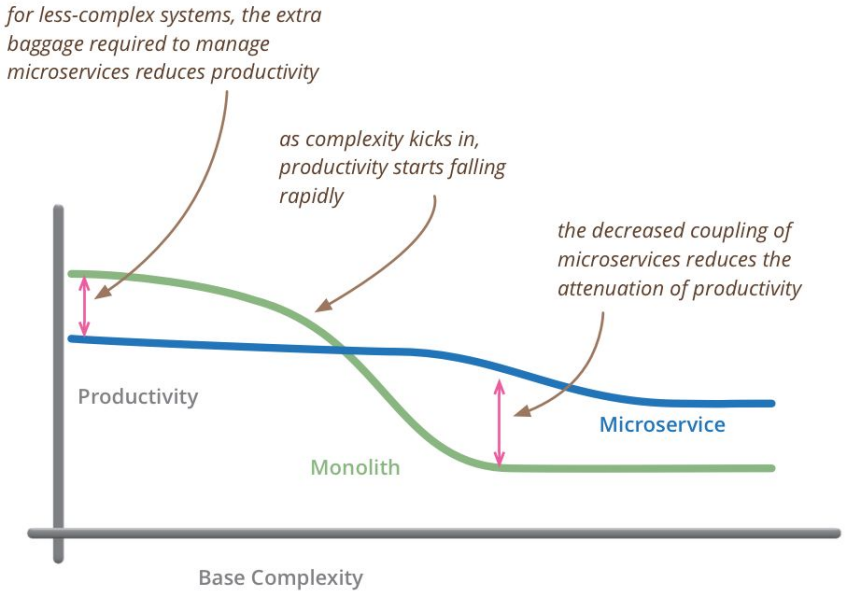
BBVA

...pero tienen costes

- Distribución
  - Los sistemas distribuidos son difíciles de programar, las llamadas remotas son más lentas y son siempre un riesgo de fallo.
- Eventual Consistency
  - Mantener el sistema fuertemente consistente es extremadamente difícil en un sistema distribuido, lo que significa que todo el mundo debe manejar ‘eventual consistency’.
- Complejidad Operacional
  - Se necesita un equipo de operaciones maduro para manejar un montón de servicios, los cuales son desplegados regularmente.

BBVA

- Ni siquiera debe considerarse el uso de microservicios a menos que se tenga un sistema que es demasiado complejo de manejar como un monolito.
- Evitar el ‘microservices envy’. Lo mejor es empezar con uno o dos servicios, para dar tiempo al equipo para ajustar y comprender el nivel adecuado de granularidad.



but remember the skill of the team will outweigh any monolith/microservice choice

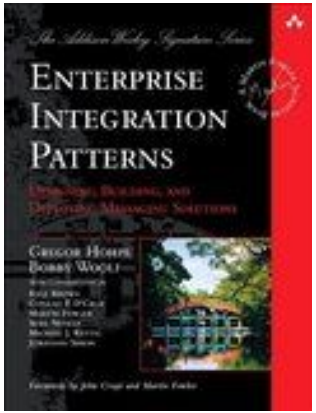
BBVA

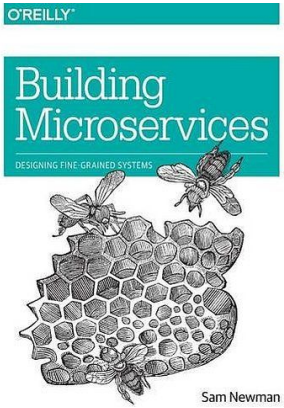
Referencias (Artículos)

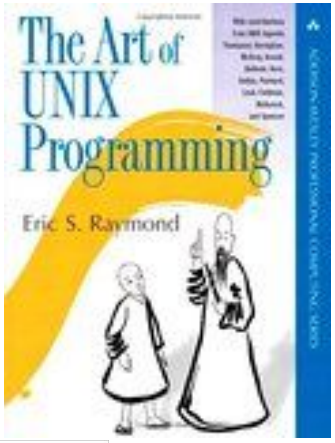
- http://martinfowler.com/articles/microservices.html
- http://martinfowler.com/microservices/
- http://martinfowler.com/bliki/MonolithFirst.html
- http://martinfowler.com/bliki/SacrificialArchitecture.html
- http://www.infoq.com/articles/seven-uservices-antipatterns
- http://highscalability.com/blog/2014/4/8/microservices-not-a-free-lunch.html
- https://www.thoughtworks.com/insights/blog/scaling-microservices-event-stream

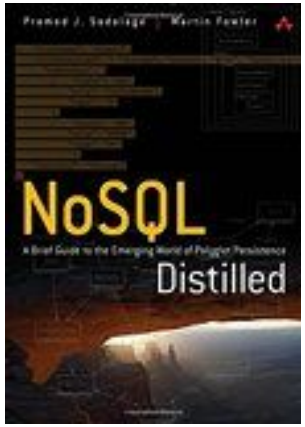
BBVA

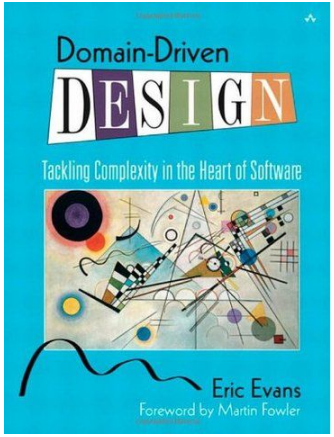
Referencias (Libros)

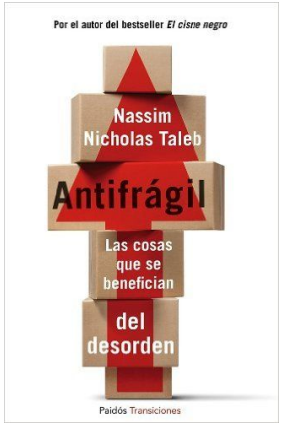


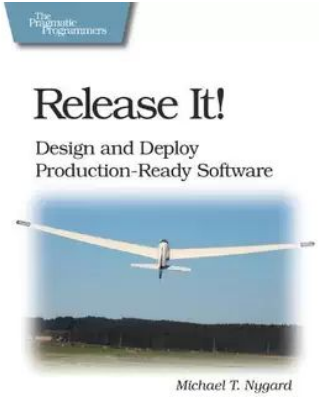


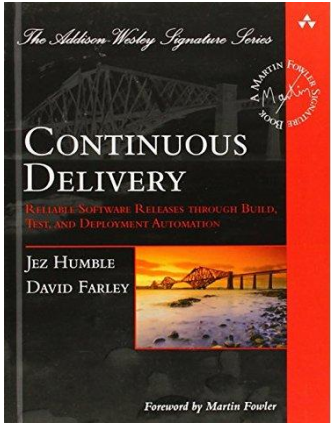












BBVA

Referencias (Presentaciones)

- http://www.infoq.com/presentations/microservices-replaceability-consistency
- http://www.infoq.com/presentations/evolutionary-architecture-microservices-cd
- https://yow.eventer.com/yow-2012-1012/micro-services-architecture-by-fred-george-1286
- https://www.youtube.com/watch?v=hsoovFbpAoE
- https://www.thoughtworks.com/insights/blog/microservices-nutshell