

BIENVENIDOS  
AL CURSO:

# Arquitectura de Microservicios en Net

SESIÓN 01





01

¿Qué es arquitectura de aplicación?

---

02

Aplicaciones monolíticas (demo simplificada de aplicación).

---

03

RPC y REST

---

04

Aplicaciones SOA (Demo simplificada de aplicación).

---

05

¿Qué son microservicios?

---



05

¿Qué son microservicios?

---

06

¿Son los microservicios adecuados  
para mi organización?

---



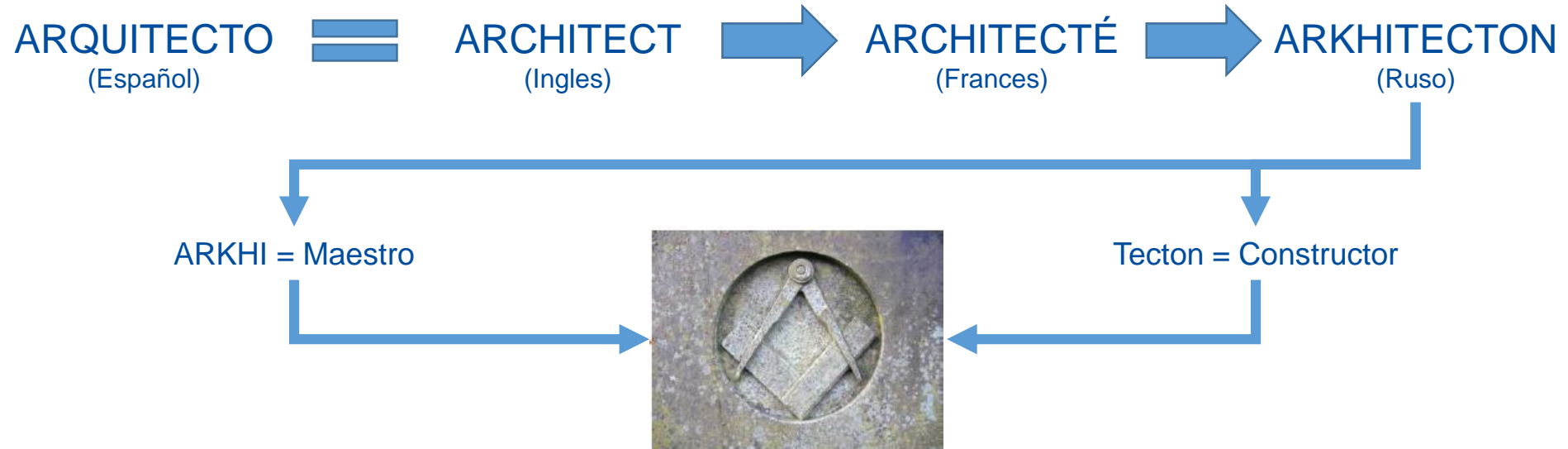
# ¿Qué es arquitectura de aplicación?

## ¿Qué es una empresa?



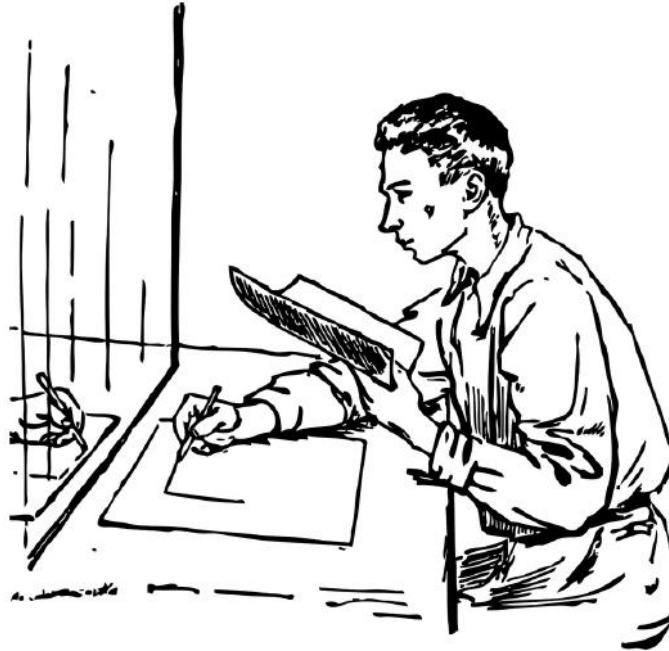
- Una sola organización
- Partes de una gran organización (como una unidad de negocios)
- Una colección de organizaciones que colaboran en una cadena de valor.
- La palabra "Empresa" cubre un amplio espectro de entidades organizacionales

## Análisis etimológico de la arquitectura



La arquitectura es el oficio de los maestros constructores

## Análisis etimológico de la arquitectura



La arquitectura empresarial se puede interpretar como el arte de crear un plan de ejecución para la empresa

■ ¿Qué es arquitectura de aplicación?

## ¿Por qué necesitamos arquitectura empresarial?

Históricamente, surgió como un mecanismo para manejar la complejidad de la implementación de sistemas de TI



En el camino, el papel de EA se transformó para abordar la arquitectura de toda la empresa en lugar de solo los componentes de TI



## Interesados



Desarrollador,  
Administrador del  
sistema, Analista  
de sistemas /  
negocios



Gerentes / jefes de  
Proyecto



Técnico / arquitecto  
de soluciones



Arquitecto  
empresarial

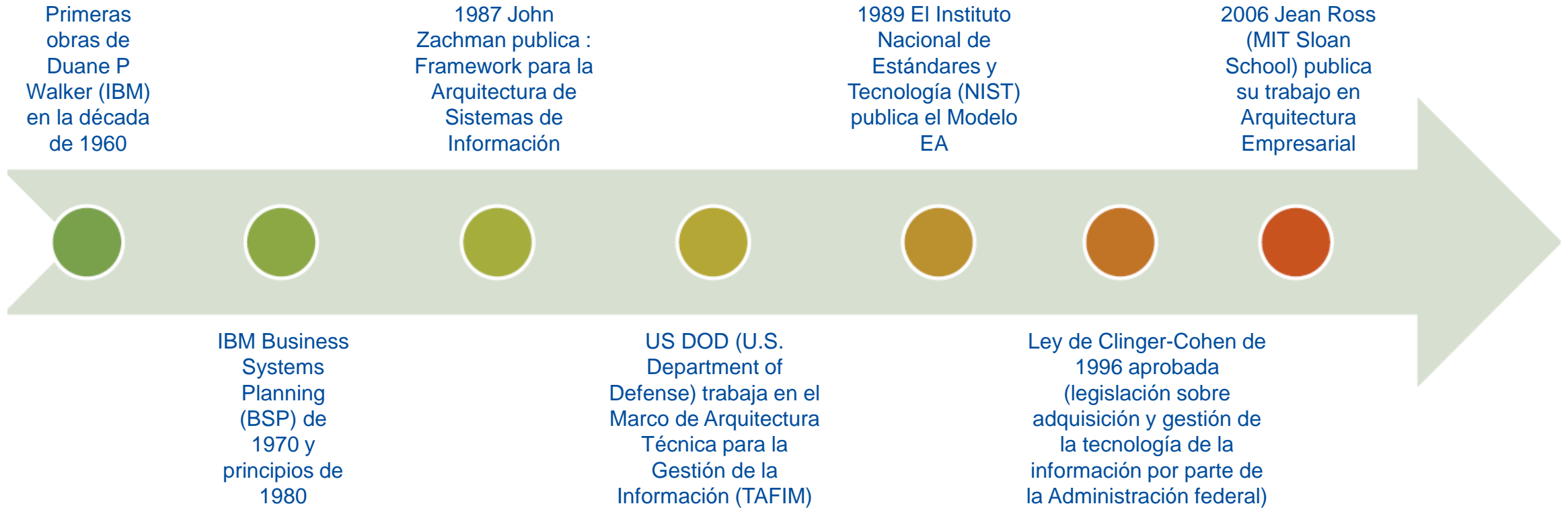


Altos ejecutivos y  
gerentes

■ ¿Qué es arquitectura de aplicación?



## Arquitectura Empresarial línea de tiempo



■ ¿Qué es arquitectura de aplicación?

## Realidad actual

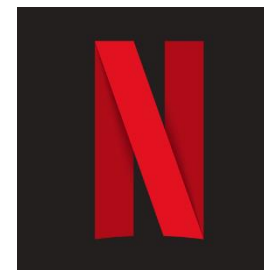


La realidad empresarial actual es significativamente diferente de la de los años sesenta y setenta  
La naturaleza de los problemas, las oportunidades y el entorno empresarial que enfrentan las  
empresas hoy en día son significativamente diferentes a las décadas anteriores.

## ■ ¿Qué es arquitectura de aplicación?



## Realidad actual



Las empresas se han vuelto cada vez más dinámicas y complejas.  
El papel de la TI dentro de las empresas tomó muchos giros inesperados en el camino.  
La mayoría de las empresas ven la TI como una competencia básica fundamental  
El ritmo del cambio solo se está acelerando

■ ¿Qué es arquitectura de aplicación?

# Arquitectura empresarial

La Arquitectura empresarial es una disciplina que permite diseñar la empresa de manera consciente y deliberada, en lugar de dejar que ocurra al azar.

El diseño se basa en la visión empresarial, la intención estratégica y los conocimientos sobre el funcionamiento de la empresa.

**La arquitectura empresarial adopta vistas tanto atómicas como holísticas**

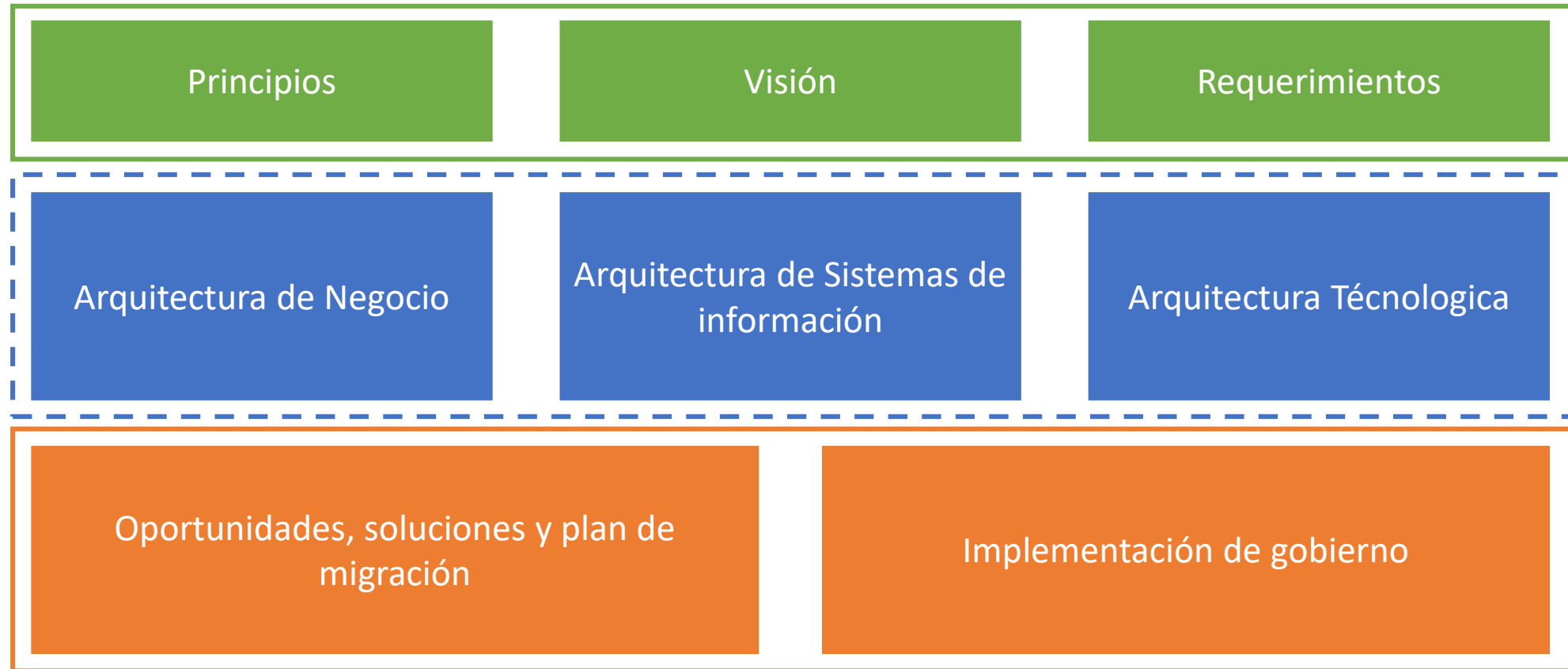


**La arquitectura empresarial adopta vistas tanto atómicas como holísticas**



■ ¿Qué es arquitectura de aplicación?

## Contenido de la arquitectura empresarial

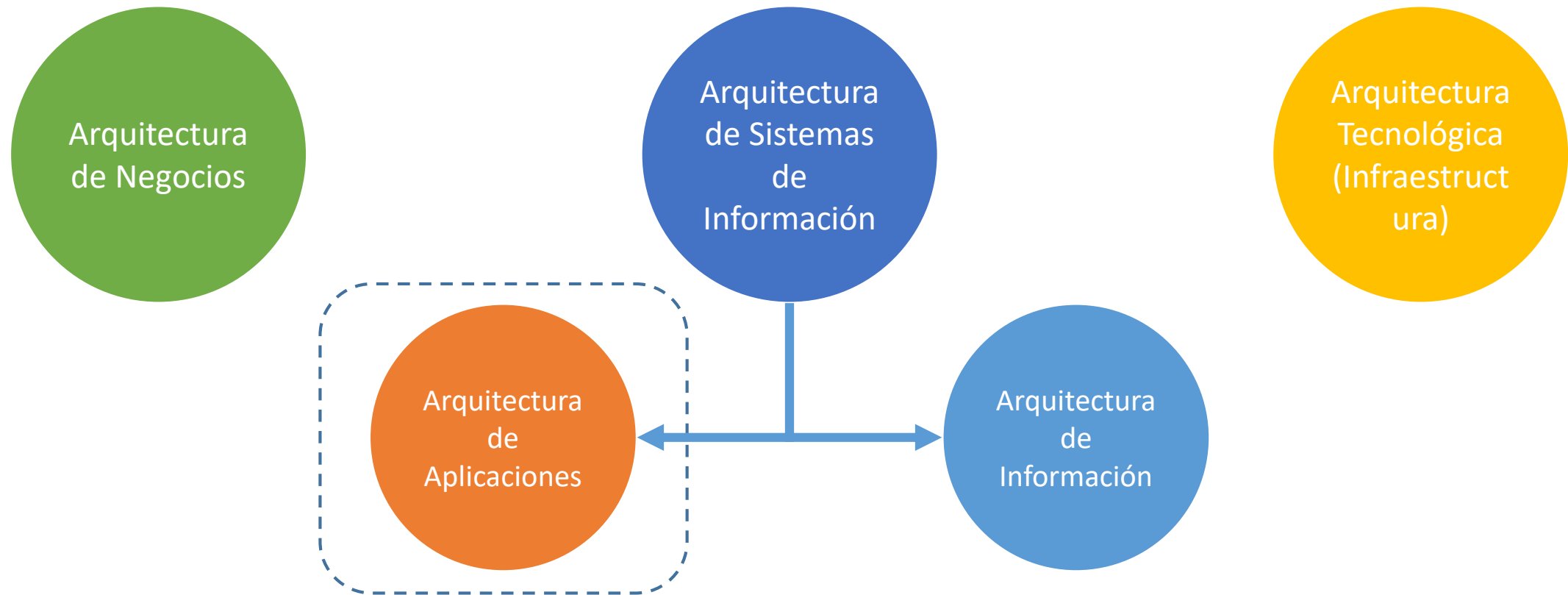


■ ¿Qué es arquitectura de aplicación?





## Dominios de arquitectura empresarial



■ ¿Qué es arquitectura de aplicación?



**En las últimas décadas, las empresas han invertido mucho en aplicaciones comerciales en todos los sectores industriales.**

**■ ¿Qué es arquitectura de aplicación?**



El arquitecto de aplicaciones empresariales crea una hoja de ruta de cartera de aplicaciones de estado objetivo, teniendo en cuenta:

**Costo total de cambio**  
**Rendimiento de las inversiones**  
**Riesgos y camino de menor resistencia.**

■ ¿Qué es arquitectura de aplicación?

## El estado objetivo podría incluir



- Brechas identificadas en las capacidades de la aplicación
- Decisión de retirar el envejecimiento y las aplicaciones de bajo valor
- Modernización de aplicaciones heredadas pero de alto valor
- Eliminar la redundancia
- Estandarización en plataforma tecnológica común
- Consolidando aplicaciones



## ¿Qué es la arquitectura de software?

Cuando las personas en la industria del software hablan de "**arquitectura**", se refieren a una noción definida de los aspectos más importantes del diseño interno de un sistema de software.

Una buena arquitectura es importante, de lo contrario se vuelve más lento y más caro agregar nuevas capacidades en el futuro.

<https://martinfowler.com/architecture/>

## ■ ¿Qué es arquitectura de aplicación?





# Arquitectura de aplicaciones

Las decisiones importantes en el desarrollo de software varían con la escala del contexto en el que estamos pensando. Una escala común es la de una aplicación, por lo tanto, "**arquitectura de aplicación**".

El primer problema con la definición de la arquitectura de aplicación es que no hay una definición clara de lo que es una aplicación. Mi opinión es que las aplicaciones son una construcción social:

- Un cuerpo de código que los desarrolladores ven como una sola unidad
- Un grupo de funcionalidades que los clientes empresariales ven como una sola unidad
- Una iniciativa que los que tienen el dinero ven como un presupuesto único

Una definición tan suelta conduce a muchos tamaños potenciales de una aplicación, que varían de unas pocas a unos pocos cientos de personas en el equipo de desarrollo. (Te darás cuenta de que miro el tamaño como la cantidad de personas involucradas, que creo que es la forma más útil de medir esas cosas.) La diferencia clave entre esto y la arquitectura empresarial es que hay un grado significativo de propósito unificado en torno a la construcción social.

<https://martinfowler.com/architecture/>

## ■ ¿Qué es arquitectura de aplicación?

## Arquitectura de aplicaciones y arquitectura empresarial

Mientras que la arquitectura de aplicaciones se concentra en la arquitectura dentro de algún tipo de límite de aplicación, la arquitectura empresarial mira la arquitectura en una gran empresa.

Tal organización suele ser demasiado grande para agrupar todo su software en cualquier tipo de agrupación cohesiva, por lo que requiere coordinación entre equipos con muchas bases de código, que se han desarrollado de forma aislada entre sí, con financiación y usuarios que operan independientemente de Uno al otro.

Gran parte de la arquitectura empresarial consiste en comprender lo que vale la pena en los costos de la coordinación central y qué forma debe tomar esa coordinación.

<https://martinfowler.com/architecture/>



## Considere la complejidad de :

Codificación a una interfaz

Servicios

Pruebas automatizadas

Domain Driven Design

Acceso a datos

Arquitectura en capas

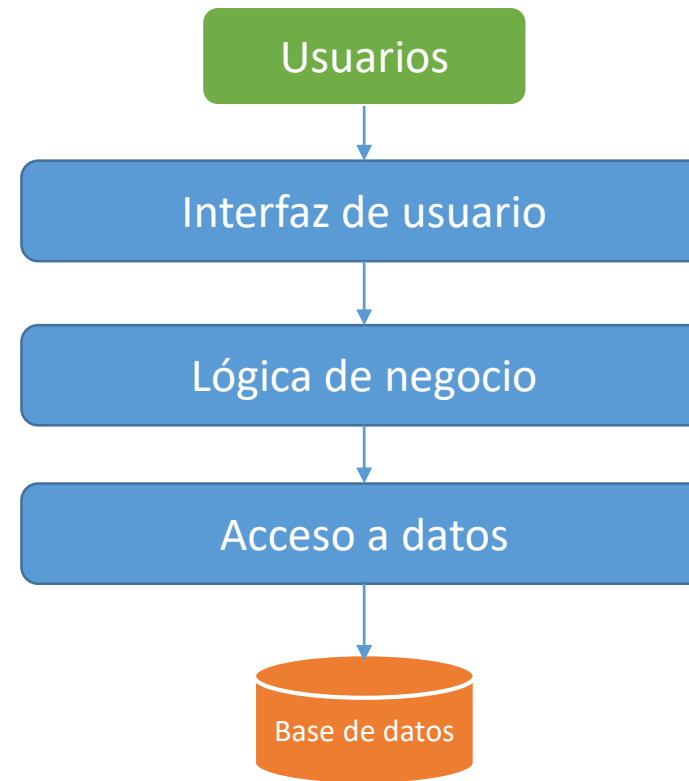
■ ¿Qué es arquitectura de aplicación?





## ¿Qué es la arquitectura de software?

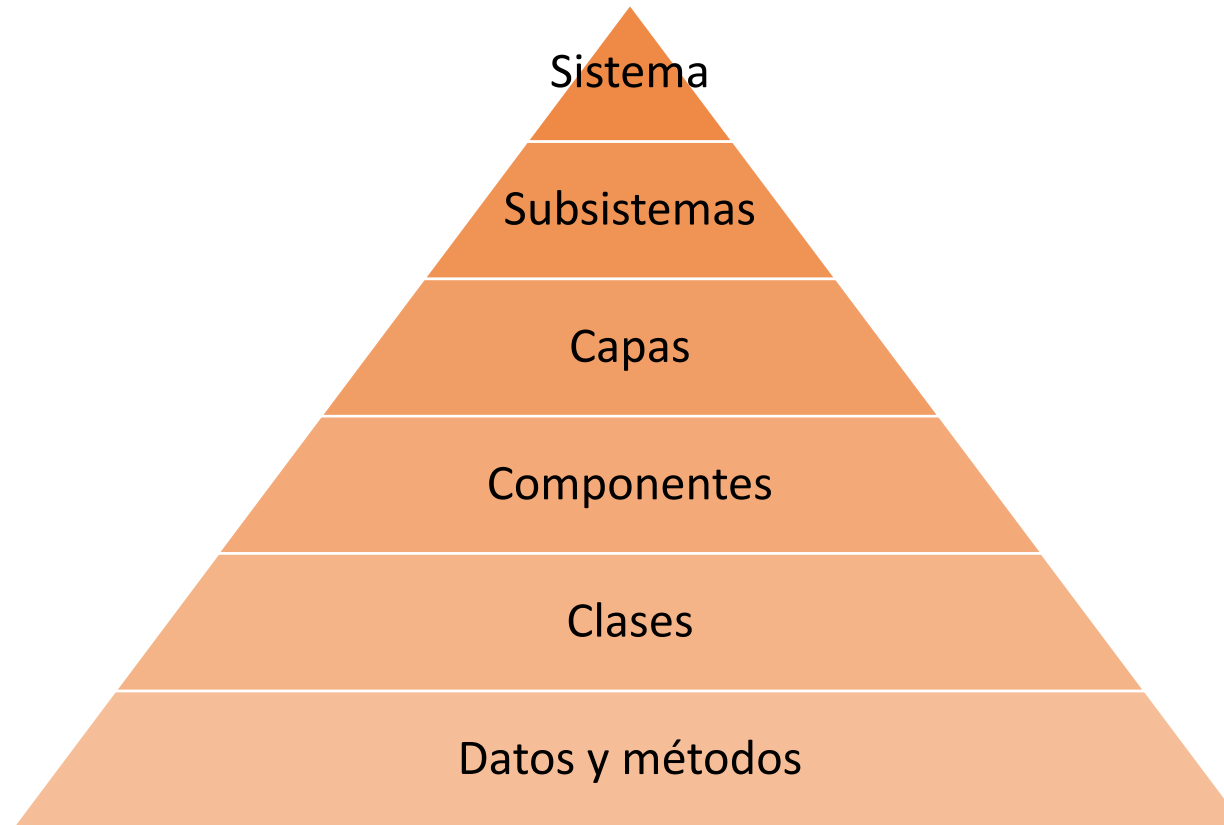
Estructura a alto-nivel de capas, componentes y la relación entre estas.



## ■ ¿Qué es arquitectura de aplicación?



## Niveles de abstracción arquitectónica



■ ¿Qué es arquitectura de aplicación?

## Arquitectura desordenada vs limpia



■ ¿Qué es arquitectura de aplicación?

## ¿Cuándo una arquitectura es mala?



- Compleja
- Incoherente
- Rígido
- Frágil
- Inestable
- Insostenible

## ¿Cuándo una arquitectura es buena?



- Sencilla
- Comprensible
- Flexible
- Emergente
- Testeable
- **Mantenible**

## ¿Qué es la arquitectura limpia?



Habitantes (Usuarios)



Arquitecto



Maquina

## ■ ¿Qué es arquitectura de aplicación?



# Aplicaciones monolíticas



# Aplicación monolítica

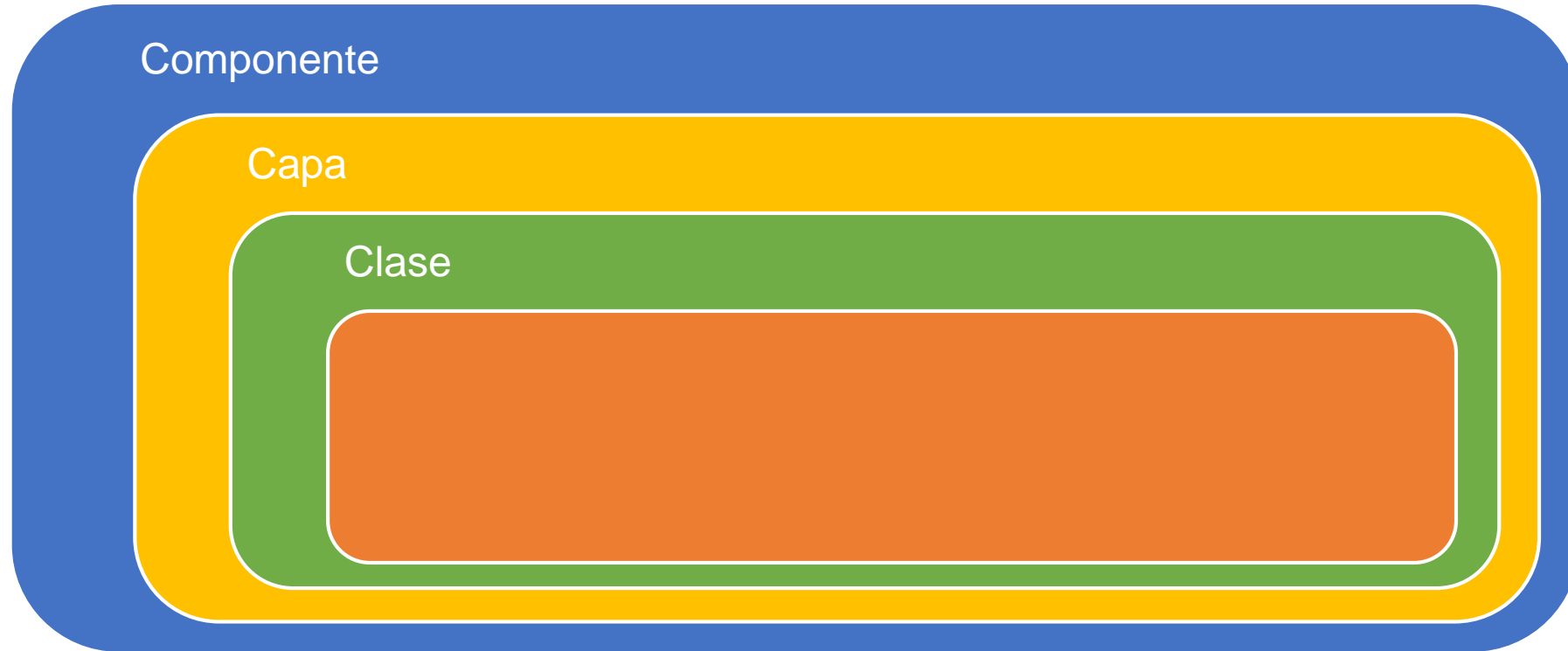
Una aplicación de software de un solo nivel en el que la interfaz de usuario y el código de acceso a datos se combinan en un solo programa desde una sola plataforma





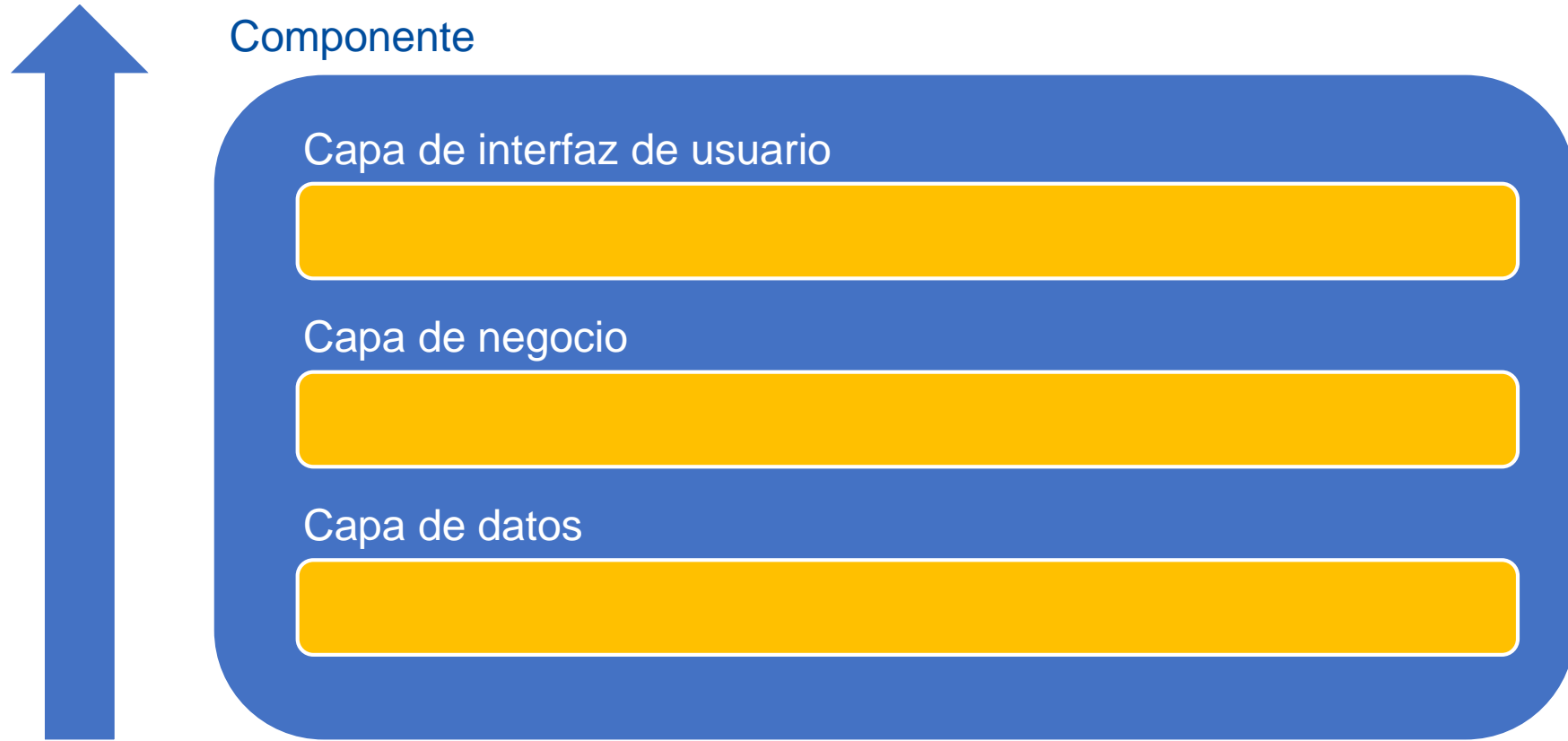
# Arquitectura general de una aplicación

Aplicación



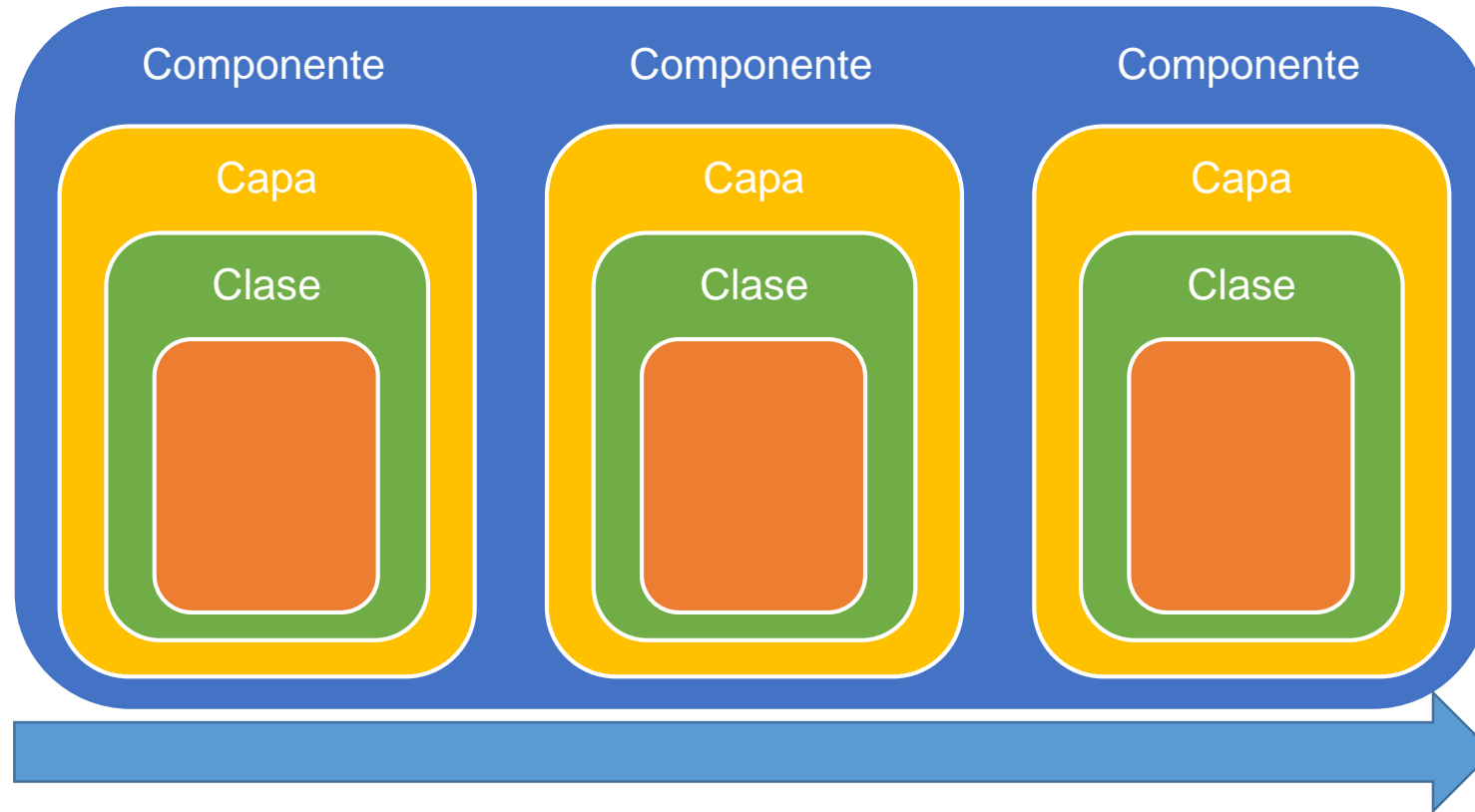


## Acoplamiento vertical



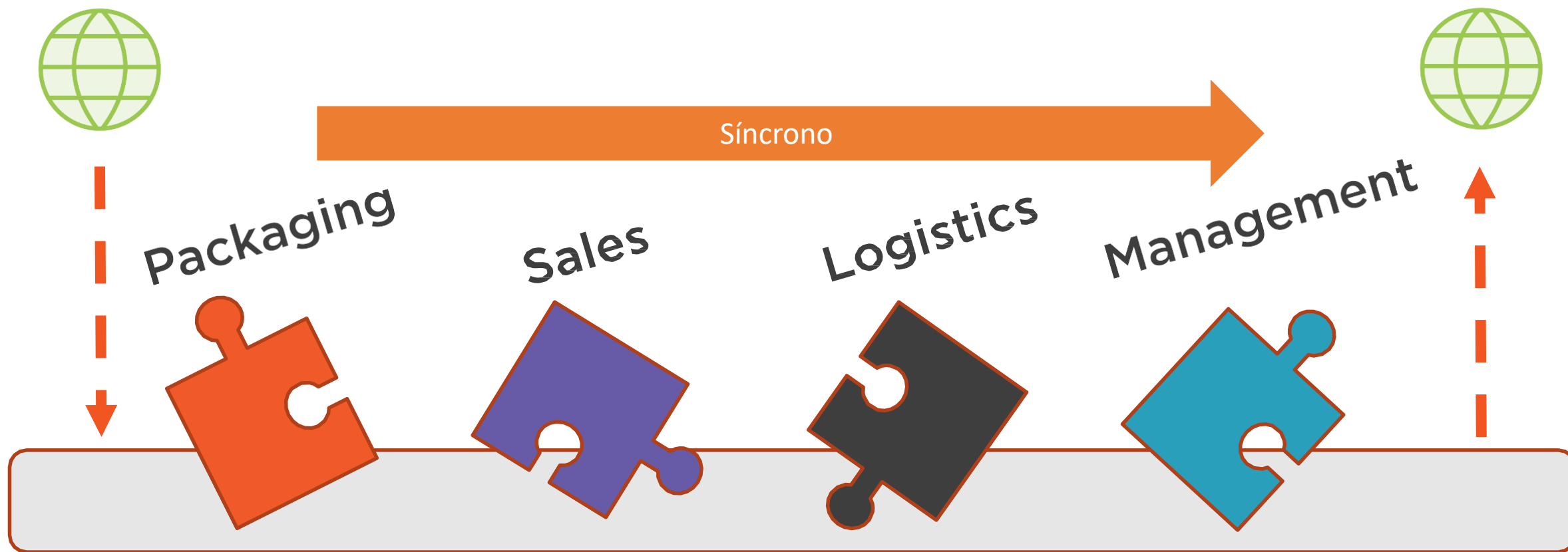
## Acoplamiento horizontal

Aplicación



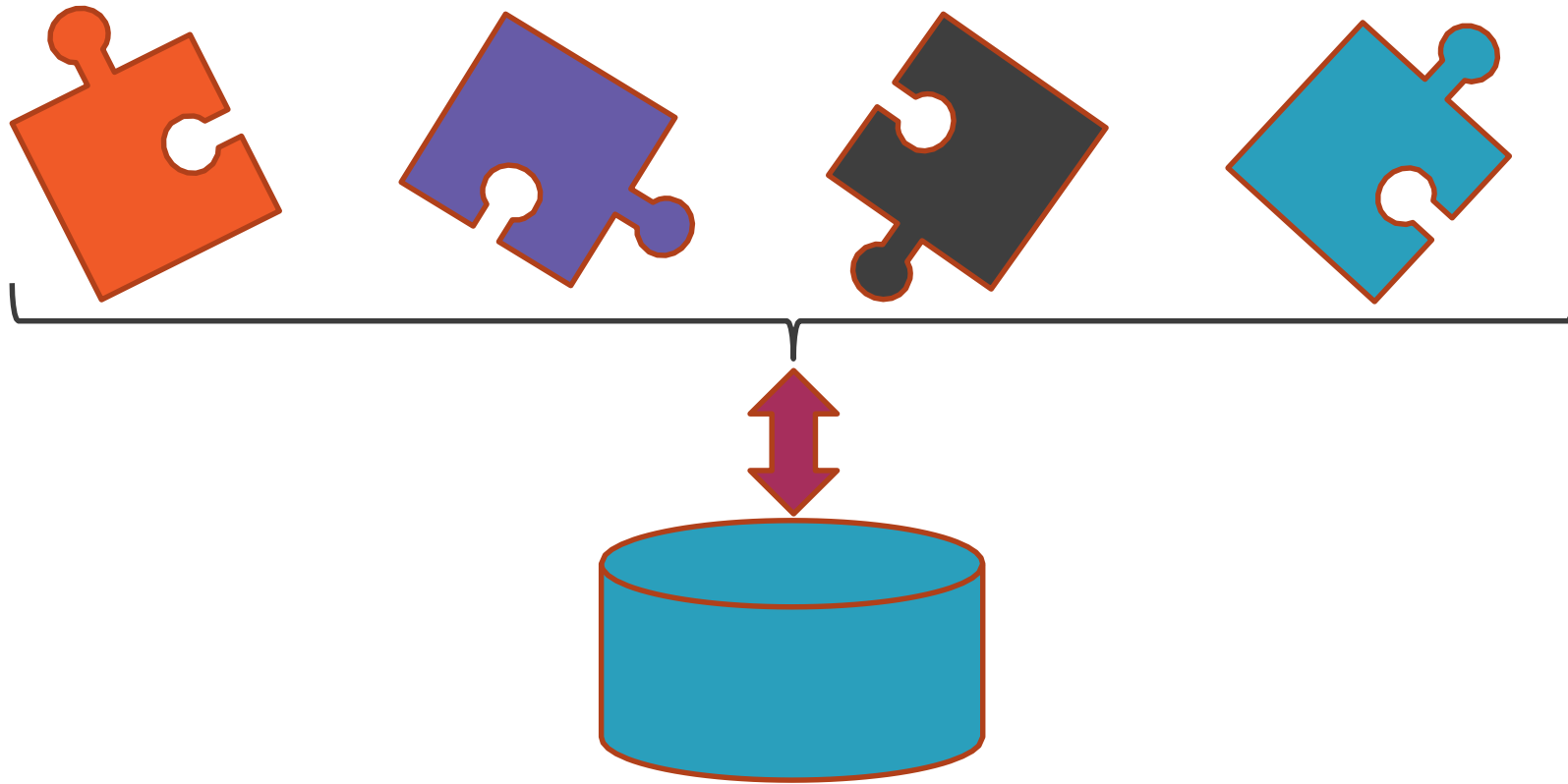


## Manejo de solicitudes





## Manejo de solicitudes





## Beneficios de las aplicaciones monolíticas

Fácil de desplegar

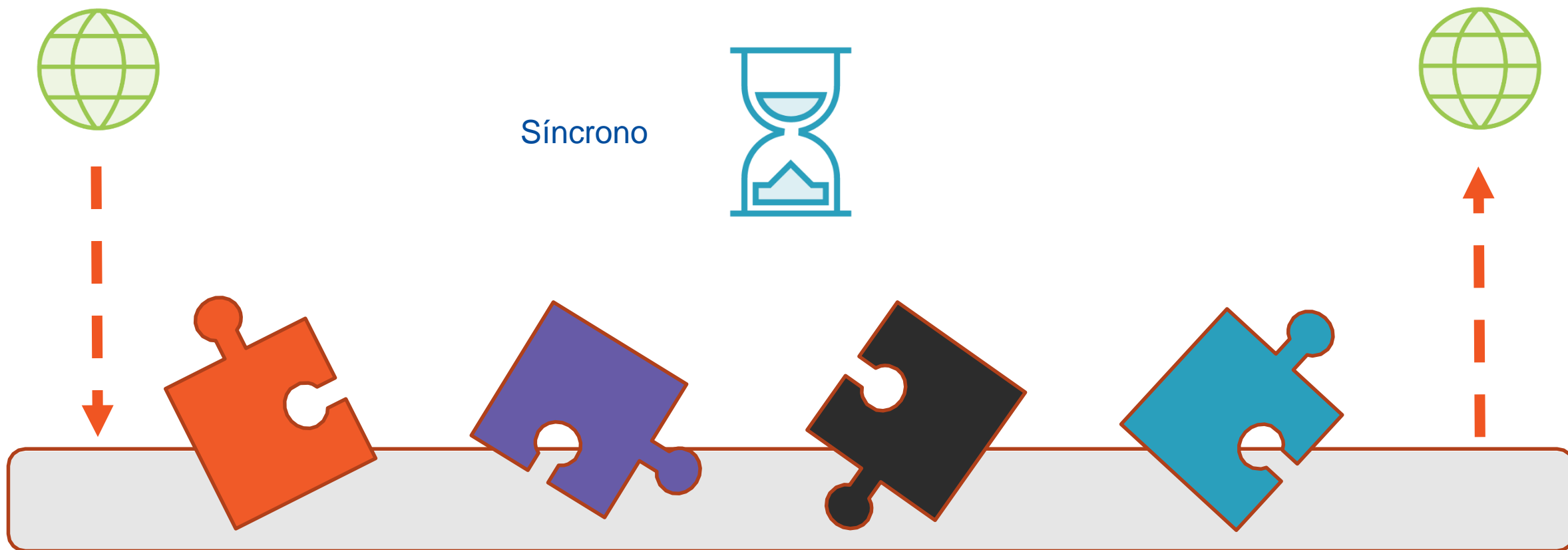
Bien conocidas

Sin dependencias externas

Amigable con los IDE's



## Posibles problemas





## Posibles problemas



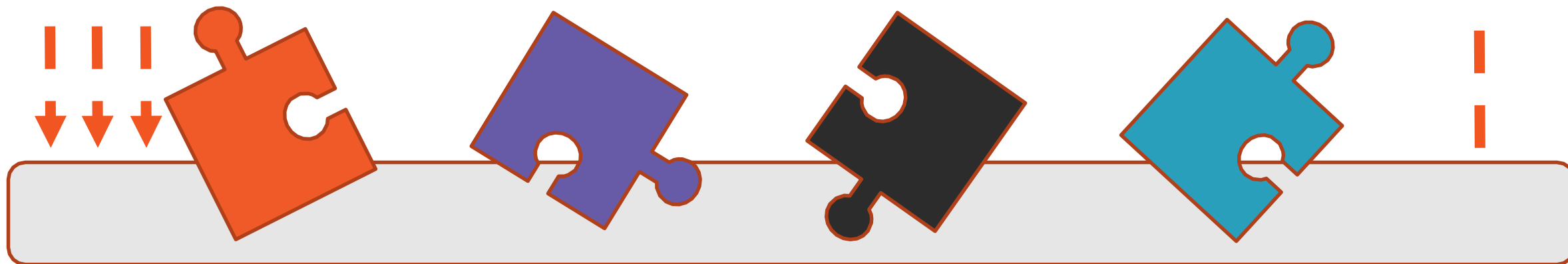




## Posibles problemas



Sobre carga de solicitudes





## Posibles problemas





## Desventajas de las aplicaciones monolíticas

Complejo y difícil de mantener

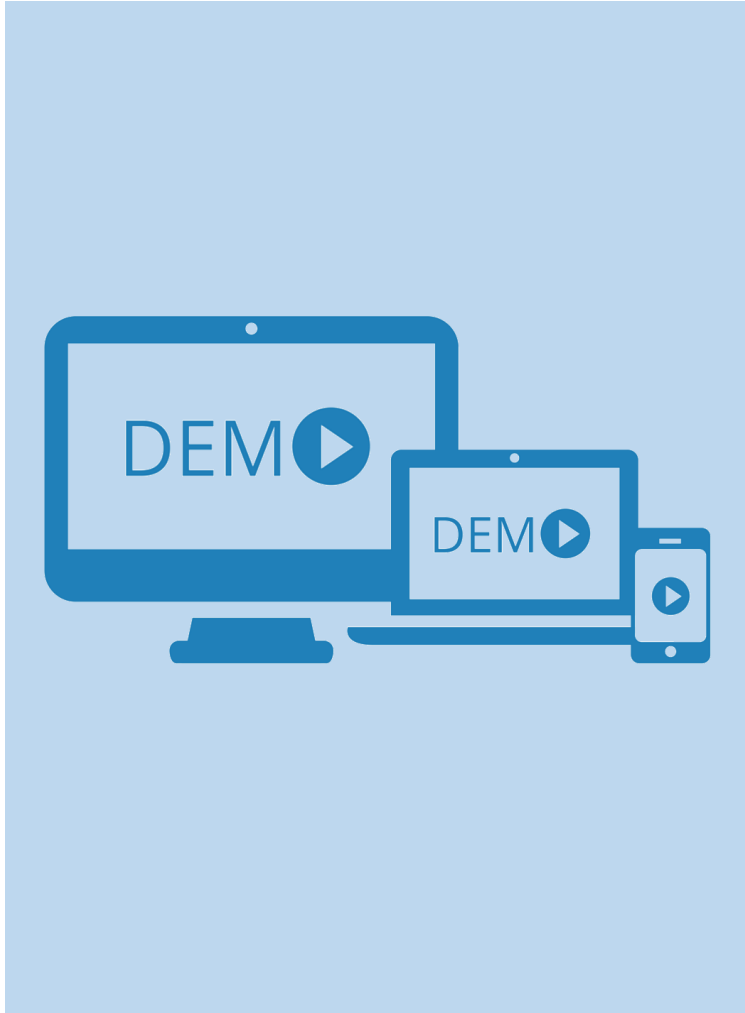
Tiende a complicarse con el tiempo

Despliegue trabajoso  
(Coordinación)

Problemas de rendimiento

Baja confiabilidad (Degradación)

One Stack (Una sola tecnología)



**Demo simplificada**



# Aplicaciones SOA

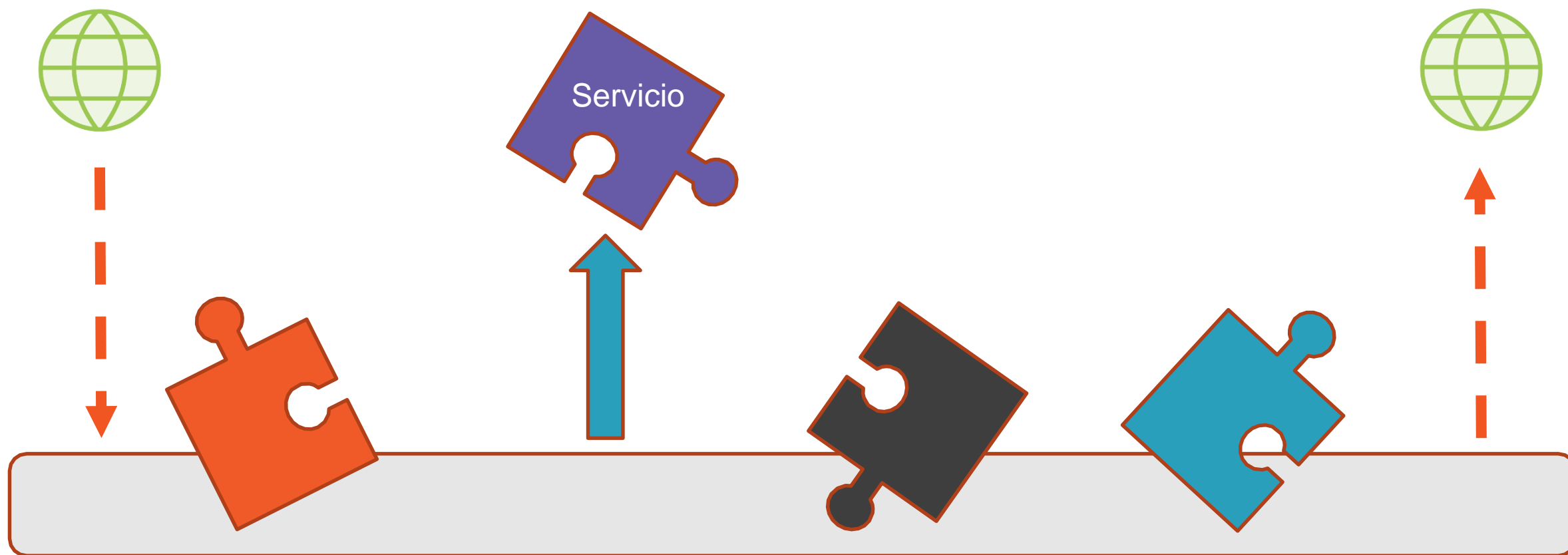


# Sistema Distribuido

Un sistema distribuido es una aplicación de software en la que los componentes están ubicados en computadoras en red y se comunican y coordinan sus acciones emitiendo llamadas o pasando mensajes

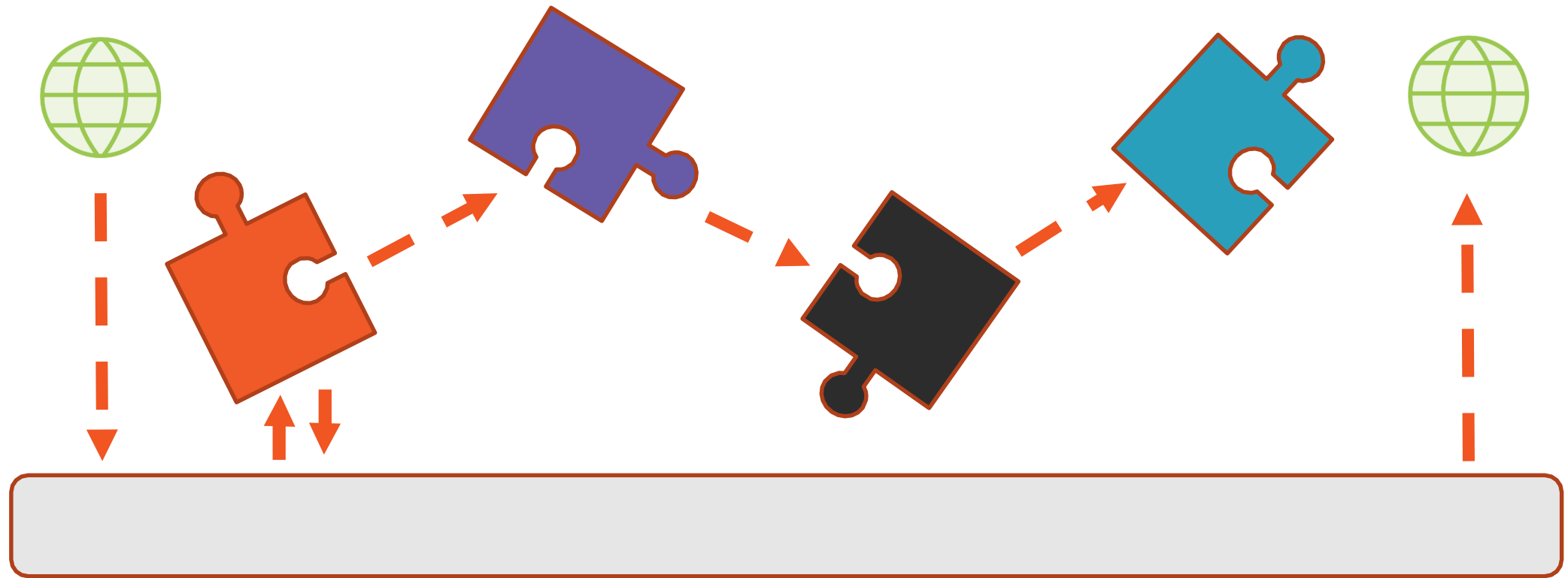


## Componentes distribuidos





## Service-oriented Architecture





## ¿Cuándo una arquitectura es buena?

### Falacias de la computación distribuida

- La red es confiable
- La latencia es cero
- El ancho de banda es infinito
- La red es segura
- La topología no cambiará
- Hay un administrador
- El costo de transporte es cero
- La red es homogénea.



## Acoplamiento

### Plataforma

(Asociado a una misma tecnología)

### Comportamiento

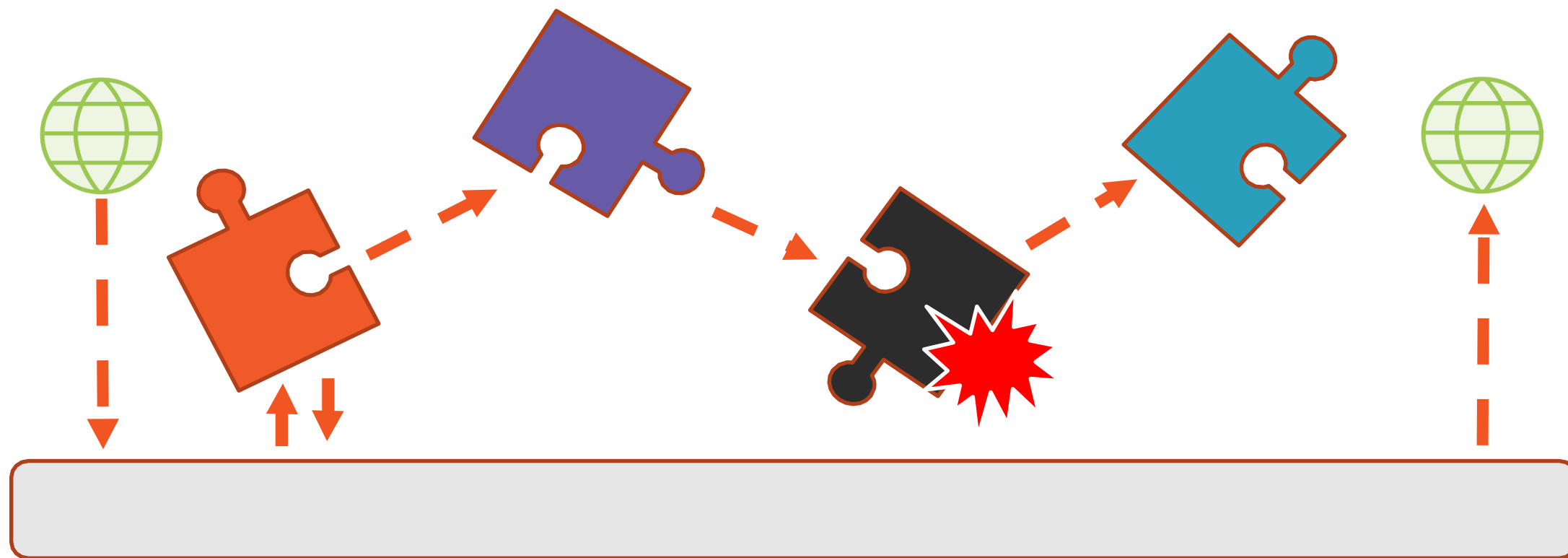
(Conocimiento de la firma)

### Temporal

(Dependencia entre servicios)



## Remote Procedure Call





# RPC y REST.



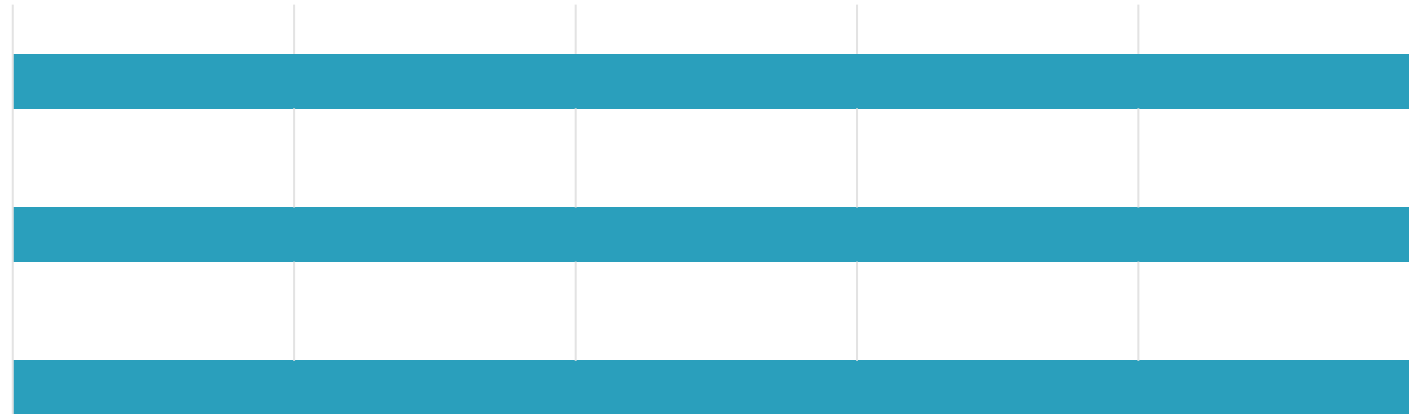
## Remote Procedure Call

Llamar a un metodo de manera remota  
.NET Remoting/Java RMI

Acoplamiento a nivel de  
plataforma

Acoplamiento a nivel de  
comportamiento

Acoplamiento a nivel  
temporal





## Remote Procedure Call

### Remote Procedure Call (RPC)

- Clases de proxy
- ¡Cuidado con las falacias de la informática distribuida!



## Remote Procedure Call: SOAP

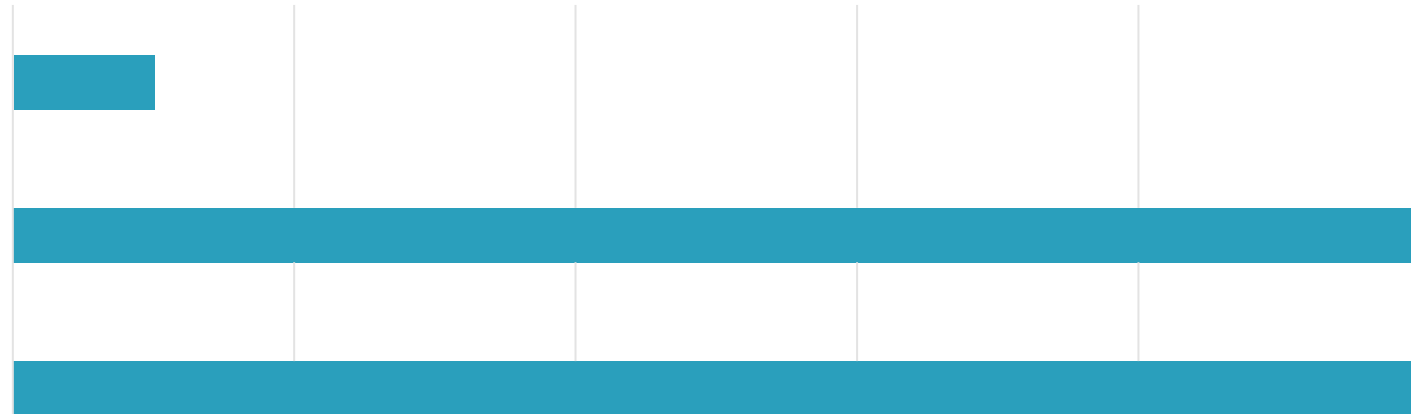
Simple Object Access Protocol  
(SOAP)

Call method using standarized XML  
(Web Services Description Language WSDL)

Acoplamiento a nivel de  
plataforma

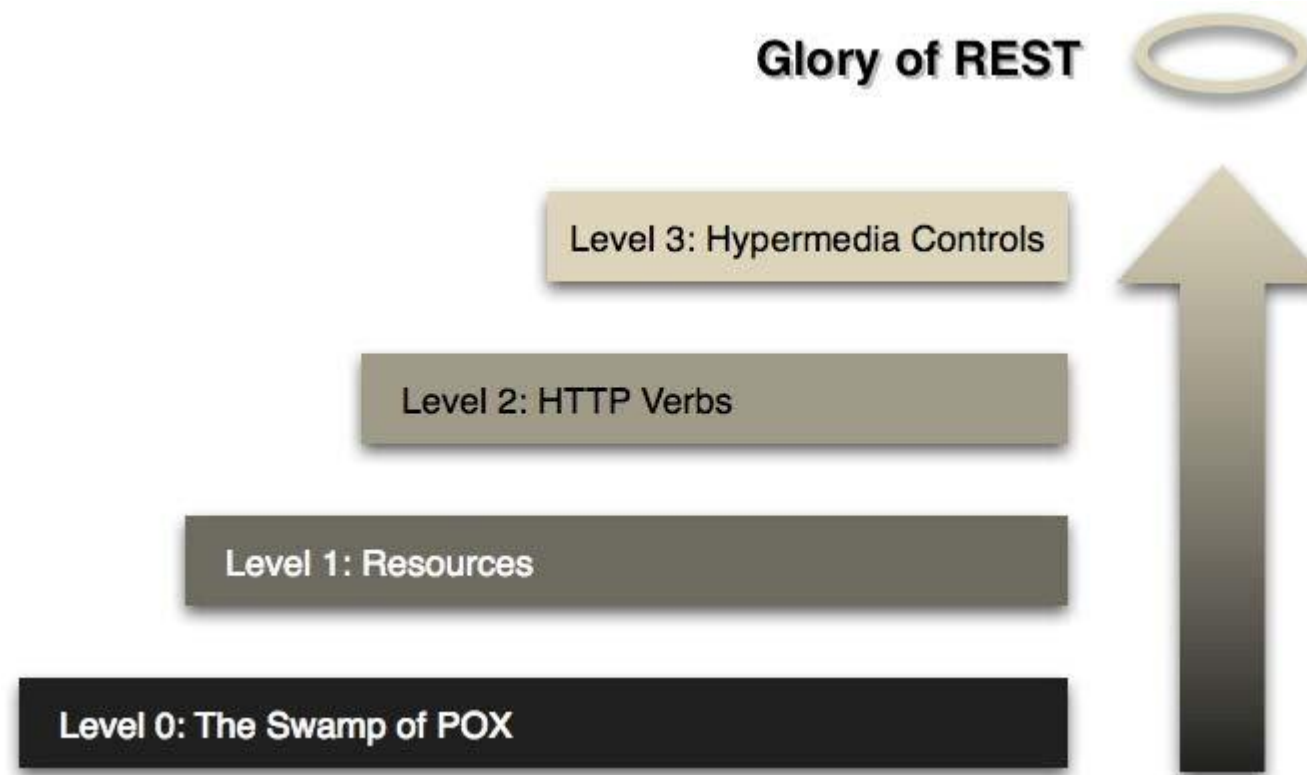
Acoplamiento a nivel de  
comportamiento

Acoplamiento a nivel  
temporal





# Representational State Transfer (REST)

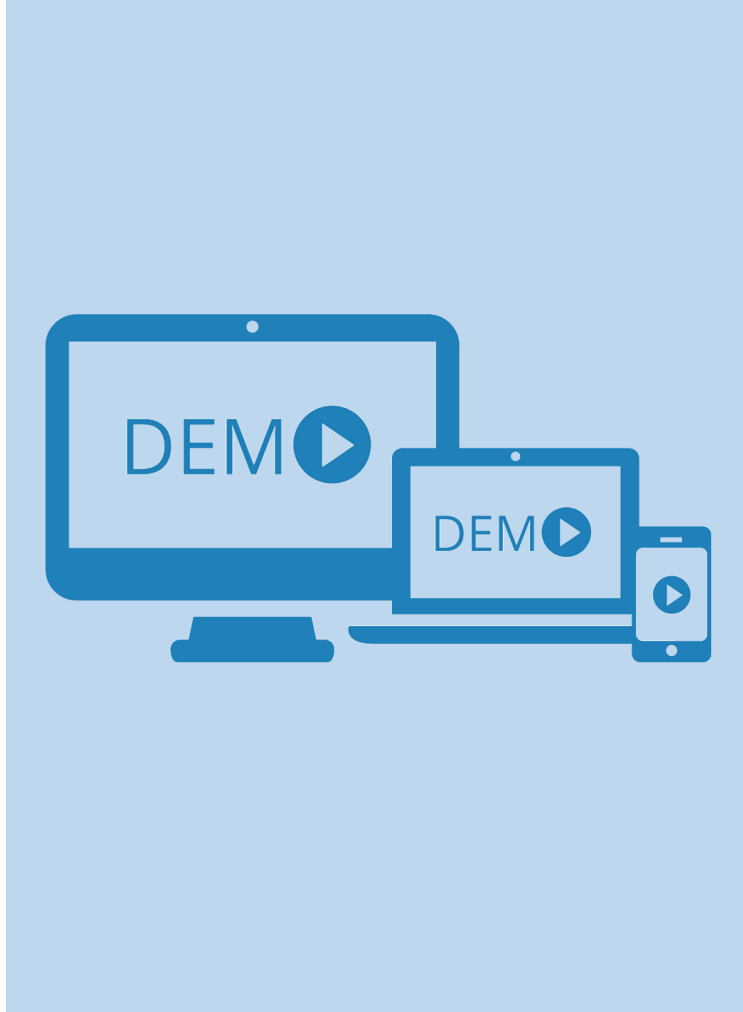






## Representational State Transfer (REST)



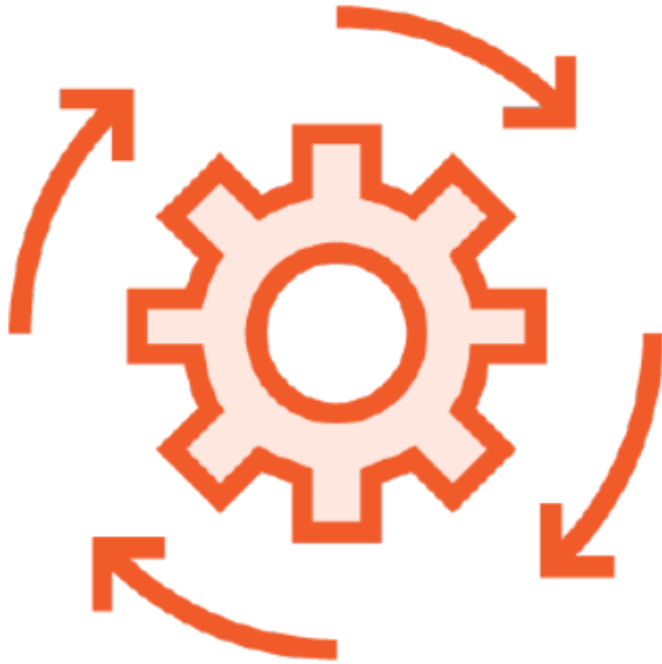


**Demo simplificada**



# ¿Qué son microservicios?

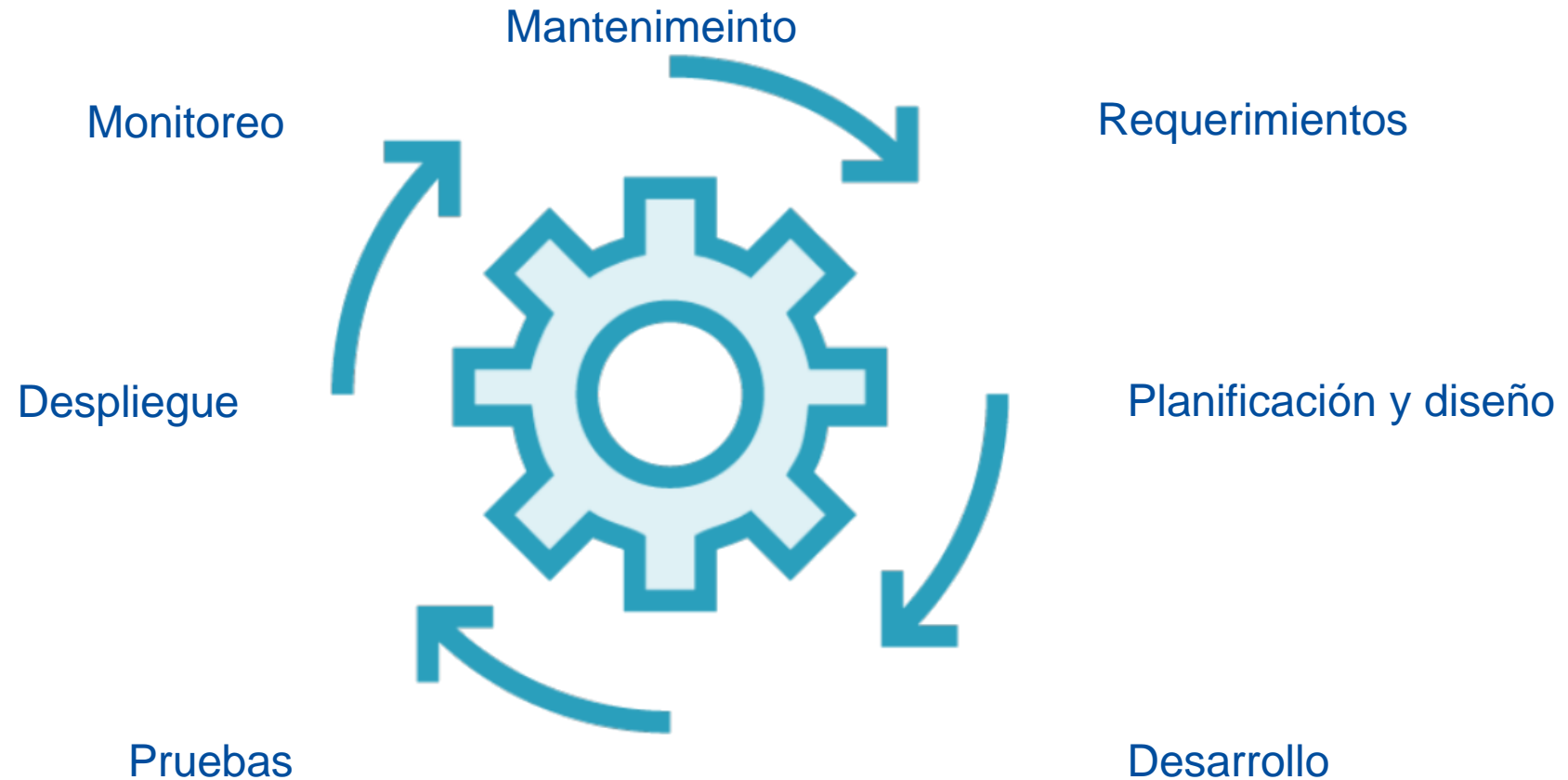
## Ciclo de vida de desarrollo de software



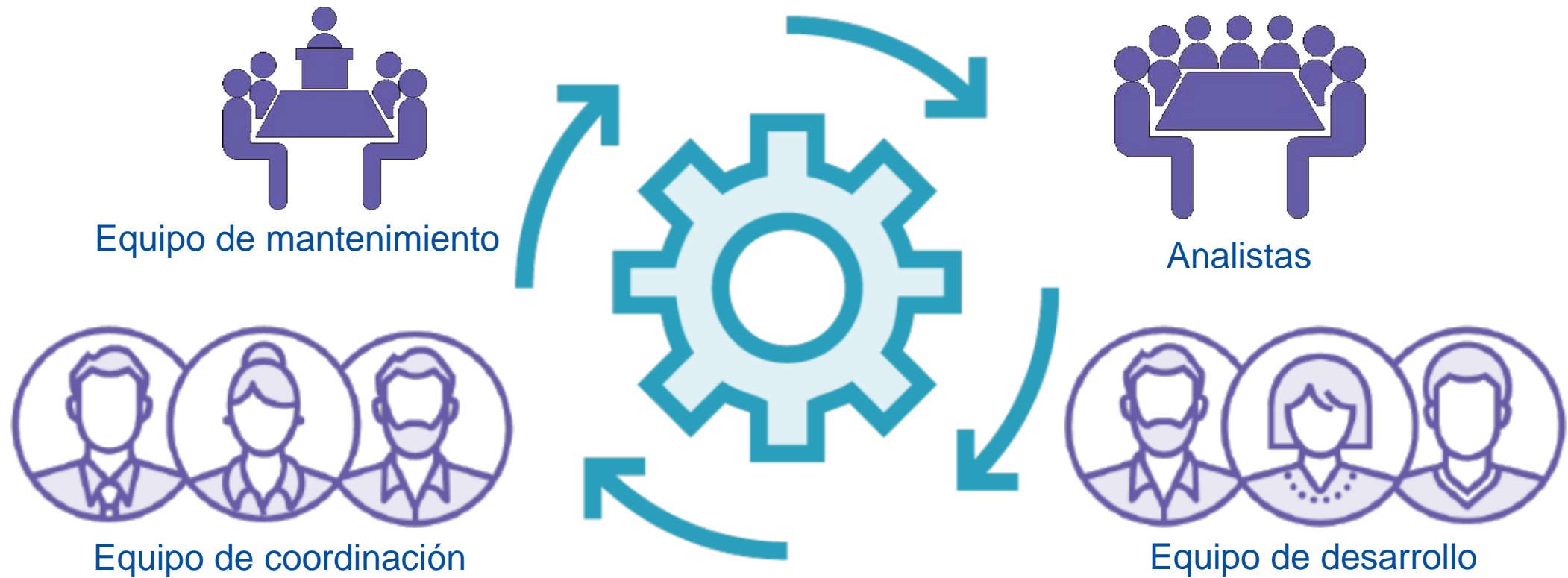
- Ciclo de vida del desarrollo de software
- Del proyecto al producto
- Organización y gestión.
- Desplegado en producción
- Los microservicios tienen un impacto en este ciclo de vida



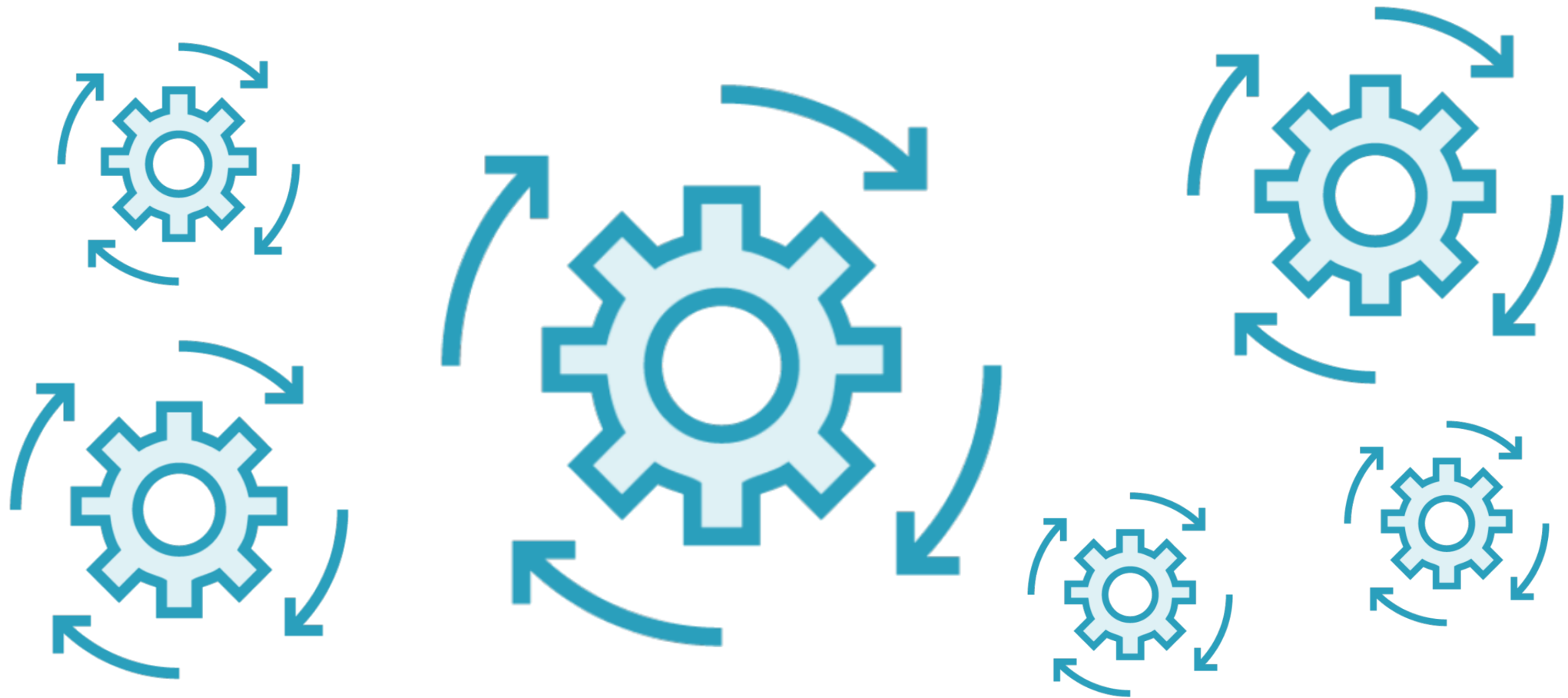
## Ciclo de vida de desarrollo de software



## Ciclo de vida de desarrollo de software

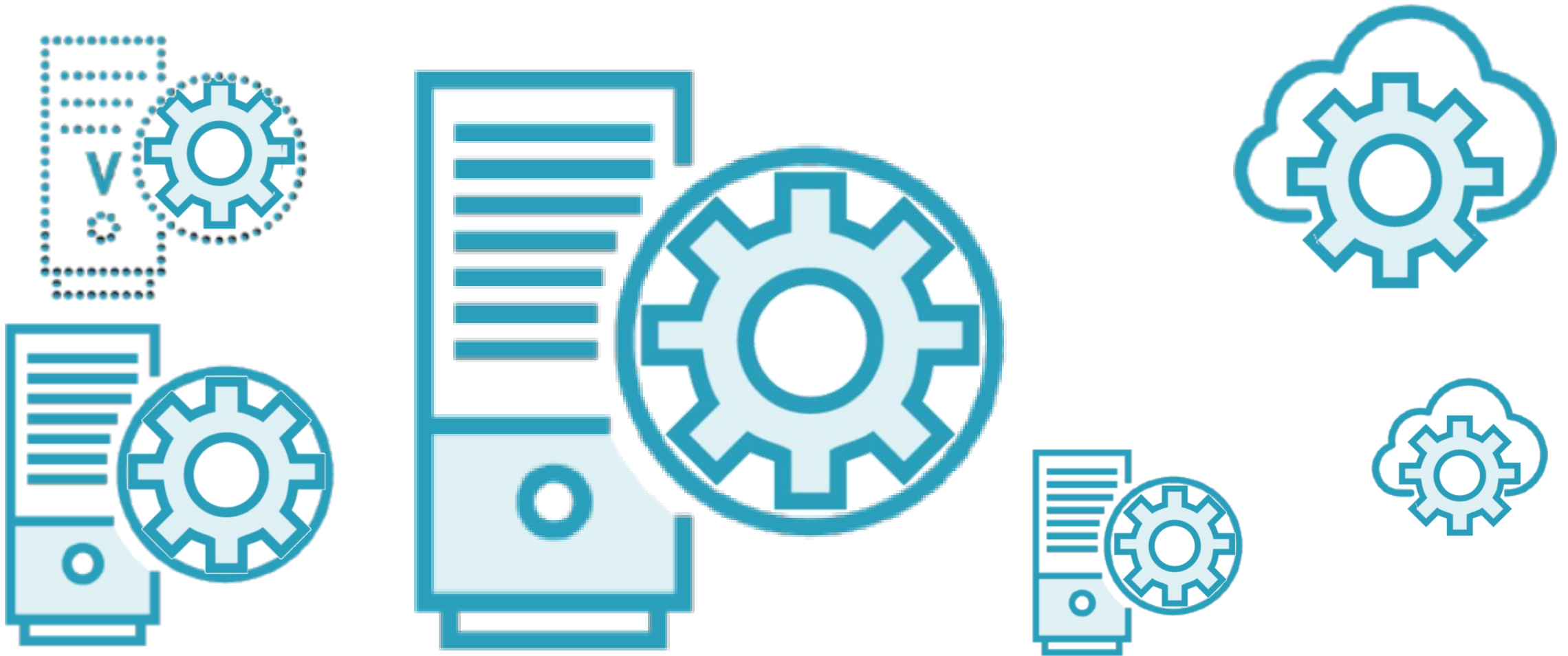


## Ciclo de vida de desarrollo de software



■ ¿Qué son microservicios?

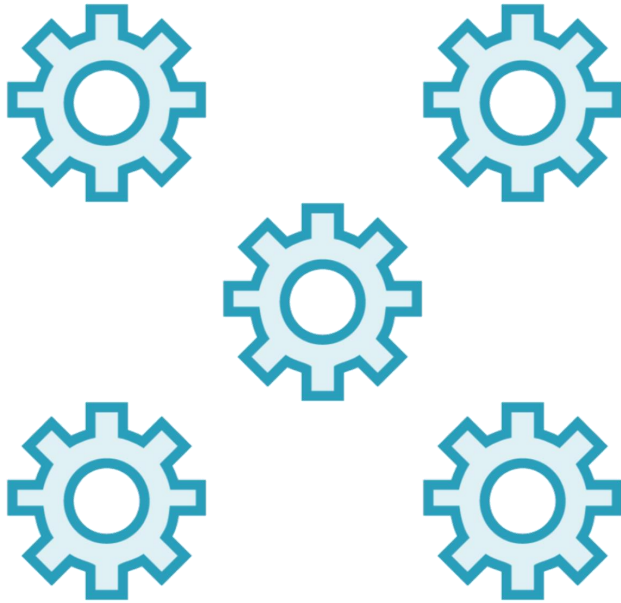
## Ciclo de vida de desarrollo de software



■ ¿Qué son microservicios?



## Microservicios

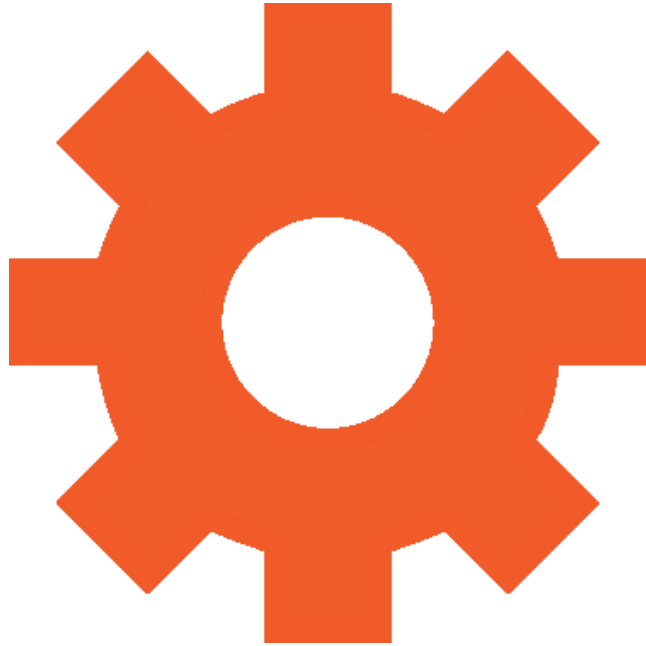


- Conjunto de practicas
- Aumentar la velocidad
- Escala
- Tecnología agnóstica
- Principios y patrones arquitectónicos.

## Micro



- Grande o pequeño
- Ninguna medida universal
- "Hace una cosa"
- Alcance de funcionalidades
- Contexto limitado
- Identificar subdominios



## Servicio

- Componente desplegable independientemente
- Interoperabilidad
- Comunicación basada en mensajes
- Arquitectura orientada a servicios (SOA)



# Microservicio (I)

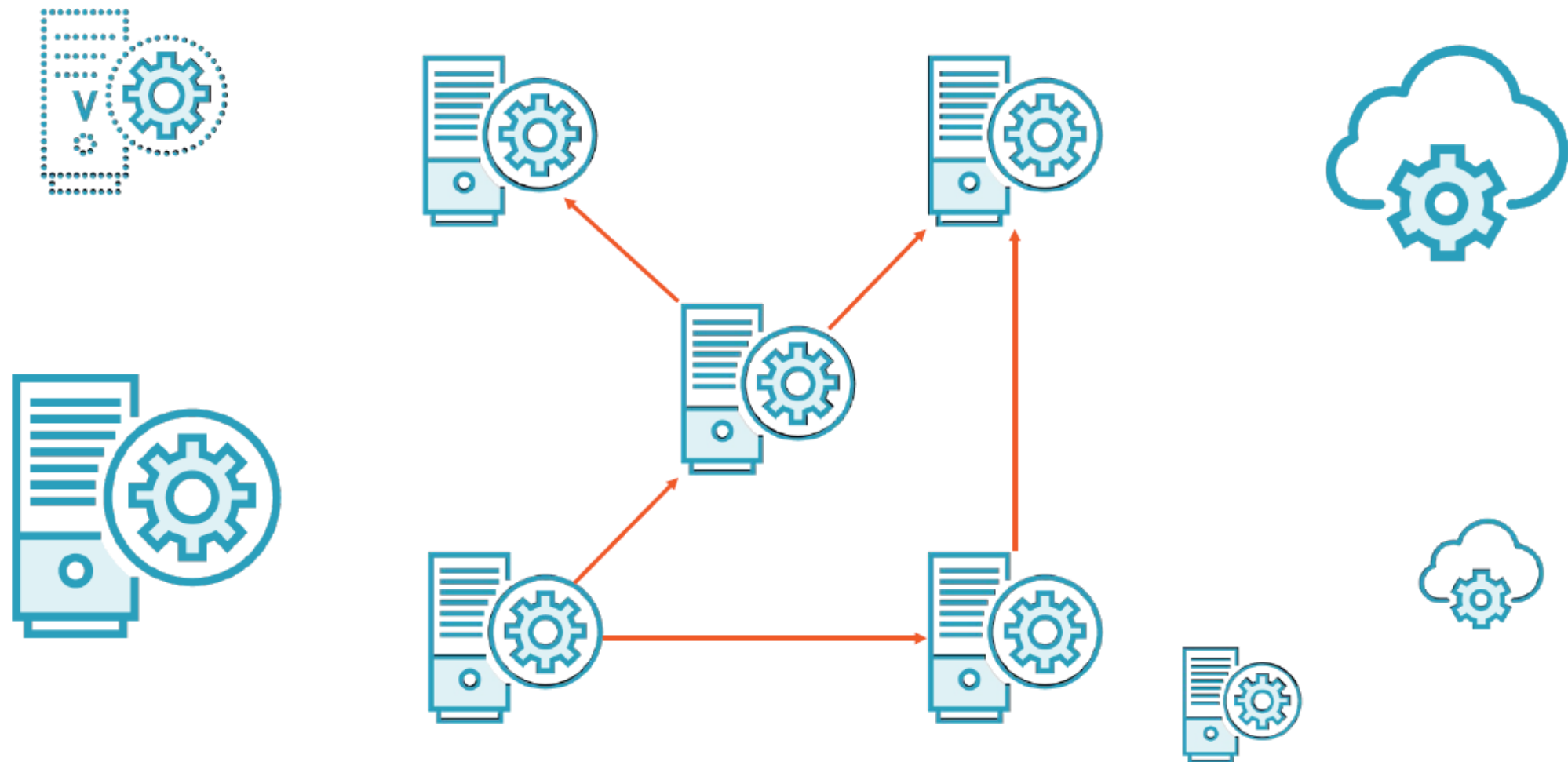
El estilo arquitectónico de microservicios es un enfoque para desarrollar una sola aplicación como un conjunto de pequeños servicios, cada uno de los cuales se ejecuta en su propio proceso y se comunica con mecanismos ligeros.

James Lewis and Martin Fowler, Thoughtworks



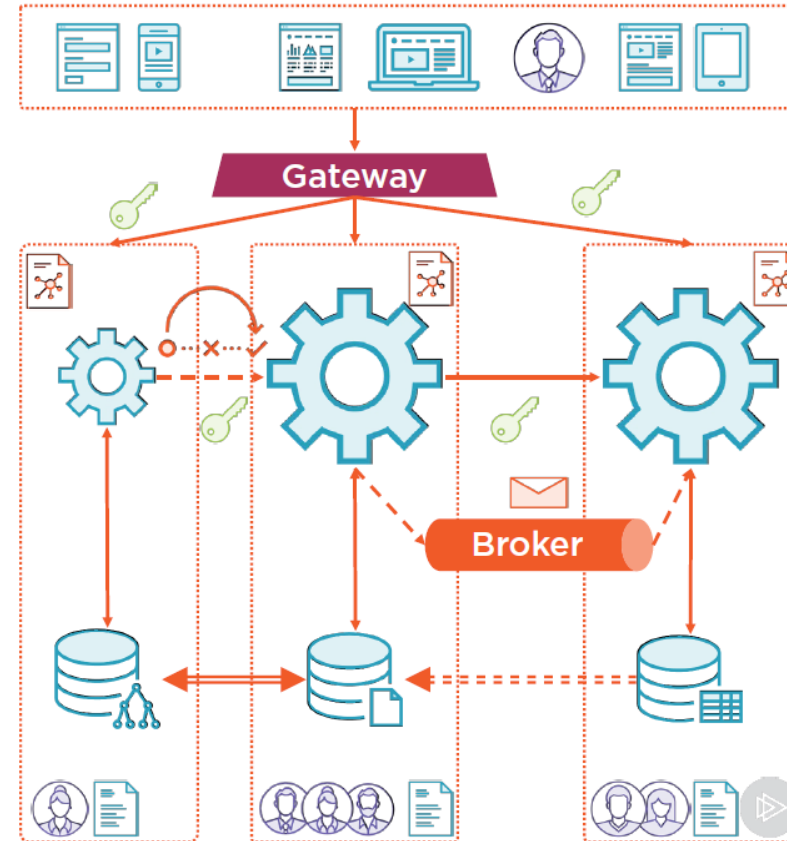
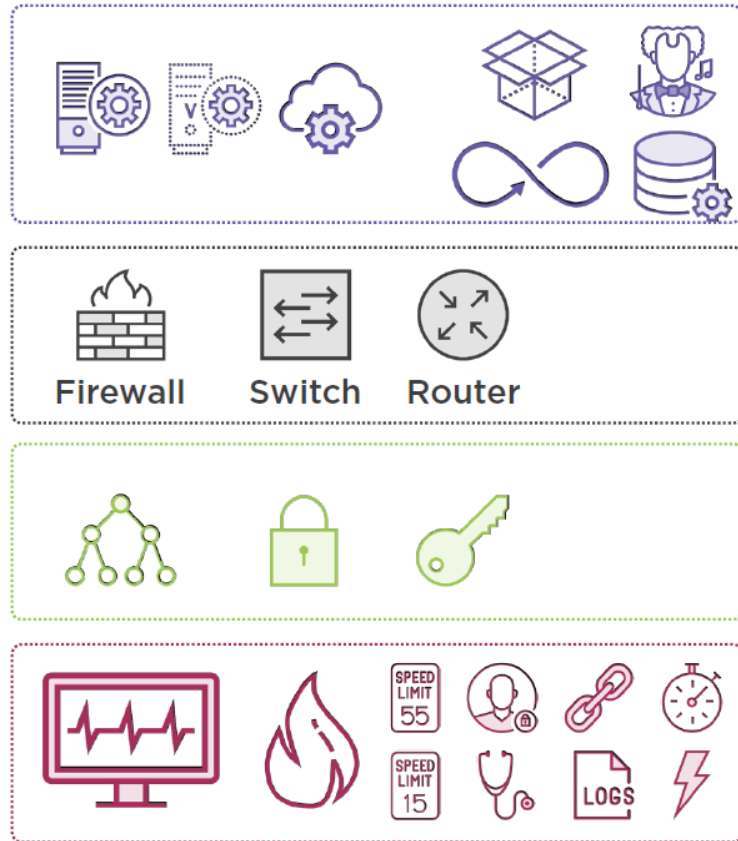
## Microservicio (II)

Un estilo de arquitectura de software, en el que las aplicaciones complejas se componen de pequeños procesos autónomos que se comunican entre sí mediante API independientes del lenguaje.



■ ¿Qué son microservicios?

## Elementos de un microservicio



¿Qué son microservicios?



¿Son los microservicios adecuados para mi organización?





## Beneficios de los microservicios

### Pequeños servicios

Puede ser propiedad de un equipo  
Más fácil de entender  
Puede ser reescrito

### Elección de tecnología

Adopta nueva tecnología  
Usa la herramienta adecuada  
Estandarizar donde  
tiene sentido

### Despliegue individual

Menor riesgo  
Minimiza el tiempo de inactividad  
Actualizaciones frecuentes

### Escalabilidad

Escalar servicios individualmente  
Económico

### Agilidad

Adaptarse rápidamente  
Reutilización más fácil

■ ¿Son los microservicios adecuados para mi organización?



## Desafíos de los microservicios

### Productividad del desarrollador

¿Cómo podemos facilitar que los desarrolladores sean productivos trabajando en el sistema?

### Interacciones complejas

Tenga cuidado para evitando comunicaciones ineficientes e innecesarias entre microservicios

### Despliegue

Necesitaras automatizar el proceso

### Monitoreo

Necesitamos un lugar centralizado para verificar los registros y monitorear los problemas

■ ¿Son los microservicios adecuados para mi organización?



# GRACIAS

POR SU PREFERENCIA