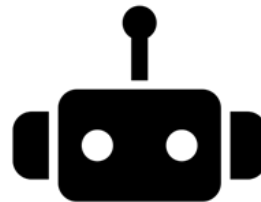


SESIÓN DE LABORATORIO

Microservicios II



HORARIO 10M1

¡Bienvenidos!

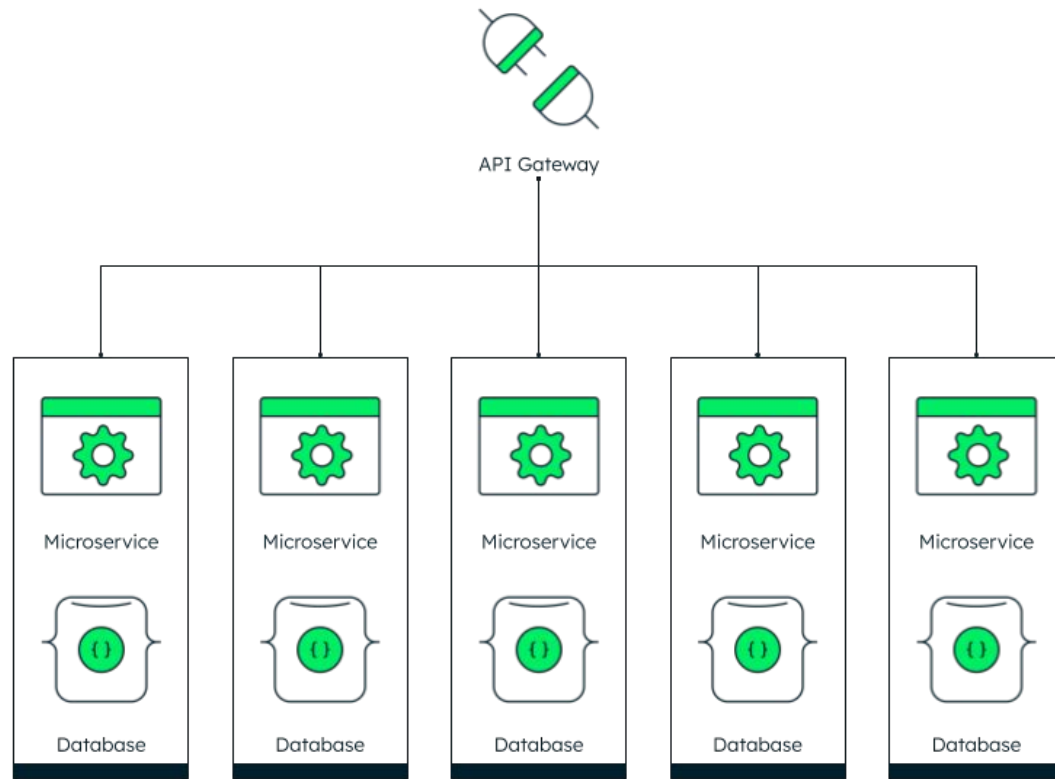
Empezaremos en unos minutos

¡Gracias!

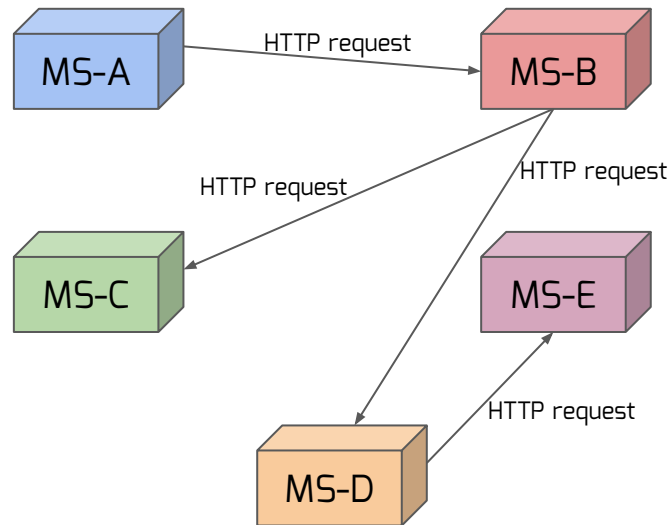


Microservicios: Desafíos

¿Qué desafíos presenta el utilizar una arquitectura de microservicios?



Desafío 1: Múltiples llamadas HTTP



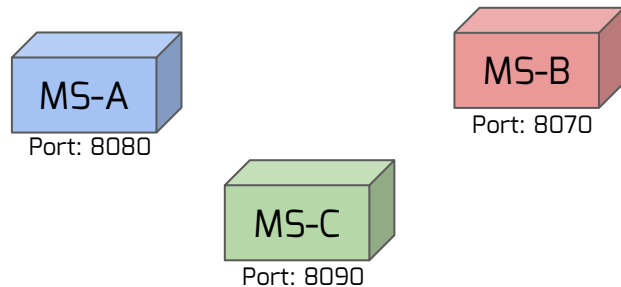
Para acceder a la información de otra base de datos, se tiene que realizar una petición HTTP hacia otro microservicio

Alternativa de solución: Feign

Simplificación de las solicitudes HTTP:

- Generación del cliente
- Conversiones de datos

Desafío 2: Centralización de información de cada microservicio



Cada microservicio tiene su propia dirección IP y puerto

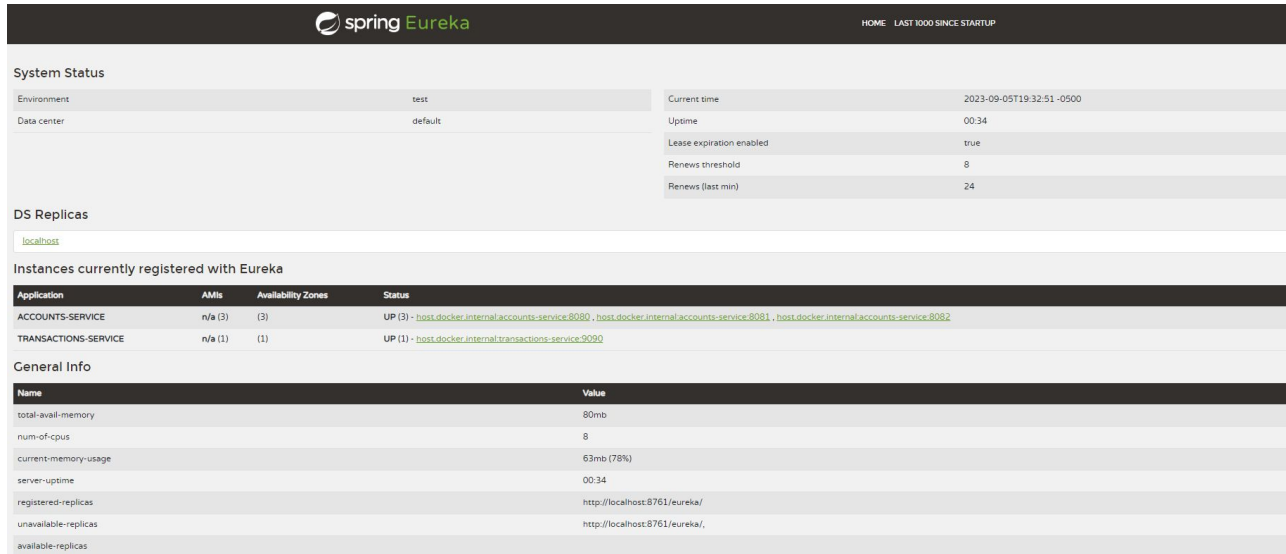
¿Dónde podemos ver toda esta información?

Alternativa de solución: Eureka

Eureka es un servidor de registro y descubrimiento de servicios (catálogo).

Beneficios de Eureka

- **Service Registry and Discovery:** Permite a los microservicios registrarse y descubrirse mutuamente sin intervención manual
- **Panel de Control:** Toda la información se encuentra centralizada en una interfaz de usuario



The screenshot displays the Spring Eureka dashboard interface. At the top, the 'spring Eureka' logo is visible on the left, and navigation links 'HOME' and 'LAST 1000 SINCE STARTUP' are on the right. The main content area is divided into several sections:

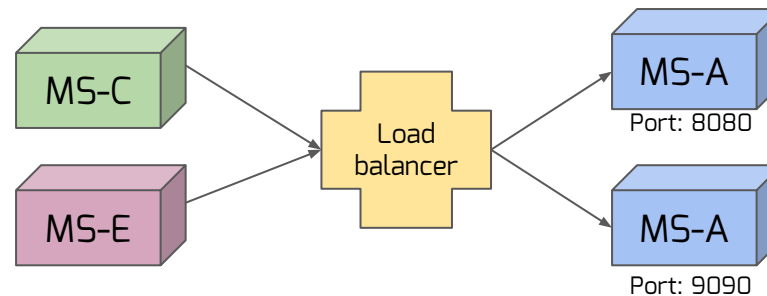
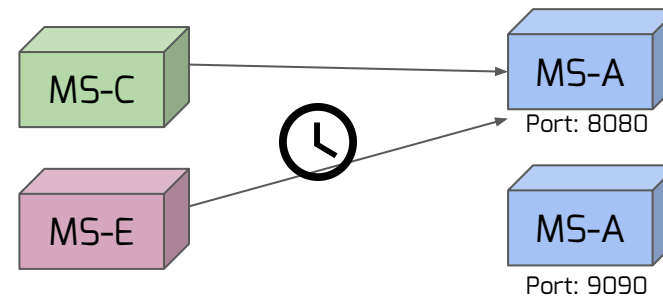
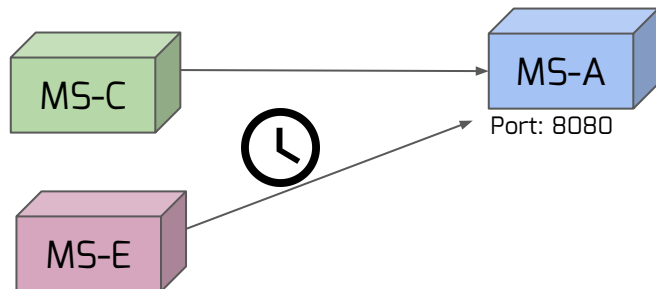
- System Status:** A table showing environment details.

Environment	test	Current time	2023-09-05T19:32:51 -0500
Data center	default	Uptime	00:34
		Lease expiration enabled	true
		Renews threshold	8
		Renews (last min)	24
- DS Replicas:** A section with a link to 'localhost'.
- Instances currently registered with Eureka:** A table listing registered services.

Application	AMIs	Availability Zones	Status
ACCOUNTS-SERVICE	n/a (3)	(3)	UP (3) - host.docker.internal/accounts-service-8080 , host.docker.internal/accounts-service-8081 , host.docker.internal/accounts-service-8082
TRANSACTIONS-SERVICE	n/a (1)	(1)	UP (1) - host.docker.internal/transactions-service-9090
- General Info:** A table showing system metrics.

Name	Value
total-avail-memory	80mb
num-of-cpus	8
current-memory-usage	63mb (78%)
server-uptime	00:34
registered-replicas	http://localhost:8761/eureka/
unavailable-replicas	http://localhost:8761/eureka/
available-replicas	

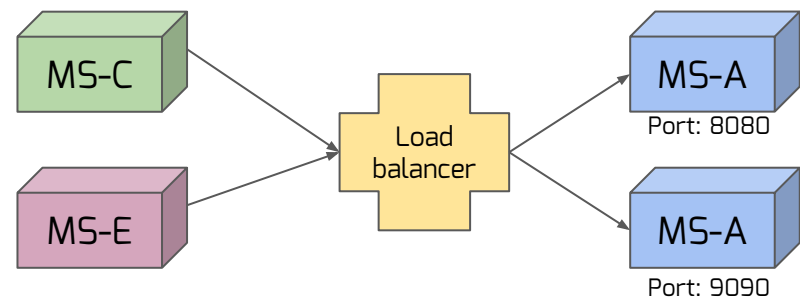
Desafío 3: Balanceo de carga



Desafío 3: Balanceo de carga

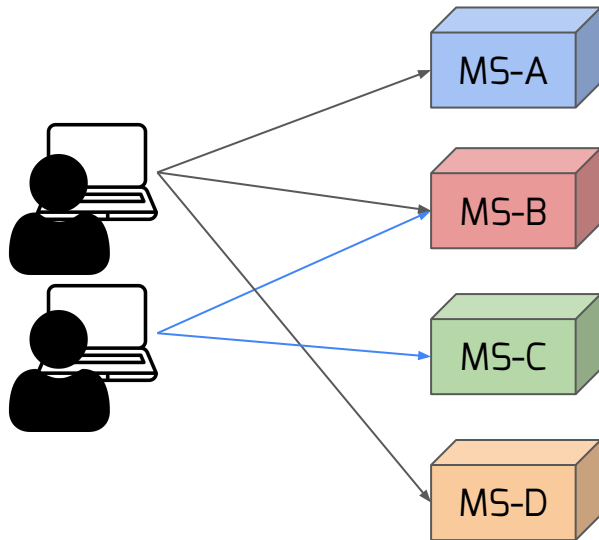
Beneficios adicionales de Eureka:

- **Balanceo de Carga:** Distribución automática de las solicitudes entre las instancias registradas
- **Failover Automático:** Si una instancia de servicio falla, Eureka redirigirá automáticamente las solicitudes a otras instancias disponibles



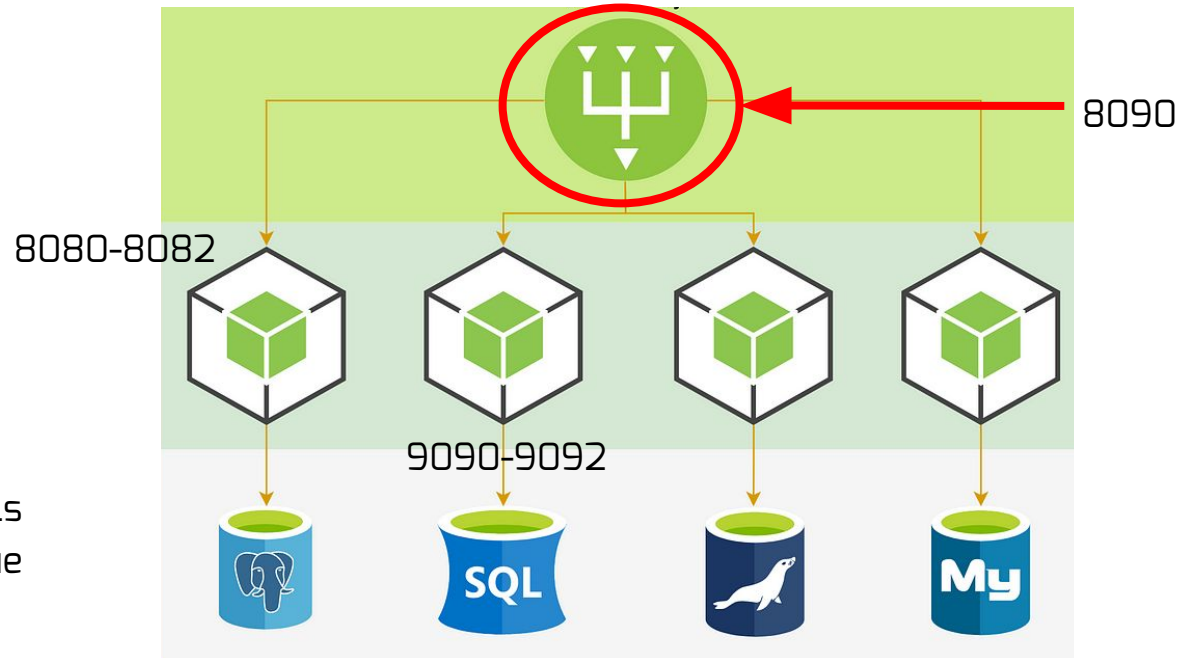
Alternativa de solución: Eureka

Desafío 4: Múltiples puntos de entrada para el cliente

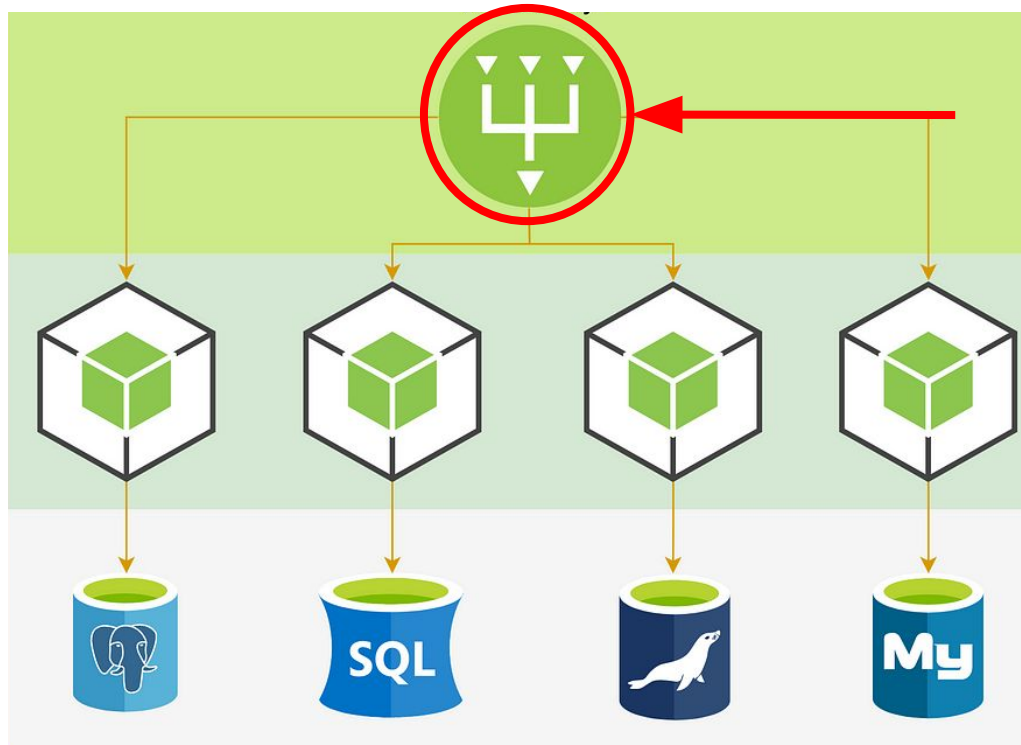


El cliente tendría que conocer las URLs de todos los microservicios a los que necesite conectarse.

Alternativa de solución: Zuul



Desafío 4: Múltiples puntos de entrada para el cliente



Alternativa de solución: Zuul

Beneficios:

- **Enrutamiento Dinámico:** Capacidad de redirigir las solicitudes de manera inteligente a los microservicios, según ciertos criterios, como la URL o los encabezados.
- **Seguridad y Autorización:** Se pueden integrar medidas de seguridad y autorización en Zuul

Ejemplo práctico: Spring Boot con Feign + Eureka + Zuul

