

MICROSERVICIOS

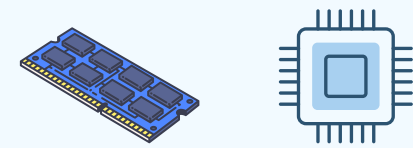
SERVICIOS DISTRIBUIDOS Y DESPLIEGUE

Daniel Blanco Calviño

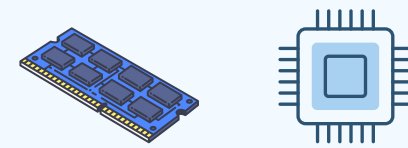
ESCALABILIDAD Y DISPONIBILIDAD



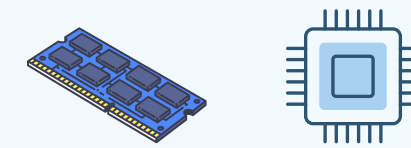
ESCALABILIDAD Y DISPONIBILIDAD



User
Microservice

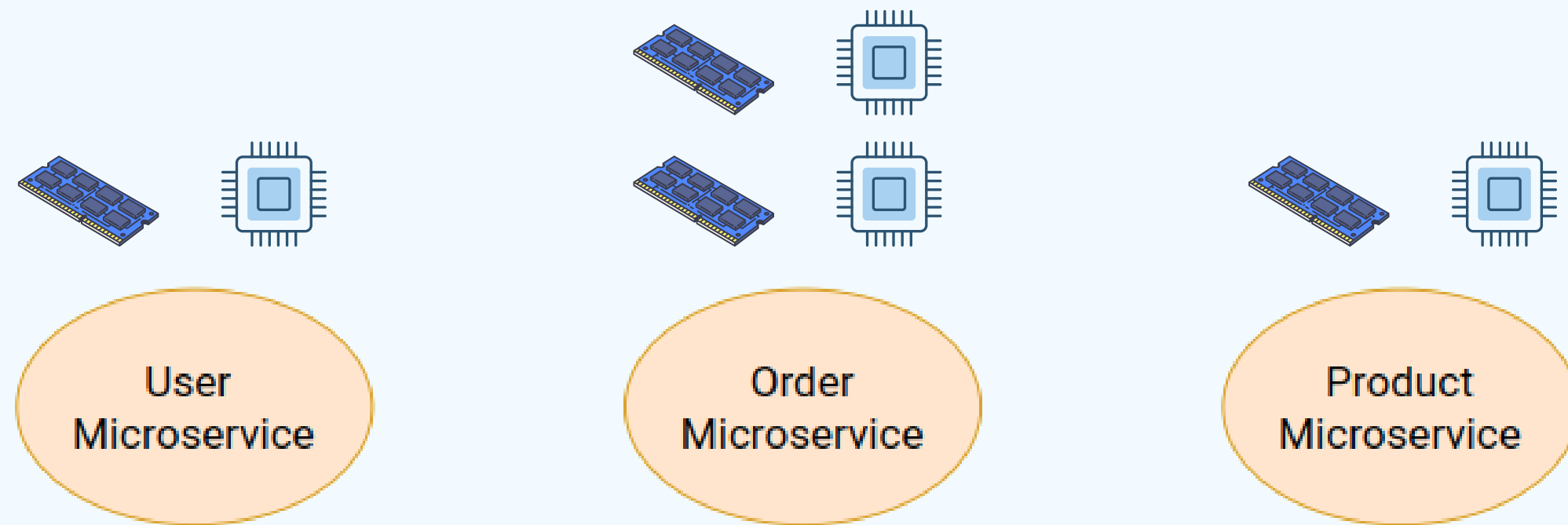


Order
Microservice

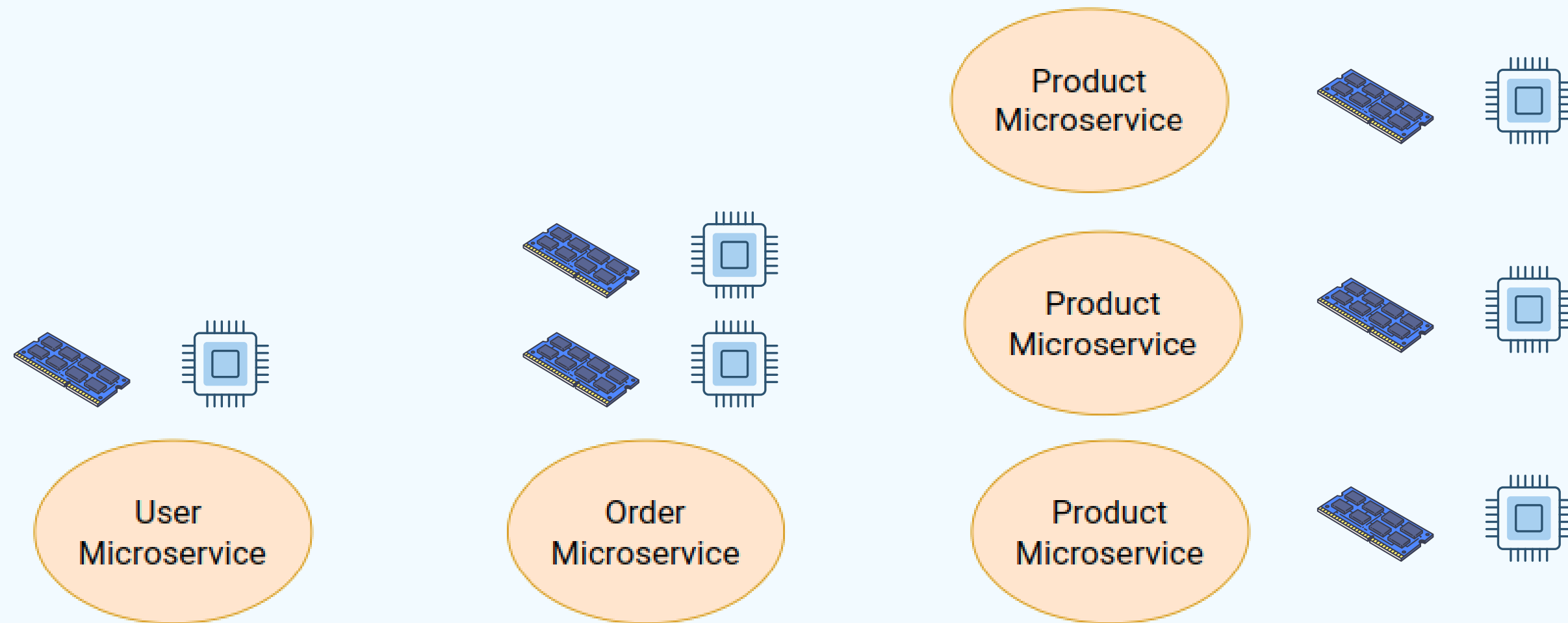


Product
Microservice

ESCALABILIDAD VERTICAL



ESCALABILIDAD HORIZONTAL



ESCALABILIDAD Y DISPONIBILIDAD

- **Escalando** nuestro sistema conseguimos **mayor disponibilidad**.
- **Escalabilidad vertical**
 - Necesidad de mayor potencia en un nodo.
 - Tareas pesadas, que necesitan mucha memoria etc.
- **Escalabilidad horizontal**
 - Mayor capacidad de procesamiento paralelo. Procesamos más peticiones en el mismo tiempo.
 - **Eliminamos puntos únicos de fallo.**

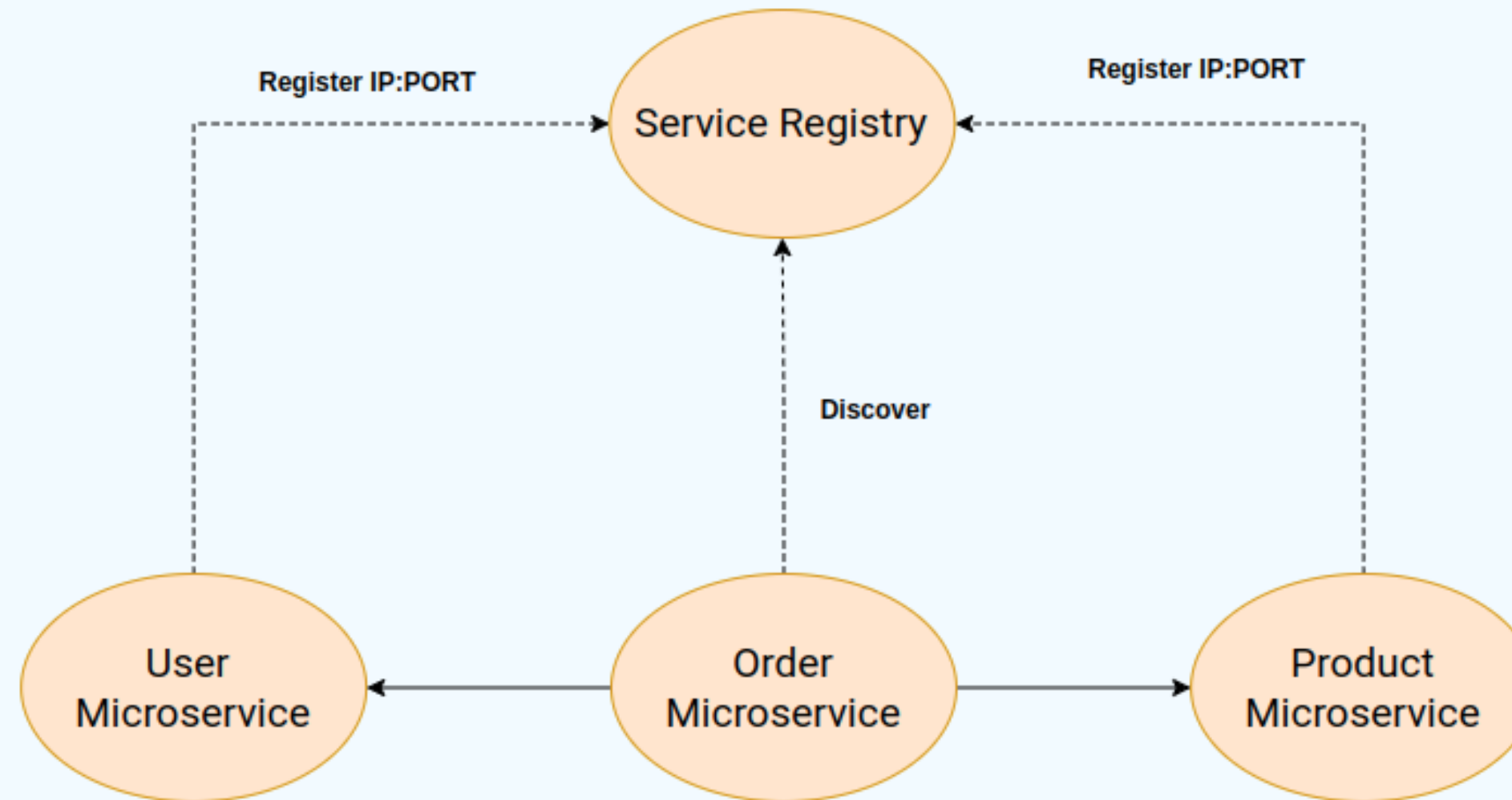
PUNTO ÚNICO DE FALLO

- **Pieza que si falla** por cualquier motivo **el sistema entero deja de funcionar.**
- Intentar que los **puntos únicos de fallo sean los menores posibles.**
- Los que existan, que sean **muy robustos.**

SERVICIOS DISTRIBUIDOS - INTERACCIÓN HTTP

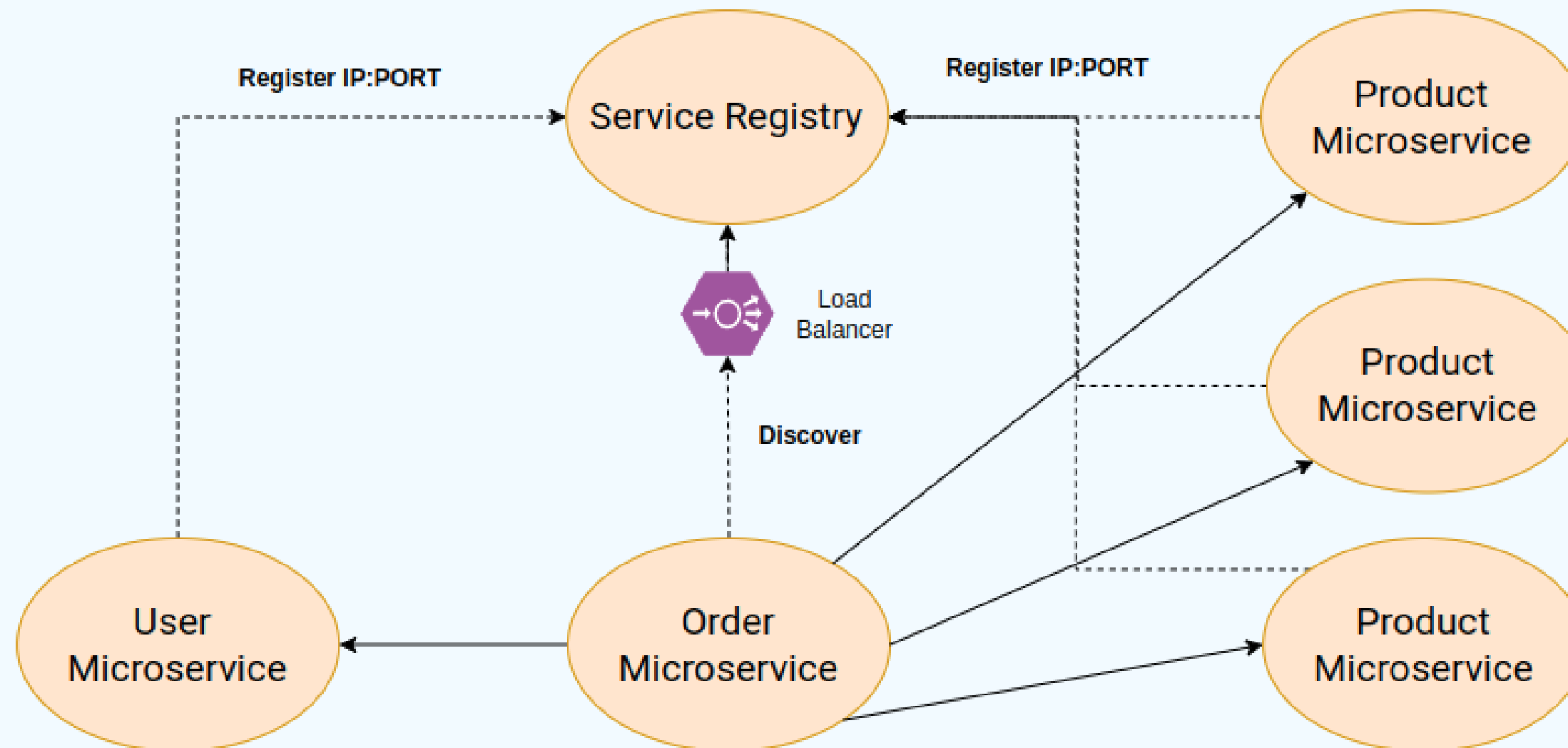
- **Despliegue en múltiples máquinas**, las cuales pueden ser dinámicas.
 - ¿Dónde reside el servicio que quiero llamar?
- **Múltiples instancias del mismo servicio.**
 - ¿A cual de ellos realizo la petición?

MÚLTIPLES MÁQUINAS - INTERACCIÓN HTTP



- Ejemplo de registro: **Eureka**

MÚLTIPLES INSTANCIAS - INTERACCIÓN HTTP



MÚLTIPLES MÁQUINAS - INTERACCIÓN POR MENSAJES

- No existe este problema.
- Los microservicios **no necesitan saber nada de los demás**.
 - Se subscriben a los eventos que les interesan.
 - Publican los eventos necesarios.



El servicio de mensajes también es un **punto único de fallo**.

MÚLTIPLES INSTANCIAS - INTERACCIÓN POR MENSAJES

- **Tampoco existe este problema.**
- El mensaje será procesado por la primera instancia del servicio que se encuentre libre.
 - Si una instancia está sobrecargada, no va a consumir eventos.

MICROSERVICIOS

SERVICIOS DISTRIBUIDOS Y DESPLIEGUE

Daniel Blanco Calviño