

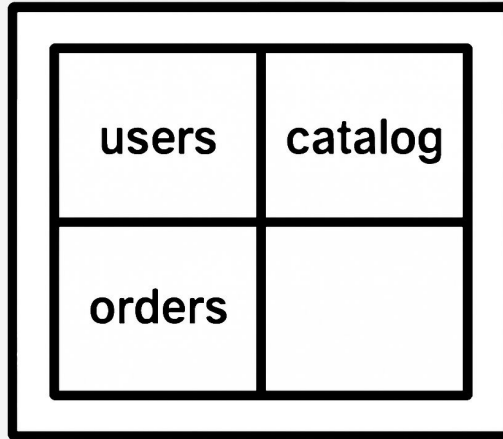
MÓDULO 2: ARQUITECTURA DE SOFTWARE

ESTILOS DE ARQUITECTURA MONOLITO MODULAR

INTRODUCCIÓN A LA ARQUITECTURA DE SOFTWARE

Estilos de arquitectura: Monolito Modular

Un único despliegue, múltiples módulos



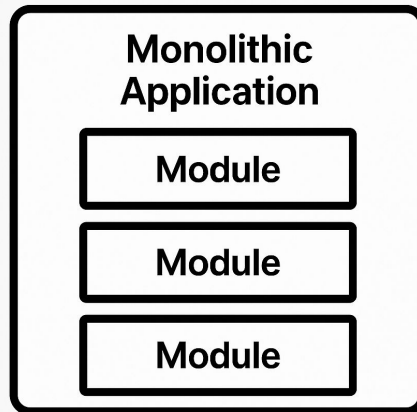
¿Qué es un monolito modular?

Un **único artefacto** desplegable

Módulos separados por **dominio**

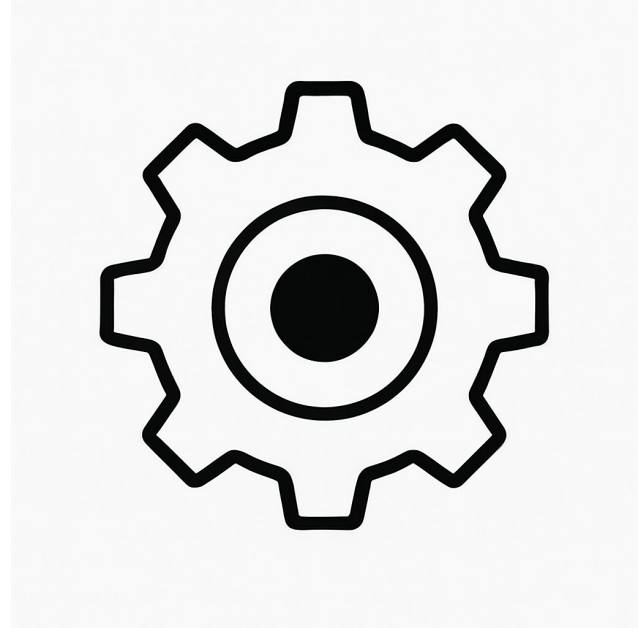
No es “bola de barro”

Se combina con Hexagonal o Clean



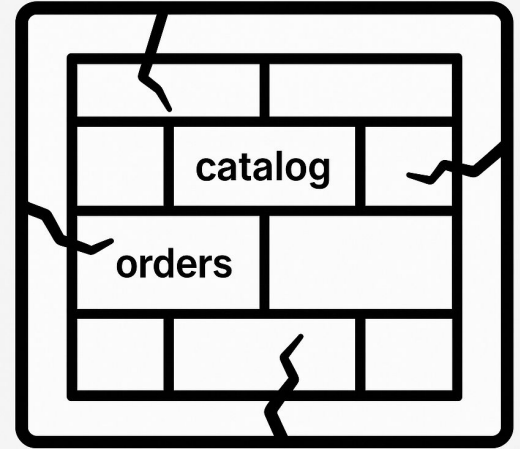
Ventajas

- ✓ Sencillez operativa (un despliegue)
- ✓ Latencia interna mínima
- ✓ Productividad alta
- ✓ Refactors más fáciles



Riesgos

- ❌ Riesgo de “bola de barro”
- ❌ Un único despliegue → cuello de botella
- ❌ Caída afecta a todo el sistema

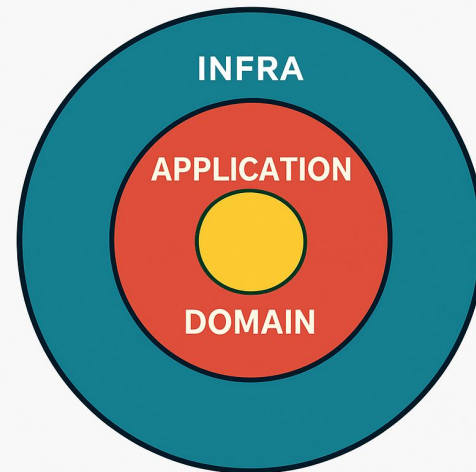


Anatomía del monolito

Módulos alineados al dominio

Capas internas: **domain** / **application** / **infra**

Dependencias siempre hacia el dominio



Ejemplo en Node/TS

```
// Dominio
class Product { changePrice() { ... } }

// Puerto
interface ProductRepo { findById(); save(); }

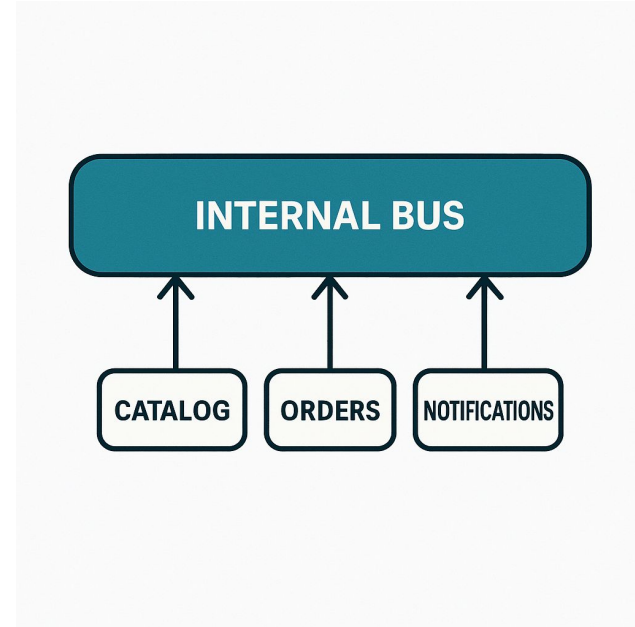
// Caso de uso
updateProductPrice(repo, {id, price})
```

Eventos internos

Desacoplar módulos sin HTTP

Eventos in-memory (**emit**, **on**)

Preparan para microservicios



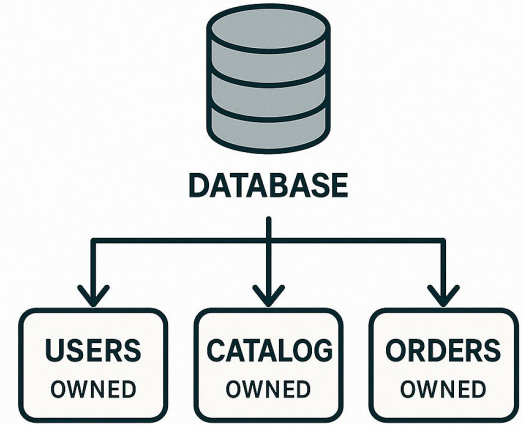
Datos y límites

Base de datos compartida (al inicio)

Propiedad clara de tablas por módulo

Evitar consultas cruzadas

Usar casos de uso o eventos



Testing y observabilidad

Unit tests en dominio

Integration tests en adaptadores

Contract tests entre módulos

Logs con `correlationId`



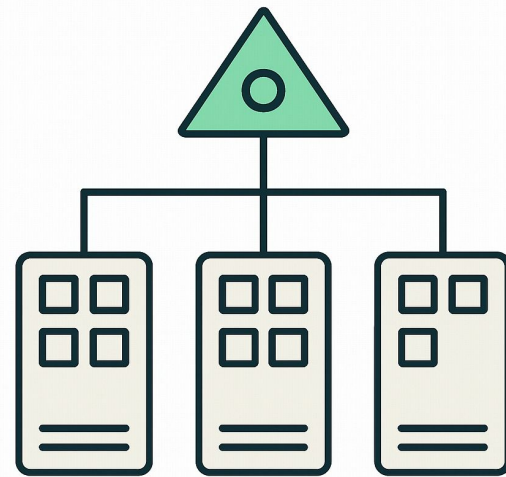
Escalar el monolito

Escalar horizontalmente (varias instancias)

Sesiones fuera del proceso (Redis)

Caché en paths críticos

Extraer módulo sólo si lo requiere



*“Un buen monolito modular te da **velocidad hoy** y **opciones mañana**.”*

➡ Próxima lección: **Microservicios**

