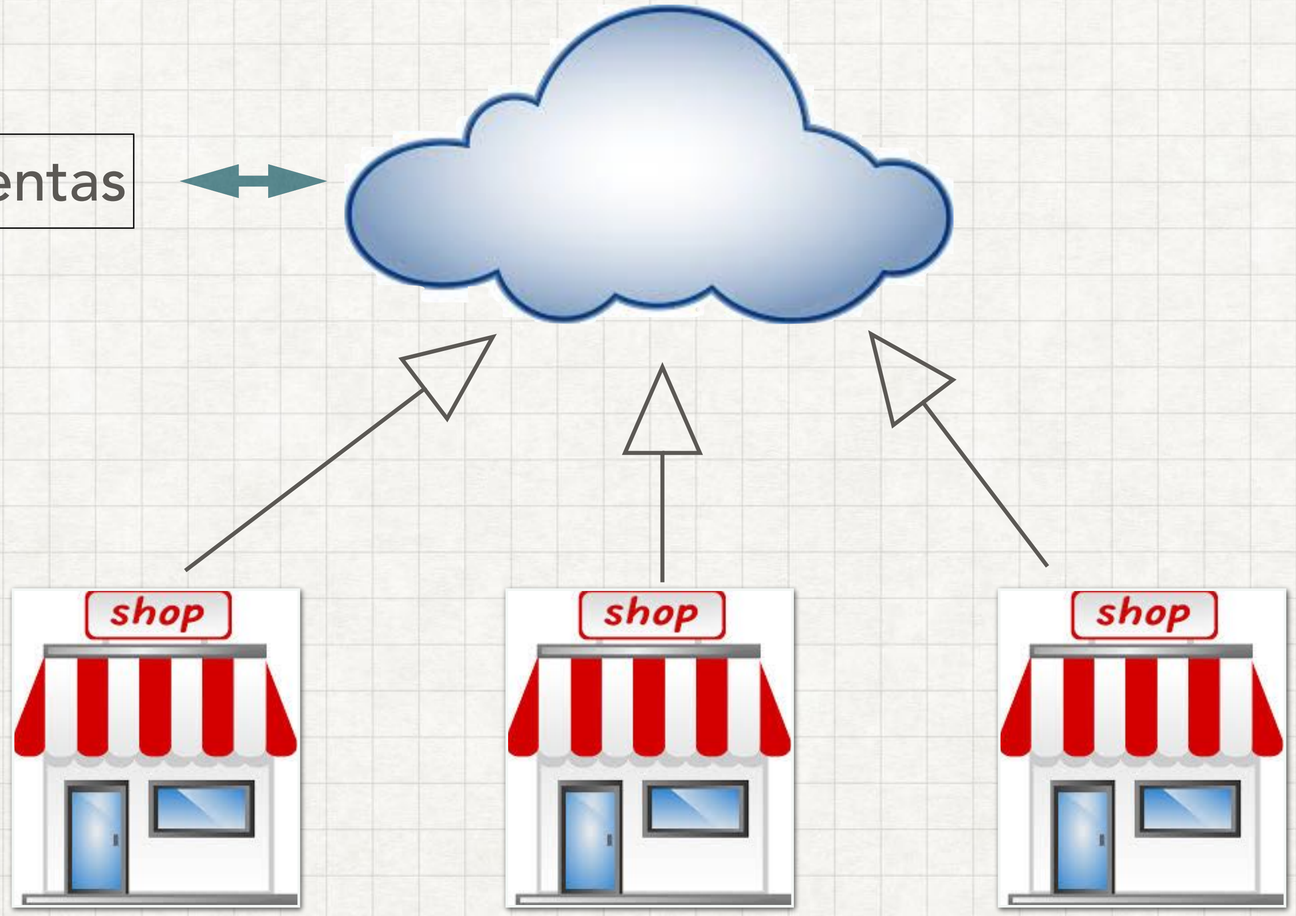
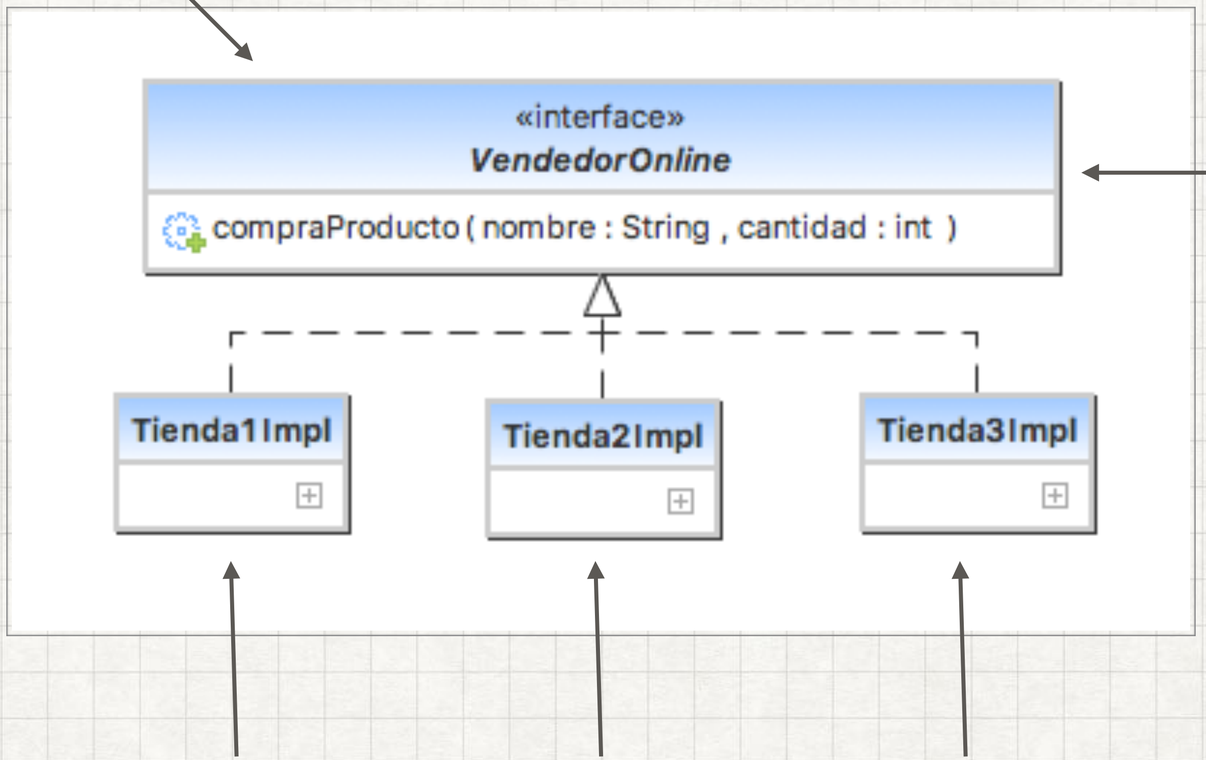
BRIDGE PATTERN

Interface de ventas



tienda A *tienda B* *tienda C*

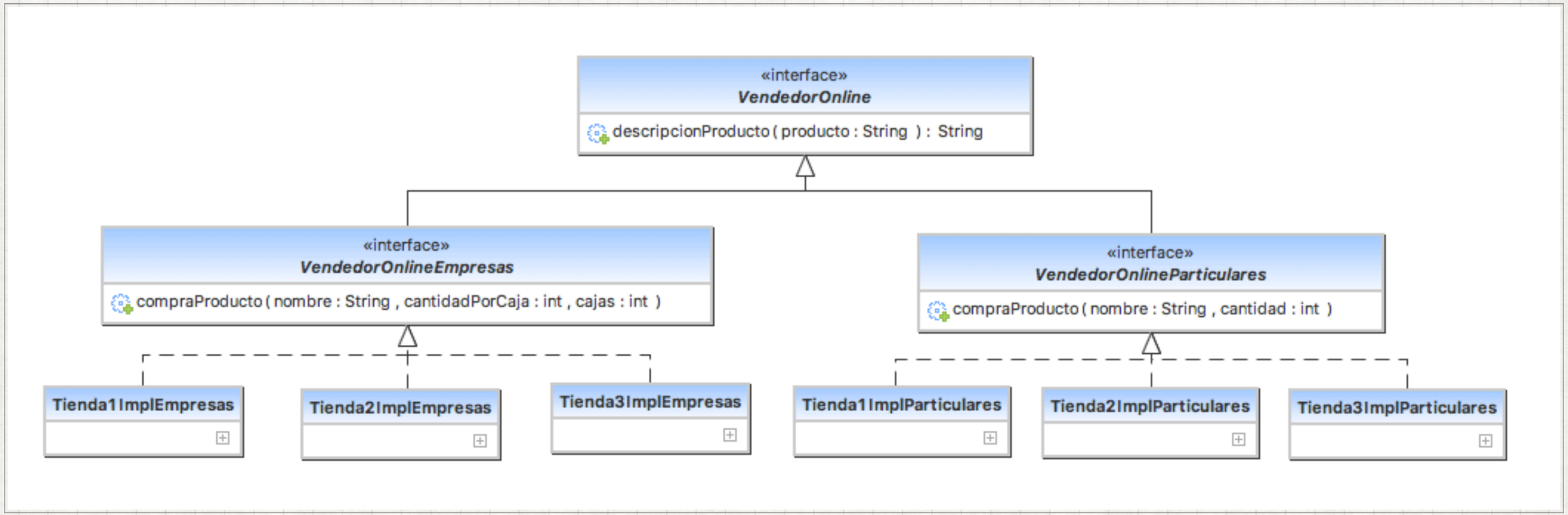
Interface que creamos para publicar a los clientes



¡Aún en desarrollo!

Las distintas implementaciones en cada tienda deben ir adaptándose a los cambios que vamos introduciendo en la interface VendedorOnline

"Los clientes particulares y los de al por mayor son diferentes, necesitamos interfaces particulares para cada uno de ellos"



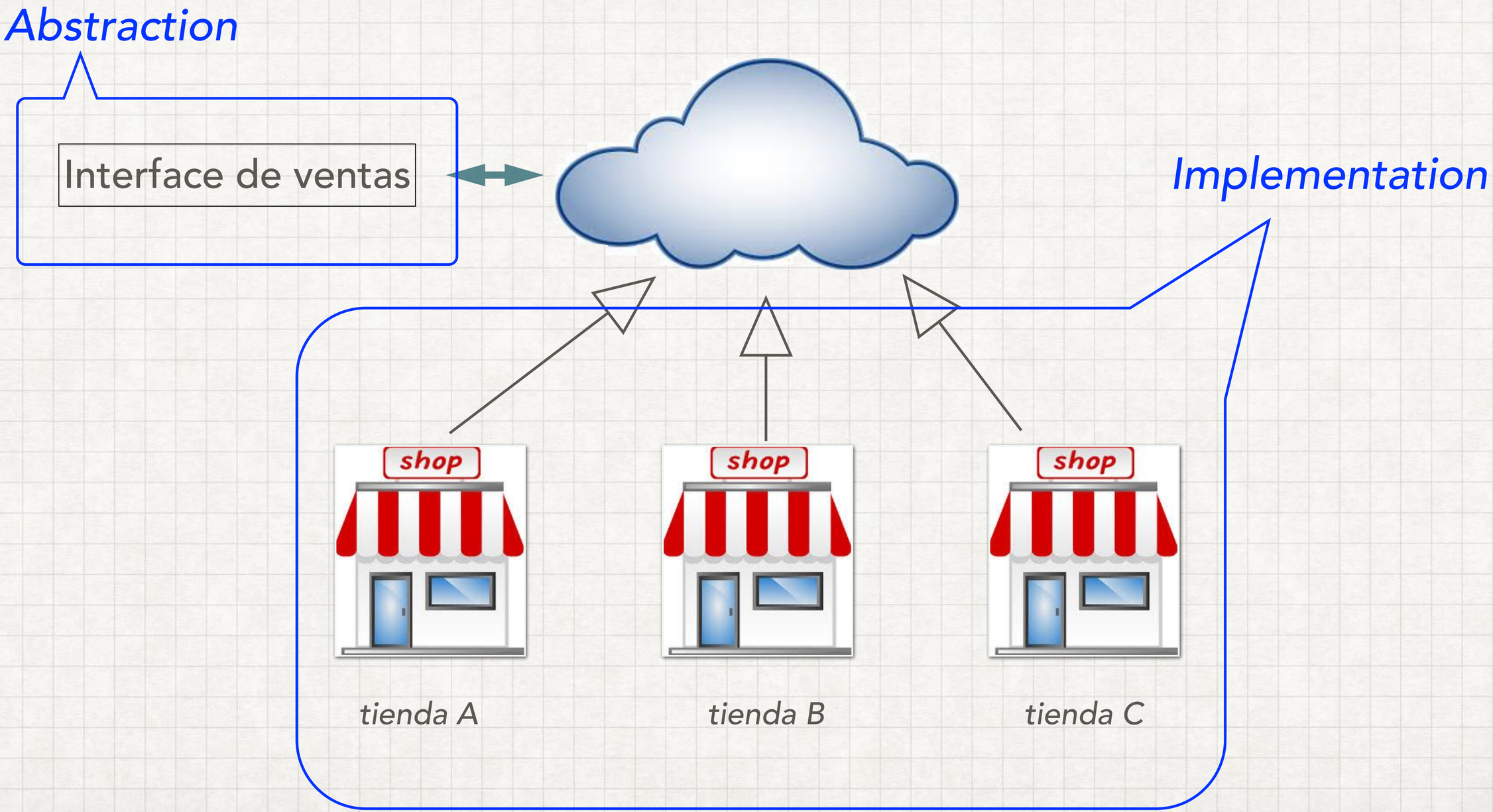
No parece la forma más flexible de trabajar con una interface cambiante

“

Decouple an abstraction from its implementation so that the two can vary independently.

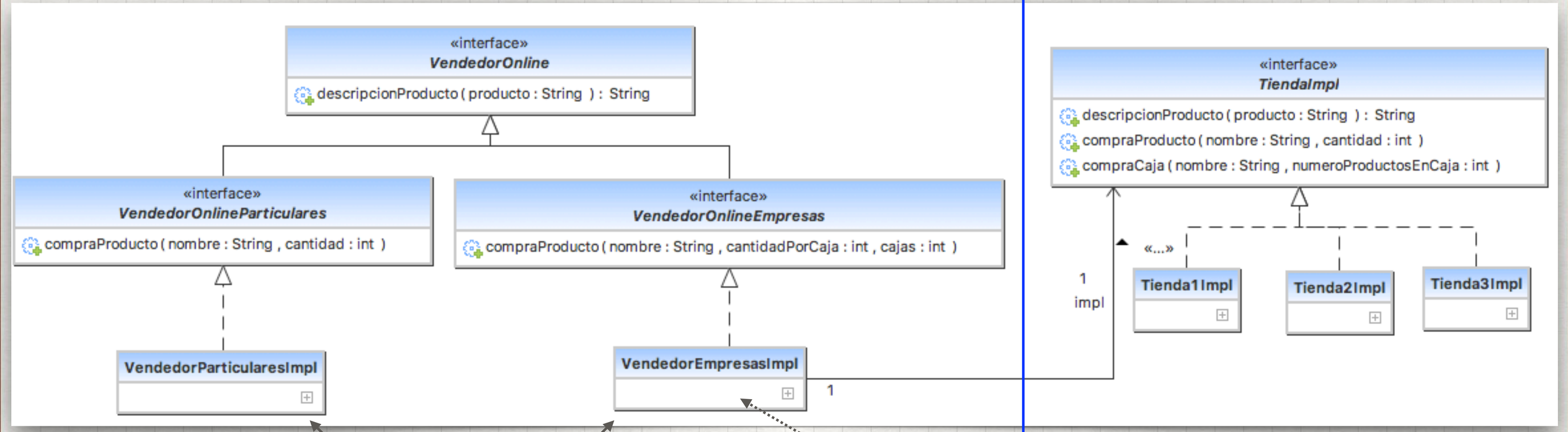
— *GoF*

”


Abstraction
Interface de ventas
Implementation
shop
shop
shop
tienda A
tienda B
tienda C

Abstraction

Implementation



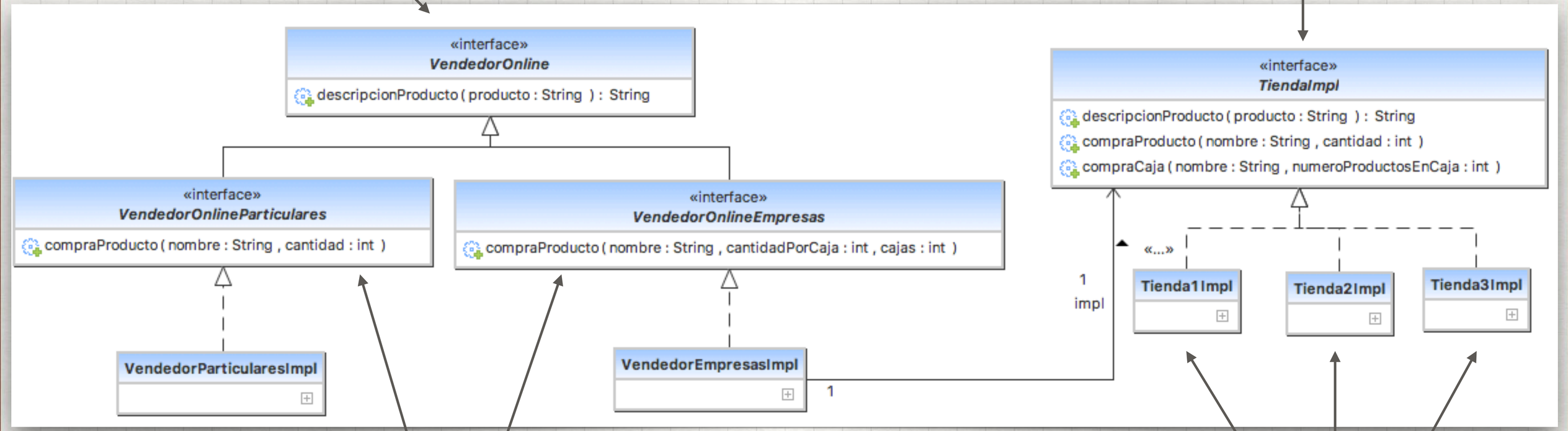
"Traducen" las llamadas según la API de la abstracción a llamadas según la API de la implementación

```
for (int i = 0; i < cajas; i++) {
    impl.compraCaja(nombre, cantidadPorCaja);
}
```

Terminología del pattern

Abstraction

Implementor



Refined abstraction

Concrete implementor

BRIDGE PATTERN