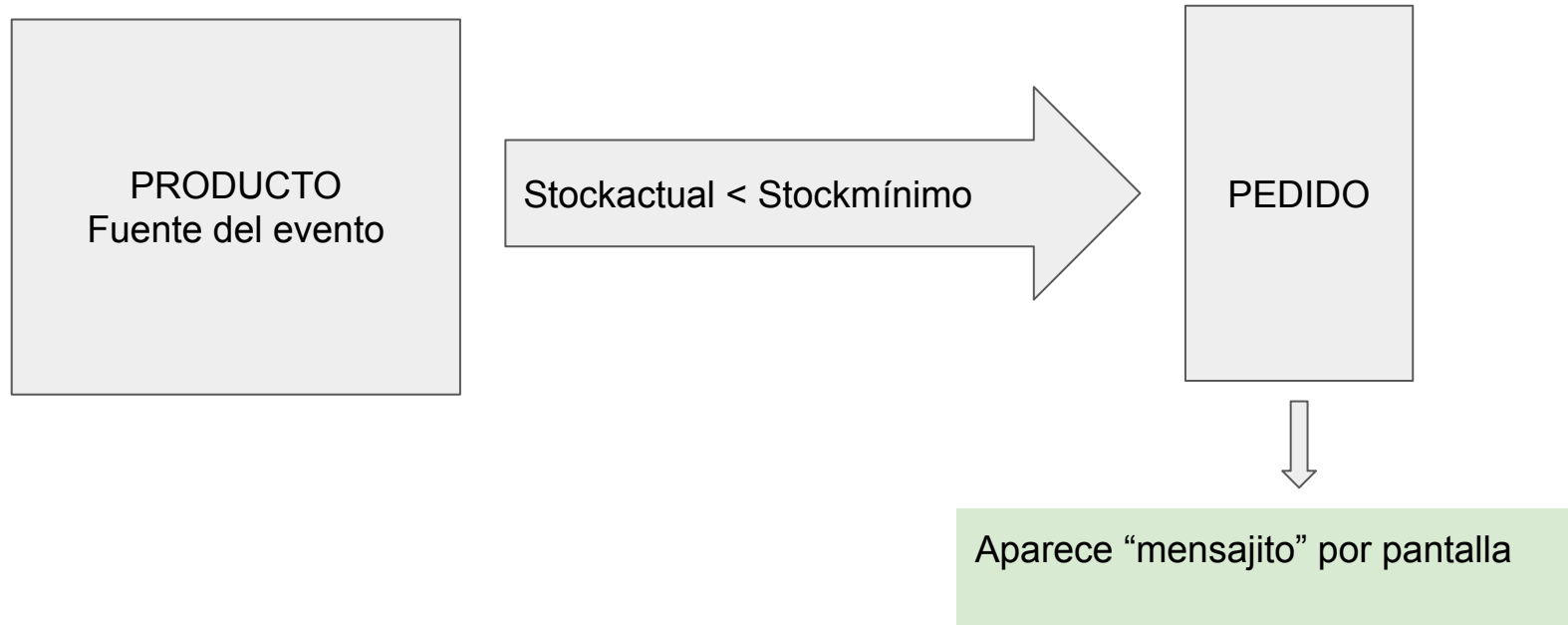


Acceso a base de datos desde JavaBean

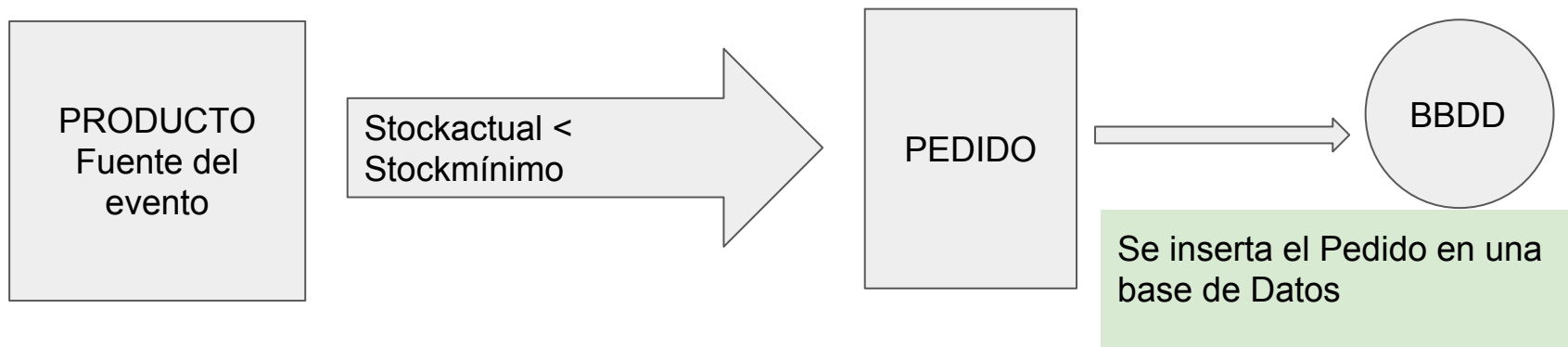
Introducción

- Vamos a montar una aplicación por componentes utilizando JavaBeans.
- Esta aplicación nos servirá para introducir el acceso a datos desde un JavaBean
- Como punto de partida tomaremos el ejemplo de la sesión anterior donde teníamos dos tipos de objetos: Pedido y Producto.
- Producto era el **JavaBean Fuente de Eventos** y Pedido era el **JavaBean Receptor de Eventos**.
- Cuando se modificaba un el stock de un producto (`producto.setStockactual`) se generaba un evento. Este evento era capturado por un pedido y aparecía un mensajito.



Objetivo

- Vamos a modificar el *propertyChange* de Pedido para que en vez de generar un “mensajito” inserte el pedido en una base de datos.
- Utilizaremos la base de datos de objetos *Neodatis*.



Acceso a la base de datos

- Se ha de crear una clase que contenga todos los métodos que permitan abrir la base de datos, cerrar, guardar ventas, pedidos y productos en la base de datos.
- En concreto, tendrá:
 - Propiedad de tipo `org.neodatis.odb.ODB`
 - Un constructor que abrirá la base de datos: `this.odb = ODBFactory.open("Producto_Ped.BD");`
 - Un método para cerrar la base de datos
 - Un método para insertar un producto
 - Un método para insertar un pedido
 - Un método para insertar una venta.

Problemas Acceso-JavaBeans

1. Genera el código de la clase “BaseDatos”. Debe contener todos los métodos mencionados. Trabaja con *Neodatis*.
2. Genera un programa que llamarás “LlenarProductos”. Insertará productos en la base de datos. Utilizará los métodos de la clase “BaseDatos”. Los productos a insertar son estos:

```
Producto p1 = new Producto(1, "Duruss Cobalt", 10, 3, 220);  
Producto p2 = new Producto(2, "Varlion Avant Carbon", 5, 2, 176);  
Producto p3 = new Producto(3, "Star Vie Pyramid R50", 20, 5, 193);  
Producto p4 = new Producto(4, "Dunlop Titan", 8, 3, 85);  
Producto p5 = new Producto(5, "Vision King", 7, 1, 159);  
Producto p6 = new Producto(6, "Slazenger Reflex Pro", 5, 2, 80);
```

Problemas Acceso-JavaBeans

3. Genera un programa que llamarás “VerProductos” que permita visualizar los productos y sus características de la base de datos.
4. Modifica el *propertyChange* de la clase Pedido para que, cuando se produzca un evento desde Producto - es decir, cuando Stockactual supere el Stockmínimo - se inserte un pedido en la base de datos. Se debe considerar:
 - a. El número del pedido (propiedad *numeroPedidoI*) a insertar en la base de datos se debe de generar. Para ello, se utilizará el número de Pedido más alto más uno de entre los Pedidos que ya están almacenados en la base de datos (Es decir, reproducir un “auto_increment” mediante programación).
 - b. El campo fecha del pedido debe ser la fecha actual.
5. Modifica el programa “VerProductos” para que permita visualizar los Pedidos además de los productos.

Problemas Acceso-JavaBeans

6. Escribe un programa que sirva para testear todos los componentes. Deberá de:
 - a. Mostrar productos
 - b. Mostrar pedidos
 - c. Disparar la generación de un pedido y su posterior almacenamiento en la base de datos. Para ello, lógicamente, modificará el Stockactual de algún producto de tal forma que quede por debajo de su Stockmínimo.
 - d. Volver a mostrar los productos. NOTA: el Stockactual del producto que ha provocado la generación del pedido NO se modificará. Si, en cambio, la modificación del Stockactual no provoca la generación del pedido (es decir, el Stockactual modificado no queda por debajo del Stockmínimo del producto) entonces sí que debe modificarse el Stockactual ($Stockactual = Stockactual - cantidad$)
 - e. Volver a mostrar los pedidos.