

Desarrollo de Software I

Patrones – Parte III

Ing. Eric Gustavo Coronel Castillo

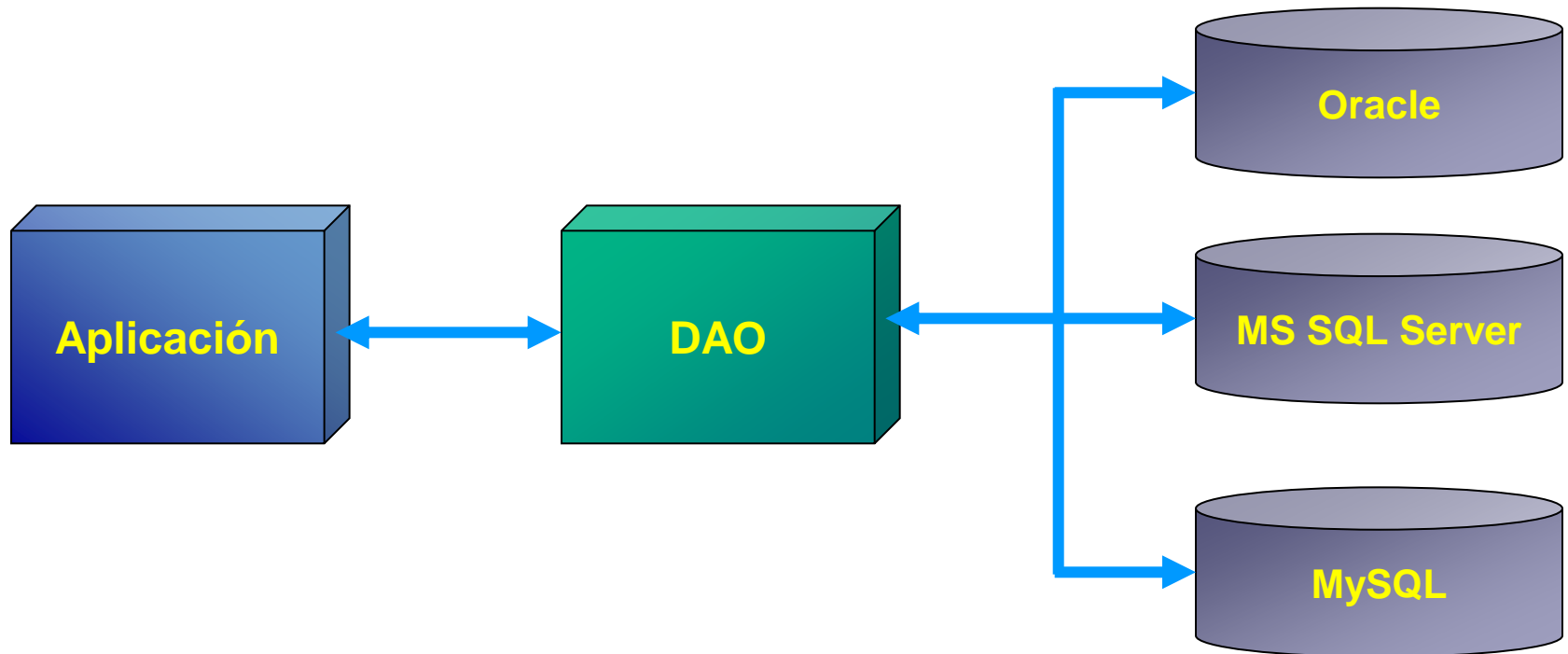
gcoronelc@gmail.com
gcoronelc.blogspot.com

Índice

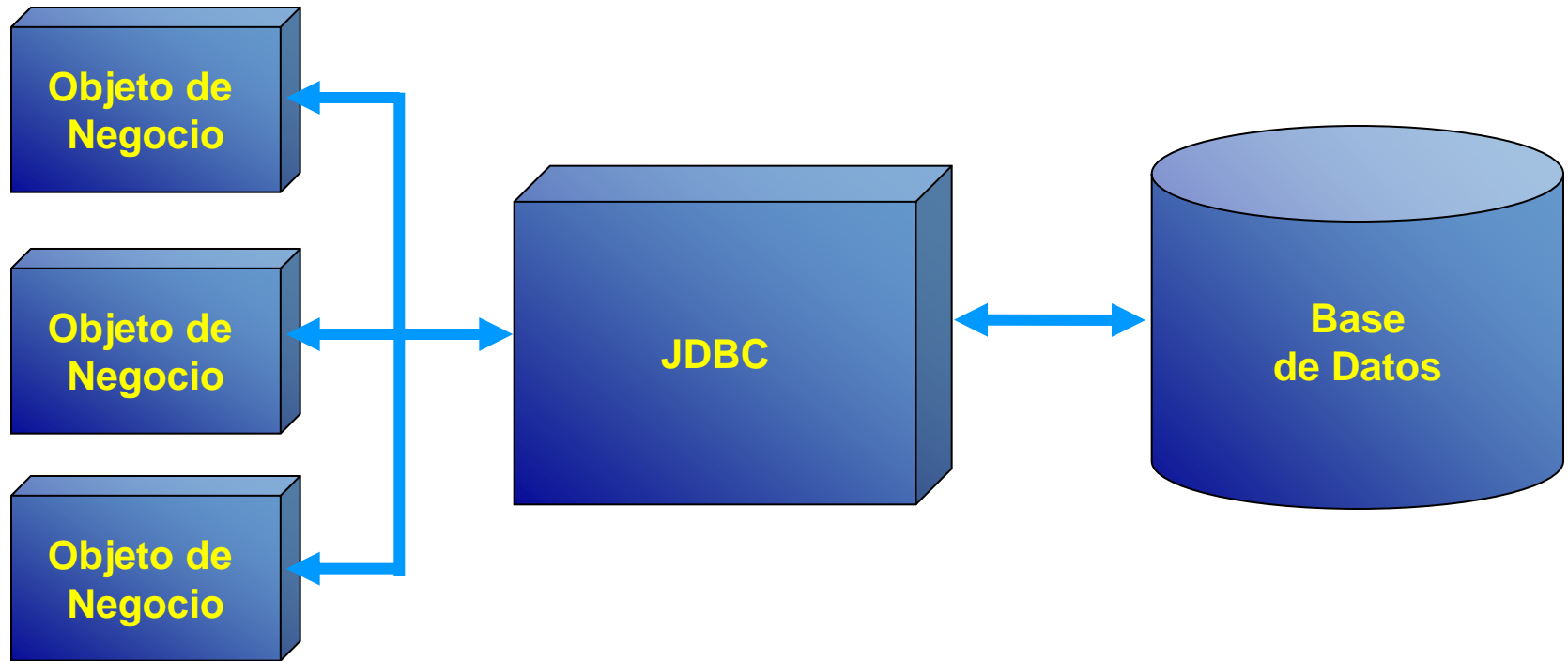
- ❖ Objetivo
- ❖ Desarrollo Clásico
- ❖ Patrón de Diseño DAO
- ❖ DAO con Factory Method
- ❖ DAO con Abstract Factory

Objetivo

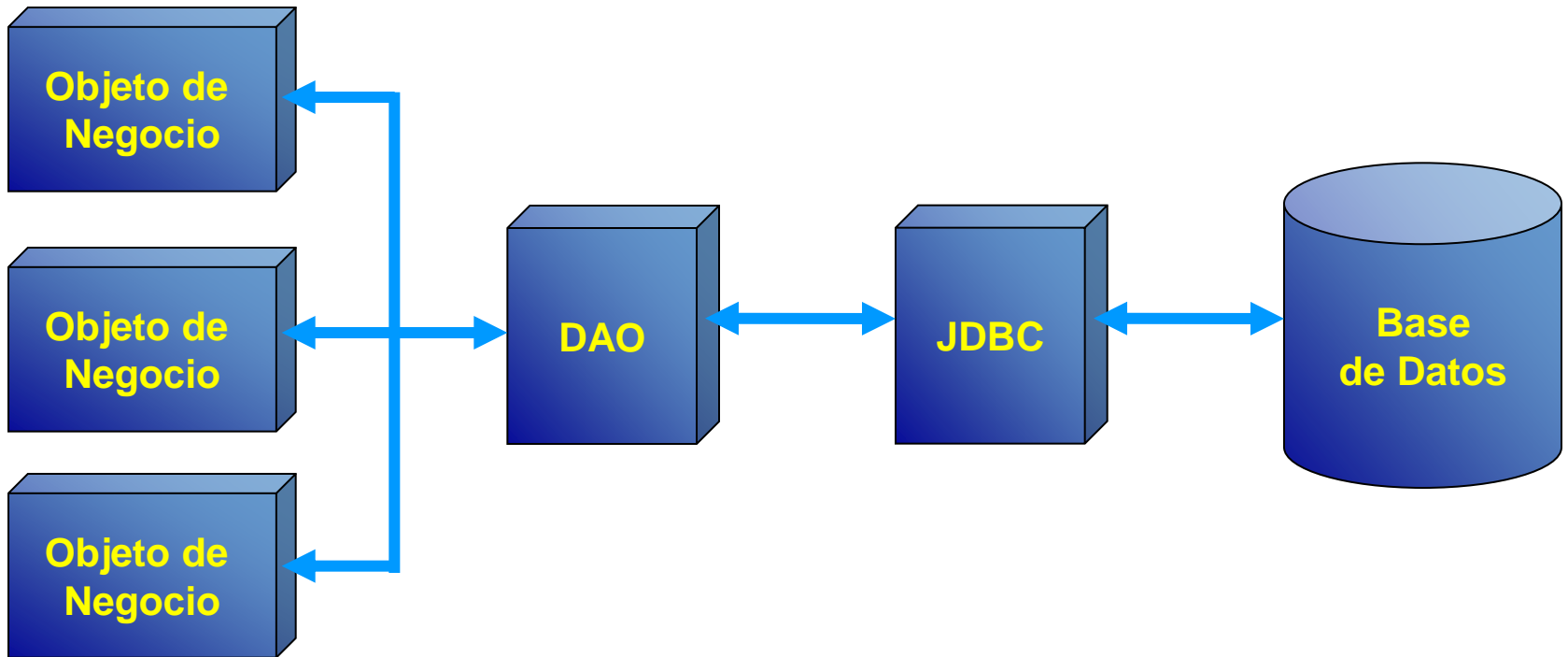
Estandarizar el acceso a fuentes de datos utilizando el patrón de diseño DAO y Factory.



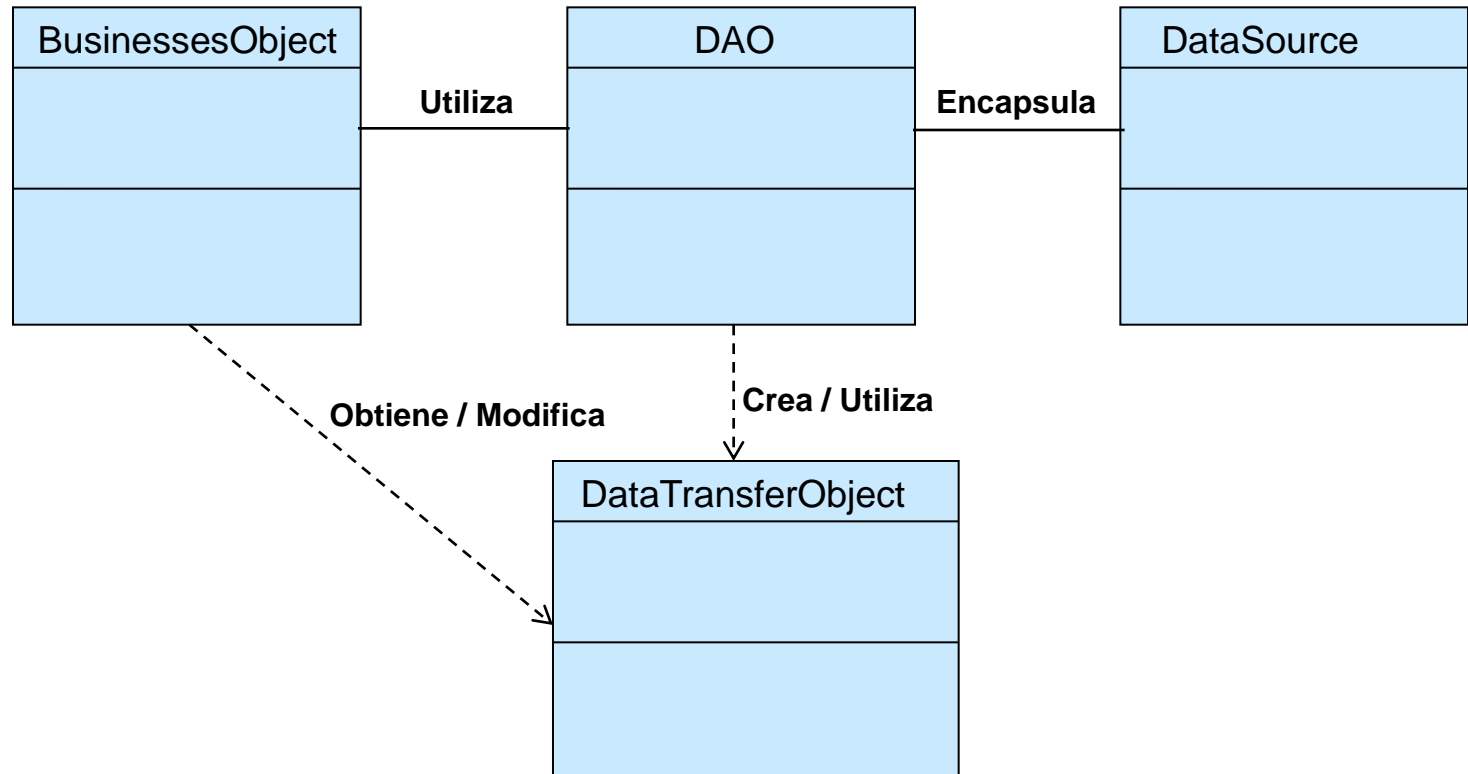
Desarrollo Clásico



Patrón de Diseño DAO

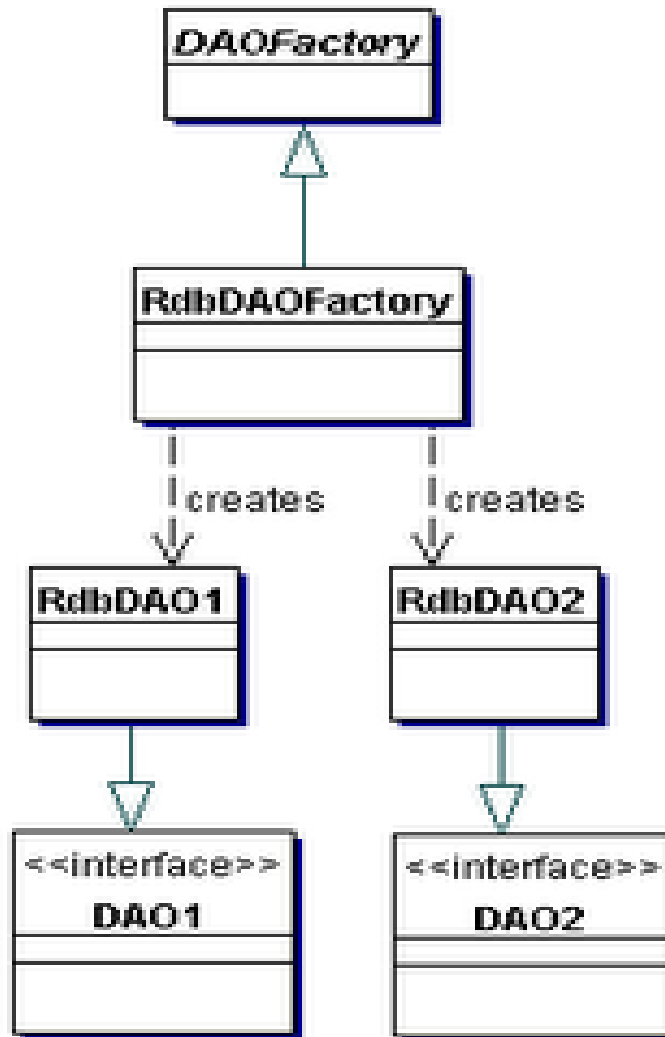


Patrón de Diseño DAO



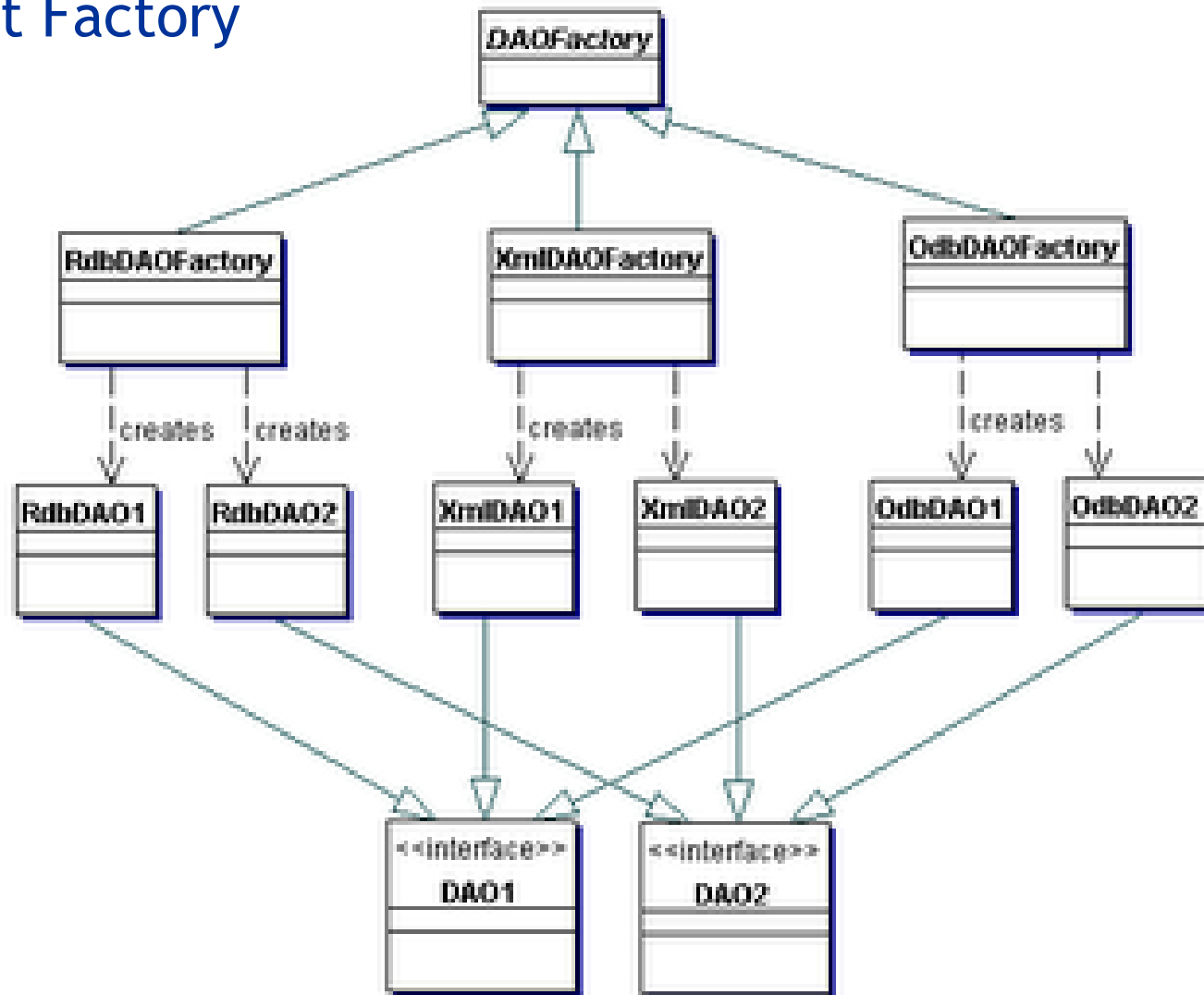
DAO con Factory Method

❖ Factory Method



DAO con Abstract Factory

❖ Abstract Factory



Conclusiones

- ❖ El patrón de diseño DAO se usa para separar las operaciones de bajo nivel de acceso a datos (low-level data access), de las operaciones de alto nivel de la lógica del negocio (high-level business logic).
- ❖ La estrategia de implementar DAO con Abstract Factory y Factory Method permite construir varios tipos de DAO factories, cada uno soportando un tipo diferente de fuente de datos.
- ❖ El uso de DAO no solo estandariza el desarrollo de aplicaciones, sino, también las simplifica; además permite que varios equipos participen en diferentes componentes dependiendo de la especialidad de cada persona.

Bibliografía

- ❖ Desarrollando Soluciones con Java y MySQL Server
Eric Gustavo Coronel Castillo
- ❖ Piensa en Java
Bruce Eckel
- ❖ Como Programar en Java
Deitel y Deitel
- ❖ Java 2
Steven Holzner
- ❖ La Biblia de Java 2 v5.0
Herbert Schildt
- ❖ Acceso a Bases de Datos con Java-JDBC
Ángel Esteban

Referencias

- <http://java.sun.com/blueprints/corej2eepatterns/Patterns/DataAccessObject.html>
- <http://www.programacion.net/java/tutorial/patrones2/8/>
- <http://weblogs.javahispano.org/hip/category/Java>