

Patrones de Diseño

❑ Hacer software no es fácil

Diseñar software orientado a objetos es difícil, y diseñar software Orientado a objetos **reutilizable** es todavía **más difícil**.

Chapter 1: Introduction. Design Patterns, The Gang of Four

...y un software capaz de evolucionar tiene que ser reutilizable
(al menos para las versiones futuras)

Patrones de Diseño

□ Diseñar para el cambio

- El software cambia.
- Para anticiparse a los cambios en los requisitos hay que diseñar pensando en qué aspectos pueden cambiar.
- Los patrones de diseño están orientados al cambio.

Patrones de Diseño

- Cómo llegar a ser un maestro de ajedrez
 - **Primero aprender las reglas del juego**
 - Nombres de las piezas, movimientos legales, geometría y orientación del tablero, etc.
 - **A continuación aprender los principios**
 - relativo valor de las piezas, valor estratégico de las casillas centrales, jaque cruzado, etc.
 - **Sin embargo, para llegar a ser un maestro, hay que estudiar las partidas de otros maestros**
 - Estas partidas contienen patrones que deben ser entendidos, memorizados y aplicados repetidamente
 - **Hay cientos de estos patrones**

Patrones de Diseño

- Cómo llegar a ser un maestro de software
 - **Primero aprender las reglas del juego**
 - Algoritmos, estructuras de datos, lenguajes de programación, etc.
 - **A continuación aprender los principios**
 - Programación estructurada, programación modular, programación OO, programación genérica, etc.
 - **Sin embargo, para llegar a ser un maestro, hay que estudiar los diseños de otros maestros**
 - Estos diseños contienen patrones que deben ser entendidos, memorizados y aplicados repetidamente
 - **Hay cientos de estos patrones**

Patrones de Diseño

□ ¿Qué es un patrón de Diseño?

Christopher Alexander (1977):

*“Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, y **describe la esencia de la solución** a ese problema, de tal modo que pueda utilizarse ésta solución un millón de veces, sin hacer lo mismo dos veces”*

□ Un patrón es:

- Una solución a un problema en un **contexto particular**.
- **Recurrente** (lo que hace la solución relevante a otras situaciones).
- **Enseña** (permite entender cómo adaptarlo a la variante particular del problema donde se quiere aplicar)
- Los patrones **facilitan la reutilización de diseños** y arquitecturas.
- Software que **ha tenido éxito**.

Patrones de Diseño

- Un patrón tiene 4 elementos esenciales:
 - El **Nombre**:
 - Permite identificar en una o dos palabras el patrón.
 - Incrementar vocabulario y compartirlo.
 - El **Problema**:
 - Describe cuando aplicar el patrón, explica el problema y su contexto.
 - A veces el problema incluye las condiciones que deben darse para aplicar el patrón.

Patrones de Diseño

- Un patrón tiene 4 elementos esenciales:
 - La **Solución**:
 - Describe los elementos que constituyen el diseño, sus relaciones, responsabilidades y colaboraciones.
 - No describe una implementación en concreto, sino mas bien como una plantilla general.
 - Las **Consecuencias**:
 - Son los resultados.
 - Ventajas y desventajas de aplicar el patrón.
 - Impacto sobre la flexibilidad, extensibilidad, portabilidad.

Patrones de Diseño

□ Motivación

- **Capturan la experiencia** y la hacen accesible a los no expertos
- El conjunto de sus nombres forma un vocabulario que ayuda a que los desarrolladores se comuniquen mejor
 - Lenguajes de patrones
- Ayudan a la gente a comprender un sistema más rápidamente cuando está documentado con los patrones que usa.
- Los patrones pueden ser la base de un manual de ingeniería de software
 - Si el software se convierte en una ingeniería, las prácticas exitosas deben ser documentadas sistemáticamente y ampliamente difundidas

Patrones de Diseño

□ Motivación

- **Facilitan la reestructuración** de un sistema tanto si fue o no concebido con patrones en mente.
- **Reutilización**
 - Los patrones de diseño soportan la reutilización de arquitecturas software.
- **El software cambia**
 - Para anticiparse a los cambios en los requisitos hay que diseñar pensando en qué aspectos pueden cambiar.
 - Los patrones de diseño están orientados al cambio.

Patrones de Diseño

- Organización del Catalogo de Patrones (según GoF)
 - Están clasificados según 2 criterios:
 - Propósito
 - Ámbito

Tabla 1.1: Patrones de diseño				
Propósito				
		De Creación	Estructurales	De comportamiento
Ámbito	Clase	Factory Method (99)	Adapter (de clases) (131)	Interpreter (225)
	Objeto	Abstract Factory (79) Builder (89) Prototype (109) Singleton (119)	Adapter (de objetos) (131) Bridge (141) Composite (151) Decorator (161) Facade (171) Flyweight (179) Proxy (191)	Template Method (299) Chain of Responsibility (205) Command (215) Iterator (237) Mediator (251) Memento (261) Observer (269) State (279) Strategy (289) Visitor (305)

Patrones de Diseño

- Organización del Catalogo de Patrones (según GoF)
 - Según la relación entre ellos.

