

Patrones de Diseño

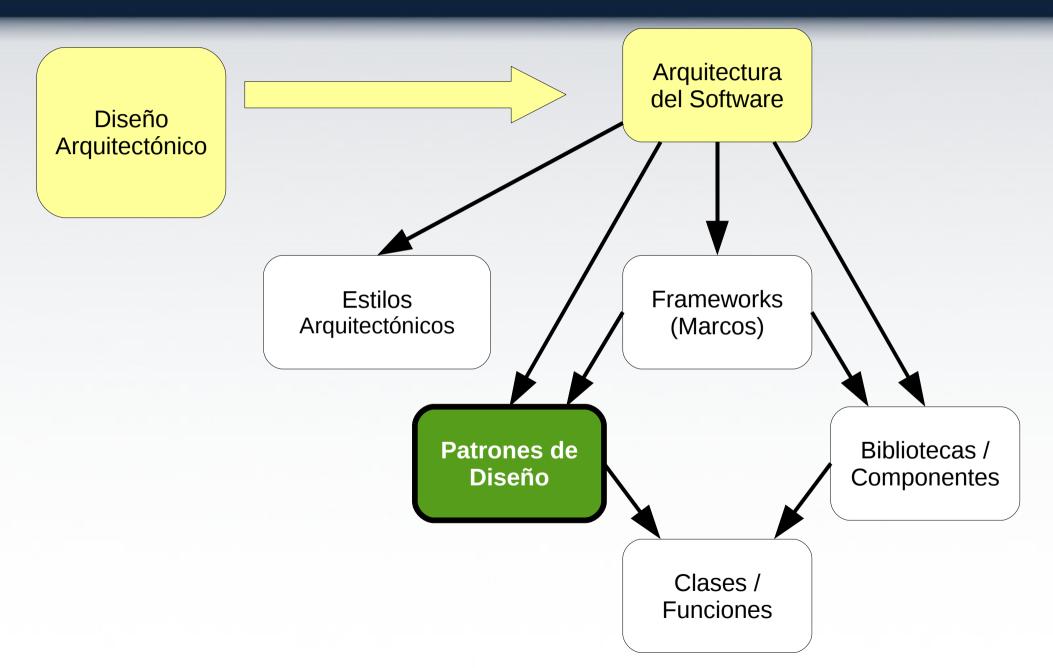
(...o bien, que tiene que ver la costura con el software...)

Universidad de los Andes

Demián Gutierrez Marzo 2010

Diseño Arquitectónico







¿Qué es un patrón?

¿Qué es un patrón de diseño?



Según RAE:

Un patrón es "Modelo que sirve de muestra para sacar otra cosa igual"

¿Qué es un Patrón de Diseño?





Por ejemplo, los patrones (de diseño) se utilizan en la costura

¿Qué es un Patrón de Diseño?

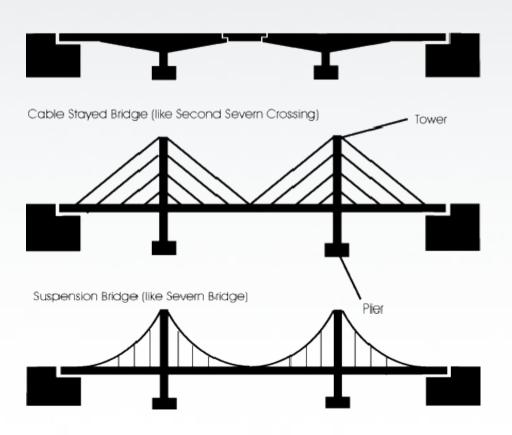




Simple Beam Bridge



Cantilever bridge (like the Forth Rail Bridge)



Los patrones de diseño también se utilizan en la Ingeniería Civil

(¿Recuerda usted esto de estilos arquitectónicos?)

¿Christopher Alexander? Patrones de Diseño en el Área de Arquitectura (Civil)



Christopher Alexander

Who's Who

Archives

<u>Buildings</u>

Paintings

Books

<u>Film</u>

<u>C Vitae</u>

Computing

Wiki

Software Patterns



Christopher Alexander is Professor in the Graduate School and Emeritus Professor of Architecture at the University of California, Berkeley.

He is the father of the Pattern Language movement in computer science, and A Pattern Language, a seminal work that was perhaps the first complete book ever written in hypertext fashion.



Trabajó en el desarrollo de patrones arquitectónicos (para ingeniería civil, no para software)

Acuñó el término "lenguaje de patrones" que no es más que un conjunto de patrones y sus relaciones con los que se puede construir algo en un contexto determinado



Su libro "A Pattern Language" (1977) contiene 253 patrones, mientras que otro de sus libros "The Timeless Way of Building" (1979) va mucho más allá describiendo la motivación y razón de ser de su trabajo



Se refirió a *problemas comunes* de la construcción de edificios y urbanismo y como estos problemas deberían ser resueltos

Las soluciones propuestas incluyen sugerencias que van desde cómo las ciudades y/o pueblos deberían estar estructurados hasta dónde deberían ser ubicadas las ventanas en una habitación



Christopher Alexander define un patrón de diseño como "Una descripción detallada de una solución a un problema recurrente dentro del contexto de un edificio"



Cada patrón describe un problema que ocurre una y otra vez en un determinado contexto y describe una solución al problema de tal manera que esta solución puede ser utilizada millones de veces de forma recurrente



Bien, pero ¿A qué viene todo esto en la Ingeniería del Software?

¿Qué es un Patrón de Diseño? Desde el Punto de Vista del Software



En general, hacer diseño orientado a objetos es una tarea *particularmente compleja* (en especial si se buscan altos niveles de reusabilidad)

Generalmente no es buena idea tratar de resolver todos los problemas desde cero, usualmente es mejor *tratar de reutilizar* soluciones (conceptos) que ya han funcionado en el pasado

Si ya existe, si ya esta hecho, si ya alguien dedicó largas horas a solucionar un problema, ¿Por qué emplear tiempo y esfuerzo para volver a solucionarlo?

¿Qué es un Patrón de Diseño? <u>Desde el Punto de Vista del</u> Software



Los diseñadores (desarrolladores / programadores) experimentados suelen hacer buenos diseños, porque *conocen* (de forma consciente o no) la *forma de resolver los problemas comunes de diseño*

Los diseñadores inexpertos generalmente suelen confundirse o no conocer todas las posibles soluciones a un problema, de modo que usualmente tienen que generar soluciones desde cero



Cuando un ingeniero encuentra una buena solución la *usa una y otra vez* siempre que la necesite



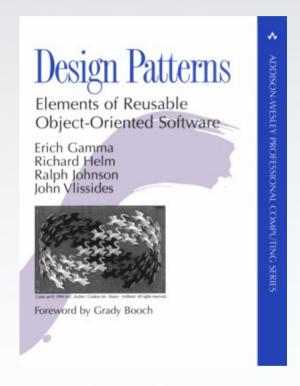


Tiene sentido tratar de utilizar patrones para diseñar e implementar software

Es lógico pensar que existen problemas típicos (con sus respectivas soluciones típicas) que se repiten a lo largo de diferentes sistemas

¿Qué es un Patrón de Diseño?





Gamma, Helm, Johnson, Vlissides, Design Patterns, Addison-Wesley 1ra, Edición Un libro *clásico de la década de los 90* escrito
por the "Gang of Four" que
describe a detalle un
conjunto de 23 patrones
de diseño

Desde entonces se han realizado incontables publicaciones sobre patrones de diseño



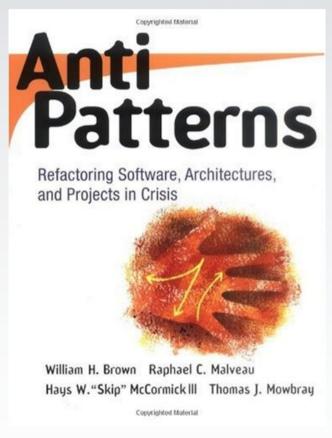
Advertencia: El libro de Patrones de Diseño de GoF no utiliza UML para describir los patrones, utiliza OMT (Object Modeling Technique).

OMT es muy parecido a los diagramas de clases de UML, pero hay varias diferencias a tener en cuenta

http://en.wikipedia.org/wiki/Object-modeling_technique

¿Qué es un Anti-Patrón?





William J. Brown, Raphael C. Malveau, Hays W. McCormick, Thomas J. Mowbray Wiley / 1ra. Edición Si el concepto de patrones (buenas soluciones a problemas conocidos) resulta interesante, puede que sea aún más interesante el concepto de "anti-patrón" (errores comunes solucionando problemas conocidos)



Regresando a los patrones de diseño...

Nombre del Patrón: ¿Cómo se llama el patrón? Es una forma general de describir un problema de diseño sus soluciones y sus consecuencias en una o dos palabras (Comunicación / Vocabulario Técnico)

Descripción del Problema: Una descripción de cuándo (¿En qué casos?) aplicar el patrón. Debe explicar con más detalle el problema y su contexto. La descripción del problema describe en que casos (contexto) debería aplicarse el patrón y en que casos NO debería aplicarse el patrón



Solución: Describe los elementos de diseño que constituyen la solución, sus relaciones, responsabilidades y colaboraciones. Una solución no describe un diseño o implementación concreta para un caso particular, un patrón es más bien una plantilla que describe de forma abstracta de forma que se puede aplicar el patrón a casos particulares

Consecuencias: Son los resultados de aplicar el patrón. Generalmente describen las ventajas y desventajas de utilizar el patrón de diseño, son una descripción detallada que sirve para definir los costos y los beneficios de aplicar el patrón

Nombre v Clasificación

Participantes



Pronósito

Colaboraciones

Tiombre y Clasmoacion	1 Toposito
Otros Nombres	Motivación
Aplicabilidad	Estructura

Consecuencias Implementación

Código de Ejemplo Usos Conocidos

Patrones Relacionados

Segun GoF...

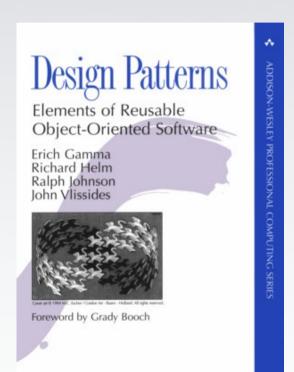


Nombre y Clasificación	Propósito
Otros Nombres	Motivación
Aplicabilidad	Estructura
Participantes	Colaboraciones
Consecuencias	Implementación
Código de Ejemplo	Usos Conocidos

Patrones Relacionados

Segun GoF...





Ejemplo:

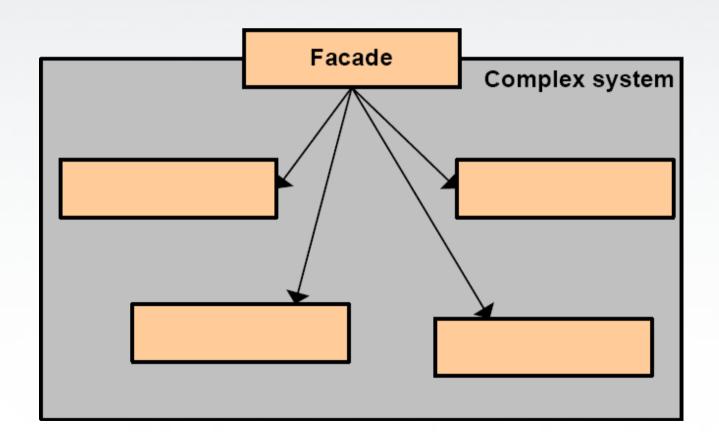
Facade (171 / --95--)

...siguiente lámina antes de ir al libro...

Ejemplo: Patrón Facade



...ocultar una interfaz compleja distribuida a lo largo de múltiples objetos detrás de un único objeto...



¿Cómo se Dividen los Patrones de Diseño? (Según GoF)



Patrones de Creación: Están relacionados con los procesos de creación de objetos (¿Cómo se obtienen y construyen los objetos?)

Patrones Estructurales: Definen estructuras típicas entre clases y objetos (¿Cómo se acoplan, relacionan, componen y comunican un conjunto de clases?)

Patrones de Comportamiento: Caracterizan la forma en que los objetos se distribuyen responsabilidades

Los patrones GOF (Según su Dificultad Aproximada de Aprendizaje)



Fáciles:

Facade

Singleton

Mediator

Iterator

Strategy

Command

Builder

State

Template Method

Factory Method

Memento

Prototype

Intermedios:

Proxy

Decorator

Adapter

Bridge

Observer

Patrones de Comportamiento

Patrones de Estructurales

Patrones de Creación

Difíciles:

Composite

Interpreter

Chain of Responsability

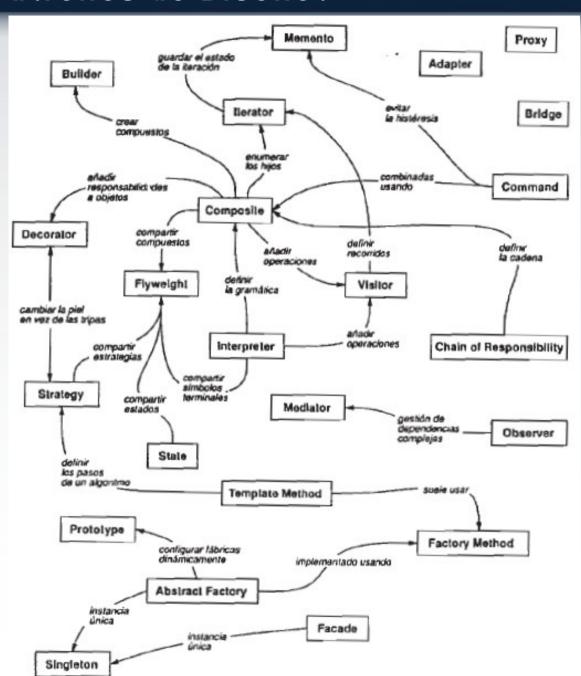
Abstract Factory

Flyweight

Visitor

¿Relaciones Entre los Distintos Patrones de Diseño?





Existen relaciones entre los distintos patrones de diseño, por ejemplo, existe una relación entre un Singleton y un Facade: El Singleton me permite obtener una instancia del Facade

Tomado de: Gamma, Helm, Johnson, Vlissides, Design Patterns, Addison-Wesley 1ra. Edición

Ejemplo: Patrones de Diseño en Acción



Patrones a ver en clases:

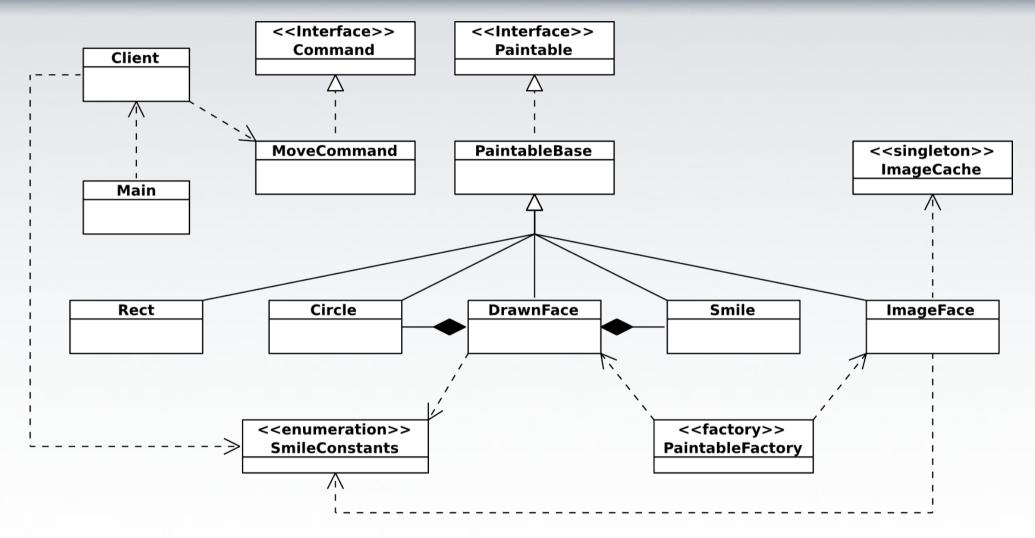
```
Singleton (119 / --69---)
Composite (151 / --85--)
Factory (99 / --59--)
Command (215 / --117--)
Facade (171 / --95--)
```

(Las páginas son referencias a la versión en español del libro de Patrones de Diseño de GoF)

...patrones en el libro...
...ejemplo en java, funcionamiento, código y patrones...

Ejemplo: Patrones de Diseño en Acción

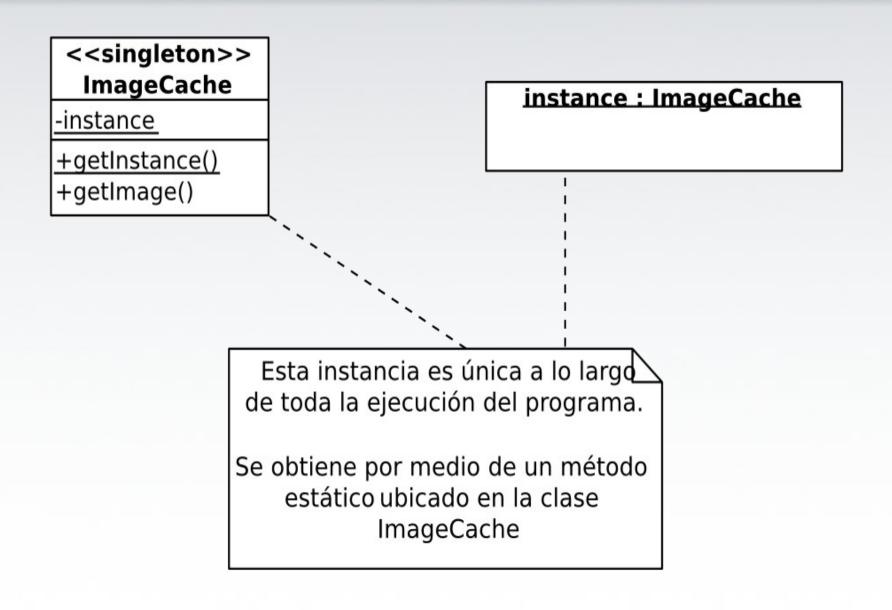




¿Recuerdan el concepto de "Responsabilidad" que vimos en diseño de software? ¿Cuál es la responsabilidad de cada clase?

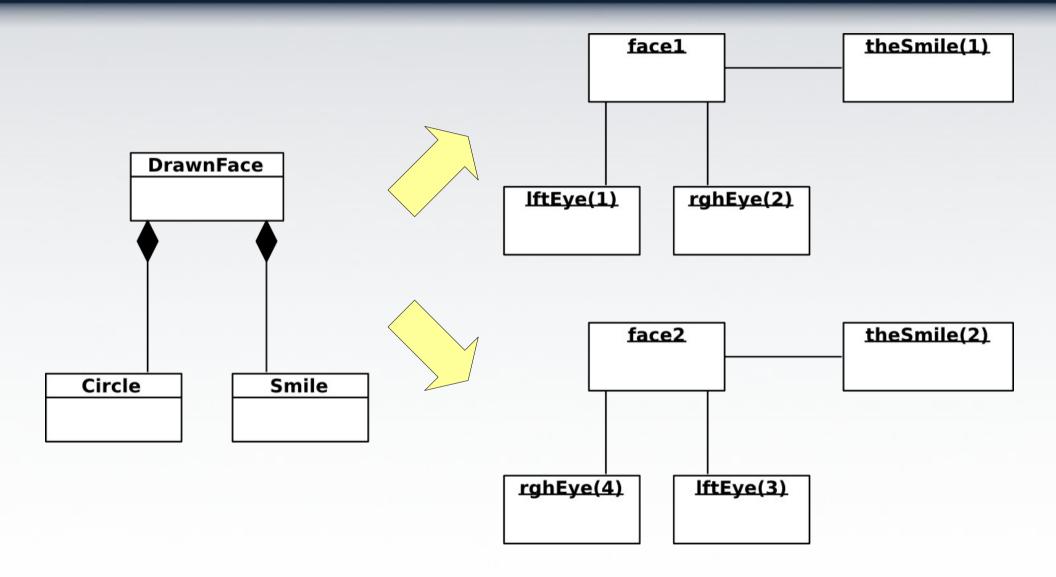
Ejemplo: Patrones de Diseño en Acción (Singleton)





Ejemplo: Patrones de Diseño en Acción (Composite)





El objeto DrawnFace se encarga de manejar los objetos que lo componen: Circle y Smile



En conclusión:

Deje de pensar en términos de código menudo: (if, else, while, for, etcétera)

y piense mas bien en términos de:

- 1) **Estructura:** ¿Cómo organizar el código y distribuir responsabilidades a lo largo de distintas clases
 - 2) Patrones de Diseño: ¿Qué patrones me pueden ayudar a estructurar el código adecuadamente?



¡Gracias!

