



Pontificia Universidad
JAVERIANA
Bogotá

ESPECIALIZACIÓN EN
ARQUITECTURA EMPRESARIAL DE SOFTWARE

Patrones de Software

Juan Carlos Cerón Barreto

ceron.juan@javeriana.edu.co



Agenda

- Patrones Estructurales
 - Layers



Pontificia Universidad
JAVERIANA
Bogotá

ESPECIALIZACIÓN EN
ARQUITECTURA EMPRESARIAL DE SOFTWARE

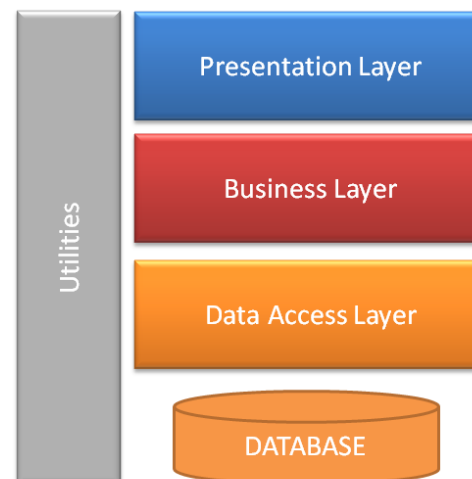
PATRONES ESTRUCTURALES





Layers

- Es un patrón de arquitectura que apoya la estructura de la aplicación.
- Promueve la descomposición en grupos de subtarefas
- Cada uno de los grupos representa un nivel de abstracción





Layers - Principios de Diseño

Principios SOLID

- ☐ **S**ingle Responsibility Principle
- ☐ **O**pen Close Principle
- ☐ **L**iskov Substitution Principle
- ☐ **I**nterface Segregation Principle
- ☐ **D**ependency Inversion Principle



Layers

Contexto

Problema

Fuerzas

Solución

Consecuencias



Contexto

- Diseñar una aplicación empresarial compleja compuesta por un número considerable de componentes de diferentes niveles de responsabilidad.



Problema

- Estructurar una aplicación para soportar requerimientos complejos operacionales y disponer de una buena mantenibilidad.



Fuerzas

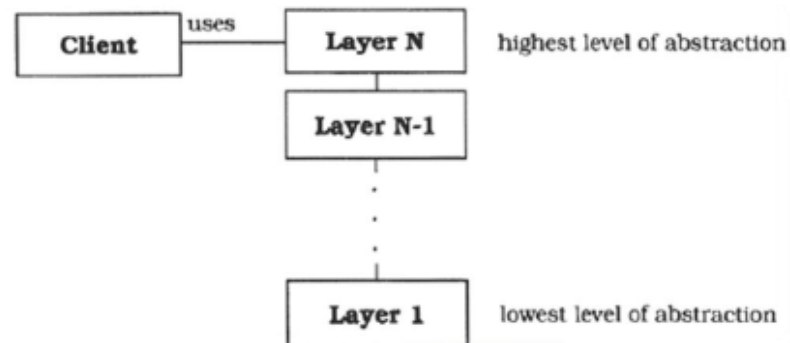
- Localización de Cambios
- Roles y Funcionalidades
- Reutilización (Interna-Externa)
- División del trabajo
- Despliegue y actualización independiente
- Realización de Pruebas



Solución

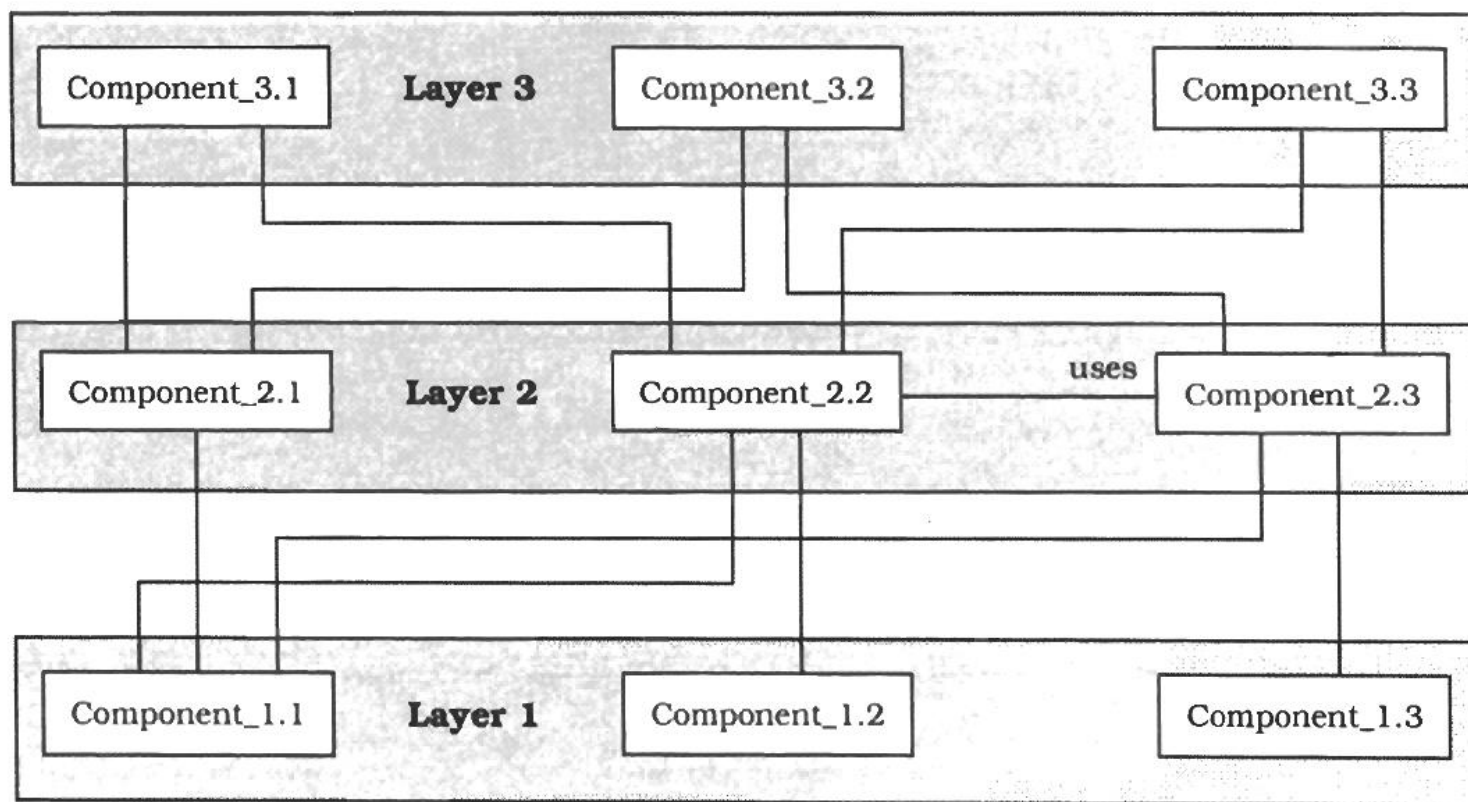
- Estructurar la aplicación de modo que pueda ser descompuesta en grupos de subtarefas, donde cada grupo tiene un nivel de abstracción particular

Class	Collaborator
Layer J	• Layer J-1
Responsibility <ul style="list-style-type: none">• Provides services used by Layer J+1.• Delegates subtasks to Layer J-1.	





Solución



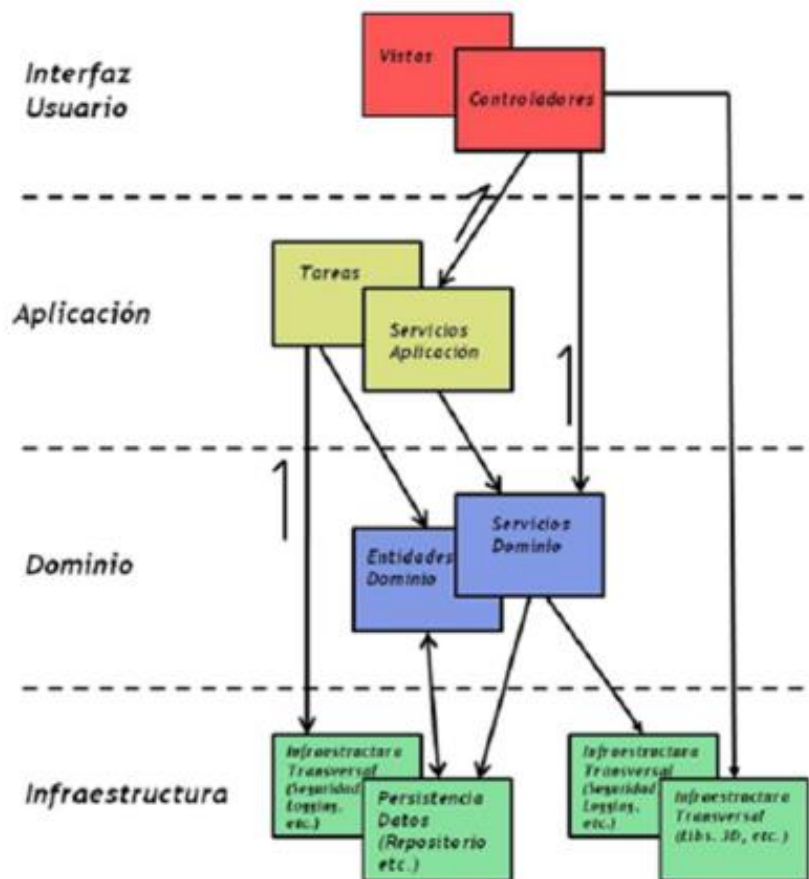


Diseño en Capas

- 1. Top-Down
- 2. Bottom-up
- 3. Middle
- 4. Flujos (1) (2)



Diseño en Capas



Estricto

Laxo

Herencia



Implementación

1. Determine la estrategia de agrupamiento
 - Granularidad
 - Separación lógica o física
 - Trade-offs atributos de calidad
2. Determine las capas requeridas
3. Distribución de layers en tiers
4. Determine si necesita colapsar capas
5. Estructura capas de forma individual
6. Manejo de Errores



Implementación

7. Funcionalidades transversales
8. Determine las reglas de interacción
 - Estricta
 - Interacción débil
9. Interfaces entre las capas
 - Abstract Interface
 - Dependency Inversion
 - Mensajes
10. Protocolos de Comunicación



Consecuencias

- Cambios en cascada
- Baja eficiencia
- Trabajo innecesario
- Nivel correcto de granularidad



Bibliografía

- Pattern-Oriented Software Architecture: A System of Patterns, F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. John Wiley & Sons, 1996
- Patterns of Enterprise Application Architecture: Martin Fowler, Addison-Wesley Professional, 1 edition ,November 15, 2002.



Preguntas

