

# Curso de Programación en Java



Juan Francisco Maldonado León  
**Arquitecto de Software**



# Java

# Patrones de Diseño



Juan Francisco Maldonado León  
**Arquitecto de Software**



## **Patrones de Diseño**

### Historia

Christopher Alexander, aportó al mundo de la arquitectura el libro A Pattern Language; en él proponía el aprendizaje y uso de una serie de patrones para la construcción de edificios.

Su uso iba desde la forma de estructurar una ciudad a como debían disponerse las ventanas en una habitación.

## Patrones de Diseño

### Historia

"Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces..."

Christopher Alexander

## Patrones de Diseño

### Historia

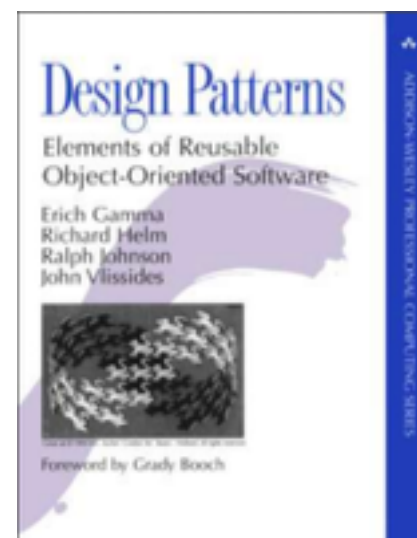
En 1987 Ward Cunningham y Kent Beck se dieron cuenta del paralelo que existía entre la buena arquitectura propuesta por Alexander y la buena arquitectura OO.

Publicaron el artículo *Using Pattern Languages for OO Programs*.

## Patrones de Diseño

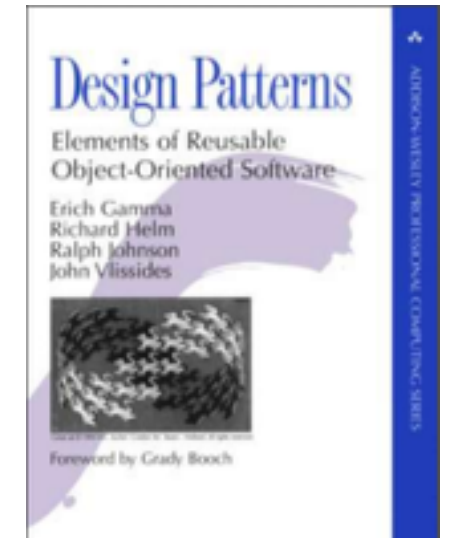
### Historia

En 1994 el grupo Gang of Four (La banda de los cuatro) compuesto por Erich Gamma, Richard Helm, Ralph Johnson y John Vlissides, publicaron el libro Design Patterns.



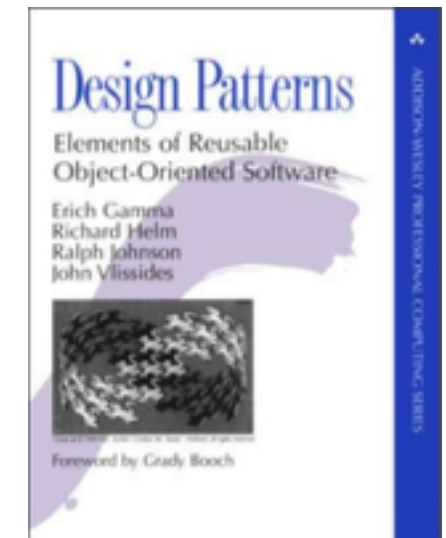
## Patrones de Diseño

### Historia



“Un patrón de diseño nombra, abstrae e identifica los aspectos fundamentales de una estructura común de diseño que la hacen útil para la creación de un diseño orientado a objetos reutilizable”





## Patrones de Diseño

gof

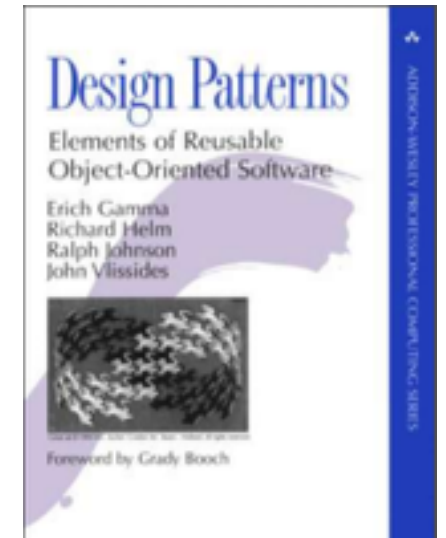
Ellos recopilaron y documentaron 23 patrones de diseño

Nivel/Tipo	Creación	Estructura	Comportamiento
<b>Clase</b>	Factory method/ Método de fabricación	Adapter/Clase adaptadora	Interpreter/Intérprete Template method/Método plantilla
<b>Objeto</b>	Abstract factory/ Fábrica abstracta Builder/Constructor virtual Prototype/Prototipo Singleton/Ejemplar único	Adapter/Objeto adaptador Bridge/Puente Composite/Objeto compuesto Decorator/Decorador Facade/Fachada Flyweight/Peso ligero Proxy/Intermediario	Chain of responsibility/ Cadena de responsabilidad Command/Orden Iterator/Iterador Mediator/Mediador Memento/Recuerdo Observer/Observador State/Estado Strategy/Estrategia Visitor/Visitante



## Patrones de Diseño

### Estructura según gof



- **Nombre**
  - Suele usarse el original en inglés del libro de GoF
- **Propósito**
  - Descripción de la situación problemática o compleja en la que parece conveniente aplicar el patrón
- **Solución**
  - Estructuras de diseño involucradas
  - Elementos y Relaciones entre estos elementos
  - Responsabilidades y colaboraciones que mantienen estos elementos
- **Consecuencias**
  - Discusión acerca de las ventajas y los inconvenientes del patrón



# Introducción a Java

Patrones de Diseño

## Patrones de Diseño

### Características

- 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)
- tiene un nombre para referirse al patrón

## **Patrones de Diseño**

### Síntesis

Los patrones de diseño son soluciones para problemas típicos y recurrentes que nos podemos encontrar a la hora de desarrollar una aplicación.

## Singleton

### Patrones Creaciones

### Propósito

Asegurar que una clase tiene una sola instancia y proporcionar un punto de acceso global a ella

### Solución

- El constructor de la clase DEBE SER PRIVADO
- Se proporciona un método ESTÁTICO en la clase que devuelve LA ÚNICA INSTANCIA DE LA CLASE:  
`getInstance()`



# Introducción a Java

Patrones de Diseño

## Singleton

### Patrones Creaciones

Singleton
- <u>singleton : Singleton</u>
- Singleton()
+ <u>getInstance() : Singleton</u>



**fcfm**

Ciencias de la  
Computación  
FACULTAD DE CIENCIAS  
FÍSICAS Y MATEMÁTICAS  
UNIVERSIDAD DE CHILE

## **Abstract Factory**

### Patrones Creaciones

### **Propósito**

Debemos crear diferentes objetos, todos pertenecientes a la misma familia.



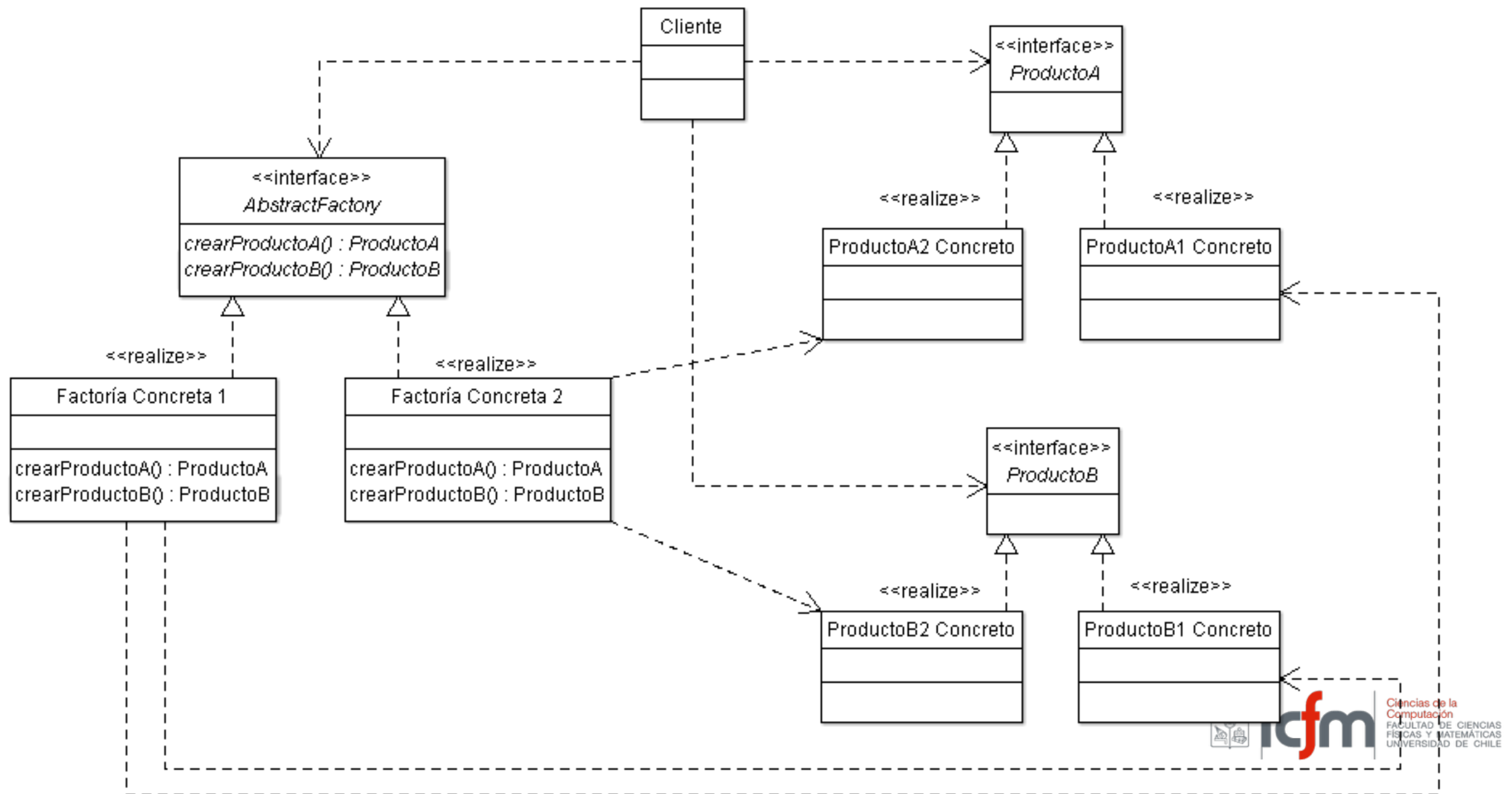


# Introducción a Java

Patrones de Diseño

## Abstract Factory

### Patrones Creaciones



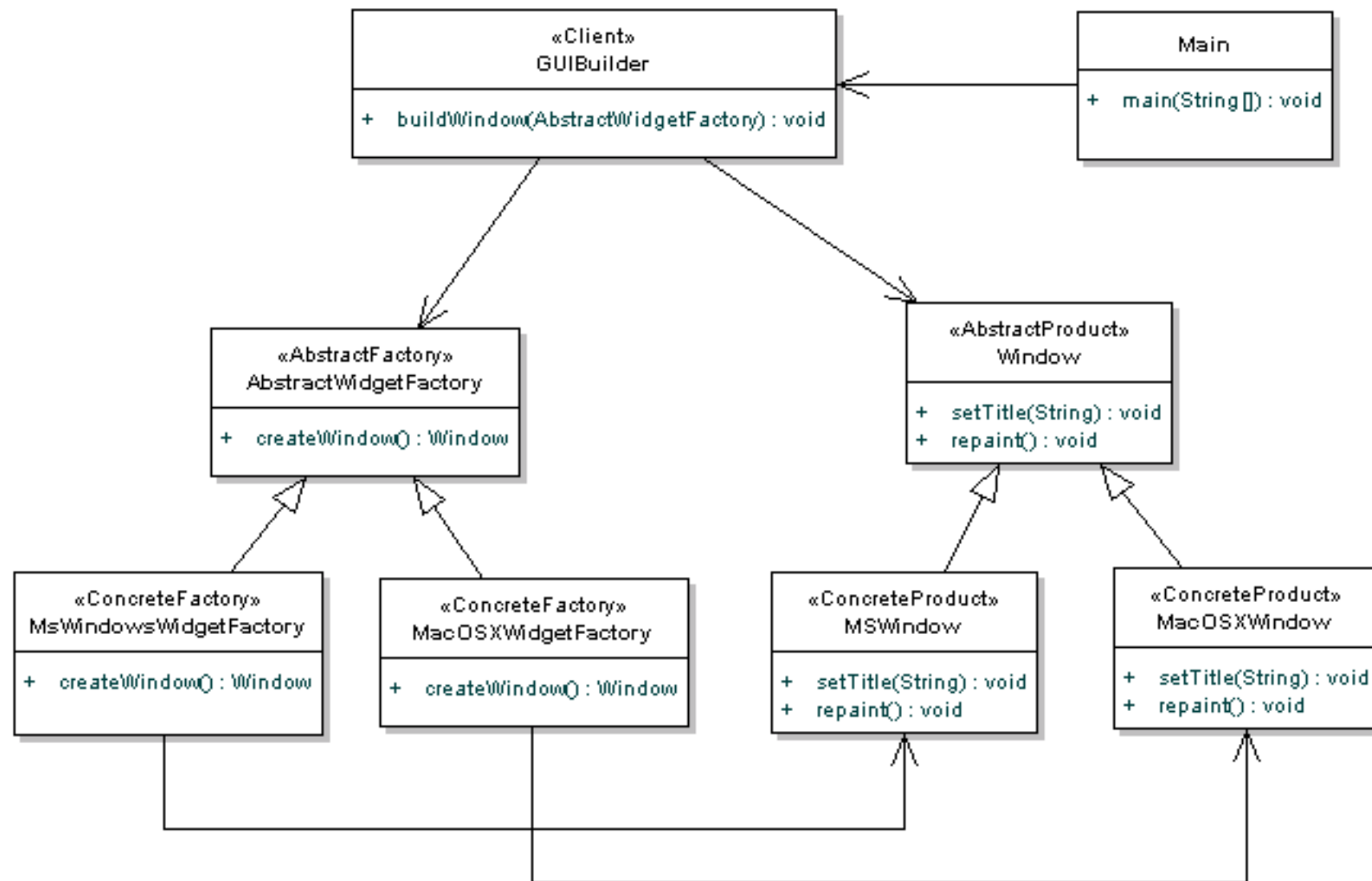


# Introducción a Java

Patrones de Diseño

## Abstract Factory

### Patrones Creaciones



**fcfm**

Ciencias de la  
Computación  
FACULTAD DE CIENCIAS  
FÍSICAS Y MATEMÁTICAS  
UNIVERSIDAD DE CHILE

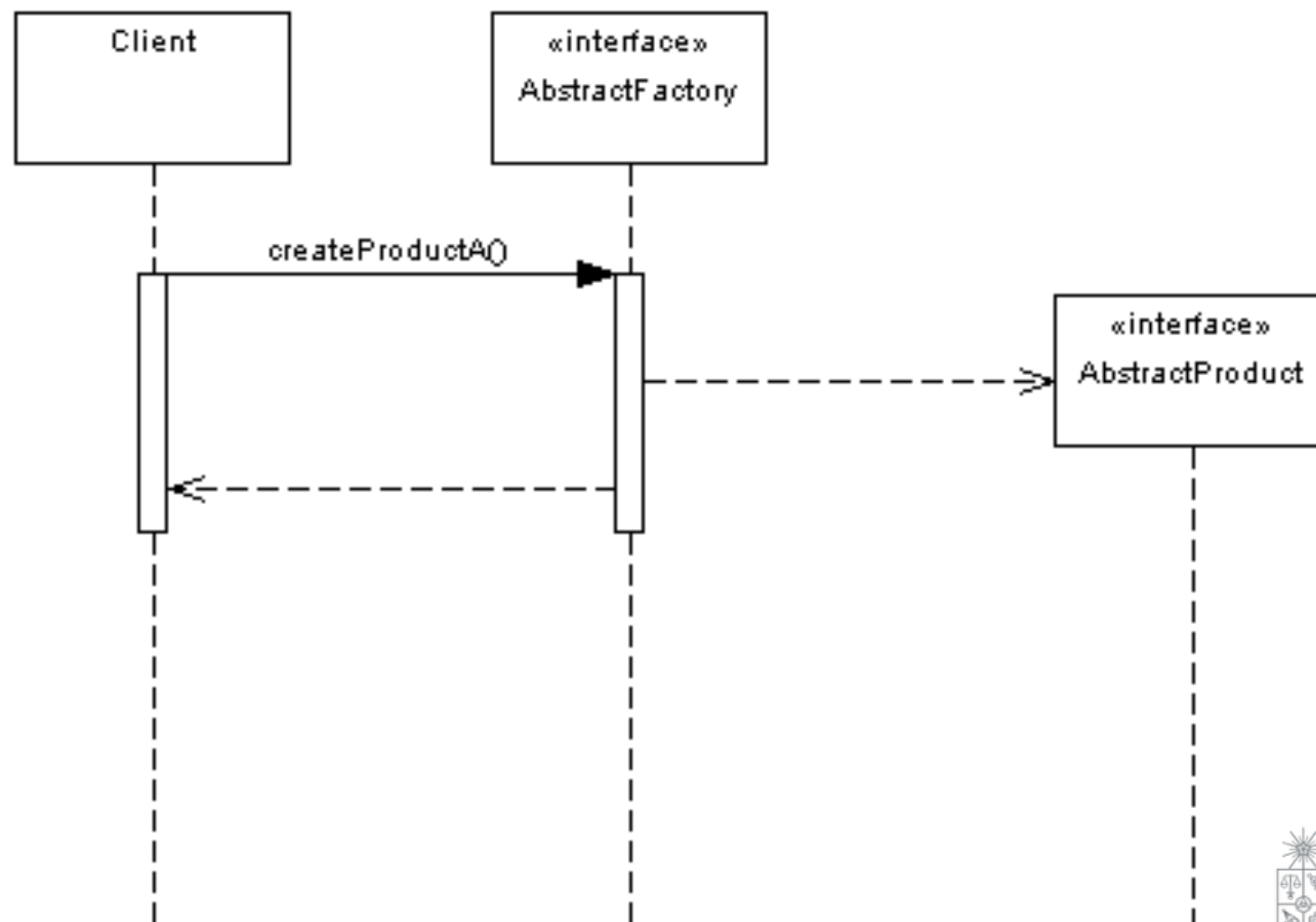


# Introducción a Java

Patrones de Diseño

## Abstract Factory

Patrones Creaciones



**fcfm**

Ciencias de la  
Computación  
FACULTAD DE CIENCIAS  
FÍSICAS Y MATEMÁTICAS  
UNIVERSIDAD DE CHILE