

# Abstract Factory

*Angelo Afeltra*

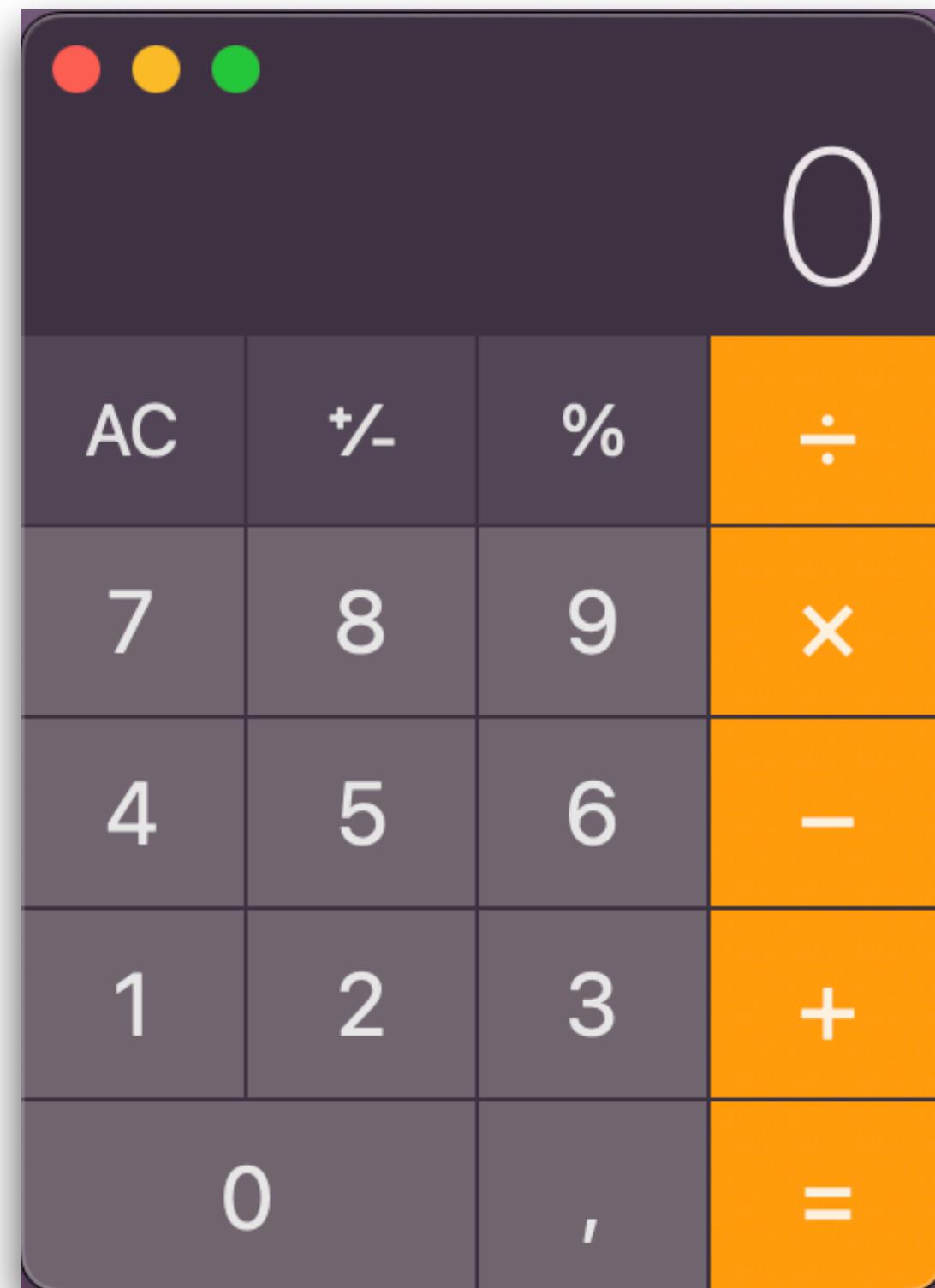
# Intent

Fornisce un interfaccia per creare una famiglia di oggetti correlati o dipendenti senza specificare la loro classe concreta.

# Problema

Immaginiamo di dover sviluppare una calcolatrice che cambia i propri widget in base al sistema operativo su cui viene eseguita.

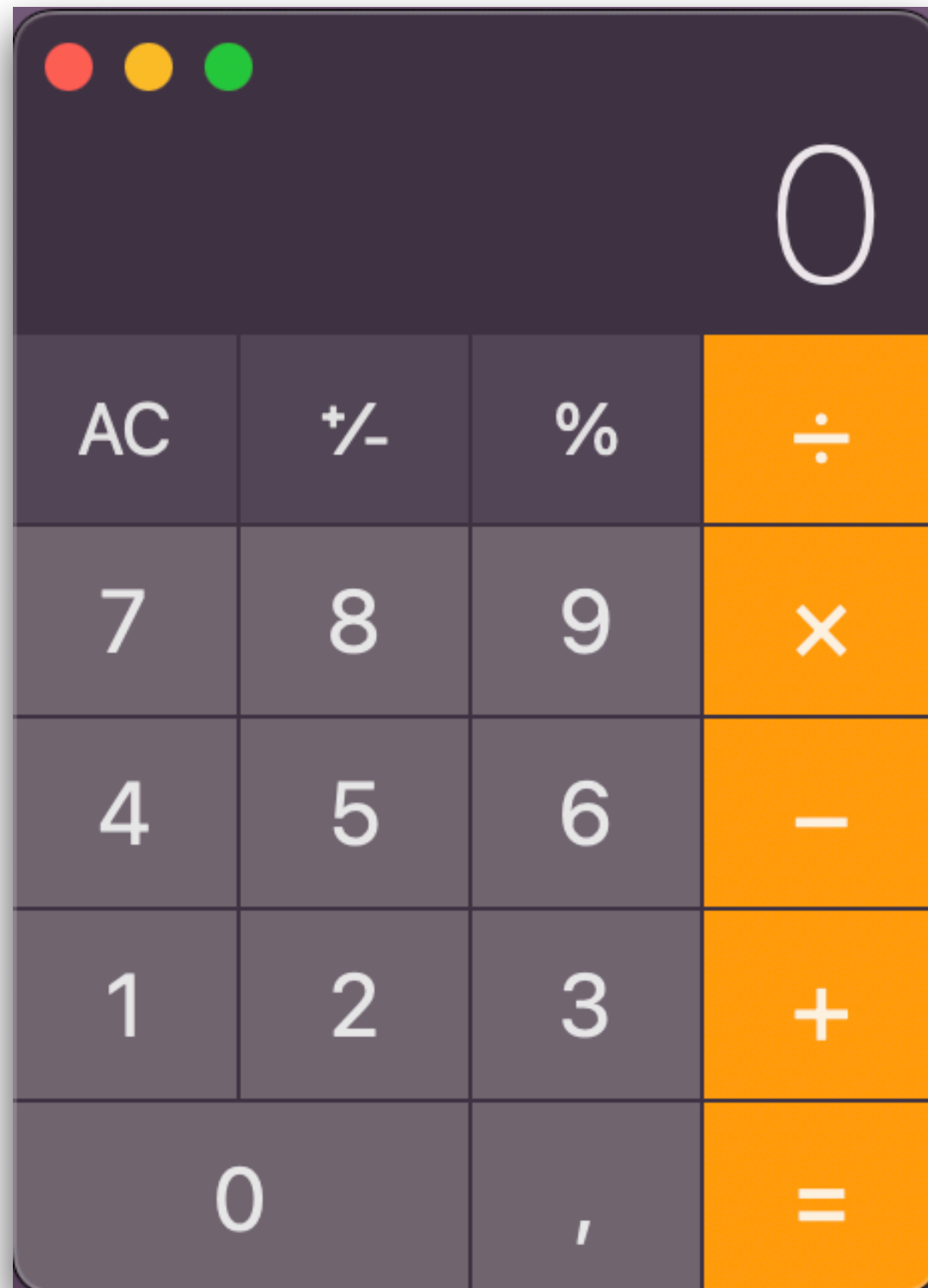
**Calcolatrice Mac**



**Calcolatrice Windows**



# Problema



Main:

```
Calcolatrice calcolatrice = new Calcolatrice()  
Se il sistema operativo è Windows:  
    calcolatrice.add(new WindowsButton(0))  
...  
    calcolatrice.add(new WindowsDisplay())  
Se il sistema operativo è Mac:  
    calcolatrice.add(new MacButton(0))  
...  
    calcolatrice.add(new MacDisplay())
```



# Problema



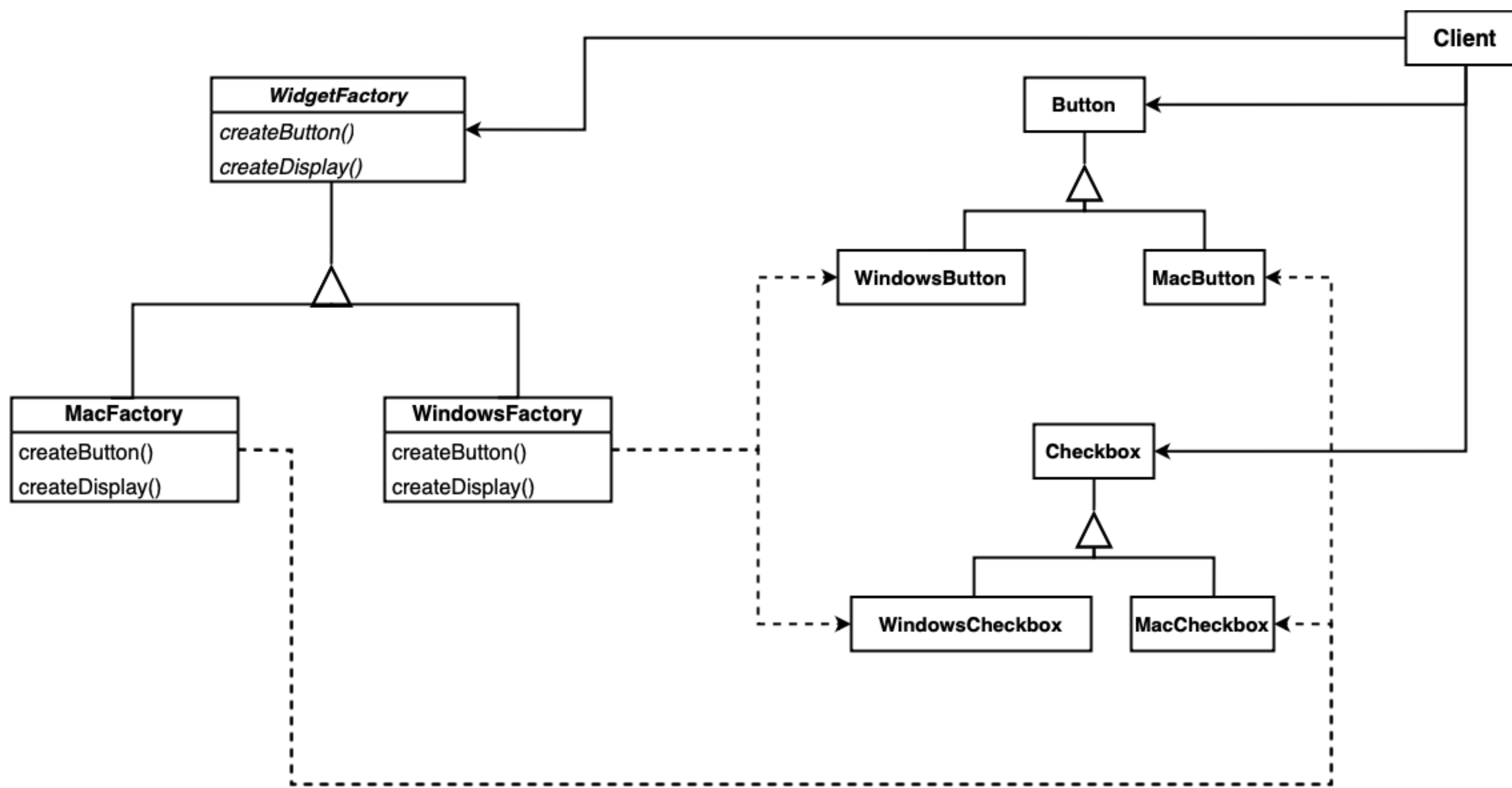


# Problema



E possibile superare questa problematica attraverso l'implementazione di una classe astratta denominata "WidgetFactory". Questa classe astratta offre un'interfaccia per la creazione di diversi tipi di componenti UI. Per ciascun tipo di componente, si definisce una classe astratta corrispondente, oltre a numerose sottoclassi concrete che implementano il componente specifico per uno standard particolare. I client che utilizzano la WidgetFactory sono in grado di istanziare componenti senza dover conoscere le classi concrete, garantendo così un alto grado di flessibilità e facilità di manutenzione nell'adattamento dell'aspetto dei componenti UI.

# Motivazione



# Quando usare l'Abstract Factory?

Il sistema deve essere indipendente da come i suoi prodotti sono creati, composti e rappresentati

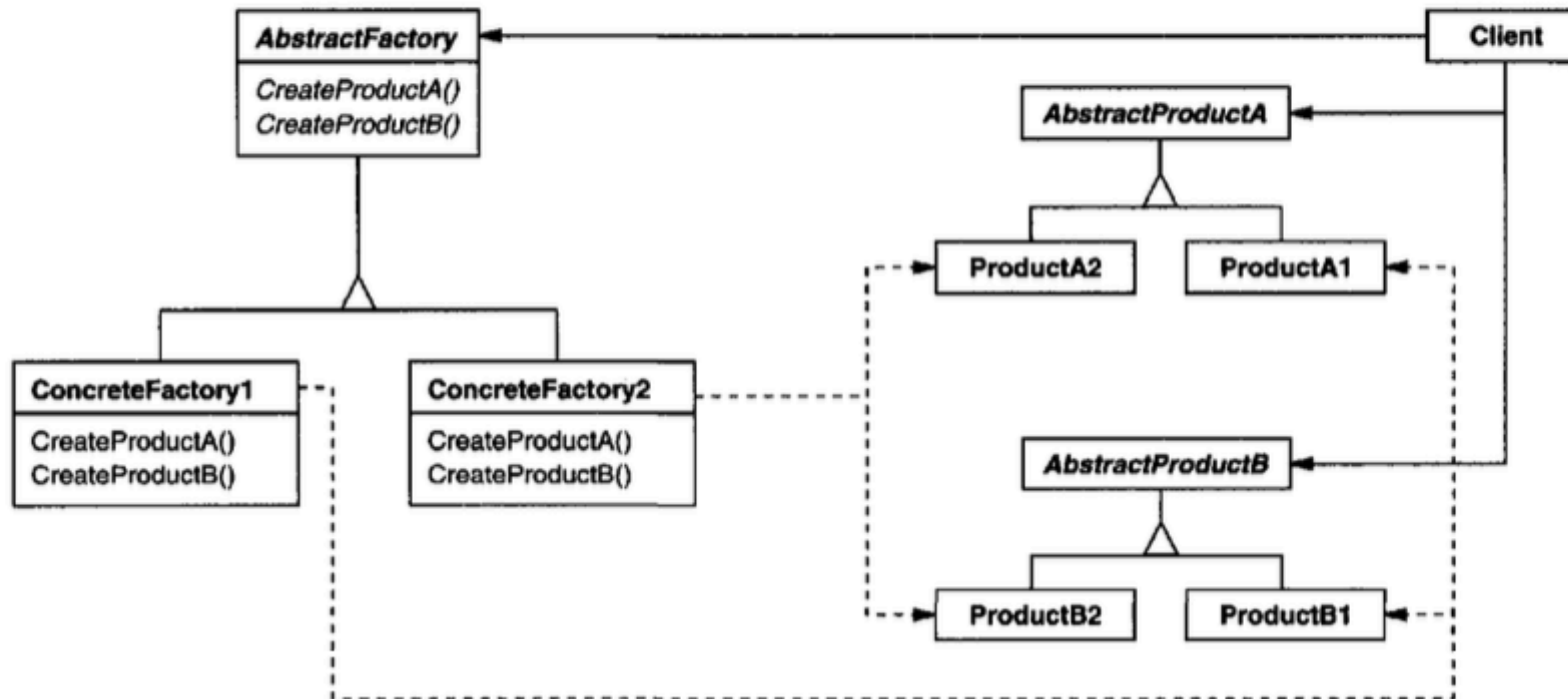
Il sistema deve configurarsi con più famiglie di prodotti

Una famiglia di prodotti correlati è progettata per essere utilizzata insieme ed è necessario applicare questo vincolo

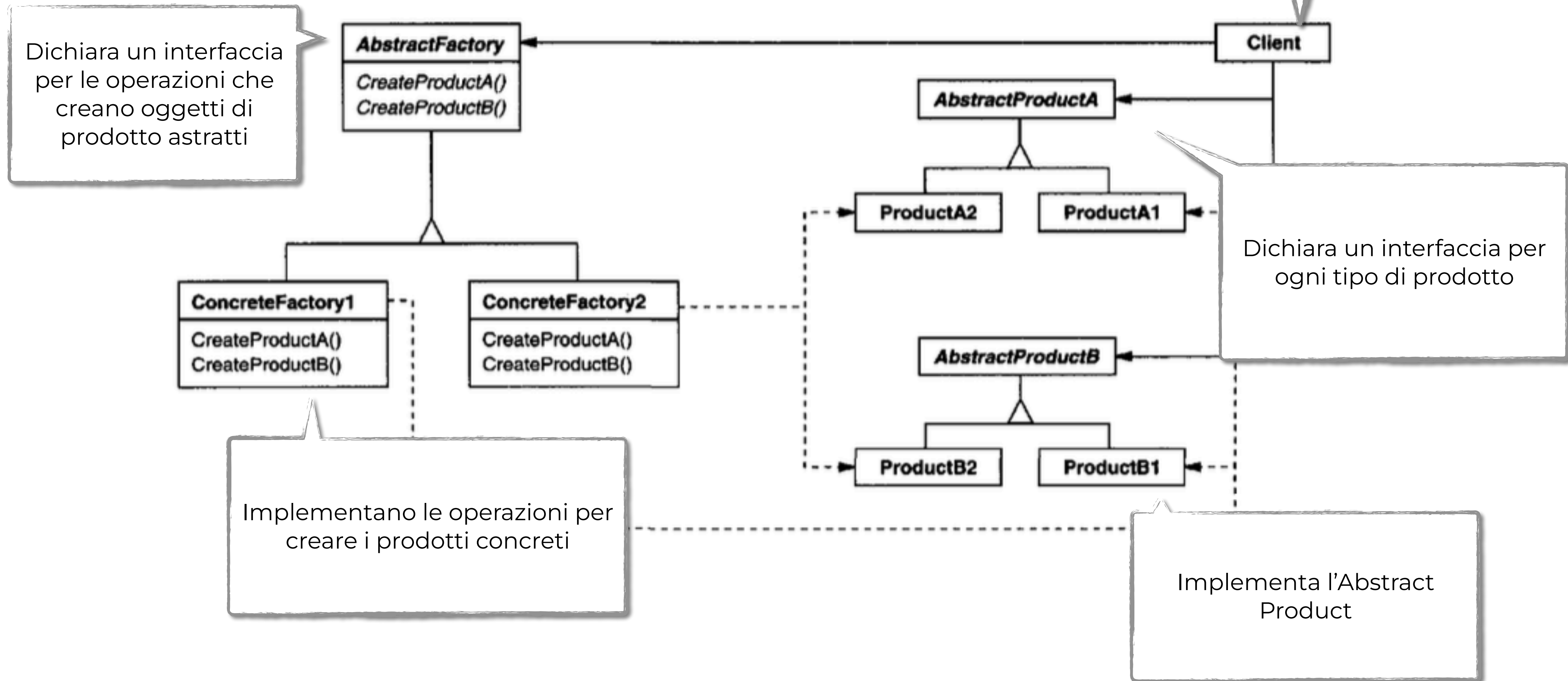
Si vuole rilasciare una libreria contenente classi di prodotti, senza rilevare le loro implementazioni ma solamente le rispettive interfacce



# Struttura



# Struttura



# Conseguenze

## PRO

Isola le classi concrete

Rende facile lo scambio di  
famiglie di prodotti

Promuove la consistenza tra  
prodotti

## CONTRO

Aggiungere nuove tipologie  
di prodotti è difficile

# Conseguenze

## PRO

Isola le classi concrete

Rende facile lo scambio di famiglie di prodotti

Promuove la consistenza tra prodotti

## CONTRO

L'Abstract Factory consente di controllare le classi di oggetti creati da un'applicazione. Poiché una fabbrica incapsula la responsabilità e il processo di creazione di oggetti prodotto, isola i client dalle classi di implementazione. I client manipolano le istanze attraverso le loro interfacce astratte.

# Conseguenze

## PRO

Isola le classi concrete

Rende facile lo scambio di famiglie di prodotti

Promuove la consistenza tra prodotti

## CONTRO

La classe di una concrete factory appare solamente una volta in un'applicazione, cioè dove viene istanziata. Questo rende facile cambiare una concrete factory



# Conseguenze

## PRO

Isola le classi concrete

Rende facile lo scambio di famiglie di prodotti

Promuove la consistenza tra prodotti

## CONTRO

Quando gli oggetti di prodotto di una famiglia sono progettati per funzionare insieme, è importante che un'applicazione utilizzi oggetti di una sola famiglia alla volta. AbstractFactory lo rende facile da applicare.

# Conseguenze

## PRO

Estendere le fabbriche astratte per produrre nuovi tipi di prodotti non è facile. Il supporto di nuovi tipi di prodotti richiede l'estensione dell'interfaccia di fabbrica, il che comporta la modifica della classe AbstractFactory e di tutte le sue sottoclassi.

prodotti

## CONTRO

**Aggiungere nuove tipologie di prodotti è difficile**

Intent

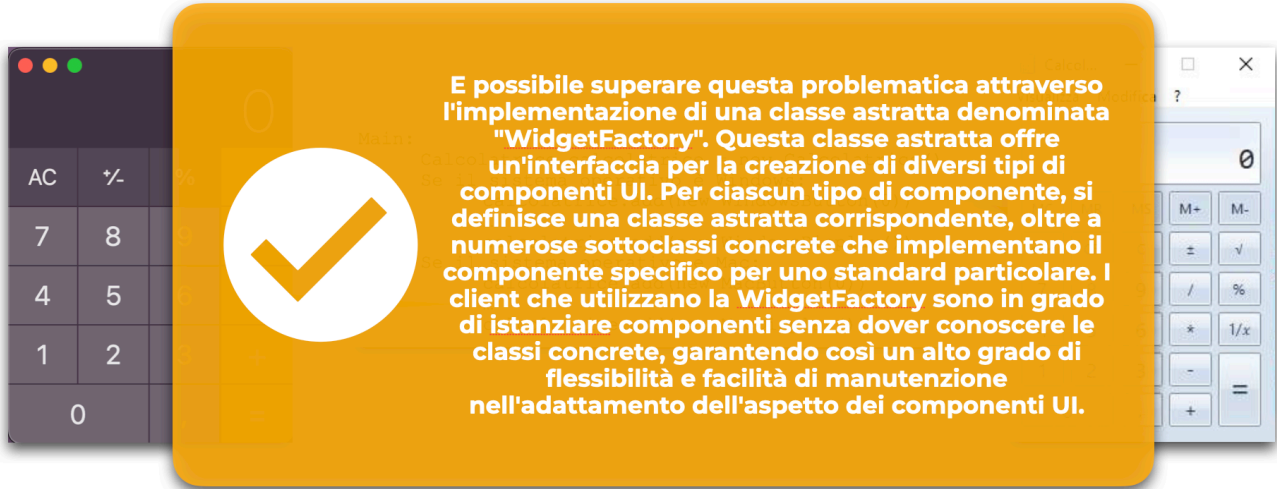
Fornisce un'interfaccia per creare una famiglia di oggetti correlati o dipendenti senza specificare la loro classe concreta.

Problema

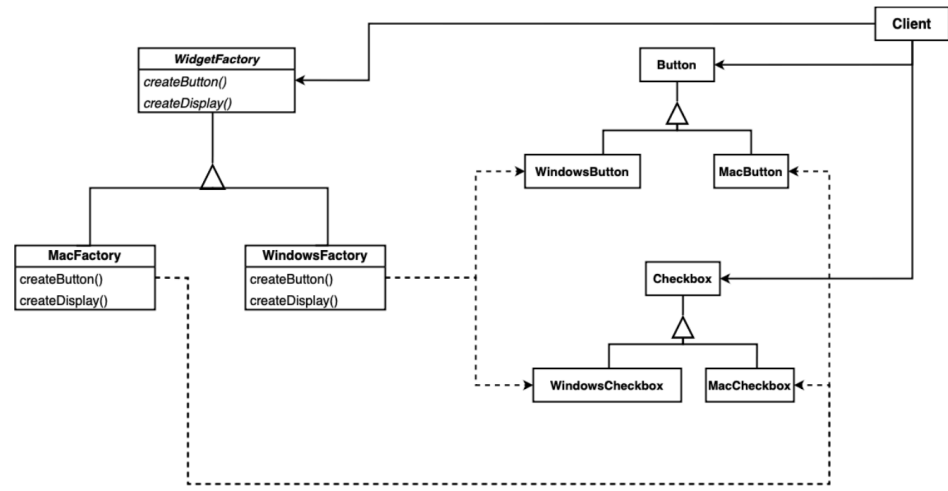
Immaginiamo di dover sviluppare una calcolatrice che cambia i propri widget in base al sistema operativo su cui viene eseguita.



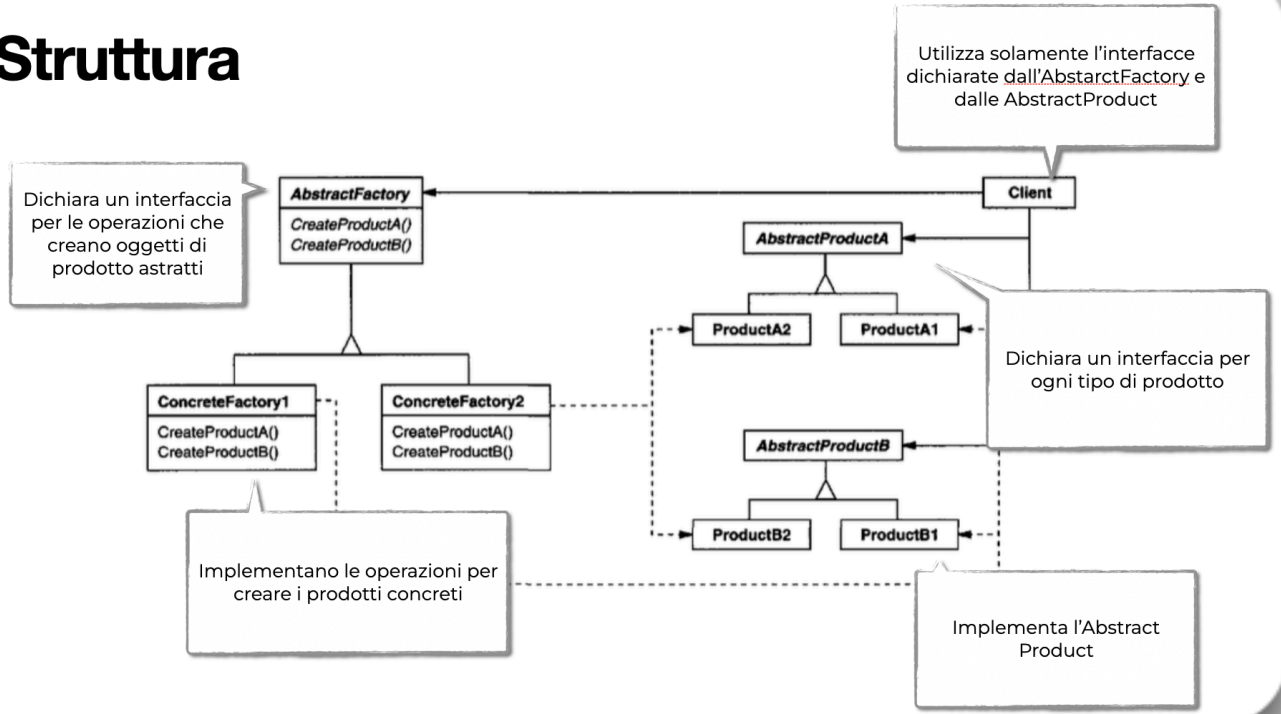
Problema



Motivazione



Struttura



Conseguenze

PRO

- Isola le classi concrete
- Rende facile lo scambio di famiglie di prodotti
- Promuove la consistenza tra prodotti

CONTRO

Aggiungere nuove tipologie di prodotti è difficile

# Abstract Factory