

1. Abstract Factory

У будь-якому разі, використовуючи спорідненні інтерфейси чи незалежні ієрархії інтерфейсів, у нас наплодиться дуже багато інтерфейсів фабрик і, в десять раз більше, конкретних фабрик

```
IHumanFood_Volume_Expirable_Factory
    Volume_Expirable_Dairy_Factory чи
    MilkFactory?
    BewarageFactory
IHumanFood_Weight_Expirable_Factory
    Weight_Expirable_Dairy_Factory чи
    CheeseFactory?
    MeatFactory
```

2. Factory Method

Також, на мою думку, буде створено багато класів.

3. Builder

Для продуктів білдер, на мою думку, не потрібен, так як не передбачено створення великої кількості однакових об'єктів, тобто достатньо одного об'єкта на один продукт.

Для Storage, на мою думку, така ж само ситуація.

4. Singleton

Singleton Storage має нюанс в тому, що він є дженеріком. Тобто патерн працює, є вірогідність створення великої кількості об'єктів (при тому, що потенційно існує десятки інтерфейсів, які наслідуються від IProduct), що знищує концепт паттерна.