

Builder



- 包裝複雜的建構邏輯，產生多種型態的物件
- Client只要透過Director取得建構出的物件
- Client需要知道要產生的物件的型態，但不需知道如何建構

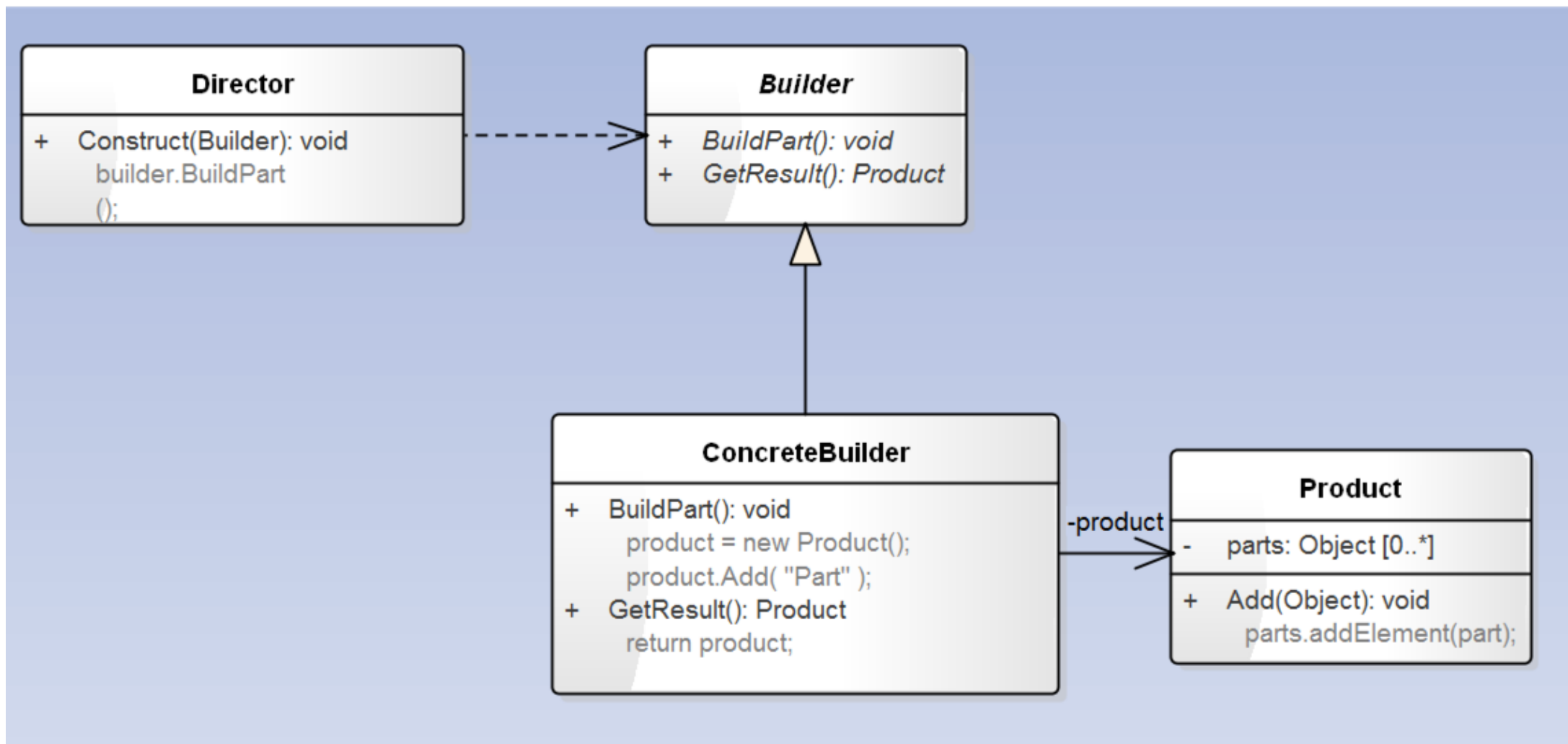
問題描述

當一個Business的概念(Concept)有複雜的結構，且會根據不同的狀況，產生不同的建構方式時，此時，可以運用Builder Pattern來進行概念的建構。

解決方案

利用Builder Interface把複雜的Create物件的過程分成多個BuildPart的Method
由Director Class決定使用哪個特定的ConcreteBuilder來建構物件結構
ConcreteBuilder會決定要回傳的Product Result為何

結構



使用Builder Pattern有以下假設點：

- Client需要認識所有的ConcreteBuilder(Director在操作時，是透過Client Pass ConcreteBuilder給他來操作的)
- Builder Interface必須窮舉所有的BuildPart Method
- Product並不需要是抽象化的一般化類別，相反地，可能是任何結構的物件

Builder談的是把「建構行為」抽象化，因此，其重點在於Director物件的建構一般化，而非結構的一般化

案例描述

假設要開發一個辦公室表單自動化的系統，已知要自動化的表單包括請假單，費用核銷單，資產申請單...等。

要如何設計一個一般化的表單，讓上述的單據可以很簡單的產生呢？

辦公室自動化範例

