

系統分析 95 (地特)

一、請扼要說明下列可以替代傳統資訊系統發展生命週期 (SDLC) 的發展方法：
(每小題 5 分，共 25 分)

(一) 雛型法 (Prototyping)

(二) 快速應用發展法 (Rapid Application Development)

(三) 聯合應用設計法 (Joint Application Design)

(四) 物件導向發展法 (Object-Oriented Development)

(五) 元件導向發展法 (Component-based Development)

(一) 雛型法 (Prototyping): 調 94.5

(二) 快速應用發展法 (Rapid Application Development): 調 96.1、高考 100.1

(三) 聯合應用設計法 (Joint Application Design): 高考 95.1

(四) 物件導向發展法 (Object-Oriented Development): 調 92.3

✓ (五) 元件導向發展法 (Component-based Development):

(1) 以物件導向方式來開發元件，也可以直接使用現存元件以節省開

發時間，具有組合 (composition) 及重用 (reuse) 的特性。

(2) 快速整合各種可以重複使用的物件模組，形成一個新的、獨特的應

用程式，例如整合使用者介面和訂單模組，形成一個線上購物系統。

(3) 使得系統發展時程縮短、成本降低、生產力提昇、容易客製化和

維護，促使提昇軟體再用性和可維護性。

(4) 軟體元件通常可分為白箱元件 (white box components) 和黑箱元件 (black

box components)，前者以原始碼呈現，後者以二進位碼呈現，只能從

文件的描述得知其功能。黑箱元件雖不能直接修改元件，但可以

透過建立新包裹元件擴充原來元件功能，而不更動原來元件功能

，使得原來元件升級後仍可使用，具有較佳的可維護性。

(5) CBD過程中，需要尋找適當元件進行修改整合，如何促進擴大元件

供應，以利後續發展過程中的修改和整合，成了元件式軟體發

分數

題號

展最核心的問題。

(6) 元件開發前的系統分析工作必須從領域分析著手，找出相同領域中不同應用間的相同部分，再進行程式的分析設計，才能確保所開發的元件能在後續的發展過程中被 reuse。