

第13章 設計模型

- 13-1 再談物件導向設計
- 13-2 建立互動圖
- 13-3 建立狀態機圖
- 13-4 設計模型的類別圖
- 13-5 物件設計：類別正規化
- 13-6 本章總結





13-2 再談物件導向設計

- 13-1-1 物件導向設計的主要工作
- 13-1-2 分析與設計類別的差異
- 13-1-3 設計類別的特點



13-1-1 物件導向設計的主要工作- 互動塑模

- 物件導向設計就是在執行類別屬性、行為和類別關係的細部設計來建立設計模型（**Design Model**）的類別圖。
- 物件導向設計的主要工作是進行「互動塑模」（**Interaction Modeling**），其主要目的如下所示：
 - 指派實體、控制和邊界物件的行為。
 - 顯示實體、控制和邊界物件之間的詳細互動過程。
 - 決定類別操作。

13-1-1 物件導向設計的主要工作- 設計類別的來源

- 設計類別的來源主要有兩種，如下所示：
 - 更新分析類別：這是互動塑模的主要工作，我們可以透過建立互動圖來進一步更新分析類別成為設計類別。
 - 類別函數庫（**Class Library**）：在設計模型加入類別函數庫的類別，這是一些解決方案的現成元件，可以如同零件一般的加入設計模型的類別圖，而不用自行建立，例如：時間、日期、字串和集合類別等。

13-1-1 物件導向設計的主要工作-建立的UML圖形

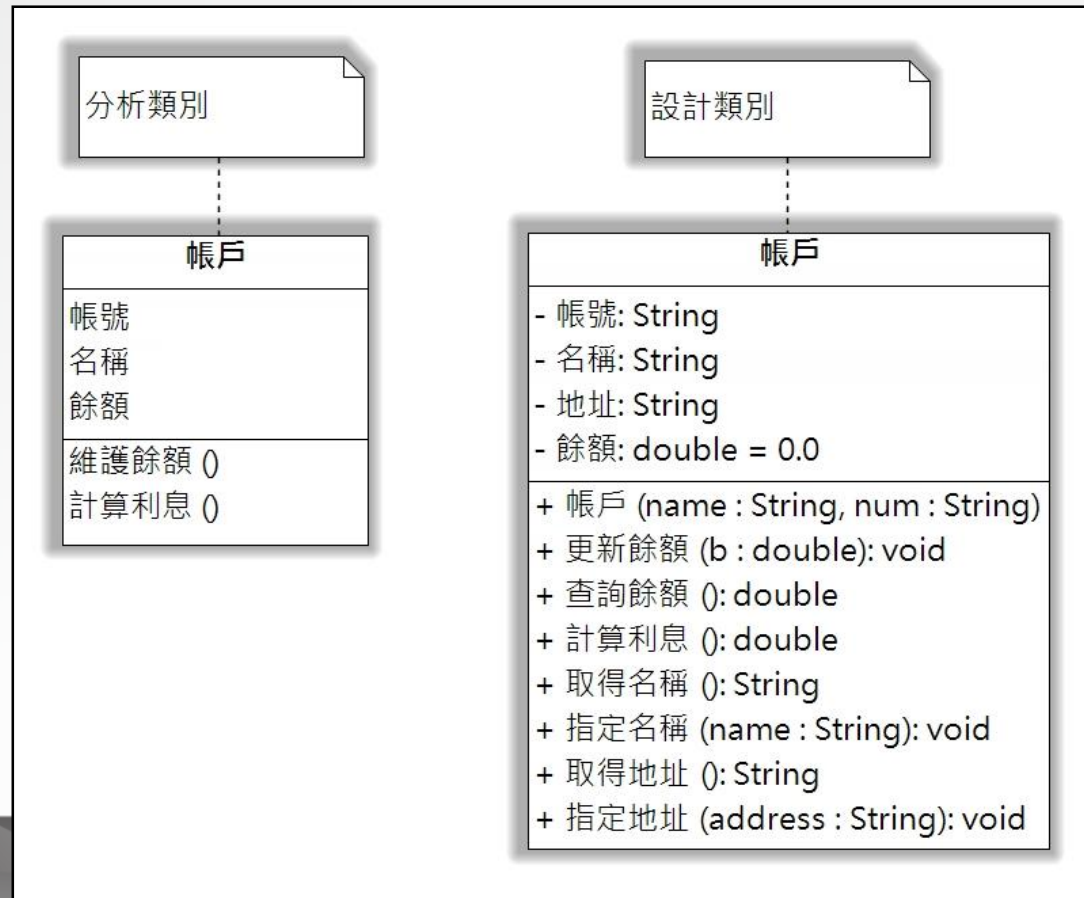
- 在物件導向設計階段需要建立的UML圖形，包含：
 - 互動圖（Interaction Diagrams）、狀態機圖（State Machine Diagrams）和套件圖（Package Diagram，詳見第15章）。
 - 設計類別圖（Design Class Diagrams）：包含類別、類別關係、屬性、方法、能見度和資料型態等。

13-1-2 分析與設計類別的差異-說明

- 分析類別只需掌握系統行為，並不用考量實作問題的細節；設計類別需要定義完整類別的詳細內容來執行其責任，包含類別屬性和操作，如下所示：
 - 類別屬性：詳細定義屬性名稱、資料型態、能見度和初始值。
 - 類別操作：詳細定義操作名稱、參數清單和傳回值型態。

13-1-2 分析與設計類別的差異-圖例

- 使用【帳戶】類別為例，說明分析與設計類別的差異，如下圖所示：



13-1-3 設計類別的特點- 完整性

- 完整性（Completeness）是指設計類別完全滿足類別使用者的需求，和其合理的期望功能，例如：購買電腦，電腦擁有CPU、主機版、記憶體、硬碟、連接線和機殼，如果缺一項，就表示它不完整。
- 如果類別可以滿足使用者合理的期望功能，就表示設計類別是一個完整性類別。

13-1-3 設計類別的特點- 充分性

- 充分性（**Sufficiency**）是指設計類別的操作剛剛好滿足類別可以實現的功能，簡單的說，每一個設計類別都是各有所長的平凡人，大家通力合作來完成特定的功能；而不是一個超人，一個人就可以完成所有的功能。
- 例如：帳戶類別可以更新和查詢餘額是十分正常的，但是，如果擁有信用管理或處理信用卡等操作，這個設計類別就不滿足充分性。

13-1-3 設計類別的特點- 單純性

- 單純性（**Primitiveness**）是指設計類別提供的操作是一種單純且不可分割的服務，換句話說，我們不應該在同一設計類別提供多種相同功能的操作。例如：當類別已經提供單一帳戶的存款操作時，就不需要提供多帳戶的批次存款操作。

13-1-3 設計類別的特點- 高內聚力

- 內聚力（**Cohesion**）是單一物件或軟體元件之中互動關係的緊密程度，互動愈多，表示它的內聚力愈高。
- 高內聚力是指一個物件提供的功能愈單純愈好，一個好的設計類別應該專注其責任，不應該作太多事和分派太多的責任，換句話說，每一個設計類別應該只需少量的操作即可。

13-1-3 設計類別的特點- 低耦合度

- 耦合度（**Coupling**）是指一個物件或軟體元件，與另一個物件或軟體元件之間連接關係的強度。
- 耦合度是用來衡量類別之間的相依關係，高耦合度表示類別程式碼會十分難更改，因為相關互動的類別太多，更改一個類別的程式碼，可能影響一大串類別。換句話說，一個良好設計類別如果更改程式碼，應該不會影響太多類別，因為它是一個低耦合度的類別。



13-2 建立互動圖

- 13-2-1 建立系統循序圖
- 13-2-2 從強韌圖建立通訊圖
- 13-2-3 將通訊圖轉換成循序圖
- 13-2-4 直接從強韌圖建立循序圖
- 13-2-5 使用互動塑模找出屬性與操作



13-2-1 建立系統循序圖-基礎

- 對於每一個使用案例來說，系統循序圖就是一個摘要版本的循序圖，它是將系統視為一個黑盒子，不用考量系統的內部，只強調動作者如何跨越系統邊界與系統進行互動過程的訊息傳遞。
- 一般來說，我們會替使用案例規格的主流程，和一些常用且複雜的替代流程建立系統循序圖。

13-2-1 建立系統循序圖-建立步驟

- 基本上，從使用案例情節建立系統循序圖的步驟，如下所示：
 - Step 1：在右邊繪出系統的生命線。
 - Step 2：對於每一個操作系統的動作者，在左邊建立動作者的生命線。
 - Step 3：針對使用案例每一個動作者產生的系統事件，建立訊息。

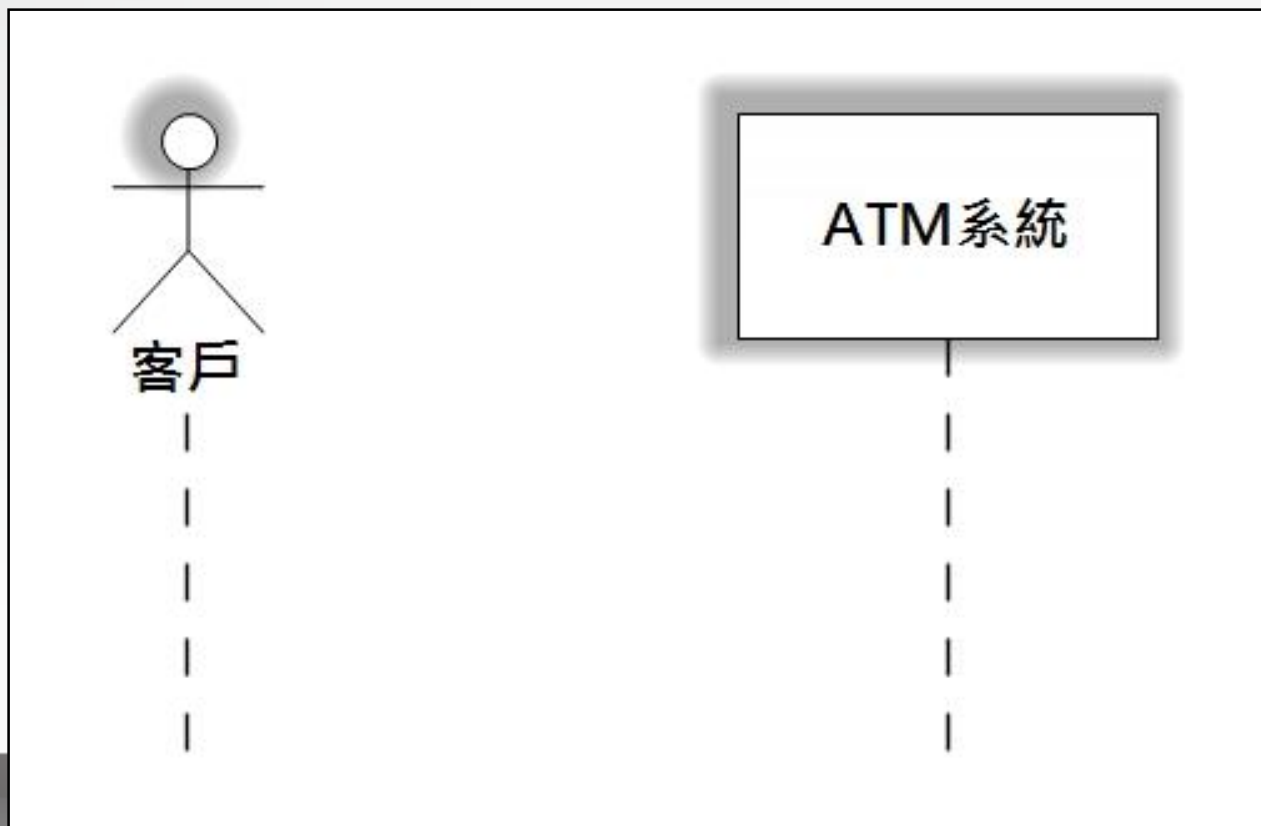
13-2-1 建立系統循序圖-案例(主流程)

■ 我們準備替ATM自動櫃員機系統的提款使用案例建立系統循序圖，其主流程的步驟，如下所示：

1. 客戶選提款選項。
2. include::取得帳戶資料。
3. 系統顯示所有可用帳戶選單，並且等待選擇。
4. 客戶選擇其中一個帳戶。
5. 系統顯示提款金額選單。
6. 客戶選擇提款金額。
- 延伸點::選擇自行輸入提款金額。
7. 系統送出款項且更新銀行的帳戶餘額。

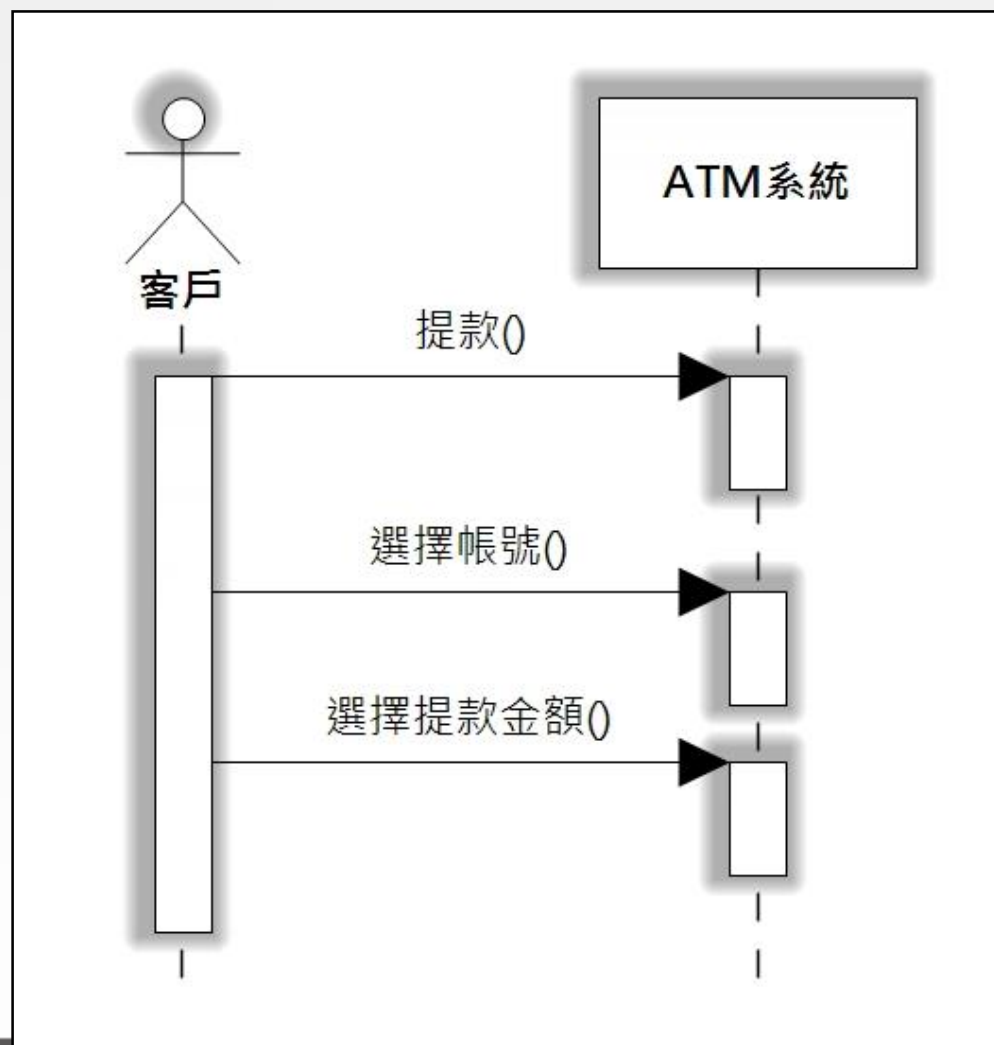
13-2-1 建立系統循序圖-案例(步驟一)

- Step 1：在右邊繪出ATM系統的生命線，左邊是客戶生命線，如下圖所示：



13-2-1 建立系統循序圖-案例(步驟二)

- Step 2：檢視主流程中屬於動作者送出的訊息，即步驟1、4和6，然後一一從上而下繪出從客戶送至系統的訊息，即可完成系統循序圖的建立，如右圖所示：

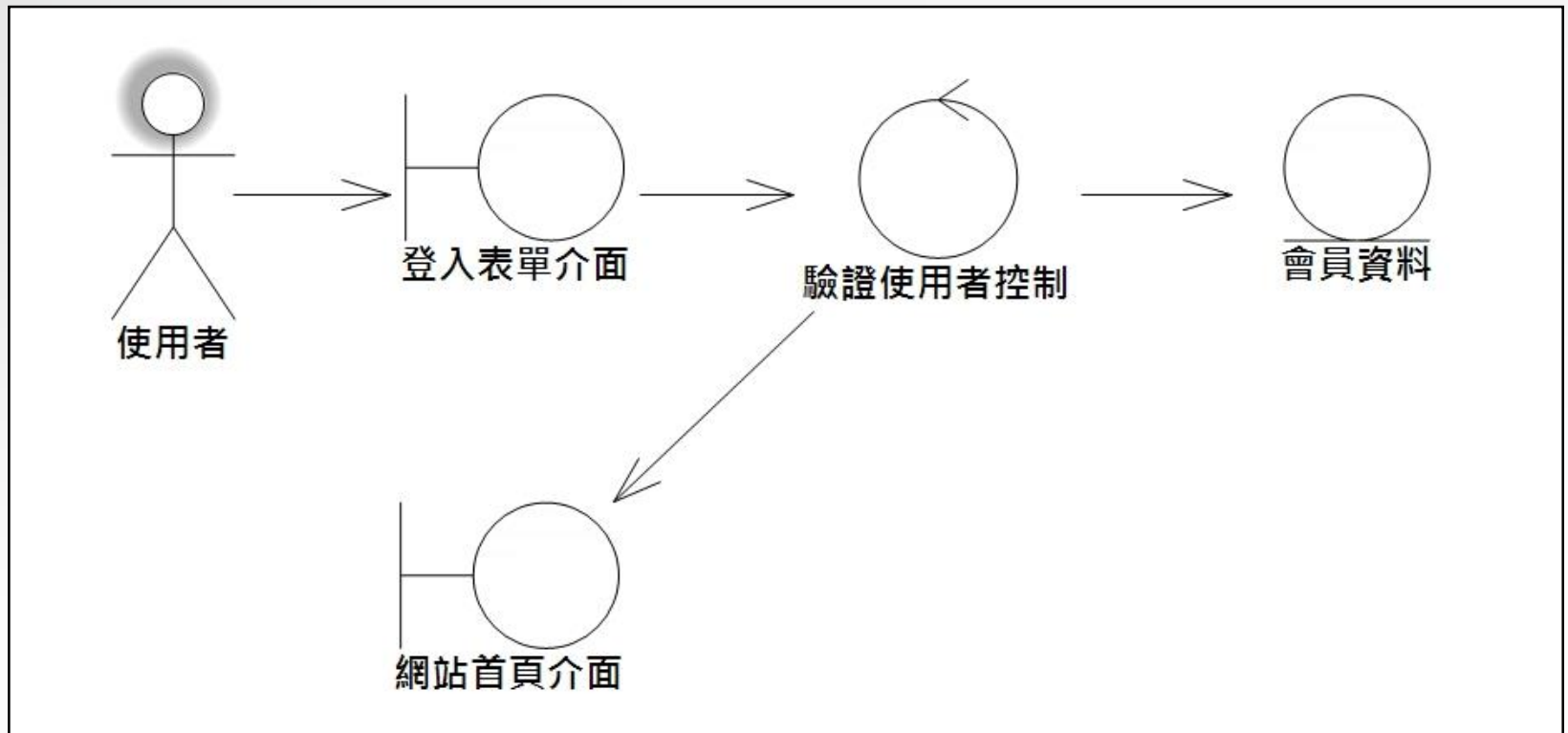


13-2-2 從強韌圖建立通訊圖-說明

- 強韌圖（Robustness Diagram）和UML通訊圖（Communication Diagram）的觀念十分相似，我們可以將強韌圖視為是一種簡化版本的通訊圖，只需在物件之間加上訊息，就可以從強韌圖建立成通訊圖。

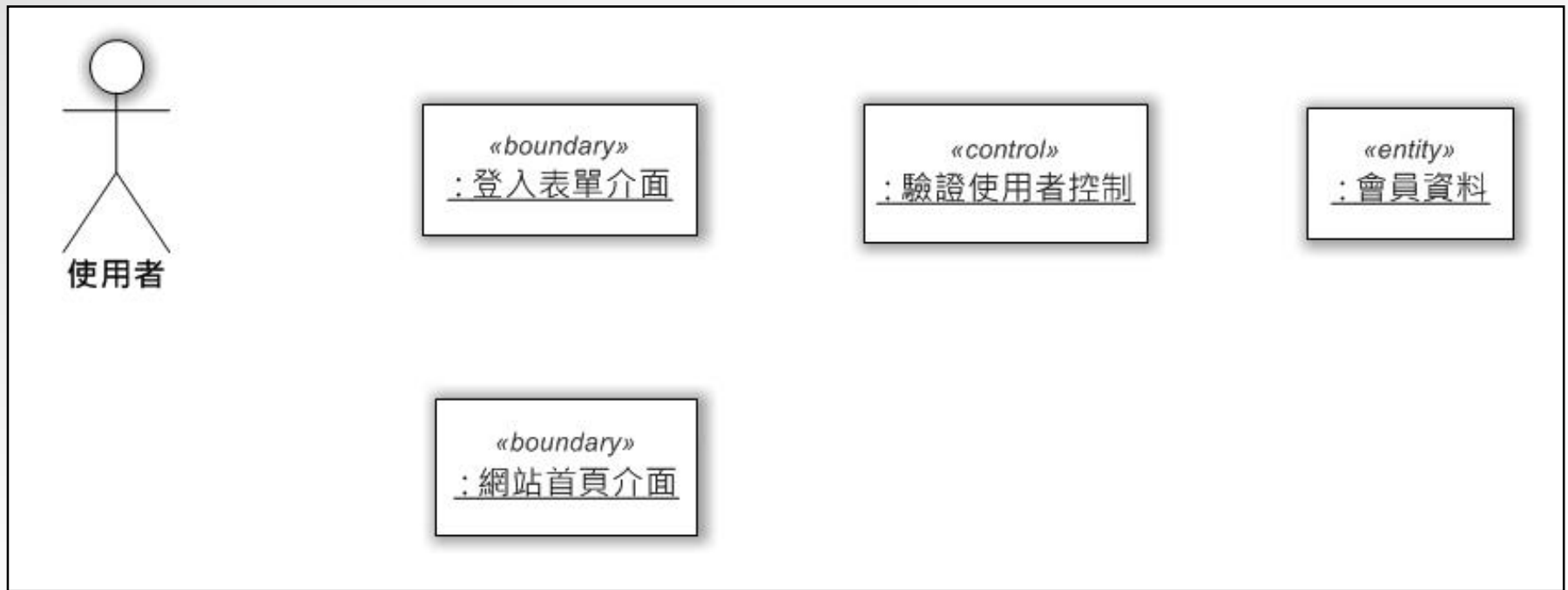
13-2-2 從強韌圖建立通訊圖-強韌圖

- 例如：第10-5-3節登入使用案例的強韌圖，如下圖所示：



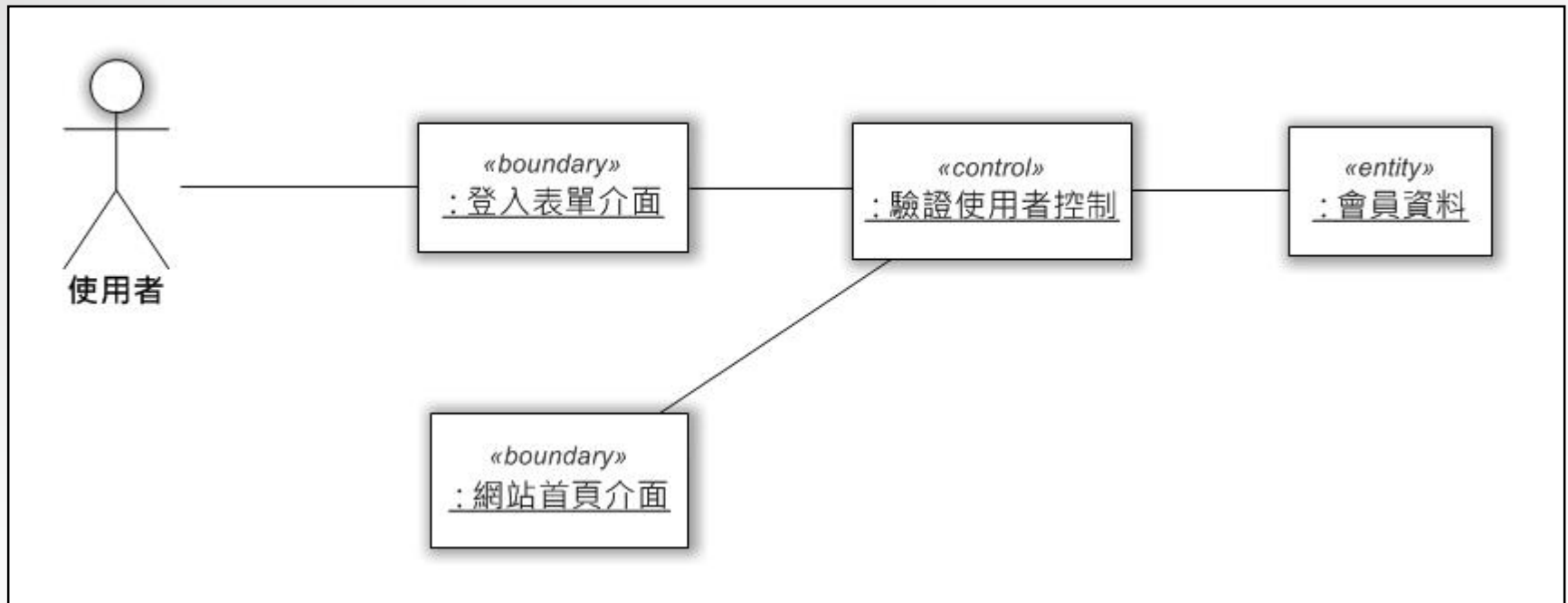
13-2-2 從強韌圖建立通訊圖-步驟一

- Step 1：首先參考強韌圖一一建立動作者和各物件的參與者，分別標示為邊界、控制和實體物件，如下圖所示：



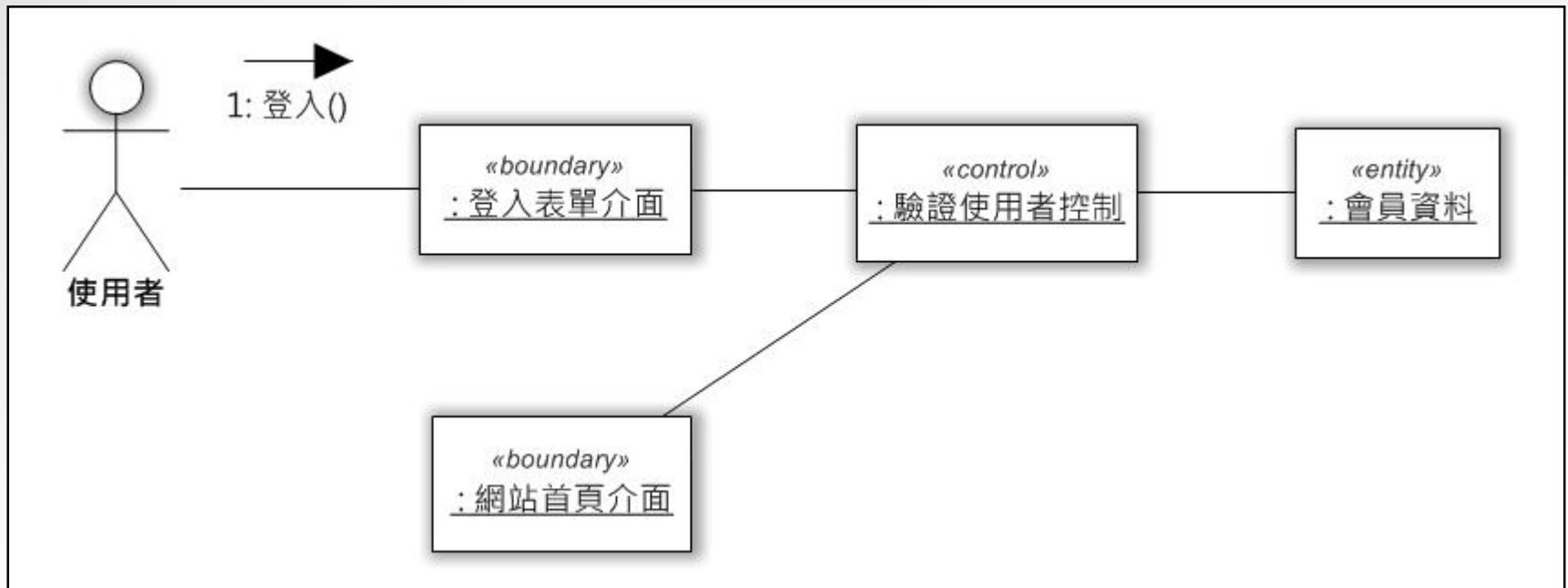
13-2-2 從強韌圖建立通訊圖-步驟二

- Step 2：然後參考強韌圖一一建立各物件之間的連接，如下圖所示：



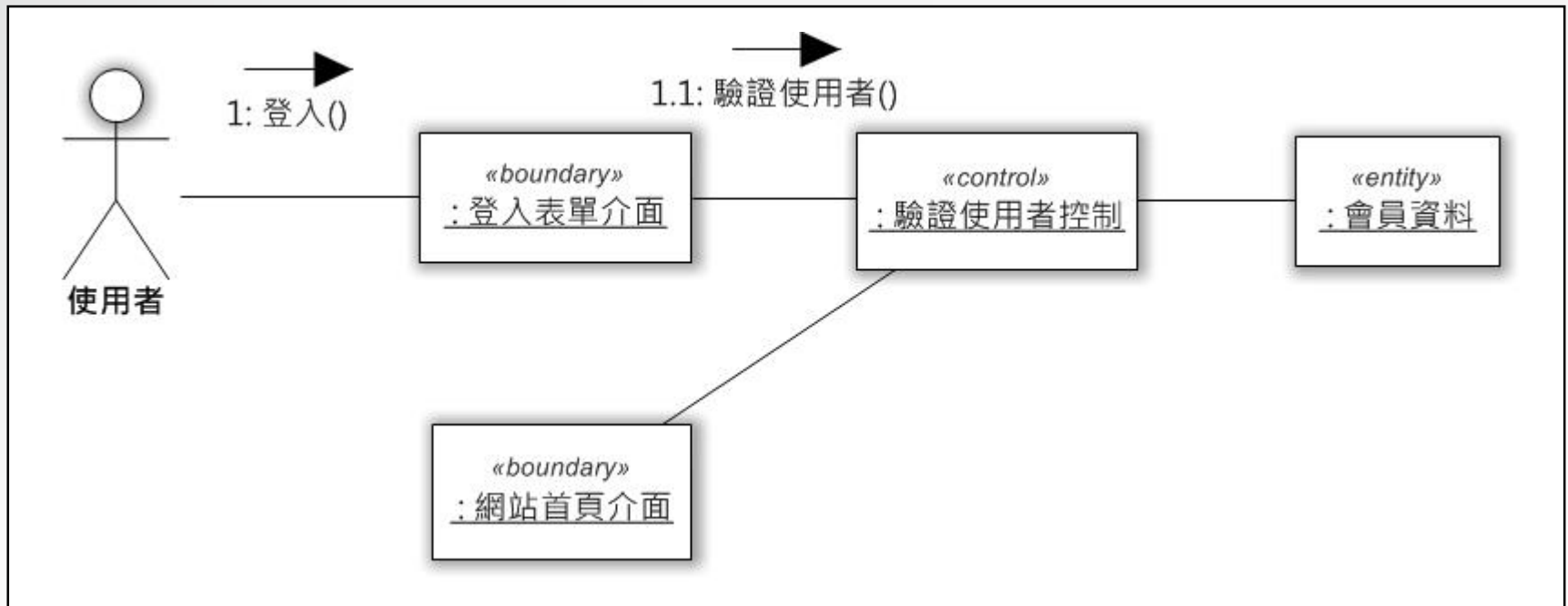
13-2-2 從強韌圖建立通訊圖-步驟三

- Step 3：接著建立訊息，首先從動作者送出第一層訊息來執行登入，如下圖所示：



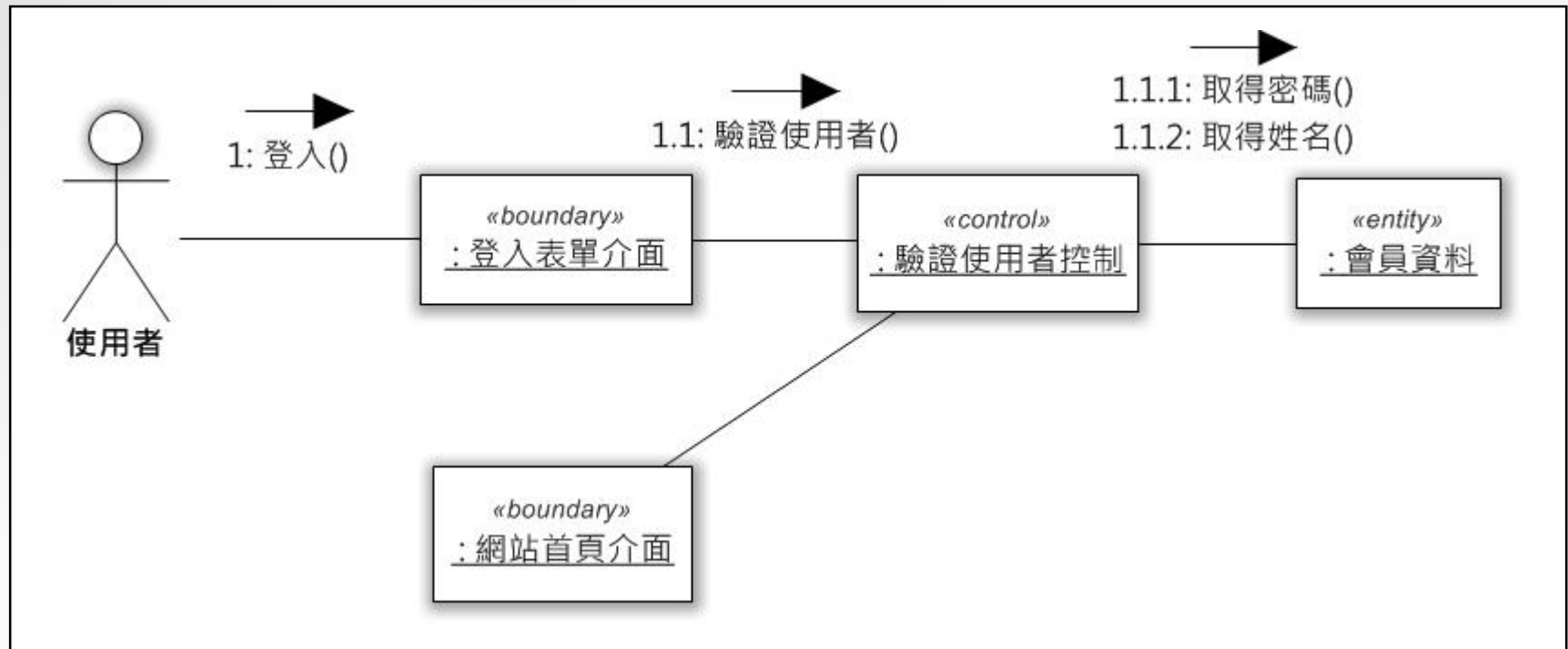
13-2-2 從強韌圖建立通訊圖-步驟四

- Step 4：登入表單介面物件送出第二層驗證使用者訊息至控制物件，如下圖所示：



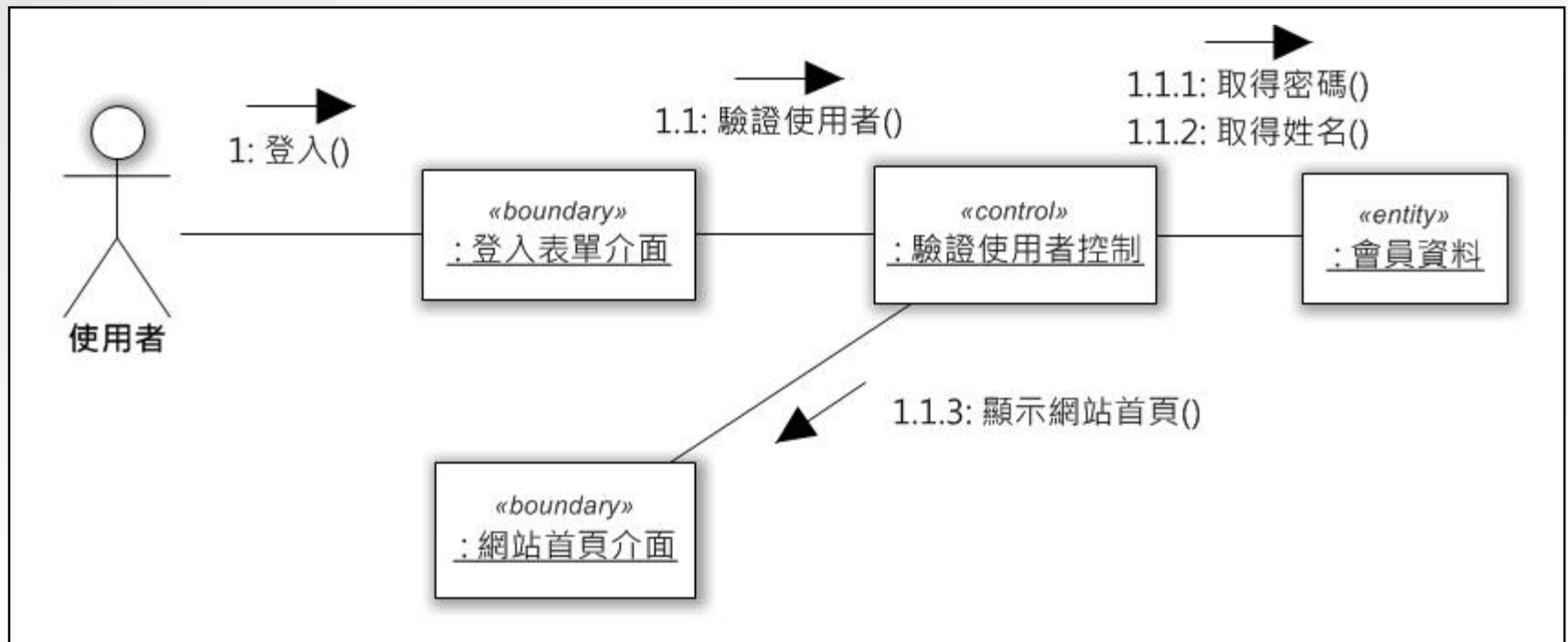
13-2-2 從強韌圖建立通訊圖-步驟五

- Step 5：控制物件送出第三層訊息取得使用者的密碼和姓名至實體物件，傳回值是姓名與密碼，如下圖所示：



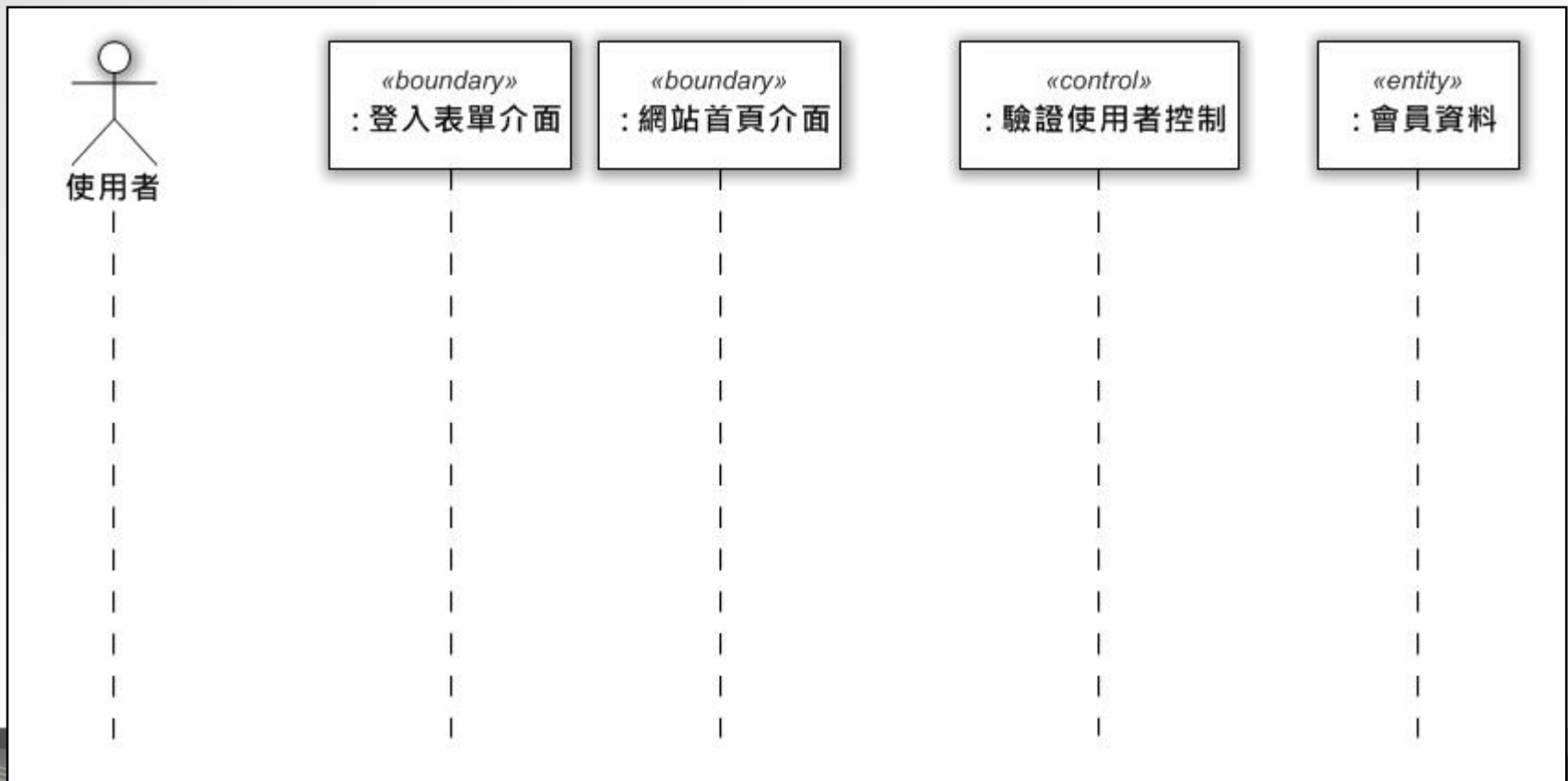
13-2-2 從強韌圖建立通訊圖-步驟六

- Step 6：傳送第三層訊息給網站首頁介面物件，訊息參數是會員姓名，顯示歡迎會員進入網站的訊息文字，如下圖所示：



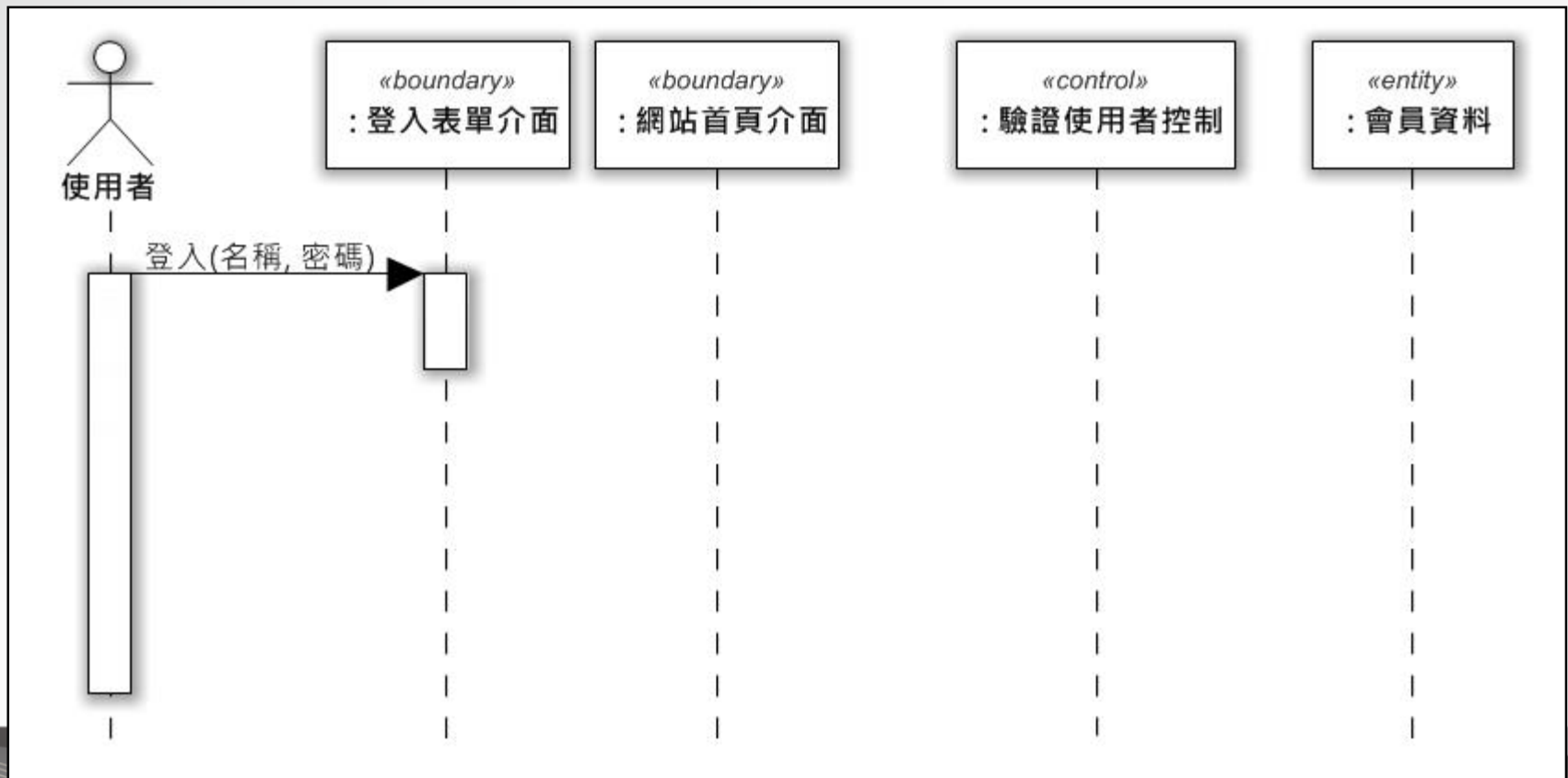
13-2-3 將通訊圖轉換成循序圖-步驟一

- Step 1：請依序將通訊圖的動作者和所有分析物件都從左至右排列成群組，如下圖所示：



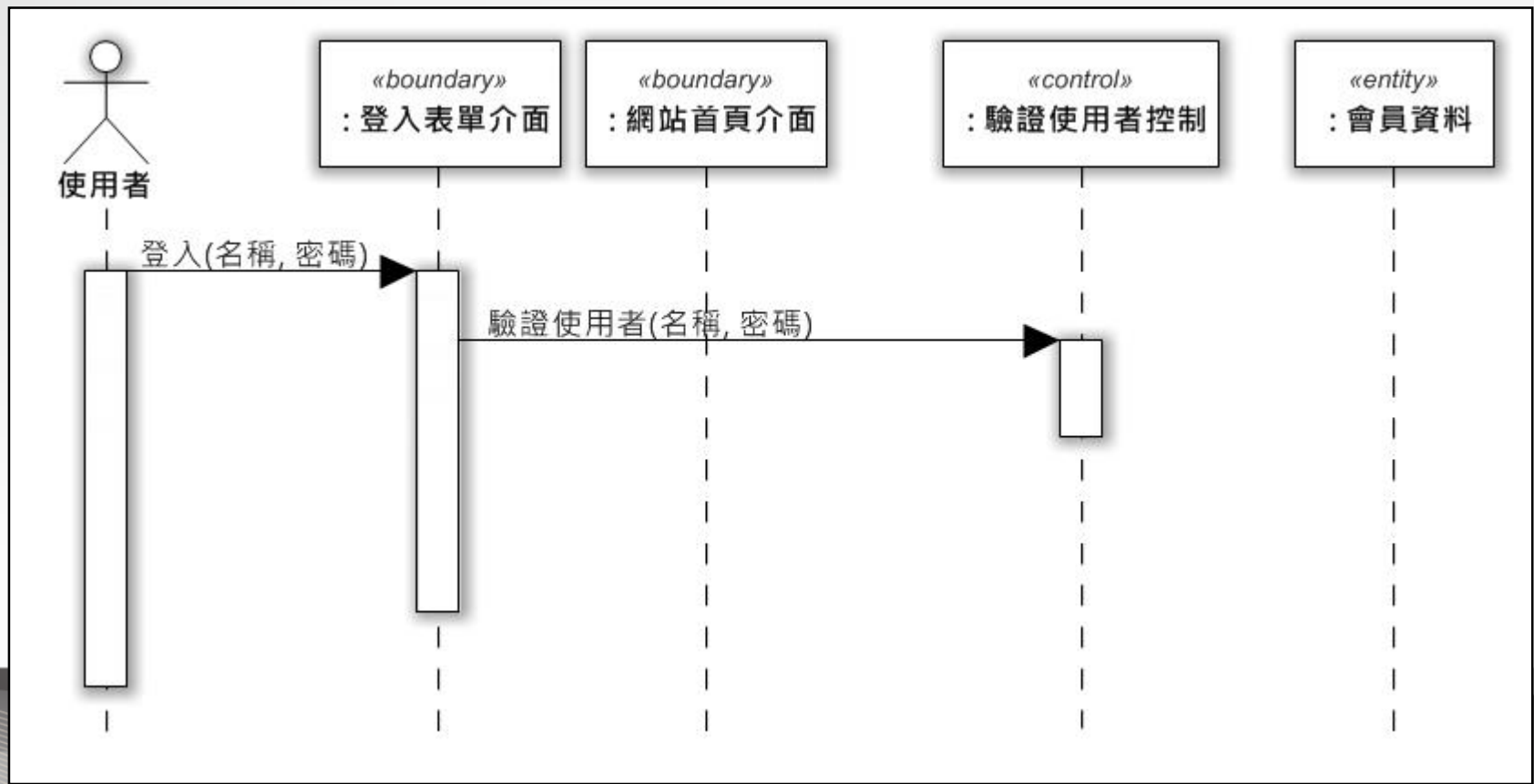
13-2-3 將通訊圖轉換成循序圖-步驟二

- Step 2：繪出第一層訊息，從動作者傳送登至登入表單介面物件的登入訊息，如下圖所示：



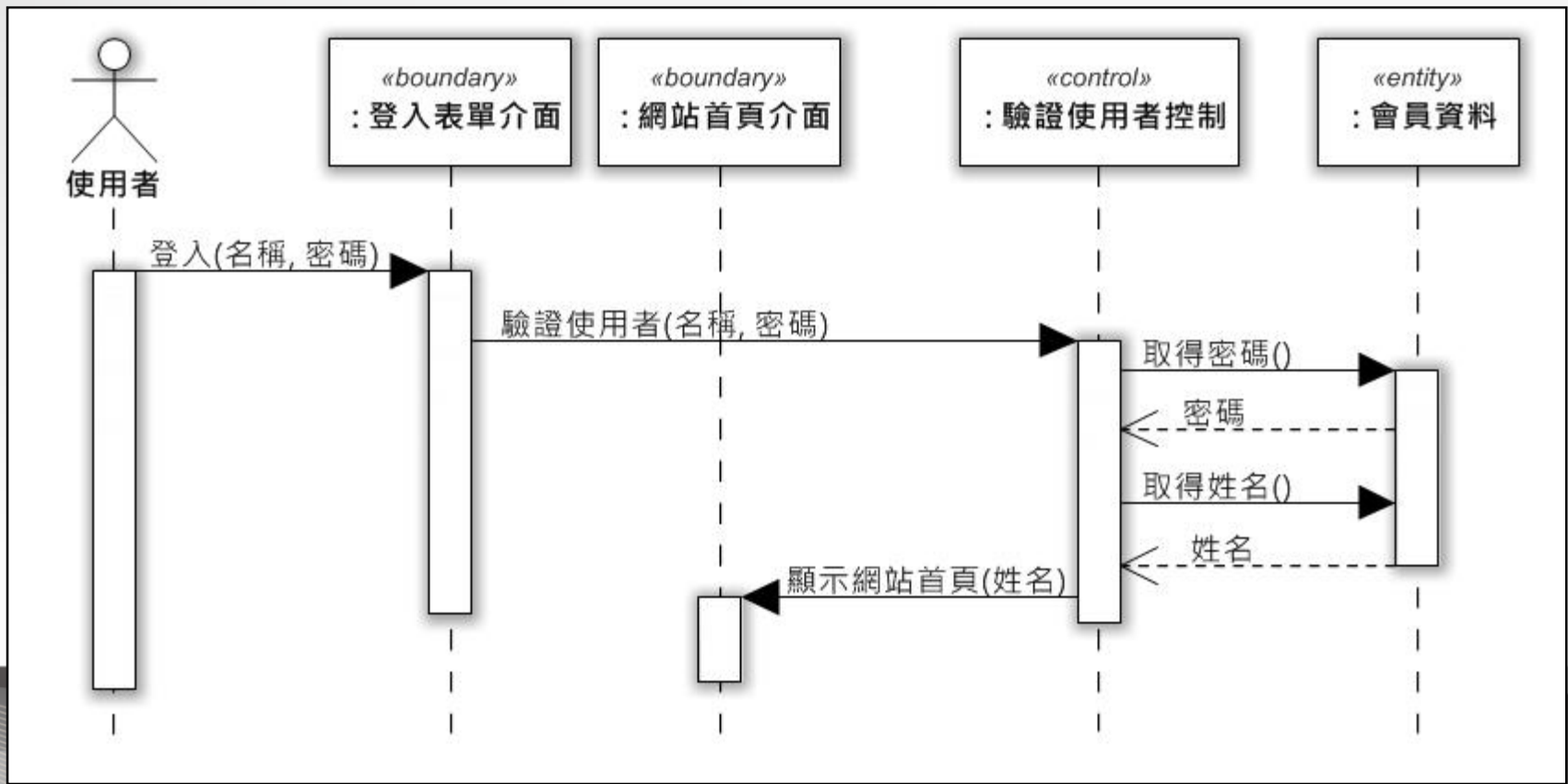
13-2-3 將通訊圖轉換成循序圖-步驟三

- Step 3：繪出第二層訊息，在登入表單介面物件需要再送出第二層的驗證使用者訊息至控制物件，參數也是名稱與密碼，如下圖所示：



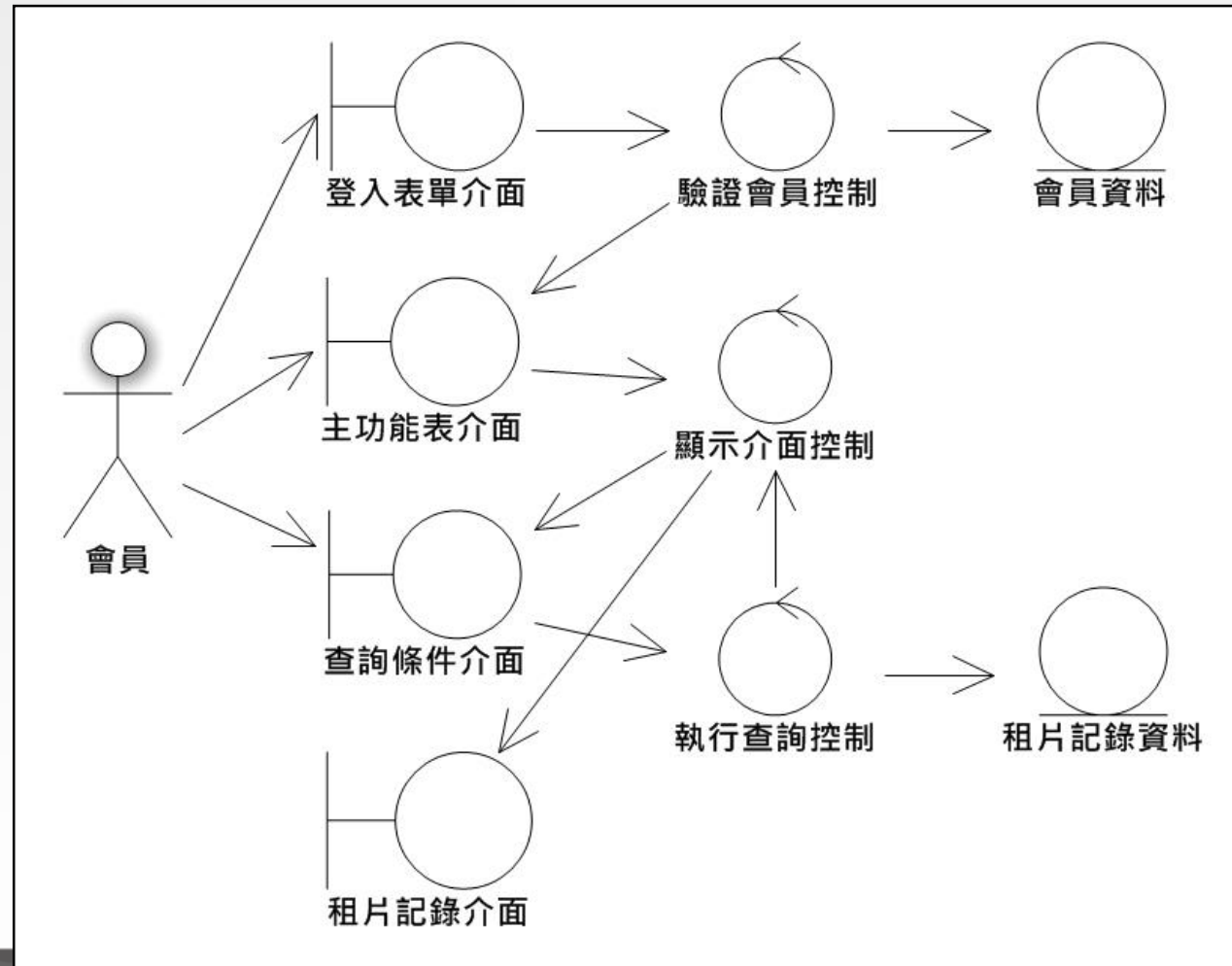
13-2-3 將通訊圖轉換成循序圖-步驟四

- Step 4：繪出第三層訊息，控制物件送出第三層訊息取得密碼和姓名後，通過驗證，即可再送出第三層訊息顯示網站首頁，完成循序圖的建立，如下圖所示：



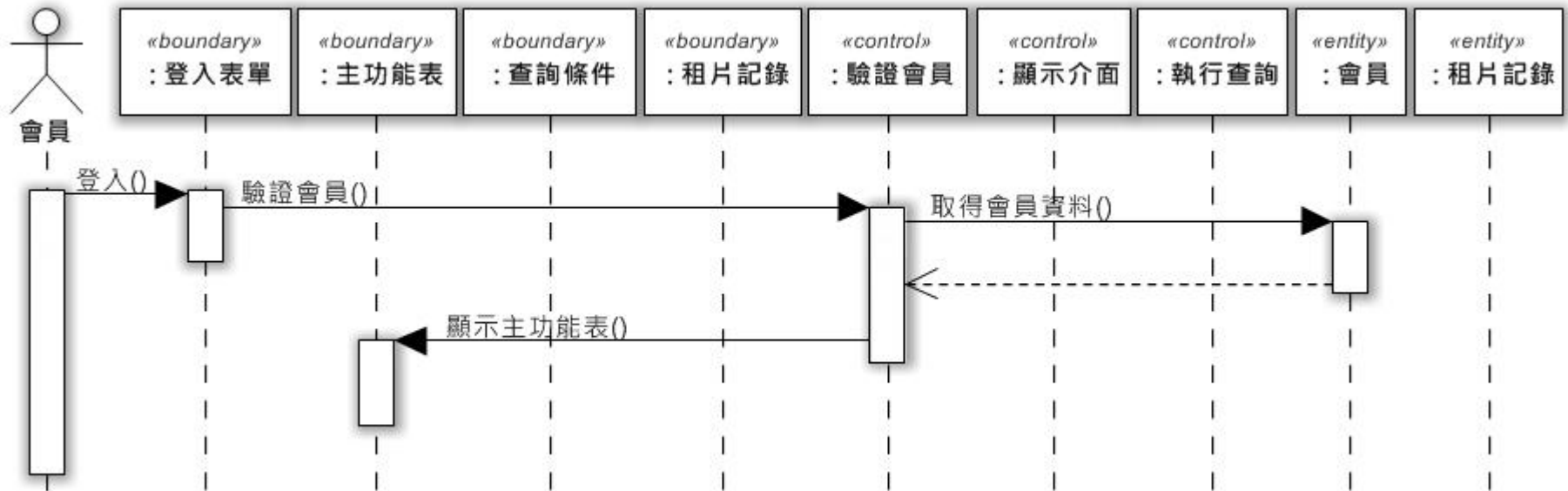
13-2-4 直接從強韌圖建立循序圖-強韌圖

■ 將第10-5-3節查詢租片記錄使用案例的強韌圖建立成循序圖，其強韌圖如右圖所示：



13-2-4 直接從強韌圖建立循序圖-步驟一

- Step 1：請依序將強韌圖的動作者和所有分析物件都從左至右排列成群組，如下圖所示：

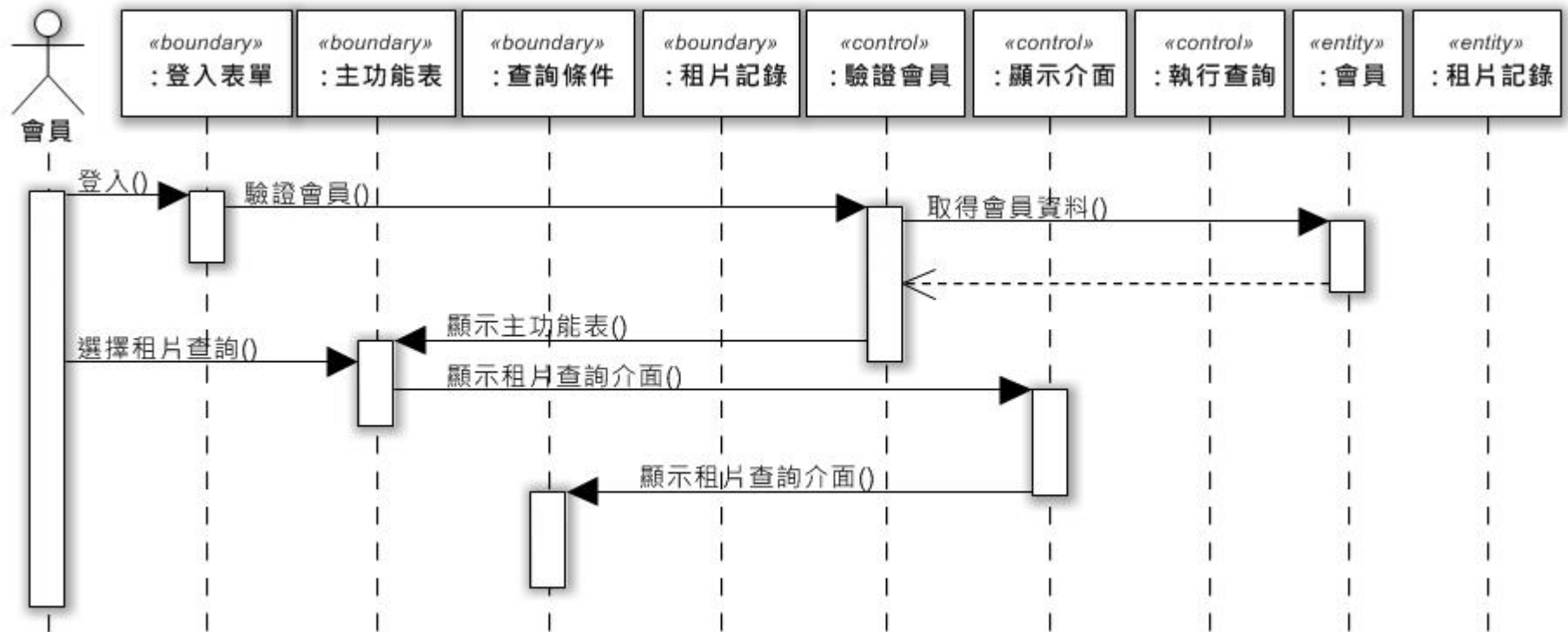


13-2-4 直接從強韌圖建立循序圖-步驟二

- Step 2：參考第10-5-3節查詢租片記錄的主流程和活動圖，從動作者送出的第1個登入訊息開始，依據活動圖的動作轉換順序一一繪出訊息，如上述圖例。

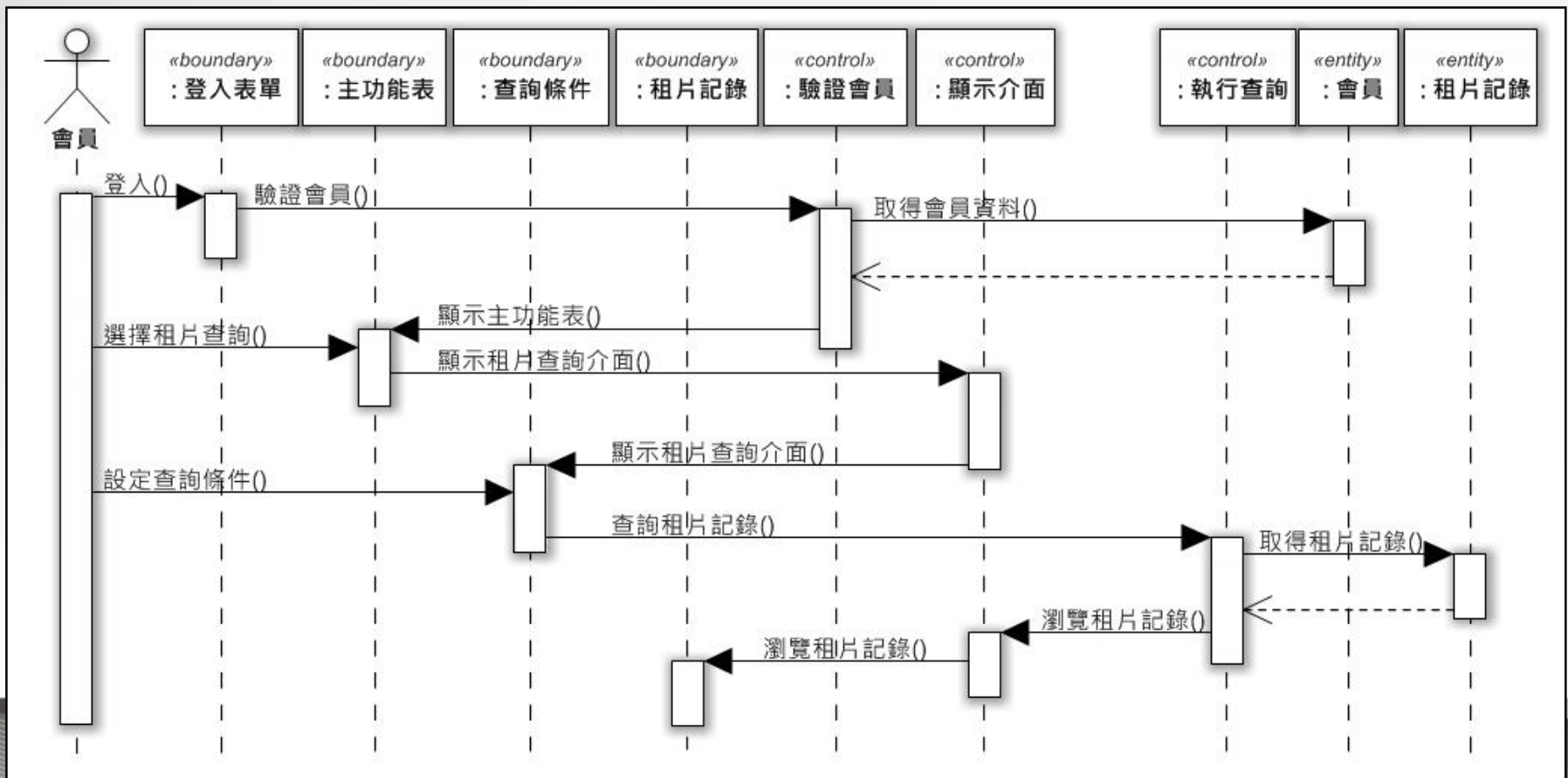
13-2-4 直接從強韌圖建立循序圖-步驟三

- Step 3：從動作者送出的第2個選擇租片查詢訊息開始，依據活動圖的動作轉換順序繪出傳送的訊息，如下圖所示：



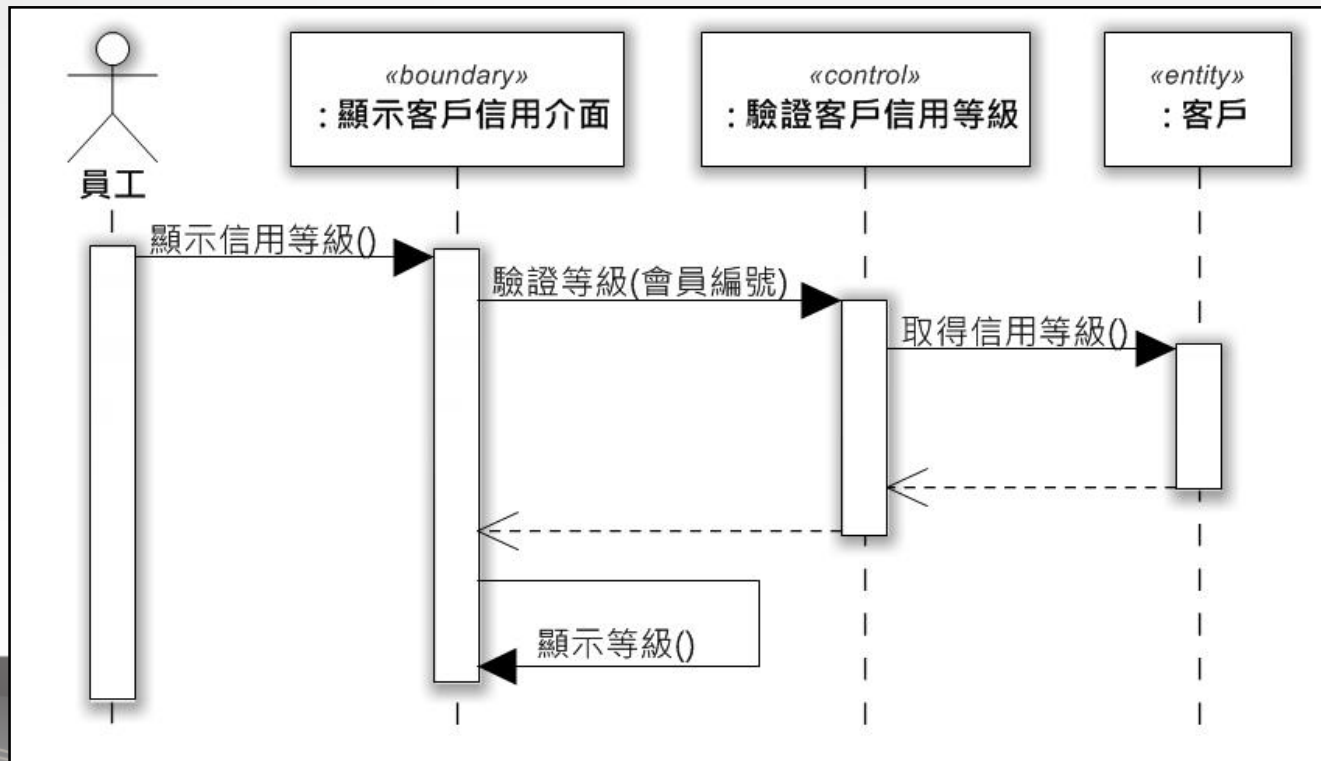
13-2-4 直接從強韌圖建立循序圖-步驟四

- Step 4：從動作者送出的第3個設定查詢條件訊息開始，依據活動圖的動作轉換順序繪出傳送的訊息，如下圖所示：



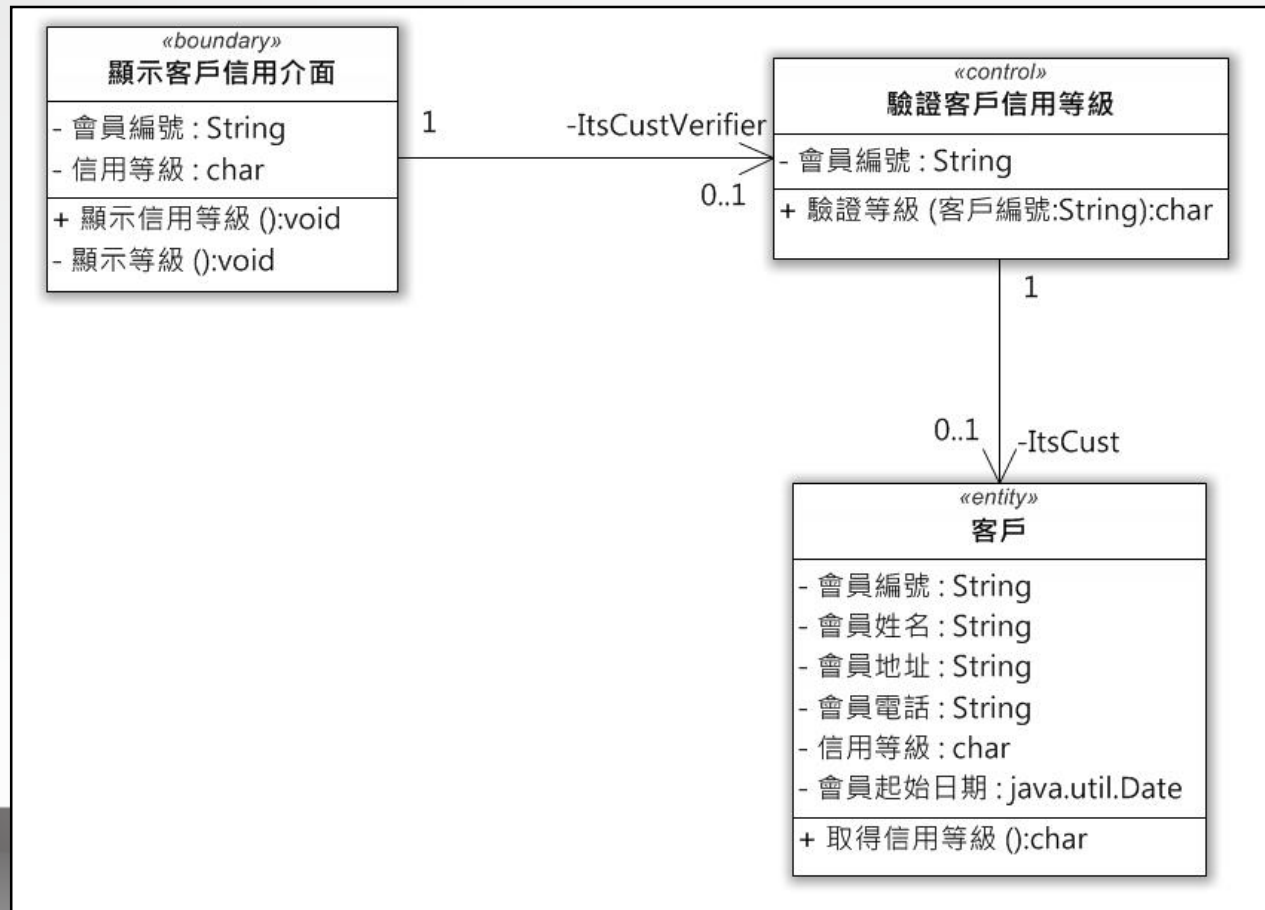
13-2-5 使用互動塑模找出屬性與操作- 循序圖

- 在建立分析模型後，就可以使用互動塑模找出屬性與操作，從建立的循序圖或通訊圖找出分析類別的屬性與操作。例如：在租片管理系統的【顯示客戶信用等級】使用案例，其循序圖如下圖所示：



13-2-5 使用互動塑模找出屬性與操作-設計類別

- 從上述循序圖可以找出分析類別的屬性與操作來建立設計類別，如下圖所示：



13-2-5 使用互動塑模找出屬性與操作- 基本規則

■ 從互動圖找出屬性和操作的基本規則，如下所示：

	類別屬性	類別操作
邊界類別	從第10-5-1節使用案例主流程活動圖的輸入或輸出註解文字，可以找出邊界類別的屬性	在循序圖中傳遞至邊界類別的訊息，就是邊界類別的操作，通常都是一些顯示介面的操作
控制類別	從循序圖傳遞訊息的參數可以找到控制類別的屬性	在循序圖中傳遞至控制類別的訊息，就是控制類別的操作
實體類別	從第10-5-1節活動圖的註解文字、使用案例規格和與實體連接的控制類別中，可以找出實體類別的屬性	在循序圖中傳遞至實體類別的訊息，就是實體類別的操作，通常都是一些資料存取的操作



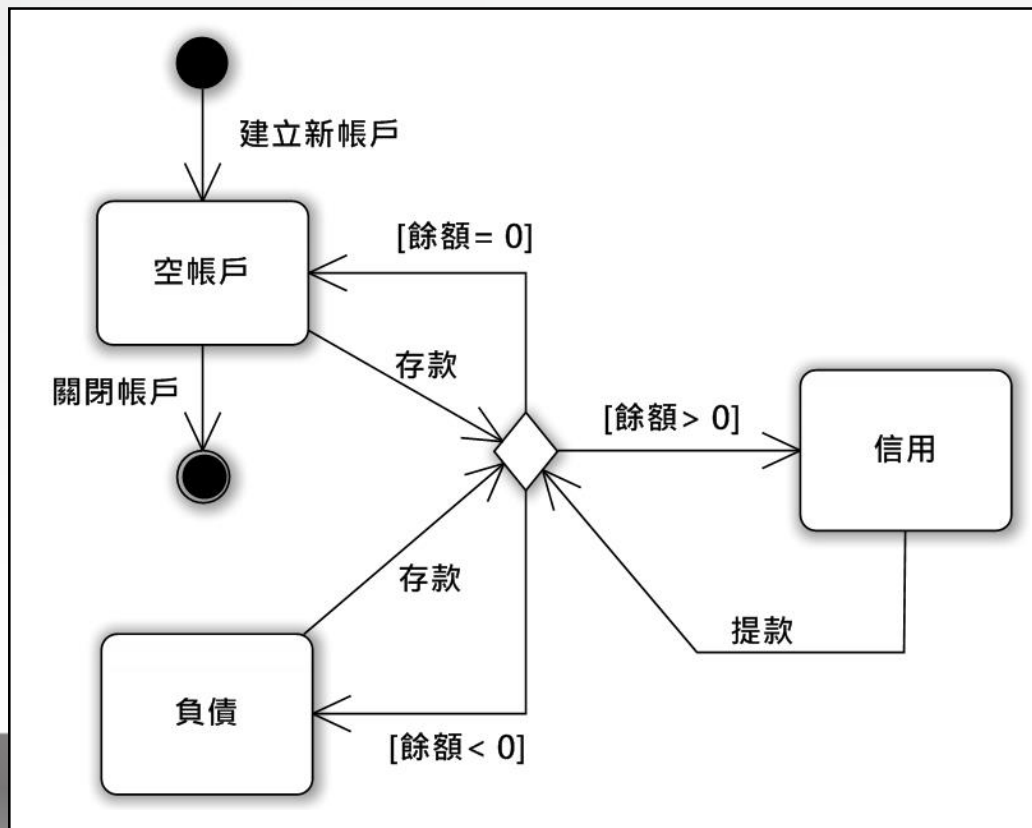
13-3 建立狀態機圖-說明

- 狀態機圖適合描述單一類別且跨越多個使用案例的行為變換過程，在實務上，狀態機圖會搭配其他互動圖來描述類別行為，我們並不會替每一類別都建立狀態機圖，只會針對哪些需要描述物件整個生命周期的行為，才需要建立狀態機圖來幫助我們了解到底是怎麼一回事。
- 一般來說，在模型中的每一物件都一定擁有狀態（**States**），狀態機圖可以表示在物件生命周期中，不同時間點的狀態變更。如果物件是狀態控制物件（**State Controlled Object**），表示此物件在接收訊息後，就會因不同的狀態而產生不同的行為，換句話說，我們需要針對模型中的狀態控制物件建立狀態機圖。

13-3 建立狀態機圖-

案例：銀行帳戶物件的狀態說明

- 銀行帳戶的餘額會因存款和提款事件而改變其值，依帳戶餘額的金額大小，帳戶物件擁有：空帳戶、信用和負債三種狀態，其狀態機圖如下圖所示：





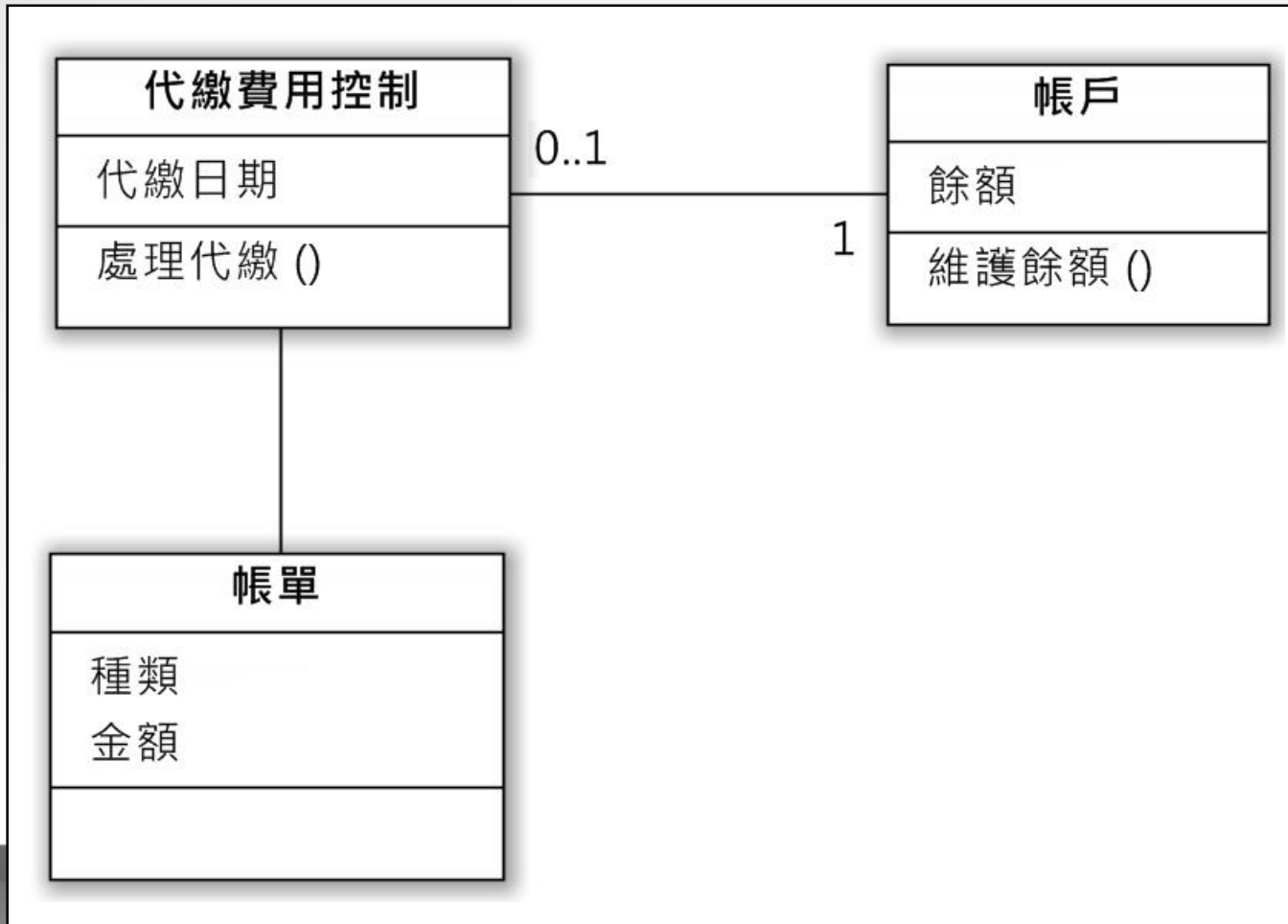
13-4 設計模型的類別圖

- 13-4-1 建立設計模型類別圖的基本步驟
- 13-4-2 在設計模型加入集合類別



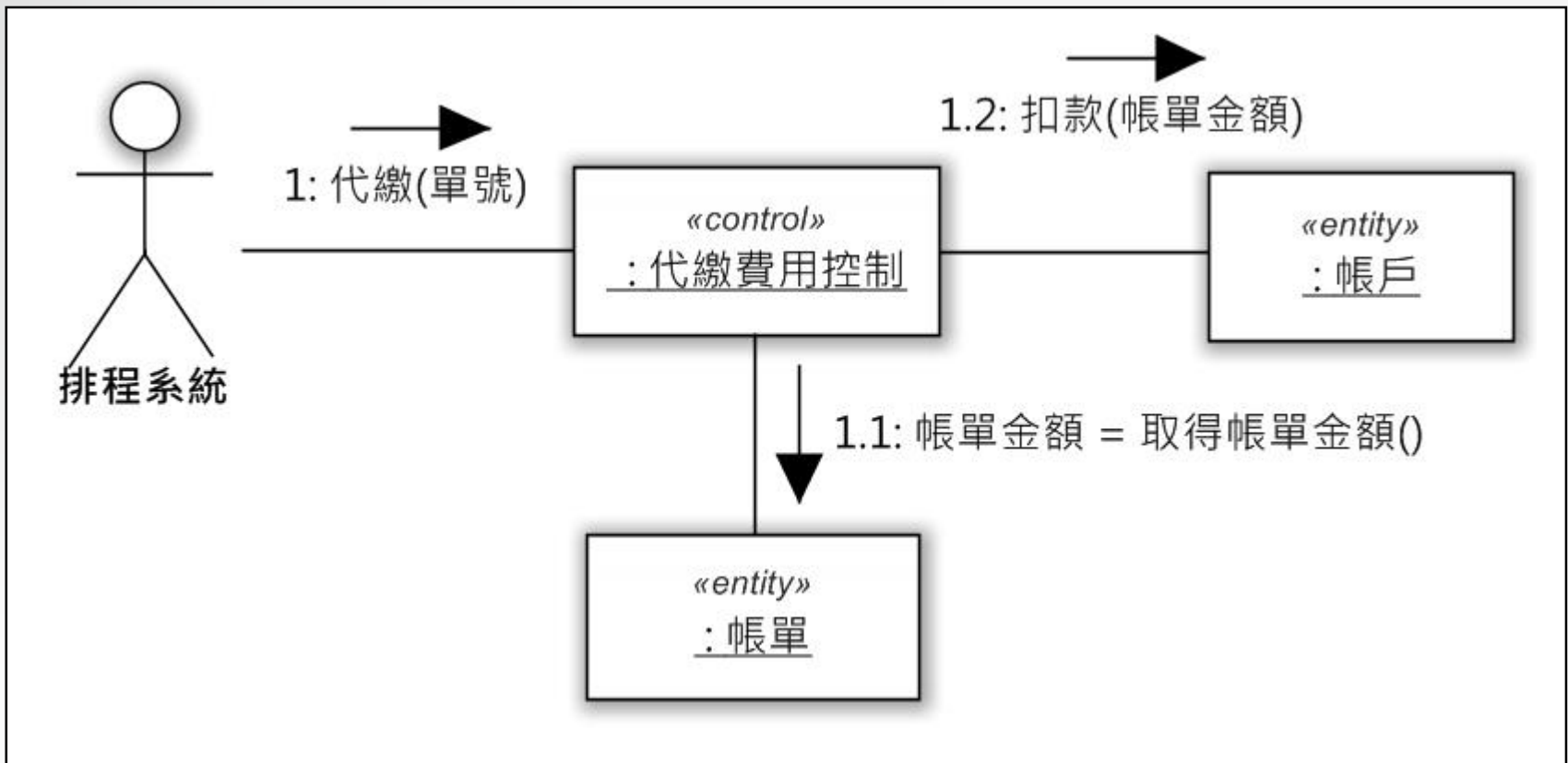
13-4-1 建立設計模型類別圖的基本步驟- 範例(分析模型)

■ 例如：銀行代繳帳單的分析模型片斷，如下圖所示：



13-4-1 建立設計模型類別圖的基本步驟- 範例(通訊圖)

■ 依據上述分析模型建立的通訊圖，如下圖所示：



13-4-1 建立設計模型類別圖的基本步驟-

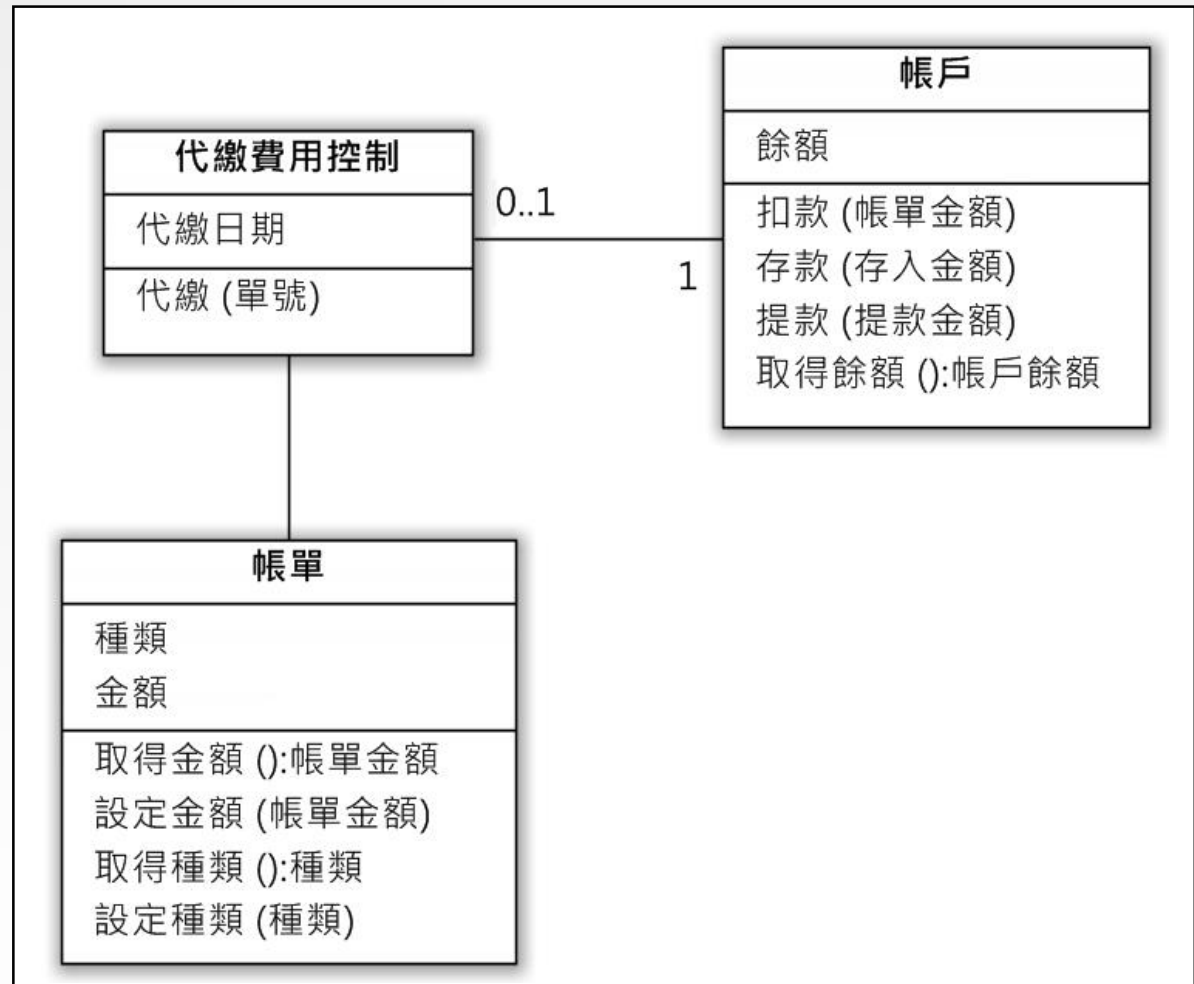
步驟一：加入類別操作(找出訊息)

- 類別操作可以從通訊圖中傳遞的訊息找出，在上述通訊圖的三個類別所接收的訊息，可以分別轉換成類別操作，其說明如下表所示：

類別	責任	訊息
代繳費用控制	處理代繳	代繳(單號)
帳單	N/A	取得金額():帳單金額
帳戶	維護餘額	扣款(帳單金額)

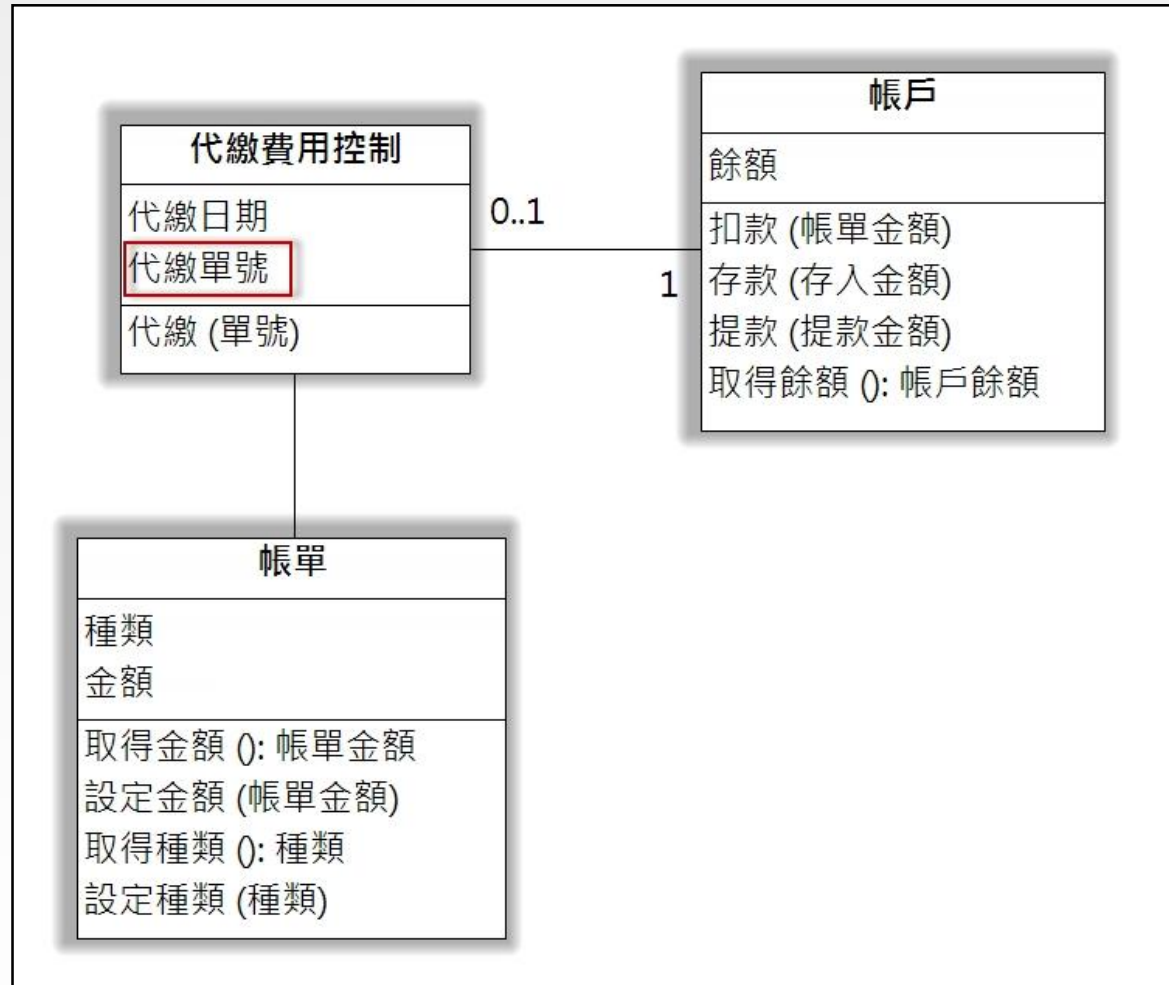
13-4-1 建立設計模型類別圖的基本步驟- 步驟一：加入類別操作(更新類別圖)

- 依據上表的訊息說明，我們可以將分析類別的責任改為類別操作，和在類別加入從訊息找到的操作。



13-4-1 建立設計模型類別圖的基本步驟- 步驟二：加入類別屬性

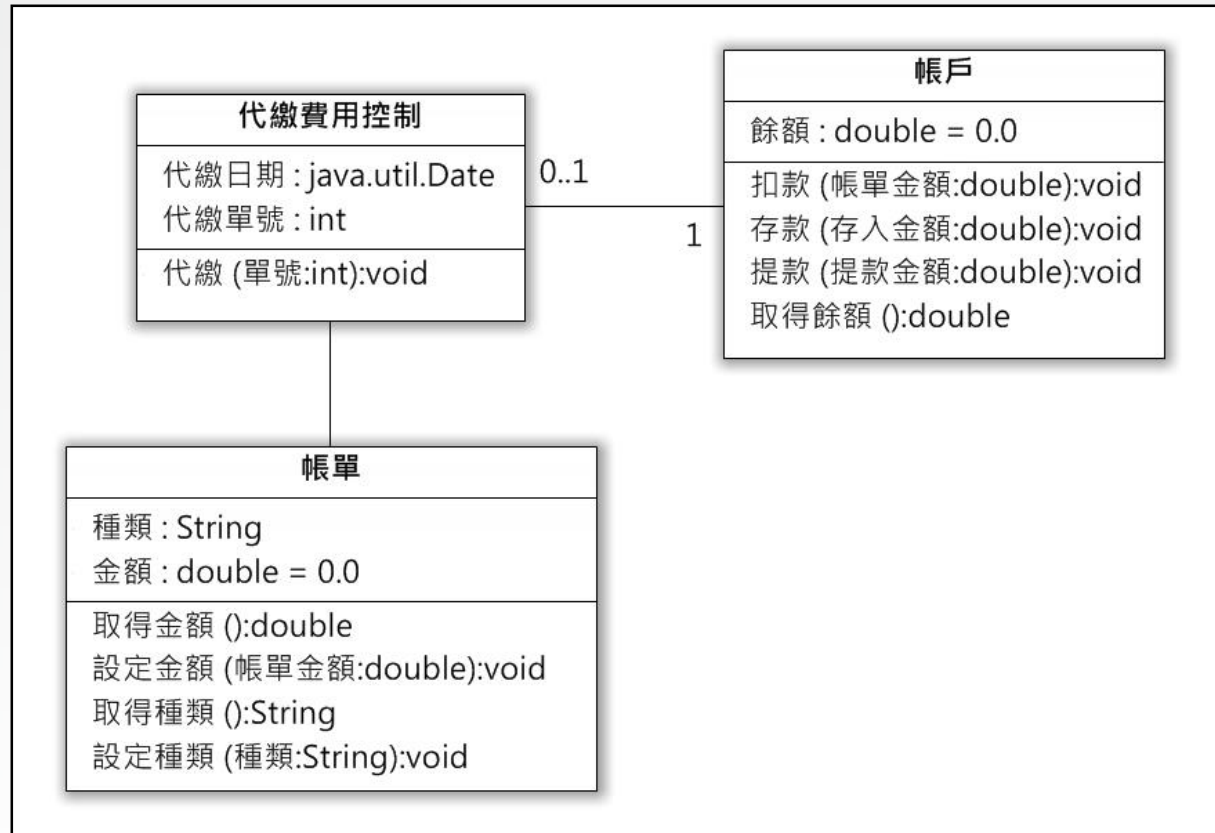
- 對於通訊圖中傳遞的訊息參數，我們需要考量是否需要加入類別，例如：【代繳費用控制】類別接收的訊息有【單號】參數，我們應該在此類別新增【代繳單號】屬性，如右圖所示：



13-4-1 建立設計模型類別圖的基本步驟-

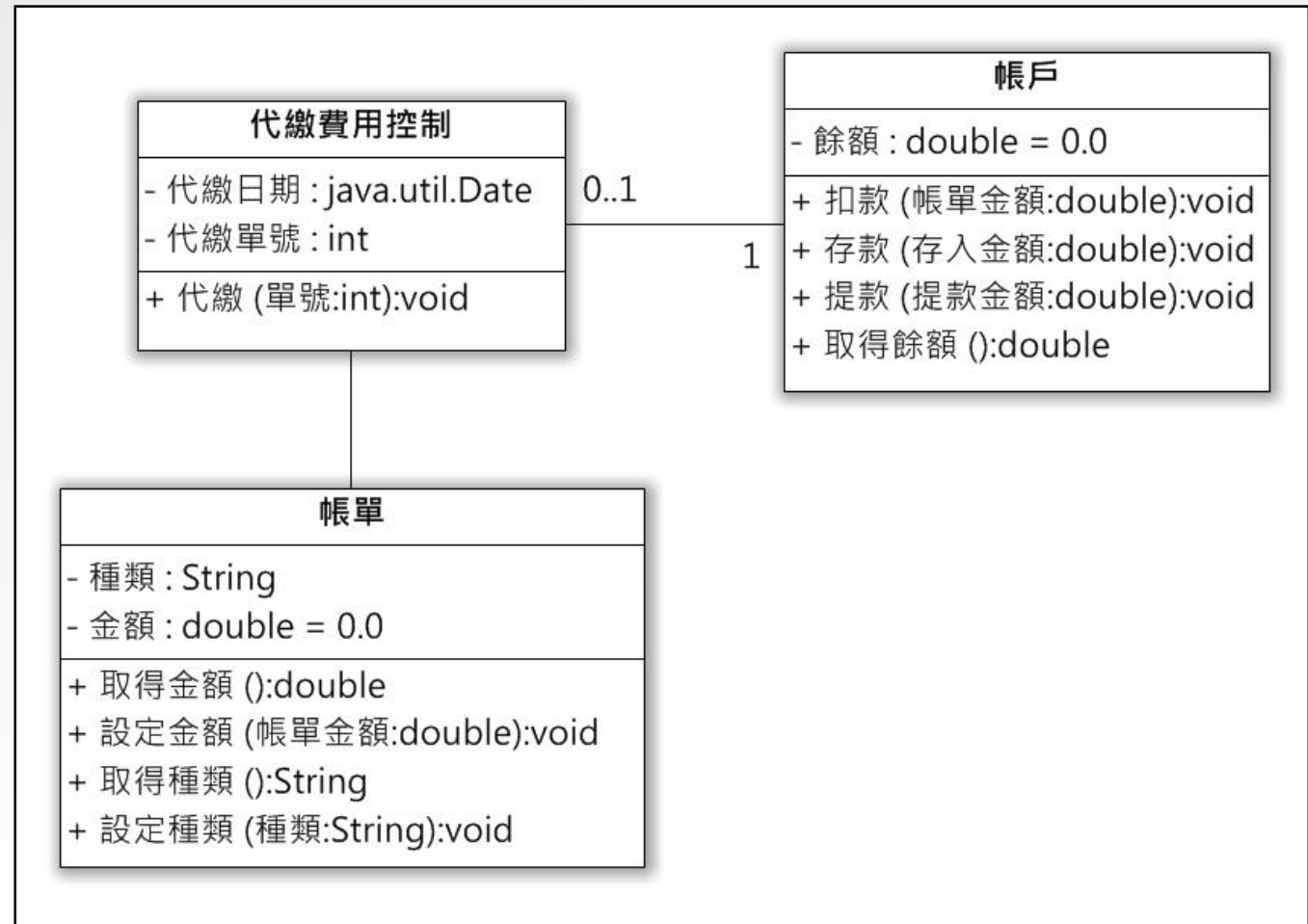
步驟三：決定資料型態

- 對於類別的屬性，我們需要進一步決定其資料型態，如果屬性有初值，也應該一併指定。例如：
【帳戶】類別的餘額屬性為double型態，其初值為0.0。



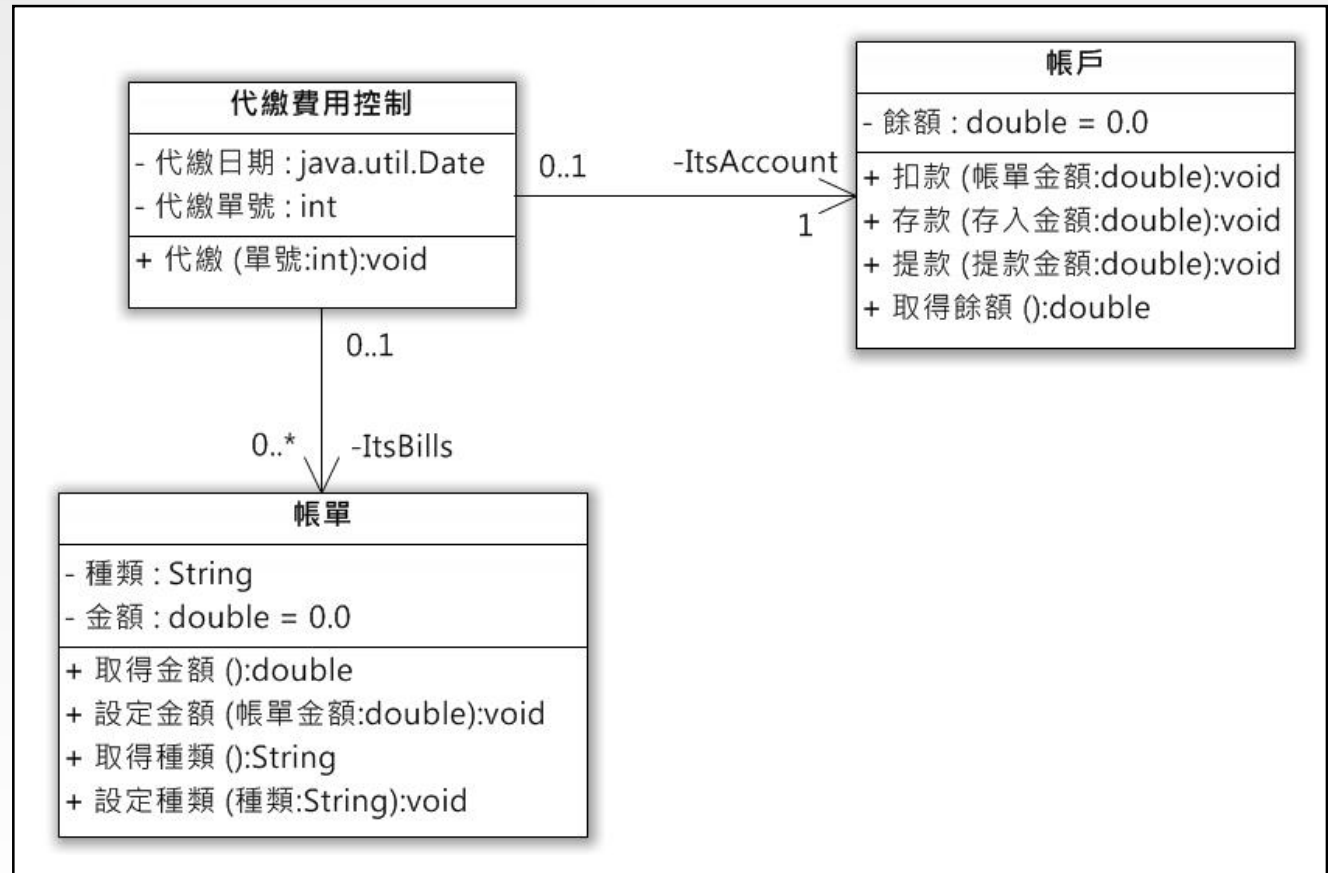
13-4-1 建立設計模型類別圖的基本步驟- 步驟四：決定能見度

- 在類別的屬性和操作都需要決定其能見度（**Visibility**），如右圖所示：



13-4-1 建立設計模型類別圖的基本步驟- 步驟五：加入類別關係

- 類別關係
包含：結合關係、
聚合關係、
組合關係和一般
關係。

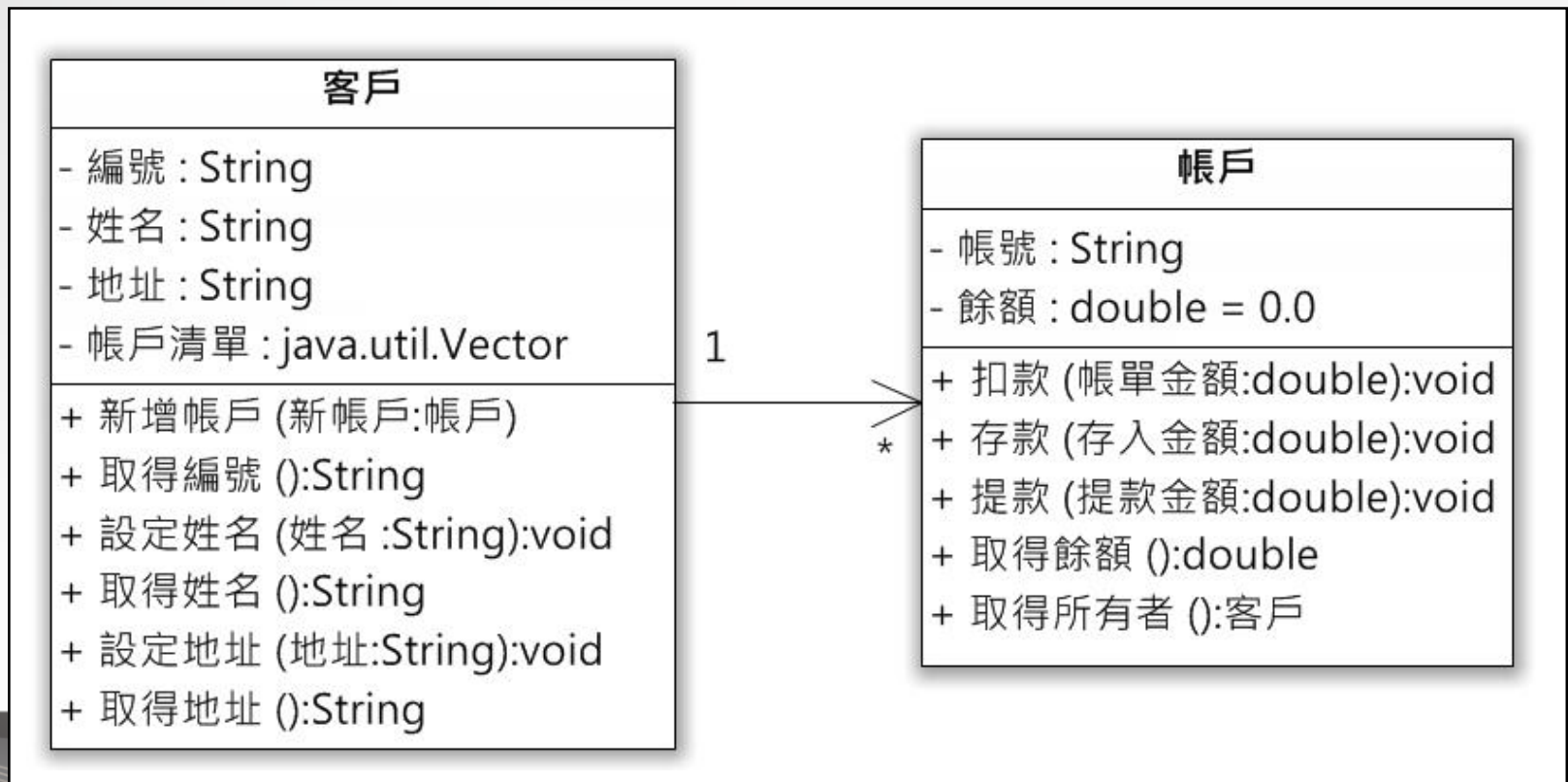


13-4-2 在設計模型加入集合類別- 集合物件的基礎

- 集合物件（**Collections**）可以視為是一個容器，其儲存的內容是物件，能夠儲存不定數量的物件，並且提供相關方法來新增、刪除和取出儲存的物件元素。
- 事實上，集合物件就是物件導向程式的資料結構，這是實作UML類別圖中一對多和多對多類別關係（**Relationships**）的最佳選擇。在Java平台最常使用的是**Vector**、**ArrayList**和**Hashtable**等物件。

13-4-2 在設計模型加入集合類別- 在一對多關係加入集合類別

- 現在，我們就可以在【客戶】類別加上【帳戶清單】屬性來實作一對多的類別關係，和新增帳戶的操作，並且將它改為單向結合關係，如下圖所示：





13-5 物件設計：類別正規化

- 13-5-1 第一階物件正規化型式
- 13-5-2 第二階物件正規化型式
- 13-5-3 第三階物件正規化型式



13-5 物件設計：類別正規化- 說明

- 類別正規化（**Class Normalization**）是一個重組物件內部結構的過程，讓我們可以重新調整物件屬性和操作的結構來提高內聚力和降低耦合度，簡單的說，就是幫助我們建立良好的設計類別。
- 不過，在實作上我們更少採用類別正規化來建立良好的設計類別，因為它和第14章設計樣式（**Design Patterns**）在功能上是重疊的，但是，類別正規化的觀念十分重要，因為它是讀者進入物件設計（**Object Design**）學習設計樣式的重要橋樑。

13-5 物件設計：類別正規化- 三個階段

- 如同資料庫設計理論的資料表正規化，類別正規化基本上也分為三個階段，如下所示：
 - 第一階物件正規化型式（First Object Normal Form，1ONF）。
 - 第二階物件正規化型式（Second Object Normal Form，2ONF）。
 - 第三階物件正規化型式（Third Object Normal Form，3ONF）。

13-5-1 第一階物件正規化型式- 說明

- 第一階物件正規化型式是針對類別中集合物件的屬性，如果類別擁有集合物件的屬性，此屬性的相關操作只能針對集合物件本身，不能是其儲存的元素，否則會造成重複的群組操作。
- 當設計模型符合**1ONF**，表示在模型中的所有類別都符合**1ONF**。

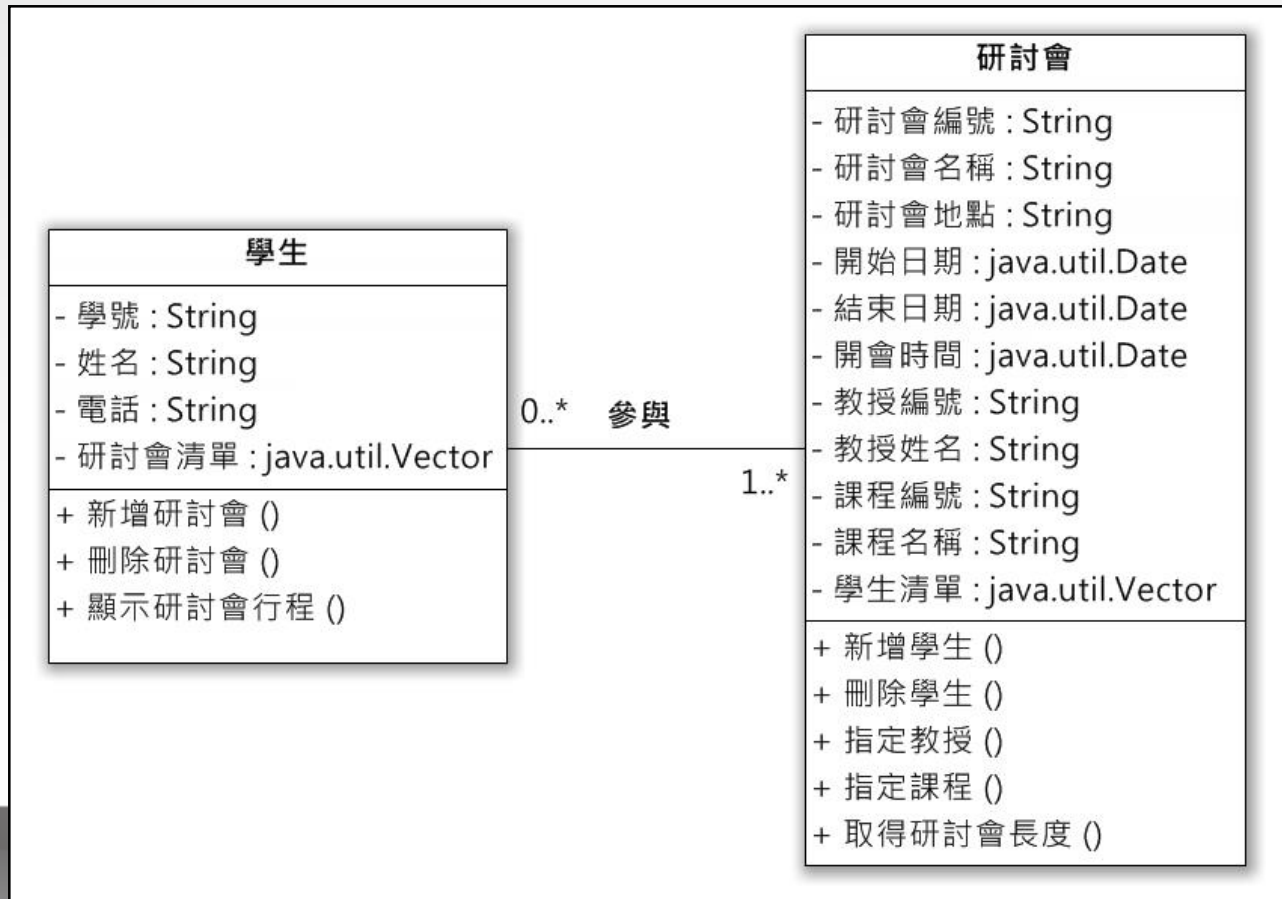
13-5-1 第一階物件正規化型式- 範例(不符合1ONF)

■ 例如：【學生】類別，如下圖所示：

學生
<ul style="list-style-type: none">- 學號 : String- 姓名 : String- 電話 : String- 研討會清單 : java.util.Vector
<ul style="list-style-type: none">+ 新增研討會 ()+ 刪除研討會 ()+ 顯示研討會行程 ()+ 指定教授 ()+ 指定課程 ()+ 取得研討會長度 ()

13-5-1 第一階物件正規化型式- 範例(符合1ONF)

- 1ONF的作法是將屬於集合物件元素的相關操作切割成另一個【研討會】類別，如下圖所示：

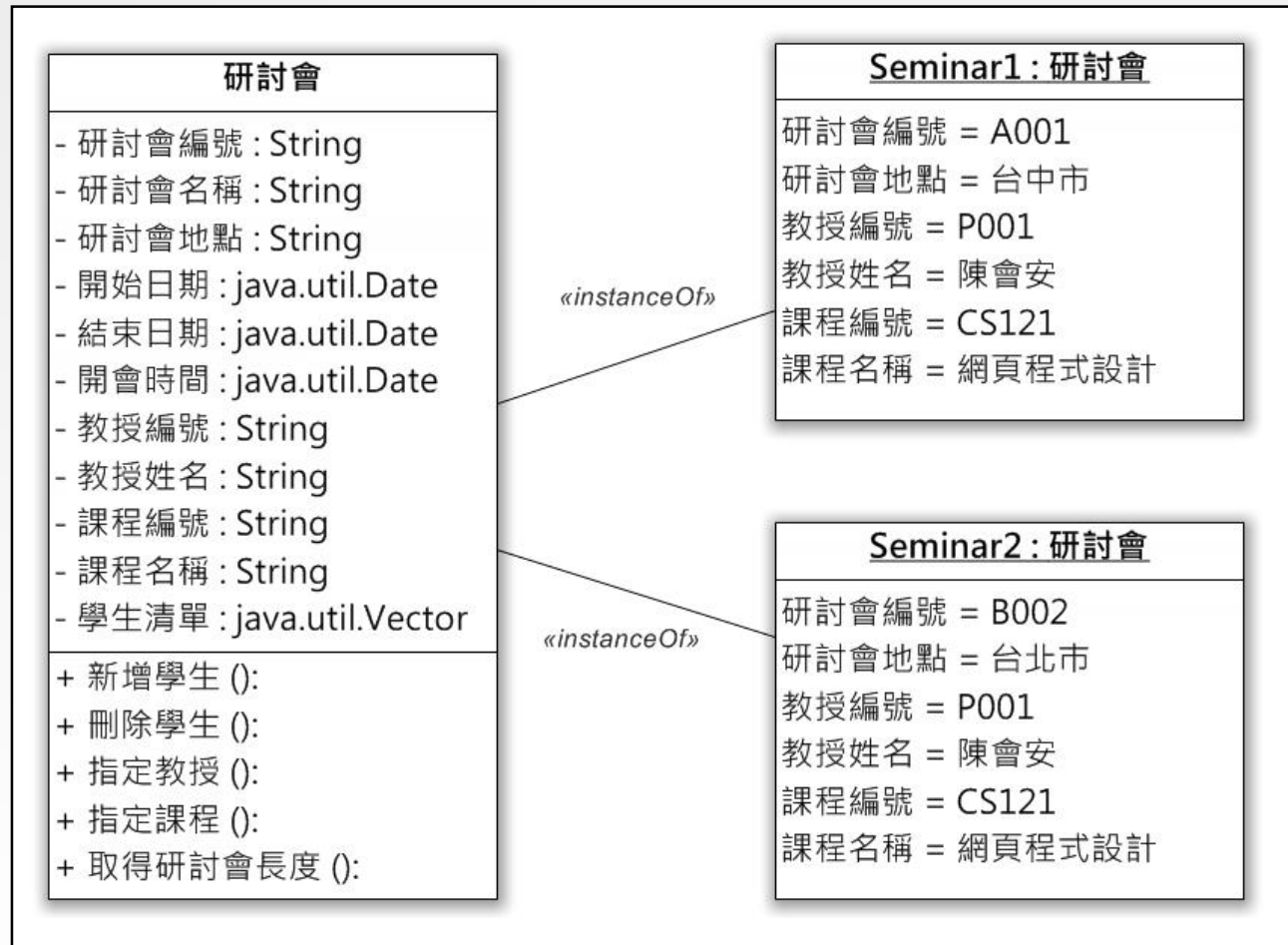


13-5-2 第二階物件正規化型式- 說明

- 第二階物件正規化型式是針對類別中多個群組屬性共用同一組操作的情況（將分屬多個觀念的群組屬性放在同一個類別），當更改其中一個屬性時，我們需要更改所有相同屬性值的物件。

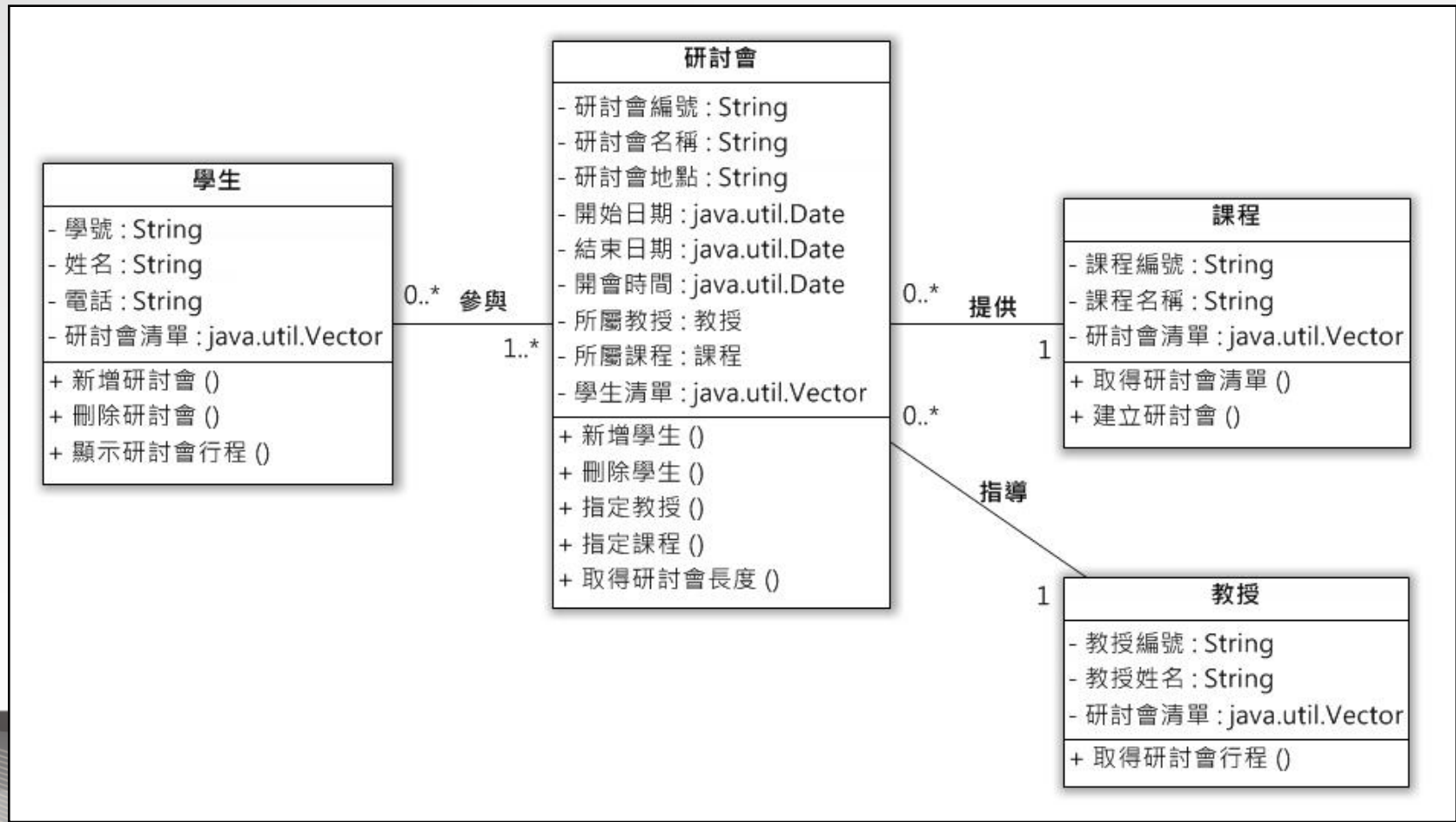
13-5-2 第二階物件正規化型式- 範例(不符合2ONF)

- 例如：更改【研討會】類別的【課程名稱】或【教授姓名】屬性，我們需要更改所有屬於同一個課程名稱或教授姓名的研討會物件，如右圖所示：



13-5-2 第二階物件正規化型式- 範例(符合2ONF)

- 我們可以將分屬不同觀念的群組屬性一一分割成類別，建立起符合2ONF的類別，如下圖所示：



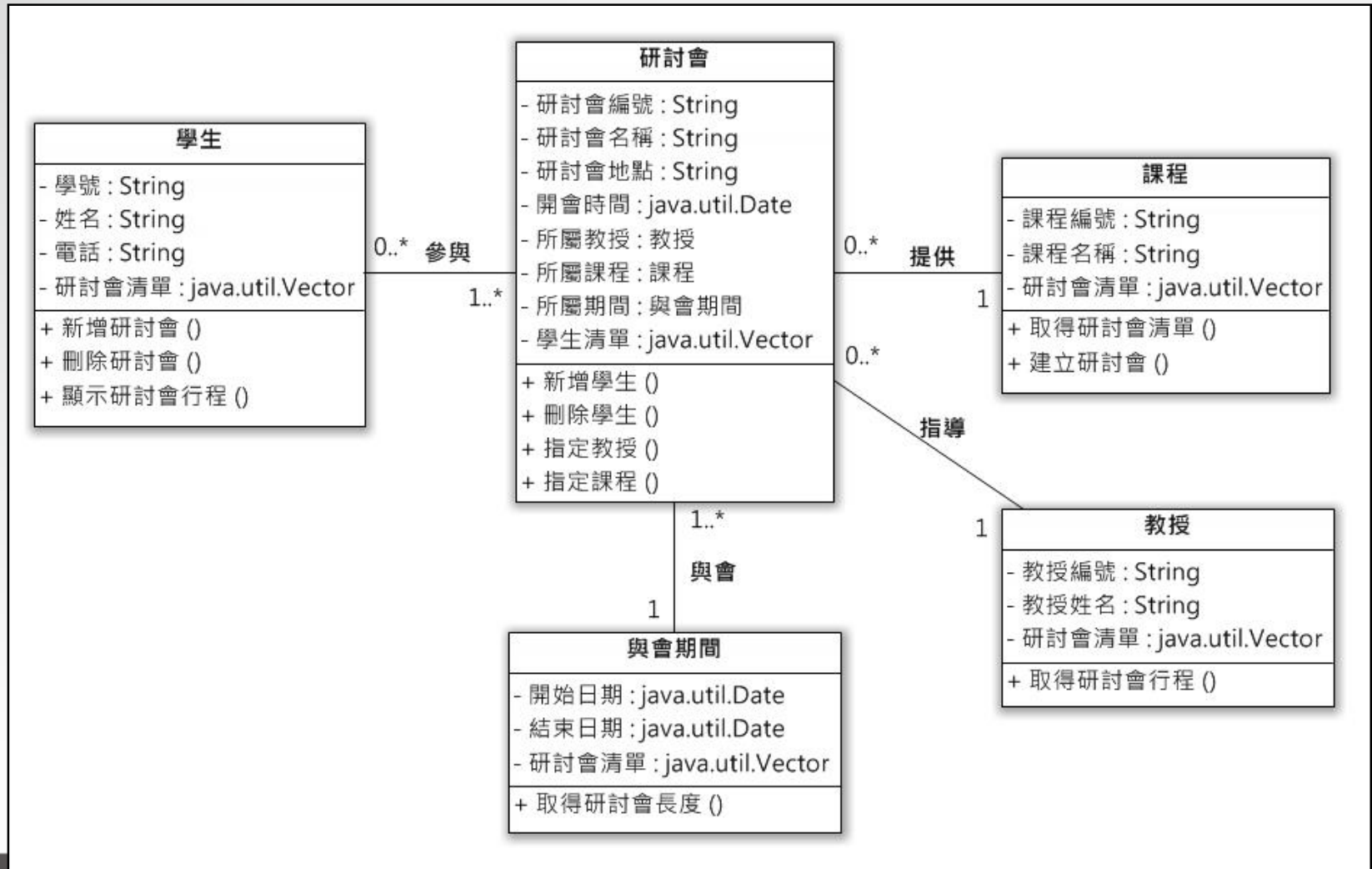
13-5-3 第三階物件正規化型式- 說明

- 第三階物件正規化型式需要同時考量類別的屬性和操作，當類別封裝不只一組的內聚操作，我們就需要進行3ONF，簡單的說，3ONF是在處理良好設計類別的充分性（**Sufficiency**）特點，設計類別的操作剛剛好滿足類別可以實現的功能。

13-5-3 第三階物件正規化型式- 範例(說明)

- 例如：【研討會】類別的操作包含一組處理研討會的操作和另一個計算開會期間的操作，所以不滿足充分性特點，我們可以將計算開會期間的操作切割成另一個處理日期範圍的【與會期間】類別，也就是進行3ONF。

13-5-3 第三階物件正規化型式- 範例(圖例)



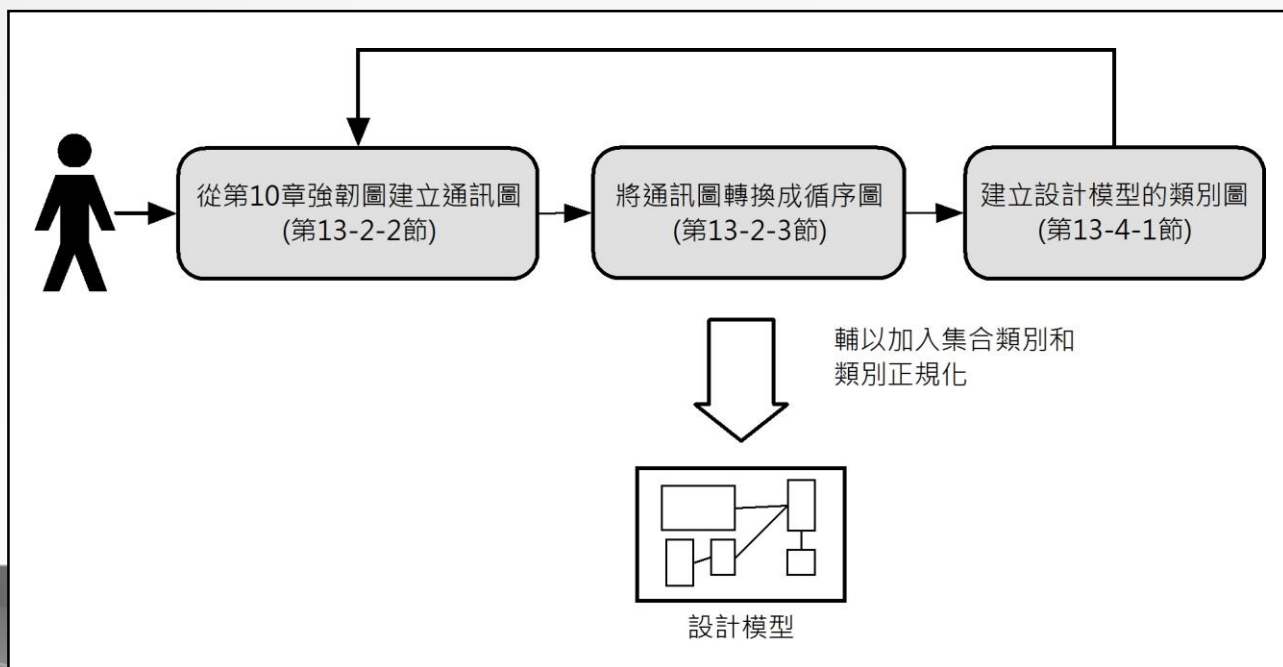


13-6 本章總結-說明

- 本章內容主要是說明如何建立物件導向設計的設計模型，筆者是從建立使用案例的系統循序圖開始，一步一步幫助讀者逐步建立起互動塑模所需的互動圖，然後更新分析模型來建立設計模型的類別圖。
- 筆者是從第10章的強韌圖開始，因為已經找出分析類別，所以只需一步一步加上執行活動圖動作送出的訊息，就可以將強韌圖轉換成通訊圖，再將它轉換成循序圖。當然，我們也可以直接從強韌圖建立循序圖。

13-6 本章總結-步驟

- 基於教學考量，為了讓初學者能夠實作物件導向設計（使用最少的經驗法則來實作），筆者的建議流程是從第10章的強韌圖開始，繪出通訊和循序圖的互動圖後，就可以使用第13-2-5節的規則找出屬性與操作來建立設計模型的類別圖，其流程如下圖所示：



End

