



第4章 資料庫設計與實體關聯模型

- 4-1 資料庫設計的基礎
- 4-2 實體關聯模型
- 4-3 建立實體關聯圖
- 4-4 實體關聯圖的常見錯誤
- 4-5 將實體關聯圖轉換成關聯表綱要
- 4-6 關聯式資料庫的邏輯資料模型





4-1 資料庫設計的基礎

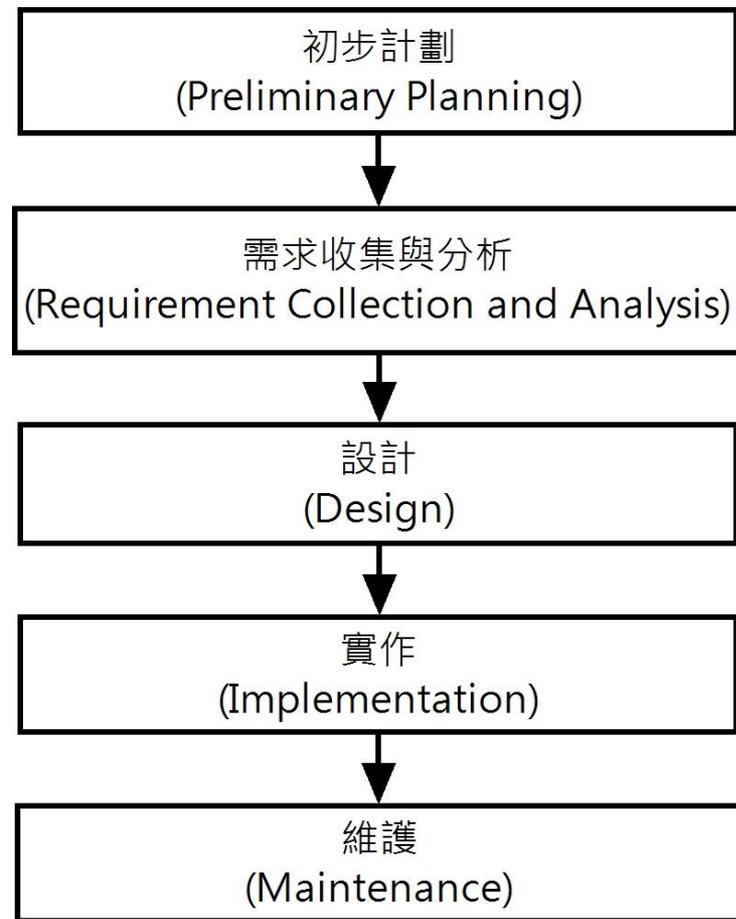
- 4-1-1 資料庫系統開發的生命周期
- 4-1-2 資料庫設計方法論





4-1-1 資料庫系統開發的生命周期 – 圖例

- 資料庫系統開發的生命周期就是資料庫系統的開發流程，它和其他應用程式的開發過程並沒有什麼不同。資料庫系統開發的生命周期可以分成五個階段，其流程圖如右圖所示：





4-1-1 資料庫系統開發的生命周期 – 種類

- 完整資料庫設計分成兩個部分，如下所示：
 - 資料庫設計（**Database Design**）：依照一定程序、方法和技術，使用結構化方式將概念資料模型轉換成資料庫的過程，在第4章和第5章將說明關聯式資料庫系統的資料庫設計。
 - 應用程式設計（**Application Design**）：設計程式來建立使用者介面，並且將商業處理流程轉換成應用程式的執行流程，以便使用者能夠輕易存取所需的資訊，即第11章的「資料庫程式設計」（**Database Programming**）。



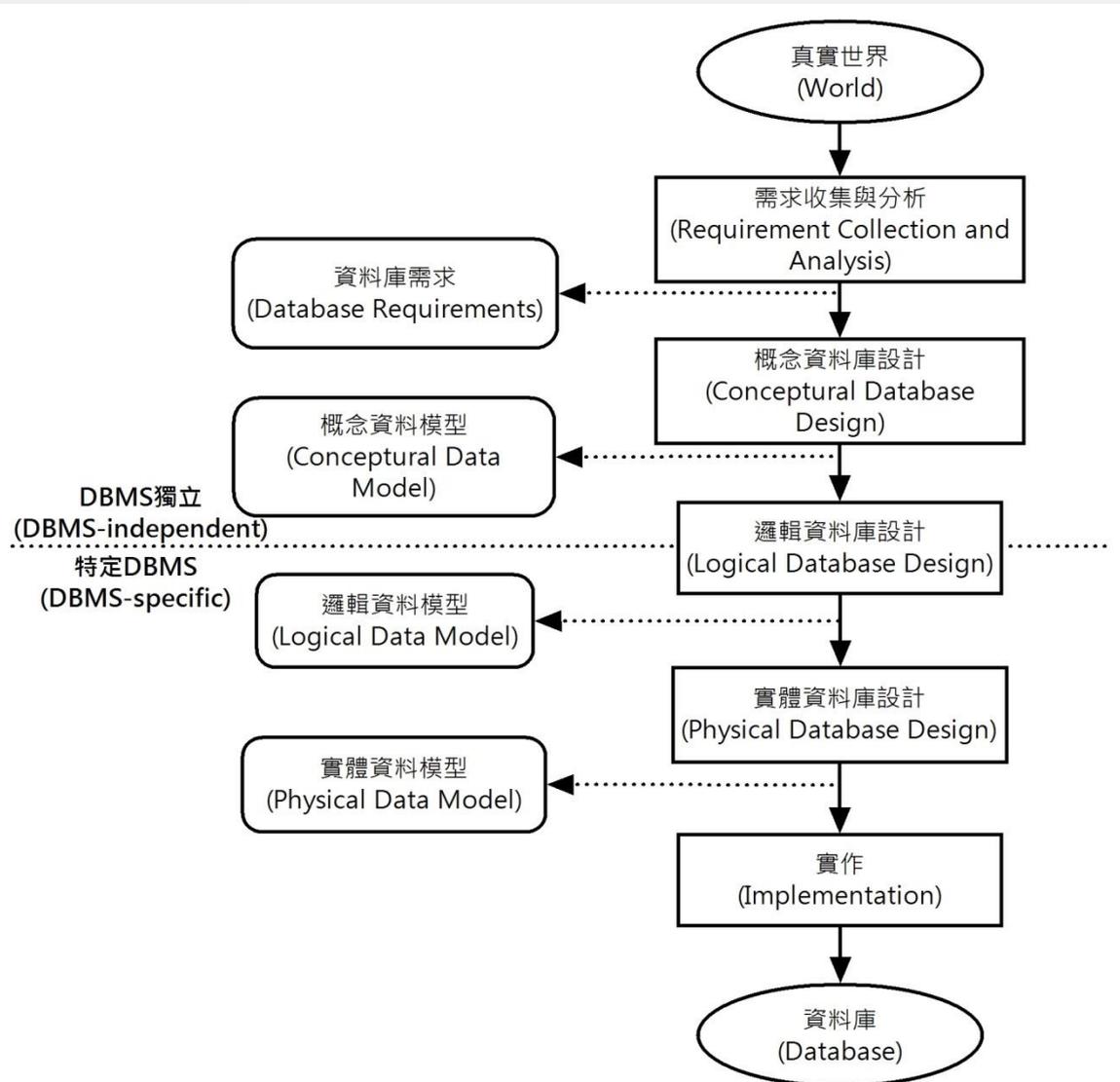
4-1-2 資料庫設計方法論 – 說明

- 資料庫設計方法論（Database Design Methodology）是使用特定程序、技術和工具的結構化設計方法，一種結構化的資料庫設計方法。簡單的說，這是一種計劃性、按部就班來進行資料庫設計，其優點如下所示：
 - 提供一步一步指引，可以幫助我們完成整個資料庫設計。
 - 減少或根本不會造成資料庫設計的錯誤。
 - 方便資料庫系統相關文件的整理，以便撰寫資料庫系統的規格書。
 - 當公司或組織的商業流程變更時，容易修改資料庫設計。



4-1-2 資料庫設計方法論 – 圖例

- 完整資料庫設計分成三個階段：概念、邏輯和實體資料庫設計，如右圖所示：





4-1-2 資料庫設計方法論 – 三個階段的說明

- 概念資料庫設計（**Conceptual Database Design**）：將資料庫需求轉換成概念資料模型的過程，並沒有針對特定資料庫管理系統或資料庫模型。實體關聯圖是目前最廣泛使用的概念資料模型之一。
- 邏輯資料庫設計（**Logical Database Design**）：將概念資料模型轉換成邏輯資料模型的過程，邏輯資料庫設計是針對特定資料庫管理系統（同時指定資料庫模型）來建立邏輯資料模型，例如：關聯式資料庫模型。
- 實體資料庫設計（**Physical Database Design**）：針對特定資料庫管理系統，例如：**SQL Server**，將邏輯資料模型轉換成指定資料庫管理系統的實體資料模型。



4-2 實體關聯模型

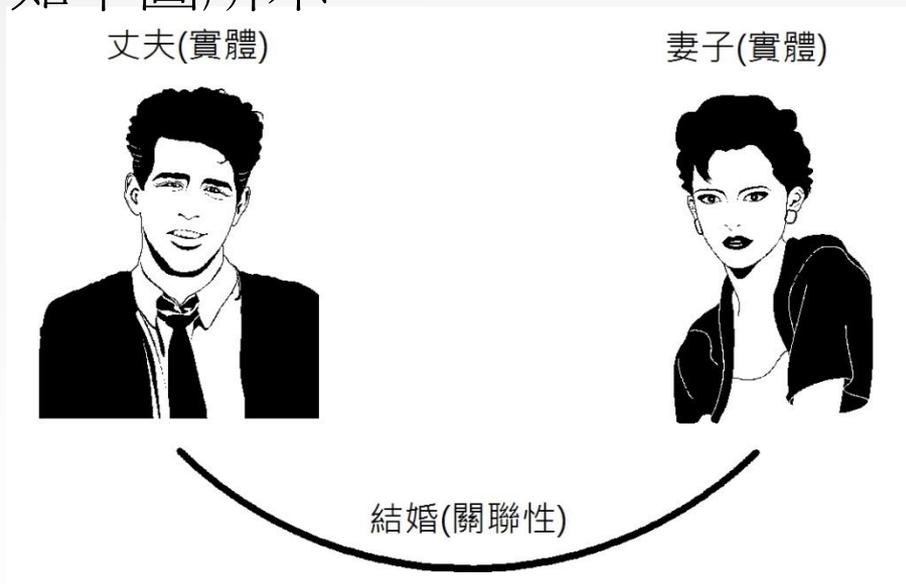
- 4-2-1 實體關聯模型的基礎
- 4-2-2 實體關聯模型與自然語言
- 4-2-3 實體關聯模型與資料庫設計





4-2 實體關聯模型

- 「實體關聯模型」(Entity-Relationship Model; ERM) 是 1976 年 Peter Chen 開發的資料塑模方法，它是目前資料庫系統分析和設計最常使用的方法。實體關聯模型是使用實體 (Entity) 與關聯性 (Relationship) 來描述資料和資料間的關係，如下圖所示：





4-2-1 實體關聯模型的基礎 – 說明

- 實體關聯模型是將商業領域的公司或組織的資料以邏輯方式呈現，實體關聯模型相信實體與關聯性（Relationship），是真實世界最自然的資料塑模（Data Modeling）方式。我們可以使用實體和關聯性來描述真實世界的資料。
- 換個角度來說，實體關聯模型是將真實世界的資料塑模成邏輯關聯資料（Logically Related Data），這就是儲存在資料庫的資料。



4-2-1 實體關聯模型的基礎 – 實體與實體型態

- 實體（**Entities**）是從真實世界的資料中識別出的東西。例如：人、客戶、產品、供應商、地方、物件、事件或一個觀念，也稱為實體實例（**Entity Instances**），其特性如下所示：
 - 實體一定屬於資料庫系統範圍內的東西。
 - 實體至少擁有一個不是鍵（即關聯表主鍵）的屬性。
- 實體可以分類成不同的實體型態（**Entity Type**），並且以實體型態名稱（**Entity Type Name**）來命名。
 - 同一類實例的集合稱為實體集合（**Entity Set**），即關聯表。



4-2-1 實體關聯模型的基礎 – 關聯與關聯型態

- 關聯性（Relationships）是二個或多個實體間所擁有的關係，也稱為關聯實例（Relationship Instances）。
- 將關聯實例收集起的集合稱為關聯集合（Relationship Set），可以將它歸類成關聯型態（Relationship Type）。



4-2-1 實體關聯模型的基礎 – 屬性與屬性型態

- 屬性（**Attributes**）是實體所擁有的特性。例如：學生實體擁有學號、姓名、地址和電話等屬性。屬性的所有可能值稱為「屬性型態」（**Attribute Type**）或值集合（**Value Set**），相當於是關聯表的定義域（**Domain**）。



4-2-2 實體關聯模型與自然語言 – 說明

- 實體關聯模型與自然語言之間擁有對應關係，我們可以直接分析自然語言的句子結構來找出實體與關聯性（Relationship），如下表所示：

英文文法結構	實體關聯圖
普通名詞（Common Noun）	實體型態（Entity Type）
專有名詞（Proper Noun）	實體（Entity）
及物動詞（Transitive Verb）	關聯型態（Relationship Type）
不及物動詞（Intransitive Verb）	屬性型態（Attribute Type）
形容詞（Adjective）	實體的屬性（Attribute）
副詞（Adverb）	關聯性的屬性（Attribute）



4-2-2 實體關聯模型與自然語言 – 範例

- 描述系統的英文句子如果擁有下列句型，需要先進行句型轉換，如下所示：

“There are A in B.” -> “B hasA.”

- 上述B與A是實體型態，has是關聯型態。舉一個實例，如下所示：

“There are many employees in every department.”
-> “Every department has many employees.”

- 上述employees和department是實體型態，has是關聯型態。



4-2-3 實體關聯模型與資料庫設計

- 實體關聯模型是目前最廣泛使用在軟體系統分析和資料庫設計的塑模技術，其目的是希望能夠正確塑模使用者的需求，建立符合使用者需求的軟體系統和資料庫。
- 以資料庫設計來說，實體關聯圖可以在概念資料庫設計建立概念資料模型，描述使用者和設計者眼中的真實世界，作為之間的溝通橋樑。因為實體關聯模型的實體型態可以對應關聯表，關聯性就是外來鍵參考，邏輯資料庫設計一樣可以使用實體關聯圖來建立邏輯資料模型。



4-3 建立實體關聯圖

- 4-3-1 實體型態
- 4-3-2 關聯型態
- 4-3-3 關聯限制條件
- 4-3-4 屬性
- 4-3-5 弱實體型態





4-3 建立實體關聯圖 – 步驟

■ 實體關聯圖是使用圖形符號建立的實體關聯模型，以資料庫設計來說，通常是使用在概念資料庫設計，以便了解使用者的需求。實體關聯圖的基本建立步驟，如下所示：

Step 1：從系統需求找出實體型態。

Step 2：找出實體型態與其他實體型態間的關聯性（Relationship）。

Step 3：定義實體型態間的關聯型態種類是：一對一、一對多或多對多關聯型態。

Step 4：定義實體型態的屬性型態與主鍵。



4-3 建立實體關聯圖 – 圖形符號

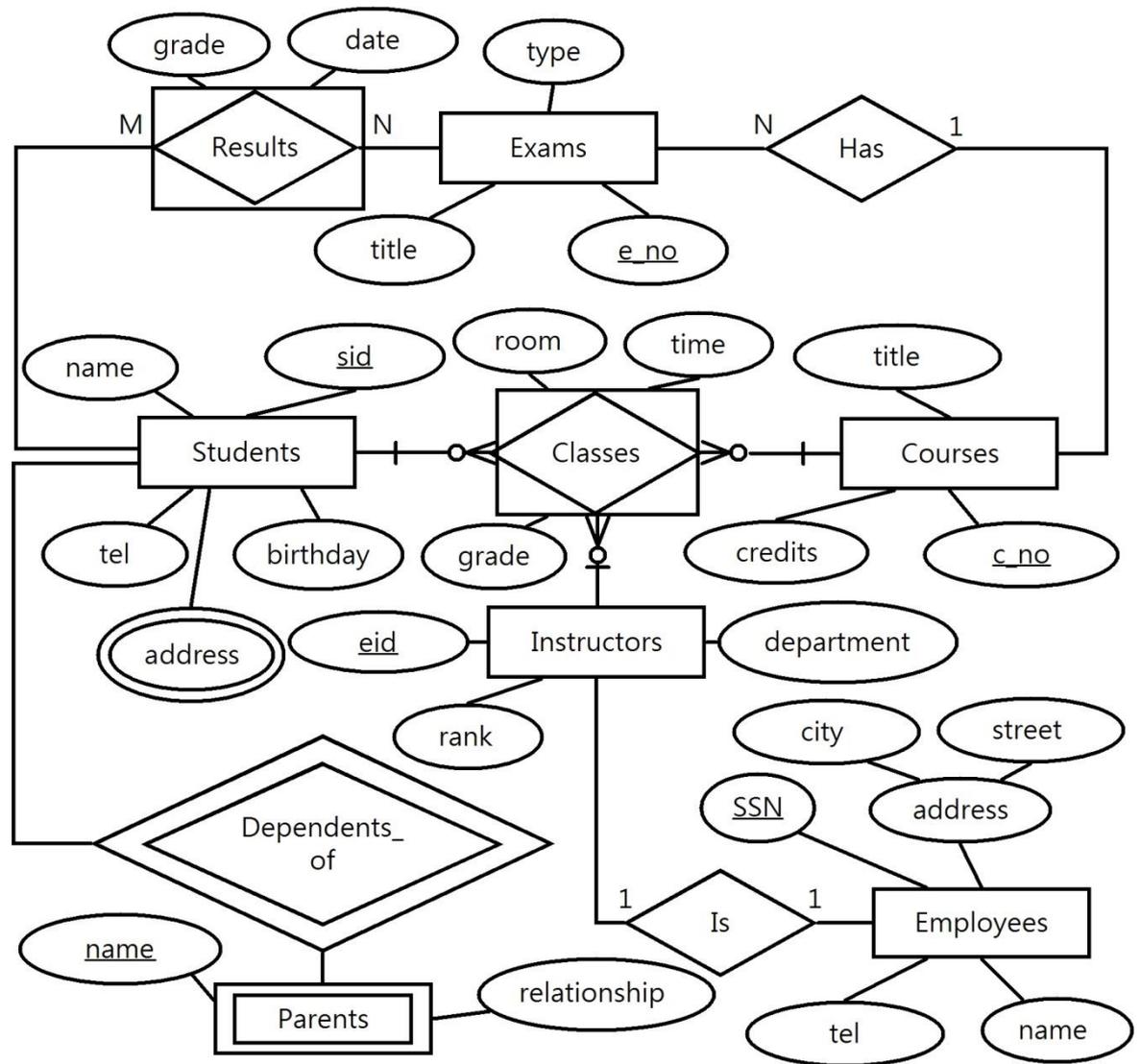
- 在實體關聯圖使用的圖形符號，整理如右表所示：

實體關聯圖的種類	圖形符號
實體 (Entity)	
弱實體 (Weak Entity)	
關聯性 (Relationship)	
識別關聯性 (Identifying Relationship)	
屬性 (Attribute)	
鍵屬性 (Key Attribute)	
複合屬性 (Composite Attribute)	
多重值屬性 (Multivalued Attribute)	
導出屬性 (Derived Attribute)	
E1 全部參與(Total Participation) R	
E2 部分參與(Partial Participation) R	



4-3 建立實體關聯圖 - 範例

■ 學校School資料庫的實體關聯圖，如右圖所示：





4-3-1 實體型態 – 說明

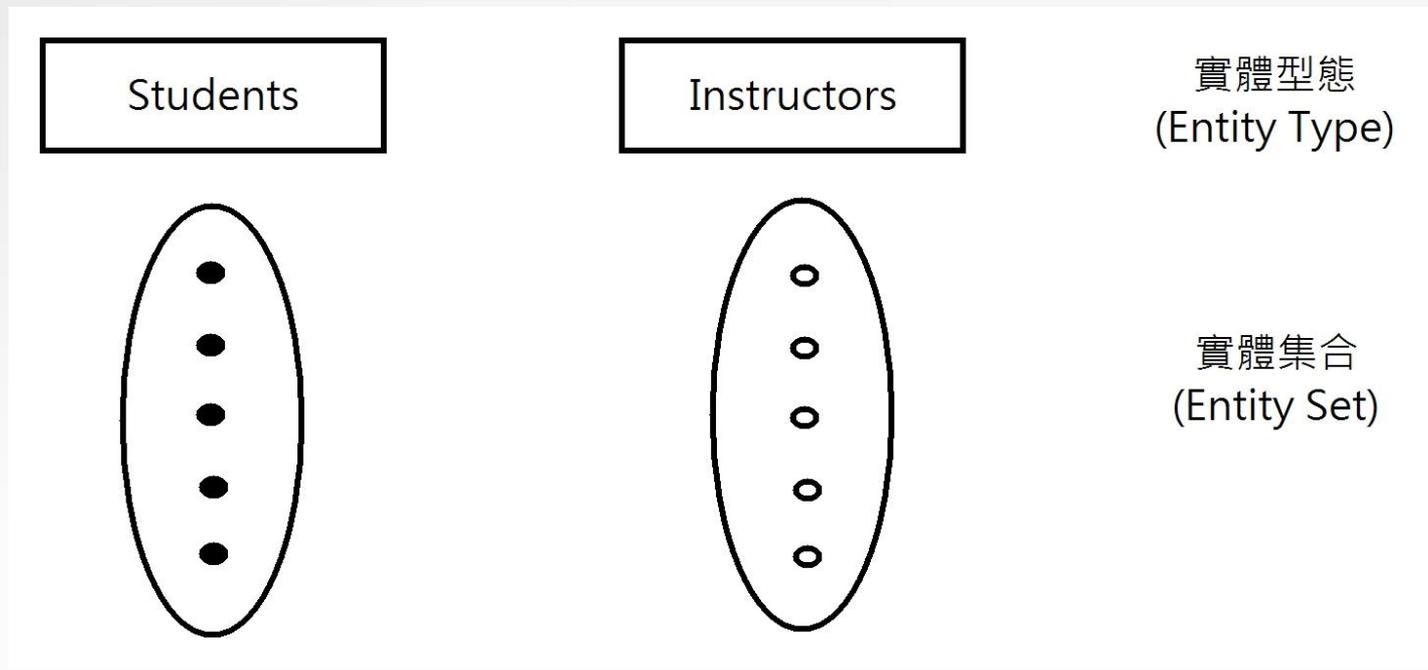
- 實體型態（Entity Types）是分類的實體集合，也就是同一類東西。其圖形符號是長方形節點，內為實體型態的名稱，如下圖所示：





4-3-1 實體型態 – 實體集合

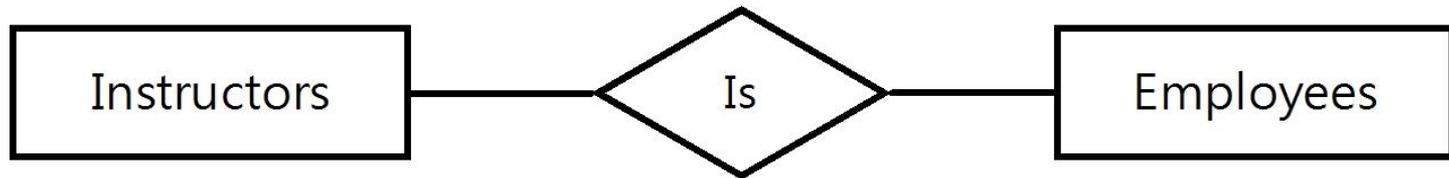
- 每一位學生稱為實體型態的實例（Instances），或簡稱為實體，其集合稱為「實體集合」（Entity Set），如下圖所示：





4-3-2 關聯型態 – 說明

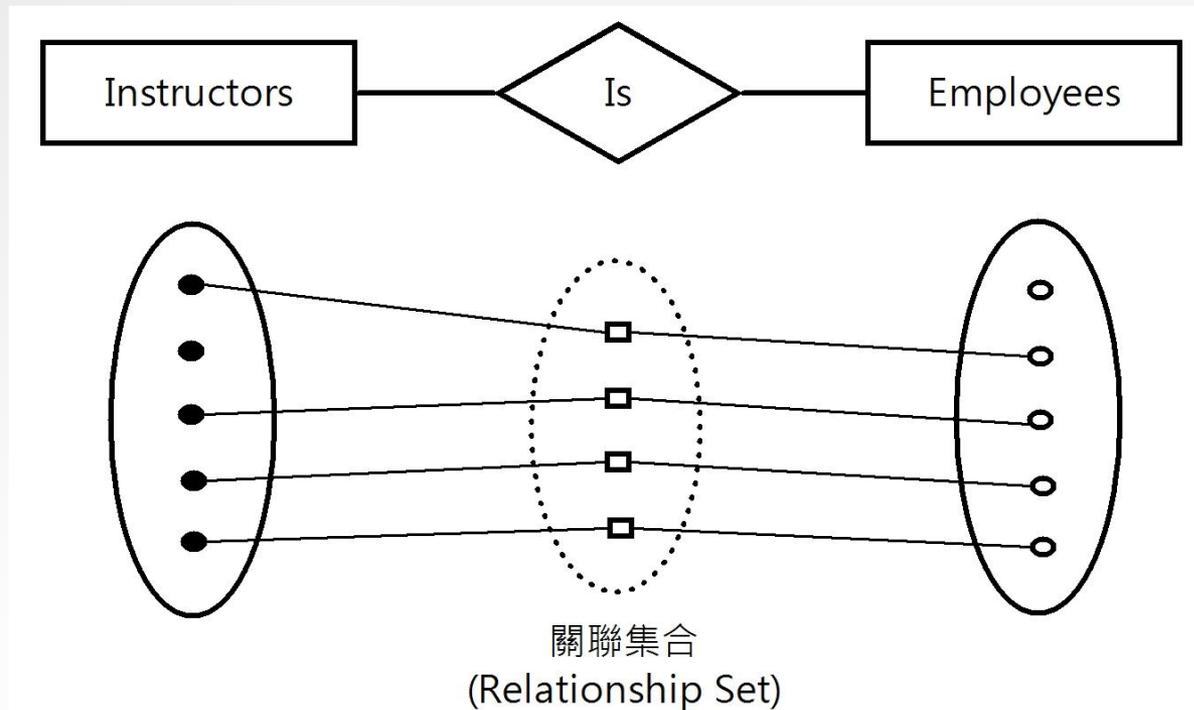
- 關聯型態（Relationship Types）也稱為「結合實體型態」（Associate Entity Type），其目的是連接一、二個或以上相關的實體型態。圖形符號是使用菱形節點，在菱形端點使用實線與關聯性（Relationship）的實體型態連接，如下圖所示：





4-3-2 關聯型態 – 關聯集合

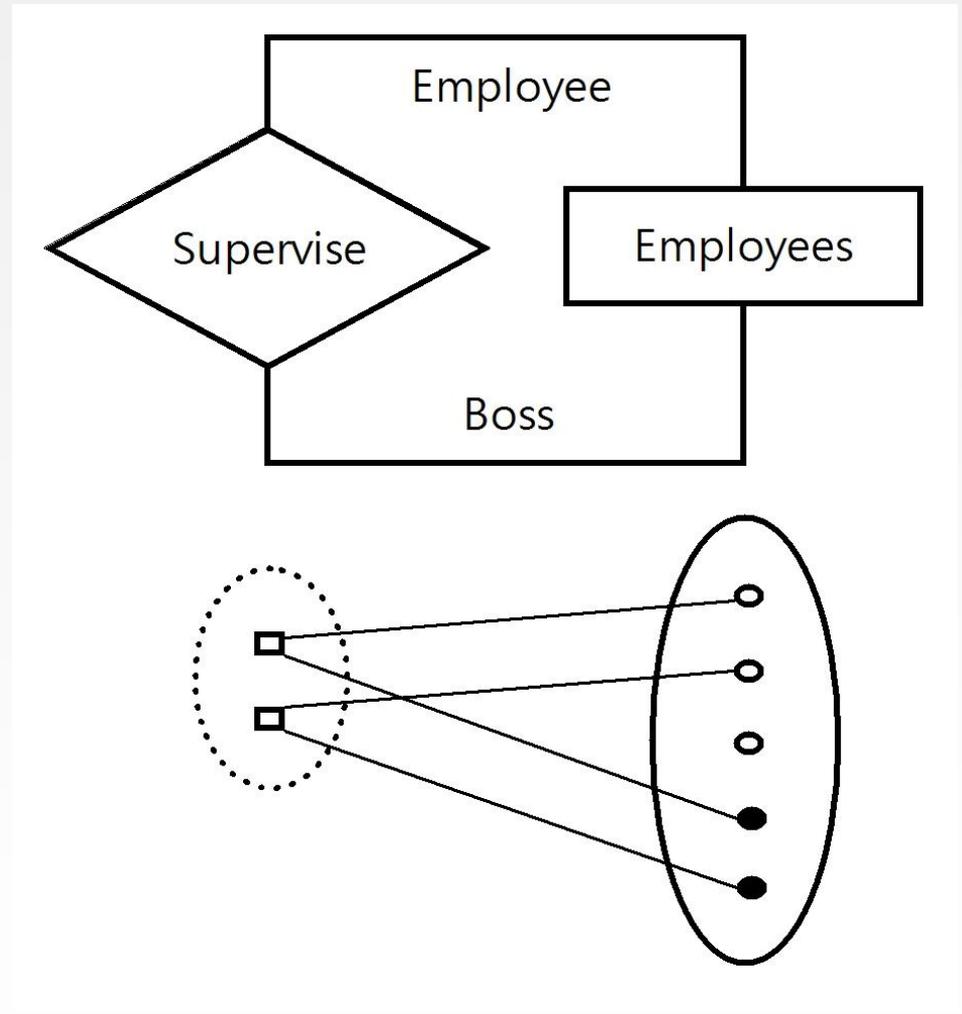
- 關聯型態也可以建立實例，實例的集合稱為「關聯集合」（Relationship Set），如下圖所示：





4-3-2 關聯型態 – 自身關聯性

- 實體型態本身會參與自己的關聯型態，稱為「遞迴」（Recursive）或「自身關聯性」（Self Relationship），如右圖所示：





4-3-3 關聯限制條件 – 說明

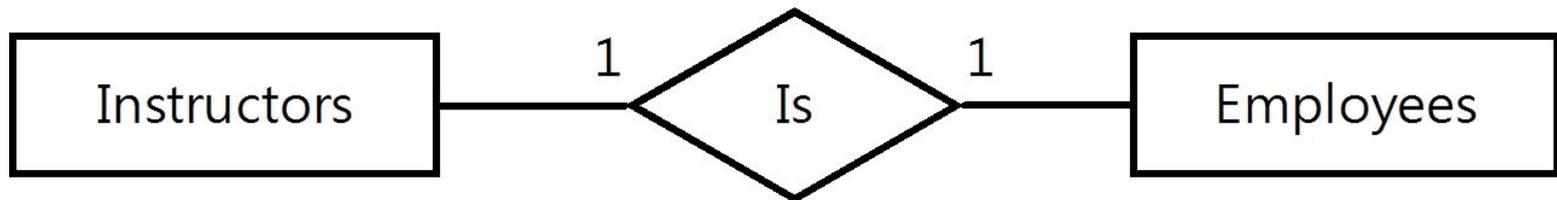
- 在實體關聯圖中，關聯型態連接的實體型態可以指定限制條件，稱為「關聯限制條件」（**Relationship Constraints**）。
- 關聯限制條件有三種，如下：
 - 基數比限制條件
 - 基數限制條件
 - 參與限制條件



4-3-3 關聯限制條件 – 基數比限制條件(一對一)

■ 基數比限制條件（Cardinality Ratio Constraints）限制關聯實體型態連接的實體個數，主要可以分為三種，如下所示：

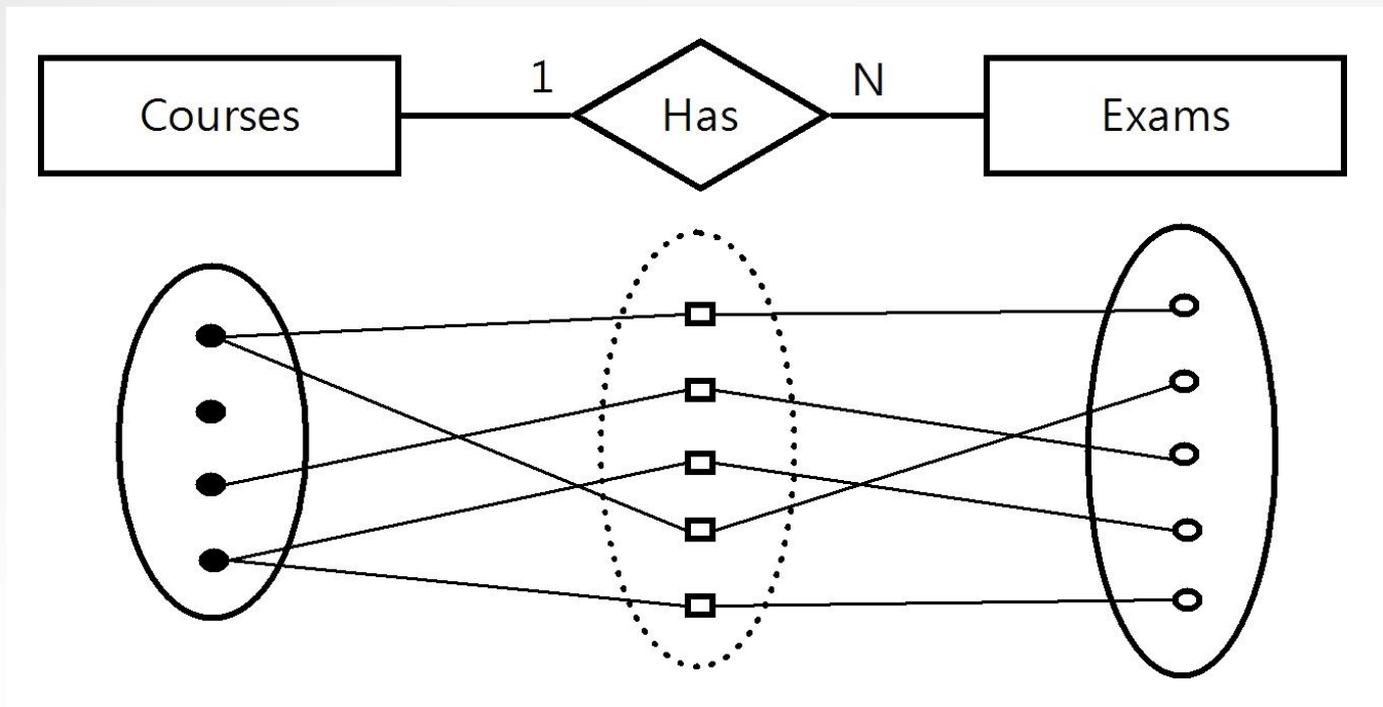
- 一對一關聯性（One-to-one Relationship；1:1）：指一個實體只關聯到另一個實體。例如：一位講師只能是學校的一位員工，如下圖所示：





4-3-3 關聯限制條件 – 基數比限制條件(一對多)

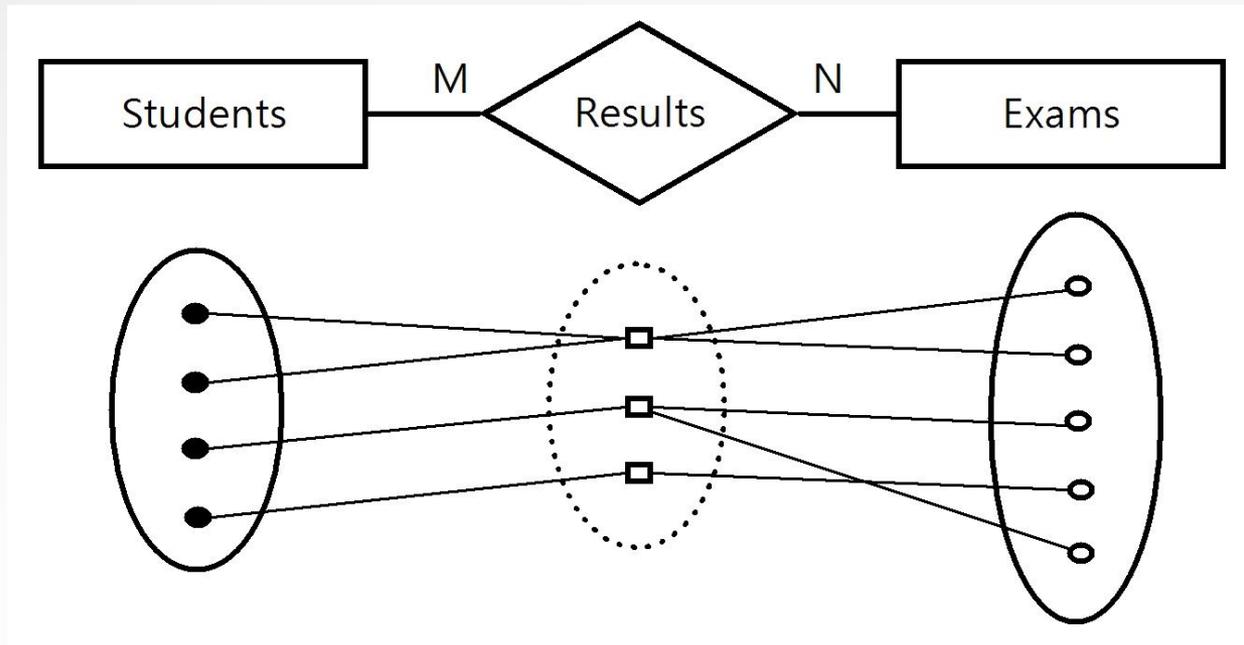
- 一對多關聯性（One-to-many Relationship；1:N）：指一個實體關聯到多個實體。例如：一門課程擁有小考、期中考和期末考等多次考試，如下圖所示：





4-3-3 關聯限制條件 – 基數比限制條件(多對多)

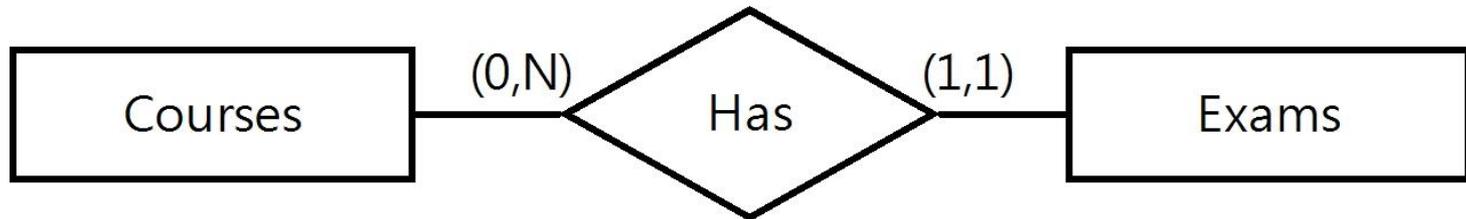
- 多對多關聯性（Many-to-many Relationship；M:N）：
指多個實體關聯到多個其他實體。例如：學生可以參加多次考試，反過來，考試可以讓多位學生應試，如下圖所示：





4-3-3 關聯限制條件 – 基數限制條件

- 基數限制條件（Cardinality Constraints）是在關聯型態的連接線上，標示實體允許參與關聯型態的數量範圍： $(1,N)$ 、 $(0,N)$ 、 $(1,1)$ 和 $(0,1)$ 等從0到N個。
 -
 - 例如：課程Courses關聯型態是以 $(0,N)$ 範圍參與Has關聯型態，考試Exams關聯型態是以 $(1,1)$ 參與，如下圖所示：





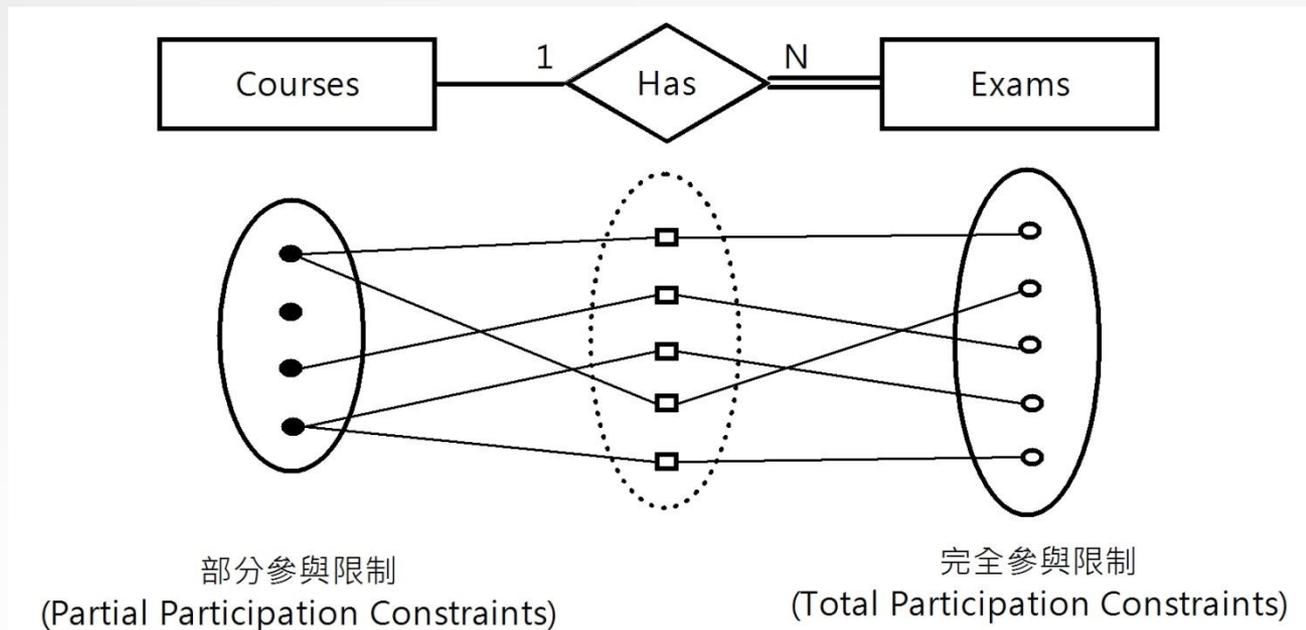
4-3-3 關聯限制條件 – 參與限制條件(說明)

- 參與限制條件（**Participation Constraints**）是指實體集合的實體全部或部分參與關聯型態，可以分為兩種，如下所示：
 - 全部參與限制條件（**Total Participation Constraints**）：所有實體集合的實體都參與關聯型態，圖形符號是使用雙線來標示，也稱為「存在相依」（**Existence Dependency**）。
 - 部分參與限制條件（**Partial Participation Constraints**）：在實體集合只有部分實體參與關聯型態，圖形符號是使用單線來標示。



4-3-3 關聯限制條件 – 參與限制條件(圖例)

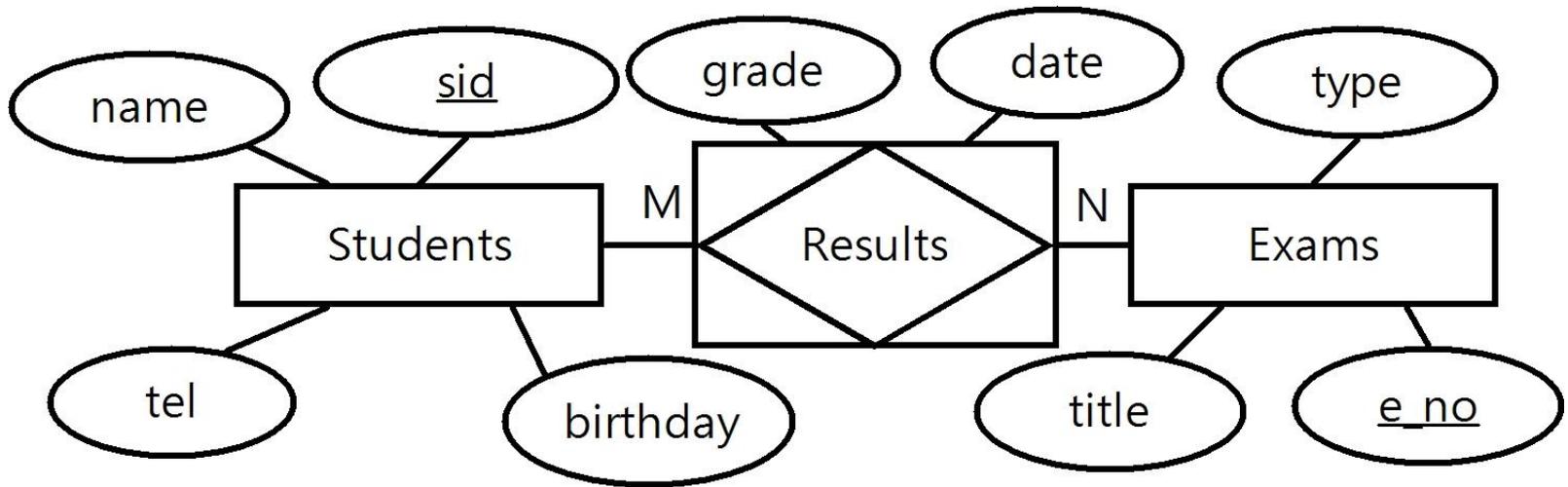
- 例如：在Courses與Exams實體型態的一對多關聯性中，Courses實體只有部分參與，因為課程可能沒有考試；Exams實體全部參與關聯，因為如果課程有考試，就一定存在考試實體，不會有考試而沒有課程，如下圖所示：





4-3-4 屬性 - 說明

- 實體關聯圖的實體與關聯型態可以擁有0到多個屬性（Attributes），屬性是使用橢圓型圖形符號的節點，使用單線與實體或關聯型態來連接，如下圖所示：





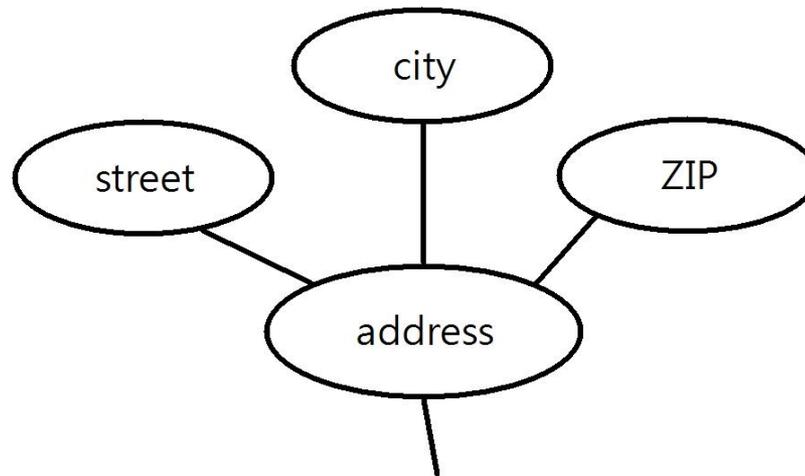
4-3-4 屬性 – 單元值屬性型態

- 屬性是一組值的集合，這些值是屬性的可能值，稱為值集合（**Value Set**），即定義域。屬性可以分成很多種，如下所示：
 - 單元值屬性型態（**Atomic Attribute Types**）：實體與關聯型態的最基本屬性型態是單元值。例如：**Students**實體型態的**sid**、**name**、**birthday**和**tel**屬性。



4-3-4 屬性 – 複合屬性型態

- 複合屬性型態（Composite Attribute Types）：屬性是由多個單元值屬性組成，使用樹狀的單元值屬性圖形符號來表示。例如：員工Employees實體型態的address複合屬性是由street、city和ZIP單元值屬性組成，如下圖所示：





4-3-4 屬性 – 多重值屬性型態

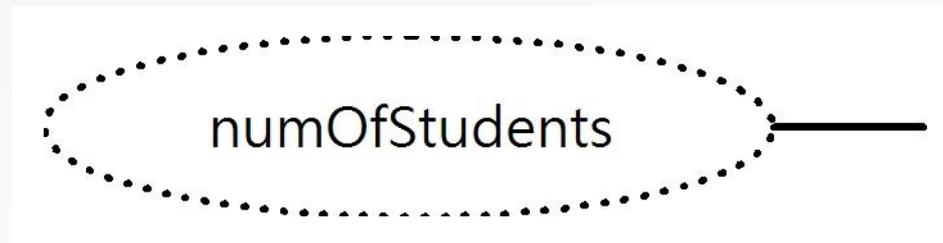
- 多重值屬性型態（Multivalued Attribute Types）：屬性值不是單元值，而是多重值，使用雙線的橢圓形節點符號標示。例如：學生Students實體型態的address屬性，用來記錄學生擁有多個通訊地址的多重值屬性，如下圖所示：





4-3-4 屬性 – 導出屬性型態

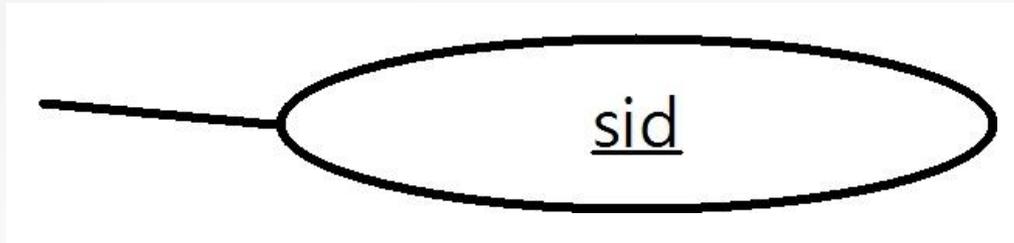
- 導出屬性型態（**Derived Attribute Types**）：由其他屬性計算出的屬性，它是使用虛線的橢圓形節點符號來標示。例如：**Exams**實體型態的**numOfStudents**屬性是記錄參加考試的學生數，屬性值可以從**Results**關聯型態計算而得，如下圖所示：





4-3-4 屬性 – 鍵屬性型態

- 鍵屬性型態（Key Attribute Types）：如果屬性是實體型態中用來識別實體的屬性，其角色相當於關聯表的主鍵，鍵屬性型態是在名稱下加上底線來標示。例如：
： Students實體型態的主鍵是sid屬性，如下圖所示：





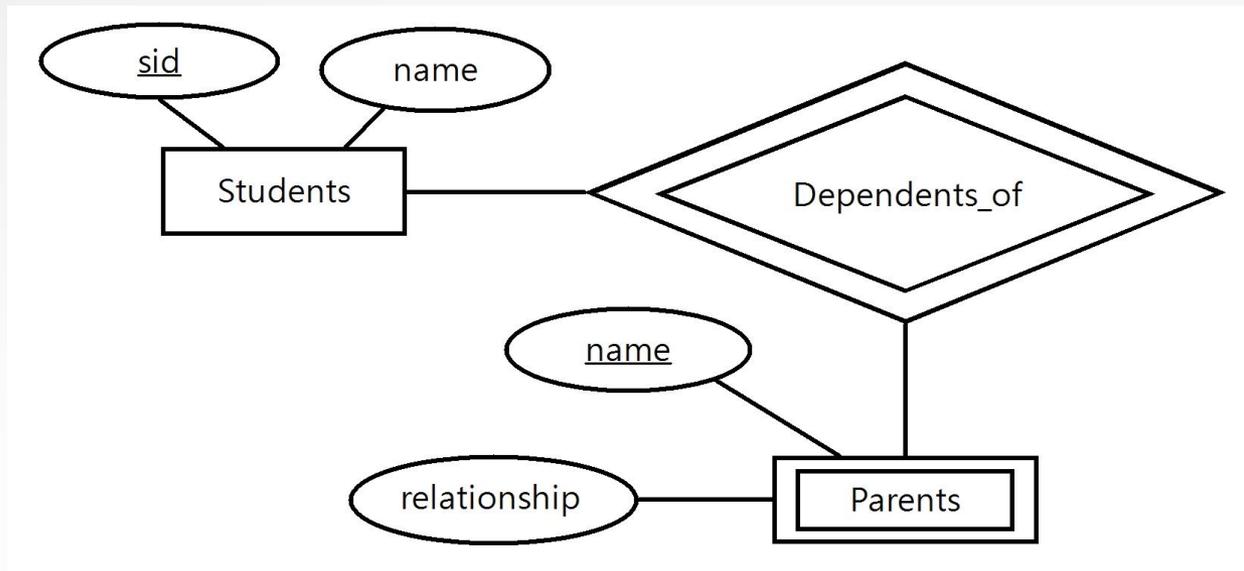
4-3-5 弱實體型態 – 說明

- 弱實體型態（**Weak Entity Types**）是一種需要依賴其他實體型態才能存在的實體形態，簡單的說，這是一種沒有主鍵的實體型態。例如：學生家長是一種弱實體，因為只有學生實體存在，家長實體才會存在。
- 相對的，擁有主鍵的實體型態稱為「一般實體型態」（**Regular Entity Type**）或強實體型態（**Strong Entity Type**）。在實體關聯圖的弱實體型態是使用雙框的長方形圖形符號來標示。



4-3-5 弱實體型態 – 圖例

- 弱實體型態一定需要關聯到一個強實體型態，以便識別其身份，強實體型態稱為「識別實體型態」（Identifying Entity Type），使用的關聯型態稱為「識別關聯型態」（Identifying Relationship Type），以雙框的菱形圖形符號來表示，例如：Dependents_of，如下圖所示：





4-4 實體關聯圖的常見錯誤

- 4-4-1 扇形陷阱
- 4-4-2 斷層陷阱





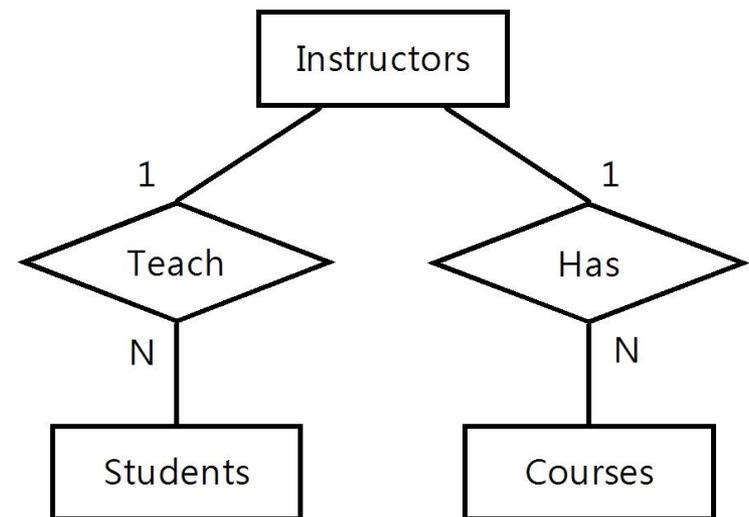
4-4 實體關聯圖的常見錯誤

- 在實體關聯圖不只需要識別出實體，還需要找出實體間的關聯性（**Relationship**）。但是常常因為錯誤解釋實體間的關係，反而造成實體關聯圖的錯誤，稱為「**連接陷阱**」（**Connection Traps**）。
- 實體關聯圖最常見的兩種連接陷阱為：
 - 扇形陷阱（**Fan Traps**）
 - 斷層陷阱（**Chasm Traps**）



4-4-1 扇形陷阱 – 說明

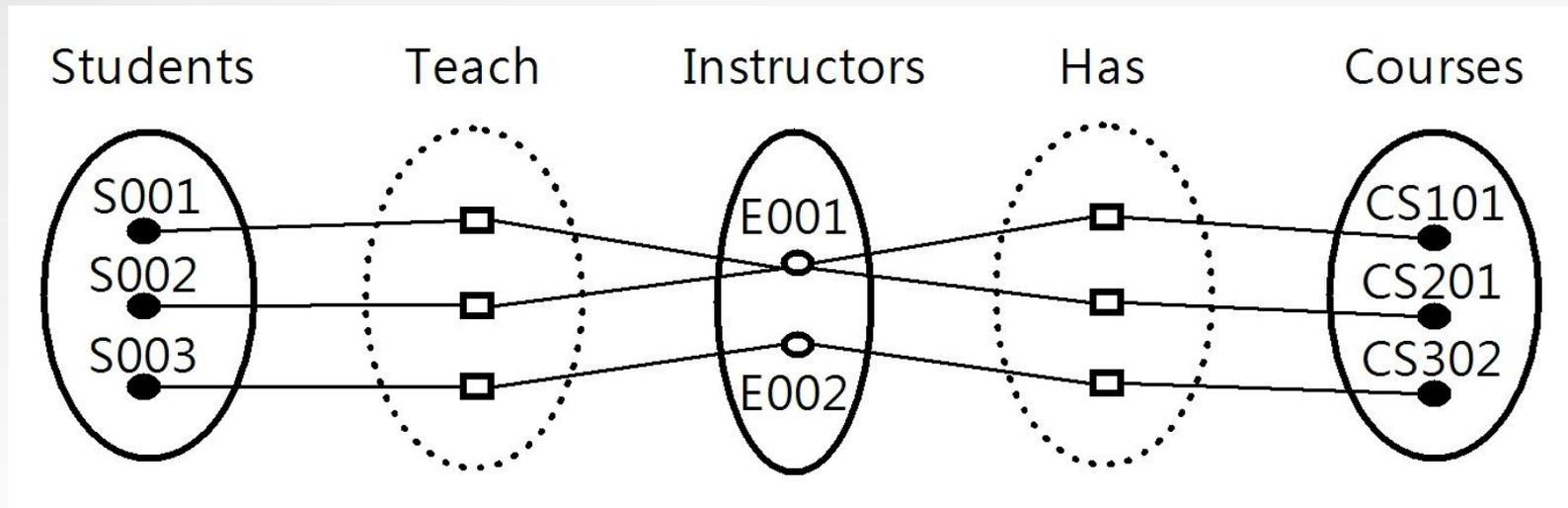
- 實體關聯圖的扇形陷阱是指實體間的關聯性（**Relationship**）會令人產生混淆，扇形陷阱通常發生在各實體型態間擁有多個一對多關聯性，如同扇形散開（這是名稱的由來）。但是各實體型態間的關聯性產生令人混淆的問題。
- 例如：學生、講師與課程實體型態間擁有**3**個一對多關聯性，如下所示：
 - 講師可以教多位學生。
 - 講師可以教多門課程。
 - 學生可以選修多門課。





4-4-1 扇形陷阱 – 問題

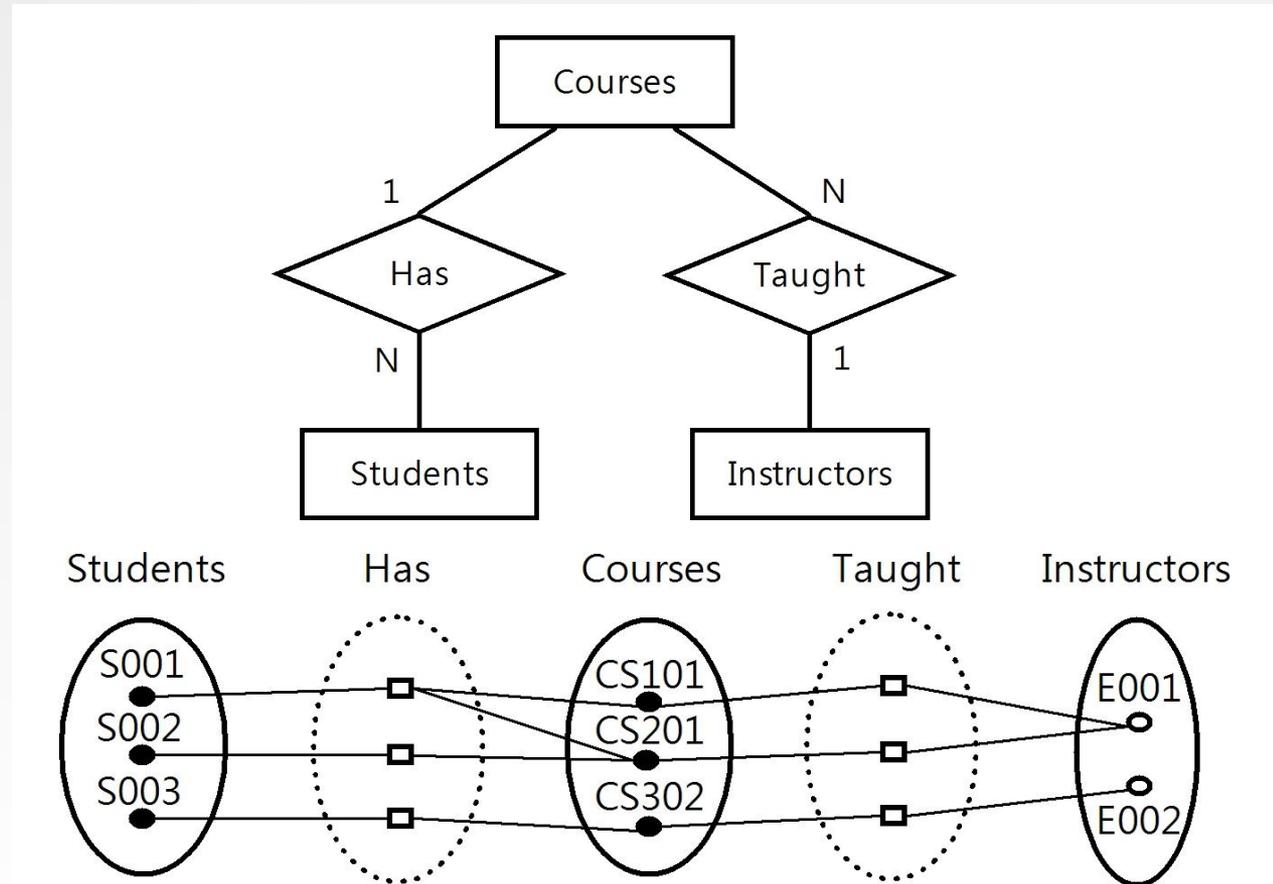
- 但是，學生選修多門課程的一對多關聯性並沒有辦法建立在講師與學生和講師與課程的兩個一對多關聯性上，如此就會產生混淆，如下圖所示：





4-4-1 扇形陷阱 – 解決方案

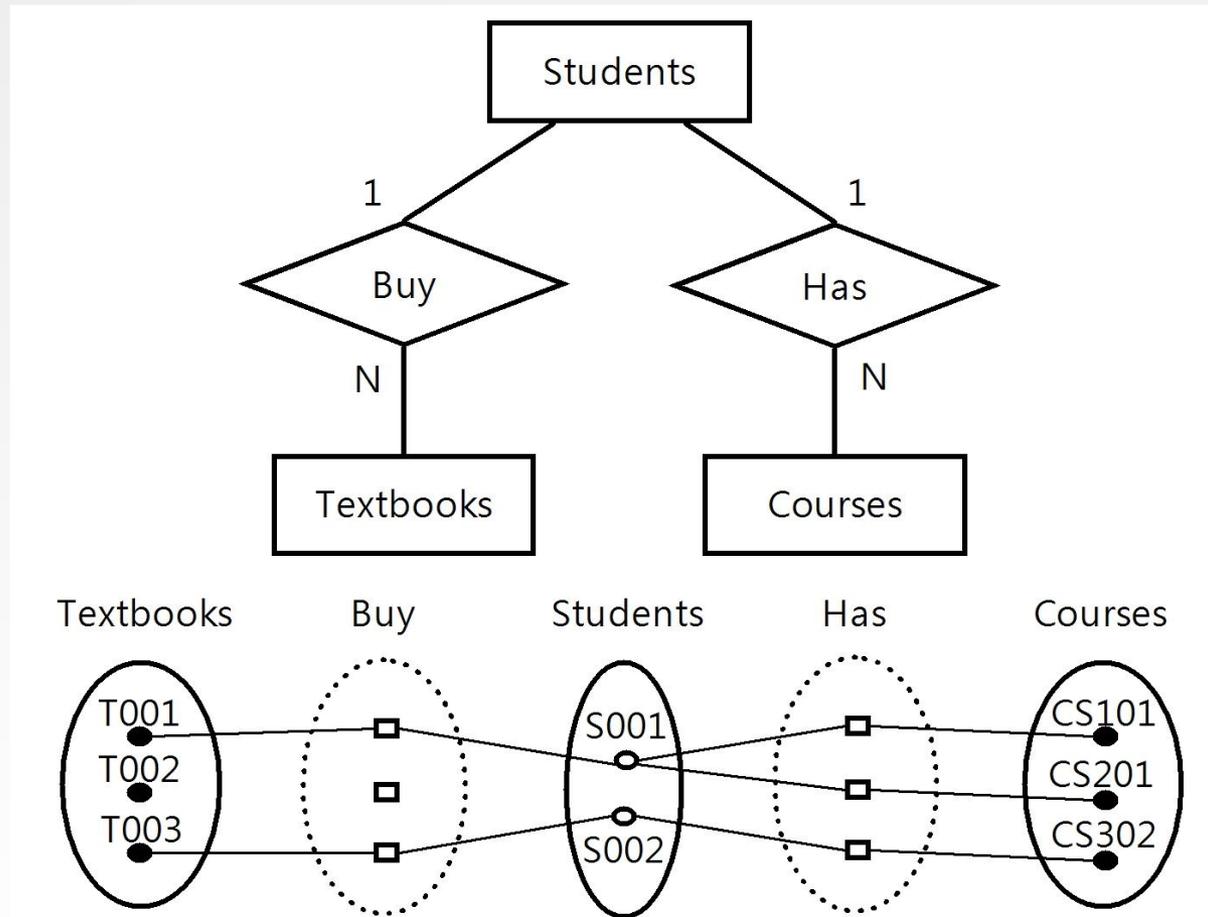
- 扇形陷阱發生的原因是誤解關聯的意義，關聯正確的解釋應該是以課程 **Courses** 建立關聯性的橋樑，如右圖所示：





4-4-2 斷層陷阱 – 說明

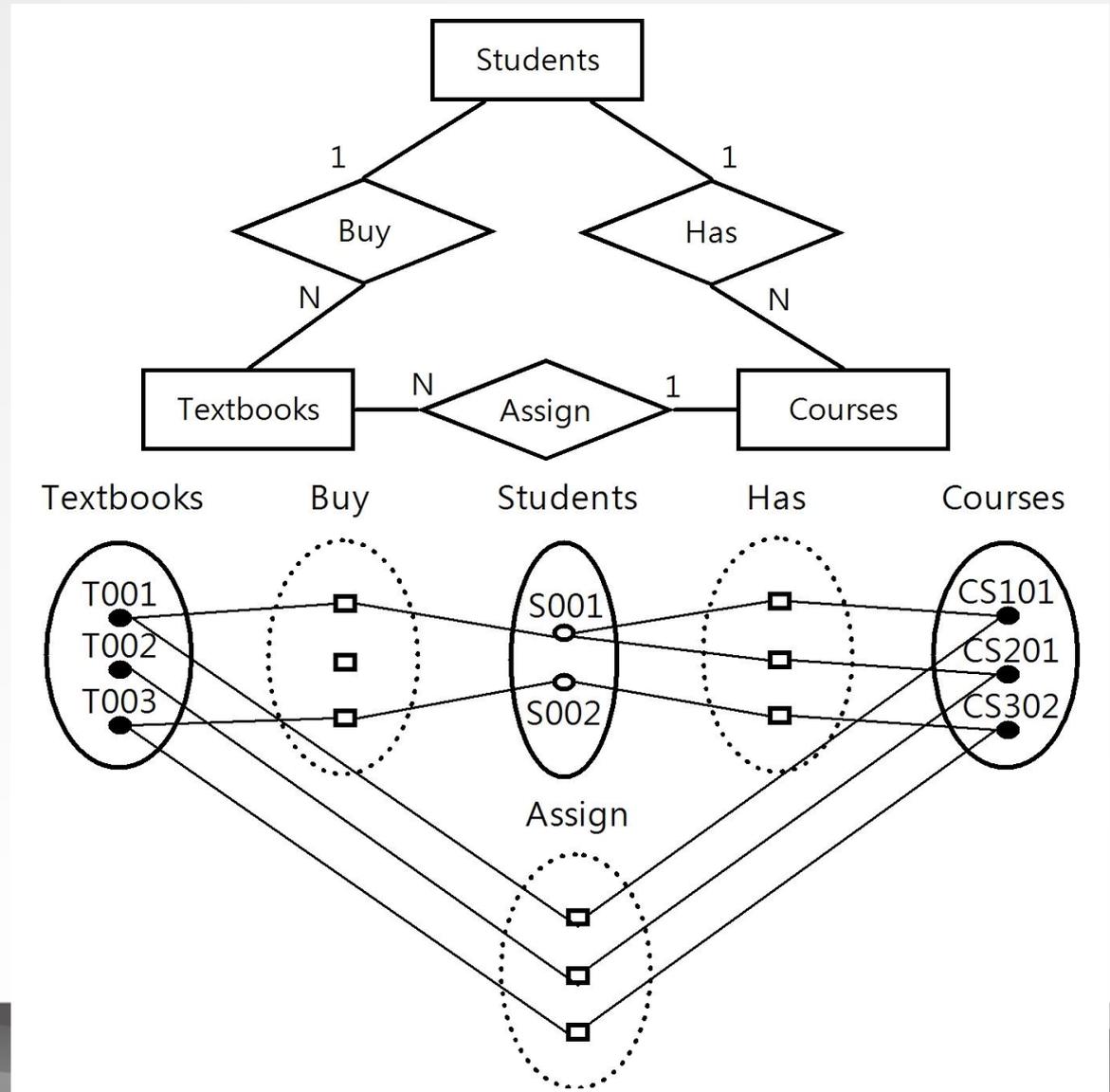
- 實體關聯圖的斷層陷阱是指實體間應該存在的關聯性根本不存在，在兩個實體間沒有辦法找到一條路徑來連接。例如：一位學生可以修多門課程，每一位學生購買多本教課書，如右圖所示：





4-4-2 斷層陷阱 – 解決方案

- 斷層陷阱 (Chasm Traps) 的解決方式是新增關聯型態，以此例是在 **Courses** 和 **Textbooks** 實體型態新增一對多的關聯型態 **Assign** 來解決此問題，如右圖所示：





4-5 將實體關聯圖轉換成關聯表綱要

- 4-5-1 將強實體型態轉換成關聯表
- 4-5-2 將關聯型態轉換成外來鍵
- 4-5-3 轉換多重關聯型態
- 4-5-4 多重值屬性轉換成關聯表
- 4-5-5 弱實體型態轉換成關聯表



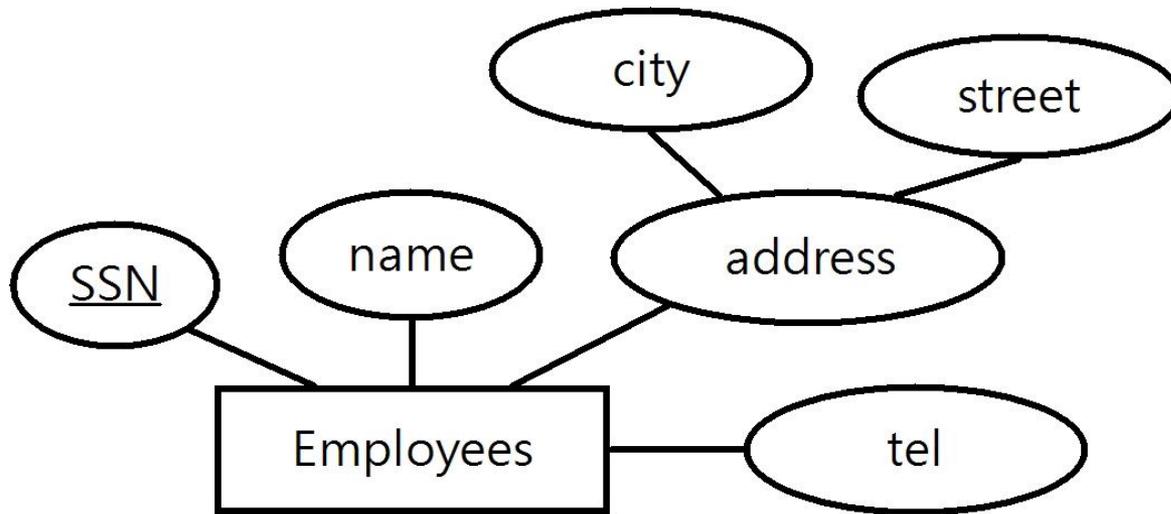


4-5-1 將強實體型態轉換成關聯表 – 規則

- 在實體關聯圖的強實體型態（即一般實體型態）是對應關聯表，將強實體型態轉換成關聯表綱要的規則，如下所示：
 - 建立新的關聯表綱要，其名稱是實體型態名稱。
 - 在關聯表綱要包含單元值屬性型態和複合屬性型態。
 - 關聯表綱要不包含多重值屬性型態、外來鍵和導出屬性型態。
 - 將鍵屬性（Key Attribute）指定為關聯表綱要的主鍵。



4-5-1 將強實體型態轉換成關聯表 - 圖例





4-5-2 將關聯型態轉換成外來鍵 – 說明

- 實體關聯圖的關聯型態可以轉換成關聯表綱要的外來鍵，在關聯表綱要新增參考其他實體型態的外來鍵，可以分為三種：
 - 一對一關聯型態
 - 一對多關聯型態
 - 多對多關聯型態

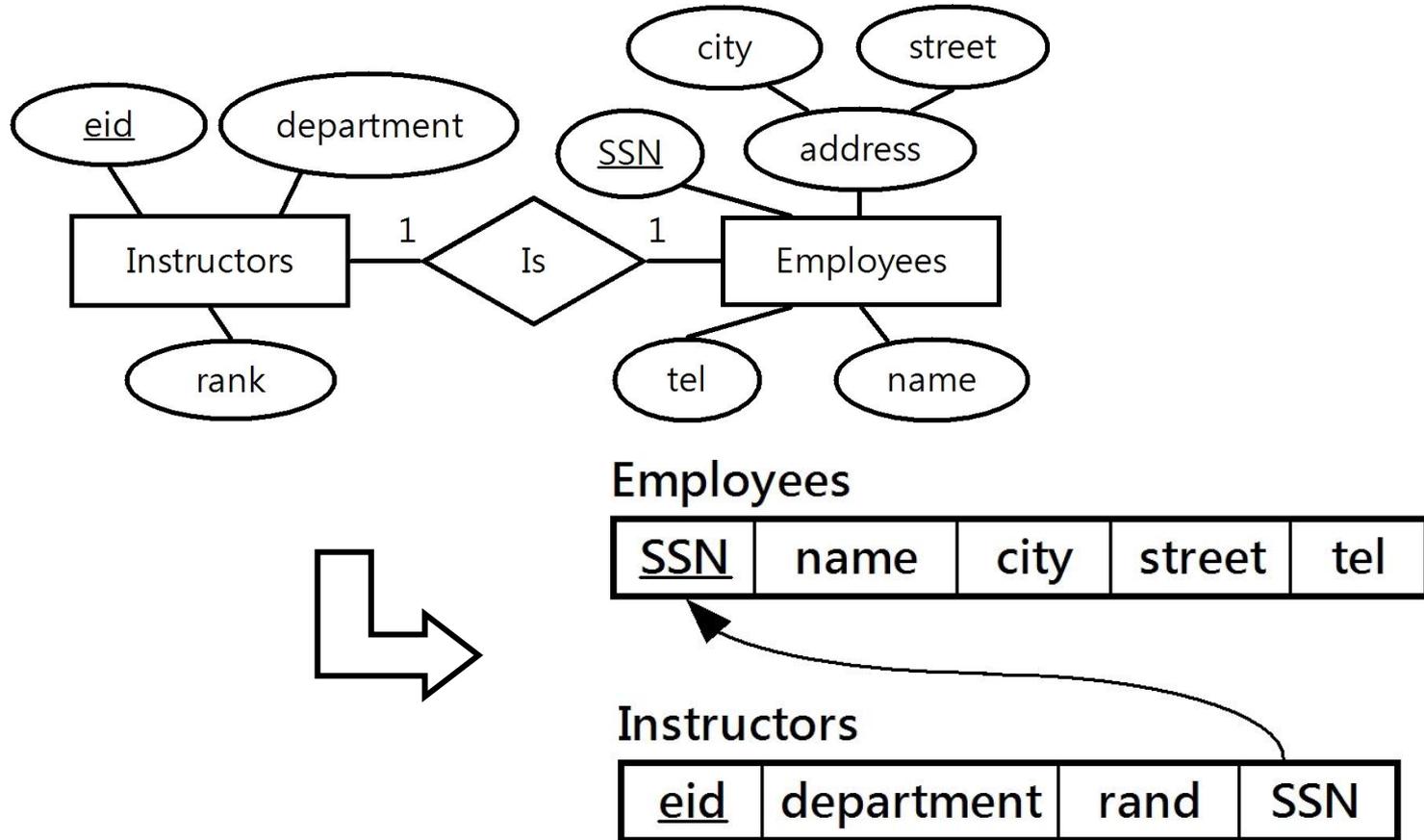


4-5-2 將關聯型態轉換成外來鍵 – 一對一關聯型態(規則)

- 一對一關聯型態轉換成關聯表綱要的規則，如下所示：
 - 在參與關聯性的關聯表綱要新增參考到另一個關聯表綱要的外來鍵（FK）。
 - 若關聯型態擁有單元值屬性，也一併加入新增外來鍵的關聯表綱要。



4-5-2 將關聯型態轉換成外來鍵 – 一對一關聯型態(圖例)



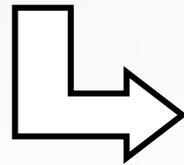
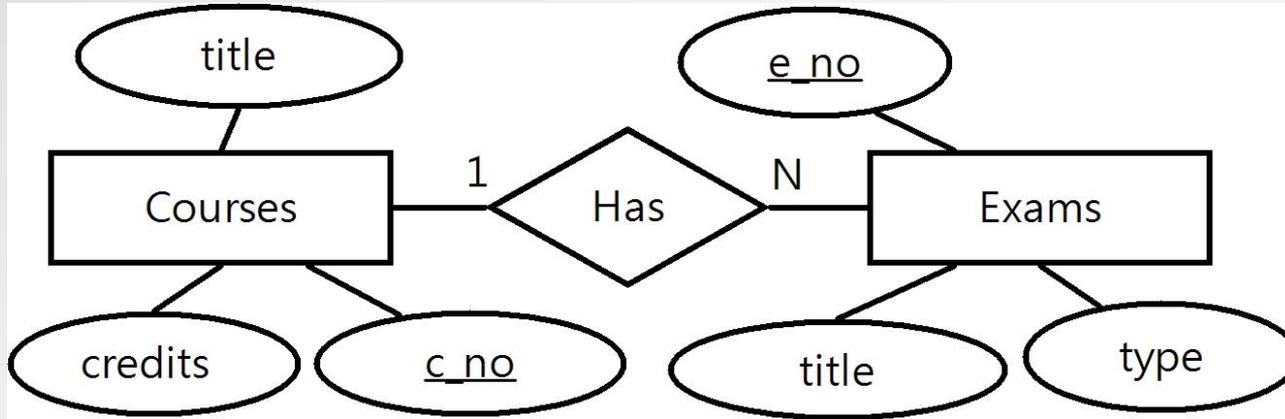


4-5-2 將關聯型態轉換成外來鍵 – 一對多關聯型態(規則)

- 一對多關聯型態轉換成關聯表綱要的規則，如下所示：
 - 在N端的關聯表綱要新增參考到1端關聯表綱要的外來鍵（FK）。
 - 若關聯型態擁有單元值屬性，也一併加入新增外來鍵的關聯表綱要。



4-5-2 將關聯型態轉換成外來鍵 – 一對多關聯型態(圖例)



Courses

<u>c_no</u>	title	credits
-------------	-------	---------

Exams

<u>e_no</u>	title	type	c_no
-------------	-------	------	------



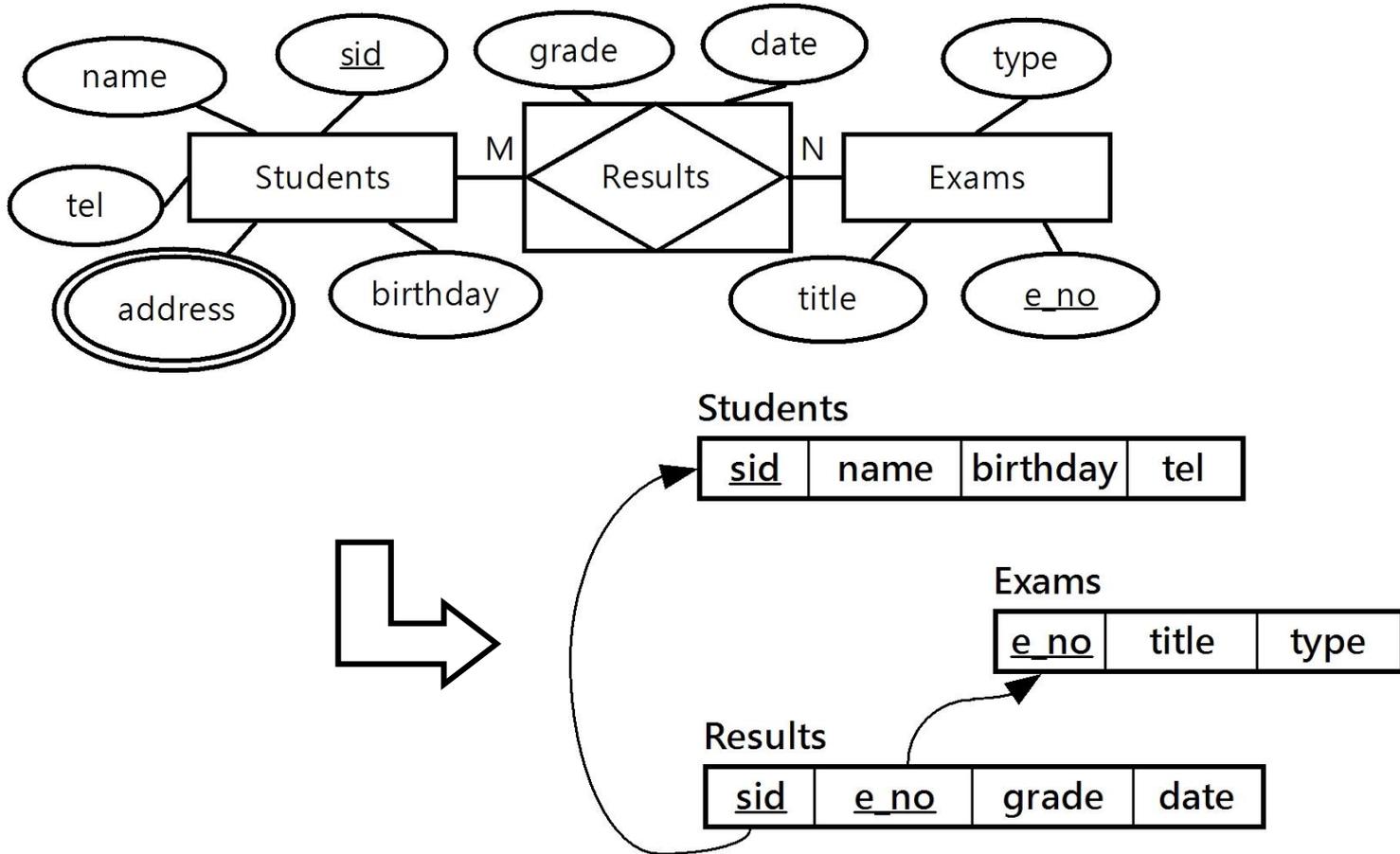


4-5-2 將關聯型態轉換成外來鍵 – 多對多關聯型態(規則)

- 多對多關聯型態轉換成關聯表綱要的規則，如下所示：
 - 將關聯型態建立成新的關聯表綱要，名稱為關聯型態名稱，在新關聯表綱要擁有兩個外來鍵（FK），分別參考關聯到的實體型態。
 - 若關聯型態擁有單元值屬性，一併加入新的關聯表綱要。
 - 關聯型態建立的關聯表綱要主鍵是兩個外來鍵的組合鍵，有時，可能需要新增幾個關聯型態的屬性作為主鍵。



4-5-2 將關聯型態轉換成外來鍵 - 多對多關聯型態(圖例)



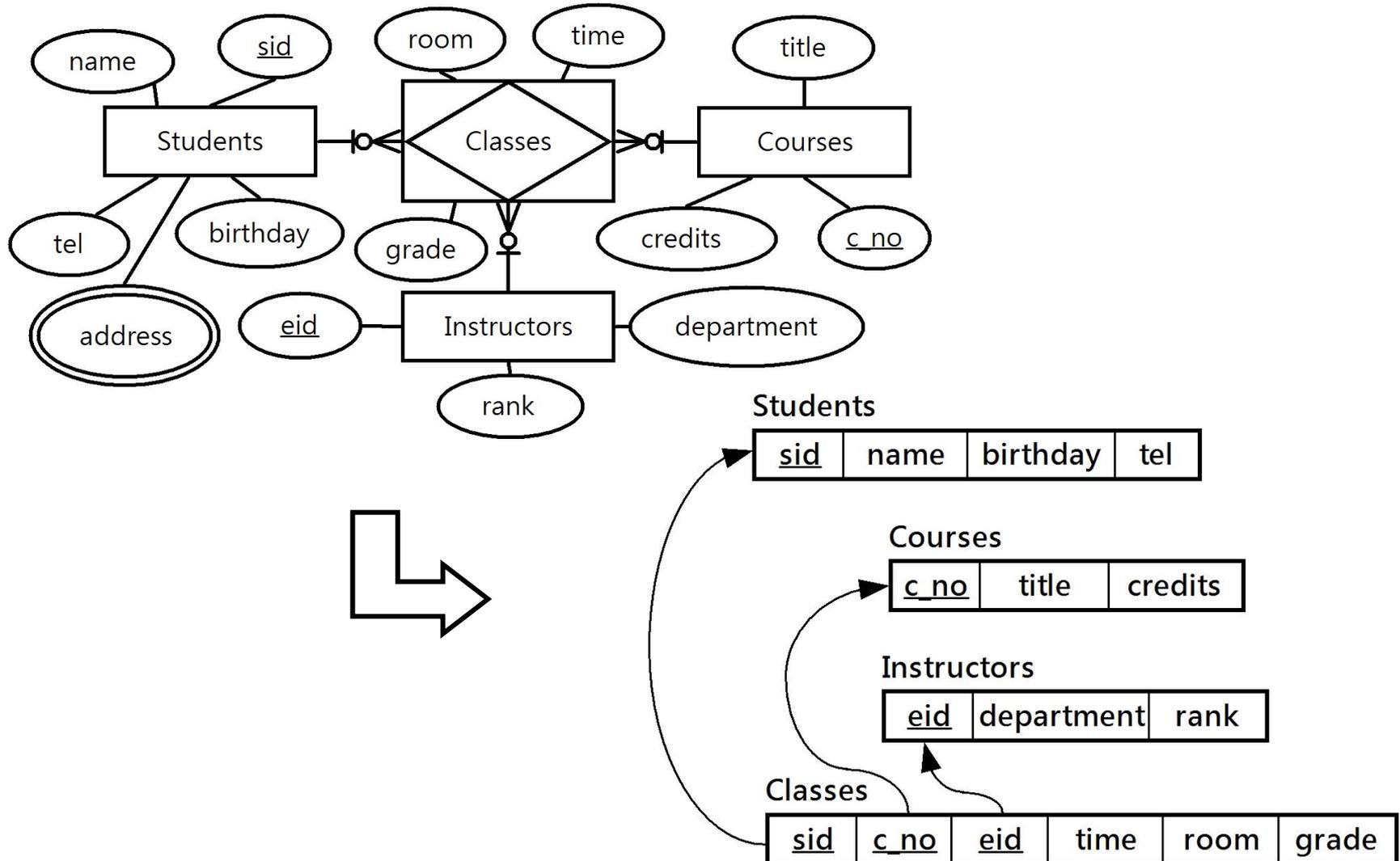


4-5-3 轉換多重關聯型態 – 規則

- 關聯型態可能擁有3個或更多實體型態間的關聯性，稱為「多重關聯型態」（Ternary Relationship Type），其轉換規則如下：
 - 將關聯型態建立成新的關聯表綱要，名稱是關聯型態的名稱，關聯表綱要擁有多個外來鍵（FK）分別參考關聯到的實體型態。
 - 若關聯型態擁有單元值屬性，也一併加入新建立的關聯表綱要。
 - 關聯型態建立的關聯表綱要主鍵通常是所有外來鍵的組合鍵，不過，可能需要新增幾個關聯型態的屬性，或部分外來鍵來作為主鍵。



4-5-3 轉換多重關聯型態 - 圖例



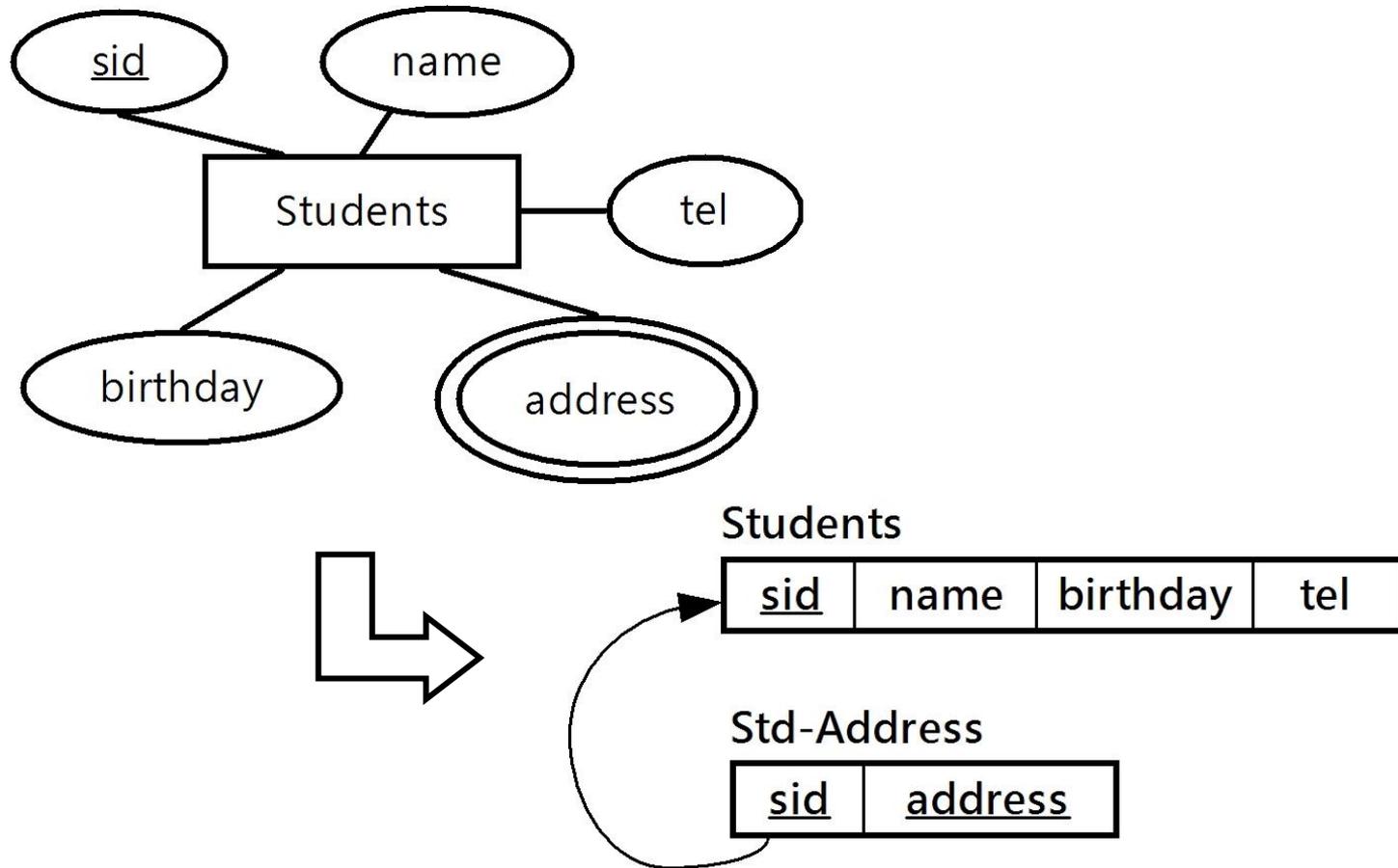


4-5-4 多重值屬性轉換成關聯表 – 規則

- 實體型態如果擁有多重值屬性，多重值屬性也需要轉換成關聯表綱要，其規則如下所示：
 - 建立新的關聯表綱要，名稱可以是屬性名稱或實體與屬性結合的名稱。
 - 在新關聯表綱要新增參考到實體型態主鍵的外來鍵。
 - 新關聯表綱要的主鍵是外來鍵加上多重值屬性，如果多重值屬性是複合屬性，可能需要加上其中一個屬性或是全部屬性。



4-5-4 多重值屬性轉換成關聯表 - 圖例



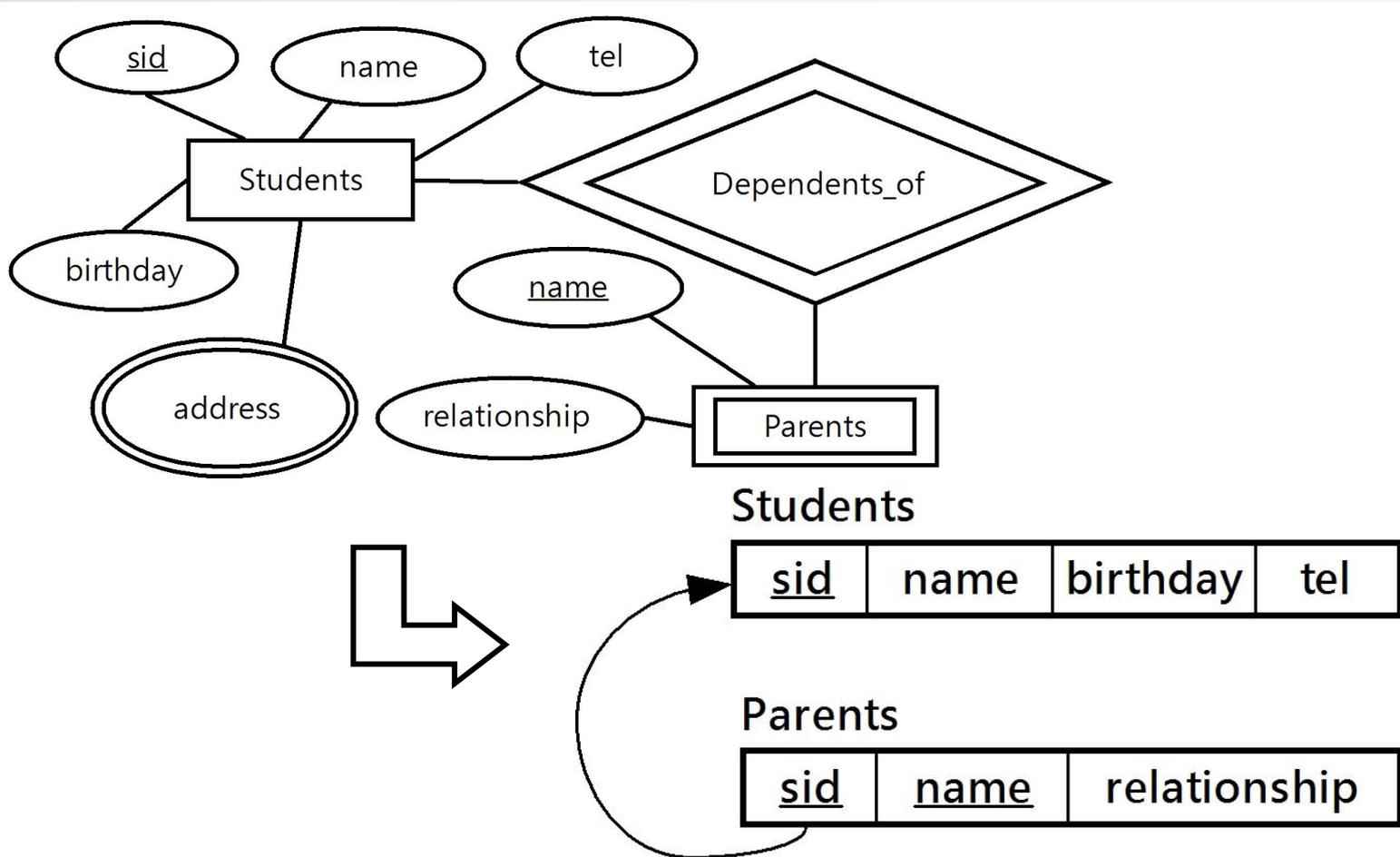


4-5-5 弱實體型態轉換成關聯表 – 規則

- 弱實體型態如同實體型態也是轉換成關聯表綱要，但弱實體型態一定擁有一個對應的識別實體型態，所以在轉換上稍有不同，其規則如下所示：
 - 建立新的關聯表綱要，其名稱為弱實體型態的名稱。
 - 新關聯表綱要包含單元值屬性型態。
 - 在新關聯表綱要新增識別實體型態的主鍵作為參考的外來鍵。
 - 將弱實體型態的「部分鍵」(Partial Key) 加上外來鍵，以便指定成新關聯表綱要的主鍵。

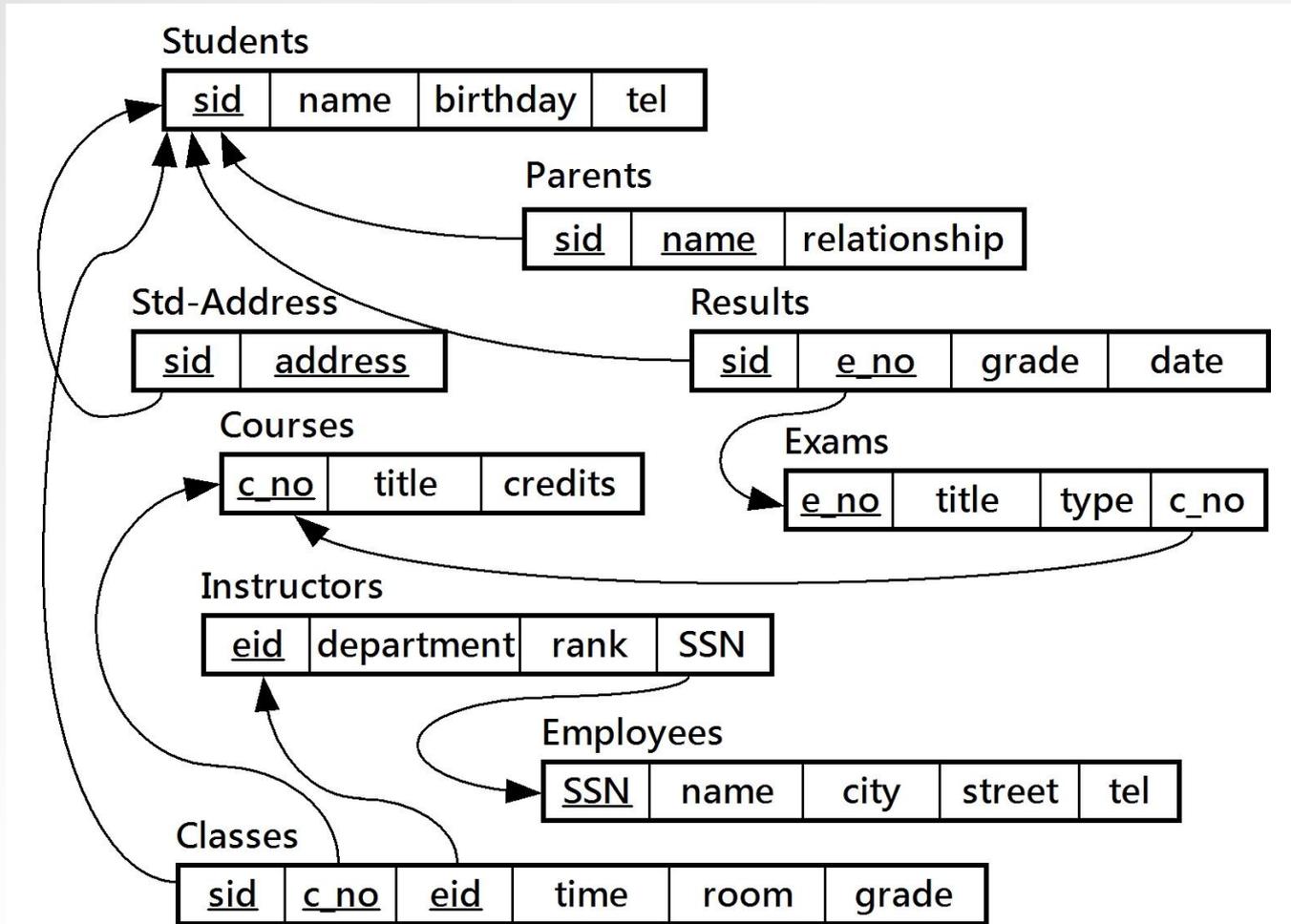


4-5-5 弱實體型態轉換成關聯表 - 圖例





4-6 關聯式資料庫的邏輯資料模型 – 外來鍵參考圖





4-6 關聯式資料庫的邏輯資料模型 – 關聯式資料庫綱要

- 學校School關聯式資料庫綱要，如下所示：

Students (sid, name, birthday, tel)

Parents (sid, name, relationship)

Std-Address (sid, address)

Results (sid, e_no, grade, date)

Exams (e_no, title, type, c_no)

Courses (c_no, title, credits)

Instructors (eid, department, rank, SSN)

Employees (SSN, name, city, street, tel)

Classes (eid, sid, c_no, time, room, grade)



End
