

行政院國家科學委員會專題研究計畫 成果報告

一個正規化整合式軟體系統程序文件之發展與研究

計畫類別：個別型計畫

計畫編號：NSC91-2626-E-164-001-

執行期間：91年08月01日至92年07月31日

執行單位：修平技術學院資訊管理系

計畫主持人：盧志偉

計畫參與人員：李乾麟、陳振庸

報告類型：精簡報告

處理方式：本計畫可公開查詢

中 華 民 國 92 年 12 月 9 日

行政院國家科學委員會補助專題研究計畫成果報告

一個正規化整合式 軟體系統程序文件之發展與研究

計畫類別： 個別型計畫 整合型計畫
計畫編號： NSC 91 - 2626 - E - 164 - 001 -
執行期間： 91年08月01日至92年07月31日

計畫主持人：盧志偉 副教授

共同主持人：朱正忠 教授、謝志明 副教授

計畫參與人員：李乾麟、陳振庸

執行單位：修平技術學院 資訊管理系

中 華 民 國 92 年 7 月 23 日

1. 簡介

首先感謝國科會惠予我們支援與協助，讓我們得在適當的時間與經費支援下進行這個名為『一個正規化整合式軟體系統程序文件之發展與研究』的前導型專案研究計畫。在這個專案計畫的目的下，我們提出並成功的架構了一個以 XML 為基礎的正規化整合式軟體系統文件模型(Formal Model for System Document Integration, FM/SDI)，將可用來協調、整合一個軟體系統與過程中，由一群不同團隊，於各個不同開發歷程按照任務需求，依循不同軟體標準所完成的軟體系統子模組。以這個整合模型為基礎，如蒙貴會繼續惠予支持，我們將得以於後續正式的應用性研究計畫中，應用這個整合式系統模型的延伸設計，透過 XML 的結構與連結描述特性，串連大部分業界實際於開發過程中設計出的抽象與實做模型，而達成一個正規化、整合性、應用性的系統文件規範，以協助解決長久以來存在軟體系統整合發展與維護作業上的問題。以下針對本前導型研究計畫的階段性成果與其相關說明，簡述如下。

由於系統開發與軟體系統的角色扮演，在現代的業界操作與應用上，其所佔有的重要比例與成敗因素愈發地明顯與提高，為了提昇軟體開發的效率與速度及軟體可維護性，許多的軟體標準(software standard)被建議實用於軟體開發流程以提昇軟體的品質，例如 UML(Unified Modeling Language)[3]及 XML (eXtensible Markup Modeling Language) [2] 提供了標準的模型表示法及資料格式，使得許多軟體模型的描述得以統一。而 Design Pattern[4]則對於特定的設計問題提供了可再使用的標準設計規範，使軟體設計有一個標準規範可循。Framework 則針對一特定領域的問題提供了一個標準系統架構，並提供了相對的一組程式碼提供程式開發人員客制化的空間，加快程式開發的速度。而就軟體開發的觀點而言，這些軟體標準確實提供了不少的貢獻。

軟體標準雖然提昇了軟體的開發效率，但個別的軟體標準往往只有支援軟體開發的部分過程 例如 UML 提供了需求分析及系統設計的軟體模型 而設計樣板(Design Pattern) 則只有提供設計階段的軟體模型。至於元件技術(Component-based technology)主要是強調在實作階段。

要建置一個高維護性的軟體必須包含軟體生命週期的每個階段包含需求分析、軟體設計、系統實作及軟體測試。單純的利用軟體標準以支援特定軟體開發階段往往是不夠的，而是必須要將所有階段的所有軟體模型整合起來以達成軟體模型間的同步性，才能真正的達成軟體可維護性的需求。要整合所有軟體模型則是等於要整合所有的軟體標準，但以目前而言，各個標準的軟體模型往往需要倚賴人為的思考從中穿針引線而得以完成，這樣的過程也容易發生人為的疏失而降低的軟體可維護性。因此，FM/SDI 的提出，一個正規、可串連在軟體生命週期所需的軟體標準的整合機制，便成為提高軟體品質可行的途徑。

2. 相關研究

物件導向技術及 UML

物件導向是軟體工程領域的一項重要的技術，物件導向的觀點在於將現實世界人、事、物等許多概念的靜態資料及動態行為表示成為物件，而這些概念之間的互動也會對映至物件之間的交互關係。一個物件往往被定義成為一個角色，擁有各自的特徵與行為，而所謂的商業模型(Business Model)就是將某個領域的資訊定義成個別的物件。利用物件導向技術可以使軟體的開發更快、成本更低、及得到較高的品質[5]。物件導向分析與設計(OOA/D) [1]基礎於物件導向的概念成為了軟體開發方法的主流。

Unified Modeling Language (UML)[3]是用來將一個軟體系統描述成視覺化軟體模型的標準表示法。UML 提供了許多既定的圖示用來描述軟體開發過程不同階段的抽象概念，例如使用案例圖(Use case diagram)、類別圖(Class diagram)、合作圖(Collaboration diagram)等等。利用這些圖示法可以讓使用者很容易的了解軟體的分析與設計過程。也因此 UML 已成為物件導向軟體開發的一項重要的技術。但 UML 的使用往往過於彈性而缺乏整合性，不同的 diagram 有不同的適用範圍，但不同的 diagram 卻缺乏一個串連的機制，在系統分析規劃時一旦需要跨不同 diagram，就有賴程式設計師以人工方式轉換、驗證，導致需求分析與系統設計的模式容易產生不完整及不一致性。

XML

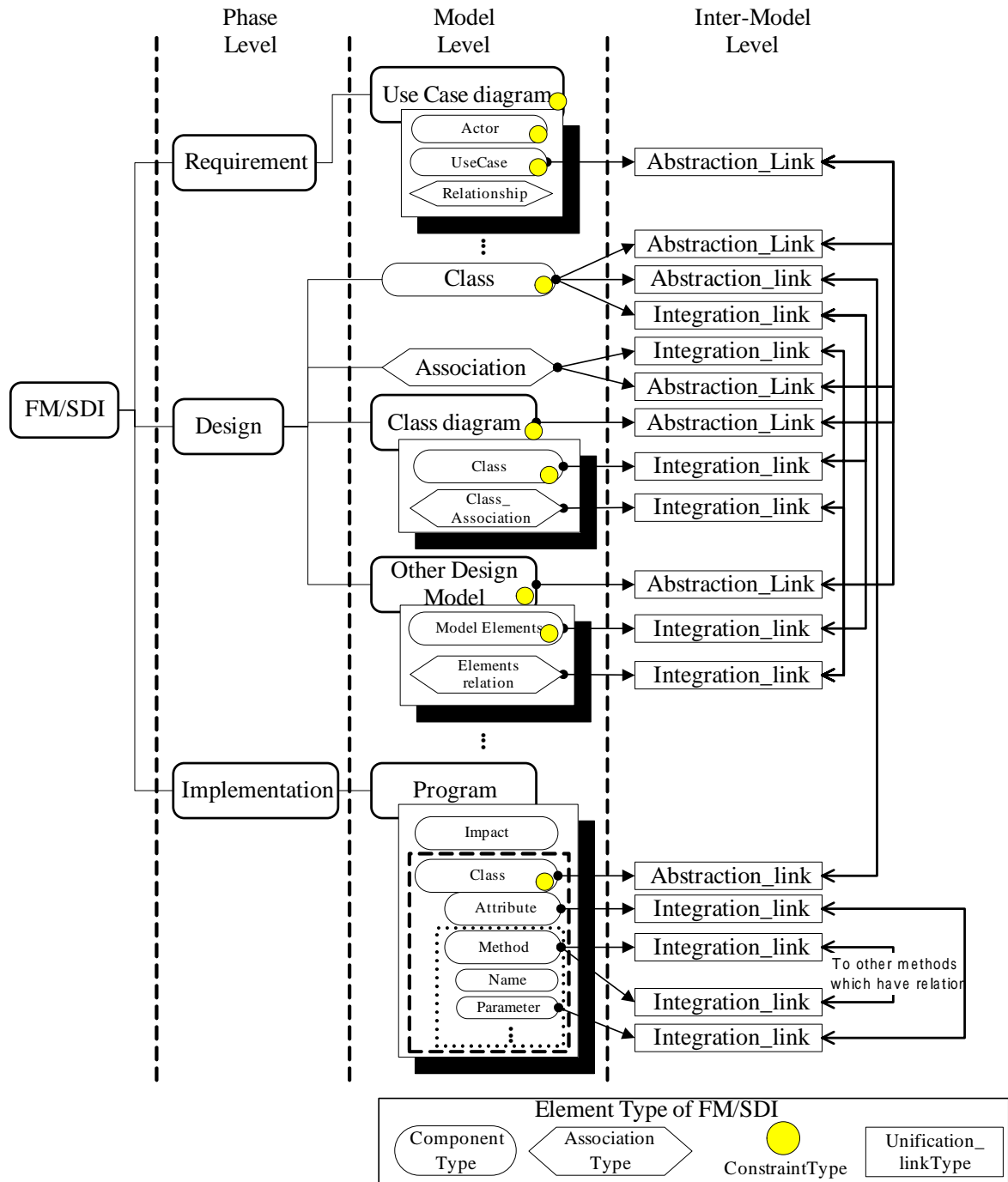
XML [6]是由 World Wide Web Consortium (W3C)所制定的標準，一種延伸式標記語言，具有擴展性(Extensibility)、結構性(Structure)、描述性(Description)、確認性(Validation)等特性。同時 XML 具有跨平台的功能，對於不同的作業系統、硬體設備、應用軟體、多元的輸入模式，開發者可以自行制定符合己身需求的標記(tag)，做結構性的描述，促使相同的一份文件呈現不同的規格，適用於不同的軟體，符合不同的設備、滿足多重的輸入方式。

XUM (XML-based Unified Model)

XUM[7]是一個利用 XML 為基礎，將軟體生命週期中，不同階段產出物加以串連、整合，XUM 印證了不同型態的軟體模型、程式碼能有效的加以整合成為具相同表示格式，卻又不失各軟體模型原本特質，並且透過 XML 的特性，可以呈現出多重不同樣式。本計畫主要就是以 XUM 的研究為基礎，進一步將系統文件加以正規化、整合成為一個一致的模式，並具有支援後續相關處理的表現模型。

3. 本計畫執行成果：FM/SDI

透過 XML-based 系統文件整合架構，我們實現將系統文件以一致的表示方法呈現，透過 XML Schema 的規範，我們訂定了一個正規化整合式軟體系統文件模型(FM/SDI)結構如下圖一所示：



圖一、正規化整合式軟體系統文件模型(FM/SDI)結構

在這個文件結構中，主要定義了四種基本元素(primitive)：ComponentType、AssociationType、ConstraintType 與 UnificationType。其中，ComponentType 是用來描述

原本系統文件中的塑模元件(Modeling elements)中必要的語義要素以及相關塑模所需的元件資訊；AssociationType 則是用來描述塑模元件間的關係及其相關資訊，ConstraintType 是用來描述塑模元素的條件限制如先決條件(precondition)、後續條件(postcondition)等，根據述語描述性質，可以區分成結構性(Structural) 述語、行為性(Behavioral) 述語以及一般性(General) 述語三類；UnificationType 則是用來連接不同模型觀點中，相同或衍生元素間的關係，其中根據連接對象不同，UificationType 衍生出兩種不同的連結：抽象連接(Abstraction Link)，用以連接不同階段中有關連的元件；以及整合連結(Integration link)，用以連接在同一設計階段中，不同塑模觀點中相對應的元件。這四種基本元素定義了整合式軟體系統程序文件結構所需的基本元素，在整合式軟體系統程序文件結構中所有的物件都是由這四種基本元素衍生而來。

表一為一般常用的軟體塑模元素對應到正規化整合式軟體系統文件模型中所呈現的語法架構：

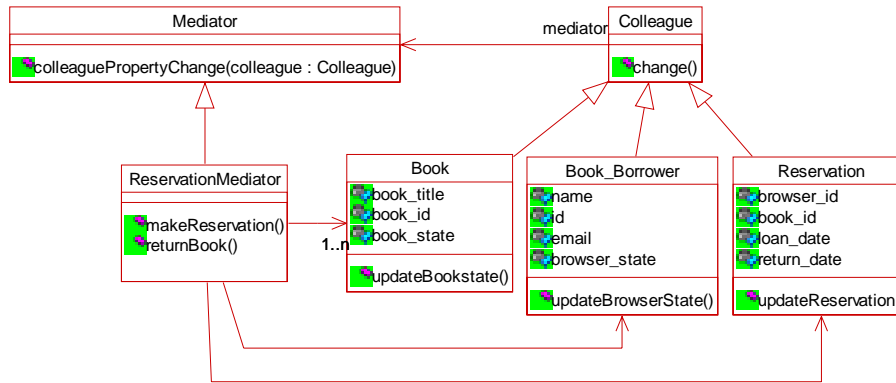
表一、塑模元素與 FM/SDI 對應關係

<i>Models /Standards</i>	<i>Model elements</i>	<i>FM/SDI element Representations</i>
UML Use Case diagram	Actor	<Actor>
	Use Case	<Usecase>
	Association	<association_type>association </association_type>
	Generalization	<association_type>generalization </association_type>
	Extend	<association_type> extend </association_type>
	Include	<association_type>include </association_type>
UML Class, Collaboration, Sequence diagram	Class	<Class>
	Attribute	<Attribute>
	Operation	<Method>
	Interface	<Interface>
	Parameter	<Parameter>
	Association	<association_type>association </association_type>
	Composition	<association_type>composition </association_type>
	Generalization	<association_type>generalization </association_type>
	Dependency	</association_type>dependency </association_type>
	Message	<Message>

其詳細的語法架構的 Schema 如附錄所示。

3.2 範例

舉例來說，下圖二是一個圖書館借書系統部分的類別圖，轉換成正規化整合式軟體系統文件模型的話，其結構與內容如圖三 a、圖三 b、圖三 c 所示。



圖二、圖書館借書系統類別圖(部分)

```

<design_Class>
  <name>Mediator</name>
  <Integration_link xlink:label="D_Mediator" xlink:title="Class of
Mediator" />
  <Abstraction_link xlink:from="D_Mediator"
xlink:to="S_Mediator" />
  <method attribute="public"> colleaguePropertyChange
(colleague:Colleague)" </method>
</design_Class>
... ..
</design_Class>

```

圖三 a、類別(classes)的整合式軟體系統程序文件表示

```

<Association>
  <ass_from>Mediator</ass_from>
  <ass_to multi="1..n"> ReservationMediator</ass_to>
  <Intgration_link
xlink:label="Mediator_ReservationMediator"
xlink:title="Association: Mediator_ReservationMediator" />
</Association>
... ..
</Association>

```

圖三 b、關連(associations)的正規化整合式軟體系統文件模型表示

```

<Class_Diagram>
  <class>Mediator</class>
    <Integration_link xlink:href="D_Mediator"/>
  <class>ReservationMediator</class>
  ... ..
  <class>Book</class>
  ... ..
  <Class_ass>
    <ass_from>Mediator</ass_from>
    <ass_to multi="1"> Reservation Mediator</ass_to>
    <association_type>generalization</association_type>
    <Integration_link xlink:title="Mediator_ReservationMediator"
      xlink:label=" Association of Mediator_ ReservationMediator"
      xlink:href="Mediator_ReservationMediator"
      xlink:from="D_Mediator " xlink:to="D_ReservationMediator" />
  <class_ass>
    <ass_from>ReservationMediator</ass_from>
    <ass_to multu="0..n">Reservation</ass_to>
    <association_type>dependency</association_type>
  ... ..
</Class_Diagram>

```

圖三 c、類別圖(class diagram)的正規化整合式軟體系統文件模型表示

圖三中，三個子圖分別代表類別 class、類別間的關連及類別圖(class diagram)相關描述資訊以 FM/SDI 的表示所呈現的內容。將類別資訊、類別關連與塑模圖分離之主要目的在於相同之類別資訊可能會在多個不同的塑模圖之中出現，一方面不宜將類別資訊與塑模圖合併，再者這樣也可以達到資訊共用及後續的類別整合追蹤。在整個正規化整合式軟體系統文件模型存在許多這類的元件資訊，這些元件資訊會在許多不同的塑模圖中出現，這些可能具有不同名稱或特質但實際上是相同物件的元素，我們就利用 <Integration_link> 與 <Abstraction_link> 將這些元素加以串連，以便後續的處理中，當程式設計師對於塑模圖中做修改時，可以輕易的將所有可能被影響的物件、塑模圖乃至程式碼表列出來，提醒程式設計師。

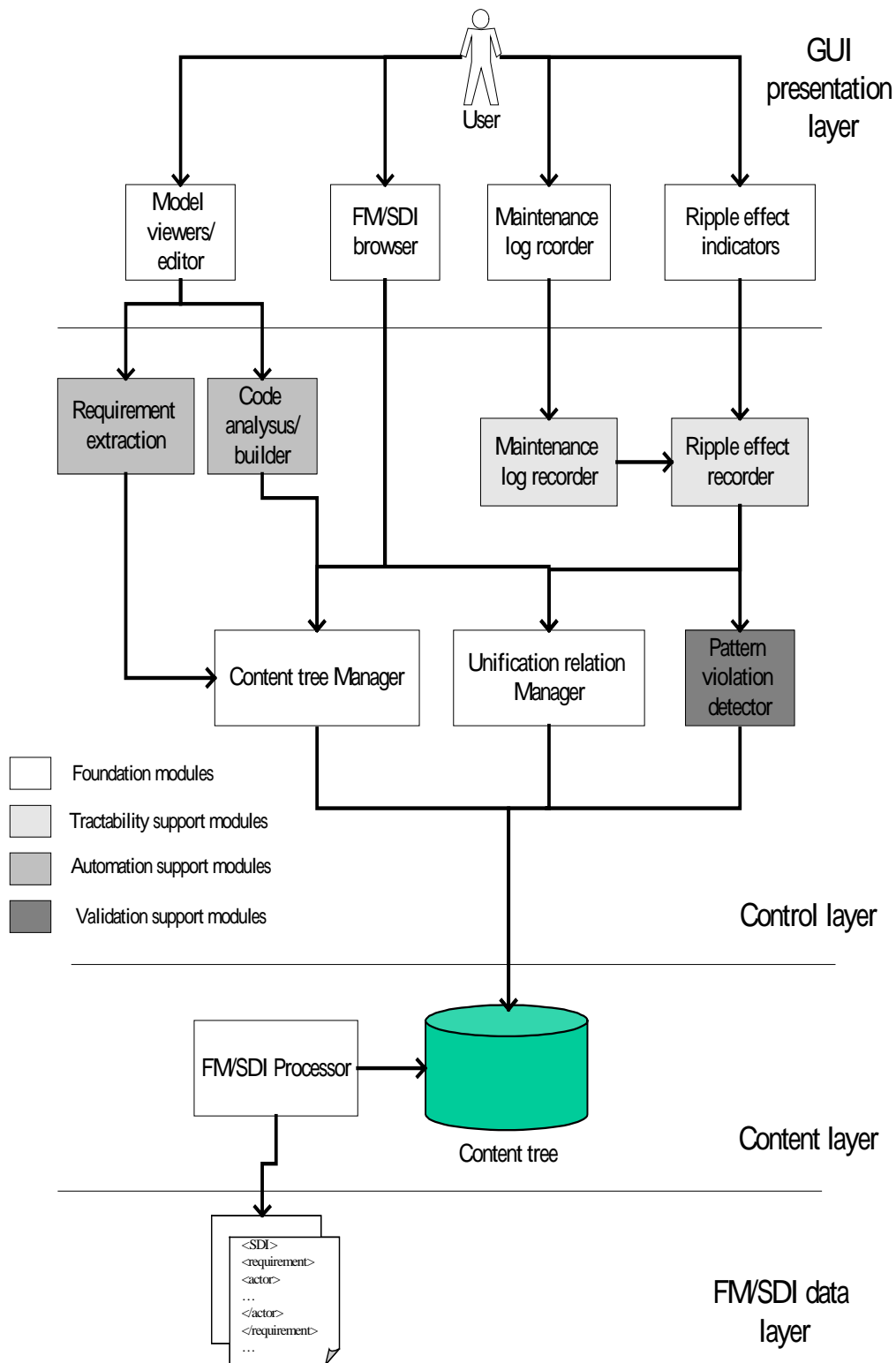
4. 後續研究計畫之工作內容

基於以上這個具有正面意義的階段成果，我們很有信心能在後續的研究中，將 FM/SDI 的模型實際導入，並串連大部分業界於開發過程中所採用與設計出的抽象及實做模型，而達成一個正規化、整合性、應用性的系統文件規範，以協助解決長久以來存在軟體系統整合發展與維護作業上的問題。為了達成這個目的，以下我們列舉幾項接下來的具體研究目標與工作項目：

4.1 整合式軟體塑模發展環境

透過整合式軟體系統程序文件表示模型，已經可以將軟體生命週期中所需要用到的需求規格、軟體塑模文件、程式碼適當地結合，然而對程式設計師而言，若沒有適合輔助工具，要撰寫 XML 格式的系統程序文件，也是相當不容易。但是透過現有程式語言處理 XML 的函數套件，如 Java 的 JAXB 等，使得實作一個可以視覺化撰寫系統規格、系統塑模、程式碼開發、並儲存符合整合式軟體系統程序文件的整合開發環境可以被實現。在本計畫的後續研究中，將實作此一整合開發環境，同時作為本計畫中其他相關研究子題的實驗平台。

本整合式軟體塑模發展環境預計將以 JAVA 為開發語言，並依循 N-tier 架構，預計架構如下圖四。其主要可分為四個階層，分別為使用者介面層(GUI presentation layer)、控制層(control modules)、內容層(content layer)及資料儲存層(XUM data layer)。



圖四、FM/SDI 整合式軟體塑模發展環境系統架構

使用者介面層提供了圖形化之介面如 FM/SDI browser 提供使用者瀏覽所有模型及其 unification link。Model viewers 提供 UML 標準如使用案例圖(Use case diagram)、類別圖(Class diagram) 等模型的顯示。

控制層是本系統的核心部分，此部份可分為以下模組：

- 基礎功能模組(Foundation modules)

基礎功能模組提供了整合模型基本的功能，包含整合模型內容管理(Content tree manager)、統一關聯管理(Unification relation manager)等。

- 追蹤性支援模組(Tractability support modules)

追蹤性支援模組主要提供了模型的可追蹤性，如漣漪效應的追蹤(Ripple effect tracer)。在使用者修改軟體模型的過程中，系統將隨時紀錄這些修改的模型，並追蹤其直接或間接關聯的模型，輔助使用者作軟體的維護。

- 自動化支援模組(Automation support modules)

自動化支援模組提供了軟體開發的自動化機制包含了支援需求分析至系統設計自動化的需求擷取模組(Requirement extractor)與系統設計至實作自動化的程式碼分析/建立模組(Code analysis/building modules)。

- 驗證支援模組(Validation support modules)

驗證支援模組主要是提供了模型驗證的機制，FM/SDI 裡預計整合常見的軟體規範，如整合了 UML 與設計樣版，由於許多軟體規範往往具有架構或屬性上的條件限制，例如設計樣版是一種規範性的軟體設計架構，因此當 UML 整合了設計樣版之後，系統設計人員就不能夠在任意的使用 UML 建置設計模型，而必須遵循於設計樣版的規範。因此本系統提供了驗證的機制。在使用者修改設計模型的過程中驗證使用者是否違反了設計樣版等特殊軟體規範的限制。

內容層以物件樹狀結構的方式對映至 XML 中的模型資訊，系統的其他模組只要透過管理此內容樹就可以達成間接存取 XML 的機制，以免除複雜的文字解析動作。

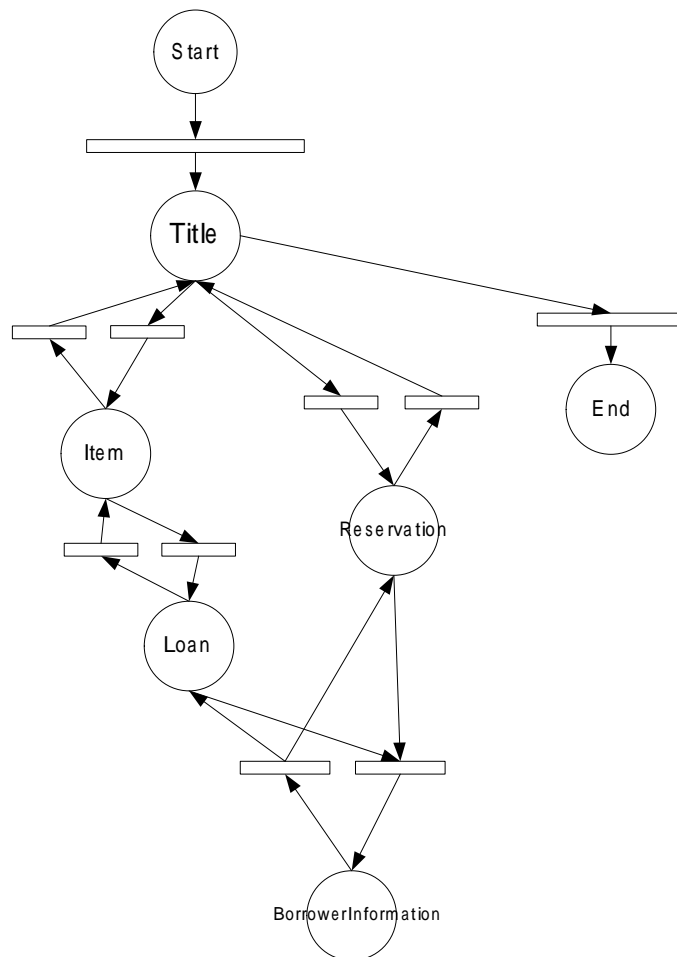
資料儲存層以 XML 的格式將統一模型儲存於檔案中。

4.2 軟體文件驗證

大部分的塑模方法都是採用視覺化圖形的表示方式，然而有些細部的條件，卻無法完全用視覺化方式來表達，而這些條件往往表達了塑模文件中，一些行為上、結構上的特徵，這些特徵在模型間檢查時，卻可以扮演重要角色，常用的塑模方法中，其實也都有這些加註條件的機制，然而一方面現有的塑模輔助工具，並沒有特別強調這方面的特性，而且一般的塑模方法有其涵蓋範圍的限制，並無法兼顧軟體開發過程中不同的需求，因此這方面的塑模限制語言並未被程式設計師善加運用。在先前的整合式軟體系統程序文件表示模型中，我們保留了加入條件限制的機制(即前一章所提之 ConstraintType 衍生之條件元素)，以便徹底運用各種塑模方式的特性，例如 UML 中的 OCL 可以將原本視覺化的塑模圖形加入先決條件、後置條件等限制，即可套用適當的一致性檢查機

制，如派翠網路(Petri Net)等，可進行模型內、模型間的一致檢查，加上合式軟體系統程序文件表示模型本身已經容納各種常用塑模方式的表示結構，使得以 FM/SDI 為基礎的塑模一致性檢查，可以跨越塑模方法的限制，檢查出不同塑模模型間不一致的現象，減少程式設計師因使用人工驗證可能造成遺漏。

圖五是模型一致性檢查派翠網路圖，當程式設計師需要驗證或檢查模型中，或不同模型間是否一致時，系統會從 FM/SDI 中取得相關模型的資訊及條件轉換成派翠網路中所需的 place、transaction，以及每個 place 的先決條件、後續條件與限制，而每條路徑也會有其條件限制。利用派翠網路的特性，當 token 從 Start 部分 fire 之後，如果所有的 place 其條件都符合，最後 token 將可以抵達 End 處，若有一個或數個 place 有條件不符，token 無法達到 End，那就在表此模型中的元素不一致，此時可檢查 token 最後卡在何處，就可以知道哪個元素可能與其他元素間無法匹配。



圖五、模型一致性檢查派翠網路圖

4.3 後續計畫之分年進度與目標規劃

第一年

第一年目標主要在將先導計畫中所完成的整合式軟體系統程序文件表示模型整合開發環境加以實作，以便提供後續相關研究的實習發展平台，因此本年度計畫主要在於對 Java 語言與 XML 相關技術的研究與 FM/SDI 之間的結合，並實作其系統雛形。

此外，在軟體文件驗證方面，則是著重於不同軟體塑模方式中限制語言資料的收集、整理、分類，以便後續研究之進行。

第二年

第二年的進度在於主要在於軟體文件驗證方面的研究，包含軟體文件驗證的理論，正規化架構與表示式及其相關演算法的建立，並與第一年中所發展的整合開發環境相結合。

第三年

第三年的進度主要在驗證整合式軟體系統程序文件表示模型運用於軟體開發過程，包含軟體整合開發環境及軟體文件驗證等功能，並希望應用於技職教育中，培養學生完整的軟體開發概念，同時視學生使用狀況與系統運作情形，對於整個系統理論架構加以驗證、調整。

4.4 總結

軟體工業將是我國在未來國際社會競爭力不可或缺的一環，如何提高軟體品質與產能，將是我國能否在世界軟體工業中佔有舉足輕重地位的重要因素。本系列性的研究計畫若能繼續推展延伸，其成果將可對於目前軟體開發過程中軟體品質不穩定、高成本、高人力等亟待突破的瓶頸，將提供有效的解決之道，並可在世界軟體技術中，取得相當正面的突破。

目前國內軟體產業中的人力，多是過去十年所培養出來的，由於這些人員在學校中尚未學習到這些新的軟體技術規範，進入業界後也因為缺乏適合的輔助工具，使得這些軟體產業的中堅對於引入新軟體規範卻步不前，而技職教育中，資訊人員將是未來國內軟體產業中主要的開發人力來源，如何在學校中，讓學生學習到新的軟體規範，讓他們樂於在實務中依循這些軟體規範，將是未來國內軟體產業是否能進入到新一階段的關鍵。

本計畫希望將過去在軟體開發過程中斷斷續續的軟體塑模方法以及軟體規範做一整合，並開發出讓使用者容易使用的整合開發環境，同時透過整合在 FM/SDI 中的資訊，

對於軟體文件一致性加以檢查，提高系統文件的品質，進一步提升軟體品質與產能。

希望貴會能繼續惠予支持，讓整個計畫成果能更加完整而得以將理論落實至實際執行面，以期獲取實際之 SE 環境與功效。

參考資料

- [1] Booch, G. (1994). *Object-oriented analysis and design with applications* 2nd ed. Redwood City, Calif. : Benjamin/Cummings Pub. Co., 3-25.
- [2] Deitel, H., Deitel, P., Nieto, T., Lin, T., & Sadhu, P. (2001). *XML how to program*. Upper Saddle River, NJ : Prentice Hall.
- [3] Object Management Group. (2001, August). *OMG unified modeling language specification*. Version 1.4, Retrieved July 16, 2001 from http://www.omg.org/technology/documents/recent/omg_modeling.htm
- [4] Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: elements of reusable object-oriented software*. Reading, MA.: Addison-Wesley.
- [5] Wirfs-Brock, R. J., & Johnson, R. E. (1990). Surveying Current Research in Object-Oriented Design. *Communications of the ACM*, 33(9),105-124.
- [6] World Wide Web Consortium (W3C), Extensible Markup Language (XML), Retrieved July 16, 2003 from <http://www.w3.org/XML/>
- [7] Lu C.W., Chu C.W., Chang C.H., Lian W.D., Yang D.L. (2003). Integrating Divers Paradigms in Evolution and Maintenance by an XML-based Unified Model , *Journal of Software Maintenance and Evolution*

附錄：FM/SDI Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSPY v5 rel. 4 U (http://www.xmlspy.com) by cwlu (hit) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:complexType name="componentType">
    <xs:sequence>
      <xs:element name="ID" type="xs:ID"/>
      <xs:element name="Name" type="xs:string"/>
      <xs:element name="integration_link" type="unificationType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="abstraction_link" type="unificationType" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="associationType">
    <xs:sequence>
      <xs:element name="ass_from" type="xs:IDREF"/>
      <xs:element name="ass_to" type="xs:IDREF"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="conditionType">
    <xs:sequence>
      <xs:element name="precondition" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="type">
            <xs:simpleType>
              <xs:restriction base="xs:NMTOKENS">
                <xs:enumeration value="general"/>
                <xs:enumeration value="structural"/>
                <xs:enumeration value="behavioral"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element name="predicate" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="type">
            <xs:simpleType>
              <xs:restriction base="xs:NMTOKENS">
                <xs:enumeration value="general"/>
                <xs:enumeration value="structural"/>
                <xs:enumeration value="behavioral"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
      <xs:element name="postcondition" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:attribute name="type">
            <xs:simpleType>
              <xs:restriction base="xs:NMTOKENS">
                <xs:enumeration value="general"/>
                <xs:enumeration value="structural"/>
                <xs:enumeration value="behavioral"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:attribute>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>

```

```

    </xs:element>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="unificationType">
  <xs:sequence>
    <xs:element name="link_from" type="xs:anyURI"/>
    <xs:element name="link_to" type="xs:anyURI"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="declarationType">
  <xs:choice>
    <xs:element name="array" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="array_type"/>
          <xs:element name="identifier" type="xs:string"/>
          <xs:element name="array_size" maxOccurs="unbounded"/>
          <xs:element name="initialize" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="variable" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="variable_type"/>
          <xs:element name="identifier" type="xs:string"/>
          <xs:element name="initialize" minOccurs="0"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:choice>
</xs:complexType>
<xs:complexType name="statementType">
  <xs:choice maxOccurs="unbounded">
    <xs:element name="assignment" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="ass_target" type="xs:string"/>
          <xs:element name="ass_source"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="methodinvocation" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element name="method"/>
          <xs:element name="parameter" minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="control" minOccurs="0" maxOccurs="unbounded">
      <xs:complexType>
        <xs:choice maxOccurs="unbounded">
          <xs:element name="chosed" minOccurs="0">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="condition"/>
                <xs:element name="then">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element ref="statement"/>
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:choice>
</xs:complexType>

```



```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="else" minOccurs="0">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="statement"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="switch" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="condition"/>
      <xs:element name="case" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="expression"/>
            <xs:element ref="statement"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="default">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="statement"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="while" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="condition"/>
      <xs:element name="do">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="statement"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="for" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="initialize" minOccurs="0"/>
      <xs:element name="test" type="xs:anyType" minOccurs="0"/>
      <xs:element name="modify" minOccurs="0"/>
      <xs:element ref="statement"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="do" minOccurs="0">
  <xs:complexType>

```

```

        <xs:sequence>
            <xs:element ref="statement"/>
            <xs:element name="while">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="condition"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="comment" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="break" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:attribute name="identifier" type="xs:string"/>
    </xs:complexType>
</xs:element>
<xs:element name="continue" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:attribute name="identifier" type="xs:string"/>
    </xs:complexType>
</xs:element>
<xs:element name="return" minOccurs="0">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="expression" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="throw" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="expression" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="synchronized" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="expression"/>
            <xs:element ref="statement"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="try" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:sequence>
            <xs:element name="catches" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element name="catch"/>
                        <xs:element ref="statement"/>
                        <xs:element name="formal_parameter" type="formal_parameterType"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="finally" minOccurs="0">

```

```

    <xs:complexType>
      <xs:choice>
        <xs:element name="catches" minOccurs="0">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="catch"/>
              <xs:element ref="statement"/>
              <xs:element name="formal_parameter" type="formal_parameterType"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element ref="statement"/>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <xs:sequence>
    <xs:complexType>
      <xs:element>
        <xs:element name="expression" minOccurs="0" maxOccurs="unbounded"/>
      </xs:choice>
    </xs:complexType>
  </xs:complexType>
  <xs:complexType name="formal_parameterType">
    <xs:sequence>
      <xs:element name="para_type"/>
      <xs:element name="identifier" type="xs:string"/>
    </xs:sequence>
    <xs:attribute name="final" type="xs:boolean" use="optional"/>
  </xs:complexType>
  <xs:element name="class">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="name" type="xs:string"/>
        <xs:element name="id" type="xs:IDREF"/>
        <xs:element name="condition" type="conditionType"/>
        <xs:element name="abstract_link" type="unificationType"/>
        <xs:element name="integration_link" type="unificationType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Actor">
    <xs:complexType>
      <xs:complexContent>
        <xs:restriction base="componentType">
          <xs:sequence>
            <xs:element name="ID" type="xs:ID"/>
            <xs:element name="Name" type="xs:string"/>
            <xs:element name="integration_link" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:complexContent>
                  <xs:restriction base="unificationType">
                    <xs:sequence>
                      <xs:element name="link_from" type="xs:anyURI"/>
                      <xs:element name="link_to" type="xs:anyURI"/>
                    </xs:sequence>
                  </xs:restriction>
                </xs:complexContent>
              </xs:complexType>
            </xs:element>
            <xs:element name="abstraction_link" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:complexContent>

```

```

        <xs:restriction base="unificationType">
            <xs:sequence>
                <xs:element name="link_from" type="xs:anyURI"/>
                <xs:element name="link_to" type="xs:anyURI"/>
            </xs:sequence>
        </xs:restriction>
    </xs:complexContent>
</xs:complexType>
</xs:element>
<xs:choice>
    <xs:element name="responsibility" type="xs:string"/>
    <xs:element name="desc" type="xs:string"/>
    <xs:element name="condition">
        <xs:complexType>
            <xs:complexContent>
                <xs:extension base="conditionType"/>
            </xs:complexContent>
        </xs:complexType>
    </xs:element>
</xs:choice>
</xs:sequence>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="UseCase">
    <xs:complexType>
        <xs:complexContent>
            <xs:restriction base="componentType">
                <xs:sequence>
                    <xs:element name="ID" type="xs:ID"/>
                    <xs:element name="Name" type="xs:string"/>
                    <xs:element name="integration_link" minOccurs="0" maxOccurs="unbounded">
                        <xs:complexType>
                            <xs:complexContent>
                                <xs:restriction base="unificationType">
                                    <xs:sequence>
                                        <xs:element name="link_from" type="xs:anyURI"/>
                                        <xs:element name="link_to" type="xs:anyURI"/>
                                    </xs:sequence>
                                </xs:restriction>
                            </xs:complexContent>
                        </xs:complexType>
                    </xs:element>
                    <xs:element name="abstraction_link" minOccurs="0" maxOccurs="unbounded">
                        <xs:complexType>
                            <xs:complexContent>
                                <xs:restriction base="unificationType">
                                    <xs:sequence>
                                        <xs:element name="link_from" type="xs:anyURI"/>
                                        <xs:element name="link_to" type="xs:anyURI"/>
                                    </xs:sequence>
                                </xs:restriction>
                            </xs:complexContent>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:restriction>
        </xs:complexContent>
    </xs:complexType>
    <xs:choice>
        <xs:element name="responsibility" type="xs:string"/>
        <xs:element name="desc" type="xs:string"/>
        <xs:element name="condition">
            <xs:complexType>

```

```

        <xs:complexContent>
          <xs:extension base="conditionType"/>
        </xs:complexContent>
      </xs:complexType>
    </xs:element>
  </xs:choice>
  <xs:element name="eventflow" type="conditionType" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="exception_eventflow" type="conditionType" minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="UCDiagrations">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Actor" maxOccurs="unbounded"/>
      <xs:element ref="UseCase" maxOccurs="unbounded"/>
      <xs:element name="UC_ass" maxOccurs="unbounded">
        <xs:complexType>
          <xs:complexContent>
            <xs:restriction base="associationType">
              <xs:sequence>
                <xs:element name="ass_from" type="xs:IDREF"/>
                <xs:element name="ass_to" type="xs:IDREF"/>
              </xs:sequence>
            </xs:restriction>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="condition">
        <xs:complexType>
          <xs:complexContent>
            <xs:extension base="conditionType"/>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Requirement">
  <xs:complexType>
    <xs:choice>
      <xs:element ref="UCDiagrations"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
<xs:element name="design_Class">
  <xs:complexType>
    <xs:complexContent>
      <xs:restriction base="componentType">
        <xs:sequence>
          <xs:element name="ID" type="xs:ID"/>
          <xs:element name="Name" type="xs:string"/>
          <xs:element name="integration_link" type="unificationType" minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="abstraction_link" type="unificationType" minOccurs="0" maxOccurs="unbounded"/>
          <xs:element name="attribute" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
              <xs:attribute name="attribute">
                <xs:simpleType>
                  <xs:restriction base="xs:NMTOKENS">

```

```

        <xs:enumeration value="private"/>
        <xs:enumeration value="protect"/>
        <xs:enumeration value="public"/>
    </xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
<xs:element name="method" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <xs:attribute name="attribute">
            <xs:simpleType>
                <xs:restriction base="xs:NMTOKENS">
                    <xs:enumeration value="private"/>
                    <xs:enumeration value="protect"/>
                    <xs:enumeration value="public"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
    </xs:complexType>
</xs:element>
<xs:element name="condition">
    <xs:complexType>
        <xs:complexContent>
            <xs:extension base="conditionType"/>
        </xs:complexContent>
    </xs:complexType>
</xs:element>
</xs:sequence>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="Class_diagram">
    <xs:complexType>
        <xs:sequence>
            <xs:element ref="class" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="class_ass" minOccurs="0" maxOccurs="unbounded">
                <xs:complexType>
                    <xs:complexContent>
                        <xs:restriction base="associationType">
                            <xs:sequence>
                                <xs:element name="ass_from">
                                    <xs:complexType>
                                        <xs:simpleContent>
                                            <xs:extension base="xs:IDREF">
                                                <xs:attribute name="multi">
                                                    <xs:simpleType>
                                                        <xs:restriction base="xs:NMTOKENS">
                                                            <xs:enumeration value="0"/>
                                                            <xs:enumeration value="1"/>
                                                            <xs:enumeration value="n"/>
                                                            <xs:enumeration value="0..1"/>
                                                            <xs:enumeration value="0..n"/>
                                                            <xs:enumeration value="1..n"/>
                                                        </xs:restriction>
                                                    </xs:simpleType>
                                                </xs:attribute>
                                            </xs:extension>
                                        </xs:simpleContent>
                                    </xs:complexType>
                                </xs:element>
                            </xs:sequence>
                        </xs:restriction>
                    </xs:complexContent>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```

```

</xs:element>
<xs:element name="ass_to">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:IDREF">
        <xs:attribute name="multi">
          <xs:simpleType>
            <xs:restriction base="xs:NMTOKENS">
              <xs:enumeration value="0"/>
              <xs:enumeration value="1"/>
              <xs:enumeration value="n"/>
              <xs:enumeration value="0..1"/>
              <xs:enumeration value="0..n"/>
              <xs:enumeration value="1..n"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="association_type"/>
<xs:element name="restriction"/>
</xs:sequence>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="condition" type="conditionType"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Collaboration_diagram">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="class" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="collaboration_ass" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:complexContent>
            <xs:restriction base="associationType">
              <xs:sequence>
                <xs:element name="ass_from" type="xs:IDREF"/>
                <xs:element name="ass_to" type="xs:IDREF"/>
                <xs:element name="message">
                  <xs:complexType>
                    <xs:simpleContent>
                      <xs:extension base="xs:string">
                        <xs:attribute name="sequence" type="xs:int"/>
                      </xs:extension>
                    </xs:simpleContent>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:restriction>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="condition" type="conditionType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="Sequence_daigram">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="class" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="sequence_ass" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:complexContent>
            <xs:restriction base="associationType">
              <xs:sequence>
                <xs:element name="ass_from" type="xs:IDREF"/>
                <xs:element name="ass_to" type="xs:IDREF"/>
                <xs:element name="message" minOccurs="0" maxOccurs="unbounded">
                  <xs:complexType>
                    <xs:attribute name="sequence" type="xs:int"/>
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:restriction>
          </xs:complexContent>
        </xs:complexType>
      </xs:element>
      <xs:element name="condition" type="conditionType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Design_pattern"/>
<xs:element name="Design">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="design_Class" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Class_diagram" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Collaboration_diagram" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element ref="Sequence_daigram" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="code_class">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="id" type="xs:ID"/>
      <xs:element name="abstraction_link" type="unificationType"/>
      <xs:element name="attributes" type="declarationType" minOccurs="0" maxOccurs="unbounded"/>
      <xs:element name="methods" minOccurs="0" maxOccurs="unbounded">
        <xs:complexType>
          <xs:choice>
            <xs:element name="formal_parameter" type="formal_parameterType" minOccurs="0"
maxOccurs="unbounded"/>
            <xs:element name="local_parameter" type="declarationType" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element ref="statement" minOccurs="0" maxOccurs="unbounded"/>
            <xs:element name="localclass" minOccurs="0" maxOccurs="unbounded">
              <xs:complexType>
                <xs:choice>
                  <xs:element ref="statement" minOccurs="0" maxOccurs="unbounded"/>
                </xs:choice>
              </xs:complexType>
            </xs:element>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```



```
</xs:element>
<xs:element name="statement" type="statementType"/>
<xs:element name="declaration" type="declarationType"/>
<xs:element name="Implementation">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="code_class"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="FMSDI">
  <xs:annotation>
    <xs:documentation>FMSDI</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Requirement"/>
      <xs:element ref="Design"/>
      <xs:element ref="Implementation"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>
```