

南 華 大 學

資訊管理學系

碩士論文

建構以知識本體為基礎之大學院校會議紀錄

語意檢索系統

Building an Ontology-Based Semantic Query System
for College's Meeting-Records

研 究 生：李國嘉

指 導 教 授：邱英華

中華民國 101 年 5 月

南 華 大 學

資訊管理學系(所)

碩 士 學 位 論 文

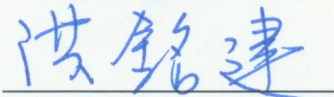
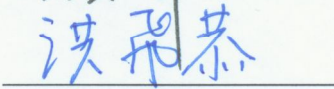
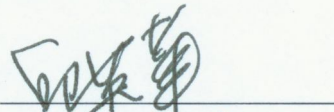
建構以知識本體為基礎之大學院校會議紀錄語意檢索系統

研究生：(請學生親筆簽名)

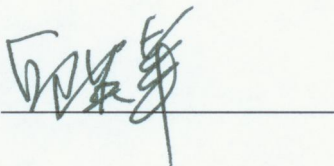


經考試合格特此證明

口試委員：



指導教授：



系主任(所長)：



口試日期：中華民國 101 年 4 月 18 日

誌 謝

經過多年間斷的努力後，終於衝破研究上的瓶頸而有所產出，期間也學習到了許多新的知識，充實了許多，也希望該研究能對於會議記錄檢索的領域上有所貢獻。在這期間，感謝邱英華老師有恆心和毅力的不斷地追蹤進度與指導，也感謝凱貞、憶伶、秋婷、宜美、妃娜、逸為、福生、咸伍、麗美大姊及士軒大哥等同學們的彼此協助與共同努力打拼，再感謝吳光閔所長、洪銘建老師、洪飛恭老師、趙偉勛老師、瑞楹學長、政璋哥、文欽、慶源、嘉彬及公司主管等同事們給予我的鼓勵、建議及技術上的指導，推動我在不斷地修正論文系統架構下終於有了新的突破，使得我有機會成功取得學業，以上還有許多未能一一詳加介紹。總之，對於這些幫助我的親朋好友們再次深表感謝之意。謝謝大家！！

在口試期間，承蒙洪飛恭老師及洪銘建老師能撥空前來，給予學生我不吝指教與寶貴的建議，使得這份論文內容經過修飾後能更加完善。內容或許平易，所寫也未必登大雅之堂，然拋磚引玉，讓更多人能看到這塊領域可被發展的可能性，盼能集思廣義，提出更多不一樣的見解，得以來完善會記錄檢索系統，讓它更能精確提供使用者所須資訊。

建構以知識本體為基礎之大學院校會議紀錄語意檢索系統

學生：李國嘉

指導教授：邱英華

南 華 大 學 資 訊 管 理 學 系 碩 士 班

摘 要

語意網技術 (Semantic Web Technology) 的主要功能，就是對現有的網路資訊添加知識本體 (Ontology) 讓機器或軟體代理人 (Software Agent) 判讀，用以改善資料檢索上的窘境。而現今大學院校的會議紀錄大部份皆以文書檔案型式發布於全球資訊網上供人瀏覽，檢索資料上較事倍功半。部份雖已有建置資料庫的會議紀錄也僅限於關鍵字的查詢，且會議紀錄的資料描述多數較不拘於型式，因而使得在檢索上較不易找到我們所要的資訊。

為了解決上述的問題，本文使用語意網技術並配合同義詞集 (Synonym Set, SynSet) 的應用，將會議內容適時嵌入相關領域的詞彙及區域文化上不同的用字遣詞，建置了一個以知識本體為基礎的大學院校會議紀錄語意檢索系統 (Semantic Query System for College's Meeting-Records, SQSCMR)，用以改善會議紀錄查詢在關鍵字應用上的限制問題。

關鍵字：語意網技術、知識本體、同義詞集、會議紀錄、軟體代理人

Building an Ontology-Based Semantic Query System for College's Meeting- Records

Student : Kuo-Chia Lee

Advisor : Dr. Yin-Wah Chiou

Department of Information Management
The M.I.M. Program
Nan-Hua University

ABSTRACT

The main function of *Semantic Web Technologies (SWT)* is to extend World Wide Web (WWW) with *ontology* to let machines or *software agents* to interpret the meaning of data. It is used to improve the searching capability of information. Presently, most of the colleges have documented their school *meeting-records* and uploaded into the WWW for users to browse freely. However, the users usually have donated a lot of efforts in searching and may not find the expected results. Although some *meeting-records* have been documented in the database, the traditional search engines only provide the keywords-matching. Therefore, it is difficult to quickly find the accurate information for meeting-records.

To solve the above problem, we employ the SWT with *Synonym Set (SynSet)* to build an ontology-based *Semantic Query System for College's Meeting-Records (SQSCMR)*. Using this system, the users can quickly and efficiently find the accurate meeting-records information.

Keywords: Semantic Web Technologies, Ontology, Synonym Set, Meeting- Records , Software Agent .

目 錄

第一章、緒論.....	1
第一節 研究背景.....	1
第二節 研究動機.....	2
第三節 研究目的.....	2
第四節 研究限制與範圍.....	3
第五節 研究方法.....	3
第六節 論文架構.....	3
第二章、文獻探討.....	6
第一節 會議紀錄.....	6
第二節 搜尋引擎.....	10
第三節 語意網技術與知識本體.....	11
第四節 查詢工具與語言.....	19
第五節 語意分析概念.....	22
第三章、系統分析與設計.....	23
第一節 系統架構.....	23
第二節 會議紀錄知識本體架構.....	25
第四章、系統實作.....	40
第一節 系統實作開發環境與工具.....	41
第二節 建構會議紀錄知識本體.....	42
第三節 建構查詢服務系統環境.....	45
第四節 建構查詢服務系統知識本體.....	49
第五節 使用者查詢介面建置.....	52
第六節 系統應用實例.....	80
第五章、結論與建議.....	100
第一節 結論.....	100
第二節 未來展望.....	101

表 目 錄

表 2-1 會議規範架構表.....	7
表 2-2 某科技大學會議架構表.....	8
表 2-3 建立知識本體之七項步驟	13
表 3-1 主題分類說明.....	27
表 3-2 提案單位近義詞、等義詞及相關詞彙的描述	29
表 3-3 提案型態說明.....	30
表 3-4 會議地點近義詞、等義詞及相關詞彙的描述	33
表 3-5 會議時間近義詞、等義詞及相關詞彙的描述	35
表 3-6 會議種類資訊的描述.....	36
表 3-7 會議種類資訊的物件屬性之 Domain 及 Range 的描述.....	36
表 3-8 物件屬性之 Domain 及 Range 的描述	38
表 3-9 註解屬性的描述.....	39
表 4-1 SQSCMR 系統開發工具列表	42
表 4-2 物件屬性之 Domain 及 Range 的描述	49
表 4-3 註解屬性的描述.....	50
表 4-4 依「主題」查詢之 SPARQL 查詢指令	54
表 4-5 依「提案型態」查詢之 SPARQL 查詢指令	56
表 4-6 依「提案單位」查詢之 SPARQL 查詢指令	58
表 4-7 依「會議時間」查詢之 SPARQL 查詢指令	60
表 4-8 依「會議地點」查詢之 SPARQL 查詢指令之一	62
表 4-9 依「會議地點」查詢之 SPARQL 查詢指令之二	64
表 4-10 依「會議種類資訊」查詢之 SPARQL 查詢指令	66
表 4-11 依「主題(含同義詞)」查詢之 SPARQL 查詢指令.....	68
表 4-12 依「提案單位(含同義詞)」查詢之 SPARQL 查詢指令.....	70

表 4-13 依「會議時間(含同義詞)」查詢之 SPARQL 查詢指令.....	72
表 4-14 依「會議地點(含同義詞)+會議種類資訊」查詢之 SPARQL 查詢 指令	74
表 4-15 依「主題分類+會議時間(含同義詞)」查詢之 SPARQL 查詢指令	77
表 4-16 依「提案單位(含同義詞應用)+提案型態」查詢之 SPARQL 查詢 指令	79

圖目錄

圖 1-1 研究流程.....	4
圖 2-1 語意網的層級架構圖.....	11
圖 2-2 Current Web 與 Semantic Web 的差異圖.....	12
圖 2-3 RDF 三元組結構資源描述圖形.....	14
圖 2-4 RDF 多項式三元組資源描述圖形.....	15
圖 2-5 RDF 多項式三元組資源描述.....	16
圖 2-6 N3 表示法資源描述.....	16
圖 2-7 Jena Rule 格式.....	20
圖 2-8 三個 RDF 三元組.....	20
圖 2-9 SPARQL 的查詢語法.....	21
圖 2-10 SPARQL 的查詢結果.....	21
圖 3-1 SQSCMR 之系統架構.....	25
圖 3-2 會議紀錄知識本體架構.....	26
圖 3-3 主題分類基本架構.....	27
圖 3-4 提案型態基本架構.....	30
圖 3-5 案由與其細部描述資訊關聯圖.....	32
圖 3-6 [SA96-1-01 校務-報告案 01]與其細部描述資訊關聯圖.....	33
圖 3-7 會議地點基本架構.....	34
圖 3-8 會議時間基本架構.....	35
圖 3-9 會議種類資訊基本架構.....	37
圖 3-10 會議紀錄架構的本體概念關聯圖.....	38
圖 4-1 SQSCMR 之系統實作流程.....	40
圖 4-2 建立一份 Protege 3.3.1 新的專案.....	43
圖 4-3 建立新專案類型為 OWL/RDF Files.....	43
圖 4-4 Protege 3.3.1 環境視窗介面.....	44

圖 4-5 設定查詢服務系統的 JOSEKI 環境變數.....	46
圖 4-6 使用命令提示字元視窗啟動 JOSEKI 的服務.....	46
圖 4-7 JOSEKI 服務(Port:2020)啟動成功的畫面.....	47
圖 4-8 以瀏覽器檢視 Server 首頁運作成功的畫面.....	47
圖 4-9 以瀏覽器檢視 SPARQL 預設查詢介面.....	48
圖 4-10 SPARQL 的查詢結果.....	48
圖 4-11 建立類別.....	49
圖 4-12 建立物件屬性.....	50
圖 4-13 建立註解屬性.....	51
圖 4-14 建立實例.....	51
圖 4-15 SQSCMR 使用者簡易查詢介面.....	52
圖 4-16 簡易式查詢 - 依「主題」查詢介面.....	54
圖 4-17 簡易式查詢 - 依「提案型態」查詢介面.....	56
圖 4-18 簡易式查詢 - 依「提案單位」查詢介面.....	58
圖 4-19 簡易式查詢 - 依「會議時間」查詢介面.....	60
圖 4-20 簡易式查詢 - 依「會議地點」查詢介面之一.....	62
圖 4-21 簡易式查詢 - 依「會議地點」查詢介面之二.....	64
圖 4-22 簡易式查詢 - 依「會議種類資訊」查詢介面.....	66
圖 4-23 簡易式語意查詢 - 依「主題(含同義詞)」查詢介面.....	68
圖 4-24 簡易式語意查詢 - 依「提案單位(含同義詞)」查詢介面.....	70
圖 4-25 簡易式語意查詢 - 依「會議時間(含同義詞)」查詢介面.....	72
圖 4-26 簡易式語意查詢 - 依「會議地點(含同義詞)+會議種類資訊」 查詢介面.....	74
圖 4-27 複合式語意查詢 - 依「主題分類+時間(含同義詞)」查詢介面.....	77
圖 4-28 複合式語意查詢 - 依「提案單位(含同義詞)+提案型態」查詢 介面.....	79
圖 4-29 案例 1-1 查詢畫面.....	80
圖 4-30 案例 1-1 查詢結果.....	81

圖 4-31 案例 1-2 查詢畫面	82
圖 4-32 案例 1-2 查詢結果	82
圖 4-33 案例 1-3 查詢畫面	83
圖 4-34 案例 1-3 查詢結果	84
圖 4-35 案例 1-4 查詢畫面	85
圖 4-36 案例 1-4 查詢結果	85
圖 4-37 案例 1-5 查詢畫面	86
圖 4-38 案例 1-5 查詢結果	87
圖 4-39 案例 1-6 查詢畫面	88
圖 4-40 案例 1-6 查詢結果	88
圖 4-41 案例 1-7 查詢畫面	89
圖 4-42 案例 1-7 查詢結果	90
圖 4-43 案例 2-1 查詢畫面	91
圖 4-44 案例 2-1 查詢結果	91
圖 4-45 案例 2-2 查詢畫面	92
圖 4-46 案例 2-2 查詢結果	93
圖 4-47 案例 2-3 查詢畫面	94
圖 4-48 案例 2-3 查詢結果	94
圖 4-49 案例 2-4 查詢畫面	95
圖 4-50 案例 2-4 查詢結果	96
圖 4-51 案例 3-1 查詢畫面	97
圖 4-52 案例 3-1 查詢結果	97
圖 4-53 案例 3-2 查詢畫面	98
圖 4-54 案例 3-2 查詢結果	99

第一章、緒論

在本章，我們描述本文之研究背景、研究動機、研究目的、研究限制與範圍、研究方法及論文架構。

第一節 研究背景

各家公司行號及學校、政府機關．．．等為了公布及協調公務上的瑣事，或審議一些重要之事項和法規等文件，常利用會議之召開，召集各部、會主管來商討，並訂定出一些重要的決議事項，以供日後行政業務遂行之參考依據。以學校為例，除了行政、校務等重大會議外，另亦有經費規劃、稽核、運用成效、處務、院務、系務、校教評．．．等數十種會議，會議紀錄記載了大大小小的重要決議事項，經日積月累及承辦人員和與會主管的改朝換代之下，如未能相互傳承，往往都會流失掉一些重要決議資訊。

在這資訊科技進步神速的時代裡，雖各項會議紀錄皆已被電子化，但經日積月累下，已嚴然成為一份龐大的資料庫了。然而，我們將如何從這份龐大的資料庫查詢出我們想要的資訊，全得仰賴查詢系統。綜觀現今的資訊查詢系統，大部份皆以關鍵字來查詢，雖能查詢出大部份資訊，但並非全是我們想要的資訊，甚至有部份可能因用字遣詞的不同而並未被查詢出來。面對此一問題，在全球資訊網(World Wide Web, WWW)發明人 Tim Berners-Lee 發表出語意網 (Semantic Web) 此一學說後，已獲得了明顯的改善。

第二節 研究動機

會議紀錄記載了機關內各單位一堆大大小小的報告事項及討論的議案，其內容涵蓋了有諸多的人、事、時、地、物等相關的重要資訊（魏召欣，2003）。然而，會議紀錄在歷經長時間的累積及參與人員的世代交替之後，會議紀錄嚴然成為一份龐大的資料庫，且以目前人工方式，僅憑藉著會紀錄承辦人員的印象下，調閱開啟各項會議紀錄查詢所需的之資訊可說是極為不易，如能導入電腦化作業，並建構完整的資訊查詢系統，將對行政效率有所改善。

第三節 研究目的

資訊查詢系統雖建構完成，但往往在使用者欲使用全文檢索從中調閱出相關的報告資料或討論議案資訊時，因對於相關詞彙用字遣詞不夠具體，致使查詢後的結果並非全為調閱者所想要的會議紀錄，或甚至明明有相關議題卻因用詞不同而查詢不到。為了解決此一問題，本文應用語意網技術（Semantic Web Technologies）針對此問題再進行強化改善，深入會議紀錄領域的知識內容，適度地導入該領域的相關詞彙，重新定義各知識概念間的關聯，並建構該領域的知識本體（Ontology），以利日後調閱者可以使用描述相關的語彙，並透過知識本體內的關聯推論機制，將有助於資訊查詢系統能更加精準地尋找出符合使用者所欲查詢的資訊內容。

第四節 研究限制與範圍

本研究因礙於各地方文化不同而影響用字遣詞亦不盡相同，此外，有鑑於機關內各單位、部門對於各項會議紀錄可開放查閱的程度不同致使資料獲得不易等問題，故本文僅以某一科技大學作為研究的範例，並以已完整造冊之 96 至 98 三年期間學校內部份之重大會議紀錄為主要的研究樣本，例如：行政會議及校務會議。本文的語意分析方式，經初步分析與了解後，在「提案單位」的部份較能找到同義詞的關係存在。因此，本文將針對「提案單位」的部份使用了同義詞 (Synonym) 的語意分析之延伸來查詢會議紀錄相關內容 (鍾正男，2004)。

第五節 研究方法

本文的主要研究重點，在於針對「提案單位」透過語意分析，以方便使用者更容易找到會議紀錄裡所想要查詢的主題及相關細部資訊，並能適度地導入語意方面的應用，讓該查詢介面與方式更能具有親和力且讓查詢的資訊更為精準。本文之研究流程，如圖 1-1 所示。

第六節 論文架構

本文共分為五個章節，各章節重點說明如下：

第一章 緒論：說明本研究之背景、動機與目的，並將研究範圍、研究限制與研究方法加以述明後再簡述整體論文架構。

第二章 文獻探討：主要在探討會議紀錄格式與規範及某科技大學現有的會議紀錄架構、文件檢索之搜尋引擎介紹，進而導入語意網

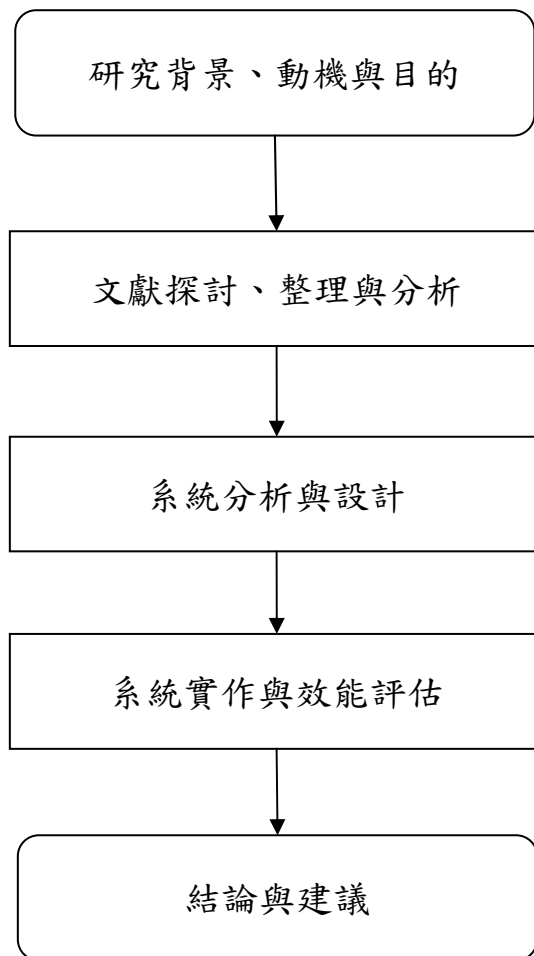


圖 1-1 研究流程

等技術的概念與應用（包含有知識本體、XML/XML Schema、RDF/RDF Schema、OWL、推論引擎、SPARQL 及同義詞...等）。

第三章 系統分析與設計：將前一章文獻探討的概念加以融合後，對整個會議紀錄架構重新分析、定義與設計一個符合該單位之會議紀錄知識本體，並簡述類別、屬性與值之間的關聯，用於後續實作上。

第四章 系統實作：經前三章的概念導入重新分析、定義與設計，須經由實作並在此加以一一述明、相關實作環境、使用的工具、實作流程，最後實際測試套用到實例應用及解說加以驗證其可行性。

第五章 結論與未來展望：總結本文的重點，並對於本系統整體設計概念有所不足的部份加以述明，以探討及延伸未來的研究發展方向。

第二章、文獻探討

本章主要描述會議紀錄的格式規範、搜尋引擎的文件檢索模式、語意網技術與知識本體、查詢工具與語言所用到的推論引擎及 SPARQL 的語法概述，以及語意分析概念的應用。

第一節 會議紀錄

有關「會議規範」於民國四十三年五月間公布試行，至今已有數十多年，時代的演變如此迅速，使得固定的條文，難以符合時勢所趨。為了改進議事秩序及提高議事效率，該「會議規範」遂根據各機關試行經驗及反映意見，並參酌其他國家最新議事規則，將其條次重新審視並予以修正，以符合時代所需。各公私立機關會議之召開迄今皆參卓該「會議規範」予以遂行之。

會議之定議：乃三人以上，循一定之規則，研究事理，達成決議，解決問題，以收群策力之效者，謂之會議（張仁青，1979）。

「會議紀錄」亦稱「議事紀錄」，乃由該會議委員經由會議之召開多數人出席予以討論並決定議（亦即決定或決議）各項議事之內容後，協同會議紀錄人員於會議中同步記載結果之文書，謂之「會議紀錄」。會議中的討論、決定議及選舉等均為與會委員所共同決定議之事項，不但須一一付諸施行，而且對與會委員有約束力，故會議紀錄應視為重要文書。依「會議規範」會議紀錄之主要內容如表 2-1 所示。

表 2-1 會議規範架構表（張仁青，1979）

項目	說明
一、標題：	即會議名稱及會次，記載其為某會第某次會議紀錄。
二、時間：	記載其集會之年月日時。
三、地點：	記載其集會之場所。
四、出席人員：	由出席人員簽名其上。如另有簽到簿，則可註明「詳見簽到簿」亦可。
五、列席人員：	由列席人員簽名其上。如另有簽到簿，則可註明「詳見簽到簿」亦可。
六、主席：	由本會議召開之主持或召集人簽名其上。
七、紀錄：	記載其紀錄人員之姓名。
八、儀式：	記載其開會儀式，大部份內容皆記載了『行禮如儀』或『開會如儀』，如不舉行儀式，此項可省略。
九、報告事項：	包括開會宗旨，過去工作，上次決議案執行情形，本次會議議程以及其他有關事項。參考順序如下： （1.宣讀上次會議紀錄。2.報告上次會議決議案執行情形。3.委員會或委員報告。4.其他報告。）
十、討論事項：	應依議程逐案記錄，每案須載明『提案單位』、『案由』及『決議辦法』。參考順序如下： （1.前會遺留之事項。2.本次會議預定討論之事項。）
十一、選舉事項：	詳記選舉票數及結果。
十二、臨時動議：	記錄各發言人之意見及表決後之結果。
十三、散會：	記明散會結束時間。

綜觀「會議規範」及其他各機關會議範本內容後，某科技大學仍參卓其架構，增刪部份項目，以符合該校現況所需之會議架構如表 2-2 所示。

表 2-2 某科技大學會議架構表

會議名稱：
時間：
地點：
主席：
出席：
列席：
記錄：
壹、宣讀上次會議紀錄：
貳、上次會議執行情形報告：
參、報告事項：
報告案一：○○室
案由：
說明：
決定：
肆、討論事項：
提案一：○○室
案由：
說明：
決議：
伍、臨時動議：
陸、散會：

「會議紀錄管理」的演進，從先前的紙本作業進步到資訊化時代的電子檔案作業模式。雖已方便了會議資料的記錄與保存，但在長年累積下的資料量甚為可觀，管理的問題逐漸浮現。我們簡易列舉三項說明如

下：

- 問題一：管理人員在不斷地世代交替之下，使得管理人員對於現有會議紀錄電子檔內容所保存的位置熟悉度有些許的差異度存在，致使日後查詢人員欲向管理人員查詢會議內容不易索引。
- 問題二：對於各提案單位名稱的世代演變（例如：招生組=菁英中心、會資系=財金系、國際貿易科=國際企業管理系=行銷與流通管理系）及會議詞彙（記錄=記載=筆錄=摘錄、教學單位=學術單位、會議=開會=議事）所造成的了解不足，使其問題同前項一般，因而日後查詢人員欲向管理人員查詢會議內容不易索引之。
- 問題三：現有的電子檔會因軟體的演進或跨平台等等諸多問題而引發了相容性的問題。

從以上所略述的問題，我們可以了解發展「會議管理系統」的必要性，以提升日後會議資料的建置、保存與查詢的便利性。現今網際網路發展已無遠弗屆，倘若只是將該「會議管理系統」以一般 C 語言、C++、Visual Basic 等等程式開發成一般套裝系統，在功用上將有所略嫌不足且使用上將受限於本機作業居多，實用性不夠完善。若能跟上時代潮流，運用網際網路技術，將其「會議管理系統」建置成網頁管理平台型態，這不但只是將「會議管理系統」資訊化，對於現今網際網路成熟度的應用，也提供管理者與資料查詢者可不受距離上的限制，給予同時間多人上網使用並符合跨平台作業服務，不論使用者是 Windows、Linux、Mac 或手持式 Android 等作業系統，只要能上網並使用自家的瀏覽器即可查閱。這對於日後系統資料建置、保存與查詢或移機等等管理維護將會更務實一些。

第二節 搜尋引擎

網際網路上的資訊在歷經無數人們的建置後已嚴然成為一個龐大的資訊載體 (Media)，而每個資訊站台端點則經由蜘蛛網般的縱橫交錯相互連結著。然而，我們要從中找尋資訊時，是經由先前經瀏覽過就已知網頁資訊位址，或是旁人引介至該網址而得以正確地找尋到我們所想要的資訊。如果想要在這龐大的資訊載體中不憑藉上述方法而要搜尋出我們想要的資訊猶如大海撈針，而搜尋引擎則是憑藉著這股需求動機被發展出來。

搜尋引擎開發廠商眾多，在相互競爭、汰舊換新與整併後得以永續成長下去的皆為優秀的代表，例：Google、Bing、yahoo...等。搜尋引擎也在歷經多次的改良後，得已自動從網際網路搜集資訊，並經由自己本身作業程序進行摘錄重點資訊與連結後紀錄到自己的資料庫中，再透過自己本身獨有的演算技術提供查詢者快速又有效的資料查詢平台。所以搜尋引擎的工作原理大致上有「搜集資訊」、「整理資訊」及「接受查詢」；其中，以「整理資訊」及「接受查詢」的過程，大量應用了各搜尋引擎開發廠商自家特有的「文字資訊檢索」技術，並依據網路超文字鏈結(即 hypertext link，是以一連串含有「底線」及「藍色」的「文字」所構成)的特點，嵌入了更多可索引的資訊。

網路時代的來臨，伴隨著網際網路的蓬勃發展，各式各樣的資訊內容都被予以網路化，可搜尋的網頁變得愈來愈多。又網頁內容的好壞參差，素質不一，搜尋引擎所搜尋的資料將變得越來越沒有品質保證。例如：討論區、知識家...等內容，良莠淆雜，亦是網際網路可搜尋資訊內容的一部份。未來的搜尋引擎是必要朝向「智慧型的語意搜尋引擎」的方向發展，加入不一樣的元素，得以改善目前的問題並提供更精確及合宜的資訊服務。

第三節 語意網技術與知識本體

壹、語意網

在 2000 年的 XML 會議上，全球資訊網（WorldWideWeb，WWW）的發明人伯納斯李（Tim Berners-Lee）發表了一則能達成語意網（Semantic Web）的層級架構圖（如圖 2-1），由下而上每個層級都包含有他所建議的語言。最底層由 Unicode、URI、XML、XML Schema...等語言為基礎，再以 RDF、RDF Schema 語言架構至上一層，而 Ontology 則是在 RDF 之上運作，再經由推論引擎運作在 Ontology 之上，得以進行邏輯推論並加以驗證。

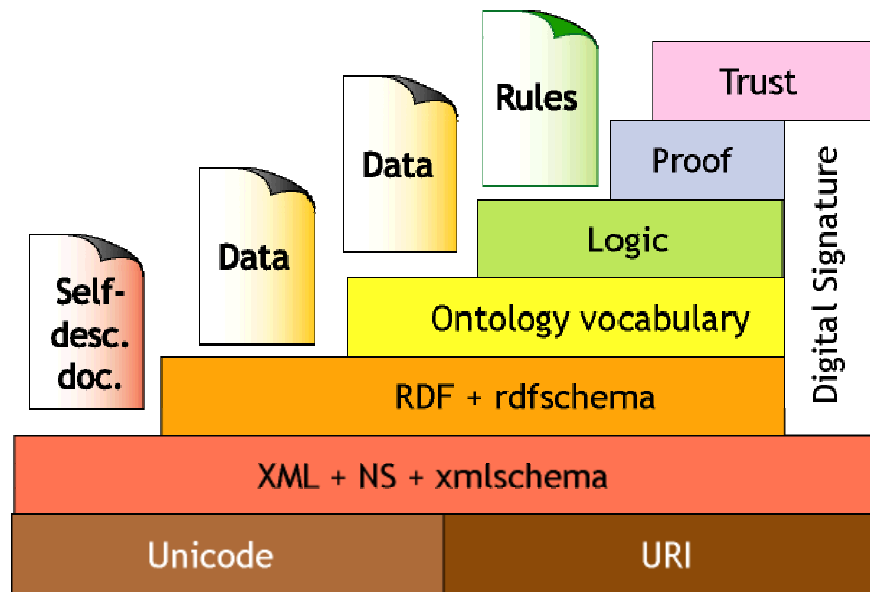


圖 2-1 語意網的層級架構圖（Berners-Lee, 2000）

語意網並非全新的網路，而是目前網路架構的延伸，只是運用知識本體與智慧型代理人提供機器可理解的語言，而不修改既存的網路內容（陳言熙，2006）。

語意網的主要目的是為了讓電腦能夠瞭解人類的語言，把目前已經在全球資訊網上的資料，經加工或轉換後，變成電腦能判讀的資訊，相對地，讓電腦亦能夠真正了解查詢人員所輸入資料的語意，而能進一步自動化替查詢人員作更精確的資料判斷與處理，提供更符合的資訊內容。

從圖 2-2 所顯示出傳統網頁(Current Web)與語意網(Semantic Web)的差異，我們可以更清楚的了解傳統的網際網路和語意網有何差別。在傳統的網際網路中，文件與文件有超連結關係，但並未指出其關係為何，如圖 2-2 左區塊，而圖 2-2 右區塊中；則為語意網之示意圖，相關的文件不但有超連結，而且會標示出此連結關係為何。換言之，在語意網中，資源連結會描述該連結之類型屬性（陳昭珍，2006）。

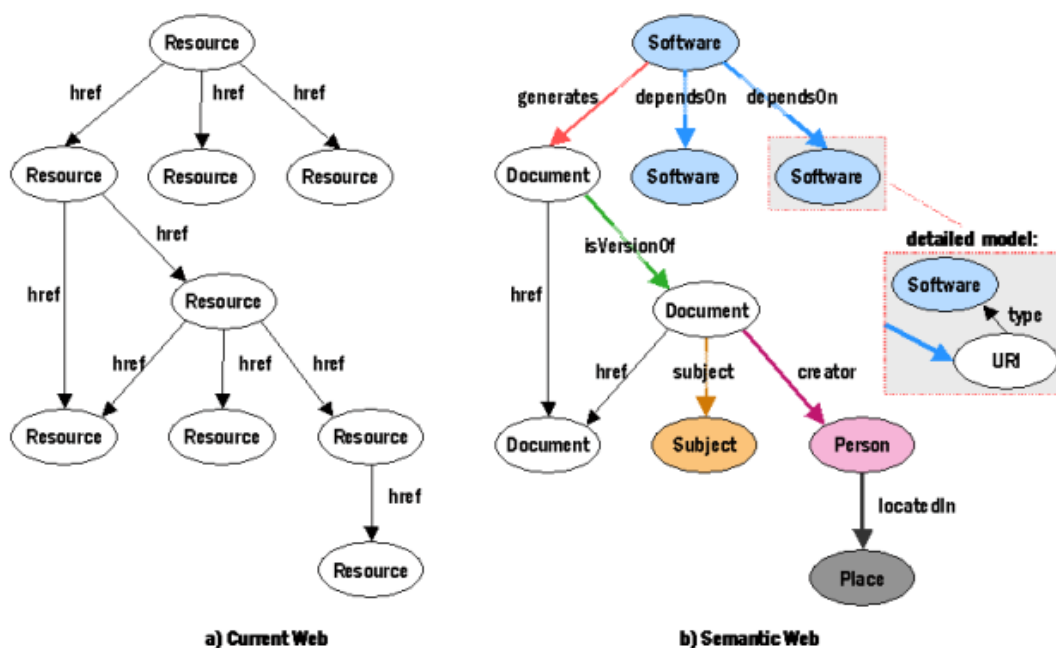


圖 2-2 Current Web 與 Semantic Web 的差異圖(Koivunen & Miller, 2001)

綜觀上述，語意網是由原生網際網路資訊附加上知識本體(Ontology)及推論機(Inference Engine)的組合，其功用就是提供能被機器所理解的網站服務。

貳、知識本體

知識本體 (Ontology) 主要是用來描述與定義某特定的應用領域內的各種知識的語言。較具體而言，知識本體是對於某一特定領域中，許多知識術語 (Knowledge Term) 所組成的集合，包含語彙 (Vocabulary)，語意上的相互連結 (Semantic Interconnection)，以及在推論 (Inference) 和邏輯 (Logic) 上的簡單規則 (Hendler, 2001)。而語意網的主要概念，就是要讓網際網路上的資料都能透過標準的知識本體的語言來相互表達，一方面可以達到知識共享的目的，另一方面也可以達到讓電腦可以讀得懂知識本體語言所描述出來的資料敘述，了解其真正語意的內涵，而能進一步地自動替使用者作更精確的資料判斷與處理。

致於要如何建構知識本體之架構，Noy et al. (2001) 總體歸納出七項步驟如表 2-3 所示。

表 2-3 建立知識本體之七項步驟

編號	項目
1.	決定出知識本體的領域及範圍 (Determine the domain and scope of the Ontology)。
2.	考慮再利用現存的知識本體 (Consider reusing existing Ontologies)。
3.	列舉出知識本體內的重要的詞彙 (Enumerate important terms in the ontology)。
4.	定義類別及類別階層 (Define the classes and the class hierarchy)。
5.	定義類別的屬性 (Define the properties of classes – slots)。
6.	定義屬性的截面或限制 (Define the facets of the slots)。
7.	建立類別的例子或物件 (Create instances or objects)。

參、RDF 與 RDFS

資源描述架構(Resource Description Framework,RDF)是語意網構成的基本要素，RDF 與 RDF Schema(RDFS)是全球資訊網協會(World Wide Web Consortium, W3C)為解決資源描述問題而創造出一種定義網際網路資訊的標準格式。RDF 基本上也是一種建構於 XML(eXtensible Markup Language)的語言格式，主要是用來描述網際網路上的資源及其相關細部描述性資訊，並允許使用者自訂標籤(tags)；也因為可以自訂標籤，所以可以詳細地描述使用者所想要描述的任何可以詮釋的資料。

RDF 運用了三元組(triple)的結構來表示任何一項資源的運算式描述，基本組成分別為主體(Subject)、謂詞(Predicate)以及客體(Object)等三個部份，主要是將兩份資訊透過一項屬性或關連表示法將之串聯起來，即為所謂之三元組(劉升平等，2009)。RDF 的三元組圖形是由節點與有方向性箭頭線條所組成的，節點代表一個資源或一個描述值，即為 Subject 或 Object，有方向性箭頭線條代表一項屬性或一個關連表示法，即為 Predicate，RDF 的三元組結構資源描述圖形如圖 2-3 所示。

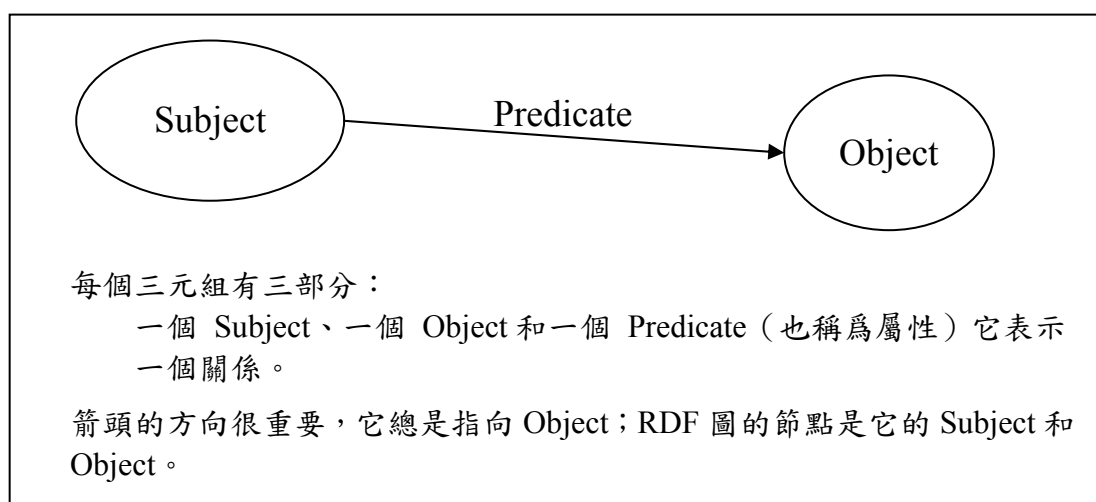


圖 2-3 RDF 三元組結構資源描述圖形 (Klyne et al., 2004)

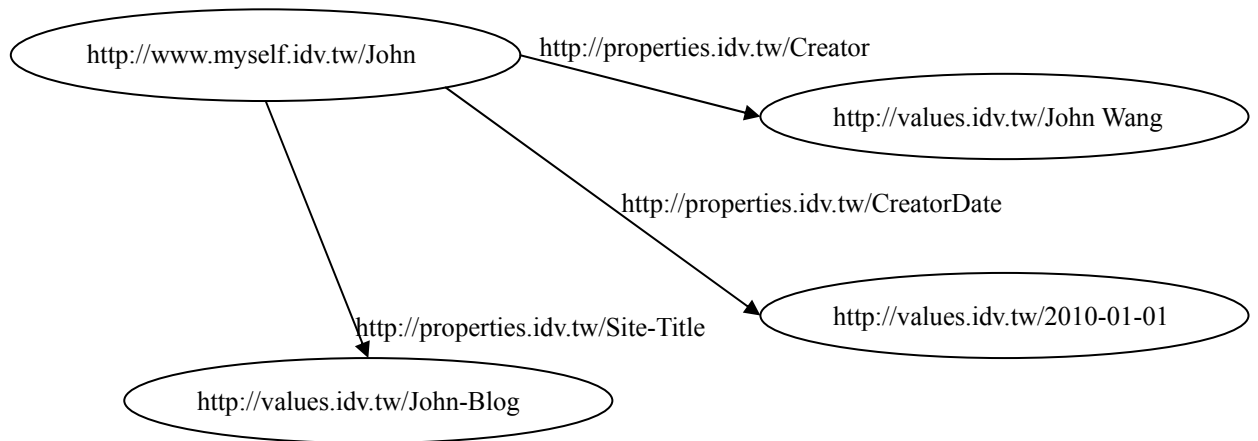


圖 2-4 RDF 多項式三元組資源描述圖形

RDF 為了讓資料內容在網際網路上不產生重覆性使其具有獨一無二的識別方式及其完整的互通性，RDF 結合了通用資源標幟碼（Uniform Resource Identifier，URI）的命名規則來做為資料間的識別與連結交換；亦即，RDF 結合了 URI 能在應用程式間做資料交換而不喪失語意的通用特性（陳亮廷，2005）。在運用三元組的結構上，如果一個 statement 為 John Wang is the Creator of the resource 而 <http://www.myself.idv.tw/John> 以一般文字表示方法為 (Creator, “<http://www.myself.idv.tw/John>”, “John Wang”)，若以 RDF 三元組圖形的表示方法並導入多項描述例子如圖 2-4 所示（王梅玲，2011）。

RDF 多項式三元組資源描述圖之說明如下（曾國峰，2010）：

1. 「<http://www.myself.idv.tw/John>」即為一個資源（Subject）。
2. 「<http://properties.idv.tw/Creator>」即為一個資源的屬性（Predicate）。
3. 「<http://values.idv.tw/John Wang>」即為一個資源屬性的值，另亦可為另一個資源的延伸（Object）。

<http://www.myself.idv.tw/John>	<http://properties.idv.tw/Creator>	<http://values.idv.tw/John Wang>
<http://www.myself.idv.tw/John>	<http://properties.idv.tw/CreatorDate>	<http://values.idv.tw/2010-01-01>
<http://www.myself.idv.tw/John>	<http://properties.idv.tw/Site-Title>	<http://values.idv.tw/John-Blog>

圖 2-5 RDF 多項式三元組資源描述 (修改自吳育賢, 2008)

依據圖 2-4 RDF 多項式三元組資源描述圖，將其轉換成敘述式的三元組資源描述表示法如圖 2-5 RDF 多項式三元組資源描述。以 RDF 三元組的表示方法，在敘述上讓一般人感覺非常冗長，尤其當資料越來越多筆時將不易檢視，如圖 2-5 RDF 多項式三元組資源描述，為了簡化該描述，於是乎有了另一種表示法 N3 (Notation 3) (Berners-Lee, 2006)的產生，N3 表示法主要是用來簡化 RDF 三元組的重複冗長的描述部份，N3 表示法利用程式設計中常用的「定義名稱」的方法，將 RDF 三元組資源描述中「所有使用到的 URI」定義成對應「精簡的前置詞 (Prefix)」，在往後描述中提及各項資源 (物件) 時，只需在其資源 (物件) 前面加上「精簡的前置詞」即可，其描述表示法如圖 2-6 N3 表示法資源描述。

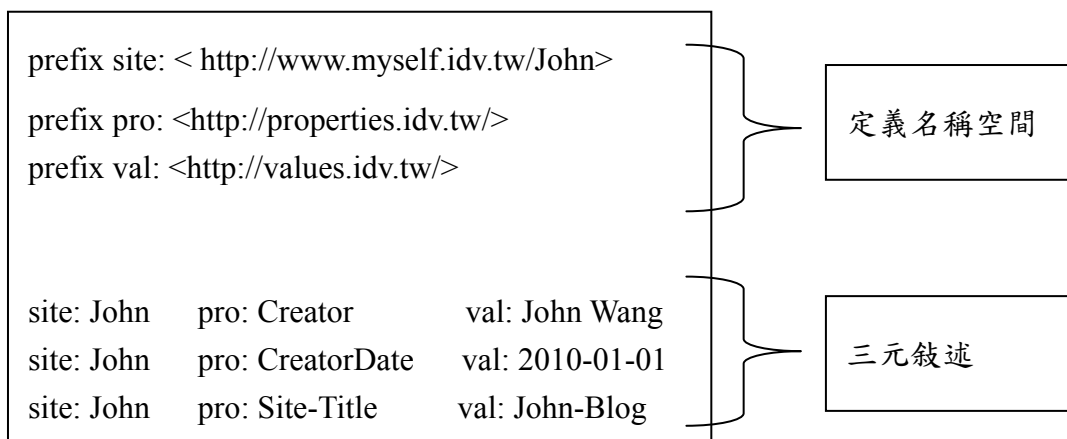


圖 2-6 N3 表示法資源描述 (修改自吳育賢, 2008)

RDF Schema (RDFS) 為 RDF 的延伸語法，主要是用來定義 RDF 的類別及屬性等詞彙之規範，並可限定描述特定應用領域的類別或次類別之間的關聯、可限定類別可以使用的屬性及其屬性所能指向的值與可使用的範圍。

肆、OWL (Web Ontology Language)

OWL 是全球資訊網協會(World Wide Web Consortium, W3C)於 2004 年正式發表的標準，主要被應用在 Web 上處理知識本體內容的語言。OWL 則算是 RDF/ RDFS 改良後的進階知識本體語言工具，等同是延伸的應用。OWL 建構在 RDF/ RDFS 之上，繼承了在 RDF/ RDFS 的語法，所以能夠與 RDF/ RDFS 向下相容使用。OWL 可以被應用於全球資訊網中來明確描述其網頁資訊內容所用到的辭彙表中術語的涵義以及術語間的關係。然這些術語及術語間的描述關係即被稱為知識本體(Ontology)。就表達涵義和語義關係描述能力方面，OWL 比起 XML、RDF 和 RDFS 有更多的表達方式來達到描述存在於網際網路上的各種資源與其應用程式之間的相互關係，使機器可以更進一步理解 Web 上所表達內容的能力（劉升平、蘭煜峰，2004）。

在語意網的概念裡 RDF/RDFS 雖然是基本的語言的應用，且能描述的能力也相當完整，也因為完整和複雜，相對地，限制最少且最寬鬆，在類別、屬性限制上或邏輯概念關聯等等部份無法詳細表述。例如：規範類別(Class)之間的「分離關係」，例如已付款、未付款都是"繳費情形"的子類別，RDF 只能顯示這樣的關係，而 OWL 卻可以強調已付款與未付款的分離關係(非已付款即未付款)。此外，在表示「布林集合的關係」的部份，例如交集、聯集、補集等等，也是 OWL 才能達到的領域（Smith et al., 2004）。

總而言之，OWL 能夠更加詳細地來定義及描述語意網所想要達到的目標，且 OWL 的語言更能讓機器理解其資料的意涵，並從邏輯推理中，得到所想要的相關資訊。不過，雖說是如此，「詳細的描述能力」與「有效的邏輯推理」其實是猶如「魚與熊掌無法兼得」的概念是相同的，而這就跟 OWL 當初所設計出來的三個子語言有關係，使用者則可參考其功能性，擇一符合需求之子語言運用於知識本體。在下列，我們簡述三個子語言的定義（李俞萱，2008）

1. OWL Lite：該子語言之優點主要是讓使用者能夠輕易實作，排除掉一些類別讓人實作起來比較簡易(不用定義一大堆東西)；而其缺點是也同於 OWL DL，缺乏 RDF 的相容性與表達性。
2. OWL DL：該子語言限制了 RDF 與 OWL 之間的使用，且更明確、精準地描述資訊 讓機器能夠做推理支援；而其缺點就是缺乏相容 RDF 的廣泛性，對於資料本身的限制較多。
3. OWL Full：該子語言可以表示的屬性很多，限制最寬鬆的 OWL，OWL Full 的文件可以完整相容 RDF/RDFS；而其缺點是定義的東西太多、太含糊會使得機器無法做推理邏輯。

第四節 查詢工具與語言

本節描述本系統所使用到的查詢工具與語言，包含：Joseki 與 Jena 推論引擎以及 SPARQL 查詢語言。

壹、推論引擎

知識本體在透過OWL及RDF等一層的語法建構下，已形成為一個有用的知識本體。那知識本體都有了，我們要如何從中得到蘊藏在裡面的知識內容呢，想當然爾就是借助推論引擎的力量將它推理萃取出來。現今常見的許多免費或付費的推論引擎有：Joseki、Jena、RacerPro、Pellet、Jess、FaCT...等。在本文，我們將以Joseki為基礎的簡易型推論引擎架構在Jena的環境裡，並自行開發查詢介面來整合SPARQL的語法，應用在本研究上較為便利且能推論出符合我們需求的資訊內容。以下則是我們對Joseki與Jena推論引擎的一個簡要的概述（鄭雅仁，2010）：

- **Joseki**(<http://www.joseki.org>)：此為 RDF 伺服器端的推論引擎，主要建立在 Jena 的操作環境上，它支援 SPARQL 語言的使用，為簡易型的推論引擎，可被安裝在任何的 Servlet Container 或 Web Application Server 環境下使用，像 Jetty 或 Tomcat 環境皆可。
- **Jena**：該推論引擎是由 HP Lab 所開發的語意網應用程式，是以 Java 平台所開發的開放式軟體，只需將 Jena Library 加入整合開發環境中，即可執行推論功能。其中 Jena 推論引擎提供許多開發語言，包含 RDF、RDFS、DMAL+OIL、OWL、RDQL、SPARQL...等語言。Jena 推論引擎除了具有知識本體的推論外，另外一大特色就是 Jena 本身有自己的 Rule 推論引擎，只要依照 Jena 所制定的格式撰寫 Rule，如圖 2-7 所示 Jena Rule 格式，即可使用 Rule 推論引擎。

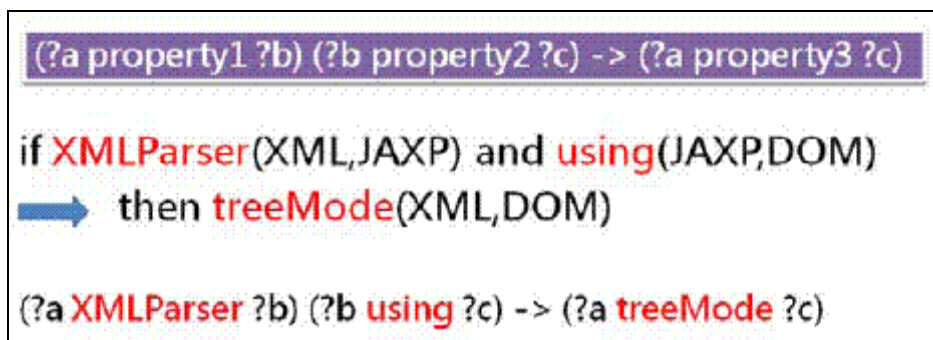


圖 2-7 Jena Rule 格式 (鄭雅仁, 2010)



圖 2-8 三個 RDF 三元組 (修改自 王怡文, 2008)

總結上述推論引擎的說明，其實就只有一個目的，就是要從建置完成的知識本體中檢視其術語間的關聯及內含的隱性資訊，並推理出我們所想要的資訊。以圖2-8為例，我們可從中歸納出小伍是鳥類動物。

貳、SPARQL 查詢語言

W3C 正在打造推論引擎以及其他多種技術，其中之一就是 SPARQL。它是一種用於 RDF 上的查詢語言，該查詢語法類似 SQL 語法。SPARQL 分為三個區塊 (Eric et al., 2008)，我們簡述如下：

- 第一個部份：是「定義名稱空間」，使用了 Prefix 的定義方式；
- 第二個部份：是「定義查詢名稱」，使用了 Select 的方式來取得欲查詢的變數內容；
- 第三個部份：是「定義查詢條件」，使用了 Where 的方式來限制查詢的條件，另也可配合 Filter Regex 或 Order By... 等語法來做進階條件的限制。

SPARQL 的查詢語法範例（如圖 2-9 所示）。而查詢後的結果，則顯示於如圖 2-10。

Example queries (or edit and write your own!). All the text boxes invoke the same "book"

SELECT - get variables (apply XSLT stylesheet)

一 {
 二 {
 三 {

```

PREFIX books: <http://example.org/book/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?book ?title
WHERE
{ ?book dc:title ?title }
  
```

Output XML: with XSLT style sheet (leave blank for none):

or JSON output:

or text output:

or CSV output:

or TSV output:

Force the accept header to text/plain regardless

圖 2-9 SPARQL 的查詢語法

SPARQLer Query Results

SPARQLer Query Results

book	title
<http://example.org/book/book7>	"Harry Potter and the Deathly Hallows"
<http://example.org/book/book2>	"Harry Potter and the Chamber of Secrets"
<http://example.org/book/book4>	"Harry Potter and the Goblet of Fire"
<http://example.org/book/book6>	"Harry Potter and the Half-Blood Prince"
<http://example.org/book/book1>	"Harry Potter and the Philosopher's Stone"
<http://example.org/book/book3>	"Harry Potter and the Prisoner Of Azkaban"
<http://example.org/book/book5>	"Harry Potter and the Order of the Phoenix"

圖 2-10 SPARQL 的查詢結果

第五節 語意分析概念

語意分析最常遇見的即為語言不同而語義相同或相近的語詞（黃居仁，2003），該資料即為**同義詞**（Synonym）。而同義詞又有分別，如果意義相同的稱為「**等義詞**」，它的意涵在於認知意義相同，且更在感情色彩或風格特點上亦是相同，例如「花生」等同「土豆」的意思，或者是「番薯」等同「地瓜」的意思，像這類例子較為少數。而同義詞的另外一個意涵為主要意義基本相同，但程度、色彩、風格等有別，這種意義相近的稱之為「**近義詞**（Near-Synonym）」，大部分同義詞皆屬此類型，如贊成、同意（教育部國語推行委員會，2007）。

在「以知識本體為基礎的語意查詢系統之研究-以圖書館為例」論文中提到：

使用同義詞（Synonym）：輸入檢索詞“電腦”=> 可找到跟電腦、計算機、微處理器相關的書籍，而不是書名裡只有含“電腦”（鍾正男，2004）。

在本文，我們經由上述概念的反推，對於語意分析這個區塊，主要是應用在「案由」、「提案單位」、「會議時間」、「會議地點」等區塊上，以「提案單位」為例，在輸入查詢詞彙時，可以是一級單位完整名稱、簡易名稱及改名前、改名後的相關歷史名稱。另外，部份單位有合併單位情形皆予以建置其合併前後的差異關係詞彙，該同義詞導入主要讓電腦能利用這些隱性詞彙經過推理後更能理解並查詢到使用者想要查詢資料內容。例如：在輸入查詢詞彙為「電子計算機中心、電算中心、圖資處、圖書資訊處」時，皆可經由詞彙的關聯比對後指向到「電子計算機中心」，並查詢出「電子計算機中心」所提案的相關資訊。

第三章、系統分析與設計

在本章，我們首先對會議紀錄文件做知識分析並描繪出整個大學院校會議紀錄語意檢索系統（Semantic Query System for College's Meeting Record, SQSCMR）之架構，然後我們再針對會議紀錄進行知識本體工程（Ontology Engineering）拆解出六大類別並做細部分析與設計整個會議紀錄知識本體之概念關係圖。

第一節 系統架構

在各單位召開眾多的會議中，如何可以迅速有效地檢索出在會議中所想要的報告或討論的議題，已成為會議查詢系統一個重要的課題。本文所開發的「會議查詢系統」之系統架構如圖 3-1 所示，該系統主要的元件包含有「使用者查詢介面模組」、「查詢系統模組」、「會議紀錄知識本體」等三大區塊所組成。其流程為先由知識工作者將會議資料整理過後透過 Protégé 免費的工具軟體建構出會議紀錄知識本體，使用者再透過使用者查詢介面模組與查詢系統模組進行溝通，並至會議紀錄知識本體將使用者所需的資料粹取出來，以供使用者閱讀使用。此三大元件之功能，我們說明如下：

- **使用者查詢介面模組**：主要是應用 Dreamweaver cs3 及 Notepad 對原有 Server 端的範本進行修改 HTML 並外加 EXT 的範本套件所建構而成，用以提供使用者一個可以簡單下達 SPARQL 指令的操作介面，該介面提供了標籤頁面的分類，CheckBox 的問題選擇，部份再適度加

入 Text 方塊供使用者自訂欲搜尋的可能關鍵字後，再透過該介面後端的程式輔以轉換成 SPARQL 語法進行 Submit 送出所想要查詢的規則與關聯，最後再由 Server 後端進行推論處理將其結果回傳至 Client 端以供使用者閱覽。

- **查詢系統模組：**在本文中的查詢系統模組是經由 Jena、Joseki 與 ARQ 所組成，其用途說明如下：
 - *Programmatic Environment for RDF (Jena)*：此為 RDF 的推論引擎，除了具知識本體推論外，本身也有自己的規則 (Rule) 推論引擎；但在本文中，Jena 僅提供一個 RDF 存取的環境，而簡單易於操作的 Joseki 架構其上進行推論使用。
 - *RDF Server (Joseki)*：此為 RDF 伺服器端的推論引擎，主要建立在 Jena 的操作環境上，它支援 SPARQL 語言的使用，為簡易型的推論引擎。它可被安裝在任何的 Servlet Container 或 Web Application Server 環境下使用，像 Jetty 或 Tomcat 環境皆可。
 - *SPARQL Implementing Program (ARQ)*：此為本文在使用推論引擎進行推論時會使用到其 SPARQL 的函式庫，它建構在 Jena 的操作環境上，並支援 request 或 bug reports。
- **會議紀錄知識本體：**此為本系統的核心知識庫載體，我們利用美國史丹佛大學醫學資訊中心 (Stanford Medical Informatics, SMI) 所研究開發的知識庫自由軟體 Protégé 來建置完成。該會議紀錄知識本體是由「主題分類」、「提案單位」、「提案型態」、「會議地點」、「會議時間」、「會議種類資訊」等六大類別所組成。

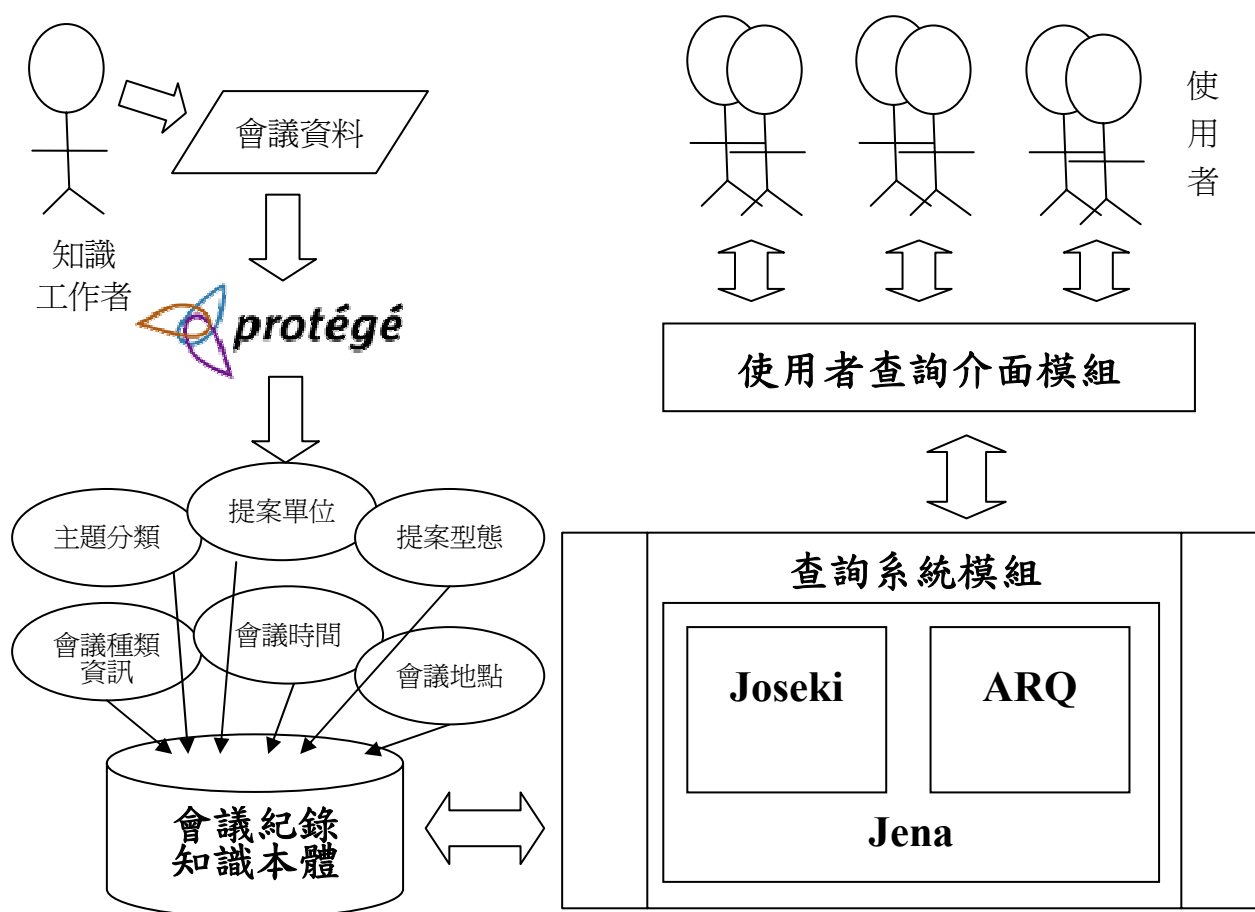


圖 3-1 SQSCMR 之系統架構

第二節 會議紀錄知識本體架構

本系統之會議紀錄知識本體的架構，是經由我們與使用者訪談過了其所需資訊後，將其知識本體設計區分為「主題分類」、「提案單位」、「提案型態」、「會議地點」、「會議時間」、「會議種類資訊」等六大類別，並藉由屬性間的交互關聯後，讓使用者能方便且清楚的查詢到所想要的會議紀錄資訊。該會議紀錄知識本體之架構如圖 3-2 所示，在此圖中會議紀錄的六大次類別被詳細描述於下列各小節。

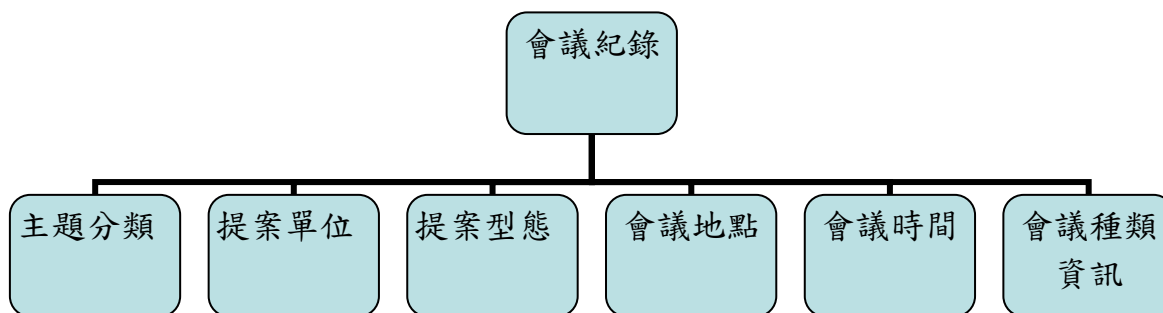


圖 3-2 會議紀錄知識本體架構

壹、主題分類之建構

在會議之主題分類的部份，我們將初步歸類包含有一般事務、人事、法規、經費及重大慶典等 5 種 Instance (例子) 分類，透過這樣的 Instance 分類並進行關聯，以利我們更容易且精準地找到我們想要的案由內容。例如：我們想搜尋有關「法規」方面的案由資訊，依關鍵字的方式對案由進行搜尋是無法較明確地找出一定的規則來進行比對，現行僅能以分類模式並將案由先行關聯分類後，於搜尋時透過關聯解析萃取出我們想要的資訊。我們將範圍縮小依法規相關隨機抽取幾筆法規案由為例：本校組織規程修正報告案；本校「專任教師評鑑辦法」修正案；廢止本校「獎勵優秀師資執行要點」，提請審議；增訂某科技大學執行「弱勢學生助學計畫」實施要點，提請討論；等等案由，各單位對於案由描述的方式無一定準則可依循，故藉由主題分類此一方式進行分類關聯，各項分類其說明如表 3-1 的會議主題分類說明，其架構如圖 3-3 的主題分類基本架構。

表 3-1 主題分類說明

分類	說明
一般事務	有關人車分道之安全問題、增減調整所系科班、財產減損報廢、校園菸害防制計畫、訪視、評鑑及自我評鑑時程規劃...等一般性事務方面主題
人事	有關教職員人事異動、獎懲及約聘僱等方面主題
法規	有關法規新增、修訂、廢止或公布等方面主題
經費	有關預算、教育部獎補助及學校自籌款等方面主題
重大慶典	有關校慶及畢業典禮...等方面主題

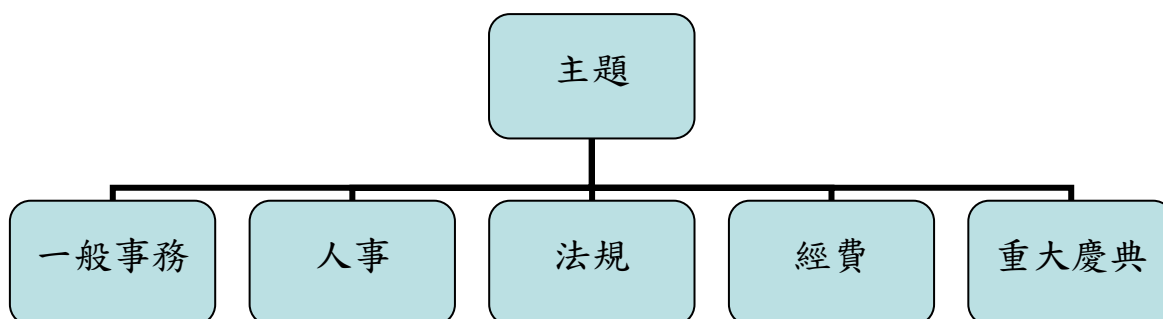


圖 3-3 主題分類基本架構

貳、提案單位之建構

本文對於語意的近義詞、等義詞及相關詞彙之應用，這個部份我們將導入到「提案單位」查詢上。這主要是在初期收集資料匯入時發現，各提案單位會隨著時間的變遷而改變了本身單位的名稱，或少部份的案由資料因先前較沒有明確的定義將以一級單位用詞方式提出，或部份單位因時間變遷而變更原有的名單，甚至兩者合併成一個單位（例如：電算中心+圖書館=圖資處）的用詞等問題。

為了解決上述的問題，我們在定義每項曾在會議紀錄中有出現的提案單位名稱時，除了定義本身單位名稱的詞彙外，皆透過 MRUnitSameAs 屬性的關聯，額外加註了一級單位名稱的近義詞及等義詞等數個單位詞彙，另也加註了改名或合併後的單位名稱之相關詞彙，以利不同的查詢者於查詢歷史記錄時對於當時提案單位用詞的認知不同，以此關聯的方式能進而查詢出所想要查詢的歷史會議紀錄。例如：在透過 MRUnitSameAs 屬性的關聯後，使用者可以用「電子計算機中心、電算中心、圖資處、圖書資訊處」等詞彙在提案單位查詢到「電子計算機中心」的相關資訊。有關「提案單位」語意的近義詞、等義詞及相關詞彙的描述顯示於表 3-2。

表 3-2 提案單位近義詞、等義詞及相關詞彙的描述

屬性(MRUnitSameAs)	值(Value)
教務處	教務處;
學務處	學務處;學生事務處;訓導處
總務處	總務處;
技研處	技研處;研發處;技術研究發展處;研究發展處;
電子計算機中心	電子計算機中心;電算中心;圖資處;圖書資訊處;
進修部	進修部;夜間部
進修學院暨進修專校	進修學院暨進修專校;進修專校;進專
進修學院	進修學院;進院;進修學院推廣教育中心;
圖書館	圖書館;圖資處;圖書資訊處;圖書館暨藝術中心
招生中心	招生中心;菁英中心;菁英服務中心;
安全工程學群	安全工程學群學群長;安全工程學群長;安全工程學群;安全學群與工程學群;安全工程學院
安全工程學院	安全工程學群學群長;安全工程學群長;安全工程學群;安全工程學院;安全學群與工程學群;工程學院籌備處與安全學院籌備處
組織與法規小組	組織與法規小組;學務處;學生事務處;
軍訓室	軍訓室;主任教官;教官室;校安中心;

註：以上 Value 乃為本次參考之會議紀錄樣本內所收集出來的值清單列表。

參、提案型態之建構

會議紀錄內容主要是記錄了各項大大小小的報告及提案之案由，本文將這些案由依型態分類後，皆定義在「提案型態」類別之下，每一個案由都清楚地描述所要報告或討論的問題，而我們將這些案由作為「提案型態」下一件一件的例子 (Instance) 或物件 (Object)，分別存在於報告案、提案討論及臨時動議等 3 種不同的提案型態之中，經過日積月累之後嚴然形成一個龐大的知識本體。提案型態類別的說明如表 3-3，其架構如圖 3-4 所示。在圖 3-4 中，SA96-1-01 校務-報告案 01、SA96-1-02 校

務-報告案 01、.....都是報告案型態的 Instance；而提案討論型態的部份則包含有 SA 96-1-01 校務-提案討論 01、SA 96-1-01 校務-提案討論 02 等等；臨時動議型態的部份則有 SA 96-1-01 校務-臨時動議 01 等等，以上數百筆的 Instance 即為提案型態之知識本體。

表 3-3 提案型態說明

型態	說明
報告案	一般已確立定案之報告事項，通常多為告知或指示照辦事項
提案討論	須經討論後再決議之事項或法規等提案
臨時動議	未於有效時間內提出列入本次會議報告或討論之提案，於會議最後臨時提出之提案

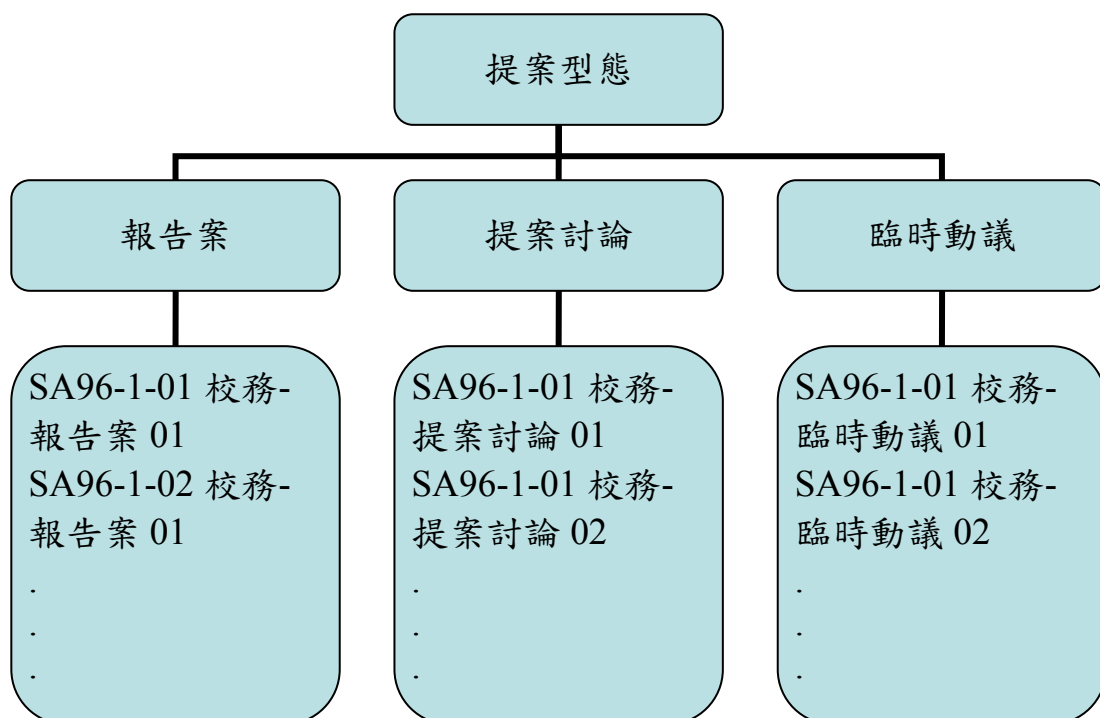


圖 3-4 提案型態基本架構

承上圖 3-4 提案型態基本架構的各項 Instance 裡，我們無法清楚地了解各項 Instance 所代表的意涵。在 Web 的資訊概念裡，在 RDF 中有些資源需要被加以描述才會具有實質上的意義。以「報告案」的 Instance 「SA96-1-01 校務-報告案」來說，這樣的資源，我們並無法了解其資源所代表的意思，更別說電腦如何向使用者解釋其代表的意義。因此，我們必須利用 RDF 的三元組來將它做更進一步的描述。

RDF 三元組的組成分別是「物件-屬性-值」，物件等同是一項資源的主體，每個物件上可以有一或多個屬性，並用以指向另一個資源或一段文字描述。我們以「提案討論」為例來表示其意涵：案由是一個物件也是一項資源的主體，而我們如果想了解更多有關案由此一物件的相關資訊則必須透過屬性的連結來了解該物件的「主旨描述、決議內容、主旨同義詞、主題分類及提案單位」等所謂三元組中的值，其關聯如圖 3-5 案由與其細部描述資訊關聯圖。例如，在圖 3-6（為圖 3-5 之 Instance），我們以「SA96-1-01 校務-報告案 01」實例來描述，並使用了 5 個屬性分別來加以連結，其關聯之說明如下：

1. Proposal01Title 用來連結「SA96-1-01 校務-報告案 01」與「本校組織規程修正報告案。」，由該屬性可進一步了解該實例的主旨詳細內容。
2. Proposal02Resolution 用來連結「SA96-1-01 校務-報告案 01」與「洽悉。」，由該屬性可進一步了解該實例的後續決定議的結果資訊。
3. Proposal03HideKnowle 用來連結「SA96-1-01 校務-報告案 01」與「法規;某科技大學組織規程;」等隱性詞彙，由該屬性可進一步了解該實例的主旨同義詞，用來加強對主旨之隱性詞彙的註解以提升可被查詢的機率。

4. ProposalType-TitleClass 用來連結「SA96-1-01 校務-報告案 01」與「法規」，由該屬性進一步了解該實例的被歸屬何項主題分類之中。
5. ProposalType-Unit 用來連結「SA96-1-01 校務-報告案 01」與「人事室」，由該屬性進一步了解該實例提由哪一個提案單位所提出的案由。以上例子之關聯如圖 3-6 [SA96-1-01 校務-報告案 01]與其細部描述資訊關聯圖所示。

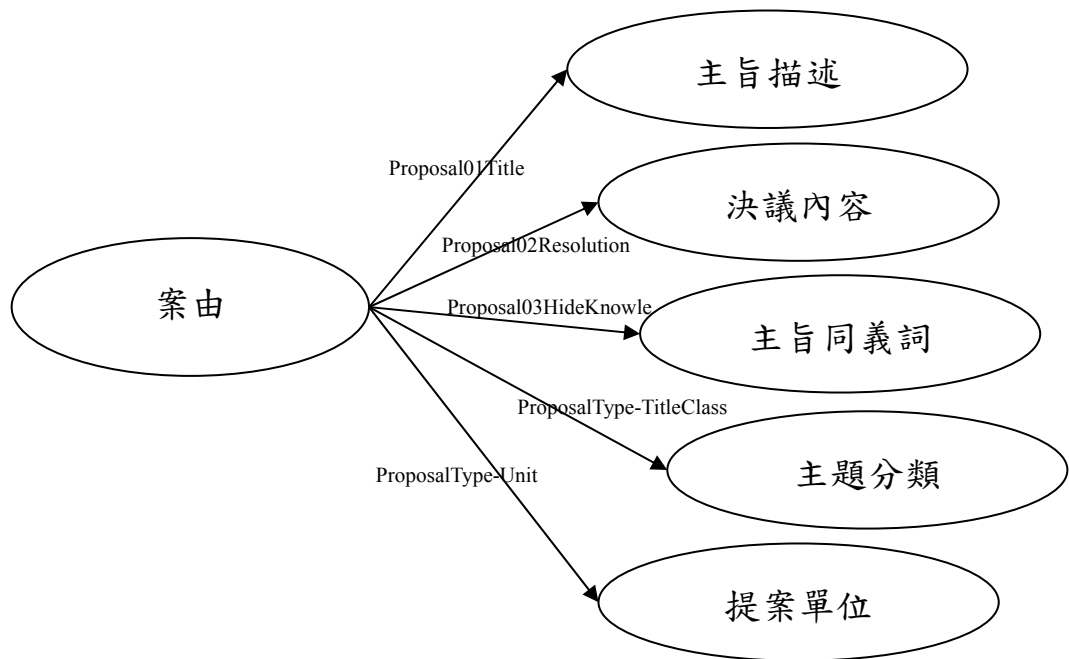


圖 3-5 案由與其細部描述資訊關聯圖

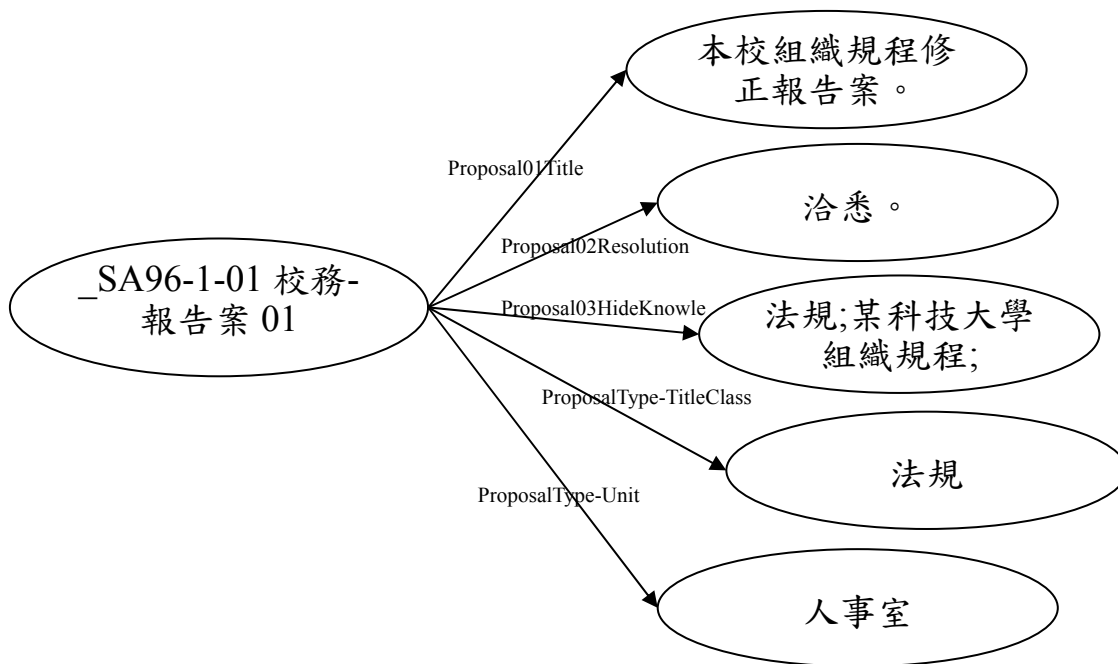


圖 3-6 [SA96-1-01 校務-報告案 01]與其細部描述資訊關聯圖

肆、會議地點之建構

會議地點類別的部份，主要是以大樓的方式來作分類，再依每間會議室為一項 Instance 來建置完整的開會地點，並透過 MP03SameAs 之同義詞的註解屬性提升可被查詢的機率（如表 3-4 所示），最後經由物件屬性的關聯後，即可清楚地了解每場會議開會的地點。從各大樓或會議室的角度來看，能清楚了解各個會議室使用的情形。會議地點的分類架構如圖 3-7 所示。

表 3-4 會議地點近義詞、等義詞及相關詞彙的描述

屬性(MP03SameAs)	值(Value)
文○樓 1F 會議室	文○樓 1F 會議室;文○樓 1 樓會議室;文○樓一 F 會議室;文○樓一樓會議室;
生○樓 9 樓第 3 會議室	生○樓 9 樓第 3 會議室;生○樓 9 樓第三會議室;生○樓九樓第 3 會議室;生○樓九樓第三會議室;
.....

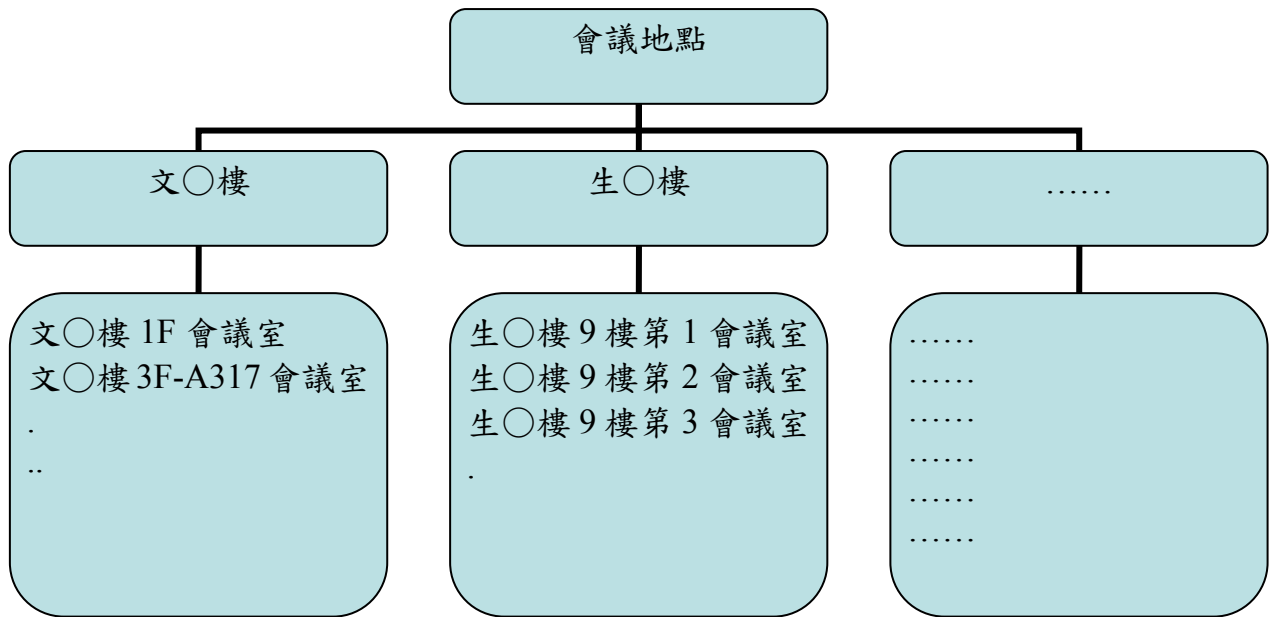


圖 3-7 會議地點基本架構

伍、會議時間之建構

會議時間的類別主要是用來對會議資料做一個較大範圍的時間歸類，此部份的設計除了以「學年度、年度」依不同年份做一群組的歸類外，另將以「某年某月」做一較小範圍的歸類，並針對每個歸類的 Instance 附加 MT03SameAs 之同義詞的註解屬性（如表 3-5 所示），其用途用以查詢時可增加被查詢時可能的機率提升，會議資料總在持續建置日積月累之下形成一個龐大的知識本體。就時間的構面，我們將以此一方式來先行歸類，以利應付日後龐大的知識本體時，查詢者可能只想知道依時間為基準的一個大概的範圍資料時，就可以依此條件將資料限縮索引出精確的一個範圍或一個值來觀看會議資料內容。會議時間的分類架構，如圖 3-8 所示。其中，所謂“學年度”與“年度”之差異，分別為「學年度」係指每年 8 月 1 日起至隔年 7 月 31 日止，「年度」係指每年 1 月 1 日起至該年 12 月 31 日止之時間區段。

表 3-5 會議時間近義詞、等義詞及相關詞彙的描述

屬性(MT03SameAs)	值(Value)
九十七學年度	九十七學年度;97 學年度;九七學年度;(97)學年度;(九十七)學年度;(九七)學年度;
九十七年度	九十七年度;97 年度;九七年度;(97)年度;(九十七)年度;(九七)年度;
九十七年一月	九十七年一月;九七年一月;97 年 1 月;97 年一月;九十七年 1 月;九七年 1 月;97-01;97-1;97/01;97/1;2008-01;2008-1;2008/01;2008/1;二〇〇八年一月;二〇〇八年 1 月;
.....

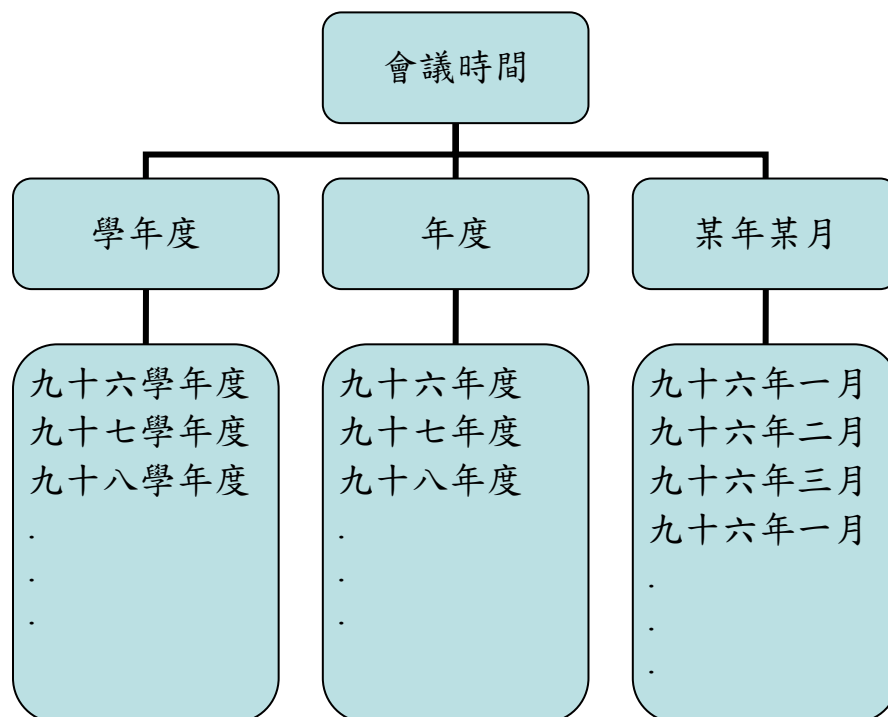


圖 3-8 會議時間基本架構

陸、會議種類資訊之建構

會議種類眾多，本文僅先以兩種校級會議為樣本，進行知識本體的建置，另我們為該會議種類類別的 Instance 透過「註解屬性」額外設計了七項會議紀錄基本資訊內容，用以記載每一筆會議紀錄召開的基本資訊，內容含包有「標題、會議時間、會議地點、主席、出席人員、列席人員及記錄人員」等資訊，以方便使用者進一步了解每場會議實際召開的人、事、時、地等詳細資訊，相關實例如表 3-6 會議種類資訊的描述。在前面各大項所提之各項案由可藉由會議種類資訊的「物件屬性」(如表 3-7)與該項資訊的關聯後進而了解其所屬會議的一些基本資訊。關於會議種類資訊之基本架構，如圖 3-9 所示。

表 3-6 會議種類資訊的描述

註解屬性	值
MRInf01Titile	九十六學年度第一學期第一次校務會議
MRInf02Time	九十六年九月十九日(星期三)中午十二時十五分整
MRInf03Place	本校生○樓九樓第三視聽室
MRInf04Chairman	校長
MRInf05Attend	全體一級主管 34 人、教師代表 45 人、職工代表 4 人及學生代表 9 人(如簽到單)
MRInf06Nonvoting	無
MRInf07Record	記錄人員

表 3-7 會議種類資訊的物件屬性之 Domain 及 Range 的描述

Object Properties	Domain	Range
TypeInf-Place	會議種類資訊	會議地點
TypeInf-ProposalType	會議種類資訊	提案型態

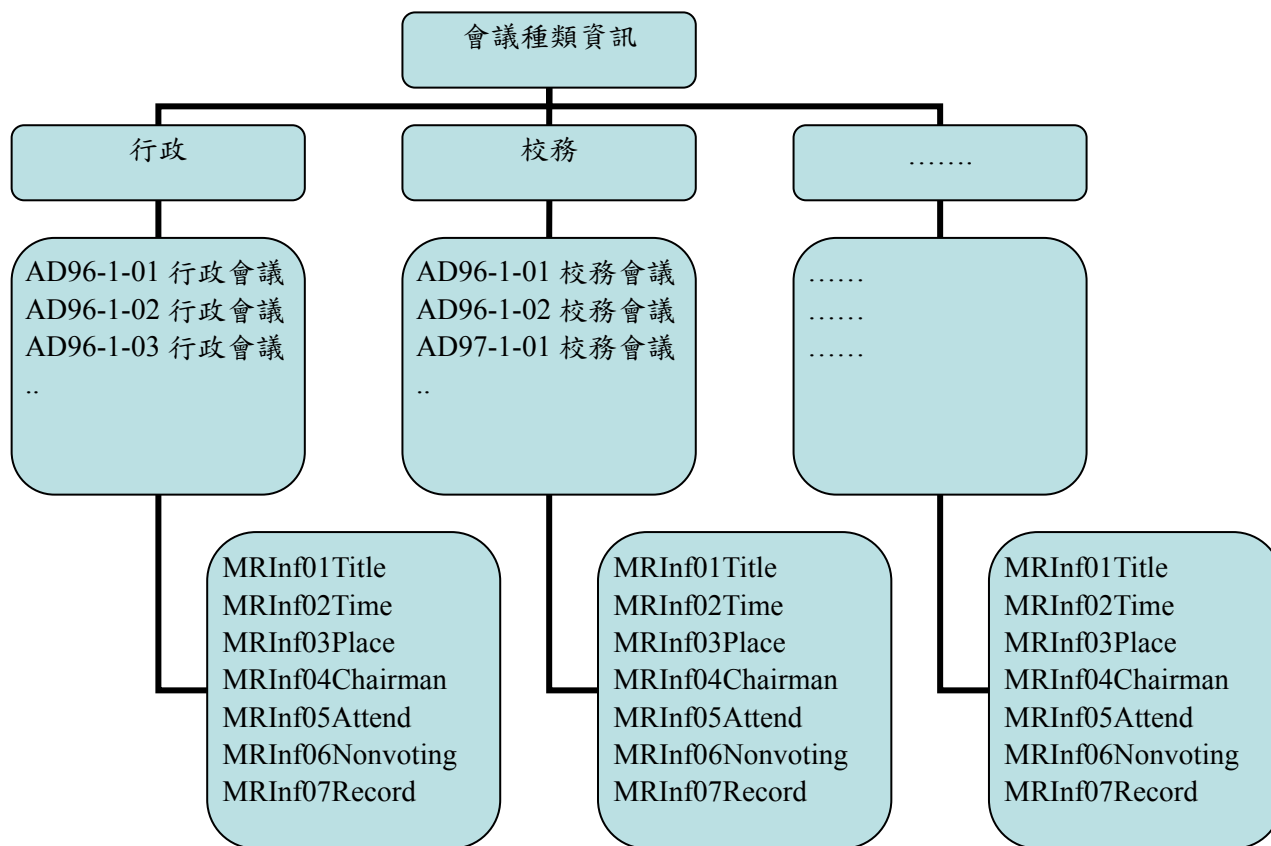


圖 3-9 會議種類資訊基本架構

綜合上述六大類，我們將會議紀錄查詢系統之完整的知識本體概念關係，呈現於圖 3-10；而各類別的物件屬性關係，則顯示於表 3-8；各註解屬性的描述，如表 3-9 所示。

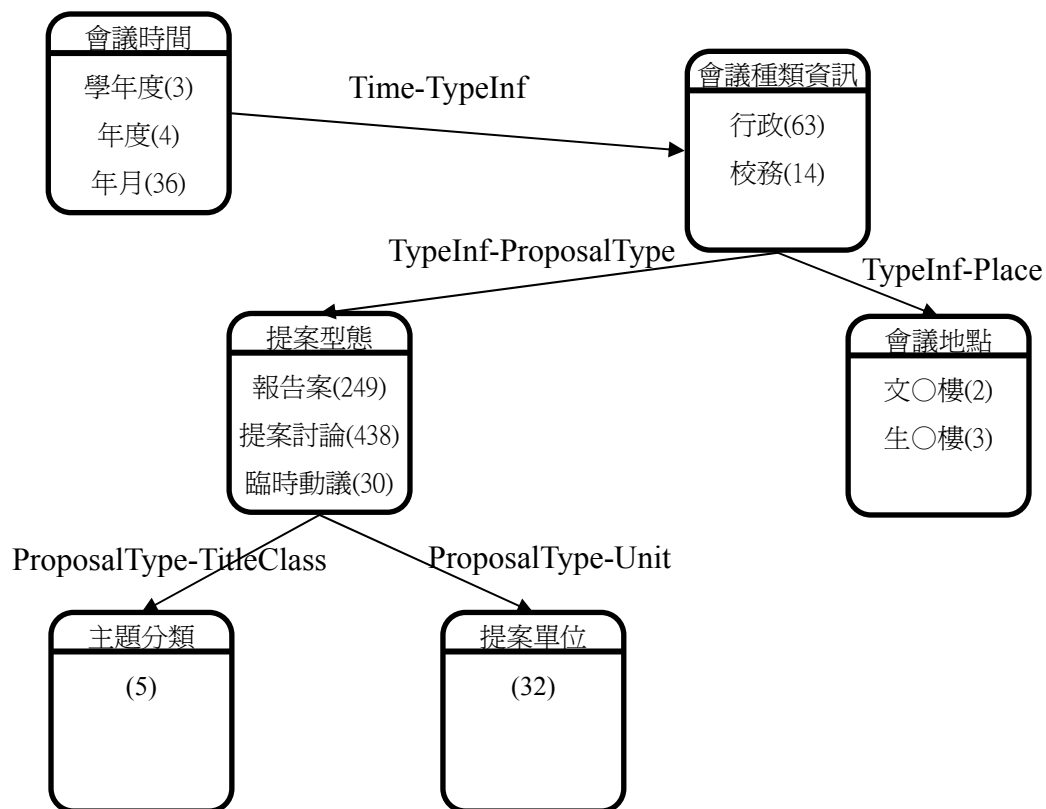


圖 3-10 會議紀錄架構的本體概念關聯圖

表 3-8 物件屬性之 Domain 及 Range 的描述

Object Properties	Domain	Range
Time-TypeInf	會議時間	會議種類資訊
TypeInf-Place	會議種類資訊	會議地點
TypeInf-ProposalType	會議種類資訊	提案型態
ProposalType-TitleClass	提案型態	主題分類
ProposalType-Unit	提案型態	提案單位

表 3-9 註解屬性的描述

Annotation Properties	Describe
MRUnitSameAs	《提案單位》同義詞
MRInf05Attend	《會議種類資訊》出席人員
MRInf03Place	《會議種類資訊》會議地點
MP03SameAs	《會議地點》地點同義詞
Proposal02Resolution	《提案型態》決議
MT02Range	《會議時間》時間範圍
MRInf02Time	《會議種類資訊》會議時間
MT01Name	《會議時間》時間名稱
MRInf04Chairman	《會議種類資訊》主席
MP01Name	《會議地點》地點名稱
MT03SameAs	《會議時間》時間同義詞
MP02Located	《會議地點》地點區域
MRInf01Title	《會議種類資訊》會議名稱
MRInf07Record	《會議種類資訊》會議紀錄人員
Proposal03HideKnowledge	《提案型態》案由同義詞
Proposal01Title	《提案型態》案由
MRInf06Nonvoting	《會議種類資訊》列席人員

第四章、系統實作

在本章，我們實作以知識本體為基礎的大學院校會議紀錄語意檢索系統 (SQSCMR)。我們使用了美國 Stanford 大學醫學資訊中心所開發的 Protégé 3.3.1 軟體來設計與建構會議紀錄知識本體，並配合 Joseki 建置於 Jena 所提供的 RDF 環境裡來使用 RDF 伺服器查詢服務。然而，因使用 SPARQL 語法頗為繁雜，故我們再使用 Dreamweaver cs3 及 Notepad 配合 Html + Ext 語法另外設計了一個簡易的查詢介面，將預先統計後所常用的問句嵌入 SPARQL 語法製作簡易式下拉式選單等介面，以供教職員工查詢所需相關資訊。

在經過前一章節的系統分析與計設並確立所要採用的架構、詞彙、分類及其屬性與關聯後，我們便可開始著手建構本系統。本系統之實作流程如圖 4-1。另外，我們將於系統建構完成後對系統依範例來做實測與修正。

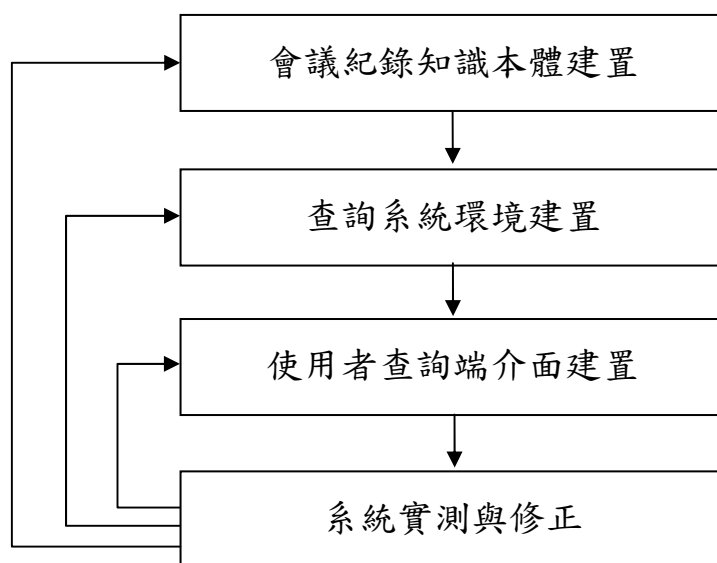


圖 4-1 SQSCMR 之系統實作流程

第一節 系統實作開發環境與工具

本文所開發之 SQSCMR 系統所使用到的開發工具如表 4-1 所示。這些開發工具，我們說明如下：

- **Protégé 3.3.1**(<http://protege.stanford.edu/>)：Protégé 知識本體工具軟體是由美國 Stanford 大學所研究開發的知識庫自由軟體，該工具軟體最大的特色是應用 Ontology 的觀念設計，並提供了 Frame-based 的知識表達方式與一個開發推論引擎的測試環境。此外，它亦加入 OKBC (Open Knowledge Base Connectivity) 模組，讓知識工作者可以建構一個以知識本體為基礎的知識管理系統，增進了對知識檢索的精準度與語意概念的關聯檢索。Protégé 是目前世界上被使用來建置知識本體中頗有名氣的平台之一，也是目前全球支援 Ontology 的平台中極為受歡迎使用的一項免費工具軟體。
- **Jena**：該推論引擎是由 HP Lab 所開發的語意網應用程式，是以 Java 平台所開發的開放式軟體，只需將 Jena Library 加入整合開發環境中，即可執行推論功能，其中 Jena 推論引擎提供許多開發語言，包含 RDF、RDFS、DMAL+OIL、OWL、RDQL、SPARQL 等等。Jena 推論引擎除了具知識本體推論外，另外一大特色就是本身有自己的 Rule 推論引擎，只要依照 Jena 所制定的格式撰寫 Rule，即可使用 Rule 推論引擎（鄭雅仁，2010）。但在本研究中，Jena 僅提供一個 RDF 存取的環境，而簡單易於操作的 Joseki 架構其上進行推論使用。
- **Joseki 3.2**(<http://www.joseki.org>)：此為 RDF 伺服器端的推論引擎，主要建立在 Jena 的操作環境上，它支援 SPARQL 語言的使用，為簡易型的推論引擎，可被安裝在任何的 Servlet Container 或 Web Application Server 環境下使用，像 Jetty 或 Tomcat 環境皆可。

表 4-1 SQSCMR 系統開發工具

系統運行環境項目	開發工具
Ontology 建置工具	Protégé 3.3.1
Server 端	
--RDF 環境	Jena
--RDF Server 推論引擎	Joseki 3.2
--相容於 Jena 的 SPARQL 查詢語言	ARQ 函式庫
Client 端	
--查詢介面	Dreamweaver cs3 及 Notepad + Ext 套件

- **ARQ**：此為本研究在使用推論引擎進行推論時會使用到其 SPARQL 的函式庫，並建構在 Jena 的操作環境上，並支援 Request 或 Bug Reports。

第二節 建構會議紀錄知識本體

本系統使用了一套頗有名氣的免費知識本體建構工具 Protégé 3.3.1，其主要的用途是用來建構各式各樣的知識本體。我們將建置的方式與步驟說明如下：

1. 執行已安裝完成的 Protégé 3.3.1，接著開啟一個新的專案（New Project），如圖 4-2 所示。
2. 選擇要創建的專案類型格式，在此本研究將以 OWL/RDF Files 類型建置，如圖 4-3 所示。
3. 創建完成後，即進行本專案的工作視窗，使用者的操作介面如圖 4-4 所示。

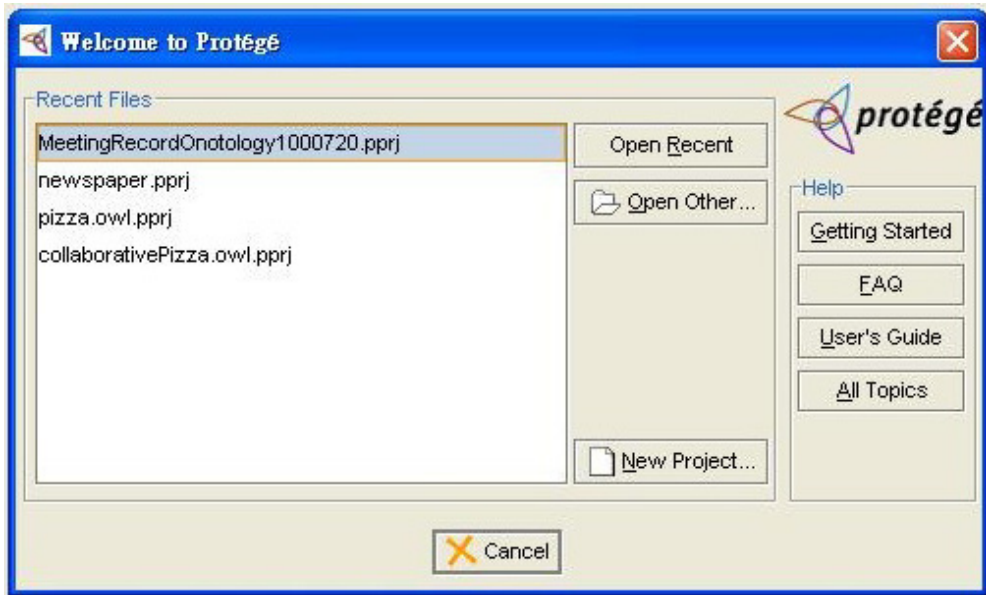


圖 4-2 建立一份 Protégé 3.3.1 新的專案

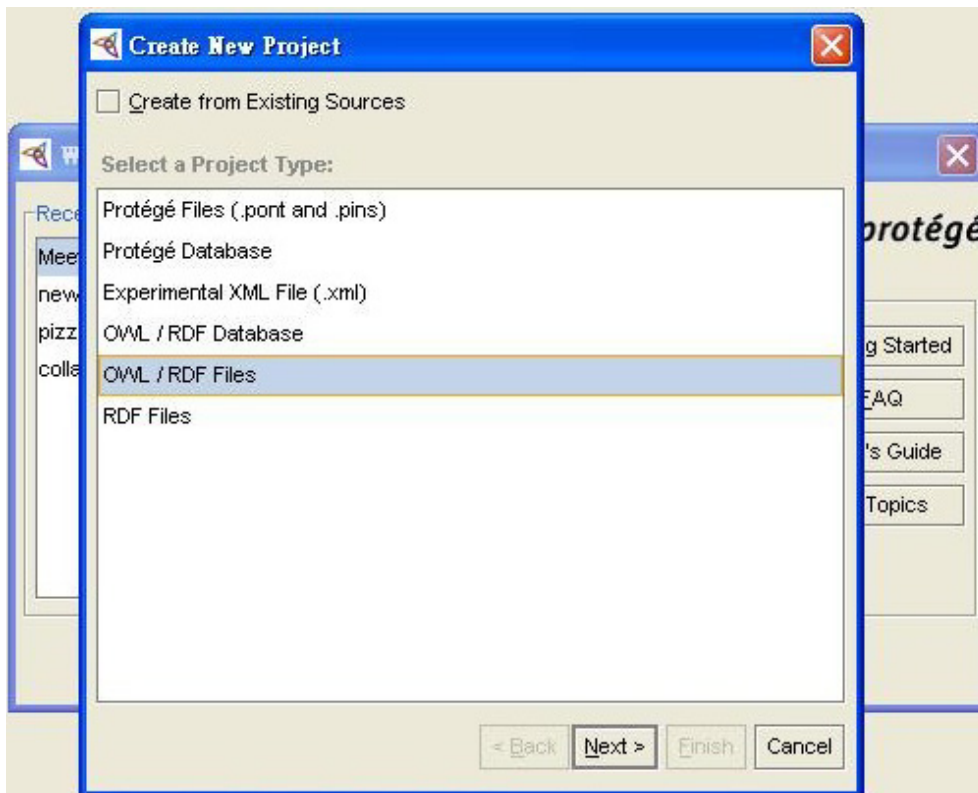


圖 4-3 建立新專案類型為 OWL/RDF Files

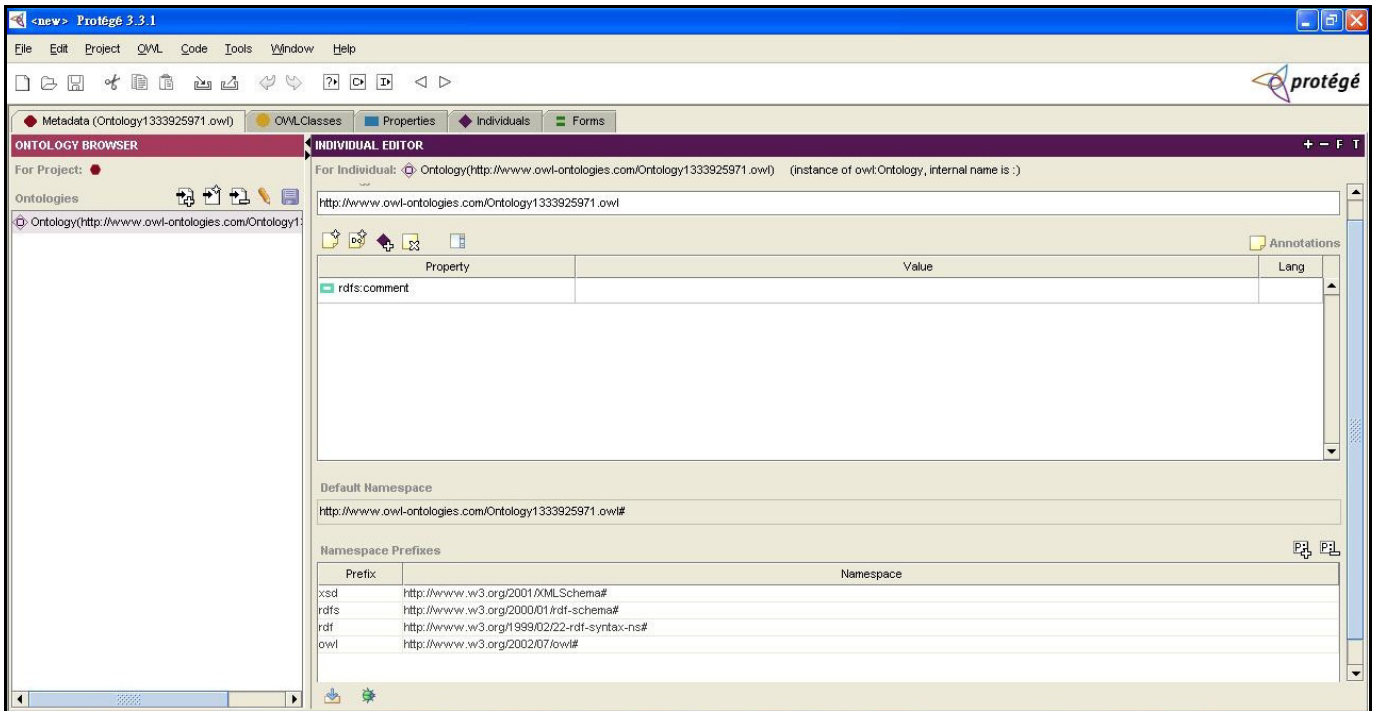


圖 4-4 Protégé 3.3.1 環境視窗介面

在圖 4-4，使用者的操作介面的工作視窗共有 5 個標籤頁，分別為：Metadata、OWL Classes、Properties、Individuals 及 Forms。我們簡述如下：

- *Metadata*：為本專案要定義的 Namespace、及 Namespace Prefixes 的宣告。
- *OWL Classes*：為本專案的 Classes 設計。
- *Properties*：為本專案的 Properties 設計及 Properties 本身的 Domain 與 Range 的規範。
- *Individuals*：為本專案的所有 Instance 依 Classes 分類與 Properties 規範皆將其細部資訊建置於此。
- *Forms*：為本專案的 Individuals 在建置資訊時的每一個 Classes 版面設計。

第三節 建構查詢服務系統環境

本文所使用的查詢服務系統環境，將採用 Joseki 網站(<http://www.joseki.org/>)所提供的 Joseki - The Jena RDF Server 來建置 RDF 伺服器的查詢服務環境，而在該網站所下載的 Joseki 套件本身就將 Jena 和 ARQ 等相關環境與套件涵蓋進來，所以我們只要下載後安裝，並做第 1 次系統環境變數設定即可使用，我們所使用的版本為 Joseki 3.2 版，其詳細安裝與設定之操作說明如下：

1. 至 Joseki 網站(<http://www.joseki.org/>)下載檔案後，並解壓縮至 C:\環境下(我們安裝後的目錄為 C:\joseki)。
2. 第 1 次系統環境變數設定，依序至作業系統=>控制台=>系統=>進階=>環境變數裡，新增一筆變數名稱：JOSEKIROOT 與變數值：C:\JOSEKI 後按確定即可生效，設定畫面如圖 4-5。
3. 執行 RDF Server 服務，開啟作業系統的命令提示字元功能後，至該安裝目錄下的 BIN 目錄 (C:\JOSEKI\BIN)，依序執行 JOSEKI_PATH 與 RDFSERVER 兩個批次檔即可啟動 RDF Server 運行服務，操作畫面如圖 4-6，若啟動正常無誤將會顯示 JOSEKI 服務(Port:2020)啟動成功畫面如圖 4-7。
4. 環境測試，開啟作業系統內的瀏覽器功能，並輸入 <http://127.0.0.1:2020/>，若系統正常運行即可看到如圖 4-8 畫面。
5. 環境查詢功能測試，在圖 4-8，點選 Query forms 的 *Form for SPARQL queries on a small books database* 可看到內建範本查詢測試如圖 4-9，按下 Get Results 後，即可看到查詢結果如圖 4-10，此即表示已可正常提供服務。

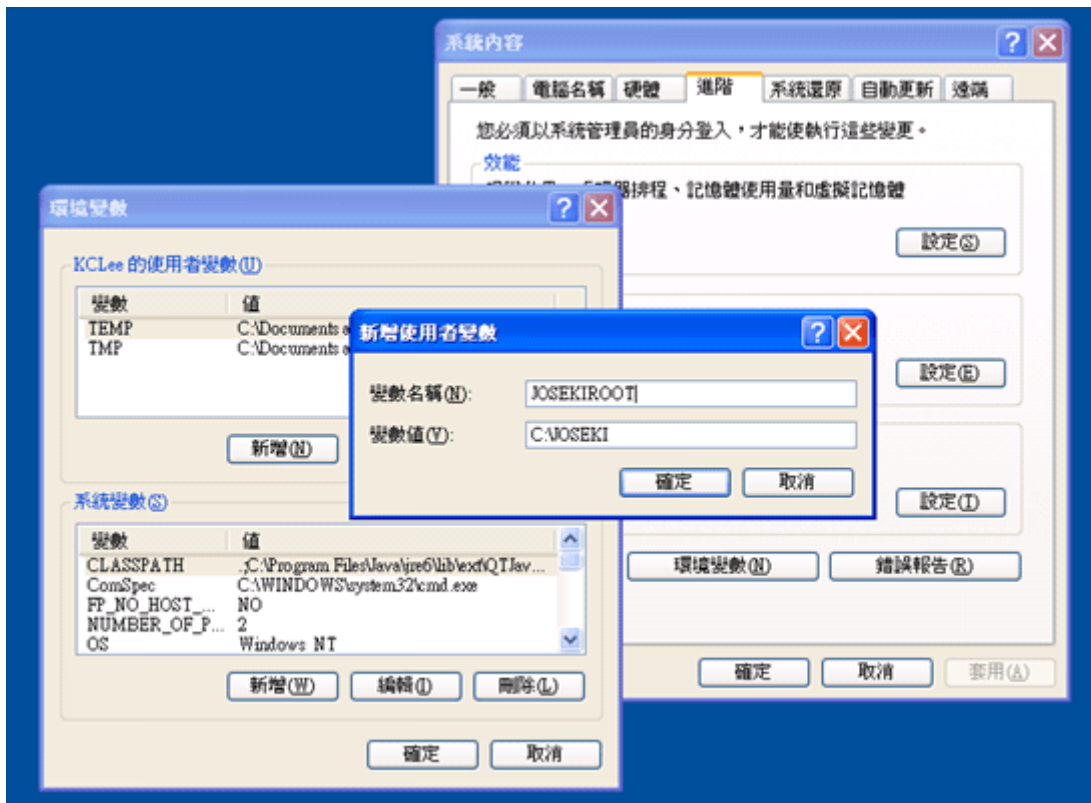


圖 4-5 設定查詢服務系統的 JOSEKI 環境變數

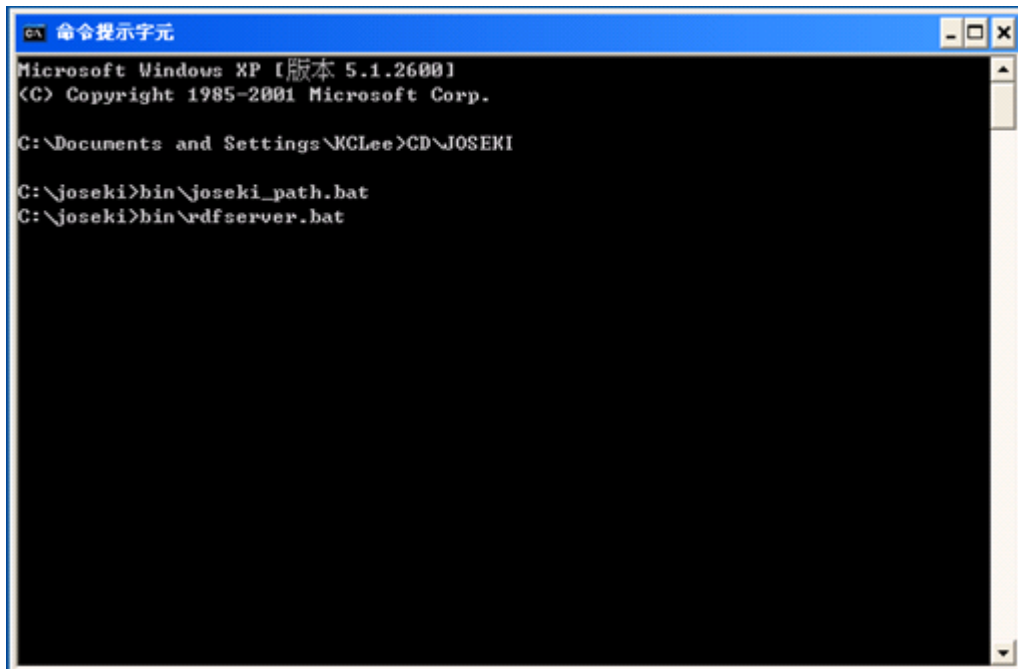
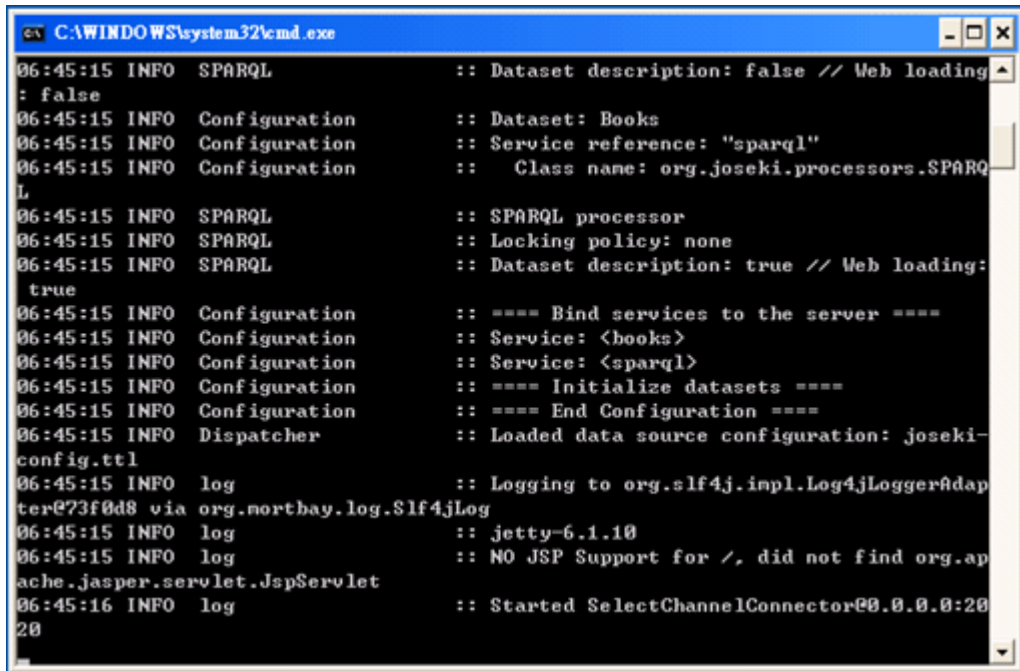


圖 4-6 使用命令提示字元視窗啟動 JOSEKI 的服務



```
C:\WINDOWS\system32\cmd.exe
06:45:15 INFO SPARQL :: Dataset description: false // Web loading
: false
06:45:15 INFO Configuration :: Dataset: Books
06:45:15 INFO Configuration :: Service reference: "sparql"
06:45:15 INFO Configuration :: Class name: org.joseki.processors.SPARQL
06:45:15 INFO SPARQL :: SPARQL processor
06:45:15 INFO SPARQL :: Locking policy: none
06:45:15 INFO SPARQL :: Dataset description: true // Web loading:
true
06:45:15 INFO Configuration :: ==== Bind services to the server ====
06:45:15 INFO Configuration :: Service: <books>
06:45:15 INFO Configuration :: Service: <sparql>
06:45:15 INFO Configuration :: ==== Initialize datasets ====
06:45:15 INFO Configuration :: ==== End Configuration ====
06:45:15 INFO Dispatcher :: Loaded data source configuration: joseki-
config.ttl
06:45:15 INFO log :: Logging to org.slf4j.impl.Log4jLoggerAdap
ter@73f0d8 via org.northbay.log.Slf4jLog
06:45:15 INFO log :: jetty-6.1.10
06:45:15 INFO log :: NO JSP Support for /, did not find org.ap
ache.jasper.servlet.JspServlet
06:45:16 INFO log :: Started SelectChannelConnector@0.0.0.0:20
20
```

圖 4-7 JOSEKI 服務(Port:2020)啟動成功的畫面

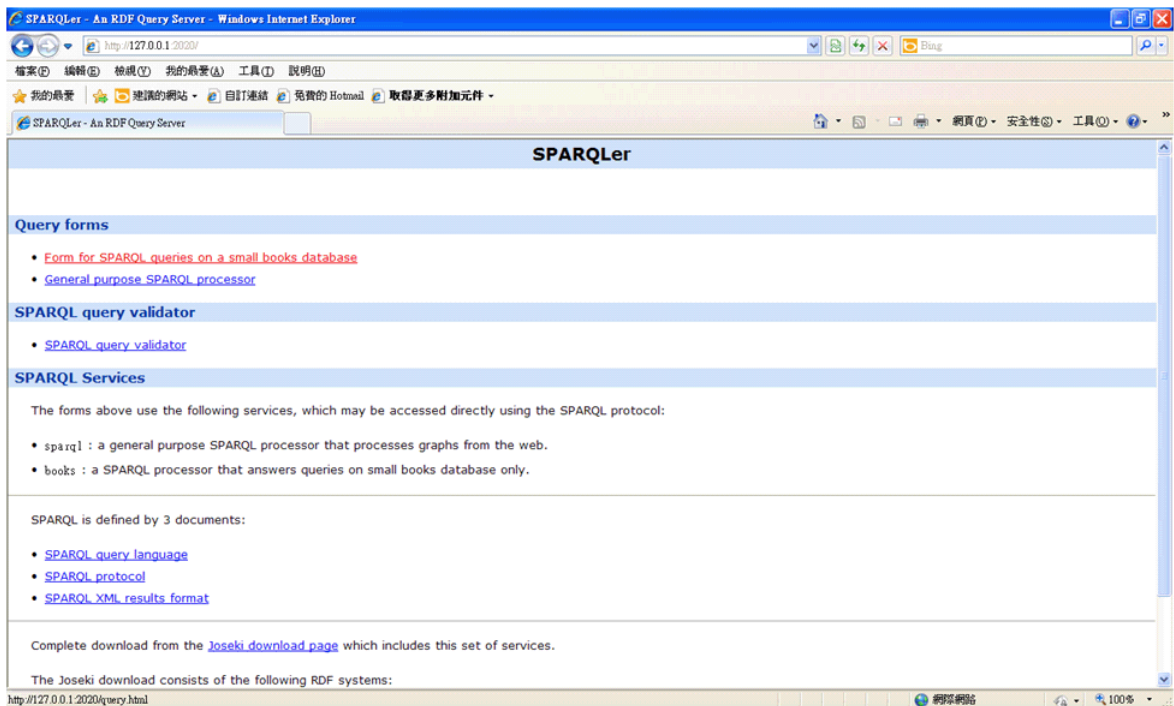


圖 4-8 以瀏覽器檢視 Server 首頁運作成功的畫面

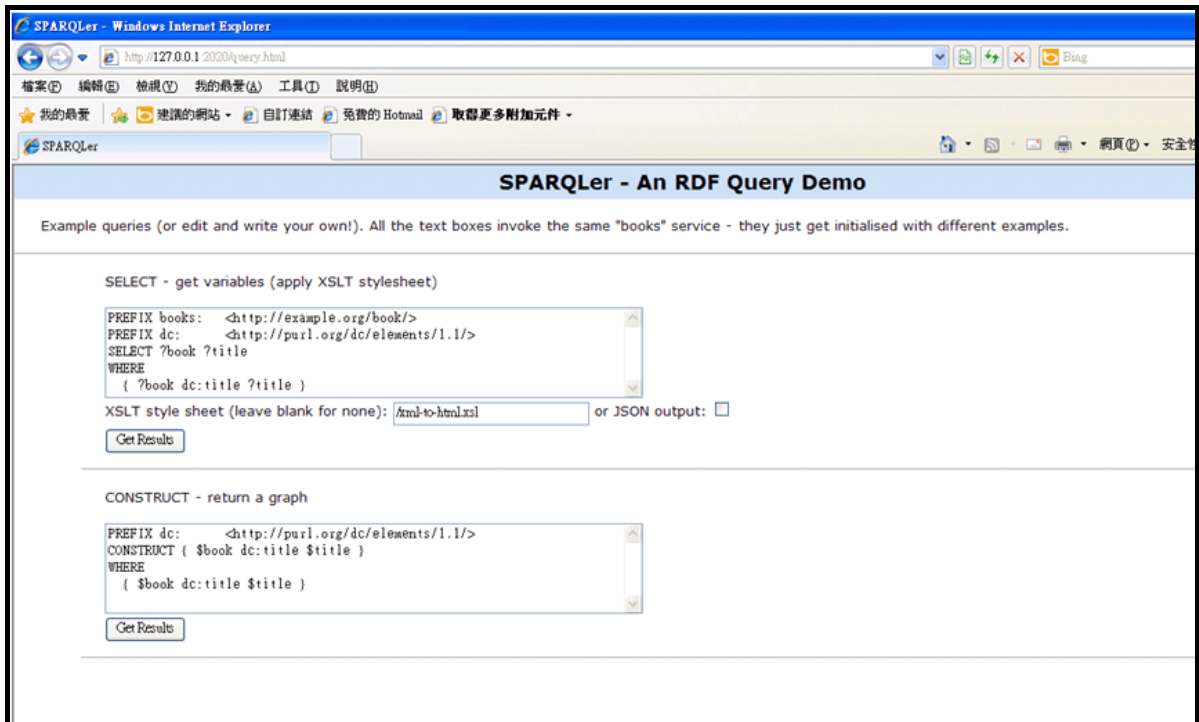


圖 4-9 以瀏覽器檢視 SPARQL 預設查詢介面

SPARQLer Query Results

book	title
<http://example.org/book/book7>	"Harry Potter and the Deathly Hallows"
<http://example.org/book/book2>	"Harry Potter and the Chamber of Secrets"
<http://example.org/book/book4>	"Harry Potter and the Goblet of Fire"
<http://example.org/book/book6>	"Harry Potter and the Half-Blood Prince"
<http://example.org/book/book1>	"Harry Potter and the Philosopher's Stone"
<http://example.org/book/book3>	"Harry Potter and the Prisoner Of Azkaban"
<http://example.org/book/book5>	"Harry Potter and the Order of the Phoenix"

圖 4-10 SPARQL 的查詢結果

第四節 建構查詢服務系統知識本體

大學院校會議紀錄知識本體之建置方法，我們以 Protégé 建立類別示範，如圖 4-11 所示。我們建立物件屬性示範(如圖 4-12)並參考表 4-2 物件屬性之 Domain 及 Range 的描述以及建立註解屬性示範(如圖 4-13)並參考表 4-3 註解屬性的描述。另外，圖 4-14 為建立實例之示範。

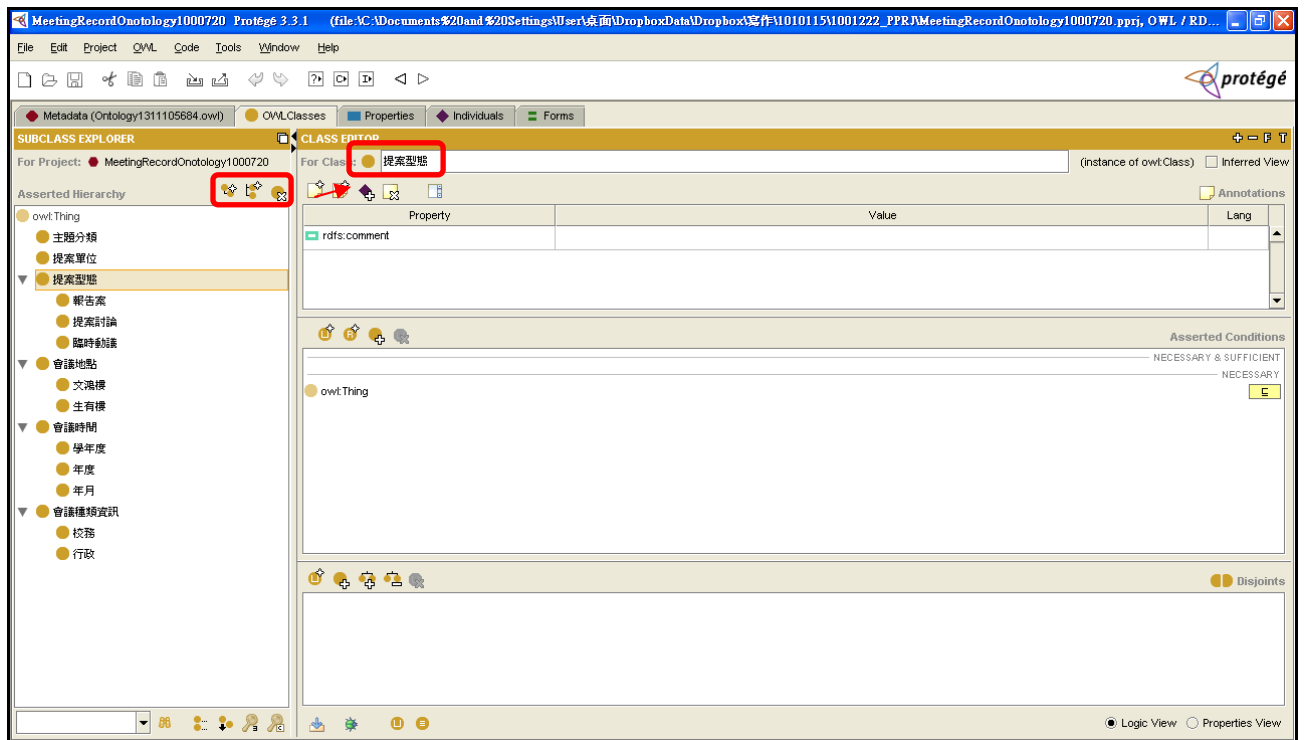


圖 4-11 建立類別

表 4-2 物件屬性之 Domain 及 Range 的描述

Object Properties	Domain	Range
Time-TypeInf	會議時間	會議種類資訊
TypeInf-Place	會議種類資訊	會議地點
TypeInf-ProposalType	會議種類資訊	提案型態
ProposalType-TitleClass	提案型態	主題分類
ProposalType-Unit	提案型態	提案單位

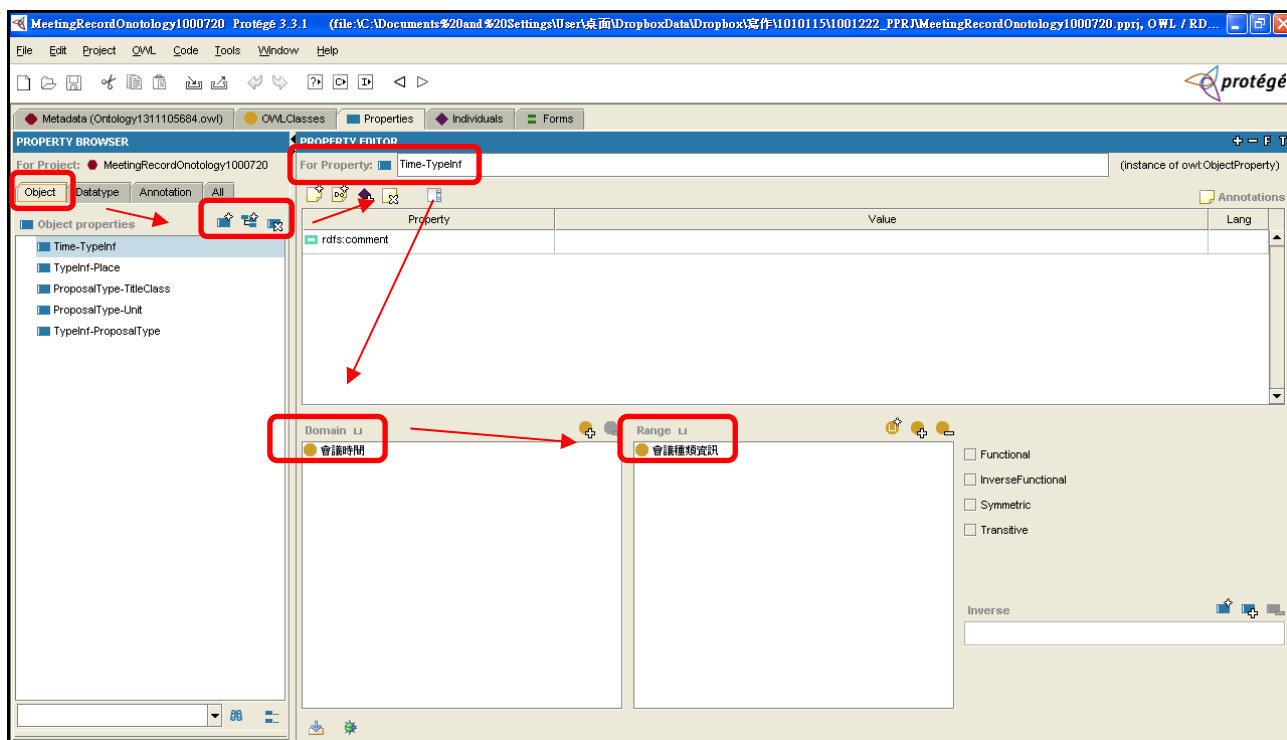


圖 4-12 建立物件屬性

表 4-3 註解屬性的描述

Annotation Properties	Describe
MRUnitSameAs	《提案單位》同義詞
MRInf05Attend	《會議種類資訊》出席人員
MRInf03Place	《會議種類資訊》會議地點
MP03SameAs	《會議地點》地點同義詞
Proposal02Resolution	《提案型態》決議
MT02Range	《會議時間》時間範圍
MRInf02Time	《會議種類資訊》會議時間
MT01Name	《會議時間》時間名稱
MRInf04Chairman	《會議種類資訊》主席
MP01Name	《會議地點》地點名稱
MT03SameAs	《會議時間》時間同義詞
MP02Located	《會議地點》地點區域
MRInf01Title	《會議種類資訊》會議名稱
MRInf07Record	《會議種類資訊》會議紀錄人員
Proposal03HideKnowledge	《提案型態》案由同義詞
Proposal01Title	《提案型態》案由
MRInf06Nonvoting	《會議種類資訊》列席人員

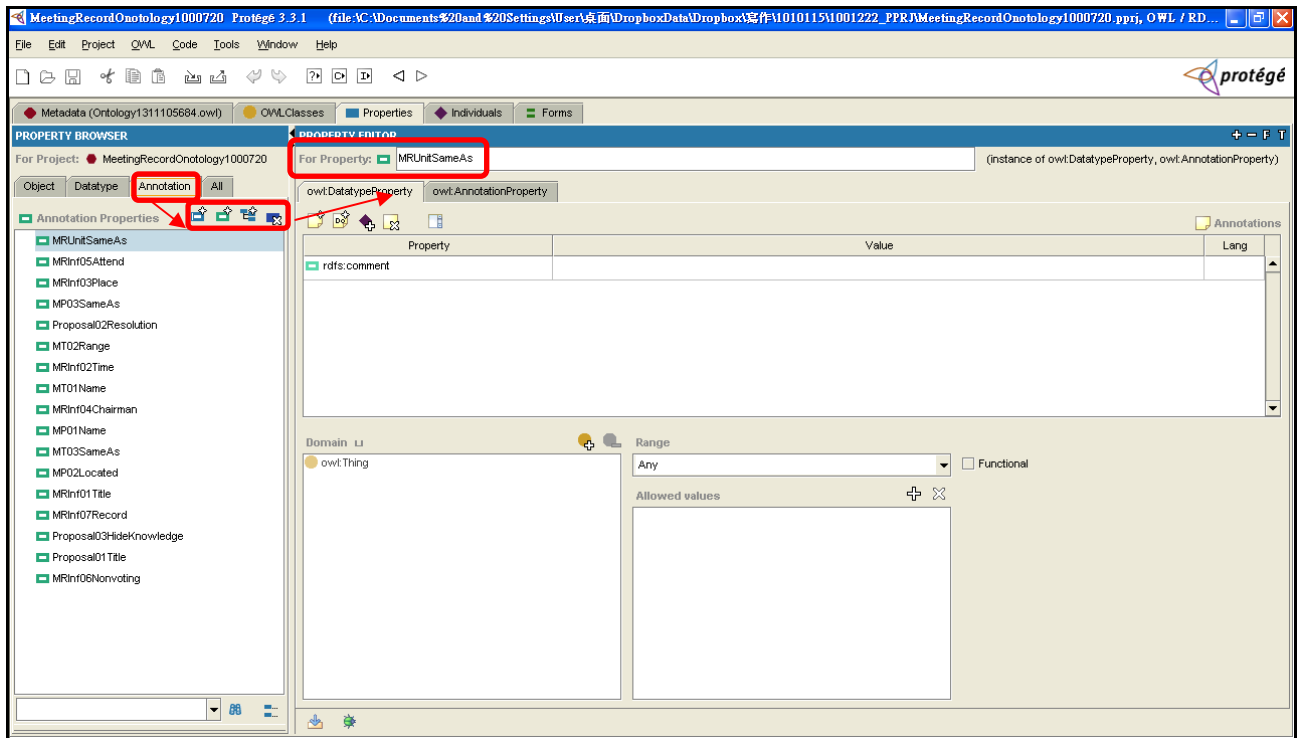


圖 4-13 建立註解屬性

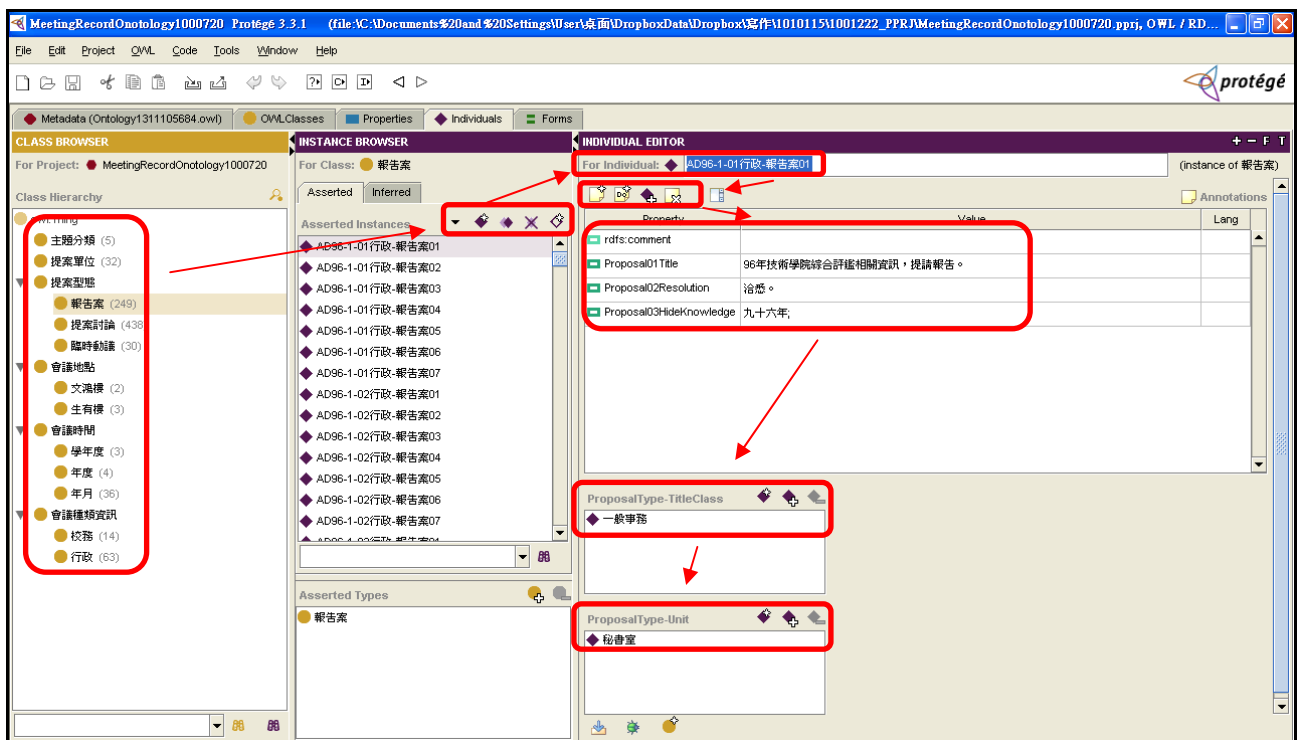


圖 4-14 建立實例

第五節 使用者查詢介面建置

第三節的圖 4-9 雖已有了簡易的查詢功能，但有鑑於並不是每一個人都對於 SPARQL 的語法有所了解。因此，為了提高使用者的使用率及方便操作，本文將以日常查詢者對於管理者所提出的常見的查詢問題進行開發設計，並配合 Joseki 的推論引擎進行查詢服務，其使用者簡易查詢介面如圖 4-15。本介面共分為三個標籤頁，其功能分別為簡易式、簡易式語意及複合式語意查詢服務，總共有 13 個問句範例，後續我們將依序說明各查詢問句的操作與指令。



圖 4-15 SQSCMR 使用者簡易查詢介面

壹、簡易式查詢—依「主題」查詢

依「主題」來查詢，並顯示「提案型態及案由、決定議」內容，其使用者操作畫面如圖 4-16 所示。我們所使用到的 SPARQL 語法如表 4-4，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「主題分類」、「提案型態」、「案由」、「決定議」，即代表依條件限制查詢後欲列出「主題分類」及關聯「提案型態」下「案由」、「決定議」之關聯內容。
- 第 7、12 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10 行分別為相互關聯過濾出符合資訊內容。第 8 行「提案型態」透過 Object properties 與「主題分類」下 Instance 化名為「主題分類」關聯，關聯屬性為 default:ProposalType-TitleClass，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行「提案型態」透過 Annotation Properties 與自己本身化名為「案由」關聯，關聯屬性為 default:Proposal01Title。第 10 行「提案型態」透過 Annotation Properties 與自己本身化名為「決定議」關聯，關聯屬性為 default:Proposal02Resolution 等以上之條件限制。
- 第 11 行為過濾出符合「主題分類」限制的條件，"" 表示條件為空值（不限制）。
- 第 13 行為排序方式，將查詢後欲顯示的資料依「主題分類」以升冪方式排序顯示。



圖 4-16 簡易式查詢—依「主題」查詢介面

表 4-4 依「主題」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?主題分類 ?提案型態 ?案由 ?決定議
7	WHERE {
8	?提案型態 default:ProposalType-TitleClass ?主題分類.
9	?提案型態 default:Proposal01Title ?案由.
10	?提案型態 default:Proposal02Resolution ?決定議.
11	FILTER Regex(str(?主題分類), "").
12	}
13	ORDER BY ASC(?主題分類)

貳、簡易式查詢—依「提案型態」查詢

依「提案型態」來查詢，並顯示「提案型態、案由、決定議及會議名稱」內容，其使用者操作畫面如圖 4-17 所示。我們所使用到的 SPARQL 語法如表 4-5，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「提案型態」、「案由」、「決定議」、「會議名稱」，即代表依條件限制查詢後欲列出「提案型態」及關聯自己部份「案由」、「決定議」及關聯「會議種類資訊」下「會議名稱」之關聯內容。
- 第 7、13 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11 行分別為相互關聯過濾出符合資訊內容。第 8 行：「提案型態」透過 Annotation Properties 與自己本身化名為「案由」關聯，關聯屬性為 default:Proposal01Title。第 9 行：「提案型態」透過 Annotation Properties 與自己本身化名為「決定議」關聯，關聯屬性為 default:Proposal02Resolution。第 10 行：「會議種類資訊」透過 Object properties 與「提案型態」下 Instance 化名為「提案型態」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 11 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「會議名稱」關聯，關聯屬性為 default:MRInf01Title 等以上之條件限制。
- 第 12 行為過濾出符合「提案型態」限制的條件，"" 表示條件為空值（不限制）。
- 第 14 行為排序方式，將查詢後欲顯示的資料依「提案型態」以升冪方式排序顯示。



圖 4-17 簡易式查詢—依「提案型態」查詢介面

表 4-5 依「提案型態」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?提案型態 ?案由 ?決定議 ?會議名稱
7	WHERE {
8	?提案型態 default:Proposal01Title ?案由.
9	?提案型態 default:Proposal02Resolution ?決定議.
10	?會議種類資訊 default:TypeInf-ProposalType ?提案型態.
11	?會議種類資訊 default:MRInf01Title ?會議名稱.
12	FILTER Regex(str(?提案型態),"").
13	}
14	ORDER BY ASC(?提案型態)

參、簡易式查詢—依「提案單位」查詢

依「提案單位」來查詢，並顯示「提案單位、案由、決定議、會議名稱」內容，其使用者操作畫面如圖 4-18 所示。我們所使用到的 SPARQL 語法如表 4-6，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「提案單位」、「案由」、「決定議」、「會議名稱」，即代表依條件限制查詢後欲列出「提案單位」及關聯「提案型態」下「案由」、「決定議」及關聯「會議種類資訊」下「會議名稱」之關聯內容。
- 第 7、14 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12 行分別為相互關聯過濾出符合資訊內容。第 8 行：「提案型態」透過 Object properties 與「提案單位」下 Instance 化名為「提案單位」為關聯，關聯屬性為 default:ProposalType-Unit，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「提案型態」透過 Annotation Properties 與自己本身化名為「案由」關聯，關聯屬性為 default:Proposal01Title。第 10 行：「提案型態」透過 Annotation Properties 與自己本身化名為「決定議」關聯，關聯屬性為 default:Proposal02Resolution。第 11 行：「會議種類資訊」透過 Object properties 與「提案型態」下 Instance 化名為「提案型態」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 12 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「會議名稱」關聯，關聯屬性為 default:MRInf01Title 等以上之條件限制。
- 第 13 行為過濾出符合「提案單位」限制的條件，"" 表示條件為空值（不限制）。
- 第 15 行為排序方式，將查詢後欲顯示的資料依「提案單位」以升冪方式排序顯示。



圖 4-18 簡易式查詢—依「提案單位」查詢介面

表 4-6 依「提案單位」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?提案單位 ?案由 ?決定議 ?會議名稱
7	WHERE {
8	?提案型態 default:ProposalType-Unit ?提案單位.
9	?提案型態 default:Proposal01Title ?案由.
10	?提案型態 default:Proposal02Resolution ?決定議.
11	?會議種類資訊 default:TypeInf-ProposalType ?提案型態.
12	?會議種類資訊 default:MRInf01Title ?會議名稱.
13	FILTER Regex(str(?提案單位), "").
14	}
15	ORDER BY ASC(?提案單位)

肆、簡易式查詢—依「會議時間」查詢

依「會議時間」來查詢，並顯示「時間別、開會時間、會議名稱、會議地點及案例」內容，其使用者操作畫面如圖 4-19 所示。我們所使用的 SPARQL 語法如表 4-7，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「時間別」、「開會時間」、「會議名稱」、「會議地點」及「案例」，即代表依條件限制查詢後欲列出「會議時間」下「時間別」、「案例」及關聯「會議種類資訊」化名為「案例」下「會議名稱」、「開會時間」、「會議地點」之關聯內容。
- 第 7、14 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議時間」透過 Annotation Properties 與自己本身化名為「時間別」關聯，關聯屬性為 default: MT01Name。第 9 行：「會議時間」透過 Object properties 與「會議種類資訊」下 Instance 化名為「案例」關聯，關聯屬性為 default: Time-TypeInf，該 Object properties 將不同區塊的 Class 進行關聯。第 10 行：「案例」透過 Annotation Properties 與自己本身化名為「會議名稱」關聯，關聯屬性為 default: MRInf01Title。第 11 行：「案例」透過 Annotation Properties 與自己本身化名為「開會時間」關聯，關聯屬性為 default: MRInf02Time。第 12 行：「案例」透過 Annotation Properties 與自己本身化名為「會議地點」關聯，關聯屬性為 default: MRInf03Place 等以上之條件限制。
- 第 13 行為過濾出符合「時間別」限制的條件，"" 表示條件為空值（不限制）。
- 第 15 行為排序方式，將查詢後欲顯示的資料依「開會時間」以升冪方式排序顯示，最久時間的資料優先排列。



圖 4-19 簡易式查詢—依「會議時間」查詢介面

表 4-7 依「會議時間」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?時間別 ?開會時間 ?會議名稱 ?會議地點 ?案例
7	WHERE {
8	?會議時間 default:MT01Name ?時間別.
9	?會議時間 default:Time-TypeInf ?案例.
10	?案例 default:MRInf01Title ?會議名稱.
11	?案例 default:MRInf02Time ?開會時間.
12	?案例 default:MRInf03Place ?會議地點.
13	FILTER Regex(str(?時間別),"學年度").
14	}
15	ORDER BY ASC(?開會時間)

伍、簡易式查詢—依「會議地點」查詢之一

依「會議地點」來查詢，並顯示「地點名稱、地點區域」內容，其使用者操作畫面如圖 4-20 所示。我們所使用到的 SPARQL 語法如表 4-8，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「地點名稱」、「地點區域」，即代表依條件限制查詢後欲列出「會議地點」下「地點名稱」、「地點區域」之關聯內容。
- 第 7、11 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9 行分別為相互關聯過濾出符合資訊內容。8 行：「會議地點」透過 Annotation Properties 與自己本身化名為「地點名稱」關聯，關聯屬性為 default:MP01Name。第 9 行：「會議地點」透過 Annotation Properties 與自己本身化名為「地點區域」關聯，關聯屬性為 default:MP02Located 等以上之條件限制。
- 第 10 行為過濾出符合「地點名稱」限制的條件，""表示條件為空值（不限制）。
- 第 12 行為排序方式，將查詢後欲顯示的資料依「地點名稱」以升冪方式排序顯示。



圖 4-20 簡易式查詢—依「會議地點」查詢介面之一

表 4-8 依「會議地點」查詢之 SPARQL 查詢指令之一

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?地點名稱 ?地點區域
7	WHERE {
8	?會議地點 default:MP01Name ?地點名稱.
9	?會議地點 default:MP02Located ?地點區域.
10	FILTER Regexp(str(?地點名稱),"").
11	}
12	ORDER BY ASC(?地點名稱)

陸、簡易式查詢—依「會議地點」查詢之二

依「會議地點」來查詢，並顯示「地點名稱、地點區域、會議時間及會議名稱」內容，其使用者操作畫面如圖 4-21 所示。我們所使用到的 SPARQL 語法如表 4-9，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「地點名稱」、「地點區域」、「會議時間」、「會議名稱」，即代表依條件限制查詢後欲列出「會議地點」下「地點名稱」、「地點區域」及關聯「會議種類資訊」下「會議時間」、「會議名稱」之關聯內容。
- 第 7、14 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議種類資訊」透過 Object properties 與「會議地點」下 Instance 化名為「會議地點」關聯，關聯屬性為 default:TypeInf-Place，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「會議地點」透過 Annotation Properties 與自己本身化名為「地點名稱」關聯，關聯屬性為 default:MP01Name。第 10 行：「會議地點」透過 Annotation Properties 與自己本身化名為「地點區域」關聯，關聯屬性為 default:MP02Located。第 11 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「會議時間」關聯，關聯屬性為 default:MRInf02Time。第 12 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「會議名稱」關聯，關聯屬性為 default:MRInf01Title 等以上之條件限制。
- 第 13 行為過濾出符合「地點名稱」限制的條件，""表示條件為空值（不限制）。
- 第 15 行為排序方式，將查詢後欲顯示的資料依「地點名稱」以升冪方式排序顯示。



圖 4-21 簡易式查詢—依「會議地點」查詢介面之二

表 4-9 依「會議地點」查詢之 SPARQL 查詢指令之二

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?地點名稱 ?地點區域 ?會議時間 ?會議名稱
7	WHERE {
8	?會議種類資訊 default:TypeInf-Place ?會議地點.
9	?會議地點 default:MP01Name ?地點名稱.
10	?會議地點 default:MP02Located ?地點區域.
11	?會議種類資訊 default:MRInf02Time ?會議時間.
12	?會議種類資訊 default:MRInf01Title ?會議名稱.
13	FILTER Regex(str(?地點名稱), "").
14	}
15	ORDER BY ASC(?地點名稱)

柒、簡易式查詢—依「會議種類資訊」查詢

依「會議種類資訊」來查詢，並顯示「MRName、MRTime、MRPlace、MRChairman、MRAttend、MRNonvoting、MRRecord」內容，其使用者操作畫面如圖 4-22 所示。我們所使用到的 SPARQL 語法如表 4-10，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「MRName」、「MRTime」、「MRPlace」、「MRChairman」、「MRAttend」、「MRNonvoting」、「MRRecord」，即代表依條件限制查詢後欲列出「會議種類資訊」下「MRName」、「MRTime」、「MRPlace」、「MRChairman」、「MRAttend」、「MRNonvoting」、「MRRecord」之關聯內容。
- 第 7、16 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12、13、14 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRName」關聯，關聯屬性為 default:MRInf01Title。第 9 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRTime」關聯，關聯屬性為 default:MRInf02Time。第 10 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRPlace」關聯，關聯屬性為 default:MRInf03Place。第 11 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRChairman」關聯，關聯屬性為 default:MRInf04Chairman。第 12 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRAttend」關聯，關聯屬性為 default:MRInf05Attend。第 13 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRNonvoting」關聯，關聯屬性為 default:MRInf06Nonvoting。第 14 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRRecord」關聯，關聯屬性為 default:MRInf07Record 等以上之條件限制。
- 第 15 行為過濾出符合「會議種類資訊」限制的條件，"" 表示條件為空值（不限制）。
- 第 17 行為排序方式，將查詢後欲顯示的資料依「MRTime」以升冪方式排序顯示，最久時間的資料優先排列。



圖 4-22 簡易式查詢—依「會議種類資訊」查詢介面

表 4-10 依「會議種類資訊」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?MRName ?MRTime ?MRPlace ?MRChairman ?MRAttend ?MRNonvoting ?MRRecord
7	WHERE {
8	?會議種類資訊 default:MRInf01Title ?MRName.
9	?會議種類資訊 default:MRInf02Time ?MRTime.
10	?會議種類資訊 default:MRInf03Place ?MRPlace.
11	?會議種類資訊 default:MRInf04Chairman ?MRChairman.
12	?會議種類資訊 default:MRInf05Attend ?MRAttend.
13	?會議種類資訊 default:MRInf06Nonvoting ?MRNonvoting.
14	?會議種類資訊 default:MRInf07Record ?MRRecord.
15	FILTER Regex(str(?會議種類資訊),"").
16	}
17	ORDER BY ASC(?MRTime)

捌、簡易式語意查詢—依「主題(含同義詞應用)」查詢

依「主題」來查詢，並顯示「提案型態及案由、決定議」內容，其使用者操作畫面如圖 4-23 所示。我們所使用到的 SPARQL 語法如表 4-11，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「主題分類」、「案由」、「會議名稱」、「案例」，即代表依條件限制查詢後欲列出「主題分類」、「會議種類資訊」下「會議名稱」及關聯「提案型態」下「案由」、「案例」之關聯內容。
- 第 7、14 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議種類資訊」下 Instance 化名為「會議名稱」透過 Object properties 與「提案型態」下 Instance 化名為「案例」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「案例」透過 Annotation Properties 與自己本身化名為「案由」關聯，關聯屬性為 default:Proposal01Title。第 10 行：「案例」透過 Annotation Properties 與自己本身化名為「PHK」關聯，關聯屬性為 default:Proposal03HideKnowledge。第 11 行：「案例」透過 Object properties 與「主題分類」下 Instance 化名為「主題分類」關聯，關聯屬性為 default:ProposalType-TitleClass，該 Object properties 將不同區塊的 Class 進行關聯並達成以上之條件限制。
- 第 12 行分別為過濾出符合「案由」或「PHK」限制的條件，""表示條件為空值(不限制)，其中，「PHK」該項過濾條件主要針對「案由」以外，另增加「案由同義詞應用」讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率；
- 第 13 行為過濾出符合「主題分類」限制的條件，""表示條件為空值(不限制)。
- 第 15 行為排序方式，將查詢後欲顯示的資料依「案例」以升冪方式排序顯示。



圖 4-23 簡易式語意查詢—依「主題(合同義詞)」查詢介面

表 4-11 依「主題(合同義詞)」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?主題分類 ?案由 ?會議名稱 ?案例
7	WHERE {
8	?會議名稱 default:TypeInf-ProposalType ?案例.
9	?案例 default:Proposal01Title ?案由.
10	?案例 default:Proposal03HideKnowledge ?PHK.
11	?案例 default:ProposalType-TitleClass ?主題分類.
12	FILTER (Regex(str(?案由), "") Regex(str(?PHK), "")).
13	FILTER Regex(str(?主題分類), "").
14	}
15	ORDER BY ASC(?案例)

玖、簡易式語意查詢—依「提案單位(合同義詞應用)」查詢

依「提案單位」來查詢，並顯示「提案單位、案由、決定議、會議名稱」內容，其使用者操作畫面如圖 4-24 所示。我們所使用到的 SPARQL 語法如表 4-12，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「PUnit」、「PTitle」、「PResolution」、「MRName」，即代表依條件限制查詢後欲列出「提案單位」下「PUnit」及關聯「提案型態」下「PTitle」、「PResolution」及「會議種類資訊」下「MRName」之關聯內容。
- 第 7、15 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12、13 行分別為相互關聯過濾出符合資訊內容。第 8 行：「提案型態」透過 Object properties 與「提案單位」下 Instance 化名為「PUnit」關聯，關聯屬性為 default:ProposalType-Unit，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「PUnit」透過 Annotation Properties 與自己本身化名為「UnitSyn」關聯，關聯屬性為 default:MRUnitSameAs，該關聯主要針對「PUnit」導入同義詞應用讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率。第 10 行：「提案型態」透過 Annotation Properties 與自己本身化名為「PTitle」關聯，關聯屬性為 default:Proposal01Title。第 11 行：「提案型態」透過 Annotation Properties 與自己本身化名為「PResolution」關聯，關聯屬性為 default:Proposal02Resolution。第 12 行：「會議種類資訊」透過 Object properties 與「提案型態」下 Instance 化名為「提案型態」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 13 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRName」關聯，關聯屬性為 default:MRInf01Title 等以上之條件限制。
- 第 14 行為過濾出符合「UnitSyn」限制的條件，""表示條件為空值(不限制)，該項(屬性-值)之建構，主要在加強「提案單位」的用字淺詞的部份，故「UnitSyn」為「PUnit」的延申註解。
- 第 16 行為排序方式，將查詢後欲顯示的資料依「PUnit」以升冪方式排序顯示。



圖 4-24 簡易式語意查詢－依「提案單位(合同義詞)」查詢介面

表 4-12 依「提案單位(合同義詞)」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?PUnit ?PTitle ?PResolution ?MRName
7	WHERE {
8	?提案型態 default:ProposalType-Unit ?PUnit.
9	?PUnit default:MRUnitSameAs ?UnitSyn.
10	?提案型態 default:Proposal01Title ?PTitle.
11	?提案型態 default:Proposal02Resolution ?PResolution.
12	?會議種類資訊 default:TypeInf-ProposalType ?提案型態.
13	?會議種類資訊 default:MRInf01Title ?MRName.
14	FILTER Regexp(str(?UnitSyn), "").
15	}
16	ORDER BY ASC(?PUnit)

拾、簡易式語意查詢—依「會議時間(合同義詞應用)」查詢

依「會議時間」來查詢，並顯示「會議時間範圍、開會時間、會議名稱」內容，其使用者操作畫面如圖 4-25 所示。我們所使用到的 SPARQL 語法如表 4-13，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「會議時間」、「開會時間」、「會議名稱」，即代表依條件限制查詢後欲列出「會議時間」下「時間區段」及關聯「會議種類資訊」下「會議名稱」、「開會時間」之關聯內容。
- 第 7、12 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議時間」透過 Annotation Properties 與自己本身化名為「時間區段」關聯，關聯屬性為 default:MT03SameAs，該關聯主要針對「會議時間」導入同義詞應用讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率。第 9 行：「會議時間」透過 Object properties 與「會議種類資訊」下 Instance 化名為「會議名稱」關聯，關聯屬性為 default:Time-TypeInf，該 Object properties 將不同區塊的 Class 進行關聯。第 10 行：「會議名稱」透過 Annotation Properties 與自己本身化名為「開會時間」關聯，關聯屬性為 default:MRInf02Time 等以上之條件限制。
- 第 11 行為過濾出符合「時間區段」限制的條件，該「會議時間」自己本身化名為「時間區段」下，共有「學年度」、「年度」及「年月」三種歸類方式，在未加入時間數值時，將以”學年度”等任一歸類來表示，條件為僅列出以”學年度”為列表的資訊；另該項(屬性-值)之建構，主要在加強「會議時間」的用字淺詞的部份，故「時間區段」為「會議時間」的延申註解。
- 第 13 行為排序方式，將查詢後欲顯示的資料依「開會時間」以升冪方式排序顯示，最久時間的資料優先排列。



圖 4-25 簡易式語意查詢—依「會議時間(含同義詞)」查詢介面

表 4-13 依「會議時間(含同義詞)」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?會議時間 ?開會時間 ?會議名稱
7	WHERE {
8	?會議時間 default:MT03SameAs ?時間區段.
9	?會議時間 default:Time-TypeInf ?會議名稱.
10	?會議名稱 default:MRInf02Time ?開會時間.
11	FILTER Regex(str(?時間區段),"學年度").
12	}
13	ORDER BY ASC(?開會時間)

拾壹、簡易式語意查詢—依「地點(含同義詞應用)+會議種類資訊」查詢

依「會議地點+會議種類資訊」來查詢，並顯示「會議地點、地點所在大樓、會議時間、會議名稱」內容，其使用者操作畫面如圖 4-26 所示。我們所使用到的 SPARQL 語法如表 4-14，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「PlaceName」、「PlaceArea」、「MRTime」、「MRName」，即代表依條件限制查詢後欲列出「會議地點」下「PlaceName」、「PlaceArea」及「會議種類資訊」下「MRTime」、「MRName」之關聯內容。
- 第 7、15 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12、13 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議種類資訊」透過 Object properties 與「會議地點」下 Instance 化名為「會議地點」關聯，關聯屬性為 default:TypeInf-Place，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「會議地點」透過 Annotation Properties 與自己本身化名為「PlaceName」關聯，關聯屬性為 default:MP01Name。第 10 行：「會議地點」透過 Annotation Properties 與自己本身化名為「PlaceArea」關聯，關聯屬性為 default:MP02Located。第 11 行：「會議地點」透過 Annotation Properties 與自己本身化名為「PlaceSyn」關聯，關聯屬性為 default:MP03SameAs，該關聯主要針對「會議地點」導入同義詞應用讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率。第 12 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRTime」關聯，關聯屬性為 default:MRInf02Time。第 13 行：「會議種類資訊」透過 Annotation Properties 與自己本身化名為「MRName」關聯，關聯屬性為 default:MRInf01Title 等以上之條件限制。
- 第 14 行為過濾出符合「PlaceSyn」限制的條件，""表示條件為空值(不限制)，該項(屬性-值)之建構，主要在加強「會議地點」的用字淺詞的部份，故「PlaceSyn」為「PlaceName」的延申註解。
- 第 16 行為排序方式，將查詢後欲顯示的資料依「PlaceName」以升冪方式排序顯示。



圖 4-26 簡易式語意查詢—依「會議地點(合同義詞)+會議種類資訊」查詢介面

表 4-14 依「會議地點(合同義詞)+會議種類資訊」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?PlaceName ?PlaceArea ?MRTime ?MRName
7	WHERE {
8	?會議種類資訊 default:TypeInf-Place ?會議地點.
9	?會議地點 default:MP01Name ?PlaceName.
10	?會議地點 default:MP02Located ?PlaceArea.
11	?會議地點 default:MP03SameAs ?PlaceSyn.
12	?會議種類資訊 default:MRInf02Time ?MRTime.
13	?會議種類資訊 default:MRInf01Title ?MRName.
14	FILTER Regexp(str(?PlaceSyn), "").
15	}
16	ORDER BY ASC(?PlaceName)

拾貳、複合式語意查詢—依「主題分類+時間(含同義詞應用)」查詢

依「主題分類+時間」來查詢，並顯示「案由、會議時間、會議名稱、提案型態」內容，其使用者操作畫面如圖 4-27 所示。我們所使用到的 SPARQL 語法如表 4-15，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「PTitle」、「MRTime」、「MRName」、「PI」，即代表依條件限制查詢後欲列出「提案型態」下「PTitle」、「PI」及「會議種類資訊」下「MRTime」、「MRName」之關聯內容。
- 第 7、18 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11、12、13、14 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議時間」透過 Annotation Properties 與自己本身化名為「TimeSyn」關聯，關聯屬性為 default:MT03SameAs。第 9 行：「會議時間」透過 Object properties 與「會議種類資訊」下 Instance 化名為「MRName」關聯，關聯屬性為 default:Time-TypeInf，該 Object properties 將不同區塊的 Class 進行關聯。第 10 行：「MRName」透過 Annotation Properties 與自己本身化名為「MRTime」關聯，關聯屬性為 default:MRInf02Time。第 11 行：「MRName」透過 Object properties 與「提案型態」下 Instance 化名為「PI」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 12 行：「PI」透過 Annotation Properties 與自己本身化名為「PTitle」關聯，關聯屬性為 default:Proposal01Title。第 13 行：「PI」透過 Annotation Properties 與自己本身化名為「PHK」關聯，關聯屬性為 default:Proposal03HideKnowledge，該關聯主要針對「案由」導入同義詞應用讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率。第 14 行：「PI」透過 Object properties 與「主題分類」下 Instance 化名為「PType」關聯，關聯屬性為 default:ProposalType-TitleClass，該 Object properties 將不同區塊的 Class 進行關聯，並符合以上之條件限制。
- 第 15、16、17 行分別為過濾出符合「TimeSyn」、「PTitle」、「PHK」及「PType」限制的條件，15 行該「會議時間」自己本身化名為「TimeSyn」下，共有「學年度」、「年度」及「年月」三種歸類方式，在未加入時間數值時，將以「學年度」等任一歸類來表示，條件為僅列

出以”學年度”為列表的資訊；另該項(屬性-值)之建構，主要在加強「會議時間」的用字淺詞的部份，故「時間區段」為「會議時間」的延申註解；16行之「案由」自己本身化為「PTitle」限制條件為””，另為的加強「案由」的資料搜尋，額外加入「PHK」主要在加強「案由」的用字淺詞的部份，故「PHK」為「PTitle」的延申註解；17行之「主題分類」自己本身化為「PType」限制條件為””，以上若””表示條件為空值（不限制）。

- 第 19 行為排序方式，將查詢後欲顯示的資料依「PI」以升冪方式排序顯示，最久時間的提案例子優先排列。



圖 4-27 複合式語意查詢—依「主題分類+時間(合同義詞)」查詢介面

表 4-15 依「主題分類+會議時間(合同義詞)」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?PTitle ?MRTime ?MRName ?PI
7	WHERE {
8	?會議時間 default:MT03SameAs ?TimeSyn.
9	?會議時間 default:Time-TypeInf ?MRName.
10	?MRName default:MRInf02Time ?MRTime.
11	?MRName default:TypeInf-ProposalType ?PI.
12	?PI default:Proposal01Title ?PTitle.
13	?PI default:Proposal03HideKnowledge ?PHK.
14	?PI default:ProposalType-TitleClass ?PType.
15	FILTER Regex(str(?TimeSyn),"學年度").
16	FILTER (Regex(str(?PTitle),"") Regex(str(?PHK),"")).
17	FILTER Regex(str(?PType),"").
18	}
19	ORDER BY ASC(?PI)

拾參、複合式語意查詢—依「提案單位(含同義詞應用)+提案型態」查詢

依「提案單位+提案型態」來查詢，並顯示「提案單位、提案型態、案由」內容，其使用者操作畫面如圖 4-28 所示。我們所使用到的 SPARQL 語法如表 4-16，該語法說明如下：

- 第 1、2、3、4、5 行為名稱空間，即定義使用到的前置詞(Prefix)，簡化查詢的敘述。
- 第 6 行 SELECT 為顯示 WHERE {} 查詢命令中的「提案單位」、「提案型態」、「PTitle」，即代表依條件限制查詢後欲列出「提案單位」下「提案單位」及「提案型態」下「提案型態」、「PTitle」之關聯內容。
- 第 7、15 行分別為 WHERE {} 查詢命令的開始與結束。
- 第 8、9、10、11 行分別為相互關聯過濾出符合資訊內容。第 8 行：「會議種類資訊」透過 Object properties 與「提案型態」下 Instance 化名為「提案型態」關聯，關聯屬性為 default:TypeInf-ProposalType，該 Object properties 將不同區塊的 Class 進行關聯。第 9 行：「提案型態」透過 Object properties 與「提案單位」下 Instance 化名為「提案單位」關聯，關聯屬性為 default:ProposalType-Unit，該 Object properties 將不同區塊的 Class 進行關聯。第 10 行：「提案單位」透過 Annotation Properties 與自己本身化名為「UnitSyn」關聯，關聯屬性為 default:MRUnitSameAs，該關聯主要針對「提案單位」導入同義詞應用讓可搜尋詞彙變成的更廣義，增加被搜尋到的機率。第 11 行：「提案型態」透過 Annotation Properties 與自己本身化名為「PTitle」關聯，關聯屬性為 default:Proposal01Title 等以上之條件限制。
- 第 12、13、14 行分別為過濾出符合「UnitSyn」、「PTitle」、「提案型態」限制的條件，其中「UnitSyn」限制條件為”人事室”，該項(屬性-值)之建構，主要在加強「提案單位」的用字淺詞的部份，故「UnitSyn」為「提案單位」的延申註解、「PTitle」限制條件為””、「提案型態」限制條件為”報告案”，以上若””表示條件為空值（不限制）。
- 第 16 行為排序方式，將查詢後欲顯示的資料依「提案型態」以升冪方式排序顯示。



以知識本體為基礎之會議查詢系統

圖 4-28 複合式語意查詢—依「提案單位(合同義詞)+提案型態」查詢介面

表 4-16 依「提案單位(合同義詞應用)+提案型態」查詢之 SPARQL 查詢指令

行號	內容
1	prefix xsd: <http://www.w3.org/2001/XMLSchema#>
2	prefix default: <http://www.owl-ontologies.com/Ontology1311105684.owl#>
3	prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
4	prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
5	prefix owl: <http://www.w3.org/2002/07/owl#>
6	SELECT ?提案單位 ?提案型態 ?PTitle
7	WHERE {
8	?會議種類資訊 default:TypeInf-ProposalType ?提案型態.
9	?提案型態 default:ProposalType-Unit ?提案單位.
10	?提案單位 default:MRUnitSameAs ?UnitSyn.
11	?提案型態 default:Proposal01Title ?PTitle.
12	FILTER Regex(str(?UnitSyn),"人事室").
13	FILTER Regex(str(?PTitle),"").
14	FILTER Regex(str(?提案型態),"報告案").
15	}
16	ORDER BY ASC(?提案型態)

第六節 系統應用實例

在本節，我們將舉出 13 個應用實例來驗證本系統的設計是否符合使用者的需求。

1. 簡易實例 1-1 「依主題」

使用者希望能透過本系統查詢有關某些主題相關的會議資料，例如：法規、經費、人事...等相關主題，並能了解該會議所決議的太致上情形，以下我們以「法規」為主題，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」－>「簡易實例一」－>並在主題下拉式選單選擇「法規」－>「查詢」即可。
- 以上操作畫面如圖 4-29 所示，查詢結果如圖 4-30 所示。



圖 4-29 案例 1-1 查詢畫面

主題分類	提案型態	案由	決定議
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-臨時動議02>	"○○技術學院經費核作業流程與注意事項(草案)，提請討論。"	"通過。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論02>	"修訂「○○技術學院附設進修學院組織規程」，提請討論。"	"通過。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論03>	"修訂「○○技術學院附設專科進修學校組織規程」，提請討論。"	"通過。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論04>	"擬修正本校通識教育委員會組織要點，提請討論。"	"提下次會議再議。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論05>	"擬訂「○○技術學院教學優良教師遴選及獎勵要點」草案，提請討論。"	"要點二、六、八修正後通過，修正後附件八。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論06>	"增訂「○○技術學院執行「弱勢學生助學計畫」實施要點」，提請討論。"	"通過。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論07>	"「○○技術學院文書處理規則」修正案，提請討論。"	"修正後通過，修正後條文如附件十。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論08>	"「○○技術學院檔案管理規則」修正案，提請討論。"	"修正後通過，修正後條文如附件十一。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論09>	"增訂法規「○○技術學院教育部獎補助整體發展資本門經費各所系分配要點」，提請討論。"	"請總務長針對會議所提之意見與各系溝通，再提會議討論。"
<http://www.owl-ontologies.com/Ontology1311105684.owl#法規>	<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-提案討論10>	"修正本校「○○技術學院長途電話使用管理要點」草案，提請討論。"	"通過。"

圖 4-30 案例 1-1 查詢結果

2.簡易實例 1-2「依提案型態」

使用者希望能透過本系統查詢所有提案資料來自於哪些會議召開時所提，並以什麼樣的提案型態提出，例如：報告案、提案討論、臨時動議等相關提案型態所提的資料內容及來自於那次會議提出情形，以下我們以「報告案」為提案型態，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」→「簡易實例二」→並在提案型態下拉式選單選擇「報告案」→「查詢」即可。
- 以上操作畫面如圖 4-31 所示，查詢結果如圖 4-32 所示。



圖 4-31 案例 1-2 查詢畫面

SPARQLer Query Results - Windows Internet Explorer

http://127.0.0.1:2020/books?selec%3D01=&RadioGroupBySample=2&selec%3D2=%E5%A0%B1%E5%91%A6%E5%A1%88&selec%3D3=01

SPARQLer Query Results

提案型態	案由	決定議	會議名稱
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案01>	"96年技術學院綜合評鑑相關資訊，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案02>	"○○技術學院校務共識營活動相關事宜，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案03>	"96年評鑑表開撰時程，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"授權各學群長調整附件二表六「96年評鑑表冊檢閱分配表」中之各學群評鑑表冊檢閱人。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案04>	"本校「96年評鑑工作特別預算申請與分配」作業，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"請各單位參閱附件三(頁2)之說明，於下星期一(9月3日)前，重新提出96年評鑑預算申請。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案05>	"96學年度新生註冊概況及全校學生人數預估，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案06>	"進修部新生訓練實施計畫摘要，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案07>	"96學年度進修學院暨進修專校新生註冊人數，提請報告。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第一次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-01行政-報告案01>	"報告截至9月5日止各系對教育部「94學年度畢業生流向平台」之填報狀況。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第二次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-報告案02>	"教育部95學年度畢業生出路調查事宜，敬請各系所該年度導師協助配合調查。" ^^<http://www.w3.org/2001/XMLSchema#string>	"洽悉，請大家共同努力。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第二次" ^^<http://www.w3.org/2001/XMLSchema#string>
<http://www.owl-ontologies.com/Ontology1311105684.owl#AD96-1-02行政-報告案02>	"擬申請成立「社團法人○○技術學院校友會」，提請討論。" ^^<http://www.w3.org/2001/XMLSchema#string>	"將說明中之「並籌募母校各項建設及校友會基金」文字刪除後通過。" ^^<http://www.w3.org/2001/XMLSchema#string>	"九十六學年度第一學期第二次" ^^<http://www.w3.org/2001/XMLSchema#string>

完成

網際網路

圖 4-32 案例 1-2 查詢結果

3.簡易實例 1-3 「依提案單位」

使用者希望能透過本系統查詢有關各提案單位於各個會議上提案的情形，例如：教務處、學務處、總務處、研發處、人事室...等相關提案單位於各會議上提案的情形，以下我們以「學務處」為提案單位，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」－>「簡易實例三」－>並在提案單位下拉式選單選擇「學務處」－>「查詢」即可。
- 以上操作畫面如圖 4-33 所示，查詢結果如圖 4-34 所示。



圖 4-33 案例 1-3 查詢畫面

提案單位	案由	決定議	會議名稱
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"增訂「○○技術學院執行「弱勢學生助學計畫」實施要點」，提請討論。"	"通過。"	"九十六學年度第一學期第二次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"修訂「○○技術學院教師輔導學生辦法」，提請討論。"	"修正後通過，修正後條文如附件四。"	"九十六學年度第一學期第四次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"教育部96年度大專校院推動性別平等訪視評鑑事宜，提請討論。"	"1.敬請各相關單位提供協助棄填。2.有鑑於目前相關法規之不適用，請學務處會同秘書室完成此組織法規之修訂。"	"九十六學年度第一學期第四次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"「○○技術學院導師制度實施要點」修訂案，提請討論。"	"修正通過，修正後條文如附件九。"	"九十六學年度第一學期第六次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"○○技術學院績優導師獎勵要點修改條文，提請討論。"	"修正通過，修正後條文如附件十。"	"九十六學年度第一學期第六次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"○○技術學院學生申訴辦法修改條文，提請討論。"	"通過。"	"九十六學年度第一學期第六次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"「第43屆校慶運動會暨園遊會」活動內容規劃案。"	"1.請學務處將幼保系兒童戲劇公演納入本屆校慶活動之一。2.各單位若有其它校慶相關活動之規劃，請儘早告知學務處，以利學務處彙整。3.請學務處於2/20(三)前召開會前會，商討第一次校慶籌備會議相關事宜。"	"九十六學年度第一學期第八次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"「○○技術學院衛生委員會組織要點」修正案，提請審議。"	"1.將本要點第二點中之進修學院主任改為進修學院校務主任；衛保組組長改為衛生保健組組長。2.其餘條文修正後通過，修正後條文如附件二。"	"九十六學年度第一學期第八次行政會議"
<http://www.owl-ontologies.com/Ontology1311105684.owl#學務處>	"本校「學生申訴處理辦法」修正案，提請討論。"	"1.將本辦法之一條條文做適當的修正。2.其餘條文修正後通過，修正後條文如附件三。"	"九十六學年度第一學期第八次行政會議"

圖 4-34 案例 1-3 查詢結果

4.簡易實例 1-4「依會議時間」

使用者希望能透過本系統查詢有關特定時間上會議召開的情形，例如：96(學)年度、97(學)年度、98(學)年度...等相關會議時間，其會議所召開情形，以下我們以「98 年度」為會議時間，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」－>「簡易實例四」－>並在會議時間下拉式選單選擇「98」－>「年度」－>「查詢」即可。
- 以上操作畫面如圖 4-35 所示，查詢結果如圖 4-36 所示。

5.簡易實例 1-5 「依會議地點之一」

使用者希望能透過本系統查詢有關會議召開到目前為止各大樓現有會議室場地有哪些地點，例如：文○樓、生○樓...等現有的相關會議地點有哪些，以下我們以「文○樓」為會議地點，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」－>「簡易實例五」－>並在會議地點下拉式選單選擇「文○樓」－>「查詢」即可。
- 以上操作畫面如圖 4-37 所示，查詢結果如圖 4-38 所示。

The screenshot shows a web interface for a knowledge-based query system. On the left, there is a vertical title: **以知識本體為基礎之會議查詢系統**. The main content area is titled **《《簡易式SPARQL查詢》》** and contains seven query examples. The fifth example, **簡易實例五：《《依會議地點》》**, is selected. It asks: **我希望藉由本系統查詢有關會議召開之** [所有大樓] **現有會議室場地有哪些地點。** The dropdown menu for [所有大樓] is open, showing options: **所有大樓**, **文○樓** (highlighted), and **生○樓**. Below the examples is a text input field containing `/xml-to-html.xsl` and a **查詢** button.

圖 4-37 案例 1-5 查詢畫面

地點名稱	地點區域
"文○樓1F會議室" ^^<http://www.w3.org/2001/XMLSchema#string>	"文○樓" ^^<http://www.w3.org/2001/XMLSchema#string>
"文○樓3F-A317會議室" ^^<http://www.w3.org/2001/XMLSchema#string>	"文○樓" ^^<http://www.w3.org/2001/XMLSchema#string>

圖 4-38 案例 1-5 查詢結果

6.簡易實例 1-6 「依會議地點之二」

使用者希望能透過本系統查詢有關會議召開到目前為止各大樓會議室使用情形，例如：文○樓、生○樓...等已使用的相關會議地點有哪些、使用情形為何？！以下我們以「生○樓」為會議地點，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」－>「簡易實例六」－>並在會議地點下拉式選單選擇「生○樓」－>「查詢」即可。
- 以上操作畫面如圖 4-39 所示，查詢結果如圖 4-40 所示。

7.簡易實例 1-7「依會議種類資訊」

使用者希望能透過本系統查詢有關各項會議召開之相關細部資訊，以利使用者更進一步了解到該次會議名稱、時間為何?!地點在哪?!該次會議主席是誰?!出席人員列表...等細部資訊內容，可了解的範圍例如：所有或行政、校務會議等細部資訊內容，進行查詢如下：

- 使用方法：請點選「以簡易式查詢」->「簡易實例七」->並在所有會議種類下拉式選單選擇「校務」->「查詢」即可。
- 以上操作畫面如圖 4-41 所示，查詢結果如圖 4-42 所示。

以知識本體為基礎之會議查詢系統

以簡易式查詢 | 以簡易式語意查詢 | 以複合式語意查詢

《《簡易式SPARQL查詢》》

簡易實例一：《《依主題》》

我希望藉由本系統查詢有關 相關案由的會議資料。

簡易實例二：《《依提案型態》》

我希望藉由本系統查詢有關 的案由說明及各來自於哪些會議召開所提出。

簡易實例三：《《依提案單位》》

我希望藉由本系統查詢有關 於各會議上提案的情形。

簡易實例四：《《依會議時間》》

我希望藉由本系統查詢有關 的時間上所召開之會議有哪些。

簡易實例五：《《依會議地點》》

我希望藉由本系統查詢有關會議召開之 現有會議室場地有哪些地點。

簡易實例六：《《依會議地點》》

我希望藉由本系統查詢有關會議召開之 會議室使用情形。

簡易實例七：《《依會議種類資訊》》

我希望藉由本系統查詢有關會議召開之 會議相關(名稱、時間、地點、主席...等)資訊。

圖 4-41 案例 1-7 查詢畫面

MRName	MRTime	MRPlace	MRChairman
"九十六學年度第一學期第一次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2007-09-19" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十六學年度第一學期第二次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2007-12-26" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十六學年度第二學期第一次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-02-27" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十六學年度第二學期第二次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-06-20" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十七學年度第一學期第一次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-09-24" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十七學年度第一學期第二次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-10-29" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十七學年度第一學期第三次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-12-30" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十七學年度第二學期第一次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-02-18" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI
"九十七學年度第二學期第二次校務會議" ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-04-22" ^^<http://www.w3.org/2001/XMLSchema#date>	"本校生○樓九樓第三視聽室" ^^<http://www.w3.org/2001/XMLSchema#string>	"○校長○○" ^^<http://www.w3.org/2001/XMI

圖 4-42 案例 1-7 查詢結果

8.簡易語意實例 2-1「依主題(含同義詞)」

使用者希望能透過本系統查詢有關某些主題相關的會議資料，例如：法規、經費、人事...等相關主題，並進一步對該主題的案由做含有同義詞應用更細部同義的查詢，以能了解該會議所決議的太致上情形，以下我們以「一般事務」為主題，並將範圍縮小至「宿舍」相關內容，進行查詢如下：

- 使用方法：請點選「以簡易語意查詢」→「簡易語意實例一」→並在主題下拉式選單選擇「一般事務」→在自訂《案由》文字方塊輸入「宿舍」→「查詢」即可。
- 以上操作畫面如圖 4-43 所示，查詢結果如圖 4-44 所示。



圖 4-43 案例 2-1 查詢畫面

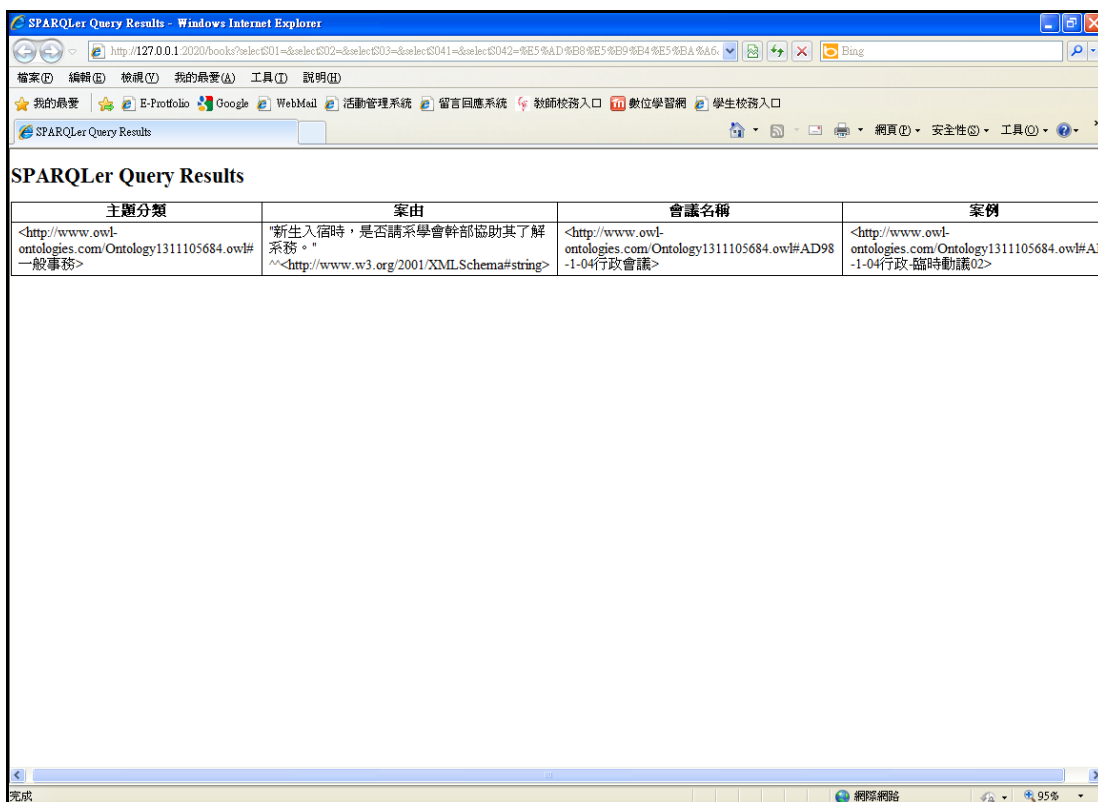


圖 4-44 案例 2-1 查詢結果

9.簡易語意實例 2-2 「依提案單位(合同義詞)」

使用者希望能透過本系統查詢有關各提案單位於各個會議上提案的情形，例如：教務處、學務處、總務處、研發處、人事室...等相關提案單位於各會議上提案的情形，然該下拉式選單僅列出常用單位列表，該項包含有同義詞應用可於自訂文字方塊輸入其他可能單位名詞來查詢，以下我們以「圖資處」或「圖書資訊處」為提案單位，進行查詢如下：

- 使用方法：請點選「以簡易語意查詢」－>「簡易語意實例二」－>並在自訂《提案單位》文字方塊輸入其他可能單位名詞「圖資處」－>「查詢」即可。
- 以上操作畫面如圖 4-45 所示，查詢結果如圖 4-46 所示。

以知識本體為基礎之會議查詢系統

以簡易式查詢 | 以簡易式語意查詢 | 以複合式語意查詢

《《簡易式SPARQL語意查詢》》

簡易語意實例一：《《依主題(合同義詞應用)》》

我希望藉由本系統查詢有關 (含自訂《案由》例：組織規程) 的會議資料提案審議的情形。

簡易語意實例二：《《依提案單位(合同義詞應用)》》

我希望藉由本系統查詢有關 or 自訂《提案單位》例：圖資處

於各會議上提案的情形。

簡易語意實例三：《《依會議時間(合同義詞應用)》》

我希望藉由本系統查詢有關 or 自訂《時間》例：98/01 的時間上所召開之會議有哪些。

簡易語意實例四：《《依地點(合同義詞應用)+會議種類資訊》》

我希望藉由本系統查詢有關 (含自訂《細部地點》例：九樓第三會議室) 場地使用情形。

圖 4-45 案例 2-2 查詢畫面

PUnit	PTitle	PResolution	MRName
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"配合「教育部大專校院校園保護智慧財產權行動方案」，圖書館之執行項目。"	"請各教學單位向啟單位老師宣導，若有書商捐贈多餘的教科書，請提供給圖書館。"	"九十六學年度第一學期第八次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"「○○技術學院圖書館使用要點」修訂案，提請討論。"	"通過。"	"九十六學年度第二學期第二次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"「○○技術學院圖書館使用要點」修訂案，提請討論。"	"1.將附件八之附件一中之「學生證」改為「學生證或身份證」。2.修正後通過。"	"九十六學年度第二學期第九次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"97學年度圖書館新生導覽簡介實施計劃，提請報告。"	"洽悉。"	"九十七學年度第一學期第三次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"「○○技術學院圖書館愛校捐書活動計畫書」，提請討論。"	"1.附件六頁6之學校logo請修正。2.依計畫辦理。"	"九十七學年度第一學期第五次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"提請修訂「○○技術學院圖書館使用要點」，提請討論。"	"1.二、...「依兒童福利法規定」...改為「依兒童福利法規定」。2.其餘條文修正後通過，修正後條文如附件七。"	"九十七學年度第一學期第五次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"提請修訂「○○技術學院圖書館諮詢委員會組織要點」，提請討論。"	"1.「○○技術學院圖書館諮詢委員會組織要點」改為「○○技術學院圖書館諮詢委員會設置要點」。2.第一點改為「一、為集思廣益，共謀圖書館業務發展，特設置「○○技術學院(以下簡稱本校)圖書館諮詢委員會(以下簡稱本會)」，促使發揮圖書館應有之功能」。3.第二點之「選介」改為「遴選」；「應聘得連任之」改為「連聘得連任之」。4.其餘條文修正後通過，修正後條文如附件四。"	"九十七學年度第一學期第六次行政會議紀錄"
<http://www.owl-ontologies.com/Ontology1311105684.owl#圖書館>	"敬請支持圖書館下列活動。"	"洽悉。"	"九十七學年度第一學期第十一次行政會議紀錄"

圖 4-46 案例 2-2 查詢結果

10.簡易語意實例 2-3「依會議時間(含同義詞)」

使用者希望能透過本系統查詢有關特定時間上會議召開及出席人員的情形，例如：96(學)年度、97(學)年度、98(學)年度、某年某月...等相關會議時間，其會議所召開情形，然該下拉式選單僅列出常用(學)年度單位列表，該項包含有同義詞應用可於自訂文字方塊輸入其他可能時間單位名詞來查詢，以下我們以「某年某月」為會議時間單位，進行查詢如下：

- 使用方法：請點選「以簡易語意查詢」→「簡易語意實例三」→並在自訂《時間》文字方塊輸入其他可能時間單位名詞「98年3月」or「98/03」→「查詢」即可。
 - 以上操作畫面如圖 4-47 所示，查詢結果如圖 4-48 所示。



圖 4-47 案例 2-3 查詢畫面

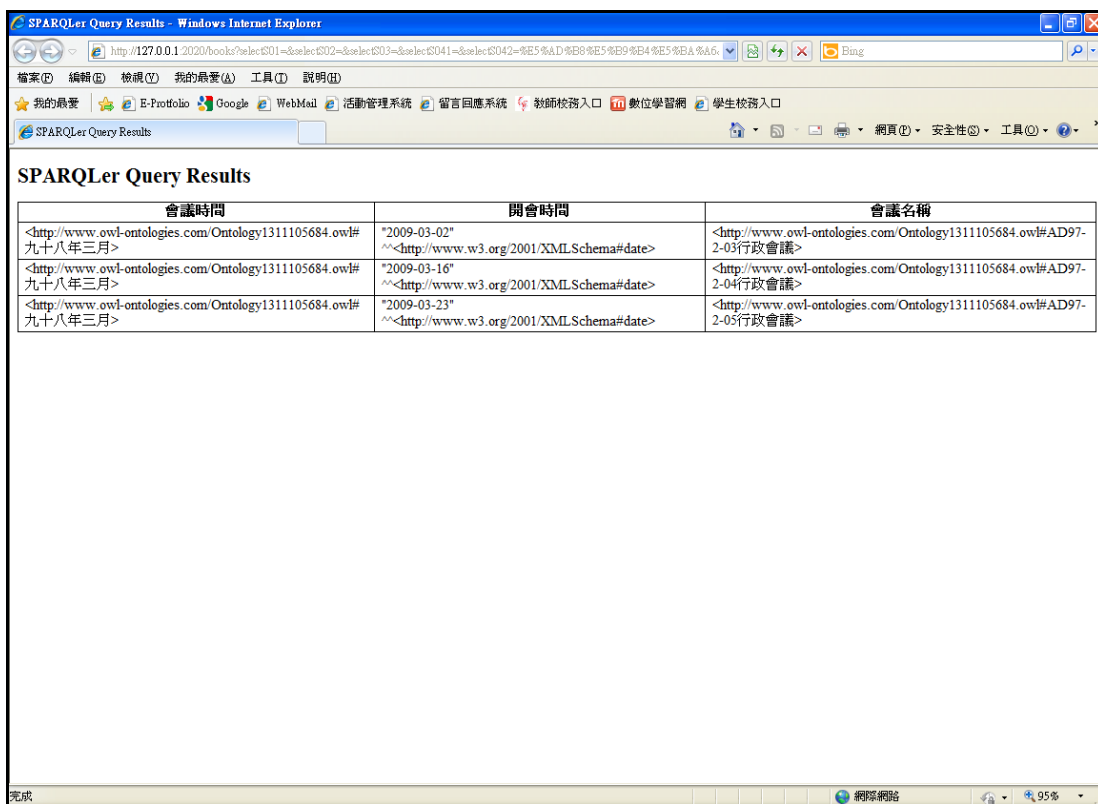


圖 4-48 案例 2-3 查詢結果

11.簡易語意實例 2-4「依會議地點之二(合同義詞)」

使用者希望能透過本系統查詢有關會議召開到目前為止各大樓會議室使用情形，並進一步對該大樓會議室做含有同義詞應用更細部同義的查詢，例如：文○樓一樓會議室、生○樓九樓第三會議室...等已使用的相關會議地點使用情形為何?!以下我們以「生○樓九樓第三會議室」為會議地點，進行查詢如下：

- 使用方法：請點選「以簡易語意查詢」－>「簡易語意實例四」－>並在會議地點下拉式選單選擇「生○樓」－>在自訂《細部地點》文字方塊輸入「九樓第三會議室」－>「查詢」即可。
- 以上操作畫面如圖 4-49 所示，查詢結果如圖 4-50 所示。

以知識本體為基礎之會議查詢系統

以簡易式查詢 | 以簡易式語意查詢 | 以複合式語意查詢

《《簡易式SPARQL語意查詢》》

簡易語意實例一：《《依主題(合同義詞應用)》》

我希望藉由本系統查詢有關 (含自訂《案由》例：組織規程) 的會議資料提案審議的情形。

簡易語意實例二：《《依提案單位(合同義詞應用)》》

我希望藉由本系統查詢有關 or 自訂《提案單位》例：國資處

於各會議上提案的情形。

簡易語意實例三：《《依會議時間(合同義詞應用)》》

我希望藉由本系統查詢有關 or 自訂《時間》例：98/01 的時間上所召開之會議有哪些。

簡易語意實例四：《《依地點(合同義詞應用)+會議種類資訊》》

我希望藉由本系統查詢有關 (含自訂《細部地點》例：九樓第三會議室) 場地使用情形。

圖 4-49 案例 2-4 查詢畫面

PlaceName	PlaceArea	MRTime	MIRName
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2007-09-19" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十六學年度第一學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2007-12-26" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十六學年度第一學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-02-27" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十六學年度第二學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-06-20" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十六學年度第二學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-09-24" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第一學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-10-29" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第一學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2008-12-30" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第一學期第三次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-02-18" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第二學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-04-22" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第二學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-06-10" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十七學年度第二學期第三次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-09-23" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十八學年度第一學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2009-12-28" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十八學年度第一學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2010-03-03" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十八學年度第二學期第一次校" ^^<http://www.w3.org/2001/XMLSchema#string>
生○樓9樓第3會議室 ^^<http://www.w3.org/2001/XMLSchema#string>	*生○樓* ^^<http://www.w3.org/2001/XMLSchema#string>	"2010-06-09" ^^<http://www.w3.org/2001/XMLSchema#date>	"九十八學年度第二學期第二次校" ^^<http://www.w3.org/2001/XMLSchema#string>

圖 4-50 案例 2-4 查詢結果

12.複合實例 3-1 「依主題+會議時間(含同義詞)」

使用者希望能透過本系統查詢有關某些主題相關（可含進階主題有同義詞應用關鍵字查詢）的會議資料提案通過的時間情形或限定於某些範圍上的時間資料即可，例如：法規、經費、人事...等相關主題，進階搜尋，例如：組織規程、智財權、宿舍等資訊，並限定於(學)年度或某年某月範圍內的資訊，並從中能了解該會議所決議的太致上情形，以下我們以「一般事務」為主，"智慧財產權"為輔的主題並再縮小時間範圍內的資料，以了解其會議通過的相關資訊，進行查詢如下：

- 使用方法：請點選「以複合式語意查詢」→「複合語意實例一」→並在主題下拉式選單選擇「一般事務」→在自訂《案由》關鍵字鍵入「智財權」→在自訂《時間》文字方塊輸入其他可能時間單位名詞「96年9月」or「96/09」or「2007/09」「查詢」即可。
 - 以上操作畫面如圖 4-51 所示，查詢結果如圖 4-52 所示。



圖 4-51 案例 3-1 查詢畫面

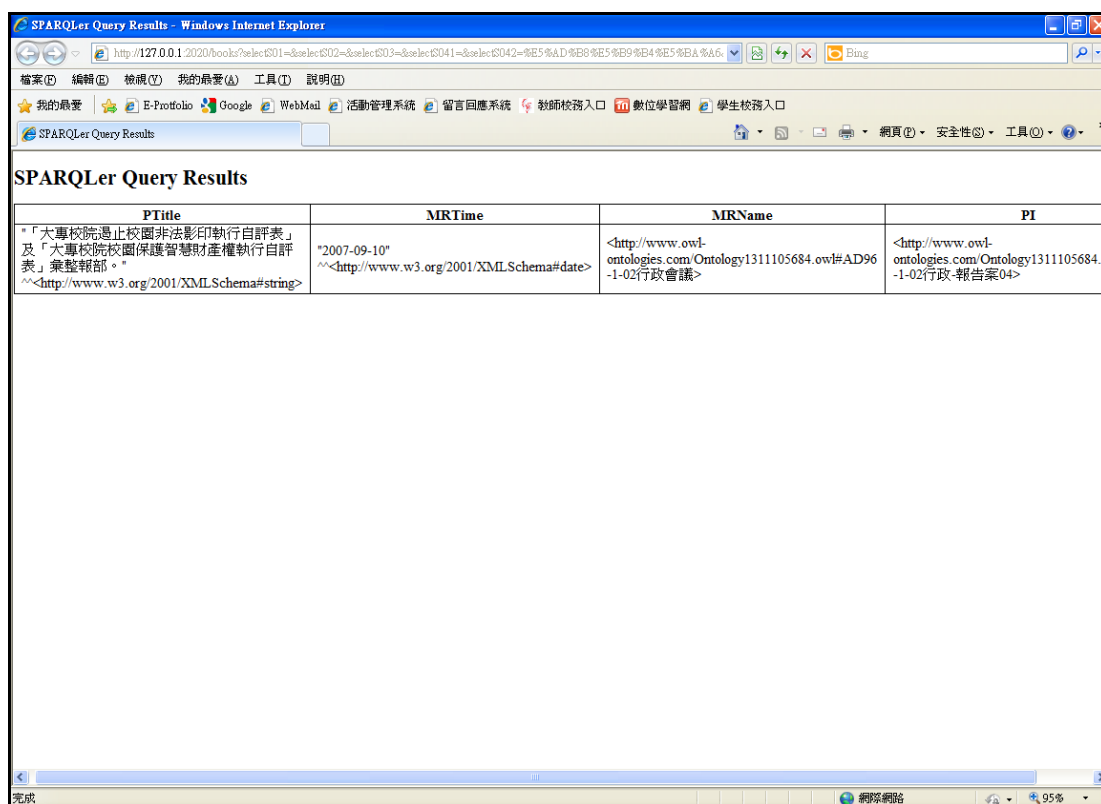


圖 4-52 案例 3-1 查詢結果

13.複合實例 3-2 「依提案單位(合同義詞)+提案型態」

使用者希望能透過本系統查詢有關各提案單位（合同義詞應用）於會議上各型態提案的情形，例如：學務處(學生事務處)、菁英中心（招生中心）、圖資處(圖書資訊處、電子計算機中心、電算中心、圖書館)...等相關提案單位（合同義詞應用）於會議上各型態提案（報告案、提案討論、臨時動議）及可含進階提案型態有同義詞應用關鍵字查詢的情形，以下我們以「菁英中心」為提案單位，「提案討論」為提案型態，並進階查詢有關「招生」範圍相關資訊，進行查詢如下：

- 使用方法：請點選「以複合式語意查詢」－>「複合語意實例二」－>並在自訂《提案單位》文字方塊輸入其他可能單位名詞「菁英中心」－>在《提案型態》下拉式選單選擇「提案討論」－>在自訂《案由》文字方塊輸入「招生」－>「查詢」即可。
- 以上操作畫面如圖 4-53 所示，查詢結果如圖 4-54 所示。

以知識本體為基礎之會議查詢系統

以簡易式查詢 | 以簡易式語意查詢 | 以複合式語意查詢

《《複合式SPARQL語意查詢》》

複合語意實例一：《《依(主題+時間)(合同義詞應用)》》

我希望藉由本系統查詢有關 (含自訂《案由》例：組織規程) 於 or 自訂《時間》例：年/月 範圍內的會議資料提案審議的情形。

複合語意實例二：《《依提案單位(合同義詞應用)+提案型態》》

我希望藉由本系統查詢有關 or 自訂《提案單位》例：招生中心(菁英中心) 於會議上所提之 (含自訂《案由》例：詳見) 的情形。

圖 4-53 案例 3-2 查詢畫面

第五章、結論與建議

在本章，我們總結本文所開發的大學院校會議紀錄語意檢索系統（SQSCMR）的主要功能與貢獻，並提出本系統的未來研究與發展方向。

第一節 結論

本文將會議紀錄建構在知識本體並予以系統化，以解決傳統在查詢時，管理者須一一調閱各大會議的文書電子檔，並以人工肉眼的方式快速檢視而極為不易找尋的方式。我們透過實作的方式，將會議紀錄建構在知識本體上，並藉由同義詞等語意分析與知識本體所支援的語言，以提案單位為例，將提案單位的部份予以建置相關語彙，再藉由推論引擎的邏輯推理，將原本只是關鍵字的查詢，擴展到多層次語意關聯的搜尋，進而能更廣泛地搜尋及精確且迅速地查詢出符合使用者所想要的資訊內容。

我們運用語意網技術，將 SQSCMR 系統建置於網頁管理平台上，這不只是將「會議紀錄查詢系統」資訊化，對於現今網際網路成熟度的應用，也更提供了使用者可不受時間或距離上的限制，給予同時間多人上網查詢使用，並符合跨平台作業服務。亦即，不論使用者是 Windows、Linux、Mac 或手持式 Android 等作業系統，只要能上網並使用自家的瀏覽器即可輕鬆地查閱。這除了給予使用者提供了便利性外，亦減輕了管理者自身或協助使用者查詢資料上的負擔。

關於本文的同義詞語彙建置，將由歷代的管理者一點一滴持續地被建置於該會議紀錄知識本體之中，日後將不用擔心管理者世代交替後，新進人員對於語彙的認知不足而查詢不到資料的問題。

第二節 未來展望

全球資訊網的發明人柏納李 (Tim Beners-Lee) 對於語意網的未來研究發展方向，除了語意搜尋的概念外，另外所要表達的是語意網能透過智慧型代理程式 (Software Agent) 主動蒐集並依設定好的規範持續擴充現有的知識本體。此一智慧型代理人程式的機制，在語意網上的應用也是極為重要的一環。然而，本系統初步僅針對「案由」、「提案單位」、「會議時間」、「會議地點」等做同義詞語彙的建置。在本系統的未來發展上，我們可針對其他項目等做更深入的語彙的建置，並配合智慧型代理人程式的機制，使其在主題分類上能達到自動化分類的功能。

參 考 文 獻

一、中文部份

1. 魏召欣，“智慧型企業文件檢索系統之建置-以大葉大學公文檢索為例”，大葉大學資訊管理學系碩士論文，2003。
2. 鍾正男，“以知識本體為基礎的語意查詢系統之研究-以圖書館為例”，大葉大學資訊管理學系碩士論文，2004。
3. 張仁青，“應用文”，台北市，文史哲出版社，1979。
4. 陳言熙，“運用文字探勘技術協助建構公司治理本體知識”，國立政治大學會計研究所碩士論文，2006。
5. 陳昭珍，“知識本體架構與知識組織發展新趨勢”，數位時代圖書館館藏發展研討會論文集，中興大學圖書資訊學研究所，2006，P318-P338。
6. 劉升平、武浩、詹子鵬、徐涵，“資源描述框架 (RDF): 概念與抽象語法”，W3C，2009，
<http://www.certifiedchinesetranslation.com/openaccess/rdfconcepts-tc.html#section-Graph-syntax>。
7. 陳亮廷，“以 RDF 為基礎之 XML DTD 整合研究”，朝陽科技大學資訊管理系碩士論文，2005。
8. 王梅玲，“資源描述架構/資源描述綱要”，技術服務小百科，2011，
<http://techserviceslibrary.blogspot.com/2011/04/rdf-resource-description.html>。
9. 曾國峰，“開發語意查詢系統支援國小體適能訓練處方之擬定”，南華大學資訊管理學系碩士論文，2010。
10. 吳育賢，“開發語意查詢系統協助教案之編寫”，南華大學資訊管理學系碩士論文，2008。
11. 劉升平、蘭煜峰，“OWL Web 本體語言概述 推薦標準”，W3CHINA.ORG 開放翻譯計畫 (OTP)，2004，<http://www.transwiki.org/tw/owloverview.htm>。
12. 李俞萱、陳盈儒、鄭筱頻、陳毓文，“OWL”，宅學習 - Social Learning Space 社交學習空間@FJU CSIE，2008，<http://sls.weco.net/node/9195>。
13. 鄭雅仁，“基於語意技術規劃軍事想定應用”，國立虎尾科技大學資訊工程系

- 碩士論文，2010。
- 14.王怡文，” 語意網，整合資訊好身手”，科學人雜誌專題文章【資訊科技】，2008。
 - 15.黃居仁，” 語意網、辭網與知識本體：淺談未來網路上的知識運籌”，佛教圖書館館訊 33 期，2003。
 - 16.教育部國語推行委員會，” 教育部重編國語辭典修訂本”，2007，<http://dict.revised.moe.edu.tw/>。

二、西文部份

1. Berners-Lee, T., "Semantic Web - XML2000", December 2000, <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>.
2. Koivunen, M. R., Miller, E., "W3C Semantic Web Activity", Nov., 2001, <http://www.w3.org/2001/12/semweb-fin/w3csw>.
3. Hendler, J., "Agents and the Semantic Web.", Intelligent System, IEEE, Vol. 16, Issue 2, Mar-Apr 2001. 30-37.
4. Noy, N. F. & McGuinness, D. L., "Ontology Development 101: A Guide to Creating Your First Ontology", Knowledge Systems Laboratory, Stanford University, CA, USA, March 2001.
5. Klyne, G. , Carroll, J., and McBride, B., "Resource Description Framework (RDF):Concepts and Abstract Syntax", February 2004, <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/> & <http://www.certifiedchinesetranslation.com/openaccess/rdfconcepts-tc.html>.
6. Berners-Lee, T., "Notation 3 Logic - An RDF language for the Semantic Web", August 2006, <http://www.w3.org/DesignIssues/Notation3.html>.
7. Smith, M. K., Welty, C., and McGuinness, D. L., "OWL Web Ontology Language Guide", February 2004, <http://www.w3.org/TR/2004/REC-owl-guide-20040210/> & <http://www.certifiedchinesetranslation.com/openaccess/owlguide-tc.html>.
8. Eric, P., and Andy, S., "SPARQL Query Language for RDF", W3C Recommendation, 2008.