

行政院國家科學委員會八十八年下半年及八十九年度科技行政研究發展報告

政府研究計畫全服務系統設計與開發之研究

執行期間：88年7月1日至89年12月31日

研究人員：徐玉梅
張東森
呂龍盈
高秋芳

執行單位：行政院國家科學委員會科學技術資料中心

中華民國九十年二月十五日

摘 要

我國政府重視研究發展資源之運用，自八十二年即開始建置「政府研究計畫基本資料檔(Government Research Bulletin, GRB)。近幾年，因網際網路的蓬勃發展，以及立法院及監察院日益關切政府研究資源之產出，人民亦殷切期盼政府資訊的公開，國科會科資中心乃積極進行 GRB 系統之研究開發工作。

本研究歷經需求確立、資料項目分析及輸入之規範訂定後，進入系統分析與設計，完成之系統置放於所建立之網站 (<http://www.grb.gov.tw>) 提供多元服務。本系統共建置完成四個子系統：

1. 識別碼管理子系統：以系統化方式管理計畫主持人、計畫主管機關之業務承辦人及計畫管考人，以及科資中心系統管理人員之識別碼及密碼。本子系統具管理彈性，各計畫主管機關可視計畫多寡自行決定採單階或二階管理模式。
2. 線上資料輸入及審查子系統：分研究計畫摘要、期中報告摘要及期末報告摘要等三種輸入介面，資料填寫、修改、刪除及審核等功能完備且控管周密，對計畫及其成果資料之正確性，以及避免項目內容之缺漏具提升效果。
3. 資料庫查詢子系統：檢索介面易用，查詢功能簡明，系統並將研究計畫資料庫、研究報告資料庫及研究報告全文檔案相互連結，除查詢功能外，亦提供瀏覽及列印等功能。
4. 統計列印子系統：系統提供數十種統計報表，包括研究經費及人力之配置資訊，這些統計分析結果已成為國家研究資源分配之重要參考。

GRB 系統之建構已陸續完成多數功能，對計畫管考機關以及民意機關整體掌握及重點觀察政府研究計畫之投入產出深具意義，對部會署主管計畫之管理亦發揮效能，而政府研究資訊的公開更藉 GRB 系統達成。但本系統在各層級使用者殷切期望下，仍需在快速通報系統之建立、計畫的不遺漏、資料交換與提供機制之建立、分類架構之調整及朝全服務目標推進等方面持續努力，以滿足各層級使用者之需求，亦須密切觀察網路環境之急速變化與採用更新之資訊技術，俾發揮系統的最大效益。

摘要
目次
表目次
圖目次

目 次

壹、緒論	1
一、研究緣起	1
二、研究目的	1
貳、國內外現況分析	3
一、GRB 計畫依據與建檔發展歷程	3
二、國內相關資料庫之建置及服務情形	3
三、國外研究計畫資訊提供情形	5
參、研究方法與步驟	7
一、研究方法	7
二、研究步驟	7
(一)資料分析與項目設定	12
(二)確定所需代碼	12
(三)確定資料建檔流程	13
(四)訂定資料輸入規範	14
(五)確定資料內容查驗方式	14
(六)確定資料庫群組化範圍	15
(七)確定全文檔案格式	15
(八)確定資料庫架構	16
(九)確定系統架構	16
(十)資料重整、查驗與饋入	18
肆、研究結果與討論	20
一、研究結果	20
(一)建立識別碼管理子系統	20
(二)建立線上資料輸入與審查子系統	21
(三)建立線上資料庫查詢子系統	24

(四)建立統計列印子系統	27
(五)建立 GRB 專屬網站	27
二、討論	28
(一)對政府推行總體計畫管考及資源分配的影響	28
(二)對各機關審核、管理及公開研究資訊的影響	28
(三)對資料庫使用者的影響	29
(四)對民意機關的影響	29
(五)對科資中心及 GRB 建檔的影響	29
(六)目前面臨的問題	29
伍、建議事項	32
參考文獻	34
附錄	
一、行政院所屬各機關委託研究計畫管理辦法	
二、委託研究計畫作業規定	
三、國內外研究計畫資訊網址一覽表	
四、資料項目內容說明表	
五、檢索項目說明表	
六、輸出項目說明表	
七、代碼內容說明表	
八、資料填寫規範	
九、資料庫表格定義及關聯	
十、GRB 網站網頁範例	
十一、主管機關一覽表	
十二、八十八下半年及八十九年度各機關登錄情形	
十三、電話訪談結果	

表 目 次

表 1：GRB 建檔及服務發展歷程	3
表 2：GRB 系統建置之作業流程	8

圖 目 次

圖 1：GRB 作業關連圖	13
圖 2：資料庫群組化概念圖	15
圖 3：系統架構概念圖	17
圖 4：資料填寫及上傳模式	20
圖 5：識別碼申請及發放示意圖	21
圖 6：研究計畫摘要輸入與確認示意圖	22
圖 7：期中報告摘要輸入與確認示意圖	23
圖 8：期末報告摘要輸入與確認示意圖	24
圖 9：資料庫之間與全文檔案之連結	26
圖 10：GRB 網站首頁	27

壹、緒論

一、研究緣起

在開始本研究之前，有必要對「政府研究計畫」的定義加以釐清。根據八十八年二月十二日行政院修正發布之「行政院所屬各機關委託研究計畫管理辦法」（如附錄一）第二條指出「本辦法所稱委託研究計畫，指各機關依業務需要，動用公務預算或其主管運用屬政府所有之基金作為研究經費，委託大專院校、研究機構、團體或個人執行具研究性質之計畫」。換言之，凡動用公務預算或政府機關主管運用之基金作為研究經費者都屬於政府研究計畫。

有關這些計畫的建檔，始自八十二年，當時國科會企劃處依行政院台八十二研展字第 四七 號令訂定發布之「行政院所屬各機關委託研究計畫管理辦法」開始辦理，所建檔案主要供科學技術統計要覽使用。科資中心則在八十六年七月接辦此項業務，並被要求公開上網提供服務。

檢視政府研究計畫的辦理方式，不外乎計畫主管機關委託專家或學研機構進行研究的「委託研究」、學者專家向機關申請的「補助研究」、機關內人員與其他機關人員或機關內人員與專家或研究機構合作的「共同研究」或「合作研究」，以及機關內人員的「自行研究」等。依計畫性質則分成「科技類」及「行政與政策類」，分別由國科會及研考會為中央主管及監督機關。推行多年以來，政府透過何種方式管理研究發展資源、資訊是否公開、計畫是否重複、是否同一個人在同一時間接案太多，以及成果應用為何等問題，屢次被立法院及監察院質疑，這也是國科會及研考會積極要求及督導政府研究計畫基本資料檔(Government Research Bulletin, 簡稱GRB)建檔及提供服務的原因，也是本研究必須在迫切的時間內邊研究邊開發，以獲得結果並付諸實施的原因。

二、研究目的

本研究探討的層面包括建檔機制的規劃及系統的設計與開發，具體的研究目的如下：

- (一)開發資料登錄與查核系統：研究設計可供各類型研究計畫與成果之摘要登錄與資料查核之系統，有效建立線上貼寫的網

路作業環境，由計畫主持人自行輸入摘要資料，再由計畫管考人審核，以提高資料的正確性及降低計畫遺漏的可能性。

- (二)建置完整的國家研究計畫及成果檔案：將研究計畫摘要及計畫執行結束後之成果報告之摘要與全文等予以電子化長期保存，使國家研究成果得以累積並延續發揮其效益於日後各領域之研究工作上。
- (三)提供多元化的資料庫服務：除提供資料庫查詢功能外，結合文字及影像，使研究計畫、研究成果報告、國科會專利等資料得以相互連結，並藉由網際網路提供服務，以擴大政府研究資訊的應用層面。
- (四)有效應用統計功能於資源分配上：每年可明確統計出各領域、學術研究機構、計畫主管機關及個人所分配獲得之研究發展資源，以及產出之成果所發揮的效益程度，藉以評估國家整體資源之應用是否妥當，並因應世界科技發展趨勢，檢討及調整未來之資源配置政策。

貳、國內外現況分析

一、GRB計畫依據與建檔發展歷程

早在八十二年國科會企劃處即根據八十二年二月八日行政院台八十二研展字第 四七 號令訂定發布之「行政院所屬各機關委託研究計畫管理辦法」開始GRB建檔工作，科資中心係在八十六年七月接辦此項業務，並在八十七年六月完成資料上網的雛型系統。八十八年二月行政院重新修訂管理辦法，國科會為因應網際網路的普及與發展趨勢，乃彙整研考會及相關機構意見，在管理辦法第五條、第十條及第十一條下訂定相關規範，包括研究計畫登錄及查詢、研究報告寄存作業，以及研究報告摘要資料庫的建立與服務作業規範等（如附錄二），並責成科資中心針對建檔方式及應用層面進行大幅度的變革。有關GRB的建檔及服務發展情形簡要歸納如表1。

表 1：GRB 建檔及服務發展歷程

82/02	建檔情形	由計畫主管機關填寫 GRB 表交國科會鍵入。
	服務情形	僅供科技動態統計參考。
86/07	建檔情形	由計畫主管機關填寫 GRB 表交科資中心鍵入。
	服務情形	公開上網，免費提供資料庫查詢。
87/09	建檔情形	由計畫主管機關上網填寫 GRB 表。
	服務情形	公開上網，免費提供資料庫查詢。
88/07 (本研究內容)	建檔情形	由計畫主持人上網填寫 GRB 表，並於計畫執行期間依進度上網填寫期中報告摘要及期末報告摘要，所填資料皆由計畫主管機關自行上網審核。
	服務情形	全文上網，免費提供查詢、列印及下載，並可提供統計報表數十種。

二、國內相關資料庫之建置及服務情形

國內最早處理及提供研究計畫資訊服務的單位，當屬國科會科資中心及農資中心在「科技發展資訊體系」下所建立的資料庫，農資中心係以農委會之農業科技及發展研究計畫為收錄範圍；科資

中心建置資料庫的歷史最早可回溯至七十一年九月開始，但資料庫正式有系統的對外提供服務是從七十七年十二月全國科技資訊網路(STICNET)對外服務開始。屬於政府研究資訊範疇的「中華民國研究計畫資料庫」係從 75 年 7 月開始蒐集，「中華民國研究報告資料庫」則回溯自 59 年度之資料。

爾後，在網際網路盛行的情形下，國科會、經濟部、衛生署、環保署等機關皆陸續將其主管之研究計畫上網提供查詢。本研究所稱之政府研究計畫資訊當然也包括這些機關的資料，換言之，除了在上述機關的網站中可以找到其主管之研究計畫資訊外，這些資料亦已納入本研究範圍。

由於學術研究活動對人類知識的累積有其延續性，為擴展民眾對政府研究資訊的瞭解及運用，本研究係在原有資料庫的基礎上，結合以往未公開的 GRB 檔，重新整建資料庫，並進行資料庫群組化的工作，期使計畫的執行與成果一體呈現。目前資料庫建置情形如下：

(一)研究計畫資料庫

本資料庫從民國 75 年之資料開始蒐集建置，但 82 年度起之資料才放入 GRB 系統。資料內容包括國科會專題研究計畫、各機關委託研究計畫及自行研究計畫，目前 GRB 系統內資料量約 11 萬筆。資料項目在重整之後，已與早期的「中華民國研究計畫資料庫」差異極大，主要是因為使用目的擴張所致。此資料庫除保留以往書目資料庫慣有的一些項目外，為能提供各機關的計畫管考需求，以及具體呈現研究資源的配置資訊，新增許多與經費及人力相關之資料項目。資料更新頻率原為每週一次，自九十年元月起提升為每日更新。

(二)研究報告資料庫

本資料庫從民國 59 年之資料開始蒐集建置，但 82 年度起之資料才放入 GRB 系統。資料內容包括國科會專題研究計畫、各機關委託研究計畫及自行研究計畫之報告摘要，GRB 系統內資料量 7 萬餘筆。資料項目與原有的「中華民國研究報告資料庫」相近，但可連結研究計畫資料庫及報告全文影像檔，便捷的回顧原有計畫內容及瞭解計畫執行成果。資料更新頻率維持每週一次。

(三)國科會專利資料庫

國科會多年來持續推動學術界與產業界之合作研究，支援補助各領域之研究計畫，迄至八十八年九月底止，研究成果共獲得國內外專利 1,024 件，包括中華民國專利 603 件、美國專利 321 件、日本專利 68 件，以及其他國家專利 32 件。為能將這些有市場價值且具實用之研究成果產出之專利，落實及擴散於產業界，科資中心與國科會綜合處於 87 年 8 月合作建立「國科會專利資料庫」。此資料庫目前仍維持獨立之查詢介面，未來將陸續與前二資料庫連結。

三、國外研究計畫資訊提供情形

資料庫製作從1970年代開始發展，早期要瞭解各國研究計畫資訊大多透過國際知名資訊系統，例如Dialog系統的Federal Research in Progress、Electric Power Database；以及STN系統的UFORDAT、FORKAT、FORIS等資料庫。但近幾年，因為網際網路的盛行，使得資訊的呈現與取得管道多元而豐富，其中不乏各國研究計畫資訊。例如，美國國家科學基金會(NSF)的FastLane、國家衛生研究院的NIH Grants Database及CRSP Database、環境保護署的Programs and Initiatives、航空太空總署的Technical Reports Server等，又如商務部的NTIS、能源部、農業部、國防部、National Academy Sciences等亦可在其網站中找到研究計畫。其中NTIS已建置多年，資訊來源包括全美200多個政府單位，如美國能源部 (Department of Energy)，國防部 (Department of Defense)、國家航空太空總署 (National Aeronautics and Space Administration) 及環境保護署 Environmental Protection Agency)等單位(NTIS 亦收錄其他國家政府及研究單位的研究報告，以擴大其收藏範圍)，可以說是全球最具規模的研究計畫成果報告資訊系統。

在歐洲方面，則以歐盟的ERGO (European Research Gateways On-Line)系統較具規模，該系統蒐集歐盟資助的所有研究計畫與研究報告資訊，資料提供者包括Austria、Belgium、Denmark、Finland、Germany、Greece、Iceland、Israel、Netherlands、Norway、Portugal、Slovenia、Sweden、United Kingdom等國家及EU本身的相關資料庫製作單位，該系統除參與單位在CERIF (Common European Research Information Format)共通規範下提供SGML (Standard Generalized Markup Language)檔建成資料庫外，它也是各參與單位相關系統的閘道。

當然，除了前述系統外，網路上可查詢的研究計畫資訊非常豐富，為能協助使用者方便取得國內外研究計畫資訊，本研究特整理國內外可供連結之研究計畫資訊網址如附錄三。

參、研究方法與步驟

本研究之範圍包括委託研究計畫資訊之蒐集與資料項目分析、系統設計與開發，以及網站建置等。本章針對研究方法及研究步驟說明如下：

一、研究方法

(一)現況探討

針對國內各機關委託研究計畫之建檔情形及國外相關系統建置及服務情形進行瞭解，做為本研究系統建構之參考。

(二)資料項目分析

將過去 GRB 之建檔結果、科資中心原有資料庫內容及資料庫群組化發展之需求，加以分析，俾提供系統最佳化設計之參考。

(三)系統設計與開發

因應網際網路的發展，從使用者介面、分散處理到中介軟體的應用，引進資料庫與 WWW 的連結技術，規劃設計一全功能之服務系統及建立 GRB 的專屬網站。

(四)系統工作日誌檔的分析與電話訪查

針對系統運作的實際執行情況，瞭解系統建構的成效與待改進之處。

二、研究步驟

本研究涉及所有行政院所屬各機關，運作機制的建立極為重要，所幸本案規劃及執行過程中，渥蒙國科會企劃處及研考會研展處處長及相關人員之協助，並督促各機關配合辦理，使本研究得以順利進行。全案參與者尚包括本中心索引人員及資訊人員，以及各機關計畫管考人、業務承辦人與計畫執行者。全程作業規劃可以下表流程說明：

表2：GRB系統建置之作業流程

作業內容	說 明
1.確定資料之涵蓋範圍及收錄年限	依據管理辦法，凡各機關依業務需要，動用公務預算或其主管運用屬政府所有之基金作為研究經費，委託大專院校、研究機構、團體或個人執行具研究性質之計畫都涵蓋在內。另國科會企劃處於88年10月發函通知各單位將科技類之自行研究計畫亦包含在內。此外，GRB於八十二年開始建檔，故資料收錄年限回溯自八十二年度。
↓	
作業內容	說 明
2.確定系統適用範圍	依據管理辦法須上網填寫資料者包括行政院所屬各機關、省府及地方政府機關(目前監察院亦考量使用本系統)。至於資料庫之查詢，則免費提供全國大眾使用。另外，統計及清單列印目前僅供各機關及計畫管考單位，如國科會及研考會使用。未來將考慮開放供大眾使用。
↓	
作業內容	說 明
3.資料分析與項目設定	檢討原有GRB表資料項目，訂定資料格式，決定檢索項目及輸出格式，並編訂資料項目內容說明表。
↓	
作業內容	說 明
4.確定所需代碼	資料編號採用科資中心資料庫編碼原則。職級、職別、參與性質等資料

	項目使用GRB原有代碼；分類碼除GRB分類碼之外，再加上科資中心分類碼；計畫編號依照各機關原有方式處理；機構代碼則使用國科會編訂之機構代碼，再輔以科資中心之代碼檔，補充代碼檔內容。
--	--



作業內容	說明
5.確定資料建檔流程	為提高資料的正確性，改變以往由國科會統一鍵入的方式，改由計畫主持人自行填寫，由業務承辦人或計畫管考人確認，再由科資中心進行邏輯檢誤。



作業內容	說明
6.訂定資料輸入規範	編輯設計網頁模式之研究計畫摘要、期中報告摘要及期末報告摘要登錄表格，並針對填寫方式、資料內容著錄規則、如何給予關鍵詞、如何撰寫摘要等加以規範，並撰寫說明文件。



作業內容	說明
7.確定資料內容查驗方式	包括系統自動檢查、人工檢查、批次檢查、個別線上更新等各種查驗方式。



作業內容	說明
8.確定資料庫群組化範圍	第一階段先連結研究計畫、期中報告、期末報告及全文檔案。未來再陸續連結國科會專利、機構資料及其他

	資料庫。
--	------



作業內容	說明
9.確定全文檔案格式	決定全文檔案格式、命名方式、放置方法，收錄年限，以及與資料庫之連結使用方法。



作業內容	說明
10.確定資料庫架構	本系統資料項目繁多，所需設計之表格包括研究計畫、期中報告、期末報告的所有基本資料、人力資料、中英文摘要資料，以及使用者資料及工作紀錄等。至於資料庫管理系統方面，資料填寫及更新之管控資料庫使用 Microsoft Access，對外提供之資料庫查詢服務則採用 IBM DB2。



作業內容	說明
11.確定系統架構	為因應網際網路的發展及免除使用者困擾，採用連結WWW與資料庫系統的方案，以瀏覽器當作前端介面，微軟 IIS (Internet Information Server)為網頁伺服器。為能連結不同平台(如UNIX或WIN)的資料庫及檔案系統，選用互動式資料庫中介軟體為開發工具，藉由其資料存取機制，只要是ODBC可以連結的物件，本系統都能存取；在UNIX平台上的資料庫，則藉由中介軟體所提供的native driver，配合該資料庫管理系統所提供的gateway軟體，達到連結的目的。



作業內容	說 明
12. 網頁設計與編輯等作業	向 TWNIC 申請 www.grb.gov.tw 的 Domain Name。由科資中心同仁自行架構網站及編輯網頁。在編輯網頁方面，使用 Front Page 及可做為 HTML 輔助編輯器的 Cold Fusion Studio，達到網頁設計及傳回資料動態顯示的效果。



作業內容	說 明
13. 資料重整、查驗與饋入	將82-88年度單機鍵入之資料與科資中心之研究計畫資料庫進行比對，去除重複，並補充缺項。將重整後之資料饋入ACCESS及DB2系統。



作業內容	說 明
14. 撰寫使用手冊	撰寫使用手冊、線上求助、問與答等輔助說明文件，並上線供使用者參考。



作業內容	說 明
15. 建立GRB系統	包括識別碼管控、線上填寫與審查、資料庫查詢、統計及清單列印等子系統。

以下一至十節將針對上述作業流程之執行情形加以說明。

(一)資料分析與項目設定

原有紙本式 GRB 表僅適用於研究計畫基本資料，已沿用多年，主要資料項目大致確定，很少更改，但本研究資料範圍除研究計畫外，尚有期中報告及期究報告，故針對以下各項重新加以確定。

1.資料項目

總計研究計畫、期中報告、期末報告所有資料項目多達百餘個；其中不需要經人力產生的自然資料項目(Natural Data Item)，包括計畫名稱、計畫期程、計畫經費、參與人力等皆由計畫主持人自行鍵入。需經人力思考判斷才產生的加值資料項目(Added Value Data Item)，如分類、關鍵詞等，由計畫主持人自行填寫上傳後，再由科資中心主題索引人員進一步加以修改。詳細資料項目及格式請參考附錄四資料項目內容說明表。

2.檢索項目

針對各資料庫內容分別訂定檢索項目，如計畫編號、計畫人員、計畫名稱、關鍵詞、年度、主管機關、研究性質、研究方式、研究領域等。詳細檢索項目如附錄五。

3.輸出項目

除細項經費、身份證字號、報告審查意見之外，其餘全部對外公開。詳細輸出項目如附錄六。

4.展現形式

為使檢索結果一目了然，讓一般大眾都看得懂，本系統的資料展現形式係直接冠以欄位名稱，並以表格方式呈現。在輸出媒體方面，目前僅提供網頁瀏覽，未來再配合新檢索引擎的應用，增加可供下載之文字檔案，以及電子傳遞等方式。

(二)確定所需代碼

本研究所需代碼包括資料編號、計畫編號、計畫屬性、研究性質、研究方式、計畫依據、職級代碼、學歷、性別、參與性質代碼、分類碼、機構代碼等。詳細代碼表內容說明請見附錄七。

(三)確定資料建檔流程

提高資料的正確性及降低計畫遺漏的可能性是 GRB 作業變革的主要目的之一，因此，除了行政院訂定發佈的管理辦法之制約外，在作業流程的設計上，亦期藉由計畫執行者、計畫主管機關之業務承辦人及計畫管考人，以及系統開發者的參與及互動，形成一維護責任共同體。換言之，除國科會專題研究計畫由國科會資訊小組提供外，其他部會署及地方政府機關之委託研究計畫資料，皆由最瞭解計畫內容的計畫執行者自行填寫資料，由委辦機關的業務承辦人或計畫管考人負責審核確認資料，由國科會科資中心負責系統開發維護及服務，國科會及研考會則分別就「科技類」及「行政及政策類」計畫進行監督。各環節應負責任與關連詳如圖 2。

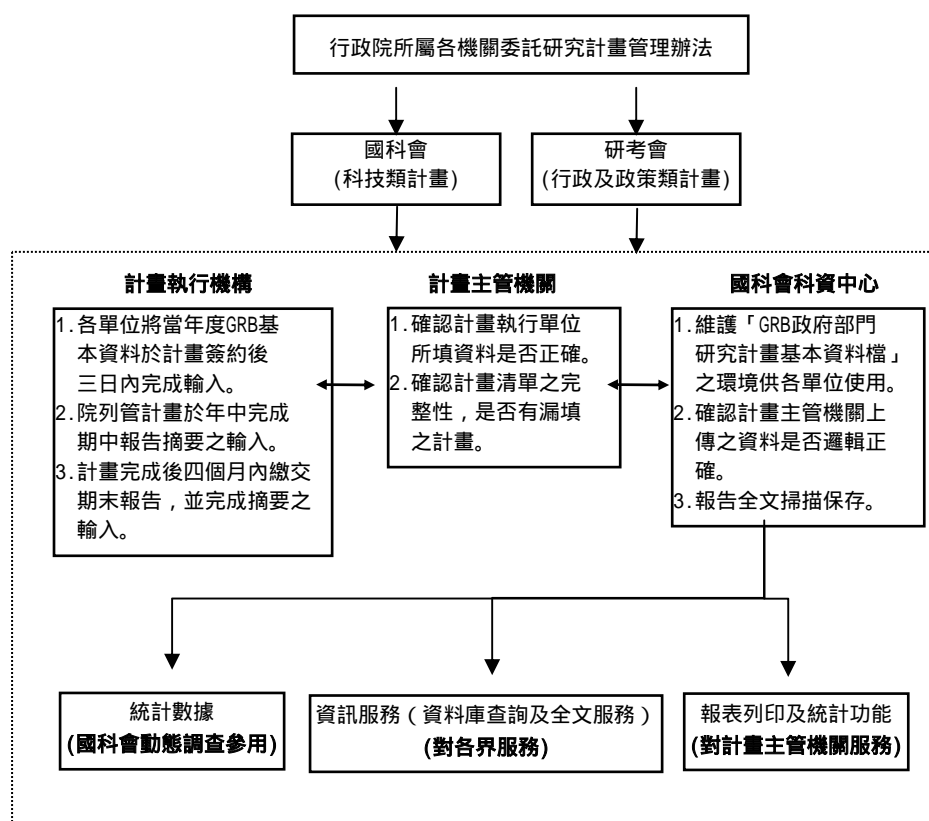


圖 1：GRB 作業關連圖

(四)訂定資料輸入規範

資料著錄的一致性，使資料呈現方式整齊劃一，是提高資料庫效能與品質的基本條件。由於 GRB 資料的填寫人對網路使用環境的熟悉度差異甚大，更不是具有圖書館學背景的編目人員，為確保資料的一致性，特訂定作業規範如下：

1.GRB 基本資料表填寫說明

包含研究性質、研究領域、計畫屬性、經費、人員專長、職級、學歷、參與性質等各欄位的說明及代碼。

2.資料內容著錄規則說明

包括中英文題名、標點符號、大小寫、縮寫、專有名詞、研究者中英文姓名、中英文關鍵詞的表示方式等等。

3.如何給予關鍵詞

主要說明關鍵詞的選取原則。

4.如何撰寫摘要

主要說明摘要的功能、摘要的種類、摘要長度、摘要內容，以及摘要撰寫注意事項等。

詳細規範說明文件如附錄八。

(五)確定資料內容查驗方式

在資料內容的管控上，則分成以下幾個層次：

1.系統自動檢查

系統可根據欄位的屬性及設定，在計畫主持人傳送資料時自動偵測錯誤並提示訊息。

2.人工檢查

計畫主持人填寫上傳之資料，需由計畫主管機關的業務承辦人或計畫管考人加以確認及上傳，最後再由科資中心於必要時輔以人工的邏輯判斷及修正，尤其是著錄格式最常被各機關所忽視。

3.批次列印作業

計畫主管機關可自行列印清單，進行書面的查核及通知執行單位上網更新。

4.個別線上更新作業

資料填寫人如發現錯誤，可隨時行上網更新，但仍需經由第二點所述的人工檢查流程進行各階段的確認工作。

(六)確定資料庫群組化範圍

為能使國內研究活動的呈現更為完整，本研究嘗試進行資料庫群組化的工作，將不同類型研究資訊加以整合，使計畫的執行與成果一體呈現。群組化設計概念如圖 2。

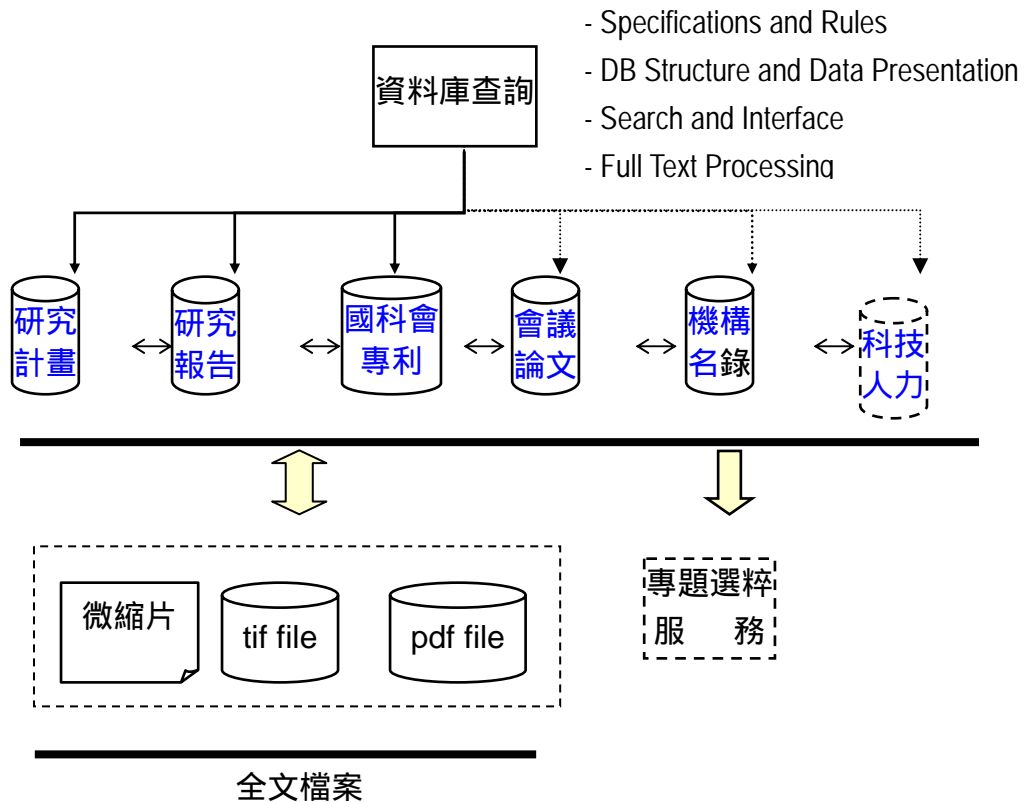


圖 2：資料庫群組化概念圖

(七)確定全文檔案格式

全文檔案格式分成兩種，紙本報告掃描以 tif 格式存檔，以電子繳交方式之 word 檔以 pdf 格式存檔。前者包括國科會完整報告及各機關委託研究成果報告，後者主要是國科會專題計畫的精簡報告。二者命名方式略有不同，國科會專題計畫報告以原計畫編號直接命名，各機關計畫編號之命名因原則不一，故無法直接使用為全文檔案之名稱，目前係以科資中心編號命名。全文檔案依類別及格式建置不同目錄來存放，資料庫中僅記錄檔案所在路徑，每筆記錄依科資中心編

號及計畫編號進行查詢。按照作業規範，該些檔案線上服務必須維持十年，原始影像圖檔則須永久保存。

(八)確定資料庫架構

由於此一管理系統為中型資料庫系統，可以採用之資料庫軟體相當多；在考慮方便性及自行開發等因素，最後選用 Microsoft Access 當做資料填寫及更新之管理資料庫，另外在提供給使用者查詢方面，為了提升整體查詢效能則採用 IBM DB2 當做後端資料庫系統。

在 Access 方面，由於使用者區分為執行機構、計畫主管機關業務承辦人及計畫管考人以及科資中心系統管理人員等不同層級，因此根據使用者之不同以及研究計畫、期中報告與期末報告等，設立相對應之表格(table)。另外，為了管控各個層級的填寫狀況，亦設立不同之紀錄表格來記錄各個使用者的填寫情形。為提供線上核發識別碼的功能，則必須另設表格來記錄系統全部使用者的個人資料。各個表格的定義及關聯性如附錄九。

(九)確定系統架構

由於本系統是經由網際網路來進行各項填寫及審核之工作，因此在進行構思系統架構之前，必須要先了解 Internet 的特性。Internet 具有多人同時上線及僅僅透過瀏覽器來與主機溝通之特性，因此若要開發相對應之應用軟體，宜採較接近 3-tier 的網路程式架構。亦即必須有一應用程式伺服器來執行所有程式邏輯，而使用者只是利用瀏覽器來下命令或編輯或顯示結果等。再者，開發工具亦會影響到程式流程的設計，在此系統的開發初期，我們曾比較過目前市面上常用之套裝軟體，最後選定使用 Allaire 公司的 Cold Fusion 產品當做中介軟體；再配合微軟公司的 IIS 網頁伺服器軟體當做提供服務的平台。在查詢服務方面，則是以 UNIX 平台搭配 Apache Web Server 再加上 IBM DB2 資料庫來提供服務，其中 DB2 中的表格定義完全與 Access 相同。整個系統設計原理及架構概念如圖 3。

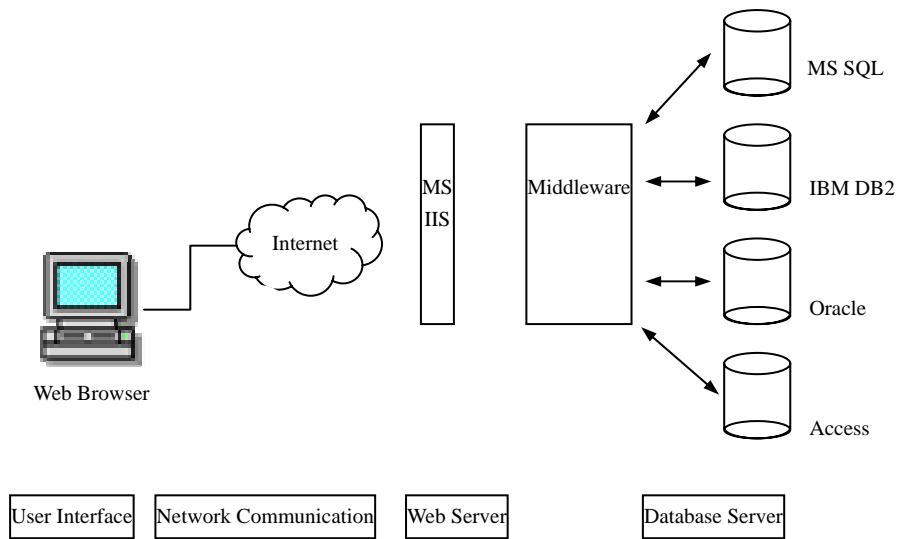


圖 3：系統架構概念圖

在系統功能設計方面則是按照需求，再根據 HTML 語法的特性逐一撰寫。主要的功能有使用者登入系統、選擇所要填寫或查看之項目(包括計畫、期中報告及期末報告)、填寫、更新、上傳、查詢、列印及各項統計等；當然針對不同層級的使用者有不同的功能。由於整個程式的開發有極大的時間壓力，因此是討論與寫碼同時進行，以致開發期間未及留下完整的程式流程圖。以下為使用者在填入帳號及密碼之後的驗證程式片斷：

```
<CFQUERY NAME="Checking" DATASOURCE="GRBDB">
    SELECT * FROM UserData
    WHERE  UserData.usr_id = '#form.T1#' and UserData.usr_pw =
'#form.T2#'
</CFQUERY>

<Html>
<head>
<meta charset="big5">
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
</head>
<body bgcolor="#FFFFCC">

<CFSET RQ_type = '0' >
<CFOUTPUT QUERY="Checking">

    <CFSET session.AUsrID=#usr_id# >
    <cfset session.UsrFld=#usr_field#>
    <cfset session.UsrKnd=#usr_kind# >
    <cfset RQ_status = #usr_status# >

    <CFIF #usr_kind# eq 1 >
        <CFSET RQ_type = '1' >
```

```

<CFELSEIF #usr_kind# eq 2 >
  <CFSET RQ_type = '2' >
  <cfelseif #usr_kind# eq 11 >
    <cfset RQ_type = '11' >
  <cfelseif #usr_kind# eq 12 >
    <cfset RQ_type = '12' >
  <CFELSE>
    <CFSET RQ_type= '3' >
  </CFIF>
</CFOUTPUT>

<CFIF RQ_type eq '0' >
  <H3> ID 及密碼錯誤, 請重新輸入!! </h3> <br>
  <a href = "index.htm" > 返回 </a>

<CFELSEIF RQ_type eq '1'>
  <cfif #RQ_status# is 1><!-- 主管機關 --->
    <cflocation url="/master/grb_special_2.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<CFELSEIF RQ_type eq '2'>
  <cfif #RQ_status# is 1 ><!-- 系統管理者 --->
    <cflocation url="/manage/manage.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<cfelseif RQ_type eq '11' > <!-- 執行機構專用 --->
  <cfif #RQ_status# is 1 >
    <cflocation url="/ExcuOrgan/excuOrg.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<cfelseif RQ_type eq '12' > <!-- 業務承辦人 --->
  <cfif #RQ_status# is 1 >
    <cflocation url="/undertaker/grb_special_2.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<CFELSE>
  <cflocation url="index.html" addtoken="Yes">
</CFIF>
</body>
</html>

```

(十)資料重整、查驗與饋入

如前所述, 本研究係在原有「中華民國研究計畫資料庫」及「中華民國研究報告資料庫」的基礎上, 結合以往未公開的 GRB 檔, 重新整建資料庫, 而科資中心以往建置書目資料

庫之目的在提供資料庫查詢服務，與現有系統的目的並不完全吻合，尤其是做為資源統計的項目大都缺乏，因此本研究須依據資料項目分析結果，進行所有資料之查驗，以及缺項之補充，再依新的資料庫架構饋入系統。

肆、研究結果與討論

一、研究結果

本研究在完成上述作業後，建立了GRB系統及專屬網站。此系統共分為四大子系統—識別碼管理子系統、線上資料輸入與審查子系統、資料庫查詢子系統、統計列印子系統。本章將針對各子系統之功能逐一說明。

(一) 識別碼管理子系統

在 GRB 的資料填寫及上傳作業中，各計畫主管機關可視計畫的多寡自行決定採單階或二階管理，輸入模式可歸納成下圖 4 之四種。

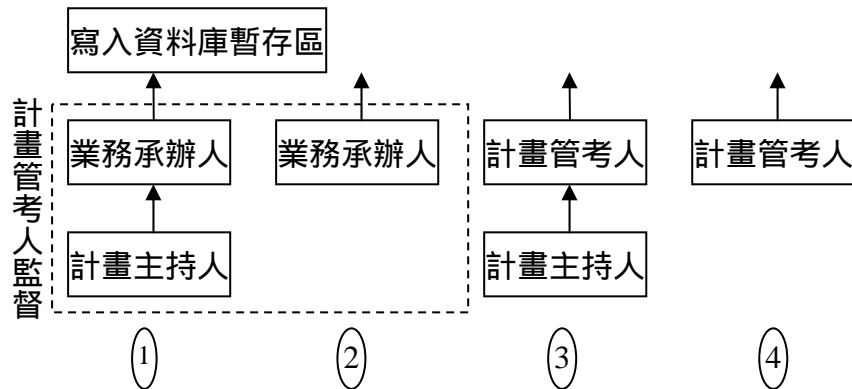


圖4：資料填寫及上傳模式

各計畫主管機關不論是採那一種資料填寫及確認模式，都只能有一個計畫管考人，所有計畫主持人與計畫主管機關業務承辦人之識別碼皆由計畫管考人審核發放；而業務承辦人則可以有多個，業務承辦人與計畫主持人之間的關係可以是一對一，也可以是一對多，二者之關連在計畫主持人審核發放識別碼時即須加以指定，以確保業務承辦人可以看到及修改自己主管的計畫。有關識別碼的申請及發放作業，系統皆自動以 email 方式通知申請者及發放者。申請與發放作業流程請參見圖 5 之示意圖。

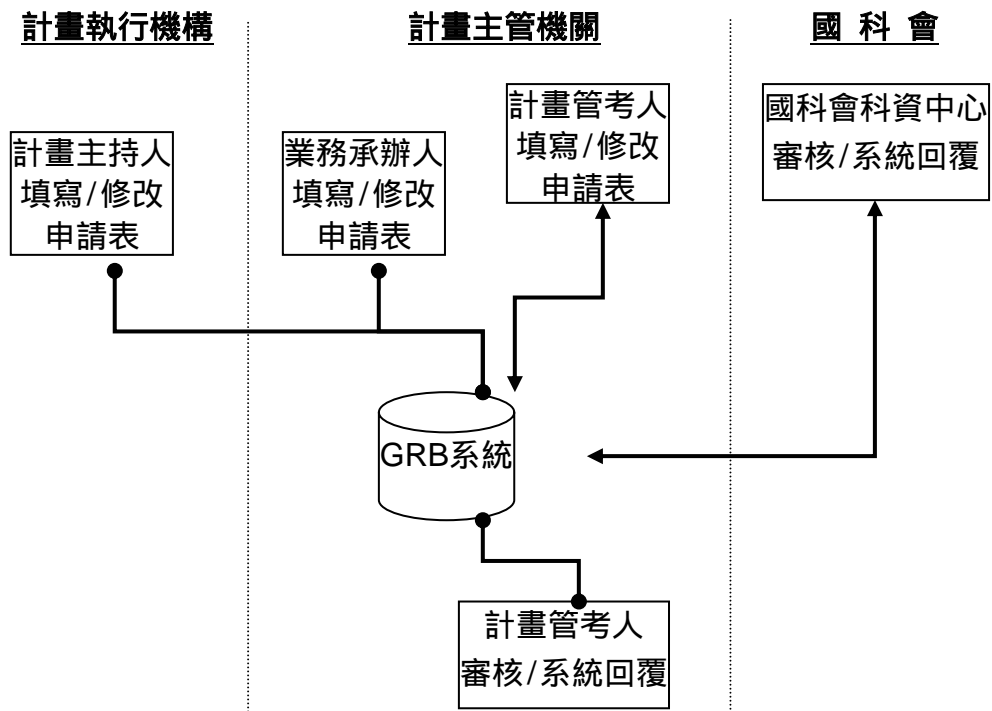


圖 5：識別碼申請及發放示意圖

(二)線上資料輸入與審查子系統

本子系統可提供研究計畫摘要、期中報告摘要及期末報告摘要之資料輸入介面，包括新增、刪除、修改等基本功能，並提供資料自動檢誤、人工審查確認等功能。以下即分別加以說明。

1.研究計畫摘要輸入介面

計畫簽約後三日內，計畫執行單位即應上網填寫。計畫類型不同所需填寫之內容不同，執行機構所填新增及修改之資料由計畫主管機關上網確認，系統會自動發 email 通知主管機關之承辦人。而計畫主管機關確認或新增之資料再由科資中心做最後確認後寫入資料庫，其作業流程概念如圖 6。

登錄後，主管機關未審查前，登錄者仍可修訂及刪除資料，唯資料一經主管機關審查後，登錄者僅能看到自己過去登錄的情形，但不能再去修改那些資料。待資料正式寫入對外提供查詢服務的資料庫後，登錄者才能再進行更新，資料更新流程如資料填寫流程，仍須經資料審查步驟才能寫入對外服務之資料庫。

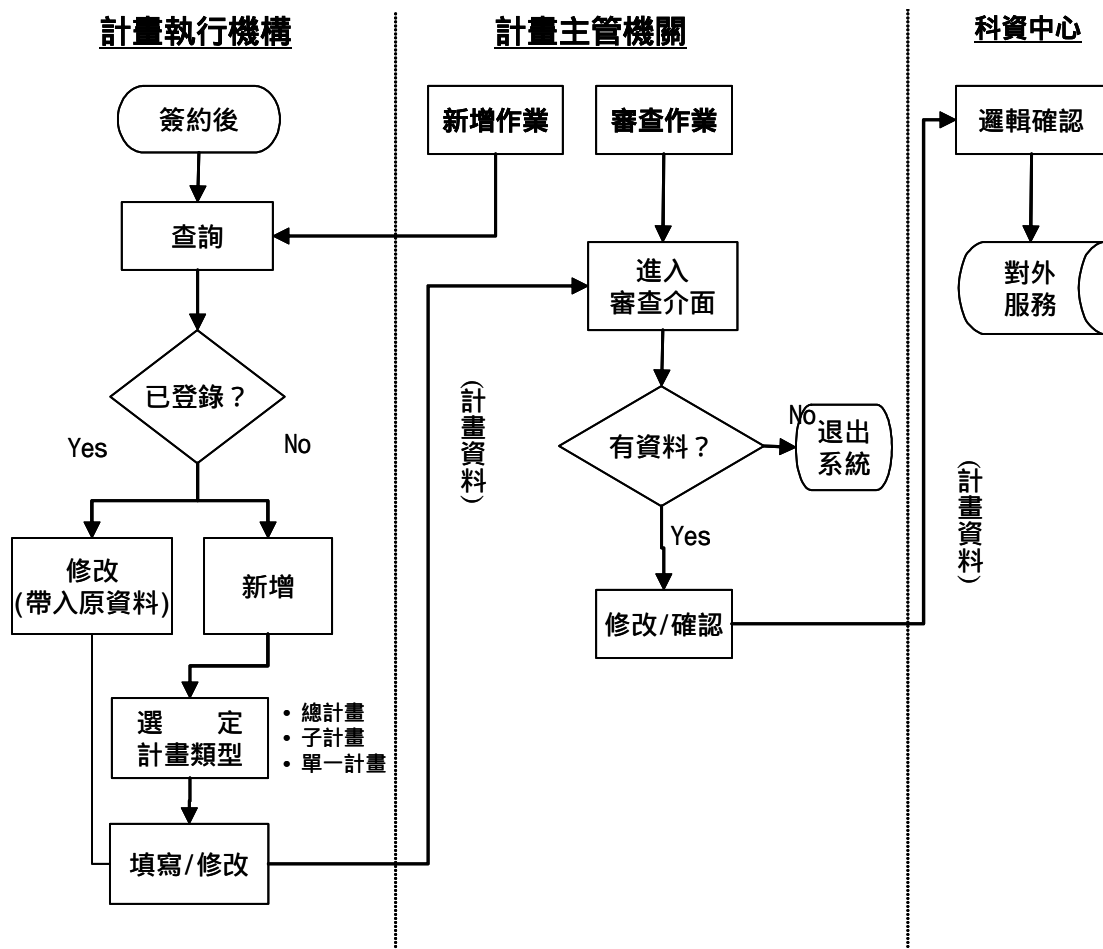


圖 6：研究計畫摘要輸入與確認示意圖

此輸入介面系統設有自動偵錯功能，例如：細項經費加總不等於本期總經費，同一計畫出資單位百分比加總不等於100%，參與計畫人力資料填寫不完整等，系統皆會顯示錯誤訊息，並提示使用者修正錯誤。

2.期中報告摘要輸入介面

登錄及確認作業流程與研究計畫摘要資料之功能大致相同，如圖 7。唯查無研究計畫基本資料者，應回研究計畫基本資料新增作業流程，予以補填。院列管計畫規定一定要填寫期中報告摘要，並繳交期中報告全文，其餘計畫由各計畫主管機關決定是否填寫。

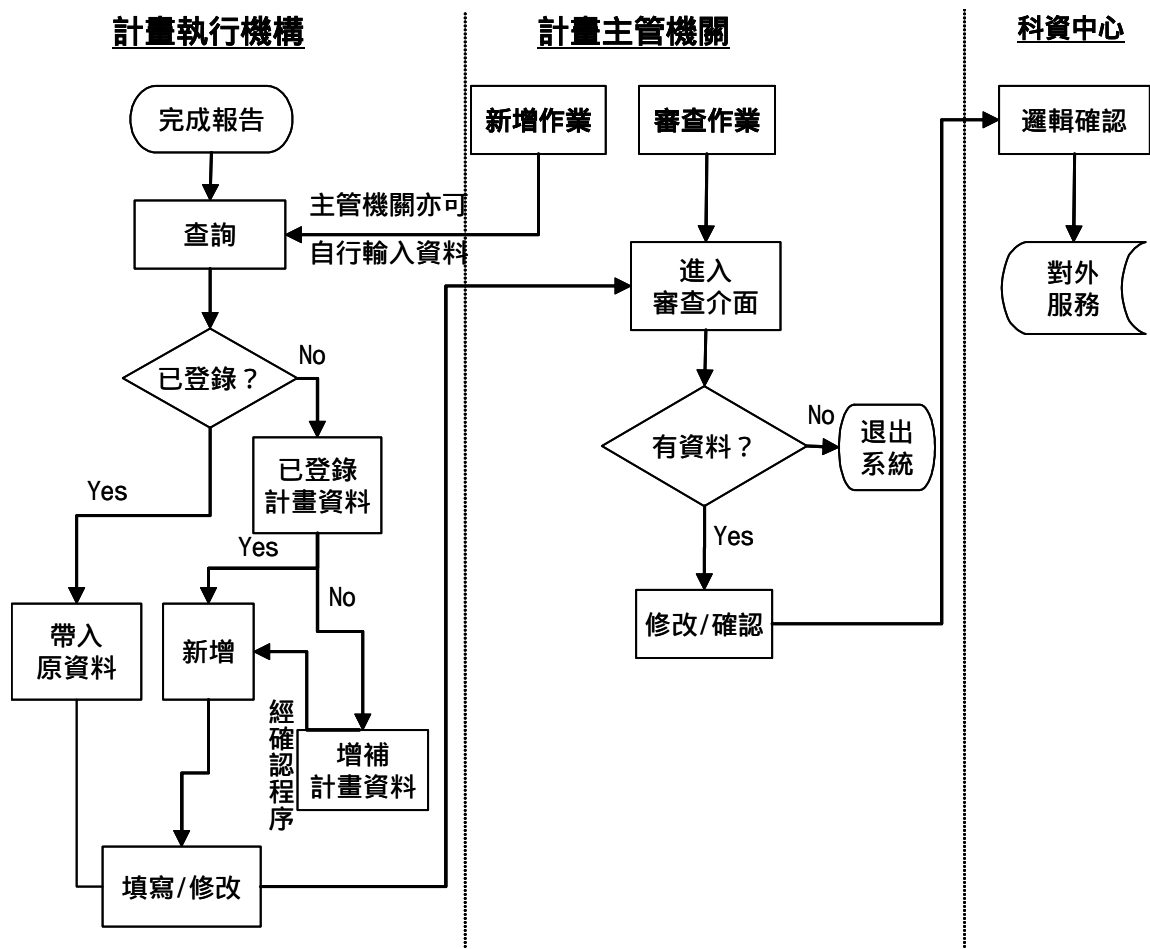


圖 7：期中報告摘要輸入與確認示意圖

3. 期末報告摘要輸入介面

登錄及確認作業流程與研究計畫摘要資料之功能大致相同，如圖 8。唯查無研究計畫基本資料者，亦應回研究計畫基本資料新增作業流程，予以補填。所有計畫在完成後四個月內應依管理辦法第十一條之研究報告摘要資料庫作業規範，上網完成摘要之輸入。並依據第十條之研究報告寄存作業規範，將非機密及限閱之期末報告，繳交給國家圖書館寄存，以及供科資中心掃描建檔與上網提供服務。

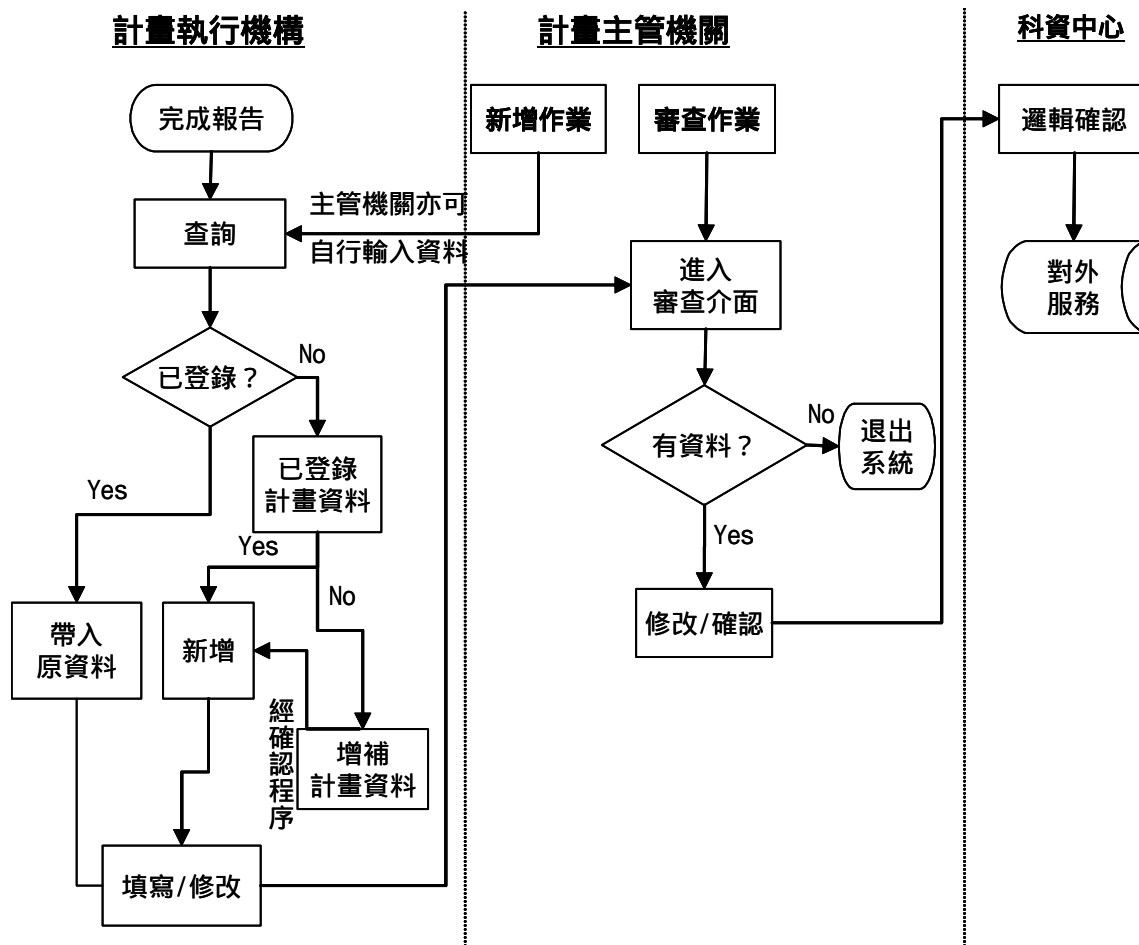


圖 8：期末報告摘要輸入與確認示意圖

4.資料確認、修改、刪除介面

此項功能係供業務承辦人或計畫管考人審核計畫主持人上傳之研究計畫摘要資料、期中報告摘要及期末報告摘要，並可加以修改或刪除。各業務承辦人僅能看到自己主管之計畫，計畫管考人則可以看到機關內所有之計畫。經由此確認程序所上傳之資料，將由科資中心進行最後之檢誤及著錄格式之查驗，最後饋入對外服務之資料庫。

(三)資料庫查詢子系統

本系統資料庫之查詢服務係以 Internet 廣大使用群為對象，使用者只要想知道國內有那些研究計畫在進行？計畫規模有多大？什麼人在研究什麼主題？是否有研究報告？都可以上網免費使用，且除了前述資料提供與審查子系統需要識

別碼外，此項對大眾提供服務的資料庫查詢子系統並未設定任何控制機制。檢索介面及查詢功能亦極簡易，使用者不需要經過訓練，即可直接使用，系統備有線上求助及使用手冊供使用者參考。以下即針對資料庫之查詢服務一一說明。

1. 研究計畫資料庫

可提供「計畫編號」、「計畫人員」、「計畫名稱」或「關鍵詞」等字串的查詢；另外，針對計畫的特定欄位進行查詢，可以同時設定多項欄位做組合查詢，目前提供之項目包括年度、計畫主管機關、研究性質、研究方式、研究領域等項，並可自行決定每次顯示查詢結果的筆數。

使用者在獲得查詢結果並選擇顯示資料時，可在顯示畫面進行「查詢結果再查詢」，此項功能目前有姓名、中文關鍵詞及英文關鍵詞等三個欄位。在這三個欄位出現的人名、中文關鍵詞及英文關鍵詞下都畫有底線，使用者只要點一下，即代表用此人名或關鍵詞再進行查詢。

另外，系統在顯示詳細資料時，將自動搜尋有無期中報告摘要、研究報告摘要或全文檔案，如果沒有，則該些欄位將略過，不予顯示；如果有，則會出現一個可連結的編號，點選該編號即可顯示期中報告摘要或期末報告摘要之內容。至於研究報告全文影像檔，如果有，則會出現「顯示全文影像(另一個螢幕)」字樣，並提供連結，點選該處即可顯示研究報告全文影像檔之內容。如果有報告微縮片之拍攝紀錄，則會出現該報告共有幾片幾幅，使用者可根據計畫編號向科資中心申請影印或複製。有關各資料庫及全文檔案之間的連結，請參考圖 9。

2. 研究報告資料庫

可提供「計畫編號」、「計畫人員」、「計畫名稱」與「關鍵詞」及「年度」的查詢。查詢結果、顯示資料，以及資料庫連結之功能皆與研究計畫相同。

3. 國科會專利資料庫

本資料庫提供關鍵詞、專利名稱、技術內容簡介、國際專利分類號、獲得專利權國家、證書號碼、公告編號、適用產業、學門等九個檢索項目。查詢結果顯示方式則有書目瀏覽及詳細資料兩種。其中書目瀏覽僅顯示專利名

稱、國際專利分類號、適用產業、授權情形等四項資料；詳細資料則顯示全部資料內容，共十一項欄位資料，並可直接連結全文影像檔案。本資料庫之國際專利分類號採用五階分類，適用產業則採用經濟部工廠普查及國科會科學園區產業分類法分類，共分成十九類。在資料庫群組化方面，未來亦將利用計畫編號與前兩項資料庫進行連結。

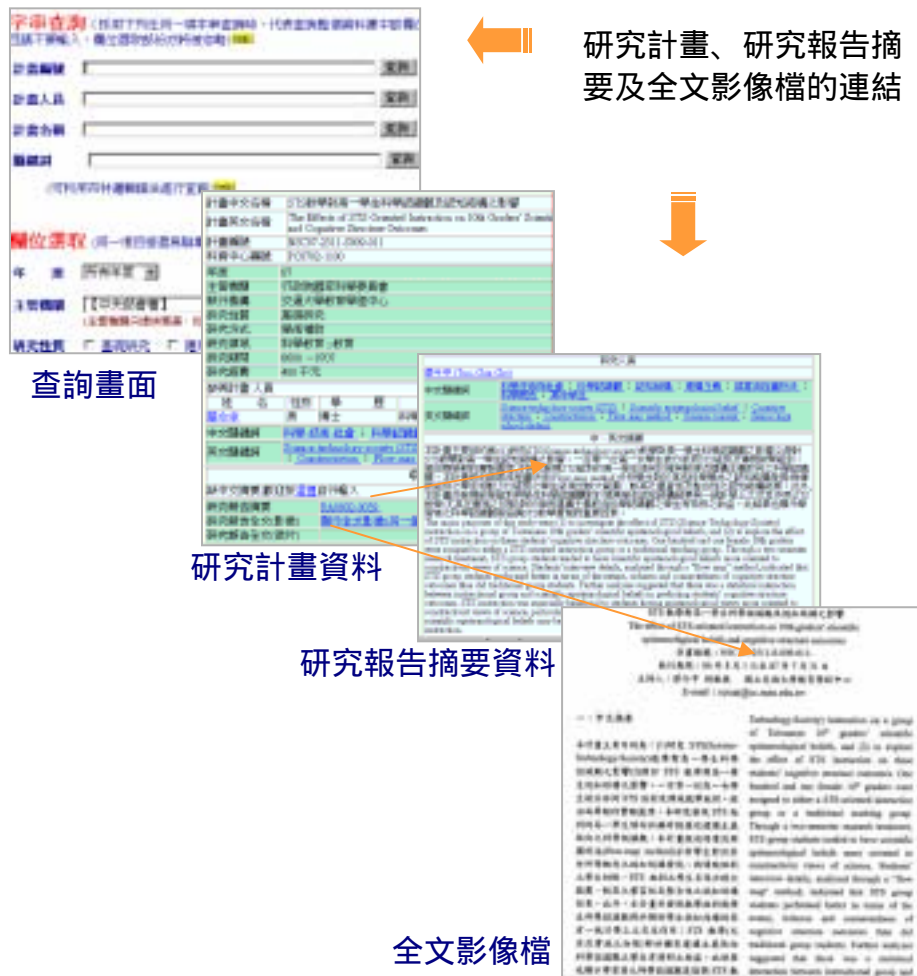


圖 9：資料庫之間與全文檔案之連結

(四)統計列印子系統

本子系統目前僅提供計畫主管機關使用，並未對一般大眾開放。其中計畫管考人可使用計畫清單列印功能，選擇列印不同年度不同類別之計畫，亦可列印計畫件數超過兩件以上之計畫主持人，以做為計畫審核之參考。另外，針對國科會及研考會這兩個分別主管科技類及行政政策類的中央主管機關，系統可提供數十種統計報表，包括研究經費及科技人力之配置資訊，內容包括研究經費、研究性質、研究領域，以及研究人員的職級、學歷、性別、年齡層等統計項目。這些資料可供整體性及重點計畫的觀察，也是呈現研究成果及將來分配預算的重要參考資料。

(五)GRB 專屬網站

由於本系統實際上是由各機關共同維護，並非科資中心所個別擁有，為提供一個讓大眾容易記憶的網址，乃向 TWNIC 申請 www.grb.gov.tw 的 Domain Name，並由科資中心同仁自行架構及編輯網頁，完成建立 GRB 專屬網站，首頁樣式如下圖 10。網站內容以前述各項子系統的功能為主，再輔以線上求助、使用手冊、問與答等說明文件。網頁範例請見附錄十。



圖 10：GRB 網站首頁

二、討論

GRB的建檔在經過這一年多以來的服務後，已引起廣泛的注意，計畫主持人及各機關在管理辦法及相關作業規範下皆盡力配合。本節將針對GRB對涉及各個層面有何影響，以及目前遭遇的困難加以探討，以期進一步對未來發展方向提出具體的意見。

(一)對政府推行總體計畫管考及資源分配的影響

本研究實施以來，目前系統內已累積計畫主管機關共有 100 個，其清單如附錄十一；在八十八下半年及八十九年度透過本系統上網填寫及確認資料的主管機關有 70 個，各機關填寫情形請見附錄十二。從參與機關的名單看來，GRB 的建檔在計畫的完整性方面應可逐漸提高比率。此外，系統所提供的多種統計報表，可提供研究經費及科技人力之配置資訊，是呈現研究成果及將來分配預算的參考資料，亦可做為管考機關整體掌握及重點觀察政府研究計畫輔助參用。

(二)對各機關審核、管理及公開研究資訊的影響

國科會及研考會都已將 GRB 的建檔納入其監督的委託研究計畫管理制度，目的除公佈政府研究資訊外，更希望各單位在進行選題或徵求研究主持人時，避免計畫重複研究或讓同一人於同一時間參與太多的研究計畫，對於連續幾年委託同一研究主持人之計畫亦期特別加以管理。因此，GRB 已成為各機關委託研究計畫的選題、研究成果的公開與運用等一連串管理機制中的重要一環。

另外，依據已登錄之計畫件數統計，行政院所屬各機關在八十九年度計畫件數 5 件以下者有 25 個，6-30 件者有 25，31-100 件者有 16 個，101-200 件者有 4 個，201 件以上之機關有 3 個。根據本研究小組以電話訪談 70 個計畫主管機關的結果顯示，共有 49 個計畫管考人表示，GRB 的建檔對其掌握機關內研究計畫的作業情形有幫助，另有 22 個計畫管考人認為沒幫助。後者大都是計畫項數很少，或者是計畫件數很多自有其管理系統之機關。另外，在受訪單位中，有將研究計畫及成果報告建檔者(包括非電子檔)有 47 個，尚未建檔者有 24 個。有對外提供服務者 44 個(包括圖書室陳列)，研究成果僅供內部參考者有 27 個。從這些數字看來，

各機關實際已累積研究計畫資訊者較已經在機關網站提供服務者普遍許多。上述電話調查訪問結果請見附錄十三。

(三)對資料庫使用者的影響

GRB 網站是提供我國研究計畫及成果資訊最為完整的網站，無論是學者專家或一般大眾，凡關心政府研究計畫者皆能免費上網查詢所需資料。使用介面易學易懂，適合各種不同層面的使用者。系統開放以來，每月平均上網人次約一萬人，以 login 的 ip 分析，學術研究界使用者占 70%、政府機關占 25%，產業界占 5%。全文傳遞服務則如前述，電子檔可線上瀏覽、列印及下載，微片資料則需向科資中心申請影印。其中全文服務對使用者的影響可能最大，這項服務已打破以往必須提出申請才能獲得全文的限制。

(四)對民意機關的影響

最近幾年，立法院及監察院經常針對政府投入研究經費及計畫執行成效提出質詢及糾正，GRB 的建檔成果，更方便民意機關引用為執行任務時之輔助資料。

(五)對科資中心及 GRB 建檔的影響

過去 GRB 的建檔採單機作業，工作期間有一定限制，國科會及後來接續建檔的科資中心必須負擔所有的鍵入工作，計畫常有遺漏，在應用上也不是很普遍。GRB 系統的建置，可以說在建檔方式及應用方面都進行了大幅度的變革，也改變了中心資料庫的服務層面。在系統的開發上，更首次完全由中心同仁自行負責，由於時間緊迫，規劃及系統文件的準備皆未完備，但在功能調整及系統更新上，具有高度的彈性，並節省了委外開發的大筆經費。

(六)目前面臨的問題

1.各機關管理方式不一，計畫管考人更換頻繁

在前述的電話受訪單位中，表示已針對委託研究計畫訂定有管理辦法或相關作業規範者有 58 個，尚未訂定者 13 個。從這些數字看來，大部份的機關皆已在委託研究計畫管理辦法第四條的規範下，訂定委託研究計畫作業規定，統籌管理所屬機關(單位)之委託研究計畫，所訂定內容大致與母法相去不遠。但估且不論管理制度是否完備，以

實際的執行內涵而言，各機關因為計畫件數的差距而有所不同。僅以最簡單的計畫編號為例，目前還有部分機關尚未賦予所管計畫一個有意義的計畫編號。另外，各機關計畫管考專責單位不一，有的並未指定專責單位，有些單位的總計畫管考人及業務承辦人更換頻繁(此次電話訪問即發現許多計畫管考人已經更換)，對計畫管理作業的延續性，造成極大的困擾。除此之外，各機關因科技計畫預算規模的差異，在管理上，計畫的形成往往也是標準不一，有以總計畫當作一個計畫而登錄者，有以子計畫登錄者，在計畫的從屬關係上並未有效釐清，這也是 GRB 為什麼先進行跨資料庫之間的群組化工作，卻將總計畫與子計畫的關連應用放在下一階段的原因。

2.分類架構的合理性問題

GRB 的分類表已使用多年，此分類表非常簡易，採單階架構，可能是早期為方便操作或統計而設計。在 82 個類別中，自然科學 9 類、工程 22 類、醫學 11 類、農業 19 類、人文社會 21 類，在架構上顯得並不平衡，新學門的歸屬亦有困擾，應有檢討的必要。

3.全文檔案傳輸問題

目前 GRB 系統內研究報告的檔案格式以 tif 及 pdf 為主，使用者只要安裝閱讀軟體或利用瀏覽器內建的應用程式，原則上都可以閱讀，但檔案大小因原文內容的規模差異極大，其中國科會的專題研究計畫成果報告，平均頁數為 6 頁，檔案大小在 300Kbyte 左右，但各機關的完整報告平均頁數為 160 頁，檔案大小可能高於 10 幾個 Megabyte 以上，傳輸問題在頻寬尚未解決之前確有其困難。

4.使用者介面的問題

編輯過網頁的人都能體會 Netscape 及 IE 兩大瀏覽器的戰爭所造成的困擾，GRB 的設計從網頁瀏覽效果、網址的描述，到資料傳輸都受到它的干擾。另外，資料填寫人對網路使用環境的熟悉度差異甚大，許多使用者甚至從來不曾上網，雖然系統內已備有操作手冊、Q&A 及線上求助，但仍然得耗費人力在電話中即時答覆及訓練使用者。

5.資料更新時效的問題

在研考會於八十九年十一月中旬至十二月初所主辦的「八十九年度行政院所屬各機關行政及政策類委託研究查核作業」中，部份機關反應由於 GRB 處理速度不夠快，從計畫主持人填寫、計畫主管機關確認資料，到科資中心上線，所需時間超過一週，以致無法即時知道計畫主持人是否同時向其他機關申請委託研究計畫，導致同一人獲得太多的計畫，並非各機關所願，雖然本研究小組已表示將把資料更新頻率從一週改為每日更新，但若各環節的相關人員無法即時配合，處理速度似乎仍無法滿足各機關期望即時掌握的需求。

伍、建議事項

本研究小組利用中介軟體連結後端資料庫的技術，成功的建置一套完整的、可實際應用的系統，在技術的應用方面，也許不是創新做法，但在短時間內要瞭解多層次管理的作業流程及複雜的網路作業環境，且需面對廣大且經驗不一的使用者，系統開發的困難度不難想像。經過這一年半的運作，在各機關的配合下，政府研究資訊的建檔、處理與利用方得以順利進行並深受重視。檢視前述的研究開發結果及所遭遇的問題，本研究小組特提出以下建議事項，以供未來開發全服務系統之參考。

一、建立快速通報系統

為能即時報導各機關的委託情形，建議開發快速通報系統，使各機關業務承辦人一旦簽約可立即上線填寫，且即時上網供各機關查詢，為求速度，此系統應只提供簡單的資料項目，俾供各機關隨時掌握計畫主持人接案情形。

二、建立計畫項目事前提交事後查驗的機制

本系統在資料蒐集的完整性方面一直被要求及詢問是否達到100%。事實上，目前GRB與各機關的關係發生在計畫簽約後，因此我們沒有完整的計畫清單得以與系統比對，俾進一步進行催缺的工作，僅能一年多次發函（含email）通知各機關計畫管考人自行清查並要求所屬單位儘快補登。未來若能建立計畫審議階段即提供細部計畫清單，系統事後分階段進行比對查驗，定期提醒各機關上網登錄，則資料的完整性方能達到各界的期望。

三、研訂資料交換與提供的機制

本研究執行期間多次接到其他機構提出想要連結系統資料之需求，在分析各種資料庫的整合做法，如Z39.50資訊檢索應用協定或SGML、XML標識語言的利用等，並衡酌國際相關發展趨勢後，我們認為XML技術的應用是較具發展性的做法，但在此之前，GRB的文件類型定義 (Document Type Definition, DTD)需先訂定。

此外，GRB的原始資料已越來越多人索取，索取目的不一，站在政府資訊公開的角度上，除了回饋各機關供其管理用途外；一般大眾為進行研究或統計所需之資訊，也應在政府資訊公開的服務範圍內。因此，民眾如何申請、科資中心如何提供、提供什麼內容等等，其辦法都有待進一步的釐清與訂定。

四、重新調整分類架構

現有分類表除GRB使用外，並未使用於其他地方，機構代碼編碼原則亦無法立即因應國際間的分類標準，因此當所提供資料需與其他統計資料來源比較分析時，可能造成困擾，建議參考國際相關標準，重新訂定分類表，或以新增方式，針對機構性質加以分類，以符合國際間如OECD之統計項目。

五、加速朝全服務目標推進

應繼續進行資料庫群組化的工作，利用專利資料庫之計畫編號與研究計畫及研究報告連結，進一步還可規劃學術研究機構資料庫、學術研討會訊息、學術會議論文資料庫，以及目前規劃建立之博士級科技人力資源庫的連結與服務方式。亦應發展個人化專題選粹資訊服務 (Selective Dissemination of Information, SDI)，儘快完成使用者profile管理、mailing list建立與維護，以及自動傳送等相關功能之規劃與開發，提供使用者定期接收最新研究計畫資訊的現代化個人服務。

六、建議更改系統中文名稱

「政府研究計畫基本資料檔」這個名稱已沿用多年，唯系統以單機方式初建之時，確實只有基本資料，但發展至今，資料內容已大幅擴充，且加入服務功能，原有名稱似乎無法反映系統內涵，因此建議系統名稱修改為「政府研究計畫資訊公告系統」。

參考文獻

一、標準與規範

1. 行政院所屬各機關委託研究計畫管理辦法，行政院於 88 年 2 月 12 日行政院台八十八研展字第 00812 號令修正發布。
2. 各機關委託研究計畫登錄及查詢作業規定，行政院國家科學委員會於 88 年 7 月 17 日(88)臺會企字第 025543 號函通知各機關辦理
3. 各機關委託研究報告寄存作業規定，行政院國家科學委員會於 88 年 9 月 19 日(88)臺會企字第 030212 號函公布。
4. 各機關委託研究報告摘要作業規定，行政院國家科學委員會於 88 年 7 月 17 日(88)臺會企字第 025543 號函通知各機關辦理。
5. 政府科技計畫審議管考評估作業手冊，國科會編印，民國八十七年。

二、圖書與論文

1. 科學技術資料中心，書目資料庫製作：文獻分析與處理，台北：科資中心，民國八十一年。
2. 張東森，“分散式書目資料庫整合技術概況”，華文書目資料庫合作發展研討會論文集，頁 171-177，民國八十九年。
2. 張永慶，“資料庫中介軟體與 WWW”，網路通訊，頁 44-49，民國八十五年。
3. Baru, Chaitanya. (2000). "Information Integration Technologies for Current Research Information Systems," Paper presented at the *CRIS 2000---Knowledge at Work - Research Information for Society*, Helsinki, Finland, May 25-27, 2000.

三、技術文件

1. Allaire Corporation , Administering ColdFusion Server , 1999.
2. Allaire Corporation , Developing Web Applications with ColdFusion Server , 1999.

3. Allaire Corporation , CFML Language Reference , 1999.
4. Calene Janacek and Dwaine Snow , DB2 Universal Database Certification Guide - second Edition , IBM International Technical Support Organization, 1997.

附 錄

- 附錄一：行政院所屬各機關委託研究計畫管理辦法
- 附錄二：委託研究計畫作業規定
- 附錄三：國內外研究計畫資訊網址一覽表
- 附錄四：資料項目內容說明表
- 附錄五：檢索項目說明表
- 附錄六：輸出項目說明表
- 附錄七：代碼內容說明表
- 附錄八：資料填寫規範
- 附錄九：資料庫表格定義及關聯
- 附錄十：GRB 網站網頁範例
- 附錄十一：主管機關一覽表
- 附錄十二：八十八下半年及八十九年度各機關登錄情形
- 附錄十三：電話訪談結果

【附錄三】

國內外研究計畫資訊網址一覽表

一、國內

- 政府研究計畫基本資料檔(GRB)
<http://www.grb.gov.tw>
- 國科會專題研究計畫
<http://www.nsc.gov.tw/project.htm>
- 研考會：行政院列管計畫資訊查詢系統
<http://w3.rdec.gov.tw/grpmis/gpmweb/main.asp>
- 衛生署資訊走廊
<http://www.doh.gov.tw/lane/research/>
- 環保署委託研究計畫查詢
<http://www.epa.gov.tw/tech/>
- 經濟部科專技畫總覽
<http://203.67.30.32/success/plan.asp>
- 經濟部科技專案計畫執行成果摘要資料庫
<http://sql.tier.org.tw/dbase/execresult.asp>
- 經濟部科技專案計畫可移轉技術資料庫
<http://sql.tier.org.tw/dbase/techmove.asp>
- 經濟部水資源局水力櫥窗計畫成果
<http://www.wrb.gov.tw>

二、國外

1.美國

- 國家科學基金會(NSF)/ Award Search | NSF FastLane
<https://www.fastlane.nsf.gov/a6/A6Start.htm>
- 國家衛生研究院/CRISP Databases
<https://www-commons.cit.nih.gov/crisp/>
- 環境保護署(EPA)/Programs

<http://www.epa.gov/epahome/programs.htm>

- 航空太空總署(NASA)/NTRA - NASA Technical Reports Server

<http://techreports.larc.nasa.gov/cgi-bin/NTRS>

- 航空太空總署(NASA)/Center for AeroSpace Information
Technical Report Server

<http://www.sti.nasa.gov/casitrs.html>

- 能源部(DOE)/ DOE R&D Project Summaries

<http://www.osti.gov/rnd/dbhome.html>

- 農業部(USDA)/ USDA Research Database (CRIS)

<http://www.usda.gov/search.html>

- 國防部技術資訊中心資料庫

<http://stinet.dtic.mil>

- National Technical Information Service (NTIS)/Technical Reports

<http://www.fedworld.gov/ftp.htm>

- National Academy Sciences/The Current Projects System

<http://www4.nationalacademies.org/cp.nsf>

2.加拿大

- 加拿大卑詩省科學委員會/Funding Programts

<http://www.scbc.org/programs/>

3.歐盟

- CORDIS (The Community Research and Development
Information Service)

<http://www.cordis.lu/>

- ERGO (European Research Gateways On-Line)

<http://www.cordis.lu/ergo/home.html>

- PROSOMA_The Technology Archive

<http://www.prosoma.lu>

2.英國

- EPSRC - Engineering and Physical Science Research

Council/Programmes <http://www.epsrc.ac.uk/EPSRCWEB/DIPS/progs.htm>

- MRC - Medical Research Council/Current Research
http://www.mrc.ac.uk/c_r1.html
- CCLRC - Council for the Central Laboratory of the Research Council
<http://www.cclrc.ac.uk/>
- NERC (Natural Environment Research Council)
<http://www.nerc.ac.uk/>
- PPARC - Particle Physics and Astronomy Research Council
<http://www.bbsrc.ac.uk/>
- BBSRC - Biotechnology and Biological Sciences Research Council
<http://www.bbsrc.ac.uk/>
- ESRC - Economic & Social Research Council
<http://www.esrc.ac.uk/>
- The British Academy
<http://www.britac.ac.uk/>

3. 德國

- BMBF
<http://www.bmbf.de/>
- The Max Planck Society (MPG)
<http://www.mpg.de/>
- The Fraunhofer-Gesellschaft (FhG)
<http://www.fhg.de/english/index.html>
- Deutsche Forschungsgemeinschaft (DFG)
<http://www.dfg.de/>

4. 法國

- SECRETARIAT D'ETAT A LA RECHERCHE
<http://www.mesr.fr/>

5. 奧地利

- Data Exchange on Research Projects and Institutions
<http://derpi.tuwien.ac.at/>

- New Austrian Research Information System (AURIS)

<http://www.auris.ac.at>

6. 比利時

- Flemish Research Database IWETO

http://www.vlaanderen.be/IWETO/DOCS/thema_pag.html

7. 瑞典

- The Swedish Research Council/NFR Allocated grants 1995/96 and 1997

<http://www.nfr.se/bevbidr/bevbidr.html>

- NUTEK (<http://www.nutek.se/>)

- TFR (<http://www.tfr.se/>)

8. 瑞士

- Swiss Education

<http://www.switch.ch/edu/edu-map.html>

- Scientific Report - Federal Institute of Technology Lausanne (EPFL)

http://research.epfl.ch/research/Webdriver?MIval=IRS_nav_index&MIlang_en=2

- Research Report - Federal Institute of Technology Zurich (ETHZ)

<http://www.rereth.ethz.ch/>

9. 義大利

- National Research Council

<http://www.newtours.it/geology2004/cnrsmc.htm>

10. 挪威

- Norwegian Current Research Database

<http://taxus.uib.no/fnsig/forsk/welcome.html>

- FORSKDOK

<http://www.bibsys.no/bibsys.html>

11. 以色列

- INDARD (The Israel National Databases for Academic Research)

and Development)

<http://www.israd.ac.il/INDARD>

12. 愛爾蘭

- Irish Internet Server Sites

<http://ireland.iol.ie/resource/ireland/irsites.html>

13. 冰島

- World Wide Web Servers in Iceland

<http://www.isnet.is/WWW/servers.html>

14. 荷蘭

- Dutch Research Database

<http://www.niwi.knaw.nl/us/homepag.htm>

15. 丹麥

- DANDOK Database (The Danish National Research Database)

<http://www.forskningsdatabase.dk/uk/indexuk.htm>

16. 芬蘭

- VTT Projects

<http://otatrip.hut.fi/vtt/ture/search.html>

- TUHTI

<http://www.helsinki.fi/lumme>

17. 克羅埃西亞

- Ministry of Science and Technology

<http://www.mzt.hr/mzt/eng/index.htm>

【附錄五】

檢索項目說明表

研究計畫資料庫

序號	欄位名稱	查詢方式	備註
1	計畫編號	字串	
2	計畫人員	字串	檢索結果再查詢，布林邏輯
3	計畫名稱	字串	布林邏輯
4	關鍵詞	字串	檢索結果再查詢，布林邏輯
5	年度	下拉選單	
6	主管機關	下拉選單	
7	研究性質	Check box	
8	研究方式	Check box	
9	研究領域	下拉選單	

研究報告資料庫

序號	欄位名稱	查詢方式	備註
1	計畫編號	字串	
2	計畫人員	字串	檢索結果再查詢，布林邏輯
3	計畫名稱	字串	布林邏輯
4	關鍵詞	字串	檢索結果再查詢，布林邏輯
5	年度	下拉選單	

國科會專利技術成果

序號	欄位名稱	查詢方式	備註
1	關鍵字查詢	字串	
2	專利名稱	字串	
3	技術內容簡介	字串	
4	專利發明人	字串	
5	國際專利分類	字串	
6	獲得專利國	字串	
7	證書號碼	字串	
8	公告編號	字串	
9	適用產業	下拉選單	
10	學門	下拉選單	

【附錄六】

輸出項目說明表

研究計畫資料庫

序號	欄位名稱	備註
1	計畫中文名稱	
2	計畫英文名稱	
3	計畫編號	
4	科資中心編號	
5	年度	
6	主管機關	代入主管機關名稱
7	執行機構	
8	研究性質	代入研究性質名稱
9	研究方式	代入研究方式名稱
10	研究領域	代入研究領域名稱
11	研究期間	含起迄時間
12	研究經費	以千元表示
13	參與計畫人員姓名	僅列出計畫主持人、共同主持人及協同主持人，檢索結果可再查詢
14	性別	代入性別名稱
15	學歷	代入專長名稱
16	專長	代入專長名稱
17	職級	代入職級名稱
18	參與性質	代入參與性質名稱
19	中文關鍵詞	檢索結果可再查詢
20	英文關鍵詞	檢索結果可再查詢
21	中文摘要	

期中報告摘要

序號	欄位名稱	備註
1	PG8901-0417	
2	計畫名稱	
3	審議編號	
4	原計畫編號	
5	主管機關	
6	執行單位	
7	計畫主持人	

8	計畫連絡人	
9	聯絡電話	
10	傳真號碼	
11	期程	含起迄時間
12	全程經費	
13	本年度經費	
14	當年預定執行進度	
15	當年實際執行進度	
16	當年完成進度百分比	
17	全程預定執行進度	
18	全程實際執行進度	
19	全程完成進度百分比	
20	當年預定支用經費	千元
21	當年實際支用經費	千元
22	當年支用經費百分比	
23	全程預定支用經費	千元
24	全程實際支用經費	千元
25	全程支用經費百分比	
26	目標達成狀況	
27	資源使用情形	
28	主要執行成果	
29	計畫變更說明	
30	落後原因	
31	因應對策(檢討與建議)	

研究報告資料庫

序號	欄位名稱	備註
1	計畫中文名稱	
2	計畫英文名稱	
3	計畫編號	
4	科資中心編號	
5	補助機構	代入補助機構名稱
6	執行機構	
7	年度	
8	研究期間	含起迄年月
9	研究人員	含中英文姓名，檢索結果可再查詢
10	中文關鍵詞	檢索結果可再查詢

11	英文關鍵詞	檢索結果可再查詢
12	中文摘要	
13	英文摘要	

國科會專利技術成果

序號	欄位名稱	備註
1	專利名稱	
2	專利發明人	
3	發明人服務單位	
4	國際專利分類	
5	獲得專利權國家	
6	公告編號	連結 tif 全文檔案
7	專利類型	
8	證書號碼	
9	專利有效期限	
10	適用產業	
11	授權情形	
12	可提供授權時間	
13	國科會計畫編號	將來與研究計畫及研究報告之連結點
14	學門	
15	技術內容簡介	摘要說明

【附錄九】

資料庫表格定義及關聯

```
-- Table Definitions -----
CREATE TABLE ADMINMIDLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR VARCHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE VARCHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE ADMINMIDLOG
ADD CONSTRAINT PK_ADMINMIDLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_ADMINMIDLOG_PROJ
ON ADMINMIDLOG (PROJKEY)
;
CREATE INDEX S_ADMINMIDLOG_USER
ON ADMINMIDLOG (USER_ID)
;
CREATE TABLE ADMINPRJLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR VARCHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
UL_STATUS VARCHAR(10),
PNCH_DESC VARCHAR(100)
)
;
ALTER TABLE ADMINPRJLOG
ADD CONSTRAINT PK_ADMINPRJLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_ADMINPRJLOG_USER
ON ADMINPRJLOG (USER_ID)
;
CREATE TABLE ADMINRPTLOG
```

```

(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(20),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR VARCHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE VARCHAR(10),
LASTDATE VARCHAR(50)
)
;
ALTER TABLE ADMINRPTLOG
ADD CONSTRAINT PK_ADMINRPTLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_ADMINRPTLOG_PROJ
ON ADMINRPTLOG (PROJKEY)
;
CREATE INDEX S_ADMINRPTLOG_USER
ON ADMINRPTLOG (USER_ID)
;
CREATE TABLE CODEREF
(
UNIONCODE INTEGER DEFAULT 0,
DEPT_CODE VARCHAR(20),
NSC_CODE VARCHAR(20),
HR_CODE VARCHAR(20),
STIC_CODE VARCHAR(20),
ORGAN_DESC LONG VARCHAR,
FORMAL_NAME VARCHAR(80)
)
;
CREATE INDEX S_CODEREF_DEPT_COD
ON CODEREF (DEPT_CODE)
;
CREATE INDEX S_CODEREF_NSC_CODE
ON CODEREF (NSC_CODE)
;
CREATE INDEX S_CODEREF_STIC_COD
ON CODEREF (STIC_CODE)
;
CREATE TABLE DEPTNAME
(
ORGAN_CODE CHAR(9) NOT NULL,
GRB_CODE VARCHAR(4),
ORGAN_DESC VARCHAR(60),
ORGAN_EXCU_CODE CHAR(1),
EXCU_ORGAN_KIND CHAR(2),
ACCT_EXCU_CODE VARCHAR(4),

```

```

ORGAN_KIND CHAR(1),
ORGAN_ADDR VARCHAR(90),
ORGAN_TEL VARCHAR(25),
ORGAN_FAX VARCHAR(25)
)
;
ALTER TABLE DEPTNAME
  ADD CONSTRAINT PK_DEPTNAME
    PRIMARY KEY (ORGAN_CODE)
;
CREATE INDEX S_DEPTNAME_GRB_COD
ON DEPTNAME (GRB_CODE)
;
CREATE TABLE EXCU04
(
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
PRJTYPE CHAR(10),
LOGDATE DATE,
USERID VARCHAR(20)
)
;
CREATE INDEX S_EXCU04_USERID
ON EXCU04 (USERID)
;
CREATE TABLE EXCU05
(
PROJKEY CHAR(15),
PNCH_DESC VARCHAR(250),
DELI_NO VARCHAR(50),
PLAN_NO VARCHAR(50),
PLAN_ORGAN VARCHAR(50),
EXCU_ORGAN VARCHAR(50),
NAME VARCHAR(50),
CON_NAME VARCHAR(50),
TEL VARCHAR(50),
FAX VARCHAR(50),
PERIO_STYM VARCHAR(50),
PERIO_ENYM VARCHAR(50),
WHOLE_AMT INTEGER DEFAULT 0,
PLAN_AMT INTEGER DEFAULT 0,
YRATE_1 INTEGER DEFAULT 0,
YRATE_2 INTEGER DEFAULT 0,
YRATE_3 INTEGER DEFAULT 0,
TRATE_1 INTEGER DEFAULT 0,
TRATE_2 INTEGER DEFAULT 0,
TRATE_3 INTEGER DEFAULT 0,
YPAYOFF_1 INTEGER DEFAULT 0,
YPAYOFF_2 INTEGER DEFAULT 0,
YPAYOFF_3 INTEGER DEFAULT 0,
TPAYOFF_1 INTEGER DEFAULT 0,
TPAYOFF_2 INTEGER DEFAULT 0,

```

```

TPAYOFF_3 INTEGER DEFAULT 0,
AB_GOAL LONG VARCHAR,
AB_SOUR LONG VARCHAR,
AB_RESU LONG VARCHAR,
AB_CHAN LONG VARCHAR,
AB_SCHE LONG VARCHAR,
AB_DISC LONG VARCHAR,
LOGDATE DATE,
EDITORVfy NUMERIC(1),
USER_ID VARCHAR(20)
)
;
CREATE INDEX S_EXCU05_USER_ID
ON EXCU05 (USER_ID)
;
CREATE TABLE EXCU06
(
PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PNCH_DESC LONG VARCHAR,
PENG_DESC LONG VARCHAR,
PLAN_YEAR CHAR(3),
EXCU_ORGAN VARCHAR(44),
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
AS_ORGAN_1 VARCHAR(44),
AS_ORGAN_2 VARCHAR(44),
AS_ORGAN_3 VARCHAR(44),
STIC_CS1 CHAR(6),
STIC_CS2 CHAR(6),
STIC_CS3 CHAR(6),
STIC_CS4 CHAR(6),
STIC_CS5 CHAR(6),
STIC_CS6 CHAR(6),
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PAGE CHAR(4),
LANGUAGE VARCHAR(20),
CLASSIFY CHAR(1),
LOGDATE DATE,
USERID CHAR(15),
EDITORVfy NUMERIC(1),
PLAN_AMT INTEGER DEFAULT 0
)
;
CREATE INDEX S_EXCU06_USERID
ON EXCU06 (USERID)
;
CREATE TABLE EXCU07
(
UNION_CODE INTEGER DEFAULT 0,
PROJKEY CHAR(15),
NAME_C CHAR(14),

```

```

NAME_E CHAR(14),
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_EXCU07_USERID
ON EXCU07 (USERID)
;
CREATE TABLE EXCU08
(
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_EXCU08_USERID
ON EXCU08 (USERID)
;
CREATE TABLE EXCUINPUTA
(
PROJKEY CHAR(15),
MN_PLAN_NO VARCHAR(25),
MN_PN_DESC VARCHAR(60),
MAIN_ORGAN VARCHAR(50),
MN_LEADER VARCHAR(20),
MAIN_STYM CHAR(5),
MAIN_ENYM CHAR(5),
MAIN_PNAMT INTEGER,
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_EXCUINPUTA_IDPWD
ON EXCUINPUTA (USERID)
;
CREATE TABLE EXCUINPUTB
(
PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PNCH_DESC VARCHAR(200),
PENG_DESC VARCHAR(200),
PLAN_CODE CHAR(1),
PLAN_DESC VARCHAR(30),
EXCU_ORGAN VARCHAR(44),
SIGN_NAME VARCHAR(20),
PLAN_YEAR CHAR(3),
PLAN_ORGAN VARCHAR(44),
REAT_TYPE CHAR(1),

```

```

REAT_CODE CHAR(1),
REFI_CODE1 CHAR(2),
REFI_CODE2 CHAR(2),
REFI_CODE3 CHAR(2),
REFI_CODE4 CHAR(2),
REFI_CODE5 CHAR(2),
REFI_CODE6 CHAR(2),
PNAT_CODE CHAR(1),
WHOLE_STYM CHAR(5),
WHOLE_ENYM CHAR(5),
WHOLE_AMT INTEGER,
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
PLAN_AMT INTEGER,
EXCU_PERC INTEGER,
AS_ORGAN_1 VARCHAR(44),
AS_PERC_1 INTEGER,
AS_ORGAN_2 VARCHAR(44),
AS_PERC_2 INTEGER,
AS_ORGAN_3 VARCHAR(44),
AS_PERC_3 INTEGER,
LAND_AMT INTEGER,
EQUIP_AMT INTEGER,
CAPT_OTAMT INTEGER,
PERS_AMT INTEGER,
MART_AMT INTEGER,
OTHER_AMT INTEGER,
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PLAN_MONTH INTEGER,
MN_PLAN_NO VARCHAR(25),
PLAN_TYPE CHAR(1),
WORK_MON INTEGER,
SIGN_DATE CHAR(6),
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_EXCUINPUTB_USERI
ON EXCUINPUTB (USERID)
;
CREATE TABLE EXCUINPUTC
(
UNION_CODE INTEGER DEFAULT 0,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PLAN_SEQ CHAR(3),
NAME VARCHAR(20),
IDNO CHAR(10),
BIRTH_YY CHAR(3),
SPFI_CODE CHAR(2),
OPGR_CODE CHAR(1),

```



```

EDUC_CODE CHAR(1),
SEX CHAR(1),
WORK_MON INTEGER,
WIAT_CODE CHAR(1),
LOGDATE DATE,
USERID VARCHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_EXCUINPUTC_IDNO
ON EXCUINPUTC (IDNO)
;
CREATE INDEX S_EXCUINPUTC_USERI
ON EXCUINPUTC (USERID)
;
CREATE TABLE EXCUMIDLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN CHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE EXCUMIDLOG
ADD CONSTRAINT PK_EXCUMIDLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_EXCUMIDLOG_PROJK
ON EXCUMIDLOG (PROJKEY)
;
CREATE INDEX S_EXCUMIDLOG_USER
ON EXCUMIDLOG (USER_ID)
;
CREATE TABLE EXCUPRJLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO CHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),

```

```

PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE EXCUPRJLOG
  ADD CONSTRAINT PK_EXCUPRJLOG
    PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_EXCUPRJLOG_PROJK
ON EXCUPRJLOG (PROJKEY)
;
CREATE INDEX S_EXCUPRJLOG_USER
ON EXCUPRJLOG (USER_ID)
;
CREATE TABLE EXCURPTLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN CHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(44),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE EXCURPTLOG
  ADD CONSTRAINT PK_EXCURPTLOG
    PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_EXCURPTLOG_PROJK
ON EXCURPTLOG (PROJKEY)
;
CREATE INDEX S_EXCURPTLOG_USER
ON EXCURPTLOG (USER_ID)
;
CREATE TABLE GRB_DISP
(
DISP_TITLE VARCHAR(50),
DISP_CODE VARCHAR(50)
)
;
CREATE TABLE GRB_ORG
(
ORG_TITLE VARCHAR(10) NOT NULL PRIMARY KEY,
ORG_CODE VARCHAR(50)
)
;
CREATE TABLE MAST01

```

```

(
PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PNCH_DESC VARCHAR(200),
PENG_DESC VARCHAR(200),
PLAN_CODE CHAR(1),
PLAN_DESC VARCHAR(30),
EXCU_ORGAN VARCHAR(44),
SIGN_NAME VARCHAR(20),
PLAN_YEAR CHAR(3),
PLAN_ORGAN VARCHAR(44),
REAT_TYPE CHAR(1),
REAT_CODE CHAR(1),
REFI_CODE1 CHAR(2),
REFI_CODE2 CHAR(2),
REFI_CODE3 CHAR(2),
REFI_CODE4 CHAR(2),
REFI_CODE5 CHAR(2),
REFI_CODE6 CHAR(2),
PNAT_CODE CHAR(1),
WHOLE_STYM CHAR(5),
WHOLE_ENYM CHAR(5),
WHOLE_AMT DOUBLE,
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
PLAN_AMT DOUBLE,
EXCU_PERC integer,
AS_ORGAN_1 VARCHAR(44),
AS_PERC_1 DOUBLE,
AS_ORGAN_2 VARCHAR(44),
AS_PERC_2 DOUBLE,
AS_ORGAN_3 VARCHAR(44),
AS_PERC_3 DOUBLE,
LAND_AMT DOUBLE,
EQUIP_AMT DOUBLE,
CAPT_OTAMT DOUBLE,
PERS_AMT DOUBLE,
MART_AMT DOUBLE,
OTHER_AMT DOUBLE,
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PLAN_MONTH DOUBLE,
MN_PLAN_NO VARCHAR(25),
PLAN_TYPE CHAR(1),
WORK_MON DOUBLE,
SIGN_DATE CHAR(6),
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_MAST01_USERID
ON MAST01 (USERID)

```

```

;
CREATE TABLE MAST02
(
PROJKEY CHAR(15),
MN_PLAN_NO VARCHAR(25),
MN_PN_DESC VARCHAR(60),
MAIN_ORGAN VARCHAR(50),
MN_LEADER VARCHAR(20),
MAIN_STYM CHAR(5),
MAIN_ENYM CHAR(5),
MAIN_PNAMT DOUBLE,
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVfy NUMERIC(1)
)
;
CREATE INDEX S_MAST02_IDPWD
ON MAST02 (USERID)
;
CREATE TABLE MAST03
(
UNION_CODE INTEGER DEFAULT 0,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PLAN_SEQ CHAR(3),
NAME VARCHAR(20),
IDNO CHAR(10),
BIRTH_YY CHAR(3),
SPFI_CODE CHAR(2),
OPGR_CODE CHAR(1),
EDUC_CODE CHAR(1),
SEX CHAR(1),
WORK_MON DOUBLE,
WIAT_CODE CHAR(1),
LOGDATE DATE,
USERID CHAR(15),
EDITORVfy NUMERIC(1)
)
;
CREATE INDEX S_MAST03_IDNO
ON MAST03 (IDNO)
;
CREATE INDEX S_MAST03_USERID
ON MAST03 (USERID)
;
CREATE TABLE MAST04
(
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
PRJTYPE CHAR(10),
LOGDATE DATE,
USERID VARCHAR(20)
)

```

```

)
;
CREATE INDEX S_MAST04_USERID
ON MAST04 (USERID)
;
CREATE TABLE MAST05
(
PROJKEY CHAR(15),
PNCH_DESC VARCHAR(250),
DELI_NO VARCHAR(50),
PLAN_NO VARCHAR(50),
PLAN_ORGAN VARCHAR(50),
EXCU_ORGAN VARCHAR(50),
NAME VARCHAR(50),
CON_NAME VARCHAR(50),
TEL VARCHAR(50),
FAX VARCHAR(50),
PERIO_STYM VARCHAR(50),
PERIO_ENYM VARCHAR(50),
WHOLE_AMT INTEGER DEFAULT 0,
PLAN_AMT INTEGER DEFAULT 0,
YRATE_1 INTEGER DEFAULT 0,
YRATE_2 INTEGER DEFAULT 0,
YRATE_3 INTEGER DEFAULT 0,
TRATE_1 INTEGER DEFAULT 0,
TRATE_2 INTEGER DEFAULT 0,
TRATE_3 INTEGER DEFAULT 0,
YPAYOFF_1 INTEGER DEFAULT 0,
YPAYOFF_2 INTEGER DEFAULT 0,
YPAYOFF_3 INTEGER DEFAULT 0,
TPAYOFF_1 INTEGER DEFAULT 0,
TPAYOFF_2 INTEGER DEFAULT 0,
TPAYOFF_3 INTEGER DEFAULT 0,
AB_GOAL LONG VARCHAR,
AB_SOUR LONG VARCHAR,
AB_RESU LONG VARCHAR,
AB_CHAN LONG VARCHAR,
AB_SCHE LONG VARCHAR,
AB_DISC LONG VARCHAR,
LOGDATE DATE,
EDITORVfy NUMERIC(1),
USER_ID VARCHAR(20)
)
;
CREATE INDEX S_MAST05_USER_ID
ON MAST05 (USER_ID)
;
CREATE TABLE MAST06
(
PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PNCH_DESC LONG VARCHAR,
PENG_DESC LONG VARCHAR,

```

```

PLAN_YEAR CHAR(3),
EXCU_ORGAN VARCHAR(44),
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
AS_ORGAN_1 VARCHAR(50),
AS_ORGAN_2 VARCHAR(50),
AS_ORGAN_3 VARCHAR(50),
STIC_CS1 CHAR(6),
STIC_CS2 CHAR(6),
STIC_CS3 CHAR(6),
STIC_CS4 CHAR(6),
STIC_CS5 CHAR(6),
STIC_CS6 CHAR(6),
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PAGE CHAR(4),
LANG VARCHAR(20),
CLASSIFY CHAR(1),
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1),
PLAN_AMT INTEGER DEFAULT 0
)
;
CREATE INDEX S_MAST06_USERID
ON MAST06 (USERID)
;
CREATE TABLE MAST07
(
UNION_CODE INTEGER DEFAULT 0,
PROJKEY CHAR(15),
NAME_C VARCHAR(20),
NAME_E VARCHAR(20),
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_MAST07_USERID
ON MAST07 (USERID)
;
CREATE TABLE MAST08
(
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_MAST08_USERID
ON MAST08 (USERID)

```

```

;
CREATE TABLE MASTMIDLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE MASTMIDLOG
ADD CONSTRAINT PK_MASTMIDLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_MASTMIDLOG_PROJK
ON MASTMIDLOG (PROJKEY)
;
CREATE INDEX S_MASTMIDLOG_USER
ON MASTMIDLOG (USER_ID)
;
CREATE TABLE MASTPRJLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE MASTPRJLOG
ADD CONSTRAINT PK_MASTPRJLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_MASTPRJLOG_PROJK
ON MASTPRJLOG (PROJKEY)
;
CREATE INDEX S_MASTPRJLOG_USER
ON MASTPRJLOG (USER_ID)
;

```

```

CREATE TABLE MASTRPTLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY CHAR(15),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN CHAR(50),
LOGDATE DATE,
DW_STATUS INTEGER DEFAULT 0,
PNCH_DESC VARCHAR(150),
PRJTYPE CHAR(10),
LASTDATE CHAR(12)
)
;
ALTER TABLE MASTRPTLOG
ADD CONSTRAINT PK_MASTRPTLOG
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_MASTRPTLOG_PROJK
ON MASTRPTLOG (PROJKEY)
;
CREATE INDEX S_MASTRPTLOG_USER
ON MASTRPTLOG (USER_ID)
;
CREATE TABLE NSC01
(
PROJKEY CHAR(15) NOT NULL primary key,
PLAN_NO VARCHAR(50),
PNCH_DESC VARCHAR(200),
PENG_DESC VARCHAR(250),
PLAN_CODE CHAR(1),
PLAN_DESC VARCHAR(50),
EXCU_CODE VARCHAR(20),
EXCU_ORGAN VARCHAR(44),
SIGN_NAME VARCHAR(20),
PLAN_YEAR CHAR(3),
PLAN_ORGAN VARCHAR(44),
REAT_TYPE CHAR(1),
REAT_CODE CHAR(1),
REFI_CODE1 CHAR(2),
REFI_CODE2 CHAR(2),
REFI_CODE3 CHAR(2),
REFI_CODE4 CHAR(2),
REFI_CODE5 CHAR(2),
REFI_CODE6 CHAR(2),
PNAT_CODE CHAR(1),
WHOLE_STYM CHAR(5),
WHOLE_ENYM CHAR(5),
WHOLE_AMT DOUBLE,
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),

```



```

PLAN_AMT DOUBLE,
EXCU_PERC integer,
AS_ORGAN_1 VARCHAR(44),
AS_PERC_1 DOUBLE,
AS_ORGAN_2 VARCHAR(44),
AS_PERC_2 DOUBLE,
AS_ORGAN_3 VARCHAR(44),
AS_PERC_3 DOUBLE,
LAND_AMT DOUBLE,
EQUIP_AMT DOUBLE,
CAPT_OTAMT DOUBLE,
PERS_AMT DOUBLE,
MART_AMT DOUBLE,
OTHER_AMT DOUBLE,
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PLAN_MONTH DOUBLE,
MN_PLAN_NO VARCHAR(30),
PLAN_TYPE CHAR(1),
WORK_MON DOUBLE,
SIGN_DATE CHAR(6),
UPLOAD_DATE CHAR(6),
JUDG_NO VARCHAR(50)
)
;
ALTER TABLE NSC01
  ADD CONSTRAINT U_NSC01_PROJKEY
    UNIQUE (PROJKEY)
;
CREATE TABLE NSC01_FRESH
(
PROJKEY CHAR(15) NOT NULL PRIMARY KEY,
PLAN_NO VARCHAR(50),
PNCH_DESC VARCHAR(200),
PENG_DESC VARCHAR(250),
PLAN_CODE CHAR(1),
PLAN_DESC VARCHAR(50),
EXCU_ORGAN VARCHAR(44),
SIGN_NAME VARCHAR(20),
PLAN_YEAR CHAR(3),
PLAN_ORGAN VARCHAR(44),
REAT_TYPE CHAR(1),
REAT_CODE CHAR(1),
REFI_CODE1 CHAR(2),
REFI_CODE2 CHAR(2),
REFI_CODE3 CHAR(2),
REFI_CODE4 CHAR(2),
REFI_CODE5 CHAR(2),
REFI_CODE6 CHAR(2),
PNAT_CODE CHAR(1),
WHOLE_STYM CHAR(5),
WHOLE_ENYM CHAR(5),
WHOLE_AMT DOUBLE,

```

```

PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
PLAN_AMT DOUBLE,
EXCU_PERC integer,
AS_ORGAN_1 VARCHAR(44),
AS_PERC_1 DOUBLE,
AS_ORGAN_2 VARCHAR(44),
AS_PERC_2 DOUBLE,
AS_ORGAN_3 VARCHAR(44),
AS_PERC_3 DOUBLE,
LAND_AMT DOUBLE,
EQUIP_AMT DOUBLE,
CAPT_OTAMT DOUBLE,
PERS_AMT DOUBLE,
MART_AMT DOUBLE,
OTHER_AMT DOUBLE,
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PLAN_MONTH DOUBLE,
MN_PLAN_NO VARCHAR(30),
PLAN_TYPE CHAR(1),
WORK_MON DOUBLE,
SIGN_DATE CHAR(6),
UPLOAD_DATE CHAR(6)
)
;
CREATE TABLE NSC02_FRESH
(
PROJKEY CHAR(15),
MN_PLAN_NO VARCHAR(30),
MN_PN_DESC VARCHAR(200),
MAIN_ORGAN VARCHAR(50),
MN_LEADER VARCHAR(20),
MAIN_STYM CHAR(5),
MAIN_ENYM CHAR(5),
MAIN_PNAMT DOUBLE,
UPLOAD_DATE DATE
)
;
CREATE TABLE NSC03
(
UNION_CODE INTEGER NOT NULL PRIMARY KEY,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PLAN_SEQ CHAR(3),
NAME VARCHAR(20),
IDNO CHAR(10),
BIRTH_YY CHAR(3),
SPFI_CODE CHAR(2),
OPGR_CODE CHAR(1),
EDUC_CODE CHAR(1),
SEX CHAR(1),
WORK_MON DOUBLE,

```

```

WIAT_CODE CHAR(1),
UPLOAD_DATE DATE,
PLAN_ORGAN CHAR(4)
)
;
ALTER TABLE nsc03
add constraint FK_NSC03
foreign key ( projkey) references nsc01
;
CREATE INDEX S_NSC03_PROJKEY
ON NSC03 (PROJKEY)
;
CREATE TABLE NSC03_FRESH
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PLAN_SEQ CHAR(3),
NAME VARCHAR(20),
IDNO CHAR(10),
BIRTH_YY CHAR(3),
SPFI_CODE CHAR(2),
OPGR_CODE CHAR(1),
EDUC_CODE CHAR(1),
SEX CHAR(1),
WORK_MON DOUBLE,
WIAT_CODE CHAR(1),
UPLOAD_DATE DATE,
PLAN_ORGAN CHAR(4)
)
;
ALTER TABLE NSC03_FRESH
ADD CONSTRAINT PK_NSC03_FRESH
PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_NSC03_FRESH_IDNO
ON NSC03_FRESH (IDNO)
;
CREATE TABLE NSC04
(
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
PRJTYPE CHAR(10),
LOGDATE DATE,
USERID CHAR(20)
)
;
CREATE INDEX S_NSC04_USERID
ON NSC04 (USERID)
;
CREATE TABLE NSC04_FRESH
(

```

```

PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
PRJTYPE CHAR(10),
LOGDATE DATE,
USERID VARCHAR(20)
)
;
CREATE INDEX S_NSC04_FRESH_USER
ON NSC04_FRESH (USERID)
;
CREATE TABLE NSC05
(
PROJKEY CHAR(15),
PNCH_DESC VARCHAR(250),
DELI_NO VARCHAR(50),
PLAN_NO VARCHAR(50),
PLAN_ORGAN VARCHAR(50),
EXCU_ORGAN VARCHAR(50),
NAME VARCHAR(50),
CON_NAME VARCHAR(50),
TEL VARCHAR(50),
FAX VARCHAR(50),
PERIO_STYM VARCHAR(50),
PERIO_ENYM VARCHAR(50),
WHOLE_AMT VARCHAR(50),
PLAN_AMT VARCHAR(50),
YRATE_1 VARCHAR(50),
YRATE_2 VARCHAR(50),
YRATE_3 VARCHAR(50),
TRATE_1 VARCHAR(50),
TRATE_2 VARCHAR(50),
TRATE_3 VARCHAR(50),
YPAYOFF_1 VARCHAR(50),
YPAYOFF_2 VARCHAR(50),
YPAYOFF_3 VARCHAR(50),
TPAYOFF_1 VARCHAR(50),
TPAYOFF_2 VARCHAR(50),
TPAYOFF_3 VARCHAR(50),
AB_GOAL LONG VARCHAR,
AB_SOUR LONG VARCHAR,
AB_RESU LONG VARCHAR,
AB_CHAN LONG VARCHAR,
AB_SCHE LONG VARCHAR,
AB_DISC LONG VARCHAR,
UP_DATE CHAR(10)
)
;
CREATE TABLE NSC02
(
PROJKEY CHAR(15),
MN_PLAN_NO VARCHAR(30),
MN_PN_DESC VARCHAR(200),

```

```

MAIN_ORGAN VARCHAR(50),
MN_LEADER VARCHAR(20),
MAIN_STYM CHAR(5),
MAIN_ENYM CHAR(5),
MAIN_PNAMT DOUBLE,
UPLOAD_DATE DATE
)
;
CREATE TABLE NSC05_FRESH
(
PROJKEY CHAR(15),
PNCH_DESC VARCHAR(250),
DELI_NO VARCHAR(50),
PLAN_NO VARCHAR(50),
PLAN_ORGAN VARCHAR(50),
EXCU_ORGAN VARCHAR(50),
NAME VARCHAR(50),
CON_NAME VARCHAR(50),
TEL VARCHAR(50),
FAX VARCHAR(50),
PERIO_STYM VARCHAR(50),
PERIO_ENYM VARCHAR(50),
WHOLE_AMT INTEGER DEFAULT 0,
PLAN_AMT INTEGER DEFAULT 0,
YRATE_1 INTEGER DEFAULT 0,
YRATE_2 INTEGER DEFAULT 0,
YRATE_3 INTEGER DEFAULT 0,
TRATE_1 INTEGER DEFAULT 0,
TRATE_2 INTEGER DEFAULT 0,
TRATE_3 INTEGER DEFAULT 0,
YPAYOFF_1 INTEGER DEFAULT 0,
YPAYOFF_2 INTEGER DEFAULT 0,
YPAYOFF_3 INTEGER DEFAULT 0,
TPAYOFF_1 INTEGER DEFAULT 0,
TPAYOFF_2 INTEGER DEFAULT 0,
TPAYOFF_3 INTEGER DEFAULT 0,
AB_GOAL LONG VARCHAR,
AB_SOUR LONG VARCHAR,
AB_RESU LONG VARCHAR,
AB_CHAN LONG VARCHAR,
AB_SCHE LONG VARCHAR,
AB_DISC LONG VARCHAR,
LOGDATE DATE,
EDITORVFY NUMERIC(1),
USER_ID VARCHAR(20)
)
;
CREATE INDEX S_NSC05_FRESH_USER
ON NSC05_FRESH (USER_ID)
;
CREATE TABLE NSC06
(
RPTKEY CHAR(15) NOT NULL,

```

```

PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PNCH_DESC LONG VARCHAR,
PENG_DESC LONG VARCHAR,
PLAN_YEAR CHAR(3),
EXCU_ORGAN VARCHAR(44),
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
AS_ORGAN_1 VARCHAR(44),
AS_ORGAN_2 VARCHAR(44),
AS_ORGAN_3 VARCHAR(44),
STIC_CS1 CHAR(6),
STIC_CS2 CHAR(6),
STIC_CS3 CHAR(6),
STIC_CS4 CHAR(6),
STIC_CS5 CHAR(6),
STIC_CS6 CHAR(6),
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PUBLISH_DA CHAR(10),
PAGE CHAR(4),
LANG CHAR(10),
CLASSIFY CHAR(1),
UPLOAD_DAT CHAR(10)
)
;
CREATE TABLE NSC06_FRESH
(
RPTKEY CHAR(15),
PROJKEY CHAR(15),
PLAN_NO VARCHAR(30),
PNCH_DESC LONG VARCHAR,
PENG_DESC LONG VARCHAR,
PLAN_YEAR CHAR(3),
EXCU_ORGAN VARCHAR(44),
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
AS_ORGAN_1 VARCHAR(44),
AS_ORGAN_2 VARCHAR(44),
AS_ORGAN_3 VARCHAR(44),
STIC_CS1 CHAR(6),
STIC_CS2 CHAR(6),
STIC_CS3 CHAR(6),
STIC_CS4 CHAR(6),
STIC_CS5 CHAR(6),
STIC_CS6 CHAR(6),
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PAGE CHAR(4),
LANGUAGE CHAR(20),
CLASSIFY CHAR(1),
LOGDATE DATE,
USERID CHAR(15),

```

```

EDITORVFY NUMERIC(1),
PLAN_AMT INTEGER DEFAULT 0
)
;
CREATE INDEX S_NSC06_FRESH_USER
ON NSC06_FRESH (USERID)
;
CREATE TABLE NSC07
(
RPTKEY CHAR(15),
PROJKEY CHAR(15),
NAME_C VARCHAR(20),
NAME_E VARCHAR(20)
)
;
CREATE TABLE NSC07_FRESH
(
UNION_CODE INTEGER DEFAULT 0,
RPTKEY CHAR(15),
PROJKEY CHAR(15),
NAME_C VARCHAR(20),
NAME_E VARCHAR(20),
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_NSC07_FRESH_USER
ON NSC07_FRESH (USERID)
;
CREATE TABLE NSC08
(
RPTKEY CHAR(15) NOT NULL,
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR
)
;
CREATE TABLE NSC08_FRESH
(
RPTKEY CHAR(15),
PROJKEY CHAR(15),
ABSTRACT_C LONG VARCHAR,
ABSTRACT_E LONG VARCHAR,
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_NSC08_FRESH_USER
ON NSC08_FRESH (USERID)
;
CREATE TABLE NSC09

```

```

(
UNIONCODE INTEGER DEFAULT 0,
PLAN_NO VARCHAR(50),
NAME VARCHAR(20),
EXCU_ORGAN VARCHAR(200),
PNCH_DESC VARCHAR(250),
DATE DATE,
MICRO1 INTEGER,
MICRO2 INTEGER
)
;
CREATE TABLE NSC10
(
UNIONCODE INTEGER DEFAULT 0 NOT NULL,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(50),
YEAR CHAR(3),
SOURCE LONG VARCHAR
)
;
ALTER TABLE NSC10
ADD CONSTRAINT PK_NSC10
PRIMARY KEY (UNIONCODE)
;
CREATE TABLE PAGECOUNT
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
LG_USRKIND INTEGER DEFAULT 0,
LG_ADDR VARCHAR(150),
LG_DATE DATE,
LG_SERVER VARCHAR(50)
)
;
ALTER TABLE PAGECOUNT
ADD CONSTRAINT PK_PAGECOUNT
PRIMARY KEY (UNION_CODE)
;
CREATE TABLE PAGECOUNT2
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
LG_USRKIND INTEGER DEFAULT 0,
LG_ADDR VARCHAR(150),
LG_DATE DATE,
LG_SERVER VARCHAR(50)
)
;
ALTER TABLE PAGECOUNT2
ADD CONSTRAINT PK_PAGECOUNT2
PRIMARY KEY (UNION_CODE)
;
CREATE TABLE REGLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,

```



```

LG_COUNT INTEGER DEFAULT 0,
LG_ADDR VARCHAR(150),
LG_DATE DATE
)
;
ALTER TABLE REGLOG
  ADD CONSTRAINT PK_REGLOG
    PRIMARY KEY (UNION_CODE)
;
CREATE TABLE REJCTUSER
(
UNION_ID INTEGER DEFAULT 0 NOT NULL,
USR_ID VARCHAR(20),
USR_NAME VARCHAR(20),
USR_PW VARCHAR(20),
USR_KIND INTEGER DEFAULT 0,
USR_DATE DATE,
USR_STICCODE CHAR(15),
USR_FIELD VARCHAR(50),
USR_EMAIL VARCHAR(40),
USR_ADDR VARCHAR(50),
USR_TELE1 VARCHAR(30),
USR_TELE2 VARCHAR(30),
USR_FAX VARCHAR(50),
USR_EDU VARCHAR(20)
)
;
ALTER TABLE REJCTUSER
  ADD CONSTRAINT PK_REJCTUSER
    PRIMARY KEY (UNION_ID)
;
CREATE INDEX S_REJCTUSER_USERID
ON REJCTUSER (USR_ID)
;
CREATE TABLE STICNOINIT
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
PRJ_HEADING VARCHAR(100),
INIT_NO INTEGER DEFAULT 0,
PLAN_NO VARCHAR(50)
)
;
ALTER TABLE STICNOINIT
  ADD CONSTRAINT PK_STICNOINIT
    PRIMARY KEY (UNION_CODE)
;
CREATE TABLE TBLNAME
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
TABLE_NAME VARCHAR(50),
P_KEY VARCHAR(50),
WRITE_TO VARCHAR(50)
)

```

```

;
ALTER TABLE TBLNAME
  ADD CONSTRAINT PK_TBLNAME
    PRIMARY KEY (UNION_CODE)
;
CREATE TABLE UPLOADLOG
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
USER_ID VARCHAR(20),
PROJKEY VARCHAR(20),
PLAN_ORGAN VARCHAR(50),
PLAN_YEAR CHAR(6),
PLAN_NO VARCHAR(20),
EXCU_ORGAN VARCHAR(50),
LOGDATE DATE,
UL_STATUS CHAR(10)
)
;
ALTER TABLE UPLOADLOG
  ADD CONSTRAINT PK_UPLOADLOG
    PRIMARY KEY (UNION_CODE)
;
CREATE INDEX S_UPLOADLOG_USER_I
ON UPLOADLOG (USER_ID)
;
CREATE TABLE USER_STATUS
(
USR_STATUS INTEGER DEFAULT 0 NOT NULL,
USR_STATUS_DESC CHAR(10)
)
;
ALTER TABLE USER_STATUS
  ADD CONSTRAINT PK_USER_STATUS
    PRIMARY KEY (USR_STATUS)
;
CREATE TABLE USERDATA
(
UNION_ID INTEGER DEFAULT 0 NOT NULL,
USR_ID VARCHAR(20),
USR_NAME VARCHAR(20),
USR_PW VARCHAR(20),
USR_KIND INTEGER DEFAULT 0,
USR_DATE DATE,
USR_STICCODE CHAR(15),
USR_FIELD VARCHAR(50),
USR_EMAIL VARCHAR(40),
USR_ADDR VARCHAR(50),
USR_TELE1 VARCHAR(30),
USR_TELE2 VARCHAR(30),
USR_FAX VARCHAR(50),
USR_EDU VARCHAR(20),
USR_STATUS INTEGER DEFAULT 0,
USR_FIELD_APPLY VARCHAR(100),

```

```

USR_MASTER VARCHAR(50),
USR_ROWS INTEGER DEFAULT 0,
PRJ_DESC LONG VARCHAR
)
;
ALTER TABLE USERDATA
ADD CONSTRAINT PK_USERDATA
PRIMARY KEY (UNION_ID)
;
CREATE INDEX S_USERDATA_USERID
ON USERDATA (USR_ID)
;
CREATE TABLE USERINPUT
(
UNION_CODE INTEGER DEFAULT 0 NOT NULL,
PROJKEY CHAR(15),
PLANNO VARCHAR(50),
PLANNAME VARCHAR(255),
LOGUSERNAME CHAR(15),
LOGUSREMAIL VARCHAR(80),
LOGUSRTEL VARCHAR(50),
PROJMASTER VARCHAR(20),
MASTEREMAIL VARCHAR(80),
MASTERTEL VARCHAR(20),
LOGDATE DATE,
EDITORVFY NUMERIC(1),
ABSTRACT_C LONG VARCHAR
)
;
ALTER TABLE USERINPUT
ADD CONSTRAINT PK_USERINPUT
PRIMARY KEY (UNION_CODE)
;
CREATE TABLE USERINPUTA
(
PROJKEY CHAR(15),
MN_PLAN_NO VARCHAR(25),
MN_PN_DESC VARCHAR(60),
MAIN_ORGAN VARCHAR(50),
MN_LEADER VARCHAR(20),
MAIN_STYM CHAR(5),
MAIN_ENYM CHAR(5),
MAIN_PNAMT DOUBLE,
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_USERINPUTA_IDPWD
ON USERINPUTA (USERID)
;
CREATE TABLE USERINPUTB
(

```

```

PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PNCH_DESC VARCHAR(200),
PENG_DESC VARCHAR(200),
PLAN_CODE CHAR(1),
PLAN_DESC VARCHAR(30),
EXCU_ORGAN VARCHAR(44),
SIGN_NAME VARCHAR(20),
PLAN_YEAR CHAR(3),
PLAN_ORGAN CHAR(4),
REAT_TYPE CHAR(1),
REAT_CODE CHAR(1),
REFI_CODE1 CHAR(2),
REFI_CODE2 CHAR(2),
REFI_CODE3 CHAR(2),
REFI_CODE4 CHAR(2),
REFI_CODE5 CHAR(2),
REFI_CODE6 CHAR(2),
PNAT_CODE CHAR(1),
WHOLE_STYM CHAR(5),
WHOLE_ENYM CHAR(5),
WHOLE_AMT DOUBLE,
PERIO_STYM CHAR(5),
PERIO_ENYM CHAR(5),
PLAN_AMT DOUBLE,
EXCU_PERC integer,
AS_ORGAN_1 VARCHAR(44),
AS_PERC_1 DOUBLE,
AS_ORGAN_2 VARCHAR(44),
AS_PERC_2 DOUBLE,
AS_ORGAN_3 VARCHAR(44),
AS_PERC_3 DOUBLE,
LAND_AMT DOUBLE,
EQUIP_AMT DOUBLE,
CAPT_OTAMT DOUBLE,
PERS_AMT DOUBLE,
MART_AMT DOUBLE,
OTHER_AMT DOUBLE,
KEYWORD_C LONG VARCHAR,
KEYWORD_E LONG VARCHAR,
PLAN_MONTH DOUBLE,
MN_PLAN_NO VARCHAR(25),
PLAN_TYPE CHAR(1),
WORK_MON DOUBLE,
SIGN_DATE CHAR(6),
LOGDATE DATE,
USERID VARCHAR(20),
EDITORVfy NUMERIC(1)
)
;
CREATE INDEX S_USERINPUTB_USERI
ON USERINPUTB (USERID)
;

```

```

CREATE TABLE USERINPUTC
(
UNION_CODE INTEGER DEFAULT 0,
PROJKEY CHAR(15),
PLAN_NO VARCHAR(25),
PLAN_SEQ CHAR(3),
NAME CHAR(14),
IDNO CHAR(10),
BIRTH_YY CHAR(3),
SPFI_CODE CHAR(2),
OPGR_CODE CHAR(1),
EDUC_CODE CHAR(1),
SEX CHAR(1),
WORK_MON DOUBLE,
WIAT_CODE CHAR(1),
LOGDATE DATE,
USERID CHAR(15),
EDITORVFY NUMERIC(1)
)
;
CREATE INDEX S_USERINPUTC_IDNO
ON USERINPUTC (IDNO)
;
CREATE INDEX S_USERINPUTC_USERI
ON USERINPUTC (USERID)
;

```

由於整個程式的開發完全是與使用者邊討論邊進行寫碼，因此開發未留下完整的程式流程圖。以下為使用者在填入帳號及密碼之後的驗證程式片斷：

```

<CFQUERY NAME="Checking" DATASOURCE="GRBDB">
    SELECT * FROM UserData
        WHERE UserData.usr_id = '#form.T1#' and UserData.usr_pw = '#form.T2#'
</CFQUERY>

```

```

<Html>
<head>
<meta charset="big5">
<META HTTP-EQUIV="Pragma" CONTENT="no-cache">
</head>
<body bgcolor="#FFFFCC">

```

```

<CFSET RQ_type = '0' >
<CFOUTPUT QUERY="Checking">

    <CFSET session.AUsrID=#usr_id# >
    <cfset session.UsrFld=#usr_field#>
    <cfset session.UsrKnd=#usr_kind# >
    <cfset RQ_status = #usr_status# >

    <CFIF #usr_kind# eq 1 >
        <CFSET RQ_type = '1' >
    <CFELSEIF #usr_kind# eq 2 >

```

```
<CFSET RQ_type = '2' >
<cfelseif #usr_kind# eq 11 >
  <cfset RQ_type = '11' >
<cfelseif #usr_kind# eq 12 >
  <cfset RQ_type = '12' >
<CFELSE>
  <CFSET RQ_type= '3' >
</CFIF>
</CFOUTPUT>

<CFIF RQ_type eq '0' >
  <H3> ID 及密碼錯誤, 請重新輸入!! </h3> <br>
  <a href = "index.htm" > 返回 </a>

<CFELSEIF RQ_type eq '1'>
  <cfif #RQ_status# is 1><!-- 主管機關 --->
    <cflocation url="/.master/grb_special_2.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<CFELSEIF RQ_type eq '2'>
  <cfif #RQ_status# is 1 ><!-- 系統管理者 --->
    <cflocation url="/.manage/manage.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<cfelseif RQ_type eq '11' > <!-- 執行機構專用 --->
  <cfif #RQ_status# is 1 >
    <cflocation url="/.ExcuOrgan/excuOrg.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<cfelseif RQ_type eq '12' > <!-- 業務承辦人 --->
  <cfif #RQ_status# is 1 >
    <cflocation url="/.undertaker/grb_special_2.cfm" addtoken="Yes">
  <cfelse>
    <H3> 您的帳號及密碼已被停用, 請與管理人員連繫, 謝謝</h3>
    <a href = "index.htm" > 返回 </a>
  </cfif>

<CFELSE>
  <cflocation url="index.html" addtoken="Yes">
</CFIF>
</body>
</html>
```

【附錄十三】

電話訪談結果

本研究原預定針對計畫管考者、計畫執行者，以及資訊查詢者進行問卷調查（問卷內容如附），後因人力及時間問題，無法完成，誠屬遺憾。最後僅能以最小規模，選擇系統使用者 table 中所有計畫管考人共 77 位，針對問卷中有關各機關研究計畫建檔與服務情形進行電話訪談。訪談結果有效樣本 71 位，結果如下表：

題號	問卷題目	有	沒有
1	貴單位有沒有專門針對研究計畫的管理辦法或相關作業規範？	58 ^{註1}	13
2	貴單位迄今有沒有將研究計畫及研究報告自行建檔管理(紙本或電子檔)？	47 ^{註2}	24
3	貴單位迄今有沒有將建檔結果對外提供服務，以何種方式提供服務？	44 ^{註3}	27
4	GRB系統對貴單位的計畫管考作業有沒有幫助？	49 ^{註4}	22 ^{註4}

註 1：問卷題號一回答“有”者，如經濟部工業局、國營事業委員會、水資源局、國際貿易局等，皆係依循其上級機關所訂定之「經濟部暨所屬機關(構)行政及政策類委託研究計畫作業規定」，故雖無訂定自己的管理辦法，亦列入“有”。

註 2：部分主管機關係將歷年委託研究計畫摘要內容及成果報告建置於機關網頁，供內部管考作業使用。

註 3：續註 2，同時亦提供外界查詢計畫摘要及下載成果報告或陳列於圖書室提供服務者。

註 4：大部分主管機關計畫管考人認為 GRB 系統有助於該機關之管考作業，部分機關因為 GRB 系統並不符合他們的管考作業流程，而認為沒有幫助；有些計畫管考人因為較熟悉紙本作業，不慣使用網路系統，而認為工作負擔增加。