

國立交通大學建築研究所

碩 士 論 文

虛擬發表空間：虛擬環境中數位媒材與設計溝通探討

**Virtual Presentation Space : An Approach to Digital Media for
Design Communication in Virtual Environment**

研 究 生：施文禮

指導教授：劉育東

中 華 民 國 九 十 四 年 七 月

虛擬發表空間：虛擬環境中數位媒材與設計溝通探討

**Virtual Presentation Space : An Approach to Digital Media for
Design Communication in Virtual Environment**

研 究 生：施文禮

Student : Wen-Li Shih

指 導 教 授：劉育東

Advisor : Yu-Tung Liu

國 立 交 通 大 學
建 築 研 究 所
碩 士 論 文

A Thesis

Submitted to Graduate Institute of Architecture

College of Humanities and Social Sciences

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Master

in

Architecture

July 2005

Hsinchu, Taiwan, Republic of China

中華民國九十四年七月

中文摘要

近來越來越多設計者習於利用電腦等數位媒材作為他們主要的設計媒材，設計者透過數位媒材的使用，能夠呈現出傳統媒材無法做到的效果，例如：即時的光影、沉浸式的空間體驗以及動態的內容展示等。然而，數位媒材的使用也限制了其他使用者與媒材互動的能力。在傳統的設計溝通過程中，所有的參與者都能夠直接地與設計者所使用的媒材互動，他們可以拿起模型觀看，或是在版面上註記。受到了電腦輸入介面的限制，在現在多數的設計發表場合中，只有發表者或是電腦的操作者能夠直接地操作數位媒材，而造成溝通上的問題。本研究的主要目的在於探討設計發表的過程中所使用的數位媒材以及與其互動的形式，並開發實作一套軟體系統 — 一個讓所有使用者都能夠與直接地與虛擬空間中的數位媒材互動同時也能夠彼此相互溝通的虛擬環境。

關鍵字：數位媒材、CAVE、多人互動、設計發表

Abstract

More and more designers nowadays are familiar with employing digital media like computers as their major design media. With digital media, design could be represented in many ways that exceeds traditional media's capability, like realtime lighting/shadow, immersive spatial navigation and dynamic content. However, adoption of digital media blocks other participants from directly interacting with media. In traditional design presentation process, each participant could interact with the design media directly, look on the model with hand-rotating it and mark on the pin-ups. Due to the input interface of computer, only the presenter or the operator of the computer is capable of manipulating the digital media directly in most design presentation occasions at present and the limitation of interaction forms some problems in communication. This research focused on digital media used in design presentation process and schemes of interacting with them, and implemented a software system - a virtual environment that all users could interact with various digital media directly and communicate with each other concurrently in it.

Keywords : digital media, CAVE, multiple interaction, design presentation

誌謝

感謝 Aleppo 老師在論文寫作的過程中給予的鞭策與教誨，讓我更明白自己的不足與缺點，也找到了激勵自己持續前進的方向。

蔡文祥教授在大學時期教導我的積極研究態度，以及帶領我去體會並享受研究過程中的樂趣與痛苦。

侯君昊老師及唐玄輝老師在口試時給予的指正與寶貴建議。

謝謝所上的學長姐，大師兄、小黃、勝誠學長、千惠、楚卿、表妹、月華、淳鈺、彥良，讓我從入所之初就適應良好過的非常快樂。

同時謝謝一起打拼過來的數位組同學們，Bob、凱鳴、Max、琇貞、To 哥、Daniel、洋基、發哥、apple、@hno、hugo、唐董，祝福大家。也謝謝郁鈞、慶倩及婉寧在研究上的協助。

還有一直為我擔心的爸媽，感謝你們面對任性的兒子不時帶給你們的驚嚇卻始終全力支持。

目 錄

第一章 導論	1
1.1 研究背景.....	1
1.2 研究問題與目標.....	2
1.3 研究步驟與方法.....	5
第二章 文獻回顧	7
2.1 人機互動.....	7
2.2 多人互動.....	9
2.3 人與 CAVE 的互動.....	16
第三章 系統分析與規劃	20
3.1 系統分析.....	20
3.2 系統規劃.....	24
3.3 VPS 系統架構.....	28
第四章 系統實作	30
4.1 裝置管理.....	31
4.2 訊號處理.....	33
4.3 情境控制.....	35
4.4 資源管理.....	36
4.5 資料運算.....	43
4.6 顯示輸出.....	44
4.7 系統整合結果.....	44
第五章 系統評估	46
5.1 傳統設計發表過程.....	46
5.2 運用 VPS 進行的設計溝通過程.....	46
5.3 比較與討論.....	52
第六章 結論	54
6.1 研究貢獻.....	54
6.2 研究限制與後續研究.....	54
參考文獻	56

第一章 導論

1.1 研究背景

在電腦進入人類的歷史以後，以相當驚人的速度佔據了我們的桌面，並取代了許多傳統的設計媒材，由於資訊技術的進步，電腦已經成為設計者進行設計時相當重要甚至是主要的媒材。而設計過程中使用的媒材在進行設計溝通時會以設計發表(design presentation)的方式呈現以與其他設計者溝通，因此隨著電腦在設計過程中扮演越來越重要的角色，以往在實體空間中的協同設計行為中許多設計者互相討論以獲取靈感或是修正設計的方式，也漸漸依賴電腦以進行圖面或是文字的表達，也就是電腦所扮演的角色已經從個人化的設計工具漸漸轉變成多人的互動討論平台[Achten, 2002]。

然而，在今日較常見的場景則是許多人盯著一部投影機投影電腦裡的畫面，操作電腦的人擁有大多數的操控權，其他參與的人則是透過雷射筆的指指點點企圖操控該電腦以說明或展示他個人的想法，但是操控者往往不能完全地依照其他人的想法操控電腦，人們常常浪費很多時間在操控者換位子、換電腦、以及各種在操控溝通上的失誤。

從許多關於未來人與電腦的互動關係的討論，如：Dertouzos[2001]、Achten[2002]等人的觀點，都強調了在未来的數位環境中提供協同工作機制的重要性。要能達到多人協同工作的目的，就要讓所有的參與者都能夠與共同的主題互動，任何一位使用者都能夠直接將想法透過在數位環境中操作虛擬物件的方式傳達給其他所有的參與者，以分享共同的討論主題，進而能夠相互地溝通。換言之，多人互動是達成協同工作的基本條件。

因應這個重要議題，許多的研究開始發展能提供多人互動機制的數位環境，其中的一個方式就是透過網路的多人互動形式[Cera, 2002; Endo, 2003]。然而透過網路的方式要進行溝通，不論是聊天室、論壇或是視訊，都讓人與人之間失去了用最自然而直覺的方式相互溝通的行為。因此，有相當多關於應用數位媒材與人機互動等研究在探索如何讓多人能夠在同一個實體共享空間(physical shared space)中最直覺的方式溝通，而又能夠同時地操作數位媒材進行設計溝通或討論。

在這樣的發展下，單一顯示群組軟體(single display groupware, SDG)與多型態輸入的概念應運而生，提出了能夠在共同的虛擬環境裡提供多人同時互動的同時又能相互直覺地溝通的具體概念。透過 SDG 的運作可以讓複數使用者同時用多隻滑鼠在同一部電腦裡同時進行繪圖，同時每位使用者彼此之間仍然可以利用最直覺的面對面溝通方式相互交換意見與討論 [Stewart, 1998]。SDG 的概念讓電腦這類的數位系統不再只是個人使用的工具，而是能夠與他人共享想法，以及共同操作媒材的溝通環境。但這樣的概念僅僅應用於單一個電腦上，而且也僅僅將複數個輸入裝置視為滑鼠或是鍵盤的延伸，而沒有善加利用在未來輸入裝置上也可能具有的運算能力，未能擴展未來人與機器間互動形式的更大可能性。

在各種由數位媒材構成的虛擬環境中，包覆型自動虛擬環境(cave automatic virtual environment, CAVE)的沉浸式空間展示能力已備受肯定，CAVE 的沉浸式空間經驗則讓原本多是平面的數位媒材轉變為立體的內容呈現，透過 CAVE，使用者可以獲得接近真實空間的感官經驗。在建築設計上，相較於縮小的實體模型，CAVE 更能快速提供身處其中的認知，對於設計的溝通與傳達是相當具有幫助的，例如交大建築所應用在重現已消失的城市，將中國唐代的長安城，包括都市的外貌、規劃、以及生活在其間的居民的活動，在經過詳細的考證後，透過 CAVE 展示出來[Liu, 2003]。現在則有許多的研究在探索 CAVE 能夠帶來的更多應用，例如在日本有結合擴充實境(augmented reality)的 MR 計畫[Tamura, 2001]、或是普林斯頓大學對於類 CAVE 的大型顯示牆所進行的討論[Li, 2000]。CAVE 也逐漸地由單純的顯示目的，轉為能夠提供多人共同溝通討論的平台，然而由於 CAVE 的多電腦分散式架構，CAVE 上的輸入裝置一直相當的受限，多半都得使用特製的輸入設備，而大多數的研究也都在討論空間輸入的可能性，忽略了多人輸入的部份，也使得 CAVE 雖然提供多人同時瀏覽的空間，卻缺乏了多人同時互動的功能。

1.2 研究問題與目標

在設計溝通的過程中，設計者運用多種媒材的呈現以傳達設計概念，並與其他參與者相互討論設計議題。在傳統的設計溝通過程中，設計者藉由口頭說明、肢體動作配合包括圖面、模型等等同時運用多種媒材傳達設計概念，其他參與者也在同時同地透過操作這些媒材給予各自的意見與評論(圖 1.1 左)。而在電腦影響了設計後，數位媒材的運用提供了設計者更多傳達設計概念的方式，現在的設計者除了以往的圖面、模型外，還可以額外用 Powerpoint 簡報、虛擬空間中的立體模型、動畫情境模擬等更多樣的數位媒材，藉以更完善地傳達設計想法。電腦在現在的設計發表場合中扮演了極為重要的角色，因為數位媒材的部份都必須透過電腦來呈現(圖 1.1 右)。

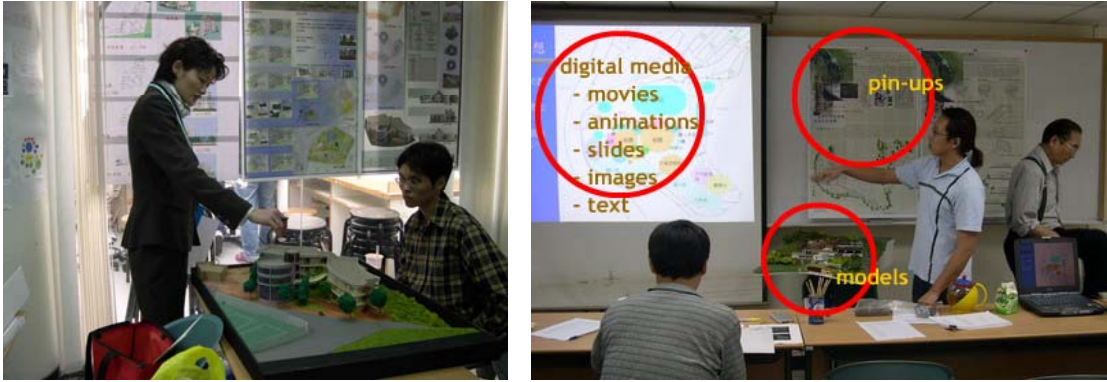


圖 1.1 (左) 傳統設計溝通過程 (右) 運用數位媒材的設計發表場合

電腦同時也帶來一個傳統設計溝通場合中沒有發生的問題：傳統上，設計者與其他參與者之間可以透過直接面對面的溝通，也可以透過共同與各種媒材互動的形式來相互表達意見(圖 1.2)；然而電腦上的軟體都是針對單一使用者設計的。換言之，除了操作電腦的人之外，其他的參與者都無法直接與數位媒材進行互動，這讓設計溝通過程出現了阻礙與不順暢，其他參與者希望能夠與數位媒材進行互動時都必須間接透過操作電腦的人，他們必須通知控制電腦者以他們希望的方式操作數位媒材以獲取需要的資訊或是呈現希望表達的內容，然而溝通上的失誤與認知上的差距(例如：控制電腦者因為沒有完全了解參與者的意見，而沒有按照其希望的方式動作)，讓這樣的溝通過程變的很沒有效率(圖 1.3)。

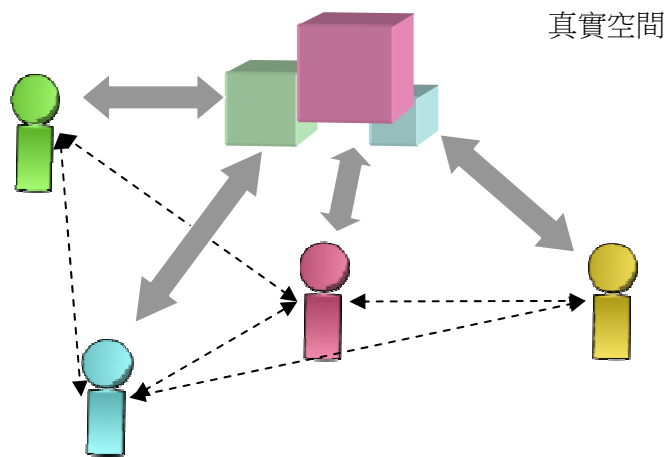


圖 1.2 傳統的設計溝通過程中互動的機制

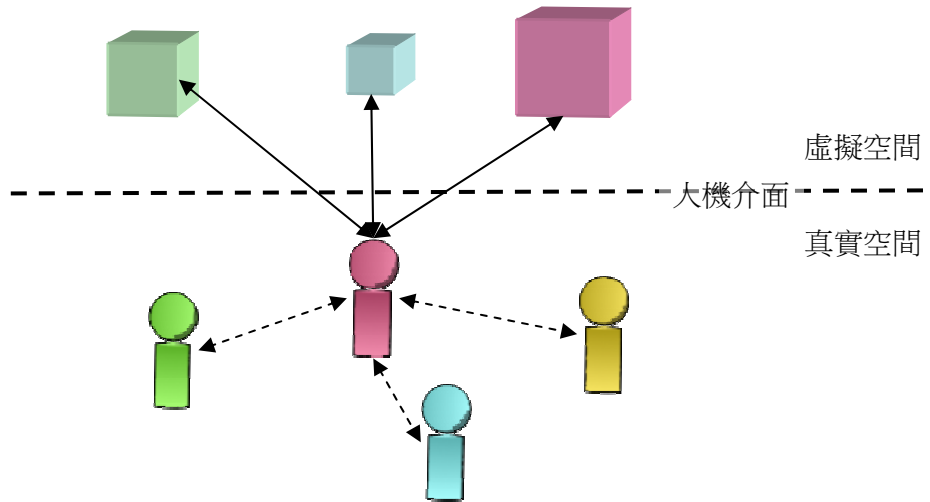


圖 1.3 使用數位媒材時只能透過操控電腦者與媒材互動

本研究的主要目標在於將 SDG 與 CAVE 結合，提出一套能讓多個使用者同時在 CAVE 裡與數位媒材進行互動的系統 - 虛擬發表空間(virtual presentation space, VPS)，讓 CAVE 發展成為設計溝通的平台。使用者只需要利用各自的個人運算設備(如：個人數位助理、筆記型電腦、手機)便可以直接與虛擬物件之間互動，而不需要透過操控電腦的特定使用者，也不需要特製的設備。使用者分別使用各自的個人運算裝置作為輸入，並在螢幕上區別個人。每個使用者都可以同時地在虛擬空間中繪圖、抓取物件、開啟檔案、打字等互動行為。也可以讓使用者直接將各自裝置中的資料貼至 CAVE 環境中與他人共享以供彼此討論，或是直接在不同的數位媒材格式間轉換，省卻了事前的檔案傳輸、整理等繁複的工作。本系統也承繼 SDG 的特性，所有使用者在與虛擬空間中的物件互動的同時，彼此之間仍保有最自然的溝通方式 - 面對面的溝通(圖 1.4)。

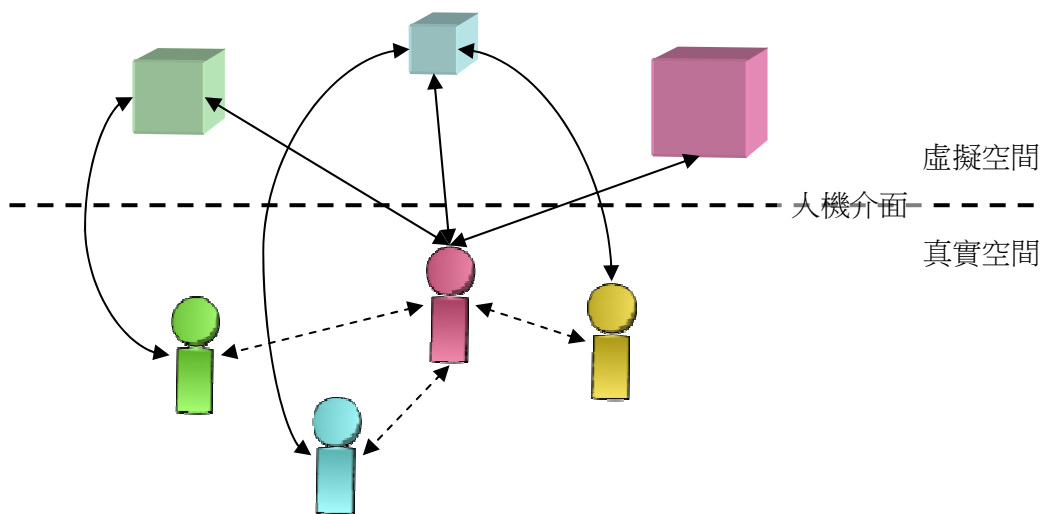


圖 1.4 VPS 下的人與人之間、人與媒材間的互動關係

VPS 是能夠在 CAVE 環境下進行最常見的設計溝通過程 - 設計發表的虛擬環境，設計者將自己的設計透過多種數位媒材的呈現，其他的使用者(其他設計者、評審)除了參與討論外，同時也透過隨身的裝置與設計者的媒材物件互動，也可以將存放在各自的運算設備記憶體中自己帶來的數位媒材內容放置進此空間中與他人共享或作為討論上的補充之用，以期達到更有效更順暢的設計溝通過程。透過這樣的系統，設計溝通的過程不再因為電腦的介入而阻絕了其他參與者直接操作媒材的機會，讓所有人都有機會透過媒材完整地表達自己的想法。

1.3 研究步驟與方法

本研究希望透過將傳統設計發表中人與媒材間互動的形式放置到虛擬空間中，除了保留傳統設計發表過程中人與人間、人與媒材間的互動關係外，並能完整利用數位媒材所帶來的好處，例如：動態的內容呈現、直接修改、立體且為真實比例的空間再現等。本研究透過 VPS 的實作並嘗試在此系統上呈現設計發表過程中人與人之間、人與媒材間互動的行為，作為系統的評估以及本研究的結論。研究的過程分為對於系統的分析與規劃、系統的實作過程以及系統雛形的評估等三個部份。

步驟與方法如下：

1 系統分析與規劃：

系統分析：分析現有環境的軟硬體設備，以了解系統在實作時必要採用的技術，並討論先前研究中的類似系統模型，並討論要加入完整呈現傳統發表過程中所需的互動需求。

系統規劃：將運用本系統呈現設計發表過程的互動特徵切割為以下元素

- ❖ 多種媒材形式：觀察設計發表所使用到的多種數位媒材的形式。
- ❖ 透過網路傳輸：分析在 CAVE 這樣的分散環境下，所有資料的傳遞都必須透過網路傳輸所產生的問題並討論如何在實作中克服。
- ❖ 使用者的輸入：分析傳統設計溝通轉換到運用數位媒材進行設計溝通時不足的部份，讓所有參與者都能與虛擬空間中的數位媒材互動，並能提供額外的補充資料。

- ❖ 分辨各使用者：分析在共有的顯示環境與軟體系統下，在虛擬空間中分辨各使用者的機制。
- ❖ 分散式的 SDG 架構：SDG 原先的設計是針對在單一個人電腦上連結多個輸入裝置(如：鍵盤、滑鼠)，要將 CAVE 與 SDG 結合則必須修改 SDG 的概念成為符合多部電腦的型態。
- ❖ 設計者自訂主題式的媒材展示與互動方式：發表的過程具備有主題性以及發表者(presenter)的特性，要讓本系統能夠呈現完整的發表過程，主題以及使用者角色的區隔是必要的。

2 系統實作

將系統切割為以下數個子系統分別進行實作：

- ❖ 裝置管理：管理及分辨多個輸入裝置
- ❖ 訊號輸入：從輸入裝置取得訊號並轉化為指令
- ❖ 情境管理：由發表者設定的情境調整系統目前提供的互動功能與形式，以進行各種主題的討論
- ❖ 資源管理：管理及載入各種支援的數位媒材資料
- ❖ 資料運算：從輸入部分取得使用者輸入並運算各種媒材的回應
- ❖ 顯示輸出：將運算後的視覺回饋呈現在畫面上完成互動的過程

- 3 系統評估：利用虛擬發表空間呈現一個娛樂設施空間規劃的設計案，此設計案原先是採用傳統的媒材進行設計的溝通與討論，在重新以本研究中提出的系統進行設計的溝通與討論後與原先傳統的方式進行比較與評估。

第二章 文獻回顧

2.1 人機互動

所謂的人機互動(human-computer interaction)指的就是人類與電腦等機器之間的互動關係，最常見的人機互動關係為：人類對電腦下達指令，電腦根據這些指令進行運算後，將結果以人類感官能感知的形式(如：視覺、聽覺等)將運算的結果回饋給人類(圖 2.1)。

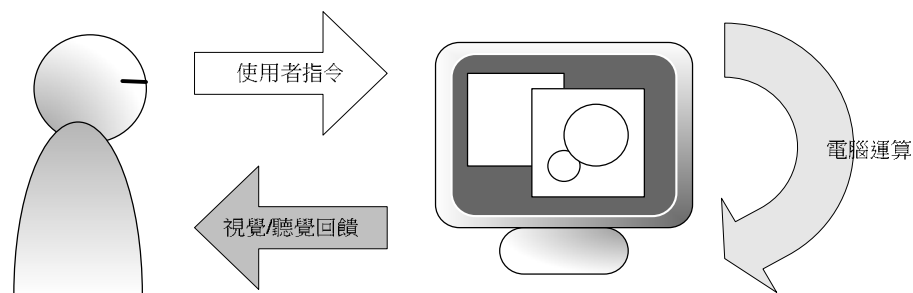


圖 2.1 人機互動(human-computer interaction)

人類與電腦間的互動歷史可以回溯到世界上第一部電腦的產生開始，一般都認為第一部電腦是在 1946 年開發出來的 ENIAC。隨著電腦的發展，電腦與人之間的互動形式也漸漸地在改變。Nielson[1993]將人類與電腦間的互動形式整理後分為五個時期(1946 ~ 1993)：

1. 批次處理系統 (batching systems)

在電腦發明的初期，電腦是昂貴的貴重物，多隸屬於大學或是研究單位，並由多人共享。要使用電腦的人必須將程式寫在卡片上，送進電腦的工作排程中，過幾天後才會得到程式執行或是計算的結果。由於電腦的反應實在太慢，跟人類間的互動並非即時的，因此從目前的觀點來看甚至不認為當時的電腦能稱得上能跟人類互動。

2. 條列式命令系統 (line-oriented systems)

隨著電腦處理速度的增加，接下來的電腦已經可以接受人類以鍵盤輸入一行指令，並馬上針對該指令做出反應，不論是執行工作或是回應訊息。這個時期的系統包括 MS-DOS 與純文字模式的 UNIX 系統。

3. 全螢幕介面 (full-screen interfaces)

全螢幕的介面讓人機互動的方式跳脫了一行一行、一個指令一個動作的互動形式，讓使用者可以用按鍵切換的方式選擇螢幕畫面裡的視覺元件，例如：表單(form)、選單(menu)等，再用鍵盤輸入訊息，這個時期由全錄研究中心(Xerox Palo Alto Research Center)所發展出來的許多視覺元件也一直延用至今，成為互動介面的一種標準[Meyes, 1998]。

4. 直接控制 (direct manipulation)

幾乎是現今電腦標準配備的滑鼠在 1965 年時由史丹佛實驗室(Strandford Research Laboratory)提出其原型，並在 1981 年首次商業化成為 Xerox 的 UNIX 作業系統 Star system 的一部分，也成為之後 Apple 發表的個人電腦 Lisa(1982)跟 Macintosh(1984)的標準配備，更在現在成為所有電腦的標準互動裝置。滑鼠的發明讓人類得以與電腦進行直接的互動，例如：點選畫面裡的按鈕、選擇文字、拖放動作(drag'n drop)等，而電腦也提供即時的視覺回饋，讓人機互動的形式就如同在真實世界中操作物件一樣。這樣的互動形式也有了一個常見的代名詞，WIMP(windows, icons, menus, pointing device)。

5. 次世代介面 (next generation interfaces)

從 1991 年後，許多的研究都開始認為人機的互動形式將走向自然與直覺，讓電腦來適應人類的行為模式，而不是讓人類去學習複雜的操作方式以與電腦進行互動。

Laurel[1993]就認為，在未來的人機互動形式中，人類將不會感受到與機器之間的互動會有所謂的介面存在，他稱之為消失的介面(the vanishing interface)，人類將會很直覺地使用電腦，猶如日常生活一般地自然。Norman[1998]同樣認為電腦已經成為一個過度複雜的機器，因為現在的個人電腦工作越來越複雜，幾乎想把所有人類的生活與工作都在一部機器裡完成，也因此讓電腦越來越難用；他認為未來應該要發展針對特定工作的多種小型而易於使用的裝置，而不是一個超大型但難用的超級電腦。

Weiser 在 1991 年所提出的遍佈運算概念(ubiquitous computing)，乃至於後來 Ishii 發展出來的可觸式媒材(tangible media)同樣闡明了未來的人機互動將不會再以人與電腦的形式存在，而是隨著科技的進步，運算設備的縮小化以及嵌入化，所謂的電腦將以無所不在的運算裝置形式存在，人類將處於一個遍佈各式各樣運算設備的環境中，互動的形式也將有很大的改變[Ishii, 1997]。

Dertouzos[1997]在書中說明他認為未來人與電腦將共生於一個大型的資訊市集 (information marketplace)中，人與人間、人與電腦間、電腦與電腦間彼此交換、買賣或共享資訊。接下來他更提倡人本運算的概念(human-centric computing)，包含五大重點：自然互動、自動化、個人化資訊存取、協同工作、個人化[Dertouzos, 2001]。

2.2 多人互動

在傳統的協同設計(collaborative design)行為中，人們在共享的空間(physical shared space)中藉由大量的手勢(gesture)、繪圖(drawing)與書寫(writing)相互表達自己的想法以進行共同討論與發想[Tang, 1989]。其中，多人互動(multi user interaction)是達到協同工作必須的條件，協同工作的成員，必須能同時地在同一個共享空間中進行溝通與互動[Tang, 1990; Stewart, 1998]。

在電腦這種數位媒材出現後，現在的設計者都習慣於運用電腦裡的應用軟體進行設計，然而目前多數的應用程式都是針對單一個人使用的，這使得上述多人同時的互動變得困難。

Gross[1999]認為在數位媒材介入設計後將使得設計教育與溝通有重大的改變，他提出了六種新的設計工作室(design studio)型態將在傳統設計工作室整合數位媒材後出現，其中便包括虛擬設計工作室(virtual design studio)。他雖然認為即使在數位媒材介入後仍無法讓參與者有興致參與協同設計，但他仍然肯定透過數位媒材進行的多人同時互動在設計溝通與評審時帶來的方便性。

Achten[2002]也同樣對於如何將協同工作應用在電腦輔助建築設計提出他的看法。他認為要建置一個符合協同設計的電腦輔助建築設計環境必須能夠提供以下幾個特徵，包括：多人參與、能感知其他參與者在同一個空間中、所有參與者都能與設計的內容進行互動、參與者之間可以彼此分享或交換資訊內容等，他更強調了溝通語言(communication language)、溝通環境(communication environment)、溝通行為(communication behavior)等等多人溝通相關的議題是在未來協同設計該著重的研究重點。

2.2.1 透過網路的多人互動

要達到人與人之間的溝通而又能與數位媒材進行互動，最常見的方式就是透過網路的形式。由於網路可以讓在不同地方的電腦相互連接，人類在使用電腦與數位媒材互動的同時，人與人彼此間還可以利用語音電話(internet phone)、視訊會議(video conference)、聊天室(chat room)等網路技術相互溝通，也可以分享各自的數位資料，在 Gross[1999]的研究中就指出有許多虛擬設計工作室就是透過這樣的方式進行溝通與協同工作。而為了讓透過網路的多人互動更為易用，也有許多的研究在進行開發整合數位媒材與人際溝通的虛擬環境。

多人群組概念發展與原型設計系統(multiuser groups for conceptual understanding and prototyping)系統就是讓所有使用者透過網路在同一個共享的虛擬環境下從事設計。彼此間設計的虛擬物件會在共享的虛擬環境中依據被設計的特性直接相互地影響，例如提供支撐或是相互的結合，並同步地反應給所有參與者，達到協同設計的目的；使用者間也將語音經由 Moving Picture Expert Group(MPEG) Layer 3 (mp3)的即時壓縮再透過網路傳輸，以讓彼此能利用語音進行溝通，達到多人互動的目的[Cera, 2002]。

而透過語音、文字甚至是視訊有時仍無法讓使用者間自然地溝通，因此也有研究專注在如何讓網路上的使用者能在除了文字或視訊之外，利用虛擬人物(avatar)的肢體動作以表達更完整的情緒，而能讓使用者之間有更自然與直覺的溝通方式[Endo, 2003]。

2.2.2 在實體共享空間的多人互動

然而以目前的科技技術，透過網路的多人互動仍然與真實面對面(face-to-face)的溝通有所差異而不夠自然與直覺，為了讓多人可以在同一個空間中彼此溝通，而又能與數位媒材進行互動，最簡單的做法或許就是給所有人一部電腦，然後在房間中放置一部投影機以提供共享的視覺空間，這也是目前在進行設計溝通或評審時最常見的情景[Pedersen, 1993]。

數位桌(UbiTable)是一個不同使用者間透過一塊在實體環境中的互動式桌面彼此透過桌面交換數位資料，使用者各自操作自己的電腦，同時也能面對面溝通(圖 2.2)。系統在使用者各自的電腦上切割出共享區與私人區，使用者只要將電腦私人區裡的物件拖放置共享區，系統便會自動將物件透過投影的方式投射在桌面上以供分享，對方便可以從桌面上取得共享的物件[Shen, 2003]。

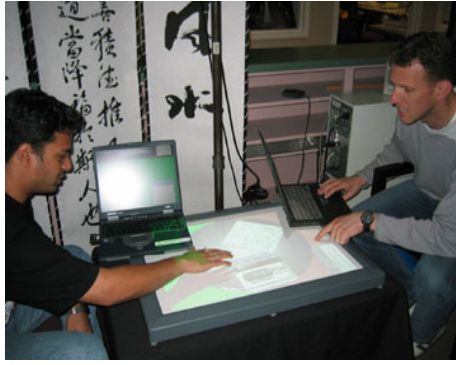


圖 2.2 數位桌 (取自 Shen, 2003)

加州大學柏克萊分校(University of California at Berkeley)發展的整合便利貼(Post-it)與投影系統的互動式牆面(圖 2.3)，實際應用在資訊建築設計(information architectural design)上，讓多使用者同時參與互動，並結合視覺辨識的技術，由後端的電腦判讀後協助架構設計與每個設計單元的管理，每個不同的設計單元以一個便利貼呈現，電腦在判讀內容後便會依據其內容而在牆面上繪製連接線以表示各便利貼之間的相互關係並顯示在牆面上，而隨意移動便利貼後電腦也會追蹤並自動重新計算畫面[Klemmer, 2002]。



圖 2.3 結合實體便利貼的互動式牆面 (取自 Klemmer, 2002)

Rekimoto 的研究是讓每個人操作一部個人數位助理(personal digital assistant, PDA)共享的投影螢幕，彼此可以透過個人數位助理交換數位資料，或是將資料內容經由投影機投射在投影螢幕上而能據以相互溝通，使用者利用筆的點選放置動作(pick' drop)在不同的電腦系統間交換資料(Fig.2.4)[Rekimoto, 1998]。



圖 2.4 點選放置讓兩個使用者能夠交換資料進行溝通 (取自 Rekimoto, 1998)

之後 Rekimoto 更進一步發展出擴增平面系統(augmented surface)，將他之前的多項研究整合在一起，呈現出一個多人互動的數位媒材環境，除了有共享的桌面，架在上方的掃描器也會自動掃描桌面上實體物件(如：錄影帶)的條碼在經過視覺辨識的運算後，將其相關的訊息投影在物體上(圖 2.5)，除此之外，使用者也可以透過手指的點選或拖拉等動作彼此交換資料及檔案[Rekimoto, 1999]。



圖 2.5 擴增平面系統 (取自 Rekimoto, 1999)

Ishii 也將其團隊所發展的光桌(luminous table)嘗試應用在實際的都市設計案中(圖 2.6)，而結果更說明透過傳統媒材的表現，搭配數位媒材的運算能力以及多人的互動能在協同設計情境下得到相當大的優勢，包括在傳統模型上投射數位模擬的風以呈現傳統媒材無法呈現的動態內容、設計內容的傳達容易被專業與非專業者所感知、鼓勵參與者之間的互動溝通等等[Ishii, 2002]。



圖 2.6 光桌(luminous table)應用在都市設計中的情形 (取自 Ishii, 2002)

2.2.3 單一顯示多人群組軟體

相較於單人使用的軟體，群組軟體(groupware)所指的就是多人能共同使用的軟體。所有使用者所觀察的、操作的，都是共同的虛擬物件，某個使用者對特定物件做了變更，其他使用者將會察覺到相同的改變。

單一顯示多人群組軟體 SDG 是群組軟體的一個分支，單一顯示多人群組軟體讓同一地的使用者，透過同一個顯示裝置作為輸出，透過各自的輸入裝置在同一個虛擬空間中與虛擬的物件互動的同時，也在實體的共享空間以人類自然的溝通方式與真實的人類進行互動 [Stewart, 1998]。Stewart 指出，透過單一顯示多人群組軟體，使用者可以同時地與軟體系統互動，不再需要與其他人交換控制權，而共享的顯示裝置也讓所有參與者都能得到訊息，而不需要在多個電腦系統間切換，也能直接看到與電腦互動的回饋反應，真正達到多人同時互動的效果。

相較於在透過網路進行的協同工作應用程式，單一顯示多人群組軟體讓系統的使用者之間能進行更完整的互動與溝通，例如：使用者可以容易地看到所有參與者、可以觀察彼此的行為、可以用手勢、表情等自然的溝通方式[Tang, 1991]；更重要的是，單一顯示多人群組軟體讓使用者可以做任何能用電腦做的事，使用者不但可以享受電腦帶來的便利性，彼此之間還能保有最自然的溝通方式(圖 2.7)[Zanella, 2001; Stewart, 1999]。



圖 2.7 兩個人用各自的輸入裝置在同一個顯示裝置上互動 (取自 Zanella, 2001)

關於單一顯示多人群組軟體的研究，MMM[Bier, 1991]是最早提出 SDG 概念的研究，也企圖要將這樣的觀念實作在作業系統上。

而 Inkpen[1997]在其博士論文中進行的研究便指出，兒童在運用 SDG 作為系統的應用程式中獲得較多的樂趣以及較不會分心等具正面意義的結果。

在卡內基-美隆大學(Carnegie-Mellon University)的人機互動研究所(Human-Computer Institute)所進行的多裝置互動計畫(Pebbles project)進行了一連串關於 SDG 應用的研究，利用 PDA 與個人電腦的連結性，讓多個使用者能同時透過不同的輸入裝置在同一部個人電腦上繪畫、操控軟體、控制游標鍵盤等等工作[Myers, 1998; 2000; 2001]。

多人群組軟體工具(SDGToolkit)則是一套讓研究者能夠簡易利用複數的鍵盤與滑鼠建置 SDG 環境的快速開發工具，利用 Microsoft 的 DirectInput 函式庫實作以提升電腦的反應速度，能夠讓多人互動更為即時[Tse, 2004]。

多型態輸入(multimodal input)則是跟單一顯示多人群組軟體相類似的概念，所指的是使用不同的輸入裝置操控同一部電腦系統，不同的是有關多型態使用者輸入的研究比較著重在輸入方式與系統架構的研究上。例如：Ilmonen[2002]所提出的 FLUID 系統架構，將系統分為 1)輸入層 2) 資料處理層 3)應用程式(圖 2.8)，透過在各個層之間傳遞各種不同的訊息資料以達到完成輸入指令的動作；Cohen[1999]則提出了一個大型的架構 - Quickset，這個系統可以同時地支援手勢與語音的輸入。TYCOON 是一個多型態使用者輸入的模型，定義了處理不同裝置間訊息的方法(圖 2.9)[Martin, 1998]。

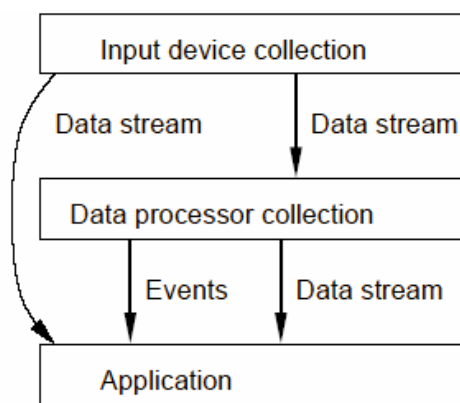


圖 2.8 FLUID 所定義的多型態輸入軟體的架構 (取自 Ilmoenn, 2002)

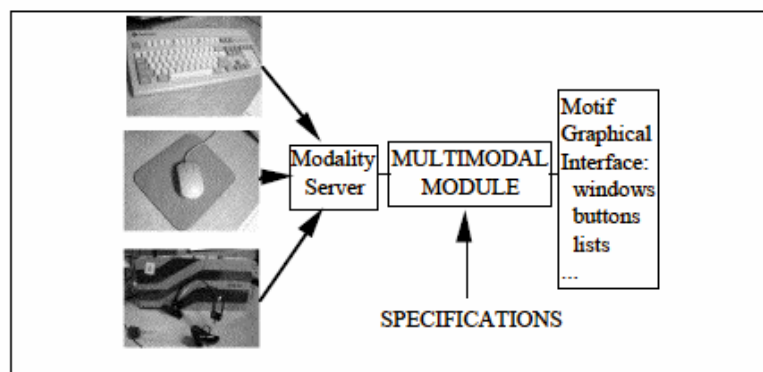


圖 2.9 TYCOON 所定義的多型態輸入裝置 (取自 Martin, 1998)

2.3 人與 CAVE 互動

2.3.1 包覆式自動虛擬環境

在各種數位系統中，包覆式自動虛擬環境 CAVE 模擬出的虛擬共享空間(virtual shared space)，相較於其他媒材更容易在人類心智模型中形成空間感知 [Huang, 2002]，因此已被廣為應用在設計傳達(design representation)上。

CAVE 是一種由多部電腦構成的數位環境，最早是 1993 年建置在伊利諾大學(University of Illinois)[Cruz-Neira, 1993]，之後出現了許多不同種類的 CAVE 系統，但架構上基本都是由多部電腦進行運算並將運算後的視覺輸出至複數的平面顯示牆上。CAVE 提供使用者沉浸式的、擬真的、具空間感的導覽體驗。CAVE 的視覺輸出會成像在大型的牆面尺寸螢幕上，利用佔滿人類視覺感知區域的方式，讓人產生有如身陷其中的空間感覺，許多研究也都在探索這種沉浸式的視覺經驗所能帶來的應用方式。

日本的 MR 計畫(MR project)利用小型攝影機及都市的實體縮小模型，讓攝影機在模型中穿梭，並利用 CAVE 作為視覺呈現工具，讓觀者真實體驗沉浸式空間瀏覽的感覺(圖 2.10) [Tamura, 2001]。



圖 2.10 (左)1/87 的縮小街道實體模型 (右)CAVE 的空間感 (取自 Tamura, 2001)

普林斯頓大學顯示牆計畫(Princeton Display Wall Project)更提出了許多利用這樣多牆面顯示系統的應用，包括運用在遊戲上、設計上以及設計傳達上。例如：建築的空間導覽(圖 2.11)、利用 CAVE 進行簡報的播放(圖 2.12)或是將 CAVE 視為草圖設計的環境(圖 2.13)等等 [Li, 2000]。研究中也提到，由於牆面大小的影像會佔滿人類的可見區域，人的視覺感知不會受到外框或是邊界的影響，而能完整地接收影像訊息，這是小螢幕所無法帶來的效果。



圖 2.11 建築的空間導覽 (取自 Li, 2000)

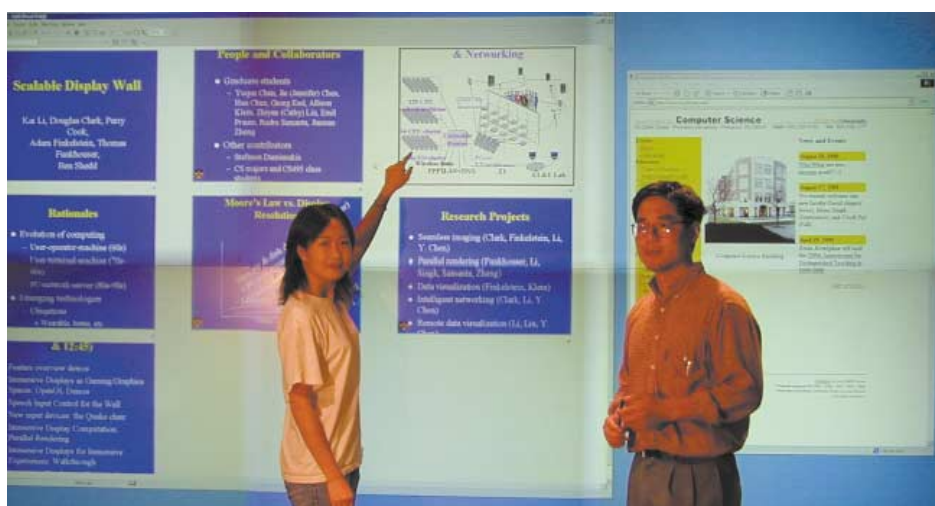


圖 2.12 利用顯示牆進行 Powerpoint 簡報 (取自 Li, 2000)



圖 2.13 將顯示牆作為草圖的展示空間 (取自 Li, 2000)

而 Welch[2000]的研究也描述了未來的辦公室會善加利用大尺寸的空間模擬器，進行高畫質影像的投射(圖 2.14)。



圖 2.14 未來辦公室的想像圖 (取自 Welch, 2000)

2.3.2 CAVE 上的輸入裝置

然而受限於單一電腦所能使用的輸入裝置，再加上 CAVE 是屬於一種由多部電腦所構成的分散式的複雜系統，CAVE 上的輸入裝置不但需要使用特製的裝置，也多只能提供單一使用者的操作，因此 CAVE 雖然可以提供同時多人的瀏覽，而且是提供沈浸式的空間體驗，但在互動部分還是多是由個人所進行的，許多的研究則思考利用透過網路的方式讓多人在同一虛擬共享空間中進行互動[Endo, 2003; Cera, 2002]。

在 CAVE 上的實體輸入裝置相關的研究，主要都在空間輸入(spatial input)的部份以期能發揮 CAVE 表達空間的能力[Bowman, 2001]。赫爾辛基科技大學(Helsinki University of Technology)的電訊與多媒體實驗室(Telecommunication and Multimedia Laboratory)的小型 CAVE 系統(HutCAVE)(圖 2.15)，則使用了多種特製的輸入裝置以在 CAVE 的模擬空間中進行互動[Kontkanen, 2002]。

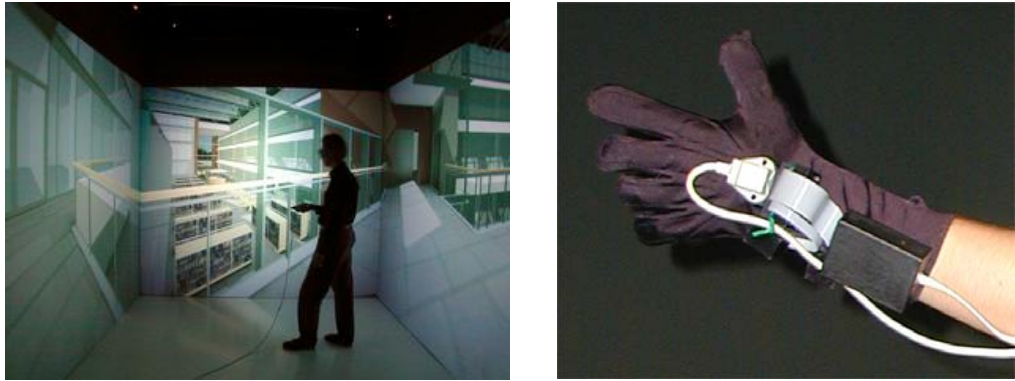


圖 2.15 在 HutCAVE 中 (左)瀏覽情形 (右)特製的輸入裝置 (取自 Kontkanen, 2002)

CAVE 的創生者 Cruz-Neira 也在一份研究中提到，CAVE 之類的虛擬環境(virtual environment)到今日為止還有一些問題尚未克服，其中便包括：CAVE 裡的互動以及溝通。他指出，由於對於如何在沉浸環境中進行互動的研究尚未能有更進一步的突破，使得在人類一直無法在 CAVE 環境中進行較為直覺的互動，也讓使用者始終無法從桌面使用環境轉移至虛擬環境，因此在未來的研究需要開發新的互動模式[Cruz-Neira, 2003]。

第三章 系統分析與規劃

3.1 系統分析

本研究提出的虛擬發表空間 VPS，讓設計發表時每位參與者的互動形式能夠在 CAVE 中展現，並能完整利用數位媒材所能提供的優點，例如：立體視覺、資料檔案交換、動態內容等。除此以外，VPS 在規劃上也必須保留擴充的空間，以在未來支援更多的輸入裝置以及更多的數位媒材格式。

3.1.1 設計發表過程中人與數位媒材間的互動關係

由於每一種媒材都有其表達能力的優勢與限制，用任何單一的媒材都無法完整傳達複雜的設計概念，因此在設計發表的過程中，發表者都會同時使用多種媒材以完整地傳達設計的想法。這些媒材除了最直接的口頭說明、手勢及肢體動作外，在傳統媒材上，有文字(報告書)、圖面(照片、各種平面與透視圖面、版面)以及實體模型；在數位媒材部分則會運用 Powerpoint 簡報、2D 數位影像檔案、3D 數位模型檔案、渲染的影片動畫以及以影像為基礎的虛擬實境(image-based virtual reality, 如：QuicktimeVR)等。

所有使用者(包括發表者與其他參與的設計者或評審)在目前運用數位媒材的發表的過程中與這些數位媒材之間互動的關係如表 3-1 所示：

使用者	互動方式	數位媒材
發表者	翻頁 (上一頁/下一頁)、指示	Powerpoint 簡報
參與者	指示、觀看	
發表者	變換圖片	2D 影像
參與者	觀看	
發表者	操作視角	3D 模型
參與者	觀看	
發表者	播放/暫停/停止	影片動畫
參與者	觀看	
發表者	旋轉/傾斜	以影像為基礎的虛擬實境
參與者	觀看	

表 3.1 發表過程中使用者與數位媒材之間的互動關係

為了要讓 VPS 裡的參與者不再是只能進行觀看的動作，而能夠真正地直接操作虛擬空間中的數位媒材，系統必須能夠讓所有的使用者都利用相同的互動方式與媒材溝通；然而這樣一來每個人都在操作媒材，會造成整個發表過程的混亂，所以使用者角色的區分還是必要的，發表者也必須能夠控制整個發表過程要展現的東西，因此需要一個只有發表者能夠操作的情境控制介面。

而為了使 VPS 能夠善加利用數位媒材相較於傳統媒材的最大優勢：動態內容，本研究也提出了一種新的數位媒材 - 虛擬版面(virtual pin-ups)。使用者可以將自己的隨身裝置中的媒材檔案放置到虛擬版面上，也可以直接將 VPS 中的各種數位媒材直接轉換成影像檔貼至虛擬版面上(例如：將 3D 模型展示中自己關注的某個視角畫面擷取下來即時貼至版面上)，或將其上的檔案下載至個人裝置後帶走，達到資訊交流的目的。透過虛擬版面的運用，

整理以上的討論，在 VPS 裡使用者與各種媒材間的互動關係如表 3-2：

使用者	互動方式	數位媒材
發表者 參與者	翻頁 (上一頁/下一頁)、指示、觀看	Powerpoint 簡報
發表者 參與者	變換圖片、觀看	2D 影像
發表者 參與者	操作視角、觀看	3D 模型
發表者 參與者	播放/暫停/停止、觀看	影片動畫
發表者 參與者	旋轉/傾斜、觀看	以影像為基礎的虛擬實境
發表者 參與者	切換情境、控制 VPS 的互動能力	情境控制介面
發表者 參與者	上傳圖片、媒材轉換、下載圖片	虛擬版面

表 3.2 VPS 中使用者與數位媒材之間的互動關係

3.1.2 CAVE 環境

VPS 的 CAVE 硬體環境為工業技術研究院 (ITRI, Industrial Technology Research Institute)所研發，已建置在國立交通大學浩然圖書館七樓發展館裡作為主要的展示空間，本研究基於地利，直接利用該系統作為 VPS 的基礎設備。主要運算系統是由七部電腦所構成。硬體環境分為一部控制伺服器(control server)與六部顯示客戶端電腦(display client)，控制伺服器負責協調六部客戶端機器的運作，client 則負責各畫面的運算與輸出至投影螢幕(圖 3.1)。控制伺服器裡負責記錄目前虛擬環境的狀態，例如：鏡頭座標、目前播放格數等，並以固定的間隔時間(10 毫秒)將目前的狀態資料不斷地傳送到三面牆的各兩部電腦，透過兩部電腦運算左右眼畫面，並將其輸出透過偏光濾鏡(polarized filter)投影至同一面投影牆上，使用者戴上偏光眼鏡後左右眼各自接受具有視差的影像便可在腦中形成立體的視覺。

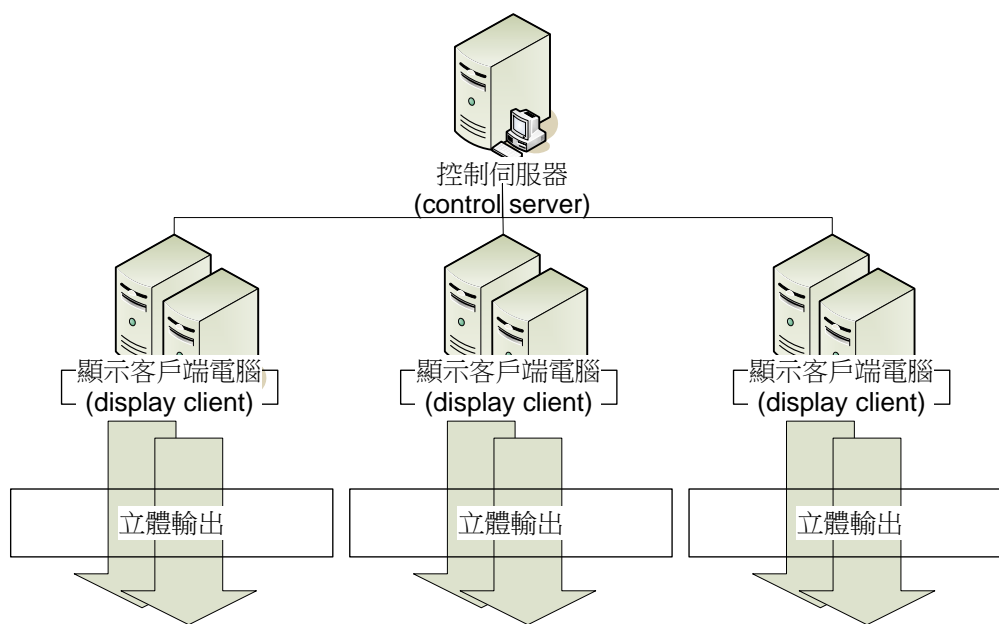


圖 3.1 CAVE 的硬體架構

在內容軟體的部份，此 CAVE 環境原先的設計僅支援立體動畫的播放，因此重新設計及製作新的檔案資源管理軟體是必要的，以讓 CAVE 可以載入及呈現更多樣的數位媒材形式。而 CAVE 的分散式硬體架構，則帶來透過網路傳輸資料的必要性以及新的系統必須處理同步的問題。舊有的設計也未考慮到輸入及與使用者互動的機制，要呈現不同的內容都必須透過系統操作者在控制伺服器上的操控，無法由使用者決定，而動畫的呈現方式也讓使用者不易取得特定的內容以進行討論。

3.1.3 輸入裝置

由於個人電腦的作業系統在設計上僅能接受一組的輸入裝置，因此為了達成多人同時互動，必須在系統中自行實作輸入的功能。一般在 CAVE 裡的輸入裝置都是連接在控制伺服器上的特製裝置，要使用這些裝置使用者不但需要學習，這些裝置也幾乎都是供單人使用。所以 VPS 必須要尋找其他的輸入裝置的可能性。

隨著科技的發展，結合了運算能力、儲存能力以及通訊能力的個人數位設備，如：PDA 或是手機也漸漸普及，成為現代人隨身攜帶的裝置。這些個人裝置不但可以進行運算，也可以儲存數位媒材檔案，運用這些裝置就能讓使用者進到空間中，取出自己隨身的裝置與控制伺服器連結後便可以直接參與互動討論及資訊的分享。

VPS 上以具備無線通訊能力(wireless communication)的個人數位裝置作為主要輸入裝置，例如：具備藍芽通訊的手機、以及具備無線網路的 PDA 或筆記型電腦，藉以讓多人透過這些裝置的互動得以在 CAVE 上實現。而傳統的輸入裝置仍然可以使用作為多模式輸入的一部分。

選用具無線通訊能力的主要原因就是要讓使用者可以不用受到線的羈絆，能自由自在走動、與他人溝通。而這些隨身裝置的小體積，也讓使用者沒有攜帶大型裝置的負擔，能更自然地運用手勢及肢體動作表達意見；許多廠商也都在開發這類無線而且小尺寸的裝置作為輸入或是簡報的控制裝置(圖 3.2)。

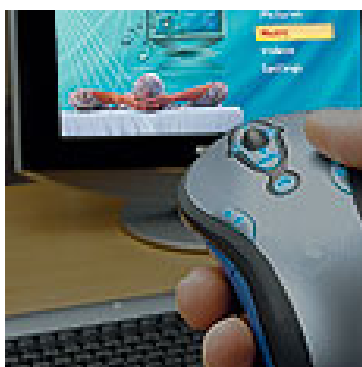


圖 3.2 (左)羅技開發的無線遙控器滑鼠 (取自 Logitech.com); (右)蘋果總裁 Steve Jobs 在 WWDC (Worldwide Developers Conference) 2005 上使用小型無線裝置進行簡報 (取自 Apple.com)

VPS 支援的硬體設備：

1. 具備藍芽傳輸功能或無線網路連線能力的 PDA
2. 搭配藍芽接收器或無線網路的筆記型電腦
3. 具備藍芽傳輸功能的手機
4. 傳統的輸入裝置：滑鼠、鍵盤及搖桿(joystick)

因為要與這些裝置進行連線，我們必須了解連線的方式，以及如何取得訊號。而這樣多人同時地進行輸入，系統就必須能夠同時地接收輸入訊號，並能分辨訊號的來源以在螢幕上能夠分辨是哪位使用者在控制什麼媒材。

3.1.4 單一顯示多人群組軟體的適用性

SDG 雖然已有許多實作出來的系統，然而這些實作的系統都著重在使用一部個人電腦接上兩隻滑鼠，並沒有實作在如同 CAVE 的分散系統上，因此我們必須修改單一顯示多人群組軟體的架構以符合 CAVE 的分散式架構。而要修改 SDG 為分散式的架構，就必須先把輸入裝置跟顯示分開來處理，也就是由控制伺服器統一負責輸入的部份，顯示則交由原本的顯示客戶端即可。

再加上我們需要的環境並不是只有多人同時用就可以，還要必須能夠在特定的設計議題下進行溝通與討論，因為設計發表時主要目的並不是讓許多人一起操作電腦而已，而是發表者有想要表達的主題與想法，互動的目的是希望能夠增進所有參與者之間的溝通，而不是為了互動而互動，這也表示系統必須能夠讓設計者自訂主題式的媒材及互動形式。

3.2 系統規劃

綜合上節各點的討論，VPS 必須具備以下的功能規劃：(1)支援多種媒材格式、(2)透過網路傳輸、(3)使用者的輸入、(4)分辨使用者、(5)分散式的 SDG 架構、(6)設計者自訂主題式的媒材展示及互動方式及(7)可擴充性。

3.2.1 支援多種媒材格式

VPS 必須支援多種數位媒材，然而不同的數位媒材又各包含多種的檔案格式，種類繁雜，因此 VPS 在初期的發展階段先支援最常見的檔案格式，未來還可以加以擴充：

1. Powerpoint 簡報檔：*.ppt
2. 2D 數位影像：BMP(點陣圖檔)、JPG(JPEG 壓縮圖檔)、PNG(可攜式網路影像)
3. 3D 數位模型：3DS
4. 影片動畫：AVI、MPG、WMV
5. 以影像為基礎的虛擬實境：QuicktimeVR

3.2.2 透過網路傳輸

CAVE 環境中所有的資料都是透過網路傳輸，而因為 CAVE 中的顯示客戶端電腦有六台之多，因此不會把輸入裝置分散連接至顯示客戶端電腦而造成管理上的不便，所有的輸入裝置都會連接到控制伺服器，然後再透過網路把所有的訊息傳輸給顯示客戶端電腦。

而任何使用者要跟其他人分享的資料也是透過網路的方式傳送至控制伺服器，再由伺服器判斷將資料轉送至哪部顯示客戶端電腦以提供與他人共享的視覺回饋。

透過網路傳訊息造成的延遲，使得控制伺服器必須肩負起同步的責任確保所有的顯示客戶端電腦能夠顯示同一個時間點的畫面。最簡單的方法是透過控制伺服器在特定的時間間隔送一次同步訊號要求所有的客戶端確認彼此的同步情形，或是把現在整個虛擬空間中的狀態(例如：視角座標或是播放影格)傳送給所有的客戶端要求更新畫面，這樣的方式可以讓客戶端與控制伺服器都維持在相同的狀態下。

3.2.3 使用者的輸入

使用者的輸入裝置除了有線的傳統輸入裝置外，最主要的就是具備無線連結能力的藍芽裝置。

藍芽(bluetooth)是一種無線傳輸的技術，可以讓桌上及膝型電腦、PDA、行動電話、印表機、掃描器、數位相機，甚至是家電用品之間進行短距離的無線連結。每一種具備藍芽技術的裝置皆擁有標準的位址，可以讓您使用低功率的無線電進行一對一或一對七(形成一個小型的網路)的連結，傳輸範圍標準為 10 公尺，最遠則可達 100 公尺。藍芽不僅具有每秒 1MB 的高傳輸率，同時也可以進行加密編碼。由於具備每秒 1600 次的跳頻率，因此很難被電磁波攔截及阻斷。

與藍芽裝置間的連線相當容易，透過藍芽標準中的序列埠傳輸模式(serial port profile, SPP)，讓電腦產生虛擬的連接埠，可以讓直接取得裝置上操作的所有訊號。

3.2.4 同時取得並分辨各裝置的訊號

所有藍芽裝置上都有一組獨一無二的識別碼，能藉以區別各個不同的裝置。VPS 本身是屬於一種動態的環境，裝置可能隨時連線及離線，並不是特定的裝置也不是固定的數量，因此必須規劃裝置管理的系統以監視環境裡的裝置連線情形，再透過輪詢(polling)的方式取得所有在連線狀態下的裝置的輸入訊號進行運算。

任何裝置離線及連線的動作都會通知裝置管理系統，以讓整個系統能夠確知目前各個裝置的連線狀態。

3.2.5 分散式的 SDG 架構

要將以往用在單一個人電腦上的 SDG 架構修改為適合分散式環境，VPS 採用了混合了 TYCOON[Matin, 1998]與 FLUID[Ilmonen, 2002]的架構設計，將系統分為位於控制伺服器上的裝置管理層、輸入訊號處理層，以及位於顯示客戶端上的應用程式層(圖 3.3)。

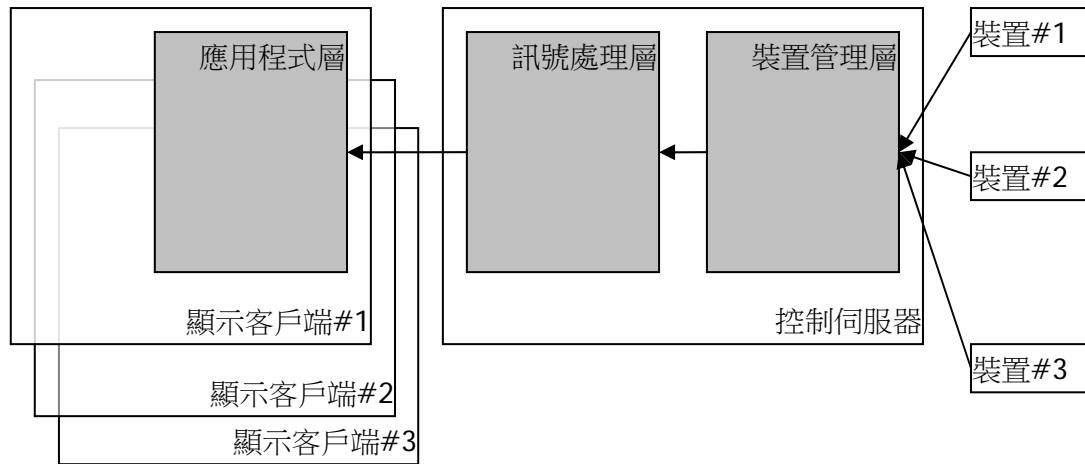


圖 3.3 VPS 的分散式系統架構概圖

3.2.6 設計者能自訂主題式的媒材展示及互動方式

要能夠主題式的進行媒材展示，以符合設計議題的討論，就不能提供無目的的單純互動，而是要能夠讓使用者設定目前的互動方式，以及決定要使用的媒材。

VPS 將參與互動的所有使用者分為兩種身分：發表者(presenter)與參與者(participant)。並將過程分為兩個階段：設計階段(design stage)與執行階段(execution stage)。發表者在執行階段安排整個互動的流程，包括媒材的安排或是設定參與者參與互動的限制，並在實際互動過程中能夠控制流程與調整虛擬環境中所有內容的呈現。參與者則只能在執行階段時，在發表者的安排下與各項媒材互動。例如：發表者設定目前為停止互動狀態，那所有參與者都無法透過裝置進行互動；而當發表者將系統設定為虛擬貼圖狀態時，任何一個參與者都可以自由地將各自個人裝置中的圖檔貼至 CAVE 中，並能夠調整位置大小，或是取得他人貼上去的檔案資料。

3.2.7 可擴充性

因應分散式的 SDG 系統架構以及擴充性的考量，VPS 的系統架構必須被細分為數個模組，如此一來只要透過對各個模組的修改便可以達到擴充的目的而不需要修改整個系統。VPS 在經過切割後分為下列六個模組：

1. 裝置管理：管理及分辨多個輸入裝置，透過對裝置管理模組的擴充，可讓 VPS 支援更多種類的輸入裝置
2. 訊號處理：取得來自輸入裝置的訊號
3. 情境管理：針對目前主題的討論控制不同的互動機制，發表者可以自由設定各種情境，VPS 會依據這些設定提供各種不同的互動的可能性，透過各種自由的設定可以嘗試更多樣化的互動行為
4. 資源管理：管理及載入多種不同媒材的格式，透過對資源管理模組的擴充，VPS 可以支援更多種類的媒材格式
5. 資料運算：從訊號處理部分取得使用者輸入並將其結果反映在媒材上
6. 顯示輸出：於顯示客戶端繪製最後的視覺呈現畫面，對此模組的擴充則可以提供更豐富的視覺感受

3.3 VPS 系統架構

完成系統規劃的 VPS 系統架構如下圖(圖 3.4)所示：

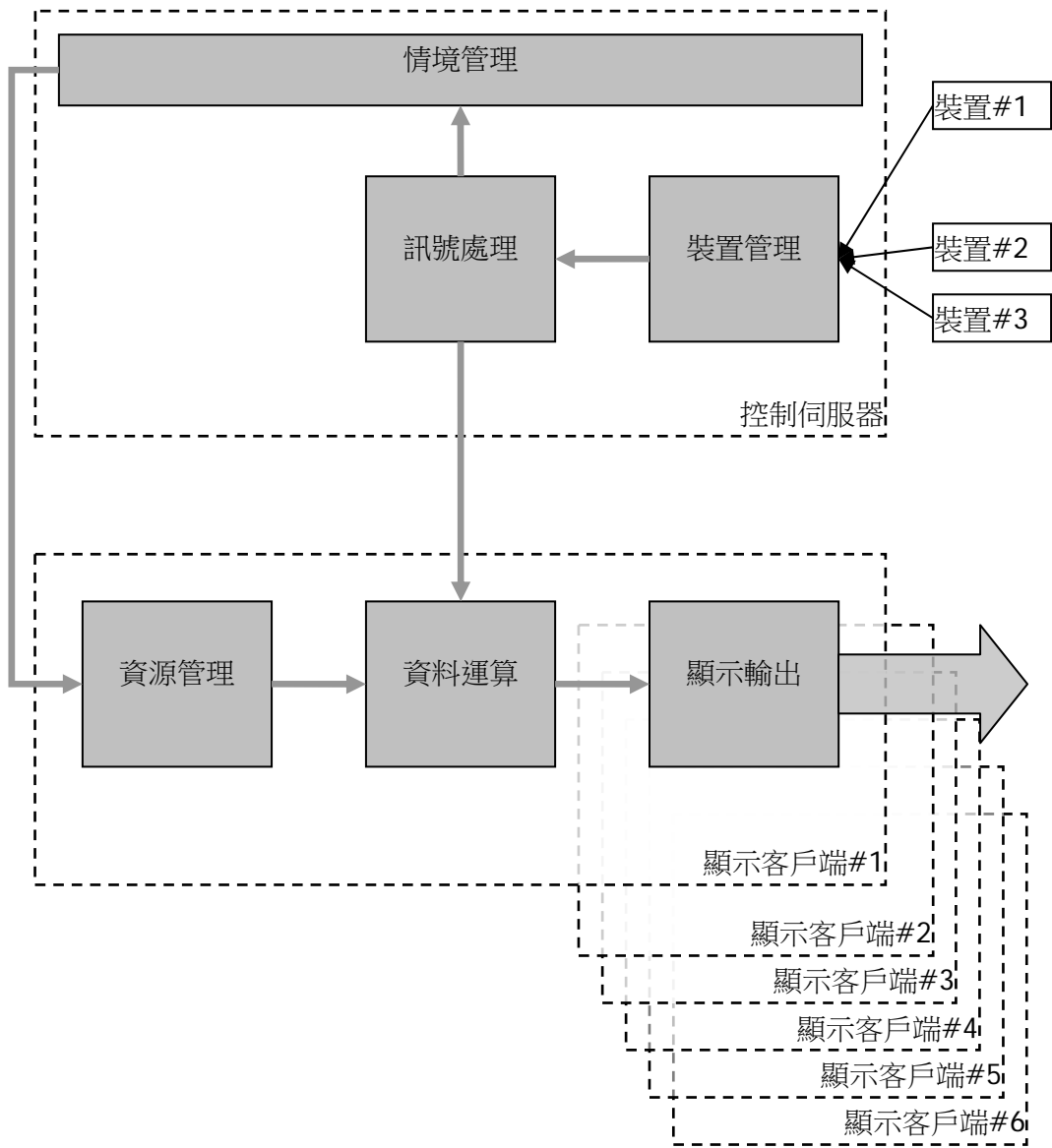


圖 3.4 VPS 系統架構圖

第四章 系統實作

本章為實作(system implementation)的細節部份，說明各子系統的實作過程以及整合各個子系統的系統運作機制。VPS 為架構在 CAVE 環境下的純軟體系統，所有實作皆以 C/C++ 程式語言完成，編譯器為 Microsoft Visual C++以及 Borland C++ Builder，實作涵蓋的相關技術及背景知識有：

- ❖ 多重執行緒 (multi-threading)
- ❖ 動態連結函式庫 (dynamic linking library, DLL)
- ❖ 序列埠通訊 (serial port communication)
- ❖ 網路程式 (windows socket programming)
- ❖ 動態型別轉換 (dynamic casting)
- ❖ 視窗訊息處理 (windows message handler)
- ❖ 物件連結與嵌入 (object linking and embedding, OLE)
- ❖ 通訊協定 (communication protocol)
- ❖ AT 指令集 (AT command set)

系統規劃的結果，VPS 被分為六個模組，分別實作為六個子系統，這些子系統各自有各自的任務，可獨立擴充，但是要透過一個方式將這些子系統串聯整合，才能確保各個子系統在擴充後仍能夠與其他子系統溝通。例如：將來裝置系統擴充支援了新的裝置，但是新裝置的訊號並無法被其他的系統理解，所以在任何子系統實作之前，必須先制定一個共通的協定 (protocol)作為子系統間溝通之用。

這個共通的協定在 VPS 裡被稱為“動作指令”(action command)，動作指令是一種文字系統，任何輸入的訊號都會被轉成相對應的動作指令再傳遞至各個子系統，動作指令穿梭在整個 VPS 系統裡，進行傳遞信息的任務。這個協定在情境管理子系統裡設定，同樣也是可以加以擴充的，有關動作指令的細節會在情境管理一節有詳細的說明。

4.1 裝置管理子系統

裝置管理系統負責管理所有的裝置以及維護已連線裝置的清單，並在有裝置要求與 VPS 連線時啟動訊號處理的執行緒去負責接受訊號，並賦予該訊號處理所負責的裝置的性質描述。而在裝置離線時，也會將該裝置隸屬的訊號處理執行緒關閉以節省系統資源，而在期間則會每隔一段時間(30 秒)送出確認信號給每個訊號處理執行緒以確認裝置仍處於連線狀態。

裝置管理子系統的工作模式如表 4-1。

狀態	工作流程
裝置要求連線時	<ol style="list-style-type: none">1. 取得裝置的識別碼2. 辨識裝置的類型 (手機、PDA、電腦....等)3. 啟動訊號處理執行緒4. 將裝置的性質描述傳入執行緒5. 將裝置名稱新增至已連線裝置清單
裝置離線或斷線時	<ol style="list-style-type: none">1. 輪詢(polling)訊號處理執行緒的狀態2. 查詢連線狀態3. 確認斷線4. 停止訊號處理執行緒，回收資源5. 將裝置名稱從連線裝置清單中移除

表 4.1 裝置管理子系統的工作模式

裝置管理系統除了維護清單之外，也有著辨識裝置以及描述的功能。VPS 目前支援的三種輸入裝置就各有不同的性質描述。性質描述用來說明該裝置可能輸入的訊號內容，以供訊號處理執行緒能用適當的方式處理訊號。

- ❖ 手機：各種按鍵值 (0~9、*、#、音量大小聲....等)
- ❖ 滑鼠：左鍵、右鍵、螢幕座標、單按事件(click)、雙按事件(double click)
- ❖ PDA：按鍵值(上下左右)、模擬滑鼠的事件

如果連線的裝置是屬於藍芽的裝置，則裝置管理系統會透過藍芽的 SPP 模式與裝置進行連結，再將連線後的資源轉交給訊號處理執行緒進行訊號的接收與後續處理，訊號處理執行緒並不負責連結的工作，這樣的工作分配是為了讓裝置管理系統能夠完整地掌握各裝置的連線情形。

而目前具備藍芽連線功能的手機，在啟動的時候必須要傳入一系列的 AT 指令，讓之後所有的輸入都會被直接傳入連線的電腦，所以必須有額外的初始化動作。以下為將藍芽手機初始化的虛擬碼(pseudo code)：

```
//開啟 Serial Port
hComm = CreateSerialPort(bluetooth_virtual_serial_port);
if(hComm != INVALID_HANDLE_VALUE)
{
    //設定該裝置多久沒回應即視為離線
    SetCommTimeouts(hComm, commTimeout);

    //設定序列埠的狀態
    BuildCommDCB("115200,N,8,1", dcbCommPort);
    SetCommState(hComm, dcbCommPort);

    //傳入 AT 指令設定手機狀態
    hcmd = "at+cmer=3,2,0,0,0\r";
    WriteFile(hComm, hcmd);
    FlushFileBuffers(hComm);Sleep(250);

    //傳入 AT 指令將手機鎖定，所有按鍵輸入都直接傳進電腦，手機不會反應
    hcmd = "at+clck=\"CS\",1\r";
    WritePort(hComm, hcmd);
    FlushFileBuffers(hComm);Sleep(250);

    //傳入 AT 指令設定字串編碼方式
    hcmd = "at+cscs=\"8859-1\"\r";
    WriteFile(hComm, hcmd);
    FlushFileBuffers(hComm);Sleep(250);

    //啟動執行緒
    ReadMobileThread = new TReadMobile(true);
    //傳入開啟完成的裝置資源
    ReadMobileThread->SetComm(hComm, &DeviceDesc);
    //訊號處理執行緒開始運作
    ReadMobileThread->Resume();
}
```

訊號處理執行緒在面對不同的裝置時就依據裝置管理系統傳入的性質描述而能用正確的方式取得訊號，再加以處理。實作上是用物件的方式記錄裝置的名稱、類別以及描述，啟動後便是將物件傳入訊號處理執行緒，以下為物件的範例：

```

Class DeviceDesc
{
    String DeviceName;
    int DeviceType;
}

Class MouseDesc extends DeviceDesc
{
    Private:
        button LB, MB, RB;
    Protected:
        event Click, DbClick;
}

Class PDADesc extends Mouse
{
    button Navi;
}

```

由於對裝置的描述是採用物件的方式來定義，目前在系統裡已經建立了一些基礎的輸入型態，即使在未來出現更多功能的輸入裝置，例如：**PDA** 不但可以輸入文字或是傳輸檔案，同時還具有模擬滑鼠的作用，這時就可以利用物件的繼承方式，很容易地擴充裝置的描述方式。而透過對裝置管理系統的擴充，或是增加額外的新裝置描述就可以讓 **VPS** 支援更多種的輸入裝置。

4.2 訊號處理子系統

訊號處理子系統是由執行緒(**thread**)構成。裝置管理系統在接受到有裝置要求連線時會啟動一個訊號處理的執行緒負責於該裝置交涉與溝通。之後訊號處理執行緒會等待是否有訊號傳入，一旦接受到來自於負責的裝置的輸入訊號，會將這些訊號以及屬於該裝置的識別碼一起送至情境管理系統進行對應的工作，將訊號轉換為動作指令。訊號處理系統所負責接收的是直接來自裝置的輸入訊號，例如：按下手機的按鍵 **0** 送出的訊號，送至情境管理系統後依照發表者的設定，被轉換為簡報下一頁的動作指令，此動作指令再送到實際進行資料運算的子系統便會被執行。

訊號處理的執行緒在實作上是透過兩種方式，一種是標準的輸出輸入裝置，如：鍵盤、滑鼠，直接利用 windows 的事件訊息(event message)就可以取得輸入；另一個方式則是要透過序列埠取得訊號。取得訊號後將訊號內容傳進情境管理系統，而為了避免多個訊號處理執行緒同時與情境管理系統溝通而相互干擾，在傳遞訊號時是用鎖定資源的方式限定一次只能有一個執行緒能夠與情境管理系統溝通，以下為處理訊息部份的虛擬碼：

```
//類型為標準輸入裝置，滑鼠或鍵盤
while(1)
{
    switch (uMsg)
    {
        //如果是鍵盤
        case WM_KEYDOWN:
            InBuffer = wParam;
            break;
        case WM_KEYUP:
            ....
        case WM_CHAR:
            ....
        //如果是滑鼠，各有左、中、右鍵
        case WM_LBUTTONDOWN:

        case WM_LBUTTONUP:

        case WM_LBUTTONDOWNBLCLK;

        case WM_MBUTTONDOWN:
            ....
        case WM_RBUTTONDOWN:
            ....
        case WM_MOUSEMOVE:
            ....
    }

    //類型為序列埠輸入裝置
    if (Terminated) return;
    //讀取裝置訊號
    ReadPortMsg(threadhComm, InBuff, 80, &dwBytesRead, NULL);
    FlushFileBuffers(threadhComm);
    //有輸入訊號
    if(dwBytesRead)
    {
        InBuff[dwBytesRead] = 0;
        //將訊號送進情境管理系統
        Synchronize(ScenarioControl);
    }
}
}
```


4.3 情境管理子系統

情境管理系統的實作本體是在控制伺服器上的一個執行檔 VPS_Master.exe，在控制伺服器負責了最重要的工作 - 管理整個發表的流程，此系統也同時肩負發送同步化訊號，而在訊號處理執行緒接受到訊號後也必須依照情境的設定將這些裝置的訊號轉換成動作指令。

VPS 啟動時，情境管理系統必須讀取及分析發表者的設定檔(present profile)以了解情境之間的關係與切換的機制(圖 4.1 右)，發表者設定檔是一個純文字檔案，目的就是要將整個發表的過程做完整的設定，在發表時就能照著這些設定切換各種情境提供多樣的互動方式或是媒材訊息。分析完成接著開始嘗試跟顯示客戶端連線，這透過一個紀錄所有要連線的顯示客戶端的位址 IP 清單的 `iplist.txt` 來完成，系統並沒有限制顯示客戶端的數量，所以是可以有彈性的設定，例如只要修改 `iplist.txt` 為單一部個人電腦，就可以在該電腦上進行多人互動，此時的模式就會跟標準的 SDG 一樣，當然如果未來 CAVE 的環境擴充為六面(十二部顯示客戶端)或甚至更多，VPS 系統也只要修改這部份就可以繼續供應多人互動的功能。由於控制伺服器的部份是在後端處理的，並不會提供給使用者操作用，所以只有一個簡單的介面(圖 4.1 左)作為偵測錯誤跟紀錄訊息之用。

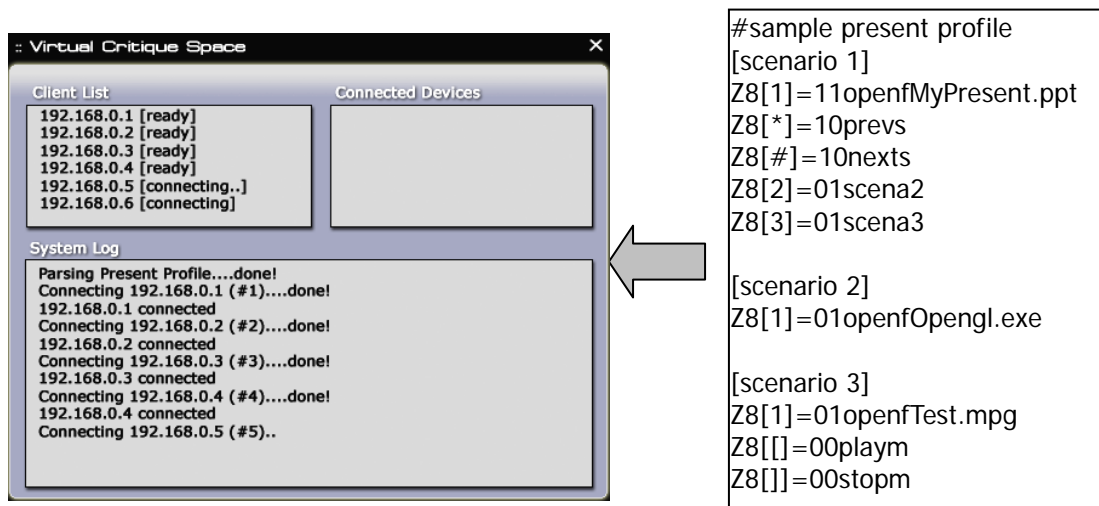


圖 4.1 VPS 控制伺服器啟動 (左)畫面; (右)發表者設定檔範例

動作指令為一種文字規範，由兩個數字、五個字母以及後面若干文字參數組成，用以傳遞執行命令，讓位於顯示客戶端的電腦能夠正確的執行命令。第一個數字代表的是控制第幾群組的顯示客戶端電腦，由於 CAVE 環境是由三面牆(六部電腦)構成，而為了立體成像，同一面投影牆的兩部電腦顯示的是相同的內容，只是有視角的差異；在一般的 CAVE 環境都是用這許多面牆共同構成一個空間，然而在進行設計發表時，往往需要同時用多種媒材呈現才能表達複雜的設計概念，因此 VPS 必須能被切割為多種媒材空間同時使用，但是在必要的時候也要可以共同構成一個完整的虛擬空間以提供真實比例的空間模擬。在這樣的考量下，所有的顯示客戶端電腦被用群組的方式來管理，1 就代表第一群組，在目前的環境裡代表的就是左邊那面投影牆的兩部電腦，0 就表示全部的電腦。第二個數字表示的是後面跟著的參數數量，然後的五個字母表示的就是要執行指令，最後跟著若干的文字就是參數內容。透過這樣的指令，發表者就可以依照自己的想法安排整個發表發表的過程，並控制整個流程(圖 4.2)。

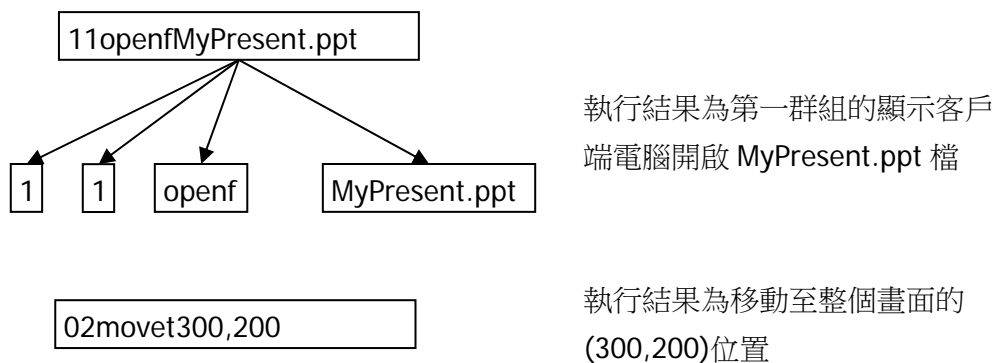


圖 4.2 動作指令

4.4 資源管理子系統

資源管理系統位在每一部顯示客戶端電腦上，依照動作指定負責載入各種不同的媒材，在載入後會把媒材的資源傳給資料運算系統，之後的互動就由資料運算系統負責。為了擴充的目的，以及發表過程中會不斷在各種媒材間切換，而每一種的媒材的操作方式差異又非常大，因此這部份必須用分離的架構，VPS 採用了動態聯結函式庫設計媒材外掛(plug-in)的方式進行實作，再用製作出來的外掛去支援各種媒材格式(圖 4.3)。

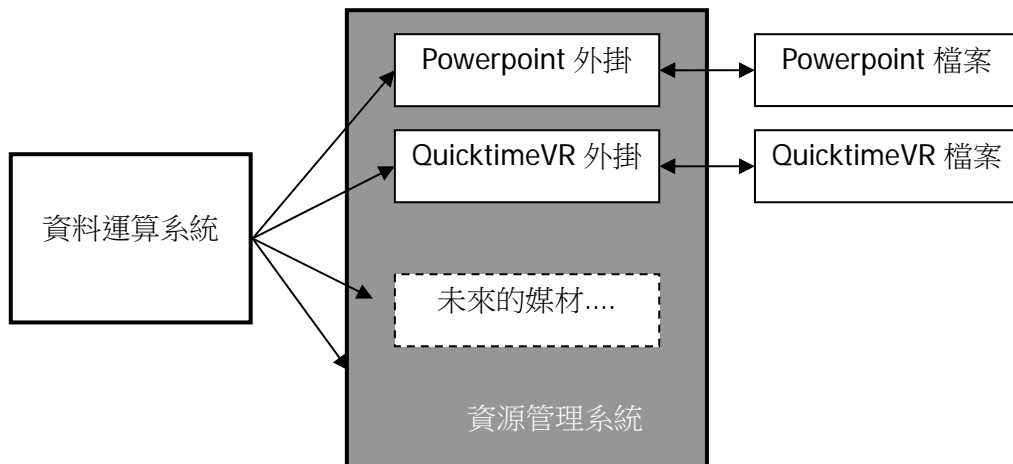


圖 4.3 採用外掛設計的資源管理系統

4.4.1 媒材外掛設計

動態連結函式庫的目的是為了讓執行檔不會佔用系統太多的資源，而且要擴充的時候也不需要修改很大的部份，只要讓執行檔去呼叫擴充過後或是新增加的 DLL 即可。在 VPS 裡，執行檔的角色就是資料運算系統，資料運算系統的工作是負責接受透過網路傳來的動作指令資料並將指令傳進各種媒材以完成互動，工作內容並不會有什麼變動，所以資料運算系統只要製作完成後就不需要再更動，往後的擴充就藉著資源管理系統來完成。

要不更動執行檔(資料運算系統)的架構，首先就是要設計一個共通的函式介面(common function interface)，只要執行檔都是透過相同的函式介面進行呼叫(function call)，就不需要擔心修改後不相容的問題，因為新的 DLL 也會提供相同的函式介面供執行檔呼叫，只是功能換成新的。依照資料運算系統的工作項目，共通函式介面需要以下幾項：

- ❖ DllInitial()：初始化函式庫，包括一些資源的配置
- ❖ DllSendCommand(char*, char*)：讓資料運算系統將命令送進媒材外掛
- ❖ DllShow(char*)：載入媒材及顯示畫面
- ❖ DllUninitial();：卸載函式庫時的清理工作

實作便是透過 DLL 的 export 方式將這些函式介面提供出去，實作如下：

```

//Powerpoint 外掛的內容
extern "C" __declspec(dllexport)int _stdcall DllInitial();
extern "C" __declspec(dllexport)int _stdcall DllSendCommand(char*, char*);
extern "C" __declspec(dllexport)int _stdcall DllShow(char*);
extern "C" __declspec(dllexport)int _stdcall DllUninitial();

int _stdcall DllSendCommand(char* dll_cmd, char* dll_param)
{
    switch (dll_cmd)
    {
        case "fopen":
            Powerpnt->filename = AnsiString(dll_param);
            Powerpnt()->Open();
            break;
        case "nexts":
            Powerpnt->NextSlide();
            break;
        case "prevs":
            Powerpnt->PrevSlide();
            break;
        case "close":
            Powerpnt->Close();
            break;
    }
    return 0;
}

```

資料運算系統只要呼叫這四個共同函式介面就可以運作，而切換不同的 DLL 時，就是操作不同的媒材，這樣的設計可以達到分離又可擴充的目的，資料運算系統的運作機制部分程式碼如下：

```

//資料運算系統的內容

//宣告的部份
int (*DllInitial)();
int (*DllSendCommand)(char*, char*);
int (*DllShow)(char*);
int (*DllUninitial)();

//DLL_NAME 可以是任何外掛
hInst = LoadLibrary(DLL_NAME);

(FARPROC &)DllInitial = GetProcAddress(hInst,"DllInitial");
(FARPROC &)DllSendCommand = GetProcAddress(hInst,"DllSendCommand");
(FARPROC &)DllShow = GetProcAddress(hInst,"DllShow");

//將訊息處理系統送來的動作指令送至媒材外掛
DllSendCommand(action_command, parameter);

```

4.4.2 支援各種媒材的外掛

VPS 支援的設計發表時常用的數位媒材格式：

- ❖ Powerpoint 簡報：開啟、翻頁
- ❖ 2D 影像：可以載入.bmp, .jpg 這兩種最常用的數位影像格式
- ❖ 3D 模型：即時的 3D 運算，可以載入.3ds、.obj 檔案格式，也可作畫面擷取
- ❖ 影片動畫：載入及播放影片，支援 .avi、.mpg、.wmv、.mp4....等
- ❖ 以影像為基礎的虛擬實境：播放 QuicktimeVR 的環場虛擬實境 (.qt)

- ❖ 執行檔：可以開啟他人製作完成之執行檔案(.exe)，例如：Director 的播放檔，並可以虛擬滑鼠或鍵盤的指令與該執行檔互動
- ❖ Flash：可以載入 Macromedia 的 Flash 檔案(.swf)，虛擬版面的實作是透過 Flash 來完成的

各種媒材的載入方式都有很大的差異，每種媒材外掛除了提供共同介面外，還必須透過各種不同的方式與技術實作才能在 VPS 這樣的共同空間下同時呈現，因此也透過外掛的設計方式實作出以下多個 DLL。

Powerpoint 簡報檔(VPS_Powerpoint.DLL)：控制 Powerpoint 檔案必須透過 OLE(Object Linking and Embedding)的機制呼叫 Powerpoint 的物件以執行使用者的命令。"OLE 是一種機制，可允許使用者建立和編輯包含由多個應用程式所建立的項目或「物件」的文件。OLE 文件，傳統上被稱為複合文件，可完全整合不同型別的資料或是元件，音效短片、試算表和點陣圖都是 OLE 文件中可見的一般元件範例。透過 OLE 的支援可以允許使用者在使用 OLE 文件時不必費心在不同應用程式之間切換，OLE 會負責切換....您可以使用容器應用程式 (container application) 建立複合文件和伺服應用程式，或是使用元件應用程式建立容器文件內的項目....OLE 所加入的許多不同概念，都是為了達成應用程式間無縫隙互動的目的" [MSDN, 2003]。OLE 的操作可以讓系統不需要重新實作讀取載入 Powerpoint 簡報檔的程式，而可以直接呼叫 Powerpoint 並嵌入作為容器應用程式的 VPS 系統內負責相關的工作。

2D 影像檔(VPS_Image.DLL)：顯示影像(.bmp, .jpg)，切換其他影像檔，也可以對影像進行放大/縮小(zoom in/out)。這部份由於 Borland C++ Builder 直接支援載入這兩類型的圖檔，透過以下函式的呼叫就完成了：

```
Image->Picture->LoadFromFile(ImageFile);
```

3D 模型檔(VPS_OpenGL.DLL)：要呈現即時的 3D 立體模型，VPS 必須能夠載入模型檔案並以即時運算(realtime rendering)的方式呈現。模型檔案格式相當多，3ds 及 obj 檔案格式則是最常用的兩種，因此 VPS 以實作這兩種為主。在載入模型檔後，由顯示輸出系統計算視差，並自動依照所在的位置分成左右眼的畫面輸出，提供立體視覺，設計者只要將模型建置好即可，不需要考慮鏡頭視差的計算。然而在設計發表的過程中，不同的評審(或其他設計者)經常會專注在模型的不同角度的畫面細節上，因此系統提供螢幕擷取(screen capture)的功能，讓每個使用者都可以將想要深入探究的畫面擷取後放置到虛擬版面的區塊以與發表者或是其他使用者討論。擷取下來的畫面會上傳到控制伺服器端的特定資料夾以供其他外掛(例：虛擬版面)存取。

動畫影片檔(VPS_Mediaplayer.DLL)：影片的部份是呼叫 Microsoft Mediaplayer 進行播放影片的功能，但是考量畫面的同步處理，因此無法支援串流型態(stream-based)的影片檔，而僅能支援以影格為基礎(frame-based)的影片，如：AVI、MPEG 等

環景影片檔(VPS_QTVR.DLL)：QuicktimeVR 是一種常用的以影像為基礎的環場虛擬實境(image-based panorama virtual reality)，製作方式是先拍攝 360 度的環景照片，透過製作軟體將兩端連接並製作成 qtvr 的格式，就可以提供使用者 360 度觀看現場環境的體驗，由於具備方向性以及鏡頭效果，可以提供擬真的現場感，製作上也相當容易只要透過照片即可，在設計發表中是很常用的媒材形式(圖 4.4)。而要讓 VPS 系統支援 QTVR 方式則是利用 Quicktime 軟體開發套件(Quicktime software developer kit)，以載入並進行瀏覽。



圖 4.4 環景照片在設計發表中常用來進行基地評估及討論

其他執行檔(VPS_EXE.DLL)：讓 VPS 可以啟動其他的執行檔(.exe)，並取得該執行檔的視窗控制資源(handle)，然後透過模擬鍵盤或滑鼠的指令讓來自使用者裝置的指令也可以跟其他已完成的執行檔互動，例如遊戲、或是利用 Macromedia Director 製作產生的播放檔。實作方式是利用 Win32 SDK 中的 FindWindow 函式找到欲控制的視窗或是用 LoadModule 函式啟動特定的執行檔，這兩個函式都可以傳回欲控制的視窗的資源，接著就可以利用 windows 的訊息系統“假造”鍵盤或滑鼠的輸入訊息，再用 SendMessage 函式送進這些被控制的視窗程式，這些視窗程式就會“以為”有滑鼠或鍵盤的輸入，透過與訊息處理系統的連結，就能讓原本只能用滑鼠或鍵盤輸入的程式與 VPS 連動，控制方式如下：

```
//啟動並取得其他執行檔的資源
int _stdcall DllInitial()
{
    hD = FindWindow(windowClass,windowName);
    if (!hD)
    {
        hD = LoadModule(Execute_FileName);
    }
    return 0;
}

//透過 Handle 及 windows 訊息系統的使用模擬鍵盤或滑鼠的訊息
int _stdcall DllSendCommand(char* dll_cmd, char* dll_param)
{
    switch (dll_cmd)
    {
        case "keyin":
            //模擬鍵盤訊息
            SendMessage(hD,WM_KEYDOWN,dll_param,1);
            break;
        case "mmove":
            //模擬滑鼠移動訊息
            SendMessage(hD,WM_MOUSEMOVE,dll_param);
            break;
        case "mlbtn":
            //模擬滑鼠左鍵按下或放開的訊息
            SendMessage(hD,WM_LBUTTONDOWN,dll_param);
            break;
    }
    return 0;
}
```

虛擬版面(VPS_Flash.DLL)：在傳統設計發表時相當重要的版面，用以同時呈現大量的媒材內容，VPS 則將原本靜態的版面設計加入了新的動態效果，讓版面不只是發表者的設計呈現，而且能夠動態反映討論的過程中使用到的各種數位媒材，甚至能夠紀錄討論後產生的變更，讓版面不再是靜態的固定內容，也讓整個設計溝通的過程中訊息的傳遞能夠更加完整。實作是透過了 Flash 作為呈現的介面，Flash 可以很快速地打造圖像介面，很適合用來快速建立動態效果，在未來則可能將這部分整合進即時 3D 的媒材中。VPS 利用 Flash 中 Action Script 的 LoadMovieNum 函式將已存放在控制伺服器上的影像載入，這些影像可能來自於(1)發表者預先放置、(2)討論的過程中使用者上傳的影像或是(3)3D 模型檔的螢幕擷圖。取出影像後也可以在畫面上移動這些影像或是對影像放大縮小，這些控制則是透過了 Flash 提供的 SetVariable 函式設定虛擬版面中各影像的座標資料(圖 4.5)。

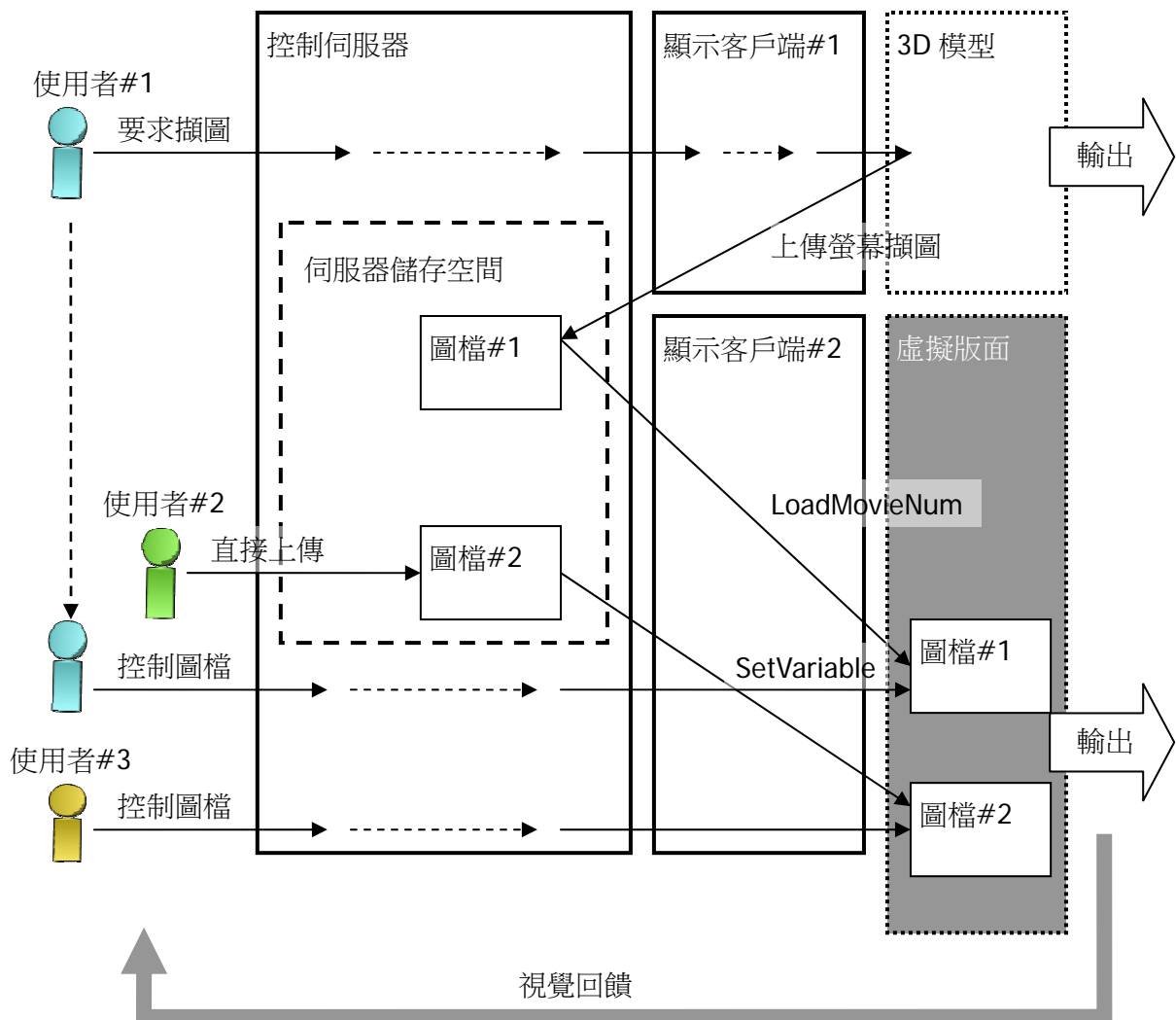


圖 4.5 虛擬版面的運作情形

4.5 資料運算子系統

資料運算系統的實作本體是位於各顯示客戶端電腦上的 `VPS_Slave.exe`，啟動後便會自動進入待命狀態，等待控制伺服器連線。連線建立後會回應成功訊息給情境管理系統，一旦需要操作特定的媒材資料，便會載入相對應的 `DLL`，在做完初始化動作後載入媒材檔案。之後任何來自訊號處理系統的訊息都會透過由資料運算系統的轉送進資源管理系統中。資料運算系統不負責實際繪製畫面的動作，但要維護目前空間的狀態以協調顯示輸出系統跟資源管理系統之間的運作，另一項更重要的工作就是接受來自控制伺服器的同步訊號並修正目前媒材內容的狀態以確保顯示輸出系統能夠顯示正確的畫面。

而在某些情況下，有可能將二到三面投影牆視為同一個空間，這時資料運算系統就會依照自己是隸屬於哪一個群組判斷物件是否已經超出自己的畫面，若超出就會通知顯示輸出系統不要繪製該物件，另外則是在合併空間後要重新指定起點(畫面左上角)座標以讓顯示輸出系統能將物件繪製在正確的位置(圖 4.6)。



空間未合併時：

兩個物件都在第二群組(中間牆)的畫面內，第三群組(右邊牆)上面則沒有任何東西或是用作其他的媒材。

第二群組的左上角座標(端點 1)為(0,0)

第三群組的左上角座標(端點 2)為(0,0)



空間合併時：

有一個物件在兩個群組之間，此時兩個群組都需要繪製該物件，以達到空間的相通。

第二群組的左上角座標(端點 1)為(0,0)

第三群組的左上角座標(端點 2)為(800,0)

圖 4.6 資料運算系統必須負責的部份：(上)空間未合併時；(下)空間合併時

4.6 顯示輸出子系統

顯示輸出系統是實際負責繪製輸出畫面的部份，依照資源管理系統跟資料運算系統提供的媒材跟資訊，將要顯示的畫面繪製出來。因為資源管理系統上的媒材外掛並不知道本身在空間中的位置為何，所以顯示輸出系統在繪製的時候便會依照目前空間的配置狀態(合併/不合併)，修正畫面中物件顯示的位置。所以 VPS 上任何的媒材在繪製出來前都會經過兩次的修正，第一次是讓媒材依照使用者的輸入產生反應(例如：改變視角、更換圖檔...等)，第二次則是依據所處的空間位置以及配置修正畫面中物件顯示的位置。

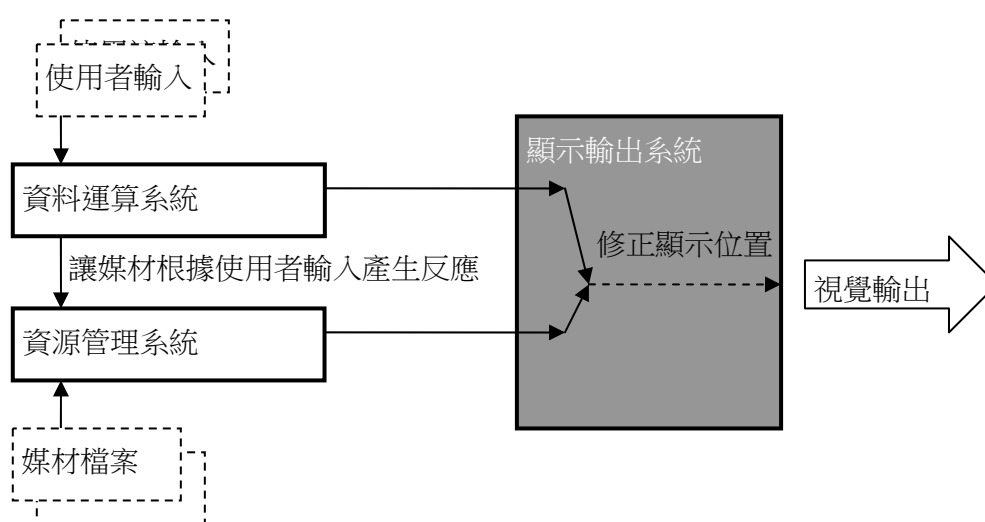


圖 4.7 媒材在繪製前會經過兩次修正

4.7 系統整合結果

經過所有的子系統實作後，各個子系統分別運作也共同整合，讓 CAVE 環境呈現一個完全不同的功能性虛擬空間。為了驗證 VPS 的實用性，在第五章會以利用 VPS 模擬一個建築設計案的發表流程作為系統評估的範例。以下為系統實際運作的畫面(圖 4.8 至圖 4.11)：

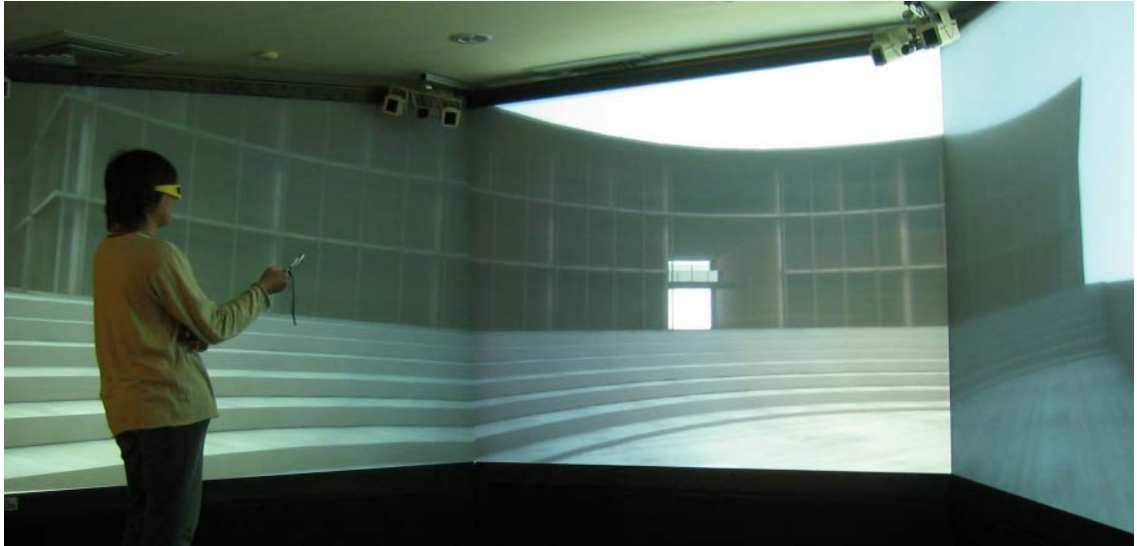


圖 4.8 啟動畫面



圖 4.9 使用三面牆播放立體動畫



圖 4.10 使用單面牆操作 Powerpoint 簡報

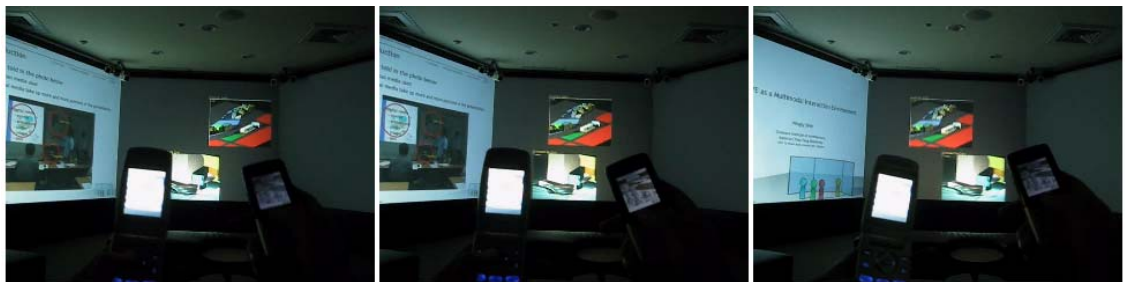


圖 4.11 多人共同操作虛擬版面，同時 Powerpoint 簡報仍可操作

第五章 系統評估

為了能夠評估將虛擬環境運用在設計發表過程中與傳統發表程序的優劣比較，本研究利用一個關於建築設計的案例，討論以舊有的媒材形式進行的發表過程，再嘗試運用 VPS 系統以相同的媒材來進行設計溝通，然後比較兩者的差異與優劣。

5.1 傳統設計發表過程

在傳統的發表過程中使用到的媒材有版面、簡報、模型以及動畫。設計者將必要的正圖置放在版面上後，將整個設計過程中用到的媒材，包括草圖，案例分析等細節都放在簡報中，簡報裡也包含了空間計畫用的圖表、法規等各種文字或表格。模型則是另一個發表必備的條件，動畫則由於是較新的媒材，尚未被廣為接受，然而也已經有一些發表場合會要求設計者必須提供動畫。

在發表的過程，首先設計者必須先將版面一一貼在牆上，然後將電腦接上投影機開始進行簡報。在簡報的過程中，設計者會不斷地以手勢指示畫面，有時不夠精確就得藉著光筆來進行位置的標示，或是走到版面的圖面旁進行解說；有時候評審提問，設計者就會翻閱簡報到特定的頁數，或是利用版面來回答。而有時候為了回應評審的問題要補充顯示一些圖，就得先將 Powerpoint 最小化，在電腦裡尋找那些圖來播放，之後再將 Powerpoint 還原，繼續剛剛簡報的內容。要播放影片的時候也要先中斷簡報的內容，播放動畫。

接著在利用版面來講述設計概念時，評審會拿起模型以各種角度觀察，或是拿至眼前近看嘗試體驗身在其中的感覺，也會將模型拿來跟版面上的圖對照。最後的講評，評審會要求將簡報翻至特定的頁面，然後進行討論。在討論的過程中，也會針對模型的某些特定細節或是就版面上的特定圖面相互交換意見，或是在版面上註記繪圖。評審也很常提到自己曾經做過的一些設計案例，提出來給設計者作為參考。

5.2 使用 VPS 進行的設計溝通過程

由案例的設計者(發表者)依自己的習慣安排流程以及各種情境的互動機制後，利用 VPS 進行一次設計的發表，參與的人有設計者以及三名參與者 A, B, C。設計者安排的设计發表過程分為以下四個情境。

5.2.1 情境一：簡報+版面+QuicktimeVR

概要：這個階段主要是由設計者進行有關設計案的簡報(圖 5.1)，並不需要特別的互動性，流程的焦點在於簡報。版面的部份則是有一些圖面的說明必須講解細部時會用到，由於版面上的圖面尺寸比牆面大，所以透過上下捲動以瀏覽完整的圖面。設計師的設定是用自己的手機操控簡報檔(圖 5.2 左)，而有時候會因為需要講解圖面的部份或是強調細部，因此除了同樣用手機上下捲動圖面外，也必須對圖面做放大及縮小的動作(圖 5.2 右)。另外設計者在進行設計規劃時有拍攝了基地的三百六十度環景照片，也將這些照片製作成了 QuicktimeVR 的格式，QuicktimeVR 的使用是為了增進對基地環境的理解，為了說明基地的環境的潛力面與限制，設計者必須轉動場景並加以解說。在講解過設計的計畫後，設計者按下手機的按鈕進入下個情境。

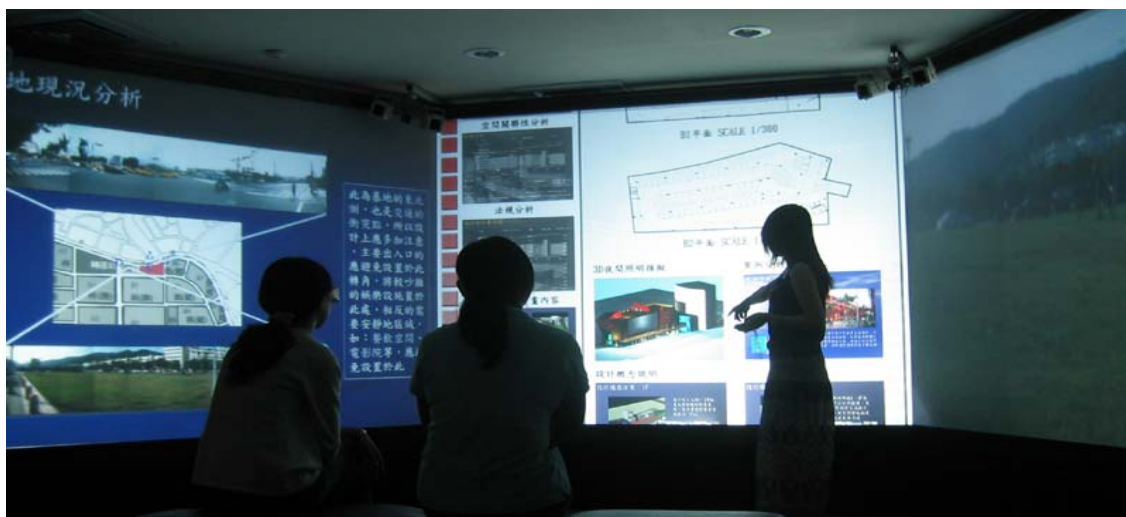


圖 5.1 情境一：簡報+版面+QuicktimeVR

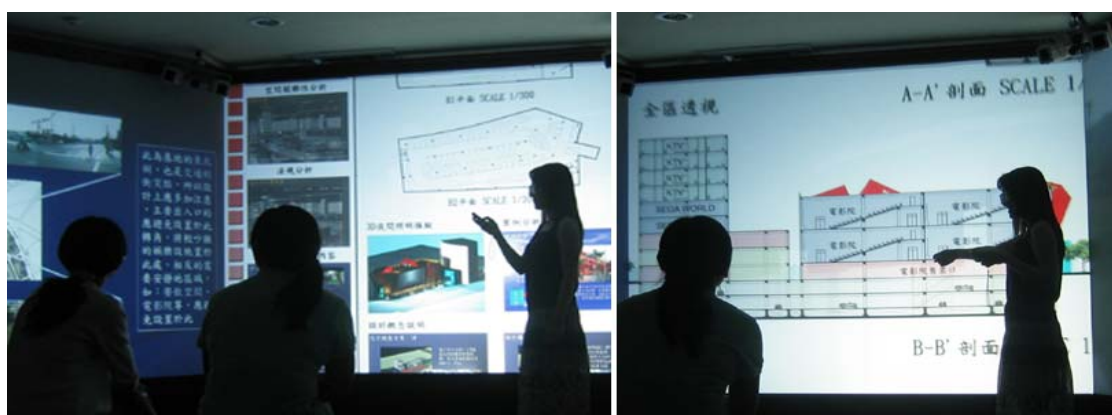


圖 5.2 使用手機操作簡報(左) 將圖面放大講解(右)

討論：VPS 的硬體環境所使用的背投式投影方式，讓設計者可以站在自己的簡報前使用肢體手勢進行指示或說明，不會被投影畫面打在臉上干擾閱讀，也不需要透過辨識力較差的光筆進行指示。而可以放大縮小以及捲動的版面，透過大尺寸的顯示，可以讓設計者將圖面的細部放大進行說明，使用者一致認為能夠將版面放大縮小對於理解以及講解都有相當正面的幫助。設計者認為同時顯示簡報及版面更是在設計發表時相當必要的，因此雖然在電腦上也可以進行圖檔的放大縮小，但是限於一次只能有一部電腦進行投影的狀況，操作圖檔時就無法繼續顯示原本簡報的內容，有時會造成參與者因為觀看圖檔而遺忘了不久前進行的內容。QuicktimeVR 的部份因為內容僅是基地的環場影像，沒有引起太大的反應，然而設計者則認為有些設計者就會偏好運用 QuicktimeVR 來進行基地的描述。

5.2.2 情境二：簡報+版面+模型

概要：在透過簡報講解設計的基本計劃後，就進入設計本身的說明。在這個階段，主要的焦點在於版面及模型(圖 5.3)，設計者會在版面上配置平面、立面、剖面及透視圖等圖面，並輔以文字說明設計的概念，並要能夠隨時放大縮小。簡報則用來輔助版面，做補充的說明。模型則是另一個重點，透過各個角度的觀察，能更精確而完整地感受設計中空間與實體間的關係，而設計者認為自己要講的內容都已做成圖面放置在版面上(圖 5.4 左)，所以不需要操作 3D 模型，模型的旋轉移動縮放等動作都是開放給參與者自行操作(圖 5.4 右)。



圖 5.3 簡報+版面+數位模型



圖 5.4 利用圖面說明(左) 參與者操作模型詢問設計者問題(右)

討論：在一般發表的狀況下，評審或其他參與者常會希望能更動模型中的某些部份以進行討論，然而實體模型這麼做的話會破壞原本的模型，設計者認為如果是數位模型的話，讓其他參與者可以更動其中的元件是必須的，VPS 僅提供視角的旋轉變換，但並沒有提供修改模型的功能，這部份是相當重要的。參與者則認為視覺上數位模型的呈現已經相當接近實體模型，然而操作的不直覺是一項缺點；因為 VPS 支援的裝置都是平面的輸入裝置，用平面的裝置來操作立體模型還是讓使用者覺得不夠方便也不太能適應。而可以讓其他人自由操作這點則讓使用者覺得很有用，在以往的發表場合，有時候因為必要而須開啟模型檔案，但是現在的 3D 造模軟體(如：3DS MAX、Alias MAYA 等)啟動的時間都相當久，常因此讓評審感到不耐，而在開啟的過程，也無法繼續進行簡報，只能單純地等待軟體啟動完成。啟動完成後，也只能透過設計者操作模型，經常因為操作上的問題花掉許多時間。立體的顯示以及控制的介面雖然能夠讓使用者自行操作虛擬模型如同在轉動實體模型一樣的效果，但如果能夠再加上對於模型的修改以及直覺的操作輸入方式，就能更加發揮數位模型的優點。在版面部份，解析度是最受到質疑的部份，因為在傳統的發表過程中所使用的版面是相當精細的，雖然 VPS 提供了更大尺寸的顯示，但是並沒有更精細的畫面表現，這部份讓使用者覺得畫面不夠清晰，是一個缺點。

5.2.3 情境三：動畫+動畫+動畫

概要：設計者本身習慣以動畫作為整個設計發表的結尾，所以在講完設計的概念後，開始播放具沉浸式效果的立體動畫。這個階段只有設計者操作播放與暫停，其他使用者無法進行輸入(圖 5.5)。

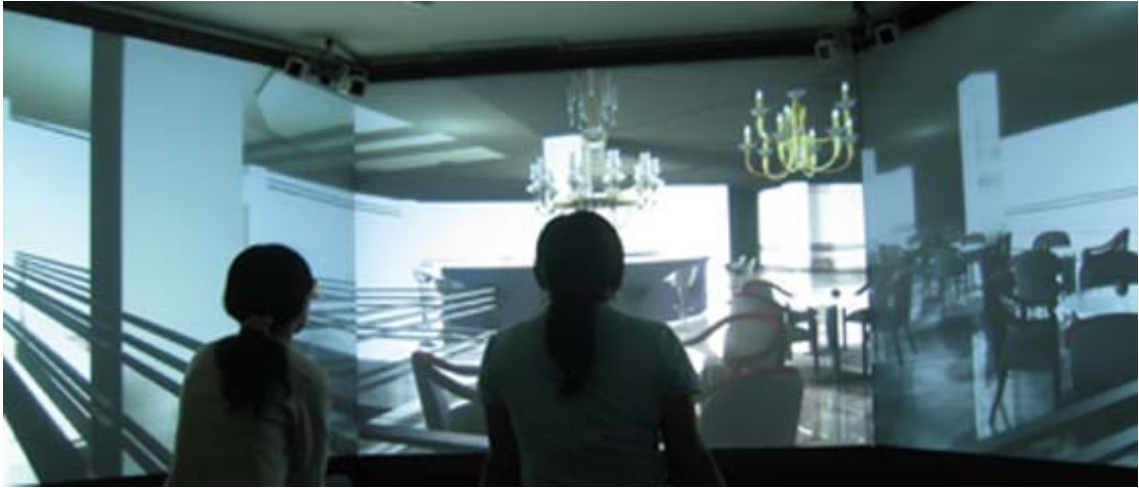


圖 5.5 播放沉浸式立體動畫

討論：相較於一般的數位媒材或是傳統媒材，沉浸式效果的立體動畫是 CAVE 這樣的虛擬環境最大的優勢，透過大尺寸的顯示，以及精細的模型、貼圖及正確的光影效果，讓觀者產生置身其中的空間感受，而在以往要操作這樣的虛擬環境必須在背後進行許多的設定，相當繁雜。現在則可以在整個發表過程中的任何一個階段隨時安排展示，也可以隨時切換，這樣的方便性會讓使用者較願意利用 VPS 作為設計發表的環境。

5.2.4 情境四：動態討論

概要：動畫播放完畢後進入討論與講評的階段，設計者選擇左邊及中間牆面是呈現自己設計的靜態版面，右邊是數位場景模型(圖 5.6)。隨著時間的進行，參與者也開始提供多種意見，中間的牆面也轉換成虛擬版面。參與者 A 透過操作數位模型，就特定的細節與設計者討論，參與者 B 則在這個時候將手上 PDA 裡的圖片上傳進伺服器，在之後顯示在畫面上參與討論(圖 5.7)，參與者 C 則是在觀看左面牆的版面。經過了一陣子的討論後，設計發表結束，設計者選擇將中間的虛擬版面畫面記錄下來，以供日後參考。



圖 5.6 利用中間的虛擬版面進行討論

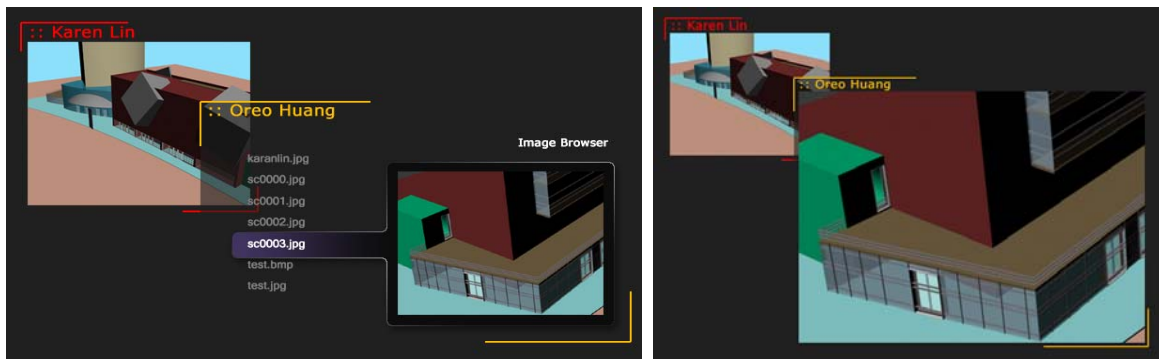


圖 5.7 選取控制伺服器內儲存圖檔的介面(左) 顯示圖檔(右)

討論：虛擬版面是所有使用者都覺得很有意思的部份，因為他們可以把以往只能口述的東西用圖面顯示出來，減少了認知上的模糊，能更清楚表達意見，但是使用者普遍希望能夠在版面上加註記或是繪圖，在溝通上會有更大的幫助。設計者也提及在以往的發表經驗中，評審會希望能夠將他們自己的一些案例拿來作為例子，所以都會帶著一些書籍來參加發表，以便在提及這些內容的時候能夠展示給設計者參考，雖然他們有時候也會希望播放存放在自己電腦裡的內容，但常常因為還要轉接投影機或是用隨身碟傳輸檔案開啟檔案等等不便太浪費時間而作罷。使用者認為透過虛擬版面的利用，可以讓未來在溝通時能夠動態地提供更多的媒材刺激彼此的思考。設計者則認為把討論結果記錄下來對於之後的檢討會相當有幫助。而 VPS 提供的 3D 場景模型的擷圖功能雖然有用到，但是使用者認為擷取下來的圖必須能夠在上面加以註記才有意義，否則就只要能夠自由操作角度觀看以及討論即可，不必然需要將畫面擷取下來。

5.3 比較與討論

相較於傳統的發表流程，再綜合以上的討論，使用 VPS 這樣的虛擬環境進行設計溝通會有的優點：

- ❖ 製作快速：有許多的設計者的設計過程現在都是利用繪圖軟體的輔助先畫好平面圖，然後利用造模軟體將平面延展為立體並進行造型以及細部規劃等設計，之後所有的正圖圖面都可以藉由角度切換的方式產生，所以數位模型是在設計過程中自然會有的產物，直接運用這些模型進行展示與體驗，讓設計者不需要花費額外的時間製作其他的媒材。
- ❖ 沉浸式體驗：相較於縮小化的實體模型以及範圍有限的影像，CAVE 提供了足以度量真實空間感受的體驗，使用者不需要再眯著一隻眼睛近看模型想像自己身在其中的感受，而能夠真正的感受設計中的空間。
- ❖ 雙向的媒材訊息：在以往的發表空間中，設計者與評審之間意見的交換只有語言是雙向的，媒材扮演的角色只有單向的表達設計者的概念，而沒有提供來自評審的內容，這也使得評審表達的方式幾乎就只有語言，相較於設計者能夠使用的複合媒材，也讓評審的想法較難以被完整傳達。
- ❖ 可攜式媒材：在 VPS 中，所有的媒材都是可以動態地上傳下載的，這也讓媒材資訊的交換更加的頻繁。使用者不再需要受限於電腦裡準備好的資料，而能夠與他人快速地交換內容，增進溝通。
- ❖ 動態內容：在討論的過程，內容能夠動態地視覺化呈現，而不會只能藉由口述進行溝通。

然而，VPS 也有著一些尚未克服的缺點：

- ❖ 實體空間的限制：傳統發表空間可以容納各種媒材，包括實體與虛擬，例如設計者可以攜帶模型也可以使用電腦；然而在 VPS 的實體空間中，由於需要使用投影機，環境必須是無光源的狀態，這使得實體的媒材難以與虛擬的媒材同時展示。
- ❖ 介面的混淆：使用手機或 PDA 等裝置雖然帶來方便，但是由於這些裝置也有螢幕，使用者在使用的時候會因為習慣性看著裝置進行輸入(實際上的視覺回饋只在顯示牆上)，而看著顯示牆操作裝置又跟這些裝置平時的使用方式相異，造成使用上的混淆。例如：使用者操作手機時因為沒有看著手機操作，容易誤觸其他按鈕。

- ❖ 情境控制設定不易：目前 VPS 的情境控制檔是由設計者以文字的方式輸入，不直覺而難以操作，應該要提供視覺化的介面協助設計者安排內容，而且能讓設計者自己設定後，直接將整個內容與設定檔上傳至控制伺服器，伺服器便會自動分配各個媒材至適當的顯示客戶端電腦。
- ❖ 介面不便：在平面與空間之間的輸入介面始終是 CAVE 上的一個重要研究議題，即使使用個人化的隨身裝置有其優點在，但是平面的輸入介面也讓在立體虛擬空間中的操作有其不足的地方。
- ❖ 解析度太低：發表時用到的圖面都必須非常精確而細緻以避免遺漏細節，在現有的硬體限制下，VPS 目前的解析度僅有 3072x768，無法與版面那樣的精細度相比，但是隨著硬體的進展，這部分在未來是必然不構成問題的。

第六章 結論

本研究探討了在一般發表空間中常使用的數位媒材，以及因為使用這些數位媒材使得參與者無法如同與實體媒材般直接與媒材互動，造成的設計者與參與者之間溝通上的問題。為了解決這樣的問題所提出來的系統 - 虛擬發表空間 VPS，在規劃、實作以及評估後，可以看到這樣的問題在透過許多技術與媒材的整合後是可以被初步解決的。

VPS 除了讓發表空間中的所有人都能擁有與媒材互動的能力外，也嘗試讓設計溝通的過程增添了新的元素，如：訊息的分享、動態的內容呈現、媒材與媒材間的互動/轉換等，希望讓溝通的過程能夠更豐富更快速更有效。透過整合 CAVE 與 SDG，數位媒材能以最豐富的方式呈現，使用者能以最直接的方式操作數位媒材，也能以最直覺的方式與同處一個空間下的其他人溝通。

6.1 研究貢獻

任何單一的媒材都有其表達上的限制，無法用來傳達複雜的設計概念，設計者長久以來都希望同時運用多種媒材以完整地傳達自己的設計概念，從最早的草圖、平面，然後加入了透視圖、模型，隨著越來越多媒材的使用，設計溝通的過程也因此更加豐富，也越來越能夠表達設計的想法。在數位媒材加入後，更是被大量地運用在設計溝通過程，從 Powerpoint 簡報、數位影像、虛擬模型，到近來動畫也逐漸被採納運用。而從未停下腳步的科技進展，更是讓新的媒材逐漸的成形，帶來了包括人機互動(human computer interaction/interface, HCI/I)、建築輔助設計與製造(CAD/CAM)、虛擬空間與網際網路等新媒材，可以預見的，是這些新媒材也將在未來成為設計溝通過程中重要的媒介。

本研究提出並實作了一個能善用數位媒材優勢，同時也提供一種新的人機互動形態的設計溝通平台 - 虛擬發表空間 VPS。透過這個平台，讓設計的溝通方式能更多樣化，原本只能透過語言傳達訊息的其他參與者也能夠透過媒材的使用表達意見，另外也可以初窺在未來新媒材將會以什麼樣的方式進入設計的溝通過程中以及設計將會以什麼樣的方式被呈現出來。

6.2 研究限制與後續研究

虛擬發表空間做了新的嘗試，但目前仍有許多的軟硬體限制，讓這個系統還無法成為一個理想的設計溝通環境。

藍芽的通訊技術仍是一種發展中的技術，雖然在目前已有越來越多裝置都具備了藍芽通訊的能力，例如：手機、PDA、電腦、耳機、視聽設備...等，也讓在本系統上人機互動的形態會有更多樣變化的可能性，但是傳輸速度與訊號干擾等技術問題仍然是個困擾，在未來硬體的改良後這部份的限制應該會有很大的進展。

VPS 在設計之初即是以平面的概念進行設計，所以介面的設計是平面的，然而媒材中有使用到 3D 的立體模型；而讓使用者使用隨身的個人設備進行互動也是基於可以便於取得以及容易交換個人資料，這些個人設備卻也都是平面的輸入介面。這樣介面上的差異造成了操作立體模型時的不直覺，也形成了本系統的一大限制。未來 VPS 該加入更直覺的輸入方式，如：手勢、電腦視覺等，讓整個輸入的介面更為易用，而裝置的連線能力則仍能作為交換資料之用。

VPS 雖然對於設計溝通的過程進行了初步的研究與探討並且實作了一個虛擬的環境供使用者與數位媒材互動，但是距離真正完成提供一個發表空間的環境還有許多後續的研究，包括：發表過程中所有參與者角色的關係(例如：評審與設計發表者之間的上下關係)、參與者角色差異對彼此溝通的形式造成的影響等，經由這些研究才能讓 VPS 在設計溝通的過程中提供必要的環境與互動機制，以真正達到虛擬發表空間的功能性。

本研究的成果成功地在 CAVE 建置了一個多人能夠同時進行互動的環境，在目前雖然僅作為設計溝通使用，然而隨著介面的改善以及電腦運算能力的提升，未來這樣多人互動的機制將可以成為協同設計的基礎平台。有別於現在的兩種協同設計型態：(1)面對面溝通後分別用各自的電腦做設計，媒材與媒材的溝通不直接；(2)人與人透過網路在同一個平台上做設計，人與人的溝通不直接。在未來，使用者不僅僅將在這樣的多人平台上進行設計，同時相互面對面溝通，媒材與資訊也相互流通，再透過發展改良後直覺的介面進行操作，人與人、媒材與媒材、人與媒材間，都將會具備最清楚最單純的溝通方式。

參考文獻

Achten, H.H. "Requirements for Collaborative Design in Architecture", Timmermans, H. (ed), Proceedings of the 6th Design & Decision Support Systems in Architecture & Urban Planning Conference, pp. 1-13, Ellecom, 2002.

Bier, E. and Freeman, S. "MMM: A user interface architecture for shared editors on a single screen", Proceedings of 4th annual ACM Symposium on User Interface Software and Technology, pp. 79-86, ACM Press, 1991.

Bowman, D.A., Kruijff, E., LaViola Jr. J.J. and Poupyrev I. "An Introduction to 3-D User Interface Design", Presence, Vol. 10, No. 1, pp. 96-108, 2001

Butz, A., Hollerer, T., Feiner, S., MacIntyre, B., and Beshers, C. "Enveloping Users and Computers in a Collaborative 3D Augmented Reality", International Workshop on Augmented Reality, San Francisco, CA, 1999.

Cera, C.D, Regli, W.C., Braude, I., Shapirstein, Y. and Foster, C.V. "A Collaborative 3D Environment for Authoring Design Semantics", IEEE Computer Graphics and Applications, pp. 43-55, 2002.

Cohen, P.R., McGee, D.R., Oviatt, S.L., Wu, L., Clow, J., King, R., Julier, S. and Rosenblum, L. "Multimodal interactions for 2d and 3d environments", IEEE Computer Graphics and Applications, pp. 10 - 13, 1999.

Cruz-Neira, C., Sandin, D.J., and DeFanti, T.A. "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE", Proceedings of SIGGRAPH'93 Conference, pp. 135-142, ACM SIGGRAPH, 1993.

Cruz-Neira, C. "Computational Humanities: the New Challenge for Virtual Reality", IEEE Computer Graphics and Applications, pp. 10-13, 2003.

Dertouzos, M.L. What Will Be, HarperCollins Publishers Inc, 1997.

Dertouzos, M.L. The Unfinished Revolution: Human-centered computers and what they can do for us, HarperCollins Publishers Inc, 2001.

Endo, M., Yasuda, T. and Yokoi, S. "A Distributed Multiuser Virtual Space System", IEEE Computer Graphics and Applications, pp. 50-57, 2003

Gross, M. and E. Do. "Integrating Digital Media in Design Studio: Six Paradigms", Proceedings of the ACSA (American Collegiate Schools of Architecture) National Conference'99, 1999.

Ilmonen T. and Kontkanen, J. "Software Architecture for Multimodal User Input – FLUID", Proceedings of the 7th ERCIM Workshop, pp. 223-242, Chantilly, France, 2002

Inkpen, K., McGrenere, J., Booth, K., and Klawe, M. "The effect of turn-taking protocols on children's learning in mouse-driven collaborative environments", Proceedings of Graphics Interface (GI'97), pp. 138-145, Morgan Kaufmann Publishers, 1997.

Ishii, H. and Ullmer, B. "Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms", Proceedings of Human Factors in Computing Systems (CHI'97), pp. 234-241, ACM Press, 1997.

Ishii, H., Underkoffler, J., Chak, D., Piper, B., Ben-Joseph, E., Yeung, L. and Kanji, Z. "Augmented Urban Planning Workbench: Overlaying Drawings, Physical Models and Digital Simulation", Proceedings of Conference on IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '02), 2002.

Klemmer, S., Newman, M., Farrell, R., Meza, R., Landay, J. A. "A Tangible Evolution: System Architecture and Participatory Design Studies of the Designers' Outpost", Proceedings of the 14th annual ACM symposium on User interface software and technology (UIST 2001), 2001.

Kontanen, J. "Architecture for High Level Processing of Multimodal User Input", Helsinki University of Technology, Master Thesis, 2002.

Laurel, B. Computers as theatre, Addison-Wesley Publishing Company, 1993.

Li, K., H. Chen, Y. Chen, D.W. Clark, P. Cook, S. Damianakis, G. Essl, A. Finkelstein, T. Funkhouser, T. Housel, A. Klein, Z. Liu, E. Praun, R. Samanta, B. Shedd, J.P. Singh, G. Tzanetakis, and J. Zheng. "Building and Using A Scalable Display Wall System", IEEE Computer Graphics and Applications, pp. 29-37, 2000.

Liu, Y.T., Tang, S.K. "Space, Place and Digital Media: Towards a Better Simulation of a City that has now Disappeared", International Journal of Architectural Computing, Volume 1, Issue 1, pp. 112-129, 2003.

Martin, J. "TYCOON: Theoretical Framework and Software Tools for Multimodal Interfaces", Intelligence and Multimodality in Multimedia interfaces. (ed.) John Lee, AAAI Press, 1998.

Microsoft Developer Network(MSDN) <http://msdn.microsoft.com>

Myers, B.A., Stiel, H., and Gargiulo R. "Collaboration Using Multiple PDAs Connected to a PC", Proceedings CSCW'98: ACM Conference on Computer-Supported Cooperative Work, pp. 285-294, Seattle, WA, November 14-18, 1998.

Myers, B.A., Lie, K.P. and Yang, B.C. "Two-Handed Input Using a PDA And a Mouse, Human Factors in Computing Systems", Proceedings CHI'2000, pp. 41-48, Hague, The Netherlands, Apr. 1-6, 2000.

Myers, B.A. "Using Hand-Held Devices and PCs Together", Communications of the ACM, Volume 44, Issue 11, pp.34 – 41, ACM Press, 2001.

Nielsen, J. Usability Engineering, Academic Press Inc, 1993.

Norman, D.A. The Invisible Computer : Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution, 1998.

Pedersen, E. R., McCall, K., Moran, T. P., and Halasz, F. G. "Tivoli: An Electronic Whiteboard for Informal Workgroup Meetings", Proceedings of Human Factors in Computing Systems (InterCHI 93), pp. 391-398, ACM Press, 1993.

Rekimoto, J. "A Multiple Device Approach for Supporting Whiteboard-Based Interactions", Proceedings of Human Factors in Computing Systems (CHI'98), pp. 344-351, ACM Press, 1998.

Rekimoto, J. and M. Saitoh. "Augmented Surfaces: A Spatial Continuous Workspace for Hybrid Computing Environments", Proceedings of CHI'99 Conference on Human Factors in Computing Systems, pp. 378-385, ACM Press, 1999.

Robertson, C.S., Wharton, C. Ashworth, and M. Franzke. "Dual device user interface design: PDAs and interactive television", Proceedings of CHI'96, pp. 79 – 86. ACM Press, 1996.

Shen, C., K. Everitt and K. Ryall. "UbiTable: Impromptu Face-to-Face Collaboration on Horizontal Interactive Surfaces", Proceedings of the Fifth International Conference on Ubiquitous Computing, pp. 281-288, 2003.

Stewart, J., Bederson, B.B., and Druin, A. "Single display groupware: A model for co-present collaboration", Proceeding of the CHI'99 Conference on Human Factors in Computing Systems: the CHI is the limit, pp. 286-293, 1999.

Stewart, J., Raybourn, E.M., Bederson, B., and Druin, A. "When two hands are better than one: Enhancing collaboration using single display groupware", Proceedings of the CHI'98 Conference on Human Factors in Computing Systems, pp. 287-288, 1998.

Tamura, H., Yamamoto, H., Katayama, A. "Mixed Reality:Future Dreams Seen at the Border between Real and Virtual Worlds", IEEE Computer Graphics and Applications, pp. 64-70, 2001

Tang, J. C. "Listing, Drawing and Gesturing in Design: a Study on the use of Shared Workspaces by Design Teams", Stanford Univeisity, PhD Dissertation, 1989.

Tse, E. and Greenberg, S. "SDGToolkit: A Toolkit for Rapidly Prototyping Single Display Groupware", Poster in ACM CSCW '2002 Conference on Computer Supported Cooperative Work, 2002.

Vries, B. de, Achten, H.H., and Orzechowski, M. "The Tangible Interface: Experiments as an Integral Part of a Research Strategy", International Journal of Architectural Computing, Volume 1, Issue 2, pp. 134-152, 2003.

Welch, G., Fuchs, H., Raskar, R., Towles, H. and Brown, M.S. "Projected Imagery in Your "Office of the Future" ", IEEE Computer Graphics and Applications, pp. 62-67, 2000.

Zanella, A. and Greenberg, S. "Reducing Interference in Single Display Groupware through Transparency", Proceedings of the Sixth European Conference on Computer Supported Cooperative Work(ECSCW 2001), pp. 16-20, Bonn, Germany, 2001.