

# 行政院國家科學委員會專題研究計畫 成果報告

## 具察覺情境功能之行動運算中介軟體設計

計畫類別：個別型計畫

計畫編號：NSC93-2213-E-032-026-

執行期間：93年08月01日至94年07月31日

執行單位：淡江大學資訊管理學系

計畫主持人：周清江

計畫參與人員：何承澤、曾志傑、王維君

報告類型：精簡報告

報告附件：出席國際會議研究心得報告及發表論文

處理方式：本計畫可公開查詢

中 華 民 國 94 年 10 月 31 日

# 具察覺情境功能之行動運算中介軟體設計

**關鍵字：**中介軟體、覺察情境、行動運算、內容自動調整、CC/PP、RSS、DOM  
**Keywords:** middleware, context aware, mobile computing, content adaptation, CC/PP, RSS, DOM

## 一、緒論

隨著晶片製程與無線網路軟硬體之快速發展，通訊科技涵蓋範圍益趨完備，個人運算輔助工具從個人電腦轉向小型、可隨身攜帶的行動運算裝置(Mobile Device)。

行動運算這些相關軟硬體的革新帶來了更小、運算能力更強大的行動裝置；越來越多人攜帶行動裝置來幫助自己處理日常生活工作，像是智慧型手機(Smart Phone)、個人數位助理(PDA, Personal Digital Assistant)與穿戴式電腦(Wearable Computer)等。其次，無線傳輸技術的快速發展：藍芽、紅外線及隨處可見 WiFi 存取點(AP, Access Point)，使得人們不必侷限在家中，使用笨重的桌上型電腦，隨時隨地能與廣大的網際網路、甚至是所處環境中其他的運算單元做互動、交換訊息。這種運算模式的全面革新帶動了計算機演進的下一個里程碑：「普及運算」(Pervasive Computing)[1]；運算裝置間的訊息傳遞方式，也由傳統的集權、主從，轉向分權、協同的合作模式。

普及運算中很重要的一個研究重點是覺察情境(Context Aware)的功能，也就是利用各種行動運算裝置對於使用者週遭環境的偵測，提供使用者所在的情境狀態的感知、並且能相互交換訊息，幫助使用者自動完成許多工作，提供使用者一個「最小使用者干擾」(Minimal User Distraction)[2]、智慧型的運算環境。以下對於察覺情境設計工具之需求做一探討：

1. 互相操作性(interoperability)：在這麼多各式各樣的設備、服務發現和互動技術裡，開發人員如何處理各種不同的設備表達能力與互動的模型？顯然地，建立一個能與各種不同的設備和服務互動的應用程式，對開發人員是一個相當大的挑戰。因此，從開發人員的角度來看，相當需要統一的資料表示方法與各種設備和服務互動。
2. 可擴充性(Scalability)：如果基礎架構都是以分散式之點對點(peer to peer)儲存與通訊，當環境裡的使用者、行動設備與服務愈來愈多的時候，網路上到處都將塞滿著服務發佈的資料，對於各種動態服務發現和設備互動會

造成相當大的負荷。因此需要一個中介軟體架構負責溝通協調，以應付大量資料交換之所需。

3. 運算能力(Computing Capabilities)：爲了使行動設備變得更小、更輕和電池續電力更長，也就是說行動設備運算能力與使用者希望之功能間必須要做一妥協。然而，面對與日俱增的運算和資料處理能力，如何動態決定一個平衡點是相當困難的。如何利用察覺情境工具，以符合大量運算與感應的需求，也是一個相當大的難題。

本研究的主要目的，在實作一個情境感知的系統，利用行動裝置對於使用者所在情境的覺察[8]，以中介軟體的方式提供相關的服務。我們並以「博物館導覽」爲應用的劇本，以參觀博物館的遊客的角度找出相對應的情境以及可能的應用並且實作相關功能。我們將情境分爲三個大項：首先以「人」爲角度，找出同在博物館參觀的遊客，並且定義其可以和使用者互動的項目；其次以「展覽品」爲角度，列出展覽品相關的說明及推薦機制，最後以「資訊」爲角度，我們提供一個單一版本的機制(Re-authoring-Free Mechanism)讓博物館管理人員只需提供單一版本的展覽品網頁內容服務(Web Content Service)就可以滿足博物館遊客各式各樣的行動運算裝置，並保證可以將適合於情境之內容顯示在裝置上，毋須針對裝置特性再特別撰寫多種版本的網頁。最後我們以實作驗證本系統的可行性，並指出相關的研究技術。

## 二、相關研究

實作一個具備情境感知(Context-Awareness)功能的系統，首先需要對於情境有一明確的定義，我們將再第三章有一詳細介紹。目前對於「情境」依據應用的環境有些許的差異，我們列舉較具代表性的相關研究如下：

Dey[3]發展了一套察覺情境設計工具(Context Toolkit)。該工具提供如情境精靈(Context Widget)、情境匯集器(Context Aggregator)、情境解析器(Context Interpreter)及情境發現器(Context Discoverer)等基礎類別，讓開發人員可以利用這些類別衍生出特定應用程式，輕易地開發一個察覺情境應用程式的雛型[5]。然而，在大量使用這類型的設計工具後發現，預期在未來的普及運算環境裡，將有愈來愈多的特定化之服務與互動技術同時存在，會在統一性(unification)、互相操作(interoperability)和一致性(consistency)上引發不少問題[4]。

利用察覺情境設計工具所引發的問題，最根本是受限於行動設備本身的能力。因此，Hong 和 Landay[6]提出基礎架構的方式(infrastructure approach)來解決：讓服務可被任何行動設備、任何應用程式取用，以取得與處理情境資訊。爲了在

普及運算環境的基礎架構順利執行，中介軟體必需有能力找到、調整和傳達適合的資訊至使用者的行動設備上[7]。

除了上述察覺情境工具需考慮的議題外，Saba 等人也討論建置行動運算中介軟體基礎架構引發的其他議題[1]：

1. 整合性(Integration)：雖然普及運算元件已經在許多的環境中使用，然而要整合這些元件到一個單一的平台裡，仍然是一個不容易解決的課題。當各種行動設備與應用程式增加時，整合的問題變得更加複雜。例如，當許多的使用者連線擁進時，如何動態的分配頻寬以求負載平衡。
2. 隱形性(Invisibility)：爲了不讓使用者感受到大量干擾，一個智慧型的環境和在這環境中的各個物件，都必須能夠自行調整以符合使用者的需求。例如，當無線網路環境由不穩或斷線回復正常時，察覺情境應用程式必須能自動設定網路連線。
3. 情境資訊管理(Context Management)：基礎架構必須要輔助蒐集、建構、維護與使用者有關的情境資訊，包含：行動設備的運算能力、使用者喜好、目前的活動和進行中的運算連線(computing sessions)[9]。當偵測到使用者所需的情境資訊時，基礎架構要能建立情境資訊的表達方式以供轉換格式或修改回應，配合使用者行動設備的運算能力，提供所需要的資訊內容。

### 三、情境的分類

情境是對於使用者目前所處環境的一個具體且可以讓機器解析的描述，感知器蒐集情境訊息之後可以將結果傳送給行動運算裝置做進一步的處理。不同的應用的功能、所處的狀態等對於所需情境的種類都不相同。在此，我們以系統中存在的物件間的從屬關係分類情境，並以此設計相關應用：

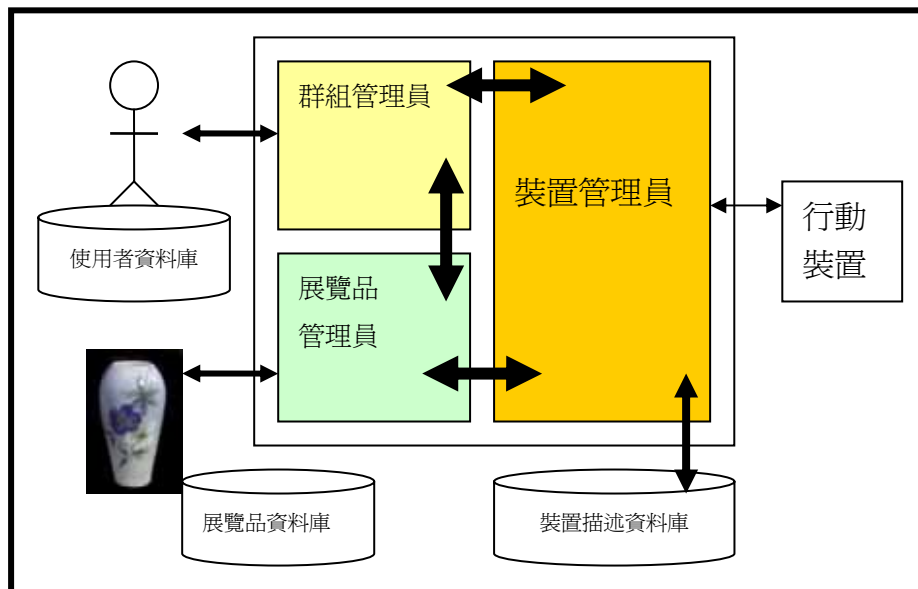
1. 使用者情境 – 使用者本身的狀態，例如使用者所在的位置或喜好等與使用者相關的情境；
2. 裝置情境 – 裝置的處理能力、顯示能力等與裝置相關的情境；
3. 環境情境 – 所處環境的狀態，通常爲系統中某部分的使用者或裝置所共有的特性，例如：溫度、溼度、高度等，也可以用來代表相關應用之內容。

在本系統中，我們將依據這三種情境，以一個博物館導覽系統爲範例，分別舉出三種在瀏覽博物館時可能會考慮的情境項目，並加以處理，提供使用者相關服務，實現普及運算所提供給我們的願景，我們所會用到的三個項目如下：

1. 使自動調整用者與其他使用者的互動
2. 行動裝置的顯示能力
3. 使用者所可以取得的展覽品內容相關資訊

#### 四、系統架構

依據上述情境，我們定義了一個情境整合管理員(Context Coordination Manager)處理所有的情境，並且依照情境的種類分為群組情境管理員(GCM, Group Context Manager)、展覽品情境管理員(ECM, Exhibition Context Manager)以及裝置情境管理員(DCM, Device Context Manager)三個模組，系統架構如圖一。



圖一、三個情境管理元件與外部環境的互動關係

群組情境管理員模組的功能在管理使用者與其群組間相對應的互動關係，使用者可以依據所在的位置決定其群組，並且與相同群組的使用者互動，互動的模式包含下列三種：

1. 資訊的交換：同一個群組的使用者間可以相互交換訊息
2. 成員尋找：依據某些特徵值尋找群組內是否有符合該特徵值的成員
3. 註冊/移除成員：利用登入/登出介面讓使用者可以選擇加入/退出群組

展覽品情境管理員模組提供一介面方便讓博物館管理人員以及遊客可以與展覽品做互動。在一個博物館導覽系統中，展覽品的資訊對遊客而言是最主要的情境資訊。我們讓遊客可以透過行動裝置觀看展覽品更多額外的介紹，其他參觀

者對於展覽品的評價、回饋或是補充資料，參觀者也可以自行撰寫對於展覽品的回饋等功能。我們針對每一個展覽品的資料儲存方式依據 RSS[13]對一個頻道(Channel)所要求的屬性儲存，以方便進行分類、檢索、推薦、回饋等機制。總結來說，展覽品情境管理員提供下列功能：

1. 展覽品的尋找：我們可以在每項展覽品之上設計一個簡單的藍芽發送器，利用藍芽傳輸距離短的特性(約 10 公尺)讓使用者可以選擇在其附近的展覽品資訊；方便做更進一步的應用。
2. 展覽品評分機制：博物館的參觀者觀看之前參觀者對於該展覽品的相關評價，並且能夠給予評分。
3. 展覽品回饋知識的管理：每一個參觀者都可以對展覽品給予回饋以及補充資料，管理員可以檢視這些回饋並且保留有用的回饋給其他參觀者參考。

裝置管理員模組則必須考慮到使用者裝置的變異性。行動裝置的日益普及，使用者對於資訊的需求及處理也由個人電腦轉移到行動裝置上。在博物館導覽系統中，我們希望參觀者可以採用任何形式的終端裝置作為導覽系統與使用者溝通的介面。但是我們又不希望博物館的內容管理員必須針對各種不同裝置設計出資訊所要呈現的格式。

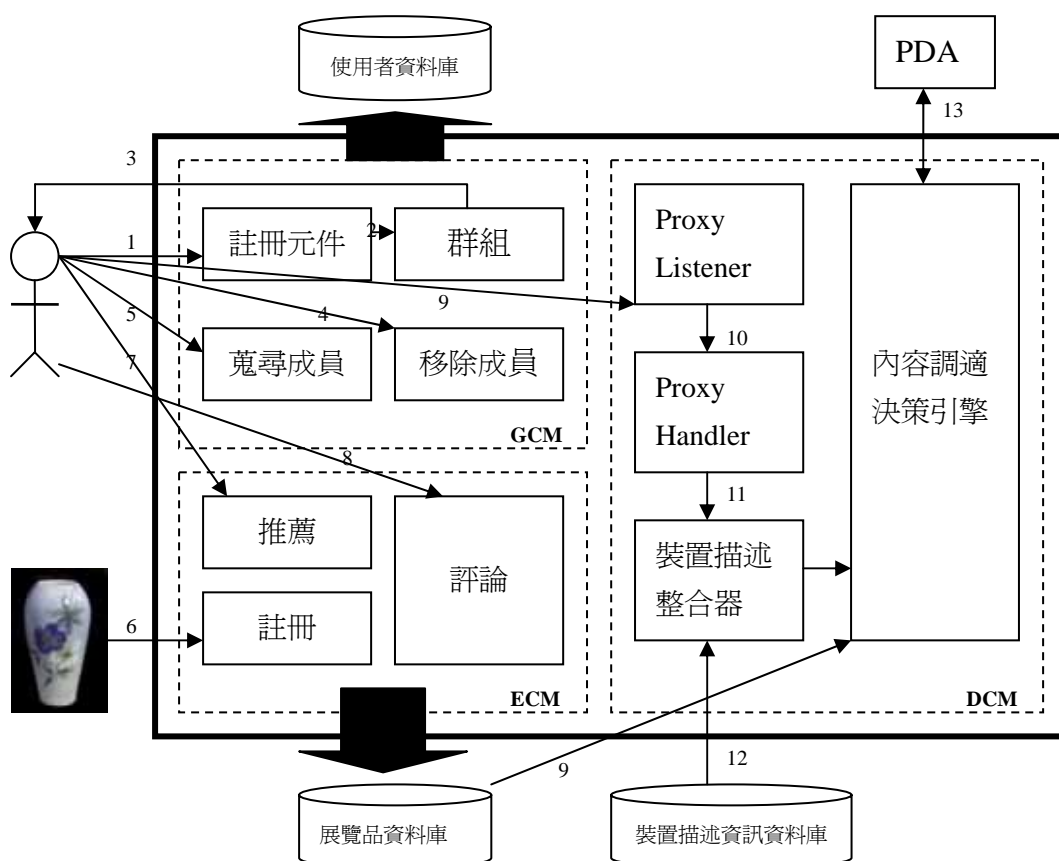
為了達到一個單一版本的機制，我們設計出一套中介軟體來克服這項挑戰，利用對於客戶端情境資訊的抓取，配合呈現內容產生出最符合客戶端裝置呈現格式的樣式表(Style Sheet)，讓每一個參觀博物館的參觀者都可以得到最佳的瀏覽經驗。在內容自動調整的各項研究中，對於應該在哪裡進行內容自動調整的功能是一個值得討論的議題。HTTP 1.1[9]列舉出內容自動調整進行的位置，可以分別在：Server-Side、Edge-Based(也就是 Proxy-Based)、Client-Based。我們選擇使用 Edge-Base 的方式執行內容調適的功能，除了一方面減少客戶端行動裝置處理內容的時間，另一方面也是要幫助博物館內容管理員可以達到單一內容版本的機制。

總結來說，裝置管理員模組提供下列功能：

1. 客戶端情境資訊的解析，所謂情境資訊包含客戶端裝置所處的連線環境、行動裝置能力特性以及使用者偏好。
2. 對內容伺服器進行內容的抓取，包含展覽品描述檔、圖片或是其他關於展覽品的額外資訊。
3. 利用頁面剪裁、非必要資訊的移除等演算法考慮情境資訊產生出適合於各項客戶端裝置瀏覽的樣式表。
4. 整合內容及樣式表，將結果傳送給客戶端裝置。

## 五、系統處理單元

情境整合管理員的系統架構如圖二所示，情境整合管理員依據系統的需求可以分為三種主要的情境，分別是展覽品情境、群組情境以及裝置情境，我們將之分為三個模組分別處理之，每一個模組又有其獨立的處理元件負責各項功能，在此，我們依據各項功能以及訊息傳遞的模式分別說明如下：



圖二、情境管理員各項單元元件及互動關係

### 1. 群組情境管理員模組

- A. 註冊元件：使用者首次參觀或是使用系統時需要先填入相關資訊以供日後展覽品的評分以及推薦機制使用。
- B. 群組：依據情境種類可以將使用者歸為各種不同的群組，比方說對某特定展覽品特別有興趣的參觀者可以組成一個群組；或是依照參觀者間的關係，像是同一學校來參觀的使用者，組成一個群組。
- C. 搜尋成員：使用者可以依據需求搜尋所對應的的成員，組成一個群

組，並且可以相互交換訊息，例如我們可以將「所在樓層」視為組成群組的一個情境。

- D. 移除成員：每一個成員有權決定是否參加/退出某群組，如果選擇退出群組，則該群組的其他成員無法與之互動。
- E. 使用者資料庫：儲存博物館參觀者的資料

## 2. 展覽品情境管理員模組

- A. 註冊：博物館人員可以透過註冊程序新增一個新的展覽品，並且給予相關描述資訊。
- B. 推薦：每個展覽品預設五個評鑑等級，參觀者在看過展覽品後可以對該展覽品評分，以供日後參觀者參考。
- C. 評論：展覽品的回饋機制，參觀者可以針對特定展覽品下評論或是彌補博物館所提供的展覽品描述資訊的不足。利用此共筆機制強化博物館知識庫，並且可以豐富展覽品的內涵。
- D. 展覽品資料庫：儲存展覽品的各項屬性、評鑑等級、回饋等資料。

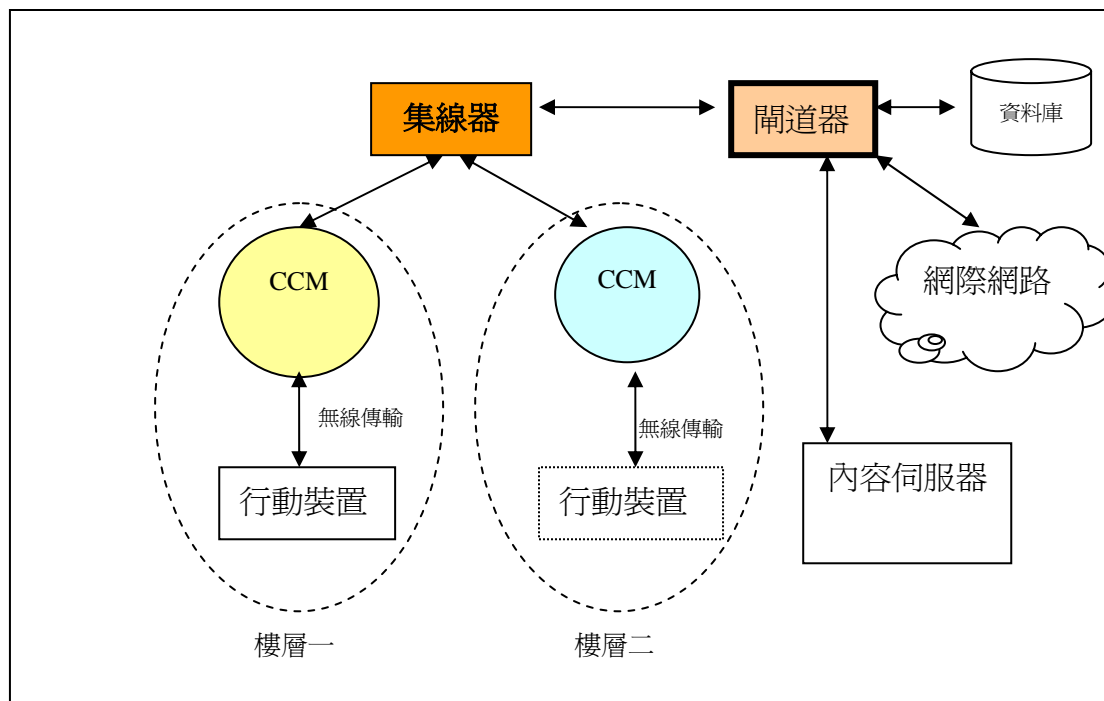
## 3. 裝置情境管理員模組

- A. Proxy-Listener：Listener 為裝置情境管理員處理行動裝置對內容服務請求的端口，Proxy Listener 收到請求後會產生一個 Proxy Handler 執行緒處理請求。
- B. Proxy-Handler：Proxy Handler 會接收客戶端裝置的請求 (HTTP 1.1 所提供的操作) 及標頭檔並作相關處理。Proxy-Handler 會將標頭解析並且分類，最後將結果傳送給裝置描述整合器。
- C. 裝置描述整合器：解析完 HTTP 的請求及標頭檔後，得到關於裝置的情境資訊，裝置的情境我們使用 HTTPext[10]及 CCPPex[11]所定義的標頭夾帶在 HTTP 請求的標頭檔之中，Proxy Handler 收到標頭後會依據 CCPPex 所定義的標頭內容到裝置描述抓取對應的裝置描述檔，再根據使用者所傳送的額外偏好資訊整合成一份完整的裝置情境描述，最後到內容伺服器抓取內容，將情境資訊及內容送給決策引擎處理，在此我們使用 HP Semantic Lab 所提供的 Jena[12]將情境資訊解析成 RDF[14]模型方便決策引擎處理。
- D. 決策引擎會依據 RDF 模型產生出適合客戶端裝置最佳的樣式表並且將結果回傳給客戶端。

## 六、訊息的傳遞及整合



訊息傳遞為計算機內軟體或硬體元件完成某項功能的資料傳送流程。所以在本系統的訊息傳遞模式中，我們以功能導向的方式來解釋每一個訊息傳遞的過程，以及訊息傳遞時所使用到的相關規格及技術。圖三展現了系統與外部資料的協同方式：



圖三、博物館導覽系統與外部環境的訊息傳遞模式

圖三中 CCM 為情境整合管理員模組(Context Coordination Manager)的縮寫，每一個樓層依據規模提供一個或一個以上的情境整合管理員提供服務。各個樓層的情境整合管理員提供使用者及外部資料互動的一個介面，所有的資料庫也都視為外部資料方便各個情境整合管理員使用。

我們以下列幾個主要功能說明圖二標示各步驟之訊息傳遞之細部功能如下：

1. 群組成員管理
  - A. 步驟 1：每一個成員登入時即可得到一個 SessionID
  - B. 步驟 2：群組管理員依據使用者情境找出使用者可以加入的群組
  - C. 步驟 3：群組管理員列出目前使用者可加入的群組
  - D. 步驟 4：使用者移除掉沒有興趣的群組
  - E. 步驟 5：搜尋成員模組列出特定群組內的成員
2. 展覽品回饋機制的運行
  - A. 步驟 6：對每一筆新加入的展覽品註冊，設定其預設屬性

- B. 步驟 7：參觀者對於展覽品的評鑑等級
- C. 步驟 8：因為每一個展覽品都是一個頻道，這裡提供回饋訊息內容

3. 網頁內容調適的步驟

- A. 步驟 9：瀏覽器(UA, User Agent)向 Proxy Server 提出內容服務的請求，Proxy Server 收到服務之後產生一執行緒(Proxy Handler)專門處理該請求。
- B. 步驟 10：Proxy Handler 處理請求、HTTP 請求標頭(HTTP Request Header)。
- C. 步驟 11：抓取內容伺服器中客戶端所需要的內容，合併動態及靜態的 CC/PP Profile 產生裝置屬性的 RDF 模型，如圖四所示。動態的裝置情境資訊表示針對使用者特定裝置產生出來的情境描述，內容包含連線方式、使用者偏好等時常變動的屬性，因此，每一次的連線都必須傳送「整個」動態情境以確保結果可以符合使用者目前的屬性。靜態的裝置描述資訊為廠商為某類型設備撰寫之屬性，例如：是否支援音效、色深、螢幕大小等，此類屬性依照機器型號為固定的，所以只要針對機器型號傳送一 URI(Unification Resource Indicator)[15]指到有儲存該裝置屬性的 RDF 檔即可，不需要傳送整個屬性檔以節省網路頻寬。動態以及靜態的屬性的合併依據合併規則(Resolution Rule)[16]決定所要採用屬性的先後次序以及重複的屬性的選擇方式產生出一份針對此次訊息交握最合適的裝置屬性檔。

```

GET /a-resource HTTP/1.1
Host: mis.im.tku.edu.tw
Man: http://www.w3.org/1999/06/24-CCPPexchange; ns=25
25-Profile: http://repository.im.tku.edu.tw/nokia.rdf, "1-23695f5b60aee55906fb2cd21829cee2"
25-Profile-Diff: <?xml version="1.0"?><rdf:RDF
xmlns:prf="http://www.wapforum.org/profiles/UAPROF/ccppschem-20010430#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"><rdf:Description
rdf:ID="MyDeviceProfile"><prf:component><rdf:Description
rdf:ID="HardwarePlatform"><rdf:type
rdf:resource="http://www.wapforum.org/profiles/UAPROF/ccppschem-20010430#HardwarePlatform"/><prf:SoundOutputCapable>No</prf:SoundOutputCapable></rdf:Description>
Proxy-connection: Keep-Alive

```

圖四、客戶端傳送給 Proxy 的請求範例

- D. 步驟 12：將裝置屬性及內容送交決策引擎，決策引擎會將網頁以及其他嵌於網頁內的 MIME 物件分離，處理如下：
  - 以 DOM(Document Object Model) Tree 剖析網頁，並走訪各個

節點，以經驗法則找出最佳的調整版本。最後產生出符合客戶端裝置的最佳樣式表，樣式表可以利用 XSLT 或是 CSS 技術產生。

- 若為嵌在網頁內的 MIME 物件，可以依據裝置情境資訊決定是否剔除或是改寫，比方說不支援音效的裝置我們就可以將音效檔砍除或是將音效檔轉成文字檔。只有長寬為 176 x 140 pixels 的螢幕我們可以將欲傳送的 640 x 480 pixels 的照片改寫為螢幕可以顯示的大小，一方面方便使用者觀看，減少客戶端裝置處理的時間，一方面也可以減少網路傳送的頻寬。

最後決策引擎搭配樣式表或嵌入於網頁內的 MIME 物件並且將改編的結果傳回給行動裝置。

## 七、實作

下面列舉出本系統中所實作的介面及相關說明：

### 1. 群組管理員模組



圖五、進入系統後使用者所見畫面



圖六、列出群組成員狀態

圖五及圖六為群組管理員模組在使用者端所呈現的畫面，使用者進入系統選擇所要互動的群組後可以看到同一群組成員的清單，在 Member 文字欄也可以輸入關鍵字找尋符合關鍵字的使用者。圖六為搜尋結果，系統將列出符合條件的使用者以及其所在樓層。

圖七為利用「興趣」及「所在位置」當作情境條件所構成的群組，使用者可以選擇興趣的項目，比方說程式設計(Program)以及所在位置。圖八為群組成

員間相互溝通的介面，使用者可以在 Message 文字欄中輸入要傳送的訊息，直接將訊息群播(Multicast)給所有的群組成員，或是在 Receiver 中指定要傳送的群組成員，只將私密訊息傳送給該群組成員。



圖七、利用「興趣」及「所在位置」作為群組依據



圖八、群組成員間利用行動裝置溝通的畫面

## 2. 展覽品管理員模組



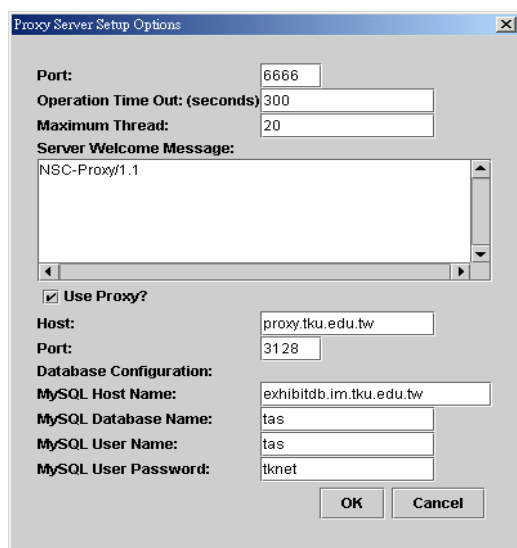
圖九、使用 Palm PDA 進入展覽品管理模組



圖十、使用 J2ME 平台瀏覽博物館內展覽品資料

圖九為使用 Palm PDA 進入系統之畫面，我們測試的客戶端系統可以選擇所要傳送給裝置管理員的裝置描述檔，並且觀看不同裝置描述檔所產生結果之不同。圖十為使用 J2ME 所提供的模擬器瀏覽博物館展覽品資料，使用者可以在此時為該展覽品評分，並且輸入相關的回饋，展覽品管理員會將回饋資料加入該展覽品所對應到頻道的回饋中。

### 3. 裝置管理員模組

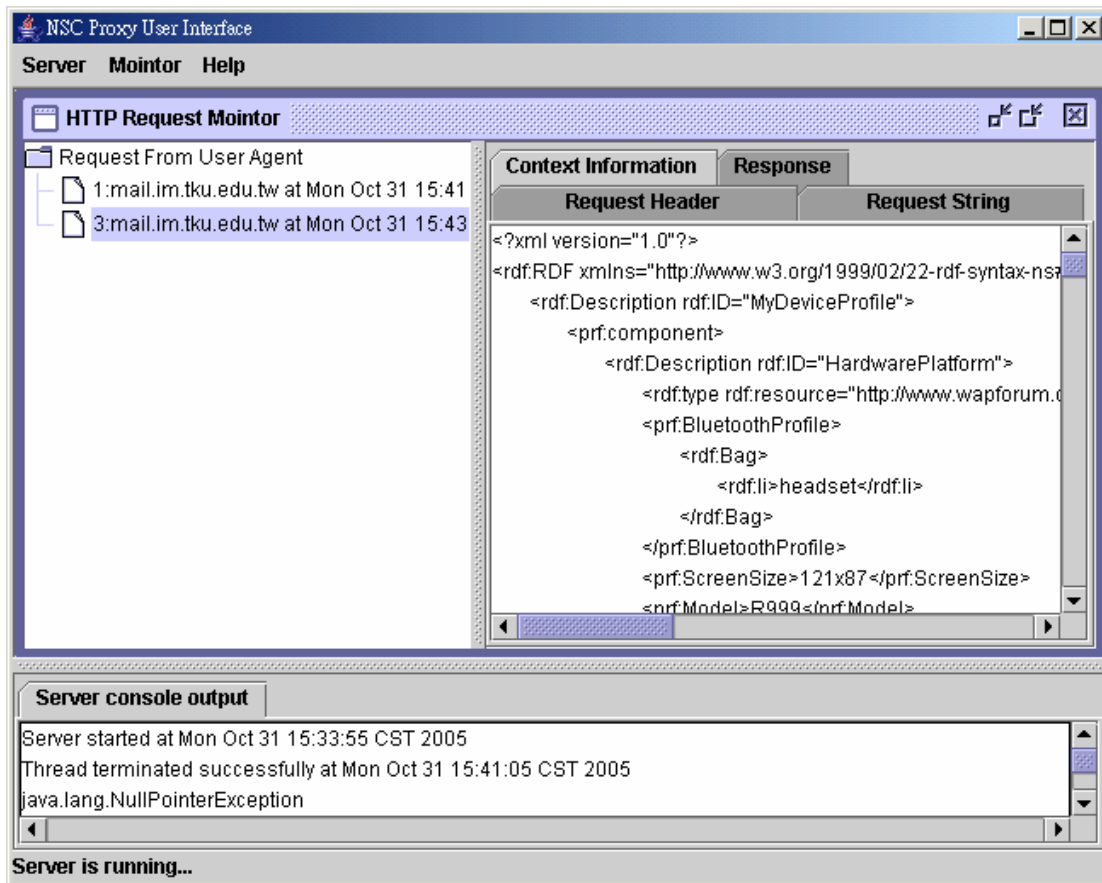


圖十一、裝置管理員模組設定畫面



圖十二、決策引擎所產生出的畫面

裝置管理員中，我們實作一圖形化介面方便管理並且觀察 Proxy Server 服務客戶端的情形，圖十一為決策引擎所產生出的結果；圖十二為裝置管理員模組設定的畫面：Port 部分可以設定裝置管理員所要傾聽連接埠編號；Operation Time Out 可以設定連線時間避免網路不穩定的程序佔用系統資源；Maximum Thread 可以限制 Proxy Handler 產生的數目以避免阻斷服務攻擊(DOS, Deny of Service)；Server Welcome Message 為設定 HTTPext 中所定義的 Proxy-Agent 所要設定的數值。如果 Proxy 會用到下一層(next-hop)的 Proxy Server 則可在 Use Proxy? 選取盒中打勾，並填入相關資訊於 Host 及 Port 中。Database Configuration 中設定了展覽品資料庫所在的位置以及相關連線參數。



圖十三、裝置管理員系統畫面

圖十三為設備管理員模組收到行動裝置請求後所處理的過程，我們利用下列四個部分顯示，分別是：行動裝置所要傳送給裝置管理員的表頭資訊(Request Header)、行動裝置所要傳送給裝置管理員的請求訊息(Request String)、合併動態及靜態情境之後的情境資訊(Context Information)以及決策引擎依照情境資訊所產生要給行動裝置的內容(圖十二)。

## 八、結論

本研究利用情境資訊營造一個智慧型的空間，設計及實作包含群組管理員、展覽品管理員、裝置管理員等 3 個模組之中介軟體，運用 CC/PP、RSS、DOM 等技術，在示範之行動博物館導覽系統中，讓使用者可以憑藉著行動裝置對於使用者所處環境的覺知、蒐集、解析、處理等方式幫助使用者可以和博物館內的各種環境有更多的互動，透過互動的過程，也可以得到關於參觀品更多的訊息。透過群組管理員的設計，我們可以利用行動裝置以及無線傳輸技術與其他博物館的參觀者互動，相互交換參觀心得或對於某個展覽品的感想。展覽品管理員可以讓每個參觀遊客不只單向的從博物館接收資訊，也可以回饋、貢獻新的知識給博物館，提供博物館與遊客雙向的互動介面。裝置管理員可以確保使用者的行動裝置在接

收訊息的時候可以依據本身裝置的能力得到一個良好的視覺效果，為上述兩功能的基礎。藉由這三個模組的相互合作，可以完成一個基本的博物館導覽智慧型系統，提供每一個到博物館參觀的遊客一個截然不同的參觀經驗。

在下一階段的研究中，我們預計在我們的智慧型空間提供更多的功能並且帶入更多的標準，這可以分為三個部分來說明：首先是內容的服務上，目前仍在使用傳統網頁的方式提供使用者內容服務，並且輔以少部分的文件檢索機制 (RSS)，在下一階段我們希望能夠全面的將文件內容以及格式的排版作徹底的抽離，第一個部分是引入 Web Service 的概念，將所有的服務寫成 Web Service 並且搭配 CSS 或 XSLT 做格式上的調適；而傳統的內容調適研究無法呈現及處理表單，我們將研究最近提出的 XForms[16] 規格，以產生出各種裝置都可以使用的表單格式，發展出更強大的使用者回饋系統，比方說投票、電子形式的問卷等等。第二個部分是決策引擎的加強，目前我們仍舊使用經驗法則來直觀判斷改寫或者是切割網頁，可能沒有辦法適用所有的狀況。未來，希望可以使用 RDF 對每一份文件做自動的註記，並且根據註記搭配情境屬性可以更聰明的覺察出使用者最想要的版面格式。透過決策引擎對於網頁調適規則的學習可以讓決策引擎更聰明。第三個部分是群組使用者間互動的訊息傳遞上，目前系統仍舊使用自訂的格式，未來希望可以引入其他既有、行之有年的標準，比方說跨平台之 Jabber 的 XMPP 協定，讓系統可以採用符合 XML 格式之標準介面，以具有更強的延展性及相容性。

## 九、參考文獻

- [1] D. Saba, A. Mukherjee, "Pervasive Computing: A Paradigm for the 21<sup>st</sup> Century", IEEE Computer, Vol.36(3), pp 25-31, 2003.
- [2] M. Satyanarayanan, "Pervasive Computing: Vision and Colleges", IEEE Personal Communication, Vol.6(8), pp 10-17, 2001.
- [3] A. K. Dey, D. Sallber, G. D. Abowd, "A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications", Human-Computer Interaction, Vol.16(2-4), pp 97-166, 2001.
- [4] A. Friday, N. Davies, E. Catterall, "Supporting Service Discovery, Querying and Interaction in Ubiquitous Computing Environments", 2nd ACM International Workshop on Data Engineering for Wireless and Mobile Access MobiDE 2001, pp 7-13, 2001.
- [5] S.F. Chen, "Designing Context Server for Context-Aware Computing", master thesis, Department of Information Engineering, National Tsing Hua University, July, 2002.
- [6] J. I. Hong and J. A. Landay, "An Infrastructure Approach to Context-Aware

- Computing*", Human-Computer Interaction (HCI) Journal, Vol. 16(2-3), 2001.
- [7] G. Banavar, and A. Bernstein, "Software Infrastructure and Design Challenges for Ubiquitous Computing Applications", Communications of the ACM, Dec., 2002, Volume 45, No. 12, pp. 92-96.
- [8] Schilit B, Theimer M. "*Disseminating active map information to mobile hosts*", IEEE Network, Vol.8, pp.22-32, 1994.
- [9] R. Fielding, J Gettys, J. C. Mogul, H. Frystyk, T. Berners-Lee, "*Hypertext Transfer Protocol*", UC Irvine, DEC, MIT/LCS, January, 1997.
- [10] H. Nielsen, P. Leach, S. Lawrence, "*An HTTP Extension Framework*", RFC 2774, February, 2000.
- [11] H. Ohto, J. Hjelm, "*CC/PP exchange protocol based on HTTP Extension Framework*", W3C, June 1999.
- [12] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, K. Wilkinson, "*Jena: implementing the semantic web recommendations*", International WWW Conference, pp.74-83, May, 2004.
- [13] RSS Specification, <http://blogs.law.harvard.edu/tech/rss>, 2003
- [14] P. Hayes, "RDF Semantics", World Wide Web Consortium, <http://www.w3.org/TR/rdf-nt>, 2004.
- [15] T. Berners-Lee, R. Fielding, L. Masinter, "*Uniform Resource Identifiers (URI): Generic Syntax*", Network Working Group, 1998
- [16] XForms - The Next Generation of Web Forms, <http://www.w3.org/MarkUp/Forms/>