

# The Fault-Tolerance of a Multi-Tier Roaming System over Internet

陳祈男 柯雅萍 劉崇汎 蔡志明  
C.N.Chen Y.P.Ke Z.F.Li uC.M.Tsai  
國立成功大學工程科學系

Institute of Engineering Science National Cheng Kung University  
Tainan, Taiwan, R.O.C.

Email: [cnchen@system.es.ncku.edu.tw](mailto:cnchen@system.es.ncku.edu.tw)

## 摘要

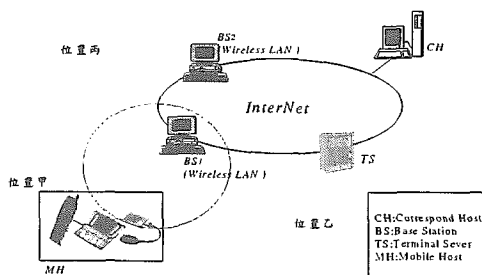
為了讓行動式計算機不受環境的限制，在 Internet 上漫遊，我們提出一行動電腦多型漫遊系統 (Multi-Tier Roaming System)，用以整合網際網路上的多種通訊裝置，其中包括(1)High Tier 通訊裝置(如 Cellular Phone)、(2)Low Tier 通訊裝置(如 Wavelan Card、紅外線)以及(3)Fixed Tier 通訊裝置(如 RS232、Ethernet)，將這些裝置充分地協調運用，使行動電腦達到無接縫(seamless)和通透性(transparent)的漫遊。

我們以 Sliding Agent-based Group Communication Model[1][2][3]為基礎，在行動電腦多型漫遊系統上架構容錯機制，解決因基地台損毀所引起的斷訊和資料流失等問題，並針對斷訊的原因作最迅速的處理，在最短的時間內，重新建立連線，藉此提高系統在行動計算通訊環境上的效率與可靠性

## 1、簡介

現今，無線通訊愈來愈受到重視，如何讓行動式計算機不受環境的限制在 Internet 上漫遊，一直是大家所關心的課題，但由於行動式計算包含了行動電腦技術和通訊網路的整合，並必須提供多元性的服務，所以要使行動電腦適應各種外在環境，漫遊於 Internet 上，是一件非常困難的事。而且和有線的通訊系統相比，無線網路有較低的頻寬，較不穩定的傳輸品質，會經常性或間歇性的失去連接，所以，在這樣不可靠的通訊環境底下，我們必須考慮如何處理錯誤發生、預防資料流失等問題。

如圖一所示，行動電腦(Mobile Host，以下簡稱 MH)可經由網路上的基地台(Base Station，以下簡稱 BS)。



圖一、環境架構

端機伺服器 (Terminal Sever，以下簡稱 TS )或直接利用 Ethernet 介面卡與網際網路上的伺服器 (Correspond Host，以下簡稱 CH) 連結，以存取 CH 端的資料。假設一開始 MH 在位置甲利用 某個 Wireless Lan 經由 BS1 與 CH 連線，並執行其通訊應用程式，當 MH 需要移入 BS1 的涵蓋範圍時，它可移至位置乙或位置丙，在此可能出現三個問題，(1) MH 由位置甲移到位置乙時，乙地不提供與位置甲相同的通訊服務，所以 MH 此時如要繼續存取 CH 端的資源，他必須更換與乙地相同的通訊介面，重新與網路連線，如此，在 TCP/IP 環境下的漫遊勢必終止。(2) MH 由位置甲移至位置丙時，丙地提供與位置甲相同的通訊服務，MH 雖然可以繼續利用和甲地相同的通訊裝置經由 BS2 繼續與 CH 連線，延續先前應用程式的連結，但如果 BS1 與 BS2 的通訊範圍不重疊，漫遊仍會中斷。(3) 通訊中的裝置因為不明原因而失效時，MH 極可能突然和基地台失去連絡，導至無線傳輸部份無法正常運作，更嚴重的是造成無法挽救的資料流失。

以上三種問題都會使 MH 與 CH 之間的連接中斷，我們針對情況(1)(2)，建構一個 Multi-Tier Roaming system[4]的系統來解決 MH 因移動產生的斷訊問題。簡單來說，所謂 Multi-Tier Roaming system 就是將現有的通訊裝置，如 Ethernet Card、RS232、Infrared、Cellular Phone 等加以整合，讓 MH 移動時可以快速又順利的切換到恰當的通訊裝置，利用其存取網路上的資源。拿情況(1)來說，MH 由甲地移到乙地，雖然乙地不支援和甲地相同的通訊裝置，但是本系統在 BS 和 MH 之間設有相對應的 Tier Manager，並訂定許多通訊協定，可以使 MH 移到乙地時，在使用者毫不知情的情況下將通訊裝置切換到和乙地相同的通訊裝置。再拿情況(2)來說，甲地與丙地的通訊裝置相同、發射範圍沒有重疊，依照舊有的單一 Tier 通訊方式，MH 與 CH 的連接會中斷，但是本系統可以在離開甲地時切換到不同的通訊裝置如發射範圍廣大的 Cellular Phone，等移入丙地時再換回較合適的通訊裝置。

雖然 Multi-Tier 的觀念可以成功地解決情況(1)(2)的斷訊問題，但是卻無法解決情況(3)軟體或硬體因不明原因

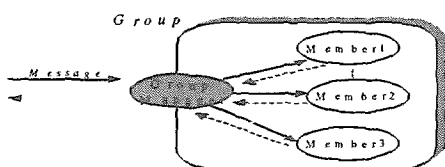
毀損而造成的斷訊，所以本系統必需再加進一些容許裝置毀損的容錯機制，用以解決情況(3)。

本論文以 Sliding Agent-based Group Communication Model 為基礎，在 Multi-tier System 上架構容錯機制，偵測何時斷訊，並針對斷訊的原因作最迅速的處理，在最短的時間內重新建立連線，使系統達到以下目的：

- (1) 減少因系統切換通訊介面(Tier hand off)所造成的資料流失。
- (2) 降低因設備損壞(如基地台當機)所帶來的影響。
- (3) 以最短的時間回復系統發生的錯誤。
- (4) 在使用者完全不知情的狀況下，將發生的錯誤回復，也就是做到錯誤的通透性(failure transparency)。

## 2、 相關研究

群組通訊[5][6][7]提供一個良好的通訊方式，有效地增加了系統的可靠度。如圖二，



圖二、 群組通訊

群組(group)是由群組成員(Group member)所組成的，而一個群組成員是一個通訊端點(communication endpoint)，可以接收和發送訊息。外界發出的訊息會送給群組的每一個成員(一致性)，各個群組成員的回應訊息則會被收集起來，只送給使用者一個回應訊息。對群組之外的使用者而言，群組是一個單一個體。

以下將針對群組通訊之基本觀念逐一討論，並說明這些觀念如何運用在容錯設計上。

### (1) 利用特定的群組名稱對外通訊

當外界要傳訊息給一個群組時，並不需要知道這群組內有哪些成員以及有多少成員，只要將訊息傳給這群組所提供的特定群組名稱即可，所以在對外界毫無影響的情況下，我們可以將一些成員或是程序組成一個群組，如成員為相同的程式，則只要有一個成員存活，便能持續運作，而達到容錯之目的。此外，在群組通訊的基本函式中，提供了可靠的通訊，即使訊息因為底層通訊實體的緩衝記憶體產生滿溢，或是傳輸實體產生錯誤，致使訊息流失，群組通訊仍然可以藉著可靠的通訊再將訊息重送。

### (2) 群組通訊的回復措施

群組通訊提供了回復(recovery)的功能，一旦群組中有成員發生故障，則群組進入回復階段，將群組中尚能正常運作的成員，再重建成一群組。而此種回復協定，

必須保證下列兩點：

- (a) 群組重建後的所有成員，必須要能收到原群組在產生故障前的所有訊息。
  - (b) 群組重建後的所有成員，必須要能收到故障產生後，這些成員所廣播的訊息。
- (3) 維持群組成員之間狀態的一致性

群組通訊保證所有送達特定群組的訊息，都能被該群組的成員依相同順序收到，以維持此群組內所有成員之間狀態的一致性。維持一致性的作法有兩種：

#### (a) Two-phase commit

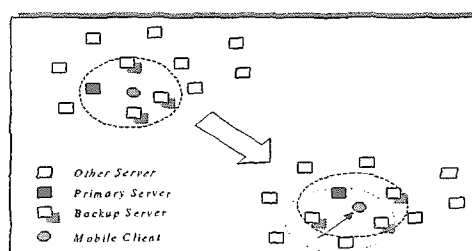
- 送方首先將資料廣播給群組中的其他成員。
- 接收者先將資料放入優先次序佇列(priority queue)中，並將它標記為不可傳遞(undeliverable)，然後將訊息的優先權設定為目前此佇列中最高值，且將此值傳送給送方。
- 送方收集所有群組中其他成員回送的優先權值，再從其中算出最大值作為此訊息之優先權值，並再將此值廣播給其他成員。
- 所有的接收者收到這個優先權值後，便將此值作為此資料在佇列中之優先權值，並將此資料改為可傳遞(deliverable)。然後將此優先次序佇列由小而大重新排序，再開始從此佇列中，由最小優先權值之資料傳遞給應用程式，重複這個動作，一直做到佇列中沒有任何資料，或是最小值的資料是不可傳遞的為止。

#### (b) Centralized

所有群組中的接受者收到廣播訊息後，透過一個集中成員(central site)指定訊息順序，然後群組中的成員再依此順序將訊息傳遞給應用程式。例如 Amoeba 便是利用送方先將訊息送給調序者 (Sequencer)，由其指定訊息的順序數字，然後再由調序者廣播給目的群。

目前利用群組觀念嘗試去解決行動電腦連接網際網路所產生問題的研究有[8][9][10]，比較著名的是 MobileChannel[7]。

MobileChannel 係以 ISIS System[11]為基礎，利用群組通訊的觀念，解決行動電腦移動問題的系統。其中採用兩種方法來實現一致性的機制：(1) Primary-backup Approach。(2) State Machine Approach。



圖三、 MobileChannel

如圖三，剛開始，能提供服務的 Servers 組成 Server Group，選定一個 Primary Server，其他的 Server 狀態則為 Backup，由 Primary Server 提供 Service 的一致性。只要 Mobile User 一移動，Mobile User 的 Server 群組就會跟著改變，加入相鄰的 Server，刪除不相鄰的成員，隨後再達成 Server 的一致性。因此，如果發生 Server Failure 如當機或通訊路線壅塞等因素，MobileChannel 能自動切換服務的 Server，讓計算能力和容量都較小的 Mobile Client 能繼續存取所需要的資訊。

### 3、 滑動代理器群組模式

我們針對行動主機在行動計算環境下漫遊所遭遇到的問題，提出一個滑動代理器群組的模式(Sliding agent-based group communication model，以下簡稱 SAGC 模式)，並以這個模式為基礎，考量系統出錯時，該如何恢復，使系統的可靠性提高。

#### 3.1 基本特性

以下分成三小點說明 SAGC 模式的特性。

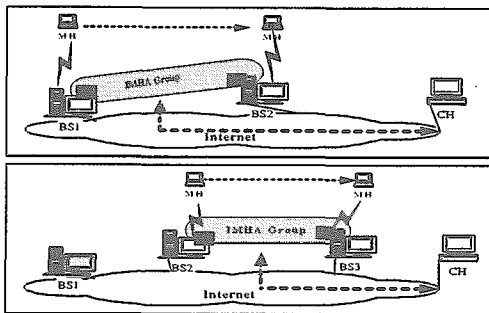
##### (1) 三端點通訊

為了使資料在有線網路與無線網路間順利地傳輸，除了在有線網路與無線網路間有一實體的連接點(即 BS)外，我們另外為每一個 MH，在其所處 cell 中的 BS 上，建立一代理器，稱為智慧型行動電腦代理器(Intelligent Mobile Host Agent)。因此，行動電腦、代理器和伺服器(或 CH)形成了三點的通訊模式。

##### (2) 滑動群組

將相鄰 cell 中 BS 上，代表同一 MH 的代理器(IMHA)集結成一個代理器群組，MH 透過群組與 CH 通訊，群組成員的數量保持最多不超過三個，以簡化群組通訊的機制、增加執行效率和避免網路資源的浪費。利用代理器群組作通訊的最大用處是保留一至多個備用代理器，以防通訊裝置出錯時產生通訊中斷及資料流失等問題。

當 MH 隨著使用者移動而跨越 cell 時，代替此 MH 的代理器群組必須隨著 MH 移動，所以我們在代理器群組中，動態地加入新的群組成員，刪除與 MH 相隔最遠的群組成員，整個代理器群組便可跟著 MH 到處滑動，



如圖四所示。因為 CH 只以整個代理器群組為通訊對象，  
圖四、 Sliding Agent Group

所以儘管群組成員有所改變，CH 卻是感覺不出來的，這種代理器群組滑動的觀念可以讓代理器群組如影隨行地跟著 MH 移動，隨時代理 MH 在 Internet 上作通訊，解決了 MH 因交接而造成的資料流失或資料重送等問題。

##### (3) 控制權的轉移

只有與 MH 在同一 cell 的代理器(IMHA)，能實際地傳送與接收 MH 對外通訊的資料；但因 MH 會隨時的移入新的 cell，所以實際替 MH 傳送與接收資料的代理器，也會隨著 MH 移動而改變，在我們的設計中，利用群組成員控制權的轉移[16]，來處理因 MH 跨越 cell，所必須進行的交接手續。

IMHA 共有四種狀態：Initialization、Active、Working 和 Backup，

- Initialization IMHA: IMHA 建立中的狀態，此為一暫時的狀態。
- Active IMHA: 當 IMHA 建立完畢，MH 可藉由此狀態的 IMHA 實際與外界通訊，此狀態的 IMHA 也是代理器群組的群組管理者，負責群組成員的管理和群組成員資料的一致性。
- Working IMHA: 當 MH 移至另一 wireless cell，此時 IMHA 的狀態就由 Active 轉為 Working，Working IMHA 繼續代替 IMHA 群組與 CH 通訊，可避免因交接造成資料的流失，此亦為一暫時的狀態。
- Backup IMHA: 當新 cell 中的 Active IMHA 與 CH 完成連接的轉移，且 Active IMHA 與 Working IMHA 完成資料的一致性時，IMHA 狀態就由 Working 變成 Backup，可作為容錯的目的，將來若 MH 再移入另一 wireless cell 時，可將此 Backup IMHA 刪除以完成群組的滑動，釋放系統資源，減輕網路負載。

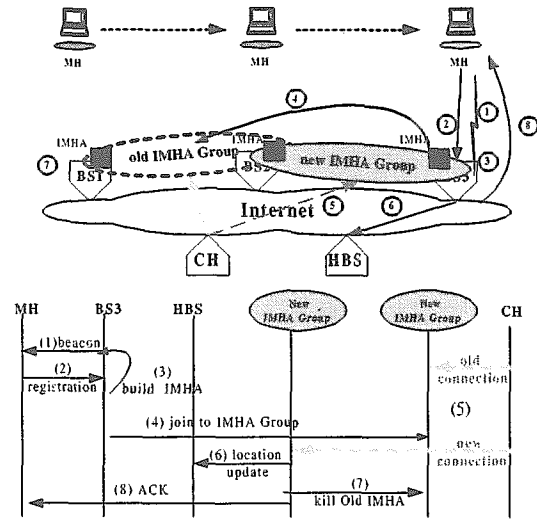
#### 3.2 運作模式

在我們的設計中，如圖五所示，就是以代理器群組的滑動來完成註冊程序，

因為代理器群組跟著 MH 而滑動，所以 CH 對代理器群組的實際連接，也要跟著轉移，下面為本系統註冊交接程序的步驟：

- 步驟① 行動電腦端的通訊駐留程式接收到基地台子系統廣播的訊息(beacon)，此訊息告知行動電腦現在所在的 cell 中，BS 的位址，通訊駐留程式決定是否要進行交接，至於通訊駐留程式如何做決定，不在本論文討論範圍，若決定要進行交接，就繼續進行以下的步驟。
- 步驟② 行動電腦端的通訊駐留程式發送註冊訊息給基地台子系統，此訊息包括代表 MH 的代理器群組名稱和 CH 的資料。
- 步驟③ 基地台子系統為此行動電腦產生一個新的代理

器，並將其加入代替行動電腦的代理器群組，若行動電腦是移回上一個經過的 cell，則此步驟可省略。  
 步驟④ 新的代理器對代理器群組發出 JOIN 訊息，要求



加入代理器群組。

圖五、註冊與交接程序

步驟⑤ 群組管理者通知 CH 端的通訊駐留程式，將對代理器群組的连接，從舊的代理器群組轉移到新的代理器群組。

步驟⑥ 群組管理者將 MH 位置改變訊息，通知 MH 的 HBS 上之基地台子系統，名稱伺服器會更改 MH 代理器群組的位置資料庫，記錄目前代理器群組所在的位置，在本系統中，即為 IMHA<sub>s</sub> 所在 BS 的 IP 位址。

步驟⑦ 群組管理者將多餘的代理器(IMHA<sub>i</sub>)刪除。

步驟⑧ 群組管理者回傳註冊完成的訊息，給行動電腦端的通訊駐留程式。

## 4、容錯之系統設計

### 4.1 系統錯誤之種類與分析

目前，無線通訊的可靠度仍然比有線通訊差很多，經常會發生斷線、資料流失等問題。

當有線端的固定成員 CH 與無線端的 MH 連結時，必須透過有線網路傳至 BS 上的代理器 IMHA，再經由無線網路傳至 MH 端，以下我們就整個傳輸路徑來討論所有斷訊的原因：

#### (1) CH 端

CH 端是網路上的現有成員，架構較簡單，出錯的機率較小，在硬體方面，CH 端可能因主機毀損造成斷訊，但這是屬於本系統不可掌控的範圍，所以在此不與討論，視為與 MH 失去連接即可。至於軟體方面，CH Daemon 錯誤是造成斷訊的主因，錯誤發生的原因可能是程式設計不良或是傳入大量封包導致封包遺失等。

#### (2) MH 端

MH 端在硬體方面的錯誤可能是主機故障或是通訊裝置損壞，主機故障的問題我們也不與討論，至於通訊裝置損壞，正是我們支援 Multi-Tier System 之容錯所要解的問題，軟體方面，Tier Adapter 接收不良、MH Daemon

當機等都是造成斷訊的主因。

#### (3) BS 端

BS 端是無線與有線傳輸的重要樞紐，必須與 MH 協調使用哪種 Tier 傳輸，還需要管理代理 MHs 的代理器，在硬體方面，造成斷訊的因素有：電腦故障、通訊裝置錯誤、斷電等，軟體方面的錯誤有：IMHA 毀損、Tier Adapter 失效、IMHA Manager 管理不當等。

#### (4) 連結 BS 端與 MH 端的無線網路。

無線傳輸的訊號在開放的空間中傳遞，非常容易受其他訊號干擾(interference)，有很高的錯誤率，造成斷訊的因素有：通訊干擾與超出通訊範圍等問題。此外，不穩定的通訊環境也經常會有資料封包遺失的錯誤發生。

#### (5) 連結 BS 端與 CH 端的有線網路。

目前一般有線網路都是使用 Ethernet 居多，其傳輸效率高，不易發生錯誤，造成斷訊的因素有：網路中斷、網路擁塞以及網路區隔(network partition)。在此，我們不考慮有線網路錯誤的情況，假設有線端永遠正確。

由上可知，在這整個傳輸的過程中，有許多不可預期的因素會照成斷訊，所以當系統發生錯誤時，必須有能力重建連線，並且保有斷訊前的重要資料。

### 4.2 錯誤偵測

針對以上 4.1 節的錯誤，我們提出兩項偵測措施，當通訊兩端因不明原因斷訊時，系統可藉由偵測措施迅速察覺錯誤，即時回復錯誤。偵測措施如下：

#### (1) 由基地台定時對外廣播 S 所在的 IP 位址

由基地台的 Tier Manager 管理基地台所有已裝設的通訊裝置，若無線通訊裝置處於可用狀態，就由該通訊裝置定時對外發出 beacon，直到該通訊裝置無法正常運作為止。這個偵測措施可以檢查出無線傳輸部分的錯誤，舉個例子來說，正在通訊的 MH 若超過一段時間沒有收到廣播，先假設此 Tier 出錯，切換到另一種 Tier，若再收不到廣播訊息，則可斷定基地台毀損或已離開此基地台的發射範圍。

#### (2) 群組成員定時確認其他成員是否處於有效運作狀態

這個偵測措施主要是想偵測有線端部分可能發生的錯誤，例如 IMHA 毀損、BS 當機等。本系統所設計的 IMHA 群組，在正常的狀態下，都是由 Active IMHA 代理 MH 對外作通訊，Backup IMHA 則處於備用狀態，偵測的辦法是由 Backup IMHA 定時發出確認訊息，確認 Active IMHA 是否處於有效的運作狀態。如果 Backup IMHA 連續發出的偵測訊息，一直沒收到 Active IMHA 的 ACK，則可判定 Active IMHA 異常，而造成 Active IMHA 異常的情況很多，如 IMHA 毀損、IMHA Manager 出錯、基地台當機等等，也就是說這樣的偵測機制可以把以上這些錯誤偵測出來。

### 4.3 錯誤回復

我們在系統上考慮容錯，容許 Multi-Tier Roaming System 發生以下四種錯誤：

#### (1) 資料遺失

基本上造成資料封包遺失的原因有兩個：(1) Transmission errors、(2) buffer overflows。transmission errors 主要是因為在傳輸線上產生訊號減弱(attenuated)及訊號失真(distorted)，而這兩者發生的機率，主要和下列三項有關：

- 傳輸媒介的種類：不同的傳輸媒介，有不同的封包遺失機率。
- 傳輸速率：單位時間內傳輸的資料愈多，封包遺失的機率愈高。
- 傳輸距離：傳輸的距離愈遠，封包遺失的機率愈高。

而 buffer overflows 發生的時機，則是當單位時間內，網路上傳輸的資料量，快於從 buffer 中取出資料的速度，這時所有的 buffer 都儲存未處理的資料時，則接下來在網路上的資料便無法再存入 buffer 中，這時如果送端繼續不斷地傳送資料便會產生資料包遺失的狀況。

基於對無線通訊環境的研究與探討，我們決定用 UDP(User Datagram Protocol) 作為本系統的通訊協定 [12]，但是 UDP 只負責將資料段送出，不保證接收端是否接收到資料，所以沒辦法提供可靠的通訊功能。為了預防資料的遺失，收受端每收到一個訊息，一定要回送一個回應(ack)訊息，若送方一直收不到 Ack，代表資料封包已經遺失，於是啟動重傳機制，重傳封包至收到為止。

除此之外，本系統在每一個通訊成員底層加設一 flow control 的機制，每個端點在收到封包後，都要先透過 flow control 確認次序後才可上傳給端點，如此一來，便可維持資料封包的次序性。至於送出封包部分，端點會先將封包送往 flow control 的 buffer 暫存，再送往目的地，若收到目的地傳來的封包 ACK 時，便把此封包由 buffer 清除，若 buffer 一直沒有收到 ACK，可由彼端的 buffer 直接再送該信息，因此，可提昇重送的效率。

#### (2) Tier 出錯

BS 的 Tier manager 會將所有可用裝置的 Tier Adapter 設為 Available 狀態，由 Tier Adapter 定時地對外發出 beacon，MH 如果接收到 beacon 就可以知道哪個 BS 的哪個 Tier 是可用的，相對的，若 MH 正在使用某個裝置通訊時，一樣會不斷地收到來自 Tier 的 beacon，若在設定的時間內收不到來自此 Tier 的任何 beacon，則判定是 Tier 出錯(MH 移出訊號發射範圍也視為 Tier 出錯)，於是，MH 會搜尋其他可用的通訊裝置，切換至其中一個可用的通訊裝置，優先順序為 Fixed Tier、Lower Tier、High Tier，若有相同種類的裝置存在，則選取通訊速率最快

的裝置，然後進行進一步的連接，最後才進行交接，若原 Tier 已經修復的話，MH 可對這個裝置重新註冊，切換回原來的 Tier。而 MH 切換到可用的裝置重新註冊，可能發生以下兩種情況，

##### (a) MH 移至同一 BS 上的通訊裝置

MH 不管用什麼樣的通訊裝置連接，都可由統一的 IMHA 代理，所以，當通訊裝置出錯時，不需要重新建立代理器，只要 BS 與 MH 之間的 Tier Manager 協定更換哪一種通訊裝置即可。

##### (b) MH 移至不同 BS 上的通訊裝置

因為無線網路卡、紅外線等通訊裝置的發射範圍較小，比較不易重複覆蓋通訊範圍，所以在一般的情況下，MH 若移到不同 BS 上的通訊裝置，大多都是切換至發射範圍廣大的 Cellular Phone，

#### (3) IMHA 毀損

IMHA 毀損時，由 CH 端來看，當 CH 端不斷將資料送給 MH 端時，資料會經由 BS 傳給 IMHA，再由 IMHA 依序地將資料傳給 MH，但是 IMHA 已經毀損，所以 BS 無法將傳資料給 IMHA，相對的，CH 端也就無法收到先前資料的 ACK，於是不斷重送，直到 flow control 的 buffer 滿了為止，CH 端的 Daemon 便通知使用者連線已中斷。同理，若是 MH 端送資料至 CH 端，遇到 IMHA 毀損的情況，也無法偵測出是 IMHA 毀損，只能判斷出是連線中斷，進而停止傳送資料而以。

但是由群組來看，群組成員中的 Backup IMHA 會定時發出確認訊息詢問 Active IMHA 是否處於有效運作狀態，若連續三次確認訊息沒有得到來自 Active IMHA 的回應，便判定 Active IMHA 毀損，由 Backup IMHA 發出訊息通知 IMHA Manager 清除 Active IMHA 舊有的記錄，再由 IMHA Manager 通知 MH 重新註冊，此時重覆註冊步驟便可回復系統狀態。

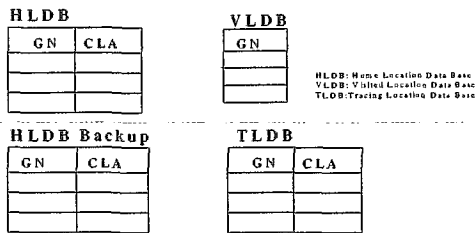
但是，這樣的回復機制無法解決 IMHA Manager 毀損的情況，只能等到 MH 要對外連接時，發現無法對外通訊，再移到可用的通訊裝置重新註冊，才可回復。

#### (4) Base Station 當機

當基地台當機時，所有的 Tier 裝置都無法正常運作，MH 也就無法再收到這個 BS 上所有 Tier 的 beacon，此時，MH 必須儘速對另一個基地台重新註冊，尋得新的代理器，並與先前的 Backup IMHA 結合成新的代理器群組，利用 Backup IMHA 備份的資料，繼續代理 MH。

另一方面，由於 BS 儲存許多 MH 的相關資料，所以 BS 在人工修復後，必須回復這些相關訊息。在 BS 中有兩個資料結構，一是 VLDB(Visited Location Data Base)，記錄在此 BS 暫時註冊的 MH 之唯一性的群組名稱(global unique Group name，以下簡稱 GN)，另一是 HLDB(Home Location Data Base)，記錄 HLDB 記錄在基

地台永久註冊的 MHs 之 GN，以及這些 MH 目前所在的位址，如圖六為 HLDB 與 VLDB 的資料結構



圖六、基地台的資料結構

如果沒有這些記錄，外界便無法藉由 MH 的 HBS 找到 MH，所以 HBS 修復後，最重要的一件事就是把資料回復，當 BS 重新修復時，沒有任何 MH 在此基地台暫時註冊，所以 VLDB 不做任何回復，保持空的狀態即可，至於 HLDB，我們提出以下四種方式來回復，

(一) 被動式回復法:

所謂被動式回復法，就是在 HBS 修復後，不做任何回復的動作，被動地等待 MH 因移出 Cell，重新註冊，自然會回 HBS 作更新 CLA 的動作。

(二) 被動式改良法:

利用方式(1)可以被動式地更新記錄，但是如果 MH 長時間不移出某個 Cell，就算 HBS 已修復也無法更新 MH 目前所在的位址，所以在 MH 端設定一時間計時器，從 MH 跨 cell 算起，每隔一段時間便到 HBS 更新 MH 目前所在的位址。

(三) 鏈結式回復法:

為了支援容錯，我們在本系統增加一個 HLDBb(Home Location Data Base backup)資料結構，如圖七，HLDBb 是將 HLDB 寫入硬碟備份的記錄，HLDBb 所存的訊息必須和 HLDB 達一致性，當 HBS 修復後，利用 HLDBb 的記錄回復基地台毀損前的記錄，但在 HBS 毀損期間，MH 可能移往其他 BS，所以，HLDB 所備份的是無法找到 MH 的舊記錄，基於以上考量，我們在 BS 上加設一資料結構 TLDB(Tracing Location Data Base)，TLDB 是儲存無法更新 HBS 之 MH 的目前行蹤，當 MH 到 HBS 更新目前所在位址失敗時，便把記錄存到 Backup IMHA 所處之 BS 的 TLDB 中，等 HLDB 修復後，利用 TLDB 的 CLA 記錄，逐一找到 MH 所在的基地台為止，最後，要求代理 MH 的 IMHA 群組回 HBS 更新 CLA，如此一來，HLDB 的記錄得以回復，以下是鏈結式回復法之演算法。

Link Recovery Algorithm(LRA):

- (1) HLDB restore from backup HL DB  
 $HLDB.GN \leftarrow HLDBb.GN$   
 $HLDB.CLA \leftarrow HLDBb.CLA$
- (2) MH update HL DB  
 if MH update HLDB success  
 $HLDB.GN \leftarrow MH.GN$   
 $HLDB.CLA \leftarrow MH.CLA$

```

else
  TLDBp.GN=MH.GN
  TLDBp.CLA=MH.CLA
(3) HLDB recovery CLA table
  VLDB.GN←-0
  while(!found)
  { CLA=lookup CLA(CLA) }
  lookup CLA(CLA)
  { if VLDB.= true
    found = true
    else if TLDB = true
    return CLA
    else HLDB.GN← 0 }
  
```

就 HBS 毀損期間，MH 沒有移往其他 BS 的情況來說，當 HBS 修復後，首先 HBS 會利用備份的 HLDB 重新取回 HBS 毀損前 HLDB 所記錄的資料(參考 LRA(1))，此時 HLDB 所記錄的 MH 目前所在位址得以回復，再拿 MH 在 HBS 毀損期間有移往其他 BS 的情況來看，HBS 修復後，利用 HLDBb 取得的是毀損前 MH 所在的位址，不是 MH 目前真正所在的位址，但演算法第二步我們將 MH 在 HBS 毀損期間 MH 所移入的 BS 一一記錄在 Backup IMHA 所在的 BS 上，所以 HBS 修復後，可利用這些記錄循序找到 MH 的行蹤(參考 LRA(3)Lookup(CLA))，由此可知，這個演算法不但能解決 HBS 毀損期間 MH 沒有移往其他 BS 的情況，還能解決 HBS 在毀損期間，MH 移往其他 BS 的情況。

但是這樣的鏈結方式解決不了一種特殊的情況--- 在 HBS 毀損期間，MH 由 HBS 移往其他 BS，MH 往 HBS 更新 CLA 自然不成功，因為 HBS 已經毀損，但 MH 將新位址轉存至 Backup IMHA 所在的 BS 也會失敗，因為 Backup IMHA 所在的位址正是 HBS 所在的位址，這種情況會造成 HBS 修復後，無法藉由 HLDB 的記錄找到 MH，所以當 HBS 找不到 MH 時，便將 MH 視為已關機，清除 MH 在 HLDB 上的記錄，等待 MH 重新註冊，便可回復(參考 LAR(3)HLDB.GN← 0)。

(四)鏈結式改良法:

HLDB 記錄 MH 目前所在位址，提供 CH 要連接 MH 時查詢用，所以我們在 HBS 修復後，不需要將所有 MH 的記錄回復，只需將 CH 所要找的 MH 之記錄回復即可。於是，我們對鏈結式回復法作改良，在 HBS 修復後，先回復 HBS 毀損前的記錄，但不作鏈結搜尋，直到 MH 主動更新，或是有 CH 到 HBS 詢問 MH 行蹤時再作追蹤(參考法(三))，此法會在 HLDB 增設一欄位記錄 MH 在 HBS 修復後是否已被更新過，等 HBS 的所有 MH 都被更新後，HLDB 便完成回復動作。

◆ 四種方法的優缺點

方法(一)被動式回復法，是利用 MH 跨 Cell 後，重

新註冊交接時的更新步驟來作回復，所以系統幾乎不需要任何的更動，其優點有(1)原理簡單，(2)節省系統資源，但此法屬被動式的作法，在 HBS 修復後，MH 若不跨 Cell 則 HLDB 無法回復，因此，方法(一)適用於使用者多且行動主機移動性高的大型系統。

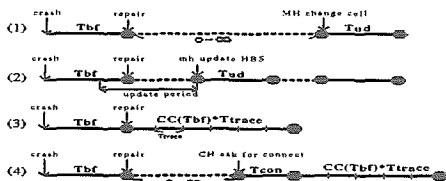
**方法(二)被動式改良法**，最大的優點就是改善了方法(一)的缺點，儘管 MH 不跨 cell 時，利用 MH 定時更新 HBS 仍可回復 HBS 的 HLDB 記錄，但是這個方法不管 HBS 有沒有損壞，所有的 MH 都必須做定時更新的動作，會造成系統資源不必要的浪費，因此這個方法也是適用於 MH 移動性高的系統，移動性高，跨 Cell 的次數多，定時更新 HBS 的次數就少。

**方法(三)鏈結式回復法**，在 HBS 修復後，主動尋找 MH 的行蹤，更新 MH 目前的位址，此法是三種方法中，最快的回復方法，但是其原理較複雜，尋找時會浪費許多系統資源，所以適用於使用者少的小型系統。

**方法(四)鏈結式改良法**，如果沒有任何 CH 想找某台 MH，則用法(三)回復其在 HLDB 的記錄，便是一種系統資源的浪費，法(四)在 CH 端要連接 MH 時，HBS 才去追蹤、更新，可以節省許多系統資源，但是這樣的做法，會拖長 CH 連接 MH 的時間，此法適用於不講究即時性的系統。

#### • 四種方法的效能分析

將以上四種方法的回復時間以時間數線圖來表示，如圖七。其中，Tbf 為基地台毀損到修復所需要的時間，



圖七、四種方法的回復時間

Tcon 表 CH 向 HBS 詢問 MH 位址所花的時間，Tud 為 MH 到 HBS 更新目前位址所需要的時間，Ttrace 為利用鏈結尋找 MH 時，到一個 BS 尋問 CLA 所需要的時間。CC(T)代表在 T 時間內 MH 平均跨 Cell 的次數，由以上時間數線圖可明顯看出，方法(三)所需的回復時間最短，方法(一)的回復時間最難預測，可能馬上回復，但如果 MH 不跨 Cell 的話，也可能永遠無法回復。

## 5、Implementation

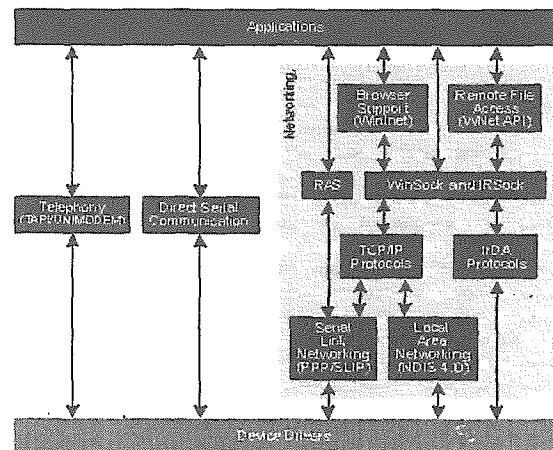
### 5.1 平台之選擇

本系統的作業平台架設在 Windows 系統之上。因為其安裝軟體的便利，目前已經相當普遍，另外 Windows 系統目前也已支援多工的作業方式，並利用訊息導向 (message driven) 的方式達到資源的有效利用；在網路支援上，Windows 系統採用 WOSA(Windows Open Service

Architecture)架構，可以提供多個網路服務及多種通訊協定的安裝，加上了作業系統易於安裝程式的特性，正好適合本系統的發展。

如圖八所示，為 Windows 系統支援連結網路的架構，其中表示此行動電腦可以使用支援多種不同的方式來控制各種通訊裝置。

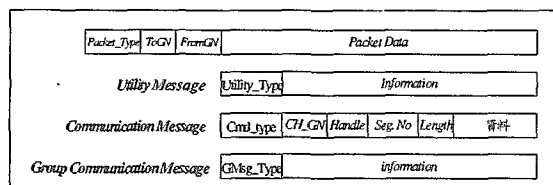
另外為了支援物件導向的觀念，我們採取 C++ 做為我們實作的程式語言，來製作本系統物件，有可擴充性及方便除錯的優點。



圖八、Windows 通訊架構

### 5.2 封包格式

下圖九為本系統內部元件彼此通訊所使用的封包 (Packet) 格式，網路協定則採用 UDP，因為其為一非連接性通訊協定，適合使用於行動電腦移動時協定連接層中斷的環境。



圖九、系統封包格式

Packet\_Type 表示後面所包裝的訊息為何種種類，我們將之分成三類，分別為 Utility Message，Group Communication Message 和 Data Communication Message。ToGN 為所要通訊的對象的群組名稱，FromGN 則是來源的群組名稱。

### 5.3 系統效能

以下對系統效能進行實際的量測。量測的環境為：  
 MH(Notebook): CPU Intel Pentium 166、2.1G H.D、32M DRAM。  
 Use Tier: Infrared、DBTEL WirelessLAN card、Cellular Phone、Ethernet card。  
 Network: 10M based Ethernet。  
 BS: CPU Intel Pentium 133、3.2G H.D、64M DRAM。

本系統除了可以達到漫遊的目的，更希望系統有效率地運作，因此針對影響系統效率最大的交接時間，作了以下的量測。也藉由下面的數據，提供未來系統修改

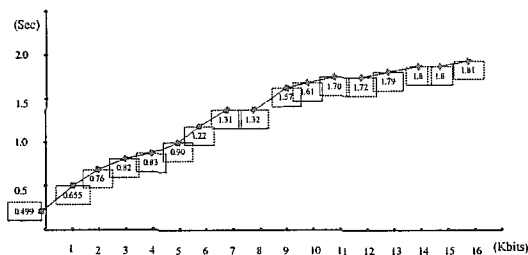
的參考。

#### (a) 交接所需時間

當 MH 找尋到一個可用的裝置，從發出註冊訊息到 IMHA 回應註冊成功的時間平均為 0.4961538 秒。此花費必須考慮當時之網路狀況。

#### (b) 一致性資料的傳輸

下圖十為根據 IMHA Data Store 中資料存放的大小，所需花費的交接註冊時間。



圖十、交結時一致性資料的傳輸時間

由上圖計算出每增加一筆資料(1Kbits)在 IMHA 的備份區(STORE)，交接時所需要付出的額外花費平均為 0.120667 秒。

#### (c) 使用單一代理器和使用代理器群組的效能差別

量測結果，使用單一 IMHA 代理 MH(以下簡稱 Case1) 平均處理一筆資料(每筆 1152byte)的時間需要 0.016 秒，也就是平均每秒可傳送 72Kbyte 的資料，而使用 IMHA 群組代理 MH(以下簡稱 Case2)平均處理一筆資料需要 0.042 秒，也就是平均每秒可傳 27.428Kbyte 的資料。不過這是 Case2 效能最差的情況，因為在 IMHA 的設計中，consistency 與 forward 是分開執行的兩個動作[17][18]，所以，對 CASE2 而言，整個 IMHA 群組運作時，效能最好的情況是當 forward 送出一筆資料給 MH，在等待 MH 回 ack 的期間，Consistency 正在處理另一筆資料的一致性，這樣的情況所花的處理時間就與 Case1 相當，反之，最差的情況就與 CASE2 相當，也就是每一筆資料都必須等待 Consistency 的動作完成，才可以對 MH 作 Forward 的動作。

由上述可知，利用 IMHA 代理器群組代理 MH 的 case2 比只用一個 IMHA 代理 MH 的 case1，要多 0~1.625 倍的資料處理時間，但 case2 可以達到比 case1 更高的可靠度，在這不甚穩定的行動式通訊環境，還是一個相當值得的作法。

## 6、結論與未來工作

本論文在 Multi-tier Roaming System 上架構容錯措施，解決因基地台損毀所引起的斷線和資料流失等問題，其中包括錯誤的偵測與重建，藉此提高系統在行動通訊環境上的可靠度

本系統目前所作的貢獻歸納如下：

- (1) 容許代理器出錯、通訊裝置毀損、基地台當機等問題發生，並保證本系統的回復措施仍可使通訊繼續運作。
- (2) 本系統在錯誤發生後的斷訊期間仍可保留斷訊前的資料，並保證斷訊期間暫時停止資料傳送，以防止資料流失。
- (3) 若資料流失，保證會不斷重傳資料封包至收到為止且收到的資料封包具有次序性。

## References:

- [1] Chyi-Nan Chen, Chung-Fann Liou, Ching-Yuan Wu, Rey-Yu Wu and Ting-Wei Hou, "Solving Location Problem of a Mobile Host by an Agent Group", Proceedings of The Seventh IEEE International Symposium on PIMRC, volume 2, p.708-712, 1996.
- [2] Chyi-Nan Chen, Chung-Fann Liou, Ching-Yuan Wu, Rey-Yu Wu, "The Design of Intelligent Mobile Host Agent", Proceedings of The Third Workshop on Mobile Computing, p132-139, 1997
- [3] Chyi-Nan Chen, Chung-Fann Liou, Ching-Yuan Wu and Rey-Yu Wu, "The Study of a Reliable Intelligent Moving Agent for a Mobile Host", Proceedings of Workshop on Distributed System Technologies & Applications, p.298-305, 1996.
- [4] Z.M.Cai, "A Roaming System over Internet with Multi-Devices", Master thesis, Dept. of Engineering Science, National Cheng Kung University, June 1999.
- [5] Kenneth P. Birman, Andre Schiper, and Pat Stephenson, "Lightweight Causal and Atomic Group Multicast", ACM Transactions on Computer Systems, Vol.9, No.3, pp.274-314, August 1991.
- [6] M.Frans Kaashoek, Andrew S.Tanenbaum, Kees Verstoep, "Using Group Communication to Implement a Fault-Tolerant Directory Service", Dept. of Mathematics and Computer Science Vrije Universiteit.
- [7] Kenjiro Cho and Kenneth P. Birman, "A Group Communication Approach for Mobile Computing". Proceeding of Workshop on Mobile Computing System and Applications December 8-9, 1994
- [8] Dhiraj K.Pradhan, P.Krishna, Nitin H. Vaidya, "Recoverable Mobile Environment: Design and trade-off Analysis", Proceedings of FTCS-26, 1996
- [9] Tason P.Jue, Dipak Ghosal, "Design and Analysis of Replicated Servers to Support IP Host Mobility in Enterprise Networks", IEEE, 1997
- [10] William Woo and Victor C.M.Leung, "HANDOFF ENHANCEMENT IN MOBILE-IP ENVIRONMENT", 1996
- [11] Kenneth Birman, Robert Cooper, "The ISIS Project : Real Experience with a Fault Tolerant Programming System", Operating Systems Review, volume 25, 1991, p.103- 107
- [12] Rey-Yu Wu, "A Reliable Wireless Communication Software for Sliding Agent-Based Group Communication", Master thesis, Dept. of Engineering Science, National Cheng Kung University, June 1997.
- [13] Chyi-Nan Chen, Chung-Fann Liou, Ching-Yuan Wu, Rey-Yu Wu, "The Design and Implementation of A Mobile Host Roaming System", Proceedings of National Computer Symposium Republic of Taiwan, volume 3, p.F107-F112., 1997
- [14] Ching-Yuan Wu, "The Design of Sliding Agent-based Group Communication for Mobile Computing Environment", Master thesis, Dept. of Engineering Science, National Cheng Kung University, June 1997
- [15] Chyi-Nan Chen, Chung-Fann Liou, Ching-Yuan Wu, Rey-Yu Wu, "The Sliding-Agent-Group Communication Model for Mobile Host Roaming", Proceedings of National Computer Symposium Republic of China, Tai-Chung, Taiwan, volume 3, p.F101-F106, 1997
- [16] Robert van Renesse, Kenneth P. Birman, and Silvano Maffei, "Horus: A Flexible Group Communication System", Communication of the ACM, Vol. 39, No. 4, pp. 76-83, April 1996.
- [17] Ya-Ping, Ke, "The Fault-Tolerance of a Multi-Tier



Roaming System over Internet', Master thesis, Dept. of Engineering Science, National Cheng Kung University, June 1999. C.N. Chen, C.M. Tsai, Y.P. Ke, "The fault-Tolerance of a Multi-Tier System over Internet", Proceedings of The 5th Workshop on Mobile Computing, 1999