

出國報告（出國類別：實習）

飛航訊息處理系統（AMHS）

軟體維護訓練出國報告

服務機關：民用航空局飛航服務總台

姓名職稱：邱家駒分析師、林泰祈分析師、郭錦源助理設計師

派赴國家：法國

出國期間：93/08/27～93/10/09

報告日期：93/12/21

公務出國報告提要

頁數：158 含附件：否

報告名稱：

飛航訊息處理系統（AMHS）軟體維護訓練出國報告

主辦機關：

交通部民用航空局

聯絡人／電話：

陳碧雲／(02)23496197

出國人員：

邱家駒 交通部民用航空局飛航服務總臺 資訊管理中心 分析師

林泰祈 交通部民用航空局飛航服務總臺 資訊管理中心 分析師

郭錦源 交通部民用航空局飛航服務總臺 資訊管理中心 助理設計師

出國類別： 實習

出國地區： 法國

出國期間： 民國 93 年 08 月 27 日 ~ 民國 93 年 10 月 09 日

報告日期： 民國 93 年 12 月 21 日

分類號/目： H2／航空 H2／航空

關鍵詞： AMHS, AFTN, cluster, X.400, X.500

內容摘要：(二百至三百字)

飛航訊息處理系統的目的在于轉送飛航訊息（例如：飛航計畫、航空氣象等的訊息）。AMHS 是新一代的系統，採用航空通信網路（ATN，註：這正是未來 CNS/ATM 的通信骨幹）作為傳輸的媒介，提供高度的系統可靠性，也保障了資料的完整性。這正是它優於現有 AFTN 系統之處。

本篇報告首先對整個系統的架構作個簡介，然後再分別介紹各個重要子系統的操作與維護。另外，為了避免因為硬體發生故障而導致服務中斷，本系統所採取的解決方案就是在各個重要子系統的主機均使用叢集（cluster）的架構，以提昇系統的可用性。本報告也將就叢集的原理加以介紹。

本文電子檔已上傳至出國報告資訊網

目錄

目的	1
第一章 AMHS 軟體訓練之目的	3
壹、 前言.....	3
貳、 軟體人員受訓之目的.....	3
一、 了解 AMHS 的運作原理	3
二、 熟悉軟體的架構以具備日常維護能力	4
三、 練習系統的重新安裝與設定.....	4
過程	5
第二章 訓練過程.....	7
壹、 訓練綱要.....	7
貳、 軟體人員訓練過程之說明.....	8
第三章 AMHS 系統介紹.....	11
壹、 AMHS 系統說明.....	11
一、 AMHS 系統組成份子.....	11
二、 高可用性的架構.....	13
三、 使用 AMHS 進行通信.....	14
四、 AMHS 系統管理席.....	15
五、 A2M 人機介面(MMI)功能.....	16
六、 AMHS 伺服器與各客戶端間的資料交換.....	16
七、 AMHS 伺服器與 A2M 之間的資料交換.....	17
八、 AMHS 伺服器與 IAT 的資料交換	18
九、 AMHS 伺服器與 UA 的資料交換.....	18
十、 AMHS 伺服器與作業印表機資料交流.....	19
貳、 ATN 路由器介紹.....	19
一、 ATN 路由器基本原理.....	19
二、 ATN 路由器硬體架構.....	20
三、 ATN 路由器軟體架構.....	21
四、 各子系統硬體架構	25
第四章 AMHS 伺服器安裝程序	35
壹、 建立 AMHS 伺服器使用者帳號.....	35
貳、 AERMAC 應用程式之安裝.....	36
一、 Aeramac 應用程式壓縮檔安裝	36
二、 產生 AERMAC 應用程式 process 之程序.....	36

三、 AERMAC 應用程式執行環境產生之程序.....	37
四、 首次啓動 AERMAC 應用程式之準備	39
參、 <i>ATS Message</i> 伺服器之安裝.....	44
肆、 <i>RMA</i> 軟體模組安裝程序及設定.....	45
一、 環境	45
二、 安裝及啓動程序.....	46
三、 <i>RMA</i> 軟體模組設定程序	48
伍、 <i>DBMS</i> 伺服器安裝程序及設定.....	49
一、 環境	49
二、 安裝及啓動程序.....	51
三、 <i>DBMS</i> 設定程序.....	51
陸、 <i>A2M</i> 應用程式安裝程序.....	52
一、 <i>A2M</i> 應用程式需求.....	52
二、 <i>A2M</i> 應用程式	52
三、 <i>A2M</i> 應用程式安裝程序.....	54
四、 <i>A2M</i> 應用程式設定.....	55
五、 <i>A2M</i> 應用程式之啓動.....	56
柒、 <i>AMHS</i> 伺服器環境變數、靜態及動態設定檔.....	59
一、 <i>AMHS</i> 伺服器環境變數.....	59
二、 靜態及動態設定檔	60
捌、 伺服器磁碟目錄組織.....	64
一、 AERMAC 目錄組織.....	64
二、 <i>ATS Message</i> 伺服目錄組織.....	69
玖、 <i>AMHS</i> 伺服器 <i>PROCESS</i> 檢視程序.....	71
一、 AFTN message switching and AFTN/ <i>AMHS</i> Gateway 應用程式 PROCESS 之檢視	71
二、 <i>ATS Message Server</i> PROCESS 之檢視.....	72
三、 <i>RMA</i> 軟體模組 PROCESS 之檢視.....	73
四、 <i>DBMS</i> 資料庫 PROCESS 之檢視	73
壹拾、 <i>AMHS</i> 系統啓動、重新啓動及 <i>switchover</i>	73
一、 啓動 <i>AMHS</i> 伺服器應用程式	73
二、 終止 <i>AMHS</i> 伺服器應用程式	78
三、 啓動及終止 <i>A2M</i> 應用程式	81
壹拾壹、 <i>AMHS</i> 維護程序	81
一、 AERMAC 應用程式	81
壹拾貳、 系統工具程式.....	86
一、 AERMAC 應用程式之工具程式.....	87
二、 磁性媒體儲存操作 (MAGNETIC MEDIA STORAGE OPERATIONS)	

.....	90
三、 A2M 工具程式	90
壹拾參、 <i>Thales-Is Linux Cluster</i> (簡稱 TILC)	92
一、 TILC 之介紹	92
二、 TILC 之硬體架構及工作原理	93
三、 TILC 之軟體架構及工作原理	95
四、 TILC 安裝	98
五、 TILC Configuration 設定	104
六、 TILC 管理	108
第五章 簡介 X.400 的標準和協定	114
壹、 前言	114
貳、 關於 MHS, MTS, MTA, UA, AU, MS 與 AMHS 的觀念	114
一、 X.400 網路	114
二、 AMHS 系統的優點	116
三、 AMHS 的各式 routing 型態	117
參、 訊息、 <i>report</i> 、 <i>probe</i> 、 <i>IPM</i> 、 <i>IPN</i>	121
一、 訊息	121
二、 <i>Probe</i>	123
三、 <i>Report</i>	123
肆、 <i>message access</i> 、 <i>message transfer</i> 與 <i>message store</i>	124
伍、 <i>P1</i> 、 <i>P2</i> 、 <i>P22</i> 、 <i>P3</i> 與 <i>P7</i> 協定	125
陸、 <i>O/R</i> 地址、 <i>O/R</i> 名稱與 <i>MF</i> 、 <i>XF</i> 及 <i>CAAS</i> 地址	126
一、 由 AFTN indicator 轉成 AMHS <i>O/R</i> 地址之步驟	128
二、 由 AMHS <i>O/R</i> 地址轉成 AFTN indicator 之步驟	130
柒、 <i>X.500</i> 目錄伺服器	131
心得	133
第六章 出國受訓之心得	135
壹、 明白 AMHS 的運作流程	135
貳、 了解叢集 (<i>cluster</i>) 架構	136
參、 熟悉軟體維護程序	136
一、 操作指令的練習	136
二、 系統重新安裝的練習	136
建議	137
第七章 建議事項	139
附錄	141
附錄 A ASN.1 的簡介	143

壹、 前言.....	143
貳、 資料格式.....	143
一、 Type	143
二、 Length.....	145
三、 Value	148
參、 實例.....	148
一、 例子一	148
二、 例子二	148
三、 例子三	149
四、 例子四	149
附錄 B TILC 的功能規格.....	150
壹、 前言.....	150
貳、 架構介紹.....	150
參、 CLUSTER 的運作.....	151
一、 CLUSTER 的啓動	151
二、 CLUSTER 的停止	151
三、 手動進行 Cluster Relocation.....	152
肆、 Cluster 的效能.....	156
伍、 注意事項.....	158

目的

第一章 AMHS 軟體訓練之目的

壹、前言

AMHS 是新一代的轉報系統，能夠提供最佳的資料完整性以及高可用性，而其軟體和硬體架構與目前所用的 AFTN 系統截然不同。爲了能夠讓民用航空局飛航服務總台的同仁具備自行維護以及操作的能力，因此派遣資訊管理中心的軟體人員以及台北航空通信中心人員分別赴法國原廠受訓。

貳、軟體人員受訓之目的

目前所使用的轉報系統（AIMS）的功能只是新轉報系統功能的一部分，因爲新系統包括一套 AFTN gateway，目的是和現存其他的 AFTN 系統作銜接，所以新系統也具備處理舊格式電報的能力，也因此而使得新系統的功能比舊系統複雜許多。派遣軟體維護人員受訓的目的則是爲了能夠具備自行維護軟體的能力，詳見以下各項：

一、了解 AMHS 的運作原理

由於 AMHS 不像 AFTN 是以專線固接的方式來傳送電報，而是透過網路來傳遞，因此 AMHS 的子系統有 ATN 路由器、MS(message store)、MTA(message transfer agent)、UA (user agent) 和目錄伺服器 (directory server) 等。而這些子系統則是使用各種不同的通信協定彼此溝通。

軟體維護人員不僅要明白每一個子系統的功能，也要了解各個子系統之間如何互動。惟有弄清楚基本的運作原理，才能對日後的系統維護與故障排除有所助益。

二、熟悉軟體的架構以具備日常維護能力

軟體的日常維護項目通常就是指以下幾點：

維護項目	說明
啓動系統	雖然理想上這是一個很少用得到的動作，可是還是要明白啓動的程序。
關閉系統	跟啓動系統一樣，這也是很少用得到的動作，不過卻是不能不懂的程序。
監看系統狀態	了解系統目前的運作情況對於維護而言是相當重要的。
系統資料的備份與還原	爲了保存重要的資料，一般而言，系統都會提供資料備份的機制。
故障排除	萬一系統出現異常，維護人員必須根據系統的紀錄檔（註：就是所謂的 LOG 檔）的內容加以判讀，以釐清原因並加以排除。

事實上，軟體的架構決定了整個系統的維護方式。因此，AMHS 系統的軟體架構也是一個學習的重點。

三、練習系統的重新安裝與設定

雖然整個系統爲高可用性的設計，發生故障的機會不大。但爲了保險起見，軟體人員還是有必要學習如何將整個系統從頭開始安裝，以及要如何設定參數。

對於系統的安裝與設定，雖然廠商提供了書面的標準作業程序，但絕對不是瀏覽文件就足夠，最重要的是我一定要實際演練。實地操作的話，一定會比只有光看文件還要來得印象深刻，並且也更有具體概念。

過程

第二章 訓練過程

壹、訓練綱要

本次 AMHS 系統國外教育訓練針對不同專長人員分三階段進行，每階段都有一星期的重疊期（詳如表 二 - 1），以便不同專長人員對新系統啓用建立良好銜接觀念。

訓練類別	人數	訓練期間	課程重點
硬體人員	3	93 年 8 月 9 日 ? 93 年 9 月 3 日	1、系統硬體架構 2、ATN ROUTER 介紹 3、網路監控與管理 4、伺服器叢集架構 5、陣列磁碟架構
軟體人員	3	93 年 8 月 30 日 ? 93 年 10 月 8 日	1、AMHS 系統簡介 2、ATN ROUTER 介紹 3、X.400 通訊協定介紹 4、AMHS 系統軟體程式架構說明 5、AMHS 系統軟體資料流程 6、伺服器叢集簡介 7、AHMS 子系統軟體建置 8、FIP(Front-End IP Processor)原理 9、LINUX 作業系統 kernel Rebuild 10、AMHS 系統網路安全管理 11、陣列磁碟原理 12、A2M、IAT 子系統功能介紹
操作人員	2	93 年 10 月 4 日 ? 93 年 10 月 15 日	1、AMHS 系統簡介 2、A2M 子系統操作

表 二 - 1 訓練類別一覽表

貳、軟體人員訓練過程之說明

訓練的期間共計六個星期，茲將過程列舉於表 二 - 2：

星期別	訓練期間	課程說明
第一週	93年8月30日 ~ 93年9月3日	<ul style="list-style-type: none"> ● AMHS 系統簡介：介紹整個系統的架構、子系統的功能與彼此間的關聯性，並比較新舊系統之間的主要差異。 ● ATN ROUTER 介紹：說明 ATN ROUTER 在 AMHS 系統中所擔任之角色，並介紹其功能。
第二週	93年9月6日 ~ 93年9月10日	<ul style="list-style-type: none"> ● X.400 通訊協定介紹：X.400 為 AMHS 系統傳送訊息的主要通訊協定，本週的課程介紹傳送過程中各階段所使用的協定及其功能。
第三週	93年9月13日 ~ 93年9月17日	<ul style="list-style-type: none"> ● AMHS 系統軟體程式架構說明：介紹 AMHS 系統軟體程式架構，各程式模組功能及相互間關聯性。 ● AMHS 系統軟體資料流程：說明各軟體程式模組資料處理流程，以及各模組間程序介面的連接方式。
第四週	93年9月20日 ~ 93年9月24日	<ul style="list-style-type: none"> ● 伺服器叢集簡介：伺服器叢集原理介紹，以及如何設定 AMHS 系統叢集架構參數。 ● AMHS 子系統軟體建置：練習如何重新建置 AMSS、AERMAC、DATABASE、RMA 各子系統。 ● FIP(Front-End IP Processor)原理：說明 FIP 在 AMHS 系統中的功能以及設定方式。
第五週	93年9月27日 ~ 93年10月1日	<ul style="list-style-type: none"> ● LINUX 作業系統 kernel rebuild：當作業系統核心參數值變更後如何重新建置作業系統。 ● AMHS 系統網路安全管理：介紹如何利用 Lorientpro 網管軟體來監控 AMHS 系統的線上流量。
第六週	93年10月4日 ~ 93年10月8日	<ul style="list-style-type: none"> ● 磁碟陣列的簡介與建置：實際操作磁碟陣列的建置程

		<p>序。</p> <ul style="list-style-type: none">● A2M、IAT 子系統功能介紹：介紹 A2M 及 IAT 子系統在 AMHS 系統架構中所擔任的角色及功能，並實地演練如何建置 A2M 及 IAT 子系統。
--	--	--------------------------------------------------------------------------------------------------------------------------------------------

表 二 - 2 訓練過程一覽表

第三章 AMHS 系統介紹

壹、AMHS 系統說明

AMHS 系統所扮演的角色為轉送 (routing) AFTN 與 AMHS 訊息的 ATN 中心，並提供能夠管理 routing 以及處理訊息的一切設備。

一、AMHS 系統組成份子

整個系統是由下列各子系統所組成：

(一) AMHS 伺服器叢集

依照功能的不同，我們可以再把它分成以下幾個方面來討論：

1、AFTN 訊息交換伺服器

- (1) 確保傳統的 AFTN 訊息得以持續地進行貯存與發送的交換（switching）服務。
- (2) 以 FIP（Front-End IP Processor）處理國內以及國際電報線路的連線。

2、AFTN/AMHS 閘道器（gateway）

- (1) 形成 AFTN 訊息交換網路的一部分。
- (2) 藉以逐漸地從傳統的 AFTN 訊息服務進步到 ATN 網路服務。
- (3) 從事 X.400 和 AFTN 之間的格式轉換、定址轉換以及服務轉換。

3、ATS 訊息服務

- (1) 提供存取 X.400 協定的服務（MTA 或 Message store）。

(二) 目錄服務伺服器叢集

提供存取 X.500 協定的服務（目錄服務）。

(三) ATN 路由器叢集

確保能夠透過 ATN 網路進行訊息的轉送與通信。

(四) 管理控制席(Administrative Positions)

操作人員能夠透過個人電腦和印表機取得警示與報表，並可以控制訊息的交換、網路的架構以及轉送訊息的路徑。

(五) 客戶端(Client Positions)

依使用型態可區分：

- 1、使用 IAT（intelligent AFTN user terminal）處理 AFTN 訊

息。

2、使用 UA (user agent) 處理 AMHS (X.400) 訊息。

(六) SNMP 網路監控席(SNMP Control and Monitoring Positions)

負責監督網路以及連接於網路上的設備之狀態。

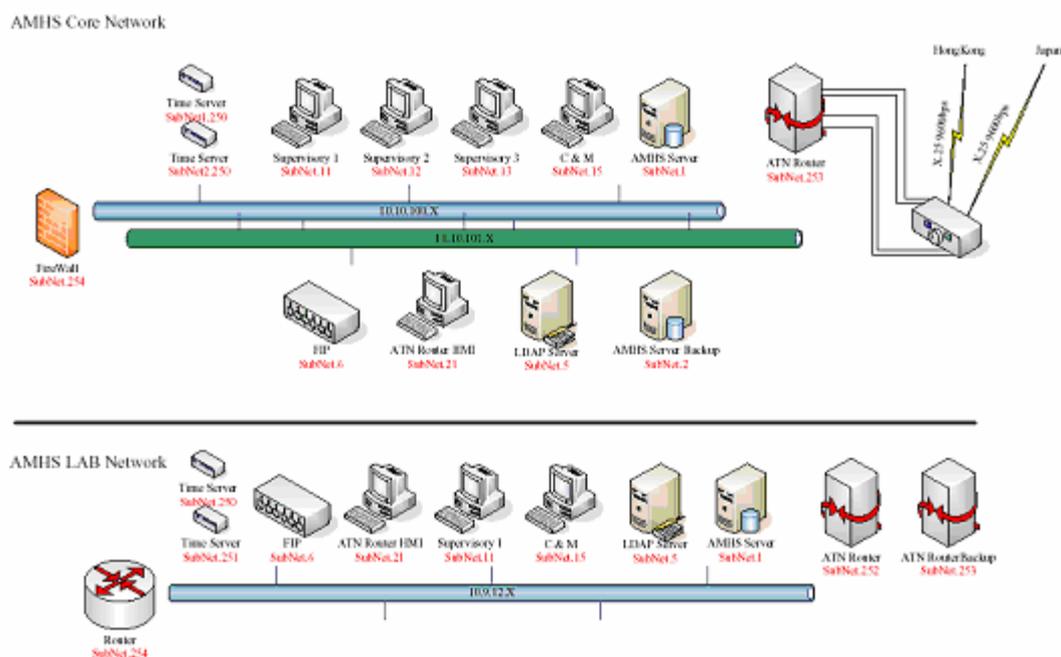
二、高可用性的架構

爲了確保萬一硬體發生故障時，系統仍舊可以繼續運作，而且服務也不會降級，因此必須把 AMHS 系統中的重要硬體元件以雙套備援。具有雙套備援的元件如下：

- 伺服器主機，它具備能夠將外部線路自動切換到工作主機上。
- 雙網路（含雙網卡、集線器 (HUB)及 FIPs）。
- 重要的設備均具備雙套電源供應器。

詳細情形請參考圖 三 - 1 AMHS 系統雙網架構圖。

圖 三 - 1 AMHS 系統雙網架構圖



三、使用 AMHS 進行通信

(一) 使用非同步介面

利用 Front-End IP Processor (FIP) 和使用非同步線路的 AFTN 交換中心相連。

(二) 透過網路

使用網路與系統連接的使用者當中，有些是位於區域網路內的近端使用者，有些則是在遠端的使用者，他本身要具備連上 WAN 的設備，例如：router 和 PDN。

MTA (Message Transfer Agent)、MS (Message Store)、DS (Directory Server) 彼此之間在傳輸層的通訊可選擇下列兩種方式：

1、ATN TP4 OVER LAN

2、RFC1006 (TP0 OVER TCP/IP)

至於 AHMS 系統的主機則是透過區域網路與 NTP 伺服器作時間同步。

(三) 使用公眾電話網路

原廠工程師可以利用數據機透過 PTN (公眾電話網路) 撥接連上系統的 Remote Software Maintenance modem, 以便進行軟體錯誤的分析與修復。

四、AMHS 系統管理席

AMHS 系統管理席的組成元件包括：系統終端機、個人電腦及印表機 (操作人員及系統管理者可以透過這些元件來控制系統及顯示系統警示訊息。)

(一) 系統終端機

- 控制作業系統。
- 修改作業系統及應用系統統計架構。
- 啟動及停止應用程式。
- 執行維護作業之工具程式。

(二) A2M (AMHS Management and Monitoring) 席

- 管理 AMHS 系統訊息之傳遞。

(三) 其他管理 ATN 路由器之個人電腦。

(四) 作業人員印表機

連續列印下列資料

- 系統警示。
- 各種報表。
- 統計資料。
- 高優先等級之資訊。

五、 A2M 人機介面(MMI)功能

A2M 人機管理監控程式原始碼是由 JAVA 語言所撰寫。完全相容於 Windows、UNIX、UNIX-like 作業系統，可提供對伺服器之資料及系統架構參數修改、讀取以及儲存之功能。

A2M 應用程式提供作業管理人員下列功能：

(一) AFTN 訊息的交換

- 監控訊息佇列。
- 訊息的讀取及再傳送。
- 修改錯誤訊息。
- 定義 AFTN 網路架構相關參數（例如：channel、circuit）。
- 定義 AFTN 訊息傳送位址。

(二) 控制 ATS 訊息服務

- MTA (Message Transfer Agent)架構參數設定。
- AFTN /AMHS Gateway 架構參數設定。
- UA (User Agent)架構參數設定。
- 控制 MTA 連接。
- 控制 MTA 訊息佇列。
- 目錄服務伺服器架構管理。

六、 AMHS 伺服器與各客戶端間的資料交換

AMHS 伺服器與下列的元件進行指令、回覆與其他資訊的交換：

- (一) **A2M 監督席**
- (二) **IAT (AFTN) 之客戶端。**
- (三) **UA (AMHS) 之客戶端。**
- (四) **作業印表機。**

七、 **AMHS 伺服器與 A2M 之間的資料交換**

所有重要的資料(例如：網路的架構、傳遞訊息路徑的架構、處理 AFTN 訊息的細節)都是貯存在伺服器本身而不是存放於個別的系統管理席。A2M 席本身只存有訊息預設路徑、格式及相關地址及各人偏愛之環境參數設定，每一個 A2M 席都與 AMHS 伺服器建立實體連線，如連線成功會顯示綠色“CONNECTION OK”圖示，失敗則顯示紅色，作業人員須輸入使用者名稱及密碼經系統認證後，方可對伺服器取得或傳送訊息以及取得網路相關資訊、動態本身架構之參數，並可管理訊息佇列。

伺服器端會定時對簽入成功之 A2M 端傳送以下的資料：

- (一) **目前 UTC 時間**
- (二) **統計訊息資料**
- (三) **系統告警旗標會在以下三種情形會被暫時設定為 1：**

- 定期列印統計報表。
- 列印訊息日誌檔。
- diversions 或 overflows 發生時。

當 TCP 連線失敗時，會收到上三項接收失敗封包，A2M 將會以紅色

圖形顯示，連線失敗，並會嘗試自行再連線。

伺服器連線成功、斷線、重新連線，都是經由 A2M 系統發出警示，除上述功能外，A2M 與 AMHS 伺服器間，彼此尚會交換以下的訊息：

- 由 A2M 向 AMHS 伺服器提出需求。
- AMHS 伺服器回應 A2M 需求。
- A2M 提出修改系統參數需求。
- AMHS 伺服器回應拒絕或接受前述需求。

如 A2M 九十秒內未接到 AMHS 伺服器回應，將自動斷線，並嘗試再度與 AMHS 伺服器連線。

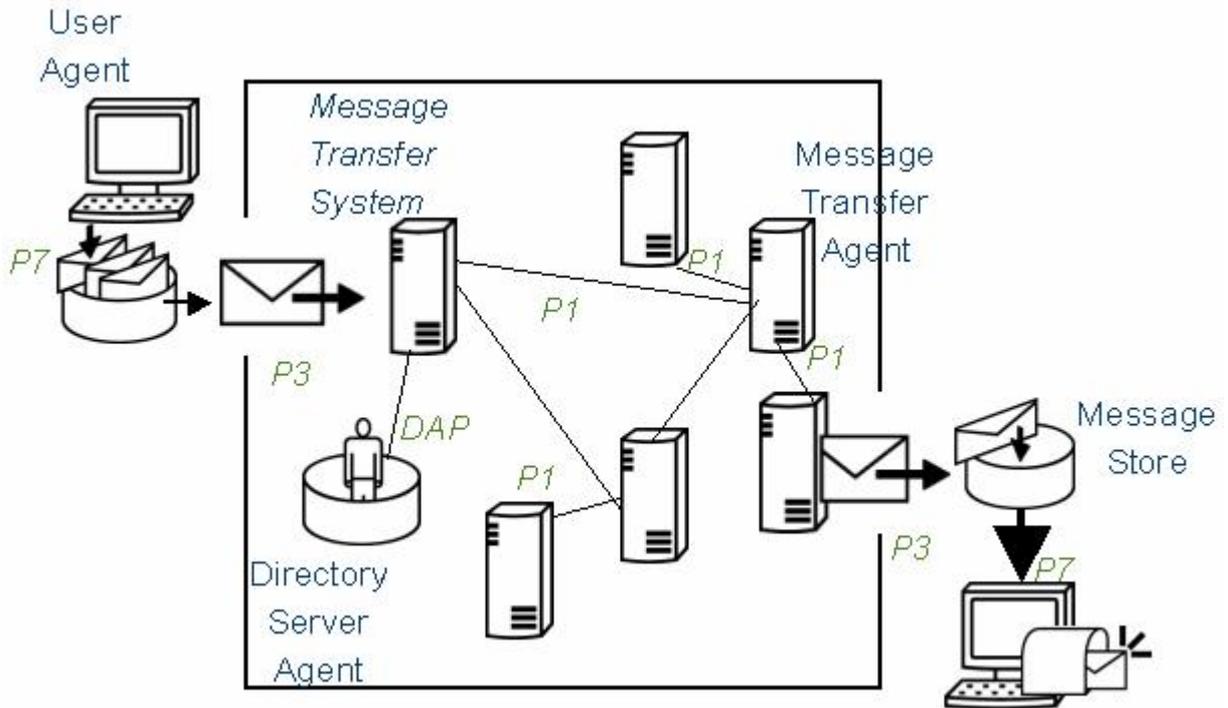
八、AMHS 伺服器與 IAT 的資料交換

IAT (Intelligent AFTN Terminal) 是一般 AFTN 傳統之使用者端，使其能正常傳送、接收 AFTN 訊息，依其連線方式可支援 TCP/IP 或非同步遠端連接介面。

九、AMHS 伺服器與 UA 的資料交換

UA (User Agent) 於 X.400 正常使用者端，使其能傳送、接收 X.400 協定之訊息，並支援 TCP/IP 連接。請參考圖 三 - 2。

圖 三 - 2 X.400 網路相關協定



十、AMHS 伺服器與作業印表機資料交流

AMHS 伺服器會送出不同種類文字訊息到作業印表機：

- 作業警示。
- 訊息列印。
- 各種統計。

貳、ATN 路由器介紹

一、ATN 路由器基本原理

ATN 網路是由所代表各種不同之使用者工作站及伺服器的節點所構成。

地對地 ATN 路由器的目的在於連接 AMHS 系統與 ATN 網路，可使彼此互傳訊息以及路由器間路徑資料的交換。(請參考圖 三 - 3 ATN 路由器之連

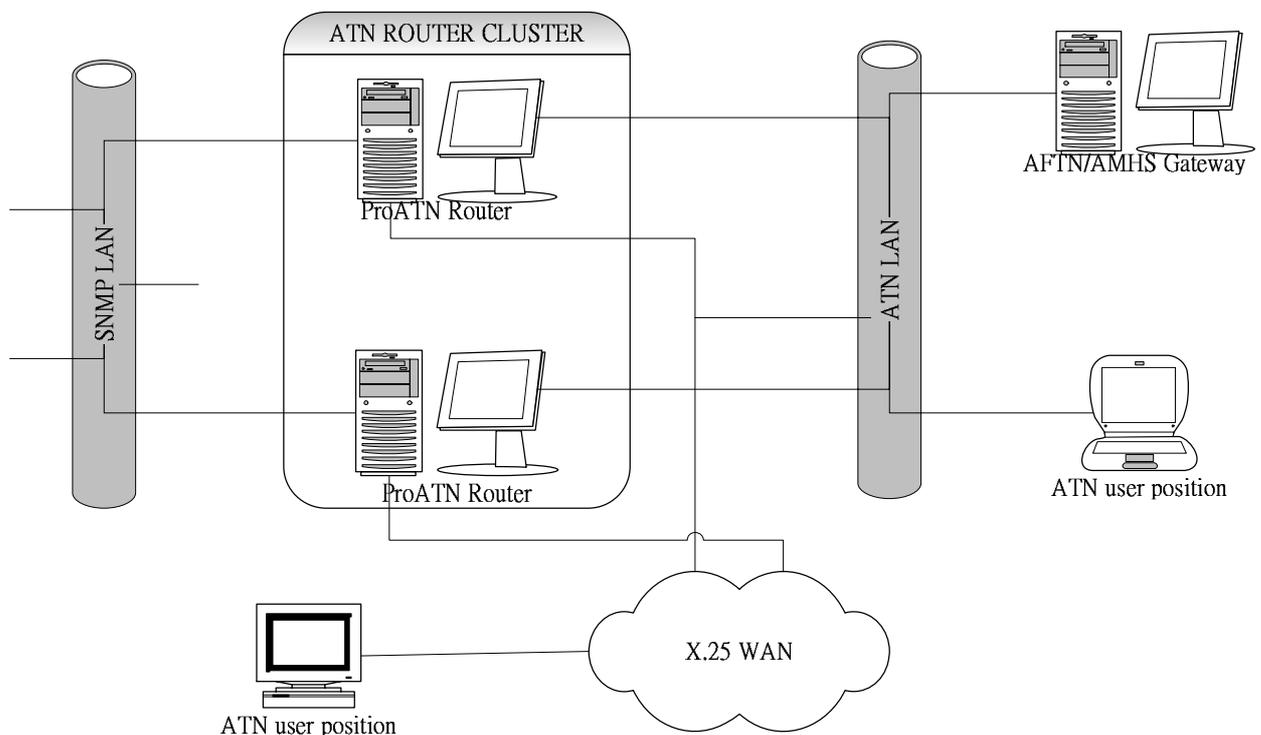
接)

爲了取得最新的 ATN 網路各節點連接資訊，所以路由器間必須進行路徑的資訊交換，爲了達成此目的必須透過以下三種協定：

- IDRP (Inter-Domain Routing Protocol)
- IS-IS protocol (Intermediate System)
- ES-IS protocol (End System-Intermediate System)

藉由使用 CLNP(Connectionless Network Protocol)和 SNDCF(Sub-Network Dependent Convergence Functions) 協定，就能夠讓訊息在 X.25 和乙太網路之間傳輸。路由器與系統其他主機之間的時間同步，則是透過 LAN 與外部之時間伺服器進行校時。

圖 三 - 3 ATN 路由器之連接



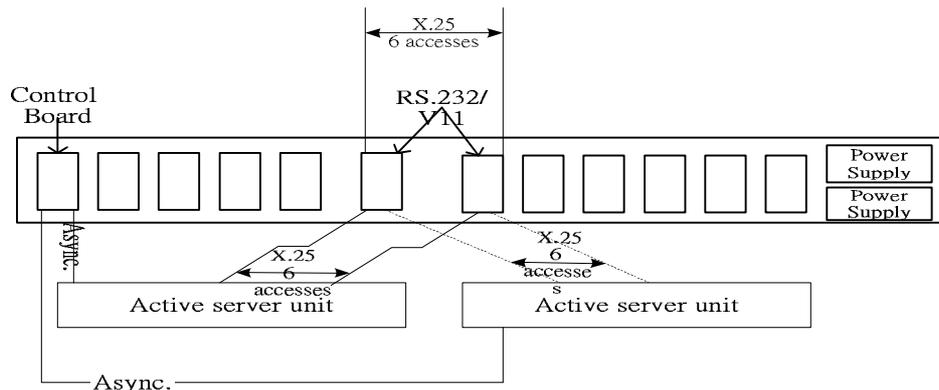
二、 ATN 路由器硬體架構

ATN 路由器爲安全理由，是由二部伺服器主機加上叢集硬體設備所構成

的叢集架構，詳細請參考圖 三 - 4。

構成叢集的二個成員會不斷的交換 HEARTBEATS 訊號，以便藉此得知彼此的狀況是否正常，如發現硬體有任何問題就會自動進行切換，以確保由正常的主機負責工作。

圖 三 - 4 ATN ROUTER SERVER CLUSTER



三、 ATN 路由器軟體架構

ProATN 路由器軟體是由下列三種主要模組構成：

- ProATN Lower Layers (LL) ATN Stack
- ProATN SNMP Agent
- ProATN SNMS Configuration Tool

(一) ProATN LL ATN Stack

ProATN Lower Layers (LL) ATN Stack 是 ATN 空對地或地對地 BIS (Boundary Intermediate System) 路由器，執行於 Linux 作業系統之應用軟體，是由下列元件所組成：

- ground SNDCEF (LAN, X.25, IP)
- CLNP, ES-IS, IS-IS protocols
- IDRP protocol

LL ATN Stack 可經由作業人員使用 oper 指令進行組態管理及監控。

LL ATN Stack 軟體是由許多 UNIX 程序所組成，其中的“IDRP subsystem”和“Router subsystem”等二個程序專門處理在 stack 方面的主要功能。至於全部的程序名稱如下：

- IDRPs subsystem
 - IDRPs protocol handler
- Router subsystem
- Transport layer handler
- PSUP-Supervisory processes
 - Launches all the other processes
 - Supervises the other processes
 - Auto-restarts processes that have stopped
 - Display events on local operator interface (if running)
 - Generates SNMP traps on events
 - Update history and trace files
- Operator processes (oper , several possible in parallel)
 - Require presence of PSUP process
 - Offer operator interface to ProATN function
 - Permit configuration of the stack processes
 - Enable start and stop of BIS connection
 - Receive notification messages generated by the processes
 - Permit dynamic adjustment to Router tracing levels
- SNMP Network Management Agent

由圖 三 - 5 可看出 ATN stack 處理程序、SNDCF 功能及通訊介面驅動程式之間的關係。

SNDF 採取所謂的 “Circuit Object” 來和下一網路層連接，而 “Circuit Object” 的成員則包含下列各項：

1、乙太區域網路 (or “Broadcast”)

ATN 資料透過乙太網路進行交換。

2、X.25 廣域網路

- 可以有許多個 “Circuit Object”，其個數視 X.121 位址數目而定
- 可以是所謂的 “static in”，也就是本地端的站台扮演接收來自遠端

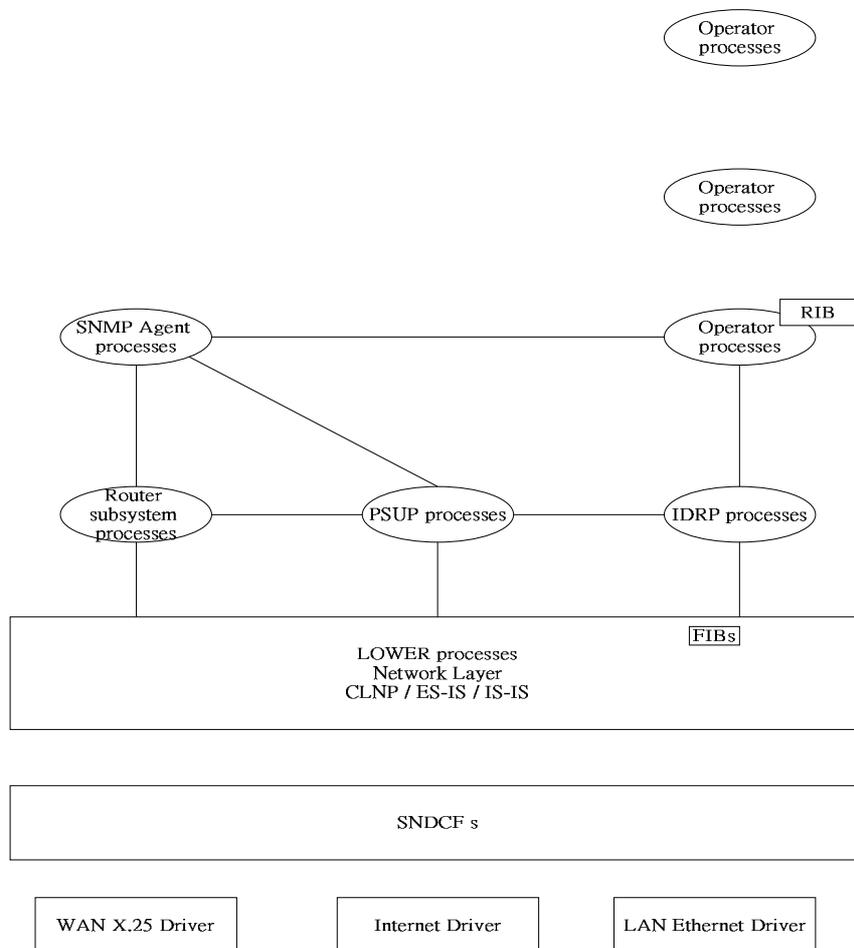
所傳來的“in coming call”封包的角色，此時仍維持連線狀態

- 可以是所謂的“static out”，也就是本地端的站台負責發送“in coming call”的封包到遠端的站台，此時仍維持連線狀態
- 可以是所謂的“Dynamic”，也就是本地端的站台只有在上一層有資料流通時才會嘗試建立連線，一旦過了一段閒置時間就切斷連線

3、廣域網路 IP 位址

- ATN 資料透過 IP 介面與外界交換。

圖 三 - 5 ATN ROUTER PROCESSES



(二) ProATN SNMP Agent

網管代理程式收集網路資訊，可提供作業人員網路連線狀態圖形及連接之障地。

(三) ProATN SNMS 架構管理工具

提供良好之圖形介面，管理 LL ATN Stack 及 ATN 網路路由器架構之工具程式。

(四) ATN 路由器其他特徵

- 1、開機後自動執行
- 2、ATN 路由器支援速度可達 64 Kpbs
- 3、依 ICAO 規定，有效管理訊息之優先順序、流量管制及過期封包處理
- 4、顯示所有錯誤、不正常之特殊事件日誌記錄
- 5、PSUP 處理程序可維護歷史檔及包含使用指令之記錄
- 6、可對不同通訊介面狀態作快速的追蹤及有效的離線分析工具
- 7、可利用 telnet 或 rlogin 作遠端的維護管理

(五) 符合各種協定之資訊

- 1、在網路間互相傳遞須符合以下的子網路協定：
 - ISO/IEC 8208 (applicable to the X.25 WAN accesses)
 - ISO 8802-3 (applicable to the Ethernet LAN)
 - ISO/IEC 8473 connection – less mode network protocol (CLNP)
- 2、關於路徑資料表(routing table)必須符合以下協定：
 - inter-Domain Routing Information Exchange Protocol – ISO/IEC 10747 (IDRP)
 - Intermediate System to Intermediate System Routing Information Exchange Protocol – ISO/IEC (IS – IS) Level 1 and 2
 - End System to Intermediate System Routing Information Exchange Protocol – ISO/IEC 9542 (ES-IS)
- 3、ATN 路由器符合 ICAO Document 9705 Edition 2 (ATN

SAPs Class 4 recommendations)。

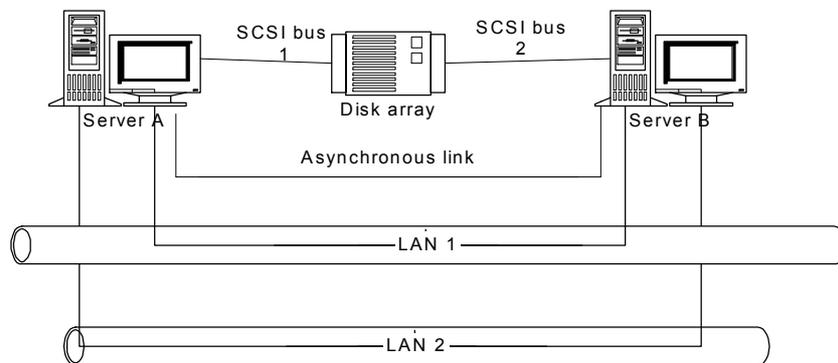
四、各子系統硬體架構

AMHS 設計的最主要特徵就是所有重要的組成份子都有 2 套。這是為了保證當系統某一組成份子之硬體發生故障系統並不會因此降低 AMHS 應用系統的服務。

由雙套所組成的子系統主要有以下各項：

- 所有伺服器的雙機都是由叢集架構所組成（請參看圖 三 - 6）
- 硬碟採 RAID 1 磁碟陣列的架構
- 所有的網路連接都支援雙網架構。(包含所有設備都具備雙乙太網路卡，含各伺服器、管理席、使用者客戶端及集線器)

圖 三 - 6 伺服器叢集架構



(一) AMHS 子系統伺服器叢集

AMHS 子系統組成包括 3 套 AFTN/AMHS/GATEWA 伺服器叢集，2 套目錄伺服器叢集和 2 套 Front-End IP Processors (FIPs)。硬體規格如表 三 - 1 所示。

伺服器系統名稱	硬體名稱	硬體規格	
		硬體元件	規格
AFTN	HP/Compaq Proliant Model	Processor	2 x XEON 2400 / 400 MHz

server / AMHS server / gateway server	ML370G3	Memory	4 GB
		Disk system	2 x 36 GB(Mirrored RAID 1)
		Ethernet Access	2 ports 10/100 MB
		Auxiliary storage	2.3 GB 3.5"
		Power Supply	Redundant Hot – plug
		Fans	4 fans Hot-plug
	HP/Compaq Shared Disks	Disk system	2 x 36 GB(Mirrored RAID 1)
Directory Services server	HP/Compaq Proliant Model ML370G3	Processor	2 x XEON 2400 / 400 MHz
		Memory	2 GB
		Disk system	2 X 36 GB(Mirrored RAID 1)
		Ethernet Access	2 ports 10/100 MB
		Auxiliary storage	2.3 GB 3.5"
		Power Supply	Redundant Hot – plug
	Fans	4 fans Hot – plug	
	HP/Compaq Shared Disks	Disk system	2 x 36 GB(mirrored)

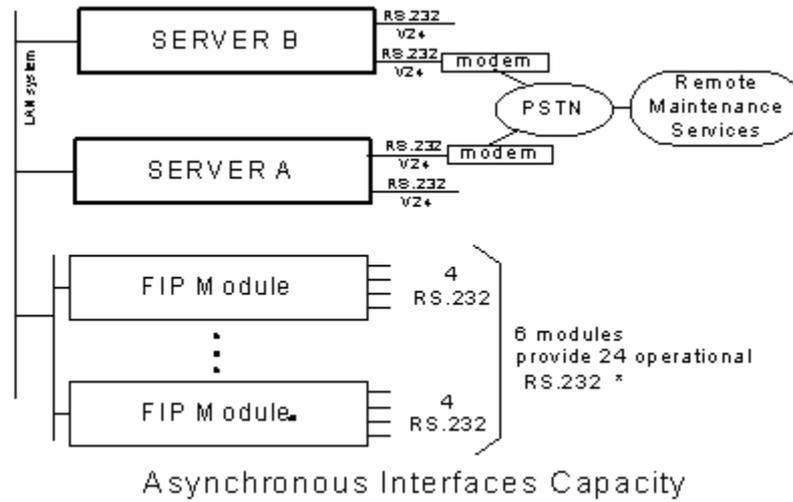
表 三 - 1 伺服器叢集的硬體規格表

(二) AMHS 伺服器通訊連接介面

1、AMHS 伺服器連接介面種類

- 非同步訊號直接連接
 - 在架構連線時要在伺服器及 AMHS 應用程式二方面進行
- 非同步訊號經由 FI 連接（請參考
- 圖 三 - 7）。
- TCP/IP 架構於乙太網路(遠端或近端使用者)
 - 所有遠端的使用者可經由乙太網路連接，但需提供廣域網路的設備如路由器、公眾網路.....等等
 - 正常都以雙網架構存在

圖 三 - 7 AMHS 伺服器叢集之連接



- AMHS 系統包含 OSI 的底層，可支援 Connection Oriented Network Service (CONS)、Connectionless Network Service (CLNS)協定、Collectionless Protocol (CLNP)，提供不同種類通訊堆疊服務，以確保 MTA、MS 和 DS 間資料傳輸。這些 OSI 的底層包括以下幾點：
 - ◆ ATN TP4 over LAN
 - ◆ RFC1006 (TP0 over TCP/IP)
- MTA 可經由 transport 堆疊 (stack) 直接與鄰近 MTA 傳輸或透過 ATN 路由器。(請參考圖 三 - 8 及圖 三 - 9)

圖 三 - 8 AMHS Transport stacks

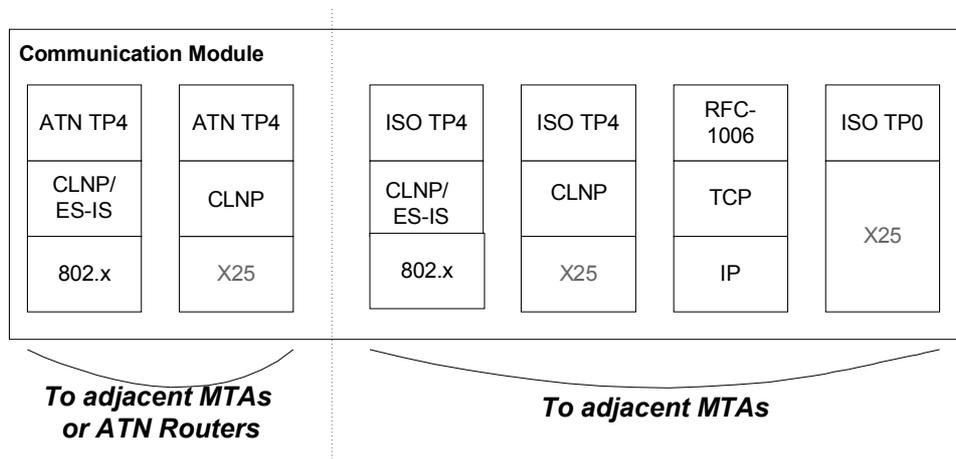
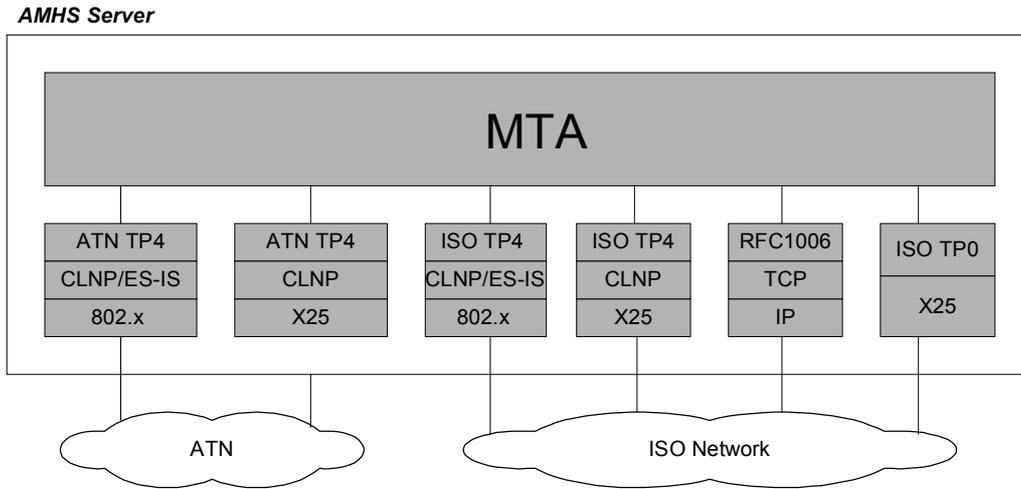


圖 三 - 9 MTA Transport stacks



2、AMHS 伺服器與外部時間伺服器的連接

AMHS 系統的伺服器一旦收到校時的信號之後時間就可以與外部的時間伺服器（註：時間伺服器的規格請參看表 三 - 2。）同步，而外部的校時信號可經由非同步線路或者經由區域網路來傳送。以下分別敘述這二種不同的方式：

(1) 非同步模式

- 直接以非同步介面與時間伺服器相連
- 時間伺服器會定時向伺服器傳送目前時間封包
- 經伺服器上專屬軟體解碼，調整伺服器時間

(2) 區域網路模式

- 經由區域網路 AMHS 伺服器接受時間伺服器依相關協定 NTP 信號作時間同步

NTP TIME SERVER	
Manufacturer	MITAC
Manufacturer Supply Code	Symmetricom TS2100

表 三 - 2 時間伺服器的規格

3、AMHS 伺服器與 RSM 的連接

RSM (Remote Software Maintenance) 是指系統在保固期間，原廠 (Thales-IS) 系統維護人員可利用數據機撥接，經由公眾電話網路直接進入系統進行維護及分析工作。為此，AMHS 具備 V.24/RS-232 介面，以便和 RSM 數據機相連。

4、Front-End IP Processors (FIP)

- Front End Processors (FEPs) 有集線器功能，能讓 AMHS 系統控制許多條電報線路。每單一伺服器可供 V.24/RS-232 介面數可達 16 個。
- Front End IP Processors (FIP) 硬體可架構於標準 TCP/IP 網路
 - FIP 經由雙乙太網路與 AMHS 系統連接。
 - 內部 FIP 模組是使用 TCP/IP 架構於雙乙太網介面。
- FIP 的架構具有下列的關鍵優勢
 - 高度的系統可擴充性
 - ◆ 每個 FIP 機櫃可以有 4 條到 192 條的 RS-232 線路。
 - ◆ 伺服器透過網路所能連接的 FIP 機櫃，並無數量上的限制。
 - 可以利用最新的 LAN/WAN 技術所提供的寬頻服務。
 - 以分散式架構建置於標準的 WAN 架構。
 - 可經由一般商用 SNMP 軟體管理及監視。
 - 快速及容易維護的硬體操作。
- FIP 系統所配備的 IP 模組最多可達 48 個，而每個 IP 模組的功能如下：
 - 4 個 RS-232 非同步介面
 - 二個乙太網路介面
 - 一個處理器
- 包含以下 AFTN 協定的特性：
 - 可偵測訊息中 message start 與 message end 的字串。
 - 可偵測 QTA 訊息格式。
 - 可處理 ASCII (IA-5) 及 BAUDOT (ITA-2) 二種不同的訊息格式。
 - 訊息長度計算。
- 有額外的實體介面可用來和電報線路相互連接
- FIP 可以透過 TCP/IP 的廣域網路與位於遠端的伺服器相連。
- FIP 為提高妥善率而在重要的零件採用雙套備援：
 - 雙乙太網路骨幹
 - 雙乙太網路連接介面
- 硬體維護時可線上熱抽拔更換的模組：

- IP 模組
- 電源供應器模組
- 乙太網路集線器

(三) ATN 路由器之硬體架構

ATN 路由器是由二部伺服器主機以叢集架構存在，此二伺服器成原，彼此不斷相互交換主機現況訊息（heartbeats check），如有任何成員硬體設備產生問題，系統會自動切換主機。詳細叢集架構裝備請參考。

伺服器系統名稱	硬體名稱	硬體規格	
		硬體元件	規格
ATN ROUTER Server	HP/Compaq Proliant Model ML370G3	Processor	2 x XEON 2400 / 400 MHz
		Memory	2 GB
		Disk system	2 X 36 GB U320 SCSI 1000 rpm Mirrored RAID 1 (RAID Controller : SMART 532 PCI)
		X.25 Telecommunication	GCOM X.25 PCI board-4 ports GCOM X.25 PCI board-2 ports
		Ethernet Access	ports 10/100 MBRJ-45 interfaces
		Auxiliary storage	2.3 GB 3.5"
		Power Supply	Redundant Hot-plug
		Fans	4 fans Hot-plug
	HP/Compaq Shared Disks	Disk system	2 x 36 GB(Mirrored RAID 1)
	VIR Circuit Switching Unit	Control	1 control module
Power Supply		1 redundant power supply	

		Networking	7 lines switching unit boards supporting data speed up to 64 Kpbs. 1 out of the 7 lines is used for internal switching
		Cabling	Set of cables
NTP Time Server	MITAC Symmetricom TS2100	(無)	(無)

表 三-3 ATN 路由器叢集架構裝備

(四) AMHS 系統管理席硬體架構

此席位包含 A2M 席、控制及管理席 (C&M position) 以及 ATN 路由器 MMI 席。

1、A2M 席之裝備規格

作業系統為 MS Windows XP Professional Traditional Chinese，連接方式為 TCP/IP。硬體設備規格如下：

ACER FT100	
Processor	INTEL P4-3.0G
Memory	512 MB
Disk system	40 GB
Ethernet Access	2 ports 10 / 100 MB
CDRW	52X CDRW

Supervisory Monitor	
Manufacturer Supply Code	Samsung 173V
Manufacturer Part Number	173V-TS (Touch Screen)

Supervisory Storage	
Manufacturer Supply Code	Fujitsu DynaMO
Manufacturer Part Number	MCJ3230AP

Supervisory Printer	
Manufacturer Supply Code	HP Color LJ9000 – 10/100 Base TX

2、控制及管理席之裝備規格

作業系統為 MS Windows XP Professional Traditional Chinese，連接方式為 TCP/IP。硬體設備規格如下：

ACER FT100	
Processor	INTEL P4-3.0G
Memory	512 MB
Disk system	40 GB
Ethernet Access	2 ports 10 / 100 MB
CDRW	52X CDRW

C&M Monitor	
Manufacturer Supply Code	Samsung 173V
Manufacturer Part Number	173V-TS (Touch Screen)

C&M Storage	
Manufacturer Supply Code	Fujitsu DynaMO
Manufacturer Part Number	MCJ3230AP

C&M Printer	
Manufacturer Supply Code	HP Color LJ5500 – 10/100 Base TX

3、ATN 路由器 MMI 席之裝備規格

作業系統為 MS Windows XP Professional Traditional Chinese，連接方式為 TCP/IP。硬體設備規格如下：

ACER FT100	
Processor	INTEL P4-2.66G
Memory	512 MB
Disk system	40 GB
Ethernet Access	2 ports 10 / 100 MB
CDRW	52X CDRW

ATN Router MMI Monitor	
Manufacturer Supply Code	Samsung 173V
Manufacturer Part Number	173V-TS (Touch Screen)

(五) 客戶端硬體架構

所謂的客戶端包含 AMHS 系統使用者席位（AMHS user position）和 Client Notebook position。前者所使用的軟體包括：User Agent (MS OUTLOOK and P7 MAPI)和 Intelligent AFTN Terminal application (IAT)。

1、AMHS user position 裝備規格

作業系統為 MS Windows XP Professional Traditional Chinese，連接方式為 TCP/IP。硬體設備規格如下：

ACER FT100	
Processor	INTEL P4-2.66G
Memory	512 MB
Disk system	40 GB
Ethernet Access	2 ports 10 / 100 MB
CDRW	52X CDRW

Client Monitor	
Manufacturer Supply Code	Samsung 173V
Manufacturer Part Number	173V-TS (Touch Screen)

Client Printer	
Manufacturer Supply Code	Futek 80 + Parallel/RS-232C

2、Client Notebook position 的裝備規格

Client Notebook	
Manufacturer Supply Code	IBM R40
Manufacturer Part Number	R40

第四章 AMHS 伺服器安裝程序

壹、建立 AMHS 伺服器使用者帳號

USER :	amss(5000)
GROUP :	amss(5000)
HomeDirectory :	/home/amss
Shell :	/bin/bash2

AMHS 系統建置完成後，目錄應有下列之主目錄，建置包括 AERMAC 應用程式、ATS MESSAGE 伺服器、RMA 模組軟體及 DBMS 伺服器等。

AMHS 系統 CLIENT 端部分，包括 A2M 軟體、IAT 軟體及 UA 軟體等。

/home/amss	.profile (rma 、 aermac 、 amhs)		
	apps/	aermac/	ctu/
			ctuobj/
			aftn/ (bin 、 data 、 cms 、 tools..)
	apps/	amhs/	mxms/
			tools/
apps/	dbms/	data 、 postgre 、 tools 、 etc 、 scripts 等目錄	
apps/	rma/	bin 、 conf 、 dev 、 log 、 dump 、 temp 、 tools 、 jdk 等目錄	

以上目錄之詳細資料，可參考本章節之各應用程式的安裝說明。

貳、AERMAC 應用程式之安裝

AERMAC 應用程式及 AFTN/AMHS GATEWAY 應用程式乃將 aermac_2000_src_v0.2.tgz【目前最新版本】及.aermac_profile 複製至伺服器之硬碟 /tmp/aermac 目錄中。

一、Aermac 應用程式壓縮檔安裝

以【root 身分】登入，解壓縮 aermac 應用程式，並修改 amss、aermac 相關屬性。

```
# mkdir /home/amss/apps/aermac↵          【建立 aermac 目錄】
# cd /home/amss/apps/aermac↵
# tar xzvf aermac_2000_src_v0.2.tgz↵      【解壓縮】
# cp /tmp/aermac/.aermac_profile ↵
# cd /home↵
# chgrp -R amss amss↵                    【amss、aermac 目錄屬性設定】
# chown -R amss amss↵
# chmod -R a+w amss↵                     或 【#chmod -R a+w amss/apps/aermac↵】
# chmod a+x amss/apps/aermac/.aermac_profile↵
# chmod a+x amss/apps/aermac/aftn/tools/.tool_dump.sh
```

二、產生 AERMAC 應用程式 process 之程序

以【amss 身分】登入，並執行以下步驟：

- (一) 檢查.aermac_profile 檔案中，
\$CTUOBJ 環境變數設定是否正確。

```
#mkdir $CTUOBJ↵          【利用此命令檢查】
```

(二) 編譯 CTREE 程式庫及產生所支援之機器碼 makefiles 。

```
#cd $CTU↵  
#make ctree↵          【檢查 libctsglib.a 必須存在於  
$CTUOBJ/lib】  
#make ms_genmk↵
```

(三) 編譯程式庫 。

```
#cd $CTU↵  
#make ms↵  
【若編譯發生錯誤時，make 預設值會停止編譯，可使用 make -i 忽略錯誤，繼續編譯】  
【檢查 aermac 程式庫 ( libms.a , libmscom.a 及 libmsamhs.a ) 必須存在於 $CTUOBJ/lib】
```

(四) process、工具程式之 makefiles 產生及編譯 。

```
#make exe_genmk outil_genmk↵  
#make exe outil↵  
【若編譯發生錯誤時，make 預設值會停止編譯，可使用 make -i 忽略錯誤，繼續編譯】  
【每一個 aermac 應用程式原始碼編譯，如果需要，則必須檢查正確無錯誤】
```

備註：以上完整四步驟，可用下列之指令完成

```
#cd $CTU↵  
#make -i all >res_make 2>&1↵
```

三、 AERMAC 應用程式執行環境產生之程序

(一) 首次安裝 AERMAC 應用程式

在首次安裝 AERMAC 應用程式時，因其執行環境並不存在，則需要產生 AERMAC 應用程式執行環境。

1、AERMAC 應用程式執行環境之產生。

則依據執行下列命令

```
# mkcible.sh ↵
```

環境變數\$SYSPATH 則必須先定義（預設值為 /home/amss/apps/aerac/aftn），此環境變數\$SYSPATH 已被.aerac_profile 檔案定義。

這 mkcible.sh script 工具建立 AERMAC 執行環境整個樹狀目錄結構，其建立目錄開頭乃依據\$SYSPATH 名稱，執行此命令後，以下之目錄必須被產生而存在。

\$SYSPATH/bin :	AERMAC 應用程式之執行檔案。
\$SYSPATH/dump :	當 AERMAC 應用程式終止時【人為停止或 crash】，此 dump 目錄被建立，此目錄資料從 log 移至此目錄。
\$SYSPATH/log :	當 AERMAC 應用程式之目前執行時，儲存所有 process 之日誌及輸出檔案等資料。
\$SYSPATH/tmp :	由 AERMAC 應用程式所產生之暫存檔案。
\$SYSPATH/tmp/fax :	由 FAX process 所產生之暫存檔案及日誌檔案。
\$SYSPATH/tools :	AERMAC 應用程式之工具程式。
\$SYSPATH/data :	AERMAC 應用程式所使用相關聯之資料。
● \$SYSPATH/data/conf :	AERMAC 應用程式所使用之系統設定檔案。
● \$SYSPATH/data/system :	內部 process 溝通用途，其次目錄包含有 mp、pipe 及 sx 等目錄。
● \$SYSPATH/data/traffic :	此目錄儲存所有報文由 AERMAC 應用程式接收及傳送等資料，其次目錄將包含有 days、file、h_stat、months、overflow 及 stat 等目錄。

AERMAC 執行環境一旦建立完成後，mkcible.sh 會自動呼叫另一個 script 工具 **livcible.sh** 立即實行自動搬移所有 AERMAC 應用程式之可執行檔案及工具程式至 \$EXE and \$OUTILS 目錄，以利 AERMAC 應用程式運作。

最後，mkcible.sh 提出下列問題：

```
Shared memory initialization ...  
Are you sure? (y/n) : 回答 n
```

之後，mkcible.sh 工具程式會執行 demarrage.x 命令，將指定

AERMAC 應用程式重新啓動之模式（自動或手動）、終止 AERMAC 應用程式之密碼及重新啓動之 TIMEOUT 等，之後完成此程序後，則進入首次啓動 AERMAC 應用程式之準備。

(二) 非首次安裝 AERMAC 應用程式

如果 AERMAC 應用程式不是首次安裝，則不需要重新建立 AERMAC 應用程式執行環境；在此種情況，AERMAC 應用程式軟體重新產生後，執行下列命令

```
# livcible.sh ←
```

這個 script 工具 livcible.sh 則立即實行自動搬移所有 AERMAC 應用程式之可執行檔案及工具程式至\$EXE 和 \$OUTILS 目錄，以利 AERMAC 應用程式運作；其包括下列之命令：

- 1、vaild_aermac.sh：執行搬移 AERMAC 應用程式之可執行檔案至\$EXE 目錄。
- 2、valid_util_aermac.sh：執行搬移 AERMAC 應用程式之工具程式至\$OUTILS 目錄。

此階段 AERMAC 應用程式之所有可執行檔案及程式已全部產生，並放置於執行環境裡，首次啓動準備。

四、 首次啓動 AERMAC 應用程式之準備

針對 AERMAC 應用程式，在執行環境已經被建立及亦依據先前程序將所有可執行的程式及檔案產生及移至相關目錄後，下列步驟乃針對 AERMAC 應用程式被啓動執行前，有關 AERMAC 應用程式之系統初始環境及靜態設定需被設定。

以【amss 身分】登入，並執行以下各個步驟：

(一) 移除已存在之分享記憶體區塊

script 工具 cleanipc.sh 移除已存在之分享記憶體區塊，包括 shared memory segment、semaphres 及 letterboxes。

```
# cd $CTU ↵  
# cleanipc.sh ↵
```

(二) 檢查在\$CTU/conf 之相關靜態設定檔是 否正確。

檢查在\$CTU/conf 之相關靜態設定檔是否正確；特別注意 cf_affaire.tin、cf_x400.tin 及 cf_option.tin 等。

ENTRY correctly the address for ETH_ADDR_1 142.3.22.240 ETH_ADDR_2 142.250.0.240 (if shall)	cf_affaire.tin
ANORMALLY NO MODIFICATION -AU CIDIN option 0: KOX_CIDIN KOX_SVCID KOX_CYNCI -KOX_SUR_HMT 0 KOX_ROBGX 0 KOX_TLC_CBM 0 KOX_TLC_SVC 0 KOX_TLC_PVC 0 KOX_TLC_TLX 0 KOX_TLC_PSTN 0 KOX_TLC_FAX 0 KOX_TLC_ASY 1 KOX_TLC_LAN 1	cf_option.tin
Check gateway name GATEWAY_NAME 'GATERCTP' MTA_LDAP_PORT 5389 MTA_IP_ADDR= IP ADDRESS OF SORT	cf_x400.tin

(三) 產生靜態設定檔 (*.table)，提供 AERMAC 應用程式讀取。

使用 script 工具程式 vaild_conf.sh 將靜態設定檔 (*.tin) 於\$CONF 目錄產生 (*.table)，提供 AERMAC 應用程式讀取，可參考 AERMAC 應用程式之工具程式。

(四) 重新建立及起始分享記憶體區塊，並建立預設之 IT SUP/A2M 席位 channel

重新建立及起始分享記憶體區塊(shared memory segment、semaphres 及 letterboxes) 並建立預設之 IT SUP/A2M 席位 channel: (creation.x 將提示鍵入 A2M 位址參數)

此工具程式 creation.x 初始分享記憶體區塊及建立第一個連接至 CCS 系統之 IT SUP/A2M 席位，它將要求輸入有關此席位連線形態 (TCP/IP 或非同步方式) 及 channel 相關特性【如果 channel 選擇非同步方式，所選擇之 port 必須正確】

```
# cd $CTU ↵  
# creation.x ↵
```

就會出現以下畫面

```
NETWORK CONFIGURATION INITIALIZATION =  
Do you really want to re-initialize the configuration (y/n)? y  
Starting Initialization.....  
Reinitialization of the channels table (1->TRUE,0->FALSE): 1  
CHANNELS TABLE INITIALIZATION.....  
It is mandatory to create an ITSUP channel  
It can be connected via TCP/IP or Asynchronous line  
Give the telecom to be used : ASY (0) or TCP (1) 1  
Give the primary internet address :142.3.22.238  
Do you want to give a secondary address ? Yes(1) No (0) 1
```

Give the secondary address :142.150.0.235

Give the calling port :3000

Give the called port :3000

OR Give the telecom to be used : ASY (0) or TCP (1) **0**

Give the device name (/dev/term/451) :

OK

Reinitialization of the circuits table (1->TRUE,0->FALSE) : **1**

CIRCUITS TABLE INITIALIZATION.....

OK

Reinitialization of the FE table (1->TRUE,0->FALSE) : 1

OPERATION FUNCTIONS INITIALIZATION.....

OK

Reinitialization of the indicators table (1->TRUE,0->FALSE) : **1**

INDICATORS TABLE INITIALIZATION.....

OK

Reinitialization of the CVP table (CIDIN) (1->TRUE,0->FALSE) : **1**

CVP TABLE INITIALIZATION.....

OK

Reinitialization of the SVC table (CIDIN) (1->TRUE,0->FALSE) : **1**

SVC TABLE INITIALIZATION.....

OK

Reinitialization of the CIDIN indicator table (1->TRUE,0->FALSE):

CIDIN INDICATORS TABLE INITIALIZATION.....

OK

Reinitialization of CIDIN stat counters table (1->TRUE,0->FALSE): **1**

CIDIN STATISTIC COUNTERS INITIALIZATION...

OK

Reinitialization of the priority table (1->TRUE,0->FALSE) :

PRIORITY TABLE INITIALIZATION.....

OK

Reinitialization of the SITA type B priority table (1->TRUE,0->FALSE) : **1**

SITA type B PRIORITY TABLE INITIALIZATION.....

OK

Reinitialization of the robex conf table (1->TRUE,0->FALSE) : **1**

ROBEX CONFIGURATION INITIALISATION...

OK

Reinitialization of wmo indicators table (1->TRUE,0->FALSE) : **1**

WMO INDICATORS TABLE INITIALIZATION...

```
OK
Reinitialization of SITA->AFTN indicators table (1->TRUE,0->FALSE) : 1
AFTN --> SITA INDICATORS TABLE INITIALIZATION...
OK
INDICATORS SITA TYPE B TABLE INITIALIZATION.....
OK
Reinitialization of MD Table (1->TRUE,0->FALSE) : 1
Reinitialization of MF Table (1->TRUE,0->FALSE) : 1
Initialization Done.....
Respond to questions concerning the type of telecommunications support to be employed
(ASY/TCP) for the initial IT SUP channel.
```

(五) 執行 **AERMAC** 應用程式重新啓動檔案 (**\$CONF/demarrage.sys**)

此工具程式 `demarrage.x` 指定 AERMAC 應用程式重新啓動模式（自動或手動）、終止 AERMAC 應用程式之密碼、及重新啓動之 TIMEOUT 等，待此設定完成後，AERMAC 應用程式已可執行啓動程序。

```
# cd $CTU ↵
# demarrage.x ↵
```

`demarrage.x` 將出現以下的提示：

```
restart type: a or m
maximum number of restarts :
time interval for maximum number of restarts:
```

- 1、**a**：AERMAC 應用程式因故停止後，AERMAC 應用程式會自動啓動
- 2、**m**：AERMAC 應用程式因故停止後，AERMAC 應用程式需手動啓動
- 3、**maximum number of restarts**：在自動啓動模式，於允許之時間的期間自動啓動之最大次數 通常設定為 3。
- 4、**time interval for maximum number of restarts**：通常設定為 5-30 分鐘

AERMAC 應用程式允許完全終止執行，其預設密碼為 amss。有關 AERMAC 應用程式啓動或終止程序將於其敘述將於其後。

參、ATS Message 伺服器之安裝

ATS Message 伺服器之執行程式及設定相關資料必須安裝在 AERMAC 應用程式之相同磁碟 partition 中。

ATS Message 伺服器安裝需求包括：

- 需要有/bin/bash2 這個檔案。
- 從光碟機下載 ATS Message 伺服器 release 軟體，進行軟體安裝。
- ATS Message 伺服器應用程式之不同 process 的參數及個別設定。

ATS Message 伺服器安裝程序因太繁雜，而無法在此陳述，需參考下列安裝手冊，進行逐項 process 之設定。

- AMHS OPTION”annex of ICM (Installation、Configuration and Maintance Manual)
- MXMS Server 3.6 for UNIX,installation Manual (E141-INST)【目前最新版本】

肆、RMA 軟體模組安裝程序及設定

一、環境

(一) 描述

此 RMA 軟體安裝於 AMHS 伺服器之相同平台機器上，它使用於 DBMS 及 AMHS 伺服器。

(二) 需求描述

此 RMA 軟體乃全由 JAVA 語言所編寫而成，執行於 Linux 平台。

(三) 磁碟組織

圖 四 - 1 是 RMA 軟體模組之目錄環境，至於 RMA 發展環境則如圖 四 - 2 所示。這二個圖當中的各個目錄的作用如下：

圖 四 - 1 RMA 軟體模組總體環境

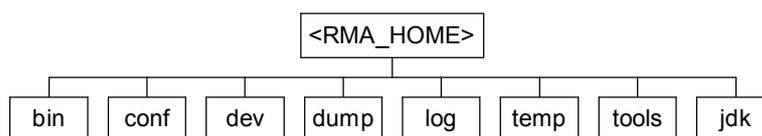
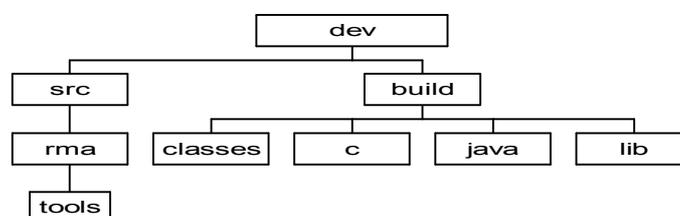


圖 四 - 2 RMA 發展環境 (<RMA_HOME>/dev 目錄)



1、<RMA_HOME>目錄:

bin:	RMA 之可執行檔案
conf:	RMA 設定檔案
dev:	RMA 發展環境

dump:	RMA 依據日期產生之日誌檔案
log:	目前執行之 RMA 日誌檔案
temp:	當 RMA 應用軟體執行時所使用之暫存檔案
tools:	可執行工具程式
jdk:	Java 發展工具

2、<RMA_HOME>/dev 目錄:

src	原始檔案
build	產生之檔案

3、<RMA_HOME>/dev/build 目錄:

classes:	類別 (Class) 檔案
c:	C 語言檔案
java:	產生之 java 檔案
lib:	產生之 jar 檔案及 C 程式庫

4、工具程式

所在目錄	Tool	描述
<RMA_HOME>/bin	rma.sh	RMA 模組 start/stop/status script
<RMA_HOME>/dev/ src/rma/tools	build.sh	Build the RMA application.

二、安裝及啓動程序

(一) JDK (Java Development Kit) 之安裝

- 1、<RMA_HOME> 目錄通常為/home/amss/apps/rma，若此目錄不存在，請先建立此目錄。
- 2、JDK 安裝於/home/amss/apps/rma/jdk。
- 3、解壓縮 JDK 檔案，進行安裝。

上述的所有步驟必須以【amss 身分】登入之後才能進行安裝，至於相

關的軟體則是從 CD-ROM 將 (jdk.zip 或 jdk.tgz) 解壓縮至 <RMA_HOME> 目錄。

```
#login amss↵          【以 amss 身分登入】
#cd /home/amss/apps/rma↵      【變更目錄至/home/amss/apps/rma】
#tar xzvf /mnt/cdrom/jdk.tgz↵  【解壓縮 JDK 檔案】
```

(二) RMA 模組二進制檔案之安裝

必須以【amss 身分】登入才能進行安裝。首先變更工作目錄至 /home/amss/apps/rma，然後再執行以下之步驟：

- 1、從 CD-ROM 將 RMA 模組二進制檔案解壓縮至 <RMA_HOME>目錄。

```
#login amss↵          【以 amss 身分登入】
#cd /home/amss/apps/rma↵      【變更目錄至/home/amss/apps/rma】
#tar xzvf /mnt/cdrom/rma-bin_vY.XX.tgz↵
或                               【解壓縮 RMA 模組二進制檔案】
#unzip /mnt/cdrom/rma-bin_vY.XX.zip↵
```

- 2、更改 RMA 啟動 SCRIPT 之權限。

```
#cd /home/amss/apps/rma/bin↵
#chmod a+x rma.sh↵
```

(三) RMA 模組原始檔案之安裝

必須以【amss 身分】登入才能進行安裝。首先變更工作目錄至 /home/amss/apps/rma，然後再執行以下之步驟：

- 1、從 CD-ROM 將 RMA 模組二進制檔案解壓縮至 <RMA_HOME>目錄。

```
#login amss↵
#cd /home/amss/apps/rma↵
#tar xzvf /mnt/cdrom/rma-src_vY.XX.tgz↵
或
#unzip /mnt/cdrom/rma-src_vY.XX.zip↵
```

2、更改 RMA Build SCRIPT 的權限。

```
#cd /home/amss/apps/rma/dev/src/rma/tools↵  
#chmod a+x build.sh↵  
#chmod a+x ant/bin/ant
```

3、產生 RMA 應用程式。

```
#cd /home/amss/apps/rma/dev/src/rma/tools↵  
#./build.sh distribution↵
```

(四) RMA 模組之啟動及終止

在啟動 RMA 軟體模組前，必須先指定 RMA 模組之設定參數。

RMA 模組之啟動及終止，必須於<RMA_HOME>/bin 目錄執行。

- 啟動 RMA 模組，執行 ./rma.sh start。
- 終止 RMA 模組，執行 ./rma.sh stop。
- 獲得 RMA 模組狀態，執行 ./rma.sh status。
- 獲得 RMA 模組版本，執行 ./rma.sh version。
- 重新啟動 RMA 模組，執行 ./rma.sh restart。

三、RMA 軟體模組設定程序

在<RMA_HOME>/conf 目錄，編輯 rma-service.xml 檔案，每次修改設定參數後必須重新啟動 RMA 模組，每個 mbean 使用一個 mbean 標籤來儲存它的設定資料，該 mbean 標籤包含 mbean 的名字及它的 java class 名字 (code 碼)。

(一) RMIRegistry 參數

- Name : com.thalesis.rma:service=RMIRegistry
- Code : com.thalesis.rma.services.RMIRegistryService

Attribute Name	Default Value	Comments
ListenPort	9999	This is the port on which the registry is listening

(二) JMXConnector 參數

- Name : com.thalesis.rma:service=JMXConnector

- Code : com.thalesis.rma.services.JMXConnectorService

Attribute Name	Default Value	Comments
ServiceURL	Service:jmx:rmi:///jndi/rmi://<localhost>:9999/server	This is the URL on which the JMX connector accept connection

(三) DatabaseService 參數

- Name : com.thalesis.rma:service=DatabaseService ◦
- Code : com.thalesis.rma.services.DatabaseService ◦

Attribute Name	Default Value	Comments
Driver	org.postgresql.Driver	The JDBC driver
ServerURL	jdbc:postgresql://127.0.0.1:5432/amss_db	The URL of the database
User	amss	The user of the database
Password		The user password

(四) DirectoryManager 參數

- Name : com.thalesis.rma:type=DirectoryManager ◦
- Code : com.thalesis.rma.providers.directory.DirectoryManager ◦

Attribute Name	Default Value	Comments
ProviderURL	Ldap://<dir_host>:5389	The URL of the directory
Principal	Cn=diradmin, ou=AGENT, ou=MTA-CH, o=localadmin, c=CH	The user of the directory
Credentials	ADMIN	The user password.

伍、DBMS 伺服器安裝程序及設定

一、環境

(一) 描述

此 DBMS 伺服器軟體安裝於 AMHS 伺服器之相同平台機器上。

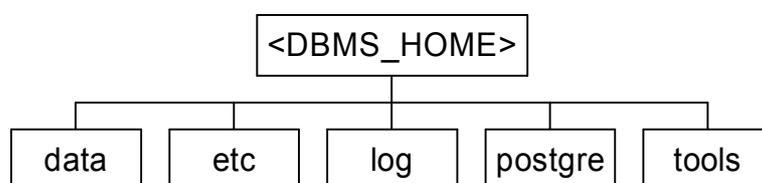
(二) 需求

無。

(三) 3.磁碟組織

圖 四 - 3 為 DBMS 伺服器之目錄環境，至於各個目錄的作用請參看表 四 - 1。

圖 四 - 3 DBMS 一般環境



圖一: DBMS 一般環境

data:	資料庫檔案
etc:	Dbms 設定檔案
log:	dbms 目前之日誌檔案
postgre:	資料庫管理伺服器
tools:	可執行工具程式

表 四 - 1 <DBMS_HOME> 底下的目錄的作用

(四) 工具程式

所在目錄	Tool	描述
Tools	dbms	DBMS server start/stop/status script.

二、安裝及啓動程序

(一) DBMS 伺服器之安裝

1、<DBMS_HOME> 目錄通常爲
/home/amss/apps/dbms，若此目錄不存在，建立此目錄。

2、解壓縮 dbms_vY.XX.tgz。

請以【amss 身分】登入之後進行安裝。先從 CD-ROM 將
(dbms_vY.XX.tgz) 解壓縮至<DBMS_HOME>目錄。

```
#login amss↵  
#cd /home/amss/apps/dbms↵  
#tar xzvf /mnt/cdrom/dbms_vY.XX.tgz↵
```

3、進行安裝

登出後，再以 amss 使用者帳號登入，以獲得新的環境變數，並進行安裝。

```
#logout↵  
#login amss↵  
#cd /home/amss/apps/dbms/tools↵  
#./install_dbms↵
```

(二) DBMS 伺服器之啓動及終止

在啓動 DBMS 伺服器前，必須先檢查 DBMS 之設定參數。DBMS 伺服器之啓動及終止，必須於<DBMS_HOME >/tools 目錄執行。

- 啓動 DBMS 伺服器，執行 ./dbms.sh start。
- 終止 DBMS 伺服器，執行 ./dbms.sh stop。
- 獲得 DBMS 伺服器狀態，執行 ./dbms.sh status。
- 獲得 DBMS 伺服器版本，執行 ./dbms.sh version。
- 重新啓動伺服器，執行 ./dbms.sh restart。

三、DBMS 設定程序

在<DBMS_HOME>/etc 目錄，編輯 dbms.cfg 檔案，每次修改設定參數後必

須重新啓動 DBMS 伺服器。

Attribute Name	Default Value	Comments
DB_NAME	amss_db	This is the name of the database
DB_ADMIN	amss	This is the administrator user name
DB_PORT	5432	This is the TCP port on which the DBMS server is listening.
DB_USER	rma	

陸、A2M 應用程式安裝程序

一、A2M 應用程式需求

(一) 硬體需求

- 1、256Mbytes 以上記憶體
- 2、建立於雙網路架構

(二) 作業系統需求

- 1、Windows 版本：Windows NT4.0 (SP6)、Windows 2000 及 XP
- 2、Redhat：Redhat 6.1 或以上

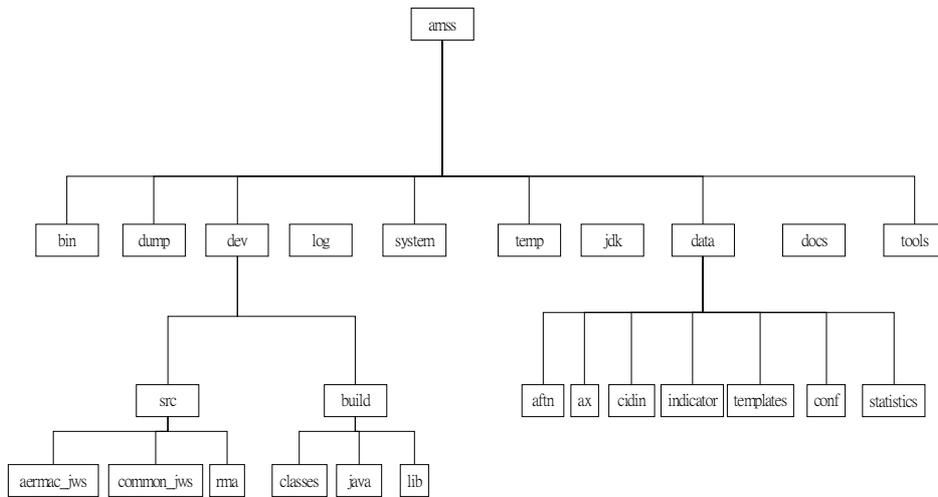
二、A2M 應用程式

(一) 環境

1、磁碟編制

<A2M_HOME>目錄：A2M 應用程式運作環境，爲 A2M 應用程式執行之目錄，預設爲 C:\a2m。

2、A2M 應用程式總體目錄



(1) <A2M_HOME>目錄：

- bin：A2M 應用程式可執行之檔案。
- data：所有需要執行 SUP 應用程式之所有檔案被拷貝至此目錄的次目錄下。當在 PC 的 SUP 應用程式版本需要更新時，這個目錄之內容將被使用。
- dev：A2M 應用程式之發展環境目錄。
- dump：A2M 應用程式依日期儲存日誌檔。
- log：目前 A2M 應用程式之日誌檔。
- system：A2M 應用程式系統檔案及應用程式。
- temp：A2M 應用程式執行時所使用之暫時檔案。
- tools：A2M 應用程式操作工具。
- jdk：JVA 發展環境工具
- docs：輔助文件檔案（HTML 格式）

(2) <A2M_HOME>\dev 目錄：

- src：A2M 應用程式之原始程式檔案。
- build：A2M 應用程式 generated 檔案。

(3) <A2M_HOME>\dev\build 目錄：

- classes：class 檔案。
- java：generated java 檔案。
- lib：generated jar 檔案。

(4) <A2M_HOME>\data 目錄：

- aftn：AFTN Proforma（來自 Creation menu）。
- ax：AX 檔案（Address book）。
- cidin：CIDIN Proforma（來自 Creation menu），未來不使用。
- Indicator：Indicator 檔案（Address book）。
- conf：Configuration 設定檔案。
- Templates：templates 檔案。
- statistics：統計檔案。

3、A2M 應用程式工具

Source 目錄	工具	說明
bin	Start.sh	Linux：A2M 應用程式之執行 【不使用】
	Start.bat	Windows：A2M 應用程式之執行
dev/src/aermac_jws/tools	Build.sh	Linux：A2M 應用程式之建立 【不使用】
	Build.bat	Windows：A2M 應用程式之建立

三、A2M 應用程式安裝程序

(一) 建立 a2m 帳號

以【Windows 之 Administrator】登入，建立一個 a2m 帳號使用者，步驟如下：

- 1、開始 → 控制台 → 使用者帳號 → 建立新帳號
- 2、新帳號名稱爲 a2m 密碼爲 a2m

(二) A2M 應用程式安裝

1、建立<A2M_HOME>目錄

使用者以 a2m 帳號登入 Windows，並於 c:\ 建立一個 a2m 子目錄，c:\a2m 將被視爲<A2M_HOME>目錄

```
C:>mkdir a2m
```

2、安裝 JDK

解壓縮光碟中 JDK 檔案（jdk.zip）至 C:\a2m 目錄，JDK 將被安裝

在 C:\a2m\jdk 目錄。

3、安裝 A2M 應用程式

解壓縮光碟中 A2M 檔案(a2m-bin_vn.nn.zip)至 C:\a2m 目錄，A2M 所有目錄及檔案皆被建立。

4、建立 A2M 應用程式執行捷徑

在 C:\a2m\bin 目錄，選擇 start.bat 檔案，按右鍵後選擇建立捷徑至桌面。

四、A2M 應用程式設定

(一) 設定 SUP 席位螢幕解析度

SUP 席位之螢幕解析度最低應為 1024x768，若為 TFT LCD 則更佳。步驟如下：桌面按右鍵 → 內容 → 設定值，設定螢幕解析度，至少要設成 1024x768 的解析度。

(二) 設定 TIME ZONE 為格林威治時間

時間設定使用者需設定正確之 TIME ZONE，為格林威治時間。步驟如下：桌面右下角時間按右鍵 → 時區 → 設定為格林威治時間。

(三) 鍵盤設定，若為 Windows 2000 則不需設定。

(四) 時間同步設定。

此項目乃針對 Windows 2000 的作業系統，目的是修改 c:\a2m\bin\ntpconfiguration.reg，步驟如下：

- 1、設定 NTPServer 參數為所屬 NTP Server IP 位址。
- 2、設定 Period 參數為每天時間同步之次數(值為 16 進制)。
- 3、存檔。
- 4、雙擊此檔及回答”是”增加至 registry。
- 5、我的電腦右按 → 管理 → 服務及應用程式 → 服務
- 6、尋找 “Window Time”
- 7、“Window Time”右按 → 內容 → 啟動類型
- 8、定為自動 → 啟動此服務

五、 A2M 應用程式之啟動

(一) 啟動 A2M 應用程式

雙擊桌面之 A2M 應用程式執行之捷徑

(二) 登入 A2M 畫面

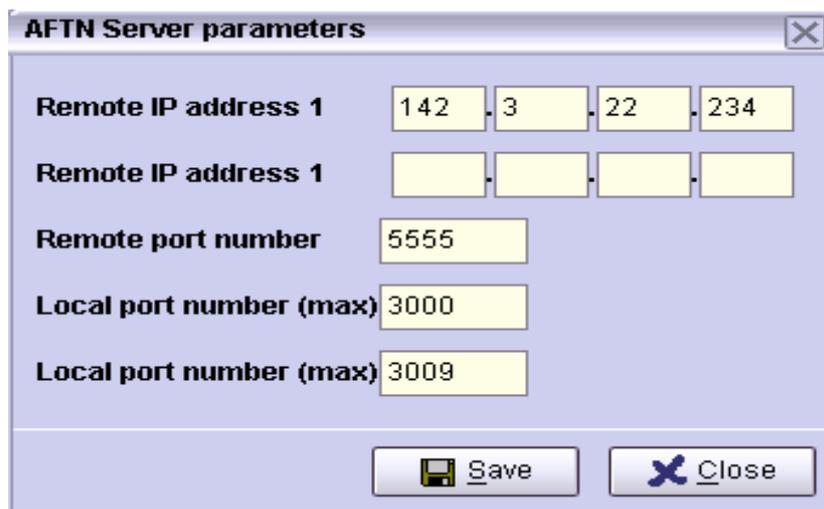
依據下列數據，鍵入正確之 USER Name 及 Password

User Name	SUPER
Password	EXTRA



需注意下列事項：

- 1、右下角連線燈號是否為綠燈，綠燈表示網路正常，若紅燈則將指標移至此處雙擊，修改 AFTN Server parameters，如下圖。



- 2、登入程序一直無法進入，可能必須檢查 RMA Parameters 是否正確，修改完成存檔後，再重新啓動 A2M 應用程式。



RMA Parameters

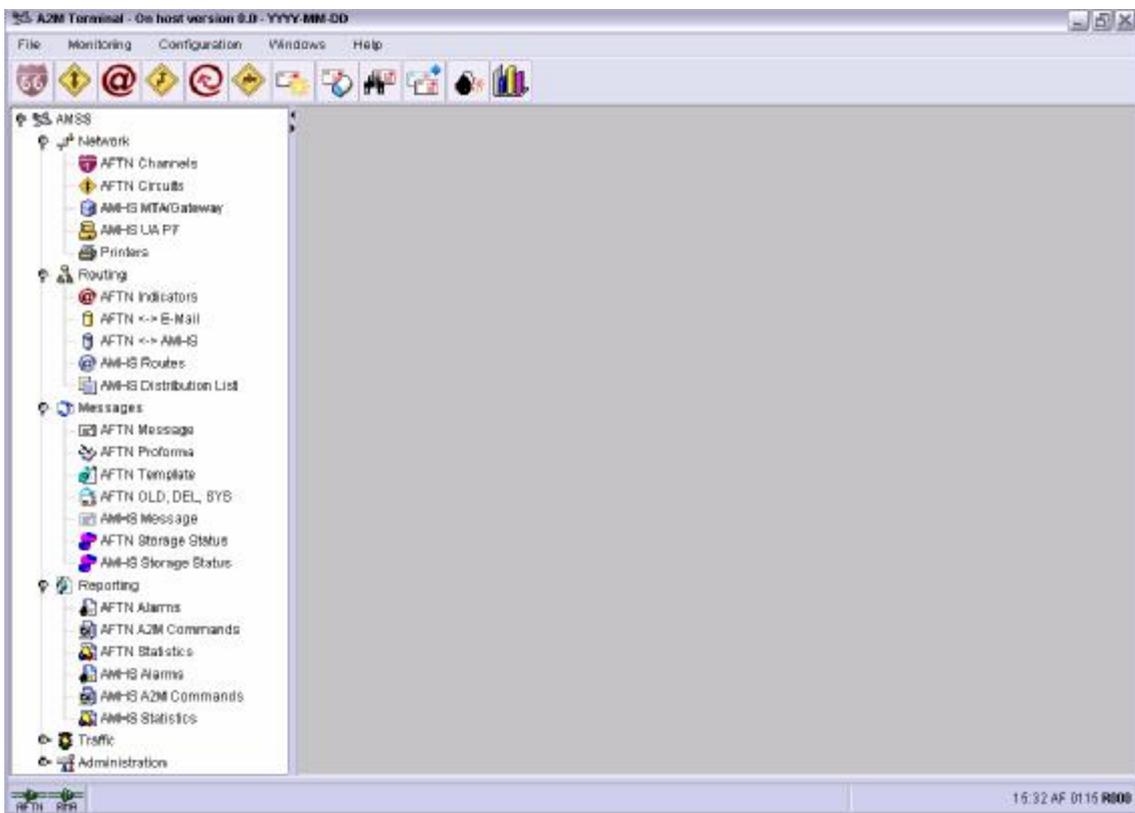
Name: Local

Url 1: service:jmx:rmi:///jndi/rmi://142.3.22.234:9966/server

Url 2:

Save Cancel

(三) A2M 登入成功畫面



柒、AMHS 伺服器環境變數、靜態及動態設定檔

一、AMHS 伺服器環境變數

當 AMHS 伺服器之相關應用程式及工具乃使用環境變數進行安裝產生及執行，故環境變數必須正確地設定。

其中 AERMAC 應用程式環境變數定義於

`/home/amss/apps/aermac/.profile` 檔案，當需要使用環境變數，要在環境變數前加上一個\$字元。

下列表格之環境變數需在安裝時其定義並被使用於產生 AERMAC 應用程式。

環境變數	說明
\$CTU	AERMAC server 原始程式碼之發展環境目錄
\$CTUOBJ	AERMAC server 程式碼及執行程式之發展環境目錄
\$UTIL	針對 AERMAC server 之發展環境目錄之可用工具 \$UTIL/util_dev/
\$SYSPATH	AERMAC server 運作的環境目錄 /home/amss/apps/aermac/aftn/
\$EXE	AERMAC 運作的 processes 之可執行程式之家目錄 \$SYSPATH /bin/
\$OUTILS	AERMAC 運作的可用工具之可執行程式之家目錄 \$SYSPATH/tools/
\$DATA	由 AERMAC server and AFTN/AMHS Gateway 所參考引用之運作的檔案及目錄 \$SYSPATH/data/
\$CONF	AERMAC server 應用程式靜態設定檔案之家目錄 (內部運作之格式): \$DATA/conf/
\$TRACE	由 process 所產生之日誌檔及 trace 檔案之家目錄

	\$SYSPATH/log/
\$DUMP	存放由 AERMAC server and AFTN/AMHS Gateway 先前軟體當機之 dump 檔案之家目錄，給軟體工程師分析用 \$SYSPATH/dump/
\$DAT_PATH	磁帶裝置之路徑(如果來源須從磁帶讀取則將被使用)
\$OSIROOT	此目錄名稱包含 ATS Message 伺服器可執行程式及 ATS Message 伺服器管理子目錄之主目錄。 /home/amss/apps/amhs/mxms/exe

AERMAC 應用程式之 AFTN message switching 元件之執行需要一個環境變數。\$KCF_INDCENT 必須包含 AFTN indicator 跟 center 一致。

所有環境變數以預設定義，且大部分不需修改。

全部環境變數清單可於 .profile 檔案中獲得且可修改；另可執行下列命令：env | more，以獲得目前環境變數設定之情形。

```
#env | more
```

二、靜態及動態設定檔

(一) ATS Message Server:

1、“static”靜態設定檔於 ATS Message 伺服器安裝期間，依據安裝手冊設定，可參考下列文件：

(1) AMHS OPTION Annex of ICM (Installation、Configuration and Maintance Manual)

(2) MXMS Server 3.6 for UNIX,installation Manual (E141-INST)【目前最新版本】

2、“dynamic”動態設定之參數與 MTAs, UA users, MS 及 X.500 目錄所關聯組合而成，且參數可在 A2M 位置藉由

ATS Message Server 功能進行修改，有關這些功能之詳細說明，可參考下列文件：

(1) Supervisory Operator Manual

(二) AFTN Message Switching (AERMAC) and AFTN/AMHS Gateway 應用程式

1、AERMAC 應用程式之動態設定

(1) operators 在 supervisory 位置初始及更新所有動態設定之參數。

(2) 所有存取 AFTN 網路之 circuits 及 channels 用戶之設定參數。

(3) Routing 參數允許交談與 AFTN indicators。

(4) 針對區域所連接之 supervisory 終端及 operator 印表機至通信中心的連接設定參數。

(5) 目前等候傳送之 messages 佇列。

(6) 所有動態參數儲存及更新皆在分享記憶體區塊，此區域為所有 AERMAC and AFTN/AMHS Gateway processes 有權存取；這些分享記憶體區塊之最新副本保存至 AERMAC 應用程式 (server) 磁碟檔案中。

2、AERMAC 應用程式靜態設定

靜態設定之資料組成爲一組應用程式參數，它們在 AERMAC 應用程式及 Gateway 安裝時期就已被鍵入與設定，它們大部分都被接受，只有極少數因特定需求需要修改，這類參數包括：

- AERMAC 應用程式之主要功能設定。
- 基本設定，關於應用程式之固定的通信參數。
- incoming AFTN 報文分析之方法及拒絕報文之標準之設定。
- 警示的處置及處理模式之設定。
- 靜態設定資訊被儲存於一組磁碟檔案，當參數需要修改，這些必須於系統終端機使用文字編輯器手動修改。

(1) 下列程序為修改靜態設定資料程序之說明：

- 首先針對特定需求的檔案 (text version) 修正其值存檔，然後使用系統工具程式("valid_conf.sh")將此修正後之 text version 檔案，轉換成二進制檔案，此二進制檔案格式將於下次應用程式啟動時被讀取後生效。
- 完整之靜態設定檔案，及它們的用途，可查閱 "Application Configuration" section of the "CCS Application" chapter of the Installation, Configuration and Maintenance Manual.
- 可修改之 text versions 檔案存放於/home/apps/amss/aermac/conf 目錄，它們的名稱以 cf_開頭，及它們 extension 的名稱為*.tin，可以使用 text editor (vi) 修改 該特定需求之檔案 (*.tin)。
- 每一個.tin 檔案都相關聯一個相同名稱的.dd 檔案(extension 名稱)，這些資料說明定義檔案指定相關聯之*.tin 檔案所有參數之格式、可能型態欄位資料說明及使用之文法（下列表格舉例說明，在此例子中，一個欄位是字元或長度最大為 30 之字串或為正常大小之二元碼(包含 15 bits)或是一個大的二元碼 (包含 31 bits))，這些*.dd 檔案正常情況並不需要修改。

cf_filename.dd	cf_filename.tin
fields: FIELD1 char;	/ =FIELD1 'B'
FIELD2 char(30);	=FIELD2 'abcdefgh'
FIELD3 binary(15);	=FIELD3 675
FIELD4 binary(31);	=FIELD4 5000000
End;	

- 針對需要修改之 text version 檔案後，每個相同名稱之.tin 檔案及.dd 檔案成雙成對產生一個相同名稱之.table (extension 名稱)；這些被產生的*.table 檔案被放置在目錄 (\$data/conf)。
- 此外，最重要的是，要使新產生之*.table 檔案生效，必須將 AERMAC 應用程式停止後，再重新啟動 AERMAC 應用程式，由 AERMAC 應用程式在啟動期間讀取這些檔案的內容後生效。
- 但僅此下列兩個檔案 (cf_mdp.table 及 cf_menu.table) 乃修改後即時生效，不需重新啟動應用程式。

(2) 下列程序為 valid_conf.sh 程序之說明：

- 使用系統工具程式 valid_conf.sh 將 *.dd 及 *.tin 成雙成對產生新的*.table。

- 如果執行時跟隨一個參數(檔案名稱不需要 extension 名稱)，只有這特定檔案被產生，如果執行時無參數，則目錄裡所有*.dd 及*.tin 成雙成對產生新的 *.table 檔案。請參看圖 四 - 4 以及圖 四 - 5。注意：圖 四 - 4 之中跟在 valid_conf.sh 之後的引數為一個檔案名稱；而圖 四 - 5 中的 valid_conf.sh 之後並沒有任何引數，這就表示執行的對象是目錄中的所有檔案。

圖 四 - 4 Validating 單一 AERMAC configuration 檔案

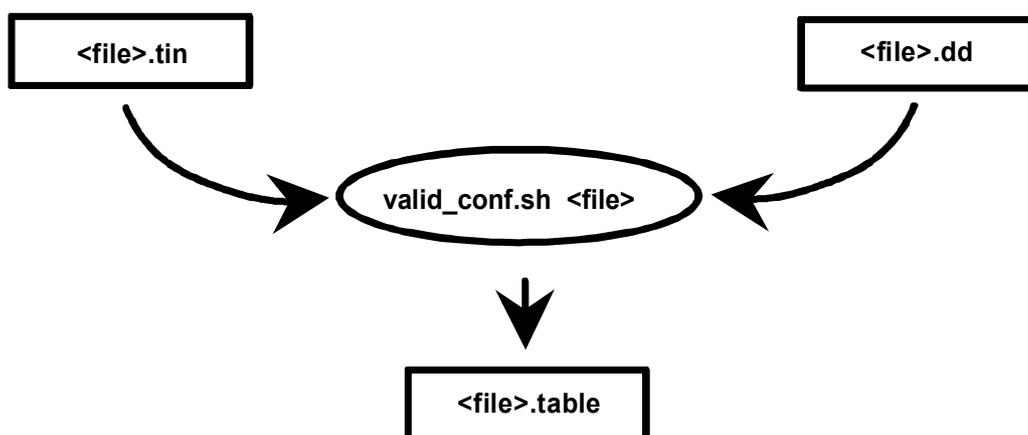
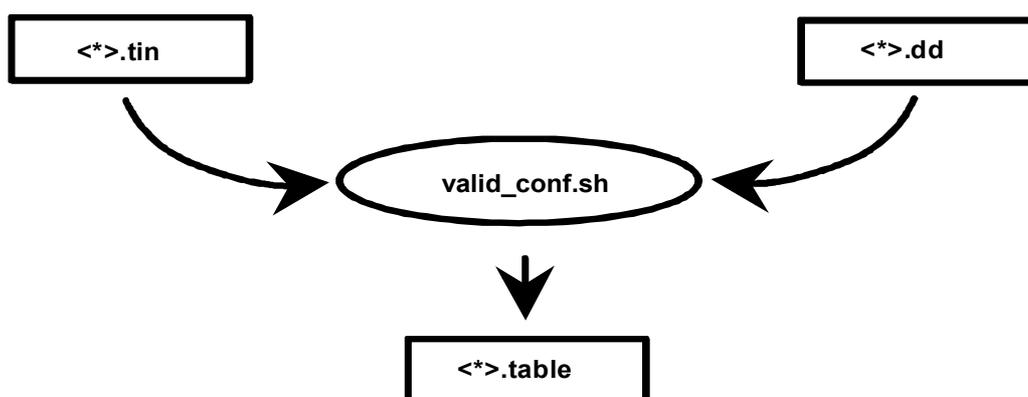


圖 四 - 5 Validating 所有 AERMAC configuration 檔案

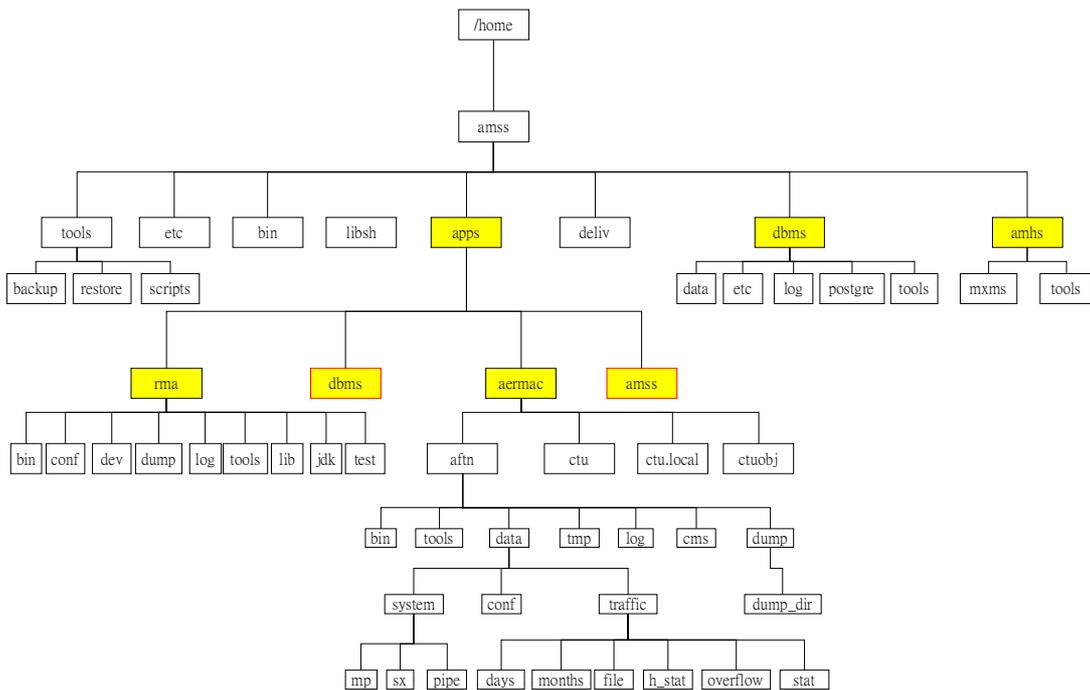


(3) 列出一些較常修改之靜態設定檔案：

檔案	功能
cf_affaire	連接埠設定之定義、伺服器及 cluster address 資訊、一些系統參數最大值

cf_al_definition	警示設定：severity、警示至 A2M、警示至印表機、有無 beep 警示等
cf_ana_antse	針對 incoming AFTN 報文分析、問題報文之拒絕、警示號碼, SVC 種類之反應”之規則
cf_auf_actse	針對 incoming AMHS 報文分析、問題報文之拒絕、警示號碼”之規則
cf_exp_err	由伺服器將問題報文之 texts 由伺服器返至 to A2M 席位 for display
cf_option	系統選項：certain messages hard copy，例如. SS, SS Ack, SVC, expired; X.400 and X.500 Directory options
cf_x400	本地 AFTN/AMHS Gateway 名稱、ldap 埠號碼、最大的 AMHS 報文大小處理

捌、伺服器磁碟目錄組織



一、AERMAC 目錄組織

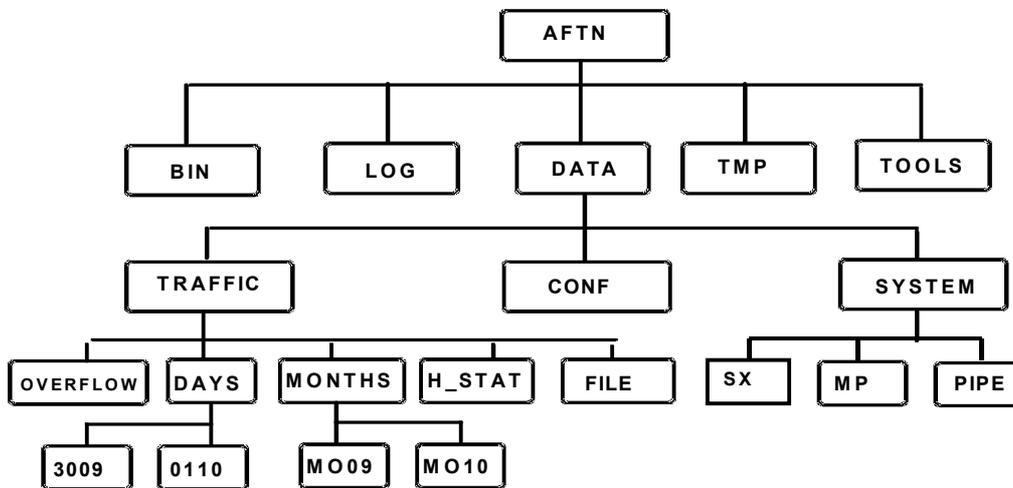
AERMAC (AFTN message switch and AFTN/AMHS Gateway)應用程式之

目錄為/home/amss/apps/aermac :

(一) AERMAC (AFTN message switch and AFTN/AMHS Gateway)應用程式執行環境之目錄

執行環境之目錄(\$SYSPATH)，子目錄 aftn 之次目錄如圖 四 - 6 所示，請注意：其中的 3009, 0110, MO09 及 MO10 僅是例子，此為日誌檔產生之目錄。

圖 四 - 6 AERMAC 執行環境目錄



1、aftn 目錄下之次目錄：(其 cms 及 dump 目錄不在圖 四 - 6 當中顯示)

bin:	參照環境變數\$EXE, 包含 AERMAC 應用程式 and AFTN/AMHS Gateway processes 可執行之檔案
log:	參照環境變數\$TRACE, 包含由目前 AERMAC 應用程式之 process 產生之所有日誌及 trace 檔案
data:	由 AERMAC 應用程式及 AFTN/AMHS Gateway 所參照之執行中檔案之樹狀結構
tmp:	暫存檔案
tools:	參照環境變數\$OUTILS, 包含所有 AERMAC 應用程式 and AFTN/AMHS Gateway 之工具程式

cms:	CMS system 所使用之目錄
dump:	存放由 AERMAC 應用程式 and AFTN/AMHS Gateway 先前軟體當機 (人為或 CRASH) dump 檔案之家目錄，提供軟體工程師分析用

2、aftn / data 目錄下之次目錄

traffic:	此目錄存放有關 traffic:包括報文、overflows 及統計資料
conf:	此目錄存放有關 AERMAC 應用程式之靜態設定資料 (在內部運作格式)
system:	此目錄包含所有 inter-process 溝通訊息. (share memory、pipe file 等)

3、aftn / data/traffic 以下之子目錄

overflow:	此目錄存放在 circuit overflow 之所有報文
days:	此目錄依照日期建立目錄 ("DDMM")，每個目錄代表該日 traffic 資料，包含報文檔案、統計資料、警示及 supervisory commands 等資料
months:	此目錄依照月份建立目錄 (named "MOMM")，每個目錄代表該月報文統計資料
h_stat:	此目錄包含由伺服器維護之每小時統計資料
file:	此目錄檔案乃是儲存所接收到異常之長報文.
stat	

4、aftn / data/system 下之次目錄

sx:	此目錄檔案包含儲存伺服器應用程式之 semaphore
mp:	此目錄檔案包含儲存伺服器應用程式之 shared memory segments
pipe:	此目錄包含 pipes (FIFOs)檔案名稱

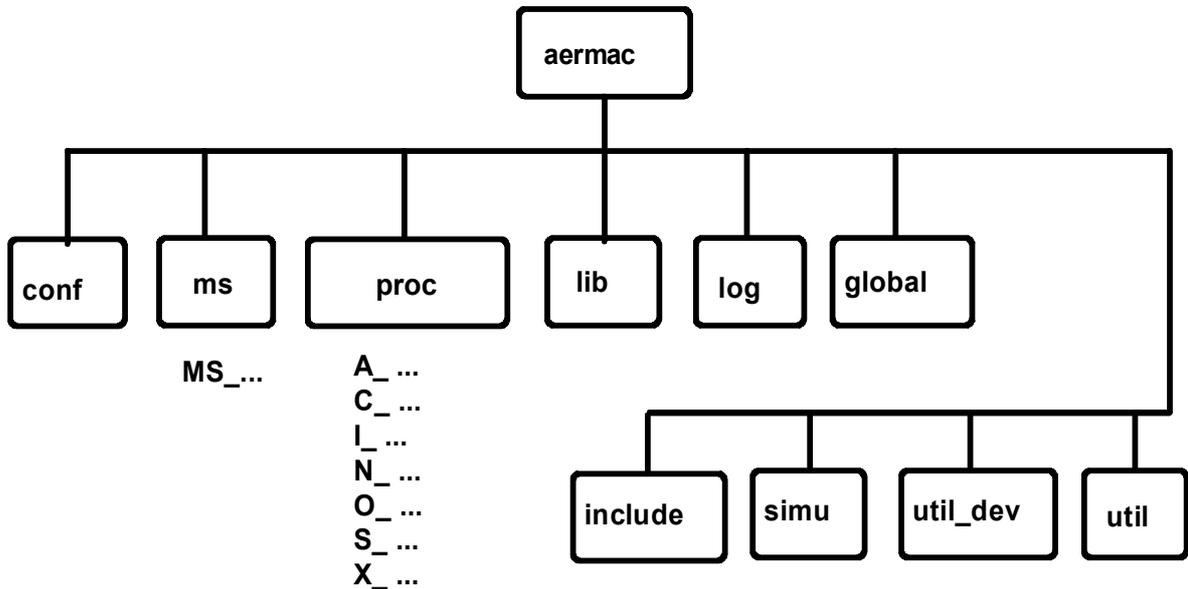
(二) AERMAC (AFTN message switch and AFTN/AMHS Gateway)應用程式發展環境之目錄

AERMAC 應用程式發展環境建立完成，THALES-IS 公司僅能介入 AERMAC 應用程式之軟體修改維護部分，也就是 AFTN Message Switch 及 AFTN/AMHS Gateway 應用程式，並沒有提供 ATS Message Server 軟體之

發展環境。

發展環境目錄起自於 aermac 次目錄，請參看圖 四 - 7。注意：圖 四 - 7 之中的 ms 及 proc 不是子目錄之字面意義上的名字；ms 表示其子目錄之出現的檔案目錄成員名稱開頭為 ms_ ；而 proc 表示其子目錄之出現的檔案目錄成員名稱開頭為一個字母加上底線，以包含其不同 processes 之起源。

圖 四 - 7 AERMAC 發展環境目錄



子目錄/home/amss/aermac 以下之次目錄

conf:	設定檔規格之工作匣 (edit formt)
ms_*:	不是工作匣名稱,但代表包含一些組合針對所支援機器之原始檔案；其子工作匣之名稱以 ms_開頭，即每一個機器函數有一個工作匣，因所支援機器有許多，在此僅列出一些例子
ms_circ	circuit 處理
ms_ind	AFTN indicator 處理
ms_sf	系統檔案處理
ms_smem	系統記憶體服務
ms_stlcm	系統通信(telecomm)處理
ms_tmes	message table 處理
ms_to	time-outs 處理

	(完整之 ms_list 可查閱在 "CCS Application" 章節之 Installation, Configuration and Maintenance Manual)
proc:	不是工作匣名稱,但代表包含一些組合包含 processes 之原始檔案;其子工作匣之名稱以一個英文字元加上底線,如下列
a_*:	搜尋 processes, e.g. a_rch, a_rtx
c_*	通信 (telecommunications) processes, e.g. c_tcp
i_	介面 processes
m_	多元協定 processes
n_	CIDIN communications (CIDIN systems only)不使用
o_	message routing and duplicating processes
s_	CIDIN operator service messages (CIDIN systems only)
x_	supervisory processes
	(完整之 process 原始目錄可查閱在 "CCS Application" 章節之 Installation, Configuration and Maintenance Manual)
lib:	產生支援機器程式庫 (參考至\$CTUOBJ)
log:	工具程式使用之暫存檔案
global:	原始程式參考之全域 include 檔案
include:	一群對所有應用程式之 include 檔案的 symbolic links
simu:	針對應用程式功能模擬之工具程式
util_dev:	發展工具程式
util	執行運作之工具程式
ctree:	CTREE 原始檔案目錄 (程式庫管理 ISAM files, 不顯示於圖中)

環境變數名稱 \$CTUOBJ 被定義,其所產生 object 檔案及執行檔,就放在磁碟中的 \$CTUOBJ 區域,預設將位於這此區域之子目錄檔案都將被編譯。

有時受限於系統容量,須暫時將眾多之 objects 及 可執行檔之原始儲存區域轉存至其他磁碟區域。(因目前系統磁碟空間已足可容納發展環境所需,故不需搬移)

\$CTUOBJ 之預設值為 \$CTU。

建議是在子目錄 \$SYSPATH 中建立一個次目錄,名稱爲 ctobj 及設定 \$CTUOBJ=\$SYSPATH/ctobj。

\$CTUOBJ 子目錄下之次目錄：

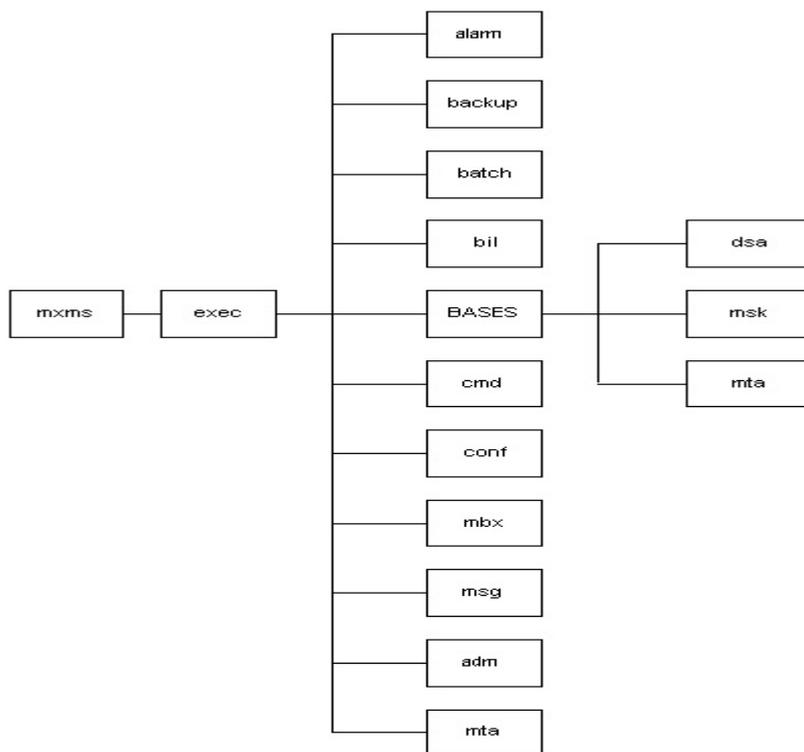
ms_*:	(參考先前解釋) 此 object 檔案來自支援機器原始編譯過，一個機器包含一個工作匣
proc:	(參考先前解釋) object 及 可執行檔案來自編譯 process 原始檔案，一個 process 包含一個工作匣
lib:	產生支援機器程式庫(AMHS: libms.a, libmscom.a, libmsamhs.a; CTREE: libctsglib.a)
simu:	針對應用程式功能模擬之工具程式
util_dev:	發展工具程式 object 及 可執行檔案
util	運作執行工具程式 object 及 可執行檔案
ctree:	CTREE object 檔案.

二、ATS Message 伺服目錄組織

ATS Message Server (MTA, MS and DS)利用從名稱爲 mxms 之子目錄以下之檔案及目錄，其實際位置選擇於安裝時期，可能爲/home/amss/amhs/mxms。

在此樹狀架構（請參考圖 四 - 8），環境變數名稱 **\$OSIROOT** 被指派至子目錄 **mxms/exec**，**\$OSIROOT** 是針對 ATS Message Server 之執行環境之樹狀目錄之起始點。

圖 四 - 8 ATS Message 伺服器目錄



mxms:	一組交付之 ATS Message Server 檔案，針對伺服器安裝檔案及設定檔案。 ATS Message Server 目錄之起始點。
-------	---------------------------------------------------------------------------

mxms/ exec:	- ATS Message Server 可執行檔案 (processes: agt cms cnv dsa dua ibo lla ltmd mce mds msm mta trf wan) - 也就是 \$OSIROOT
----------------	---------------------------------------------------------------------------------------------------------------------

\$OSIROOT (mxms/exec)子目錄以下之子目錄

alarm:	來自伺服器警示訊息產生日誌檔案
BASES:	子目錄包含 MTA, MS and DSA DataBases
backup:	每日 backup 命令自動產生 backup 檔案
batch:	啟動、停止、backup 及目錄管理命令之 shell scripts
bil:	伺服器產生之 billing 檔案
cmd:	operator 命令 scripts
conf:	針對伺服器 process 之設定檔案
log:	來自伺服器產生日誌檔案

mbx:	Process 間起始 mailboxes 溝通
msg(e):	檔案包含 administrative error message texts: msg : 法文 - msge : 英文
users:	使用者程式連結至伺服器產生或查閱之檔案.

\$OSIROOT / BASES 子目錄以下之子目錄

dsa:	DSA DataBases 目錄
msk:	MS DataBases 目錄
mta:	MTA DataBases 目錄.

玖、AMHS 伺服器 PROCESS 檢視程序

AMHS 伺服器程序可分為四個主要部份：

一、AFTN message switching and AFTN/AMHS Gateway 應用程式 PROCESS 之檢視

AERMAC 應用程式之 process 名稱清單列於下表中，當以 amss 使用者登入後，鍵入 pu 命令時，在螢幕上可列出正執行中之 process 名稱。

pu_a_rch*:	針對搜尋報文、警示及 supervisory commands 之處理
pu_a_rtx:	針對特定 TEXT 字串搜尋報文之處理
pu_c_asy	經由 Asynchronous line 之 telecommunications 之處理
pu_c_fip	經由 FIP 之 telecommunications 之處理
pu_c_tcp	經由 TCP/IP 之 telecommunications 之處理
pu_i_cnv	於 AFTN/AMHS 位址轉換之介面
pu_m_auf	AFTN/AMHS 存取單元(gateway) 功能
pu_o_ana	process 分析及 incoming messages 之 routing
pu_o_edt	針對傳送之 switched messages 編輯
pu_o_ftr	完成最終傳送之處理

pu_o_irv	主要 process 處理初始化及監督工作
pu_o_per	process 起始週期性的功能
pu_o_rec	報文接收之處理
pu_o_rpc	報文重新處理之處理
pu_o_rvy	應用程式停止之後，recovery 之處理
pu_o_tim	計時及 timeout 處理
pu_x_ecr	伺服器及 supervisory 終端間，命令反應介面
pu_x_sta	線上統計資料處理
pu_x_sup	upervisory 命令之處理
pu_x_xed	多元化的輸出至印表機及螢幕之編輯

二、ATS Message Server PROCESS 之檢視

ATS Message Server process 名稱清單列於下表中，當以 amss 使用者登入後，鍵入 status 命令時，在螢幕上可列出正執行中之 process 名稱。

agt	server administration
cms	message 編碼 and 解碼 (ASN.1 code)
cnv	server administration
dsa	目錄伺服器 DAP 及 DSP 功能
dua	存取服務發送 MTA 及 AGT 需求至 DSA
ibo	存取至 information base operations
lla	存取至 communications facilities
ltmd	server administration
mce	報文轉換字元
mds	針對目錄需求有關 X.400 位址及 X.500 參考之處理
msm	message data base 維護
mta	MTA message exchanges, routing, delivery 及 non-delivery notification, message recovery 之處理
trf	建立連線及 message 傳送於: MTA/MTA, MTA/MS, MS/UA 之間
wan	OSI Presentation (ASN.1), Session and Transport layers 之處理, including RTS.

三、RMA 軟體模組 PROCESS 之檢視

```
#cd /home/amss/apps/rma/bin↵  
#./rma.sh status ↵
```

當以 amss 使用者登入後，變更目錄至<RMA_HOME>/bin，鍵入./rma.sh status 命令時，在螢幕上可列出正執行中之 process 情形。

四、DBMS 資料庫 PROCESS 之檢視

```
#cd /home/amss/dbms/tools↵  
#./dbms.sh status ↵
```

當以 amss 使用者登入後，變更目錄至<DBMS_HOME>/tools，鍵入 ./dbms.sh status 命令時，在螢幕上可列出正執行中之 process 情形。

壹拾、AMHS 系統啓動、重新啓動及 switchover

關於 cluster 主機運作系統之終止及啓動，請參考前面章節。

此章節乃關於 AMHS 應用程式之執行，手動或自動之重新執行、手動或因軟體問題而停止，這些完整程序敘述於“CCS Application” chapter of the Installation, Configuration and Maintenance Manual。

一、啓動 AMHS 伺服器應用程式

如果是第一次執行 AMHS 伺服器應用程式，此執行環境必須事先建立及所有可執行程式須正確建立，請參考第四章 貳、AERMAC 應用程式之安裝。

(一) 首次啓動 AMHS 伺服器應用程式

1、首次啓動 AFTN message switch and AFTN / AMHS Gateway

(1) 以【amss 身分】登入。

(2) 執行 AERMAC.sh，啓動主 process (pu_irv.x)。

```
# AERMAC.sh ←
```

(3) 執行 log_irv.x，啓動所有 process【共 20 個】。

```
# cd $EXE ←  
# log_irv.x ←
```

(4) 此程式提示下列問題要求操作者輸入

```
Exit to UNIX command level? (y/n)  
          回答: n  
Network configuration reintroduction? (y/n)  
          回答: n  
Next restart automatic? (y/n)  
          回答: y  
Restart type? :  
0  COLD      : message queues and traffic days all lost  
1  WARM-1   : message queues and traffic of crashing day lost  
2  WARM-2   : message queues (only) lost  
3  HOT      : message queues and traffic days all recovered  
          回答: 3
```

(5) 執行 pu 指令，確認所有 process 已啓動（所有 processes 名稱請參考第四章 玖、一、AFTN message switching and AFTN/AMHS Gateway 應用程式 PROCESS 之檢視。）

```
#pu ←
```

2、啓動 ATS Message Server

(1) 以【mta/amss 身分】登入。

(2) 執行 goagt 指令

```
# goagt NONE ←
```

(3) 執行 status 指令，確認所有 process 都在執行中(所有 processes

名稱請參考第四章 玖、二、ATS Message Server PROCESS 之檢視)

3、啓動 RMA 模組軟體

- (1) 以【amss 身分】登入。
- (2) 執行./rma.sh status 指令。

```
#cd /home/amss/apps/rma/bin↵  
#./rma.sh start ↵
```

- (3) 執行 RMA 之指令./rma.sh start，確認 RMA 是否執行中（請參考第四章 玖、三、RMA 軟體模組 PROCESS 之檢視）。

4、啓動 DBMS 資料庫

- (1) 以【amss 身分】登入。
- (2) 執行./rma.sh status 指令。

```
#cd /home/amss/dbms/tools↵  
#./dbms.sh start ↵
```

- (3) 執行 RMA 之指令 ./rma.sh start，確認 RMA 是否執行中（請參考第四章 玖、四、DBMS 資料庫 PROCESS 之檢視）。

(二) 非首次啓動 AMHS 伺服器應用程式

1、AFTN message switch and AFTN / AMHS Gateway：

- (1) restart type 爲自動模式：

AERMAC 應用程式因軟體或硬體問題及人爲正常程序終止後，若 demarrage.x 定義 restart type 爲自動模式 (a)，AERMAC 應用程式則會自動重新啓動。

- (2) restart type 爲手動模式：

以下情形需 AERMAC 應用程式之手動啓動程序

- demarrage.x 定義 restart type 爲手動模式 (m)
- 若 restart type 爲自動模式，但自動啓動之次數於規定時間內已超過其重新啓動次數(請參考 AERMAC 應用程式之工具程式 demarrage.x 設定)。
- 手動啓動型態有兩種"total" or "partial"，其取決定於主 process

(pu_o_irv.x) 是否執行。

- 若 pu_o_irv.x 正執行中，則進行 partial 啟動。【log_irv.x】
- 若 pu_o_irv.x 並未執行，則進行 total 啟動。【AERMAC.sh、log_irv.x】

AERMAC 應用程式一旦當機，主 process (pu_o_irv.x) 通常仍繼續執行中，因為 AERMAC 應用程式可能因其他 process 當機而終止；pu_o_irv.x 則會自動偵測此種狀況時就終止所有 processes (包含它自己)。若 restart mode 為自動，此控制之 script AERMAC.sh 會重新執行 pu_o_irv.x，將 AERMAC 應用程式之其他所有 processes 喚起；但若 restart mode 為手動，它將不喚起 AERMAC 應用程式之其他所有 processes，等待人工啟動。

在 AERMAC 應用程式當機，如果 pu_o_irv.x 仍執行中，其 partial 手動啟動如下：

- 以【amss 身分】登入。
- 執行 pu，確認 pu_irv.x 存在而其它 process 不在(若其他 process 存在，則先執行 stop 後再執行，請參考終止 AERMAC 應用程式程序)
- 執行 log_irv.x，啟動 AERMAC 應用程式所有 process

```
# cd $EXE↵  
# log_irv.x↵
```

此程式提示下列問題要求操作者輸入：

```
Exit to UNIX command level? (y/n)  
          回答: n  
Network configuration reintroduction? (y/n)  
          回答: n  
Next restart automatic? (y/n)  
          回答: y  
Restart type? :  
0  COLD      : message queues and traffic days all lost  
1  WARM-1   : message queues and traffic of crashing day lost  
2  WARM-2   : message queues (only) lost  
3  HOT      : message queues and traffic days all recovered  
          回答: 3
```

- 執行 pu 指令，確認所有 process 已啟動 (所有 processes 名稱請參考第四章 玖、一、AFTN message switching and AFTN/AMHS Gateway

應用程式 PROCESS 之檢視)。

在 AERMAC 應用程式一旦當機，如果 pu_o_irv.x 並未執行中，舉例由人為完全停止，其 total 手動啟動如下：

- 請依照第四章 壹拾、一、(一) 1、首次啟動 AFTN message switch and AFTN / AMHS Gateway 之程序執行

2、啟動 ATS Message Server

(1) 以【mta/amss 身分】登入。

(2) 執行 goagt 指令

```
# goagt NONE ↵
```

(3) 執行 status 指令，確認所有 process 都在執行中(所有 processes 名稱請參考第四章 玖、二、ATS Message Server PROCESS 之檢視)

3、啟動 RMA 模組軟體

(1) 以【amss 身分】登入。

(2) 執行 ./rma.sh status 指令。

```
#cd /home/amss/apps/rma/bin↵  
#./rma.sh start ↵
```

(3) 執行 RMA 之指令 ./rma.sh status，確認 RMA 是否執行中(請參考第四章 玖、三、RMA 軟體模組 PROCESS 之檢視)。

4、啟動 DBMS 資料庫

以【amss 身分】登入。

執行 ./dbms.sh start 指令。

```
#cd /home/amss/dbms/tools↵  
#./dbms.sh start ↵
```

執行 DBMS 之指令 ./dbms.sh status，確認 DBMS 是否執行中(請參考第四章 玖、四、DBMS 資料庫 PROCESS 之檢視)。

二、終止 AMHS 伺服器應用程式

(一) 終止 AERMAC 應用程式程序

終止 AERMAC 應用程式可能之情形：

1、人為終止 AERMAC 應用程式

主 process (pu_o_irv.x) 在人為終止 AERMAC 時，所執行之步驟。

- (1) 刪除其中之一的 processes (pu_a_rch.x)。
- (2) 偵測到即將終止之 process 終止訊號。
- (3) 執行終止應用程式之其他的 process
- (4) 最後終止自己之 process。(pu_o_irv.x)

2、由於程式軟體 CRASH，導致應用程式終止。

3、應用程式之其中 process 終止(人為或 process crash)。

4、主 process (pu_o_irv.x) 被終止。

主 process (pu_o_irv.x) 在程式軟體 CRASH、其中 process 終止時，所執行之步驟。

- (1) 偵測到即將 crash 之 process 終止訊號。
- (2) 執行終止應用程式之其他的 process
- (3) 最後終止自己之 process。(pu_o_irv.x)

若 restart type 事先定義於 demarrage.x 為自動(a)，終止 AERMAC 應用程式後，系統不需人工介入會自動啟動 AERMAC 應用程式及回復。

若 restart type 事先定義於 demarrage.x 為手動(m)，終止 AERMAC 應用程式後，系統需人工介入才能重新啟動 AERMAC 應用程式及回復。

當主 process (pu_o_irv.x) 已經終止，AERMAC 應用程式則會自動重新啟動 AERMAC_S.sh (此 Shell-script 會被 AERMAC.sh 重新喚起)，以啟動主 process (pu_o_irv.x)。

- 如果 restart type 為自動，則主 process (pu_o_irv.x) 自動被啟動後，啟動其它 processes。
- 如果 restart type 為手動，則主 process (pu_o_irv.x) 自動被啟動後，

不啓動其它 processes，等待"kick-off" 訊息或"total stop"訊息。

【"kick-off" 訊息或"total stop"訊息產生於人工介入執行 log_irv.x 命令產生，以上所提之 "kick-off" 訊息是當 log_irv.x 在應用程式啓動時，由 log_irv.x 送給 pu_o_irv.x】

(二) 終止 AERMAC 應用程式程序，可分爲 total 或 partial

系統維護人員如果需要終止 AERMAC 應用程式，例如維護系統，則首先需要執行 demarrage.x 將 restart type 由 a 改爲 m，平常時間之 restart type 應設爲 a 模式。

1、AERMAC 應用程式手動完全終止程序（total）

AERMAC 應用程式手動完全終止程序（total），乃是 AERMAC 應用程式正常執行中，將所有 processes 終止，採兩步驟完成。

(1) 終止其中之 processes (pu_a_rch.x)。

以【amss 身分】，鍵入 stop_armac.x 或 kill -9 pu_a_rch.x。

```
#stop_armac.x ↵  
或  
#kill -9 pu_a_rch.x ↵
```

執行 pu 命令，確保除 pu_irv.x 外之所有 process 都已終止。

```
#pu ↵
```

(2) 終止主 process (pu_o_irv.x)

確認先前之 pu_o_irv.x 已經終止及新的 pu_o_irv.x 已被執行。

```
#ps -aef | grep irv ↵
```

CPU 使用時間欄位須顯示 0:00 以表示一個新的 process。如果沒有，則需要等待到 pu_o_irv.x 終止並重新啓動後。

終止新的 pu_o_irv.x

```
#cd $EXE ↵  
#log_irv.x ↵
```

畫面出現

```
"Exit to UNIX command level (y/n)?" 回答 y  
enter the password : 此密碼爲 amss
```

這將送出"total stop"訊息給正等待之主 process (pu_o_irv.x)，並終止主 process (用 pu 檢視)及它執行 script AERMAC_S.sh 。

2、AERMAC 應用程式手動部分停止程序 (partial)

AERMAC 應用程式手動部分停止程序 (partial)，乃是因人為或軟體 crash，造成 AERMAC 應用程式執行之 process，僅剩唯一 process (pu_o_irv.x)，AERMAC 應用程式手動部分停止程序 (partial) 相當簡單，只要執行 AERMAC 應用程式手動完全終止程序 (total) 之步驟 2 程序，換言之只要執行 log_irv.x 即可。

(三) 終止 ATS Message Server 程序

(四) 終止 RMA 模組軟體

- 1、以【amss 身分】登入。
- 2、執行 ./rma.sh stop 指令。

```
#cd /home/amss/apps/rma/bin↵  
#./rma.sh stop ↵
```

- 3、執行 RMA 之指令 ./rma.sh status，確認 RMA 是否執行中（請參考第四章 玖、三、RMA 軟體模組 PROCESS 之檢視）。

(五) 終止 DBMS 資料庫

- 1、以【amss 身分】登入。
- 2、執行 ./dbms stop 指令。

```
#cd /home/amss/dbms/tools↵  
#./dbms.sh stop ↵
```

- 3、執行 DBMS 之指令 `./dbms.sh status`，確認 DBMS 是否執行中（請參考第四章 玖、四、DBMS 資料庫 PROCESS 之檢視）。

三、 啓動及終止 A2M 應用程式

(一) 啓動 A2M 應用程式程序

雙擊桌面之 icon (A2MSUP)或檔名爲 `start_aermac.bat`。

(二) 終止 A2M 應用程式程序

選擇 menu item → File.. → Exit。

壹拾壹、 AMHS 維護程序

一、 AERMAC 應用程式

(一) 系統問題追蹤

檔案	描述
errlog.h	<p>此檔案位於 \$TRACE 目錄裡，它包含 AERMAC 應用程式啓動及終止之日期時間，每次 AERMAC 應用程式啓動及終止都會被更新。</p> <ul style="list-style-type: none"> ● 當 AERMAC 應用程式被啓動後，下列之敘述也附加至 errlog.h 檔案裡。 Aermac Software Starting at (date):(time) ● 當 AERMAC 應用程式被終止後（crash 或人爲從終端機使用 stop_aermac.x 工具）， <ul style="list-style-type: none"> ①如果一個 core 檔案被產生，下列之敘述也附加至 errlog.h 檔案裡。 Aermac Software failure detected at (date):(time) ②如果沒有 core 檔案被產生，下列之敘述也附加至 errlog.h 檔案裡。 No core file <p>當 AERMAC 應用程式因一個 process 而終止，會產生一個系統 core 檔案，</p>

	<p>此 core 檔案將包含該 process 當掉瞬間之內容。</p> <p>gdb 系統偵錯工具使用分析 core 檔案。</p> <p>如果當 AERMAC 應用程式終止而產生 core 檔案（它不總是如此），一個特別目錄被建立，包含所有 AERMAC 應用程式之日誌檔案。</p>
*.out 檔案	<p>AERMAC 應用程式之每個 process，當它被啟動時，就會開啓一個日誌檔（process_name.out），此日誌檔記載該 process 於生命週期所有訊息。</p>
.trace/*.trace.anc 檔案	<p>AERMAC 應用程式之每個通信 process(c_asy、c_pvc、c_svc、c_tlx、c_fhx、n_c3b、c_isn、c_tcp、c_fip、c_sgt)，當它被啟動時，就會開啓一個日誌檔（process_name.trace），由 process 記載該 process 所有處理完成之日誌。</p> <p>舉例：process 【pu_c_tcp.x】，其日誌檔為 c_tcp.trace。</p> <p>process_name.trace 檔案將從該 process 啟動後，記載它呼叫過之所有函式功能；此檔案為循環，當檔案大小超過 2MB 時，它將被搬移至 process_name.trace.anc 及新的 process_name.trace 檔案被開啓；AERMAC 應用程式被啟動時，先前之 process_name.trace 檔案將被搬移至 process_name.trace.anc 及新的 process_name.trace 檔案也被開啓。</p> <p>m_auf process（系統中唯一執行 x.400 選項）也是維護追蹤用途，其名稱爲 auf.trace，提供在處理 x.400 訊息及其它物件之主要紀錄。</p>
AERMAC 軟體問題追蹤	<p>當 AERMAC 應用程式被終止後（crash 或人爲從終端機使用 stop_aermac.x 工具），如果一個 core 檔案被產生，一個 dump 目錄（名稱爲：dump_（data）（time））被產生於 DUMP 目錄（\$SYSPATH/dump）。</p> <p>此 DUMP 目錄將包含下列訊息</p> <ul style="list-style-type: none"> ● *.trace/*.trace.anc 檔案 ● *.out 檔案 ● core 檔案 <p>日誌檔（*.trace/*.trace.anc、*.out）針對軟體當機給予相關直接訊息。</p> <p>o_irv.out 檔案（pu_o_irv.x process 日誌檔）記在哪一個 process 造成 AERMAC 應用程式終止，在依據該 process 之 out 日誌檔去查明哪一個函式造成。</p> <p>舉例：</p> <p>①因 SGT process dead，造成 AERMAC 應用程式被終止，於 o_irv.out 檔案最後內容出現：</p> <pre> PROCESS : pu_o_irv.x OPERATION : OE_IRDEAD(pu_c_sgt) ERREUR : 16086 Erno : 0 </pre> <p>在該 process 之日誌檔（out）造成 AERMAC 應用程式被終止，於其.out 日誌檔案最後內容出現：</p> <pre> PROCESS : process_name (e.g: pu_c_sgt) </pre>

<p>OPERATION : function executed when the process stopped</p> <p>ERREUR : AERMAC error code (e.g:16216)</p> <p>Errno : UNIX error code (e.g:11)</p> <p>◎關於 AERMAC error code 可查閱\$CTU/ms_err/ms_err.inc</p> <p>◎關於 UNIX error code 可查閱/usr/include/sys/errno.h</p> <p>②使用 stop_aeremac.x，終止 AERMAC 應用程式，process (pu_a_rch.x 被終止)，於 o_irv.out 檔案最後內容出現：</p> <pre>PROCESS : pu_a_rch.x OPERATION : BYE_BYE ERREUR : 16069 Errno : 0</pre> <p>③如果當 AERMAC 應用程式終止沒有 core 檔案產生，不會產生 DUMP 目錄。</p> <p>④工具程式 PURGE_DUMP.sh 使用清除 DUMP 目錄。</p>

(二) 備份

此工具程式針對以下的各種資料提供備份及還原之功能：

- 1、AERMAC 產品之原始碼
- 2、AERMAC 應用程式之靜態設定資料
- 3、AERMAC 應用程式之網路設定資料
- 4、報文日誌

狀況	AERMAC 應用程式之靜態設定資料
備份 (save_dir.sh) 1.AERMAC 應用程式之靜態設定資料 2. AERMAC 應用程式之原始碼	此工具程式使用於備份儲存 AERMAC 應用程式之靜態設定資料及原始碼，可儲存於磁帶或檔案裡 <ul style="list-style-type: none"> ● 儲存 AERMAC 應用程式之靜態設定資料 <ul style="list-style-type: none"> 【儲存於磁帶】 #cd \$CTU/util_dev↵ #save_dir.sh -d \$CTU/conf↵ 【儲存於檔案/CD】 #cd \$CTU/util_dev↵ #save_dir.sh -f \$CTU/conf↵ ● 儲存 AERMAC 應用程式之原始碼 <ul style="list-style-type: none"> 【儲存於磁帶】 #cd \$CTU/util_dev↵

	<pre>#save_dir.sh -d \$CTU↵ 【儲存於檔案/CD】 #cd \$CTU/util_dev↵ #save_dir.sh -f \$CTU↵</pre>
<p>還原 (restore_dir.sh)</p> <p>1. AERMAC 應用程式之靜態設定資料</p> <p>2. AERMAC 應用程式之原始碼</p>	<p>此工具程式使用於將儲存於磁帶或檔案裡之 AERMAC 應用程式之靜態設定資料及原始碼還原</p> <ul style="list-style-type: none"> ● 還原 AERMAC 應用程式之靜態設定資料 <ul style="list-style-type: none"> 【還原從磁帶】 <pre>#cd \$CTU/conf↵ #restore_dir.sh -d ↵ 【還原至目前目錄】 #restore_dir.sh -d <dest>↵ 【還原至<dest>目錄】 【還原從檔案/CD】 #cd \$CTU/conf↵ #restore_dir.sh -f ↵ 【還原至目前目錄】 #restore_dir.sh -f <dest>↵ 【還原至<dest>目錄】</pre> ● 儲存 AERMAC 應用程式之原始碼 <ul style="list-style-type: none"> 【還原從磁帶】 <pre>#cd \$CTU↵ #restore_dir.sh -d ↵ 【還原至目前目錄】 #restore_dir.sh -d <dest>↵ 【還原至<dest>目錄】 【還原從檔案/CD】 #cd \$CTU↵ #restore_dir.sh -d ↵ 【還原至目前目錄】 #restore_dir.sh -d <dest>↵ 【還原至<dest>目錄】</pre>
<p>備份 (saveconf.sh)</p> <p>AERMAC 應用程式之網路設定資料</p>	<p>此工具程式 saveconf.sh 備份儲存 AERMAC 應用程式之網路設定資料，可儲存於磁帶或檔案裡，此工具程式首先將有關網路設定資料之 share memory 區塊寫入檔案後，拷貝至磁帶或檔案。</p>
<p>還原</p> <p>AERMAC 應用程式之網路設定資料</p>	<p>從磁帶或檔案還原 AERMAC 應用程式之網路設定資料，必須終止 AERMAC 應用程式及重新啓動它，指定網路設定資料需重新還原，有關 AERMAC 應用程式重新啓動之程序請參考第四章 壹拾、一、(一) 和第四章 壹拾、一、(二) 之說明。</p> <p>當重新啓動 AERMAC 應用程式，必須回答 log_irv.x 之問題</p> <pre>Exit to UNIX command level? (y/n) 回答: n 【表示要啓動 AERMAC】 Network configuration reintroduction? (y/n) 回答: y 【回答: y 表示要重新載入舊的 AERMAC 應用程式之網</pre>

	<p>路設定資料】</p> <p>插入含有舊的網路設定資料之磁帶或 floppy disk。</p> <p>Is the drive ready? (y/n)</p> <p>回答: y</p> <p>reload 完成，移除磁帶或 floppy disk。</p> <p>當完成 AERMAC 應用程式之網路設定資料重新下載後，不做 message recovery，所有報文資料皆被刪除。</p>
備份 (archiv_mes.sh) 每日報文日誌	此工具程式 archiv_mes.sh 備份儲存 AERMAC 應用程式之每日報文資料，儲存至磁帶。
還原 (reintro_mes.sh) 每日報文日誌	<p>此工具程式 reintro_mes.sh 從磁帶還原已被刪除之日報文資料。每次執行還原程序僅能以一日為單位，且今日與昨日不能執行還原程序。</p> <p>工具程式 list_disk.x 給予在磁碟中針對日報文儲存提供下列訊息</p> <ul style="list-style-type: none"> ● 如果此日報文已被儲存至磁帶，記載儲存時間 ● 如果此日報文被 reintro，以 flag 表示 ● 日報文之期間接收報文之數量 ● 日報文之期間傳送報文之數量 ● 傳送之平均比率 <p>此訊息被提供從 ITSUP 席位。</p>

(三) 維護

此節解釋說明需要針對 AERMAC 應用程式、Clock process 及 CMS proxy-agent 維護之行動項目。

此維護項目需週期性執行。

項目	維護項目
AERMAC 應用程式	<p>硬體檢查</p> <ul style="list-style-type: none"> ● CCS board 檢查 (Stratus 或 PC 或 Digital Alpha server) ● FEP board 檢查 <p>因為非同步連接在 CCS 非同步埠及 MODEM 之間有問題，如果與 CAC 連接沒有正確設定，此命令將產生下列結果</p> <pre> -/sbin/rsntrans -r1 -sHub -z -f -x0 - waiting for connect message - time out opening device </pre> <p>因為非同步連接在 PSTN 之間有問題，如果與 CAC 連接沒有正確設定，此命令將產生下列結果</p> <pre> -/sbin/rsntrans -r1 -sHub -z -f -x0 </pre>

- waiting for connect message
- 2 or 3 minutes later.....
- 14:55:26 C no connect after 4 attempts
- Lan 檢查
如果 AERMAC 應用程式使用雙網路，週期性檢查雙網路之狀態。如果單一網路故障，系統會自動切換至另一網路，但若雙網路都故障則所有 tcp subscriber 都無法到達。
- Printer 檢查
應週期性連接至 CCS 之 HSP（高速印表機）。
檢查印表機紙匣是否充足及 printer queue 應為空的。
問題發生時，檢查與 CCS 及印表機之實體連接、檢查印表機之設定。
軟體檢查
- 磁碟空間檢查
— 當磁碟空間 LOW 時，AERMAC 應用程式會自動刪除磁碟中之舊的日報文日誌。
- Alarm handling
當磁碟空間不足時，AERMAC 應用程式會產生 alarm。
【請參考 ANNEX H】
— 當 Alarm <<121 WARNING:DISK SATURATION <%used>>>, 建議『立即使用 PURGE_DUMP.sh 刪除存在之 DUMP 目錄』。
- 建議每週刪除已存在之 DUMP 目錄及輸出之統計資料檔案【請參考 ANNEX G】
- 離線統計檢查

壹拾貳、系統工具程式

完整系統工具程式可查閱 "Tools" Annex of the Installation, Configuration and Maintenance Manual.

一、AERMAC 應用程式之工具程式

(一) 工具程式的種類

- 1、為伺服器的可執行應用程式在發展環境之產生及傳送至執行環境之工具程式

```
genmk  
make  
mkcible.sh  
livcible.sh  
valid_aermac.sh  
valid_util_aermac.sh
```

- 2、關於應用程式之首次啟動之工具程式

```
cleanipc.sh  
creation.x  
demarrage.x
```

- 3、應用程式之設定檔案生效之工具程式。

```
valid_conf.sh      + create_table.sh  
create_table.x
```

- 4、應用程式啟動及終止之工具程式

```
AERMAC.sh      + AERMAC_S.sh  
log_irv.x  
stop_aermac.x
```

- 5、儲存及回存應用程式之 **source**、設定及 **traffic** 檔案之工具程式

```
save_dir.sh  
restore_dir.sh  
saveconf.x  
newconf.sh      + errconf.x,  
finconf.x  
archiv_mes.sh   + archiv_mes.x  
err_reintro.x, fin_reintro.x  
restore_log.sh
```

6、維護之工具程式

```

PURGE_DUMP.sh
change_time.x
cnf_edt.x
list_disk.x
fiptool.x
extractpl.x
calculmp.x
simu.x
statsys.sh
statsystofloppy.sh
ala.sh          +   ala.x
sup.sh          +   sup.x
    
```

(二) 工具程式的說明

Utility	描述	Utilities called by this utility
AERMAC.sh	AMHS 應用程式之 start-up	AERMAC_S.sh
ala.sh	list of alarms issued on a specified date DDMM	ala.x
archiv_mes.sh	日期 traffic 儲存工具	archiv_mes.x
calcul_mp.x	計算 shared memory 全部使用之空間	
change_time.x	修改系統時間	
cleanipc.sh	IPC 重初始化(share memory)	
cnf_edt.x	設定檔清單	
creation.x	網路設定初始	
demarrage.x	應用程式重新啟動參數	
DUMP_AERMAC.sh	從 process crash 之 dump 儲存	
extractpl.x	從 traffic day 檔案的 X400 之 ASN.1 code 檔案之 extractor	
findfile.sh	檔案搜尋	
fiptool.x	NAMIP 單元設定	
gene_code.x	從 regular expression specifications 產生 C code	

Utility	描述	Utilities called by this utility
genmk	Makefiles 之產生	
link_ctu.sh	階層目錄之建立包含連結至 reference process sources	
link_ctu_ms	階層目錄之建立包含連結至 reference 支援機器 sources	
list_disk.x	顯示在系統磁碟所有之 traffic 日期資料	
livcible.sh	可執行檔之 delivery 至 target 環境	valid_aermac.sh valid_util_aermac.sh
mkcible	建立 target 環境	
mkconf.sh	建立連結至設定檔案	
mklninc.sh	建立連結至 include 檔案	
newconf.sh	configuration restoration (called by log_irv.x utility)	errconf.x finconf.x
PURGE_DUMP.sh	清除不需要之 dumps	
reftocible	拷貝從 reference 發展檔案	
reintro_mes.sh	重建日期之 traffic	reintro_mes.x err_reintro.x fin_reintro.x
saveconf.x	儲存系統設定檔	
statsys.sh	系統統計檔日誌	
statsystofloppy.sh	拷貝系統統計檔案至磁碟片	
stop_aermac.x	終止 AMHS 應用程式	
sup.sh	supervisory 命令之清單在指定日期 DDMM	sup.x
transfer_dump	轉送 dumps 資料至別處，避免磁碟 saturation	
trdir	transfer directories	
valid_aermac.sh	搬移所產生之可執行的 process 至執行環境 (\$EXE)	
valid_conf.sh	確認 AERMAC server 設定檔案	create_table.sh create_table.x (create configuration file)
valid_util_aermac.sh	搬移所產生之可執行的工具程式至執行環境 (\$OUTILS)	

二、 磁性媒體儲存操作 (**MAGNETIC MEDIA STORAGE OPERATIONS**)

針對備份及系統檔案之儲存，磁性媒體儲存操作更顯其重要性。

- (一) 執行環境之檔案。
- (二) 每日 **traffic** 報文檔案，包含相關聯之警示、統計及 **supervisory** 所下之命令。
- (三) 應用程式之原始碼檔案

使用備份檔案之儲存裝置有：

- 1、DAT 磁帶。(不使用)
- 2、MO (Magneto-Optical Disk) drive (Operations using the MO drive)。
- 3、磁碟片(diskette)

三、 **A2M** 工具程式

在此主要說明一些較常在 A2M 席位上使用的系統工具程式。

- (一) 觀察 **A2M** 磁碟使用之情形

※開啓瀏覽器 (我的電腦)
※在 C:\icon 按右鍵
※選擇內容.

(二) 從 A2M 席位之檔案傳送

※開啓 MS-DOS 提示命令
※使用 CD 命令至你希望 (GET 或 PUT) 之目錄
※鍵入 ftp <hostname> 或 ftp <IP address>
輸入正確之使用者帳號及密碼
※確認你在遠端主機之目前目錄
ls 或 ls -l
※改變你在遠端主機目錄位置至你希望之目前
cd <directory path name>
※選擇檔案傳送之模式
bin 或 ascii
※從遠端主機得到一個檔案或多個檔案
get <filename> (一個)
mget <filename> (多個) 使用*/?
※放置到遠端主機一個檔案或多個檔案
put <filename> (一個)
mput <filename> (多個) */?
※離開 ftp
quit 或 bye

(三) 其它 A2M 工具程式

A2M utility	說明
createPack.sh	使用於應用程式 CD 之建立
InstallAermac.bat	使用於應用程式之安裝
InstallFonts.sh	字型安裝期間，由 installAermac.sh 呼叫之內部的 shell script
env.bat	內部的 shell script 使用去建立環境變數。
start_aermac.bat	啓動 Aermac 應用程式
aermac.bat	從 dev 目錄，啓動 Aermac 應用程式 from the dev directory.
aermacJar.bat	從 dev 目錄，啓動 Jar 檔案
make_ans.bat	Makes ans 檔案(dev)
make_aermac.bat	Makes aermac 檔案(dev)
make_jar.bat	Makes a Jar 檔案(dev)
make_all.bat	Makes 所有檔案.
make_only.sh	Makes a selected 檔案.
akeJarCopy.bat	Makes a Jar 檔案 and 拷貝它至 bin 目錄.
make_class.bat	Makes a selected 檔案(OLD)
trStat.bat	每日從伺服器至本地端 PC 傳送 STAT.OMT 檔案

A2M utility	說明
autologftp.txt	Internal 檔案使用去自動連接從本地端 PC 至伺服器以傳送統計檔案 via ftp.
planStat.bat	使用於計劃每日 STAT.OMT 檔案之傳送.
getStat.bat	使用於 transf 傳送一個 selected STAT.OMT 檔案.
stopSup.sh	使用於終止 supervision
stopSession.sh	使用於終止 session
InstRpm.sh	Script to install the package for Mwm (TEST ONLY)

壹拾參、Thales-Is Linux Cluster (簡稱 TILC)

一、TILC 之介紹

AMHS 系統所有相關之伺服器，如 AMHS 伺服器、目錄伺服器 (Directory Access Server)、ATN Router 伺服器，其主機架構採用雙套主機並存，其一主機為運作模式 (Active)，另一主機為備份模式 (Stand-by)，並採用叢集控制方式 (Cluster)，以確保系統資料不遺漏，服務不中斷，TILC 則為提供叢集控制方式之軟體，且安裝於 AMHS 系統所有相關之伺服器上。

當提供服務執行 (AMHS、ATN ROUTER、LDAP) 中之運作伺服器 (ACTIVE) 硬體故障，TILC 軟體會自動偵測且切換至 Stand-by 伺服器，並取代成為運作伺服器 (ACTIVE)，繼續提供系統運作及服務，不需人工切換；系統維護人員則可盡速完成修護故障之伺服器。

二、TILC 之硬體架構及工作原理

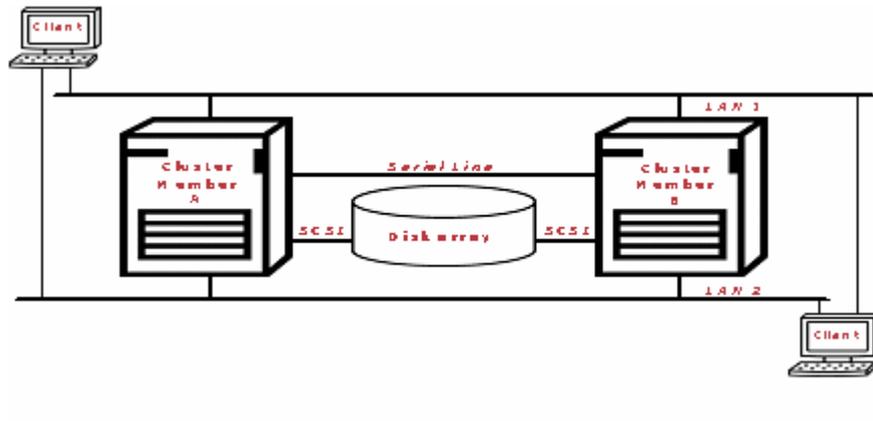
(一) TILC 之硬體必須架構

- 1、兩部含有雙網路介面連接 (Dual LAN) 之伺服器等級電腦。
- 2、一個外接式磁碟陣列器 (External Shared Disk Array)
- 3、兩條 parallel SCSI 介面線材
- 4、兩部 Linux 作業系統伺服器間非同步連結

(二) TILC 之硬體工作原理

- 1、TILC 使用連結 (LAN、ASYN) 使用
 - (1) 檢查叢集主從伺服器間系統硬體狀況。
 - (2) 執行遠端命令切換叢集主從伺服器。
- 2、TILC 所使用之叢集主從伺服器之作業系統資料皆分別置於各叢集伺服器之系統磁碟中。
- 3、屬於 TILC 所管理之 (AMHS、LDAP 等) 服務資料皆儲存於共享的磁碟陣列。
- 4、起始叢集主從伺服器溝通之共享磁碟陣列分割區稱為『quorum』。
 - (1) 檢查叢集主從伺服器之另一伺服器硬體狀況。
 - (2) 與叢集主從伺服器之另一伺服器交換訊息。
- 5、一個『quorum』區域保留給叢集主從伺服器之每一個伺服器。
- 6、TILC daemon 於每秒在 quorum』保留區域儲存日期，及更新本叢集伺服器服務執行情形。

圖 四 - 9 TILC 的硬體架構圖



三、TILC 之軟體架構及工作原理

(一) TILC 之軟體必須架構

- 1、叢集主從伺服器成員必須先安裝 Linux 作業系統，支援之作業系統有 Linux RedHat ES2.1 Update 3、Linux RedHat ES3.0 Update 1、Linux RedHat 8.0 等作業系統。
- 2、作業系統之 kernel，IP Aliasing 必須被支援設定。
- 3、叢集主從伺服器成員必須設定 NTP。
- 4、叢集主從伺服器成員必須分別執行 RSH 服務。
- 5、叢集主從伺服器成員必須先設定 TIME 服務。
- 6、一個有效之磁碟分割給叢集 quorum。

(二) TILC 之軟體架構包括

- 1、Configuration 檔定義。
- 2、叢集主從伺服器間溝通連結。
- 3、quorum 分割。

(三) 管理工具。

- 1、啟動服務。
- 2、停止服務。
- 3、重新安置服務。
- 4、檢視服務狀態。
- 5、顯示警示。

(四) 低階工具。

- 1、藉由 SCSI BUS 傳遞命令檢查磁碟陣列情形。

- 2、當 IP Alias 切換從叢集一伺服器主機轉至另一部時，利用網路傳送更新之 arp 至其他主機。

(五) TILC 之軟體架構流程

- 1、叢集主從伺服器成員於 Linux 作業系統啓動 (boot) 時期，啓動主 TILC Daemon。
- 2、這 TILC Daemon 隨時監視叢集之另一伺服器硬體狀況。
- 3、另一 Daemon 被啓動監視 TILC Daemon 是否執行，若沒執行，則自動啓動監視 Daemon。

StandBy 伺服器之監視 Daemon 偵測至 Active 伺服器硬體異常，則於本地端 StandBy 伺服器啓動服務。

(六) TILC 之軟體架構自動切換流程

- 1、第一階段：在啓動執行中叢集服務之 TILC 動作。
 - (1) 如果叢集所設定之服務已在執行，則檢查執行情況。
 - (2) 檢查網路及存取分享磁碟陣列。
 - (3) 指定 IP 別名 (Alias)
 - (4) 掛載分享磁碟檔案系統。
 - (5) 啓動相關應用程式。
 - (6) 更新『quorum』及叢集檔案用服務新狀態。
- 2、第二階段：在停止執行中叢集服務之 TILC 動作。
 - (1) 如果叢集所設定之服務正執行中，則檢查執行情況。
 - (2) 停止相關應用程式。
 - (3) 卸載分享磁碟檔案系統。
 - (4) 卸除指定 IP 別名 (Alias)。
 - (5) 更新『quorum』及叢集檔案用服務新狀態。
- 3、第三階段：在啓動執行中叢集服務之 TILC 動作。
 - (1) 如果叢集所設定之服務正執行中，則檢查執行情況。
 - (2) 檢查網路及存取分享磁碟陣列。
 - (3) 停止目前 Active 伺服器之服務，步驟如第二階段。
 - (4) 啓動目前 StandBy 伺服器之服務，步驟如第一階段。
 - (5) 如果以上之重新安置步驟失敗，則重新針對先前之 Active 伺服器之服務重新啓動服務，若再失敗則 STOP。

(七) TILC 叢集之 AMHS 軟體異常處理

諸如 AFTN message switching、AFTN/AMHS Gateway、ATS message Server function 等軟體功能異常，將不喚起叢集主從伺服器自動切換，此現

象解決乃相關軟體應用程式將於同一伺服器重新啓動【自動或手動】。

(八) TILC 叢集之關機或 boot 處理

- 1、叢集間成員之一伺服器在下列情形需要關機及重新啓動動作：
 - (1) 系統必須關機維護。
 - (2) 安裝新版的作業系統後，kernel 核心修改。
- 2、關機前，先進行叢集主從伺服器切換至 Standby 伺服器後，在將該伺服器中所有執行的應用軟體遵循其所屬之正常關機程序執行

四、TILC 安裝

(一) TILC 軟體解壓縮

TILC 安裝軟體為壓縮形式之檔案，它必須安裝於\$CLUSTER_PATH，此\$CLUSTER_PATH 環境變數被定義在~root/.tilcrc 檔案中，其預設值為 /usr/local/tilc；如果這預設值不被使用，後來會自動安裝之.tilcrc 檔案將被叢集一伺服器主機成員更新。

建立 cluster 根目錄，解壓縮。

```
#export CLUSTER_PATH=/usr/local/tilc↵  
#mkdir -p $CLUSTER_PATH↵  
#cd /usr/local/tilc↵  
#tar xzvf <archieve_path>/tilc.tgz↵          【archieve_path 為檔案所在之目錄】
```

會在 cluster 根目錄 (/usr/local/tilc)，產生以下子目錄

alarm :	警示及事件 log 檔
conf :	TILC config 每一服務處理的資料 舉例 conf/services/amss 有 amss.scr ,amss.dump

etc :	TILC config 資料 (伺服器 LAN 及 SLIP 位址及 quorum partition 資料)
libsh :	TILC SCRIPT 使用之資訊程式庫
sbin :	執行程式及 SCRIPT

(二) TILC 軟體安裝

- 1、在 cluster 根目錄編譯 TILC 工具及安裝軟體。

```
#cd $CLUSTER_PATH ↵
#make ↵
```

- 2、在/etc/rc.d/init.d 目錄安裝 cluster 服務及在/etc/rc.d/rcx.d 目錄建立 start/stop 連結。
- 3、建立/root/.tilcrc 檔案定義 CLUSTER_PATH 環境變數。
- 4、在~/.bashrc 檔案鍵入 call ~/.tilcrc 檔案。
- 5、在/etc/inittab 檔案中建立一行 for monitor。

```
c1:2345:respam:/usr/local/tilc/sbin/monitor.d
```

- 6、init 要求重讀 configuration。
- 7、拷貝 tilc_log_rotate 檔案給 logrotate daemon 處理 log 管理。

(三) rsh configuration

- 1、cluster scripts 使用 rsh 命令在叢集伺服器另一成員執行命令，叢集伺服器彼此成員間可藉由 rsh 使用 root 帳號登入，故在叢集伺服器之每一成員的/root/.rhosts 各自增加遠端伺服器之 interface

Server A 之/root/.rhosts	Server B 之/root/.rhosts
rocamhsdeva-1 root	rocamhsdevb-1 root
rocamhsdeva-2 root	rocamhsdevb-2 root
rocamhsdeva-3 root	rocamhsdevb-3 root

- 2、修改叢集伺服器之每一成員的/root/.rhosts 之屬性，具有讀

寫能力。

```
#chmod 600 /root/.rhosts ↵
```

3、修改/etc/pam.d/rsh 檔案

```
# vi /etc/pam.d/rsh ↵
```

將原本 `auth required /lib/security/pam_securetty.so` 改為 `auth optional /lib/security/pam_securetty.so`。

```
##%PAM-1.0
# For root login to succeed here with pam_securetty, "rsh" must be
# listed in /etc/securetty.
auth      required /lib/security/pam_nologin.so
auth      optional /lib/security/pam_securetty.so
auth      required /lib/security/pam_env.so
auth      required /lib/security/pam_rhosts_auth.so
account   required /lib/security/pam_stack.so service=system-auth
session   required /lib/security/pam_stack.so service=system-auth
```

4、修改/etc/xinetd.d/rsh 檔案

```
# vi /etc/xinetd.d/rsh ↵
```

將原本 `disable yes` 改為 `no`。

```
.....
.....
disable = no
```

重新載入 xinetd configuration

```
# service xinetd reload ↵
```

(四) time configuration

叢集伺服器使用者由於使用 `rdate` 命令，故需起始 `xinetd` 服務中 `time`，所以修改 `/etc/xinetd.d/time`。

```
# vi /etc/xinetd.d/time ↵
```

將原本 `disable yes` 改為 `no`。

```

# default: off
# description: An RFC 868 time server. This protocol provides a \
#   site-independent, machine readable date and time. The Time \
#   service sends back to the originating source the time in \
#   seconds since midnight on January first 1900. This is the \
#   tcp version.

service time
{
    type      = INTERNAL
    id        = time-stream
    socket_type = stream
    protocol  = tcp
    user      = root
    wait      = no
    disable   = no
}

```

重新載入 xinetd configuration。

```
# service xinetd reload ↵
```

(五) ntp configuration

叢集伺服器使用者由於使用 ntp 服務及 quorum 檔案系統時間同步叢集之伺服器成員間。

- 1、叢集伺服器成員必須具有相同內容 configuration 檔案 /etc/ntpd.conf 及 /etc/ntp/step-tickers，首先修改 /etc/ntpd.conf。

```
# vi /etc/ntpd.conf ↵
```

將 fudge 作註解，並加入 time server（舉例 mario）。

```

.....
Prohibit general access to this service.
restrict default ignore
# Permit all access over the loopback interface.  This could
# be tightened as well, but to do so would effect some of

```

```

# the administrative functions.
restrict 127.0.0.1
restrict 142.3.22.23 nomodify noquery notrap # BY THALES
server 142.3.22.23 # BY THALES
restrict 142.3.22.240 # BY THALES
peer 142.3.22.240 # BY THALES
restrict 142.150.0.240 # BY THALES
peer 142.150.0.240 # BY THALES
restrict 10.0.0.240 # BY THALES
peer 10.0.0.240 # BY THALES

# -- CLIENT NETWORK -----
# Permit systems on this network to synchronize with this
# time service. Do not permit those systems to modify the
# configuration of this service. Also, do not use those
# systems as peers for synchronization.
# restrict 192.168.1.0 mask 255.255.255.0 notrust nomodify notrap

# --- OUR TIMESERVERS -----
# or remove the default restrict line
# Permit time synchronization with our time source, but do not
# permit the source to query or modify the service on this system.

# restrict mytrustedtimeserverip mask 255.255.255.255 nomodify notrap noquery
# server mytrustedtimeserverip

# --- NTP MULTICASTCLIENT ---
#multicastclient # listen on default 224.0.1.1
# restrict 224.0.1.1 mask 255.255.255.255 notrust nomodify notrap
# restrict 192.168.1.0 mask 255.255.255.0 notrust nomodify notrap

```

```

# --- GENERAL CONFIGURATION ---
#
# Undisciplined Local Clock. This is a fake driver intended for backup
# and when no outside source of synchronized time is available. The
# default stratum is usually 3, but in this case we elect to use stratum
# 0. Since the server line does not have the prefer keyword, this driver
# is never used for synchronization, unless no other other
# synchronization source is available. In case the local host is
# controlled by some external source, such as an external oscillator or
# another protocol, the prefer keyword would cause the local host to
# disregard all other synchronization sources, unless the kernel
# modifications are in use and declare an unsynchronized condition.
#
#server 127.127.1.0 # local clock
#fudge 127.127.1.0 stratum 10

#
# Drift file. Put this in a directory which the daemon can write to.
# No symbolic links allowed, either, since the daemon updates the file
# by creating a temporary in the same directory and then rename()'ing
# it to the file.
#
driftfile /etc/ntp/drift
#broadcastdelay 0.008

#
# Authentication delay. If you use, or plan to use someday, the
# authentication facility you should make the programs in the auth_stuff
# directory and figure out what this number should be on your machine.
#
authenticate no

#

```

```
# Keys file.  If you want to diddle your server at run time, make a
# keys file (mode 600 for sure) and define the key number to be
# used for making requests.
#
# PLEASE DO NOT USE THE DEFAULT VALUES HERE. Pick your own, or
remote
# systems might be able to reset your clock at will. Note also that
# ntpd is started with a -A flag, disabling authentication, that
# will have to be removed as well.
#
#keys          /etc/ntp/keys
```

2、修改/etc/ntp/step-tickers。

```
# vi /etc/ntp/step-tickers↵
```

增加 time server 名稱（舉例 mario）。

```
server mario #如果 mario 已定義在/etc/hosts，可使用 mario
142.3.22.23 #如果 mario 未定義在/etc/hosts，使用 IP Address
```

五、TILC Configuration 設定

(一) Cluster quorum partition

使用 fdisk 在共享磁碟陣列建立至少 1M bytes quorum 之分割區。

1、顯示目前 partition 狀態

```
fdisk -l /dev/cciss/c0d0 (OS DISK)
```

2、fdisk -l /dev/cciss/c0d1 (共享磁碟陣列)

分割共享磁碟陣列（以下舉例）

Device	start	end	blocks
/dev/cciss/c0d1p1	1	1	4064
/dev/cciss/c0d1p2	2	8717	35561

(二) Cluster Configuration

- 1、Cluster Configuration 設定檔位於
\$CLUSTER_PATH/etc/cluster.cfg，此檔包含 cluster 叢集
間伺服器成員之 IP 位址及 quorum partition 分割區。
- 2、TILC 使用非同步連結檢查叢集遠端一方之主機或透過一
個 SLIP 點對點網路介面送出命令；serial port 之 COM1、
COM2 可以使用，但其它非同步卡（ISICOM PCI cards）
則不能使用。（註：COM1 在 Linux 的裝置名稱爲
/dev/ttyS0；COM2: Linux 的裝置名稱爲/dev/ttyS1）

```
# Cluster configuration file
# Hostname of member A of cluster on LAN 1
CLUSTER_mba_lan1="rocamhsdeva-1"
# Hostname of member A of cluster on LAN 2
CLUSTER_mba_lan2="rocamhsdeva-2"
# Serial port link of member A
CLUSTER_mba_serial_port="/dev/ttyS1"
# Hostname of member A of cluster on slip LAN 3
CLUSTER_mba_lan3="rocamhsdeva-3"
# Hostname of of member B of cluster on LAN 1
CLUSTER_mbb_lan1="rocamhsdevb-1"
# Hostname of member B of cluster on LAN 2
CLUSTER_mbb_lan2="rocamhsdevb-2"
# Serial port link of member B
CLUSTER_mbb_serial_port="/dev/ttyS1"
# Hostname of member B of cluster on slip LAN 3
CLUSTER_mbb_lan3="rocamhsdevb-3"
# Cluster quorum partition
CLUSTER_quorum_partition="/dev/cciss/c0d1p1"
# Alarms E-mail
CLUSTER_alarms_email="root"
```

- 3、檔案之所有 interface 名稱須在/etc/hosts 檔案事先定義，及預設 SLIP cluster 連結之 IP 位址為 10.0.0.240 及 10.0.0.241。

/etc/hosts 檔案之內容（舉例）如下：

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1 localhost.localdomain    localhost
142.3.22.240 rocamhsdeva-1        rocamhsdeva rocamhsdeva.mkf.syseca
142.150.0.240 rocamhsdeva-2
10.0.0.240    rocamhsdeva-3
142.3.22.241 rocamhsdevb-1        rocamhsdevb
142.150.0.241 rocamhsdevb-2
10.0.0.241    rocamhsdevb-3
142.3.22.242 rocamhsdev-a
142.150.0.242 rocamhsdev-b
```

(三) Cluster Service Configuration

Cluster Service Configuration 設定檔位於 \$CLUSTER_PATH/conf/services 目錄中，每個 service 皆擁有自己 service 名稱之子目錄（\$CLUSTER_PATH/conf/service/<service-name>），且此 service 必須定義於 \$CLUSTER_PATH/conf/services.conf 檔中。

由於在 AMHS 系統僅提供一個 amss 服務，因此，在 \$CLUSTER_PATH/conf/services.conf 檔案內容只有一行，內容如下：

```
amss
```

只有在 \$CLUSTER_PATH/conf/services 目錄中產生一個 amss 子目錄，其 amss 子目錄檔案有 amss.scr、amss.dump、ipalias1、ipalias2 及 partition1 等。

1、ipalias1、ipalias2

每個服務必須針對所有 LAN IP 建立 ipalias 檔案，一個 LAN 分別建立一個 ipalias 檔案。(舉例)

2、partition1

每個服務的磁碟分割區必須指定，建立 partition1 檔案。partition1 之內容如下：

```
DEV=/dev/cciss/c0d1p2
MOUNTPOINT=/home/amss
FSTYPE=ext3
```

3、amss.scr

如果服務必須啟動一個應用程式，使用一個名稱爲 amss.scr 之 script，進行 start/stop/kill/status 等選項。

```
#!/bin/ksh
case "$1" in
    start) #start application
        echo "Start application"
        /usr/X11R6/bin/xclock &
        exit 0
        ;;
    stop) #stop application
        echo "Stop application"
        killall xclock
        exit 0
        ;;
    kill) #kill application
        echo "kill application"
        kill -9 xclock&
        exit 0
        ;;
    status) #status application
        ps -e |grep xclock
```

```
        exit  $?
    ;;
    *) #error
echo "usage :$0 {start|stop|kill|status}"
exit 1
;;
esac
```

六、 TILC 管理

(一) TILC 啓動及終止

伺服器每次重新啓動，叢集伺服器 Cluster 服務將依設定而啓動。

TILC 之手動啓動及終止方法有二：

1、採用 the Foot icon：

Foot → Programs → System → Service Configuration，在 table of services 選擇 "cluster"，然後選擇 "stop" or "start"

2、採用 Linux command "service"：

```
# service cluster start ↵ 或
# service cluster stop ↵
```

(二) TILC 管理工具 (clumgr.sh)

採用 Linux command，TILC 管理工具執行 clumgr.sh

```
# clumgr.sh ↵
```

一個選單方式，提供以下選項：

display status of service
start a service
stop a service
relocate a service
display cluster alarm
display service configuration
display cluster configuration

1、display status of service

(舉例)

<u>Member Id</u>	<u>Member Id</u>
[1] amss	[2] xclock
UP	DOWN

2、START A SERVICE

- (1) enter the name of the service (amss)
- (2) enter the member id on which the service is to be started (1,2)
- (3) clumgr replies that the service has started successfully on the local or remote member

3、STOP A SERVICE

- (1) enter the name of the service (amss)
- (2) type in "y" (yes) to confirm service stop
- (3) clumgr replies that the service has started successfully on the local or remote member

4、RELOCATE A SERVICE

- (1) enter the name of the service to be relocated (amss)
- (2) type in "y" (yes) to confirm service relocation
- (3) clumgr replies that the service has started successfully on the local or remote member and started successfully on the alternate member

5、DISPLAY CLUSTER ALARMS

- (1) clumgr returns a listing of the latest alarms concerning management of the cluster

(ALARMS 範例)

```
2004 Aug 14 04:26:51 Network interface rocamhsdevb-1 is reachable
2004 Aug 14 04:32:25 Network interface rocamhsdeva-2 is unreachable
2004 Aug 14 04:43:02 Member [2] does not have updated quorum time
2004 Aug 14 04:43:02 Remote member [2] is DOWN
2004 Aug 14 04:45:16 Remote member [2] is UP
```

(2) once a set of latest alarms has been displayed in this way, they are not included in the next clumgr alarm-display listing, but are stored in the alarms log file, name as follows :

(3) \$CLUSTER_PATH / alarm / alarms.log

(4) you can consult this log file at any time using one of the editors (vi, more, cat, ...)

6、DISPLAY SERVICE CONFIGURATION

(1) enter the name of the service (amss)

(2) clumgr returns a listing of the IP alias and disk partition configuration parameters for that service

範例: TAIWAN AMHS SERVER :

- IP alias :
 - Device : eth0 : 1
 - Name : rocamhsdev-1
 - IP : 142.3.22.234
- IP alias :
 - Device : eth1 : 1
 - Name : rocamhsdev-2
 - IP : 142.150.0.234
- Partition :
 - Device : /dev / cciss / c0d1p2
 - Mount point : / home / amss
 - File system type : ext3

7、DISPLAY CLUSTER CONFIGURATION ("READ DATABASE")

(1) clumgr returns a summary of the network and device names for the local server cluster (amss)

範例: TAIWAN AMHS SERVER :

- CONTENTS OF LOCAL CONFIGURATION
- MBA_LAN1 : rocamhsdeva-1
- MBA_LAN2 : rocamhsdeva-2
- MBA_TTY : / dev / ttyS1
- MBA_SLIP : rocamhsdeva-3
- MBB_LAN1 : rocamhsdevb-1
- MBB_LAN2 : rocamhsdevb-2
- MBB_TTY : / dev / ttyS1
- MBB_SLIP : rocamhsdevb-3

(三) TILC 日誌檔

1、TILC 維護日誌儲存在/var/log 目錄中，目前日誌檔名為 tilc_log。

2、tilc_log 日誌檔由 LOGROTATE 機制維護。

當 tilc_log 日誌檔到達規定容量（650000bytes）時，它將自動更改名字:

(1) /var/log/tilc_log .1

(2) /var/log/tilc_log .2

(3) /var/log/tilc_log .3

(4) /var/log/tilc_log .4

3、如果需要研究特定之 TILC 問題，最佳的方法是將目前的日誌檔拷貝至其他目錄後，在讀取其資料內容。

第五章 簡介 X.400 的標準和協定

壹、前言

X.400 是作為訊息網路的規範，它明定 Message Handling System (MHS) 的某些特性，該特性與系統使用者之間彼此交換訊息有關。稱為發信者的系統使用者發出訊息到 MHS，該訊息經由 message transfer system 的轉送，最後將傳遞到它的收信者（或收信者們）。用以表示訊息已經成功地傳遞到收件者或者該訊息無法傳遞的報告可以回報給發信者。X.400 的標準由以下的文件所規範：

- CCITT Recommendation X.400-X.420 Message Handling Systems (1984, 1988, and 1992), ISO/IEC 的對等文件為「ISO/IEC 10021-1,2,3,4,5,6,7:1990」

至於 X.500 則是用來規範目錄位址存取的標準，請參考以下文件：

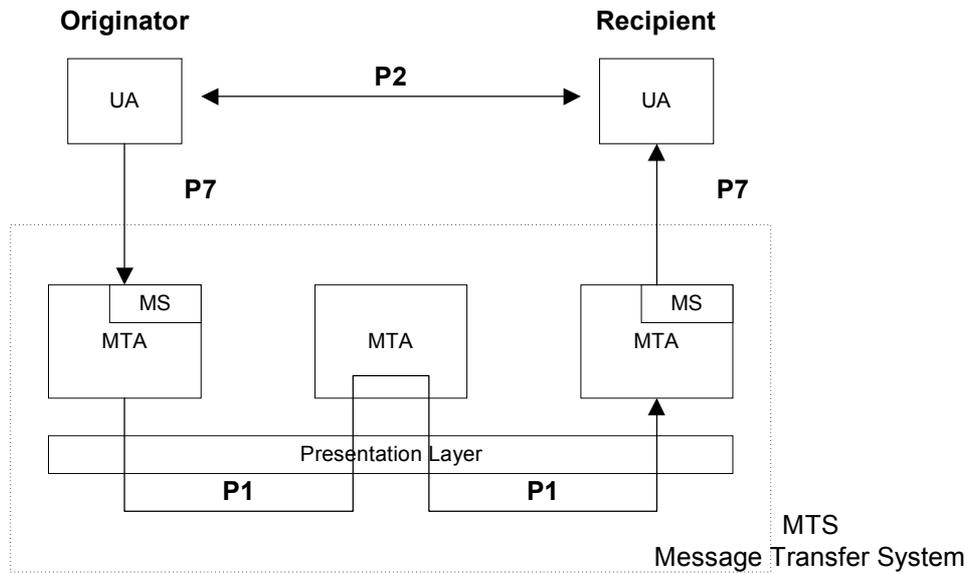
- CCITT Recommendation X.500-X.521 The Directory (1988), ISO/IEC 的對等文件為「ISO/IEC 9594-1,2,3,4,5,6,7:1990」

貳、關於 MHS, MTS, MTA, UA, AU, MS 與 AMHS 的觀念

一、X.400 網路

以下的圖 五 - 1 用以表示一個一般化的 X.400 網路，其中的各個方塊的功能解釋，請參看表 五 - 1 的說明：

圖 五 - 1 X.400 網路之方塊圖



名詞	說明
MHS	MHS 為 message handling system 的縮寫，一般而言，它就是指 X.400 網路。MHS 是由 UA, AU 和 MTS 所組成。
UA	UA 為 user agent 的縮寫，它是位於使用者和 MTS 之間的軟體介面。UA 是作為存取 MTS 服務的一個點。
AU	AU 為 access unit 的縮寫，也稱為 gateway。在 AMHS 中，AFTN/AMHS gateway 就是一個 AU。它是位於使用者和 MTS 之間的軟體介面。UA 是作為存取 MTS 服務的一個點。
MTS	MTS 是 message transfer system 的縮寫，它是由 MTA 所構成的網路。
MTA	MTA 是 message transfer agent 的縮寫，它是網路上的節點，用來提供 X.400 訊息的路由。
MS	MS 是 message store 的縮寫。每個 MS 都和 local 的 MTA 有關聯。凡是 UA 要發信或者是收信，都要透過 MS 來完成。
AMHS	AMHS 是 ATS message handling system 的縮寫，它是一種 MHS，可在 X.400 的協定下提供所有的 AFTN 服務。AMHS 使用一種稱為 AFTN/AMHS gateway 的 AU，提供 AFTN 與 X.400 協定之間的位址轉換以及服務，可藉此讓既有的 AFTN 系統逐漸轉移到 AMHS。

表 五 - 1 AMHS 子系統之名詞解釋

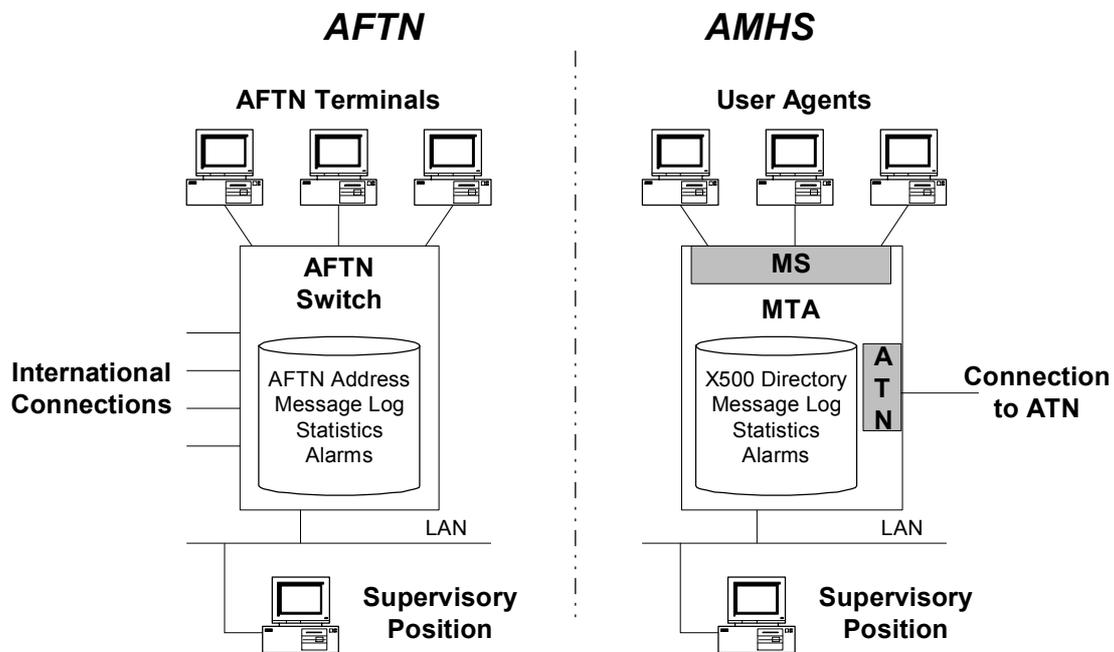
UA 使用一種名為 P7 的 X.400 協定來和它所關聯的 MS 溝通；而 MTA 和 MS 的對話則是透過稱為 P3 的協定；MTA 與 MTA 之間的轉送信件則是使用

P1 的協定；至於 UA 與 UA 彼此的對談則是使用 P2 協定。

ATN 是 Aeronautical Telecommunication Network 的縮寫，它是一種全球性的網路，用以連接國際上航空相關的社群，提供飛航管制、天氣預報和機場服務等資料的交換。讓運用現代標準格式及通訊協定的各式資料型態都能夠安全地進行交換，這就是 ATN 的目標。在 ATN 中，intermediate system (IS) 扮演 ATN router 以及 gateway (在各種不同協定之間轉介) 的角色。

二、AMHS 系統的優點

圖 五 - 2 從舊的 AFTN 系統轉換成 AMHS 系統



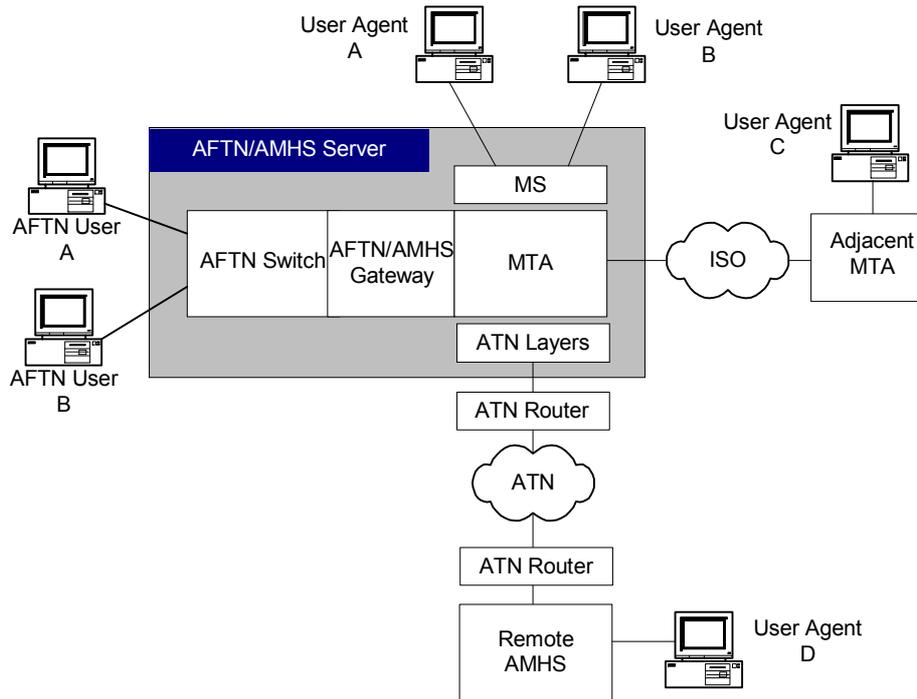
從傳統的 AFTN 環境轉換到 AMHS 之後，可以取得以下的優點：

- 傳遞訊息時有較佳的完整性。(這是由於使用 X.400 標準的緣故。)
- 提高對於訊息大小的限制(在傳統 AFTN 的訊息不可大於 2100 個字元)
- 放寬對於內容的限制。(因此，現在可以傳遞二進制的氣象以及航圖資料。)
- 可以放棄老舊的非同步通訊方式。

三、AMHS 的各式 routing 型態

以下我們將以許多實例來解釋 AMHS 系統中透過 ATN 所作的各種不同型態的 routing。而這些例子所使用的網路組態就如圖 五 - 3 所示，至於圖中各個方塊圖的說明，請看表 五 - 2 的解說。

圖 五 - 3 以這個 AMHS 網路組態為例來說明各式各樣的 routing 型態



- AFTN User A 和 AFTN User B 都是連到 local AFTN switch 的 AFTN 直接使用者。
- User Agent A 和 User Agent B 都是連到 MTA 的 AMHS 直接使用者。
- User Agent C 是鄰接 MTA 上的 AMHS 直接使用者。
- User Agent D 是經由 ATN 連到 local AMHS 上的 AMHS 直接使用者。
- Local MTA 直接連到鄰接的 MTA。
- Local AMHS 經由 ATN 連到遠端的 AMHS。

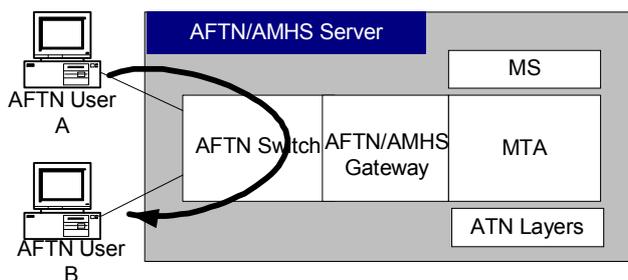
表 五 - 2 關於圖 五 - 3 之方塊圖的說明

(一) 由 AFTN 使用者連到 AFTN 使用者

1、訊息由 AFTN User A 送到 AFTN User B

來自 AFTN User A 的訊息是透過 local AFTN switch 而直接轉到 AFTN User B，請參看圖 五 - 4。

圖 五 - 4 訊息由 AFTN User A 送到 AFTN User B



(二) 由 AFTN 使用者送到 AMHS 使用者

1、從 AFTN User A 到 User Agent A

訊息先被 switch 收到，然後由 AFTN/AMHS gateway 進行轉換，並由 local MTA 轉送到與 User Agent A 有關聯的 MS，最後才到 User Agent A。請參看圖 五 - 5。

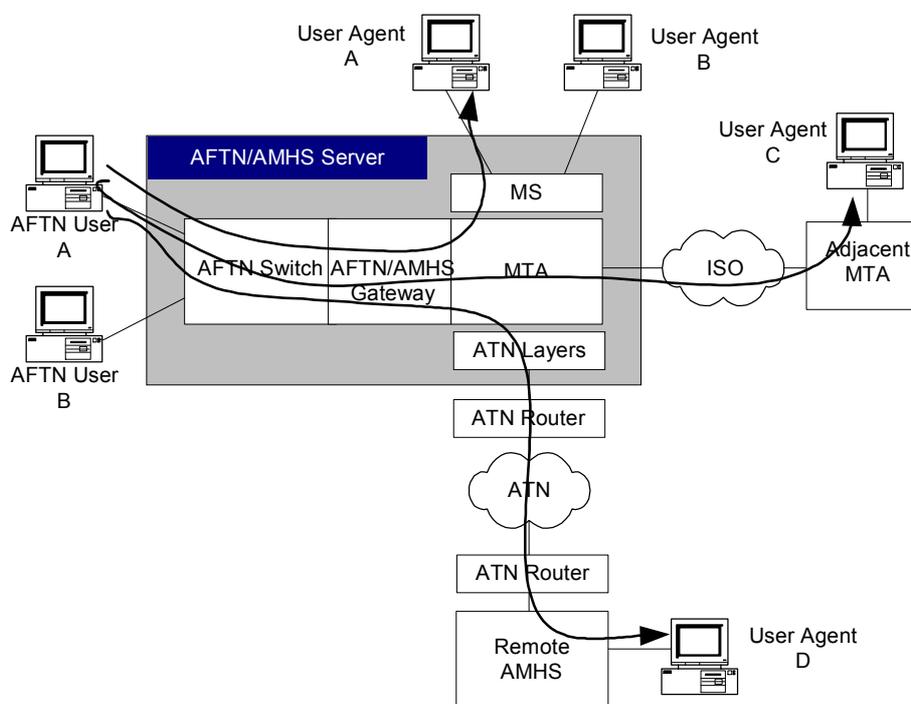
2、從 AFTN User A 到 User Agent C

訊息先被 switch 收到，然後由 AFTN/AMHS gateway 進行轉換，並由 local MTA 轉送到鄰接的 MTA，最後從鄰接的 MTA 傳送到 User Agent C。請參看圖 五 - 5。

3、從 AFTN User A 到 User Agent D

訊息先被 switch 收到，然後由 AFTN/AMHS gateway 進行轉換，並由 local MTA 透過 ATN 轉送到遠端的 AMHS 系統，然後傳送到 User Agent D。請參看圖 五 - 5。

圖 五 - 5 由 AFTN 使用者送到 AMHS 使用者



(三) 由 AMHS 使用者送到 AFTN 使用者

1、從 User Agent A 到 AFTN User A

User Agent A 發出的訊息傳給 MS，然後由 local MTA 傳遞到 AFTN/AMHS gateway，在 gateway 進行格式的轉換後再交給 local AFTN switch，最後經由 AFTN 網路交給 AFTN User A。請參看圖 五 - 6。

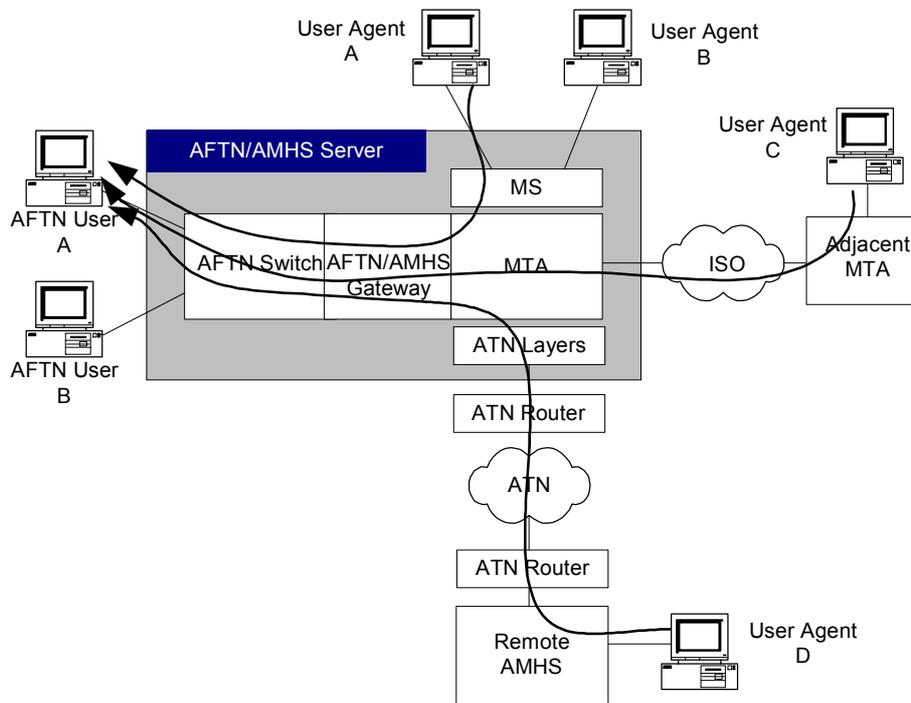
2、從 User Agent C 到 AFTN User A

User Agent C 發出的訊息會被鄰接的 MTA 傳給 local MTA，然後再傳遞到 AFTN/AMHS gateway，在 gateway 進行格式的轉換後再交給 local AFTN switch，最後經由 AFTN 網路交給 AFTN User A。請參看圖 五 - 6。

3、從 User Agent D 到 AFTN User A

User Agent D 發出的訊息會被遠端的 AMHS 系統透過 ATN 傳給 local MTA，然後再傳遞到 AFTN/AMHS gateway，在 gateway 進行格式的轉換後再交給 local AFTN switch，最後經由 AFTN 網路交給 AFTN User A。請參看圖 五 - 6。

圖 五 - 6 由 AMHS 使用者送到 AFTN 使用者



(四) 由 AMHS 使用者送到另一位 AMHS 使用者

1、從 User Agent A 到 User Agent B

User Agent A 發出訊息到 MS，而 local MTA 把訊息放到 MS 中保留給 User Agent B 的區域。請參看圖 五 - 7。

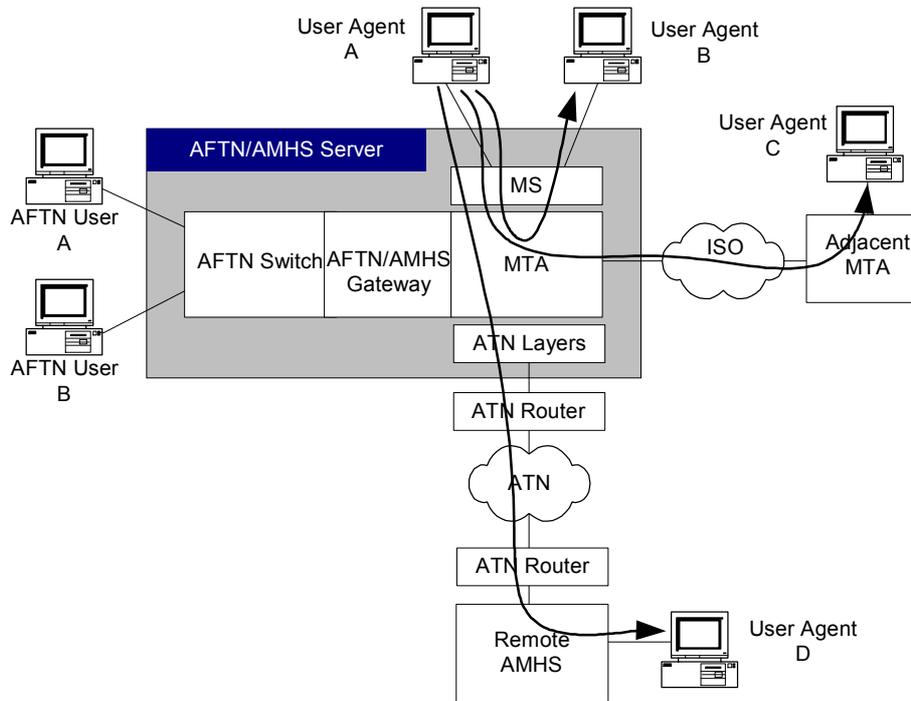
2、從 User Agent A 到 User Agent C

User Agent A 發出訊息到 MS，而 local MTA 把訊息轉給鄰接的 MTA，它再給 User Agent C。請參看圖 五 - 7。

3、從 User Agent A 到 User Agent D

User Agent A 發出訊息到 MS，而 local MTA 把訊息透過 ATN 轉給遠端的 AMHS 系統，它再給 User Agent D。請參看圖 五 - 7。

圖 五 - 7 由 AMHS 使用者送到另一位 AMHS 使用者



參、 訊息、report、probe、IPM、IPN

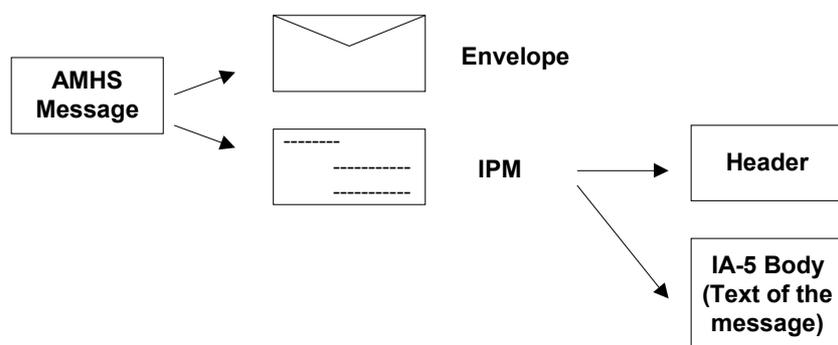
在 X.400 網路上所交換的訊息一般通稱為 object，而這些 object 分成三種基本型態：

- Message(訊息)
- probe
- report

一、 訊息

一個 AMHS 的訊息(Message)是由若干部分所構成，請參看圖 五 - 8：

圖 五 - 8 AMHS 訊息的結構



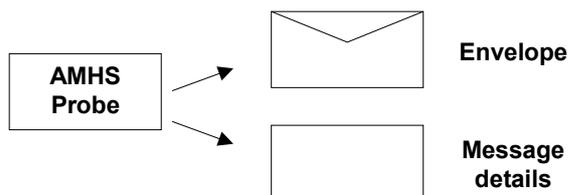
訊息(Message)傳輸的封套(envelope)包含了訊息的內容，在 AMHS 系統中它通常就是 IPM(inter-personal message)。IPM 是由一個 header 和一個 body 所構成。Header 所包含的資訊也可以在一般的電子郵件中找到，例如：主旨、寄件人的地址、收件人的地址、複本的收受者、電子郵件的 reference id、所回覆的舊信內容等等。Body 則是由一個或多個 body part 所構成，每個 body part 都代表一份 text，這就相當於一般電子郵件中的本文和附件。不過，在 AMHS 中 IPM 的 body 只有一個 body part，而它的內容就是飛航電報。事實上，通過 AMHS 系統所傳遞的訊息中，大多數都是屬於 AMHS message。另外，AMHS gateway 的主要功能之一就是從事 AMHS message 與 AFTN 之間格式的轉換。

裝在訊息傳輸的封套(envelope)中的內容型態也可以是 IPN (inter-personal notification)。IPN 可用來表示收件者已讀取了 IPM，在此情況下的 IPN 稱為收到通知 (receipt notification, RN)；另外，它也可以用來表示 IPM 無法寄達，此時稱為無法寄達通知 (non-receipt notification)。只有在 IPM 的收件者欄位中明白地註明需求時，才會有 IPN 送出。在 AMHS 中傳送 RN 就相當於在 AFTN 協定中，為了確認收到等級為 SS 的高優先權訊息而回傳 SS acknowledge 型態的 service 訊息。請注意，只有在優先權為 SS 的訊息中，它的 IPM 才會包含 RNR (receipt notification request, RN 需求)，而 IPM 並不是使用單獨的一個 RNR 欄位來規範所有收件者的需求，事實上，每個 IPM 當中所提及的收件者欄位中都有 RNR 的資訊，以便對每個收信者訂出個別的 RNR 需求。

二、Probe

AMHS 的 probe 的構造如圖 五 - 9 所示：

圖 五 - 9 AMHS 的 probe 的構造

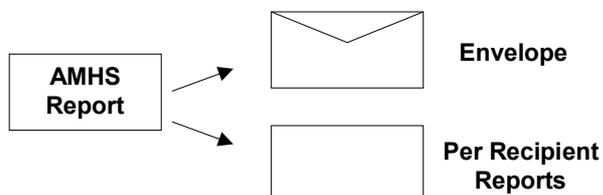


裝在 Probe 傳輸封套(envelope) 中的是一組傳送 (transfer) 欄位 (field)，它用來描述遞送訊息 (message) 的路徑。MTA 必須檢查 probe 的內容，藉以判斷是否可行，並以 report 回覆。Probe 和它所對應的 message 之間非常類似，唯一的差別在於 probe 實際上並不會把 IPM 給送出去，它只會指出 IPM 的大小而已。MTA 若看到 probe 所提出的 IPM 長度太大而無法處理的話，可以據此回覆拒絕遞送。

三、Report

AMHS 的 report 的構造如圖 五 - 10 所示：

圖 五 - 10 AMHS 的 report 的構造



裝在 Report 傳輸封套(envelope) 中的資訊包括一組屬性，藉以描述對傳送前一個 message 或 probe 的反應。AMHS 的 report 明白指出所要報告的 probe 或訊息 (message) 的主旨 (subject)，而且它有一串的報告資料，每個都是對不同的收件者所作的資料。針對 message 所作的 report 用以描述是否成功地傳遞到收件者；至於對 probe 所作的 report 則是用來表示可否成功地遞送。對於無法成功遞送的情況，Report 之中會有診斷的資訊，用以解釋失敗的原因。

肆、 message access、message transfer 與 message store

在 X.400 中，message access、message transfer 與 message store 都有明確的定義，詳情請看以下的說明。

message access 指的是以下的程序：message submission、message delivery、probe submission、report delivery 以及和 X.400 P3 協定相關的功能。在 1984 年版的 X.400 協定中，由於沒有 message store，所以 message submission 與 delivery 是直接在 user agent(UA)和 MTA 之間進行。

1988 年版的 X.400 引入 message store(MS)之後，MS 和 MTA 之間有了 message access 的運作，而 UA 和 MS 直接對話，MS 可以貯存 UA 想要寄出的訊息，也可以用 P7 協定的命令從本地端的 MTA 收取訊息。

Message transfer 則是指在 message transfer system 中對於 message、probe 或 report 物件的交換以及路由的決定。這些在 MTA 之間傳遞訊息的協定就稱為 X.400 的 P1 協定。請參看表 五 - 3。

伍、 P1 ， P2 ， P22 ， P3 與 P7 協定

X.400 是由一組稱為 P1、P2、P22、P3 與 P7 的交換協定所構成，並且在 OSI 的表現層（presentation layer）是使用 ASN.1（abstract syntax notation 1）的方式來表示。請參看表 五 - 3。

名詞	名詞解釋
P1	這是一種 message transfer 協定，它支配以下三類的程序： <ol style="list-style-type: none"> 1. message、probe 與 report 等物件在 MTS（message transfer system）之中傳遞時路徑的決定。 2. message、probe 與 report 等物件在 MTA 彼此之間的交換。 3. MTA 所提供的服務。
P2	這是 1984 提出的一種 inter-personal messaging protocol，在對話時它是對等（peer）的 user agent 所關注的一種協定。它規定在 UA 之間所交換的 IPM 與 IPN 的屬性內容，以及要如何使用 IPN 來回應 IPM。
P22	這是 1988 提出的一種 inter-personal messaging protocol，它是 P2 的加強版，因為它包含更多的屬性，特別是 heading 與 body 的擴充屬性。
P3	這是一種 message access protocol，它支配 UA 與 MTA 之間或者是 MS 與 MTA 之間的 message submission、message delivery、probe submission、report delivery 等的相關程序。
P7	這是一種 message store protocol，用來支配 UA 和 MS 之間的資料交換。UA 可以傳送 message 到另一個 user 或者傳送 probe 到遠端的 MTA。UA 可以對 MS 發出請求，要它列出貯存在 MS 的 message（以及 RN 和 report）的清單。UA 可以從 MS 一個一個地抓取（fetch）前述的物件，並檢查其內容（對於 IPM 而言，這就構成了「收信」或「讀信」）。UA 也可以刪除貯存在 MS 之中的 message。
ASN.1	以上述的 P1 到 P7 等協定在 X.400 網路進行交換的資料與命令，其格式都是屬於 abstract syntax notation 1（ASN.1）。ASN.1 是一種表現層（presentation layer）的規範，要透過 X.400 網路來傳輸的各種型態的資料與各種大小的資料的話，就要使用此規範來表示。凡是要透過 X.400 交換的資料都要套用這個格式的規定，包括每個命令的 prefix，也包括已標準化的 Type 與 Length 欄位的屬性。

表 五 - 3 X.400 相關的名詞解釋

陸、O/R 地址、O/R 名稱與 MF、XF 及 CAAS 地址

O/R 是個縮寫，意謂“originator/recipient”。O/R 地址與名稱是依據 X.400 協定中用來表示地址（尤其是 message 的收件者與寄件者的地址）的定址規格。

在 AMHS 的應用中，使用者的 O/R 地址也稱之為 MF 地址（MHS form address），以便和 AF 地址（AFTN form address 或 AFTN indicator）有所區別。MF 地址可以再細分為 XF、CAAS 和 other 地址。以下將針對此再詳加說明。

O/R 地址是一個與順序有關的屬性所構成的集合，在給定的領域中它可以定義出唯一的使用者。特別是 MTA，它根據收件者的 O/R 地址就可以把訊息遞送給這位特定的使用者。

建立 O/R 地址的方法有很多種，可以由各種不同的屬性型態、字母及數字所構成；但在這裡所用的是指 O/R 地址的助憶（mnemonic）形式，包括以下各種的屬性：

- Country Name
- Administration Management Domain (ADMD) Name
- Private Management Domain (PRMD) Name
- Organization Name", "Organizational Unit Name
- Common Name

O/R 名稱可以是由以下二者所構成，也可以是由以下任何一個所構成：

- O/R 地址
- X.500 目錄名稱（directory name）

其中目錄名稱是個選項，如果出現目錄名稱而且又當 O/R 地址沒有被使用或者是個無效的 O/R 地址的話，那麼目錄名稱就會被 MTA 用來作為遞送的依據。

MF 名稱當中的 Country Name、ADMD Name 和 PRMD Name 等屬性就構成了 AMHS 的管理領域（AMHS management domain），而這些屬性的資料是由 ITU-T

以及 ICAO 所指定，其規定如下：

- Country Name：資料為“XX”。
- ADMD Name：資料為“ICAO”。
- PRMD Name：資料為國家名稱或者是 ICAO 的國家代碼（ICAO nationality letters）。

MF 名稱中其餘屬性的內容就和它所使用的“定址方案（addressing scheme）”有關，而方案則有三種，請參看表 五 - 4。

定址方案的名稱	使用的時機	概要說明
CAAS (Common AMHS Addressing Scheme)	這是用來簡化地址產生程序的方案	基本上就是把 AFTN 的 indicator 存在 common name 的屬性中。
XF (XF-Addressing Scheme)	對於尚未建置 AMHS 的 ICAO 會員國或者是宣布採用此方案的會員國，此方案提供產生這些國家地址的預設方法。	<ul style="list-style-type: none"> ● Organization Name 設成 AFTN ● Organizational Unit Name 設成 8 個字元的 AFTN indicator 或 AFTN 名稱
其他的 addressing scheme	如果前述的二種方案都不可行的話	強制到 X.500 的目錄伺服器中去找 AFTN indicator 或者 O/R 名稱。

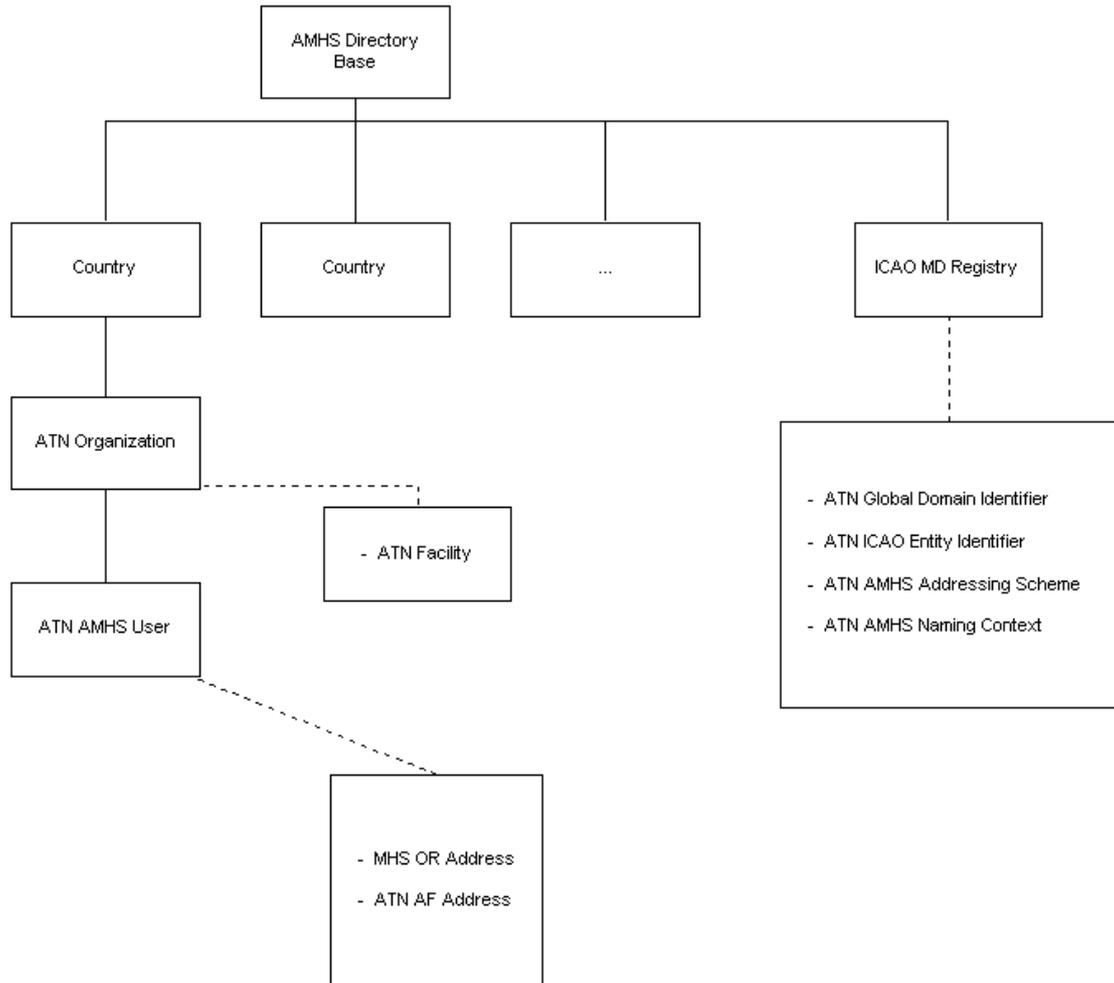
表 五 - 4 AMHS 的定址方案

為了能夠進行 AF 與 MF 之間的地址轉換，ATS message server 必須能夠存取 X.500 目錄伺服器。至於轉換的細節則可以參考 ICAO Document 9705 ATN Standards and Recommended Practices (ATN SARPs)。

圖 五 - 11 用來表示記載於目錄伺服器當中資料的結構，在最上層有許多的分支，其中之一是 ICAO MD Registry，其他的分支則是對應到每一個所要定址的國家。圖形中標示著“ICAO MD Registry”、“Country”、“ATN Organization”和“ATN AMHS User”的長方形則代表目錄中主要的紀錄（entry）型態；其他以虛線連接的長方形則列出該紀錄的屬性，而這些屬性將在地址轉換時扮演重要的

角色。

圖 五 - 11 X.500 目錄的結構



一、由 AFTN indicator 轉成 AMHS O/R 地址之步驟

先到 ICAO MD Registry 的分支，在各個記錄中找出 “ATN ICAO Entity Identifier” 屬性與 AFTN indicator 相同或相近者，其比對的標準如下：

- 與 AFTN indicator 的前 7 個字元相同者；或者，
- 與 AFTN indicator 的第 1,2,5,6,7 個字元相同者；或者，
- 與 AFTN indicator 的前 4 個字元相同者；或者，
- 與 AFTN indicator 的前 2 個字元相同者

從該筆紀錄的“ATN AMHS Naming Context”屬性就可得知接下來要到那一個 country 的分支去找更多的資料；至於“ATN AMHS Addressing Scheme”的屬性則記載該國家的 MF 地址的型態（CAAS、XF 或者是其他）。

設定 O/R 地址屬性的過程，我們把它歸納成以下的表 五 - 5。請注意，一旦轉換的過程中出現任何的問題，將使用 XF 地址作為輸出的型態。

O/R 地址屬性	MF 地址的型態		
	CAAS	XF	其他
Country Name	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“XX”	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“XX”	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“XX”
ADMD Name	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“ICAO”	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“ICAO”	取自“ATN Global Domain Identifier”屬性的資料，通常該資料就是“ICAO”
PRMD Name	取自“ATN Global Domain Identifier”屬性的資料，例如：Aena、CFMU、France 等	取自“ATN Global Domain Identifier”屬性的資料，例如：Aena、CFMU、France 等	取自“ATN Global Domain Identifier”屬性的資料，預設值為 ICAO 二個字母的國家代碼
Organization Name	取自“ATN Organization”紀錄中名為“ATN Facility Name”屬性的資料	填入“AFTN”等四個字母	到“ATN AMHS User”紀錄中找尋屬性“ATN AF Address”與 AFTN indicator 相同者，然後由該筆紀錄的“MHS OR Address”決定本屬性的資料
Organizational Unit Name	AFTN indicator 的前 4 個字母	整個 AFTN indicator（註：8 個字母）	到“ATN AMHS User”紀錄中找尋屬性“ATN AF Address”與 AFTN indicator 相同者，然後由該筆紀錄的“MHS OR Address”決定本屬性的資料

Common Name	整個 AFTN indicator (註：8 個字母)	(沒有使用)	到“ATN AMHS User”紀錄中找尋屬性“ATN AF Address”與 AFTN indicator 相同者，然後由該筆紀錄的“MHS OR Address”決定本屬性的資料
Other O/R attributes	(不適用)	(不適用)	到“ATN AMHS User”紀錄中找尋屬性“ATN AF Address”與 AFTN indicator 相同者，然後由該筆紀錄的“MHS OR Address”決定本屬性的資料

表 五 - 5 從 AMHS 地址轉換成 AFTN indicator 之對照表

二、由 AMHS O/R 地址轉成 AFTN indicator 之步驟

先到 ICAO MD Registry 的分支，在各個記錄中找出“ATN Global Domain Identifier”的資料中與 AMHS 地址的“Country Name”、“ADMD Name”和“PRMD Name”相符者，從該筆紀錄的“ATN AMHS Naming Context”屬性就可得知接下來要到那一個 country 的分支去找更多的資料；至於“ATN AMHS Addressing Scheme”的屬性則記載該國家的 MF 地址的型態（CAAS、XF 或者是其他）。注意，如果 AMHS O/R 地址的“Organization Name”是由“AFTN”這四個字元所組成的話，那麼這個地址就被當成 XF 地址來處理。

從 AMHS O/R 地址中抽取出 AFTN indicator 的方式如表 五 - 6 所示：

AFTN	MF 地址的型態		
	CAAS	XF	其他
AFTN indicator (8 個字元)	從 AMHS O/R 地址中的“Common Name”屬性中得知	從 AMHS O/R 地址中的“Organizational Unit Name”屬性中得知	到“ATN AMHS User”紀錄中找尋“MHS OR Address”屬性與 AMHS O/R 地址相同者，然後由該筆紀錄的“ATN AF Address”決定本屬性的資料

表 五 - 6 從 AFTN indicator 轉換成 AMHS 地址

柒、X.500 目錄伺服器

目錄伺服器就是一個 X.500 地址的資料庫，提供地址目錄服務的標準功能。

而這些地址的資訊包含以下各項：

- X.400 所使用的各種 O/R 地址的欄位
- AFTN indicator 以及所對應的 O/R 地址
- 其他額外的定址資訊，例如：傳真號碼、電話號碼、電子郵件信箱等。

AMHS 廣泛地使用目錄伺服器，以從事 X.400 地址與 AFTN indicator 之間的互換。在讀取與寫入目錄資料時，是採用標準的 DAP 或 LDAP 協定。

心得

第六章 出國受訓之心得

本次的訓練課程不僅有理論的課堂課程，也有實地進行軟體操作與軟硬體安裝的實習課程。由於理論和實際得以相互印證，因此收穫頗為豐碩。以下列舉重要的心得。

壹、明白 AMHS 的運作流程

出國前由於並沒有實地操作 AMHS 系統的機會，因此先前對於 AMHS 的概念相當抽象，只有大略知道它的運作方式與電子郵件極為類似。上過訓練課程之後就得到更具體的觀念，明白訊息在 OSI 的第六層（也就是表現層，presentation layer）必須以 ASN.1 的格式編碼（註：詳細資料請參看附錄 A）；至於類似電子郵件地址的 AMHS O/R address 也有一套規定（註：詳細資料請參看第五章 陸、O/R 地址、O/R 名稱與 MF、XF 及 CAAS 地址）。

AMHS 系統是使用網路作為傳遞的媒介，不再使用非同步專線。如此一來，整個 AMHS 網路就會比舊系統強健許多，換句話說，網路上萬一有任何的線路故障發生，ATN router 就會自動繞道改走其他的路線。另外，網路化的傳輸本身就包含了資料的錯誤偵測與控制，因此就能確保資料的完整性（integrity），而這正是現行的 AFTN 系統所欠缺的部分。（註：詳見第五章 貳、二、AMHS 系統的優點）

上了這個課之後，對於 AMHS 的各個組成份子（例如：UA、MTA、MS 等）之間的通信協定（例如：P3、P7 等）才有更清楚的了解。（註：詳見第五章 伍、

P1，P2，P22，P3 與 P7 協定)

貳、了解叢集 (cluster) 架構

爲了提供高可用性，AMHS 的伺服器都是採用叢集的架構，以避免一旦重要的硬體故障就無法提供服務。

叢集架構的概念很容易明瞭，就是把重要的硬體設計成雙套，彼此互爲備援。至於硬體錯誤的偵測以及如何從故障的主機切換到另一部主機，其實就是要靠主機上的軟體。整個詳細的資料，請參看附錄 B 貳、架構介紹。

參、熟悉軟體維護程序

由於實地操作是訓練的重點之一，所以我們可以進行以下各項的練習，以利日後的軟體維護。

一、操作指令的練習

操作指令大致分成三種：啓動系統、停止系統運作以及檢視系統運行的狀態。對於線上實際運作的系統而言，除了檢視系統運行狀況的指令較常用之外，其他的平常根本很少用得到。訓練期間可以盡情地不斷練習啓動以及停止系統，可以讓軟體維護人員熟悉平時罕用的指令。

二、系統重新安裝的練習

Thales-IS 公司對於 AMHS 的每個子系統都有軟體安裝文件，我們也趁受訓的機會把每個系統按照文件重新安裝一次，如此一來除了可以熟悉安裝程序之外，同時也可以順便驗證文件的正確性。

建議

第七章 建議事項

本次出國受訓有下列之建議事項：

1. 對國外訓練師資及實習環境的提供，建議在規格書部份，作較嚴謹的規範，要求投標商必須根據規格書之規範在投標文件中提出訓練計劃及實習環境。
2. 關於國外訓練時段的安排，建議在系統已完成 FAT(陣地測試)準備階段進行，且相關維護文件皆已完成的情況下才能進行。因為如此一來才能讓派訓人員於受訓期間實際操作，一方面既能熟悉系統維護，另一方面又可以驗證文件的正確性。
3. 各種訓練以及維護相關文件廠商都需提供電子檔，以便於攜帶。
4. 此 AMHS 系統建置程序，其涵蓋 AFTN、ATS Message、ATN Router 及 LDAP 伺服器。得標廠商 Thales 依據其功能項目，分別委由不同廠家設計開發，再由 Thales 進行功能整合、測試及驗收作業，此舉雖可節省廠商開發設計時程及人力，但考量 AMHS 系統未來之更新及維護作業，Thales 也許有維修時程無法掌握以及欠缺獨立維護能力之虞；故建議爾後系統規格訂定時，需將考量廠商獨立開發系統之能力及整合能力。
5. 針對 AMHS 系統之 CLUSTER 叢集架構，它可針對硬體異常即可自動切換至備援之伺服器功能，可達到飛航情報資訊的服務不致中斷；但若亦能加入針對 AMHS 系統之軟體服務異常也能夠提供自動切換的話，那麼功能就更完美。

附錄

附錄A ASN.1 的簡介

壹、前言

ASN.1 的全名為 Abstract Syntax Notation One，AMHS 的訊息在 presentation layer 的表示法就是採用 ASN.1。本文將就 ASN.1 作一個概括的介紹，想進一步了解細節的話，請自行參閱 X.208、X.209 和 X.411 等文件。

貳、資料格式

簡單說來，一個以 ASN.1 資料格式所表示的物件，可以細分成是由 T、L 和 V（也就是 Type、Length、Value）三個部分所構成，請參看圖 A。

圖 A ASN.1 資料格式



以下將就格式中的每個部分加以介紹。至於關於格式的詳細資料，請自行參看 X.409。

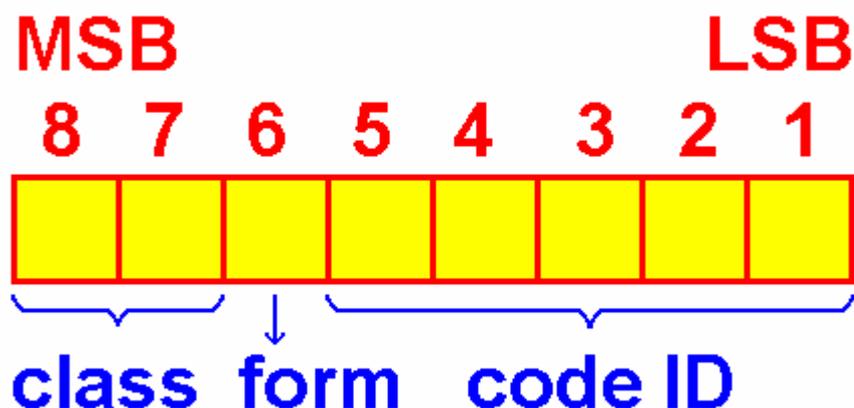
一、Type

它的目的在描述 Value 的種類，以便於解讀 Value 中的資料。其格式可以根據 byte 數目的不同而分為二種：

(一)Type 是由一個 byte 所組成

這個 byte 的八個 bit 可以再細分成 class、form 和 code ID 等欄位，請參看圖 B。

圖 B Type 的資料格式（長度為一個 byte）



第七和第八個位元構成 class 欄位，其意義如下表所示：

Class 欄位的值	所代表的意義
0 0	Universal
0 1	Application
1 0	Context
1 1	Private

第六個位元稱為 form 欄位，其意義如下表所示：

Form 欄位的值	所代表的意義	備註
0	Primitive	代表 Value 是基本的資料型態
1	Constructor	代表 Value 並非為基本的資料型態，而是一個由「T、L、V」所構成的物件。

(二)Type 是由許多個 byte 所組成

前一種形式的 Type，其 code ID 的數量受到侷限。如果想獲得更多不同

種類的 code ID 的話，請使用現在所介紹的另一種形式。請參看圖 C，其中第一個位元組的 bit 1 到 bit 5 都設成 1，這樣子代表說實際的 code ID 是由後續的位元組當中以綠色表示的位元所構成。請注意，第 2 個到第(N-1)個位元組的 bit 8 皆為 1，而第 N 個位元組的 bit 8 則為 0，以代表它是最後一個位元組。

圖 C Type 的資料格式（長度不止一個 byte）



第一個位元組的 class 和 form 的意義則和上一節所提到的完全相同。

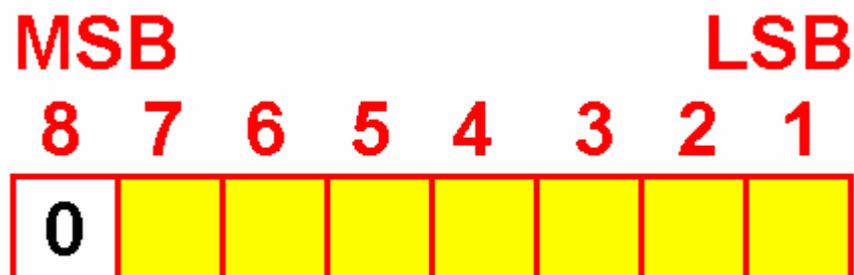
二、Length

Length 的目的在於表示 Value 的長度，而它總共有三種不同的表示法：

(一)Short form

Length 的資料僅由一個位元所組成，而且 MSB 必須為 0，Value 的長度由其餘的七個位元所表示。換言之，本表示法僅適用於 Value 的長度小於 128 個位元組。請參看以下的圖 D：

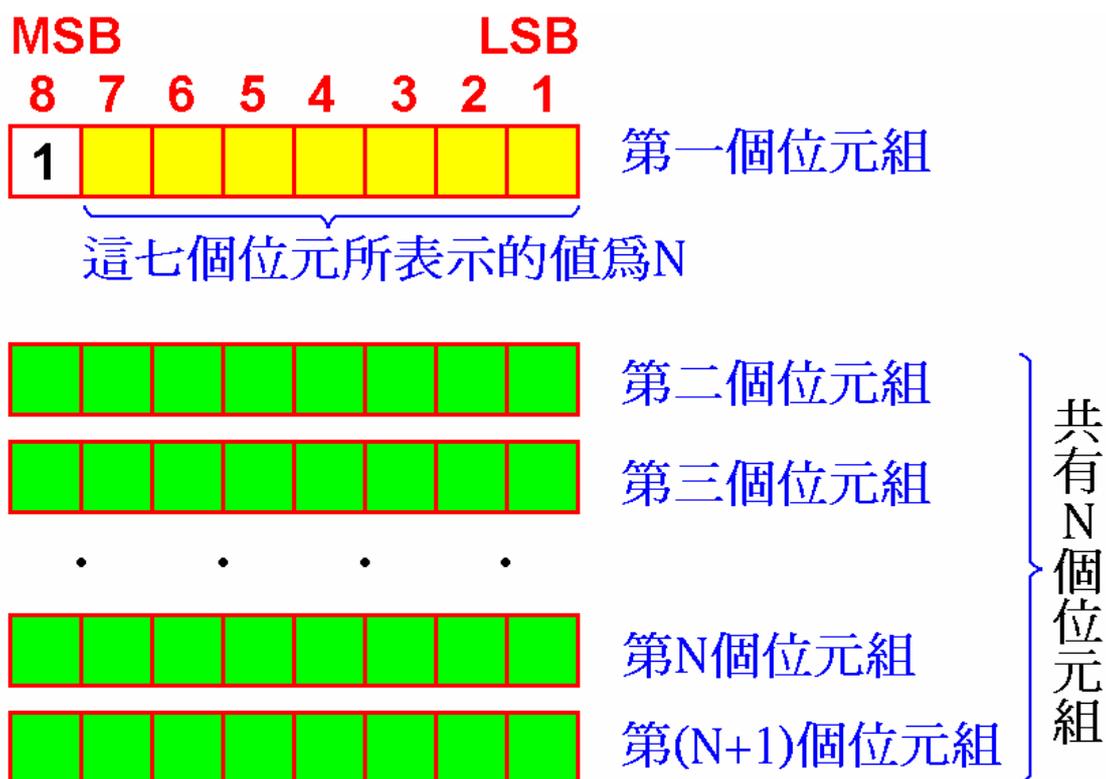
圖 D Length 的格式為 short form



(二)Long form

本表示法適用於 Value 資料的長度不小於 128 且不大於 21008 個位元組。為便於解說，請參看以下的圖 E。

圖 E Length 的格式為 long form

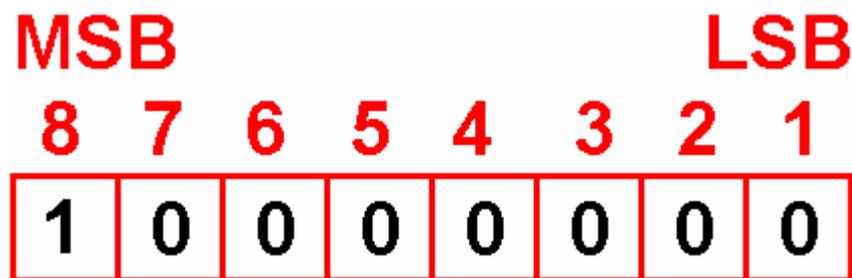


第一個位元組的 MSB 必須為 1，而其餘的七個位元所代表的數值為 N（註：N 最大為 126，因為 127 保留起來不可使用），這就表示 Value 資料的長度是由第一個位元組以後的 N 的位元組的數值來表示（註：也就是以綠色所表示的部分）。本表示法將使用(N+1)個位元組。

(三)Indefinite form

本表示法的格式為一個位元組，其數值固定為 H80，如圖 F 所示：

圖 F Length 的資料格式為 indefinite form



其意義為不指明 Value 的長度。因為 Value 的資料將從下一個位元組開始，一直到連續二個 H00 之前的位元組為止。

三、Value

Value 可以是由基本型態 (primitive) 的資料所構成，也可以是由複合型態 (constructed) 的資料所構成。所謂的複合型態的資料就是一個由 Type、Length 和 Value 所構成的。

參、實例

一、例子一

敘述	一個由「ALTAS」等 5 個字元所構成的字串
結果	04 05 41 54 4C 41 53
說明	<ul style="list-style-type: none"> 字串的 code ID 為 4，而 class 為 universal 的 primitive 資料型態，因此 Type 的編碼為 04。 由於字串長度由五個字元所構成，因此 Length 的資料為 05。 其餘的五個字元就是該字串的 ASCII 值。

二、例子二

敘述	BOOLEAN 資料 false 的表示
結果	01 01 00
說明	<ul style="list-style-type: none"> Boolean 資料的 code ID 為 01，而 class 為 universal 的 primitive 資料型態，因此 Type 的編碼為 01。 Value 只有二種可能：00 代表 false，FF 代表 true。二者都是由一個字元所構成。 由於 Value 長度由一個字元所構成，因此 Length 的資料為 01。

三、例子三

敘述	integer 資料的表示。本例子用來表示整數 2。
結果	02 01 02
說明	<ul style="list-style-type: none">integer 資料的 code ID 為 02，而 class 為 universal 的 primitive 資料型態，因此 Type 的編碼為 02。數值 2 可以用一個字元來表示，因此 Value 的資料為 02。由於 Value 長度由一個字元所構成，因此 Length 的資料為 01。

四、例子四

敘述	Bit string 資料的表示。本例子用來表示由 0、0、1 等三個位元所構成的 bit string。
結果	03 02 05 20
說明	<ul style="list-style-type: none">Bit string 資料的 code ID 為 03，而 class 為 universal 的 primitive 資料型態，因此 Type 的編碼為 03。(註：其實 bit string 也可能是 constructed 的資料，在此情況下，Type 的編碼為 23。在本例中，我們使用 primitive 的表示法)Value 由二個部分所構成：第一個部分為第一個字元所構成，其他的字元構成第二個部分。第二個部分用來表示 bit string，由於第二個部分的字元所能容納的 bit 數目不小於所要表示的 bit string，所以，在第二個部分的最後一個字元當中會有若干個 bit 是用不到的，這些用不到的 bit 數目就記錄於第一個部分的字元中。以本例而言，由於資料只有三個 bit，所以第二個部分的資料只需一個字元，該字元中有五個 bit 並不會用到，所以第一個部分的資料為 05。第二個部分的那個字元中的前三個 bit 為 0、0 和 1。其他五個 bit 的值並不重要，任何值都可以，而我們選擇把這五個 bit 都填 0，所以第二個部分的字元為 20。由於 Value 長度由二個字元所構成，因此 Length 的資料為 02。

附錄B TILC 的功能規格

壹、前言

TILC 的目的在於提供硬體的高可用性 (HA, high availability)，也就是可以容忍硬體上所出現的故障。請注意，TILC 並不是用來防範軟體上的問題，萬一它所跑的 Application 出現異常，cluster 的架構無助於防範此類問題的發生。

另外還有一點值得特別注意：TILC 在機器開機 (boot) 或重新啓動 (reboot) 之後並不會自動執行 cluster 的功能，因此也就沒有提供服務，必須由操作人員以管理工具執行之後才能恢復。

貳、架構介紹

TILC 主要是由二部 server (我們稱爲 server A 和 server B) 以及一部 shared disk array 所構成。server 和 shared disk array 是由 SCSI 纜線所連接；而二部 server 之間的連線則是由二張網路卡透二個獨立的區域網路(我們稱爲 LAN1 和 LAN2) 相連，另外也藉由一條 NULL MODEM 的纜線連接彼此的 COM port (我們稱爲 SLIP 連線)。

TILC 的主要任務在於管理以下三種的資源：

- Cluster interface name：簡言之，就是 TILC 內部的二部 server 的 IP 要如何對應到外界看到 TILC 的代表 IP。
- Shared disk array：這個共享的區域由二部 server 所分享，但同一個時間不會有二部 server 一同使用。

- Application：就是 TILC 對外提供服務的應用程式，同一個時間不會有二部 server 同時執行。

我們上面的文字中提及，只有一部 server 會對外提供服務，我們稱之為 active member (簡稱為 MBA)，另外一部則稱為 standby member (簡稱為 MBB)，因為它在必要時可以接手。

參、CLUSTER 的運作

以下我們將列舉各種不同的狀況，針對二部 server 的角色加以解釋：

一、CLUSTER 的啟動

這是由執行管理工具來進行，以下將敘述二部 server 所做的工作。

(一)Active member (MBA) 所採取的行動：

- 對於 MBA 在 LAN1 和 LAN2 上面的網路介面，產生 cluster alias interface。
- 檢查 shared disk array 的 file system。
- 掛載 (mount) shared disk array。
- 啟動由 cluster 所管理的 application。
- 啟動監督程式 (monitoring daemon)，以便檢查 active member 的狀態以及 standby member 的可用性 (availability)。

(二)Standby member (MBB) 所採取的行動：

- 啟動監督程式 (monitoring daemon)，以便檢查 active member 的可用性 (availability)。

二、CLUSTER 的停止

這是由執行管理工具來進行，以下將敘述二部 server 所做的工作。

(一)Active member (MBA) 所採取的行動：

- 停止執行 cluster 所管理的 application 。
- 移除 LAN1 和 LAN2 上面的 cluster alias interface 。
- 把所有存取 shared disk array 的 process 都停止執行。
- 卸載 (umount) shared disk array 。
- 停止執行監督程式 (monitoring daemon) 。

(二)Standby member (MBB) 所採取的行動：

- 停止執行監督程式 (monitoring daemon) 。

三、手動進行 Cluster Relocation

所謂手動進行 cluster relocation，就是操作人員藉由執行管理工具，來將二部 server 的 active 及 standby 角色互換。因為我們可以在 MBA 也可以在 MBB 執行管理工具，所以就這二種方式分別列舉各種情況加以討論可否完成 cluster relocation。

(一)從 Active member 發出 Relocation 的命令

情況	無法從 LAN1 ping 到 standby member	無法從 LAN2 ping 到 standby member	無法從 SLIP link ping 到 standby member	Active member 無法存取 shared disk array	Standby member 無法存取 shared disk array	是否能完成 manual relocation	備註
1						可以	
2	●					可以	
3		●				可以	
4			●			可以	
5				●		—	由 standby member 上的 monitoring daemon 進行 automatic relocation
6					●	不行	
7	●	●				不行	

8				●	●	—	由 standby member 上的 monitoring daemon 進行 stop cluster
9	●	●		●		—	由 standby member 上的 monitoring daemon 進行 automatic relocation
10	●	●			●	不行	
11	●	●	●		?	不行	因為無法 ping 到 standby member，所以 TILC 判定它不正常，因此無法 relocation
12	●	●	●	●	?	不行	因為無法 ping 到 standby member，所以 TILC 判定它不正常，因此無法 relocation。在此情況下建議 stop cluster，因為 active member 無法存取 shared disk array

(一) 從 Standby member 發出 Relocation 的命令

情況	無法從 LAN1 ping 到 active member	無法從 LAN2 ping 到 active member	無法從 SLIP link ping 到 active member	Active member 無法存取 shared disk array	Standby member 無法存取 shared disk array	是否能完成 manual relocation	備註
1						可以	
2	●					可以	
3		●				可以	
4			●			可以	
5				●		—	由 standby member 上的 monitoring daemon 進行 automatic relocation
6					●	不行	
7	●	●				不行	
8				●	●	—	由 standby member 上的 monitoring daemon 進行 stop cluster
9	●	●		●		—	由 standby member 上的 monitoring daemon 進行 automatic relocation

10	●	●			●	不行	
11	●	●	●	?		—	由 standby member 上的 monitoring daemon 進行 automatic relocation
12	●	●	●	?	●	—	由 standby member 上的 monitoring daemon 進行 stop cluster

(二) 自動進行 Relocation / Stop

我們首先定義何謂 cluster automatic relocation：當 standby member 偵測到符合自動進行 relocation 的標準時，它就進行 relocation。其次，所謂的 cluster automatic stop 就是當 standby member 偵測到符合自動進行 stop 的標準時，它就停下整個 cluster。至於在何種情況下才會進行 automatic relocation 或 automatic stop，請參看以下的列表：

情況	無法從 LAN1 ping 到 active member	無法從 LAN2 ping 到 active member	無法從 SLIP link ping 到 active member	Active member 無法存取 shared disk array	Standby member 無法存取 shared disk array	Automatic relocation	Automatic stop	alarm	備註
1									
2	●							●	
3		●						●	
4			●					●	
5				●		●			在 standby member 執行的 monitoring daemon 會偵測到 active member 無法存取 shared disk array，因此就會進行 automatic relocation
6					●			●	
7	●	●						●	

8				●	●		●	●	<p>在 standby member 執行的 monitoring daemon 會偵測到 active member 無法存取 shared disk array，但由於 standby member 本身也無法存取 shared disk array，所以就進行 automatic stop。</p> <p>【注意】當關閉 shared disk array 的電源時就會出現這種情況。</p>
9	●	●		●			●	●	<p>在 standby member 執行的 monitoring daemon 會偵測到 active member 無法存取 shared disk array，因此就會進行 automatic relocation</p>
10	●	●			●			●	
11	●	●	●				●	●	
12	●	●	●	●			●	●	<p>由於 standby member 無法從 LAN1、LAN2 或 SLIP link ping 到 active member，因此 TILC 認定 active member 已經當機，所以會進行 automatic relocation</p>
13	●	●	●		●		●	●	
14	●	●	●	●	●		●	●	<p>由於 standby member 無法從 LAN1、LAN2 或 SLIP link ping 到 active member，因此 TILC 認定 active member 已經當機，不過因為 standby member 也無法存取 shared disk array，所以就進行 automatic stop</p>

當 automatic relocation 發生時，系統所要完成的工作如下所示：

狀況	系統所要完成的工作
若 MBA 無法提供服務（例如：power off、system crash）	<ul style="list-style-type: none"> 由 MBB 啟動 cluster，並成為新的 active member
若 MBA 無法存取 shared disk array，但 system 並未 crash	<ul style="list-style-type: none"> 停掉所有由 cluster 所啟動的 process 移除 MBA 在 LAN1 與 LAN2 的 cluster alias interface 由 MBB 啟動 cluster，並成為新的 active member

當 automatic stop 發生時，系統所要完成的工作如下所示：

狀況	系統所要完成的工作
若 MBA 無法提供服務（例如：power off、system crash）	<ul style="list-style-type: none"> 把 MBB 所執行的 monitoring daemon 給停掉
若 MBA 無法存取 shared disk array，但 system 並未 crash	<ul style="list-style-type: none"> 停掉所有由 cluster 所啟動的 process 移除 MBA 在 LAN1 與 LAN2 的 cluster alias interface 把 MBB 所執行的 monitoring daemon 給停掉

肆、Cluster 的效能

在自動進行 relocation 時所花費的切換（switchover）時間，包含偵測硬體故障的時間以及停掉並啟動 application 的時間。

偵測硬體故障所需要的時間與故障的情況有關，以下表格列舉若干例子並加以計算：

故障情況	偵測過程	花費的時間
三個網路介面（LAN1、LAN2、SLIP link）故障	判定一個網路介面故障的標準為：每秒 ping 一次，如果連續三次都無法 ping 到，則認定為故障。由於故障的介面有三個，因此有 9 個 ping 沒有回應。	9 秒
二個網路介面	判定一個網路介面故障的標準為：每秒 ping	6 秒

(LAN1、SLIP link) 故障	一次，如果連續三次都無法 ping 到，則認定為故障。由於故障的介面有二個，因此有 6 個 ping 沒有回應。	
位於 active member 和 shared disk array 之間的 SCSI 纜線發生斷線	判定一個 SCSI 介面故障的標準為：每秒 ping 一次，如果連續三次都無法 ping 到，則認定為故障。	3 秒

根據編號為「56 638 813-306」的文件，它提到切換時間和“停掉並啓動 application”有很大的關係。該文件僅就一部分的 application (AERMAC) 進行實地測試，數據如以下的表格所示：

由 cluster 所管理的 application	手動 relocation 的 switchover 時間	自動 relocation 的 switchover 時間 (方法是把 active member 給 shutdown)	自動 relocation 的 switchover 時間 (方法是拔掉 active member 和 shared disk array 之間的 SCSI 纜線)
AERMAC	10 秒	15 秒	9 秒
AMHS	尙未測試	尙未測試	尙未測試
AMSS (即 AERMAC+AMHS)	尙未測試	尙未測試	尙未測試

測試所用的硬體／軟體如下所示：

類別	說明
硬體	伺服器：Compaq Proliant ML370 G2 SCSI 卡：Smart Array 532 Shared Disk Array：Compaq Smart Array Cluster Storage with one logical volume at RAID 1 網路介面：二個 LAN 以及一個 SLIP link
軟體	Linux 版本：RedHat 7.2 Kernel 版本：2.4.7-10 SCSI 卡的驅動程式版本：2.4.30

伍、注意事項

根據編號為「56 638 813-306」的文件，開啓電源之後，shared disk array 所花的 initialization 時間要比 server 開機的時間還要長，因此會造成 server 找不到 shared disk array 並等待使用者按下 F1 鍵。請勿按 F1 鍵，而是等 shared disk array 完成 initialization 之後，再 reboot server。