

壹、目的

「航空氣象現代化作業系統」(簡稱 AOAWS), 奉行政院核定「交通部民用航空局中長期計畫」, 此計畫的主要目的為以 MM5 中尺度數值模式建立台北飛航情報區之氣象預報資訊系統, 並整合了航空氣象資訊, 建立更人性化之顯示介面。本計畫期程為八十七年七月至九十二年六月, 共計五年。目前已經架設好軟硬體於台北、中正以及高雄氣象台和諮詢台等單位, 且正值測試階段的期間, 對於系統的使用與模式結果的信心測試, 尚有許多待努力之處。

職二人奉派接受「航空氣象現代化」第三年產品操作訓練, 旨在學習 MM5 中尺度氣象模式執执行程序、模式產品顯示操作以及低空風切預警系統的操作和顯示資料還原處理。藉由這些訓練, 期能深入瞭解整個系統的設計原理和整體架構, 進一步改善缺失並增加系統操作的熟悉度, 善盡航空氣象對飛安之責任。

貳、 過程及課程內容

職二人在美學習的課程如下

十一月

22(Thu) : 抵達 Boulder

23(Fri) : Thanksgiving Holiday

26(Mon) : Celia 為我們介紹辦公室及其它設備.並帶我們到 Building2 樓下辦通行證.碰巧照像機器故障了,因此,下午我們到 Mesa 去照證,順便參觀一下.

27(Tue): Frank 為我們介紹 MDS 的各項功能及操作.並詢問我們有關於我們在使用上的一些問題.

28(Wen)

10:00 氣象衛星 (Rm 2143D FL1 - Dave Johnson)

2:00 雲幕和能見度 (Rm 2143D FL1 - Kevin Petty)

29(Thu)

10:00 AOAWS Flight Categories and METAR products (Rm 2143D FL1 - Niles Oien)

2:00 AOAWS MDS Parameter Files (AOAWS LAB - Frank Hage)

30(Fri)

10:00 AOAWS Satellite and Winds and Temp products (Rm 2143D FL1 - Niles Oien)

2:00 Introduction to Python (Rm 2143D FL1 - Gerry Wiener)

十二月

3(Mon)

10:00 COMET case study (Rm 2143D FL1)

2:00 AOAWS WMDs Overview (AOAWS LAB - Deirdre Gavey)

4(Tue) :MM5 模式介紹和個案研究 (Rm 2143D FL1 – 徐曉明)

5(Wed) : MM5 模式介紹和個案研究(Rm 2143D FL1 – 徐曉明)

6(Thu) :MM5 模式介紹和個案研究(Rm 2143D FL1 -徐曉明)

7(Fri)

9:00 AOAWS Project Meeting (RAP Conf Rm)

11:00 FAA Oceanic Weather Research (Tenny Lindholm)

12:00 Lunch at Orchid Pavilion Restaurant

2:00 FAA Ceiling and Visibility Research (Paul Herzegh)

10(Mon)

9:30 FAA Convective Weather Research (Cindy Mueller)
 10:30 FAA Turbulence Research Program (Bob Sharman)
 1:00 Visit Vaisala

11(Tue)
 9:00 Visit NOAA FSL
 2:00 Visit NOAA NWS
 3:00 MM5 case study (Rm 2143D FL1 - Hsiao-ming Hsu)

12(Wen)
 9:00 AOAWS LLWAS case study report (Dave Johnson)
 10:00 Visit Radiometrics Corp (Stick Ware)
 2:00 RAP Christmas Party

13(Thu)
 10:00 MM5 case study (Rm 2143D FL1 - Hsiao-ming Hsu)
 2:00 AOAWS MM5 model overview (RAP Conf Rm - Jordan Powers and Jim
 Bresch)

14(Fri)
 10:00 AOAWS MOS product (Rm 2143D FL1 - Jaimi Yee)
 2:00 Introduction to Python (Rm 2143D FL1 - Gerry Wiener)

17(Mon)
 10:00 Introduction to UNIX/Linux (ASG Rm - Jeff Stolte)
 介紹基本的指令。
 2:00 Introduction to RAP computer network (Rm 2143D FL1 - Tres
 Hofmeister)
 介紹基本的 IP 原理和實際的網路設施。

18(Tue)
 10:00 Linux Installation (ASG Rm - Jeff Stolte)
 NCAR 使用 Debian Linux 的程式和系統更新方式。
 2:00 High Res. Radar Obs. using Mult. Freq. Range Imaging (Rm 1022
 FL2 - Tianyou Yu)
 余天佑介紹數值方法分析高解雷達觀測在亂流和水氣的結果。

19(Wed)
 10:00 Verification (Rm 2143D FL1 - Barb Brown)
 2:00 Windshear Meteorology (Rm 2143D FL1 - Bill Mahoney)

20(Thu)
 10:00 Introduction to 3DVAR (RAP Conf Rm - Wei Huang)
 1:30 AOAWS Icing product (Rm 2143D FL1 - Mike Dixon)
 2:30 Introduction to RAP computer network (Rm 2143D FL1 - Tres
 Hofmeister)
 徐曉明: TIMAX, TISTEP

21(Fri)

10:00 AOAWS ITFA product (Rm 2143D FL1 - Mike Dixon)

2:00 MM5 Modeling/COMET case study (Rm 2143D FL1)

Mike: the algorithm of ITFA

24(Mon) : Christmas Holiday

25(Tue) : Christmas Holiday

26(Wed) : All day - MM5 Modeling/COMET case study (Rm 2143D FL1)

27(Thu)

10:00 AOAWS LLWAS/WSP systems (Rm 2143D FL1 - Dave Johnson)

11:30 AOAWS LLWAS data analysis software (Rm 2143D FL1 - Dave Johnson)

2:00 Intro to Python (Rm 2143D FL1 - Gerry Wiener)

28(Fri)

9:30 Field Trip to ML

2:00 MM5 Model/COMET case study (Rm 2143D FL1)

31(Mon) : All day - MM5 Model/COMET case study (Rm 2143D FL1)

課程內容整理如下

一、MM5 數值預報模式

MM5 全名為 Fifth-Generation Pennsylvania State University/NCAR Mesoscale Model, 為美國賓州大學首先研發, NCAR 持續發展之, 目前為止已發展至第五代系統。MM5 在全球氣象界擁有為數眾多的使用者, NCAR 每年亦有為入門使用者舉辦課程 (MM5 Tutorial Class), 國內部分亦有相當多的 MM5 相關研究論文發表於國內外科學期刊。

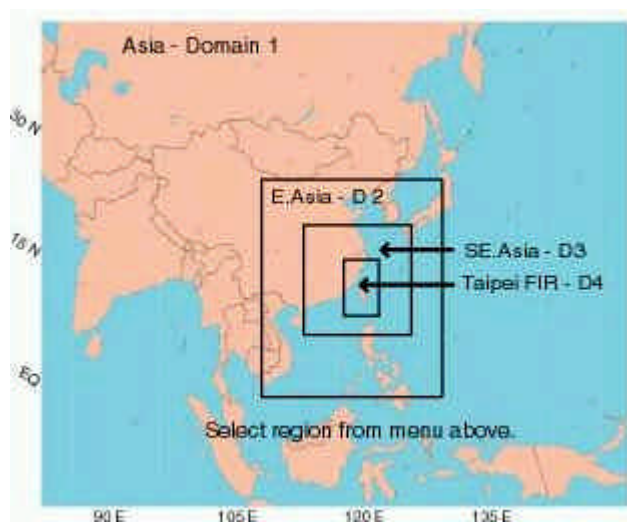
在 NCAR 受訓期間, MM5 部分主要由徐曉明博士負責介紹, 徐博士來自台灣, 所以在語言上沒有問題, 可以介紹的更為詳細, 學得的東西也較多, 感覺上就像回到研究所時代跟著指導教授一樣。

MM5 為三維空間非靜力巢狀網格數值模式, 其架構可大概區分為前置作業、主程式、後置處理三大部分。底下詳細說明之:

(1) 前置作業:

此部分又可分為地形資料、氣象場、海水表面溫度三部分。

地形資料對預報模式來說是不可或缺的最基礎資料，若以研究為目的的模式則地形資料不一定需要真實地形，可以設計若干人為設定的理想地形或簡化地形以進行模式的敏感度分析或動熱力研究。當然，本局 AOAWS 系統內的 MM5 使用的是真實地形，其範圍最廣可包含全球解析度 1 公里，地形越細越能掌握小尺度的氣象系統如局部環流與地形效應等，但也會需要更長的計算時間甚至造成模式本身或電腦硬體的不穩定而異常終止。本局目前使用四個巢狀網格，網格間距分別為 135 公里、45 公里、15 公里、5 公里。範圍如下圖所示：



MM5 中處理地形的模組為 TERRAIN。

地形資料準備完後，接著處理初始氣象場與海水表面溫度，這部分共使用三個模組分別為 REGRID、3DVAR (或 LITTLE_R) 與 INTERP。因為 MM5 為網格數值模式，所以必須在每一個網格點 (GRID POINT) 都有初始氣象資料提供給主程式計算。REGRID 模組將全球分析資料放到 MM5 所設定的水平網格點上就定位，3DVAR 模組可將觀測資料如測站地面觀測、高空觀測、雷達資料等等加入初始場中，以補初始場無法解析較小尺度天氣現象之不足。INTERP 進一步將資料插分至模式設定的較高解析度之垂直分層上，並將資料由等壓面轉到 MM5 的垂直座標 SIGMA 層上。經過這三個模組的處理後，所有網格點上所需的資料皆已備齊，且是經過客觀分析方法剔除不合理或錯誤資料後的模式所需初始場。AOAWS 系統之

MM5 初始分析資料目前使用中央氣象局全球分析資料，而一般研究用的可使用 ECWMF 或 NCEP 或其他全球分析資料，只要知道資料的格式及結構，即可以增加 VTABLE 的方式增加至 REGRID 模組內，不過這個工作並不容易。經由 INTERP 模組所輸出的檔案，除了初始環境場外亦包含地面邊界條件與側邊界條件場。

在此要說明是，這些模組都是經由參數設定檔或 SHELL SCRIPT 語言來更改模組的設定，一般使用者不需去更改以 FORTRAN 77 或 FORTRAN 90 所撰寫的原始程式碼。且對一般使用者來說，資料格式或不同參數所代表之物理意義及其對模式輸出會有何影響的瞭解較為重要。

(2) MM5 主程式

MM5 主程式為 MM5 系統中最主要的計算模組，讀取使用者依需求所設定之參數及上述之初始場進行各種氣象場的運算，最後依使用者所設定的時間間隔輸出結果，此結果為 BINARY 的檔案並非圖形檔，需再加上後置處理依使用者所需畫出圖形或進行分析運算。MM5 的參數化方法非常多包含不同氣象學家或團隊所發展的不同物理意義的方法，有些已經比較成熟有些則還在發展中。有些參數化方法可以有較詳細的物理考量，但必須花費甚多的計算時間，因此一般作業預報模式無法採用，另為自動化作業考量，作業預報模式無法如研究用模式經常改變參數化的設定。

AOAWS 所選用的參數化方式如下：

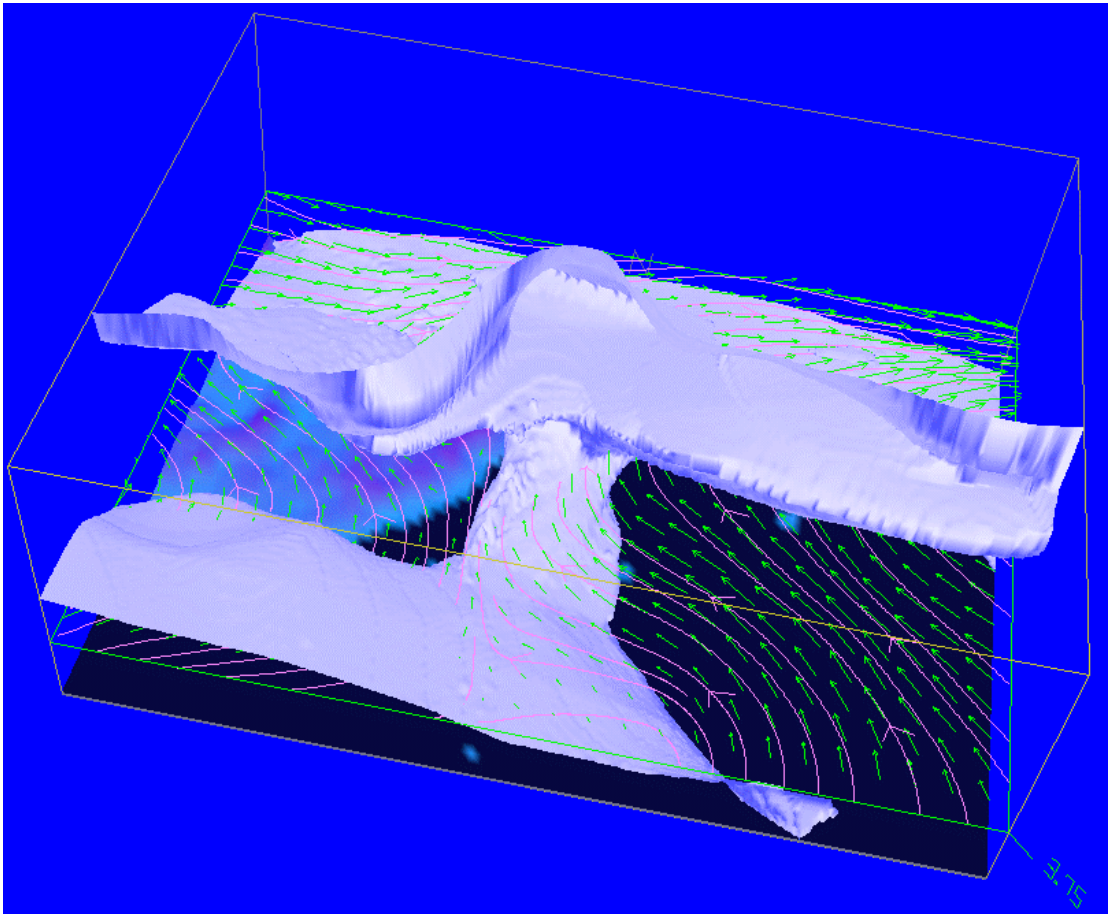
- 1.積雲參數化：使用 Kain-Fritsch 法，此法同時預報積雲內之上衝氣流與下衝氣流之特性，包含積雲降水過程，也考慮風切作用在降水效率上的影響。
- 2.行星邊界層參數化：使用 MRF 法，此法適用於高解析度的模式上。使用隱式垂直擴散法以能適用於較長之時間間距。
- 3.微物理參數化：使用 Mixed-Phase (Reisner) 法，考慮過冷水滴的存在，允許緩慢的冰融化過程。不考慮冰雹與冰淞過程，此法可搭配使用對照表以節省計算時間。
- 4.輻射參數化：使用 Simple Cooling 法，大氣輻射冷卻率只由溫度決定，不考慮雲的交互作用或日夜變化。
- 5.地表溫度參數化：使用 Blackadar 法，此法假設地表為一個平版且有

固定溫度的下墊面。此溫度決定於能量收支，以及日夜溫度變化可達之深度。

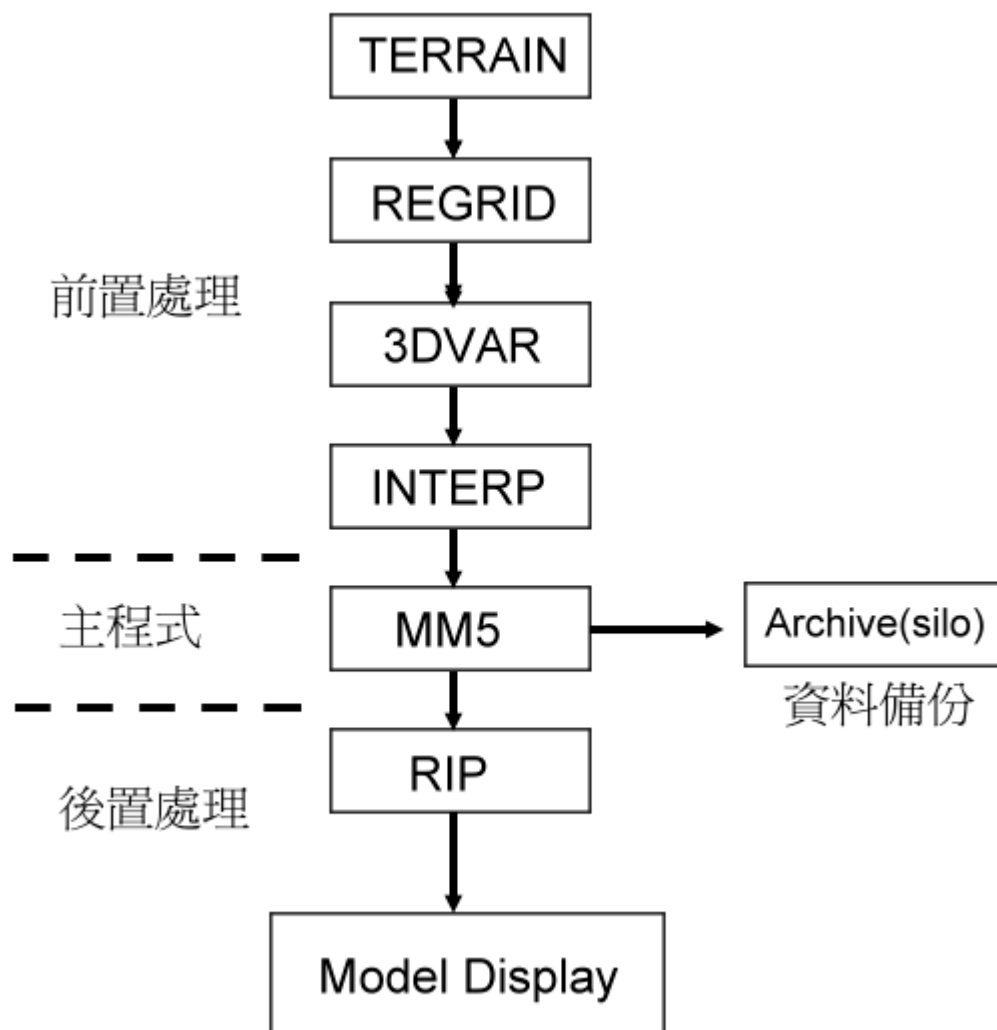
(3)後置處理

後置處理最主要為繪圖部分。不僅僅是主程式的輸出結果，前置處理部分的輸出亦可拿來繪圖以供分析使用。原先官方版本的繪圖模組為 GRAPH，但使用上及使用者設定略嫌不足，圖形輸出亦不美觀。後來有其他使用者自行依 MM5 輸出格式自行撰寫另一繪圖程式 RIP，不僅使用者操作增加及輸出美觀，還可將原先 BINARY 輸出檔拿來做資料讀取切割以利分析，最重要是當原作者覺得此程式成熟時即放到網路上供其他使用者自由下載使用。現 RIP 程式及相關說明亦收錄到 MM5 的官方網頁上，且隨者 MM5 的更新而更新。這種成果分享的作法令職等感到敬佩，別人會如此先進不是沒有原因。

此外，我們可以使用 Vis5D 這一套三度空間顯示的軟體，將模式產品作更好的展示，如圖所示為納莉颱風侵襲前台灣附近氣流的 3D 結構和風場的 2D 結構。



整個 AOAWS MM5 的流程可以下圖表示：



二、多元化氣象產品顯示系統 Multi-Dimensional Display System (MDS)

MDS 系統為一般使用者查詢航空氣象產品的最終端顯示，(另一為網頁形式的顯示稱為 WMDS，在此並不討論)。MDS 的產品顯示類似氣象局的 WINS 二代系統，但著重的內容並不一樣。氣象局的 WINS 二代偏重於一般氣象產品顯示，並不包含如詳細雲組雲霧高、能見度、亂流、積冰等等航空氣象所需的氣象場，時間與空間尺度的劃分相對於航空氣象所需亦稍嫌不足，且無法提供所選定的航路預報。MDS 系統則是專為台灣地區航空氣象所設計，產品顯示除一般氣象資訊外，最重要亦整合了航空氣象要素的觀測值與模式預報值甚至飛機報告內容，時間空間的解析度亦符合航空所需，垂直解析以飛航高度為單位而非一般的標準大氣分層。

拜網路科技發達之便，NCAR 為我們準備的這部分參考資料使用者手冊為中英文版，另還有設定操作介面的多個參數檔相關資料。

使用者手冊部分基本上沒有太多問題，在台北氣象台及中正氣象台職等已有相當程度的熟悉操作及使用產品資料，這部分僅對一些較細節部分再加以請教討論，也算是回饋使用心得給程式設計人員，程式設計人員可藉此設計出更符合人性化的操作方式及內容。

整個 MDS 系統啟動所需的參數設定則相當複雜分佈於數個檔案之中，如 MetarSelect.aoaws、taiwan_ndb.map、dbz_cwb.colors、CIDD.aoaws，由檔案名稱大概就可以知道這些設定檔是設定那個部分，其中 CIDD.aoaws 最龐大最複雜也最重要，主掌了整個系統操作介面的設定。由於整個 MDS 的作業環境是 LINUX 系統，因此檔案名稱的大小寫是有區別的，這點需特別加以留意。

參數設定檔本身為 Shell Script 寫成的檔案，主程式啟動時會呼叫這些設定檔，讀取其中的參數設定，諸如作業環境設定、程式開始時顯示哪些產品視窗、產品視窗的設定、產品本身色階、選項按鈕的連結及選定位置、... 等等，再以圖形按鈕或視窗的方式顯示在螢幕上，使用者可以用滑鼠點選所需產品按鈕，系統接獲使用者的要求後，將此要求傳給資料庫伺服器待

資料庫伺服器計算備妥資料後，再回傳給使用者所使用的系統並以圖形或文字的資料顯示出來。雖然過程複雜，但因電腦硬體的進步及高速網路的傳輸，所需時間甚短幾乎是立即回應產品給使用者，使用者幾乎感覺不到等待時間。

有關參數設定部分 NCAR 方面僅挑選幾個較重要部分講解，其餘部分需對程式語言及 Shell Script 檔有相當程度的瞭解才能進行設定變更，而要熟悉一種程式語言並非短期內可以達成。因此這部分是需要自行進修才能有更進一步的熟悉。

舉例來說，在 CIDD.aoaws 這個 Shell Script 檔中，可以找到下面幾行設定：

```
# Note: index starts at 1
cidd.level1_lable1:      WORLD
cidd.level1_min_xkm:     -330
cidd.level1_min_ykm:     -90
cidd.level1_max_xkm:     330
cidd.level1_max_ykm:     90
.
.
.
cidd.level16_lable1:     FIR
cidd.level16_min_xkm:    116.75
cidd.level16_min_ykm:    20.75
cidd.level16_max_xkm:    125.25
cidd.level16_max_ykm:    29.25
```

第一部分是設定水平空間尺度範圍選項為「WORLD」的按鈕所代表的範圍，以經緯度設定從 330°W~330°E，90°S~90°N。

同樣，另一部份是設定台北飛航情報區（TPE FIR）的水平範圍，選項按鈕名稱為「FIR」，顯示範圍則從 116.75°E~125.25°E，20.75°N~29.25°N。

餘此類推，前面的變數代表系統主程式中所定義的變數，後面的值則是目前程式設計人員依目前使用者所需給定的變數值，後者是可因應不同需求而更改，且不需重新編譯主程式。若要更動前面的變數名稱則必須要程式設計人員才可進行，且需重新改寫原始程式碼進行編譯的動作。

三、雲霧與能見度產品及預報

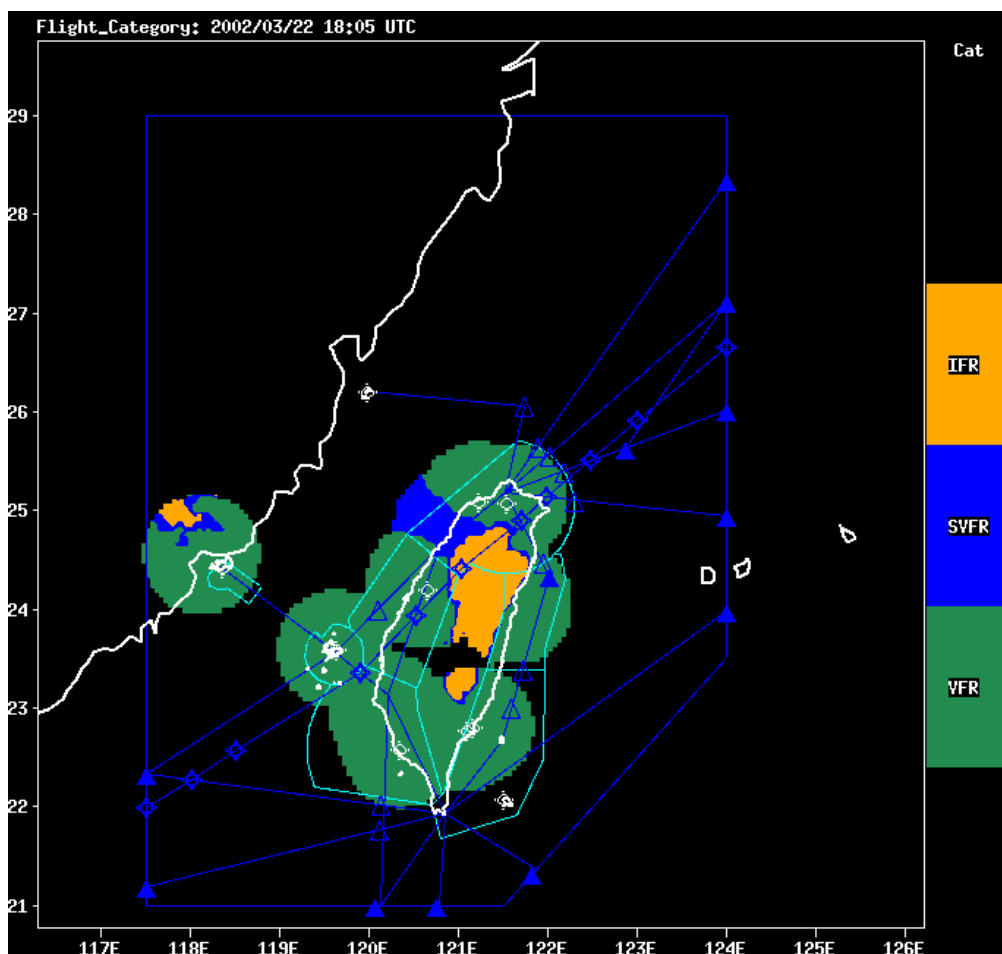
西元 1999 年 7 月 16 日美國甘乃迪家族之 JFK Jr. Tragedy 駕駛私人小飛機墜毀於美國東岸附近，NTSB 調查可能為霾（HAZE）及黑夜造成空間迷向導致 JFK Jr. Tragedy 墜毀於海面。美國遂成立 National Ceiling and Visibility Product Development Team（NCVPDT）來發展機場雲霧與能見度產品的顯示與預報，NCAR 為其中的成員之一。此部分由 NCAR 科學家 Kevin Petty 介紹，目前產品雖在發展階段尚未到達最後成熟產品，但已有不錯的表現。

雲霧與能見度的預報在氣象上來說，並不是一件容易之事，因為雲霧與能見度受外界影響甚遽變化時間可以很短暫且急遽，氣象學上並無固定之理論方程式可以解析這兩個氣象要素的變化。因此 NCAR 所使用的方法並非傳統數值預報直接由一組或數組趨勢方程式求出雲霧與能見度的預報值，而是使用類似經驗方程式，先蒐集可能影響雲霧與能見度的因子，再利用不同天氣及環境條件下不同因子配與不同的權重求得雲霧與能見度的預報。利用此方式的預報方式，資料庫及經驗方程式的建立非常重要，資料庫中所蒐集的資料越多，越可以建立更完善的經驗方程式，也就可以得到更完善的產品。

NCAR 所發展的系統中，上游資料目前包括氣象數值模式產品、衛星資料、地面觀測、雷達及飛機報告，將來若有新的考量某些因子可能會影響雲霧及能見度，亦可將這些因子加入到上游資料庫中。其中衛星部分尚包括雲種類的選定特別是低雲部分。所有資料備齊後再經由選定功能及權重比例等得到最後產品，將此產品以資料庫形式儲存，並經由網際網路方式提供使用者查尋產品及預報，其網址為 <http://www.rap.ucar.edu/~paddy/cvis2/>。

其產品包括飛航種類、能見度、雲霧高的即時觀測結果及未來數小時的預報，並可選擇之前的預報產品與即時觀測結果做比對，另還可疊加 VOF、

在使用過程中我們發現一個問題，此系統顯示介面是利用 JAVA 語言所寫成的 JAVA APPLATE，並非每種瀏覽器都支援。在 NCAR 因為基於安全性考量，並不鼓勵使用 Microsoft 之 IE 系列瀏覽器而是大部分使用 Netscape 瀏覽器，反之在台灣 IE 瀏覽器的使用率卻遠超過後者，因此在台灣地區若欲查詢相關資料可能會出現問題，經與 Dr. Kevin Petty 討論後，他說將來成熟產品會修正此操作介面的相容性問題。目前本局 AOAWS 計畫的氣象產品中也使用部分的技術及類似的產品，但操作介面不同，並無瀏覽器相容問題。本局 AOAWS 相關產品如下：



(3) 在此要說明的是，美國擁有為數甚多的 NEXRAD 二代氣象雷達及飛機報告等系統運作所需的上游資料，這對提升系統的預報準確度及預報穩定度有相當大的效用。目前國內的雷達資料尚有待整合及加強，飛機報告部分更是需要飛行員的主動提供，因為這是一個互相回饋的過程。

四、低空風切預警系統(簡稱 LLWAS)

低空風切預警系統(簡稱LLWAS)目前裝設在松山和中正航空氣象台，目前雖然已驗收，然而這只是表示系統的運作是有效的。但預警的可信度校驗和系統的調校並未完成，因此LLWAS的顧問Dave Johnson博士(NCAR)設計了一套程式，用以將LLWAS所記錄的資料重建成容易理解的圖形，以便能將其與飛機報告或是都卜勒雷達所測到的風切來做比較，並可針對觀測點及觀測位置所造成的誤差做評估。

以下為執程序與說明：

1. 錄存資料格式為一行一筆資料，如下所示

```
L,LLWAS,10Sec,RWY1D,2001,6,20,23,59,50,-91229998193783603000.000000,0,05-D 240 CALM
```

其中10Sec表此筆資料為10秒輸出一筆，
RWY1D表離開RWY1的跑道，
"2001,6,20,23,59,50"表2001年6月20日23:59:50。

2. 編輯參數檔，松山為LIwasPlot.ss，中正為LIwasPlot.cks

3. 單一時間繪圖

```
LIwasPlot -params LIwasPlot.ss -end "2001 3 24 04 29 30"
```

繪製時間結束於2001年3月24日4點29分30秒

4. 時序繪圖

```
LIwasPlot -params LIwasPlot.ss -start "2001 3 24 04 50 00" -end  
"2001 06 22 05 00 00" -delta 60
```

繪製時間開始於2001年3月24日4點50分，結束於2001年6月22日5點，
-delta 60 表示繪圖間格為60秒。

5. 所需檔案：

LIwasPlot.cks: CKS參數檔(見參考資料一)

LIwasPlot.ss: SS參數檔

LIwasPlot2: 執行檔(適用於Linux作業系統)

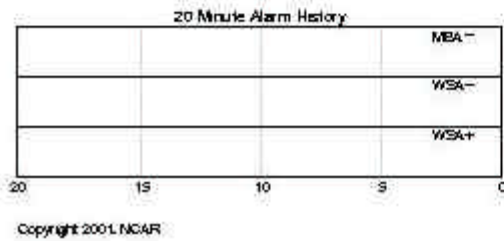
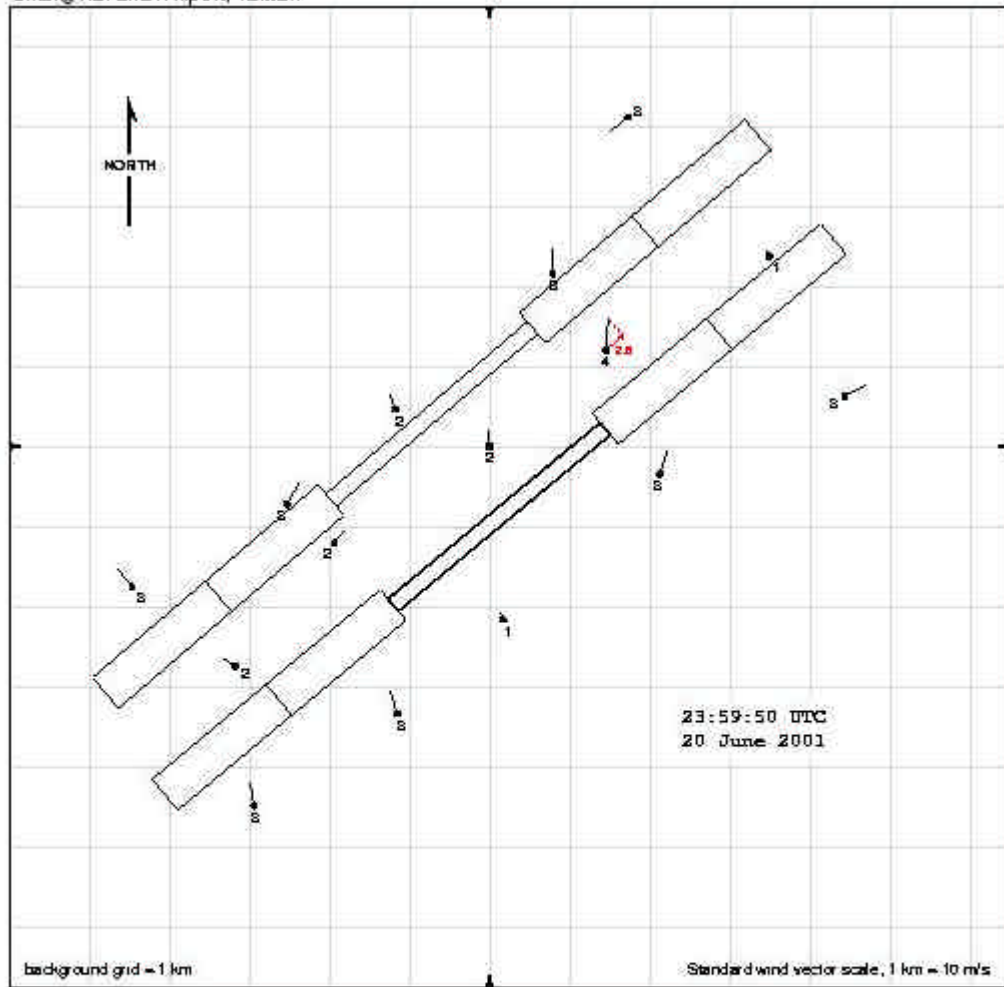
prologue_cks.ps: ps參數檔(適用於CKS見參考資料二)

prologue_ss.ps: ps參數檔(SS)

6. 執行結果

如下圖所示

LLWAS Event Reconstruction
Chiang Kai Shek Airport, Taiwan



Reconstructed AAD

05-A	200	3
05-D	240	CALM
23-A	240	CALM
23-D	200	3
06-A	190	3
06-D	230	CALM
24-A	230	CALM
24-D	190	3
CF 170	CALM	23:59:50

五、氣象衛星與衛星資料

課程內容見於<http://silk.rap.ucar.edu/~djohnson/satellite/coverage.html>及

http://silk.rap.ucar.edu/~djohnson/satellite/satellite.html#top_of_page

主要的內容包含地球同步衛星和繞極衛星原理的介紹 應用在氣象上的衛星頻道(水氣頻道和紅外線頻道)，以及氣象衛星產品的特性。

六、網路

主要介紹內容為 TCP/IP 的架構、佈線設計，以及防火牆的設計。

1. TCP/IP 的設定:

- (1) IP address:主機的數字代號
- (2)網路遮罩(netmask):用以與 IP address 位置作邏輯 AND 運算，以得到所在網路的地址。
- (3)網路位址(network address):代表某個網路的地址。
- (4)廣播位址(broadcast address):作用在於帶有此種位址的資料是要諮詢所有網路上的主機，因此每台主機看到都會接收。
- (5)閘道位址(gateway address):此位址為連到外面 Internet 主機的 IP 位址，此主機負責本地網路銜接外界路的工作。所以此主機會有二個以上的 IP 位址和乙太路卡。
- (6)名稱伺服器位址(nameserver address):此為提供主機名稱和主機位址查詢服務的主機。

2. 防火牆

- (1) IP 過濾(IP filter):保留需要的網路功能外，其餘的網路功能都取消。IP 過濾在數據包這一層工作。它依據起點、終點、埠號和所含的數據包種類信息來控制數據包的流動。這種防火牆非常安全，但是缺少有用的登錄記錄。它阻擋別人進入個別網路，但也不告訴你何人進入你的公共系統，或何人從內部進入網際網路。IP 過濾是

絕對性的過濾系統。即使你要讓外界的一些人進入你的私有伺服器，你也無法讓每一個人進入伺服器。

(2) 代理伺服器(Proxy server):防火牆內的主機需要網路服務時透過此一伺服器達到目的。代理伺服器允許通過防火牆間接進入網際網路。最好的例子是以 telnet 連上系統，然後從該處再以 telnet 連上另一個系統。在有代理伺服器的系統中，這項工作就完全自動。利用客戶端軟體連接代理伺服器後，代理伺服器啟動它的服務程式（代理），然後傳回數據。由於代理伺服器重複所有通訊，因此能夠記錄所有進行的工作。只要配置正確，代理伺服器就絕對安全，這是它最可取之處。它阻擋任何人進入，因為沒有直接的 IP 通路。

七、Linux 作業系統

Linux 的發展從 1991 年 7 月 3 日，芬蘭人 Linus 開始發展一些硬碟和周邊的驅動程式起，至今各項應用軟體已然齊備。

Linux 的檔案系統為以 / 代表之根目錄，底下有呈樹狀結構的目錄：/bin、/sbin、/etc、/dev、/lib、/usr、/var、/tmp... 等重要的檔案目錄。

使用 Linux 的原因除了可以降低購買軟體的成本，更重要的是，它是依照最新的 UNIX 標準 (POSIX)，結合了 SYSTEM V 和 BSD 的優點，並有完充足的軟體支援，是個很完善的作業 (環境) 系統。

以下為指令介紹：

1. 常用指令

man keyword 顯示 keyword 中的資料

what is keyword 顯示 keyword 中的資料

whereis 指令 顯示指令所在路徑

2. 檔案及目錄操作指令

cat 瀏覽檔案內容

cp 檔案拷貝

mv 檔案更名或是搬移

rm 刪除檔案或是目錄 (參數為 -r)

ls 瀏覽目錄內容

mkdir 產生目錄

rmdir 刪除目錄

3. 資料與檔案處理指令

cut 取出檔案內容

```
cut -d: -f 1-3 < /etc/passwd
```

以 ":" 為分隔識別字，將 passwd 的第一至第三欄取出

grep 搜尋檔案內容

tr 字元轉換或刪除

tr -s '\n' 即刪除重複之空白行

sed 批次編輯器，可用以執行簡單的命令

sed 's/linux/LINUX/g' 可將 linux 轉變為大寫

expr 數值計算指令

expr 5 * 6 即 5 乘 6，為 30

find 搜尋檔案位置

find . -name "passwd" -print 為自本目錄之下搜尋 passwd 檔

案

4. 檔案權限與檔案系統相關指令

chmod 改變檔案的屬性(包含存取和執行)

chown 改變檔案的擁有者和所屬群組

df 顯示系統硬碟使用資訊

du 顯示所在目錄使用資訊

fsck 檢查硬碟是否有錯誤發生並修復

quota 設定使用者可以使用多大的硬碟容量

ln 設定資料的鏈結

sync 將存於記憶體中的資料寫入硬碟

patch 將檔案更新的資訊檔(補丁)來更新檔案

5. 訊息顯示相關指令

w 顯示系統目前有那些使用者以及其資訊

ps 顯示系統目前執行程序的資訊

finger 顯示某個使用者的資訊

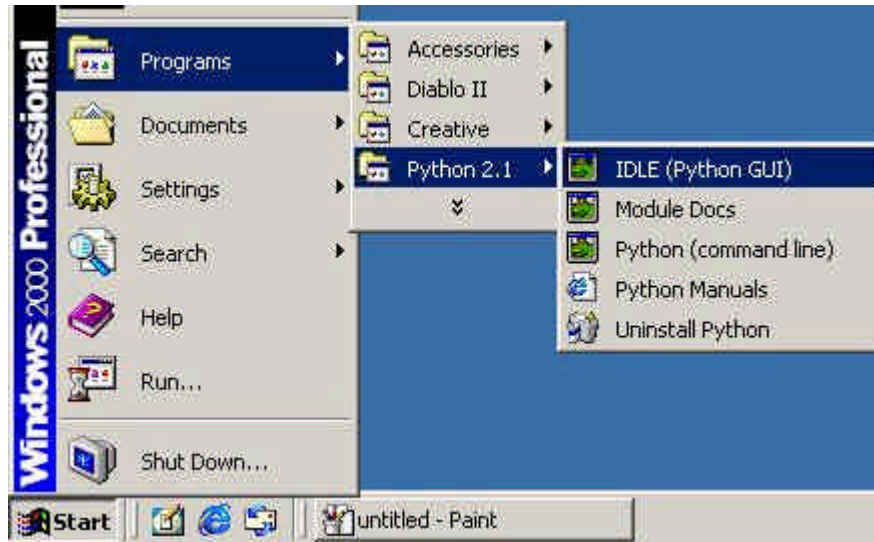
ping 顯示某個主機是否離線的資訊

八、Python 程式語言

一、編輯器的使用(以 Windows 下為例，內容見

http://hkn.eecs.berkeley.edu/~dyoo/python/idle_intro/index.html)

如下圖顯示：



二、檔案內容搜尋

一般在 UNIX 下我們可以用 grep 這個程式來搜尋檔案中的內容，我們可以下達 “grep string filename” 這個指令來搜尋在 filename 檔案中含有 string 字串並顯示在螢幕上。

完整程式和說明如下：

```
#!/usr/bin/env python 此行告訴系統如何執行python這個程式
```

```
import string 含入string模組
```

```
import sys 含入sys模組
```

```
def usage(prog): 顯示此程式的用法  
    print "usage: %s string file_name" % prog  
    return
```

```
if __name__ == "__main__":
```

```
    argc = len(sys.argv) 程式執行時得到的參數個數(含本身)
```

```
    if (argc < 3): 如果參數小於3個，則顯示程式的用法並結束程式
```

```

usage(sys.argv[0])
sys.exit(2)

search_string = sys.argv[1]    將第二個參數設定給搜尋字串
fname = sys.argv[2]           將第三個參數設定給檔案名稱

# open the file file_name
fp = open(fname, "r")          檔案以唯讀方式開啟
lines = fp.readlines()        讀取一行資料
fp.close()

# search for search_string
for l in lines:
    if string.find(l, search_string) != -1:  如果找到對應的字串則把該行顯示
        print l[:-1]  顯示字串 l

```

三、使用 python 做系統檔案管理

```

from stat import *  將stat模組中的所有物件叫出(檔案結構資訊)
import os, sys     含入os、sys模組(系統呼叫)
import getopt      含入getopt模組(外部參數處理)
import string      含入string模組(字串處理)
import time        含入time模組(時間處理)

dir_name="/tmp"
age=1440 單位為分鐘

if not os.path.isdir(dir_name):  如果dir_name目錄不存在
    print "not a directory:", dir_name
    return

files = os.listdir(dir_name)  將目錄所在的檔案名稱複製予files
for file in files:            處理任何一個在files的檔案
    try:
        path = os.path.join(dir_name, file)  將目錄和檔名合併成path
        status = os.stat(path)  將檔案資訊傳給status物件
        tim = time.time()  擷取目前的時間(注意此時間為自1970.1.1起之秒數)
        if (tim - status[ST_MTIME] > age*60):  如果現在時間比檔案時間大age分鐘
            print "rm -rf %s" % path  印出欲刪除之檔案(含路徑)
            os.system("rm -rf %s" % path)  透過系統呼叫執行刪除指令(rm -rf path)
    except:
        continue

```

四、其他應用

Python 的應用非常廣泛，除了系統程式、網路應用、圖形處理、資料庫處理，它幾乎都可以勝任。而且它是不斷發展的程式語言，所以它的能力是不可限量的。

參、心得與建議

這次赴美受訓為職等第一次出國，由於抵達時間恰為美國之感恩節，為當地重大節日，當日除放假外亦為美國人與家人團聚的重要節日，因此無人可至機場接送。藉由出發前由網際網路所收集之資料及美方機場導引設施之完備，再加上勇敢發問與美國人之熱心，職等還是順利於晚間十一點左右抵達下榻處。在此強烈建議同仁無論對方有無人員可至機場接送，第一次到陌生的地方事前的準備工作一定要做好。

在美國受訓期間，特別要感謝幾位人士：Celia Chen 女士伉儷、徐曉明博士伉儷、徐天佑博士與 Cindy Nelson 女士。Celia 與其先生陳先生來自台灣，在美期間安排我們的課程與行程，課餘亦相當關心我們的日常生活對待我們就像一家人，就像是我們在美的家長一樣。徐曉明博士除講授 MM5 的課程之外亦常邀請我們至家裡聚餐，其講授過程相當自由，會依我們的需求講授及上線操作，由於此次時間略嫌短暫無法作到太詳細的地步，徐曉明博士一直要求我們即使回到台灣有何問題亦可透過 EMAIL 與其聯絡，不論公事或是私事，就像我們的指導教授。徐天佑博士雖無擔任課程講授，但在生活上的幫忙相當多，就像學校裡學長學弟的關係。Cindy 是我們在美國的房東，藉由 NCAR 的 HOUSING OFFICE 之介紹，我們在行前就一直以 EMAIL 跟她確認我們的行程，雖然我們抵達時她並不在家，但還是可以順利由另外一位印度室友帶領安頓，平常家居就像大姊一樣的對待我們。還有其他 NCAR 人士也都給予我們相當大的關心與幫忙。在此對他們致上最高的感謝。

受訓期間我們發現 NCAR 有幾個值得我們學習與反思的地方，首先是其分工合作相互支援成果共享的觀念，有時課堂上我們的問題當事人無法解決或不是

他的專業，對方會去幫我們問更專業的人士，甚至直接安排他/她來給我們解答，講授過程中也常常講到「這部分是誰的成果」，在私底下的談話過程中，亦經常聽到跟哪幾個團隊或部門合作解決哪些問題。其次是其工作環境相當完善，技術人員專業知識與能力值得肯定，NCAR 之所以如此執全球氣象之牛耳，除了科學家所發展之理論與實驗之外，擁有能力甚強的軟硬體工程人員也是不可或缺之重要因素。最後是輕鬆生活認真工作的態度，受訓期間幾次例假日跟 NCAR 人員出遊或參加其家庭聚會，體驗到美國人享受生活的態度，所謂享受生活並不是指物資方面的追求，而是其日常休閒與人或大自然的互動，同事間經常一起規劃共度假日或舉行家庭聚會，無形中建立起良好的同事與朋友關係，這對工作上的合作有一定的助益。

底下提出四點建議事項：

- 一、加強分工與支援：AOAWS 系統之龐大與複雜，不是單純氣象人員擁有豐富氣象知識或理論可以掌控，其程式語言部分包含甚多種類，如 FORTRAN 77、FORTRAN 90、SHELL SCRIPT、PYTHON、C++、GNU MAKE、HTML、CGI、JAVA SCRIPT、資料庫結構...等等，操作系統亦包含 PC、LINUX、UNIX 與 VPP 5000 超級電腦等，硬體部分亦包含甚多不同種類之電腦與設備。建議加強各參與單位之溝通與支援，維持良好之互動關係，才能共享飛安之成果。
- 二、持續訓練人員對 AOAWS 的瞭解與操作：如前所述，AOAWS 系統不是僅派遣人員至美國短期受訓即可瞭解整個細節，如果有此預期加諸在受訓人員身上，那受訓人員將承受甚大之壓力導致參訓意願不高。建議在內部亦應定期舉行理論與技術訓練或討論，且不應只侷限於氣象人員。至於電腦程式或硬體設備之部分因進步甚快，學校所學常跟不上資訊方面的腳步，除人員自行進修外可請訓練所考慮開訓相關課程。
- 三、提升人員自行研發之能力與意願：總台雖為作業單位，維持正常而穩定的作業能力為最基本之要求，因此現行設備亦大多以此考慮。但長久考量

應提升所屬人員擁有自行研發之能力及意願，大型系統如 AOAWS 的研發現階段有其困難性，但只要能改善工作環境提昇工作效率的小程式或小型系統都值得鼓勵與支持。目前大局與總台亦鼓勵同仁提報自行研究計畫，但在所有設備都以作業需求為優先的情況下，有意願的同仁不敢也不被允許利用這些設備來嘗試，因為研發過程中一定會不斷嘗試錯誤，因此只能自行購置如筆記型電腦等設備利用公餘時間自行研究，無形中降低不少意願。

四、提倡學習 Linux 作業系統：Linux 作業系統已經是研究單位廣泛使用的作業系統之一，尤其是優異的多工多執行緒的作業環境，使得同一部個人電腦既是主機，也是終端機。但由於對於中文的支援一直不是很理想，使得其在台灣的市場上居於劣勢，但由兩岸三地的努力之下，已經有了初步的成果(CLE Project)，相信在不久的將來，這一套作業系統將會變得普遍，並且變得更容易用。因此，如果各公務單位(尤其是屬於技術性的部門)將來採用 Linux 作為主要的作業平台，不僅可以大幅降低採購軟體的成本，更可以促進同仁研究發展的能力。

建議能在長官的支持下，將有意願之同仁以不影響作業前提下組成一小型研發團隊，以少量經費購置研發用之電腦設備，將每人不同專長集合起來發揮最大功能，也可避免因研發人員離開現職而使得系統維護困難甚至停擺。另一好處是可分散負擔，不用擔心將來維護的責任是否會只落在一個人身上。

最後感謝國家提供此次受訓機會，受訓期間不僅只接收課程所講授之知識與技術，亦讓我們可以觀摩學習別人做事的方法與態度，進而思考為何人家作得到的東西我們會作不到？MM5 的理論或是 AOAWS 系統中所使用的語言程式，把它分開來看，國內並不缺這種人才，但為何無法形成一個有組織的系統是值得讓我們思考的問題。

肆、參考資料

一. LlwasPlot.cks 參數檔的內容:

```
/*
*****
* TDRP params for LlwasPlot
*****/

//=====
//
// LlwasPlot reads in data from an Llwas output file, parses it, merges
// it with a PostScript prologue file and optionally prints it.
//
//=====

//=====
//
// DEBUGGING AND PROCESS CONTROL.
//
//=====

////////// debug //////////
//
// Debug option.除錯選擇
// If set, debug messages will be printed with the appropriate level of
// detail.
//
// Type: enum
// Options:
//  DEBUG_OFF
//  DEBUG_WARNINGS
//  DEBUG_NORM
//  DEBUG_VERBOSE
//

debug = DEBUG_OFF;

////////// instance //////////
//
// Process instance.
// Used for registration with procmmap.
```

```

// Type: string
//

instance = "cks";

//=====
//
// DATA INPUT.
//
//=====

////////// llwas_data_file_path //////////
//輸入檔案之名稱 (含路徑)
// File path for LLWAS output data.
// Type: string
//

llwas_data_file_path = "./tcks_0620.txt";

////////// postscript_prologue_file_path //////////
//ps參數檔名 (含路徑)
// File path for PostScript prologue path.
// Type: string
//

postscript_prologue_file_path = "./prologue_cks.ps";

////////// airport_name //////////
//機場名稱
// Name of airport.
// Type: string
//

airport_name = "RCTP";

////////// llwas_stations //////////
//低空風切變數陣列，每一對為兩個字串，前一個為ps參數檔所用變數，後一個為資料檔案所用
的字串
// Array of LLWAS stations.
// The postscript_id is the variable name given to the station in the

```

```

// postscript prologue. The name is the station name in the LLWAS data
// file.
//
// Type: struct
// typedef struct {
//     string postscript_id;
//     string name;
// }
//
// 1D array - variable length.
//

llwas_stations = {
    { "station_CF", "WAWSCF"},
    { "station_01", "WAWS01"},
    { "station_02", "WAWS02"},
    { "station_03", "WAWS03"},
    { "station_04", "WAWS04"},
    { "station_05", "WAWS05"},
    { "station_06", "WAWS06"},
    { "station_07", "WAWS07"},
    { "station_08", "WAWS08"},
    { "station_09", "WAWS09"},
    { "station_10", "WAWS10"},
    { "station_11", "WAWS11"},
    { "station_12", "WAWS12"},
    { "station_13", "WAWS13"},
    { "station_14", "WAWS14"}
};

////////// runway_names //////////
//跑道名稱
// Array of runway names.
// Type: string
// 1D array - variable length.
//

runway_names = {
    "RWY1A", "RWY1D", "RWY2A", "RWY2D",
    "RWY3A", "RWY3D", "RWY4A", "RWY4D"
}

```

```

};

//=====
//
// DATA OUTPUT.
//
//=====

////////// output_dir //////////
//輸出檔案目錄
// Directory for output files.
// Output files will be named using the airport name and the start and
//   end times.
// Type: string
//

output_dir = "/tmp/llwas/cks";

////////// time_delta //////////
//時間間隔，單位為秒
// Frequency at which the plots are created - secs.
// This should be a multiple of 10.
// Type: int
//

time_delta = 10;

////////// alarm_history_period //////////
//警告顯示前之時間，單位為秒
// History period for alarms - secs.
// The program searches for all alarms between the start and end times,
//   and back to this period before the start time.
// Type: int
//

alarm_history_period = 1200;

////////// print_output_file //////////
//列印選項
// Option to send the output file to a printer.

```

```
// Type: boolean
//

print_output_file = FALSE;

////////// printer_name //////////
//印表機名稱，可見於/etc/printcap或設定PRINTER變數
// Name of printer.
// See 'print_output_file'.
// Type: string
//

printer_name = "${PRINTER}";
```

二. prologue_cks.ps的內容:

```
%!PS Adobe
%Chiang Kai Shek Airport, Taiwan
%-----
/inch { 72 mul } def
  /str2  2 string def
  /str3  3 string def
  /str4  4 string def
  /str5  5 string def
  /str7  7 string def
  /str8  8 string def
  /str12 12 string def
  /str27 27 string def

/F { findfont exch scalefont setfont } bind def
%-----text-handling procedure
/textdict 25 dict def
  textdict begin
    /delta 2   def
    /graytext 0 def
    /grayback 1 def
    /lino false def
    lino { 4 setflat } if
  /tcfp {true charpath flattenpath pathbbox} def
  /lcpbbox {          { strX 0 3 -1 roll put
    newpath 0 0 moveto
    strX tcfp
      dup y2 gt { /y2 exch def } { pop } ifelse
    xoff add dup x2 gt { /x2 exch def } { pop } ifelse
      dup y1 lt { /y1 exch def } { pop } ifelse
    xoff add dup x1 lt { /x1 exch def } { pop } ifelse
    strX stringwidth pop xoff add /xoff exch def
  } forall } def
  end
/text1B { textdict begin
  /back false def
  /strX (x) def /koff 0 def
  /x1 0 def /x2 0 def
  /y1 0 def /y2 0 def
```

```

/cy exch def /cx exch def
/str exch def /ang exch def
/yy exch def /xx exch def
cx 0 lt { /back true def /cx cx neg def } if
newpath 0 0 moveto
str lcpbbox
newpath
cx 0 eq { /xxx 0 def } if
cx 1 eq { /xxx x1 neg def } if
cx 2 eq { /xxx x1 x2 add neg 2 div def } if
cx 3 eq { /xxx x2 neg def } if
cx 4 eq { /xxx xoff neg def } if
cy 0 eq { /yyy y1 neg def } if
cy 1 eq { /yyy 0 def } if
cy 2 eq { /yyy y2 2 div neg def } if
cy 3 eq { /yyy y2 neg def } if
xx yy translate ang rotate
xxx yyy moveto str show back { xxx yyy moveto } if
ang neg rotate xx neg yy neg translate end } def
%-----end-text-handling procedure
%-----
7 /Helvetica F
% 90 50 moveto (file: fig2d.eps ) show
%*****
%-----station and runway data
/runway_W-NE { 0.536 1.480 } def
/runway_W-SW { -1.977 -0.670 } def
/runway_E-NE { 1.446 0.220 } def
/runway_E-SW { -1.203 -1.982 } def
/station_CF { 0.000 0.000 } def
/station_01 { -4.460 -1.754 } def
/station_02 { -2.529 -0.719 } def
/station_03 { -1.176 0.462 } def
/station_04 { 0.791 2.153 } def
/station_05 { 1.731 4.114 } def
/station_06 { -3.171 -2.747 } def
/station_07 { -1.938 -1.210 } def
/station_08 { 1.455 1.202 } def
/station_09 { 3.497 2.369 } def
/station_10 { -2.941 -4.491 } def

```



```

/station_11 { -1.155 -3.334 } def
/station_12 { 0.170 -2.163 } def
/station_13 { 2.131 -0.345 } def
/station_14 { 4.437 0.624 } def
%-----definitions
/black { 0 setgray } def
/red { 1 0 0 setrgbcolor } def
/green { 0 1 0 setrgbcolor } def
/blue { 0 0 1 setrgbcolor } def
/light_blue { 0 1 1 setrgbcolor } def

/nautical_mile { 1.853 mul } def
/runway_length_E { 3.4447 } def
/runway_length_W { 3.3072 } def
/runway_angle_E { 2.202 2.649 atan neg 90 add } def
/runway_angle_W { 2.150 2.513 atan neg 90 add } def
/runway_angle runway_angle_E runway_angle_W add 2 div def
%-----
% /airport_scale { 30 mul exch 30 mul exch } def % STARTING REFERENCE
/pixels_per_kilometer 40 def % *****
/vector_scale pixels_per_kilometer 10 div def
/main_scale_factor pixels_per_kilometer 30 div def
/airport_scale { 30 main_scale_factor mul mul exch
                30 main_scale_factor mul mul exch } def
%-----
% /station_symbol { gsave translate newpath
% 0 0 1.5 0 360 arc stroke grestore } def
/station_symbol { gsave translate newpath
0 0 1.5 0 360 arc gsave fill grestore stroke grestore } def
/zero_velocity_symbol { gsave translate newpath
0 0 2.0 0 360 arc 1 setgray fill 0 setgray
0 0 0.5 0 360 arc fill
0.5 setlinewidth newpath
0 0 2.0 0 360 arc stroke grestore } def
/zero_velocity { gsave newpath
0 0 2.0 0 360 arc 1 setgray fill 0 setgray
0 0 0.5 0 360 arc fill
0.5 setlinewidth newpath
0 0 2.0 0 360 arc stroke grestore } def
/inoperative_symbol { gsave translate red newpath

```

```

0 0 1.5 0 360 arc gsave fill grestore stroke
% 0 -2 3 mul 0 (NA) 2 2 text1B
grestore } def

/plot_station_symbols { black
station_CF airport_scale station_symbol
station_01 airport_scale station_symbol
station_02 airport_scale station_symbol
station_03 airport_scale station_symbol
station_04 airport_scale station_symbol
station_05 airport_scale station_symbol
station_06 airport_scale station_symbol
station_07 airport_scale station_symbol
station_08 airport_scale station_symbol
station_09 airport_scale station_symbol
station_10 airport_scale station_symbol
station_11 airport_scale station_symbol
station_12 airport_scale station_symbol
station_13 airport_scale station_symbol
station_14 airport_scale station_symbol
} def

/wind_vector_scale_factor 1.0 def
/wvsf { wind_vector_scale_factor } def
/wind_vector { % useage: "station_xx" dir kts wind_vector
/kts exch def /dir exch def
/station_y exch def /station_x exch def
/dir dir 180 add def
dir 360 gt { /dir dir 360 sub def } if
gsave station_x station_y translate
kts 3.88 gt
gsave % draw & label wnd vector components
dir runway_angle gt dir runway_angle 180.0 add le and
{ /delta 1.8 def } { /delta 1.8 neg def } ifelse
90 runway_angle sub rotate
/component_wind {runway_angle dir sub cos kts mul } def
0 0 moveto
runway_angle dir sub cos kts vector_scale mul wvsf mul mul 0 lineto
component_wind vector_scale mul wvsf mul 0 lineto red stroke

```

```

component_wind 2 ge
    { component_wind vector_scale mul wvsf mul 0 moveto
    -2.5 -1.5 rmoveto 2.5 1.5 rlineto -2.5 1.5 rlineto
    0.5 setlinewidth red stroke } if
component_wind -2 le
    { component_wind vector_scale mul wvsf mul 0 moveto
    2.5 -1.5 rmoveto -2.5 1.5 rlineto 2.5 1.5 rlineto
    0.5 setlinewidth red stroke } if
component_wind vector_scale mul wvsf mul 0 moveto
component_wind vector_scale mul wvsf mul
runway_angle dir sub sin kts vector_scale mul wvsf mul mul lineto
0.5 setlinewidth red [1 2] 0 setdash stroke [ ] 0 setdash
    component_wind abs 2.0 ge
    { 6 /Helvetica F
    component_wind vector_scale mul wvsf mul 0.67 mul delta 3 mul
    runway_angle 90 sub
    component_wind 10 mul round cvi 10 div str5 cvs
    2 2 text1B } if
    grestore } if

90.0 dir sub rotate % draw & label main wind vector (telltai)
    kts 0.5 gt {
0 0 moveto
    kts vector_scale mul wvsf mul 0 lineto
1 setlinewidth stroke
0.5 setlinewidth 0 0 station_symbol
7 /Helvetica F
    2 3 mul neg 0 90.0 dir sub neg
    kts round cvi str2 cvs 2 2 text1B }
    { zero_velocity } ifelse
    grestore } def

%-----alarm history panel definitions
/history_ref { -6.00 -9.25 airport_scale } def
% /history_ref { -7.75 -5.25 airport_scale } def
% /h_bar_height 25 def
/h_bar_height 20 def
/h_bar_width 2 def
/h_title h_bar_height 5 add 3 mul 2.5 add def
/h_center 120 h_bar_width mul 2 add 2 div def

```

```

/h_right 120 h_bar_width mul 2 add def
/h_box { gsave translate
        gsave 0.85 setgray 0.5 setlinewidth
        30.5 h_bar_width mul 1 add 0 moveto
        0 h_bar_height 5 add 3 mul rlineto
        60.5 h_bar_width mul 1 add 0 moveto
        0 h_bar_height 5 add 3 mul rlineto
        90.5 h_bar_width mul 1 add 0 moveto
        0 h_bar_height 5 add 3 mul rlineto    stroke
        grestore
        0 0 moveto 0.5 setlinewidth
        120 h_bar_width mul 2 add 0 lineto
        120 h_bar_width mul 2 add h_bar_height 5 add 3 mul lineto
        0 h_bar_height 5 add 3 mul lineto
        closepath stroke
        0 h_bar_height 5 add moveto
        120 h_bar_width mul 2 add
        h_bar_height 5 add lineto
        0 h_bar_height 5 add 2 mul moveto
        120 h_bar_width mul 2 add
        h_bar_height 5 add 2 mul lineto stroke
% set font and size
8 /Helvetica F
    h_center h_title 0 (20 Minute Alarm History) 2 1 text1B
7 /Helvetica F
    h_right 0 3 sub 0 (0) 2 3 text1B
    0 0 3 sub 0 (20) 2 3 text1B
    90.5 h_bar_width mul 1 add 0 3 sub 0 (5) 2 3 text1B
    60.5 h_bar_width mul 1 add 0 3 sub 0 (10) 2 3 text1B
    30.5 h_bar_width mul 1 add 0 3 sub 0 (15) 2 3 text1B
    h_right 35 sub h_bar_height 5 add 2.9 mul 0 (MBA) 1 3 text1B
8 /Helvetica F
    currentpoint 0.1 add exch 0.3 add exch 0 (\261) 1 1 text1B
7 /Helvetica F
    h_right 35 sub h_bar_height 5 add 1.9 mul 0 (WSA) 1 3 text1B
8 /Helvetica F
    currentpoint 0.1 add exch 0.3 add exch 0 (\261) 1 1 text1B
7 /Helvetica F
    h_right 35 sub h_bar_height 5 add 0.9 mul 0 (WSA) 1 3 text1B
9 /Helvetica F

```

```

currentpoint 0.2 add exch 0.5 add exch 0 (+) 1 1 text1B
    grestore    } def

/aad_ref { 0.75 -9.25 airport_scale } def

/time_convert { 10 div 1 add exch
    6 mul add exch
    360 mul add } def

%-----replicate AAD Screen
/replicate_aad { gsave
    8 /Helvetica F
/aad_line_spacing h_bar_height 5 add 3 mul 6.5 div def
    0 aad_line_spacing 2.5 mul neg moveto 177 0 rlineto
    0 aad_line_spacing 9 mul rlineto
-177 0 rlineto closepath stroke
    85 aad_line_spacing 6.5 mul 2.5 add 0 (Reconstructed AAD) 2 1 text1B
    grestore } def

/MBA- { time_convert /alarm_time exch def
    pop pop pop
    gsave history_ref translate
    current_time alarm_time sub /delta_time exch def
    delta_time 120 lt delta_time -0.1 gt and
    { 119 h_bar_width mul 1 add delta_time h_bar_width mul sub
        h_bar_height 5 add 2.0 mul 0.25 add moveto
    2 0 rlineto 0 15 rlineto -2 0 rlineto closepath
        red fill } if grestore
    } def

/WSA- { time_convert /alarm_time exch def
    pop pop pop
    gsave history_ref translate
    current_time alarm_time sub /delta_time exch def
    delta_time 120 lt delta_time -0.1 gt and
    { 119 h_bar_width mul 1 add delta_time h_bar_width mul sub
        h_bar_height 5 add 1.0 mul 0.25 add moveto
    2 0 rlineto 0 15 rlineto -2 0 rlineto closepath
        red fill } if grestore
    } def

```

```

/WSA+ { time_convert /alarm_time exch def
    pop pop pop
    gsave history_ref translate
    current_time alarm_time sub /delta_time exch def
    delta_time 120 lt delta_time -0.1 gt and
    { 119 h_bar_width mul 1 add delta_time h_bar_width mul sub
    0 0.25 add moveto
    2 0 rlineto 0 15 rlineto -2 0 rlineto closepath
    red fill } if grestore
    } def

```

%-----end alarm history panel definitions

%-----draw/fill runway and approach areas

7 /Helvetica F

```

/fill_runway_E { gsave runway_E-NE airport_scale translate
    90 runway_angle_E sub rotate
    0 0.10          airport_scale moveto
    runway_length_E neg 0.10 airport_scale lineto
    runway_length_E neg -0.10 airport_scale lineto
    0 -0.10        airport_scale lineto
    closepath 1 setgray fill

    runway_length_E neg -0.25 airport_scale moveto
    runway_length_E neg 2 nautical_mile sub -0.25
    airport_scale lineto
    runway_length_E neg 2 nautical_mile sub 0.25
    airport_scale lineto
    runway_length_E neg          0.25
    airport_scale lineto
    closepath 1 setgray fill

    0 -0.25        airport_scale moveto
    2 nautical_mile -0.25 airport_scale lineto
    2 nautical_mile 0.25 airport_scale lineto
    0 0.25         airport_scale lineto
    closepath 1 setgray fill
    grestore    } def

```

```

/fill_runway_W { gsave runway_W-NE airport_scale translate

```

```

90 runway_angle_W sub rotate
0 0.10          airport_scale moveto
runway_length_W neg 0.10 airport_scale lineto
runway_length_W neg -0.10 airport_scale lineto
0 -0.10        airport_scale lineto
closepath 1 setgray fill

runway_length_W neg -0.25 airport_scale moveto
runway_length_W neg 2 nautical_mile sub -0.25
  airport_scale lineto
runway_length_W neg 2 nautical_mile sub 0.25
  airport_scale lineto
runway_length_W neg          0.25
  airport_scale lineto
closepath 1 setgray fill

0 -0.25        airport_scale moveto
2 nautical_mile -0.25 airport_scale lineto
2 nautical_mile 0.25 airport_scale lineto
0 0.25         airport_scale lineto
closepath 1 setgray fill
grestore      } def

```

```

/draw_runway_E { gsave runway_E-NE airport_scale translate
  90 runway_angle_E sub rotate
  0 0.10          airport_scale moveto
  runway_length_E neg 0.10 airport_scale lineto
  runway_length_E neg -0.10 airport_scale lineto
  0 -0.10        airport_scale lineto
  closepath gsave clip 1.0 setlinewidth stroke
  grestore stroke
  runway_length_E neg -0.25 airport_scale moveto
  runway_length_E neg 2 nautical_mile sub -0.25
    airport_scale lineto
  runway_length_E neg 2 nautical_mile sub 0.25
    airport_scale lineto
  runway_length_E neg          0.25
    airport_scale lineto
  closepath stroke
  runway_length_E neg 1 nautical_mile sub 0.25

```

```

    airport_scale moveto
runway_length_E neg 1 nautical_mile sub -0.25
    airport_scale lineto
stroke
0 -0.25          airport_scale moveto
2 nautical_mile -0.25 airport_scale lineto
2 nautical_mile 0.25 airport_scale lineto
0 0.25          airport_scale lineto
closepath stroke
1 nautical_mile 0.25 airport_scale moveto
1 nautical_mile -0.25 airport_scale lineto
stroke
grestore    } def

/draw_runway_W{ gsave runway_W-NE airport_scale translate
    90 runway_angle_W sub rotate
    0 0.10          airport_scale moveto
runway_length_W neg 0.10 airport_scale lineto
runway_length_W neg -0.10 airport_scale lineto
0 -0.10          airport_scale lineto
closepath gsave clip 1.0 setlinewidth stroke
grestore stroke
runway_length_W neg -0.25 airport_scale moveto
runway_length_W neg 2 nautical_mile sub -0.25
    airport_scale lineto
runway_length_W neg 2 nautical_mile sub 0.25
    airport_scale lineto
runway_length_W neg          0.25
    airport_scale lineto
closepath stroke
runway_length_W neg 1 nautical_mile sub 0.25
    airport_scale moveto
runway_length_W neg 1 nautical_mile sub -0.25
    airport_scale lineto
stroke
0 -0.25          airport_scale moveto
2 nautical_mile -0.25 airport_scale lineto
2 nautical_mile 0.25 airport_scale lineto
0 0.25          airport_scale lineto
closepath stroke

```



```

1 nautical_mile 0.25 airport_scale moveto
1 nautical_mile -0.25 airport_scale lineto
stroke
grestore } def

/RWY_E { gsave runway_E-NE airport_scale translate
90 runway_angle_E sub rotate
0 0.10 airport_scale moveto
runway_length_E neg 0.10 airport_scale lineto
runway_length_E neg -0.10 airport_scale lineto
0 -0.10 airport_scale lineto
closepath 0.75 setgray fill
grestore } def

/RWY_W { gsave runway_W-NE airport_scale translate
90 runway_angle_W sub rotate
0 0.10 airport_scale moveto
runway_length_W neg 0.10 airport_scale lineto
runway_length_W neg -0.10 airport_scale lineto
0 -0.10 airport_scale lineto
closepath 0.75 setgray fill
grestore } def

/E_SW-1 { gsave runway_E-NE airport_scale translate
90 runway_angle_E sub rotate
runway_length_E neg -0.25 airport_scale moveto
runway_length_E neg 1 nautical_mile sub -0.25
airport_scale lineto
runway_length_E neg 1 nautical_mile sub 0.25
airport_scale lineto
runway_length_E neg 0.25
airport_scale lineto
closepath 0.75 setgray fill
grestore } def

/W_SW-1 { gsave runway_W-NE airport_scale translate
90 runway_angle_W sub rotate
runway_length_W neg -0.25 airport_scale moveto
runway_length_W neg 1 nautical_mile sub -0.25
airport_scale lineto
runway_length_W neg 1 nautical_mile sub 0.25
airport_scale lineto
runway_length_W neg 0.25

```

```

        airport_scale lineto
    closepath 0.75 setgray fill
    grestore } def
/E_SW-2 { gsave runway_E-NE airport_scale translate
    90 runway_angle_E sub rotate
    runway_length_E neg 1 nautical_mile sub -0.25
        airport_scale moveto
    runway_length_E neg 2 nautical_mile sub -0.25
        airport_scale lineto
    runway_length_E neg 2 nautical_mile sub 0.25
        airport_scale lineto
    runway_length_E neg 1 nautical_mile sub 0.25
        airport_scale lineto
    closepath 0.75 setgray fill
    grestore } def
/W_SW-2 { gsave runway_W-NE airport_scale translate
    90 runway_angle_W sub rotate
    runway_length_W neg 1 nautical_mile sub -0.25
        airport_scale moveto
    runway_length_W neg 2 nautical_mile sub -0.25
        airport_scale lineto
    runway_length_W neg 2 nautical_mile sub 0.25
        airport_scale lineto
    runway_length_W neg 1 nautical_mile sub 0.25
        airport_scale lineto
    closepath 0.75 setgray fill
    grestore } def
/E_NE-1 { gsave runway_E-NE airport_scale translate
    90 runway_angle_E sub rotate
    0 -0.25          airport_scale moveto
    1 nautical_mile -0.25 airport_scale lineto
    1 nautical_mile 0.25 airport_scale lineto
    0 0.25          airport_scale lineto
    closepath 0.75 setgray fill
    grestore } def
/W_NE-1 { gsave runway_W-NE airport_scale translate
    90 runway_angle_W sub rotate
    0 -0.25          airport_scale moveto
    1 nautical_mile -0.25 airport_scale lineto
    1 nautical_mile 0.25 airport_scale lineto

```

```

0 0.25          airport_scale lineto
closepath 0.75 setgray fill
grestore      } def
/E_NE-2 { gsave runway_E-NE airport_scale translate
90 runway_angle_E sub rotate
1 nautical_mile -0.25  airport_scale moveto
2 nautical_mile -0.25  airport_scale lineto
2 nautical_mile 0.25  airport_scale lineto
1 nautical_mile 0.25  airport_scale lineto
closepath 0.75 setgray fill
grestore      } def
/W_NE-2 { gsave runway_W-NE airport_scale translate
90 runway_angle_W sub rotate
1 nautical_mile -0.25  airport_scale moveto
2 nautical_mile -0.25  airport_scale lineto
2 nautical_mile 0.25  airport_scale lineto
1 nautical_mile 0.25  airport_scale lineto
closepath 0.75 setgray fill
grestore      } def
/05-ARWY { RWY_W } def  /05-DRWY { RWY_W } def
/05-A1MF { W_SW-1 } def /05-D1MD { W_NE-1 } def
/05-A2MF { W_SW-2 } def /05-D2MD { W_NE-2 } def
/23-ARWY { RWY_W } def  /23-DRWY { RWY_W } def
/23-A1MF { W_NE-1 } def /23-D1MD { W_SW-1 } def
/23-A2MF { W_NE-2 } def /23-D2MD { W_SW-2 } def

/06-ARWY { RWY_E } def  /06-DRWY { RWY_E } def
/06-A1MF { E_SW-1 } def /06-D1MD { E_NE-1 } def
/06-A2MF { E_SW-2 } def /06-D2MD { E_NE-2 } def
/24-ARWY { RWY_E } def  /24-DRWY { RWY_E } def
/24-A1MF { E_NE-1 } def /24-D1MD { E_SW-1 } def
/24-A2MF { E_NE-2 } def /24-D2MD { E_SW-2 } def

%-----draw background grid
/kilometer_grid { 0.5 setlinewidth
0.85 setgray % light_blue
6 -6.75 airport_scale moveto
6 5.50 airport_scale lineto stroke
-6.00 -6 airport_scale moveto
6.50 -6 airport_scale lineto stroke

```

-6 -6.75 airport_scale moveto
-6 5.50 airport_scale lineto stroke

-6.00 5 airport_scale moveto
6.50 5 airport_scale lineto stroke

5 -6.75 airport_scale moveto
5 5.50 airport_scale lineto stroke

-6.00 -5 airport_scale moveto
6.50 -5 airport_scale lineto stroke

-5 -6.75 airport_scale moveto
-5 5.50 airport_scale lineto stroke

-6.00 4 airport_scale moveto
6.50 4 airport_scale lineto stroke

4 -6.75 airport_scale moveto
4 -3.90 airport_scale lineto
4 -3.10 airport_scale moveto
4 5.5 airport_scale lineto stroke

-6.00 -4 airport_scale moveto
6.50 -4 airport_scale lineto stroke

-4 -6.75 airport_scale moveto
-4 5.50 airport_scale lineto stroke

-6.00 3 airport_scale moveto
6.50 3 airport_scale lineto stroke

3 -6.75 airport_scale moveto
3 -3.90 airport_scale lineto
3 -3.10 airport_scale moveto
3 5.50 airport_scale lineto stroke

-6.00 -3 airport_scale moveto
6.50 -3 airport_scale lineto stroke

-3 -6.75 airport_scale moveto
-3 5.50 airport_scale lineto stroke

-6.00 2 airport_scale moveto
6.50 2 airport_scale lineto stroke

2 -6.75 airport_scale moveto
2 5.50 airport_scale lineto stroke

-6.00 -2 airport_scale moveto
6.50 -2 airport_scale lineto stroke

-2 -6.75 airport_scale moveto
-2 5.50 airport_scale lineto stroke

-6.00 1 airport_scale moveto
6.50 1 airport_scale lineto stroke
1 -6.75 airport_scale moveto
1 5.50 airport_scale lineto stroke
-6.00 -1 airport_scale moveto
6.50 -1 airport_scale lineto stroke
-1 -6.75 airport_scale moveto
-1 5.50 airport_scale lineto stroke

-6.00 0 airport_scale moveto
6.50 0 airport_scale lineto stroke
0 -6.75 airport_scale moveto
0 5.50 airport_scale lineto stroke

black

-6.00 0 airport_scale moveto
0 -0.04 airport_scale rmoveto
0.16 0.04 airport_scale rlineto
-0.16 0.04 airport_scale rlineto
closepath fill

6.50 0 airport_scale moveto
0 -0.04 airport_scale rmoveto
-0.16 0.04 airport_scale rlineto
0.16 0.04 airport_scale rlineto
closepath fill

0 5.50 airport_scale moveto
-0.04 0 airport_scale rmoveto
0.04 -0.16 airport_scale rlineto
0.04 0.16 airport_scale rlineto
closepath fill

0 -6.75 airport_scale moveto
-0.04 0 airport_scale rmoveto

```

    0.04 0.16 airport_scale rlineto
0.04 -0.16 airport_scale rlineto
    closepath fill

-6.00 -6.75 airport_scale moveto
    6.50 -6.75 airport_scale lineto
    6.50 5.50 airport_scale lineto
-6.00 5.50 airport_scale lineto
    closepath stroke
    } def

/time_and_date_display_procedures {
/time_string ( : : ) def
    hour 10 lt { time_string 0 (0) putinterval
                time_string 1 hour str2 cvs putinterval}
{ time_string 0 hour str2 cvs putinterval } ifelse
    min 10 lt { time_string 3 (0) putinterval
                time_string 4 min str2 cvs putinterval}
{ time_string 3 min str2 cvs putinterval } ifelse
    min 10 lt { time_string 3 (0) putinterval
                time_string 4 min str2 cvs putinterval}
{ time_string 3 min str2 cvs putinterval } ifelse
    sec 10 lt { time_string 6 (0) putinterval
                time_string 7 sec str2 cvs putinterval}
{ time_string 6 sec str2 cvs putinterval } ifelse

10 /Courier-Bold F
2.4 -3.45 airport_scale 0 time_string 1 1 text1B
    ( UTC) show
2.4 -3.75 airport_scale moveto
    day str2 cvs show
    month 1 eq { ( January ) show} if
    month 2 eq { ( February ) show} if
    month 3 eq { ( March ) show} if
    month 4 eq { ( April ) show} if
    month 5 eq { ( May ) show} if
    month 6 eq { ( June ) show} if
    month 7 eq { ( July ) show} if
    month 8 eq { ( August ) show} if
    month 9 eq { ( September ) show} if

```

```

month 10 eq { ( October ) show} if
month 11 eq { ( November ) show} if
month 12 eq { ( December ) show} if
year str4 cvs show } def

```

```

%-----define Almos Station Names and Proceures

```

```

/WAWSCF { station_CF airport_scale 4 -2 roll wind_vector } def
/WAWS01 { station_01 airport_scale 4 -2 roll wind_vector } def
/WAWS02 { station_02 airport_scale 4 -2 roll wind_vector } def
/WAWS03 { station_03 airport_scale 4 -2 roll wind_vector } def
/WAWS04 { station_04 airport_scale 4 -2 roll wind_vector } def
/WAWS05 { station_05 airport_scale 4 -2 roll wind_vector } def
/WAWS06 { station_06 airport_scale 4 -2 roll wind_vector } def
/WAWS07 { station_07 airport_scale 4 -2 roll wind_vector } def
/WAWS08 { station_08 airport_scale 4 -2 roll wind_vector } def
/WAWS09 { station_09 airport_scale 4 -2 roll wind_vector } def
/WAWS10 { station_10 airport_scale 4 -2 roll wind_vector } def
/WAWS11 { station_11 airport_scale 4 -2 roll wind_vector } def
/WAWS12 { station_12 airport_scale 4 -2 roll wind_vector } def
/WAWS13 { station_13 airport_scale 4 -2 roll wind_vector } def
/WAWS14 { station_14 airport_scale 4 -2 roll wind_vector } def

```

```

%-----Procedure for generating AAD Display's

```

```

/RWY1A { /RWY1A_input  exch def } def
/RWY1D { /RWY1D_input  exch def } def
/RWY2A { /RWY2A_input  exch def } def
/RWY2D { /RWY2D_input  exch def } def
/RWY3A { /RWY3A_input  exch def } def
/RWY3D { /RWY3D_input  exch def } def
/RWY4A { /RWY4A_input  exch def } def
/RWY4D { /RWY4D_input  exch def } def
/GustString { /GustString_input  exch def } def

```

```

/aad_display_procedure {
/RWY1A_plot ( ) def
/RWY1D_plot ( ) def
/RWY2A_plot ( ) def
/RWY2D_plot ( ) def

```

```

/RWY3A_plot ( ) def
/RWY3D_plot ( ) def
/RWY4A_plot ( ) def
/RWY4D_plot ( ) def
/GustString_plot ( ) def

```

```

GustString_plot 0 GustString_input putinterval
  GustString_plot 20 time_string putinterval
RWY1A_plot 0 RWY1A_input
  putinterval
  RWY1A_plot (K) search { pop pop pop
    str7 0 RWY1A_plot 0 4 getinterval putinterval
    str7 4 RWY1A_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
  { pop } ifelse
RWY1D_plot 0 RWY1D_input
  putinterval
  RWY1D_plot (K) search { pop pop pop
    str7 0 RWY1D_plot 0 4 getinterval putinterval
    str7 4 RWY1D_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
  { pop } ifelse
RWY2A_plot 0 RWY2A_input
  putinterval
  RWY2A_plot (K) search { pop pop pop
    str7 0 RWY2A_plot 0 4 getinterval putinterval
    str7 4 RWY2A_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
  { pop } ifelse
RWY2D_plot 0 RWY2D_input
  putinterval
  RWY2D_plot (K) search { pop pop pop
    str7 0 RWY2D_plot 0 4 getinterval putinterval
    str7 4 RWY2D_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
  { pop } ifelse
RWY3A_plot 0 RWY3A_input
  putinterval
  RWY3A_plot (K) search { pop pop pop
    str7 0 RWY3A_plot 0 4 getinterval putinterval

```



```

    str7 4 RWY3A_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
    { pop } ifelse
RWY3D_plot    0 RWY3D_input
    putinterval
RWY3D_plot (K) search { pop pop pop
    str7 0 RWY3D_plot 0 4 getinterval putinterval
    str7 4 RWY3D_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
    { pop } ifelse
RWY4A_plot    0 RWY4A_input
    putinterval
RWY4A_plot (K) search { pop pop pop
    str7 0 RWY4A_plot 0 4 getinterval putinterval
    str7 4 RWY4A_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
    { pop } ifelse
RWY4D_plot    0 RWY4D_input
    putinterval
RWY4D_plot (K) search { pop pop pop
    str7 0 RWY4D_plot 0 4 getinterval putinterval
    str7 4 RWY4D_plot 16 3 getinterval putinterval
    str7 cvx cvn exec}
    { pop } ifelse
} def

%-----draw runways and generate aad displays
/draw_runways_and_generate_aad_displays {
gsave
    0.25 setlinewidth black
    draw_runway_E
    draw_runway_W
grestore
%-----replicate AAD Screen
gsave
    8 /Helvetica F
    aad_ref translate
    replicate_aad
    10 /Courier-Bold F
    5 aad_line_spacing 8.25 2.5 sub mul 0 RWY1A_input 1 1 text1B

```

```

5 aad_line_spacing 7.25 2.5 sub mul 0 RWY1D_input 1 1 text1B
5 aad_line_spacing 6.25 2.5 sub mul 0 RWY2A_input 1 1 text1B
5 aad_line_spacing 5.25 2.5 sub mul 0 RWY2D_input 1 1 text1B
5 aad_line_spacing 4.25 2.5 sub mul 0 RWY3A_input 1 1 text1B
5 aad_line_spacing 3.25 2.5 sub mul 0 RWY3D_input 1 1 text1B
5 aad_line_spacing 2.25 2.5 sub mul 0 RWY4A_input 1 1 text1B
5 aad_line_spacing 1.25 2.5 sub mul 0 RWY4D_input 1 1 text1B
5 aad_line_spacing 0.25 2.5 sub mul 0 GustString_plot 1 1 text1B

```

```
grestore
```

```
7 /Helvetica F
```

```
-6 -9.25 airport_scale 2.5 aad_line_spacing mul sub moveto
```

```
(Copyright 2001, NCAR) show
```

```
} def
```

```
%*****
```

```
%-----NORTH Symbol
```

```
/xscale 1.0 def
```

```
/north_symbol { 10 xscale div /Helvetica-Bold F
```

```
2 xscale div setlinewidth
```

```
0 0 0 (NORTH) 2 2 text1B
```

```
0 -40 xscale div moveto 0 -8 xscale div lineto
```

```
0 8 xscale div moveto 0 40 xscale div lineto
```

```
3 xscale div 32 xscale div lineto
```

```
1 xscale div 34 xscale div lineto
```

```
stroke } def
```

```
%-----initialize graphical displays, setup plots
```

```
/initialize_graphical_displays {
```

```
%-----locate grid/background and add labels
```

```
302 470 translate
```

```
kilometer_grid
```

```
10 /Helvetica F
```

```
-6.00 5.5 airport_scale 3.5 add exch 1 add exch 0
```

```
(Chiang Kai Shek Airport, Taiwan) 1 1 text1B
```

```
% 6.50 5.5 airport_scale 3.5 add exch 1 sub exch 0
```

```
% (Civil Aeronautical Administration) 4 1 text1B
```

```
-6.00 5.5 airport_scale 15 add exch 1 add exch 0
```

```
(LLWAS Event Reconstruction) 1 1 text1B
```

```
% 6.50 5.5 airport_scale 15 add exch 1 sub exch 0
```

```

% (Republic of China) 4 1 text1B
8 /Helvetica F
-5.80 -6.60 airport_scale 0
(background grid = 1 km) 1 1 text1B

% /wvsf 1.5 def

wvsf 1.05 gt wvsf 0.95 lt or
{ 6.24 -6.30 airport_scale 0
(X) -3 1 text1B
currentpoint exch 3 sub exch 0
wvsf 10 mul round cvi 10 div str3 cvs -3 1 text1B
currentpoint exch 3 sub exch 0
(Additional display factor = ) 3 1 text1B } if

6.24 -6.60 airport_scale 0
(Standard wind vector scale, 1 km = 10 m/s) 3 1 text1B

%-----plot North Symbol
gsave
-4.50 3.51 airport_scale translate 0.75 0.75 scale
0 setlinejoin 40 setmiterlimit
black north_symbol
grestore

%-----fill airport runway(s) and approach areas
gsave
fill_runway_E
fill_runway_W
grestore

%-----draw station symbols
black
plot_station_symbols

%-----draw background for history box
history_ref h_box
} def

```