

出國報告 (出國類別：實習)

第 2 代先進交談式天氣處理系統 本土化系統建置與資料導入銜接流 程測試

服務機關：交通部中央氣象局

姓名職稱：黃柏禎 技佐

派赴國家：美 國

出國期間：102 年 4 月 12 日至 10 月 17 日

報告日期：102 年 11 月 26 日

摘要

第 2 代先進交談式天氣處理系統(AWIPS II)用來整合顯示各種即時氣象資訊，以提供氣象預報及研究人員決策時參考。

AWIPS II 基於服務導向架構，最主要的兩個子系統為 AWIPS 共通顯示環境(CAVE)及環境資料交換服務(EDEX)。CAVE 建立在 Eclipse RCP/JFace/SWT 等軟體元件的基礎上，為 AWIPS II 主要的使用者介面，而 EDEX 服務架構在 Camel/Spring 軟體應用整合框架的基礎上，主要負責即時資料的解碼作業及回應 CAVE 等 Client 端程式的請求。

本次出國實習主要成果為：(1) 學習 AWIPS II 系統架構及相關技術，掌握未來發展、建置、佈署及維運的基礎知識，如：EDEX 服務安裝佈署、解碼流程、本土化、運作機制以及 CAVE 安裝佈署、本土化設定、運作機制等；(2) 針對本局需求進行相關的本土化學習，如：特定產品的解碼、顯示等本土化設定，並嘗試針對本局衛星資料撰寫對應的 EDEX 解碼外掛(Plug-in)，期望能有助於後續的發展應用。

關鍵詞：AWIPS II、CAVE、EDEX

目次

1. 目的	4
2. 過程	5
3. 成果與心得.....	7
3.1 AWIPS II 相關技術學習	7
3.2 AWIPS II EDEX 學習.....	8
3.3 AWIPS II CAVE 學習.....	12
3.4 CWB 本土化成果.....	14
4. 建議及結論.....	20
4.1 系統發展建議	20
4.2 新產品導入與本土化設定建議	22
4.3 建置、佈署與維運建議.....	22
4.4 結論.....	23
5. 名詞解釋.....	24
6. 參考資料.....	26

1. 目的

第 2 代先進交談式天氣處理系統(Advanced Weather Interactive Processing System, AWIPS II)為美國國家氣象局(NWS)自 2006 年開始委託雷神(Raytheon)公司開發的新一代 AWIPS 系統，此系統係基於服務導向架構(Service Oriented Architecture)，並採用許多開放技術、框架及軟體元件，於 2012 年已陸續在美國少部分天氣預報辦公室(WFO)上線使用及評估測試。

本局長期以來與美國海洋暨大氣總署(NOAA)地球系統研究實驗室(ESRL)全球系統組(GSD)一直保持密切的合作關係，從第 1 代的即時預報系統(AWIPS I)到圖形預報編輯系統(GFE)，均已成為本局預報作業不可或缺的工具，本著持續追求進步與發展的精神，我們期望藉由 AWIPS II 的導入，學習到最先進的資訊技術，提升即時預報系統效能、可靠性及可用度，架構上也能更開放、有彈性且容易維護，另一方面也能更深化臺美之間的技術交流與合作。

職本次出國主要學習項目為 AWIPSII 系統架構熟悉、開發環境建置、本土化、資料導入及系統流程控制/維運等，期望在這個過程中掌握導入 AWIPSII 所需的關鍵知識與技術，以利後續的發展工作。

2. 過程

職此次赴美行程及工作內容說明如下表：

日期	地點與相關工作內容
102/4/12~ 102/4/13	臺北->洛杉磯->馬里蘭銀泉
102/4/15	至美國國家氣象局(NWS)氣象發展實驗室(MDL)決策支援組(Decision Assistance Branch)報到，忻翎艷博士協助填寫一些表格、環境熟悉、工作上軟硬體、帳號設置等。
102/4/16~ 102/5/23	<ol style="list-style-type: none"> 忻翎艷博士與工程師 Ken Sperow 介紹虛擬實驗室(Virtual LAB)、協同作業網站(Collaborate Web)等學習資源，並協助設置 AWIPS II 發展環境及討論未來學習方向。 AWIPS II 基礎學習: 參與美國國家氣象局(NWS)線上學習網站課程，研讀 AWIPS II 相關技術文件。 學習如何將測試資料餵進 AWIPSII EDEX： <ol style="list-style-type: none"> 透過 LDM/pqact/EDEX Bridge 透過 RCM 雷達服務(Radar Server) 自行將資料複製到手動解碼目錄(manual endpoint) 手動發送 QPID 訊息以觸發 EDEX 解碼程序
102/5/24	參與旁聽 FFMP/AWIPS II 期中討論會議。
102/5/25~ 102/10/15	<ol style="list-style-type: none"> 研讀 AWIPS II 相關技術文件、網站資源。 學習 AWIPS II 本土化，嘗試針對 CWB 站台進行相關設定: <ol style="list-style-type: none"> 地圖資料匯入及顯示設定 地形資料客製化 底圖顯示設定 格點產品解碼及顯示設定 衛星產品顯示設定 測站產品顯示設定 RCM 雷達服務(Radar Server)設定 參與移動氣象特徵追蹤工具(Moving Meteogram Trace Tool)發展會議，研讀相關程式碼，協助將衛星、雷達、格點等產品套用到此工具，也藉此學習到一些 AWIPS II 工具及 CAVE 背後的運作機制。 EDEX 運作機制學習： <ol style="list-style-type: none"> 如何進行 EDEX 程式除錯 EDEX 啟動流程

	<p>(3) EDEX 相關設定(edex.xml、mode.xml 等)</p> <p>(4) Spring 框架、Camel 解碼流程</p> <p>(5) 過期資料清除機制</p> <p>5. 嘗試撰寫 EDEX 外掛元件(cwbsat)以解碼本局衛星資料 本局衛星中心提供的衛星資料並未使用美方的 GINI 格式資料，無法直接以 EDEX 內建的外掛元件(Plug-in)解碼。</p>
102/10/16~ 102/10/17	馬里蘭銀泉->洛杉磯->臺北

註：以上僅為職在學習過程中接觸到的項目列表，並不代表已完成或能完全掌握的工作項目。

3. 成果與心得

3.1 AWIPS II 相關技術學習

AWIPS II 用到了許多開放技術、框架及軟體元件，詳細的清單如下網頁所示：
http://www.unidata.ucar.edu/software/awips2/manual/software_freeware.html，雖然這個清單有點長，不過也不是每項都必須深入學習，有些只要知道個大概即可，有些可以暫時當成黑箱看待。

但再考量到本局有本土化、客製化、佈署及維運等需求，牽涉到的層面其實還滿廣的，以軟體發展、維運的角度而言，平常有餘力的時候還是要多方涉獵，並在重要的技術部分多花點時間深入了解，真的遇到問題時比較不會無從下手。雖然研讀文件看似很花時間，但常常是解決問題最有效率的方式。

筆者深深覺得能有機會在此學習真的還滿幸運的，因為除了技術文件外，還有許多的網路資源可以運用，包含線上課程、論壇等，當然最重要的還是氣象發展實驗室(MDL)人員的支援，每每在遇到問題時，忻博士總能指出一些方向、要研讀的文件，甚至找雷神公司(Raytheon)的工程師協助，讓從來沒有參與過大型軟體專案的筆者獲益良多。雖說 AWIPS II 相關的技術文件、學習資源還算滿充足的，不過還是沒辦法涵蓋到所有細節，即使不是進行軟體開發的工作，我們常常還是需要由原始碼下手來了解系統的運作，好在 AWIPS II 運用了 Eclipse 整合開發環境，讓原始碼的追蹤比起 AWIPS I 容易了許多。熟悉 Eclipse 整合開發環境，尤其是其除錯模式(Debug Perspective)，將能大大提升發展人員的生產力與效率，筆者認為這是除了 Java 語言本身之外，AWIPS II 發展維護人員最重要的先修項目。

雖然 AWIPS II 號稱是基於服務導向架構(SOA, Service Oriented Architecture)下設計的系統，不過在雷神公司(Raytheon)官方的 AWIPS II 軟體設計文件(SSDD, Software System Design Description)中並未出現任何服務導向架構相關的字眼，令人覺得有點訝異，反倒是在系統管理手冊(SMM, System Manager's Manual)中有提到 EDEX 是建立在服務導向架構下的服務集合。在此架構下，除了 EDEX 各項服務本身要盡量滿足高聚合度(High Cohesion)、低耦合度(Loose Coupling)等原則，也運用了 Camel、Spring 等軟體框架來管理、組合解碼流程，並以 QPID 作為訊息中介，讓各項服務能很容易地重組或置換新的實作方式。此外，由於客戶端程式可透過 QPID、http 請求等訊息傳遞的方式，或利用 uEngine 介面和 EDEX 互動，AWIPS II 將不會是封閉的系統，CAVE 也不會是唯一的使用者介面，未來在實際應用上將有很大的彈性及想像空間。

雷神公司官方文件中並未明確指出 EDEX 以外的子系統(如 CAVE)是基於服務導向架構來設計的，不過整體上的設計還是很有服務導向的精神，在其軟體設計文件中就開宗明義提到了 AWIPS II 的設計需求及主要目標為「最大化可適應性及可負擔性(Maximize Adaptability And Affordability)」，期望此系統在可負擔的成本下足以因應未來快速多樣的需求變動，當然這是建立在可維護性、延展性、穩定性、安全性等前提下，而為了達成這樣的需求及目標，AWIPS II 整體上設計是基於外掛元件(Plug-in)的架構，此架構下系統會有一組核心功能(Core Capabilities)，我們可在此基礎上加入新的外掛元件以使用、擴充這些核心功能，舉例如下：

例一：EDEX 中核心功能包含了 Camel 框架、資料庫存取、HDF5 持久化(Persistence)、請求/處理(Request/handler)服務、本土化等，而在這些核心功能上我們可以加入一些相依於特定產品外掛，像是模式格點、雷達、衛星等 COMMON/EDEX 外掛元件(Plug-in)來進行資料的解碼、儲存、取出等作業。

例二：CAVE 中核心功能包含了 Eclipse RCP 框架、JFace、SWT、本土化、Rendering 框架等，而在這些核心功能上我們可以加入一些相依於特定產品的視覺化(VIZ)外掛(如:格點、雷達、衛星產品)，以支援將這些特定產品資料繪製到畫面上。

此外 AWIPS II 在設計上遵循分層方法(Layered Approach)，意指各別程式碼中使用既有功能時，僅需知道介面的定義規範，而無須知道其實作細節，以方便未來改用不同的實作。

例一：圖形繪製的程式片段會”繪製”圖形至 IGraphicsTarget 介面，而非特定的實作類別，如 GLTarget (Open GL)。

例二：連線至 QPID 係透過 JMS 介面，無須相依於 QPID 的實作。

例三：資料庫的存取透過 EDEX 的 CoreData 類別，而不直接使用 SQL 語法，程式碼中也無須直接相依於 postgres 資料庫的 API。

例四：存取 HDF5 時透過 IDataStore 介面，程式碼無須相依於 PypiesDataStore 實作。

3.2 AWIPS II EDEX 學習

在 AWIPS II 中，EDEX 環境資料交換服務(Environmental Data EXchange)主要負責即時資料的解碼作業、過期資料清除以及回應 CAVE 等客戶端程式的請求。

EDEX (edex_camel)啟動時預設會包含 4 個獨立的程序：edex_ingest(處理大多數資料)、edex_ingestGrib(處理 Grib 資料)、edex_ingestDat(處理決策輔助工具相關資料，如 SCAN、SAFESEAS、FFMP 等)以及 edex_request(處理 CAVE 等客戶端的請求)。以下分項說明筆者針對 EDEX 的學習成果/心得。

3.2.1 EDEX 安裝佈署

AWIPS II 使用 YUM/RPM 套件安裝的方式於 Linux 主機佈署 EDEX，取得相關的 RPM 檔後須先設定好對應的 YUM 設定檔(於目錄/etc/yum.repo.d)，再用 yum 指令進行安裝，如：`# sudo yum groupinstall "AWIPS II Standalone"` 可將 EDEX 相關套件安裝至本機的/awips2 目錄下。

若要於 AWIPS 整合開發環境(ADE)修改 EDEX 相關原始碼後重新佈署到主機上，則可使用 build.edex 專案下的 ANT 建置檔 deploy-install.xml 來佈署之。

安裝完成後相關程式會被佈署為 Linux 系統服務，啟動 EDEX 服務程序的方式如下依序執行之。

```
# sudo /sbin/service edex_postgres start
# sudo /sbin/service httpd-pypies start
# sudo /sbin/service qpidd start
# sudo /sbin/service edex_camel start
```

3.2.2 EDEX 資料解碼流程

EDEX 解碼作業流程如圖 1 所示，並簡單說明如下：

- (1) LDM、RCM 雷達服務、LDAD 等程序負責將上游原始資料存到/data_store 中，並透過 QPID 的訊息佇列「external.dropbox」或「radarserver.dropbox」傳送訊息給 EDEX 分派服務(Distribution Service)，訊息須包含表頭(一般為 WMO header)及內容(原始資料檔存放位置)。
- (2) 分派服務根據\${EDEX_HOME}/data/utility/edex_static/[base|site]/distribution 下所有 XML 設定檔的正規表示法樣式與此訊息的表頭比對，若比對符合則會發送 QPID 訊息至對應的 Ingest.*佇列，以通知對應的 EDEX 解碼外掛(Plug-in)處理。
- (3) 對應的 EDEX 解碼外掛(Plug-in)依據訊息中包含的檔案路徑將原始資料做解碼的動作，並交由持久化服務(Persist Service)將處理後的資料存為 HDF5 格式的檔案。處理原始資料的 EDEX 解碼外掛(Plug-in)名稱必為分派服務設定檔的檔名(不包含副檔名.xml)。
- (4) 接著由資料索引服務(Index Service)將處理後得到的相關描述性資料(metadata)存到 PostgreSQL 資料庫「metadata」，此資料庫中原則上不存放氣象觀測資料，但有少數例外，如：飛機觀測報 pirep；另外若為雷神公司(Raytheon)開發的內建 EDEX 解碼外掛(Plug-in)，則存放的資料表名稱原則上和 EDEX 解碼外掛(Plug-in)名稱一樣，為分派服務設定檔的檔名。
- (5) 最後再透過 QPID edex.alerts 佇列傳送「資料已處理完成」的通知給 CAVE 以更新產品資訊。

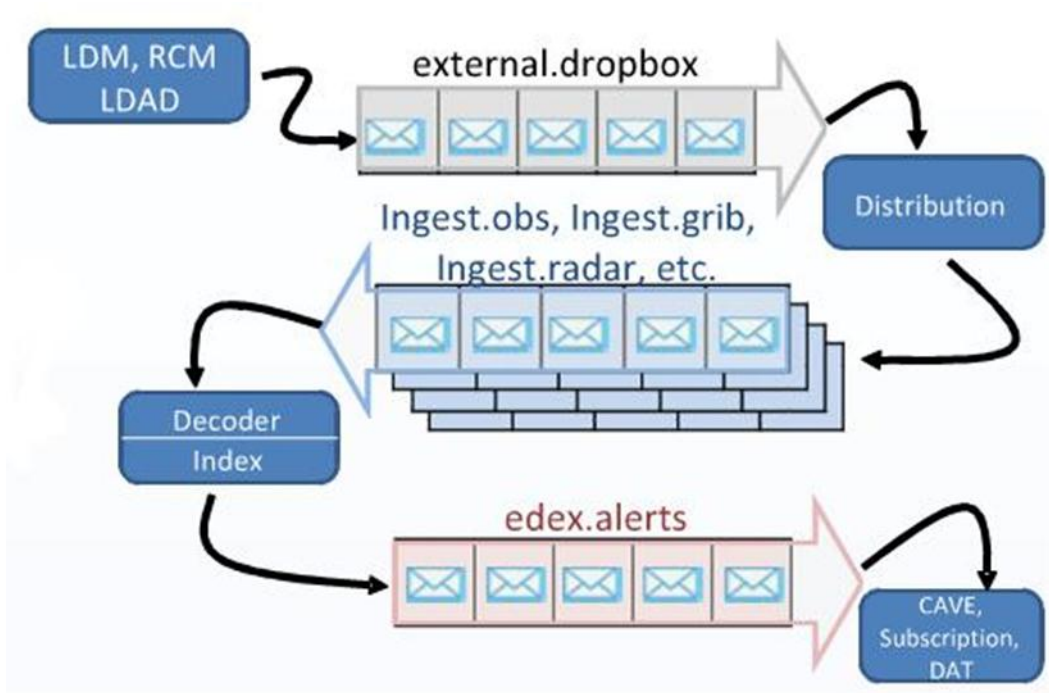


圖 1 AWIPSII 即時資料解碼作業流程圖 (摘自 NWS Learning Center EDEX Overview Course)

以上所述只是大致上的解碼流程，實際上 EDEX 使用 Camel-Spring XML 檔組合解碼流程，對於不同產品的 EDEX 解碼外掛，流程可能會些微差異，我們亦可藉由設定來客製化流程。

3.2.3 Debug EDEX 程式

要使用 Eclipse IDE 的除錯模式(Debug Perspective)來進行 EDEX 程式除錯，我們須先設定連線到 EDEX Debug Port，初次使用之詳細設定步驟如下：

(1) 以-d 選項啟動 edex_camel

```
# ${EDEX_HOME}/bin/start.sh -d ${edexMode}
```

edexMode 為 ingest、ingestGrib、ingestDat 及 request 等。

(2) 於 Eclipse 選單點選 Run -> Debug Configurations...

於對話窗左邊選單選擇 Remote Java Application -> 點選滑鼠右鍵並選 New 以開啟新的設定。

(3) 於 Connect 分頁輸入新設定的 Name、Project、Connection Properties 等參數
其中 Name 為設定檔名稱；Project 為要除錯的專案，不過其實隨便選一個 EDEX 專案即可；其中 Connection Properties 中的 Host 為要除錯的 EDEX 程序所在主機，例如 localhost，而 Port 則要參照 build.edex/esb/etc/\${edexMode}.sh 中的 EDEX_DEBUG_PORT。

(4) 於分頁 Source 點選 Add 以新增要追蹤的 Java 專案(也可以後再編輯加入新的專案)，並點選 Apply 套用。

(5) 未來只要執行 Debug 時選擇此設定，即可啟動 Debug Perspective 開始除錯。

(6) 假設我們想追蹤 EDEX 衛星產品解碼外掛(Plug-in)的 SatelliteDecoder.java，則

可在此程式碼中設定中斷點，接著嘗試丟個衛星資料以觸發解碼過程，進行到中斷點時即會暫停程式的運作。

值得一提的是除錯模式(Debug Perspective)有程式碼即時代換(hot swap)功能，例如我們可以直接修改 `SatelliteDecoder.java`，修改的部分會在觸發解碼時立即反應出來，不用重新佈署 EDEX，不過此代換動作僅是取代記憶體中的程式碼片段，結束除錯之後 EDEX 本身並不會真的被異動。

3.2.4 EDEX 啟動流程

如同前面所提到的，EDEX 基於服務導向架構，建立在 Apache Camel 整合框架下，並使用 Spring 框架來管理 Java 物件，而其對應的啟動服務名稱為 `edex_camel`。此外 EDEX 使用 QPID 作為訊息中介、`postgresql` 作為 `metadata` 資料庫以及 `HDF5` 作為實際資料儲存格式，而 `HDF5` 的存取則是透過 `httpd-pypies` 服務。這也說明了為什麼在 `edex_camel` 服務啟動之前，必須先啟動 `edex_postgres`、`httpd-pypies` 及 `qpidd` 3 個服務。

我們可藉由檢視啟動指令 `/etc/init.d/edex_camel` 得知 EDEX 服務是由 `/awips2/edex/bin/start.sh` 啟動的，其主要的參數為 EDEX 執行模式(`RUN_MODE`)，如：`start.sh ${RUN_MODE}`，作業的 `RUN_MODE` 包含 `ingest`、`ingestGrib`、`ingestDat`、`request` 等，EDEX 服務啟動前會先載入共同設定 `/awips2/edex/bin/setup.env`，接著載入 `/awips2/edex/etc/${RUN_MODE}.sh` 以匯出對應該模式的環境變數(若沒指定則是載入 `default.sh`)。

實際的啟動作業係以 `YAJSW`(Yet Another Java Service Wrapper)包裝執行，Java 執行參數設定於 `/awips2/edex/conf/wrapper.conf`，程式進入點 `main()` 函式位於 EDEX 專案 `com.raytheon.uf.edex.esb.camel.launcher` 中，EDEX 啟動時主要會先載入 `edex.xml` 的 `spring` 設定，接著參照 `/awips2/edex/conf/modes.xml` 以載入對應該執行模式的外掛(plug-in)專案 `Spring` 設定，除了上述 4 種作業模式外，我們還可修改 `modes.xml` 來設定測試用的非作業模式。

3.2.5 過期資料清除機制

EDEX 使用 `com.raytheon.uf.edex.purgesrv.PurgeSrv` 提供清除過期資料的服務。我們僅須發送 QPID 請求訊息至 `purgeRequest` 佇列，`PurgeSrv` 即會依據請求內容以及對應的規則(Purge Rule)幫我們清除過期資料。

`PurgeSrv` 支援的請求訊息內容格式如下：

`PURGE_ALL_DATA`: 清除所有資料。

`PURGE_EXPIRED_DATA`: 清除所有過期資料(根據 `data plugin purge rules`)。

PURGE_PLUGIN=[dataplugin name]: 清除某產品(data plugin)過期資料。

PURGE_ALL_PLUGIN=[dataplugin name]: 清除某產品(data plugin)所有資料。

PURGE_CRON: Purge cron message ?

Purge Rule 設定檔係用來控制某產品要保留多久/多少筆資料，若該產品資料外掛元件(Plug-in)並未設定 Purge Rule 的話，則依預設規則來清除。

實務上除了直接發送 QPID 訊息外，我們亦可透過 uEngine 網頁手動清除某產品過期資料；若 EDEX 服務安裝在本機，則 uEngine 資料清除網頁網址如下：

<http://localhost:8080/uEngineWeb/dataPurge/dataPurge.html>

除此之外，EDEX 服務本身也會定時執行過期資料清除的作業。

3.2.6 uEngine

uEngine CLI (Command Line Interface)為外部應用程式與 EDEX 介接的橋樑，透過 uEngine 執行 uEngine(Python) Script，我們可讓外部應用程式直接利用現有的 EDEX 物件，而不需撰寫 Java Plugin，我們若想撰寫個獨立的程式來測試某個 EDEX 元件，也可利用 uEngine 來執行，以省去自行建立 EDEX 相依物件的麻煩，不過筆者還不是很懂 uEngine 如何運作的，有待未來深入學習。

uEngine 用法：

```
uengine [flag(s)] < ScriptFullPathName
```

Flags:

-h (help) 印出用法說明

-s (substitution) 變數代換

-r (runner) 指定 script runner ['python', 'jscript', 'system', 'ldad']

範例：uengine -r python < /home/awips/myScripts/sample.py

3.3 AWIPS II CAVE 學習

CAVE 全名為 AWIPS 共通顯示環境(Common AWIPS Visualization Environment)，為 AWIPS II 主要的客戶端使用者介面，就如同 AWIPS I 中的 D2D，只不過 CAVE 基於 Eclipse RCP/JFace/SWT，包含數個呈現模式(Perspective)，除了 D2D 外，還有如 GFE、Hydro、Localization 等呈現模式，以提供不同目的使用。以下分安裝佈署等逐項說明。

3.3.1 CAVE 安裝佈署

於 Linux 主機上安裝 CAVE 的方式和 EDEX 一樣，都是用 YUM/RPM，如下：

```
# sudo yum groupinstall "AWIPS II Visualize"
```

這會安裝 Visualize 群組套件，包含 Alertviz 及 CAVE。

AWIPS II 的 Alertviz 對應 AWIPS I 中的 Guardian，用來接收一些警告、錯誤訊息提醒使用者。要啟動 CAVE 前須先啟動 Alertviz，如下：

```
# /awips2/alertviz/alertviz.sh
```

```
# /awips2/cave/cave.sh
```

3.3.2 CAVE 本土化設定介面(Localization Perspective)

由於本土化設定檔目錄分散在各處、設定檔同步更新問題，加上權限控管、簡化操作等考量，CAVE 本土化模式(Localization Perspective)提供了圖形化介面(GUI)來整合 CAVE 相關本土化設定的操作。

我們可以於選單 CAVE->Perspective->Localization 開啟本土化模式(Localization Perspective)，開啟後可於左方的 File Browser 展開各項本土化設定檔，雖說我們還是可以用 vi 等文字編輯器直接修改各項本土化設定檔，不過一般還是建議盡量能在 CAVE 本土化模式(Localization Perspective)上進行各項操作。

除此之外我們還可以利用設定檔 userRoles.xml 來進行權限控管，以細部規範不同使用者於 CAVE 本土化模式(Localization Perspective)下能夠編輯的本土化設定檔種類/層級。

3.3.3 Debug CAVE 程式

若要使用 Eclipse 整合開發環境的除錯模式(Debug Perspective)來進行 CAVE 程式除錯，可於 package explorer 展開專案 com.raytheon.viz.product.awips，滑鼠移到 developer.product 上按右鍵，選取 Debug As -> Eclipse Application 即可。

3.3.4 CAVE 啟動流程

專案 com.raytheon.uf.viz.application 的 VizApplication.java 會實作 Eclipse RCP 架構中 IApplication 的 start()及 stop()，其中 start()會呼叫 component.startComponent()來執行一些必要的初始化作業來啟動 IStandaloneComponent(以 CAVE 來說就是指"CAVE"這個 Java 類別)。

3.3.5 移動氣象特徵追蹤工具學習

移動氣象特徵追蹤工具(Moving Meteogram Trace Tool)用來追蹤預報人員感興趣的移動特徵，繪製出該特徵隨時間的變化曲線圖，如下圖 2 所示。雖然本工具和本局本土化發展沒有直接的關係，但藉由參與本工具的發展討論會議及檢視追蹤此工具的原始碼、程式架構，筆者學習到了許多和 CAVE 及 EDEX 有關的運作細節，例如：CAVE 的資源類別(Resource Class)架構、如何載入顯示設定(Bundle)、

EDEX 的請求/處理(Request/Handler)及 AWIPS II 工具的架構等，也更加熟悉了 Eclipse 整合開發環境的運用。

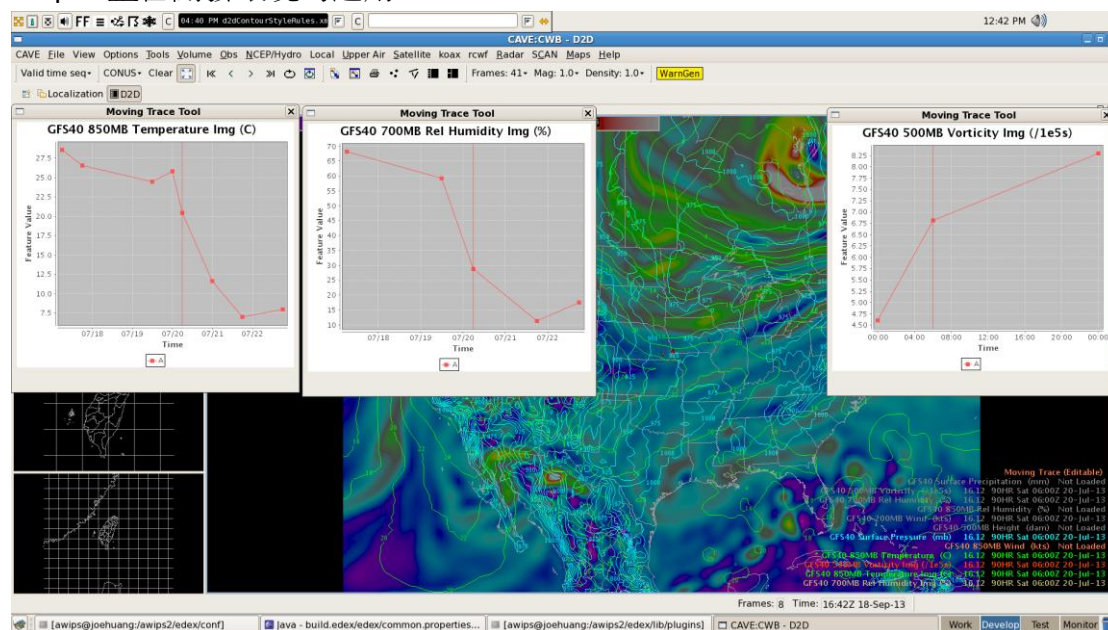


圖 2 移動氣象特徵追蹤工具(Moving Meteorogram Trace Tool)畫面截圖。

3.4 CWB 本土化成果

本局即時預報系統(WINS)目前已導入的產品為數眾多，筆者並未全數於 AWIPS II 完成相對應本土化的工作，僅針對其中較具代表性的少數幾項去試作，因為半年的時間雖不算太短，不過要學的東西實在太多了，筆者還有很多不懂的地方，需要日後不斷學習熟悉，且筆者此行的目的是盡量學會如何進行本土化，而非完成所有本土化工作。以下分項說明筆者針對 CWB 需求所進行的本土化設定成果與心得。

3.4.1 AWIPS II 本土化設定簡介

在 AWIPS II 中，本土化設定(Configuration)整併了 AWIPS I 中本土化(Localization)和客製化(Customization)的架構，延伸了 GFE Base/Site/User 的覆蓋(Override)概念，讓站台維護者/使用者能依權限客製化相關設定，這些獨立的設定也能很容易的備份及分享給其他站台使用者，另外 CAVE 並提供本土化設定圖形化操作介面(Localization Perspective)，將複雜的目錄、權限、操作、同步更新等隱藏起來。

本土化層級(Localization Level)

AWIPS II 本土化設定包含 Base/Configured/Site/Workstation/User 5 個層級，讀取的優先次序為 User > Workstation > Site > Configured > Base，舉例來說，系統針對某個設定若發現存在 User 層級的設定檔，就會載入此設定，若於 User 層級找不到，再去找 Workstation 之後的層級，以此類推。

本土化設定種類(Localization Types)

目前 AWIPS II 本土化設定檔可分為 cave_config、cave_static、common_static、edex_static 4 種，分述如下。

- (1) cave_config: 和 CAVE 偏好設定(Preference)有關的設定檔，此類設定檔只會有 Base/Site/User 三個層級的設定。
- (2) cave_static: 和 CAVE 顯示有關的本土化設定，例如：basemaps(non-shapefile geo-info)、bundles(顯示設定)、colormaps、menu、style rule、測站 plot models 等設定。
- (3) common_static: 和 CAVE 及 EDEX 都有關的共同設定，例如：user roles 權限設定。
- (4) edex_static: 和 EDEX 解碼有關的設定。
此類設定檔僅包含 Base/Configured/Site 等層級的設定，不提供工作站主機及個人等層級的客製化。

另外值得注意的是，在家目錄下有個 caveData 目錄作為 CAVE 相關設定 (cave_config、cave_static)的暫存目錄，若設定檔有異動才會從上述設定檔目錄複製過來，若 CAVE 發生異常，可嘗試刪除此目錄再重啟。

3.4.2 建立新的本土化站台 CWB

以本局為例，建立一個本土化站台 CWB 的步驟如下：

- (1) 建立本土化目錄

```
# mkdir /awips2/edex/data/utility/common_static/site/CWB
# mkdir /awips2/edex/data/utility/edex_static/site/CWB
# mkdir /awips2/edex/data/utility/cave_static/site/CWB
```
- (2) 編輯檔案 /awips2/edex/bin/setup.env，將其中的 AW_SITE_IDENTIFIER 設定值改為 CWB 並重啟 EDEX Server。
- (3) 於 CAVE 中展開 CAVE -> Preference -> Localization 將 Site 的設定值修改為 CWB 並重啟 CAVE。
- (4) 必要時參考內建的本土化站台 OAX，將其設定檔複製到上述新建的 CWB 本土化目錄後再來做客製化修改。

3.4.3 地圖/底圖的匯入與設定

- (1) 學習如何匯入地理資訊檔(shape files)至 AWIPS II PostgreSQL 資料庫，並於 CAVE 進行地圖顯示設定(Map Bundle Config)。
- (2) 學習如何建立特定投影、範圍的底圖，並配合載入上面新增的地圖以進一步客製化底圖設定。

地圖/底圖本土化成果如圖 3 所示。

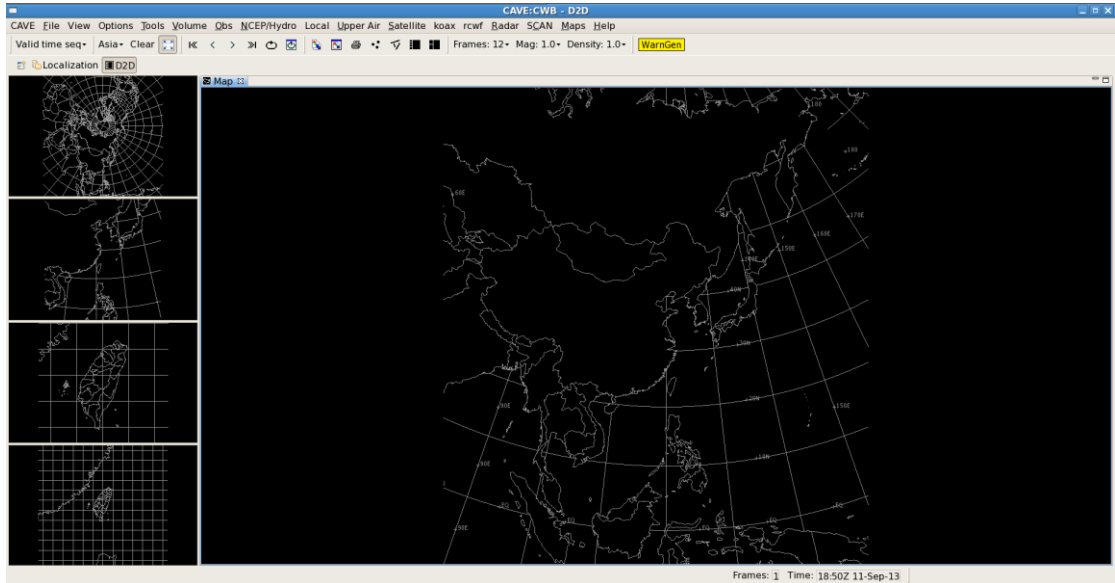


圖 3 CAVE 地圖/底圖本土化畫面截圖。

3.4.4 地形資料檔的產製

由於 AWIPS II 內建的地形資料檔(HDF5 格式)係針對美方的需要製作的，並未涵蓋到亞洲地區，故須自行產製適合本局的地形檔案，成果如圖 4 所示。

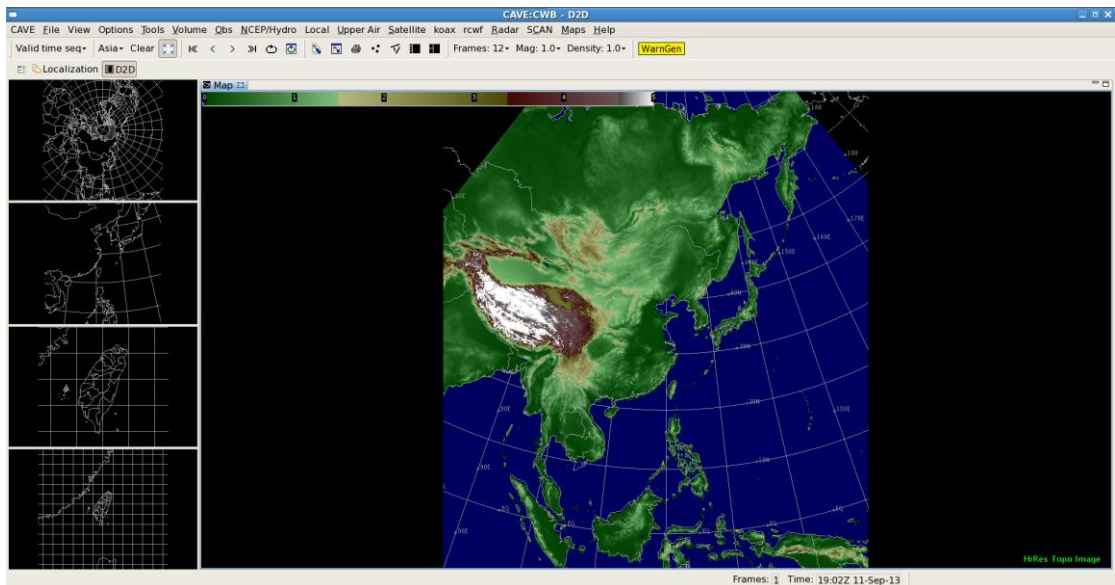


圖 4 CAVE 亞洲區地形資料載入畫面。

3.4.5 格點/模式產品的解碼及顯示設定

由於忻博士給了個無法於 EDEX 成功解碼的 Grib2 格式的檔案供職研究，所以這部份花了比較多的時間，除了 Grib2 檔案格式的研讀，也學習了相關的解碼、顯示、選單、衍生場等設定。圖 5 係針對本局提供的日本 JMA 模式 Grib2 檔進行實際進行本土化之設定成果。

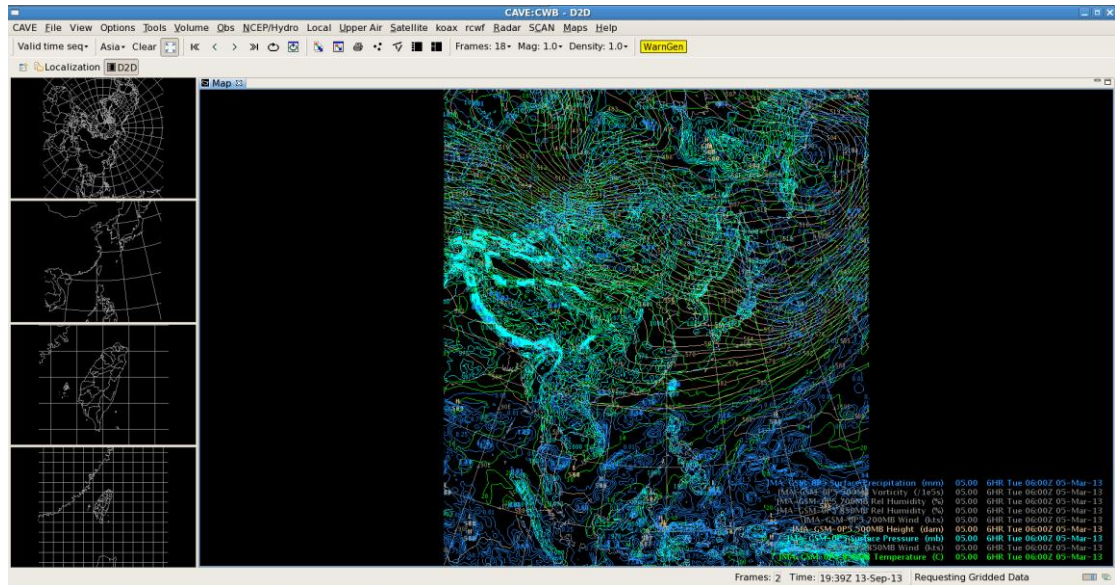


圖 5 日本 JMA 模式格點產品載入畫面。

3.4.6 衛星產品的解碼及顯示設定

衛星產品的顯示設定和格點產品大致上相同，不過在解碼流程方面，AWIPS II 內建的 EDEX 衛星解碼外掛(Plug-in)係針對美方所使用的 GINI 格式檔案撰寫，並無法接受本局衛星中心提供的資料。

雖然內建還有另外一支 EDEX 解碼外掛「regionalsat」，可以接受 AWIPS I NetCDF3 格式資料的，不過其接受的 NetCDF 格式還是和本局 WINS 產出的檔案稍有不同；此問題雖不難解決，不過用這個解碼外掛進行解碼的話，未來我們還要先將資料轉成 NetCDF3 再餵進 EDEX，流程稍嫌繁複沒效率，優點則是能相容於原本 NetCDF3 的資料。

由於本局的衛星資料格式還滿單純的，職便試著參考「regionalsat」這支解碼外掛，針對本局資料撰寫 EDEX 解碼外掛「cwbsat」以解碼之。

圖 6 為亞洲區紅外線衛星雲圖(MTSAT HRIT)載入畫面，雖然表面上看起來顯示效果還不錯，不過仔細與現有即時預報系統(WINS)的圖比對會發現似乎有點不太一致，尚待近一步比對驗證。

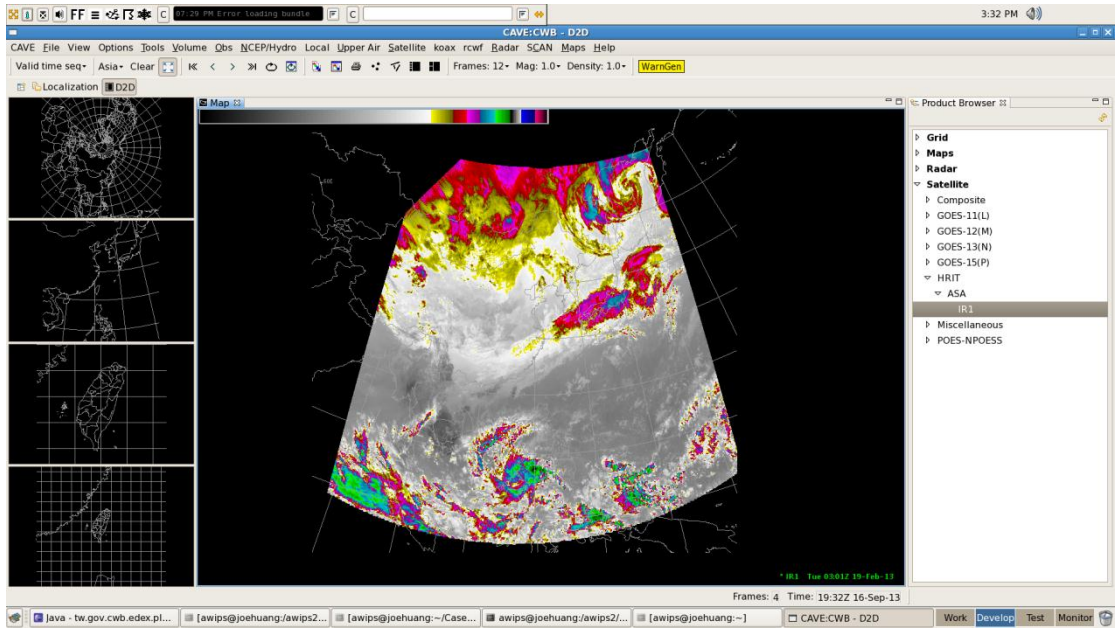


圖 6 亞洲區紅外線衛星雲圖(HRIT.ASA.IR1)顯示畫面。

3.4.7 觀測資料的解碼及顯示設定

觀測資料的本土化部分僅針對從本局提供的 BUFRUA 資料進行解碼測試，並稍微學了一點本土化相關的知識，例如：繪製模型(Plot Model, SVG 格式)、顯示(Display Bundle)及選單等設定，目前尚未實際針對本局即時預報系統(WINS)既有產品進行設定轉移。另外像本局即時預報系統(WINS)目前所接收的 FGGE、CGM、Text 等格式的資料，並無法使用內建的 EDEX 解碼外掛，要如何處理，也待未來繼續努力。BUFRUA 觀測資料載入畫面如圖 7 所示。

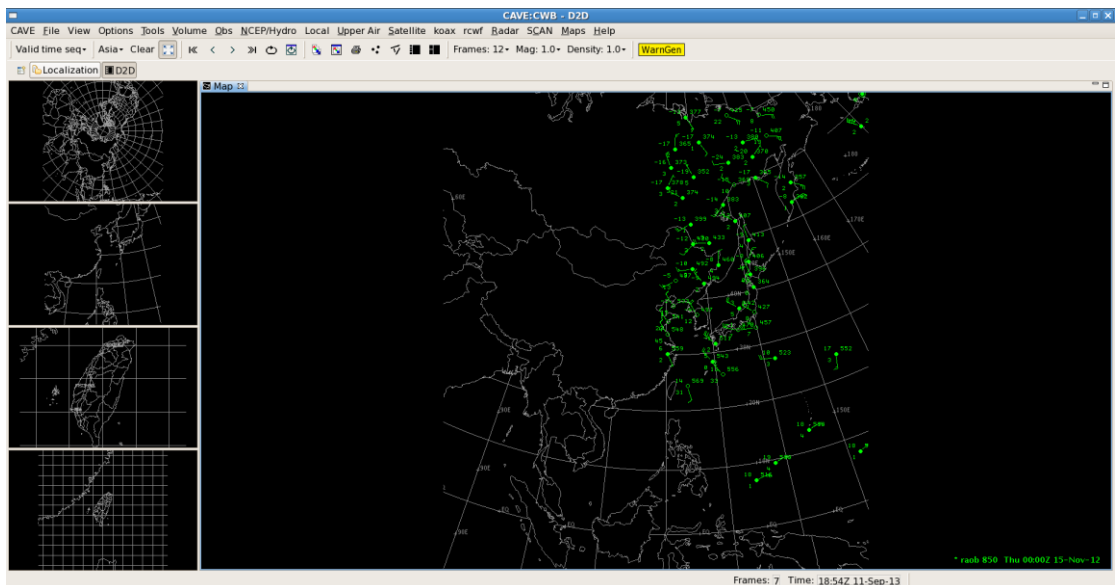


圖 7 BUFRUA 觀測資料顯示畫面。

3.4.8 雷達產品的解碼及顯示設定

雷達的部分此行尚未花費很多時間去學習，僅大概知道一點關於如何增加新雷達站台至 CAVE 選單以及如何設定 RCM 雷達服務，以作為 EDEX 與 ORPG 服務的介接橋樑，尚未將本局提供的原始資料檔透由 EDEX 成功解碼及顯示。

4. 建議及結論

4.1 系統發展建議

4.1.1 AWIPS I VS. AWIPS II

以下針對 AWIPS I 與 AWIPS II 進行比較，以供未來新系統發展時參考。這裡的 AWIPS I 係指本局客製化後的即時預報系統作業版本(WINS, 目前最新版本代號為 cwbalps)，而非美方原始版本。

	AWIPS I (WINS)	AWIPS II
主要語言	C++、Tcl/Tk，以 Makefile 建置專案。	Java、Python，並使用 ADE/Eclipse 為整合開發環境，利用 ANT 建置專案。
軟體架構	物件導向，執行效能較好但設計上較沒有彈性，不易因應未來的需求變化。	物件導向/服務導向；設計上較有彈性，容易整合/重組/置換各種服務及軟體元件以因應未來需求。
即時資料作業流程	LDM/pqact 接收儲存上游原始資料，並觸發解碼程式，將資料存為 NetCDF3 檔案供 D2D 讀取，最後發送產品產出通知至 D2D 以達到即時更新的目的。	LDM/pqact 接收儲存上游原始資料，並以 LDM/EDEX Bridge 發送 QPID 訊息以通知 EDEX 觸發解碼，最終產品的描述性資料 (metadata)存至 PostgreSQL 資料庫，實際資料存為 HDF5，最後發送產品產出通知至 CAVE 以達到即時更新的目的。
主要氣象產品顯示介面	D2D，基於 Tcl/Tk；D2D GUI 介面直接存取 netCDF 檔案。	CAVE D2D 模式(Perspective)，基於 Eclipse RCP/JFace/SWT；CAVE 發送 HTTP 請求至 EDEX Request 服務，EDEX 回傳必要的產品描述性資訊(metadata)，CAVE 再藉以存取 HDF5 檔案。
顯示設定本土化方式	修改本土化設定檔後執行 mainScript.csh 產生最終本土化檔案。	於 CAVE 本土化模式(Localization Perspective)直接進行本土化設定。

產品傳送至下游單位	以 NisSend 將解碼完成的 NetCDF 檔案傳送到 WINS 下游主機；或以遠端桌面整合軟體提供下游單位直接執行 D2D(雲端 WINS)。	無；若要傳送實際資料到下游，可能要於下游主機啟動 EDEX 服務，並傳送尚未解碼的原始資料到下游主機。
過期產品清除機制	Crontab 定時觸發清除程式或以 NICE 元件 NdmFp 以刪除過期檔案。	由管理者定義規則(Purge Rule)，EDEX 負責刪除過期資料；要手動作業的話則可發送 QPID 請求通知 EDEX 資料清除服務(Purge Service)，亦可利用 uEngine 網頁介面執行此作業。
產品備份/保存機制	Crontab 定時觸發備份程式將解碼後的檔案(NetCDF3 格式)備份至磁帶館。	EDEX Ingest 解碼服務處理完後將資料存到\${ARCHIVE_ROOT}，解碼後的資料並未備份保存。
Failover 策略	兩台 WINS 作業主機利用 NICE 元件 NisSend 同步原始資料，並同時解碼所有上游資料。	<ol style="list-style-type: none"> 1. 兩台 EDEX 服務主機(DX3/4)間透過 QPID 達到負載平衡的作用，QPID 佇列的訊息被處理後即被移除，故同一筆資料不會同時被兩台主機處理。 2. 兩台資料服務主機(DX1/2)組成 Heartbeat 叢集，運行 PostgreSQL、RCM 雷達服務、PyPIES、LDM 等服務，以 DAS(Direct Attached Storage)作為解碼後資料的儲存空間。

4.1.2 程式開發人員先備知識

建議要熟悉 AWIPS II 軟體系統架構、Java 物件導向程式設計概念、Eclipse 整合開發環境、服務導向架構概念、OSGi 框架、Spring 框架、Apache Camel、JMS/QPID 等，若為 Viz/CAVE 外掛(Plug-in)開發人員，要再熟悉 Eclipse RCP 框架、JFace、SWT 等；其它還有 Python、ANT、Hibernate、Postgresql、Thrift、JAXB 等等...

4.1.3 程式開發建議事項

(1) 遵循介面程式設計(Interface Programing)的原則以保持未來相關實作技術變更的彈性。

- (2) 遵循套件命名慣例，氣象局自行開發的專案名稱建議以 `tw.gov.cwb` 開頭(例如：`tw.gov.cwb.common.XXX`、`tw.gov.cwb.edex.XXX`、`tw.gov.cwb.viz.XXX` 等)。
- (3) 非必要時不要直接修改美方內建(Baseline)的程式碼及設定檔，可以的話就開新的專案來擴展/覆蓋原系統功能，以簡化未來和美方新版本合併的工作。
- (4) 盡量重複使用現有的程式碼，遵循既有的編碼風格。

4.2 新產品導入與本土化設定建議

- (1) 建議不要修改基礎(Base)目錄下的設定檔，以 `Configuration/Site/Workstation/User` 等層級進行本土化。
- (2) 要再評估是否開放使用者(User)層級的設定或用統一帳號，如要開放，要開放那些部分以及如何進行權限控管？

4.3 建置、佈署與維運建議

4.3.1 系統建置、佈署人員先備知識

ANT、rpmbuild、yum、版本控管(eg. Subversion, git)、heartbeat、YAJSW 等。

4.3.2 佈署策略評估

(1) Heartbeat 技術評估

AWIPS II 實際作業時是可以把不同軟體元件分開裝在多台主機，或用多台主機提供同一個服務來分散負載，並可以 heartbeat 叢集技術來達到高可用度(High Availability)的目的。

(2) 實體主機 vs. 虛擬主機？

或許未來本局作業環境可以將 EDEX 服務裝在多台虛擬主機上，一來可以很有彈性地擴充服務、提升運作效能，二來也可以很容易進行主機的備份、還原、維護及管理作業。

(3) 軟體佈署方式

要評估一下看是要在發展主機建置完成後，以 Subversion(版本控管軟體)管理建置完成的目錄，再以 `svn checkout` 佈署到各目的的主機(亦即目前本局 WINS 的佈署方式)，或是要先建置成 rpm 套件並在各主機進行安裝。

4.3.3 系統維護人員先備知識

熟悉 Linux 系統管理、shell script、AWIPSII 系統架構、log4j、Spring、Camel-Spring、PostgreSQL/pgadmin、JConsole、QPID、LDM、HDF5 格式、PyPIES Data Store 等。

4.4 結論

AWIPS II 用到了很多開放技術、框架及軟體元件，整體架構為了保持彈性以因應未來的需求變動，也比 AWIPS I 龐大了許多，要滿足本局的客製化、建置佈署、維護等需求，需學習的東西比想像中還多，此行職雖然獲益匪淺，不過尚有許多不了解的地方，有待未來繼續努力。在此感謝好心收留我的室友 Sam 及這段期間幫助過我的每個人，也謝謝氣象局的長官們願意給我學習的機會，讓我這一趟美國之旅充滿了美好的回憶。

5. 名詞解釋

ADE (AWIPS 整合開發環境)

全名為 AWIPS Development Environment，以 Eclipse 整合開發環境為主，用來協助 AWIPS II 系統開發。

AlertViz (警告工具列)

為 AWIPS II 中用來提供使用者各種異常警告訊息的常駐工具列，地位等同 AWIPS I 中的 Guardian，在 AWIPS II 中，必須確定 AlertViz 已啟動方可正常啟動 CAVE。

AWIPS II (第 2 代先進交談式天氣處理系統)

AWIPS 全名為 Advanced Weather Interactive Processing System，本系統基於服務導向架構，最主要的兩個子系統為 CAVE 及 EDEX，目的在整合顯示各種即時氣象資訊，以提供氣象預報人員決策參考。其第 1 代系統(AWIPS I)經本局本土化過後命名為天氣整合與即時預報系統(WINS)，為預報中心平日作業所使用的決策支援系統。

Camel (應用整合框架)

為一企業應用整合設計模式(Enterprise Integration Patterns)的實作，主要用來介接各種不同的軟體元件，透過規則的設定我們可以將不同的元件組織起來，實作出期望的商業流程。在 AWIPS II 中主要用來組合串接 EDEX 服務，為其服務導向架構的重要基礎建設。

CAVE (AWIPS 共通顯示環境)

CAVE 全名為 Common AWIPS Visualization Environment，建立在 Eclipse RCP/JFace/SWT 的基礎上，為 AWIPS II 主要的使用者介面，在此架構下 D2D 只是 CAVE 的其中一個「呈現模式(Perspective)」。

EDEX (環境資料交換服務)

EDEX 全名為 Environmental Data EXchange，架構在 Camel/Spring 框架下，主要負責即時資料的解碼作業及回應 CAVE 等客戶端程式的請求。

HDF5 (階層式資料格式)及 PyPIES (HDF5 存取管理系統)

HDF5 全名為 Hierarchical Data Format Version 5，為 AWIPS II 解碼後資料的儲存格式；PyPIES 全名為 Python Process Isolated Enhanced Storage，用來處理所有的 HDF5 存取作業，類似資料庫管理系統的角色，讓其他程式不用煩惱 HDF5 底層的存取方式。

NetCDF (網路共通資料格式)

全名為 Network Common Data Form，為第 1 代 AWIPS I 主要的資料儲存格式。AWIPS II 已改用 HDF5 作為資料儲存格式。

PostgreSQL (資料庫)

為一關聯式資料庫管理系統(Relational Database Management System)，在 AWIPS II 中用來儲存產品的描述性資料(metadata)及部分解碼資料。

QPID (訊息中介軟體)

為 AMQP 訊息協定(Advanced Message Queuing Protocol)的實作，AWIPS II 服務間溝通的橋樑，在 Java 程式中可透過 JMS(Java Message Service) API 來進行訊息收發等操作。

RCM (雷達服務)

全名為 RCM(Radar Coded Message)雷達服務(Radar Server)，此服務會自 ORPG/SPG(open systems Radar Product Generator/Supplemental Products Generator)產品伺服器接收雷達資料，接著發送「資料到位」的 QPID 訊息通知 EDEX 服務進行解碼作業。

Spring (軟體框架)

Spring 軟體框架使用 Spring 容器(BeanFactory/ApplicationContext)及控制反轉/相依注入(Inversion of Control/Dependency Injection)等概念來協助 Java 物件(Plain Old Java Objects)的設定、建立及使用，藉以管理 Java 應用程式複雜的相依性，簡化系統開發人員的工作。

WINS (即時預報系統)

本局將 AWIPS 本土化後重新命名為天氣整合與即時預報系統(WINS, Weather Integration and Nowcasting System)，其本質上和 AWIPS 並沒什麼不同。

6. 參考資料

(1) UCAR 的 AWIPS II 專案網頁

<http://www.unidata.ucar.edu/software/awips2>

(2) 美國國家氣象局(NWS) AWIPS Technology Infusion 網頁

<http://www.nws.noaa.gov/ost/SEC/AE>

(3) 雷神公司(Raytheon) AWIPS II 簡介網頁

<http://www.raytheon.com/capabilities/products/awips/index.html>