鄉鎮逐時天氣預報中央控管系統

李宥樺[1]　張旭[1]　蔡立夫[1]　楊淑蓉[2]
中央氣象局氣象預報中心[1]　資拓宏宇國際股份有限公司[2]

摘要

中央控管系統爲鄉鎮逐時天氣預報計劃底下子系統之一，所提供服務包含系統監控、程序流程控制、統一來源資料倉儲。鄉鎮逐時天氣預報計劃相關子系統分別爲高解析度網格點之氣象分析子系統、高解析度統計預報子系統及互動智慧天氣辨識子系統。爲達到降低維運人力以及穩定的自動化流程，提供流程控管、監控服務。使用流程控管服務須先在中央控管註冊程序啓動訊號，藉由事件驅動或定時啓動方式將透過SSH(Secure Shell)及 RSH(Remote Shell)方式完成其他系統之作業程序並且回報程序處理狀況。中央控管系統監控服務包含 CPU、硬碟空間、網路狀態，可設定警示範圍以提醒維護人員通報處理。因氣象局資資料種類繁多，爲提高存取效能給各子系統進行資料運算，因此使用高速儲存媒體 SAN(Storage Area Network)，讓各子系統能夠使用 SAN 所提供之 LUN (Logical Unit Number)直接存取資料以降低資料傳輸時間成本。FIFOW 各子系統伺服器均採用雙機備援，並且主機與輔機資料互相同步，提升系統穩定度、降低作業風險、減少維運人力、提升技術開發人力、服務不間斷持續營運之目的。

關鍵字：中央控管系統、鄉鎮逐時天氣預報計劃、系統監控、程序流程控制、雙機備援

一、前言

繼氣象業務全面電腦化後，自動化作業流程乃是處理氣象局業務重要之經驗。雖已有效降低人力成本，但自動化作業流程伴隨著維運人力的問題。目前系統數量日漸成長，系統風險也相對提高。維運人員之人力也逐漸增長，爲有效將低維運人員人力及承受系統風險之壓力，鄉鎮逐時天氣預報計劃(Fine Information of Formosa Weather 以下簡稱 FIFOW)透過中央控管系統(簡稱 CCTL)掌握流程控管、主輔機監控、雙機備援、統一資料倉儲等機制將系統穩定度提高，可將人力提升至開發領域。FIFOW 計劃上除了 CCTL 以外另有四個子系統分別爲高解析度網格點之氣象分析子系統(簡稱 GT)、高解析度統計預報子系統(簡稱 SFM)及互動智慧天氣辨識子系統(簡稱 ISWIS)均由 CCTL 負責所有子系統的流程控管、監控作業、資料來源索取、資料解碼、格式轉換等重要工作，資料來源由中央控管從上游取得後存放在資料倉儲，以利 FIFOW 所有相關系統順利完成各項任務。

二、硬體架構

FIFOW 採用 Intel Xeon 伺服器級之中央處理器並且各自獨立運作，作業系統均採 RedHat Enterprise Linux 5.5，均有相同等級之備援機、有各自獨立空間，內部光纖主幹進行資料傳遞，各子系統資料源均來自共同資料倉儲，統一資料來源管理、產品保存。每日有大量資料需求如觀測資料、網格資料、模式資料等，因此需要高速儲存媒體以降低資料傳輸時間成本。

(一) 硬體評估

中央控管評估各子系統(GT、SFM、ISWIS)需要即時運算龐大資料後發現 NAS(Network Attached Storage)的效能讓各子系統獲得資料所需時間上造成過久的延遲導致氣象相關產品無法在限定之時間內產出，而造成相當大的損失。因此提供專門儲存線上作業用資料的 SAN 資料倉儲，允許子系統透過光纖通道(Fiber Channel)以小型計算機系統介面協定(Small Computer System Interface Protocols，簡稱 SCSI Protocols)所提供的邏輯單元節點(Logical Unit Number，簡稱 LUN)。保證上游資料進入 SAN 空間後，子系統可以立即讀取資料作運算並產出 FIFOW 相關氣象產品。雖然 SAN 的成本比 NAS 高出許多，但能夠在限定時間內完成產品之產製，才是當務之急。

(二) FIFOW 整體架構

CCTL 所提供的服務如圖 1 所示，各子系統透過 LUN 直接取得位於 EVA8000 SAN 空間的即時資料，SAN 空間有限因此將非即時資料保存到磁帶櫃(Tape Library)，未來需要使用到年代久遠的資料可從磁帶櫃將資料調出，提供模式建模、歷史資料查詢等工作。

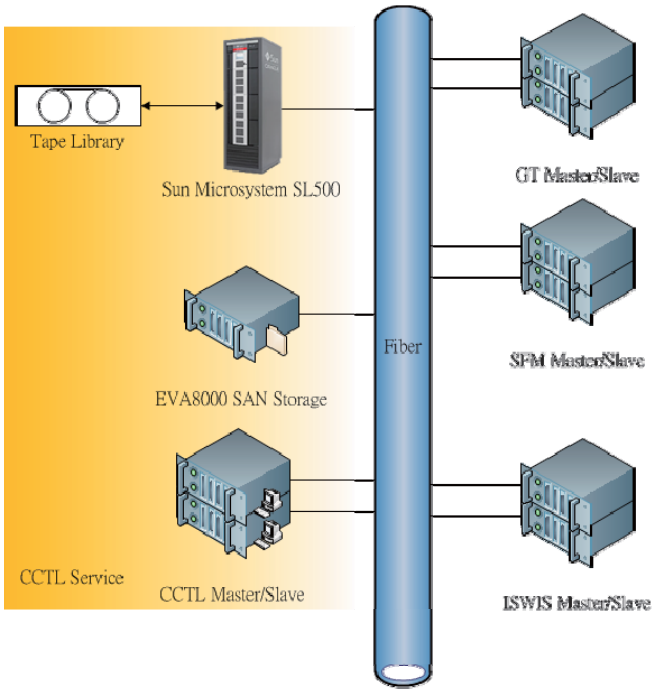

圖 1：FIFOW 硬體架構

三、監控備援機制

爲落實備援機制，各子系統主輔機會互相監控。當備援機偵測到主機無法運作時，備援機將會取代主機所有服務。主機恢復時，立刻進入資料同步之作業，待資料同步完成時再從備援機取回服務權。除此之外也可以由中央控管管理介面得知子系統目前狀態，包含 CPU 狀態、網路狀態、硬碟狀態。

(一) 主輔機監控備援

每個子系統均有被分配到資料交換用空間如圖 2 所示，此空間作爲主機與備援機之間資料傳遞之暫存空間。利用此空間主機與備援機都會將自己的狀態定時寫入此空間並且讀取對方目前狀態。不管是主機或備援機狀態沒有更新就會取得對方的服務權。資料分成檔案與資料庫，檔案同步使用 rsync[1]，資料庫同步使用環狀抄寫(Ring Replication)[2]如圖 3 所示。當主機發生無法運作時，備援機會立刻使用 IP Take Over[3]，取得主機的 IP，當主機恢復後立刻進行檔案與資料庫同步，資料同步完成主機取回 IP 與服務權。

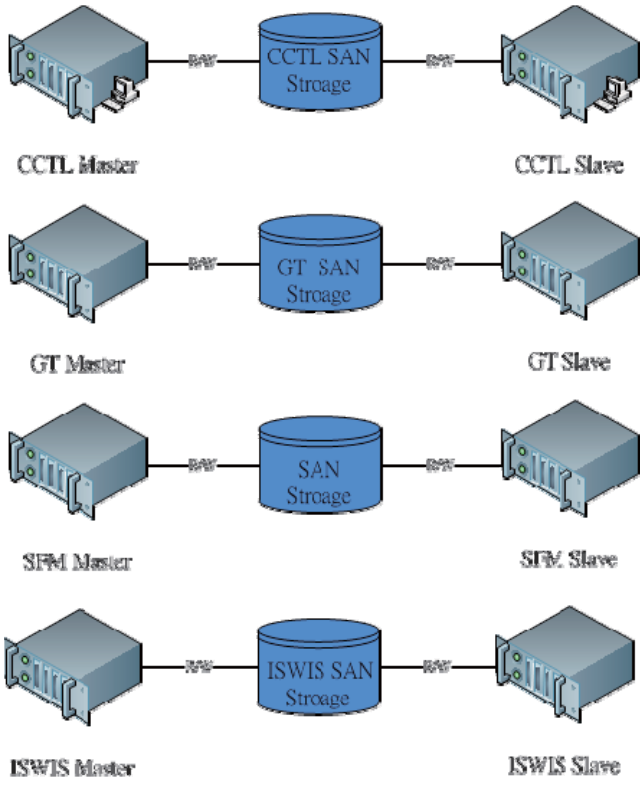

圖 2：資料交換空間

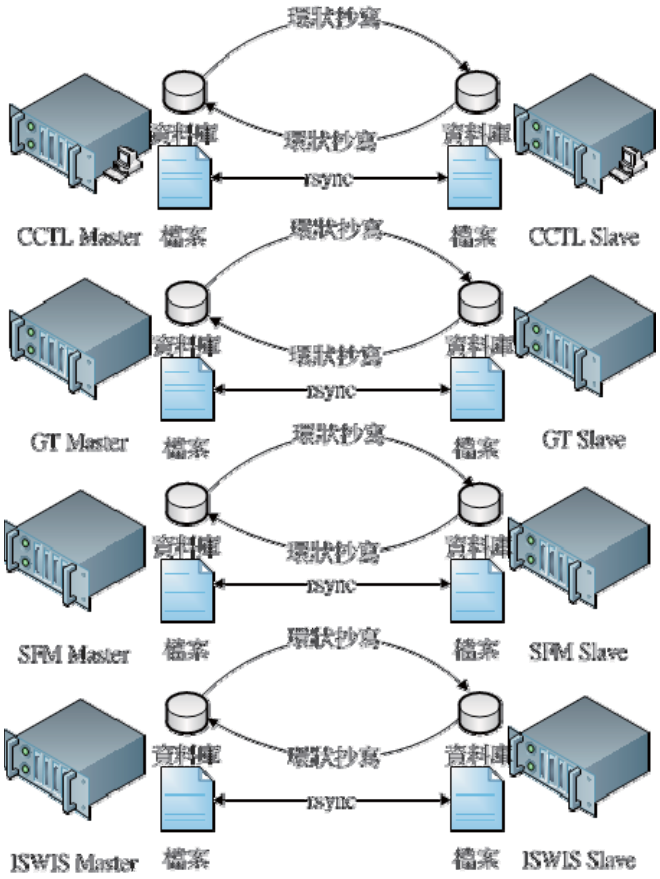

圖 3：資料同步

(二) 子系統狀態監控

在CCTL管理介面當中每分鐘偵測一次子系統狀態，結果如圖 4 至 6 分別爲 CPU、硬碟、網路每分鐘的監控資訊。

偵測時間▼	主機	警示水位(%)	目前水位(%)
2011-05-19 16:54:01	CCTLsv1	80	12
2011-05-19 16:53:01	CCTLsv1	80	8
2011-05-19 16:52:01	CCTLsv1	80	8

圖 4：CPU 每分鐘監控資訊

偵測時間▼	主機	位置	目前水位(KB)	高水位(%)	低水位(%)	百分比(%)	狀態
2011-05-19 16:50:01	CCTLsv1	/users2	1715681020	90	0	86	OK
2011-05-19 16:50:01	CCTLsv1	/home	21714272	80	0	47	OK
2011-05-19 16:50:01	CCTLsv1	/users3	360223892	90	0	74	OK

圖 5：硬碟每分鐘監控資訊

偵測時間▼	主機	偵測項目 [系統/主機]	狀態
2011-05-19 16:53:08	CCTLsv1	172.16.7.31	OK
2011-05-19 16:53:08	CCTLsv1	172.16.7.33	OK
2011-05-19 16:53:08	CCTLsv1	172.16.7.73	FAIL

圖 6：網路每分鐘監控資訊

四、資料倉儲

資料倉儲分成線上作業用與非線上作業用之倉儲，線上作業用之倉儲需要高速的資料傳遞效率因此使用 SAN，歷史資料部分爲非上線作業用均保存到磁帶櫃以節省空間成本。磁帶櫃系統爲 Sun Microsystem SL500 LTO4，使用 SAN 1TB 空間當作磁帶櫃緩衝區，平常各子系統作業用資料超過作業用時間就會透過中央控管系統將資料移到緩衝區，其他子系統只有唯讀權限。當緩衝區儲存容量超過 80%就會發送任務信號啓動 CCTL 程序，將緩衝區資料移進磁帶櫃，磁帶櫃存取機制如圖 7 所示。

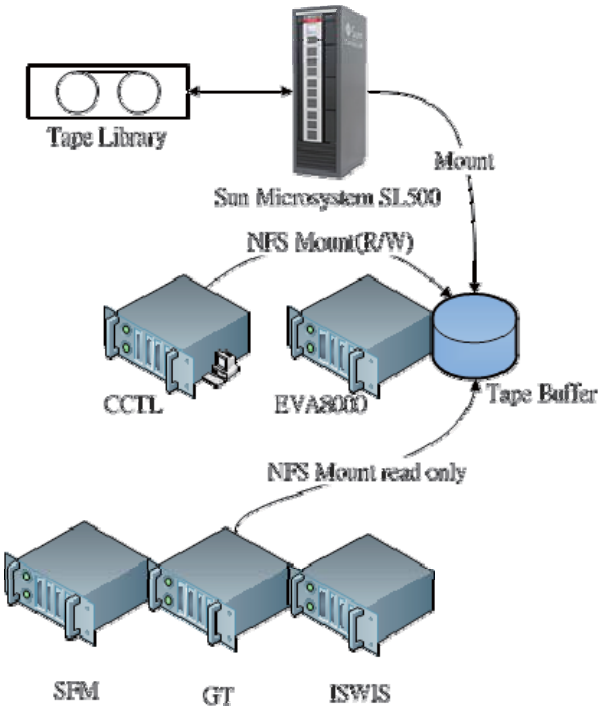

圖 7：磁帶櫃存取機制

五、流程控管服務

流程控管服務是 CCTL 業務重要一環，各子系統向 CCTL 註冊，啓動程序信號以及設定程序組態來完成程序串接工作。當上游資料進入資料倉儲後，CCTL 透過 Inotify[6]機制發送信號給子系統以啓動子系統作業流程。由 CCTL 掌控之程序可以監控其運作狀況，當程序有潛在漏洞或是例外狀況發生時，可即時反應給開發人員及維護人員針對突發

狀況進行追蹤。但流程控管主要定位在於控管程序流程進行，爲保證程序執行之穩定度，凡經由 CCTL 控管之系統程序須在本機端經過至少兩週以上之壓力測試，方可將程序流程交由 CCTL 控管。

(一) Inotify 機制

上游資料會進入到資料倉儲當中特定目錄，當資料到位時，需要呼叫子系統啓動程序來完成作業。傳統方式爲定期檢查資料夾是否有新檔案進入，隨著檔案越多資料表就越龐大，檔案比對次數也會大幅增加。此方式逐漸被事件驅動型式的目錄監控程式所取代，不再以比對檔案方式來進行資料夾監控。直接由硬體 I/O 送出檔案操作訊息當做事件通知。Linux Kernel 從 2.6.13 版本後就直接支援 Inotify，開發者可自行撰寫接收檔案 I/O 訊息之程式，並且將訊息加以定義以客製化成特定動作固定觸發特定程序之功能。Inotify 延伸功能可以達到程序託管、檔案收送、程序重做等相關延伸功能，以消除作業上因檔案傳輸失敗引發作業停滯狀況，如圖 8 爲透過 Inotify 實做遠端檔案重送機制。

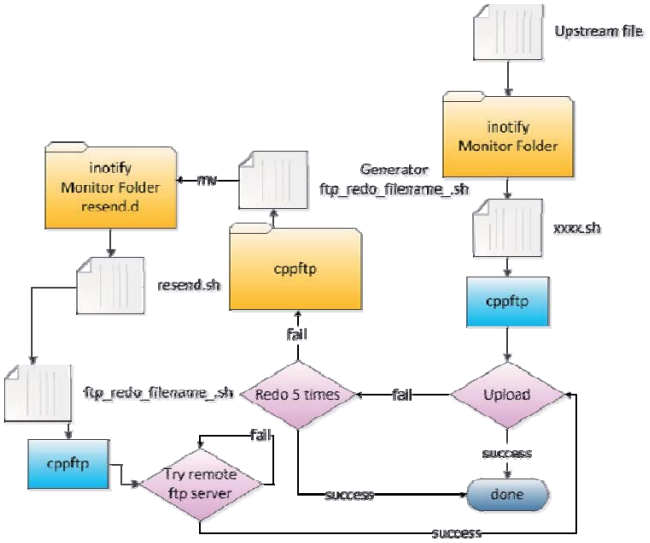

圖 8：遠端檔案重送機制

(二) 流程控管機制

各子系統的之作業流程均由 CCTL 控管，各子系統使用流程控管服務之前會先詢問 CCTL 主機位置，得知主機位置後進行註冊。註冊完畢後，CCTL 開始發送程序啓動訊號給子系統完成作業程序如圖 9 所示。

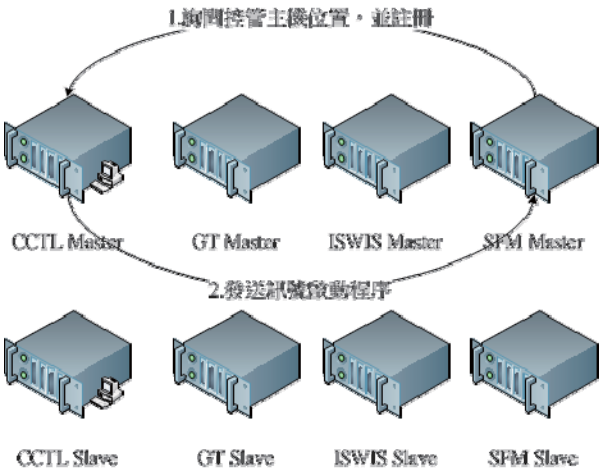

圖 9：流程控管機制

(三) 流程控管設定介面

各子系統需要由 CCTL 做流程控管時必須進行訊號註冊、工作組態設定、設定訊號觸發工作如圖 10 至 12，以防止 CCTL 收到未註冊之訊號而執行未授權之程序。

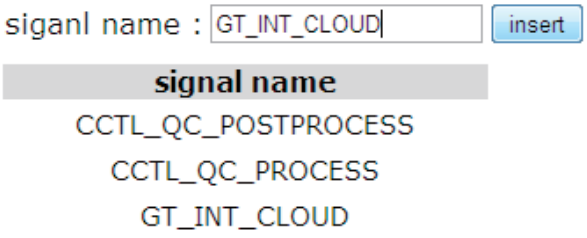

圖 10：訊號註冊

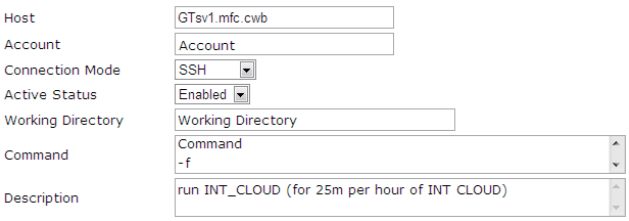

圖 11：工作組態設定

圖 12：設定觸發工作

六、結論

CCTL 系統是相當有彈性的流程控制系統，可自訂流程串接。當然流程串接越複雜會導致作業時間拉長，因此串接流程必須經管理者以及開發者做詳細溝通，以達成高效能的控管作業流程，並且程

式運作正常與否均可以確實回報處理狀況。雖然在串接過程當中需要花費較多心思以及時間與程式設計師溝通，只要將流程定義完善日後系統方可保證穩定運作並且有效率地執行。因氣象局資料來源龐大，產出產品所占用空間也不容小覷，因此也依照各子系統需求定義了合適的資料保存時間及儲存規範以節省資料倉儲整體空間。

七、未來展望

目前監控介面是純文字表示，未來將全面圖形化，使監控人員輕鬆瀏覽且容易上手。流程控制部分將串接完成之程序，自動產生流程圖，並且顯示目前正在執行之程序。若程序若發生錯誤可以選擇此流程重做，或是全部流程重做。目前資料庫同步部採用 Replication，未來使用資料庫叢集(Database Cluster)[4]來完成資料庫備援，CCTL 將進行需求評估，不排除使用運算叢集(Computing Cluster)[5]架構來擔任需要高運算量之程序。

八、參考文獻

[1] Tridgell, P. Mackerras,1996:"The rsync algorithm", Department of Computer Science Australian National University Canberra. ACT 0200 Australia.

[2] M. Kindahl, L. Thalmann,2009:" MySQL Replication Tutorial ",My SQL Conference & Expo .

[3] C. FETZER , N. SURI ,2003:"Practical Aspects of IP Take-Over Mechanisms", Proceedings of 9 th IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS 2003), pages 250 - 254.

[4] S. Gan¸carski, H. Naacke, E. Pacitti, and P. Valduriez.2002:"Parallel processing with autonomous databases in a cluster system", In Int. Conf. On Cooperative Information Systems (CoopIS).

[5] M. Baker, et al,2000:"Cluster Computing White Paper", University of Portsmouth.

[6] R. Love., 2005: " Kernel korner: intro to inotify", Linux Journal, Volume 2005, Issue 139, Pages: 8.