

封面格式

行政院及所屬各機關出國報告
(出國類別：研究)

「高階模擬架構（HLA）」實務訓練課程報告

服務機關：中山科學研究院

第一研究所

出國人職稱：荐聘技士、少校技士

姓名：白方平、盧南華

出國地區：美國佛羅里達州奧蘭多

出國期間：91年06月22日至91年06月30日

報告日期：91年07月29日

I8/
009/03228

國外公差報告

中山科學研究院

國外公差心得報告

批 示				
中山科學研究院 院長 宋大德 10456				
公年 差度	九十一	所屬單位各級 主管	政戰部	企劃處
單 位	一 模 擬 組 所	第一研究所 所長 傅鶴齡 1000	政戰部 副部長 謝芳慶 1001	請將資料上傳行政院研考會網站，並將報告裝訂五份送貴單位專責人員後轉送本處四份、計品會一份，電子檔亦送交本處科技技術組。
級 職	荐少校技士	第一研究所 副所長 何志遠 1000	政戰部 副組長 姜永銓 1001	企劃處 副處長 郭永福 1001
姓 名	白 盧 方 南 平 華	第一研究所 組長 單 弘 1031	政戰部 組長 謝芳慶 1001	企劃處 組長 邱炎川 1001

(九二)一所行會17號

916109



國外公差人員返國報告主官（管）審查意見表

本次國外公差係為參訪中佛大 IST 研發設計成果及參與 DISTI 公司舉辦之 HLA 實務訓練課程。二位出差人員參與課程時認真負責，並就未來戰車模擬器設計可能發生之問題與 DISTI 公司工程人員研討，對 HLA 之設計研發工作有極大之幫助；實務課程中帶回之程式原始碼也澄清了許多設計上之疑問，為一次成功之實務訓練課程。

所屬單位主官

第一研究所 副所長 吳中興 0800	第一研究所 副所長 何志遠 0920	第一研究所 所長 傅鶴齡 0921
---	---	--

報 告 資 料 頁			
1. 報告編號：	2. 出國類別： 研究	3. 完成日期： 91 年 07 月 29 日	4. 總頁數： 34 頁
5. 報告名稱：「高階模擬架構（HLA）」實務訓練課程報告			
6. 核准 文號	人令文號	(九一)銓鑑字第〇〇三一三七號	
	部令文號		
7. 經 費		新台幣：313,293 元	
8. 出(返)國日期		91 年 06 月 22 日至 91 年 06 月 30 日	
9. 公 差 地 點		美國佛羅里達州奧蘭多	
10. 公 差 機 構		中山科學研究院第一研究所	
11. 附 記			

行政院及所屬各機關出國報告提要

出國報告名稱：「高階模擬架構（HLA）」實務訓練課程報告

頁數 34 含附件：☒是☐否

出國計畫主辦機關/聯絡人/電話

國防部中山科學研究院第一研究所/白方平/(04)27023051 分機 503769

出國人員姓名/服務機關/單位/職稱/電話

白方平、盧南華/國防部中山科學研究院第一研究所/模擬組/技士、技士
/(04)27023051 分機 503769

出國類別：☐1 考察☐2 進修☒3 研究☐4 實習☐5 其他

出國期間：91 年 06 月 22 日 出國地區：美國佛羅

至 91 年 06 月 30 日 里達州奧蘭多

報告日期：91 年 07 月 29 日

分類號/目

關鍵詞：高階模擬架構（HLA，High Level Architecture）

內容摘要：（二百至三百字）

本院執行陸軍裝甲兵學校委製之「M48H/M60A3 組合型戰車訓練模擬器」專案，其中合約所要求之高階模擬架構（HLA，High Level Architecture），為美國軍方近年來推廣之先進模擬器聯網演訓技術，由於國內之學術界及產業界均還處於研究及測試階段，並無成熟完整之產品功能可供參考；而 HLA 在美國軍方已行之有年，DISTI 公司本身也有參與美國軍方之部份專案研發，具有相當之程式設計及開發實務經驗。

在本次實務訓練課程中，除了一般性的理論介紹外，DISTI 公司還提供其發展測試驗證過之 HLA 程式原始碼，在訓練課程中能更深入的瞭解 HLA 設計架構，也澄清了許多自我摸索設計上之疑問；同時有就未來戰車模擬器設計可能發生之問題與 DISTI 公司工程人員研討，對 HLA 之設計研發工作有極大之幫助。

所以本次國外差旅，除了對 HLA 之設計架構有更進一步的體驗外，同時瞭解國外廠商目前發展之程度及方向，並建立聯繫之管道，足可做為本院後續開發 HLA 架構很好的範本及依循的方向。

中山科學研究院公差出國人員報告目錄

項	目	頁碼
壹、出國目的及緣由	-----	1
貳、公差心得	-----	2
參、效益分析	-----	15
肆、國外工作日程表	-----	19
伍、社交活動	-----	30
陸、建議事項	-----	30

附件：課程表

壹、出國目的及緣由

本院執行陸軍裝甲兵學校委製之「M48A/M60A3 組合型戰車訓練模擬器」專案，其中合約要求以高階模擬架構（HLA，High Level Architecture）作為模擬器聯網之基礎架構，由於 HLA 為美國軍方近年來所開發推廣之先進模擬器聯網演訓技術，國內之學術界及產業界均還處於研究及測試階段，並無成熟完整之產品功能或發展經驗可供參考。

為確保專案工程品質、降低研發風險、並達成如期如質交機之目標，經由網路查詢及美國 HLA 產品開發主要廠商 MAK Technology 及 AEGIS 公司推薦，得知美國 DISTI 公司曾參與美國軍方之 HLA 相關專案研發，具有相當之程式設計及開發實務經驗，且針對 HLA 開辦一系列之訓練課程，其客戶包括美國軍方及各大國防承包商，為我方獲取 HLA 相關資源及訓練之良好來源。

DISTI 於六月底開辦 HLA 實務訓練課程(A Practical Guide to HLA Development)，除了一般性的理論介紹外，並提供其發展測試驗證過之 HLA 程式原始碼做為實務訓練課程之訓練題材，應可從中獲取其設計經驗，以澄清自我摸索設計上之疑問，因此申請本次赴美參加該公司主辦之該項課程。並期藉此建立討論管道，以做為後續設計發展諮詢之資源。

貳、公差心得

一、高階模擬架構概述：

在國防資源有限現況下，運用模擬提高武器籌獲、分析、訓練以及測評效益，已成為世界潮流。由於網路科技的進步，將既有之模訓裝備透過網路加以連結，以提供協同作戰演訓功能之分散互動式模擬更成為模擬技術發展的一門顯學，甚至可以應用在軍事用途以外，諸如工程系統設計模擬、危機處理模擬訓練、娛樂休閒等領域。

HLA 是由美國國防部「國防模式模擬辦公室 (Defense Modeling and Simulation Office，簡稱 DMSO)」所主導開發的一個分散互動式模擬的基礎架構，它是 DIS 規格與 ALSP 規格的精進版本，使得網路模擬系統的設計與發展更為制式化且更為容易，並於 2000 年 9 月成為國際 IEEE 1516 的標準。透過此基礎架構可以連結原先為不同目的所建立的系統、不同時期的技術、不同廠商的產品、以及不同的平台，而使它們可以相互運作，進而建構出一個時空一致的、複雜的、虛擬的戰場。戰機模擬器、戰車模擬器、船艦模擬器可透過此架構加以整合，模擬出整個戰場之情況，更可將兵棋系統整合進來以執行建構式(Constructive)模擬演訓，甚至單兵也可以透過個人模擬裝備進入這個模擬戰場進行戰技訓練。在系統架構上 HLA 採用了開放式物件導向的架構，並引入物件導向的軟體發展觀念。

HLA 之主要目的為整合各種型態的模擬系統(包括真人操作真實系統、模擬器系統和兵棋軟體)並且提升模擬系統各元件之重複使用性(reuse)與相容性。在 HLA 的環境下，所有獨立的模擬單元，不管是兵棋軟體、模擬器或是真實武器系統，統稱

為 Federate。一個 Federate 其實可以負責多個模擬物件之屬性運算與互動行為。而由各個 Federate 所共同組合互動所建構出之演訓環境稱為 Federation。為構築出這樣一個滿足演訓共同需求的通用元件化架構，並提供時空一致性及模擬管理之功能，HLA 在架構上主要可分為三大分系統：

1. HLA Compliance Rule：一組設計與管理模擬環境的規則。這些規則是為了要整合不同模擬系統達到完整的互動所訂下的方針與條約。
2. Object Model Template (OMT)：描述 Federation Object Model 與 Simulation Object Model 的物件描述樣板，以作為模擬實體間資料定義、交換與協調的共用標準。
3. Interface Specification: RTI是對整體模擬環境提供必要服務以達成互動性的基礎架構，為一分散式物件系統，RTI必須依照介面規範提供完整的介面函式以作為RTI與Federate間的標準界面。

其中RTI實際上就是一個分散式物件系統，主要的目的在於提供標準的介面服務 (Interface Services) 給應用軟體 (Application Software)。模擬元件與模擬元件之溝通均需以呼叫RTI API來執行，此外並透過RTI來達成模擬管理及時間一致性之功能。RTI是以軟體元件 (Software Components) 的方式實作組合而成，這些元件提供了Federates之間進行互動以及滿足各項管理功能所需之標準介面。在Federates與RTI之間定義了6個服務群組：Federation Management、Declaration Management、Object Management、Ownership Management、Time Management、Data Distribution Management，其功能請參見下節說明。

除此之外，HLA 也提供了一個模式與模擬的物件架構，主要目的是要使各個模擬系統在大型演訓的應用中，能夠很容易地達成交互作用(Interoperation of Simulations)以及模擬元件重複使用的目標。由於 RTI 以及 Compliance Rules 在整個 HLA 演訓中是固定的，因此如何建立滿足 OMT 之模擬物件以及開發滿足 RTI 介面標準之介面軟體為建構 HLA Compliant 演訓之關鍵。

二、高階模擬架構程式發展：

1. Federation Management：所提供之服務如下表：

服務內容/API	觸發者
Create Federation Execution	Simulation
Destroy Federation Execution	Simulation
Join Federation Execution	Simulation
Resign Federation Execution	Simulation
Register Federation Synchronization Point	Simulation
Synchronization Point Achieved	Simulation
Request Federate Save	Simulation
Federate Save Begun	Simulation
Federate Save Complete	Simulation
Request Federation Restore	Simulation
Federate Restore Complete	Simulation
Confirm Synchronization Point Registration	RTI
Announce Synchronization Point	RTI
Federation Synchronized	RTI
Initiate Federate Save	RTI
Federation Saved	RTI
Confirm Federation Restoration Request	RTI
Initiate Federate Restore	RTI
Federation Restored	RTI

與 RTI 溝通的第一步是建立一個 RTI Ambassador 的物件，RTI Ambassador 物件的功能是提供 Federate 與 RTI

之間的介面功能，要建立 RTI Ambassador 物件，首先需要 include `<RTI.hh>` 檔案，然後再建立一個 `RTI::RTIAmbassador` 型別之全域變數。

下一步則是建立一個 `Federate Ambassador` 物件，`RTI::FederateAmbassador` 類別是以虛擬類別的方式存在，我們所建立的 `Federate` 必須建立一個繼承自該類別的物件並實作出所需的方法(Method)。

接下來則是嘗試 `Create Federation`，所需呼叫的函式是 `RTIAmbassador::createFederationExecution`，此函式所需用到的兩個引數：`Federation` 名稱字串以及所引用 `.fed` 檔名路徑的字串。

註 1：在絕大部分的情況下，每一個 `Federate` 都需要嘗試建立 `Federation` 的動作，因為即使 `Federation` 已經由其他 `Federate` 成功建立，其他 `Federate` 在呼叫 `createFederation` 也僅會產生一個次要 `exception` 顯示該 `Federation` 已存在。但 `createFederation` 動作還有一項功能就是會去做 `.fed` 檔案一致性的檢查可避免 `.fed` 檔案不一致的情況發生。

註 2：當呼叫 `createFederation` 時，`Local RTI Component` 會與 `rtiexec` 程式溝通，`rtiexec` 程式內部儲存有所有執行中 `federation` 及其參與該 `federation` 所有 `federate` 的資料。

`CreateFederationnc` 函式會產生幾項 `exception`，我們必須實作出這些 `exception` 的 `handling` 機制以避免 `exception` 導致程式終止的情況發生。依據 RTI API 文件，`createFederation` 會產生以下幾項 `exception`：

- Federation Execution Already Exists
- Could Not Open FED
- Error Reading FED
- Concurrent Access Attempted
- RTI Internal Error

其中 Federation Execution Already Exists 僅為一警告訊息而可以忽略，但其他 exception 則必須提供 exception handling.

註 3：當成功建立 Federation 後，由於 Federation Executionvm 需要一些時間來啟動，因此在 joinFederation 之前需要等待幾秒鐘的時間，通常設定為 5~10 秒鐘，因此在 createFederation 之後呼叫 sleep(10) 來滿足這個需求。

下一個步驟則是 joinFederation，呼叫該項服務所需提供的是 Federate 名稱、所參與 Federation 名稱以及所對應使用之 Federate Ambassador 物件名稱。縮 joinFederationc 函式同樣也會產生幾項 exception，我們同樣也必須實作出這些 exception 的 handling 機制以避免 exception 導致程式終止的情況發生。依據 RTI API 文件，joinFederation 會產生以下幾項 exception：

- Federation Execution Does Not Exist
- Federation Already Execution Member
- Could Not Open FED
- Error Reading FED
- RTI Internal Error
- Concurrent Access Attempted
- Save in Progress

➤ Restore in Progress

註 4：當 *Federation Execution Does Not Exist Exception* 發生時，並不一定表示有問題發生，其原因在於 *RTI Call* 之間並沒有一個同步機制存在，因此導致 *federation creation* 及 *federation joining* 之間發生 *racing*(競逐)問題。其解決方法之一為採用註 3 的做法，在 *createFederation* 之後等待數秒鐘使得 *Federation Execution* 可以正確啟動。另外一個做法則是以一個迴圈不斷去 *catch joinFederation* 的 *exception*，若有 *Federation Does Not Exist exception* 產生，則等待一段時間後重新嘗試，直至成功 *join* 或超過預設之嘗試次數為止。

2. Declaration Management：所提供之服務如下表：

服務內容/API	觸發者
Publish Object Class	Simulation
UnPublish Object Class	Simulation
Subscribe Object Class Attributes	Simulation
UnSubscribe Object Class Attributes	Simulation
Publish Interaction Class	Simulation
UnPublish Interaction Class	Simulation
Subscribe Interaction Class	Simulation
UnSubscribe Interaction Class	Simulation
Start/Stop Registration For Object Class	RTI
Announce Synchronization Point	RTI
Turn Interaction On/Off	RTI

首先需呼叫 *getObjectClassHandle*、*getInteractionClassHandle* 函式以獲得 *Object Class* 或 *Interaction Class* 之 *RTI Handle*，此函式所需要之引數為 *FED* 檔案所定義之 *Object Class* 或 *Interaction* 名稱，而其傳回值為一 *RTI::ObjectClassHandle* 或

RTI::InteractionClassHandle 型態之變數。若需要獲得 Handle 之 class 為一子類別，則需以”父類別 . 子類別”之形式來獲得其正確之 Handle 值。接著需呼叫 getAttributeHandle 以獲得 Attribute 之 Handle 值，其次產生 AttributeHandleSet 物件，並以其 add 方法將所欲 Publish/Subscribe 之 Attributes 加入此 Attribute List 中，最後呼叫 publishObjectClass 或 subscribeObjectClassAttributes 完成 Publish/Subscribe 功能。

至於 Interaction Class 需要呼叫 getParameterHandle 函式以獲得 Interaction Class Parameter 之 Handle 值，此函式所需要之引數為 Interaction Class 之 Handle 與 Parameter 名稱。Interaction Class 及 Parameter 之 Handle 值應以全域變數之方式儲存。

其次利用 RTI Ambassador 之 subscribeInteractionClass 函式來 Subscribe 某一 Interaction Class，該函式所需之引數只有一個，即 Interaction Class Handle。

Publish 與 Subscribe 的呼叫方式非常接近，同樣也是在取得 Interaction Class 之 Handle 後呼叫 publishInteractionClass 函式，所需之引數亦為 Class Handle。

3. Object Management：所提供之服務如下表：

服務內容/API	觸發者
Register Object Class	Simulation
Update Attribute Values	Simulation
Send Interaction	Simulation
Delete Object Instance	Simulation
Local Delete Object Instance	Simulation
Change Object Transport Type	Simulation
Change Interaction Transport Type	Simulation

Request Attribute Values Update	Simulation
Discover Object Instance	RTI
Reflect Attribute Values	RTI
Receive Interaction	RTI
Remove Object Instance	RTI
Attributes In Scope	RTI
Attributes Out Of Scope	RTI
Provide Attribute Values Update	RTI
Turn Updates On/Off For Object Instance	RTI

物件管理服務包括物件的註冊/刪除，Interaction 的收發與 Interaction 機制的開關、屬性資料的更新、傳輸機制的改變等。它受到 Time Management 的管制。資料更新方式分為主動執行資料更新以及被動執行資料更新兩種方式，並提供中斷資料傳送服務之功能。

首先需執行 RTI Object 之註冊動作，將物件所屬 Class 之 RTI Handle 作為引數呼叫 registerObject Instance 以產生並註冊該模擬物件。

當註冊模擬物件之後，我們必須定時去處理屬性更新問題，透過 UpdateAnAttribute 或是 updateAttributeValues 函式可以更新一個或多個屬性資料。

當接收到由 FederateAmbassador 所產生之 FindObject 事件時需決定 discover 到物件的種類，並加入該物件之 Object List 或矩陣中。

執行 Send Interaction 功能首先需建立 Parameter Handle Value Pair Set，此 set 所儲存的是一組一組 Parameter Handle 及其對應數值的組合，RTI 就是利用這些數據組來傳送 Interaction 的參數。RTI::ParameterSetFactory 類別就是用來產生這些數據組。我們所呼叫的函式是

RTI ::ParameterSetFactory ::create()，並以所要產生的數據組數目作為其引數。

4. Ownership Management：所提供之服務如下表：

服務內容/API	觸發者
Unconditional Attribute Ownership Divestiture	Simulation
Negotiated Attribute Ownership Divestiture	Simulation
Attribute Ownership Acquisition	Simulation
Attribute Ownership Acquisition If Available	Simulation
Attribute Ownership Release Response	Simulation
Cancel Negotiated Attribute Ownership Divestiture	Simulation
Cancel Attribute Ownership Acquisition	Simulation
Query Attribute Ownership	Simulation
Is Attribute Owned by Federate	Simulation
Request Attribute Ownership Assumptions	RTI
Attribute Ownership Divestiture Notification	RTI
Attribute Ownership Acquisition Notification	RTI
Attribute Ownership Unavailable	RTI
Request Attribute Ownership Release	RTI
Attributes Out Of Scope	RTI
Confirm Attribute Ownership Acquisition Cancellation	RTI
Inform Attribute Ownership	RTI

Ownership Management 是負責 federates 轉移物件屬性所有權的控制，所有權轉移是根據 federation execution 設計規劃以及由 RTI 負責調解(仲裁)方式，轉移方式可以是主動式或是強迫式。並需事先已對此屬性宣告 publication 與 subscription 的需求。

做法上可以有強制性及非強制性兩種方式取得所有權，所

呼叫的函式分別為 `AttributeOwnershipAcquisition` 以及 `AttributeOwnershipAcquisitionIfAvailable`，當呼叫 `AttributeOwnershipAcquisition` 後，RTI 會傳送 `RequestAttributeOwnershipRelease` 訊息給 `Attribute Owner`，`Attribute Owner` 則以 `unconditionalAttributeOwnershipDivestiture`、`negotiatedAttributeOwnershipDivestiture` 或 `attributeOwnershipReleaseResponse` 方式回應。

5. Time Management：所提供之服務如下表：

服務內容/API	觸發者
Request Federation/Federate Time	Simulation
Request LBTS	Simulation
Request Minimum Next Event Time	Simulation
Set/Request Lookahead	Simulation
Time Advance Request	Simulation
Next Event Request	Simulation
Flush Queue Request	Simulation
Time Advance Grant	RTI

Time Management 之目的在於控制演習邏輯時間的進行以及協調邏輯時間與真實時間的關係。以確保事件的發生的時間順序與因果關係。

對於時間的管理我們可以有三種考量：

- 沒有一個 `federate` 會收到過去時間所發生的事件(保守的方式)。
- 當一個 `federate` 收到它的過去時間所發生的事件必須倒推回到(roll back) 該時間點以確保邏輯關係之正確(樂觀的方式)。
- 對於即時性模擬系統，採取完全忽略時間管理功能的方式，將收到該事件的時間視為該事件的發生時間。

在 Time management 的環境下，每一個 federate 的時間特性由兩個 Boolean 變數來決定：

- Time-regulating
- Time-constrained

若不需啟動 Time Management 相關功能，則需呼叫 disableTimeConstrained 及 disableTimeRegulation 函式

6. Data Distribution Management：所提供之服務如下表：

Data Distribution Management 之目的在於控制資料傳的傳輸對象，以使資料傳輸更有效率。RTI 利用此服務的資訊來決定 declaration management 所宣告之資料的傳輸對象，所依據的是在執行前所建立的 routing space，routing space 的內容係定義於 FED 檔案中。

在實際執行上首先利用 CreateRegion 函式建立一個所謂的 Region，這個 region 可以是一維的空間(如雷達頻率)以及二維或多維的空間，再利用各資料傳輸函式之 with region 版本來進行資料之更新、收送動作。

三、高階模擬架構觀念澄清：

1. HLA 並未加諸任何物件導向設計需求在我們的模式模擬應用上，事實上 HLA 並未超脫一個外部界面的層級。
2. 在 FOM 的制定上不建議採取自行建置的模式，可以採用美軍既有之 FOM 如 SISO 之 PRP FOM 或取得美國陸軍/空軍/海軍陸戰隊所使用之 FOM。

3. 對於 DMSO RTI 分散式物件的架構說明、對於分散式物件系統包括 CORBA，Java/RMI，以及 DCOM 提出比較與分析以及 MultiCast 網路傳輸方式之說明

4. MOM (Management Object Model)：

- MOM 是每個 FOM 所需強制包含的一個標準物件模型，所有的 federations 具有相同的 MOM，以使 RTI 可以根據其所定義之物件架構描述與管理 federation 的狀態。
- RTI 本身會建立 MOM 物件的 instance，並且維護 MOM 的屬性資料。例如當 federate 加入 federation 時，RTI 會建立一個 class **Manager.Federate** 的 instance，同時建立該實體物件的屬性值。如果一個 federate 想要知道有幾個 federates 參加 federation，則可以 subscribe class **Manager.Federate** 來得知。

四、模式模擬發展趨勢：

由於此次係配合陸軍「M48A/M60A3 組合型戰車訓練模擬器」專案赴美，因此特別針對美國陸軍在模式模擬應用之發展與規劃請教 DISTI 公司及 IST 人員，所獲得之答案如下：

1. 美國陸軍對模式與模擬的分類

- ✓ 研究與發展模擬(Research and Development Simulations)
- ✓ 測試與評估模擬(Test and Evaluation Simulations)
- ✓ 製造與運籌模擬(Production and Logistics Simulations)
- ✓ 分析模擬(Analysis Simulations)

✓ 教育與訓練模擬(Education and Training Simulations)

2. 美國陸軍對模式模擬之主要應用

- 合成虛擬戰場(Synthetic Theater of War, STOW)
 - ✓ 模擬器與模擬器之結合
 - ✓ 模擬器與推演性兵棋系統之結合
- 電腦整合實兵模擬、嵌入式訓練以及與 C4I 系統之結合
- 結合人工智慧之電腦合成武力
- 高階模擬架構
- 電子戰測評等相關應用
- 模式模擬於武獲、需求分析與訓練的全方位應用，稱為 SMART(Simulation and Modeling for Acquisition, Requirements and Training)

5. 未來的模擬走向必定是一種分散式的、Server Based 系統，各類模式模擬如武器模擬、目標機模擬、損害分析、碰撞偵測等均會以分散式方式成為各類運算伺服器所提供的服務。

6. 電腦合成武力除本身武器性能外應具備路徑規劃、視野運算、偵蒐規劃、互動/協同行為合理性、威脅分析以及地形因素運算等設計考量。

7. 在地形資料庫之建置與應用上可參考美軍之 SEDRIS(Synthetic Environment Data Representation and Interchange Specification)地形資料庫系統，在智慧型目標物/半自動合成武力(Semi-Automated Forces)可參考美軍 ModSAF/JSAF 系統，而在空戰模擬系統上則可參考美國空軍之 DMT(Distributed Mission Training)系統。

叁、效益分析

1. 籌辦本次訓練課程的DISTI公司本身在模擬器相關技術尤其是戰車與戰機模擬器之聯網建置上有及其豐富之實務經驗，其主要之業務領域包括軍事模擬科技、聯網模擬科技以及電腦視效科技之研發、教育訓練與顧問諮詢。客戶包括美國德州儀器、波音、洛克希德·馬丁、諾斯洛普·格魯曼、雷神、賽考斯基、休斯等國防武器大廠以及美軍各軍種之模擬訓練單位，並且是南韓及新加坡航太科技之顧問。並曾參與美國軍方部份之合作計畫，具有相當豐富之實務經驗。透過此次訓練課程除自該公司引進HLA相關技術與經驗、建立良好之互動溝通模式外，並獲得該公司負責人首肯可提供我方在HLA建置及規劃上諮詢服務。
2. 此次實務訓練課程中所提供之範例中，有應用HLA架構執行系統聯網控制、資料傳送/接收控制、簡易型戰車連線對抗及系統控制權切換等範例程式，這些範例程式架構相當完整，甚至可以在修改傳輸之Attribute及Interaction後直接套用。對於本院目前正在執行之「M48A/M60A3組合型戰車訓練模擬器委製案」可作為極佳之設計範本與依循方向。此為本次出國收益最大之處。
3. HLA 由於架構非常龐大、專有名詞繁多且定義不易了解，因此要進程式開發所需跨越之門檻相對較高，參加本次訓練由於是以實例加以講解，且程式碼完全公開，所以很容易獲得全般而深入的了解，尤其在以往完全沒有看過實際運作的 Data Distribution Management 以及 Ownership Management，講師均能透過簡單的實例詳述其運作機制以及實作方式，而使參訓人員很容易地即可了解其精要。此

外由於整個課程在名稱上即以“Practical Development Guide”為名，因此完全是以實務的觀點出發，講師無論在授課資料以及上課所補充之說明內容均是以其多年之經驗，針對程式開發過程或對系統架構容易產生混淆之處加以說明，因此對於一些原先錯誤或不清楚的觀念均得以釐清，例如 SOM 與 FOM 之間的關係與區分，OMT、OMD 與 FED 檔案所儲存之內容有何不同，Multicast 之意義以及如何設定，FED File 是否每個 Federate 都要具備且如何維持其內容之惟一性，FED 檔案內容不一致會產生何種情況等問題。

此次課程中並就戰車專案目前所規劃之 HLA 聯網架構與 DISTI 公司技術人員進行研討，目前戰車案聯網架構設計為：

- 駕駛模擬器不提供 HLA 聯網功能，執行聯網演訓課目時透過射擊模擬器進行聯網。
- 射擊模擬器間可單獨執行 HLA 聯網演訓，或搭配駕駛模擬器執行，相關 HLA 聯網介面程式均於射擊模擬器主計算機執行。
- 各模擬器本身加入/離開聯網演訓及演訓參數設定均由各射擊模擬器教官台執行
- 射擊模擬器與駕駛模擬器之聯網僅提供固定之組合（第 1 套駕駛模擬器對第 1 套射擊模擬器），不允許任意搭配。
- 射擊模擬器與駕駛模擬器間資料傳輸方式不採 HLA，而以自訂資料格式透過 TCP/IP 協定進行。
- 聯網演訓相關聯網課目設定由射擊模擬器主教官台執行設定。
- rtexec 程式於射擊模擬器主控教官台執行， M60A3 與 M48H 執行聯網演訓時，於 M60A3 射擊模擬器主控教官台執行

- 聯網演訓將產生三種 Federation(FedM48H、FedM60A3 及 FedTankSim)，分別作為 M48H 聯網、M60A3 聯網以及兩種車種同時加入演訓之用。
- 射擊模擬器主控教官台本身亦為一 Federate，將扮演 Simulation Management、Data Logger 及戰場顯示三種角色。

DISTI 公司認為此架構規劃為一合理之設計，目前美國陸軍同類型模擬器亦採相同之規劃方式，唯在網路卡之選擇上應採用較高階(如 3Com 公司為 Server 電腦所開發之產品)。

4. 在此次國外差旅中同時順道參訪中佛大 (UCF, University of Central Florida) 之 IST (Institute for Simulation and Training) 實驗室，瞭解 UCF 在訓練模擬器之研發成果：UCF 主要研究方向是在視效圖像處理及虛擬實境方面，並沒有大型實體模擬器，主要偏重於軟體模擬；虛擬實境方面已有實體展示，包括 3D 虛擬場景設計、多機連線之聲音及 3D 虛擬場景資料傳送等技術研發。
5. 美軍在模訓裝備的應用上已建構了非常完整的體系，十分值得我方學習，透過此次與 DISTI 公司人員之研討，獲得美軍在半自動合成武力、地形資料建置以及相關模訓應用之參考網至資料，可作為我方相關發展應用以及技轉之參考。
6. 此外值得一提的是在 Microsoft Windows 以及 X-Window 環境下圖形介面程式的開發上，DISTI 公司所採用的是一套稱為 FLTK(Fast Light Tool Kit)的工具軟體函式庫，此套軟體為可在網路上下載之 Freeware，在功能上較近似於 X-Windows 系統下之 Widget 開發軟體，具備跨平台

(Microsoft Windows 以及 X-Window)之特點，可將應用程式及 GUI 處理程式完全切割開來，而不會有 Windows GUI 程式視窗處理與應用程式相交雜之問題，尤其對於 HLA 程式開發在 Federate Ambassador Callback 函式之處理上可完全化解一般 Windows Programming 無法擷取 RTI 傳回 Federate 訊息之問題，頗值得以後開發 GUI 程式使用。

7. 由 DISTI 講師授課談話得知 DMSO 於本年底之後不再支援/提供 RTI 的發展，因此模擬演訓必須選擇商業版本的 RTI，由於目前市面上所僅有之 RTI 中，瑞典 Pitch 公司所發展的 Pitch RTI 是以 Java 語言發展，因此相關之 HLA 界面軟體亦必須以 Java 發展，由於 Java 在 Performance 上並不適用於即時性模擬，我們可以選擇的商業性 RTI 可能就僅剩 MAK RTI 了，但由於我國陸軍及學界對 MAK RTI 已有一些先入為主之印象，認為該公司產品尚未完全獲得 DMSO 認證，因此並未予以較高之評價，然而此次課程中講師對該 RTI 卻是讚譽有加，就其實務經驗所見認為較 DMSO RTI 更為穩定，因此我們必須密切注意 DMSO 在 RTI 策略上之發展，以作為 RTI 應用選擇上之參考以確保專案時程及品質。

肆、國外工作日程表

91/06/22(六)~91/06/23(日)

台北搭機出發，經由加州洛杉磯轉機至佛羅里達州奧蘭多。

91/06/24(一)

- 1、參訪中佛大 (UCF, University of Central Florida) IST (Institute for Simulation and Training) 實驗室：由該實驗室之副主任兼首席工程師 Ronald C. Hofer 博士親自接待。IST 建立於 1982 年，目前已成為美國國防與商用領域在模式模擬、電腦輔助訓練、虛擬實境等主題居領導地位的研究中心。IST 的研究經費來自政府以及民間企業資助的專題研究，目前共有 80 名全職的研究人員以及 30 名具教職身份的兼任研究人員，並有研究生以及大學部學生跟隨研究人員參與研究專題。在模擬與訓練領域 IST 一心要扮演規範制定者的角色，。

IST 位於中佛羅里達研究園區鄰近中佛羅里達大學，該園區為美國模擬產業重鎮，美國海軍空戰武器研究中心模擬系統部門 (Naval Air Warfare Center Trainig System Division)、美國陸軍模擬訓練司令部 (US Army Simulation, Trainig and Instrumentation Command, STRICOM)、美國空軍模式模擬局 (Air Force Agency for Modeling and Simulation) 等均位於 IST 週邊。IST、UCF 以及 STRICOM 甚至共用研究空間與設備可見其與陸軍關係之密切。

IST 在 DIS 之發展上扮演及其關鍵之角色，並曾參與 HLA 最初之實驗性測試，成功地將 M1 坦克模擬器整合進該次測試中(這是該次測試中唯一的一型由人操控(Human-in-the-Loop)的模擬器，其後在持續接受 STRICOM 的研究經費挹注下，IST 完成 HLA 閘道(Gateway)程式開發並進行商業化轉移給 MAK 科技公司以支援 STRICOM 的 HLA 整合需求。

IST 目前主要之研究領域包括

1. 多層次互動式聯合演訓：發展建構式與虛擬式模擬之連結界面，將建構式模擬之模擬單位解構為虛擬式模擬中之元件實體，或其反向之操作。
2. 高階模擬架構基礎研究
3. 網路連線研究：探討長距離網路連線中傳輸延遲及時間管理問題。
4. 電腦合成武力：包含路徑規劃、視野運算、偵蒐規劃、互動/協同行為合理性、威脅分析以及地形因素運算等研究。
5. 多人互動式虛擬實境：
 - (1)虛擬實境：個別之虛擬實境具有頭盔顯示 3D

虛擬實境場景，身體軀幹、手及腳感測器模擬人的肢體動作。

- (2)分散式虛擬實境之小組訓練：個別之虛擬實境可單獨模擬訓練，多台分散式虛擬實境也可經由網路連線，而達到小組共同訓練之功能，彼此可經由 3D 虛擬實境場景看到對方之動作。

6. 地形資料庫：

- (1) VESS (Virtual Environment Software Sandbox)：提供高階圖像資料庫，使得程式設計師不需要細部設計到各物件之動態反應，只需簡易的高階指令則可完成所需之動作；這對動態分散式網路連線模擬系統之設計有很大之助益。同時由於此系統為高階語言，所以可以很容易攜帶到各種作業系統中（如 SGI 之 IRIX、Linux 及 Windows 系統等）。
- (2) Standard Simulator Database Interchange Format 之制定與程式開發界面。

7. 其他模擬領域：包括醫學、公共安全、先進遠距教學等

2、問題研討：

- (1) 在戰車模擬器中，當既有之視效資料庫環境改變時（如原有橋樑斷裂無法通行、建築物倒塌影響道路行進或是炸彈爆炸之彈坑影響道路行進等），該如何處置：建議考量將視效資料分為背景資料庫及重要地標（如橋樑或主要道路）兩種（或更多種類），背景資料庫沒有任何屬性固定顯示，而重要地標則加入屬性參數以控制其是否顯示。

91/06/25(二)

1、實務訓練課程：

- (1). 課程簡介
- (2). DISTI 公司簡介及講師簡介
- (3). 分散互動式模擬介紹
 - i. 名詞解釋
 - ii. 分散互動式模擬發展歷程與演進
 - iii. SIMNET 系統架構及限制
 - iv. ALSP 系統架構
 - v. DIS 系統架構及限制
- (4). HLA 的誕生
- (5). HLA 系統架構介紹
 - i. 核心設計概念：C++ Base、CORBA Like Core、Everything must be Transported via RTI、Rumbaugh methodology might inavailable.
 - ii. HLA 名詞解釋
 - iii. HLA 程式發展要領
 - iv. HLA 組成項目

v. HLA 與 DIS 之比較

(6). 網路技術

- i. HLA 網路架構概述
- ii. Ethernet/ATM
- iii. Broadcasting/Multicasting
- iv. 網路設備
- v. 網路需求分析

(7). 分散式物件系統

- i. 分散式物件系統起源
- ii. CORBA
- iii. DCOM
- iv. Java RMI
- v. 分散式物件架構之比較
- vi. CORBA 與 HLA 之關係

(8). 物件模型樣板(Object Model Template)

- i. OMT 之目的與採用 OMT 之理由：OMT 之主要目的為提供模擬物件一個共通的結構化表現格式。並可方便使用者查閱各類定義、特性、數值。其所描述的是物件、屬性、互動類別(Interaction Class)以及相關細部定義。
- ii. SOM：描述在 Federation 中之個別 Federate 可以外在提供給整個演習的系統功能。因此在角色上 SOM 是一些主觀的資料，類似於 ICD。

- iii. FOM:提供一個共通且標準化之格式來描述在一個 Federation 中各 Federate 之間所有公開的資料包括 objects 、 attributes 、 interactions 。
- iv. OMT 的內容：Object Model Identification Table、Object Class Structure Table 、 Object Interaction Table 、 Attribute/Parameter Table 、 Dimension Table
- v. HLA 對物件資料的要求
- vi. Publish/Subscribe
- vii. Interaction Class
- viii. Routing Space
- ix. Lexicon Table

91/06/26(三)

1、實務訓練課程：

- (1)安裝 RTI (Run Time Infrastructure)：電腦已安裝完成 RTI；此一實務訓練課程主要在說明 RTI 系統安裝完成後，需注意之設定項目，其中包括：RTI 安裝路徑、RTI 及作業系統版本、發展測試環境之路徑宣告(作業系統宣告及 Visual Studio 發展環境宣告)等。
- (2)Class 設定：使用 OMDT (Object Model Development Tool)在 FOM(Federation Object Model)之 Class Structure Table 中增/減 Class，設定其資料為 Publish 或 Subscribe 並

設定其 Attributes。

- (3)Complex Datatypes 設定：使用 OMDT (Object Model Development Tool) 在 Complex Datatypes 中增/減 Fields，並設定其 Properties (包含初始值及資料格式等)。
- (4)Attribute Table 設定：使用 OMDT(Object Model Development Tool) 在 Attribute Table 中增/減 Attributes，並設定其 Properties (包含初始值及資料格式等)，以提供 Class 設定時選用。
- (5)Interaction Table 設定：使用 OMDT (Object Model Development Tool) 在 Interaction Table 中增/減 Interaction，並設定其 Properties 之 Parameters (包含初始值及資料格式等)。

91/06/27(四)

1、實務訓練課程：

- (1)Create and Join Federation：程式一開始先 check Federation 是否已存在，如果未存在 (第一個加入者) 則 Create Federation；否則以 Join Federation 方式執行。
- (2)Publish and Subscribe：RTI 對於傳送/接收資料均是以 Message 方式處理，以 subscribeInteractionClass 及 publishInteractionClass 之 RTI 指令接收/傳送資料。
- (3)Publish and Subscribe：RTI 對於傳送/接收資料均是以 Message 方式處理，以 subscribeInteractionClass 及 publishInteractionClass 之 RTI 指令取得接收/傳送資料之控制權。

- (4)Send Interactions：以 sendInteraction 之 RTI 指令將資料傳送出去。
- (5)Receive Interactions：以 receiveInteraction 之 RTI 指令接收資料。
- (6)DDM (Data Distribution Management)：加入設定 Enable 或 Disable DDM 之功能；如果 Enable DDM 功能後，傳送端及接收端之 DDM 頻率範圍必需有交集資料才可正常傳送，否則無法正常傳送。
- (7)程式測試驗證：可正常持續開啟視窗陸續加入 Federation；Disable DDM 功能，可任意傳送/接收資料（包含文/數字資料及音效檔資料）；Enable DDM 功能後，調整傳送端及接收端之 DDM 頻率範圍完全沒有交集，資料無法正常傳送；調整傳送端及接收端之 DDM 頻率範圍有交集，資料可正常傳送。

91/06/28(五)

1、實務訓練課程一：

- (1)建立 FED File：包括 Federation Name、FEDVersion、各層級之 Object Classes、各 Object Classes 之 Attributes、Interactions 及其 parameters 等宣告設定。
- (2)Create and Join Federation：程式一開始先 check Federation 是否已存在，如果未存在（第一個加入者）則 Create Federation；否則以 Join Federation 方式執行。
- (3)Disabling Time Management：當系統在即時環境下執行時，Join Federation 時需 Turn Off Time Management 而取得目前 Federation

Time。

- (4)Get Object and Attribute Handles：以 getObjectClassHandle 取得 Object Class 之控制權，以 getAttributeHandle 取得 Attribute 之控制權。
- (5)Publish and Subscribe Objects and Attributes：傳送/接收 Object Class 及 Attributes。
- (6)Get Interaction and Parameter Handles：以 getInteractionClassHandle 取得 Interaction 之控制權，以 getParameterHandle 取得 Parameter 之控制權。
- (7)Publish and Subscribe Interactions and Parameters：傳送/接收 Interaction Class 及 Parameters。
- (8>Create Local Tank：產生一個 Tank Object，初始位置由亂數產生，狀態為 alive。
- (9)Update Tank Attribute：可動態變更 Tank Attribute，包括 Tank 位置、砲塔仰角、得分及狀態等 Attribute。
- (10)Inserting Objects：先以 FindObject 確認 Objects 是否已存在，如果不存在則再以 AddObject 新增 Objects。
- (11)Deleting Objects：先以 FindObject 確認 Objects 是否已存在，如果存在則再以 DeleteObject 刪除 Objects。
- (12)Fire the Bullet：先確認 Tank 狀態為 alive 才可發射 Bullet；再檢查是否已發射 Bullet 且還在飛行中（程式設定一次只能發射一枚 Bullet）。
- (13)Move the Bullet：Bullet 在飛行中則依據 Tank

位置及砲塔仰角持續計算 Bullet 新位置。

(14)Detonate Interaction：當 Bullet 擊中目標或落地 (Hit/Miss) 時送出 Detonate Interaction；如果擊中目標則告知對方修改 Tank 狀態及送出音效。

(15)程式測試驗證：可正常持續開啟視窗陸續加入 Federation；本機之 Tank 為紅色，其餘均為藍色；可自由移動本機 Tank 位置及調整砲塔仰角；可按發射鍵以發射 Bullet 並可看到 Bullet 飛行位置；如果被命中之 Tank 則改成灰色並且無法移動及發射 Bullet。

2、實務訓練課程二：

(1)Ownership 控制：可正常持續開啟視窗陸續加入 Federation；每一視窗程式均可自行加入一些控制錶頭；本機開啟之控制錶頭為紅色，其餘為藍色；控制錶頭中有紅色及綠色選鈕表示控制權。

(2)錶頭控制：當視窗內錶頭之控制權為綠色時表示取得控制權，可調整及設定錶頭；當視窗內錶頭之控制權為紅色時表示未取得控制權，則無法調整及設定錶頭。

3、戰車模擬器聯網設計研討：

(1)戰車模擬器分散於兩座建築物中，網路架構應如何設計：個別建築物網路連線到統一的 Router，再以光纖將兩座建築物之網路連接，應可滿足聯網傳輸之需求。

(2)戰車模擬器中有分為 8 套射擊模擬器及 8 套駕駛模擬器，射擊模擬器與駕駛模擬器平時可分開訓練，但又必需能組合成一組戰車模擬器共同

演訓：在 8 套射擊模擬器與 8 套駕駛模擬器中各架設一部主控電腦，負責控管個別模擬器之加入或退出 Federation；此一主控電腦除了可以監看本身 8 套模擬器外，也可監看另外 8 套模擬器之狀態，當需要射擊模擬器與駕駛模擬器共同聯網演訓時，可做為配對設定之功能。

91/06/29(六)～91/06/30(日)

由佛羅里達州奧蘭多搭機經由加州洛杉磯轉機回到台北。

伍、社交活動

與 UCF 之 IST 實驗室教授廣泛交換模擬器設計方面之心得，並留下相關領域負責人之資料，可做為後續網路討論之資源。

由於課程極為緊湊，所以並無與其他公司之與課人員做深入討論，僅有 DISTI 公司製作之所有參與課程人員基本資料。

利用此次課程並與 DISTI 公司人員研討有關戰車模擬器聯網設計架構、美軍模擬應用現況等問題。由於該公司再模擬聯網相關技術具有十分堅實之經驗與基礎，因此亦曾提出要求，希望其能持續提供適當之諮詢服務已獲得同意。

柒、建議事項

1. 以本組近年來在模擬器製造領域所累積之寶貴經驗，實應超越“做”的層級而應提昇至“用”的層級，尤其在國軍各級模訓裝備採購建置已逐步成型的階段，如何將各型模訓裝備加以整合應用、發揮進一步的加成果效，實應為今後本組發展之重點。

附件：課程表

1. Introduction to Distributed Simulation

- Networked simulation environments
- Real-time & discrete event simulation
- Distributed simulation models

2. Introduction to the High Level Architecture

- Key ideas and goals
- Basic terminology
- Related simulation design concepts

3. Network Architecture and Media Overview

- UDP/IP, TCP/IP and IP Multicast
- HLA imposed network considerations
- Network architecture strategies

4. Legacy Simulation Architectures (DIS & ALSP)

- Entity state / Event communication
- Dead reckoning and coordinate systems
- Simulation management
- How is HLA different from DIS/ALSP

5. Distributed Object Processing Environments

- Distributed Object Model concepts
- Examples of Distributed Object Models and services
- Overview of 'The ACE ORB' (TAO) - CORBA

6. HLA Object Models

- The object model template
- Building object models
- Object model development tools
- HLA rules for object models

7. HLA Federates

- What is a Federate?
- Simulation Object Models (SOM)

- Sample HLA Federates and case studies
- Submitting a SOM for review
- **LAB - Building a Simulation Object Model**

8. HLA Federations

- What is a Federation?
- How Does a Federation Relate to Your Project?
- Federation Object Models (FOM)
- Federation Planning and Management Software
- **LAB - Constructing FOM and RTI configuration files**

9. The Run-time Infrastructure

- Functional View of an RTI
- What RTIs are Available and How to Choose?
- How Difficult is it to Change RTIs?
- Federation Management Services
- **LAB - Establishing a Federation Execution**
- Declaration Management Services
- Data Distribution Management
- Object Management Services
- **LAB - Sending and Receiving Objects and Attributes**
- **LAB - Sending and Receiving Interactions**
- Time Management Services
- **LAB - Exercises in Cooperative Time**
- Ownership Management Services
- **LAB - Transferring Object/Attribute Ownership**
- **LAB - Distributing and Object Across Multiple Simulators**
- The Management Object Model
- Mapping Federates to New FOM Representations
- Federate Case Studies

10. Planning your DIS to HLA Migration

- The Real-time Platform Reference FOM (RPR-FOM)
- Examples and case studies

- LAB - Demonstration of HLA Tools

11. HLA Compliance Requirements and Testing

12. HLA Data Sources