The Royal Australian Air Force Air Defence Ground Environment Simulator

Lucien Zalcman, Senior Research Scientist (Contractor)
Defence Science and Technology Organisation
Australian Department of Defence
Melbourne, AUSTRALIA
lucien.zalcman@dsto.defence.gov.au

ABSTRACT

Previously Royal Australian Air Force (RAAF) Air Defence Controllers relied on interaction with live F/A-18s to receive most of their realistic, high fidelity training. Access to such live asset training is limited in that it is prohibitively expensive and such assets have very short lifetimes before a major overhaul is required. An Advanced Distributed Simulation, virtual environment, training system known as the Air Defence Ground Environment Simulator (ADGESIM) has been developed and delivered.

A generic, Test and Training Enabling Architecture (TENA) like, composability, toolbox approach, using a mixture of Commercial-Off-The-Shelf (COTS) and customized “thin client” Government-Off-The-Shelf (GOTS) components, has been used to construct ADGESIM. The Distributed Interactive Simulation (DIS) protocol is used to communicate between ADGESIM components however the simulator architecture is such that the Higher Level Architecture (HLA) or TENA “protocols” could also be easily used in the future if required.

All ADGESIM components are developed to run on desktop/portable PCs under the Microsoft Windows XP operating system and are therefore low cost and cost effective to develop and maintain and can be easily deployed. ADGESIM was rapidly developed and fielded from a concept demonstrator to operational use bypassing the normal, system acquisition process. ADGESIM stimulates the real, same systems used by the Air Defence Controllers thus eliminating most traditional, trainer concurrency problems.

This paper describes and discusses some of the innovations, technologies, toolbox components, simulator architecture, development processes and philosophies used and the lessons learned developing ADGESIM.

ABOUT THE AUTHOR

Lucien Zalcman is a Senior Research Scientist in the Australian Defence Department’s Defence Science and Technology Organisation (DSTO) specializing in Advanced Distributed Simulation. Lucien has a B.Sc. (Hons) and a Ph.D. in Experimental Physics from Melbourne University and a Graduate Diploma in Computing Studies from the Royal Melbourne Institute of Technology. In his 21 years at DSTO Lucien has been involved in the evaluation and development of new training technologies and has provided specialized technical support to warfighter training programs such as the Royal Australian Air Force (RAAF) Airborne Early Warning and Control (AEW&C) and F/A-18 aircraft cockpit training systems and the Royal Australian Navy’s FFG Frigate’s Upgraded On-Board-Training-Systems. Lucien has been a key architect in the design and development of the RAAF ADGESIM trainer described in this paper. Lucien resigned from DSTO in May, 2005 and is now employed as a contractor at DSTO.

The Royal Australian Air Force Air Defence Ground Environment Simulator

Lucien Zalcman, Senior Research Scientist (Contractor)
Defence Science and Technology Organisation
Australian Department of Defence
Melbourne, AUSTRALIA
lucien.zalcman@dsto.defence.gov.au

INTRODUCTION

In some situations the Australian Defence Force (ADF) has no option but to use live assets to support training. The use of live assets can be expensive and can lead to increased fatigue-life accrual with the resultant increase in maintenance and support costs. Reduced operating budgets have impacted on both asset availability and frequency and scale of joint and combined exercises. The introduction of new systems, such as the Jindalee Operational Radar Network (JORN), and the Airborne Early Warning and Control (AEW&C) aircraft will require additional personnel and operations. Reliance on the availability of fighter aircraft or the use of existing, ageing and outdated simulators for the conduct of fighter control and battle management training for RAAF Surveillance Control Group (SCG) is no longer viable and enterprise level exercises that provide live training are virtually non-existent.

To address this problem the ADF is adopting advanced distributed technologies, such as DIS [IEEE 1278.1A 1,2] and HLA [HLA (2000)], to enhance training capabilities, increase training opportunities and reduce costs.

THE DSTO ADVANCED DISTRIBUTED SIMULATION LABORATORY

In 1998 the need for study into advanced distributed simulation technology to support training within the SCG was recognized by the RAAF and led to the establishment of the Australian Defence Science and Technology Organisation (DSTO) Advanced Distributed Simulation Laboratory (ADSL) at DSTO in Fishermans Bend, Melbourne funded by the RAAF's Virtual Air Environment (VAE) project.

The approach currently used by most ADF simulation projects has resulted in one-off, highly specialized, monolithic, large software applications. These large software projects are high risk, usually end up being delivered over time, over budget, do not provide the full functionality originally specified and are expensive to maintain. The ADSL promotes the use of modular, cost-effective, COTS, distributed simulation applications. However it is unlikely that an operational simulator can be completely created from available COTS applications and *customised thin-client* applications have been developed where no commercial equivalents exist. Because the in-house development focus is on these customised thin-client applications which, when compared to the COTS applications used, only require a relatively small amount of code to be developed, a simulator can be delivered cost-effectively, quickly, and at lower risk when compared to highly specialised, monolithic, large one-off software ADF simulation applications mentioned above.

A low-cost and cost-effective hardware and software development environment based on COTS (corner store) PC hardware, the Microsoft Windows XP operating system, the latest Microsoft Visual Studio C++ compiler and its integrated development environment is mandated in the ADSL for the development of software.

THE RAAF AIR DEFENCE GROUND ENVIRONMENT SIMULATOR (ADGESIM)

ADGESIM (shown in Figure 1) is comprised of eight, customised thin-client, applications interoperating with five COTS products (Solipsys [Solipsys] back and front ends, MÄK [Mak] Technologies VR-Forces back and front ends and VR-Link toolkit). The ADSL developed, thin client applications are the ADGESIM Orthographic Network Input/Output Node (ONION) application, the Pilot Interface, the DIS Radio / Intercom Communications system, the Airline Scheduler, the After Action Review (AAR) Data Logger, the World Viewer, the Chat application, and the DIS Protocol Data Unit (PDU) Data Viewer.

Each ADGESIM application is a standalone, distributed simulation, application that has no requirement, or knowledge, of the presence of any other ADGESIM application. All ADGESIM applications can be used in other virtual, simulation

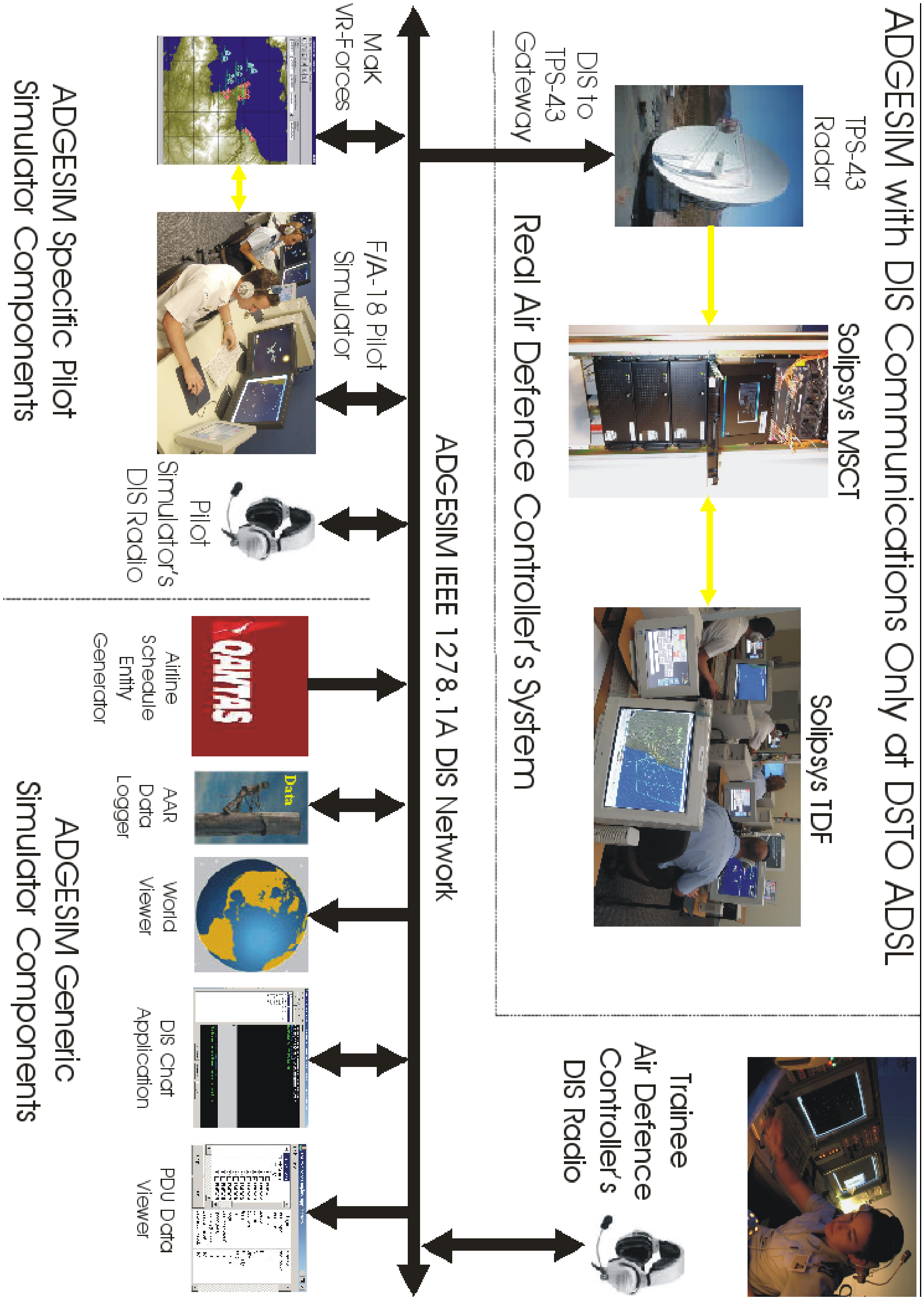

Figure 1. Royal Australian Air Force Air Defence Ground Environment Simulator (ADGESIM)

environments and are reusable, distributed simulation, applications. These ADGESIM applications are described in detail below.

The ADGESIM ONION DIS to Westinghouse Radar TPS-43 Gateway Application

A Sensor ONION is a gateway between the ADGESIM IEEE 1278.1A, DIS compliant, synthetic virtual environment and the real operational system used by RAAF Air Defence Controllers. DIS provides a network environment within which participants share information using protocols and message formats that are generally not used by operational systems. DIS also provides a ‘perfect’ picture of scenario activity. In the real world, operational systems are constrained by sensor capabilities and the performance of other systems to which they are connected and do not have a ‘perfect’ view. ONIONs are software applications that emulate (ie simulate) the performance of real world sensors and systems. ONIONs listen to the DIS environment, apply a sensor detection model to determine whether a simulated event should be detected, and then format messages about that event that resemble the output streams of real-world sensors. These streams of simulated sensor data are received and processed by operational systems in the same way as ‘live’ sensor data, with the exception that simulated data will have an indicator flag that identifies it as synthetic (ie simulated). The ADGESIM Sensor ONIONs also filter out any inappropriate DIS entity information eg dismounted infantry.

A number of sensor ONIONs, with differing performance attributes, can be run simultaneously to replicate a multi-radar or multi-sensor operational environment to stimulate operational systems. Generic sensor ONIONs have been produced for 2D and 3D radars. Rotation rates, detection probabilities, clutter performance, IFF range, primary range, and various other attributes are maintained in XML-based configuration files on a sensor type-by-type basis. Each instance of a sensor type can also have magnetic variation, location, altitude and identity label assigned.

An ONION Graphical User Interface (GUI) has been developed for management of sensor operations. Sensor performance and attributes can be managed in real-time to enable and disable functions such as primary and secondary radar operation, clutter profile and detection probability. Future development will manage Electronic Warfare operations. Current radar data output is Westinghouse TPS-43 Digital Target Extractor (DTE) format. However, the modular nature of the application allows for additional data output formats to be added simply.

ONIONs for ESM, JORN, AEW&C, ship-based radar, and space-based sensors (Infrared and Elint) are planned in the near term. Performance of the ONION application is very good with ten simultaneous sensor instances on a single 2GHz Pentium 4 PC producing an approximate processor load of thirty percent with three hundred radar targets (many being detected by several sensors simultaneously). Performance may be impacted in the future as more complex detection and electronic warfare functions are added.

Because the ONION application stimulates the same operational software used by Air Defence Controllers concurrency problems (ie the difference between the operational and trainer systems) are kept to a minimum. The ADGESIM trainer applications can be used to stimulate any sensor system that supports the Westinghouse TPS-43 DTE format via the ONION gateway application.

The ADGESIM DIS Radio / Intercom System

The ADGESIM DIS Radio/Intercom Communications system, known as DISVOX, is mainly used for demonstration purposes in the ADSL and is not used operationally. A COTS, DIS/HLA, radio and intercom software toolkit [Emdee] is used by DISVOX to produce DIS Transmitter and Signal PDUs.

Although DIS version 1278.1A does have an Intercom PDU this functionality is implemented in DISVOX by assigning radio channels with frequencies less than 1000Hz as intercoms for both DIS 1278.1 and 1278.1A systems. This is a de-facto standard method of implementing intercoms in DIS simulators.

Bmp and xml files determine the appearance and behaviour of the DISVOX GUI. DISVOX is “skinnable” and these (bmp/xml configuration file) skins are designed to duplicate the “look and feel” of real radio/intercom systems. A single DISVOX application can support up to a maximum of 18 radios and 18 intercoms on the one skin.

Many of the capabilities available in the DIS Transmitter and Signal PDUs are supported through the DISVOX configuration menus. DISVOX is a highly flexible and configurable application.

DISVOX has been demonstrated in the ADSL to be interoperable with the General Dynamics ModIOS

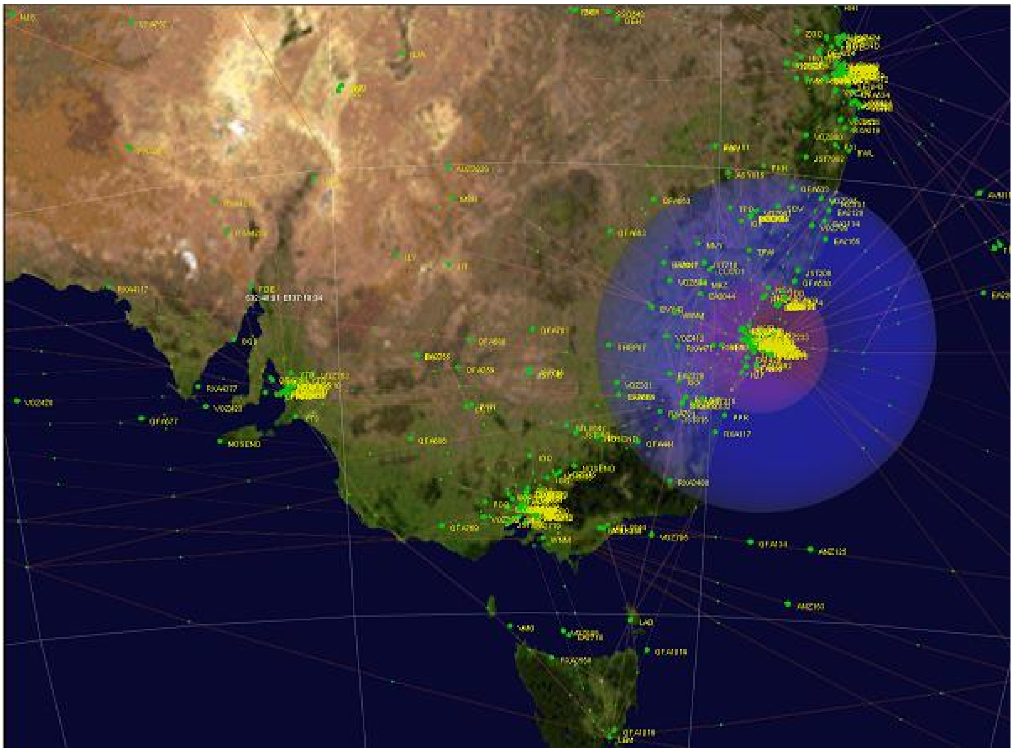

Figure 2. The ADGESIM Airline Scheduler

ToolSuite, ASTi and US Navy Battle Fleet Tactical Trainer (BFTT) DIS radio/intercom systems.

The ADGESIM Airline Scheduler Application

The ADGESIM Airline Scheduler provides a way to populate the synthetic environment with large numbers of “background” entities that are not dynamically controllable by Pilot Drivers or VR-Forces itself, and which conform to predetermined behaviours such as airline schedules or simple scripts. Whereas the Pilot Driver Interface application communicates with the VR-Forces back-end, which then produces DIS PDUs, the Scheduler application makes use of a COTS library of DIS functions, called the *VR-Link* Toolkit [Mak], to generate the required DIS PDUs.

The Airline Scheduler performs calculations to direct entity takeoff/start, navigation, height changes etc. Multiple scripts can be run simultaneously and many entities may be airborne in the Scheduler at any one time, anywhere on the planet. The Scheduler is being updated to manage surface and space traffic. This will be important for wide area surveillance training where simulated space and Over-The-Horizon-Radar sensors are required. With around one thousand entities airborne simultaneously the processor load on a 2GHz Pentium 4 PC remained steady at about two percent, indicating scalability well beyond ten thousand simultaneous entities. A typical graphical screen of the Airline Scheduler application is shown in Figure 2.

Data on air routes, waypoints, airfields, aircraft attributes, DIS enumerations, and flight schedules are held in configuration files compiled from standard data sources. The entire flight schedule for QANTAS [Qantas] will be added, and over time, schedule libraries will be created for other world airlines and various “interesting” surveillance scenarios.

The ADGESIM After Action Review Data Logger

In the ADSL COTS DIS Data Loggers have been used for many years. However in practice COTS Data Loggers usually do not provide the full functionality required during a real training exercise.

COTS DIS Data Loggers usually record and playback using a single TCP/IP port. In many exercises attended by DSTO scientists the DIS PDU stream was distributed over multiple ports to aid Wide Area Network (WAN) management. The ADGESIM DIS Data Logger application (see Figure 3) can record and filter PDU data over at least four different ports.

Simple DIS exercise performance metrics can be viewed including number of PDUs versus time, bandwidth versus time, last PDU type detected, protocol version of last DIS PDU detected, number of PDUs detected, etc. These types of metrics are useful in the setup stages of a DIS exercise to spot obvious incompatibility and interoperability problems.

Data can be played back starting at any time within the data file. The direction of playback (forward or reverse) and the playback speed can be set by the user.

Figure 3. The ADGESIM DIS Data Logger

Recorded PDU data is stored in an indexed binary file. To replay data the next sequential record (ie DIS PDU) is read from the recorded binary file and the PDU is replayed onto the simulation network according to the timing data stored in the retrieved data. This approach [Graebener et al., 2004] is more efficient than a pure Relational DataBase Management System (RDBMS) approach and, so far, has been able to replay all test data used so far, within an error of 1 millisecond, for PDU rates of up to several thousand PDUs per second. At the same time the binary PDU data is recorded, specific, critical data can also be optionally recorded using COTS RDBMS software. However because only known critical (ie not all) PDU data is recorded the average RDBMS PDU record time is now acceptable.

The ADGESIM DIS Data Logger has been designed to be the starting component of more capable (and demanding) applications such as an AAR or a DIS Interface Compliance Testing application.

In COTS Data Loggers only when data logging has stopped and the data has been saved, can logged data be processed and further analysed. The ADGESIM DIS Data Logger has been designed to allow sections of logged data to be concurrently extracted whilst data is still being logged from the network.

DSTO has been involved in the testing of DIS interfaces on military platforms where access to the system under test is severely limited. Exactly which interface compliance tests are to be carried out may depend on the results of previous tests. The concurrent access capability of the ADGESIM DIS Data Logger allows testing, analysis and logging to be carried out in a synchronous, concurrent fashion whilst network data is still being recorded, thus saving time when limited testing time is available. Similarly a large scale, WAN, distributed simulation, training exercise may take place over a period of several hours to several days. Traditionally, once such an exercise was completed the data would be stored, then processed and analysed, and then presented to the exercise team participants for further AAR processing. How long it takes to prepare data for AAR processing will partly depend on the complexity off, and the amount of data logged. It is usually considered essential to carry out the AAR process as soon as possible. Because of the concurrent access design of the ADGESIM Logger data files an AAR team could carry out preliminary processing of the data whilst the training exercise is in progress thus reducing the time required before the AAR process can start at the completion of the exercise.

The ADGESIM Pilot Interface Application

The Pilot Interface Application allows pilot simulator operators to create and fly multiple, simulated aircraft entities generated and controlled in the VR-Forces COTS back-end. Functionality provides for situation awareness and has features that allow the operator to respond in a way that approximates the behaviour of real-world pilots and formations of aircraft.

Aircraft position, performance and system information is presented in a customised GUI (see Figure 4) and is derived from information broadcast by VR-Forces. Each Pilot Interface can control up to twelve such back-end entities at any one time (although entities can be released and captured for control as required) and works by calling VR-Forces back-end functions over the network to control entity and environmental behavior. Several custom VR-Forces controllers were developed as part of the Pilot Interface client application to achieve more realistic aircraft control.

Multiple Pilot Interfaces can communicate with a single VR-Forces back-end or with several VR-Forces back-ends on the network to make the ADGESIM Pilot Interface application highly scalable.

Pilot Interfaces also communicate to provide electronic warfare indicators such as radar "spikes", radar mode changes, and weapon launch indications. In later versions, pseudo tactical data link information will also be incorporated to allow cooperative engagement of targets. Through VR-Forces (using DIS Interrogate Friend or Foe (IFF) PDUs) the Pilot Interface controls Modes 1, 2, 3/A, 4, and Mode S for all entities and manages weapons employment. The interface allows

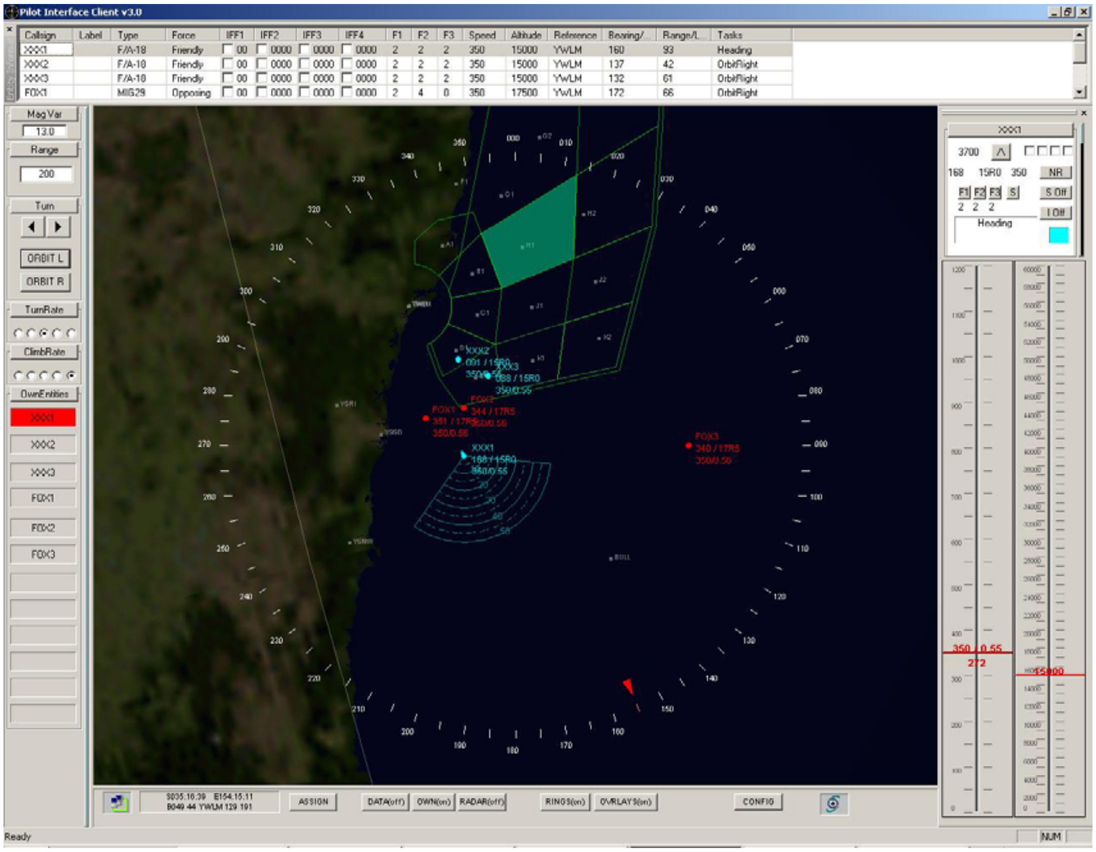

Figure 4. The ADGESIM Pilot Interface

aircraft to be grouped and flown as formations and also provides automated manoeuvres such as heading snaps, postholes, navigate to a point, and follow or patrol a route that reduces operator workload.

The ADGESIM Pilot Driver Interface is used operationally at RAAF Williamtown and operator response to the interface has been very positive with confidence and proficiency in use of all functionality achieved in around two hours of supervised use.

The ADGESIM World Viewer Application

The ADGESIM World Viewer application is a DIS compliant, psuedo-3D viewer application based on a high resolution, texture of the world. The World Viewer provides a symbolic representation of any DIS entity participating in an exercise along with airports, waypoints, routes and overlays on an orthographic projection of the world.

Datablocks of user defined, text information including callsign, type, heading, speed, IFF, lat/long etc. can be displayed for each entity. These datablocks can be permanently displayed or be displayed only when the mouse is positioned over the DIS entity symbol.

There are various display options - radar cones, primary/secondary emitter sensor volumes, entity snap-to-ground, ground-altitude entity poles, entity history dots, etc. Multiple symbology sets are supported including the MIL-STD-2525B symbology set.

The user can zoom in or out and rotate the world with the center of the world always located at the center of the screen. A pan mode is also available. The user's viewpoint may be fixed in space or locked on to any particular entity. A moving map capability is being implemented whereby the world texture is merged or replaced with a geospecific map.

Although the World Viewer is a separate application the application's functionality is sourced from a WorldLib, common code, library that is used by other ADGESIM applications. The Pilot Interface and Airline Scheduler ADGESIM applications already use this WorldLib, common code, library to provide 3D capabilities within these applications. Probably all ADGESIM applications could, in one form or another, make use of this psuedo-3D Viewer capability.

The ADGESIM Chat Application

The ADGESIM Chat application is used for chatting, via text messages, between exercise participants over a distributed simulation, exercise network.

Internet Chat type programs are generally client-server, TCP/IP applications and require access to a server that normally resides (transparently) on the Internet. Real training exercises are generally carried out on secret networks and access to the (external) Internet (ie the server application) may not be available. A client application may involve a simple, user friendly, installation but once a network server application is required the installation of the combined client-server application (set) does require the availability of computer networking, knowledgeable people.

The ADGESIM Chat application is a stand-alone, peer-to-peer application that does not require any server or external network component to operate correctly. The application is easy to install and once started, automatically broadcasts its presence on the network, to enable other ADGESIM Chat applications to automatically setup their required chat groups.

In a DIS exercise many applications may be executing over the simulation network at the same time. It is desirable to synchronously record all data to allow this data to be replayed, in correct time-step, during an AAR. Therefore chat traffic could be replayed in correct time sequence with other DIS network traffic to allow an appropriate AAR to take place. Even though available freeware/shareware Chat applications may have a record and replay capability there may not be any (easy) way to synchronise the chat data, both for record and replay, with the distributed simulation network traffic. The Chat application's DIS Comment PDUs can be recorded and replayed, synchronously, along with the other exercise DIS PDUs at AAR time using a COTS DIS data logger.

The ADGESIM DIS Chat Application can operate in one of three modes:

- Instructor mode – can participate in an exercise, does not belong to any subgroup, but can view all chat application messages that are sent over the network on the chosen port;
- Participant mode – the normal operating mode that interacts with messages and users belonging to user chosen groups; and the
- After Action Review (AAR) mode - does not participate in an exercise but exactly monitors or imitates a user that is currently participating, or has previously participated, in an exercise.

The AAR mode is not normally available in COTS applications. This mode is designed to help in the AAR process and has been demonstrated in the Chat application with the objective that it will eventually be included in all other ADGESIM applications.

The ADGESIM PDU Data Viewer Application

The ADGESIM PDU Data Viewer application displays user defined data describing selected entity (platforms and weapons) activity within the virtual battle-space environment. The user interface comprises a dynamic tree-structured list of entities and a dynamically sized area for viewing chosen entity information.

Every entity detected in the virtual battle-space can be added and sorted into the tree-structured entity list according to its identity and a dynamic structure chosen by the user. For example, the list could be split into categories of Force, then Domain, then Country. In this case the root of the tree would consist of the branches 'Friendly', 'Enemy' and 'Neutral'. Under each of these branches, branches named 'Air', 'Land' and 'Sea' would exist and under each of these Domain branches the participating Country branches would exist. If an Australian, friendly aircraft was detected in the virtual environment, it would belong under Friendly, Air, Australia.

The entity tree structure is initially determined from a configuration file read at run-time but can also be modified dynamically via menu options. A user may specify entity Force, Domain and Country in any order and each entity is listed by callsign and child branches that contain entity information including Entity Type, Enumeration, Entity ID, Site ID, Host ID, Force, Domain, Country and Category. Alternatively no category can be specified in which case the tree structure would disappear altogether and a list of entities would appear in alphabetic order sorted according to callsign.

Checking the checkbox next to each callsign indicates entity information is to be displayed for that entity in the main data display area. Exactly what entity information is displayed is initially determined at startup from a configuration file but this can also be dynamically modified via menus. PDU type, call-sign and Entity Type information must be displayed as a minimum. If a chosen entity fires a weapon or is targeted with a weapon additional Fire/Detonation information is displayed. Friendly, enemy and neutral force information are displayed using user defined colours and Fire/Detonation information is displayed using different shades of the required force colours. The PDU Data Viewer is designed to focus the observer's attention on relevant information from important entities in an exercise.

A MAJORITY COTS SOFTWARE APPROACH

The ADGESIM was designed to use as much COTS software as possible. A simple guess of the total number of lines of code in the ADGESIM software, based on the author's experience and the complexity of the COTS applications, would be between one and two million lines of code.

The customized, thin-client, ADGESIM applications developed at DSTO comprise less than 200,000 lines of code. Therefore, for the ADGESIM, the use of COTS potentially reduces the development and maintenance of software, documentation and training material by up to approximately an order of magnitude!

COMMODITY ITEM PC HARDWARE

All ADGESIM applications run on common, highly reliable, "high-end", low-cost, PC hardware. No computer hardware maintenance is required and if any hardware fails the faulty hardware is simply replaced.

PC SOFTWARE DEVELOPMENT ENVIRONMENT

A simple PC environment was required for software development of ADGESIM applications. Only the Microsoft Windows and Linux operating systems were considered. At the time of initial development Microsoft Windows was (and still is) the dominant desktop operating system. Commercial development and maintenance of the Microsoft Windows XP operating system, and unavailability of the operating system source code, was considered lower risk, more productive and more appropriate than the open-source Linux model. However Linux may be a better choice where applications are being ported from a Unix environment rather than developed from new.

The latest version, Microsoft Visual Studio C++ compiler is used to develop ADGESIM applications. Several of the ADGESIM thin-client applications are multithreaded and are thus able to take advantage of Hyper-Threading technology now and will be able to take advantage of multi-core processor technology when this technology becomes available.

THE ADGESIM, PEER-TO-PEER, DISTRIBUTED SIMULATION, STIMULATED ARCHITECTURE

ADGESIM was specifically designed to be a distributed simulation application executing over a network of commodity-item PCs.

Generally each ADGESIM application runs on its own (networked) PC. Multiple applications can run on the same PC however, usually, the limiting factors are the amount of screen "real-estate" available and the graphics and processor requirements of the particular applications. PCs with sufficiently powerful processor (at least 2 GHz) and graphics capability (nVidia 4600 or ATI 9600) are required. However with the endless advance of PC technology these requirements will shortly no longer be considered "high-end".

This peer-to-peer, distributed simulation, "black-box" approach enables each application to remain manageable and, to date, each thin-client ADGESIM application has only required development by only one software engineer.

From Figure 1 the ADGESIM architecture is comprised of three main components. These are

- The real system – the Westinghouse TPS-43 radar system (replaced by the equivalent ONION component in the ADGESIM), the Solipsys Multi-Source, Correlator, Tracker (MSCT) back-end and the Solipsys Tactical Display Framework (TDF) front-end. The use of this stimulated, real system component approach eliminates many trainer concurrency problems that occur in emulated systems:
- The ADGESIM specific, Pilot Simulator components - these components are designed to provide the main simulation functionality required by the ADGESIM - the wingman, the red force, etc. at the fidelity required: and
- The ADGESIM generic simulator components that can be reused by any simulator;

Many (if not all) real-time training simulation systems may be able to be constructed using this approach and the ADGESIM architecture can be viewed as a generic, real-time, training simulator architecture! Note also that this architecture is compatible with the Reuse, Composability, Peer-to-Peer, Test and Training Enabling Architecture (ie TENA [TENA (2002)]) whereby reuse can occur at any scale – reusable components can be combined to create an application, a system, or a system-of-systems.

SUPPORT FOR DIS, HLA AND/OR TENA

Currently IEEE 1278.1A DIS is the "communication protocol" that provides interoperability between the ADGESIM Peer-to-Peer applications. DIS was initially chosen because of the requirement that the ADGESIM would have to interoperate with other ADF and coalition DIS simulators. There is however no reason why HLA or TENA could not be used to provide inter-component interoperability if the appropriate code was used. A HLA Run-Time Infrastructure or TENA toolkit could be used or, alternatively, a protocol independent, middleware toolkit (such as MÄK Technologies VR-Link toolkit) could also be used. A middleware toolkit uses the same, higher level of abstraction, API regardless of the protocol required. The protocol to be used is selected via a make file library selection process and conditional compilation. The use of such a middleware toolkit will enable each ADGESIM application to support DIS, HLA and/or TENA when VR-Link supports TENA - as MÄK Technologies have indicated it will [Mak].

The ADGESIM Pilot Interface and the Simulation Manager applications already use VR-Link. This software was not used in every ADGESIM application as each application would then require a VR-Link run-time license. However the use of VR-Link in every application should reduce the total amount of code thus somewhat reducing the cost of maintenance as well as enabling support for DIS, HLA and/or TENA.

The use of this "protocol independent" strategy should make ADGESIM components prime candidates for simulation applications able to be used in Australian Joint Combined Training Centres to be established in Northern Australia.

THE ADGESIM EVOLUTIONARY ACQUISITION AND SPIRAL DEVELOPMENT PROCESS

As part of the Australian Defence Department, DTSO does not normally develop operational training simulators. Such simulators are usually procured from industry. However the research and development of a modern, cost-effective, architecture able to be rapidly constructed from reusable, interoperable and supportable, generic simulator components was funded under the RAAF's DSTO Virtual Air Environment (VAE) task's Advanced Concept Technology Demonstrator (ACTD) program.

A Request for Proposal (RFP) was sent to industry seeking cost and schedule information for a distributed simulation, Air Defence Controller trainer. Industry delivery estimates of three to five years, changing technology requirements, and major changes to real-world operational tempo in the Surveillance Control Group meant that neither the functionality nor the schedule specified in the RFP would have met operational and training needs. Redevelopment and issue of a new RFP to industry was considered unlikely to produce results within the required timescales. A decision to allow DSTO to continue development of, and to subsequently field, a distributed simulation system constructed from concept demonstrator components already developed in the ADSL, was made by the RAAF in January 2002. Development of the simulator application suite began in March 2002 using VR-Forces and VR-Link along with customised thin-client applications on the ADSL mandated, hardware and software development infrastructure.

ADGESIM was delivered and installed in October 2002 at the Surveillance Control Group Surveillance and Control Training Unit (SACTU) at the operational, RAAF Williamtown F/A-18 fighter base. This six to seven month development cycle time compares favourably with industry estimates of between three (60% to 65% functionality) and five years (90% to 95% functionality) in the original RFP responses. This initial ADGESIM capability was delivered at considerably less cost than solutions proposed by industry in response to the RFP.

Several versions of the ADGESIM have now been delivered and installed where each new version has been further enhanced by the evolving needs of the user as the user has gained an understanding of the capabilities of the technology.

This process is identical to the Evolutionary Acquisition and Spiral Development process defined in the US DoDD 5000.1 and DoDI 5000.2 acquisition policy documents [OSD, 2003]. The objective of the 5000 series acquisition policy is to deliver (ie acquire) interoperable and supportable, advanced technology systems faster and more cost-effectively. These objectives certainly appear to have been achieved for the ADGESIM.

CONCLUSIONS AND LESSONS LEARNED

The components that make up the ADGESIM used for training RAAF Air Defence Controllers have been described in detail in this paper.

The ADGESIM has been designed using a three sub-system architecture - real system, simulator specific component(s) and generic, reusable toolbox components. This architecture is compatible with the reuse, composability and peer-to-peer, Test and Training Enabling Architecture (ie TENA) whereby reuse can occur at any scale – reusable components can be combined to create an application, a system, or a system-of-systems. This simulator architecture may be applicable to all real-time, training simulators.

Stimulating (as against emulating) real operational systems in a training simulator reduces trainer concurrency problems considerably.

This distributed simulation, peer-to-peer architecture (ie the "black-box") approach helps keep the development and maintenance of each ADGESIM application manageable.

The use of COTS reduces the development and maintenance costs of software, documentation and training material considerably.

It is unlikely that an operational simulator can be completely created from available COTS applications and customised thin-client applications have been developed where no commercial equivalents exist.

Use of the dominant desktop operating system, programming language and corresponding software development environment (Microsoft Windows XP, Visual Studio.NET C++ compiler and its associated integrated development environment) also reduces the cost of software development and maintenance.

The use of a distributed simulation, architecture on a network of "commodity-item" PCs reduces hardware acquisition and maintenance costs to a minimum.

Several ADGESIM applications are multithreaded and are thus able to take advantage of hyper-threading and multi-core processor technology.

ADGESIM currently supports the IEEE 1278.1A DIS interoperability protocol. A strategy has been proposed whereby the use of a protocol independent, middleware API toolkit will provide each ADGESIM component/application with the ability to support DIS, HLA and/or TENA.

A DoD Series 5000 compliant, evolutionary acquisition and spiral development process was used to transition ADGESIM components from a laboratory, concept demonstrator program to an advanced technology, operational simulator that has been developed, delivered and installed in a cost-effective and timely fashion, and is highly interoperable and supportable.

In summary, a risk reduction strategy of integrating COTS products with in-house developed, customised thin-client applications, that have no commercial equivalent, has reduced considerably the purchase, development and maintenance costs of the ADGESIM training system and has contributed to the required project functionality being delivered on time, on budget and according to specifications.

REFERENCES

Graebener, R. J., Rafuse, G., Miller, R., & Yao, K. (2004). Successful Joint Experimentation Starts at the Data Collection Trail – Part II. *Proceedings of the 2004 Industry / Interservice Training Systems Conference*, Orlando, FL: National Security Industrial Association.

[IEEE 1278.1/A 1,2] IEEE 1278.1 (1995), 1278.1A (1998), *IEEE Standard for Information Technology – Protocols for Distributed Interactive Simulation Applications*.

[Emdee] www.emdee.com

[HLA (2000)] – 1516-2000, *IEEE Standard for Modeling and Simulation (M&S), High Level Architecture (HLA)*, (2000)

[Mak] www.mak.com

[OSD, 2003] http://dod5000.dau.mil/

[Qantas] www.qantas.com.au

[Solipsys] www.solipsys.com

[TENA (2002)] www.fi2010.org