Abstract. Seminar wargaming is a human-adjudicated map-based activity used by the military for experimentation, training and within the Military Appreciation Process, for course of action analysis. It differs fundamentally from typical wargaming simulations that are part of a larger grouping, described as system wargames, where the behaviour of forces and outcomes of events are determined exclusively by algorithms or rules. The Joint Seminar Wargaming Adjudication Tool (jSWAT) is a computer-based seminar wargaming environment that has been used extensively for Army and Air Force experimentation. This paper describes the outcome of a three-year development to provide jSWAT with simulation-based tools designed to add objectivity to adjudication of manoeuvre, logistics, combat, and intelligence gathering. The tools also include the jSWAT planning environment that matches military tasks, required effects and assigned force elements in a synchronisation matrix. The novelty of this work is the combination of features of seminar and system wargames in a single environment.

1. INTRODUCTION

The Defence Science and Technology Organisation (DSTO), through its Land Operations Division (LOD), provides ongoing support to the Army Experimental Framework (AEF) by the provision of staff who participate in the planning, conduct and analysis of experiments. The AEF conducts an annual program of activities to investigate force structures and options to meet future Army requirements. Seminar wargaming is used to investigate broad issues at the campaign level to determine those that merit closer study, often in a conventional simulation environment. DSTO developed the joint Seminar Wargaming Adjudication Tool (jSWAT) to support seminar wargaming by transforming the human-adjudicated seminar wargaming environment from a hard copy map with physical objects representing military forces and manual recording of play, to a computer environment with electronic map, ORBAT1 browser, scheme of manoeuvre (SOM), force status and an ability to save turns and points of interest for future reference and analysis. Recent work to develop jSWAT2, described in this paper, has provided users with a suite of simulation-based adjudication tools to add rigour to adjudication of manoeuvre, logistics, combat, and intelligence gathering. The development of jSWAT2 has coincided with the recognition that, in the modern world, the Army must adopt a holistic approach to operations and can no longer focus only on ‘red on blue’ land combat operations [1]. To that end, jSWAT2 has begun to address requirements for wargaming in this more complex environment where Army and its small specialist elements play a leading role in peace keeping and nation building.

2. APPLIED WARGAMING CONCEPTS

2.1 System Wargames and Seminar Wargames

For the purpose of this discussion, wargaming is a term used to describe any representation of military activity, short of live exercises and conflict, where human players manoeuvre their assigned forces in the context of the opposing forces and the physical, operational and sociological characteristics of the area of operations (AO). Interactive entity based and aggregate computer simulations fall within the scope of wargaming, as do the variety of board games and other wargaming environments catering to both the military professional and hobbyist.

A key development in the modern concept of professional wargaming was the adoption of Kriegsspiel by the Prussians in the early 19th Century [2]. When the concept of Kriegsspiel play was published by von Reisswitz in 1824, an intricate set of rules was provided and these were developed through the ensuing years with the result that the game became tedious; dominated by calculation and umpire’s decisions as to which rules to apply. Dissatisfaction

---

1 ORBAT is an acronym for Order of Battle.
2 For the purposes of this paper, jSWAT2 refers to the current version. jSWAT is used for earlier versions and for references to jSWAT in general.
with the difficulties of learning and applying the rules led to development of free Kriegsspiel, from about 1876, where decisions of the umpire, based on experience, replaced rules and became the means of managing play. Wargames following the original concepts of Kriegsspiel are characterised as system wargames, where the outcome of player interactions is determined by reference to rules, tables and calculations or, in the modern context, real time computer simulation. The free Kriegsspiel concepts are reflected in seminar wargames where the map-based game illustrates the situation, plans, and courses of action and outcomes are determined by discussion of players and assessment by expert umpires.

Army officers assigned to AEF experimentation come already equipped with seminar wargaming skills developed from their use of wargaming as part of their established planning processes. The AEF adapted these techniques to the broad examination of experimental scenarios, typically at the operational level, in order to determine where more detailed tactical studies were needed. These were then generally played using the Janus (real time entity-level) wargame.

2.2 Conduct and Adjudication of Seminar Wargames

Seminar wargames can be conducted with a minimum of specialised equipment; at the most basic level, little more is required than a map and some objects that can be used to represent force elements. Games can be completed rapidly. Unnecessary detail can be abstracted away so that players can concentrate on broad issues without the distraction of managing entities at the tactical level. It should be noted that the players representing friendly and enemy sides are not necessarily adversaries in the usual sense, but are representing different viewpoints in an effort to reach a common understanding of the issues raised by the scenario.

A seminar wargame is conducted on a map of the AO by players representing the viewpoints of participants in the battle, typically a friendly (blue) and enemy (red) force. The principal players represent the force commanders, who may be assisted by staff with specialist battlefield operating system (BOS) expertise forming the blue and red cells. Play is turn based under control of an adjudicator, who may also have access to specialist BOS experts collectively forming a white cell. Also present are staff to maintain a record of the game.

For a typical ‘round-table’ seminar wargame, participants share a common wargaming map and consequently situational awareness (SA). An alternative arrangement is to separate blue and red cells and isolate the viewpoints, in which case each side builds and maintains SA based on information provided by the white cell.

Play between blue and red cells represents conflict on a conventional battlefield where the identity of friendly and enemy forces is well understood. However, by contrast, contemporary military operations are characterised by more complex relationships where various insurgent and militia forces, and non-government agencies, may not be fully controlled by either (red or blue) side. These conditions present additional challenges for turns based players and adjudicators; there are additional viewpoints and therefore either more cells or combined viewpoints are required.

2.3 Computer-aided Seminar Wargaming with jSWAT

DSTO staff supporting the AEF program of experimentation became familiar with the use of Seminar Wargaming and their perceptions that many advantages could be gained by moving the process to an electronic environment led to the development of the first versions of the jSWAT software [3]. jSWAT was first used in 2003 and this version displayed the fundamental features that endure in jSWAT2; a map display with overlays, drawing tools, an ORBAT browser, and the ability to save game states. Among the wargaming innovations adopted by AEF was the use of a smart board laid horizontally for use as a ‘marked-up map’ playing surface. The screen image from jSWAT is projected onto the surface using a roof mounted projector and mirror. The smart board functions as a large touch screen and many of the inputs to jSWAT can be made by touching, or dragging a fingertip. For a ‘round-table’ game the blue and red commanders sit at opposite sends of the screen with their staff at hand. The adjudicator, together with an experimental facilitator and recorders sit to the side.

jSWAT was adopted by a small but diverse community within Defence and used for experimentation by the AEF, the Air Power Development Centre, the former Defence Systems Analysis Division and HQ 1 Division.

3. jSWAT2 - APPLICATION OF SIMULATION TO SEMINAR WARGAME ADJUDICATION

Through the observation of seminar wargames it was recognised that more robust experimental outcomes could be achieved if jSWAT was provided with tools to assist with adjudication. Realisation of these tools became possible when DSTO was given access to contract funding from the Australian Defence Simulation Office’s (ADSO) Simulation Minors Program.

Requirements for jSWAT2 were to provide an environment for conducting an adjudicated seminar wargame on a Windows platform. Emphasis was to be on highly abstracted representations for joint operational level wargames, where typically 5-10 days of operations are played out in a single day. Flexibility was required to support round table and single or double blind games.

jSWAT2 was specified to be a qualitative tool to assist in the identification of branches and sequels emerging from an operational plan. To meet this need it was
necessary to provide the ability to capture decision points and the ability to reset the game to an earlier decision point to allow investigation of alternative courses of action (branches and sequels). The saved decision points also provide a record of the game for after action review and analysis.

Tools were specified to assist with regulating the rate of movement of entities (time and space), levels of detection by opposing forces, potential outcomes of land, air and sea combat, and logistics consumption, replenishment and transport.

An ambitious goal was to provide a tool, the jSWAT planning environment that could capture a plan expressed as graphical elements and produce a synchronisation matrix.

To remain faithful to the concepts of seminar wargaming requiring only maps, transparent overlays and tokens to represent forces, jSWAT2 was specified to be capable of supporting wargames in the absence of detailed data.

4. jSWAT2 IMPLEMENTATION

4.1 Introduction

From its inception, jSWAT was developed on the OpenMap [4] application, an open-source geographical information system toolkit developed by BBN Technologies. The earliest implementation was used to support an AEF activity in 2003.

The project to provide adjudication tools within jSWAT began in mid-2005. The task was envisaged as an extension of the existing code base to incorporate additional data elements and methods. However, a review of the code uncovered a number of issues with the received architecture relating to the suitability of the underlying technologies adopted and the complexity of the programming interface.

The preferred option from the review was to rewrite the jSWAT code. However, DSTO envisaged implementing the adjudication modules sequentially to support ongoing experimentation programs, so the less favoured option of remediation of the existing code was adopted. Armed with the findings of the review, the programmers began making steady progress in implementing the land combat adjudication model (LCAM) and frameworks for the intelligence surveillance and reconnaissance (ISR) model and air combat adjudication model (ACAM).

Design of the logistics model forced a change in thinking in order to deal with the number of new data elements required to represent logistics holdings. A decision was taken to implement an architecture that was optimised for this purpose. When no longer constrained by the legacy architecture, the development team made rapid progress, routinely finishing tasks ahead of estimated schedules.

jSWAT2 retains a number of features of earlier versions. The software is built in Java and intended to run in a Windows environment, OpenMap is used for the map display and the use of a smart board as a playing surface is retained.

The jSWAT2 software was designed to provide modularity and handle the data structures required by the adjudication models. Extensive use was made of open source re-useable code.

The jSWAT2 software is implemented as a client-server architecture. The server provides plug-in communication support for networked games. The Apache Jackrabbit [5] implementation of the JSR-170 Content Repository is used to store, share and provide data persistence.

jSWAT2 clients are implemented as ‘fat clients’. Each client application provides all required functions. When networked, clients share data, but each client supports its own processing workload. Processing of changes is initiated by injection of new data rather than listening for changes. This means that only one client needs to calculate and write the data, whereas the remaining clients only have to update displays after a data change.

Data and services are separate in order to make it easy to modify one part of the system without having to change others (e.g. changes made to the Jackrabbit interface had no impact on the actual algorithm code).

The run-time configuration for jSWAT2 is managed by a combination of the Spring Framework [6] and NetBeans Framework [7].

jSWAT2 uses Apache Maven [8] for automated building, testing and deployment. Maven improves the release time and minimizes the inadvertent introduction of bugs while fixing other errors.

4.2 User Interface

OpenMap is used to display the maps and other positional information. This allows user configuration of what is displayed.

The jSWAT2 GUI was developed using the NetBeans platform to provide windows and menus. The layout has title bar, menu and tool bars at the top of the display and the remainder of the screen is taken up by the map display and a series of windows. Whilst there is insufficient space to fully describe the GUI, a brief description of features of particular significance to the user is provided. Additional features will be described in conjunction with the models they support.

4.3 Map Display

The map display is the central feature of the GUI. The map display is implemented using OpenMap within its own window. The OpenMap toolbar is located at the top of the map window and used for map navigation. jSWAT2 can provide multiple map windows on a
single workstation. OpenMap is specifically designed to display maps in the CADRG format, but can also accommodate pictorial information scanned as .jpeg files with the addition of registration and scaling information. This allows jSWAT to display scanned maps and satellite imagery.

4.4 Layers
The OpenMap display comprises a number of layers. These layers are used to display maps, images, drawing or construct map overlays. jSWAT2 uses layers to display information generated during the game and made available for the user. A layers window provides control of the visibility and ordering of layers.

4.5 ORBAT Browser
The ORBAT browser is a key feature of the user interface that displays the units available to the players as a collapsible list following the command structure using MIL-STD-2025B [9] symbols. Expanding the list shows components of a unit in greater detail to the limit set by the scenario developer. Components can also be moved between units to reflect task organisation.

Units at any degree of expansion can be displayed in the game by dragging and dropping their icons onto the map display. The ORBAT browser provides a means of visually tracking which units are in use and which are unassigned.

4.6 Drawing
jSWAT provides a capability for drawing graphical overlays using the tools provided by OpenMap. Drawing layers can be created, deleted and saved.

4.7 Properties Window
The GUI is provided with a properties window that shows the properties table for the object currently selected. Object properties are the basis for the functioning of jSWAT models. The properties window can be used to examine and modify properties and is the means to access properties when there is no other tool provided.

5. ORBAT BUILDING TOOLS

5.1 ORBAT Browser
The ORBAT browser provides tools to create new blank entities, and to copy, paste and delete existing entities. Entities created using the ORBAT Browser initially have no properties assigned. Properties can be added manually with the assistance of a wizard framework or directly using the properties window. However, low-level entities are generally created using stereotypes as described in the next section.

5.2 Stereotypes
Stereotypes used in jSWAT to assist the process of ORBAT building by providing a library of generic entities. Capability stereotypes are defined for low-level force elements, commonly individual soldiers, vehicles or small units. Organisational stereotypes are created for larger force elements from combinations of capability stereotypes or lower-level organisational stereotypes. Properties of the component stereotypes are aggregated when new organisational stereotypes are defined.

Tools for creation of capability and organisational stereotypes are provided in the stereotype window. In both cases, a wizard framework is used to assist in defining new stereotypes and assigning properties. The stereotype window also provides tools to copy, paste and delete existing stereotypes.

Stereotypes are used to populate the ORBAT by selecting the required stereotype in the stereotype window and dragging and dropping a copy to the Orbat browser.

5.3 Aggregation and Distribution of Properties
The process of aggregation and distribution of properties underpins the processes of ORBAT building and task organisation. The combat scores, logistics holdings, capable tasks and effects properties of the higher level unit are the aggregated properties of its components. When changes are made to the ORBAT as a result of task organisation, properties are aggregated with the new parent unit. When parent units are used on the map, changes in properties as a result of consumption or combat are distributed among the component units down to the lowest level.

6. MODELS

6.1 Background
The jSWAT2 models for supporting adjudication of seminar wargames are described below. They are intended to provide simple representations of the interactions between entities and between entities and the environment as discussed below. In general, entities within the wargame represent aggregated military units, not individuals or individual vehicles.

6.2 Environment
jSWAT2 is intended to be used without requiring the detailed terrain databases of a conventional simulation. Tools are provided in the Map menu to mark up a map overlay, the terrain layer, to delineate terrain types that influence operations (e.g. mountain, forest, ocean). A set of properties is defined for each terrain type that describes the effect of the terrain on sensors and mobility.
6.3 Time and Space
The time and space model provides the means of analysing route timings for movement of forces. A tool to draw routes and roads is provided in the map menu. When the route or road is appended to the properties of an entity, the model can calculate timings at the vertices (waypoints) based on a combination of the speed specified for the entity and, for land forces, the terrain crossed. When the clock is advanced, entities with a route will move to the distance calculated by the model. When the end of a route is reached, a notification appears and the route is removed from the properties for that entity. Movement along roads is not modified by terrain.

6.4 Intelligence, Surveillance and Reconnaissance
The ISR model represents the ability of sensors incorporated in each entity to detect other entities. Each entity can carry multiple sensors and each sensor has a search footprint modelled as a range ring centred on the entity. Target entities have a ground footprint with a radius depending on the size of the entity. The detection level is calculated from the overlap of the search footprint and ground footprint. The search footprint is modified by the terrain type whenever ground entities are targeted and by time of day and weather.

6.5 Land Combat Adjudication
The land combat adjudication model (LCAM) determines the outcome of a battle from the aggregated combat power of engaging forces, their initial offensive or defensive posture and the results of a series of dice rolls. Combat power is determined from the Situational Force Scoring method as developed by the Rand Corporation and modified by DSTO [10]. LCAM calculates results for 1000 battle runs determining the probabilities of wins, losses and inconclusive results, and the average time to complete the battle for each case. After the adjudicator selects the result to be used for further play, the combat scores are modified to reflect attrition. Selection of the combat tool from the menu enables selection of force elements for the battle from the GUI and opens a wizard framework used to access parameter settings and results. At the end of a battle, a marker is placed at the battle site. Clicking the marker opens a dialog box summarising the battle.

6.6 Air Combat Adjudication
The air combat adjudication model (ACAM) provides the operator with the means of drawing zones defended by anti-aircraft artillery, missiles or combat air patrols. Probability of engagement of formations of aircraft flying through a defended zone increases the longer they remain in the zone. ACAM is currently implemented as a framework without supporting algorithms. However, it is a useful adjudication tool in its present form, particularly in a double blind game, because it is capable of being used in conjunction with the time and space annotations to determine the times at which air assets are exposed to enemy forces.

6.7 Maritime Combat Adjudication
The maritime combat adjudication model (MCAM) is an adaptation of LCAM.

6.8 Logistics
The logistics model is centred on entity consumption and is supported by the holding of equipment and stores by entities, as well as warehousing, transport and re-supply. Entity properties include a table that lists holdings associated with the entity and cargo it can carry. Each entity has a maximum holding of each class of stores that can be transported and a maximum total for own consumption (this enforces the maximum load when more than one type of store is carried as cargo). The model for consumption of stores caters for daily consumption (typically food and water), consumption by distance travelled (typically fuel) and consumption as a result of combat attrition. Warehouses are entities that can be sited as required and entities can draw stocks as either supplies or cargo. Cargo can be transported, unloaded and used by other entities as required.

6.9 jSWAT Planning Environment
The jSWAT planning environment is an implementation of a concept: 'The Diagram is the Plan'. The jSWAT Planning Environment is an application of a planning tool called the Course of Action Scheduling Tool (COAST) [11] that uses Petri nets to schedule resources required to achieve battlefield effects and produces results in the form of a synchronisation matrix. The production of a synchronisation matrix is intended to show that players have the capacity to undertake planned operations, entities can be available when required to support the proposed timings, and to provide an additional record of the player’s intent for the analyst.

A menu tool is provided that allows the user to select a military task and place the task symbol on the map indicating the location and axis for the task. Underlying the symbol is a generic COAST plan for the operation. In order to produce the synchronisation matrix, COAST requires the user to assign forces to the plan that can provide the required effects (entity properties include tasks that the entity can perform and effects). The task symbols are scaleable. The size of forces required to be allocated is determined from the frontage and the time taken by the length represented by the symbol.

The compilation of COAST plans formulated to date has been time consuming and resources have not been available to provide a comprehensive plan set. However, the plans contain a number of sub-plans and the work of compiling plans has demonstrated that reuse of sub-plans should simplify adding new tasks. Plans are provided with jSWAT2 for the tasks Advance,
Attack, Clear, Defend and Lodge – sufficient to assess the utility of the jSWAT planning environment for the wargamer.

6.10 Turns Model
Maintaining a record of the progress of a wargame is an essential element of jSWAT. The turns model provides a tool to manually save the game state at the end of each turn. A series of turns records the passage of a game for subsequent analysis or review. Games can be reset to any turn and replayed to investigate alternative courses of action.

The turns model has been developed using the functionality of the Jackrabbit content repository in conjunction with custom version control. Each turn saved by the user results in an incremental upgrade to the repository. A turns history window provides the operator or analyst with a record of the saved turns arranged in a hierarchical sequence labelled by game time and date.

6.11 Permissions Model
jSWAT2 supports multi-player games through its permissions model. Networked workstations are assigned to the white (adjudication) cell and to cells representing the sides participating in the game. The group of workstations can be set by white cell to allow movement of friendly, hostile or neutral entities as separate phases within each turn. Multi-player games are also supported by providing the ability to set each workstation to a display the game from the viewpoint of the side it represents.

7. SUPPORT FOR ADAPTIVE CAMPAIGNING
Since jSWAT development began, the emphasis of military experimentation has changed from conventional combat against a similarly equipped enemy to a broader view of conflict resolution where there is greater emphasis on the role of the military in stabilisation, reconstruction, counterinsurgency, security and civil-military cooperation. In this environment, the military acts in collaboration with host governments, coalition forces, and a multiplicity of government and non-government agencies. Hostile forces may include conventional forces, but may also include militias, insurgents, terrorists, sympathisers and criminal elements that cannot be readily identified amongst the population and who compete with government for the allegiance of groups within the population.

The practice of wargaming for adaptive campaigning is in development so there are no clear user requirements that can be translated into features of a tool like jSWAT. jSWAT2 has anticipated some requirements by including capability for multi-sided wargaming, and the inclusion of ‘Neutral’ as a side for both permissions and viewpoints. The modular implementation of jSWAT should make it easy to add additional adjudication tools as the need is identified.

8. CONCLUSION
The work described was completed in June 2008. jSWAT2 was deployed for the Headline Experiment in August 2008; this event provided an opportunity to use jSWAT2 under the technical control of the development team. A number of residual bugs emerged and were eliminated after which jSWAT2 functioned in accordance with its design intent. Subsequently, some changes have been made to the way jSWAT2 accesses the repository to speed up saving and loading of large multi-player games.

jSWAT2 is now available for use within the Department of Defence. The project has demonstrated the ability to combine features of seminar and systems wargames in a single environment.

9. ACKNOWLEDGEMENTS
The authors wish to acknowledge the many contributors to jSWAT2 including Damien Presser, Cidgem Dilek, Justin King, Justin Beck, Justin Millikan, Karyn I’Anson and Brice Mitchell. Funding from the ADSO Simulation Minors Project made this work possible and our thanks are extended to ADSO staff.

10. REFERENCES