GAMOV: Towards More Affordable and Capable Modelling

[dst1]

Nick Bell

[Dstl] Defence and Security Analysis Division,
Dstl Portsdown West, Portsdown Hill Road, Fareham, Hants
PO17 6AD
03067703297
Email: nbell@dstl.gov.uk

Abstract

The current global uncertainty and emerging ways of warfare, in an era of tight budgets, present a challenge to those who seek to understand military campaigns through modelling.

The GAMOV modelling framework developed by Dstl uses cutting edge computer science to meet these challenges through the use of a modular framework.

The approach allows re-use of common modelling components to create required analytical capabilities through an object functional paradigm.

The construction and testing of the common functionality is now complete, and the GAMOV framework is now ready for models to be incorporated.

Other developments using GAMOV's object functional approach hold the possibility that semantic web technologies can be used to more rapidly analyse the emergent phenomena which manifest within modelled military campaigns.

What is GAMOV?

A toolset for building models out of common components.

- Modular library of previously used representations
  - Speed to build new models should increase as more are added
  - Eases validation, verification and changes to existing models
- Heavy use of "Off The Shelf" libraries, tools, and functionality to reduce cost and maintenance burden and improve flexibility
- Core modelling engine implemented in Python
- Operating System agnostic
- Web based, expandable Graphical User Interface
- Modular design allows parallel development and use with low risk

GAMOV Toolkit

- Core Engine is re-used each time
- Entities: Actors in the model
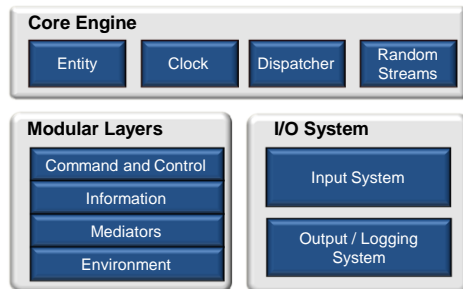- Mediators: Rules governing the interactions between entities

Core Engine

| Entity | Clock | Dispatcher | Random Streams |

Modular Layers

- Command and Control
- Information
- Mediators
- Environment

I/O System

- Input System
- Output / Logging System

Figure 1. The GAMOV system

Development History

Year	Delivery	Framework
2007 - 08	Concept Paper	Direction Set
2008 - 09	Deterministic Lanchester	Supports Modelling
2009 - 10	Components Exchanged	Supports Model Change
2010 - 11	Grand Simple Land Air Model (SLAM)	Supports New Models
2011 - 12	Aerial Delivery Model	Supports Different Models
2012 - 13	Intervention Scenario (Wartime Planning Tool)	Supports RESTful Models
2013 - 14	Logs Using Mission Planner	Supports Future Direction
2014 -15	Combat Air Model (CAM)	Productised

Where next this year?

- Spec and build an improved CAM
- Improved air to surface representation
- Use the "Mission Planner" to improve C3
  - Potential for much reduced scenario setup time
- Improve in/out system-switch to industry standard JSON format

Combat Air Model (CAM)

- Developed in response to modelling need from the Strategic Decision Support to Air project
- Intended to represent the air domain contribution to a campaign at the theatre level
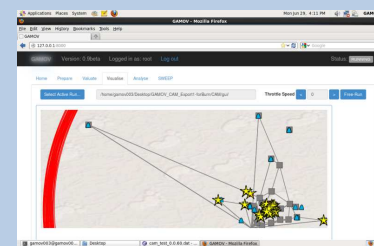- Easier to use than current capability (COMAND)
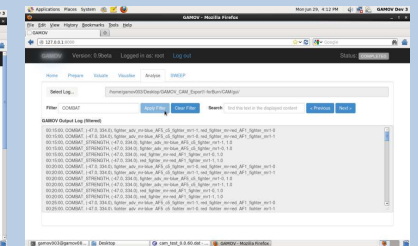- Initial focus on fast air

Figure 2. The CAM Running

Figure 3. CAM output being analysed

Future Work

- Semantic Web Examiner of Emergent Phenomena (SWEEP) tool (PhD Research with University of Portsmouth)
  - Allow more automatic analysis of model runs
  - Improved visualisation of data
- Aspire to generate a "Joint Medium Scale Land/Air Model"

What is Mission Planner?

An automation of the planning of missions within simulation models

- Uses mathematical optimisation techniques
  - Genetic Programme and
  - Simulated Annealing
- Designed to reduce the burden on the analyst in setting up complex branching scripts in stochastic models
- "Perception" based-improves representation of C4ISR
- Optimises against a number of potential courses of action
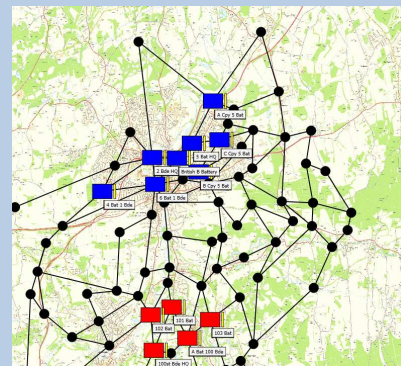- Test bed built around the Simple Brigade (SIMBRIG) model

Figure 4. The Mission Planner Plan visualiser

Summary

- GAMOV is an extant capability
- MOD is funding continued development of this capability
- Modularity:
  - Reduces risk
  - Reduces long term cost
  - Increases re-use, adaptation and exploitation
  - Increases supportability
- Exploits open source computer science developments
- Stable version whilst conducting parallel research, focusing on:
  - Reducing the time it takes to analyse a problem
  - Greater technical coverage- e.g. of C4ISR