

Validating a Scenario Assessment Tool

Caroline Morin PhD

presenting for:

Geoff Hone, Ian Whitworth & Andy Farmilo

Department of Informatics and Sensors

Cranfield University at the Defence Academy of the United Kingdom

Shrivenham SN6 8LA UK

i.r.whitworth; a.farmilo; @cranfield.ac.uk

ghone.cu@defenceacademy.mod.uk

&

David Swift PhD

Directorate of Training (Army), HQ Land Forces UK

prinpsy@ditraining.mod.uk

Why do scenarios matter?

Essentially:

**they define the starting conditions,
and the key parameters,
for just about any war-game ...
or training exercise ...
or mission rehearsal ...
or procurement study ...
etc ...**

They must be fit for purpose to be of value

Why do scenarios matter? (2)

Procure against a vague scenario:
get a vague product

Train against a vague scenario:
get vague training

Scenarios define starting conditions:
so

Start Right and Finish Right

This presentation tells a story

- The initial aim was to derive a generic architecture for scenarios (already reported)
- This exposed the team to the MSDL (Military Scenario Description Language)
- It was apparent that the initial generic architecture did not map directly on to the MSDL
- So an intermediate stage was developed (forming an extended architecture)
- Result: a multi-stage process to Validate and Verify a scenario

Training is the biggest use of scenarios

The UK has used a Systems Approach to training for more than a decade.

The Defence Systems Approach to Training (DSAT) formalises the design of training.

Scenario Architectures ensure that Scenario design matches Training needs.

MSDL gives a uniform way of describing scenarios (in a machine-readable form)

Individual Training

This is very well understood

The Defence Systems Approach to Training (DSAT) ensures that:

- the right tasks are trained**
- to the right standard**
- and in the right order**

Collective Training

This means taking a set of well-trained individuals, and getting them to function well together.

This requires the right training exercises:

In a Command Post

In a networked simulator (e.g. CCTT)

In the field

The exercise scenario MUST be the starting point.

Stage One

**The starting point is the primary architecture
This is composed of components and items
(a two-level decomposition)**

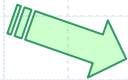
**Any scenario should conform to this architecture,
but without always needing every item**

**The most important thing is to have a
statement of purpose**

**But – primary architecture does not map onto
the MSDL**

Stage 1 effect

Initial
requirement
and scenario



Completeness Check
Primary
Architecture

Stage Two

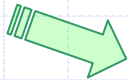
The primary architecture was decomposed to four levels down.

This gave sub-items that did map onto the MSDL.

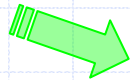
Now there is traceability from the initial requirement, through the primary architecture, through the secondary architecture to the detail of the MSDL

Stage 2 effect

Initial
requirement
and scenario



Completeness Check
Primary
Architecture



Secondary
Architecture

Maps to MSDL

From Paper to PC

The checks on the first two stages can be handled very easily by “flat-paper” forms (as well as by spreadsheet on a PC or laptop).

Either way works – the traceability requirements will usually dictate the best approach.

The MSDL can enable a scenario to be described in detail – using the Extended Mark-up Language (XML) – in a file that can be several thousand lines long.

This is best checked by computer.

Stage Three

Now the computer comes into its own, and the needs of security have to be considered.

Any scenario that needs to be described by MSDL has enough detail to reveal much of the C2 involved.

The decision was made to go for a web-based approach for checking. This can be operated over a secure network or on a stand-alone PC.

Why Stage Three?

Most MSDL descriptions are authored with appropriate tools, and it was originally assumed that these would guarantee correct MSDL syntax

In practice, it has been found that correct syntax cannot be guaranteed

Stage Three Checks

The principle is that a scenario written in MSDL is compared to the fundamental MSDL schema.

- The schema has to reside on the host computer first (this means that updates can be easily made)
- Next the checking tool is uploaded
- Then the Scenario file can be uploaded

Running the tool can now take place with both initial and detailed results output.

MSDL Check Results

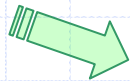
These are handled in two stages: syntax and completeness.

- The first pass of the check tool tests the scenario file to see if it conforms to the MSDL standard. This is a fast check with a **YES/NO** result.
- The second pass checks that the scenario file contains details for all the elements identified in the architecture.

This produces an XML file for further processing.

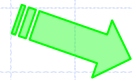
The whole process:

Initial
requirement
and scenario



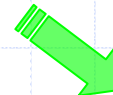
Completeness Check

Primary
Architecture



Secondary
Architecture

Maps to MSDL



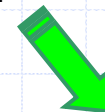
MSDL
Check 2

MSDL Detail

MSDL Syntax



MSDL
Check 1



A Verified
Scenario

Each step in the process has been validated

Validation and Verification

It can now be seen that there is now a validation process rather than a single act.

The primary architecture was validated against previously published outline scenarios, and acts to validate the initial scenario requirement.

The secondary architecture was validated against specific UK defence scenarios (but can also be used to verify that all the items in a Training Needs Analysis have been covered).

Validation and Verification (2)

The MSDL check tool was validated against a published scenario written in MSDL, and can provide internal verification that the MSDL file conforms to standard.

The XML file (the check tool output) can be fed back to the task analysis (using THE HTA TOOL) and can further verify that all training issues have been addressed.

THE HTA TOOL can then produce a file for transfer to Excel to enable a colour-coded report.

The Checked Output

This can now be inserted into a report in the required format.

The MSDL check tool, and the scenario in MSDL form, can both now be removed from the computer.

This has benefits for both security and file-space needs.

Summary

A three step process, with

Each step validated against known scenarios

Steps validated against previous step

Final step verified against known schema

**Result is a scenario that is ready for a
Subject Matter Expert to validate**

Note that few – if any – SMEs can read XML

And finally:

Geoff Hone

ghone.cu@defenceacademy.mod.uk
(gnhone@globalnet.co.uk from September '09)

Ian Whitworth

i.r.whitworth@cranfield.ac.uk

Andy Farmilo

a.farmilo@cranfield.ac.uk

**Department of Informatics and Sensors
Cranfield University at the Defence Academy of the United Kingdom
Shrivenham SN6 8LA UK**

David Swift

prinpsy@ditraining.mod.uk

Directorate of Training (Army), HQ Land Forces UK