Investigation into the Usability of MSDL in South African C2 Tactical Simulations

Willem H. le Roux
Council for Scientific and Industrial Research[1]
Meiring Naude Road
Pretoria, 0001
+27 12 841 4867
whleroux@csir.co.za

Keywords:
MSDL, Scenario Definition, Tactical Simulations, Command & Control, Modelling & Simulation

ABSTRACT: *The applicability and usability of the Military Simulation Definition Language (MSDL) is investigated in context of the South African air defence simulation community. A constructive, tactical command and control simulation environment used for acquisition decision support, concept evaluation and doctrine development support as part of a phased ground-based air defence procurement project is applied as test-case. This paper reports on the process followed, issues encountered and the outcome of translating the proprietary scenario definition format of the air defence simulation environment into an MSDL compatible one. Recommendations for both the MSDL community and developers of the South African air defence simulation environment are presented throughout the paper.*

1. Introduction

In order to promote interoperability and standardisation of scenario definitions, an investigation has been undertaken to assess the applicability and usability of the Military Scenario Definition Language (MSDL) [1] in the South African defence community. A tactical command and control (C2) simulation environment developed as a means for decision support to a Ground-based Air Defence System (GBADS) procurement program was used as test-case [2-5]. The C2 simulation environment is primarily used in a constructive mode, but is applied in virtual simulations to support operators interacting with the simulation via mock-ups of air defence consoles and the integration of external systems. These external systems range from external air picture and sensor systems, flight simulators to other similar simulations.

It is important to point out that this paper is solely based on the specification of MSDL as provided in [1], and that no other sources of information have been consulted to apply the standard to the simulation environment. A short overview of the existing scenario definition format is presented in Section 2, after which the process is explained that was followed to transform the format to be MSDL compatible (Section 3). The most important criteria for the newly defined MSDL-based scenario language are firstly to support all requirements for scenario definition of the C2 simulation environment. Secondly to be fully compatible with MSDL, but only to support the parts that is necessary for scenario definition in the C2 simulation environment. It may be extended to support more aspects of MSDL in future.

2. Existing C2 Simulation Scenario Definition Format

It should be noted that the initial aim of the definition format was not to be a generic air defence scenario language, but rather a customised, once-off definition for specific simulation and end-user environments. This resulted in some design choices that may not be optimal for generic scenario definition language compatibility.

The existing scenario definition format is rather referred to as a format than a language, as it does not provide the capability to convey new information, but only to specify the configuration of a fixed set of entities. A language is defined by a set of symbols and a grammar that governs the use of the symbols to convey information [6].

[1] The Council for Scientific and Industrial Research (CSIR) has been constituted by an Act of the South African Parliament in 1945. It is one of the leading scientific and technology research, development and implementation organisations in Africa. The organisation undertakes and applies directed research and innovation in science and technology to improve the quality of life of the country's people.

2.1 C2 Scenario Contents Overview

The existing C2 simulation scenario definition format is based on the Extensible Mark-up Language (XML) [7]. Aspects that are addressed by the existing format are summarised in Table 1.

Whereas some of the elements in Table 1 only address a single entity, aspect or uniform type, such as "Terrain", others such as "Battery" embodies lists of different types. Battery elements may include short range missile launchers, but also tracking radars, for example. Common attributes between types are used as far as possible, but type-specific attributes are also catered for. It was attempted with the present scenario definition format to capture both the military and simulation scenario definitions in one definition, which explains an item such as Execution in Table 1.

Table 1: Existing Scenario Definition Format Elements

Category	Description
Metadata	Captures version information and author details.
Defended Assets	Indicates position, type, criticality and boundaries associated with each asset. Multiple assets may be defined.
Battery	All equipment, including effectors, sensors and operator terminals are captured. It includes organisation, type, affiliation and areas of responsibility. Configurations and activations to some external systems are included as well.
Threat	All aircraft, irrespective of affiliation are specified here. References are made to flight profiles, stored in a separate configuration. This allows "libraries" of threat profiles to be used and reused.
Air zones	To define prohibited and restricted air zones that aid target hostility classification (tactical doctrine). Tunnels and lanes are also included in this section.
Areas	These are effectively overlays that are scenario specific for visualisation and planning aid.
Sectors	Segments of circles that are used during C2.
Line of sight maps	Scenario specific pre-generated line of sight maps used for planning and visualisation.
Visualisation specifics	Visualisation parameters that are scenario specific.
Terrain	Terrain definition for the scenario.
Execution	Configuration for the distributed or non-distributed execution of a scenario.

Although the existing scenario definition format is applied in a tactical C2 simulation environment, the organisation and definition of operators (humans) are not covered – these aspects are implied by the equipment, e.g. when a fire control system weapon terminal is encountered, the relevant operators are automatically created with the relevant organisation, communication networks and air defence control.

2.2 Encoding

All of the elements in Table 1 are encoded using XML, except for two. Although most threat attributes are encoded in XML, the actual profile (flight path) is encoded in a non-XML text-based format. The terrain specification is also specified in a non-XML text format.
A simplified XML encoding scheme is used without declarations, character encoding or external (XML) dependencies. Elements with attributes and child elements are used. Element content is not used to simplify reading and decoding of XML elements. Empty elements are allowed. Although the encoding scheme is well-formed, it does not result in valid XML documents since no explicit XML schema is used. This is not necessarily an optimal choice, since it requires careful use of the format in terms of structure and content. This could be done, as the development team was small and scenario definitions not overly complex. An example XML snippet is shown for the definition of a Very Short Range Air Defence Missile System (VSHORADS) in Figure 1.

```
<ELEMENT        name="VSHORADS3"
                unit_name="VFU3"
                adc_type="VSHORADS"
                force="FRIENDLY"
                latitude_deg="-33.9430990472878"
                longitude_deg="18.428644946806"
                height_aglm="1.8"
                heading_deg="252"
                pitch_deg="0"
                oem_type="STARSTREAK"
                ammo_type="vshorad"
                envelope_factors="1, 1, 1, 1"
                layer_name="">
        <FIRE_ARCHS>
                <FIRE_ARCH      name="Fire"
                                heading_deg="252"
                                full_angle_deg="95"
                                elevation_angle_deg="3
                                0"
                                range_m="5500" />
                </FIRE_ARCHS>
        <SURVEILANCE_ARCHS>
                <SURVEILANCE_ARCH
                        name="Surveillance"
                        heading_deg="252"
                        full_angle_deg="95"
                        elevation_angle_deg="30"
                        range_m="7000" />
        </SURVEILANCE_ARCHS>
</ELEMENT>
```

Figure 1: XML Snippet in Existing Definition Format

As can be seen from Figure 1 child elements are used to group information and that element attributes are used to encode all parameters values. The VSHORAD battery element is shown with its fire and surveillance arcs. Due to the fact that a document scheme is not used, parameter type information (units) is not encoded optimally. The parameter units are indicated as part of the attribute

names. This is not efficient since any change of the parameter unit will result in a change of the attribute name, limiting interoperability. Since the existing XML scenario description was not initially aimed at being a generic air defence language, it did not present a problem.

2.3 Scenario Definition Format Context

Figure 2 depicts the existing scenario definition format in context of dependencies and processes for the C2 simulation environment.

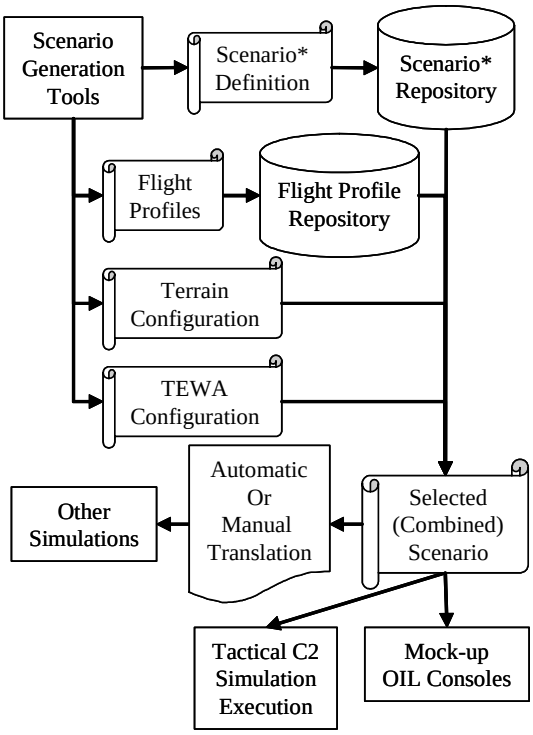

Figure 2: Existing Format Scenario Generation Process (* Indicates Partial Definition)

With the present setup, a complete scenario is defined by a partial scenario definition (mostly ground-based elements), flight profiles associated with threats (aircraft), terrain configuration and Threat Evaluation and Weapon Assignment (TEWA) configuration. The terrain and TEWA configurations are not stored in repositories, but are kept with the rest of the scenario definitions. The flight profiles and partial scenario definitions are kept in informal repositories (not databases) for reuse between different scenarios.

3. Process Followed

To ensure that the MSDL-based scenario definition language fully supports the existing format, the translation effort has been conducted using the existing format. This means that for each item in the existing format, a suitable equivalent mechanism of definition is searched for in the MSDL language.

To determine which items in Table 1 have to be encoded using MSDL, some factors have to be considered. The first is to decide which items remain constant between different scenarios. This may be for all scenarios or specific subsets of scenarios. Secondly, items that remain the same, irrespective of the physical location of the scenario should be considered. Thirdly, items that may be encoded with other, more appropriate definition languages, have to be selected. Lastly, items that are not essential to scenarios, or that will not be applicable to other simulations, have to be identified.

Although not all of the items in Table 1 may be essential to a scenario definition, it is still preferable to group them with a specific scenario. Such information may be visualisation settings or terrain overlays. Two alternatives to group essential MSDL-based scenario definitions with other peripheral scenario information is to at least have references in the main scenario file to the other applicable files, or to create an aggregated scenario definition which in essence contains references to all applicable scenario definition parts, including an MSDL-based scenario file. In any case, both techniques allow mixing and matching of different parts, i.e. a main scenario can be used with different visualisation setting files.

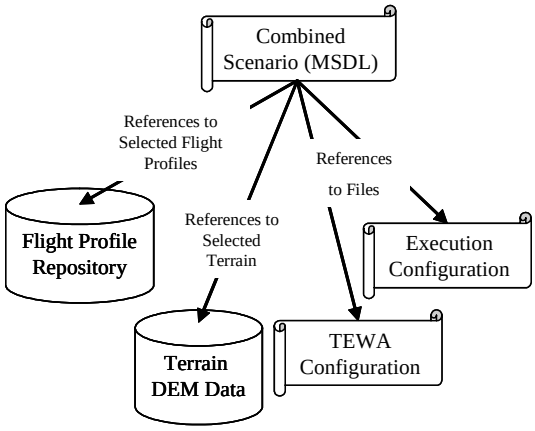

Figure 3: Combined MSDL-based Scenario with References

Figure 3 depicts a scenario definition where the MSDL-based file contains references to other configurations that are not encoded using MSDL. Figure 4 presents a slightly modified version of that shown in Figure 3. The configuration in Figure 4 is preferable since it allows mixing and matching of different scenario parts by just changing references. The one in Figure 3 may also allow a similar setup, given MSDL provides adequate definition mechanisms for external configuration references.

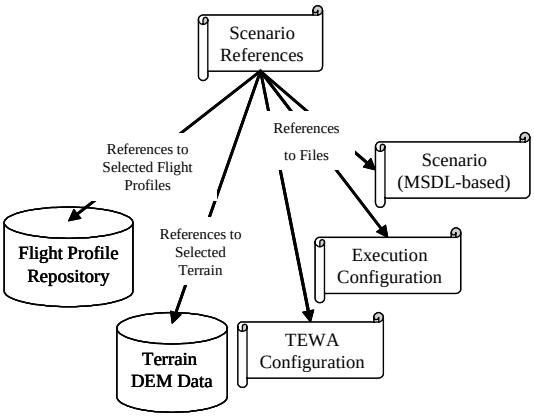

Figure 4: Scenario References Alternative

4. MSDL Overview

Before mapping the existing scenario definition format to MSDL a short overview of the elements covered by MSDL are presented. The MSDL schema supports the definition of elements as shown in Table 2 [1].

Table 2. Primary MSDL Elements (Rephrased from [1])

MSDL Element	Description
Options	Used to specify task organisations, data standards and application specific options.
Plans	Contains scenario descriptive information and executable courses of action.
Environment	Covers scenario time, terrain and weather data.
Force Structure	All participating forces are defined with their respective structures, including associations.
Task Organisations	To define equipment and units. Equipment generally relates to simulated entities and units to the simulated forces. Communication network and unit-equipment relations are also specified.
Installations	Identifies military installations and symbol modifiers.
Overlays	Dedicated and custom (user-defined) overlays are supported. It provides a mechanism to link tactical graphics to specific layers that may be displayed.
Tactical Graphics	Provide for the definition of control measures in MSDL. Graphics can be linked to specific overlays.
MOOTW Graphics	Similar to tactical graphics but to for non-war operations, such as peace keeping.
Threat	To specify non-military threats that links with MOOTW.

5. Mapping the Existing Scenario Definition Format to MSDL

In order to map the existing format to a MSDL compatible format, each item listed in Table 1 needs to be considered. The following subsections present the outcome of each mapping.

5.1 Metadata

Metadata not only captures scenario author information, date and time aspects, but also version information, used for maintaining backwards compatibility. The scenario definition language (in this case MSDL) itself, the scenario version, the version of the simulation that supports the scenario, and other important peripheral applications, such as simulation pre and post processing tools all have different versions that need to be managed.

Table 3: Metadata data storage alternatives

Alternative	Advantage	Disadvantage
XML Prolog	Allows user-definable tags.	Software and XML readers may have problems reading non-standard descriptors.
{Options | Applications | **AppOptions**}	A more structured method of storing metadata that forms part of the existing MSDL version. Minor extensions to the MSDL might accommodate all metadata requirements.	Limited to metadata per Application relevant to the MSDL-based scenario. General metadata cannot be stored here.
Extend MSDL	A structured, controlled way of ensuring minimum requirements are met.	The standard needs to be extended.

MSDL does not explicitly provide for metadata in its schema, although the MSDL version may be specified in the Options element. Alternatives to consider for storing scenario metadata are as part of the XML prolog section, which is basically everything before the root node. Secondly, application specific options {Options | Applications | AppOptions | **Application Schema**}[2] can be used. The Application Schema is not parsed and validated by means of the MSDL schema, and could therefore contain child elements and attributes to capture metadata of the scenario. This approach fits the requirement well to have

[2] The notation { X | Y | … | **Z** } is used to denote elements and child elements of the XML schema as applicable to MSDL. X is therefore the top (root) element, of which Y is a child element. Z is a child element of Y. The last element is in bold for readability purposes.

version information per application that needs to generate or process the scenario file. Lastly, MSDL may be adapted to include a metadata element. The advantage of the latter approach is that the structure and content of the metadata can be regulated and controlled such that it at least satisfies a minimum requirement. Table 3 compares the three alternatives of storing metadata in an MSDL-based scenario file.

Table 4: Metadata Criticality

Type	Criticality
Scenario Invariance	Scenario specific.
Spatial Relevance	Can differ even for the exact same spatial locations.
Appropriateness of MSDL	Should be supported by MSDL.
Essential Item	Vital for compatibility management.

Due to its importance for managing compatibility between versions, metadata is seen as a critical element in the scenario definition (Table 4).

5.2 Defended Assets

As the C2 scenarios are often executed as constructive simulations, a Threat Evaluation and Weapon Assignment (TEWA) model is used to evaluate incoming threats and assign the necessary fire power given the threat intensity and the assets protected. Minimum requirements are to support multiple defended assets as points and polygons. All defended assets have some common parameters (Table 5), as well as a set of parameters per defended asset (Table 6) in the existing format.

Table 5: Common Parameters for all Defended Assets

Parameter	Comment
Local Warning Line position and radius.	Position specified as latitude-longitude pair in decimal degrees with radius in meters.
Effector Line position and radius	Similar to the Local Warning Line.

Note that assets as referred to in the existing format are not necessarily specific installations or equipment, but rather signify geo-referenced point or area. The specific types of assets may be any arbitrary installations or equipment.

MSDL does not make explicit provision for Defended Assets. The only feasible options to consider are using tactical graphics with user defined overlays with point, line or area graphics and triggers or military installations. Applicable MIL-STD-2525B symbols for point and area defended assets are Point and Area Target, both in {Tactical Graphics | **Fire Support**}[3] the branch of the symbology hierarchy [8].

Table 6: Parameters for each Defended Asset

Parameter	Comment
Name	User definable name – has some limitations on length and allowable characters.
Type	Point or area.
Criticality	Medium (0), high (1) or Extreme (2).
Weapon Release Line (of the threat)	Distance in meters
Keep out boundary	Distance in meters
Closest point of approach Action Distance	Distance in meters
Coordinate (point asset) or list of coordinates (area asset)	Coordinates specified as latitude-longitude pairs in decimal degrees with a number per pair.

Possible MIL-STD-2525B equivalents for defended asset parameters that are used in conjunction with overlays and tactical graphics are shown in Table 7. However, none of these provide a way of encoding all the parameters indicated in Table 6 and 7, explicitly, as defended assets.

Table 7: Battery Equipment Parameters

Arc Type	MIL-STD-2525B Equivalent
Local Warning Line	{Tactical Graphics | Command and Control and General Maneuver | Defense | Areas | **Engagement Area**}, {Tactical Graphics | Command and Control and General Maneuver | Special | Areas | **Area of Operations**} or {Tactical Graphics | Fire Support | Areas | **Fire Support Coordination Line**, **Coordinated Fire Line** or **Final Protective Fire**}
Effector Line	{Tactical Graphics | Command and Control and General Maneuver | Offense | Lines | **Probable Line of Deployment**}
Weapon Release Line	{Tactical Graphics | Command and Control and General Maneuver | Defense | Areas | **Attack Position**} or {Tactical Graphics | Command and Control and General Maneuver | General | Points | **Release Point**}
Keep-Out Boundary	{Tactical Graphics | Command and Control and General Maneuver | Offense | Lines | **Limit of Advance**}
CPA Action Distance	{Tactical Graphics | Command and Control and General Maneuver | Offense | Areas | **Penetration Box**}

[3] The {X | Y | … | **Z**} is used for tactical graphics to indicate the position of a symbol in the MIL-STD-2525B hierarchy. X is the top of the hierarchy and Y one of the braches of the hierarchy. Z may be a branch or leave node. The last node is in bold for readability purposes.

By denoting defended assets as graphics, all relevance and meaning associated with them are lost. The definition of defended assets is seen as a critical element of scenario definitions, as it forms an inherent part of defensive military scenarios (Table 8).

Table 8: Defended Assets Criticality

Type	Criticality
Scenario Invariance	Scenario specific
Spatial Relevance	Can differ even for the exact same spatial locations.
Appropriateness of MSDL	Should be supported by MSDL.
Essential Item	Vital for constructive simulation executions.

It is suggested that MSDL is extended to provide for defended assets. Although the primary application of the C2 simulation environment is air defence, MSDL should be extended in such a way to provide for other operations as well, including offensive and defensive. It should also be possible to link the definition of defended assets to options, force structure, task organisations, installations, overlays, graphics and planning. Triggers and events should also be extended to provide for actions with regard to defended assets. As an interim measure, a separate configuration file may be used to define defended assets or the use of tactical graphics as shown in Table 7.

5.3 Battery

The battery element of the existing scenario definition format captures some air defence equipment. Sensors, effectors, terminals and external systems (data sources, simulators and/or simulations) are indicated as well. A key aspect of the existing format is that a fair amount of scenario definition is implied by specific entries. The C2 simulation environment is pre-programmed to associate auto-generated elements with specified elements in the scenario definition. To illustrate: All operators (modelled, human operators) of equipment are automatically created by the simulation when the relevant equipment is detected in the scenario definition. Table 9 lists the parameters for equipment.

Battery equipment is the most important elements to specify in a scenario as it directly relates to the entities in a simulation. Having no explicit MSDL support for Original Equipment Manufacturer types and complete initial orientation (pitch is not supported) is problematic. Exporting scenario definitions to other simulations may work correctly, but importing will require additional information to be supplied.

Table 9: Battery Equipment Parameters

Parameter	Comment	MSDL Equivalent
Name	User definable name – has some limitations on length and allowable characters.	{TaskOrganization | Equipment | EquipmentItem | **Name**}
Unit name	Organisational unit	{TaskOrganixation | Equipment | EquipmentItem | **CommandRelation**}
Air Defence Type	Item selected from pre-defined type list	{Task Organixation | Equipment | EquipmentItem | **Enumeration**}
Original Equipment Manufacturer Type	Model number or name if the Air Defence Type are the same and different makes of the same equipment have to be used.	No MSDL equivalent in {TaskOrganization | Equipment | **EquipmentItem**} structure
Force	The force (own, enemy) controlling the equipment.	Specified via the force affiliation of the unit commanding the equipment
Position	Latitude-longitude-above ground triplet.	{Task Organixation | Equipment | EquipmentItem | **Location** }
Initial Orientation	Heading-pitch pair	Only the heading can be specified, not the pitch.
Arcs	All arcs, including Fire, Surveillance, Area-of-responsibility, can't fire and can't see. All defined by a heading, range elevation and azimuth interval. Multiple arcs may be defined.	No explicit support, but tactical graphics with overlays may be used. User-defined overlays will be necessary to provide for all types of arcs. Translation between tactical graphic symbology and arc types necessary.
Equipment specific parameters	May be encoded in an external configuration file.	Not supported
TEWA Specific parameters	May be encoded in an external configuration file.	Not supported

Possible MIL-STD-2525B equivalents for arc types that are used in conjunction with overlays and tactical graphics are shown in Table 10.

Table 10: Battery Equipment Parameters

Arc Type	MIL-STD-2525B Equivalent
Fire	{Tactical Graphics | Command and Control and General Maneuver | Defense | Areas | **Engagement Area**} or {Tactical Graphics | Fire Support | Areas | **Free Fire Area** or **Fire Support Area**}
Surveillance	{Tactical graphics | C2 and General Maneuver | General | Areas | **General Area**} Use naming convention to indicate arc types
Area of Responsibility	
Can't See	
Can't Fire	

5.4 Threat

The threat section of MSDL is aimed at non-military threats and threat organisations, whereas in the existing format, it is aimed at defining threat aircraft attacking the defended asset(s). Internal threats, that is, threats generated within the C2 simulation environment, are handled in the threat section of the existing format, but threats based on external, online data sources, simulators or simulations are not listed in the threat section. Threats therefore have to be specified as equipment commanded by opposing forces in a similar way than battery equipment. An additional requirement is to store threat flight paths, or references to path definitions. Although a direction of movement may be specified, {TaskOrganization | Equipment | EquipmentItem | **DirectionOfMovement**}, it is not adequate to specify attack profiles.

An alternative is to specify aircraft flight paths by applying user-defined overlays with the tactical graphics. The flight path points may be specified as MIL-STD-2525B {Tactical graphics | C2 and General Maneuver | General | Points | **Waypoint**} anchor points, which have accurate locations (latitude, longitude and elevation above ground level). Aspects still lacking, are to associate a flight path, or series of anchor points with an aircraft (equipment item) and to specify behaviour (timing, speed, etc) at each anchor point. Table 11 lists threat (aircraft) parameters as used in the existing format with possible MSDL equivalents.

Table 12 lists the parameters per waypoint associated with the aircraft. Note if the aircraft state data option is used, the full three dimensional state (position, velocity, acceleration and orientation) as a function of time is used. The parameters in Table 12 may be seen as aircraft model specific and may therefore be rather encoded in an external, referenced file.

Table 11: Threat (Aircraft) Parameters

Parameter	Comment	MSDL Equivalent
Name	User definable name – has some limitations on length and allowable characters.	{TaskOrganizations | Equipment | EquipmentItem | **Name**}
File	Way points or aircraft state data	Anchor points may be used for Way points or external, referenced file for aircraft state data.
Type	Fixed, Rotary Wing, Missile, UAV etc.	{TaskOrganizations | Equipment | EquipmentItem | **Enumeration**}
Force	The force (own, enemy) controlling the equipment.	Specified via the force affiliation of the unit commanding the equipment.
Attack Profile	Input to automatic TEWA model (Profile type).	No explicit option.
Munitions	Munitions associated with the aircraft that may be launched.	To be handled the same as threats (aircraft) with implied or naming convention association.
Time extensions (start and end)	Straight path extensions based on time at the start and end of a flight profile.	No explicit option.
Creation delay time	The aircraft only becomes active in the simulation after a selected delay.	No explicit option.
Active at simulation initialisation	Aircraft may be activated at run-time with the relevant interface.	No explicit option.

Threat definitions are critical elements in air defence and tactical C2 simulations, therefore external, referenced files have to be used in the interim. Parameters that are model-specific, such as the maximum accelerations for waypoints will be included in such files.

Table 12: Waypoint Parameters

Parameter	Comment	MSDL Equivalent
Type	Fixed or Rotary Wing waypoint	No explicit option
Position	Latitude, longitude and altitude (above sea level)	{Tactical graphics | C2 and General Maneuver | General | Points | **Waypoint**}
Velocity	Scenario specific, thus required in scenario definition.	No explicit option
Maximum Positive, Negative and Lateral Accelerations	Model specific – not critical to scenario definition	No explicit option
Delay Time	For Rotary Wing waypoints. The orientation may then be specified as well. Scenario specific, thus required.	No explicit option

5.5 Air Zones, Areas and Sectors

Air zones, tunnels and lanes are all areas to aid air space control and assist in hostility classification. These areas are critical to the scenario definition and are defined by means of overlays and tactical graphics in conjunction with MIL-STD-2525B. Anchor points may be used to define points, lines or areas. Although MIL-STD-2525B does not explicitly support the area types indicated in Table 13, acceptable translations could be identified.

Areas are user-defined to indicate areas of interest or aids for planning or tactical support. These are supported similarly to air zones, but without specific MIL-STD-2525B symbols, although {Tactical graphics | C2 and General Maneuver | General | Areas | **General Area**} may be used. User-defined overlays with relevant names may be associated with areas. Areas are non-critical but relevant to scenario definitions.

Sectors have C2 tactical implications and are extensively used during planning, deployment and in battle. Sectors are handled in a similar way than areas, as the tactical graphics symbology does not support sectors explicitly.

5.7 LOS Maps

Line-of-sight (LOS) maps are generally pre-generated with a simulation tool and are used as planning and deployment aids. LOS maps are not critical to scenario definitions.

Table 13: Battery Equipment Parameters

Area Type	MSDL / MIL-STD-2525B Equivalent
Prohibited Air Zones	{Tactical Graphics | Command and Control and General Maneuver | Aviation | Areas | **Weapons Free Zone**}
Restricted Air Zones	{Tactical Graphics | Command and Control and General Maneuver | Aviation | Areas | **Restricted Operations Zone**} or {Tactical Graphics | Fire Support | Areas | **Airspace Coordination Area**}
Tunnels	{Tactical Graphics | Command and Control and General Maneuver | Aviation | Lines | **Air Corridor**}, {Tactical Graphics | Command and Control and General Maneuver | Aviation | Areas | **Missile Engagement Zone (Low or High altitude)**} or {Tactical Graphics | Command and Control and General Maneuver | Aviation | Lines | **Standard-Use Army Aircraft Flight Route**}
Lanes (Entry and Exit)	{Tactical Graphics | Command and Control and General Maneuver | Aviation | Lines | **Air Corridor**} or {Tactical Graphics | Command and Control and General Maneuver | Aviation | Lines | **Standard-Use Army Aircraft Flight Route**}

5.8 Visualisation Specifics

Visualisation specifics refer to the settings of two or three dimensional visualisation tools – these are not critical to scenario definitions, but are specific to scenarios. Settings may be stored in an external, referenced file.

5.9 Terrain

The existing scenario definition format allows for the definition of a spherical earth with no undulations, but with a custom ground height, or a spherical earth with terrain elevations. In both cases a rectangular area, with resolution of the latitude-longitude degree boundaries have to be specified. Practically not more than 10 degree-by-degree tiles can be used, and with the terrain elevations option, non-contiguous tiles may be specified. The terrain specification is critical to the scenario definition.

MSDL provides adequate terrain boundary mechanisms in the form of upper-right and lower-left corners. This is sufficient to specify a rectangular terrain, whereas the terrain database itself may be described using a free text name. The latter could be used to differentiate between spherical earth with or without terrain elevations.

5.10 Execution

Execution parameters are simulation execution specific, but also scenario specific. These are not considered critical to the scenario definition therefore it will be stored in an external, referenced file.

6. Future Work

This paper only presents the process and an attempt to translate the existing scenario definition format to an MSDL-compatible one. The first task is then to actually start retrofitting the existing C2 tactical simulation environment with the new format. Some elements covered by the existing format have also not been included in this translation effort – these are mostly applicable to non-air defence scenarios or joint air defence scenarios.

7. Conclusion

Table 14 indicates to what extent MSDL could be used to define scenarios for the GBADS C2 tactical simulation environment with some comments and suggestions.

Table 14: MSDL Applicability to the GBADS C2 Simulation Environment

Existing Format Element	Explicit MSDL Support	Suggestion / Comment
Metadata	No	Extend MSDL.
Defended Assets	No	Extend all relevant MSDL elements. In the interim use a referenced, external file.
Battery	Mostly	Some vital parameters cannot be encoded.
Threat	Partial	Aircraft can be defined, but flight path definitions not explicitly supported. Anchor points (tactical graphics) may be used with Waypoints from MIL-STD-2525B.
Air zones, tunnels, lanes, areas and sectors	Yes	Used in conjunction with MIL-STD-2525B. Translation between symbology and area types necessary. Associated user-defined overlay names are indicative of the area type or meaning.
LOS maps	Yes	Similar to Areas.
Visualisation specifics	No	Non-critical, store in an external, referenced file.
Terrain	Yes	Data source specified as free text name.
Execution	No	Non-critical, store in an external, referenced file.

Although some elements in the existing format could not be mapped to MSDL, elements addressed by MSDL are also not supported in the existing format. Some of these, such as force layout and structure, should be addressed, and not implied, if scenario definition interoperability with other simulations should be maintained. It is also preferable not to use too many user definable fields and elements, since it limits interoperability.

It is appreciated that MSDL may not have been aimed at tactical air defence engagement scenarios, but rather higher-level simulations. These simulations, such as war-gaming and theatre-level simulations, often rely on aggregated entities, rather than detail tactical information, in scenario definitions. In order to support MSDL-based scenario definitions in the C2 tactical simulation environment in the interim, external referenced scenario elements and customised overlays have to be employed with appropriate MIL-STD-2525B symbols.

8. References

[1] “Specifications for: Military Scenario Definition Language (MSDL),” Simulation Interoperability Standards Organization (SISO), Initial Draft, Orlando, April, 2005.

[2] W.H. le Roux: “Implementing a Low Cost Distributed Architecture for Real-Time Behavioural Modelling and Simulation," Proceedings of the 2006 European Simulation Interoperability Workshop, Stockholm, 19-22 June 2006.

[3] S. Naidoo and and J.J. Nel: “Modeling and Simulation of a Ground Based Air Defence System and Associated Tactical Doctrine as Part of Acquisition Support,” Proceedings of the 2006 Fall Simulation Interoperability Workshop, Orlando, September, 2006.

[4] H.J. Baird and J.J. Nel: “The Evolution of M&S as part of Smart Acquisition using the SANDF GBADS Programme as Example” Twelfth European Air Defence Symposium, Shrivenham, June 2005.

[5] R. Oosthuizen: “Doctrine Development during Systems Acquisition and the Benefits of Modelling & Simulation,” Fourth South African Joint Air Defence Symposium, Pretoria, October 2003.

[6] Wikipedia, http://en.wikipedia.org/wiki/Language, Accessed 6 February 2007.

[7] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler and F. Yergeau: Extensible Markup Language (XML) 1.0 - Origin and Goals, Fourth Edition, World Wide Web Consortium, Accessed on 23 January 2007, http://www.w3.org/TR/2006/REC-xml-20060816, October 2006.

[8] “Department of Defense Interface Standard: Common Warfighting Symbology”, MIL-STD-2525B, United States of America, 30 January 1999.

13. Acknowledgements

The author thanks his colleagues at the CSIR for taking the time to review this paper and provide valuable suggestions, as well as the Armaments Corporation of South Africa for supporting this work.

HERMAN LE ROUX has been with the South African Council for Scientific and Industrial Research since April 1998 and is at present a Principal Engineer in the Mathematical and Computational Research Group. He is involved in Modelling and Simulation-based Decision Support, specifically for the South African National Defence Force. Interests include information fusion, biometrics, artificial intelligence and software engineering. Le Roux completed a Masters Degree in Computer Engineering at the University of Pretoria in 1999 and is currently pursuing a PhD in Information Fusion.