NORTH ATLANTIC TREATY
ORGANIZATION

SCIENCE AND TECHNOLOGY
ORGANIZATION

AC/323(MSG-086)TP/562

www.sto.nato.int

STO TECHNICAL REPORT **TR-MSG-086-Part-I**

Simulation Interoperability

(Interopérabilité de la simulation)

Final Report of Task Group MSG-086.

Published January 2015

Distribution and Availability on Back Cover

NORTH ATLANTIC TREATY
ORGANIZATION

SCIENCE AND TECHNOLOGY
ORGANIZATION

AC/323(MSG-086)TP/562

www.sto.nato.int

STO TECHNICAL REPORT **TR-MSG-086-Part-I**

Simulation Interoperability

(Interopérabilité de la simulation)

Final Report of Task Group MSG-086.

The NATO Science and Technology Organization

Science & Technology (S&T) in the NATO context is defined as the selective and rigorous generation and application of state-of-the-art, validated knowledge for defence and security purposes. S&T activities embrace scientific research, technology development, transition, application and field-testing, experimentation and a range of related scientific activities that include systems engineering, operational research and analysis, synthesis, integration and validation of knowledge derived through the scientific method.

In NATO, S&T is addressed using different business models, namely a collaborative business model where NATO provides a forum where NATO Nations and partner Nations elect to use their national resources to define, conduct and promote cooperative research and information exchange, and secondly an in-house delivery business model where S&T activities are conducted in a NATO dedicated executive body, having its own personnel, capabilities and infrastructure.

The mission of the NATO Science & Technology Organization (STO) is to help position the Nations' and NATO's S&T investments as a strategic enabler of the knowledge and technology advantage for the defence and security posture of NATO Nations and partner Nations, by conducting and promoting S&T activities that augment and leverage the capabilities and programmes of the Alliance, of the NATO Nations and the partner Nations, in support of NATO's objectives, and contributing to NATO's ability to enable and influence security and defence related capability development and threat mitigation in NATO Nations and partner Nations, in accordance with NATO policies.

The total spectrum of this collaborative effort is addressed by six Technical Panels who manage a wide range of scientific research activities, a Group specialising in modelling and simulation, plus a Committee dedicated to supporting the information management needs of the organization.

AVT	Applied Vehicle Technology Panel
HFM	Human Factors and Medicine Panel
IST	Information Systems Technology Panel
NMSG	NATO Modelling and Simulation Group
SAS	System Analysis and Studies Panel
SCI	Systems Concepts and Integration Panel
SET	Sensors and Electronics Technology Panel

These Panels and Group are the power-house of the collaborative model and are made up of national representatives as well as recognised world-class scientists, engineers and information specialists. In addition to providing critical technical oversight, they also provide a communication link to military users and other NATO bodies.

The scientific and technological work is carried out by Technical Teams, created under one or more of these eight bodies, for specific research activities which have a defined duration. These research activities can take a variety of forms, including Task Groups, Workshops, Symposia, Specialists' Meetings, Lecture Series and Technical Courses.

The content of this publication has been reproduced directly from material supplied by STO or the authors.

Published January 2015

ISBN 978-92-837-0201-2

Table of Contents

List of Figures

List of Tables

MSG-086 Membership List

Nation	Full Name	Contact Information	Appointed by Nation
AUS	Johannes LÜTHI	FH Kufstein Tirol Bildungs GmbH Andreas Hofer-Strasse 7 6330 Kufstein AUSTRIA Phone: +43 5372 71819 303 Email: Johannes.Luethi@fh-kufstein.ac.at	NO
DEU	Michael BERTSCHIK	WTD 91 / GF 510 Am Schießplatz 49176 Meppen GERMANY Phone: +49 5931 43 1590 Email: MichaelBertschik@bundeswehr.org	NO
DEU **Chair**	**Matthias HAHN**	Planungsamt der Bundeswehr Oberspreestrasse 61L 12439 Berlin GERMANY Phone: +49 30 6794 - 1977 Email: matthias3hahn@bundeswehr.org	YES
DEU	Eckehard NEUGEBAUER	IABG mbH Postfach 1764/Schiessplatz 49707 Meppen GERMANY Phone: +49 59 327 343 42 Email: eckehard.neugebauer@kabelmail.de	YES
DEU	Martin ROTHER	IABG mbH Einsteinstrasse 20 85521 Ottobrunn GERMANY Phone: +49 89 6088 3974 Email: rother@iabg.de	YES
DEU Co-Chair	Robert SIEGFRIED	aditerna GmbH Otto-Hahn-Str. 13 B 85521 Riemerling GERMANY Phone: +49 89 608 755 825 Email: robert.siegfried@aditerna.de	YES
DEU	Dieter STEINKAMP	IABG mbH Ferdinand-Sauerbruch-Str. 26 56073 Koblenz GERMANY Phone: +49 261 94729-71 Email: dieter-steinkamp@gmx.de	NO

Nation	Full Name	Contact Information	Appointed by Nation
DEU	Stefan VRIELER	WTD 91 / GF 510 Am Schießplatz 49176 Meppen GERMANY Phone: +49 5931 43 2226 Email: StefanVrieler@bundeswehr.org	YES
GBR	Steve FRANK	MBDA UK Ltd 5200 Building Six Hills Way Stevenage, Herts SG1 2DA UNITED KINGDOM Phone: +44 1438 754 271 Email: steve.frank@mbda-systems.com	YES
GBR	Neil MORRIS	Dstl Portsdown West PCS/AES Portsdown Hill Road Fareham, Hampshire PO17 6AD UNITED KINGDOM Phone: +44 02392 532 936 Email: nmorris@dstl.gov.uk	YES
ITA	Valeria FONTANA	SELEX ES SpA Via Tiburtina 1231 00131 Rome ITALY Phone: +39 6 4150 5731 Email: valeria.fontana@selex-es.com	YES
NLD	Arno GERRETSEN	National Aerospace Laboratory NLR Training, Simulation and Operator Performance Dept P.O. Box 90502 Anthony Fokkerweg 2 1059 CM Amsterdam NETHERLANDS Phone: +31 88 511 3231 Email: arno.gerretsen@nlr.nl	YES
NLD	Michel KEUNING	National Aerospace Laboratory NLR Training, Simulation and Operator Performance Dept P.O. Box 90502 Anthony Fokkerweg 2 1059 CM Amsterdam NETHERLANDS Phone: +31 88 511 3155 Email: keuning@nlr.nl	YES

Nation	**Full Name**	**Contact Information**	**Appointed by Nation**
ROU	Catalin CIUL	Military Equipment and Technologies Research Agency Aeroportului Street, No 16 CP 19 OP BRAGADIRU 077025 Ilfov ROMANIA Phone: +40 21 423 30 58 Email: cciul@acttm.ro	YES
SWE	Fredrik JONSSON	Swedish Defense Materiel Administration FMV, Banérgatan 62 SE-115 88 Stockholm SWEDEN Phone: +46 878 255 34 Email: fredrik.jonsson@smart-lab.se	YES
SWE	Peter KARLSSON	Pitch Technologies AB Repslagaregatan 254 SE-58222 Linköping SWEDEN Phone: +46 13 470 5517 Email: peter.karlsson@pitch.se	YES
SWE	Björn LÖFSTRAND	Pitch Technologies AB Repslagaregatan 25 58222 Linköping SWEDEN Phone: +46 13 470 5500 Email: bjorn.lofstrand@pitch.se	YES
SWE	Lennart OLSSON	Pitch Technologies AB Repslagaregatan 25 58222 Linköping SWEDEN Phone: +46 13 470 5500 Email: lennart.olsson@pitch.se	YES
TUR	Mahmut Nedim ALPDEMIR	TÜBITAK BILGEM ILTAREN Sehit Yzb Ilhan, TAN Kislasi, Umit Mah 2432 Cad, 2489 Sok Umitkoy 06800, Ankara TURKEY Phone: +90 312 291 6084 Email: nedim.alpdemir@tubitak.gov.tr	YES
TUR	Kurtulus BEKTAS	Turkish Navy Denis K. K.ligi APGE ve BILKARDES Bsk.ligi 06650 Bakanliklar/Ankara TURKEY Phone: +90 312 403 3649 Email: bektas.k5781@dzkk.tsk.tr	YES

Nation	Full Name	Contact Information	Appointed by Nation
TUR	Mesut GUNEY	Turkish General Staff Genelkurmay BILKARDEM Bsk.ligi 06650 Bakanliklar, Ankara TURKEY Phone: +90 312 402 3842 Email: mguney@tsk.tr	YES
TUR	Ahmet KARA	TÜBITAK BILGEM ILTAREN Ümit Mah. 2432. Cad. 2489 Sok. Ümitköy 06800 Ankara TURKEY Phone: +90 312 291 6061 Email: ahmet.kara@tubitak.gov.tr	NO
TUR	Halit OGUZTÜZÜN	Middle East Technical University Department of Computer Engineering Inonu Bulvari 06531 Ankara TURKEY Phone: +90 312 210 5587 Email: oguztuzun@metu.edu.tr	YES

Simulation Interoperability

(STO-TR-MSG-086-Part-I)

Executive Summary

As simulation interoperability was the highest priority capability gap of the NMSG MORS M&S Gap Analysis Questionnaire an Exploratory Team (ET-027) of the NMSG was formed in 2009 to investigate simulation interoperability. ET-027 identified 63 issues which severely limit simulation interoperability. Based on the findings of ET-027, MSG-086 "Simulation Interoperability" was initiated in 2010 and tasked to analyse the ET-027 interoperability issues. This analysis should be used to recommend and prototype information products augmenting the Distributed Simulation Engineering and Execution Process (DSEEP) [DSEEP] to mitigate or obviate identified interoperability issues. In total, 46 interoperability issues are identified and described by MSG-086 and categorized into 9 interoperability issue groups. All issues are documented according to the same schema and relations to relevant M&S standards are identified.

Present standards for distributed simulation environments mostly focus on the technical, the syntactic and (to a limited extent) on the semantic interoperability level. To meet future demands – especially in terms of quality and reduced time and costs – simulation interoperability at higher levels (i.e., on the pragmatic, dynamic, and conceptual interoperability level) is required as well as substantial automation of development, integration, and execution of distributed simulation environments.

This requires standardization of information products created in the process of developing a simulation environment, following e.g., the DSEEP. Such standardized information products must address higher levels of interoperability than present simulation interoperability standards which focus on lower interoperability levels (i.e., on technical, syntactic, and semantic issues).

Based on the identified interoperability issues and their impact on the simulation environment engineering process, MSG-086 focused its further efforts on the issue group "Scenario". To improve simulation interoperability in context of the DSEEP, MSG-086 developed a "Guideline on Scenario Development for (Distributed) Simulation Environments". This guideline augments the DSEEP with regards to scenario development and proposes content and structure of an information product for scenario specification.

Additionally, MSG-086 delivers the following secondary outcomes and deliverables:

- Recommendations for updating AMSP-01 [AMSP-01] with respect to simulation interoperability and scenario development.
- A proposal for updating the NMSG Military Operational Requirements Sub-group (MORS) M&S Gap Analysis Questionnaire.
- Recommendations for SISO working groups (e.g., DSEEP, FEAT, and SCM), especially with regards to scenario development.

A major finding of MSG-086 (besides the detailed documentation of simulation interoperability issues) is that simulation interoperability is not primarily a technical issue, but that simulation interoperability needs to be addressed in a holistic way along the whole simulation environment engineering process. Achieving simulation interoperability requires efforts and standardization on the technical, the syntactic, the semantic, and the pragmatic level. Focusing only on standards for distributed systems or reuse of components will not lead to simulation interoperability on higher levels.

Based on these conclusions, MSG-086 worked out proposals for future activities within the NMSG and for cooperative actions with SISO. One of the most promising approaches towards improving simulation interoperability is a service-oriented approach, commonly referred to as “M&S as a Service”. It is recommended to continue initial work done by MSG-131 (“Modelling and Simulation as a Service: New Concepts and Service-Oriented Architectures”) and to investigate the potential of M&S as a Service by a dedicated Task Group. Regarding cooperative action with SISO the recommendation is to transform the “Guideline on Scenario Development for (Distributed) Simulation Environments” into an official SISO standard.

Interopérabilité de la simulation

(STO-TR-MSG-086-Part-I)

Synthèse

Puisque le questionnaire d'analyse des lacunes de M&S du MORS du NMSG a déterminé que le manque d'interopérabilité de la simulation constituait la lacune prioritaire à combler en matière de capacité, une équipe exploratoire (ET-027) a été constituée au sein du NMSG pour étudier l'interopérabilité de la simulation. L'ET-027 a identifié 63 problèmes qui limitent fortement l'interopérabilité de la simulation. Le MSG-086 « Interopérabilité de la simulation » a été créé en 2010 sur la base des conclusions de l'ET-027 ; sa mission est d'analyser les problèmes d'interopérabilité mis au jour par l'ET-027. Cette analyse devait servir à recommander et créer des prototypes d'applicatifs qui améliorent les processus de création et réalisation de simulation distribuée (DSEEP, *Distributed Simulation Engineering and Execution Process* [DSEEP]) afin d'atténuer ou éviter les problèmes d'interopérabilité identifiés. Au total, le MSG-086 identifie et décrit 46 problèmes d'interopérabilité et les classe en neuf groupes. Tous les problèmes sont documentés selon le même schéma ainsi que leur lien avec les normes de M&S associées.

Les normes actuelles relatives aux environnements de simulation distribuée se concentrent principalement sur les niveaux technique, syntaxique et (dans une certaine mesure) sémantique de l'interopérabilité. Pour répondre aux besoins futurs, notamment en termes de qualité et de réduction des délais et des coûts, l'interopérabilité de la simulation doit avoir lieu à des niveaux plus élevés (autrement dit, aux niveaux pratique, dynamique et conceptuel) de même que l'automatisation relative du développement, de l'intégration et de la mise en œuvre des environnements de simulation distribuée.

Cela exige une normalisation des applicatifs créés pendant le développement d'un environnement de simulation, à la suite, par exemple, du DSEEP. Ces produits normalisés doivent répondre à des niveaux d'interopérabilité supérieurs aux normes actuelles d'interopérabilité de la simulation, qui se concentrent sur des niveaux inférieurs (autrement dit, les questions techniques, syntaxiques et sémantiques).

En partant des problèmes d'interopérabilité identifiés et de leur impact sur le processus d'ingénierie de l'environnement de simulation, le MSG-086 a concentré ses efforts sur la problématique liée au « scénario ». Afin d'améliorer l'interopérabilité de la simulation dans le contexte du DSEEP, le MSG-086 a rédigé un « Guide au développement des scénarios dans les environnements de simulation (distribuée) ». Ce guide augmente le DSEEP relativement à l'élaboration des scénarios et propose un contenu et une structure d'applicatif pour la spécification des scénarios.

En outre, le MSG-086 délivre les résultats secondaires et documents suivants :

- Recommandations de mise à jour de l'AMSP-01 [AMSP-01] relativement à l'interopérabilité de la simulation et au développement des scénarios.
- Proposition de mise à jour du questionnaire d'analyse des lacunes du Sous-groupe des besoins opérationnels militaires (MORS) du NMSG.
- Recommandations aux groupes de travail de la SISO (par exemple, DSEEP, FEAT et SCM), en particulier au sujet du développement des scénarios.

L'une des grandes conclusions du MSG-086 (en plus de la documentation détaillée sur les problèmes d'interopérabilité de la simulation) est que l'interopérabilité de la simulation n'est pas fondamentalement une question technique, mais qu’elle doit être abordée de manière holistique tout au long du processus d’ingénierie de l’environnement de simulation. L’interopérabilité de la simulation exige du travail et une normalisation aux niveaux technique, syntaxique, sémantique et pragmatique. Il ne suffit pas de se focaliser sur les normes des systèmes distribués ou sur la réutilisation de composants pour améliorer l’interopérabilité de la simulation.

A partir de ces conclusions, le MSG-086 propose des activités futures au sein du NMSG et des actions en coopération avec la SISO. L’une des approches les plus prometteuses pour améliorer l’interopérabilité de la simulation est une approche axée sur le service fourni, couramment appelée « M&S en tant que service ». Il est recommandé de poursuivre le travail initial accompli par le MSG-131 (« Modélisation et simulation en tant que service, de nouveaux concepts et de nouvelles architectures axées sur le service ») et de confier à un groupe de travail spécial l’étude du potentiel de la M&S en tant que service. En ce qui concerne les actions en coopération avec la SISO, il est recommandé de transformer le « Guide au développement des scénarios dans les environnements de simulation (distribués) » en une norme officielle de la SISO.

Chapter 1 – INTRODUCTION

1.1 MOTIVATION

Simulation interoperability is the highest priority capability gap of the NMSG MORS M&S Gap Analysis Questionnaire [MORS Gap List], which was originally based on the US M&S CCBP [US CCBP].

The capability for simulation interoperability is required to enable the use of M&S across NATO and the Nations for joint and combined distributed simulation application across all application domains as envisioned in the NATO M&S Masterplan [NATO_MSMP], the ACT M&S vision paper [ACT MS Vision], and the NMSG Strategy and Business Plan [NMSG_SBP]. Therefore this capability has to be considered as a fundamental pre-requisite for network-enabled capabilities (NEC) in the frame of NATO transformation. This was clearly reflected by the prioritization of the gaps listed in the MORS M&S Gap Analysis Questionnaire in a survey which was conducted across all NATO Nations and ACT.

As a consequence, at the 21st NMSG business meeting in Ljubljana (May 2008) it was decided to form an Exploratory Team (ET-027) to develop a TAP/TOR "Simulation Interoperability" under consideration of the activities A-SIM-1, A-SIM-5 and A-SIM-7 of the MORS M&S Gap Analysis Questionnaire:

- **A-SIM-1**: Determine the requirements for LVC technical interoperability across all NATO Nations. Ensure that interoperability with Global Integrated Grid concepts is addressed.
- **A-SIM-5**: Identify the alternatives for specifying and sharing semantic interoperability specifications appropriate for use in integrating LVC environments. The alternatives should consider existing standards and whether they would need to be customized for applicability to LVC interoperability.
- **A-SIM-7**: Determine and prioritize the requirements for readily available and/or persistent LVC environments within and across the NATO.

The results of ET-027 were:

- The gaps and related activities regarding simulation interoperability as given/proposed by the MORS M&S Gap Analysis Questionnaire are just scratching at the surface of the problem.
- Up to now (and utilizing the available standards) "true" simulation interoperability in terms of fully logically consistent distributed simulation environments is not a priori (if at all) achievable.
- Interoperability between information technology systems (i.e., also between simulation systems, C3I systems etc.) – that means the communication between machines – is a complex topic and requires a well-structured description.
- The six-level structure as elaborated by Tolk et al. [Tolk2006] should be used as basis for further research regarding simulation interoperability.
- Focus of present standards for distributed simulation environments is basically on the technical, syntactic and – to a limited extent – on the semantic interoperability level. To enable fair fight conditions in future simulation environments and for simplification (decrease of time and effort/ costs for experts meetings, special agreements, special bridging and middleware developments) and automation of the development of future distributed simulation environments, interoperability at higher levels (pragmatic, dynamic, and conceptual level) is required.
- Aspects of higher level simulation interoperability (i.e., pragmatic, dynamic, conceptual interoperability) are not addressed in the present MORS M&S Gap Analysis Questionnaire.
- A relevant step towards higher levels of simulation interoperability is the standardization of information products created in the process of developing a simulation environment, e.g., the Distributed Simulation Engineering and Execution Process (DSEEP, IEEE Std 1730).

Such standardized information products have to consider the modelling domain in contrast to present simulation interoperability standards which focus on lower interoperability levels.

1.2 OBJECTIVE

ET-027 identified 63 issues potentially hampering simulation interoperability at various levels. Based on these findings ET-027 recommended a TAP for a follow-on Task Group (MSG-086) which is outlined in the following.

1.2.1 Area of Research and Scope

Investigation of current simulation interoperability issues and of the feasibility of standardized information products in frame of the DSEEP under specific consideration of the modelling domain in contrast to present simulation interoperability standards focusing on the simulation domain.

1.2.2 Specific Goals

- Get common understanding of simulation system interoperability and of the structure of interoperability.
- Get common understanding of interoperability aspects related to the different levels of interoperability.
- Propose content and structure of required information products and determine the relation between these information products as described in the DSEEP to support interoperability at all levels.
- Get common understanding of the development of these information products by providing an example prototype.

1.2.3 Covered Topics

- Describe simulation interoperability issues starting with ET-027 list of issues.
- Allocate these issues to a structure to be determined.
- Identify corresponding information products related to the DSEEP (e.g., objectives, conceptual model, scenarios).
- Define selected information products independent of the implementation (list of contents and structure) with specific focus on the higher levels of interoperability.
- Describe relations between information products (in context with DSEEP) (e.g., in the form of a flow diagram).
- Look at existing implementations and compare to defined information products in order to derive recommendations.
- Recommend information products (may be standard to be submitted to MS3, SISO, Wikipedia).
- Develop an example or prototype of a selected information product.

1.2.4 Deliverables

- Final Report containing proposed standards of content and structure of information products.
- Example or prototype of a selected information product which focusses on higher levels of interoperability.

The Task Group started its work in July 2010 and finished work in November 2013.

1.3 GENERAL APPROACH

ET-027 prepared a collection of 63 interoperability issues, listed as keywords or very short descriptions (see Annex C for original list of ET-027 issues). The main goal of MSG-086 was to provide a structured and systematic overview of potential interoperability problems and possible solution approaches, based on this list. Because of the very brief issue descriptions in the ET-027 list, it turned out to be necessary to provide a short problem description for every interoperability issue in the ET-027 list.

In this process, MSG-086 also tried to identify issues that rather addressed solution approaches than interoperability problems. If such issues were identified, either the underlying problem was described as interoperability issue or the issue was dropped as it did not describe an interoperability problem.

In order to provide the basis for a systematic discussion of interoperability issues and solution approaches, MSG-086 structured the issues along two dimensions:

1) In terms of the phase(s) in a corresponding system development process, where the interoperability issue may arise or may be caused; and
2) In terms of the level of interoperability that is addressed by the respective issue.

To locate each issue and its potential solutions within these two dimensions, specific approaches for both dimensions have been chosen: The DSEEP for the "time" axis and the Levels of Conceptual Interoperability Model (LCIM) proposed by Andreas Tolk et al. [Tolk2006] for the "space" axis. Both concepts have been examined with respect to their suitability for the objectives of MSG-086 and are introduced in more detail in the following sections of this report. They appear to be the most generic approaches of their kind, i.e., other approaches pointing in similar directions usually can be mapped onto the DSEEP or the LCIM, respectively.

As a first step of the structuring process the finally identified interoperability issues were categorized into 9 issue groups:

- Conceptual Model (CM);
- Federation Development (FD);
- Fidelity (FI);
- Infrastructure and tools (IN);
- LVC and C2-Sim coupling (LC);
- Organizational and Legal issues (OL);
- Scenario (SC);
- Synthetic Environment (SN); and
- Time Management (TM).

MSG-086 is aware of the fact that this is not a unique way of forming issue groups. In fact, the development of these groups has been a subject of major discussion. However, the chosen groups turned out to be helpful in providing a better hierarchical overview of the identified interoperability issues.

The main purpose of the issue groups is to have a structured discussion of the interoperability issues. This includes:

- A clear description of each identified issue;
- A systematic view (when and where does an issue exist?);
- An overview of possible solutions; and
- A discussion of existing and/or recommended implementations and information products related to an issue and/or solution approaches.

Consequently, given the issue group structure, MSG-086 provides a short introduction to every issue group including an overview of the issues in every group. Subsequently, for every issue the following information is included:

- **Problem Definition**: A very short description of the core problem that arises with the described issue.
- **Extended Problem Description**: A more detailed description of the background of the respective issue, including examples where and how the described issue may arise.
- **Related Work**: Some issues and the problems associated with them that have been discussed in other contexts.
- **Connection to LCIM**: For every identified interoperability issue a discussion is included to which level of the LCIM model it is related.
- **Connection to DSEEP Steps and Artifacts**: Description in which DSEEP step an interoperability issue arises (i.e., may be caused or identified).
- **Possible Solution Approaches**: Discussion of existing solution approaches for the described issue as well as suggestions for potential solutions that may have to be developed in more detail in ongoing and future work.
- **Existing Implementations and their Information Products**: For existing solution approaches, implementations may already exist. Furthermore, existing information products that are supposed to be provided to avoid or to reduce the effect of an issue are mentioned.
- **Recommended Information Products**: In some cases, solution approaches that are pointed out include the definition and use of additional or optimized information products. Recommendations for such information products are described here.
- **Examples and Prototypes of Selected Information Products**: If examples or prototypical versions of such recommended information products exist, they are shortly described here.

As a summarizing consequence of the issue discussions, the information product “Scenario” was selected to provide an example recommendation in more detail. Furthermore, recommendations for a number of standards and documents (e.g., DSEEP, FEAT, AMSP-01, and the NATO MORS M&S Gap Analysis Questionnaire) are provided in this report, including proposals for future NMSG activities as well as recommendations for SISO working groups.

Chapter 2 – BACKGROUND MATERIAL AND RELATED WORK

2.1 DISTRIBUTED SIMULATION ENGINEERING AND EXECUTION PROCESS (DSEEP)

The Distributed Simulation Engineering and Execution Process (DSEEP) [DSEEP] describes a 7-step process for developing and executing a simulation environment (see Figure 2-1).

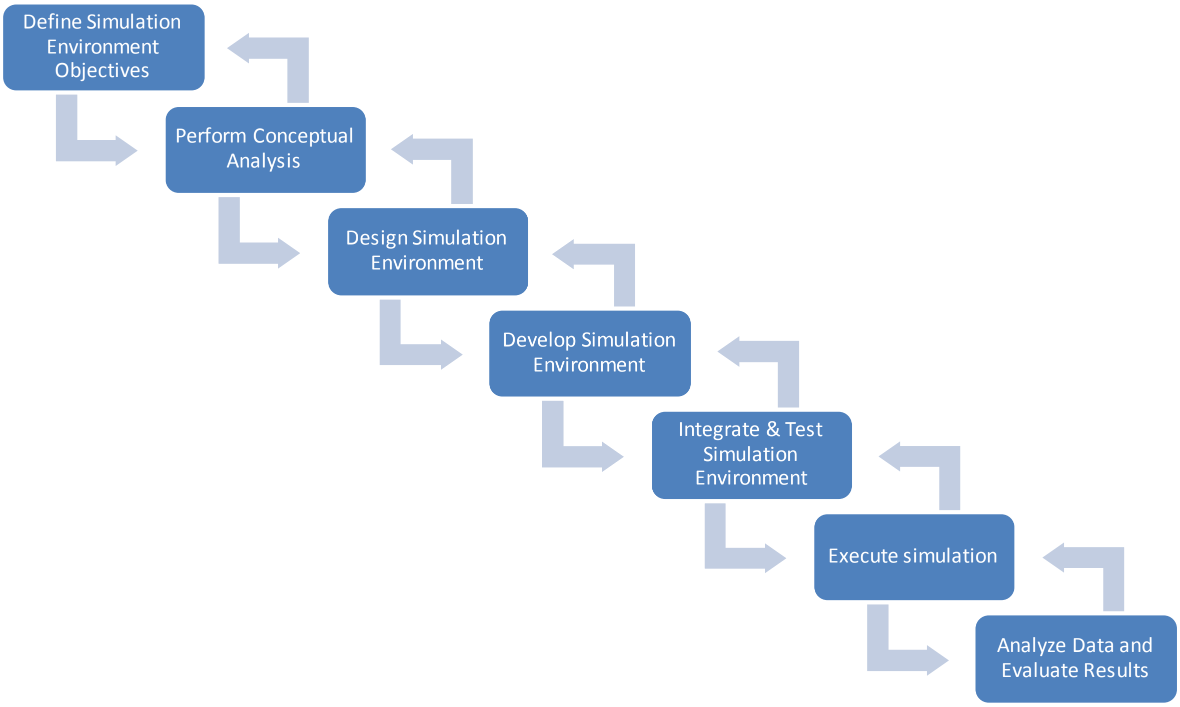


Figure 2-1: Seven-Step Process Defined by DSEEP.

The DSEEP was developed and is maintained by the Simulation Interoperability Standards Organization (SISO) and is standardized as IEEE 1730.

One basic objective during development of the DSEEP was to develop an architecture-neutral successor of the Federation Development and Execution Process (FEDEP). Whilst the FEDEP was specifically designed for setting up HLA-based distributed simulation environments, the DSEEP is not focussed on a single simulation architecture (like e.g., HLA, DIS, or TENA).

Each step of the DSEEP is subdivided into more specific activities, and for each activity inputs and outputs are specified by the DSEEP. As the DSEEP is considered as a generalized, high-level framework which has to be adapted to individual needs, the DSEEP does not provide very detailed guidance or even documentation templates for activity outcomes. It is assumed that each organization that uses the DSEEP provides this detailed guidance as part of the individual adaptation. One example for such an adaptation of the DSEEP is the German process model VEVA ("Vorgehensmodell für den Einsatz der VIntEL-Architektur", engl. "Process Model for Application of the VIntEL-Architecture") [11S-SIW-044].

Since the FEDEP is superseded by the DSEEP, MSG-086 decided to base its work on the DSEEP instead of the FEDEP. As a consequence, this report uses mainly the architecture-neutral terms as defined by the

DSEEP instead of the usual HLA parlance (e.g., the term federation is specific for HLA). This leads to the following mapping of terms:

FEDEP (HLA)	**DSEEP**
Federation	Simulation Environment
Federate	Member Application
Federation Object Model (FOM)	Simulation Data Exchange Model (SDEM)
Federation Agreements Document (FAD)	Simulation Environment Agreements

But as experience shows, the original HLA terms are still used widely: people are used to them, and in general, the HLA terms are handier to use. MSG-086 uses primarily the DSEEP terminology, but sometimes the HLA terms are used in parallel with the DSEEP terms.

2.2 LEVELS OF CONCEPTUAL INTEROPERABILITY MODEL (LCIM)

Many models have been proposed to capture and structure "interoperability" [Brownsword2004]. A widely used, systematic approach of structuring the discussion about interoperability definitions and criteria has been proposed by Tolk et al. [03F-SIW-007], [Tolk 2006], [Wang2009]. The Levels of Conceptual Interoperability Model (LCIM) identifies and characterizes six levels of interoperability (see Figure 2-2). In the following, a brief summary of the LCIM is provided, more detailed descriptions can be found in [03F-SIW-007], [Tolk 2006] and [Wang2009].

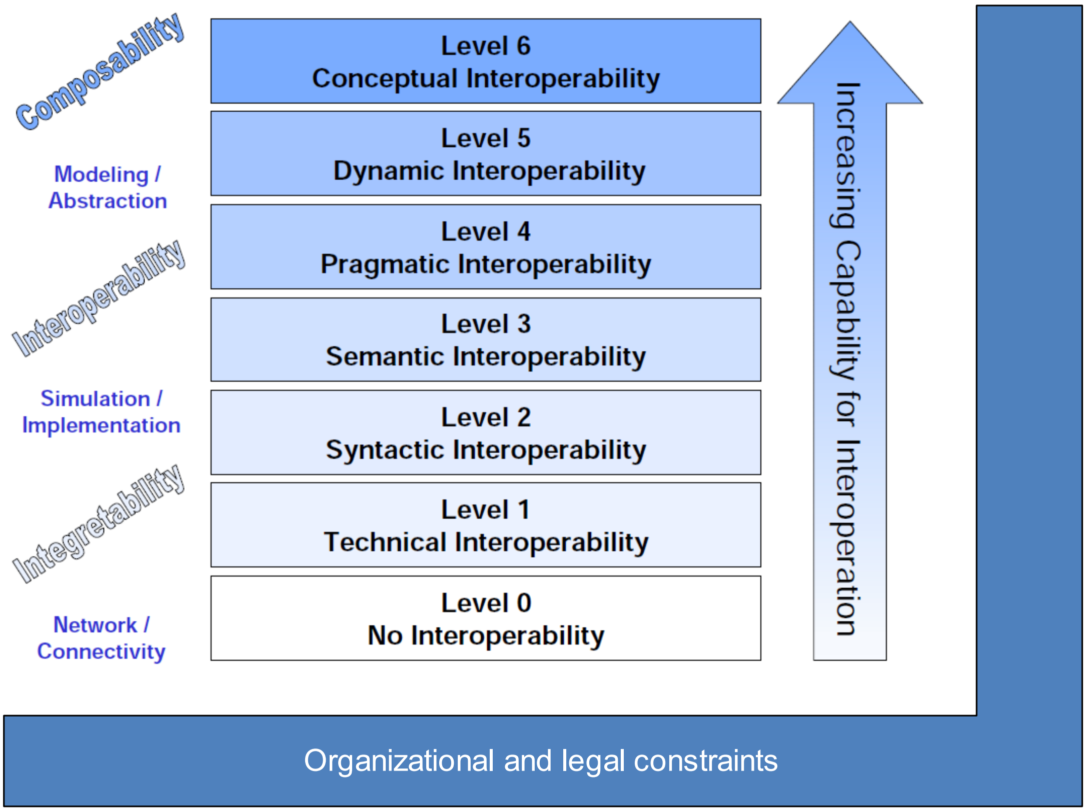


Figure 2-2: The Levels of Conceptual Interoperability Model (Figure adapted from [Tolk2006]).

As illustrated in Figure 2-2, the LCIM defines the following six levels of interoperability:

- **Level 1 – Technical Interoperability**:
  - Communication infrastructure and corresponding protocols exist. Data can be exchanged between systems.
- **Level 2 – Syntactic Interoperability**:
  - Common structures for information exchange such as common data formats exist. Protocols are agreed upon that ensure that the right forms of data are exchanged in the right order. A common view about the meaning of data is not established.
- **Level 3 – Semantic Interoperability**:
  - Shared meaning of data exists, i.e., the interoperating systems are exchanging data that can be semantically parsed.
- **Level 4 – Pragmatic Interoperability**:
  - Mutual awareness of methods and procedures between simulation systems is established. The context and meaning of exchanged information is agreed upon.
- **Level 5 – Dynamic Interoperability**:
  - Interoperating simulation systems are mutually aware of changes in assumptions and constraints over time. This also means that interoperating systems are able to re-orient their information delivery and consumption according to these changing assumptions.
- **Level 6 – Conceptual Interoperability**:
  - A fully documented overall conceptual model exists. The interoperating systems are completely aware of each other’s meaning of information, processes, contexts, and changing modelling assumptions.

Although the LCIM with its six levels of interoperability covers very well the degree as to which extent simulation systems are capable of working together from integratability via interoperability to composability (which is often used synonymously to the term interoperability), some aspects of interoperability that constrain the actual outcome of a distributed simulation environment, are not considered. During the course of this Task Group it turned out that often organizational and legal aspects determine the level of interoperability that can be achieved. Examples of such organizational and legal issues are collected in the corresponding interoperability issues group that is discussed in more detail in later sections of this report. In fact, organizational and legal aspects may restrict interoperability on any of the six LCIM levels. Consequently, MSG-086 embedded the original LCIM into a frame composed of organizational and legal constraints. An overview of the adapted LCIM is depicted in Figure 2-2. In this figure, the original LCIM concept as published in [03F-SIW-007] is extended by integrating the LCIM within a fundament and framework of organizational and legal constraints.

2.3 FAIR FIGHT

Tightly related to simulation interoperability is the term “fair fight”. The term “fair fight” is regularly used in context of distributed simulation environments and although everybody has an intuitive feeling about what is meant by this term, a good official definition is still lacking. The following definition is used by MSG-086:

> “Fair Fight exists among two simulation systems if the differences in representing reality in the simulation models do not lead to a systematic and model immanent advantage and consequently unrealistic simulation results for one of the simulation systems.” [Siegfried2011]

It is important to stress that fair fight may always only be defined for a specific simulation environment. There is no "absolute fair fight". Two (or more) simulation systems can only be in fair fight with regards to the objectives and constraints of a specific simulation environment. Therefore, fair fight problems result from differences in the representation of the real world that are not acceptable for a specific simulation environment.

Obviously, different simulation systems use a different representation of the real world. This is perfectly fine and will always be the case (as for example, simulation systems are developed for different purposes or by different developers). A different representation of the real world does not necessarily prohibit fair fight as long as these differences are compatible with the requirements for a specific simulation environment (e.g., if communication effects are not important for a specific simulation environment, fair fight is achieved even if communication between units is modelled differently by two simulation systems).

Achieving fair fight requires simulation interoperability on all levels (as described above).

2.4 RELATED NATO ACTIVITIES

In the following sub-sections NATO MSGs are described that are related to MSG-086 at the time this report was written.

2.4.1 MSG-052 "Knowledge Network for Federation Architecture and Design"

The main objective of MSG-052 "Knowledge Network for Federation Architecture and Design" was to initiate a knowledge network to promote development and sharing of information and knowledge about federation architecture and design issues among NATO and Partner Nations. These results were to be gathered and made available in a knowledge base for the M&S community. MSG-052 made use of a wiki-based Collaborative Working Environment (CWE) to develop the knowledge base and to share information. Through the CWE and associated tools experts contributed to the body of knowledge about federation architecture and design.

The knowledge network was a combination of a Community of Practice (CoP), tools, processes and knowledge bases. Activities that have been explored by the CoP as the driver for establishing the knowledge network include: Federation Architecture and Design Taxonomy, Federation and Simulation Execution Management, Federation Agreements and Federation Agility [NMSG-052].

2.4.1.1 Relevance for MSG-086

The work of MSG-052 is relevant to simulation interoperability in more than one aspect. Federation architecture and design issues are fundamental for simulation interoperability for distributed simulation environments and are also a main area for MSG-086. A knowledge network as proposed by MSG-052 might contribute to solve different issues of simulation interoperability.

2.4.2 MSG-058 "Conceptual Modeling for Military Modeling and Simulation"

The major goal of MSG-058 was to develop a guidance document on conceptual models, which can be used in the future by NATO to support its M&S requirements [NMSG-058]. The most important requirements identified are:

- Clarify the "Conceptual Model" concepts, discuss the terminology, and emphasize the utility to better formalize conceptual models, understand the relationship between conceptual modeling and related concepts (scenario definition, etc.);

- Draft a guidance document on conceptual models that can be used by different stakeholders (sponsor/user, project manager, subject-matter experts, verification and validation agents, developers, etc.);
- Investigate methodologies, simulation and software engineering processes, initiatives and technologies useful for the establishment and content of conceptual models;
- Foster the establishment of the guidance document as a SISO standard;
- Identification of the relevant stakeholders of conceptual models and considering whether a prioritization is needed;
- Addressing the needs of M&S community, identifying the way conceptual models may contribute to M&S development, and providing guidance to implementation; and
- Providing guidelines for standards in conceptual modeling for M&S; thereby specifying a conceptual model to be (re)usable by users with similar knowledge and to be accepted by the computer science community.

2.4.2.1 Relevance for MSG-086

The relevance of MSG-058 work to MSG-086 is that it provides a common understanding from the outset of the effort that a conceptual model should serve as a frame of reference for simulation development by documenting important entities/concepts, their properties, and their key actions and interactions. Conceptual models are a bridge between the requirements and simulation environment design, thereby supporting that the simulation systems used in military applications such as training and decision support are fit for purpose. Issue group CM "Conceptual Modeling" discusses interoperability issues related to conceptual modeling.

2.4.3 MSG-068 "NATO Education and Training Network (NETN)"

The objective of the MSG-068 NETN Task Group was to assess the distributed simulation and learning capabilities that NATO, Partner and Contact Nations, schools, and agencies have that could contribute to the development of a NETN capability [NMSG-068]. The Task Group also recommended and demonstrated a way forward for interoperability, technical standards and architectures to link these training and education centres to provide a shared persistent capability. Finally, MSG-068 identified and recommended roles and responsibilities of NATO, Partner and Contact Nation organizations responsible for distributing and maintaining M&S capabilities within the scope of NETN. The following topics were covered under this Task Group to meet the objectives:

- Assessment of distributed simulation and learning capabilities with potential for inclusion in NETN;
- Recommendations for interoperability and technical standards;
- Recommendations for the development of NETN architectures;
- Recommendations for the assignment of roles and responsibilities for distributing, managing and maintaining NETN capabilities;
- Identify, develop and conduct experiments enabling NATO/PfP Nation's capabilities to participate in NETN;
- Roadmaps and technical reports in support of NETN;
- Demonstration of a limited NETN realization comprising JWC, JFTC and national simulation centres and systems;
- Run preparatory tests at ACT and national facilities and evaluate the results from these tests for risk reduction of the demonstration of the feasibility of the NETN-concept; and

- Perform a demonstration of the feasibility of the NETN concept of a distributed networked training capability embracing JWC, JFTC and national simulation centres and the corresponding simulators, simulation systems and C2- systems.

2.4.3.1 Relevance for MSG-086

The relevance of MSG-068 to MSG-086 is that the MSG-068 Task Group produced recommendations for interoperability and technical standards. Mainly issues in the issues groups “CM Conceptual Model” and “FD Federation Development” have relevance to the work conducted in MSG-068.

For example, the recommendations of MSG-068 contain guidelines for service exchange among simulation members and MSG-068 also developed an object model that solved some issues within commonly used object models.

2.4.4 MSG-071 “Missionland – A Universal, Generic, Multi-Spectral Simulation Environment Database”

The prime objective of the Missionland Task Group was to make available a shared coherent dataset from which environment databases can be constructed for a wide scope of simulators operating in air, sea and land domains [NMSG-071]. These environment databases are generally needed for visual out-of-the-window and sensor views, but also terrain servers and computer generated forces applications often make use of such databases.

Problems with creating correlated synthetic environments and the limited availability of source data due to security and political limitations should be addressed by the Missionland dataset. Therefore it is preferable to create a generic and geo-unspecific synthetic environment. Using a geo-unspecific synthetic environment would also overcome the objections that result from using a real world area as basis for the synthetic environment. And besides that, it also offers the advantage of combining geologically different environments in the same synthetic environment. This makes a generic environment much more flexible in performing different types of missions within the same synthetic environment.

An open source development model has been used to ensure that participating Nations have full access to the dataset and can feed changes and improvements made back into Missionland. In support of the prime objective, there is the need to complement a deployment and continuation process to ensure proper use and continuation after the creation of Missionland, which includes guidelines and user’s support.

2.4.4.1 Relevance for MSG-086

The relevance of Missionland to MSG-086 is that the Missionland Task Group created a correlated dataset that would help solving some of the issues related to interoperability and the synthetic environment. Missionland will not solve all synthetic environment issues, but the fact that all users have access to the same correlated dataset will reduce correlation problems. Besides that the fact that Missionland can be freely shared between NATO and PfP Nations also means that it will help in reducing the legal and political issues related to sharing synthetic environments.

So of the identified issues Missionland contributes to a solution of the following issues:

- SN-01 Synthetic environment data is not correlated; and
- OL-06 Legal or political restrictions apply.

2.4.5 MSG-080 “Security in Collective Mission Simulation”

Collective mission simulation is proving an important enabler to achieve military objectives within application areas such as development, training and exercises. However, in most cases the simulation models exist within different security domains and these models need to be protected while at the same time information needs to be shared between the different simulations. Therefore, there is an increasing need for solutions that enable the sharing of simulation information across different classified security domains to establish collective simulations without a potential information leakage and confidentiality breach.

The activity will focus on how information within different security domains can be protected with a minimal impact on the operation of the collective mission simulation. Furthermore the activity will focus on a solution that does not require configuration per simulation as is the case today due to the ‘system high’ approach. The work will be based on a labelling and release mechanism that is closely bound to the simulator or (national) simulation domain that it protects [NMSG-080].

For additional information see [12S-SIW-032] or refer to SISO Standing Study Group (SSG) on “Security in Simulation” that was set up in Spring 2012.

2.4.5.1 Relevance for MSG-086

The relevance of MSG-080 to MSG-086 is that the Security in Collective Mission Simulation Task Group is addressing all issues related to security in distributed simulation environments. Although MSG-080 is addressing all security issues identified in ET-027, it is not addressing implicit issues e.g., procedure or organizational issues that arise because of security constraints. Examples of such issues are described in the issue group “Organizational and Legal issues” (OL).

2.4.6 MSG-085 “Standardization for C2-Simulation Interoperation”

Interoperation among C2 and simulation systems is required to support military activities such as: Force Readiness, Support to Operations, and Capabilities Development. Currently, interoperating systems from different manufacturers or Nations requires proprietary interfaces that require time and money to develop and maintain. Furthermore, in many cases, in addition to these vendor-specific interfaces, human intervention is required during military scenario definition, initialization and execution. The so-called “swivel-chair” interface entails feeding simulation operators with information that they must translate manually into instructions that the simulation systems can process.

Developing standards (like C-BML) that define common interfaces for the exchange of military information among C2 and simulation systems therefore can lead to significant cost-reduction and greatly facilitate systems integration. The benefits of standardizing C2-simulation interoperation include: reduced cost and workload, reduced scenario preparation time, and increased realism and overall effectiveness.

The high-level objectives of MSG-085 (as defined in MSG-085 Programme of Work) are as follows:

- Define the scope and operational and technical requirements for C-BML;
- Establish a set of reference expressions based on NATO operational procedures;
- Assess and leverage available C-BML implementations;
- Address command and control information systems and simulation initialization requirements; and
- Demonstrate and communicate the operational relevance and benefits of C-BML for improving the efficiency of military operations.

2.4.6.1 Relevance for MSG-086

Simulation interoperability problems for C2 systems are being studied by MSG-086 within issue group LC (LVC and C2-Sim coupling). Especially issue LC-05 (Limited simulation awareness of C2 systems) is related to the work of MSG-085.

2.4.7 MSG-098 and MSG-099 Task Groups on "Urban Combat Advanced Training Technology" (UCATT 3)

Since 2003 NMSG has worked with the issues, problems and limitations associated with developing Military Operations on Urban Terrain (MOUT) training facilities. The Urban Combat Advanced Training Technology (UCATT) Task Groups, i.e., MSG-032 "UCATT", MSG-063 "UCATT 2" and the ongoing MSG-098 and MSG-099 "UCATT 3" have been tasked to exchange and assess information on MOUT facilities and training/simulation systems with a view toward establishing best practices. In addition it has been required to identify interoperability requirements and a suitable architecture and a standard set of interfaces that would enable interoperability of MOUT training components.

In the last couple of years UCATT has become NATO's focal point for MOUT training technology and exchanging information with the military community and is as well regarded among industry as the driving force within the live domain.

MSG-098 is working on the architecture aspect including:

- Exchange and assess information on MOUT (live/constructive/virtual) installations and training/simulation systems to identify and maintain a comprehensive list of generic user requirements;
- Identify and maintain a suitable architecture and a standard set of interfaces that enable interoperability of MOUT training components that do not inhibit future research and enhancements; and
- Identify limitations and constraints on MOUT development with a view toward identifying areas for future research.

MSG-099 is focused on standardization of potential UCATT defined interfaces (e.g., frequency spectrum allocation and management, laser compatibility, battlefield effects simulations, firing through walls, indirect fires, tracking and position/location in built-up areas).

MSG-099 mainly supports the SISO process and works toward SISO approved standards for UCATT architecture defined interfaces that will make interoperability between MOUT training systems possible.

2.4.7.1 Relevance for MSG-086

The work of UCATT is relevant to MSG-086 regarding the live aspect of simulation interoperability (issue group LC). MSG-098 and MSG-099 identify and standardize architectures and interfaces that enable higher levels of interoperability of MOUT training facilities.

2.4.8 MSG-106 "Enhanced CAX Architecture, Design, and Methodology – SPHINX"

The MSG-106 activities are a continuation of the efforts and results of MSG-068 and the overall objectives are:

- To define a common technical framework for NETN exercises in order to facilitate persistent training capability, federation set-up, better interoperability, long-term maintenance, certification testing, configuration management and establishment of standard profile; and

- To deliver to NATO and partners a persistent, distributed combined joint training capability able to support training from the operational to the tactical level across the full spectrum of operations, through leveraging existing national expertise and capabilities.

2.4.8.1 Relevance for MSG-086

The relevance of MSG-106 to MSG-086 is that MSG-106 will produce recommendations for interoperability and design patterns for technical and architectural aspects (e.g., logistics, and multi-resolution modelling). Mainly issues in the following issue groups have relevance to the work conducted in MSG-106:

- Conceptual Model (CM);
- Federation Development (FD);
- LVC and C2-Sim Coupling (LC);
- Scenario (SC); and
- Time Management (TM).

2.4.9 NIAG SG 162

The NATO Industrial Advisory Group (NIAG) Study Group (SG) 162 on “Distributed Simulation for Air and Joint Mission Training” [NIAG-SG-162] aimed at developing a roadmap for future Mission Training through Distributed Simulation (MTDS), starting from currently available and known planned simulation architectures, to support air and joint simulated mission training. The SG 162 started activities in July 2011 with 40 members from 35 companies of 14 NATO/PfP participating Nations for a 1-year activity.

2.4.9.1 Relevance for MSG-086

NIAG SG 162 has many relationships with MSG-086, e.g.:

- It acknowledges the importance of a correlated representation of the synthetic environment (issue group “Synthetic Environment”);
- It acknowledges the importance to evaluate federates with regards to their compliance (issue groups “Conceptual Model” and “Federation Development”); and
- It encourages the use of MSDL and C-BML (issue group “Scenario”).

Furthermore, NIAG SG 162 identified 39 issues that need to be solved to achieve the MTDS requirements. The issues are described in a different way than the issues identified by MSG-086 and extend the issues described in this report.

2.5 RELATED SISO ACTIVITIES

In the following sub-sections SISO activities are described that are related to MSG-086 at the time this report was written.

2.5.1 DSEEP (Distributed Simulation Engineering and Execution Process)

The DSEEP defines a process model for planning, setting up, and executing a distributed simulation environment. The DSEEP is the architecture-neutral successor of FEDEP (which was HLA-centric) and is available as IEEE 1730 standard (free for SISO members via SISO webpage). See Section 2.1 for more details about the DSEEP.

The DSEEP Product Support Group (PSG) is concerned with all activities related to usage and maintenance of the DSEEP standard. Currently, the DSEEP PSG collects user experiences, change requests and further recommendations for updating or extending the DSEEP. A major revision of the DSEEP standard itself might start in 2015 (depending on number of change requests, etc.).

2.5.1.1 Relevance for MSG-086

The DSEEP is highly relevant for MSG-086 as all interoperability issues and information products are considered in context of the DSEEP.

2.5.2 DMAO (DSEEP Multi-Architecture Overlay)

The DSEEP Multi-Architecture Overlay (DMAO) is an overlay for the DSEEP process model that specifically addresses issues that arise in multi-architecture simulation environments (e.g., when using HLA and DIS in the same simulation environment). The DMAO is available as IEEE 1730.1 standard (free for SISO members via SISO webpage).

Ongoing user support and standards maintenance activities are pursued by the DSEEP PSG due to the close relation of DSEEP and DMAO.

2.5.2.1 Relevance for MSG-086

The DMAO is relevant for MSG-086, especially within issue group “Federation Development” as it defines many problems (“issues”) that might complicate the development of a multi-architecture simulation environment. See FD-01 (Multi-Architecture Simulation Environments) for more details.

2.5.3 FEAT (Federation Engineering Agreements Template)

The Federation Engineering Agreements Template (FEAT) is a SISO standardization effort for all agreements (i.e., documentation items) which are necessary and typically required for any simulation environment engineering effort. The FEAT defines a template (as XML Schema) to provide an unambiguous format for recording agreements about the design and use of a distributed simulation environment. The template will also enable the development of federation engineering tools that can read the schema and perform federation engineering tasks automatically.

2.5.3.1 Relevance for MSG-086

The FEAT is highly relevant for MSG-086 as it provides a structured template that covers the whole documentation of a simulation environment engineering process. Many issues encountered by MSG-086 are traced back to missing or incomplete specification and documentation. Therefore, a standardized documentation template would be helpful.

2.5.4 MSDL (Military Scenario Definition Language)

The Military Scenario Definition Language (MSDL) defines an XML Schema for describing military scenarios. Currently, MSDL is restricted to static descriptions of military situations (e.g., starting conditions of a scenario) and mostly used for specifying initial situations of a military scenario.

New development efforts are discussed, including extensions and/or updates to the current MSDL version as well as alignment with C-BML.

2.5.4.1 Relevance for MSG-086

MSDL is relevant for MSG-086 as MSG-086 focussed on scenario development and developed an information product “Scenario”. Therefore, MSDL as a formal standard for scenario specification plays an important role.

2.5.5 SCM (Simulation Conceptual Modeling)

The Simulation Conceptual Modeling (SCM) PDG will produce a stand-alone guidance document that will clarify conceptual model concepts, discuss conceptual modeling terminology, and enable different stakeholders to improve the formalization of conceptual models. The first draft of the “Conceptual Modeling Recommended Practice” will be based upon the MSG-058 final report.

2.5.5.1 Relevance for MSG-086

The SCM PDG is considered relevant for MSG-086 (especially for issue group “Conceptual Model” and information product “Scenario”) as many interoperability issues can be traced back to a missing or badly specified conceptual model. Also, unaligned or incompatible conceptual models are identified as regular cause for interoperability problems. If the SCM PDG could provide an information product on conceptual modeling (as MSG-086 did for scenario development) this would be considered an important step forward.

2.5.6 RIEDP (Reuse and Interoperation of Environmental Data and Processes)

The RIEDP PDG works on two standards to make it easier to exchange and reuse environmental data between different initiatives that exist in this field. The two products the PDG is working on are:

- **The Environmental Data Model Foundations product**, which will be a SISO Guidance document, and will include the formalization of a Reference Process Model (RPM) and a Reference Abstract Data Model (RADM); and
- **The Environmental Detailed Features Description product**, which will be a SISO Standard product, and will improve reuse and interoperability between different initiatives related to the synthetic environment.

The RIEDP PDG started in 2013, following the RIEDP Study Group (SG) that worked from 2010 to 2012 [RIEDP_2012].

2.5.6.1 Relevance for MSG-086

The RIEDP working groups are related to the issue group Synthetic Environment (SN) and therefore considered relevant for MSG-086.

2.6 RELATED NATIONAL ACTIVITIES

Based on national input this section tries to give an overview of national activities related to MSG-086. This description is not a complete overview and thus no national activities may be described for some Nations.

2.6.1 Germany

There are two main activities in Germany going on regarding simulation interoperability:

1) SuTBw (German: “Simulations- und Testumgebung der Bundeswehr”): Simulation and Test Environment of the German Armed Forces; and

2) The so-called "SD VIntEL" (German: "Systemdemonstrator Verteilte Integrierte Erprobungslandschaft"): System Demonstrator for a Distributed Integrated Test Environment.

2.6.1.1 SuTBw

Starting in 2006, the German Armed Forces initiated a project named SuTBw (German: "Simulations- und Testumgebung der Bundeswehr", Engl. "Simulation and Test Environment of the German Armed Forces") to provide the infrastructure for the coupling of simulation systems. The infrastructure offers networks, network services, software tools and technical support to all military simulation projects within the German armed forces.

2.6.1.2 SD VIntEL

Build on a generalized infrastructure distributed all over Germany at places working on military simulations provided by SuTBw, SD VIntEL offers a variety of methods, tools, and databases to achieve simulation interoperability over a wide range of aspects, e.g., methods for making, distributing and maintaining a common environmental database.

2.6.2 Italy

2.6.2.1 SimLabs

The Simulation Network SimLabs was created as the result of the activities conducted by the working group of the same name belonging to the SET2 (Simulation for Experimentation and Test, Evaluation and Training) community of Finmeccanica MindSh@re. SET2 mission includes the exploration and the development of innovative technologies and solutions for the Group in the field of M&S and the creation of a technological network connected with customers and institutional and academic partners.

SimLabs connects seven Group Companies' Simulation Labs: Selex ES (three labs) Alenia Aermacchi, Telespazio, OTO Melara, MBDA IT, setting up an "on demand", scalable, distributed, operational environment.

The resulting Distributed Simulation Infrastructure provides the following main benefits:

- Secure, high speed, low latency, priority configured, data paths on WAN links;
- A common simulation platform supporting both systems engineering development and training through a "Virtual System of Systems" testing environment;
- Promote knowledge and experiences sharing among FNM Group companies in the modeling and simulation domain;
- Costs reduction and productivity increase by "remote system development and integration"; and
- Business benefits from the exploitation of new solutions arising from the capability to test together different simulators / simulation areas.

The innovative value of SimLabs is representing one of the few examples of a federation of laboratories that do not belong to a single company and, at business level, it gives Finmeccanica the opportunity to explore advantages deriving from getting sets of simulators developed for specific different and complementary environments to work together and to create a "Simulation Hosting Service" to meet the requirements of governmental or military customers. In February 2013, SimLabs was federated with the NATO Modeling and Simulation Centre of Excellence at Cecchignola site in Rome, confirming its functionality and effectiveness through the achievement of a real-time simulation involving different systems and laboratories.

2.6.2.2 ITB Forza NEC

In the context of the Forza NEC (Network-Enabled Capability) project, undertaken by the Italian Defence Administration and a pool of Italian companies acting in partnership with the objective of modernizing the Italian Armed Forces, the ITB (Integrated Test Bed) represents a “spread” centre for concept development and experimentation, consisting of numerous military centers federated over a geographically distributed network and characterized by a completely integrated and interoperable environment.

2.6.3 Netherlands

In the Netherlands several actions to improve simulation standardization are or will be taken which will also have a positive effect on simulation interoperability. These standardization activities include:

- Limiting the number of different CGF applications for new simulation systems;
- Standardization on a common set of geographical data from which the digital terrain databases are produced; and
- Dutch MoD wide licenses for use and re-use of components and (geo) data are part of the contracts for new simulation systems nowadays.

The Dutch MoD has started a project for the development of a “common C2 – Sim” interface.

The creation of a “Joint Environmental Simulation Database” can be depicted as a (near) future activity. This database aims to improve the reuse of environmental data and databases, scenarios, doctrines and entity and weapon models by bringing them together in one repository.

2.6.4 Sweden

2.6.4.1 Defence Concept Modelling Framework (DCMF)

DCMF (Defence Conceptual Modelling Framework) is the name of a Swedish framework for the development and use of conceptual models, which itself has the potential to support the development of simulation models. DCMF has its foundation in the US military’s efforts on distributed simulation which begun in the mid-1990s. The attempt was called CMMS or Conceptual Models of Mission Space. The DCMF framework is developed by the Swedish Defence Research Agency (FOI), financed by the Swedish Armed Forces.

The process to develop a knowledge base, prior to the development of a simulation model, is a time-consuming and costly process. Moreover, the knowledge that a model represents is often not properly saved for future reuse. Even if the model is accurately stored it is difficult to reuse it in other context. This is primarily because knowledge of how a model was created is not documented to the extent necessary. In other words, for potential reuse of a model, relevant facts about knowledge acquisition are missing and hence so are their traceability to a knowledge source.

The DCMF framework includes a process whose main objective is to address how knowledge can be acquired for just such a particular purpose, as well as how it can be structured, modelled and formatted according to pre-defined criteria. It also addresses how knowledge can be used, or re-used, given diverse applications. The process consists of four main phases:

- Knowledge Acquisition (KA);
- Knowledge Representation (KR);
- Knowledge Modelling (KM); and
- Knowledge Use (KU).

The need of a (activity centric) tool for modelling the acquired knowledge has resulted in the development of the concept called “KM3 – Knowledge Meta Meta Model”. Another important component of the framework are the ontologies. An ontology suite and methodology have been developed for the DCMF purposes.

In summary, conceptual modelling in the context of military M&S is considered to be able to contribute to:

- **A cost-effective development process** – By standardising methods for the collection and representation of knowledge, as well as creating a common infrastructure for its storage, knowledge reuse can be realised on a larger scale, i.e., the same knowledge can be used in several development contexts.
- **High-quality knowledge** – The DCMF method assumes that authorised sources will be used to acquire knowledge, i.e., just anyone who has ideas about an activity may not be used. Furthermore, using methods for strict formalisation of knowledge in the form of models, quality assurance and reuse can be facilitated.
- **Support in the early stages –** Using conceptual modelling in the early phases of the development cycle (for example, for constructing a simulation model) creates a common understanding of the problems to be addressed that can be communicated to all stakeholders for a project.

DCMF is a framework for making conceptual descriptions and models of military operations. It consists of tools for their development and reusability, as well as standards for acquisition, representation, modelling, integration, management and preservation of knowledge.

A good introduction to DCMF and that also describes its components is the report [Mojtahed2005] from 2005. Although it is quite old it gives a good overview of the concept. Two other reports focus on different specific areas. The report [Mojtahed2008b] focusing on the end product and the BOM concept (and suggest BOM++). The report [Mojtahed2008] focusses on knowledge use and puts “Repository, Processes and Products” in context.

2.6.4.2 M&S Architecture and Design Guidelines for Command Training Federations Based on the SMART-Net Concept

This is a guideline that categorizes, characterizes and provides guidelines for some of the common M&S design issues encountered in federation development and specific guidelines are given for simulation supporting Command and Control (C2) Training Federations (CTF).

The guideline provides definitions of the basic concepts of federation architecture and design. It also provides references to other best practices documents, guidelines, standards and policy documents.

The document introduces the “SMART-net” concept which provides general and overarching design principles for all domain specific architectures. This includes references to and definitions of common design issues. It also provides detailed guidance based on the SMART-net principles with respect to the C2 Training domain.

In addition the document also documents federation design patterns currently not documented elsewhere. It provides more detailed explanations on the patterns of simulation interplay used to address specific federation design issues. These patterns and others defined in other referenced documents are used to describe and characterize specific designs [FMV2011].

Chapter 3 – IDENTIFIED INTEROPERABILITY ISSUES

The original list of simulation interoperability issues compiled by ET-027 contained 63 issues. After a thorough review by MSG-086 some issues were not considered as interoperability issues (e.g., because they were rather solution approaches than actual problems) and some issues were combined due to their similarity. During these discussions, MSG-086 also identified new interoperability issues. In total, 46 interoperability issues are identified (see Annex C for detailed traceability to ET-027 issues) and described by MSG-086 according to the following schema:

- Problem definition:
  - Short definition of each interoperability issue (usually one to three sentences).
- Extended problem description:
  - Detailed problem description which usually includes examples for illustrating each problem.
- Related work:
  - Pointers to related work (e.g., other NATO MSGs or SISO working groups). Also, description which element of the Federation Engineering Agreements Template (FEAT) is connected with each interoperability issue (see Section 3.5 for description of FEAT).
- Connection to LCIM level:
  - Description which level of the Levels of Conceptual Interoperability Model (LCIM) is related with each interoperability issue. It is described how each issue may cause problems, i.e., which level of interoperability may not be met because of problems due to the described issue.
- Connection to DSEEP steps and artifacts:
  - Description which steps and artifacts of the Distributed Simulation Engineering and Execution Process (DSEEP) are connected with each interoperability issue. If applicable, DSEEP artifacts that are concerned by an issue are also mentioned.
- Possible solution approaches:
  - Outline of possible solution approaches for each interoperability issue.
- Existing implementations and information products:
  - Reference to existing implementations (e.g., gateways, adaptors) and already available information products related with each interoperability issue. The term “information product” is a general term referring to any kind of information required during a simulation environment engineering process, e.g., an information product may be a document like the Federation Agreements Document (FAD) or a specific file like a scenario definition using the Military Scenario Definition Language (MSDL).
- Recommended information products:
  - Recommendation which information products would be useful for tackling this interoperability issue.
- Examples and prototypes of selected information products:
  - Identification of information products which can be considered as examples (according to the recommendations made before).

Especially by documenting the connections of the MSG-086 interoperability issues with the DSEEP and the FEAT, it is ensured that all interoperability issues have proper foundations in international standards for distributed simulation environments.

For clarity and maintainability the issues are categorized into 9 interoperability issue groups:

- Conceptual Model (CM);
- Federation Development (FD);
- Fidelity (FI);
- Infrastructure and tools (IN);
- LVC and C2-Sim coupling (LC);
- Organizational and Legal issues (OL);
- Scenario (SC);
- Synthetic Environment (SN); and
- Time Management (TM).

In the following sections, an overview of all issue groups and all issues identified by MSG-086 is given. Detailed documentation of all issues according to the presented schema may be found in Annex A.

3.1 ISSUE GROUP "CONCEPTUAL MODEL" (CM)

In order to analyze interoperability issues related to conceptual models it is necessary to understand the role of conceptual modeling in the development process for distributed simulation environments, i.e., the DSEEP [DSEEP]. The DSEEP contains "Step 2: Perform Conceptual Analysis" which is dedicated to the development of a conceptual model. According to the DSEEP the conceptual analysis step consists of three activities, as shown in Table 3-1.

Table 3-1: Activities of Conceptual Analysis Step of the DSEEP.

Activity	Resulting Information Product
2.1 Develop Scenario	Scenario
2.2 Develop Conceptual Model	Conceptual model
2.3 Develop Simulation Environment Requirements	Simulation environment requirements, Simulation environment test criteria

To understand the role of the conceptual model related to simulation interoperability problems, the dependencies of subsequent DSEEP steps of the conceptual analysis step and its information products need to be evaluated. If the conceptual modeling step of the DSEEP is not performed at all during the development of a simulation environment, the process will stop because the information products generated in this step are vital to subsequent steps. This is analyzed further in issue CM-01.

It is therefore assumed that the conceptual analysis step is carried out and that the information products are successfully generated. If this is the case, then there is no reason that the conceptual analysis step and its products will lead to any interoperability problems. The only way interoperability problems may result from conceptual analysis therefore is, that the quality of the information products is poor, caused, e.g., by incompleteness or inconsistencies. In the subsequent analysis it is assumed that this is the case.

Typical reasons for such inconsistencies or incompleteness may be:

- A lack of agreement on formats and notation (see issue CM-02);
- A lack of tools for conceptual modelling; and
- A lack of built-in support for consistency checks or transformation support in available tools for conceptual modeling (see issue CM-04).

Fortunately, the DSEEP defines the flow of information products throughout the simulation environment engineering process. It is therefore concluded, that an inconsistent or incomplete scenario will result in:

- Inconsistent or incomplete simulation environment design (DSEEP Step 3.2); and
- Inconsistent or incomplete simulation environment agreements (DSEEP Step 4.2).

Incomplete or inconsistent conceptual models or simulation environment agreements will result in:

- Selecting or designing member application that are not interoperable at the conceptual level (DSEEP Step 3.1 and 3.3) or just not including necessary member applications required to fulfil objectives;
- Inconsistent or incomplete simulation environment design (DSEEP Step 3.2);
- Inconsistent or incomplete simulation data exchange model (DSEEP Step 4.1); or
- Inconsistent or incomplete simulation environment agreements (DSEEP Step 4.2).

It is remarked here that the conceptual model information products are also consumed in DSEEP Steps 6.2 (Prepare output), 7.1 (Analyze data) and 7.2 (Evaluate results). However, these activities may only be performed if a working simulation environment has already produced results. As interoperability is a precondition for a working simulation environment, these activities are neglected in the analysis of interoperability problems caused by poor conceptual modeling.

First ideas for structuring the terminology regarding conceptual models in context of distributed simulation environments were proposed by MSG-058 [NMSG-058]. According to MSG-058 a distinction should be made between:

- The mission space conceptual model (describing the extract of real-world objects, situation and dynamics (“referent world”) which is to be represented in a simulation); and
- The simulation implementation space conceptual model (describing how things will be represented on an implementation/technical level).

Unfortunately, the terms “mission space conceptual model” and “simulation implementation space conceptual model” have not been precisely defined by MSG-058 and even the final report of MSG-058 does not follow a strict terminology. To avoid confusion, MSG-086 decided not to use the terms proposed by MSG-058, but to follow the terminology as used by the DSEEP.

Table 3-2: Conceptual Model Issues.

CM Issues	
CM-01	No explicit development of a conceptual model
CM-02	Missing standards or guidelines for development of a conceptual model for distributed simulation environments
CM-03	Missing reference conceptual models
CM-04	Lack of standard methodologies, techniques and tools for automated transition from conceptual models to Simulation Data Exchange Models (SDEMs) or (automated) comparison and verification between conceptual models and SDEMs
CM-05	Missing standards for application services

3.2 ISSUE GROUP "FEDERATION DEVELOPMENT" (FD)

The issue group "Federation Development" deals with interoperability issues that are related to either the design of a simulation environment or the simulation environment engineering process as a whole.

Issues regarding the design of simulation environment include problems related to the simulation architecture (e.g., HLA) or the interplay of multiple simulation architectures in a single simulation environment. Due to the dominance of HLA-based simulation environments, many of the issues in this group focus on HLA. However, it is stressed that most issues might also occur (slightly different, of course) when using other simulation architectures (e.g., DIS or TENA).

The issues regarding the simulation environment engineering process as a whole address topics that require special care during preparation and design of a simulation environment, like e.g., federation agreements.

Table 3-3: Federation Development Issues.

FD Issues	
FD-01	Multi-architecture simulation environments
FD-02	Different HLA versions
FD-03	Different FOM or SDEM versions
FD-04	Incompatible FOM modules
FD-05	Incomplete specification of federation agreements
FD-06	Missing comprehensive reference architectures
FD-07	Missing standards and templates for artifacts

3.3 ISSUE GROUP "FIDELITY" (FI)

Fidelity group deals with the fidelity issues described in ET-027 list of the aspects of simulation interoperability.

The NATO Modelling and Simulation Standards Profile Glossary defines fidelity as "The accuracy of the representation when compared to the real world" [AMSP-01]. Another definition states fidelity as "the degree to which a model or simulation reproduces the state and behavior of a real-world object or the perception of a real-world object, feature, condition, or chosen standard in a measurable or perceivable manner; a measure of the realism of a model or simulation; faithfulness" [99S-SIW-167].

Although the simulation community understands and uses fidelity in a general sense, a quantitative definition that delivers a well-defined tool for tackling multi-aspects and accompanying complexity of fidelity has not been agreed. Part of the complexity comes from the fact that the impact of fidelity considerations spans all of the simulation engineering lifecycle, starting from simulation objective definition (Step 1 of DSEEP), and moving along conceptual analysis (Step 2), environment design (Step 3), environment development (Step 4), integration and testing (Step 5), simulation execution (Step 6), and finally post processing (Step 7).

Another issue that contributes to the complexity is the semantic conjunction of fidelity with concepts such as accuracy, sensitivity, precision, resolution, repeatability, model/simulation validation (see [99S-SIW-167] for an in-depth discussion). Each of these concepts (including fidelity) signifies aspects that are closely related, at times to the extent of representing different facets of the same phenomenon, but at the same time require careful considerations to cater for important subtleties among them.

Looking from an interoperability point of view, the most important problem appears to be the lack of formalized or agreed "levels" or "representations" for various aspects of fidelity (such as behavioural fidelity, representational fidelity, visual, audio, temporal fidelity, etc.). Clearly lack of such agreements or formalizations introduces interpretation and compatibility problems at different levels of interoperability (see Andreas Tolk et al.'s work on "The Levels of Conceptual Interoperability Model"). In that respect, MSG-086 has consolidated findings of earlier work and identified several key interoperability issues in regard with fidelity. These key issues can be broadly categorized into three groups:

- Four of those issues (FI-01, FI-02, FI-03, FI-05) address the lack of agreement on representational fidelity levels (such as entity aggregations, entity resolutions and data resolutions), and possible ways of mappings between inconsistent representations;
- Issue FI-04 addresses the behavioral fidelity issues; and
- Issue FI-06 addresses the fidelity issues attributable to man-machine interaction inconsistencies.

Note that the first three issues FI-01, FI-02, and FI-03 can be considered as an indirect call for standardization efforts regarding the description of externally visible (or extrinsic) artifacts (i.e., entities and data) of simulations. FI-04, on the other hand, is the manifestation of a more realistic stand catering for legacy simulations too: if you cannot ensure compatible fidelity levels, try to compensate for mismatches via converters. The call here, however, is at least a standardized way of employing such mediator components. FI-05 is more related to intrinsic artifacts of simulations (i.e., behavioral aspects). Clearly, intrinsic aspects are more difficult to standardize. The implied call here is at least to try to form a common catalogue of well-known algorithms (such as computation of physical phenomena).

The relevance of these issues to DSEEP steps and activities, and also to levels of conceptual interoperability is discussed in detail in the corresponding issue descriptions in Table 3-4.

Table 3-4: Fidelity Issues.

FI Issues	
FI-01	Lack of formalized description for entity aggregations
FI-02	Lack of agreed levels for entity resolution
FI-03	Lack of agreed classifications for data fidelity levels
FI-04	Lack of formalized transfer mediation functions between incompatible entity resolution and or data fidelity levels
FI-05	Lack of agreed critical behaviours and corresponding algorithms
FI-06	Inconsistent Human Machine Interfaces

3.4 ISSUE GROUP "INFRASTRUCTURE AND TOOLS" (IN)

Infrastructure refers to services and facilities necessary for the operation of a simulation environment. Infrastructure is both hardware and software. The hardware part of the infrastructure consists among other things of computers and network equipment. The software part of the infrastructure is for example in HLA the RTI (Runtime Infrastructure) and the implementation of different network protocols.

Without a well working infrastructure, interoperability in a simulation environment will encounter problems and issues. Issues may arise in both the hardware and the software part of the infrastructure and also in the use of the infrastructure.

To avoid issues with the use of an infrastructure, there is a need for well documented standards. The design and configuration of an infrastructure is an important task for which appropriate tools are necessary. Also the experience and knowledge among the engineers is a very important part in the planning and design phase of a simulation environment.

Appropriate tools are necessary to operate and control an infrastructure and the execution of a simulation environment. For example, tools for inspection and supervision of the physical network, computers, and software parts are of great value when data overflows occur. To take care of issues and problems such as inconsistent data marshalling during execution appropriate tools are necessary for fault management and analysis of the situation.

Three different interoperability problems are identified and described in three issues noted in the table below.

Table 3-5: Infrastructure and Tools Issues.

IN Issues	
IN-01	Inconsistent data marshalling
IN-02	Data overflow
IN-03	Proper network configuration

3.5 ISSUE GROUP "LVC AND C2-SIM COUPLING" (LC)

The LVC and C2-Sim coupling issue group deals with issues concerning the coupling between different types of systems, like Live, Virtual and Constructive simulation systems and C2 systems.

The DoD Glossary [DoD M&S Glossary] describes Live, Virtual and Constructive simulations as follows:

- Live simulation: A simulation involving real people operating real systems.
- Virtual simulation: A simulation involving real people operating simulated systems. Virtual simulations inject human-in-the-loop in a central role by exercising motor control skills (e.g., flying an airplane), decision skills (e.g., committing fire control resources to action), or communication skills (e.g., as members of a C4I team).
- Constructive model or simulation: Models and simulations that involve simulated people operating simulated systems. Real people stimulate (make inputs) to such simulations, but are not involved in determining the outcomes.

There are currently few defined standard or guidelines for connecting live systems to virtual and constructive simulations. Standards like DIS, HLA and TENA support some aspects of defining and exchanging data between simulation systems and live systems, but provide little or no specific guidance for the architecture and design of a distributed simulation environment containing live/real systems. The agreements of a distributed simulation environments depend on the purpose, goal and requirements of the simulation environment and therefore the interoperability issues of bringing in a live system in the simulation environment can be very different from case to case. However, live systems often exhibit some common characteristics that may cause specific problems when integrating with virtual and constructive simulation systems, many of the problems that may arise are not unique to LVC coupling and can be tracked to other issues.

One special case of LVC coupling is the use of C2 systems in connection with simulation systems. “C2-Sim Coupling” refers to the coupling of any C2 system with any simulation system. The purpose of such a coupling might be, for example, training, mission planning or decision support. There are different ways to couple C2 systems and simulation systems. In general, one is trying to build a gateway which exposes the C2 interfaces on one side and the simulation interfaces on the other side. The gateway then has to be able to translate the C2 information to the simulation systems and vice versa. Another possibility is that a simulation system can directly interact with a C2 system or vice versa. Table 3-6 specifies the five issues associated with LVC and C2-Sim coupling.

Table 3-6: LVC and C2-Sim Coupling Issues.

LC Issues	
LC-01	Limitations due to integration of live systems
LC-02	Difference in accuracy for position
LC-03	Missing information exchange between real and simulated environment
LC-04	Limitations in coupling simulations and live systems when using operational protocols of live systems
LC-05	Limited simulation awareness of C2 systems

3.6 ISSUE GROUP “ORGANIZATIONAL AND LEGAL” (OL)

During the development of a simulation environment not only technical issues are encountered. For example also legal, political or cultural issues can play a role during the development of a distributed simulation environment. This issue group lists some of the most commonly encountered non-technical issues.

Given that the issues in this group are not of a technical nature, relating them to the LCIM levels is not really applicable. Instead, the issues can be considered part of the legal/organisational sidebar that MSG-086 has added to the LCIM diagram. In general these issues do not inhibit reaching a certain level of interoperability; however they make the process of getting there much harder and more time-consuming. But in certain cases the restrictions might be so severe that they could prevent interoperability at all.

For most of these issues, the solution is to identify them as early as possible during the development process and then try to minimize their impact by managing them correctly. Given that the solution for these issues is a management task and not at the technical level, the different issue descriptions in this issue group will not list solution approaches or recommended information products.

The Task Group combined issues from the ET-027 list and from the DSEEP Multi-Architecture Overlay (DMAO) [DMAO]. Table 3-7 specifies the organisational and legal issues.

Table 3-7: Organizational and Legal Issues.

OL Issues	
OL-01	Improper project management
OL-02	Missing or limited policy, coordination and cooperation
OL-03	Cultural aspects
OL-04	Lack of required personnel
OL-05	Selection of incompatible member applications
OL-06	Legal or political restrictions apply

3.7 ISSUE GROUP “SCENARIO” (SC)

Scenarios play an important role in planning, engineering and executing a distributed simulation environment. During the simulation environment engineering process the operational scenarios provided by the user are refined into one or more conceptual scenarios and finally the executable scenarios are derived which are used for initializing and stimulating the participating simulation systems and other member applications.

The DSEEP already requires the user to specify scenarios as part of the requirements for an intended simulation environment. These scenarios describe the intended purpose of a simulation environment and are authoritative sources of requirements for the simulation engineers that design and set up a simulation environment. Poorly specified scenarios regularly result in unacceptable or inappropriate simulation environments. Therefore, the importance of well-specified scenarios in the simulation environment engineering process can hardly be overestimated.

The issue group “Scenario” contains all interoperability issues related to scenarios used within a (distributed) simulation environment.

A definition of the term “Scenario” and a detailed discussion of the three types of scenarios which are developed in the simulation environment engineering process (operational scenario, conceptual scenario, and executable scenario) are given in the information product “Scenario” [ScenarioGuideline].

Table 3-8: Scenario Issues.

SC Issues	
SC-01	Missing authoritative operational scenarios
SC-02	Incomplete or inconsistent operational scenario description
SC-03	Missing formal scenario specification
SC-04	Use of different formats for executable scenarios
SC-05	Use of different doctrines and ROEs
SC-06	Missing or incomplete definition of application domain

3.8 ISSUE GROUP “SYNTHETIC ENVIRONMENT” (SN)

The following definition of Synthetic Environment is adopted by MSG-086. The definition is adapted to an architecture independent definition from [DOD M&S Glossary, 1P2.19.85 Synthetic Environment]:

> The Synthetic Environment is the integrated set of data elements that define the environment within which a given simulation application operates. The data elements include information about the initial and subsequent states of the terrain including cultural features, and atmospheric and oceanographic environments throughout a simulation exercise. The data elements include databases of externally observable information about instantiable simulated entities, and are adequately correlated for the type of exercise to be performed.

It should be noted that this definition does not limit the synthetic environment to the visual spectrum only. It includes all active emissions and passive reflections in the whole electromagnetic spectrum, but for example also observable movement of air or water masses. This means the synthetic environment should support observations made by different devices, for example visual, infrared, radar or sonar.

The synthetic environment can be divided into different segments. Common examples of such segments are the Synthetic Natural Environment (SNE) or the Synthetic Human-made Environment (SHME). Within MSG-086 the following division of the synthetic environment is used. All of the identified issues apply to each of these segments, although some issues might be more troublesome on some of the segments:

- Natural terrain and human-made structures;
- Instantiable simulated entities; and
- Atmospheric and bathymetric conditions.

The different interoperability problems related to the synthetic environment have been organised into four issues. These are:

- Correlation issues with the data used for the synthetic environment. This issue is addressed further in SN-01.
- Interoperability issues resulting from the usage different levels of fidelity in different simulation systems. This issue is addressed further in SN-02. This issue is also related to the more general fidelity issues of the FI issue group.
- Issues with the exchange of the synthetic environment data before the execution of the simulation. This issue is addressed further in SN-03.
- Issues with the exchange of synthetic environment updates during the execution of the simulation. This issue is addressed further in SN-04.

Issues with legal or political restrictions on the usage of synthetic environments are also common; this issue is addressed further in OL-06.

Table 3-9: Synthetic Environment Issues.

SN Issues	
SN-01	Synthetic environment data is not correlated
SN-02	Different levels of fidelity are used for synthetic environment
SN-03	Difficult to exchange synthetic environments before runtime
SN-04	Difficult to exchange synthetic environment updates at runtime

3.9 ISSUE GROUP “TIME MANAGEMENT” (TM)

This issue group deals with problems concerning the time management schemes used by member applications in simulation environments.

A member application executes a model over time. A simulation environment is the execution of multiple models over time that interoperate by exchanging data. The exchange of data can be within a process, between processes in a single computer, between computers on a LAN or across a WAN. The data exchange between member applications in a simulation environment is synchronized (managed) to ensure that the delivery of data is timely with respect to the requirements of the simulation environment.

Management of timely delivery of data is termed “Time Management” and requires agreements among participating member applications on time representation and time advancement.

Fidelity requirements for the simulation environment affect the resolution of time. The architecture and design of the simulation environment and the allocation of modeling responsibilities affect which member application are time constrained and/or time regulating.

Differences in time representation, and different time-advancement schemas within member applications and the synchronization of time between member applications are identified as the main time management issues. Network latency can also cause problems when not using a time management approach that guarantees that events are delivered in order according to their time-stamp. This is common in real-time simulation environments where member applications just time stamp their events before sending the event and the infrastructure then distributes the events to consumers without any measure regarding the time stamp.

All member applications in a simulation environment have to agree on a shared binary representation of simulation time and time synchronization methods. Each model may have different internal time representations, depending on the model requirements, software technologies and languages used. If the shared representation is different from the member application’s internal representation then it has to provide a conversion between the shared representation and the internal representation. This conversion has to be accurate enough that any loss of information during the conversion does not affect the validity of the simulation environment. It is not uncommon that information is shared at a lower time resolution than the internal models use.

In order to understand time representations there are several important concepts that need to be understood. According to the NATO Modelling and Simulation Standards Profile [AMSP-01] the following definitions that are somehow related to time:

- **Real Time:** In real-time modeling and simulation, simulated time advances at the same rate as actual time; for example, running the simulation for one second results in the model advancing time by one second. Contrast with: fast time; slow time.
- **Scalability:** The ability of a distributed simulation to maintain time and spatial consistency as the number of entities and accompanying interactions increase.

Definitions provided by [11S-SIW-049] are:

- **Wall-Clock Time:** This is the actual time in the real world when a member application is executed.
- **Simulation Time or Logical Time:** Representation of physical time within a simulation environment.
- **Time Step or Time Increment:** This is the value by which a member application repeatedly increments its simulation time. Many virtual simulators (like flight simulators) use a constant time step. These are sometimes referred to as “frame based “since the state in each step is visualized in a frame of a visualization system. The time step may also vary. In event-driven simulation the time step will typically be the time to the next known event.
- **Simulation Executive Loop:** The main loop in a member application that increments the time value is sometimes known as the simulation executive loop.
- **Time Representation:** This is the binary representation of the time value, for example a 32-bit little-endian integer or a 64-bit big-endian IEEE-754 floating-point.
- **Time Interpretation:** This is the interpretation of the time value used by the domain model. An integer representation with the value of 47 may be interpreted as 47 seconds or 47 days, depending on the interpretation used.
- **Time Implementation:** This is program code, usually a set of object-oriented classes, which can represent and perform calculation on time values, such as time stamps and time increments.
- **Time Management:** Methods and services for managing the time advancement in a simulation environment, as well as the exchange of time-stamped information.

Additional definitions:

- **Time Constrained:** In a simulation environment the participating member applications may or may not be time constrained. A time-constrained member application is not allowed to move forward in time unless the constraints are met. An example of a constraint is to match the pace to an external clock, e.g., system clock to simulate real time. A member application can be unconstrained but in a simulation environment the events of a member application often need to be synchronized with other member applications.
- **Time Regulating:** Member applications that limit the pace of other constrained member applications are called time regulating member applications.
- **Time Advancement:** The advancement of time in a member application can follow different time management methods, e.g.,:
    - A time-stepped member application advances time in steps of time increments. These increments can be of a pre-determined fixed length or (as supported by HLA) a length that can dynamically change during execution.
    - Event-driven member applications jump forward in time to the next event regardless of if it lies near or far in the future.
    - In “Optimistic Time Warp” the calculations are made ahead of the current time, and if events arrive in the past, then the calculations are retracted.

- **Time Synchronized Data:** Data is time stamped and data exchange services can be used to ensure that the delivery of data is done before granting the member applications to step forward in time. The services, algorithms and implementation used for determining when time advancement is granted can be different depending on the purpose of the member application and which simulation environment standard is being used. In some situations a mix of different methods can be applied which requires that additional care and attention is put on ensuring consistent time management.

- **The Basic Application Loop:**
  1) Process received data;
  2) Calculate and update data for the next time-period;
  3) Request to move forward in time (time advance request/next event request); and
  4) When constraints are met, the member application is granted to step forward.

Table 3-10: Time Management Issues.

TM Issues	
TM-01	TM-01 Temporal anomalies caused by differences in precision of time representation
TM-02	TM-02 Temporal anomalies caused by differences in time resolution
TM-03	TM-03 Temporal anomalies caused by unsynchronized time
TM-04	TM-04 Temporal anomalies caused by network latency

3.10 ISSUE GROUPS AND STATISTICS

3.10.1 Rationale

The general approach chosen by MSG-086 to investigate simulation interoperability issues has been discussed in Section 1.3. The issues identified by ET-027 and MSG-086 have been categorized into several interoperability issue groups. Each issue has been related to the corresponding levels of the LCIM model [03F-SIW-007] and the steps of the DSEEP [DSEEP].

This mapping of issues to LCIM and DSEEP naturally raises the question for additional findings that can be drawn from this mapping:

- Which DSEEP steps or LCIM levels are affected most/least by simulation interoperability issues?
- Why are some DSEEP steps or LCIM levels more sensitive to simulation interoperability issues than others?
- As a consequence, which DSEEP steps would benefit most from an improvement, e.g., an information product template?

In order to answer these questions, the statistical approach described in this section is used.

3.10.2 Procedure

The mapping of issues to LCIM levels or DSEEP steps forms a matrix, each row representing an issue and each column representing a LCIM level or DSEEP step respectively. This is illustrated in the following table (Table 3-11), mapping the scenario issues to LCIM levels.

Table 3-11: Equal Weight is Assigned to Every LCIM Level an Interoperability Issue is Mapped To.

Number	Issue Title	LCIM Level						
		1	2	3	4	5	6	7
SC-01	Missing authoritative operational scenarios				0.25	0.25	0.25	0.25
SC-02	Incomplete or inconsistent operational scenario description				0.25	0.25	0.25	0.25
SC-03	Missing formal scenario specification			0.25	0.25	0.25	0.25	
SC-04	Use of different formats for executable scenarios		0.33	0.33	0.33			
SC-05	Use of different doctrines and ROEs					0.33	0.33	0.33
SC-06	Missing or incomplete definition of application domain					0.33	0.33	0.33

In many cases an issue is related to several LCIM levels. In these cases a weight factor to each cell is assigned in order to give the issue a total weight of 1. Consider for example the SC-03 issue in Table 3-11 which has been mapped to LCIM level 3 to 6 with an equal weight of 0.25. Following this procedure, the columns of the matrix are summed up and the total weight assigned to each of the LCIM levels is obtained as the result. The more issues contribute to an LCIM level and the higher the weights of the individually contributing issues are, the higher the resulting weight of the LCIM level will be and the more "important" this particular level may be considered. The assignment of different weights for each LCIM level affected by an issue would have been an option for some issues, where some LCIM level feel to be more affected than others. However, this would have opened the door for arbitrariness, because there is no absolute measure for the strength of the relation between a simulation interoperability issue and a LCIM level. The assignment of equal weight chosen here therefore improves the statistical quality of the analysis.

A similar procedure is applied to the mapping from interoperability issues to DSEEP steps, thus resulting in weight factors attributed to the individual DSEEP steps and activities.

3.10.3 Results and Interpretation

3.10.3.1 LCIM

Following the procedure described in the previous section, the following result (Figure 3-1) is obtained.

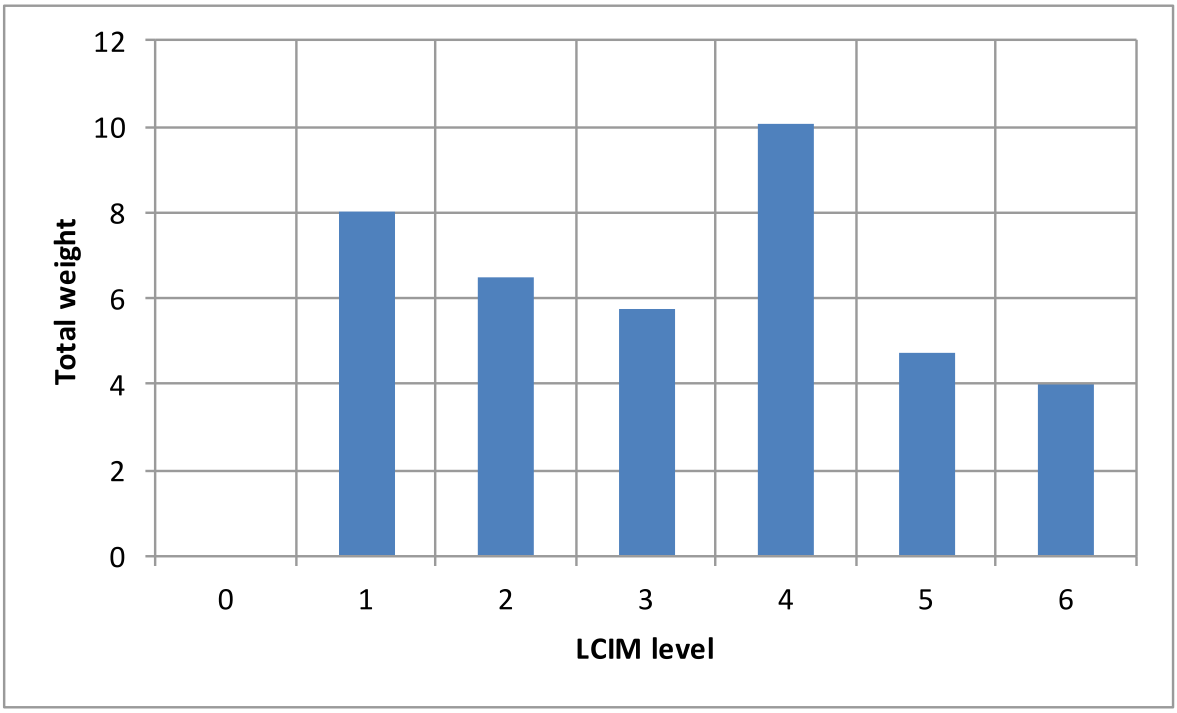


Figure 3-1: Summing up the Individual Weight Assigned by Each Interoperability Issue to a LCIM Level, the Total Weight of a LCIM Level is Calculated.

Due to the limited number of issues per LCIM level some of the total weights may have rather poor statistics and therefore may be interpreted as hints or trend rather than reliable or statistically significant values.

From Figure 3-1 it is observed that LCIM level 0 is not affected by interoperability issues. This is obvious because level 0 corresponds to technical connectivity and assigning a level 0 means that there is no connection between the individual systems. However, most of the interoperability issues discussed will appear in connected systems, so at least technical connectivity is a precondition for the occurrence of most of the interoperability issues.

Figure 3-1 also shows that levels 1 to 6 of the LCIM are approximately equally affected by simulation interoperability issues except the outstanding level 4. From this it may be concluded that the work of MSG-086 addresses all levels of interoperability without any preferences. This means that the collection of simulation interoperability issues is complete in the sense that all LCIM levels are investigated by MSG-086. It is also observed that level 4 (pragmatic interoperability) plays an outstanding role with respect to the interoperability issues discussed in this report. A reason for this may be that on one hand in present simulation environments a maximum of level 4 interoperability is reached, whereas levels 5 and 6 are not yet explored to the extent of the lower levels. On the other hand, working solution approaches are known for interoperability issues arising from the syntactic and semantic levels, thus producing a lower amount of issues with respect to these levels.

3.10.3.2 DSEEP Steps

Applying the approach from Section 3.10.2 to the mapping of simulation interoperability issues to DSEEP steps, the following figure (Figure 3-2) is obtained.

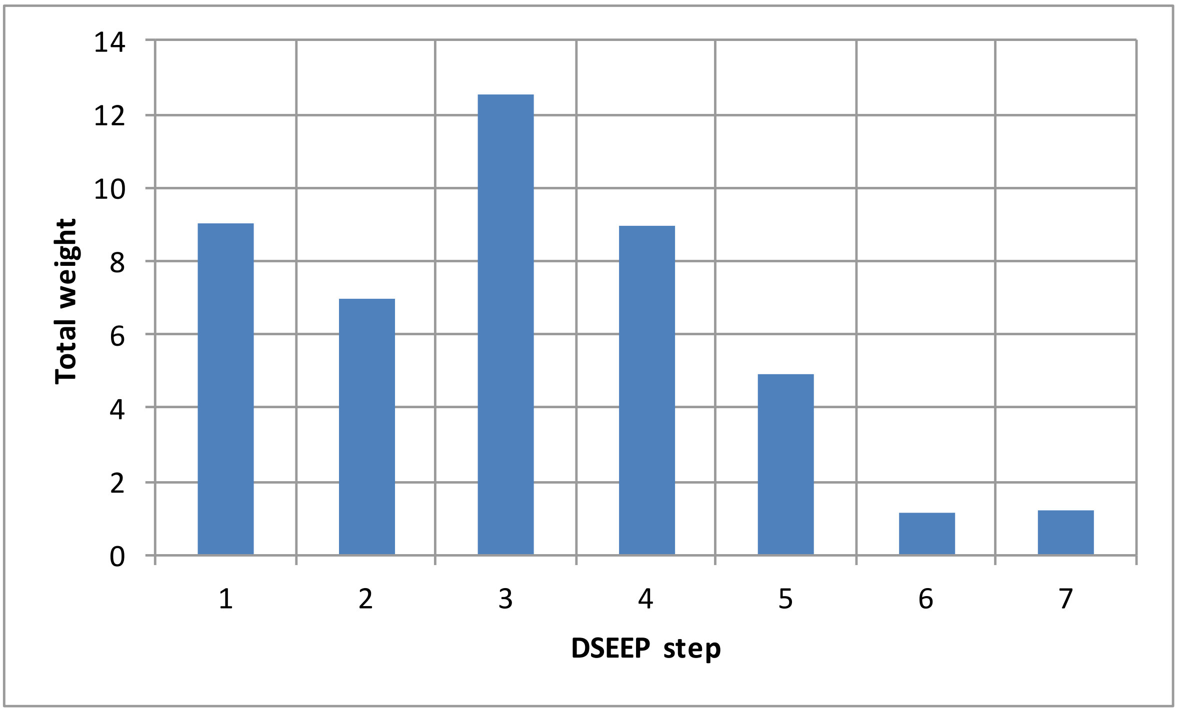


Figure 3-2: Following the Described Approach a Total Weight of Each DSEEP Step with Respect to Simulation Interoperability Can be Computed.

It is observed that most of the simulation interoperability issues discussed can be assigned to DSEEP steps:

- Step 1 – Define simulation environment objectives;
- Step 2 – Perform conceptual analysis;
- Step 3 – Design simulation environment; and
- Step 4 – Develop simulation environment.

These four steps form the planning, design and development phase of the process. A conclusion may be that many simulation interoperability problems can be avoided by proper planning, design and development of a simulation environment. This well-known practical experience again is confirmed by the presented statistical evaluation.

In contrast Steps 5 to 7 (integration, execution and analysis) seem to be a minor source of interoperability problems. An interpretation may be that simulation interoperability issues emerging in earlier DSEEP steps prevent the process from reaching Steps 5 to 7. So, if a distributed simulation environment has reached Step 5, the majority of interoperability problems has already been solved for this particular simulation environment. This seems to be in contradiction to the experience gained from implemented simulation environments where many interoperability problems often occur in these phases of a simulation environment execution. However, this must be seen in comparison to the number of interoperability problems which already have been solved to allow a simulation environment execution to reach this phase. Also, many practical problems experienced in these phases have nothing to do with simulation interoperability and therefore are not investigated by MSG-086 (e.g., hardware integration problems, missing execution plan or unclear analysis requirements). Another reason for this observation might be that the mapping from simulation interoperability issues to DSEEP steps is rather general in the sense that it does not distinguish between the true origin of a problem and its occurrence or observation, the latter being more prominent and therefore possibly creating a bias in the statistical evaluation.

3.10.3.3 DSEEP Activities

Further detailing the results from Section 3.10.3.2 the simulation interoperability issues are mapped to the activities of the DSEEP. The result is shown in the following figure (Figure 3-3).

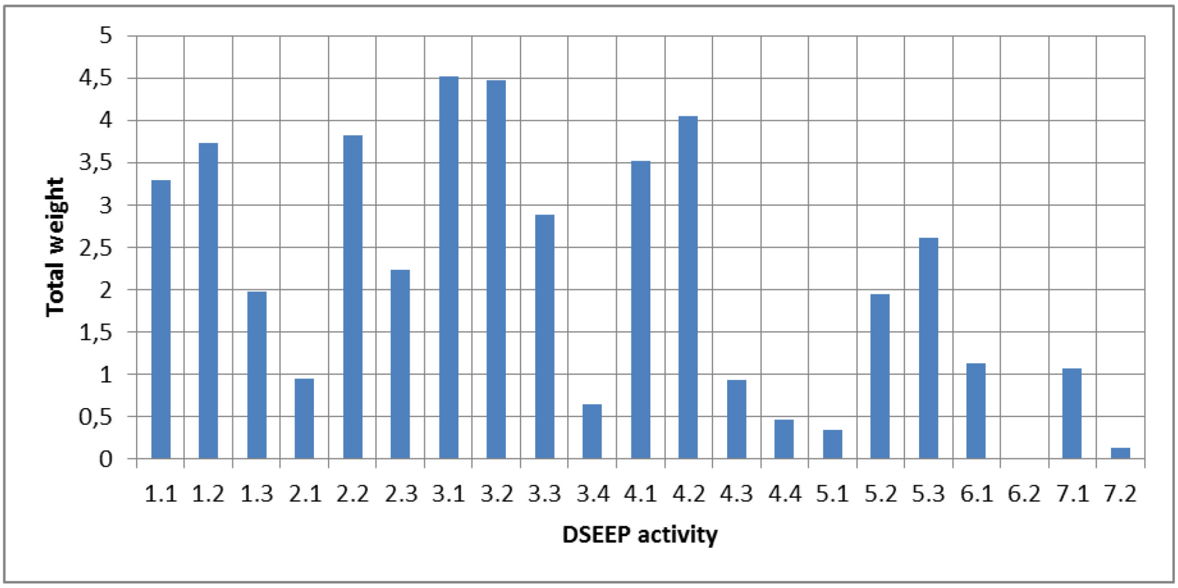


Figure 3-3: The Total Weight of Each DSEEP Activity is Calculated Following the Procedure Described in the Text.

Similar to Section 3.10.3.2 it can be concluded that the most important DSEEP activities regarding simulation interoperability are activities:

- Step 3.1 – Select member applications;
- Step 3.2 – Design simulation environment; and
- Step 4.2 – Establish simulation environment agreements.

The order of this list is from highest to lowest value of the total weight. Again it must be stated that due to poor sample size it cannot be expected that these results have a high statistical significance. This means that the order may give a hint on the importance of a particular DSEEP activity but not an absolute ranking.

As expected, the selection of member application and the development of a simulation environment play the most important role for simulation interoperability. The prominent peak at Activity 4.2 seems to be surprising at a first glance, but may be attributed to the fact, that this activity under its label "establish simulation environment agreements" collects many topics of rather low conceptual but rather high |practical importance, e.g., databases, time management, synchronization points, save/restore procedures, data distribution and security. Although each of these topics seems rather "small", it will prevent the whole simulation environment from executing properly. This emphasizes the importance of an agreement on these "minor" topics in order to successfully operate a simulation environment comprising many individual simulation systems.

The question raised at the beginning of this section, which DSEEP steps or LCIM levels are affected most by simulation interoperability issues, therefore has its answer in Figure 3-1 and Figure 3-3. Activities 3.1, 3.2 and 4.2 are affected most, so it can be concluded, that defining information products which cover the actions described in these activities, will have the highest impact.

Chapter 4 – CONCLUSIONS AND RECOMMENDATIONS

Based on an extensive investigation of current simulation interoperability issues MSG-086 evaluated the feasibility of standardized information products in context of the DSEEP to improve simulation interoperability.

4.1 CONCLUSIONS

A major finding of MSG-086 is that simulation interoperability is not primarily a technical issue, but that simulation interoperability needs to be addressed in a holistic way along the whole simulation environment engineering process (e.g., DSEEP). Achieving simulation interoperability requires efforts and standardization on the technical, the syntactic, the semantic, and the pragmatic level. Focusing only on standards for distributed systems or reuse of components will not lead to simulation interoperability on higher levels.

It is important to stress that interoperability of simulation systems and further assets (e.g., C2 systems) strongly depends on the specific application purpose. Interoperability is not an absolute characteristic of a system but can only be defined (and verified) for a specific application purpose and specific operational requirements.

Following the widely used Levels of Conceptual Interoperability Model (LCIM), MSG-086 observed that achieving technical interoperability (LCIM Level 1) is usually easily possible. Nevertheless, achieving technical interoperability still requires attention (e.g., proper network configuration) and cannot be neglected in daily practice. Similarly, syntactic and semantic interoperability (LCIM levels 2 and 3) are well achievable with current standards (e.g., using HLA and a reference FOM like the RPR-FOM).

The major challenges regarding simulation interoperability appear on the higher levels of interoperability, i.e., with regards to pragmatic, dynamic, and conceptual interoperability (LCIM Levels 4, 5, and 6). Interoperability on these levels is significantly harder to achieve as a detailed interoperability analysis of participating systems is required for each simulation environment and current standards do not address these interoperability levels. A detailed interoperability analysis requires a good specification of the users' requirements as well as substantial documentation of all involved systems. Both are often hard to obtain.

Current standards related to simulation interoperability address only technical, syntactic, and semantic interoperability (LCIM Levels 1 – 3) and thus do not provide any guidance for achieving interoperability on higher levels. In order to significantly improve simulation interoperability, standardization and guidance on higher levels (LCIM Level 4 and above) is required. This may include process guidelines and documentation templates (e.g., to ensure proper requirements specification, or to guide conceptual modelling activities) as well as standardized components (e.g., application services).

Complementary to the original LCIM interoperability levels, MSG-086 stresses that simulation interoperability may also be significantly influenced by organizational, cultural, legal and project management issues.

4.2 RECOMMENDATIONS FOR INFORMATION PRODUCT “SCENARIO”

Based on the identified interoperability issues and their impact on the simulation environment engineering process, MSG-086 chose to focus its further efforts on the issue group “Scenario”. To improve simulation interoperability in context of the DSEEP and in accordance with its Technical Activity Proposal [TAP_MSG086], MSG-086 worked out a proposal regarding content and structure of an information product for scenario development. The title of this information product is “Guideline on Scenario Development for (Distributed) Simulation Environments” [ScenarioGuideline].

The primary objective of the “Guideline on Scenario Development for (Distributed) Simulation Environments” is to provide comprehensive guidance on scenario development by defining a clear terminology, by describing relations between different steps of the scenario development process, by providing documentation templates, and by giving an overview of existing standards and tools for scenario development.

The “Guideline on Scenario Development for (Distributed) Simulation Environment” is based on the DSEEP and augments respectively refines the DSEEP with respect to scenario development.

It is the expressed intention of MSG-086 to make the “Guideline on Scenario Development for (Distributed) Simulation Environments” available to a broad community and especially to SISO with the objective of developing an official SISO guidance document. The “Guideline on Scenario Development for (Distributed) Simulation Environments” as developed by MSG-086 could serve as a first draft document of a respective SISO PDG and could finally be developed into a SISO guidance document. Within this process, recommendations and best practices regarding scenario development (e.g., which standards should be used) should be integrated into the document.

For purposes of easy reuse and distribution, the “Guideline on Scenario Development for (Distributed) Simulation Environments” was extracted from the main part of this final report and is provided as a separate document.

4.3 RECOMMENDATIONS FOR NATO AMSP-01

The primary purpose of the Allied Modeling and Simulation Publication (AMSP) 01, the NATO Modeling and Simulation Standards Profile, established by the MS3 of the NMSG, is to provide guidance on the selection and use of Modeling and Simulation (M&S) standards to promote interoperability in context with the objective “Establish a Common Technical Framework to Foster Interoperability and Reuse” of the NATO M&S Master Plan [NATO_MSMP]. It is stressed here that the term “interoperability” not only embraces interoperability between simulation systems, but also interoperability between simulation systems and live systems and C2-systems.

The first version of the NATO M&S Master Plan, which was published 1998, defined M&S interoperability as the “… capability for simulations to physically interconnect, to provide (and receive) services to (and from) other simulations, to use these exchanged services in order to effectively work together”. Meanwhile it is clear that this definition should be expanded to: “M&S interoperability is the capability for simulations to physically interconnect, to provide (and receive) services to (and from) other simulations, to use these exchanged services in order to effectively work together in a logically consistent way”.

AMSP-01 basically realizes the problem:

> “The definition… given in the… Masterplan (1998)… refers mainly to “technical interoperability”, that means the possibility to physically interconnect and communicate. A lot of additional work has to be done after interconnection is ensured, to reach higher levels of interoperability (semantic or substantive interoperability)”. [AMSP-01, Chapter 2.3]

Because of lack of standards dealing with higher levels of interoperability AMSP-01 is presently limited to interoperability standards related to Steps 3, 4, 5 and 6 of the DSEEP (e.g., HLA, DIS, TENA). These DSEEP steps address design, development, test and execution aspects of simulation environments. Aspects and questions which are addressed in DSEEP Steps 1 and 2, i.e., issues in context with the definition of the application space and problem space of a simulation environment to be developed, are not addressed. As these steps determine the conceptual model of a simulation environment they are of paramount importance to reach higher levels of interoperability.

AMSP-01 is an excellent overview of the current situation regarding M&S standards. It identifies the fundamental problem of today´s simulation application (lack of “substantive” interoperability) and points out corresponding blank areas of required standardization. However, the background and meaning of the term “substantive simulation interoperability” and the consequences of missing substantive simulation interoperability are not further elaborated and explained.

It is of fundamental importance to stress and explain the limits of present M&S solutions, particularly in basic M&S documents like the AMSP-01. It is equally important to direct the attention of involved decision makers to the essential questions addressed in DSEEP Steps 1 and 2, including the necessity to focus on modeling issues and corresponding standards (in contrast to today´s focus on systems development and application standards).

Therefore the following adjustments and additions to the AMSP-01 are recommended:

- Add a chapter or paragraph explaining the term “interoperability” in its totality using the structured approach of the LCIM and include examples for problems related to the individual levels of the LCIM (both may be taken from this report);
- Adjust existing AMSP-01 chapters correspondingly. Replace the term “substantive interoperability” by corresponding terms from the LCIM;
- Stress the particular importance of standards in context with DSEEP Steps 1 and 2, formulate corresponding requirements, point out difficulties to be expected in this context (e.g., security issues, proprietary issues);
- Address other concepts of higher level simulation interoperability (e.g., service-based simulation) and necessary standardization requirements; and
- Evaluate whether standards related to DSEEP Steps 3 – 5 need to be updated and initiate update as required.

In addition to these recommendations MSG-086 made two proposals for updates of the Gaps chapter of the AMSP-01. The proposals describe the information product “Guideline on Scenario Development for (Distributed) Simulation Environments” and the issue catalogue that MSG-086 has developed. Both proposals were submitted to MS3 in July 2013 for consideration during development of AMSP-01 version “C”.

4.4 RECOMMENDATIONS FOR NATO MORS M&S GAP ANALYSIS QUESTIONNAIRE

The MORS M&S Gap Analysis Questionnaire is a document that is supposed to be reviewed and modified at each NMSG business meeting. Although it has no official standing and does not reflect any official or legally binding national position or commitment, it may be used as a tool for assessing and prioritizing identified M&S gaps. An up-to-date M&S Gap Analysis Questionnaire will help to get a common understanding of existing M&S gaps, their structure and relations, and to direct further activities on closing prioritized gaps thus maximizing the benefit of such actions for nations and NATO bodies.

The present questionnaire is based on the US M&S Cross-Cutting and Business Plan from November 2006 [US CCBP], without any additional inputs from other nations to the gaps listed therein. Since that time the list of gaps in the questionnaire was never updated, and corresponding to the present state of knowledge it is at least in parts obsolete (if not wrong). The efforts taken by Nations to assess the individual gaps in this list therefore are rendered less useful as new important gaps may be missing and already solved problems may be attributed a high priority.

In order to keep further activities focused on the most important and current M&S gaps, the M&S Gap Analysis Questionnaire is considered to be a valuable tool and should be kept up to date. Simulation interoperability gaps form a subset of the M&S gaps addressed by the questionnaire and it is recommended that the corresponding subsection should be structured according to the structure of simulation interoperability discussed in this report, i.e., the questionnaire should adopt the issue groups and issues described in Chapter 3 and Annex A. Furthermore an alignment between issues discussed in this report and issues discussed in NIAG SG 162 seems to be reasonable.

4.5 RECOMMENDATIONS FOR FUTURE NMSG ACTIVITIES

The evaluation of the MSG-086 issue groups leads to the following recommendations for future NMSG research activities in context with simulation interoperability.

4.5.1 Investigate "M&S as a Service"

Missing are simulation standards and agreements that focus on higher levels of simulation interoperability. Based on the analysis done by MSG-086 service-based approaches seem to be a promising approach for future simulation environments. Different types of services and their impact on simulation interoperability should be analysed:

- Application services provide dedicated capabilities for a simulation environment. This may include (standard-) algorithms for critical or often used entity behaviors for different fidelity levels (e.g., calculation of weapon effects, computation of communication effects) or more technical functionalities (e.g., standardized transformation rules between entities of agreed (standardized) aggregation levels); and
- Infrastructure services provide capabilities for management and execution of a simulation environment (i.e., standardized federation execution control- and monitoring patterns and investigation of possibilities for (automated / standardized) federation execution control / monitoring / infrastructure services).

Based on the analysis, a new task group should go one step further and develop corresponding (standard-) interfaces for services.

First investigations regarding M&S as a Service have been conducted by ET-34 and MSG-131.

4.5.2 Investigate and Develop Information Products

MSG-086 recommends to investigate further information products that are required during the first steps of a simulation environment engineering process and to develop those information products if necessary.

Two primary candidates for future information products are the user requirements and the conceptual model:

- User requirements (DSEEP Step 1): Investigation of the feasibility to establish a standard, guideline, or template to describe the application domain of a simulation system to be developed. Investigation of the feasibility to establish a standard, guideline, or template to describe operational scenarios as basis for the development of the problem (mission) space of a simulation system to be developed. Assessment of existing ontologies (e.g., DCMF and MiSO [Mojtahed2005], [Mojtahed2008]) and related standards and elaboration of a (standard) military ontology to be used for the definition of the problem (mission) space and the development of the conceptual model of military simulation; and
- Conceptual model (DSEEP Step 2): Elaboration of recommended practices or guidelines for the development of conceptual scenarios and conceptual models for distributed simulation environments.

Assessment of the outcome of the corresponding SISO SCM PDG and the MSG-058 results. Investigation of the feasibility of application domain specific standardized aggregation levels for common entities in distributed simulation environments.

A possible business model could be to develop those information products within the NMSG and to handover the results to SISO for standardization and publication. This approach is foreseen for the "Guideline on Scenario Development for (Distributed) Simulation Environments" developed by MSG-086, and a similar approach was used by MSG-073 that supported the SISO GM-VV PDG.

4.5.3 Investigate Possibilities for Formal Standards and Support Tools

MSG-086 recommends investigating possibilities for developing and using formal standards within M&S applications. Many potential solution approaches for interoperability issues documented by MSG-086 include the requirements for more precise and more complete specifications. Formal standards may help to solve these issues.

Potential areas for investigation of formal standards may include:

- Investigation of the feasibility for formalized description of Rules of Engagement; and
- Development of recommended practices, methodologies and tools (if feasible standards) for the transition of conceptual models to Simulation Data Exchange Models (SDEMs).

Investigation and development of formal standards – especially if they are related to operational concerns (like rules of engagement) – should be done in close collaboration with the operational user community and should consider reuse of existing operational standards.

4.6 RECOMMENDATIONS FOR SISO WORKING GROUPS

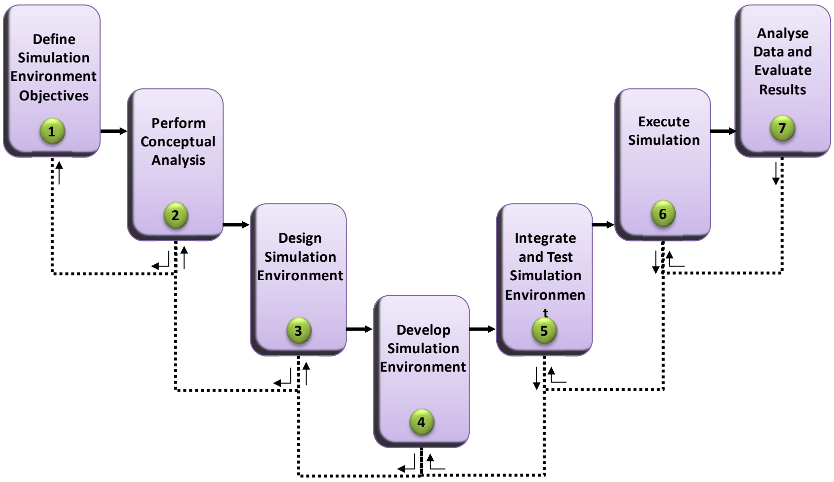
The evaluation of the MSG-086 issue groups leads to the following recommendations for related SISO working groups.

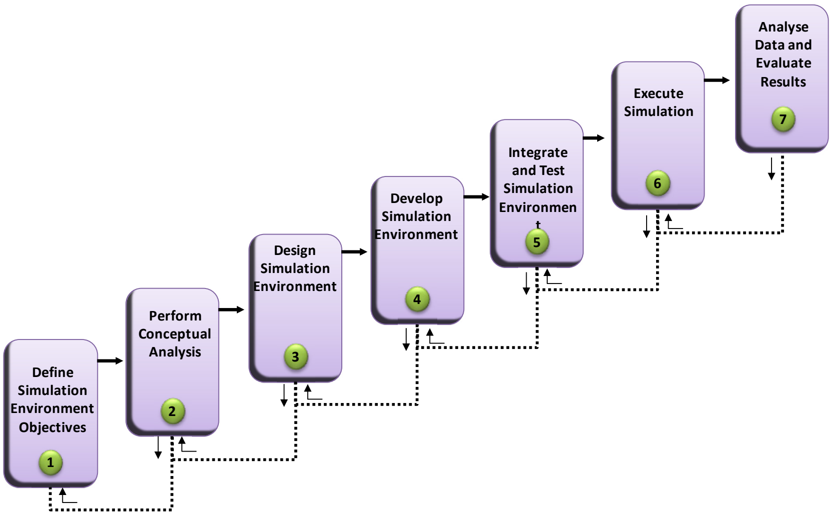
4.6.1 DSEEP

Because of the paramount importance of the early steps of a simulation engineering and execution process for achieving higher levels of interoperability the corresponding activities and related information products should be thoroughly thought through and elaborated in sufficient detail using unambiguous terms (e.g., the use of the term "scenario").

Table 4-1: Change Recommendations for DSEEP.

No.	Title	Description
1	New definition for term "Scenario"	As described in [12S-SIW-014], the definition of the term "scenario" currently used by DSEEP has several shortcomings. The proposal is to use the definition of MSG-053 instead. See [12S-SIW-014] for more details.
2	Add definition for term "vignette"	A definition of the term "vignette" is missing. The proposal is to add the following definition (DSEEP, Chapter 2.1, Page 2): "A vignette is a reusable temporally ordered set of events and behaviours for a specific set of entities."

No.	Title	Description
3	Editorial comment (table incomplete)	In Table 3-1 the activity 3.3 (“Design member applications”) is missing.
4	Adopt three-step scenario development process	Adopt the three-step scenario development process which is defined in the “Guideline on Scenario Development for (Distributed) Simulation Environments” [ScenarioGuideline] and also in [12S-SIW-014] and [12F-SIW-046]. Note: For simplicity and better traceability, this change request is split up into a set of smaller change requests.
4.1	Distinguish three types of scenarios	Clearly distinguish between “Operational scenarios”, “Conceptual scenarios” and “Executable scenarios” in the DSEEP. See [12S-SIW-014] and [12F-SIW-046] for more details. Update all terminology in the DSEEP appropriately.
4.2	Rename Step 2.1	Rename Step 2.1 from “Develop scenario” to “Develop conceptual scenario”.
5	Avoid impression that DSEEP is a waterfall process	Figure 2-1 of the DSEEP (page 4) can give first-time readers the wrong impression that the DSEEP is a waterfall process. Although back steps and iterations are shown in this figure, they may easily be missed. To avoid the impression that the DSEEP is a waterfall process, it is proposed to change the figure to more explicitly stress the iterative nature of the DSEEP. If possible, the new figure should also indicate that steps may be executed concurrently (at least, partially). The following figures may be used as starting points for further discussions: 

No.	Title	Description
5	(cont'd)	

4.6.2 FEAT

No.	Title	Description
1	Distinguish three types of scenarios	Clearly distinguish between "Operational scenarios", "Conceptual scenarios" and "Executable scenarios" in the FEAT. Introduce new elements if necessary and update existing terminology appropriately.

4.6.3 SCM

No.	Title	Description
1	Provide precise definitions of conceptual modeling terminology	Provide precise definitions of conceptual modeling terminology. Terms like "mission space conceptual model" and "simulation implementation space conceptual model" need to be defined precisely. Furthermore, these terms should be used consequently throughout the document (this is unfortunately not the case in the MSG-058 Final Report).

Chapter 5 – REFERENCES

Reference / Key	**Full Bibliographical Information**
ACT MS Vision	ACT M&S Vision, December 2008.
Baker1999	T.M. Baker, "Time Management in Distributed Simulation Models", Adacel Technologies Limited, (SimTecT99), 29 March – 1 April 1999.
07F-SIW-009	J.J. Boomgaardt, K.J. de Kraker and R.M. Smelik, (2007), "Achieving a Level Playing Field in Distributed Simulations", Fall Simulation Interoperability Workshop, 07F-SIW-009, Orlando, FL, USA.
11F-SIW-034	A. Bowers, C.L. Budde, B.C. Gregg and D. Winkowski, "JLVC Data Initialization", SISO 2011 Fall SIW, Orlando, FL, USA, 11F-SIW-034.
Cayirci2010	E. Cayirci, Tutorial on "Joint Exercise Management Module (JEMM)", 5th NATO CAX Forum 2010, Ottobrunn, Germany, August 2010.
Brownsword2004	L.L. Brownsword, D.J. Carney, D. Fisher, G. Lewis, C. Meyers, E.J. Morris, P.R.H. Place, J. Smith and L. Wrage, "Current Perspectives on Interoperability", Technical Report, CarnegieMellon Software Engineering Institute, CMU/SEI-2004-TR-009, ESC-TR-2004-009, March 2004.
DoD M&S Glossary	Department of Defense: Modeling and Simulation (M&S) Glossary, Published by Modeling and Simulation Coordination Office, October 1, 2011.
JP5-0	US Department of Defense, Joint Chiefs of Staff, "Joint Operation Planning", Joint Publication 5-0, 11 August 2011.
MSC-DMS	Department of Defense Modeling & Simulation Coordination Office: Modeling and Simulation (M&S) Community of Interest (COI) Discovery Metadata Specification (MSC-DMS), Version 1.4, 24 December 2010.
FMV2011	FMV, "M&S Architecture and Design Guidelines for Command Training Federations Based on the SMART-Net Concept", Version 2, 11FMV10110-16:1, 27 November 2011.
12F-SIW-014	A.D. Gillis and G.P. Franck, "The HLAgile Open Source Project: Design, Implementation, and Use", SISO 2012 Fall SIW, Orlando, FL, USA, Paper 12F-SIW-014.
13S-SIW-042	C. Hein, "Global Communications Network Effects Federate", SISO 2013 Spring SIW, San Diego, CA, USA, Paper 13S-SIW-042.
12S-SIW-032	W. Huiskamp, "Security in NATO Collective Mission Training – Problem Analysis and Solutions", SISO 2012 Spring SIW, Paper 12S-SIW-032.
1278.3	IEEE: IEEE Recommended Practice for Distributed Interactive Simulation – Exercise Management and Feedback, 1996 – IEEE Std 1278.3.
DSEEP	IEEE: IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP), 2011 – IEEE Std 1730-2010.
DMAO	Draft Recommended Practice for Distributed Simulation Engineering and Execution Process Multi-Architecture Overlay (DMAO), IEEE P1730.1/Dv0.2, July 2011.

Reference / Key	Full Bibliographical Information
FEDEP	IEEE: IEEE Recommended Practice for High Level Architecture (HLA) – Federation Development and Execution Process (FEDEP), 2003 – IEEE Std 1516.3, 2003.
11S-SIW-049	M. Karlsson, F. Antelius and B. Möller, "Time Representation and Interpretation in Simulation Interoperability – An Overview", Proceedings of 2011 Spring Simulation Interoperability Workshop, 11S-SIW-049, Simulation Interoperability Standards Organization, April 2011.
11F-SIW-012	R. Khayari, K.-H. Neumann, S. Devaud, J.-P. Faye, D. Desert, J. Ruiz and B. Löfstrand, "NATO Education and Training Network (NETN): Logistics and Transfer of Control FOM Modules Harmonization", 2011 SISO Fall SIW, Orlando, FL, USA, September 2011, 11F-SIW-012.
Krueckhans2012	M. Krückhans, "ISO and OGC Compliant Database Technology for the Development of Simulation Object Databases", Winter Simulation Conference 2012, Berlin, Germany.
10F-SIW-037	R. Lutz, R. Scrudder, R. Syring, M.D. Petty, J. Borah and J. Rutledge, "A Systems Engineering Perspective on the Development and Execution of Multi-Architecture LVC Environments", SISO 2010 Fall SIW, Orlando, FL, USA, Paper 10F-SIW-037.
11S-SIW-024	R. Lutz and D. Drake, "Gateway Concepts for Enhanced LVC Interoperability", SISO 2011 Spring SIW, Boston, MA, USA, Paper 11S-SIW-024.
12S-SIW-048	B. Mack, K. Roos, I. Olomski, P. Campbell, S. Smith and R. Rheinsmith, "A Multi-Faceted Approach to the Development of the HLA 1516-2010 German Maritime Federation Object Model (GMF)", SISO 2012 Spring SIW, Orlando, FL, USA, Paper 12S-SIW-048.
Mojtahed2005	V. Mojtahed, M. Garcia Lozano, P. Svan, B. Anderson and V. Kabilan, "DCMF-Defence Conceptual Modelling Framework", FOI-R –1754 –SE, ISSN 1650-1942, November 2005.
Mojtahed2008	V. Mojtahed, E. Tjörnhammer, J. Zdravkovic and A. Hanzala Khan, "The Knowledge Use in DCMF", FOI-R –2606 –SE, ISSN 1650-1942, November 2008.
Mojtahed2008b	V. Mojtahed, B. Andersson and V. Kabilan, "BOM and DCMF", FOI-R –2362–SE, ISSN 1650-1942, January 2008.
JC3IEDM	Multilateral Interoperability Programme (MIP), "Joint Consultation, Command and Control Information Exchange Data Model", (JC3IEDM), Version 3.1.4, 14 February 2012.
MORS Gap List	NMSG MORS M&S Gap Analysis Questionnaire: Consolidated National MS Gap Assessment, Version 2, 22 May 2011.
NATO_COBP	NATO: Code of Best Practice for C2 Assessment, October 2002.
NATO_COPD	NATO, "NATO Comprehensive Operations Planning Directive", 17 December 2010.
NATO_MSMP	NATO: NATO Modelling and Simulation Master Plan, Version 2.0, 14 September 2012, Document AC/323/NMSG(2012)-015.

Reference / Key	Full Bibliographical Information
NAF	NATO Consultation, Command and Control Board (NC3B): NATO Architecture Framework, Version 3, AC/322-D(2007)0048-AS1, 2007.
AAP-06	NATO Standardization Agency (NSA): NATO Glossary of Terms and Definitions, Edition 2013.
AMSP-01	NATO: NATO Modelling and Simulation Standards Profile, AMSP-01, Edition (B), Version 1, June 2011.
NMSG_SBP	NMSG Strategy & Business Plan, AC/323-NMSG(2011)SBP-012, May 2011.
NMSG-024	NATO: Final Report of MSG-024 "M&S Support to Non-Article 5 Operations", RTO-TR-MSG-024 AC/323(MSG-024)TP/27, January 2008.
NMSG-052	NATO: Final Report of MSG-052 "Knowledge Network for Federation Architecture and Design", RTO-TR-MSG-052 AC/323(MSG-052)TP/397, November 2011.
NMSG-053	NATO: Final Report of MSG-053 "Rapid Scenario Generation for Simulation Applications", RTO TR-MSG-053 AC/323(MSG-053)TP/302, May 2010.
NMSG-058	NATO: Final Report of MSG-058 "Conceptual Modeling (CM) for Military Modeling and Simulation", RTO-TR-MSG-058 AC/323(MSG-058)TP/404, July 2012.
NMSG-068	NATO: Final Report of MSG-068 "NATO Education and Training Network", RTO-TR-MSG-068 AC/323(MSG-068)TP/407, February 2012.
NMSG-071	NATO: Final Report of MSG-071 "Missionland", STO-TR-MSG-071, January 2014.
NMSG-080	NATO: Final Report of MSG-080 "Security in Collective Mission Simulation", Pre-released October 2014.
TAP_MSG086	NATO: Technical Activity Proposal (TAP) for MSG-086 "Simulation Interoperability", 2010.
ScenarioGuideline	NATO MSG-086. "Guideline on Scenario Development for (Distributed) Simulation Environment", January 2015.
Neugebauer2009	E. Neugebauer, D. Nitsch and O. Henne, "Architecture for a Distributed Integrated Test Bed", NATO RTO Modelling and Simulation Group Symposium (MSG-069), Brussels, Belgium, October 2009, RTO-MP-MSG-069 Paper 19.
NIAG-SG-162	NATO Industrial Advisory Group (NIAG) and NAFAG Aerospace Capability Group 6 (ACG 6): Final Report of NIAG SG 162 Study on "Distributed Simulation for Air and Joint Mission Training", 15 October 2012, NIAG-D(2012)0022 (PFP).
Relax NG	The Organization for the Advancement of Structured Information Standards (OASIS), "RELAX NG Specification", Committee Specification, http://relaxng.org/spec-20011203.html, 3 December 2001. (See also "ISO/IEC 19757-2:2003 Document Schema Definition Language (DSDL) – Part 2: Regular-grammar-based validation – RELAX NG "for an ISO version of RELAX NG Specification).

Reference / Key	Full Bibliographical Information
UML	Object Management Group (OMG), "Unified Modeling Language", Version 2.4.1, http://www.omg.org/spec/UML/2.4.1/, August 2011.
EU CTF	Saab Systems & Combitech: Core Technical Framework for Distributed Simulation, Part of EU Core Technical Framework Study, EDA reference number 08-R&T-004, July 2009.
11S-SIW-044	R. Siegfried, G. Herrmann, J. Lüthi and M. Hahn, "VEVA as German Approach Towards Operationalizing the DSEEP: Overview and First Experiences", SISO 2011 Spring SIW, Boston, MA, USA, Paper 11S-SIW-044.
11F-SIW-023	R. Siegfried, G. Herrmann, J. Lüthi and M. Hahn, "A Comparison of DSEEP with the German Approach VEVA", SISO 2011 Fall SIW, Orlando, FL, USA, Paper 11F-SIW-023.
12S-SIW-014	R. Siegfried, A. Laux, M. Rother, D.Steinkamp, G. Herrmann, J. Lüthi and M. Hahn, "Scenarios in Military (Distributed) Simulation Environments", SISO 2012 Spring SIW, Paper 12S-SIW-014.
12F-SIW-046	R. Siegfried, A. Laux, M. Rother, D. Steinkamp, G. Herrmann, J. Lüthi and M. Hahn, "Components and Reuse in Scenario Development Processes for Distributed Simulation Environments", SISO 2012 Fall SIW, Orlando, FL, USA, Paper 12F-SIW-046.
13S-SIW-019	R. Siegfried, H. Oguztüzün, U. Durak, A. Hatip, G. Herrmann, P. Gustavson and M. Hahn, "Specification and Documentation of Conceptual Scenarios Using Base Object Models (BOMs)", 2013 SISO Spring SIW, San Diego, CA, USA, Paper 13S-SIW-019.
Siegfried2010	R. Siegfried, A. Laux, G. Herrmann and J. Lüthi, "VEVA – A Procedure Model for Distributed Simulation Experiments", NATO MSG Symposium 2010, Utrecht, Netherlands, Paper 010.
Siegfried2011	R. Siegfried, J. Lüthi, G. Herrmann and M. Hahn, "How to Ensure Fair Fight in LVC Simulations: Architectural and Procedural Approaches", NATO MSG Symposium 2011, Bern, Switzerland, Paper 016.
BOM	SISO: Base Object Model (BOM) Template Specification, 31 March 2006 – SISO-STD-003-2006.
MSDL	SISO: Military Scenario Definition Language (MSDL), 14 October 2008 – SISO-STD-007-2008.
RIEDP_2012	SISO: Final Report of Reuse and Interoperation of Environmental Data and Processes (RIEDP) Study Group, SISO-REF-043-2012, 8 October 2012.
99S-SIW-167	SISO: Report from the Fidelity Implementation Study Group, Redacted by D.C. Gross, Spring SIW 1999, Paper 99S-SIW-167, Orlando, FL, USA, March 1999.
SUMO	Suggested Upper Merged Ontology SUMO, http://www.ontologyportal.org/.
CMMI	Software Engineering Institute: CMMI for Development, Version 1.3, Technical Report CMU/SEI-2010-TR-033, November 2010.

Reference / Key	Full Bibliographical Information
Stueber2012	R. Stüber and M. Krückhans, “Increased Sustainability of Simulation Object Databases Using International Norms and Standards”, NATO MSG Conference 2012, Stockholm, Sweden, Paper 15.
10S-SIW-027	J. Tegner and R. Suzic, “Core Technical Framework Stud: DSEEP Adaption and MODAF Mapping”, SISO 2010 Spring SIW, Orlando, FL, USA, 10S-SIW-027.
03F-SIW-007	A. Tolk and J.A. Muguira. “The Levels of Conceptual Interoperability Model”, SISO 2003 Fall SIW, Orlando, FL, USA, 03F-SIW-007, September 2003.
Tolk2006	A. Tolk, et al., “Ontology Driven Interoperability – M&S Applications”, Whitepaper for Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC) 2006.
09F-SIW-011	A. Tolk and C.D. Turnitsa, “An Extended Interoperability Framework for Joint Composability”, 2009 SISO Fall SIW, Orlando, FL, USA, Paper 09F-SIW-011.
US CCBP	US M&S Steering Committee, M&S Common and Cross-Cutting Business Plan, Version 1.2, November 2006.
OWL	W3C, “OWL 2 Web Ontology Language”, W3C Recommendation, http://www.w3.org/TR/owl2-overview/, 27 October 2009.
XMLSchema	W3C, “XML Schema”, W3C Recommendation, http://www.w3.org/XML/Schema, 28 October 2004.
Wang2009	W. Wang, A. Tolk and W. Wang, “The Levels of Conceptual Interoperability Model: Applying Systems Engineering Principles to M&S”, SCS Spring Simulation Multiconference, San Diego, CA, USA, 23-27 March 2009.
CDB	Presagis, “Common Database Specification”, Volume 1, Version 3.1, December 2011.
SRM	SEDRIS, “Spatial Reference Model”, ISO/IEC 18026:2006(E).
EDCS	SEDRIS, “Environmental Data Coding Specification”, ISO/IEC 18025:2005(E).
DRM	SEDRIS, “Functional Specification”, ISO/IEC 18023-1:2006(E).
STF	SEDRIS, “Abstract Transmittal Format”, ISO/IEC 18023-2:2006(E).
DFDD	DGIWG, “DGIWG Feature Data Dictionary”.

Annex A – DETAILED ISSUE DESCRIPTIONS

A.1 ISSUE GROUP “CONCEPTUAL MODEL” (CM)

A.1.1 CM-01 No Explicit Development of a Conceptual Model

A.1.1.1 Problem Definition

If a conceptual model is not developed explicitly for an intended simulation environment, the actual objectives of the user/sponsor may not be achieved.

A.1.1.2 Extended Problem Description

Due to various reasons (e.g., cost, schedule, knowledge) quite often a conceptual model is not explicitly developed for an intended simulation environment. When a conceptual model is lacking, each individual will form and use his own conceptual model during the development of the simulation environment.

As a conceptual model is a crucial aspect during simulation environment development, the lack of an explicit conceptual model leads to manifold problems. Examples of these are:

- The conceptual model is the traceability link between the objectives and the eventual design implementation, not having a conceptual model means this traceability is lost; and
- During conceptual modelling the static and dynamic relations between entities are identified, by not performing this analysis it is more likely that crucial relations are missed. To solve this later in the development will be more expensive or it can impact the usability of the simulation environment.

A related issue is the use of simulation components for which the conceptual model is not (fully) known. The same problems as identified above can occur in that case. An example of this is using a commercial serious game that often won’t come with a fully documented conceptual model.

A.1.1.3 Related Work

- MSG-058 “Conceptual Modeling for Military Modeling & Simulation”.
- SISO Product Development Group (PDG) on “Recommended Practices for Simulation Conceptual Modeling” (SCM):
    - Objective: “The SCM PDG will produce a stand-alone guidance document that will clarify “conceptual model” concepts, discuss conceptual modeling terminology, and enable different stakeholders to improve the formalization of conceptual models.” (Taken from SISO Product Nomination for SCM PDG)
    - Continue on work from MSG-058.
- The German process model VEVA defines documentation guidelines for the conceptual model (see [11S-SIW-044, 11F-SIW-023, Siegfried2010]).
- Swedish Defence Conceptual Modelling Framework (DCMF).
- Federation Engineering Agreements Template (FEAT):
    - Partially represented in FEAT schema by the fedagree: conceptual Model element. Most of the information is stored in external documents that are referenced. However fedagree:conceptual Model/archModel offers the possibility to link XML encoded documents, for example UML diagrams. But there is no predefined structure for the conceptual model.

A.1.1.4 Connection to LCIM Level

4-6 (pragmatic, dynamic and conceptual interoperability).

A.1.1.5 Connection to DSEEP Steps and Artifacts

This issue is related to Step 2, Activity 2.2 ("Develop Conceptual Model") of the DSEEP. However during Step 1 the operational scenario is defined and this is an important input for the conceptual model, so Activity 1.2 ("Develop objectives") is also related.

A.1.1.6 Possible Solution Approaches

The obvious solution approach for this issue is to explicitly develop a conceptual model. This solution approach has two implications:

1) A conceptual model must be developed (which always happens at least in an implicit way; and
2) The conceptual model must be made "explicit", i.e., "communicable" and documented.

Possible guidelines for conceptual modeling are listed below:

- The work of MSG-058 describes a process to perform conceptual modelling;
- The DSEEP provides basic guidelines on how to perform conceptual modelling; and
- The German process model VEVA defines documentation guidelines for conceptual models [11S-SIW-044, 11F-SIW-023].

The other issues in the issue group CM concern more specific problems with performing the conceptual modelling step. Solution approaches for these other issues will also contribute to making it easier to perform the conceptual modelling step at all and therefore are a solution approach for this issue.

A.1.1.7 Existing Implementations and Their Information Products

- The DSEEP provides some information on the information products that should be created during conceptual modelling activity.

A.1.1.8 Recommended Information Products

- An improvement of the current situation would be if there are recommendations and guidelines for conceptual modeling (possibly including templates for information products to be developed). First steps into this direction have been made by MSG-058 (see Section 2.4) and are expected to be made by SISO SCM PDG (see Section 2.5).

A.1.1.9 Examples/Prototypes of Selected Information Products

- The German process model VEVA defines documentation guidelines for the conceptual model [11S-SIW-044, 11F-SIW-023].

A.1.2 CM-02 Missing Standards for Development of Conceptual Models for Distributed Simulation Environments

A.1.2.1 Problem Definition

Lack of standard practices for conceptual model development or consensus definition of conceptual model content inhibits simulation interoperability because assumptions and constraints critical to the required level of effectiveness are unknown, undocumented, or ignored.

A.1.2.2 Extended Problem Description

Missing standards for development of conceptual models for distributed simulation environments lead to (at least) two problems:

1) Every modeler/developer has to develop his own standards/templates/guidelines. This takes a lot of time and requires a good knowledge of conceptual modelling in general. Therefore, if resources such as time, budget, knowledge or experience are limited the development of a conceptual model may not happen. Consequences hereof are described in CM-01.
2) Missing standards complicate comparability of conceptual models and familiarization of modelers with the standards to be used.

A.1.2.3 Related Work

MSG-058 asserts that “Without Conceptual Models, history has shown that simulation developers often do not sufficiently understand the military domain to be modelled and implement M&S that do not reflect the intended reality, and thus do not satisfy the user’s needs.” It follows, then that two or more simulation systems without adequate conceptual models and consequently inadequately reflecting the intended reality, will not “effectively work together” and will not “satisfy the user’s needs”.

IEEE Std 1730-2010, IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP), identifies “a sequence of seven very basic steps that all distributed simulation applications will need to follow to develop and execute their simulation environments.” Step two of this seven step process is performing conceptual analysis with Activity 2.2 being the development of a conceptual model. Obviously “performing conceptual analysis” is deemed an important part of the process since it is included as one of seven steps, but the document does not describe the problem(s) attending NOT performing one of these seven basic steps.

In October 2010, SISO approved a Product Development Group (PDG) activity whose purpose is development of a stand-alone SISO product entitled “Recommended Practices for Simulation Conceptual Modeling.” The PDG intends to provide “detailed guidance on how to perform conceptual modelling for technical teams that actively develop or modify distributed simulation environments.” [SISO Product Nomination SISO Product Development Group (PDG) on Recommended Practices for Simulation Conceptual Modeling 12 April 2010]

In Sept 2011, the Simulation Conceptual Modeling (SCM) PDG agreed to base the Recommended Practice on the MSG-058 Final Report.

A.1.2.4 Connection to LCIM Level

4-6 (pragmatic, dynamic and conceptual interoperability).

LCIM Level 6: Finally, if the conceptual model – i.e., the assumptions and constraints of the meaningful abstraction of reality – are aligned, the highest level of interoperability is reached: Conceptual Interoperability. This requires that conceptual models are documented based on engineering methods enabling their interpretation and evaluation by other engineers. In essence, this requires a “fully specified, but **implementation independent model**” as requested by Davis and Anderson; this is not simply text describing the conceptual idea.”

The LCIM Level 6 requires alignment of conceptual models in order to reach the supreme state of simulation (application?) interoperability. It follows that two or more applications that have unaligned conceptual models are not “fully” interoperable as defined by LCIM (though they may be sufficiently interoperable for the purpose for which they are being used).

A.1.2.5 Connection to DSEEP Steps and Artifacts

This issue is related to Step 2, Activity 2.2 (“Develop Conceptual Model”) of the DSEEP. The activity outcome, i.e., the artifact, for Activity 2.2 is the conceptual model.

A.1.2.6 Possible Solution Approaches

A published recommended practices guide would address this issue. The SISO SCM PDG is developing the SISO “Recommended Practices for Simulation Conceptual Modeling” (see Section 2.5).

A.1.2.7 Existing Implementations and Their Information Products

- Final Report of MSG-058.
- VEVA documentation guidelines (see MSG-086 Background and Reference Materials).

A.1.2.8 Recommended Information Products

Guideline for developing and documenting conceptual models for distributed simulation environments.

A.1.2.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.1.3 CM-03 Missing Reference Conceptual Models

A.1.3.1 Problem Definition

Currently no widely known and accepted reference conceptual models exist. It is assumed that the lack of such reference conceptual models may lead to interoperability problems during design and development of distributed simulation environments.

A.1.3.2 Extended Problem Description

According to the DSEEP pre-existing reference conceptual models may be incorporated in the design process of a concrete simulation environment as an input to the conceptual analysis step.

The DSEEP [DSEEP] does not give any detail on how a pre-existing reference conceptual model may be integrated in or will influence in any way the conceptual analysis activity. Indeed, the process step is somewhat “self-contained” and may be fully executed without any optional input of a reference conceptual model, and thus will not give rise to any interoperability problems. It is therefore concluded that the absence of reference conceptual models will not influence simulation interoperability, as long as the conceptual analysis step is done properly.

As mentioned before, interoperability problems arising from conceptual analysis can be fully attributed to a poor quality (e.g., incompleteness, inconsistencies) of a conceptual model. Therefore, the question regarding the relation of reference conceptual models and simulation interoperability may be reduced to the question, whether (and how) existing reference conceptual models, which are fed as an input to the DSEEP conceptual analysis step carried out for a specific simulation environment, can improve the quality of the resulting information products (i.e., the conceptual model) in the sense of completeness and consistency, thus preventing simulation interoperability problems in later stages.

To gain further insight, it is assumed that a reference conceptual model is explicitly fed as input into the process. Then it may influence the process in the following ways:

- The DSEEP contains a built-in iteration between the “develop (conceptual) scenario” and the “develop conceptual model” activities. This iteration may converge to any one of several possible consistent solutions. The presence of a reference conceptual model may force the iteration process to favor one solution over the others, thus providing some kind of “stability” to the process.
- The input conceptual model may serve as a direct base for modelling. If it is available in machine readable form it may be directly loaded into some modelling tool. A good example is the SUMO ontology and derived ontologies from [Mojtahed2005].

Even if a reference conceptual model is not explicitly fed into the process but is just recognized by the individuals involved in the conceptual analysis step, there may be positive effects:

- The reference conceptual model may be used as a source of “best practices” on how certain aspects may be modeled or have been modeled successfully in the past.
- It may cure at least to some extent the problem discussed in CM-01: If no “official” conceptual model is defined, then the developers will have to explicitly or implicitly define their own individual conceptual models. A commonly known and accepted reference conceptual model, although not adapted to the specific simulation environment objectives, may nevertheless provide some guidance and lead to some alignment of the individual conceptual models. In addition, it may also improve the completeness of individual conceptual models by being a source of (otherwise missing) information. Although a reference conceptual model may be helpful in this way, the situation of CM-01 is highly undesirable and should be avoided in any case.

The analysis shows, that the absence of reference conceptual models does not per se lead to simulation interoperability problems. However, the previous analysis demonstrates that the design and engineering of a simulation environment may benefit from the existence of reference conceptual models, thus reducing the overall effort in time and costs necessary for design and engineering a simulation environment.

We note here, that operational scenarios as output of DSEEP Step 1 (Define simulation environment objectives) also play an important role for design of a conceptual model (and thereby the conceptual scenarios). Operational scenarios in contrast to reference conceptual models will serve as a true input to conceptual modelling. As a consequence, the conceptual modelling activity of the DSEEP will fail, if proper operational scenarios are not provided.

A.1.3.3 Related Work

- [Mojtahed2005] describes the Defence Conceptual Modelling Framework (DCMF), which provides a rich environment for conceptual modelling.
- [SUMO] is the Suggested Upper Merged Ontology. This ontology comprises a meta-level schema of classes (“concepts” in the ontology wording) and relationships (“properties”) as well as a mid-level ontology. Many rich domain ontologies exist, that are built on the basis formed by SUMO.
- [NMSG-058] was a STO Level 3 Task Group engaged with Conceptual Modeling for Military Modeling and Simulation.
- FEAT: Partially represented in FEAT schema by the “fedagree:conceptualModel”-element. Unfortunately, the “fedagree:conceptualModel”-element provides just a reference to some external document.

A.1.3.4 Connection to LCIM Level

According to [Mojtahed2005] conceptual modelling includes agreement on commonly used terms and their definition and meaning, e.g., by defining taxonomies and ontologies as part of the conceptual model. Therefore Level 3 (semantic) and above are affected with a focus on the conceptual level.

A.1.3.5 Connection to DSEEP Steps and Artifacts

As shown above, the following DSEEP steps are affected:

- Step 3.1, Select Member Applications;
- Step 3.2, Design Simulation Environment;
- Step 3.3, Design Member Applications;
- Step 4.1, Develop Simulation Data Exchange Model; and
- Step 4.2, Establish Simulation Environment Agreements.

The following DSEEP artifacts are affected:

- Scenarios;
- Conceptual model; and
- Simulation environment agreements.

A.1.3.6 Possible Solution Approaches

It has been demonstrated that the DSEEP conceptual analysis step can be carried out properly without pre-existing reference conceptual models. If carried out properly, no interoperability problems will result from this activity.

On the other hand it has been shown, that the quality of the conceptual model information products may be significantly improved and the modelling effort reduced by incorporating proper reference conceptual models into the process. Therefore, a database of reference conceptual models might be helpful, see [Mojtahed2005].

A.1.3.7 Existing Implementations and Their Information Products

See [Mojtahed2005].

A.1.3.8 Recommended Information Products

Although several reference conceptual models have been published, no true standard or at least “community standard” is available.

Regarding ontologies the SUMO [SUMO] ontology may be recommended as it is a candidate for the IEEE “standard upper ontology” to be standardized in the future.

A.1.3.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.1.4 CM-04 Lack of Standard Methodologies, Techniques and Tools for Automated Transition from Conceptual Models to Simulation Data Exchange Models (SDEMs) or (Automated) Comparison and Verification Between Conceptual Models and SDEMs

A.1.4.1 Problem Definition

Methodologies, techniques and tools are missing for:

- Automated transition from conceptual models to Simulation Data Exchange Models (SDEMs, e.g., a FOM in HLA); and
- Automated comparison and verification between conceptual models and simulation data exchange models.

Such tools help to ensure the correctness of the SDEMs.

A.1.4.2 Extended Problem Description

No information provided by MSG-086.

A.1.4.3 Related Work

- G. Özhan, "Generating Simulation Code from Federation Models: A Field Artillery Case Study", In Proceedings of the European Simulation Interoperability Workshop, (11E-SIW-007), June 27 – 29, 2011.
- G. Özhan, A. Dinç and H. Oğuztüzün "Model-Integrated Development of Field Artillery Federation Object Model", Simul, pp. 109-114, 2010 Second International Conference on Advances in System Simulation, 2010.

A.1.4.4 Connection to LCIM Level

No information provided by MSG-086.

A.1.4.5 Connection to DSEEP Steps and Artifacts

No information provided by MSG-086.

A.1.4.6 Possible Solution Approaches

No information provided by MSG-086.

A.1.4.7 Existing Implementations and Their Information Products

No information provided by MSG-086.

A.1.4.8 Recommended Information Products

No information provided by MSG-086.

A.1.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.1.5 CM-05 Missing Standards for Application Services

A.1.5.1 Problem Definition

The absence of standards for integrating application services into simulation environments leads to interoperability problems which are caused by incompatible or proprietary interfaces as well as by differently behaving services.

A.1.5.2 Extended Problem Description

Using application services (e.g., a weapon effect service) is a means to achieve identical effects and behavior (e.g., with respect to weapon effect) across multiple simulation systems. Currently no standards exist for describing application services and for integrating application services into a simulation environment which leads to several interoperability problems:

- Missing standardized interface specifications for application services lead to a missing exchangeability and interoperability of application services. Without standardized interface specifications it is not possible to use different application services for a specific purpose (e.g., a classified weapon effect service and a non-classified weapon effect service).
- Regarding HLA-based simulation environments, application services may be integrated into the simulation environment in different ways (e.g., as a special type of federate or as a web service). Not all possible ways of integrating application services into the simulation environment are conforming with the HLA-standard (e.g., if a communication effect service realizes communication or data exchange without involving the RTI).

A.1.5.3 Related Work

MSG-068 and MSG-106

In MSG-068 a pattern for use of services was developed and implemented as a Federation Object Model (FOM) module. Successful testing was done where federates provided and consumed services among them. The following services were defined:

Service	FOM Module	Tested
Supply	X	X
Storage	X	
Repair	X	X
Transport	X	X
Combat Adjudication	X	

See [11F-SIW-012] for more details. This work is continued by MSG-106.

German VIntEL Development

Within the German R&D effort “SD VIntEL” the use of application services in HLA-based simulation environments is analyzed. See Section 2.6 and [Neugebauer2009] for more details.

A.1.5.4 Connection to LCIM Level

This issue is mainly related to Levels 2 and 3:

- Level 2 (syntax) and Level 3 (semantics) are primarily affected by this issue, as interface specifications for application services address these two levels.

The Levels 4, 5, and 6 are also affected by application services, because the selection of an application service which will be integrated into a simulation environment is as crucial as the selection of a simulation system which will be integrated. However, this issue does not consider the case of choosing an inappropriate application service but of missing interface specifications.

A.1.5.5 Connection to DSEEP Steps and Artifacts

- Steps 2.2 and 2.3 (“Develop conceptual model” and “Define Simulation Environment Requirements”).
- Steps 3.1 and 3.2 (“Select Member Applications” and “Design Simulation Environment”).
- Steps 4.1 and 4.2 (“Develop Simulation Data Exchange Model” and “Establish Simulation Environment Agreements”).

A.1.5.6 Possible Solution Approaches

The most natural solution approach is to define and to standardize interface specifications (syntax, semantic, pragmatic) for application services. Ideally, these interface specifications are developed firstly for existing (prototypical) application services (e.g., the Weapon Effect Service used in VIntEL) and usage experiences are gathered. Subsequently, more and more interface specifications for application services could be developed.

A.1.5.7 Existing Implementations and Their Information Products

DEU (“SD VIntEL” project):

- Weapon Effects Service (provided by IABG);
- Communication Effects Service (provided by KESS of Thales); and
- Exterior Ballistic Service (provided by IABG).

NLD:

- A prototype weapon effect server has been developed by TNO [07F-SIW-009].

Other:

- Global Communications Network Effects Federate [13S-SIW-042].

A.1.5.8 Recommended Information Products

- Interface specification:
  - Describes syntax, semantics, and pragmatic of an application service interface.
- Conceptual model:
  - Describes how the application service works, which assumptions are underlying, etc.

A.1.5.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.2 ISSUE GROUP "FEDERATION DEVELOPMENT" (FD)

A.2.1 FD-01 Multi-Architecture Simulation Environments

A.2.1.1 Problem Definition

Using more than one simulation architecture (e.g., DIS, HLA, TENA) within a single simulation environment often leads to interoperability problems. This is due to different underlying assumptions (e.g., real-time simulation vs. time-managed simulation), different protocols and supported fidelities, and many other issues.

A.2.1.2 Extended Problem Description

Usually, the implementation of a specific simulation environment is based on just one simulation architecture (e.g., HLA, DIS, or TENA) that is able to support the interoperation of all member applications. But sometimes it is necessary to mix several simulation architectures, because the required member applications do not support a common simulation architecture.

The *Introduction* to the DSEEP standard mentions "mixed-architecture simulation environments", which are defined as follows: "circumstances in which more than one architecture (e.g., HLA, DIS, TENA) is required to meet all of the technical, cost, and schedule constraints associated with the application". The problem is described further in the DSEEP in Section *4.3.2 Activity 3.2 – Design simulation environment:*

> *"In some large simulation environments, it is sometimes necessary to mix several simulation architectures. This poses special challenges to the simulation environment design, as sophisticated mechanisms are sometimes needed to reconcile disparities in the architecture interfaces. For instance, gateways or bridges to adjudicate between different on-the-wire protocols are generally a required element in the overall design, as well as mechanisms to address differences in SDEMs."*

The problems arising from a multi-architecture simulation environment comprise technical as well as management issues, e.g., planning, or requirements evaluation. In addition, different architectures usually rely on different engineering practices, documentation templates, etc., thus leading to communication barriers between the users of different simulation architectures. Another description of this problem is given in [11S-SIW-024] and [10F-SIW-037]:

> *"In some situations, sponsor requirements may necessitate the selection of simulations whose external interfaces are aligned with more than one simulation architecture. This is what is known as a multi-architecture simulation environment. There are many examples of such environments within the DoD, especially in areas that require broad participation across disparate user communities (e.g., joint training and experimentation). When more than one simulation architecture must be used in the same environment, interoperability problems are compounded by the architectural differences. For instance, middleware incompatibilities, dissimilar metamodels for data exchange, and differences in the nature of the services that are provided by the architectures must all be reconciled for such environments to operate properly. This not only raises additional technical risk, but the additional resource consumption necessary to adjudicate these architectural differences affects cost and schedule risk, as well."*

A.2.1.3 Related Work

In 2010, the SISO DMAO PDG *(DSEEP Multi-Architecture Overlay Product Development Group)* was established, which takes care about the design and development of multi-architecture simulation environments, see http://www.sisostds.org:

> *"Many special issues must be addressed when building a distributed simulation environment that involves multiple simulation architectures (e.g., HLA, DIS, TENA). Issues like time management, interest management, and object model reconciliation are all more difficult to resolve when multiple simulation architectures are in play. While the DSEEP provides an architecture-neutral description of the process required to build distributed simulation environments, it does not address the unique issues/solutions associated with the development and execution of multi-architecture simulation environments, leaving developers with little or no sources of practical guidance.*
>
> *The DMAO extends the process described in the DSEEP to address multi-architecture development and execution. It is designed as an overlay, associating issues and solutions relevant to multi-architecture development to existing DSEEP activities. While a baseline overlay currently exists, broader participation in this effort is requested to improve the quality and completeness of this important product."*

The DMAO PDG has developed the DSEEP Multi-Architecture Overlay [DMAO]. This document contains detailed descriptions of 37 multi-architecture-specific issues, which can be seen as sub-issues to *FD-01 Multi-Architecture Simulation Environments.*

A.2.1.4 Connection to LCIM Level

The connection of HLA, DIS and TENA concerns mainly the LCIM Levels 1 and 2 (technical and syntactic), but Level 3 (semantic) might also be affected (e.g., mapping of DIS PDUs to HLA classes, while preserving their proper meaning).

Depending on the nature and origin of the member applications, even the Levels 4 to 6 could be affected, because, e.g., the conceptual models determine the semantic models.

A.2.1.5 Connection to DSEEP Steps and Artifacts

The DMAO describes a set of simulation environment development issues that are either specific to, or exacerbated by, the need to develop a multi-architecture simulation environment. These issues are aligned with relevant activities in the DSEEP. The DMAO augments the descriptions of those DSEEP activities with additional inputs, recommended tasks, and outcomes.

In the following, just the titles of the multi-architecture-specific issues are cited, grouped under the relevant activities of DSEEP, and the affected artifacts of DSEEP respectively DMAO are indicated.

Activity 1.3: Conduct Initial Planning

Multi-architecture-specific issues:

1) Multi-Architecture Initial Planning;
2) Required Multi-Architecture Simulation Environment Expertise;
3) Inconsistent Development and Execution Processes; and
4) VV&A for Multi-Architecture Applications.

Affected artifacts:

- Simulation Environment Development and Execution Plan.

Activity 2.3: Develop Simulation Environment Requirements

Multi-architecture-specific issues:

1) Requirements for Multi-Architecture Simulation Environment; and
2) Member Application Requirement Incompatibility.

Affected artifacts:

- Simulation Environment Requirements; and
- Simulation Environment Test Criteria.

Activity 3.1: Select Member Applications

Multi-architecture-specific issues:

1) Member Application Selection Criteria for Multi-Architecture Simulation Environments; and
2) Non-conforming Member Applications.

Affected artifacts:

- List of Selected Member Applications;
- List of Requirement Gaps; and
- List of architectures support by the selected member applications.

Activity 3.2: Design Simulation Environment

Multi-architecture-specific issues:

1) Gateway Usage and Selection Decisions;
2) Object State Update Contents;
3) Object Ownership Management;
4) Time Management in Multi-Architecture Simulation Environments;
5) Interest Management Capability Differences;
6) Gateway Translation Paths;
7) DIS Heartbeat Translation;
8) Multi-Architecture and Inter-Architecture Performance;
9) Translating Non-Ground-Truth Network Data;
10) Object Identifier Uniqueness and Compatibility;
11) Cross Domain Solutions in Multi-Architecture Simulation Environments; and
12) Multi-Architecture Save and Restore.

Affected artifacts:

- Simulation Environment Design;
- Implied requirements for member applications modifications;
- Selected common communications middleware;
- List of selected gateways;
- Requirements for gateway configuration;

- Common terminology across architecture communities; and
- List of selected tool(s) for monitoring and controlling simulation execution.

Activity 3.3: Design Member Applications

Multi-architecture-specific issue:

1) New Member Application Architecture.

Affected artifact:

- Member Application Designs.

Activity 3.4: Prepare Detailed Plan

Multi-architecture-specific issues:

1) Cost and Schedule Estimating for Multi-Architecture Development.

Affected artifact:

- Simulation Environment Development and Execution Plan.

Activity 4.1: Develop Simulation Data Exchange Model

Multi-architecture-specific issues:

1) Metamodel Incompatibilities; and
2) SDEM Content Incompatibilities.

Affected artifacts:

- SDEM for each architecture;
- SDEM mappings;
- Updated requirements for gateway modifications; and
- Updated requirements for gateway configuration.

Activity 4.2: Establish Simulation Environment Agreements

Multi-architecture-specific issues:

1) Agreements to Address Multi-Architecture Development;
2) Tool Availability and Compatibility; and
3) Initialization Sequencing and Synchronization.

Affected artifact:

- Simulation environment agreements.

Activity 4.3: Implement Member Application Designs

Multi-architecture-specific issue:

1) Non-standard Algorithms.

Affected artifacts:

- None.

Activity 4.4: Implement Simulation Environment Infrastructure

Multi-architecture-specific issue:

1) Network Configuration.

Affected artifacts:

- None.

Activity 5.1: Plan Execution

Multi-architecture-specific issues:

1) Integration and Test Planning for Multi-Architecture Simulation Environments; and
2) Multi-Architecture Execution Planning Considerations.

Affected artifacts:

- Execution Environment Description.

Activity 5.2: Integrate Simulation Environment

Multi-architecture-specific issue:

1) Live Entity Time, Space, and Position Information Updates.

Affected artifacts:

- None.

Activity 5.3: Test Simulation Environment

Multi-architecture-specific issue:

1) Complexities of Testing in a Multi-Architecture Simulation Environment.

Affected artifacts:

- None.

Activity 6.1: Execute Simulation

Multi-architecture-specific issue:

1) Monitoring and Controlling Multi-Architecture Simulation Environment Execution; and
2) Multi-Architecture Data Collection.

Affected artifacts:

- None.

Activity 7.2: Evaluate and Feedback Results

Multi-architecture-specific issue:

1) Multi-Architecture Simulation Environment Assessment.

Affected artifact:

- Fault analysis for problems found to be related to or caused by multi-architecture factors.

A.2.1.6 Possible Solution Approaches

1) Closely follow the work of the SISO DMAO PDG/PSG, in general, and for the specified 37 multi-architecture issues. For all 37 issues, the DMAO describes recommended actions in detail.
2) For example, use of gateways for simulation environments using DIS and HLA.

A.2.1.7 Existing Implementations and Their Information Products

There are only partial solutions, e.g., DIS-HLA-gateways (available as COTS or MOTS products), and their user manuals.

The two M&S infrastructure vendors Pitch Technologies and VT MÄK have both COTS products to deal with this issue.

Pitch Technologies DIS Adapter enables transition from DIS to HLA. It has a graphical interface for configuration and monitoring and supports the commonly used PDUs. For systems that use older standards, such as HLA1.3 or HLA 1516-2000 that are hard to migrate Pitch provides a “1516 Adapter for HLA1.3 Federates” and you are able to connect without changing one line of code.

VR-Link from VT MÄK is a networking toolkit with a single documented API that abstracts away the details of simulation networking protocols. When you write your code to the VR-Link API, your applications become natively compliant with DIS, HLA 1.3, HLA 1516, and HLA Evolved.

VT MÄK VR-Exchange is a universal translator for distributed simulation. With VR-Exchange you can link together simulations that use different HLA RTIs, different object models, and even different protocols.

An example of a GOTS product is the German PSI-SA-Gateway (**P**rimary **S**imulation **I**nterface for **S**imulation **A**pplications) that was developed in 2009 for the German Army project SuTBw (see Section 2.6). It provides a coupling device between different DIS or HLA federations by mapping different FOM versions onto each other and by routing all the object attributes into the other federation. By doing so the PSI-SA gateway appears as a federate and allows every state and update within each federation to be logged. The PSI-SA Gateway let all simulation objects appear in every federation that is connected to the gateway. It is also possible to build up cascades of multiple gateways to connect several different federations (which would otherwise impossible to connect with each other, e.g., because of different RTIs or because of the difference of standards like HLA vs. DIS). See Figure A-1 for an example.

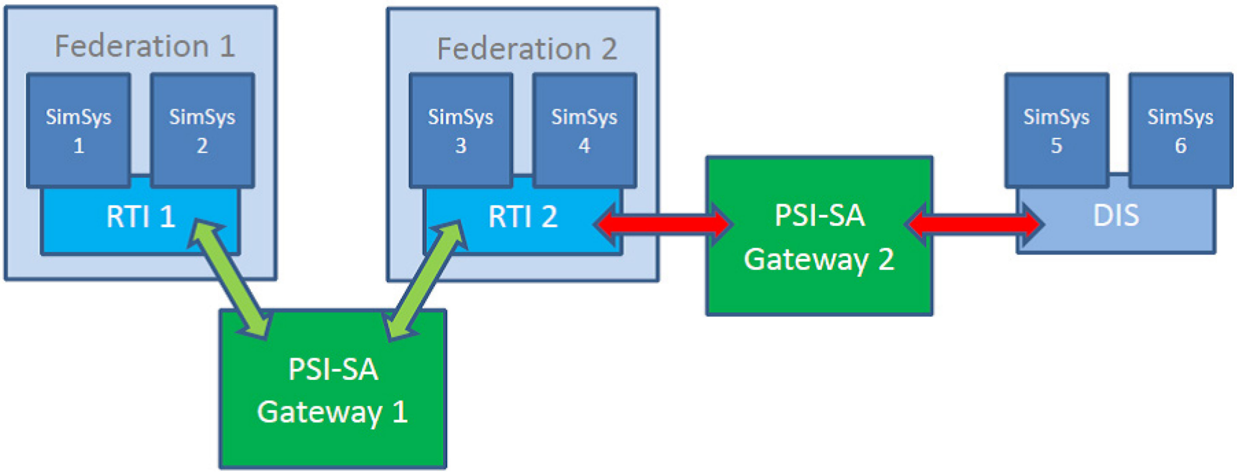


Figure A-1: Two PSI-SA Gateways Connecting Different Federations: Gateway 1 Connects Two Different HLA Federations, Gateway 2 Connects a HLA Federation with a DIS Simulation. In this case the simulation objects in the DIS simulation would also appear in Federation 1 and *vice versa*.

TENA Gateway Builder is a tool from the TENA consortium that automatically generates gateways to bridge simulation and live protocols. The TENA Gateway Builder supports mappings between TENA, DIS and HLA and message-based protocols using any object model.

A.2.1.8 Recommended Information Products

See lists with *Affected artifacts* of DSEEP in the Section *Connection to DSEEP steps or artifacts*, above.

A.2.1.9 Examples/Prototypes of Selected Information Products

In the descriptions of the recommended actions for the multi-architecture issues, the DMAO gives numerous hints to projects, including bibliographic references, where these issues have been treated.

A.2.2 FD-02 Different HLA Versions

A.2.2.1 Problem Definition

Due to the evolution of the standard, several versions of the HLA standard exist. For each version implementations from different vendors or open source implementations are available. The lack of standardization on the protocol level and differences in the capabilities of the individual versions prevent systems building upon different HLA versions or implementations from interoperating.

A.2.2.2 Extended Problem Description

The problem originates from the following facts:

- HLA has undergone a continuous evolution during which more and more features and capabilities found their way into the standard. Applications that require the use of newer features will be incompatible with older applications not capable of using new features. Currently four different version are in use:

  1) HLA 1.3 (from 1998);
  2) IEEE Std 1516-2000 (plus DMSO interpretations version 2, supported, e.g., by Pitch);

3) IEEE Std 1516-2000 (SISO-STD-004.1-2004, aka SISO DLC API version, supported, e.g., by MÄK); and
4) IEEE Std 1516-2010 (a.k.a. *HLA Evolved*).

- Each federation requires exact one RTI executable to be present and running. The RTI executable however is bound to a distinct version of the standard and moreover to a particular implementation or vendor. The decision for a particular RTI therefore determines the standard and implementer/ vendor of the HLA infrastructure to use by all federates.
- The standardization comprises the application interface only while the network protocol remained unstandardized. Thus implementations from different implementers or vendors are in general incompatible, even if they adhere to the same version of the standard.

A.2.2.3 Related Work

The problem of adapting federates to different HLA versions is sometimes subsumed under the keyword *Federation Agility [*12F-SIW-014*]. Federation Agility* also was one of the topics treated by MSG-052 *(Knowledge Network for Federation Architecture and Design)*.

FEAT: The FEAT addresses this issue with the `fedagree:middlewareAgreements` and `fedagree:middleware` elements. The latter however only refers to external documents not otherwise specified.

A.2.2.4 Connection to LCIM Level

This issue affects the interfaces of federates to HLA RTIs. So, the LCIM Levels 1 and 2 (technical and syntactic) are concerned.

A.2.2.5 Connection to DSEEP Steps and Artifacts

This issue is connected to the following steps and artifacts of DSEEP.

Activity 1.3: Conduct Initial Planning

- Simulation Environment Development and Execution Plan.

Activity 3.2: Design Simulation Environment

- Simulation Environment Design;
- Selected common communications middleware;
- List of selected gateways; and
- Requirements for gateway configuration.

Activity 3.4: Prepare Detailed Plan

- Simulation Environment Development and Execution Plan.

Activity 4.2: Establish Simulation Environment Agreements

- Simulation environment agreements.

A.2.2.6 Possible Solution Approaches

Essentially, there are two possibilities to overcome the problem of different HLA versions:

1) Adapt the interface for member applications that do not have a native API that corresponds to the API stated in the Federation Agreement Document; and
2) Split the federation into several sub-federations in such a way, that each sub-federation obeys to one HLA version, and the sub-federations are connected to each other by gateways. Sometimes, these gateways are also called *bridges* (see MSG-052 Final Report, p. xix: *The term bridge is sometimes preferred to avoid the confusion with the term gateway at network level*).

In addition limited interoperability between the different HLA versions can be provided by the commercially available RTIs, for example:

1) "Federates built with the MÄK RTI's HLA 1.3 libraries can interoperate with federates built with MÄK RTI's HLA 1516 libraries if the federation is not using ownership management, DDM, or MOM services." (from: *MÄK Interoperability Guide*); and
2) "Pitch pRTI™ provides the HLA Evolved API and all of the HLA Evolved services. In addition to this it provides the HLA 1516-2000 and HLA 1.3 APIs through adapters." (from: www.pitch.se).

This however will work only within the software product palette provided by each single vendor.

A.2.2.7 Existing Implementations and Their Information Products

The following implementations of the above mentioned solution approaches are known:

1) Commercial adapters (e.g., from MÄK or Pitch); and
2) Commercial gateways (e.g., from MÄK or Pitch) and custom-made gateways (e.g., PSI-SA gateway described in FD-01).

The distinction made here between adapters and gateways is that an adapter is attached to a system whereas gateways are usually considered to be stand-alone systems.

A.2.2.8 Recommended Information Products

As the problem is not caused by missing or inconsistent information, it cannot generally be solved using agreement documents or similar information products. For practical purposes it is however required to agree upon the HLA version and vendor implementation to use for a specific simulation environment.

A.2.2.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.2.3 FD-03 Different FOM or SDEM Versions

A.2.3.1 Problem Definition

If systems that are required to exchange data do not comply with the same data exchange model, interoperability is severely limited.

A.2.3.2 Extended Problem Description

Simulation architectures like HLA or DIS define a data exchange model for data exchange between all member applications. In case of HLA, this data exchange model is known as Federation Object Model

(FOM). An HLA federation is always based on exactly one FOM (or, in the case of HLA Evolved, a specific set of FOM modules). Federates which shall be used in this federation, have to obey to the FOM of the federation, including the encoding of data types. An interoperability problem shows up when a federate was developed for another federation, using another FOM, and cannot switch easily to the new FOM. This effect is also called missing *FOM agility* or missing *FOM independence*.

The problem can be generalized for all types of simulation architectures (e.g., DIS, HLA, or TENA): All member applications of a simulation environment have to obey to exactly one Simulation Data Exchange Model (SDEM). If the interfaces of already existing member applications have been developed for other SDEMs, they can usually not interoperate easily with a new SDEM.

A.2.3.3 Related Work

DSEEP

The DSEEP gives additional information concerning SDEM in Section 4.4.1 (p. 24/25):

> *"Depending on the nature of the application, the SDEM may take several forms. Some simulation applications are strictly object-oriented, where both static and dynamic views of the simulation system are defined in terms of class structures, class attributes, and class operations (i.e., methods) within the SDEM. Other simulation applications maintain this same object-based paradigm, but use object representations as a way to share state information among different member applications about entities and events that are being modeled internal to the member applications themselves. In this case, the SDEM is quite similar to a data model, including a defined set of format, syntax, and encoding rules. Still other simulation applications may not use an object-based structure at all in their SDEM. Rather, the focus is on the runtime data structures themselves and the conditions that cause the information to be exchanged. In general, different applications will have different requirements for the depth and nature of interaction among member applications, and while these varying requirements will drive the type and content of the SDEM, it is necessary for the SDEM to exist in order to formalize the "contract" among member applications necessary to achieve coordinated and coherent interoperation within the simulation environment.*
>
> *There are many possible approaches to SDEM development. Certainly, reuse of an existing SDEM is the most expeditious approach, if one can be found within a repository that meets all member application interaction requirements. If an exact match for the needs of the current application cannot be found, then identifying an SDEM that meets some of these needs and modifying/tailoring the SDEM to meet the full set of requirements would generally be preferable to starting from a "clean sheet." Some communities maintain reference SDEMs for their users to facilitate this type of reuse (e.g., the real-time platform reference federation object model (RPR FOM); see SISO-STD-0001.1 [B2]). Still other approaches involve assembling an SDEM from small, reusable SDEM components (e.g., Base Object Models (BOM), see SISO-STD-003 [B3]) and merging SDEM elements from the interfaces of member applications [e.g., HLA Simulation Object Models (SOM)]. In general, the simulation environment development/integration team should employ whatever approach makes most sense from a cost, efficiency, and quality perspective."*

Reference FOMs

Explanation for Reference FOMs from AMSP-01, Section B.4.6:

> *"While the HLA Standards dictate how federates exchange data, it is a Federation Object Model (FOM) that dictates what data is being exchanged in a particular federation. HLA does not mandate the use of any particular FOM, however, several "reference FOMs" have been developed to promote a-priori interoperability. That is, in order to communicate, a set of federates must agree on a common FOM (among other things), and reference FOMs provide ready-made FOMs that are*

> *supported by a wide variety of tools and federates. Reference FOMs can be used as-is, or can be extended to add new simulation concepts that are specific to a particular federation or simulation domain."* (Followed by a short description of the RPR FOM, as an established example for a reference FOM.)

FOM Modules

Definition in IEEE Std 1516-2010 (HLA – Framework and Rules):

> *"**Federation object model (FOM) module:** A partial FOM (containing some or all Object Model Template (OMT) tables) that specifies a modular component of a FOM. A FOM module may contain classes not inherent to it but upon which the FOM module depends, i.e., superclasses to the modular components. These superclasses will be included in the FOM module as either complete or scaffolding definitions."*

BOMs

Explanation for BOMs from AMSP-01(A), Section B.4.1:

> *"Base Object Models (BOMs) provide a component framework for facilitating interoperability, reuse, and composability. The BOM concept is based on the assumption that piece-parts of models, simulations, and federations can be extracted and reused as modelling building-blocks or components. The interplay within a simulation or federation can be captured and characterized in the form of reusable patterns. These patterns of interplay are sequences of events between simulation elements. The representation of the pattern of interplay is captured in the first BOM document [Reference SISO-STD-003-2006]. The second document, the "Guide for Base Object Model (BOM) Use and Implementation", introduces methodologies for creating BOMs and implementing them in the context of a larger simulation environment. The document is a means of familiarizing the reader with the concept of BOMs and providing guidance for BOM development, integration, and use in supporting simulation development. [Reference SISO-STD-003.1-2006]"*

BOMs may be used for documenting SDEMs in a standardized and architecture-neutral way, and to foster conceptual model compatibility.

A.2.3.4 Connection to LCIM Level

This issue is concerned with SDEMs (or FOMs for HLA). Since SDEMs (FOMs) define the semantics of data interchange, LCIM Level 3 (semantic interoperability) is affected by this issue.

A.2.3.5 Connection to DSEEP Steps and Artifacts

This issue is mainly connected to DSEEP Step 4 *(Develop Simulation Environment), Activity 4.1 – Develop simulation data exchange model*, and to the resulting artifact SDEM.

A.2.3.6 Possible Solution Approaches

There are several possibilities to overcome the problem of different FOM or SDEM versions:

1) Try to avoid the problem from the very beginning, by using either a specific reference FOM or FOM modules in the development of all member applications. If possible, use a standardized reference FOM, or FOM modules.
2) Adapt the interfaces of one or several member applications, so that all member applications obey to the same FOM (or SDEM).

3) Split the federation (or simulation environment) into several sub-federations in such a way, that each sub-federation obeys to one FOM (or SDEM), and the sub-federations are connected to each other by gateways.
4) Middleware to support *FOM agility* of federates (e.g., from MÄK, or PSI-SA in Germany) that either produce a conformal FOM or work as an adaptor within the middleware.

A.2.3.7 Existing Implementations and Their Information Products

- Widely used reference FOMs are RPR FOM 1.0 and 2.0 (*Real-time Platform Reference Federation Object Model*, see SISO-STD-0001), accompanied by a GRIM *(Guidance, Rationale, and Interoperability Manual).*
- Commercial gateways (e.g., from MÄK or Pitch).
- Custom-made gateways.

A.2.3.8 Recommended Information Products

Depending on the solution approach:

1) For reference FOMs or FOM modules, a kind of GRIM or Users' Guide is mandatory;
2) A FAD *(Federation Agreements Document)* is mandatory; and
3) For gateways, a Users' Manual is mandatory.

A.2.3.9 Examples/Prototypes of Selected Information Products

Example for a GRIM:

- *Guidance, Rationale, and Interoperability Manual for the Real-time Platform Reference Federation Object Model (RPR FOM)*, Version 2.0D17v3, 3 October 2003.

A.2.3.10 References

There exist tools to check the simulation environment for compliance of FOMs. Two examples from Germany are:

- FACTS (Federation Agreement Conformance Test Service) is an experimental tool for distributed HLA-based simulation environments with the capability to validate attribute and parameter values while recording and replaying data of a federation execution. It was developed within the German army project SD VIntEL (see Section 2.6). FACTS is model independent, the application can load any precompiled model at run-time. With FACTS the user can watch and log the exchange of information within the federation and is able to discover any violation of the federation agreement or federation object model FOM; and
- FIERS (Federation Integration and Experimentation Rehearsal Surrogate) serves during the integration and experimentation phase as a surrogate for other simulation systems which are not available that early or cannot be easily distributed because of license issues. However, FIERS will not re-implement the business logic of other simulation systems. It was merely developed to demonstrate and test the information exchange patterns of a federation execution. Therefore, FIERS will create and discover objects, update and reflect attribute values, and send and receive interactions as specified in the Federation Agreements Document (FAD). However, it is not expected to behave in a military meaningful way. FIERS also serves as test harness that may be used by each company with their federates prior to integration tests between different companies. Early testing of federates by each company against the same test tool shall reduce the time required to integrate the federation.

A.2.4 FD-04 Incompatible FOM Modules

A.2.4.1 Problem Definition

A sub-set of the rules for the merging of FOM modules in IEEE 1516-2010 (HLA Evolved) are unnecessarily strict and conflicts can result in federates not able to join a federation execution.

IEEE 1516-2010 allows the FOM to be divided into a number of modules. These FOM modules are loaded during the creation of the federation execution and when federates are joining the federation. When loading FOM modules the content in these modules is compared with the content from already loaded FOM modules. The comparison is done by the RTI which follows rules specified in “FOM module / SOM module merging rules” in IEEE Std 1516.2-2010.

A.2.4.2 Extended Problem Description

An example on a too strict FOM merging rule is that the transport type at attributes and interaction classes has to be equal in all loaded FOM modules (e.g., in the HLA 1516-2010 case, Best Effort or Reliable, additional transportation types can be used if agreed upon). A member application (federate) can as one of the first thing change this locally and send the message at its own preference. It had been better to let a federate specify what transportation type it wants to use when sending attributes and interaction classes when it loads its FOM modules.

Another example is that, assigning Dimensions to attributes and interaction classes in FOM modules to be used with Data Distribution Management (DDM) will prevent systems to execute in the same simulation environment (federation execution) if not that each attribute and interaction class has an equal set of dimensions specified in all loaded FOM modules. An application not developed for DDM will never be aware of any kind of dimensions assigned to the attributes/interaction classes even if other applications uses DDM in the same simulation environment. When DDM is used, only the intersection of the published and subscribed dimension sets is used for the filtering.

If tools for FOM merging are missing and a member application (federate) developed for IEEE 1516-2000 or HLA 1.3 shall execute in a simulation environment through an adapter, this system may only be able to load a monolithic FOM combined from a number of FOM modules even if not all of the modules are used. Improvement and corrections in of the legacy system unused FOM modules can prevent the legacy system or newer systems dependent of the improvements to be in the same simulation environment.

A.2.4.3 Related Work

The recent version of PSISA 2.2 contains a description and method for merging FOM modules to monolithic FOMs.

A.2.4.4 Connection to LCIM Level

This issue is concerned with how interaction classes and attributes at object classes are specified in FOM modules for IEEE 1516-2010. These specifications are syntactic definitions. LCIM Level 2 (syntactic interoperability) is affected here.

A.2.4.5 Connection to DSEEP Steps and Artifacts

This issue is connected to:

- Step 3.1 (Design Simulation Environment, Select member applications) where the improper selection of member applications and the design of the simulation environment can result in conflicts when merging FOM modules;

- Step 3.3 (Design Simulation Environment, Design Member Applications) where some member applications may have to be modified to use a create and join setup of FOM modules that not results in a conflict;
- Step 4.1 (Develop Simulation Environment, Develop Simulation Data Exchange Model); and
- Step 4.2 (Develop Simulation Environment, Establish Simulation Environment Agreements).

A.2.4.6 Possible Solution Approaches

- Modify “legacy” federates to only load FOM modules that contains the object and interaction classes that they actually use.
- Modify the set of unnecessarily strict rules for merging of FOM modules to be less strict in the next version of the IEEE 1516 standard.
- In the next version of the IEEE 1516 standard, remove the current requirement that Dimensions must be specified at class attribute and interaction classes in the FOM to be used with HLA Data Distribution Management services.
- Access to FOM development tools.

A.2.4.7 Existing Implementations and Their Information Products

- NETN FOM modules (from MSG-068 and MSG-106).
- RPRv2 (when finished).
- GMF (German Maritime FOM) [12S-SIW-048].

A.2.4.8 Recommended Information Products

No recommendations made by MSG-086.

A.2.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.2.5 FD-05 Incomplete Specification of Federation Agreements

A.2.5.1 Problem Definition

Incomplete or incompatible specification of Federation Agreements can lead to a large variety of interoperability problems.

Two examples (Object identifier uniqueness, Dead Reckoning Algorithms) are given in the following extended problem description.

A.2.5.2 Extended Problem Description

Object Identifier Uniqueness

Unique identifiers for entities in a simulation environment are desirable. Identifiers are used when updating instance attributes and referring instances in messages and events in the simulation environment. Operators of member applications (viewers) are also dependent on unique identifiers to identify entities in the scenario.

This implies that it may be necessary with different types of identifiers, one technical identifier adapted for software use and one operational identifier adapted to be human readable. The technical identifier will affect the simulation interoperability while the operational identifier will not affect the simulation interoperability, except when it is also used as a technical identifier.

In HLA Federation Executions are every instance assigned a unique identifier, the instance name according to the HLA standard, this identifier can be set by the federate during the registration or by the RTI. If the federate shall set the identifier value, it must be reserved first and not used by any existing instance, in this case the registration will fail. Most common is that the federate lets the RTI generate a unique identifier. The problem with this is that if the federate for some reason leaves the federation execution and comes back and register instances from the same ORBAT again, the identifier for the new instances will get a new value even if it shall modelling an entity that was active in the federation execution before the federate left the federation execution.

Also, in the RPR2 FOM has physical and aggregated entities two attributes that is used as identifiers, the ‘EntityIdentifier’ and ‘Marking’ (or ‘AggregateMarking’), unfortunately are not any of these guaranteed to be unique. In some messages (interactions) is the value for the ‘EntityIdentifier’ used and in some messages is the instance name used. The value at the attribute ‘EntityIdentifier’ of may also change for an entity when a federate leaves and comes back in the federation execution. The value of the ‘Marking’ attribute is often the entity name in the ORBAT, this value is limited to eleven characters for platform entities and to 31 characters for aggregate entities. Eleven characters for an entity name are often too small to get a good name. The Federation Agreements shall state which character code that shall be used. When ASCII is used, a common problem is the use of spaces in entity names. When building scenarios for different applications, human input of entity names often result in errors, especially when spaces are included in entity names.

Dead Reckoning Algorithms

To reduce the rate at which dynamic data between federates is updated, Dead Reckoning is used. Each federate has to maintain a Dead Reckoning Model for remote entities. This model is used to calculate the position of remote entities between updates. If two simulation systems use different Dead Reckoning models for the same remote entity, different positions are likely calculated between updates. The differences in positions may lead to different decisions on the actions of the two simulation systems and may result in unfair fight.

A.2.5.3 Related Work

No related work identified by MSG-086.

A.2.5.4 Connection to LCIM Level

This issue is concerned with SDEMs (or FOMs for HLA). Since SDEMs/FOMs define the semantics of data interchange, LCIM Level 3 (semantic interoperability) is affected here.

A.2.5.5 Connection to DSEEP Steps and Artifacts

This issue is connected to:

- Step 4.1 (*Develop Simulation Environment, Develop Simulation Data Exchange Model*) Define identifiers and how they shall be used if a new SDEM/FOM is developed; and
- Step 4.2 (*Develop Simulation Environment, Establish Simulation Environment Agreements*) Specify usage of identifier(s) when using existing SDEM/FOM.

A.2.5.6 Possible Solution Approaches

Generally the solution approach is to provide a complete specification of required federation agreements. Ensuring that all federation agreements are satisfied by the federates is an organizational task and may be supported by appropriate tools or verification methods (see FACTS or ET-35).

Object Identifier Uniqueness

Use a standard for identifier, e.g., the UUID defined in ISO/IEC 11578:1996 (and ISO/IEC 9834-8:2005). The UUID consists of an array with 16 bytes and has a standardized printable format that consists of 32 ASCII characters (0-9, a-f) and four hyphens on predetermined places, e.g., 550e8400-e29b-41d4-a716-446655440000. This identifier definition is used in MSDL. However UUIDs are somewhat cryptic and the relation between an UUID and the object it refers to might not be obvious (easily decodeable) in all cases, depending on the development environment, so debugging the simulation environment might be a difficult task.

Use only one type of identifier when refer to entities in all type of messages in the FOM.

The ORBAT should contain a unique identifier for every entity, and if not, the Federation Agreements Document has to specify how identifier values shall be generated for entities.

Define an identifier that has a user friendly format for entity names with no restriction on the number of characters.

Another way to ensure object identifier uniqueness is the implementation of a (non-technical) key management procedure, for example like the key management rules used in the Multi-lateral Interoperability Program (MIP).

Dead Reckoning Algorithms

Confirm that all federates use the same Dead Reckoning algorithms.

A.2.5.7 Existing Implementations and Their Information Products

In MSG-068 and MSG-106 is a FOM developed that has an attribute ‘Callsign’, primarily used for viewers and an identifier ‘UniqueID’ according to the UUID defined in ISO/IEC 11578:1996 to be used in messages to refer to instances.

A.2.5.8 Recommended Information Products

Federation Agreements Document (FAD)

The Federation Engineering Agreements Template (FEAT) may be used for a formal specification of federation agreements. Due to the novelty of the FEAT standard a detailed evaluation is missing.

A.2.5.9 Examples/Prototypes of Selected Information Products

A general approach to resolve or mitigate the problem of incomplete or incompatible specification of Federation Agreements could be the binding utilization of a (quasi)- standardized Federation Agreements and Reference Document for a certain federation execution as developed by the NATO Education and Training Network Federation Architecture and FOM Design Technical Sub-group MSG-068 (NETN Federation Agreements and FOM Reference Document v1.0). An update of this document will be provided by the presently ongoing NATO MSG-106.

A.2.6 FD-06 Missing Comprehensive Reference Architectures

A.2.6.1 Problem Definition

Established simulation architectures (DIS, HLA, or TENA) provide architecture frameworks just for the data exchange between simulation systems. But full interoperability requires more and includes also the alignment of data models, the use of services, the integration of operational systems, etc. Missing are comprehensive reference architectures (in the sense of NAF) which treat all these aspects.

A.2.6.2 Extended Problem Description

Table A-1: Hierarchy of Architectures as Defined by NAF.

NAF Terminology	Description	Examples
Overarching architecture	Architectures which are used for long-term planning. These architectures may include visionary concepts and ideas. Overarching architectures are often defined as goals or constraints which have to be applied when designing and implementing new systems.	"Our next-generation simulation framework shall be service-oriented." "Our long-term goal is to use only open (i.e., freely available) standards and protocols."
Reference architecture	Architectures which are used as a reference for a large set of applications.	Generic reference architectures: DIS, HLA, TENA Comprehensive reference architectures: VIntEL (= HLA + VIntEL-FOM + services) NETN (= HLA + NETN-FOM)
Target architecture	Architecture of a specific simulation environment.	MSG-068 demonstration at I/ITSEC

Building target architectures for specific systems on foundations from established reference architectures will increase not only the efficiency of work in time and budget, but also the quality of the results, and will lead to improved interoperability.

This general reasoning, following, e.g., the NAF schema, applies pretty well also to the case of creating *Simulation Environments*. But as a matter of fact, there are hardly any *comprehensive* reference architectures around, which embrace all aspects of creating *Simulation Environments*.

A.2.6.3 Related Work

- NETN (MSG-068, MSG-106).
- SD VIntEL (see Section 2.6).

A.2.6.4 Connection to LCIM Level

A simulation architecture affects mainly the design and the implementation of a simulation environment. Therefore, this issue concerns mainly the LCIM Levels 1 to 3 (*Technical, Syntactic*, and *Semantic*).

A.2.6.5 Connection to DSEEP Steps and Artifacts

A simulation architecture affects mainly the design and the implementation of a simulation environment. Therefore, this issue concerns mainly the DSEEP Steps 3 and 4 (*Design Simulation Environment*, and *Develop Simulation Environment*).

A.2.6.6 Possible Solution Approaches

Develop *Comprehensive reference architectures* for different application domains (like MSG-068/106 does for the application domain "training").

A.2.6.7 Existing Implementations and Their Information Products

An example of a comprehensive reference architecture is developed in the German SD VIntEL project. Figure A-2 shows the basic elements of the VIntEL reference architecture.

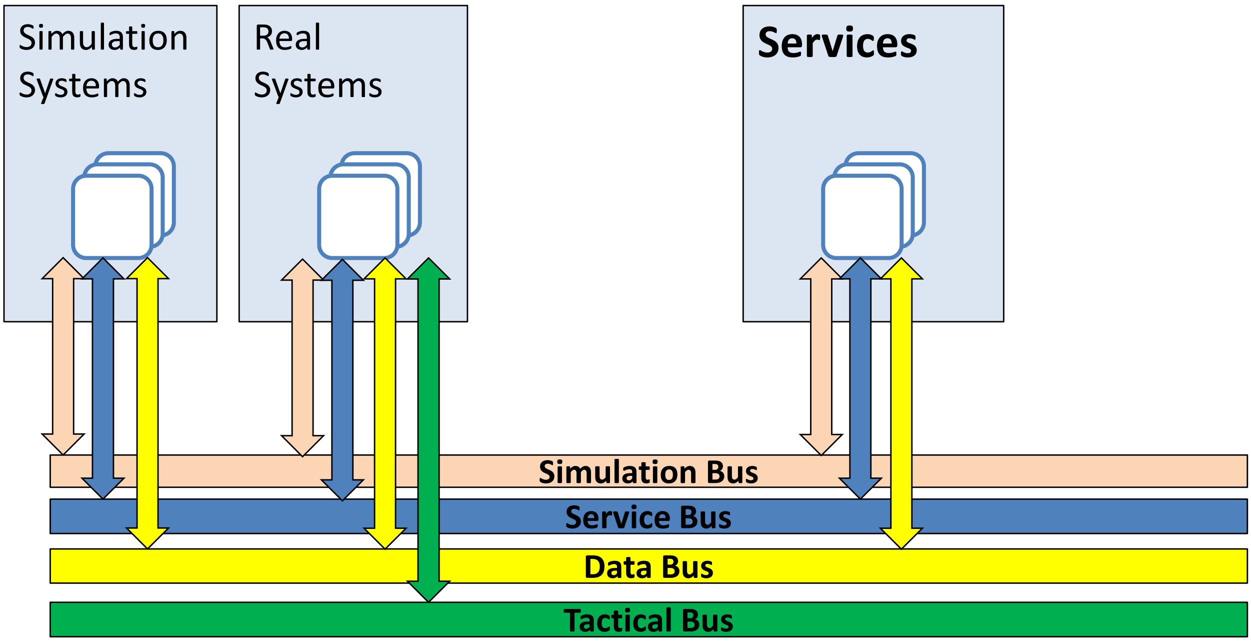


Figure A-2: VIntEL Reference Architecture.

From this *VIntEL reference architecture* several *Target architectures* for a series of evaluations were derived. An example for such a target architecture is shown in Figure A-3.

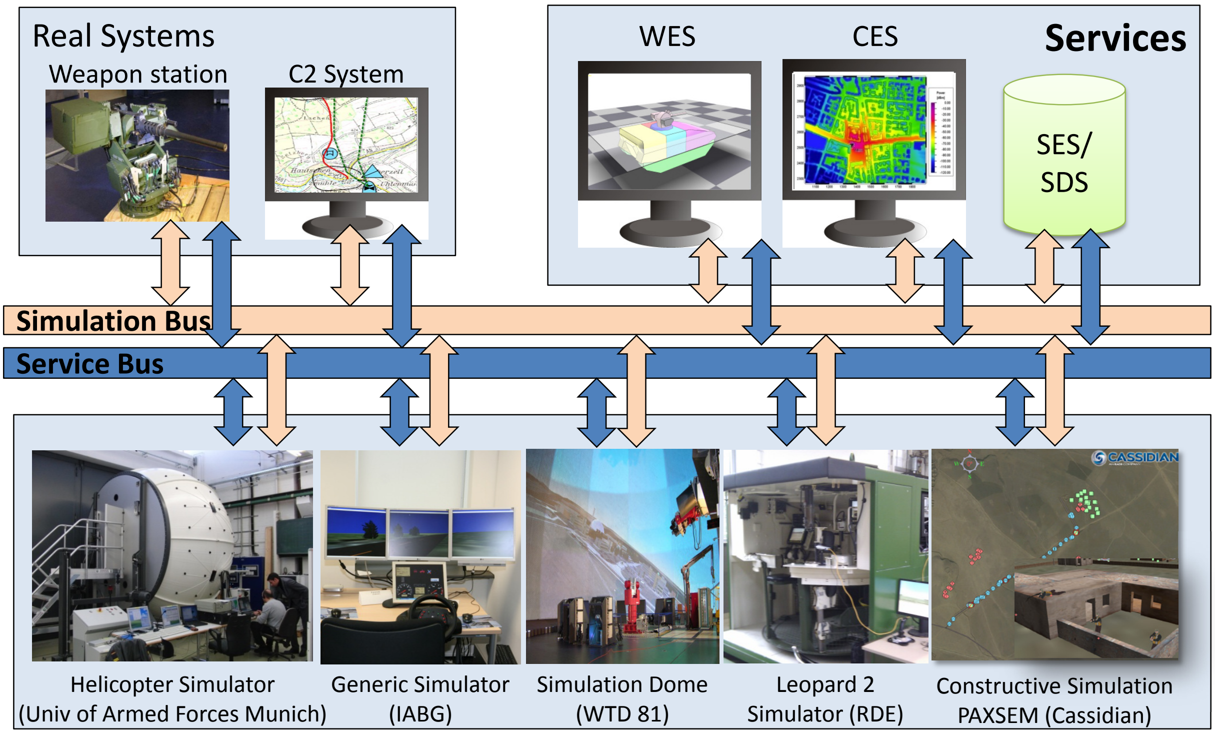


Figure A-3: VIntEL Target Architecture (Example).

A.2.6.8 Recommended Information Products

Description of *Comprehensive reference architectures,* e.g., according to NAF.

A.2.6.9 Examples/Prototypes of Selected Information Products

The guideline document of the *Reference Architecture* for the German SD VIntEL projects.

A.2.7 FD-07 Missing Standards and Templates for Artifacts

A.2.7.1 Problem Definition

If good engineering practices are applied, the development of a specific member application or the simulation environment in its entirety leads to the generation of artifacts, e.g., design documents. It is assumed, that missing standards or templates for artifacts lead to simulation interoperability problems.

A.2.7.2 Extended Problem Description

It is assumed that the engineering process of the simulation environment follows the DSEEP [DSEEP]. This process defines numerous information products to be created along the engineering from the objectives to the fully integrated and tested simulation environment. In addition, the analysis, design and implementation of individual components may be accompanied by the creation of additional artifacts, e.g., UML diagrams to describe software components.

Regarding the information products to be produced, the DSEEP is somewhat self-contained. It denotes the applicable artifacts for the process steps and roughly describes their content. If all required artifacts or information products are developed properly, obviously no interoperability problems will occur, even if the artifacts don't follow certain standards or templates.

However, interoperability problems may arise, if the created artifacts are incomplete or inconsistent. In case of incomplete artifacts, some maybe important properties of the simulation environment or its individual elements are not agreed upon, or at least an agreement if not documented. In this case the developers may tend to rely on assumptions on these properties resulting in incompatible components. In case of inconsistent artifacts basically the same symptom may be observed, the problem here being an incorrect design rather than incompatible assumption on unspecified features.

In any case standards and templates for the creation and contents of artifacts will help to create complete and consistent artifacts, thus preventing the described interoperability problems.

A.2.7.3 Related Work

DSEEP

Although the DSEEP defines contents for some of the artifacts to be generated during the execution of the process, three problems occur:

- It is not complete: there are artifacts defined in DSEEP without suggested contents, e.g., *Conceptual Model*;
- It is not sufficient: only a few high-level headings are given per artifact (if any); and
- It does not provide documentation standards or templates for the recommended information products.

VEVA

VEVA is a process model of the German MoD for the development of simulation environments (VEVA: **V**orgehensmodell für den **E**insatz der **V**IntEL-**A**rchitektur – Procedure Model for the Application of the VIntEL-Architecture, VIntEL: **V**erteilte **Int**egrierte **E**rprobungs-**L**andschaft – Distributed Integrated Testbed). See Figure A-4.

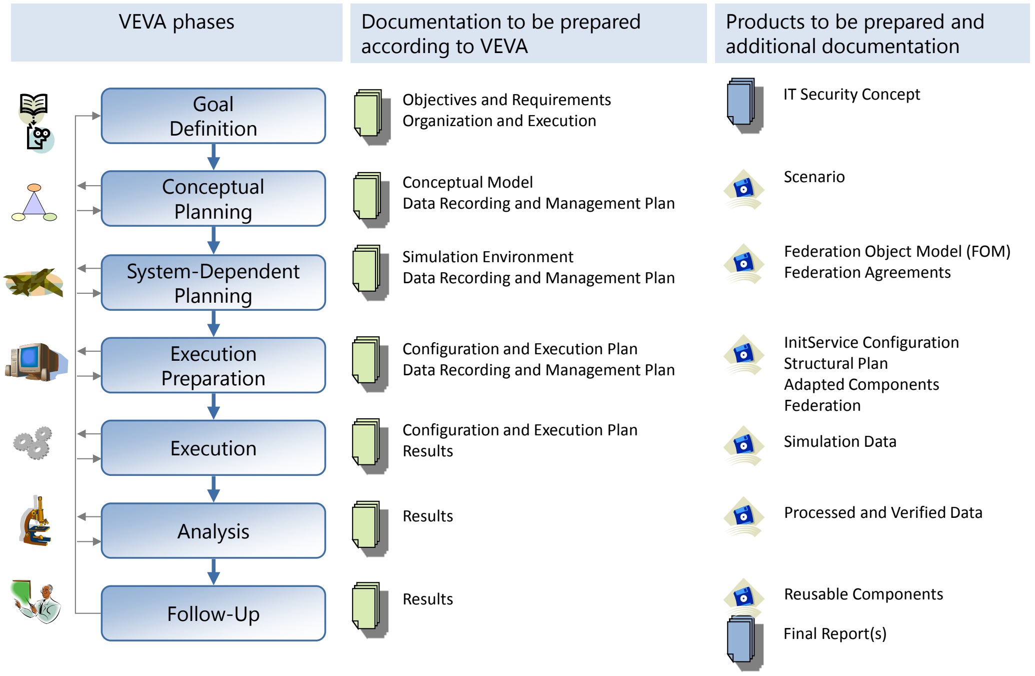


Figure A-4: Overview of VEVA Process Model.

VEVA borrows from the DSEEP, but is partly more detailed:

- VEVA provides highly detailed documentation templates; and
- VEVA defines 12 roles along with their responsibilities and duties within the development process.

For more details about VEVA, see paper 11S-SIW-044 of the SISO spring 2011 conference and paper 11F-SIW-023 of the SISO Fall 2011 conference.

NAF

Whereas the DSEEP is a *process model*, the NATO Architecture Framework NAF is an *architecture framework*, which defines numerous templates to consistently describe system architectures. DSEEP and NAF may be related as follows:

- DSEEP can be seen as a higher-level framework that defines a process, including the flow of information products and the content of information products.
- DSEEP does not define the format of information products or the dependencies of information products on the content level. These have to be provided by low-level management and engineering practices.
- On the other hand, NAF provides a well-known engineering practice (ranging from high level to low level).
- NAF may be used to augment DSEEP and to further detail the information products of DSEEP. For example, the templates of NAF may be used as input to DSEEP steps or as structured output of DSEEP steps.

As an example, the 7 operational views of NAF are compared with the 3 activities of Step 1 of the DSEEP (I = NAF views may provide input; O = NAF views may be used as result or documentation).

Table A-2: Mapping NAF Operational Views to DSEEP.

NAF Operational View		**DSEEP Step 1: Define simulation environment objectives**		
		Activity 1.1: Identify User/ Sponsor Needs	Activity 1.2: Develop Objectives	Activity 1.3: Conduct Initial Planning
NOV-1	High-Level Operational Concept	IO	IO	IO
NOV-2	Operational Node Connectivity Description	IO	IO	I
NOV-3	Operational Information Requirements	I	O	I
NOV-4	Organisational Relationship Chart	I		O
NOV-5	Operational Activity Model	I	O	I
NOV-6	Oper. Activity Sequence & Timing Description	I		
NOV-7	Information Model	I		O

Therefore NAF and DSEEP are somehow complementary:

- DSEEP provides a development process, but a poor description of structure and content of information products, and of relations between different information products.
- NAF provides templates for structuring related information, but doesn't provide a development process for architectures.
- Taken together, DSEEP provides a process for using NAF in the context of SE development:

- NAF can provide templates for DSEEP information products; and
- NAF templates constitute a common format for documenting architectures, and thus leverage the re-use of architecture elements.

NATO

One of the topics treated by MSG-052 *(Knowledge Network for Federation Architecture and Design)* are *Federation Agreements*. See Annex C: *Federation Agreements Activity* of the Final Report (available as a draft).

SISO

In 2010, the FEAT PDG *(Federation Engineering Agreements Template Product Development Group)* was established, which takes care about the development and use of HLA federation agreements, see http://www.sisostds.org:

> *"The Federation Engineering Agreements Template (FEAT) will benefit all developers, managers, and users of distributed simulations by providing an unambiguous format for recording agreements about the design and use of the distributed simulation. The template will also benefit this community by enabling the development of federation engineering tools that can read the schema and perform federation engineering tasks automatically.*
>
> *Although the FEDEP explicitly calls for federation agreements and gives some guidance about the contents of such agreements, it provides no guidance on the format or structure of these agreements. Currently federation agreements are recorded in multiple formats with ad hoc structures and content. As a result, federation agreements are often incomplete and ill-structured, leading to errors and rework resulting from misunderstanding of the agreements. The community needs a detailed, unambiguous template for recording federation agreements. Optimally, this template should be in a standardized format that can be used readily by automated federation engineering tools as well as read by federation participants."*

We can safely assume that this HLA-centric development maps perfectly to DSEEP (replace *federation agreement* by *simulation environment agreements).*

UML

UML is a framework of diagrams and related symbology for the design of software systems. Some elements of UML however may also be used outside the software domain, e.g., time sequence diagrams may generally be used to describe sequences and dependencies among events of activities. It is likely, that the strict notion of UML may be useful for the creation of at least some types of DSEEP information products as well.

A.2.7.4 Connection to LCIM Level

This issue concerns DSEEP Steps 1 to 5 (see next section). As a consequence, all six levels of the LCIM are affected by this issue.

A.2.7.5 Connection to DSEEP Steps and Artifacts

This issue concerns all information products of the DSEEP Steps 1 to 5.

Step 1 (Define Simulation Environment Objectives):

- Needs Statement;

- Objectives Statement; and
- Initial Planning Documents.

Step 2 (Perform Conceptual Analysis):

- Scenario(s);
- Conceptual Model;
- Simulation Environment Requirements; and
- Simulation Environment Test Criteria.

Step 3 (Design Simulation Environment):

- List of Selected Member Applications;
- List of Requirements Gaps;
- Simulation Environment Design;
- Member Application Designs; and
- Detailed Planning Documents.

Step 4 (Develop Simulation Environment):

- Simulation Data Exchange Model; and
- Simulation Environment Agreements.

Step 5 (Integrate and Test Simulation Environment):

- Execution Environment Description.

A.2.7.6 Possible Solution Approaches

An obvious approach would be the following:

1) Define/overwork the structure and contents for all artifacts of DSEEP Steps 1 to 5 as detailed as possible, e.g., using plain text or templates as defined by NAF;
2) Where appropriate, define artifacts as XML documents, and provide corresponding XML schema definitions (example: XML schema for FAD, as defined by SISO FEAT standard [FEAT]);
3) Else, propose other standards (example: *Conceptual Models* may be documented in UML notation or as process models); and
4) Provide hints for tailoring.

A.2.7.7 Existing Implementations and Their Information Products

The SISO FEAT standard defines an XML schema for federation engineering documentation, and a corresponding tool, called FEAT Editor, see http://www.sisostds.org:

> *"The proposed PDG will develop an XML schema designed to record all federation agreements determined to be of use to federation developers and participants. Under the M&S Steering Committee LVCAR Implementation program, a team led by JHU/APL has performed extensive research on existing federation agreements and templates. From this research, this team has derived a detailed set of requirements for a FEAT. The team has implemented an XML schema based on the requirements and performed some preliminary experimentation on applying the schema to a small,*

> *extant federation agreements document. The team has also begun development of a Java-based tool to simplify development of federation agreements conformant with the XML schema."*

Currently (March 2011), the FEAT PDG provides a template for federation agreements as an XML schema, and a first version of the FEAT Editor, accompanied by a Users' Guide (NSAD-R-2010-052_LVC_Fed_Agreements_User_Guide.pdf, see SISO website) and a Programmer's Reference.

A.2.7.8 Recommended Information Products

See referenced documents from the "related work" section.

A.2.7.9 Examples/Prototypes of Selected Information Products

Users' Guide and a Programmer's Reference for *FEAT Editor* of SISO FEAT PDG.

A.3 ISSUE GROUP "FIDELITY" (FI)

A.3.1 FI-01 Lack of Formalized Description for Entity Aggregations

A.3.1.1 Problem Definition

There is a lack of a formalized notation for description of aggregation levels of entities in conceptual models. Due to lack of such a notation, implementation independent ways of mapping between different aggregation levels cannot be facilitated. Thus, simulation members cannot access the rules for aggregating or disaggregating simulation entities that are communicated within the simulation environment.

A.3.1.2 Extended Problem Description

Aggregation and (especially!) disaggregation [1] of force structures is a common problem when coupling simulation systems which operate on different levels (e.g., single unit vs. battalion). Fair fight issues may arise when entities (e.g., single unit vs. battalion) are disaggregated in inconsistent ways. For example a team that is defined to be an aggregate of nine members can be disaggregated into twelve individuals by the receiving simulation.

A.3.1.3 Related Work

FEAT elements that are related with FI-01:

- fedagree:design: Entity aggregations should be described at design agreements:
  - fedagree:conceptualModel;
  - fedagree:objectivesAndRequirements; and
  - fedagree:memberApps.
- fedagree:execution: Entity aggregation descriptions required for data logging and replay:
  - fedagree:dataLogging; and
  - fedagree:dataReplay.
- fedagree:management: Entity aggregation descriptions would be required for VVA and test plan:
  - fedagree:vva; and
  - fedagree:testPlan.

- fedagree:data: Encodings, exchange models and data dictionary are related with entity aggregation descriptions:
  - fedagree:encodings;
  - fedagree:dataExchangeModels; and
  - fedagree:dataDictionary.
- fedagree:modeling: Algorithms for aggregation/disaggregation and procedures for adjudicating effects will require descriptions for entity aggregations:
  - fedagree:effectsAdjudication; and
  - fedagree:aggregation.

A.3.1.4 Connection to LCIM Level

LCIM 4 – Pragmatic: Entities and their semantic meanings are known, but the relations (aggregation/disaggregation) between them are unknown.

A.3.1.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 2.2, and it can be observed at 4.1, 4.2 and 7.1.

Activity 2.2: Develop Conceptual Model

Conceptual model development phase of DSEEP is closely related to this issue. Formalized descriptions for entity aggregations will help the modelers to develop a standardized and consistent representation of the problem domain.

Activity 3.1: Select Member Applications and Activity 3.3: Design Member Applications

Selection of member applications (for re-use) or design of new members should consider semantic and syntactic compatibility issues. Existing formalized descriptions will help in identifying compatible members (in the case of re-use) and ensure that the design of new members will lead to more interoperable members.

Activity 4.1: Develop Simulation Data Exchange Model and Activity 4.2: Establish Simulation Environment Agreements

The development of simulation members should cater for formalized entity aggregation descriptions both when implementing simulation state objects and their processing. In addition, if different participating members operate on different entity aggregation levels, mapping logic has to be implemented to ensure successful interoperation.

Activity 5.1: Plan Execution and Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Integration and Testing phase can be planned, prepared and executed more conveniently if standardized entity aggregation levels are clearly documented.

Activity 7.1: Analyze Data

Automated post processing of simulation results can be facilitated more easily if formalized entity aggregation descriptions exist. Processing tools could rely on those standard descriptions to interpret results.

A.3.1.6 Possible Solution Approaches

There are two different possible solution approaches for this issue:

- Development of documentation for domain specific standardized aggregation levels for common entities: Aggregation levels like team, battalion, etc., should be documented in a common standard and all relative properties should be described; and
- Description of aggregate entities with standardized specification languages: B Method [2] and Event-B [3] are specification languages that allow representation of different entities and their refinements (disaggregation) with a well-defined specification language. These languages can be used to specify the relations of attributes that effect the simulation execution at different aggregation levels. They provide validation of relation definitions, so inconsistencies between aggregation levels would be minimized.

A.3.1.7 Existing Implementations and Their Information Products

RPR FOM provides an entity class named AggregateEntity for representing entities with aggregations in a common way.

JC3IEDM can describe “compositions of things”. It has several concepts of “composition”, e.g., HOLDING, ESTABLISHMENT and OBJECT-ITEM-ASSOCIATION. Convoy, battalion, etc., appear as “size-indicator” for ORGANISATION, but don’t give any information about composition.

SISO MSDL (Military Scenario Definition Language) defines the hierarchy and also equipment at scenario units.

There is no standardized solution for using specification languages for formalizing aggregation levels.

A.3.1.8 Recommended Information Products

RPR FOM includes an AggregateEntity Object Class for representing aggregates of entities. It contains attributes of number, formation, dimensions, etc., that formalizes the description of aggregates. But it is not sufficient to be a universal solution for description of entity aggregates.

A.3.1.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.3.1.10 References

[1] P.K. Davis and J.H. Bigelow. Experiments in Multi-Resolution Modeling (MRM). RAND Corporation, 1998.

[2] J.R. Abrial. 1996. The B-book: Assigning Programs to Meanings. Cambridge Univ Pr.

[3] J.R. Abrial. 2010. Modeling in Event-B: System and Software Engineering. Cambridge Univ Pr.

A.3.2 FI-02 Lack of Agreed Levels for Entity Resolution

A.3.2.1 Problem Definition

Two semantically equivalent entities may be represented in different levels of detail. For example, a human may be defined with less or more properties. Simulation members may disagree on level of detail of shared entities in a simulation environment.

A.3.2.2 Extended Problem Description

Entity resolution can be represented with its attributes, such as the number of various weapons held by each battalion rather than assigning the battalion a net strength.

Talking about the relative resolutions about two entities is meaningless, because the resolution relationships between them are likely to be complex and confusing, with one entity's resolution being higher in some attributes, lower in others, and the same in still others. [1]

A.3.2.3 Related Work

FEAT elements that are related with FI-02:

- fedagree:design: Entity resolution levels should be described at design agreements:
    - fedagree:conceptualModel;
    - fedagree:objectivesAndRequirements; and
    - fedagree:memberApps.
- fedagree:execution: Entity resolution levels should be described for data logging and replay:
    - fedagree:dataLogging; and
    - fedagree:dataReplay.
- fedagree:management: Entity resolution levels would be required for VVA and test plan:
    - fedagree:vva; and
    - fedagree:testPlan.
- fedagree:data: Encodings, exchange models and data dictionary are related with entity resolution levels:
    - fedagree:encodings;
    - fedagree:dataExchangeModels; and
    - fedagree:dataDictionary.
- fedagree:modeling: Algorithms for aggregation/disaggregation and procedures for adjudicating effects will require descriptions for entity resolution levels:
    - fedagree:effectsAdjudication; and
    - fedagree:aggregation.

A.3.2.4 Connection to LCIM Level

LCIM 4 – Pragmatic: Semantically equivalent entities are represented in different resolutions/views.

A.3.2.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 2.2, and it can be observed in Activities 4.1, 4.2 and 7.1.

Activity 2.2: Develop Conceptual Model

Conceptual model development phase of DSEEP is closely related to this issue. Agreed levels for entity resolutions will help the modelers to develop a standardized and consistent representation of the problem domain.

Activity 3.1: Select Member Applications and Activity 3.3: Design Member Applications

Selection of member applications (for re-use) or design of new members should consider semantic and syntactic compatibility issues. Existing formalized descriptions will help in identifying compatible members (in the case of re-use) and ensure that the design of new members will lead to more interoperable members.

Activity 4.1: Develop Simulation Data Exchange Model and Activity 4.2: Establish Simulation Environment Agreements

The development of simulation members should cater for standardized levels of entity resolutions both when implementing simulation state objects and their processing. In addition, if different participating members operate on different entity resolution levels, mapping logic has to be implemented to ensure successful interoperation.

Activity 5.1: Plan Execution and Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Integration and Testing phase can be planned, prepared and executed more conveniently if standardized entity resolution levels are clearly documented.

Activity 7.1: Analyze Data

Automated post processing of simulation results can be facilitated more easily if formalized entity resolution descriptions exist. Processing tools could rely on those standard descriptions to interpret results.

A.3.2.6 Possible Solution Approaches

FI-02 has three possible solution approaches, similar to FI-01:

- Development of documentation for domain specific standardized resolution levels for common entities: Resolution levels for different entities should be documented in a common standard and all relative properties should be described;
- Development of mediation functions to convert detail level of semantically equivalent entities; and
- Description of entities at different resolutions with standardized specification languages: B Method and Event-B are specification languages that allow representation of different entities and their refinements (high-resolution entity) with a well-defined specification language. These languages can be used to specify the relations of attributes that effect the simulation execution at different resolution levels. They provide validation of relation definitions, so inconsistencies between resolution levels would be minimized.

A.3.2.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.3.2.8 Recommended Information Products

No recommendations made by MSG-086.

A.3.2.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.3.2.10 References

[1] P. Davis and R. Hillestad, “Families of Models that Cross Levels of Resolution: Issues for Design, Calibration and Management”, in Proceedings of the 25th Conference on Winter Simulation, ACM, 1993, pp. 1003-1012.

A.3.3 FI-03 Lack of Agreed Classifications for Data Fidelity Levels

A.3.3.1 Problem Definition

The ingredients of data fidelity are **accuracy** and **precision.** Data resolution determines precision. Sensitivity of the models determines their ability to respond to differences in high resolution data values.

A.3.3.2 Extended Problem Description

Higher resolution leads to higher precision which in turn facilitates higher accuracy. Different fidelity levels in data exchanged between simulation members may cause disagreement in perception of the environment. For instance; location of entities can be represented in different precision in different simulation members.

Description of data fidelity includes the definition of terms precision and accuracy:

- ***Precision*:**
  1) The quality or state of being clearly depicted, definite, measured or calculated.
  2) A quality associated with the spread of data obtained in repetitions of an experiment as measured by variance; the lower the variance, the higher the precision.
  3) A measure of how meticulously or rigorously computational processes are described or performed by a model or simulation. [1]
- ***Accuracy:*** The degree to which a parameter or variable or set of parameters or variables within a model or simulation conform exactly to reality or to some chosen standard or referent. See resolution, fidelity, precision. [1]

Lack of agreed fidelity levels for LINK systems Simulations of LinkX systems (Joint Tactical Information Distribution System – JTIDS) may not implement the LinkX communication protocol in the same level of fidelity. For example different LinkX node (unit) simulations may reduce LinkX state transition diagrams in different ways. Transmission delays may be computed as in real implementation or randomized. LinkX node simulations with different levels of fidelity may not be able to communicate in the simulation environment.

A.3.3.3 Related Work

FEAT elements that relate with FI-03:

- fedagree:design: Data fidelity classifications should be described at design agreements:
  - fedagree:conceptualModel;
  - fedagree:objectivesAndRequirements; and
  - fedagree:memberApps.

- fedagree:management: Data fidelity classifications would be required for VVA and test plan:
    - fedagree:vva; and
    - fedagree:testPlan.
- fedagree:data: Encodings, exchange models and data dictionary are related with data fidelity classifications:
    - fedagree:encodings;
    - fedagree:dataExchangeModels; and
    - fedagree:dataDictionary.
- fedagree:modeling: Algorithms for aggregation/disaggregation and procedures for adjudicating effects will require descriptions for data fidelity classifications:
    - fedagree:effectsAdjudication; and
    - fedagree:aggregation.

A.3.3.4 Connection to LCIM Level

LCIM 3 – Semantic: Different data fidelity levels will result in different data representations, accuracy and precision problems and cause semantic mismatch.

A.3.3.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 2.2, and it can be observed at 4.1 and 4.2.

Activity 2.2: Develop Conceptual Model

Conceptual model development phase should ensure that correct data types are assigned to entity attributes to cater for required data precision levels. In that respect, agreed levels of data resolutions for specific domains would facilitate a more systematic modeling process.

Activity 3.2: Design Simulation Environment and Activity 3.3: Design Member Applications

Data fidelity requirements should be considered during the design phase so that inputs, outputs and state variables are typed in correct precision. Agreed fidelity classifications should be an input to the design phase so that different members are aligned in terms of data fidelity.

Activity 4.1: Develop Simulation Data Exchange Model and Activity 4.2: Establish Simulation Environment Agreements and Activity 4.3: Implement Member Application Designs

Data simulation exchange model and simulation environment agreements should be implemented in accordance with data fidelity requirements. In addition, if known data fidelity mismatches exist between participating members, mapping functions need to be implemented

Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Simulation integration and testing should explicitly involve steps or procedures to ensure that participating members exchange data with expected fidelity at both ends and that data conversions (if any) preserve contractual fidelity constraints.

A.3.3.6 Possible Solution Approaches

Domain specific classification of data fidelity levels would facilitate more coherent simulation environments. An example from the domain of bioluminescence tomography is the data fidelity levels L1 and L2 [2].

Development of mediation functions to convert data fidelity levels of semantically equivalent entities.

A.3.3.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.3.3.8 Recommended Information Products

No recommendations made by MSG-086.

A.3.3.9 Examples/Prototypes of Selected Information Products

- DTED levels provide a convenient classification for raster elevation data resolution.
- Fidelity levels for Link-16 network simulations are defined by SISO (SISO-STD-002 Standard for Link-16 Simulations, 2006).

A.3.3.10 References

[1] 99S-SIW-167 Report from the Fidelity Implementation Study Group.

[2] H. Gao and H. Zhao, Multi-Level Bioluminescence Tomography Based on Radiative Transfer Equation, Part 2: Total Variation and l1 Data Fidelity, Optics Express, 2010.

A.3.4 FI-04 Lack of Formalized Transfer Mediation Functions Between Incompatible Entity Resolution and/or Data Fidelity Levels

A.3.4.1 Problem Definition

When models or simulators with different fidelity levels need to interoperate within the same simulation, information exchange between those models is likely to involve various transformation and mapping procedures to ensure that models are aligned in terms of their interpretation of exchanged data elements. It is desirable to develop a systematic (and if possible a formal) approach to this model alignment requirement, rather than leave it to custom and often ad- hoc methods employed by the model implementers.

This issue is slightly related to issue number FI-05 “Lack of agreed critical behaviours and corresponding algorithms”.

A.3.4.2 Extended Problem Description

The issue of multiple resolution entity descriptions pertaining to the same real-world entity appears in two distinct forms:

1) When constructing the model of a system the modeler may have to develop multiple representations of the same system each with different resolutions (normally to form an ordered set of multi-resolution models) to meet the requirements of different customers or different analysis requirements of the same customer. This is known as Multi-Resolution Modeling (MRM).
2) When developing simulation applications, the simulationist may have to compose or orchestrate models with different resolutions (hence map entities with different resolutions), paying particular attention to resolution mapping requirements. This is known as Cross-Resolution Modeling (CRM).

The problem described here is mostly related to the second form (i.e., cross-resolution modeling). In a cross-resolution modeling scenario it is often necessary to employ appropriate conversion procedures between incompatible resolution levels. Unfortunately the current practice is mostly to implement ad-hoc conversion logic somewhere in between the models exchanging incompatible entities.

Similarly, data fidelity mismatches are likely to appear between models with different resolution due to either differences in algorithmic (behavioral) complexity or other types of fidelity mismatches. Such mismatches have to be compensated via appropriate conversion procedures.

It is worth noting that standardization of (or formalization support for) conversion rules needs to be addressed separately from the standardization of mechanisms or implementation frameworks for those rules. The former has more to do with representational or denotational semantics (both for entities or data), whereas the latter has more to do with operational semantics.

A.3.4.3 Related Work

FEAT elements that relate with FI-04:

- fedagree:management: Transfer mediation functions would effect VVA and test plan:
  - fedagree:vva; and
  - fedagree:testPlan.
- fedagree:data: Transfer mediation functions are directly related with Encodings and exchange models:
  - fedagree:encodings; and
  - fedagree:dataExchangeModels.
- fedagree:modeling: Algorithms for aggregation/disaggregation and procedures for adjudicating effects will require descriptions for transfer mediation functions:
  - fedagree:effectsAdjudication; and
  - fedagree:aggregation.

A.3.4.4 Connection to LCIM Level

LCIM 5 – Dynamic: Systems will be able to re-orient information production and consumption based on understood changes to meaning.

A.3.4.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 3.3, and it can be observed at 4.1, 4.2 and 4.3

Activity 3.1: Select Member Applications and Activity 3.3: Design Member Applications

During selection of exiting members (for re-use) or design of new members mediation/mapping requirements should be carefully considered. At this stage, the designer should document the design of such adapter functions in terms of formalized or agreed descriptions or readymade (off-the shelf) components if any.

Activity 4.1: Develop Simulation Data Exchange Model and Activity 4.2: Establish Simulation Environment Agreements and Activity 4.3: Implement Member Application Designs

Development process should make use of any available standard implementations (libraries) for entity or data mapping. Verification of the developed components should be done against design artifacts that explicitly address mapping requirements.

Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Integration and testing process should involve dedicated procedures that explicitly address verification of the implemented mapping functions.

A.3.4.6 Possible Solution Approaches

A systematic approach to categorization, tagging, discovery, deployment and use of transformation functions should be encouraged. It is desirable to develop formal approaches and methodologies for such mapping tasks. In fact, at least domain specific converter libraries can be developed for commonly used, community-agreed mapping functions. It is also desirable to define standardized fidelity levels for various data (such as geographical terrain data, visual image resolution, etc.) which would then facilitate standardization of conversion procedures.

A.3.4.7 Existing Implementations and Their Information Products

In service-oriented architectures similar requirements exist. In particular, automation of business process via orchestration of web services requires automatic insertion of mapping functions between the interfaces of incompatible web services.

In Defence Concept Modelling Framework (DCMF) standardises methods for the collection and representation of knowledge, as well as creating a common infrastructure for its storage, knowledge reuse can be realised on a larger scale, i.e., the same knowledge can be used in several development contexts.

A.3.4.8 Recommended Information Products

- EU 7th FP Project: CONNECT: Emergent Connectors for Eternal Software Systems http://www.connect-forever.eu.
- M. Autili et al., “Towards a Connector Algebra”, In Proceedings of ISoLA 2010.

A.3.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.3.5 FI-05 Lack of Agreed Critical Behaviours and Corresponding Algorithms

A.3.5.1 Problem Definition

Critical behaviours include dynamic models of humans and systems, such as platforms, sensors, weapons, equipment and networks. Generation of behaviors is accomplished by algorithms that execute the models over time. In some cases an algorithm is a direct representation of some behavior. Lack of agreement on the levels of fidelity of models and lack of information on their associated algorithms lead to inconsistent joint behavior in distributed simulation.

A.3.5.2 Extended Problem Description

Unrealistic model performance effects are results of the lack of fidelity. For example, unrealistic radar coverage and 100% hit probability could result in a behavior of a fantastically successful air defence system. For the models of physical systems (i.e., systems that have a physical counterpart in the real world) the model's behavior are implemented via certain algorithms that are based on physical laws.

If two interacting simulators do not implement those behavioral algorithms in the same way then even if data mappings are provided and technical interoperability is achieved, due to inconsistent level of physical realism fair fight issues may arise.

For example, consider two combat simulations that interoperate with each other. The effects of weapons must be consistent and realistic from both sides of a weapons engagement. Suppose simulation A represents air forces and simulation B represents ground forces. Assume that an aircraft modeled in simulation A has a route over a certain country, and simulation B has a ground target equipped with Surface-to-Air Missiles (SAM) located in a faraway location. If simulation B models the SAM system unrealistically with a very long range and very high hit probability, aircraft in simulation A could be destroyed without noticing the attacking ground target.

Human behavior modeling is an issue that presently seems to elude agreed fidelity descriptions. In [1] the authors suggest to study socio-cultural attributes with consideration of four characteristics (abbreviated MALO): Mission, Area of responsibility, Level of command, and Operator expertise.

Algorithms defining communication models determine the performance of the communication systems. If these algorithms are implemented inconsistently among participating simulations fair fight issues may arise. For instance, one simulator may account for the interference effect of surrounding systems while the other may ignore them. Similarly one simulator may consider ducting effects of radar propagation while the other may not.

A.3.5.3 Related Work

FEAT elements that relate with FI-05:

- fedagree:management: Critical behaviours and corresponding algorithms would effect VVA and test plan:
    - fedagree:vva; and
    - fedagree:testPlan.
- fedagree:modeling: Algorithms for aggregation/disaggregation and procedures for adjudicating effects will require agreed critical behaviours and corresponding algorithms:
    - fedagree:effectsAdjudication; and
    - fedagree:aggregation.

A.3.5.4 Connection to LCIM Level

LCIM 6 – Conceptual: Systems should be aware of each other's processes, contexts, and modeling assumptions.

A.3.5.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 3.1 and 3.3, and it can be observed at 4.3

Activity 3.1: Select Member Applications and Activity 3.3: Design Member Applications

Design phase should include selection of correct and relevant algorithms across the simulation consistently across the whole simulation. Agreed behaviour implementations (e.g., known reference implementations (SW libraries) or reference technical documentations in the literature) should be an input to the design phase.

Activity 4.3: Implement Member Application Designs

The development phase should consider the re-use of verified off-the shelf components if available. If developed in house, such algorithms should be verified against reference technical documentations, reference test data, etc.

Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Integration and test procedures should ensure that participating members implement common critical behaviours in a consistent way. Most of the issues/inconsistencies should be prevented through the use of verified off-the-shelf algorithm implementations. However in the case of in house development, especially when re-using coarse grain simulation members with under documented behaviour implementations, integration testing becomes much more crucial.

A.3.5.6 Possible Solution Approaches

Specifying critical dynamic behaviours can be achieved by first enumerating most widely used behaviours in modeling of physical systems (including humans). Creating a catalogue that involves standardized descriptions of alternative algorithms.

A catalogue of algorithms can then be formed for certain most widely used dynamical behaviours, including alternative algorithms of the same behaviours for different fidelity levels.

Model implementers can consult this catalogue to implement most widely used physical behaviours and clearly document their selection so that simulation application developers can choose the right models for consistent interoperation.

Since algorithms would be specified to indicate their data input requirements as well as their execution logic, these data requirements should be specified in terms of entities found in standardized conceptual models where possible. This in turn, would contribute to consistent interpretation of data and consistent generation of behaviours.

Service-oriented architectures help sharing algorithm implementations through services. Critical algorithms are implemented only once and are provided to all simulation systems in the synthetic environment, e.g., via network.

A.3.5.7 Existing Implementations and Their Information Products

Dead reckoning algorithms used in DIS can be given as an example of agreed critical dynamic behaviour.

German SD VIntEL approach is an example of service-oriented algorithm implementation sharing.

A.3.5.8 Recommended Information Products

No recommendations made by MSG-086.

A.3.5.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.3.5.10 References

[1] S.K. Numrich and P.M. Picucci, New challenges: Human, Social, Cultural, and Behavioral Modeling, In "Engineering Principles of Combat Modeling and Distributed Simulation", Edited by A. Tolk, Wiley, 2012.

A.3.6 FI-06 Inconsistent Human Machine Interfaces

A.3.6.1 Problem Definition

Even if the underlying object or conceptual models of two simulations have the same level of fidelity, Human Machine Interface can cause different level of fidelity perceptions due to different interpretation and presentation (visual, audio, tactile, motion) at the user interface level.

Simulators participating in a simulation scenario should present consistent user interfaces to the user with respect to objectives of the Simulation, so that users are exposed to an equivalent level of look feel and interaction experience.

A.3.6.2 Extended Problem Description

Due to the different HMI implementation of Simulation Systems, the users of these systems may encounter different levels of perceptions and interpretations. Two simulations having the same entity and data resolution levels might have different HMI fidelity levels due to the difference in man-machine interface. Simply to overcome this problem, each simulation systems should present consistent user interfaces to the users. Otherwise inconsistent cues presented to user may cause fair fight issues. Fair fight issues should be avoided to ensure an effective training experience. The following definitions of fair fight are given to clarify the implications of this issue further for the reader:

- ***Definition of Fair Fight:*** *Two or more simulations may be considered to be in a fair fight when differences in the simulations' performance characteristics have significantly less effect on the outcome of the conflict than actions taken by the simulation participants*
- ***SEDRIS Fair Fight Definition:*** *A simulation or exercise conducted such that differences in the simulator or training system technology do not unduly result in one force or entity having an advantage over another.*

Several examples for fair fight violation cases can be given. For instance wide Out-The-Window (OTW) vs. narrow OTW displays would cause different user perception experience of the environment. Another example is two opposing ships of the same model in a war game where bridge displays in one of the ships is Electronic Chart and Display Information System (ECDIS) compliant but the other is not. Alerts that are available to the first player may not be available to the second.

In simulation environment Human Machine Interface (HMI) can be studied in terms of four modalities:

1) Visual;
2) Audio;
3) Tactile; and
4) Motion.

A.3.6.3 Related Work

FEAT elements that relate with FI-06:

- fedagree:design: Human machine interfaces should be described at design agreements:
    - fedagree:conceptualModel;
    - fedagree:objectivesAndRequirements; and
    - fedagree:memberApps.
- fedagree:execution: Human machine interfaces will effect time management and update resolution:
    - fedagree:timeManagement; and
    - fedagree:updateRates.
- fedagree:infrastructure: Secondary communication channels and hardware configurations are required for human machine interface design:
    - fedagree:secondaryCommChannels; and
    - fedagree:hardwareConfiguration.

A.3.6.4 Connection to LCIM Level

LCIM 4 – Pragmatic: View and meaning of data might be different at different Human Machine Interfaces.

A.3.6.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is at Activity 1.1, and it can be observed at 5.2.

Activity 1.1: Identify User/Sponsor Needs

The fidelity level of Human-Machine interfaces should be carefully considered early in the process when setting the objective of the simulation exercise and identify user/sponsor needs. Domain specific agreed HMI fidelity level specifications should be an input to this step if available.

Activity 2.1: Develop Scenario and Activity 2.2: Develop Conceptual Model and Activity 2.3: Develop Simulation Environment Requirements

Scenario development, development of the simulation environment requirements, and conceptual model development should address HMI fidelity issues explicitly and ensure a consistent experience across the simulation exercise (for all simulation users). Required HMI fidelity levels specified in Step 1 should be an input to this step.

Activity 3.1: Select Member Applications and Activity 3.3: Design Member Applications

Selection of member applications and design of the new members that present any interaction interfaces to the users should involve references to HMI fidelity requirements in a consistent way.

Activity 4.1: Develop Simulation Data Exchange Model and Activity 4.2: Establish Simulation Environment Agreements and Activity 4.3: Implement Member Application Designs and Activity 4.4: Implement Simulation Environment Infrastructure

All phases and activities of this step should respect the required HMI fidelity level. Otherwise the implemented product cannot meet the objectives and requirements of the user/sponsor.

Activity 5.2: Integrate Simulation Environment and Activity 5.3: Test Simulation Environment

Integration and testing process should involve explicit references to required HMI fidelity levels.

Activity 7.1: Analyze Data and Activity 7.2: Evaluate and Feedback Results

Analysis and evaluation of the results cannot be done in a systematic way unless HMI fidelity is well documented and is used as input at this stage.

A.3.6.6 Possible Solution Approaches

- Although related with fidelity of conceptual models, the fidelity of Man-machine interface needs to be considered as a distinct fidelity issue; and
- Community specific standardization of visual, audio, tactile and motion cues can help to create a common user interaction experience.

A.3.6.7 Existing Implementations and Their Information Products

An example of standardization of visual symbols is S57/S52 standards where S57 is the standard representation of maritime vector data, and S52 defines a standard set of visual symbols and representations of how these data are presented to the user.

Another standardization example is Allied Procedural Publication 6A (APP-6A, APP-6B, APP-6C), a NATO standard for military map marking symbols.

A.3.6.8 Recommended Information Products

- European Aviation Safety Agency Certification Specifications for Aeroplane Flight Simulation Training Devices CSFSTD(A).
- European Aviation Safety Agency Certification Specifications for Helicopter Flight Simulation Training Devices CSFSTD(H).

A.3.6.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.4 ISSUE GROUP “INFRASTRUCTURE AND TOOLS” (IN)

A.4.1 IN-01 Inconsistent Data Marshalling

A.4.1.1 Problem Definition

An inconsistency in how the member applications encode/decode data violates agreements on data exchange and can cause crashes, exceptions and unexpected behaviour.

Member applications exchange information using a common Information Exchange Data Model (IEDM), e.g., a HLA Federation Object Model (FOM). During execution of a simulation environment, data is encoded and decoded by the member applications.

A.4.1.2 Extended Problem Description

This issue occurs in runtime when data is exchanged between member applications. However the root of the problem usually can be found in process-steps pre-runtime. Example: Member application A encodes velocity as a 64 bit double Big Endian. Member application B decodes velocity as 32 bit float Little Endian.

This should have been detected during the design of the simulation environment or the integration and test of the simulation environment.

Possible causes to this issue include:

- Misinterpretation of specification;
- Different implementations caused by ambiguous or unclear specification; and
- Bug (erroneous implementation).

In simulation environments that use the Simulation Execution Control State Transition pattern (MSG-052) this issue can be shown during transition between **Ready to Initialize and Initialized** for member applications when instances gets an initial update in the simulation environment.

A.4.1.3 Connection to LCIM Level

This issue occurs on the syntactic level of interoperability and may affect all levels above the syntactic level.

A.4.1.4 Connection to DSEEP Steps and Artifacts

The issue is related to the steps of DSEEP where testing and execution with other systems are conducted. However the root cause of the issue may be found in the design step where the foundations for federation agreements are set:

- 4.1 Develop Simulation Data Exchange Model;
- 4.3 Implement Member Application Designs;
- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment; and
- 6.1 Execute Simulation.

This problem should be managed during:

- 3.2 Design Simulation Environment; and
- 3.3 Design Member Application.

A.4.1.5 Possible Solution Approaches

Make sure there are runtime analysis tools available for checking compliance with federation agreements including data encoding. This will enable early detection of this issue.

If this issue occurs during step Execute Simulation the following solutions are available:

- Always clarify federation agreements (Design step);
- Use bridging methods to separate out erroneous applications and apply conversion and/or adaptations to comply with agreements; or
- Use capabilities of underlying infrastructure to apply runtime receiver- and/or sender-side data modification (requires the infrastructure can support this method).

If this issue occurs during step Integration and Test Simulation Environment and the following solutions are appropriate:

- Always clarify federation agreements (Design step);

- Adapt member application and re-test and re-integrate (re-iterate previous DSEEP steps); or
- Use bridging methods to separate out erroneous applications and apply conversion and/or adaptations to comply with agreements; or
- Use capabilities of underlying infrastructure to apply runtime receiver- and/or sender-side data modification (requires the infrastructure can support this method).

If this issue occurs during step Develop Simulation Environment the following solutions are available:

- Clarify agreements and adapt member applications.

A.4.1.6 Existing Implementations and Their Information Products

The DSEEP provides some information on the information products that should be created during Design simulation environment and Integrate and test simulation environment activities.

A.4.1.7 Recommended Information Products

Federation Agreements Document, Member Application Documentation.

A.4.1.8 Examples/Prototypes of Selected Information Products

Recorders and logging equipment for HLA data traffic.

A.4.2 IN-02 Data Overflow

A.4.2.1 Problem Definition

Data overflow occurs when:

- A receiver gets too much data to manage. The receiver cannot process all incoming data in a proper manner and this may affect the quality of the output data.
- The network load gets too high and there are packet losses. The receiver will not receive all addressed data and this may affect the quality of the output data.

Data out of bounds:

- A member application cannot handle received data correctly because the expected/designed value range is exceeded.

A.4.2.2 Extended Problem Description

Too much input data for a member application to manage:

- The member application executes on a computer with not sufficient performance; and
- The amount of input data is above the expected during the Simulation Environment Design (DSEEP) step.

Network load to high:

- The network connections does not have enough bandwidth; and
- The amount of data is above the expected data amount.

Data out of bounds:

- A member application does not match the requirements in the federation agreement. Values may become out of bounds when a legacy application is selected to be a member application due to improper application analysis.

In simulation environments that use the Simulation Execution Control State Transition pattern (MSG-052) this issue can be shown during transition between **Ready to Initialize** and **Initialized** for member applications not designed for the number of instances that is registered and updated in the simulation environment.

A.4.2.3 Connection to LCIM Level

This issue occurs on the technical level of interoperability.

A.4.2.4 Connection to DSEEP Steps and Artifacts

This problem may be detected during one of the following steps, sooner is better:

- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment;
- 6.1 Execute Simulation; and
- 7.1 Analyze Data.

This problem should be managed during:

- 3.2 Design Simulation Environment; and
- 3.3 Design Member Application.

A.4.2.5 Possible Solution Approaches

- Redefine the Simulation Environment Design.
- Use high performance hardware (computers and network equipment).
- Redefine member applications subscription declarations.
- Split the simulation environment in sub-simulations and use filters (class and value).
- In a HLA simulation environment, use Data Distribution Management (DDM).
- In a HLA simulation environment, if data is sent with transportation Reliable, change to Best-Effort if it is possible.
- In a HLA Evolved simulation environment, subscribe with Update Rate Reduction

A.4.2.6 Existing Implementations and Their Information Products

The DSEEP provides some information on the information products that should be created during Design simulation environment and Integrate and test simulation environment activities.

A.4.2.7 Recommended Information Products

Federation Agreements Document, Member Application Documentation.

A.4.2.8 Examples/Prototypes of Selected Information Products

Recorders and logging equipment for HLA data traffic.

A.4.3 IN-03 Missing Proper Network Configuration

A.4.3.1 Problem Definition

During development of a simulation environment a lot of time is often spent on locating network problems hampering proper connection of attached simulation systems.

A.4.3.2 Extended Problem Description

Network problems affecting simulation interoperability come in many different flavors. For a structured approach the ISO model for open systems interconnection (OSI reference model) sketched in Figure A-5 is helpful [1].

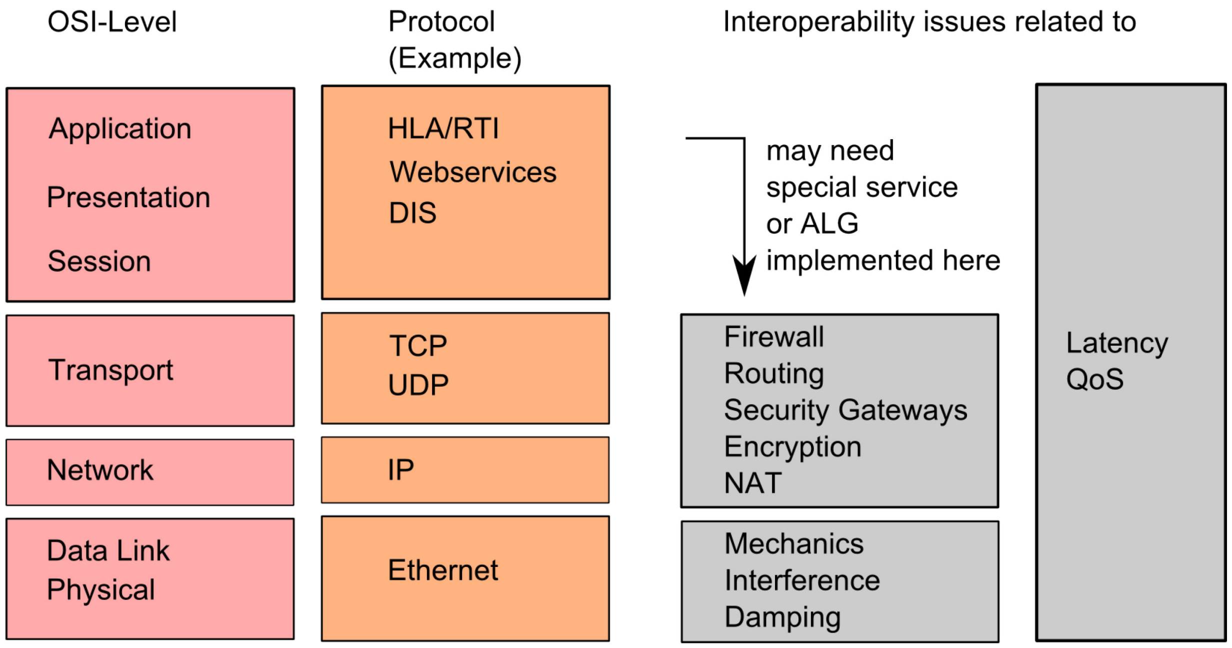


Figure A-5: OSI Layer Model.

We chose to group the 7 OSI layers into 4 different groups and give examples of typical protocols on these layers used in distributed simulation environments (bottom up order):

- The Data Link / Physical layer provides mechanical connections and is responsible for transmitting bits coded as signals over these connections. Depending on the type of the connection the signals may be of electrical, optical, RF or acoustical nature. Most often electrical Ethernet connection (RJ45 plugs) is employed, as this is the most common network connection for computer systems.
- The Network layer is responsible for routing data to its destination endpoint by properly switching necessary (electrical) connections and/or determining the forwarding of individual data packets between network nodes. Most often the Internet Protocol (IP) is employed for this purpose.
- On the Transport layer protocols like the Transmission Control Protocol (TCP) or the User Datagram Protocol (UDP) are used to provide additional services like correction of errors and packet loss, flow control, encryption, multiplexing and so on.

- Finally the Application/Presentation/Session layer forms the interface to the simulation application by providing higher level or domain specific interface functions. The most common protocols for distributed simulation are the Distributed Interactive Simulation (DIS) protocol (definition of “entities” and their attributes or domain specific events like “munition detonation”) and the High Level Architecture (HLA) (definition of common data model FOM and publish/subscribe mechanisms of sections of the FOM).

It is obvious, that each of these levels may be a source of networking problems of very different nature. Each one of these problems prevents a simulation application from properly transmitting/receiving data to/from a network and is therefore an interoperability problem on LCIM Level 0 or 1 (technical connection), as seen from the simulation applications point of view. Some examples are provided in the next section. Due to the very different nature of these problems, the possible solution approaches also form a wide spectrum of measures and therefore are discussed along with the examples.

Data Link and Physical Layer

This layer deals with mechanical, electrical, optical and RF connections. Typical problems are:

- Mechanical problems of plugs or electrical contacts (bent, broken, dirty), plugs not properly fixed to connectors or slipping out (cheap RJ45 connectors are especially prone regarding these problems). A possible solution is to check each connection during setup and to rigorously sort out any damaged connector.
- Broken cables. This may just lead to some distortion or high error rate for electrical connections. It may also be hard to detect because the liner of the cable may not show any visible damage (fiber/optical cables are particularly sensitive to this kind of problem). Besides rigorously sorting out visibly damaged cabling, a network monitoring tool may help to identify broken connections or high rates of transmission errors. More expensive testing equipment is able to precisely locate the damage within a cable.
- Electrical interference into cables due to RF sources (e.g., radio equipment or radars) or high power consuming equipment in the vicinity of the cable. This may easily be solved by proper mounting of the cable far enough away from RF sources and by using shielding and electrical filters, or simply by using fiber connections.
- RF interference in radio/RF connections. These may be caused by emissions from strong RF sources collected by the antenna, and then may be eliminated by electrical filtering or using a wired connection instead. Or they may be caused by improper frequency management, when several network connections transmit in the same frequency range.
- Damping of the signal may result in poor signal to noise ratio and therefore high error rates on a particular connection (resulting in reduction of bandwidth) or even connection loss. This is seldom a problem in wired connections when using high quality cables (however it is a typical problem in using Digital Subscriber Line (DSL) connections because of long distances and low quality cables). It is a typical problem in WIFI networks, where transmission is heavily influenced by RF damping and reflection by or within buildings. A solution approach is the use of repeaters or employing high quality cabled connections.

Transport and Network Layer

This layer deals with routing of data and low level services like flow control or error correction. Routing and other services are provided by software components, often augmented by specialized hardware to boost performance (Routers, Firewalls, Gateways, Crypto boxes). All of these components have in common, that (packetized) data traffic is inspected, interpreted, routed or even modified by the component according to a widely configurable set of rules. Typical components for IP networks are:

- **Routers**: These are connected to at least two different (in physical medium or logical segment) networks, read source and destination address/port from each incoming IP packet and forward the packet according to the configured routing tables. Additional functionality may be built into a router, like for example Network Address Translation (NAT, mapping between a single external IP address and many internal IP addresses), media conversion (“bridge”), and pure network switching (“switch”, similar routing, but works on the Link/Ethernet layer).
- **Firewalls and Security Gateways**: These are similar to routers, but provide additional functionality like monitoring IP connections, allowing connections to be initiated only from certain hosts or network partitions, dropping data packets with unknown or dangerous content or allowing flow of IP packets in one direction only (network A to network B, but not the other way round) in order to separate open and secret parts of the network.
- **Crypto Boxes and Tunneling Devices**: They work similar to Firewalls or Routers, but will also modify the data content of the IP packet, e.g., by encryption or by wrapping the content into another data packet. This of course renders the data unusable to an application, which means, that there always must be another element in the network to decrypt or unwrap the data content. This is most often used to “tunnel” data through unsecure parts of the network or to combine physically separated parts of a network into one single logical network (e.g., connect two simulation networks at different locations via unsecure DSL line). Another application is to forward data to other parts of the network, which are not accessible by standard routing, e.g., forward UDP broadcast data through routed parts of the network which will normally block broadcasts.

Nearly all problems arising from the Transport and Network layer can be attributed to improper configuration of the devices employed. The only solution approach is a detailed network planning, a deep understanding of the device function and configuration options and a careful configuration of the devices. It should be mentioned, that there may also be the chance of unexpected or erroneous device behavior. This can only be fixed by the manufacturer of the device, typically by an update of the firmware.

Application Presentation and Session Layer

Beside the above mentioned processing issues by HLA software or web services there are no specific simulation interoperability issues related to these network layers (protocol specific issues are covered in other issue groups). However, as these layer build upon the lower layers of the OSI reference model, certain specifics from lower layers may impact simulation interoperability on the Application, Presentation and Session layer indirectly, for example:

- The implementation of the HLA components (the protocol) on these layers (between the HLA API and the TCP/UDP/IP level is not standardized and proprietary to the manufacturer of these components. There may be undocumented assumptions on components on lower layers, e.g., certain ports to be open in firewalls, certain routing to be performed, which make it impossible to configure components on the Network and Transport layer in a way to work in conjunction with a certain HLA/RTI implementation. If for example it is unlikely to use HLA across different security domain, because HLA implementation most likely needs two-way-communication between simulation applications. On the other hand this can easily be accomplished using DIS, because only one-way-broadcast communication is required.
- DIS employing broadcast communication will not work in routed networks, because normally broadcasts are not routed to different network segments. It may be possible to configure this feature at the router; otherwise a special component is needed to “tunnel” broadcast communication through the routed parts of the network.

These and similar issues could be solved by equipping the components of the Transport and Network layer with a specialized Application-Layer Gateway (ALG) or similar functionality. This is a well-known practice in networking technique and is used, e.g., to receive incoming VoIP phone calls through firewalls (SIP-ALG). However, these ALGs must be built into the router or firewall components, which are typically “closed source”. Also it is vital, that the ALG can “understand” the traffic on the Application level (i.e., HLA or DIS). This is not possible for HLA, because the HLA standard does not specify a network protocol for HLA.

Latency and Quality of Service (QoS)

This topic cannot be attributed to a specific OSI layer but appears in all layers, albeit in different quality.

Latency means, that it may take some time for data being sent by one network node to appear ready to use at the receiving network node:

- On the Physical and Data Link layer the connection may have poor signal quality, thus forcing the sender to transmit at lower data rate (low bandwidth). Therefore the transmission time of the data packet increases. Also additional data traffic on the same line will lower the bandwidth available for an individual transmission (multiplexing, Ethernet collision).
- On the Network layer it is unpredictable, how fast intermediate routers and other equipment will process and forward the data, thus delaying the reception of the data packet at the receiver in an unpredictable way.
- On the Transport layer, flow control mechanisms may intentionally delay data packets or may request a re-send of certain packets, thus providing another source of latency.
- On the Application, Presentation and Session Layer the data is finally processed and handed over to the application. Again, the delay until the data becomes visible to the receiving application, heavily depends on the processing performed by the software components in this layer. Thereby it also strongly depends on the CPU resources of the receiving computer; see issue IN-02 (Data Overflow).

It should be noted here, that the term “latency” in this discussion also includes “jitter” in the data stream. This means, that the latency normally is not the same for each data packet, but may vary from packet to packet in an unpredictable way.

Quality of Service (QoS) in IP networks means a collection of requirements regarding the performance of a network connection. Beside requirements on latency and jitter there may be requirements on packet loss and throughput.

It is obvious, that a network, which does not provide the Quality of Service required by the connected simulation applications, will lead to interoperability problems, because the data required by the application won’t be provided too late (latency), not at the correct frequency (jitter) or incomplete (packet loss and throughput).

As shown above, the QoS is a combination of various factors and a result of all components of the network at all layers of the OSI reference model working together properly. The solution approach therefore must be a clear specification of the QoS (regarding at least the mentioned criteria) by the simulation applications and a high quality design and setup of the network to meet these requirements.

It should be noted, that the IP protocol offers some flags to assign data to different QoS classes (used, e.g., to prioritize time critical speech or video transmission data). However, to make use of this feature, all affected network components must be capable to take into account these flags while performing their work.

A.4.3.3 Related Work

See standard literature on computer networking.

FEAT: The fedagree:infrastructure element captures most of the networking issues, as far as they are related to proper configuration of network hardware or protocol issues. However, most of the related FEAT elements refer to external documents only.

A.4.3.4 Connection to LCIM Level

All topics discussed here may be attributed to LCIM level L1 (technical).

A.4.3.5 Connection to DSEEP Steps and Artifacts

The following DSEEP steps are affected:

- 3.2 design simulation environment;
- 3.4 prepare detailed plan;
- 4.2 establish simulation environment agreements;
- 4.4 implement simulation environment infrastructure;
- 5.2 integrate simulation environment;
- 5.3 test simulation environment; and
- 6.1 execute simulation.

The following DSEEP artifacts are affected:

- Simulation environment design (from 3.2);
- Detailed planning documents (from 3.4);
- Simulation environment agreements (from 4.2);
- Implemented simulation environment infrastructure (from 4.4);
- Integrated simulation environment (from 5.2); and
- Tested simulation environment (from 5.3).

A.4.3.6 Possible Solution Approaches

The discussion above shows, that there is a broad spectrum of interoperability issues resulting from networking issues. Therefore solution approaches also form a wide spectrum of individual measures and have been discussed along with the issues. Summing up, the following approaches are considered helpful:

- Proper network setup comprising:
  - Precise specification of QoS and networking capabilities (protocols, address/port ranges) required by the individual simulation systems and other components (e.g., RTI) of the simulation environment;
  - Precise specification of additional networking requirements (encryption, security gateways, firewalls) and boundary conditions (availability/quality of lines, mounting options, RF interference);
  - Proper planning of network configuration (components, mountings, connections, logical network configuration, component configuration); and

  - Careful setup und component configuration.
- Develop standardized, open HLA network protocol; and
- Develop and integrate dedicated simulation ALGs into networking components.

A.4.3.7 Existing Implementations and Their Information Products

See standard literature on computer networking.

A.4.3.8 Recommended Information Products

See standard literature on computer networking.

A.4.3.9 Examples/Prototypes of Selected Information Products

See standard literature on computer networking.

A.4.3.10 References

[1] OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection, Hubert Zimmermann, IEEE Transactions on Communications, Vol. 28, No. 4, April 1980, pp. 425-432.

A.5 ISSUE GROUP “LVC AND C2-SIM COUPLING” (LC)

A.5.1 LC-01 Limitations Due to Integration of Live Systems

A.5.1.1 Problem Definition

Live systems without existing external interfaces are of course hard to integrate without major changes in the system itself. Proprietary protocols require the involvement of the owner of the protocol to make any changes and/or solutions to adapt the system to the distributed environment. In some cases external interfaces are defined but not accessible due to security constraints, e.g., causing a live system to crash due to inappropriate use of interfaces or accessing secure information through an external interface. Furthermore are live systems not primarily designed for simulation. Some live systems implement simulation support and can be said to be simulation-aware. The level of simulation-awareness and simulation capabilities of the live system determines the level of interoperability that can be achieved in a distributed simulation environment.

A.5.1.2 Extended Problem Description

Most legacy live systems like vehicles are made for combat or for live training but not for a simulated training or for operating in a simulated environment. So they are not simulation-aware. Modern systems often have a simulation mode which allows to train with the original equipment, e.g., an anti-aircraft system can attack simulated targets or a tank can operate in a virtual 3D-scene which is displayed on a monitor. An interface for coupling the live system with external simulations is usually not available.

For the coupling of a live system a precondition is that it has a minimum of simulation awareness, i.e., it must be possible to operate the system in a simulated mode (e.g., pressing the fire button of a weapon system without really shooting).

If this precondition is met, the first interoperability issue is the physical connection of the live system with other simulation systems. Simulation systems usually have a network adapter while live systems have any proprietary interface (often for technical test purposes). In the worst case the live system does not have any interface so that a reengineering or the help of the producer of the system is necessary. When the physical connection is established the next issues might occur with the transfer protocol and the semantics of the protocol. Another issue is caused by security constraints. Live systems often produce classified data which cannot be shared with other participants of a distributed simulation or the whole simulation network has to run in classified mode.

One fundamental characteristic of live systems is the fact that they are designed to run in real (operational) environments and in real time:

- Initialisation of system state based on simulation scenario may not be available.
- Interruptions in the execution (start/pause/stop) may not be supported.
- Such system may also require input in real time from the distributed simulation environment.

The real-time requirements may limit the use of the live system in distributed simulation environments. The real-time requirements of the live system may put additional requirements and limitations on the distributed simulation environment.

A.5.1.3 Related Work

- TM-01 – Temporal Anomalies caused by differences in Precision of Time Representation.
- TM-02 – Temporal Anomalies caused by differences in Time Resolution.
- TM-03 – Temporal Anomalies caused by Unsynchronized Time.
- TM-04 – Temporal Anomalies caused by Network Latency.

A.5.1.4 Connection to LCIM Level

This issue is concerned with the connection between a live system and a simulation. LCIM Levels 1, 2 and 3 are affected here:

- LCIM Level 1 (technical interoperability): physical connection (plug, connector, cable);
- LCIM Level 2 (syntactic interoperability): protocols, data format; and
- LCIM Level 3 (semantic interoperability): common understanding of the data.

A.5.1.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is located in Steps 3.1 (“Select member applications”) and Step 2.3 (“Develop simulation environment requirements”).

The issue may be observed in Step 5.3 (“Test simulation environment”).

A.5.1.6 Possible Solution Approaches

- Change requirements.
- Select different member applications.
- Run simulation in real time.
- Extend VC-Simulation with real-time dynamics (e.g., acceleration, speed, g-forces).

- Make sure that inputs/changes from virtual/constructive Simulation are represented in the real word.
- Design live systems with simulation in mind (make them simulation aware from the beginning) and *vice versa*:
    - Example: the German AFV PUMA offers a connector to simulation systems in a way that the operator is able to train with the original equipment in a simulated environment that is shown on the display of the AFV.
- Use adapters/gateways to overcome differences in protocol and transmission.
- Use a simulation network which is approved for the necessary security level.

A.5.1.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.5.1.8 Recommended Information Products

No recommendations made by MSG-086.

A.5.1.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.5.2 LC-02 Difference in Accuracy for Position

A.5.2.1 Problem Definition

Live systems use measured positions with limited accuracy whereas simulation systems use calculated positions with high accuracy.

A.5.2.2 Extended Problem Description

When combining live and virtual/constructive systems, the position of the live system is subject to systematic and random (space and time) errors. Therefore Virtual/Constructive systems may experience that Live-entities “jump around”.

For example live systems often use GPS for measuring their position. The accuracy of GPS can be several meters and may differ at different locations in space and time. Virtual and Constructive systems do not have this problem when positioning simulated entities (Ground Truth position).

A.5.2.3 Related Work

- VIntEL II: A German simulation experiment, where live systems were connected to virtual and constructive systems. During the experiment the GPS position of a vehicle was transferred to the simulation where the related object jumped around due to the finite accuracy of the GPS position.

A.5.2.4 Connection to LCIM Level

This issue is connected to Level 4 (pragmatic), because it is related to the interpretation of position data. It is not a semantic problem because position data may in both systems be expressed in Lat/Long-Coordinates, however many simulation systems tend to interpret all Lat/Long-Coordinates as errorless values.

A.5.2.5 Connection to DSEEP Steps and Artifacts

The origin of this issue is located in Steps 3.1 (select member applications) and Step 3.3 (design member applications).

The issue may be observed in Step 5.3 (test simulation environment).

A.5.2.6 Possible Solution Approaches

Two solution approaches seem feasible. The first approach is the improvement of the measured position data and the incorporation of additional data:

- Use sensor data fusion;
- Use additional position systems (inertial navigation, fix point navigation); and
- Velocity measurement by wheel rotation.

The second approach is to make the simulation systems aware of erroneous live data:

- Incorporate additional knowledge on position information, for example by ground clamping or Kalman-Filtering.

A.5.2.7 Existing Implementations and Their Information Products

Documentation of simulation systems should include information whether a system is aware of erroneous position data. A Federation Agreements Document should include information about ground clamping.

A.5.2.8 Recommended Information Products

Federation Agreements Document, Member Application Documentation.

A.5.2.9 Examples/Prototypes of Selected Information Products

SD VIntEL Federation Agreements Document Fall 2011 Vignette.

A.5.3 LC-03 Missing Information Exchange Between Real and Simulated Environment

A.5.3.1 Problem Definition

Changes in real environment are not easily reflected in virtual or constructive simulation systems and vice versa.

A.5.3.2 Extended Problem Description

Environmental conditions which are changed in virtual or constructive simulation systems are not easily reflected in live systems and vice versa. There are various examples of this problem:

- If a munition explodes in a constructive simulation system it is hard to notify a live player that a crater now exists that has implications for traversability or could be used for cover.
- If damage is created in a constructive simulation system it is hard to realistically reflect that damage in a live player. A system such as AWES uses flashing lights to signify damage, but there would be other effects such as hot-spots or smoke making a target more detectable that are not included.
- The effect of losses on morale in a constructive simulation system which would impair effectiveness is difficult to transfer to a live system in a way that is compatible with ethical constraints.

- The state of gate (e.g., open or closed) in the real world is not reflected in simulation systems due to missing instrumentation.
- Weather information is not transmitted into simulation systems.

A.5.3.3 Related Work

- SN-04 Difficult to exchange synthetic environment updates at runtime.
- UCATT.

A.5.3.4 Connection to LCIM Level

This issue is connected to Level 1 (technical).

A.5.3.5 Connection to DSEEP Steps and Artifacts

The issue is related to Step 2.3 (develop simulation environment requirements).

A.5.3.6 Possible Solution Approaches

- Indicate changes from simulated environment into real environment (e.g., sign/marking).
- Use displays (optronics) to broadcast changes in the simulated environment.
- Use additional sensors to detect changes in real world and transfer those changes to the simulated environment (e.g., gate state, rain/fog sensors).

A.5.3.7 Existing Implementations and Their Information Products

DEVIL – Demonstrator for the Connection of Live and Virtual Simulation 99f-siw-050.pdf.

A.5.3.8 Recommended Information Products

Specify as part of “Simulation Environment Requirements” which information must be exchanged between real and simulated world.

A.5.3.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.5.4 LC-04 Limitations in Coupling Simulations and Live Systems When Using Operational Protocols of Live Systems

A.5.4.1 Problem Definition

When coupling simulation and live systems limitations in the interoperability between these systems can exist due to restriction in the information exchange between the systems. These limitations can be caused by physical properties, available interfaces and available protocols. In the most extreme conditions this might even result in non-existent interoperability between the systems.

A.5.4.2 Extended Problem Description

Live systems are primarily designed for use in real/operational environments. In general the interfaces to other systems are also designed for the specific operational environment and its specific requirements.

However, within a distributed simulation environment additional information may be required when exchanging information with virtual/constructive simulations. Making this additional information available is often not an easy task and might require changes to the existing operational interfaces.

It is important to consider during the conceptual modelling phase that not all available interfaces of the live system provide information at the required fidelity for the simulation environment.

Example 1: Link 16 is based on a time-slot scheme and systems send out updates about their position information (PPLI) with an update rate of one second. And the resolution of the latitude and longitude values in the Link 16 message results in a positional accuracy of around 8 feet. This is sufficient for the display of a symbol of the entity in operational systems, but it is not sufficient to feed a simulation entity with the state of the system.

Example 2: A live tank might send its position via an operational protocol. However, a simulation system might also need information on platform and turret orientation which is not transmitted by the operational protocol.

A.5.4.3 Related Work

- LC-01 Limitations due to integration of live systems.

A.5.4.4 Connection to LCIM Level

This issue is connected to Level 4 (pragmatic) Interoperability and above.

A.5.4.5 Connection to DSEEP Steps and Artifacts

In DSEEP Step 3.2 “Design Simulation Environment” the problem of the coupling limitations arises.

A.5.4.6 Possible Solution Approaches

- In the conceptual model for the simulation environment there has to be a clear separation between tactical information of the live system (perceived truth) and the state of simulated objects (ground truth).
- This may require enhancing the available interfaces so that all required information can be exchanged. However making such changes to operational systems is often costly, might require involvement from the manufacturer of the system and certification of the system might restrict the changes that can be made.
- Additional transmitters and receivers can be employed to exchange information between the live systems and the simulation environment. It might be that additional data is required from the simulation environment to be able to stimulate the live systems realistically.

A.5.4.7 Existing Implementations and Their Information Products

An example for “additional transmitters” is an Air Combat Training System pod that can be attached to an aircraft. These pods add additional data link capabilities to exchange training related information between aircraft equipped with such a pod. This information can then also be replayed to the simulation environment. Within the German simulation experiment “VIntEL I” there was a real flight profile recorded and replayed into the simulation. However, the use of pods does not solve the problem mentioned above, but it merely bypasses it.

A.5.4.8 Recommended Information Products

- Documentation of operational protocols.
- Simulation Data Exchange Model (should contain information about necessary/required information exchange between systems).

A.5.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.5.5 LC-05 Limited Mutual Awareness of Simulation Systems and C2 Systems

A.5.5.1 Problem Definition

Most C2 systems are not designed for interoperation with simulation systems. Some C2 systems implement simulation support and can be said to be simulation-aware. The level of simulation-awareness and simulation capabilities of a C2 system determines the level of interoperability that can be achieved in a distributed simulation environment.

Similarly the problem may appear the other way round. If a simulation system is not “C2-aware” problems may result from mixing perceived truth and ground truth (see also LC-04).

A.5.5.2 Extended Problem Description

When using LVC systems and C2 systems in the same simulation environment:

- Different protocols are used by LVC systems and C2 systems;
- C2 systems often use proprietary (national/company) protocols;
- Different time concepts are used by LVC systems and C2 systems; and
- Data models differ in resolution and accuracy.

C2 systems work with “Perceived Truth” in reports (input) and orders and tasks (output) containing information of friendly positions and actions and activity at opposing forces. Simulation systems usually work with “Ground Truth”, e.g., position in real time (simulation time). It is stressed that a C2 system works with perceived truth only (because no other “truth” exists in reality), whereas a simulation system handles ground truth (no pendant in reality) and perceived truth independently (at least when modelled correctly, see also LC-04).

Additional problems may occur when C2 systems and simulation systems operate on different resolution levels. An example of a special case that was observed in practice is the following – Orders in a C2 system were given on a high aggregation level where subordinates had to make out details how orders shall be executed. Orders in a connected simulation system were given on a low aggregation level.

Information exchange between simulation systems and C2 systems usually requires protocol translation. Common C2 protocols / standards / data formats are MIP/DEM, ADatP-3, OTH Gold, NFFI, whereas common protocols / standards / data formats for simulation are HLA/RPR FOM, MSDL. Translation between these protocols may be complicated due to different data types, data structure, contained information, etc.

A.5.5.3 Related Work

- MSG-048 (C-BML).
- MSG-085 (Standardization for C2-Simulation Interoperation).
- MSG-119 (C2 to Simulation Interoperability (C2SIM)).

A.5.5.4 Connection to LCIM Level

Level 1 (Technical) and above.

A.5.5.5 Connection to DSEEP Steps and Artifacts

Step 2 (*Perform Conceptual Analysis*):

- 2.2 Develop Conceptual Model (information exchange relations between simulation systems and C2 systems need to be defined).

Step 3 (*Design Simulation Environment*):

- 3.1 Select Member Applications; and
- 3.3 Design Member Applications.

A.5.5.6 Possible Solution Approaches

Special adapters may provide a connection between simulation systems and C2-systems. Adapters that work with standardized protocols can be reused and reduce the cost and make it easier to use in different simulation environments.

There are different developments of a so-called C2SimProxy, which are able to mediate between a certain set of C2 systems and HLA simulations. In principle, this C2SimProxy transmits orders from the C2 systems to the simulation objects, and transmits the updated states of objects (friendly or hostile) to the C2 systems (in some cases represented by reconnaissance messages). The transmission of states and orders requires a common data model with an abstract formulation that can be translated to the demanded standards and protocols.

A standardized format for reports to C2 systems should be used. MSDL can be used which is a picture (static state description) of the current state.

A standardized format for orders and tasks to simulation systems should be used (like C-BML).

To be interoperable and easy to use in different simulation environments, systems shall use standard protocols:

- Simulation systems should produce RPR2 FOM data or specializations of it;
- Adapters shall convert “ground truth” data (RPR2 FOM data and specializations) to produce “perceived truth” data (MSDL);
- C2 systems shall understand (consume) MSDL;
- C2 systems shall produce C-BML output; and
- Simulation systems shall understand (consume) C-BML or use adapters that transforms C-BML data to understandable orders for the simulation systems.

A.5.5.7 Existing Implementations and Their Information Products

The German C2SimProxy: A Short Overview

In 2006, the German Armed Forces initiated a project named SuTBw (see Section 2.6) to provide the infrastructure for the coupling of simulation systems. One element in this infrastructure is the so-called C2SimProxy (Command and Control to Simulation Proxy). Its purpose is to couple simulation systems and C2-systems with each other, i.e., to transfer of positions and types of simulated units to C2-systems. By doing this it is possible to feed the C2-systems with simulated units, e.g., for training users at the specific C2-system. There exist former versions of the C2SimProxy which, in addition, allow orders to be sent from the C2-system to simulated units, but this feature is not implemented in the current version of the C2SimProxy.

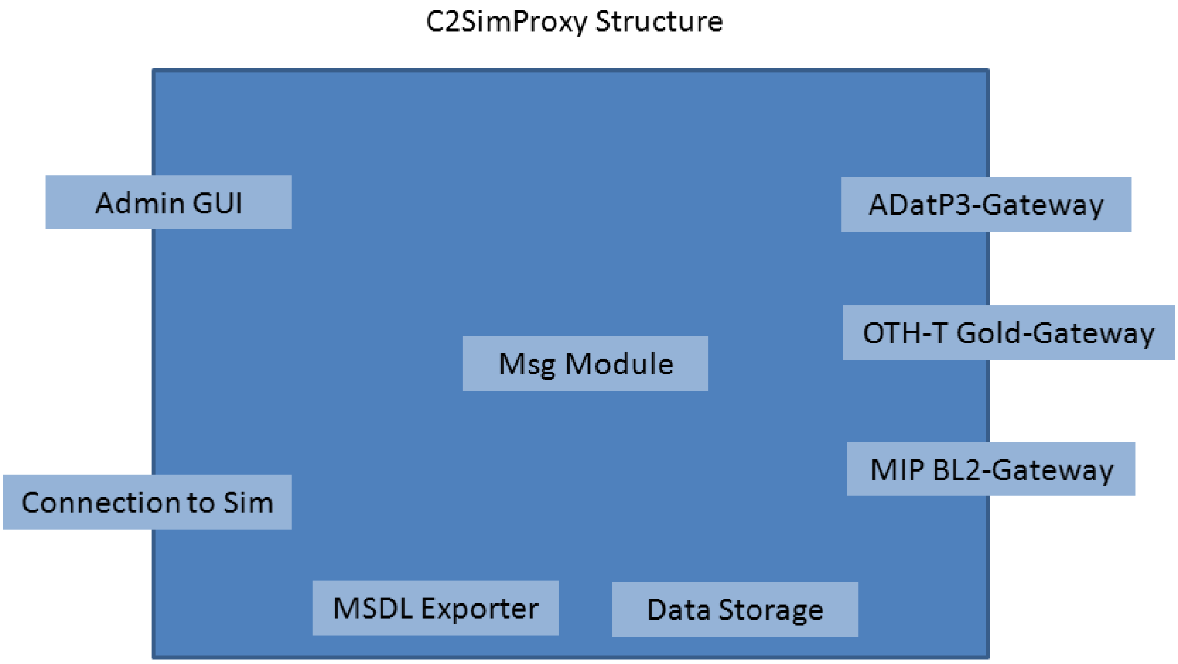


Figure A-6: Overview of C2SimProxy.

The C2SimProxy consists of different modules; some of them are possibly connected to other systems or to the user:

- Via the “Admin GUI” the user can manage the C2SimProxy;
- “Connection to Sim” represents usually the connection to a HLA-Federation;
- The “Msg Module” forms the messages from the simulation states;
- The “MSDL Exporter” constructs MSDL messages;
- The “Data Storage” contains data regarding the simulation environment; and
- There are three gateways which connect the C2SimProxy to the C2 systems.

The current C2SimProxy is able to exchange data via ADatP3, OTH Gold and MIP BL2.

Example: Imagine a friendly simulated unit which wants to report the position of an (simulated) enemy unit to the C2-system. In this case the assumption is that the C2-system knows and expects ADatP3. The information route shows up as follows.

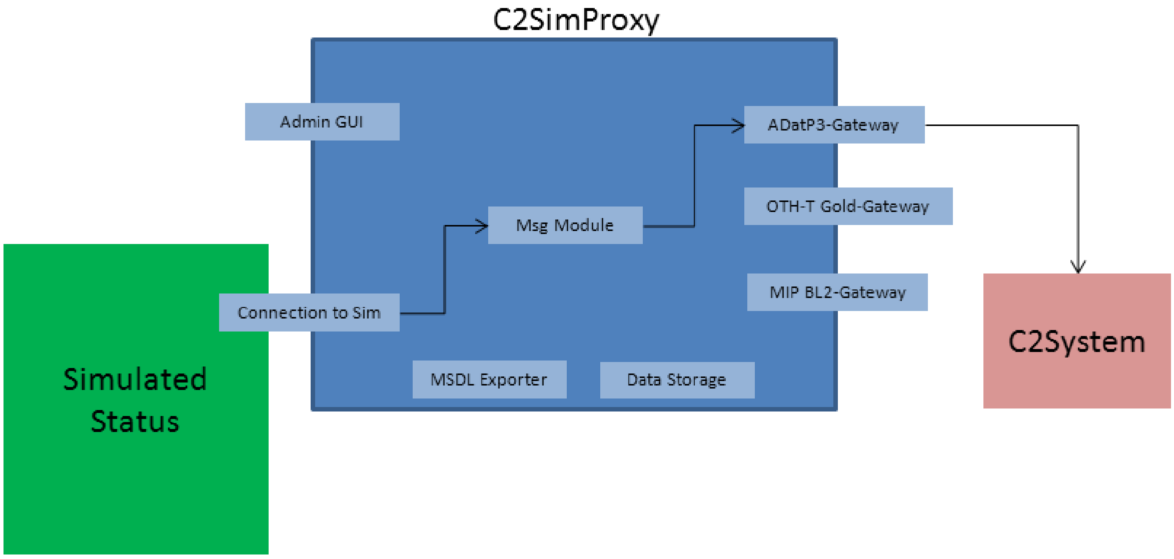


Figure A-7: Example Configuration of C2SimProxy.

In this example the Msg Module extracted the essentials out of the information from the simulated state and sends them to the ADatP3 Gateway. This Gateway formed the appropriate message for the C2-system. The C2-system now shows a simulated status, reported by a simulated entity.

NETN SIM-C2

NETN SIM-C2 is an activity within MSG-106 for the coupling between C2 systems and simulation systems. The NETN (NATO Education and Training Network) SIM-C2 concept is based on the Coalition Battle Management Language (C-BML) standard and requires the use of a C-BML server to connect and exchange Battle Management Language (BML) information between C2 and simulation systems.

The use of C-BML information (orders, reports, etc.) depends on the modelling granularity of simulation models. High Level simulations (aggregated units) can process global orders, i.e., real BML orders. Low level simulation (platforms or small aggregated units like platoons) can only process elementary military tasks.

NETN has two FOM modules to cover C-BML, one for the standard C-BML (high level) and a low level module where orders and reports are broken down for low level simulation models. The transformation is done by a C2 Agent. The C2 Agent also compiles the reports from low level simulation systems into BML reports to the C2 systems.

The C2 Agent is a separate federate in the NETN federation, not integrated in a simulation system or in a C2 system.

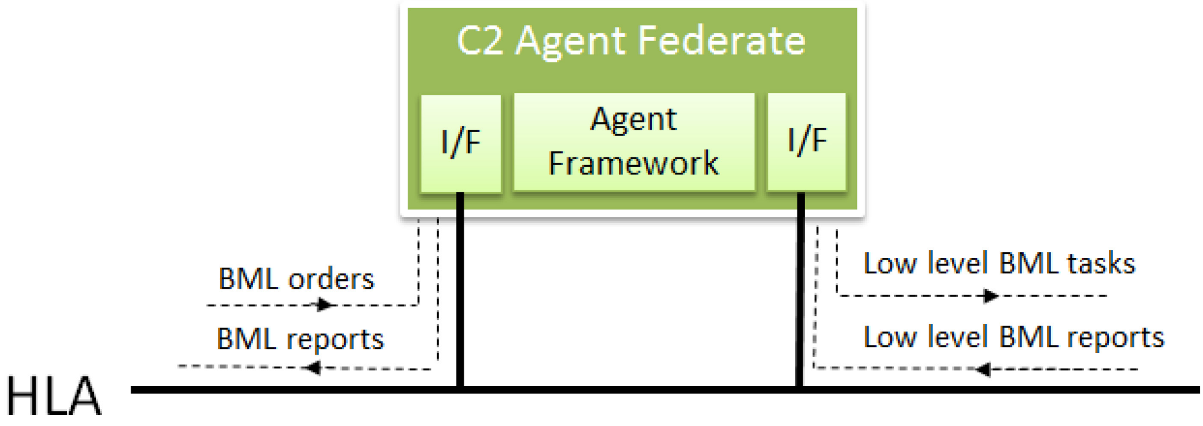
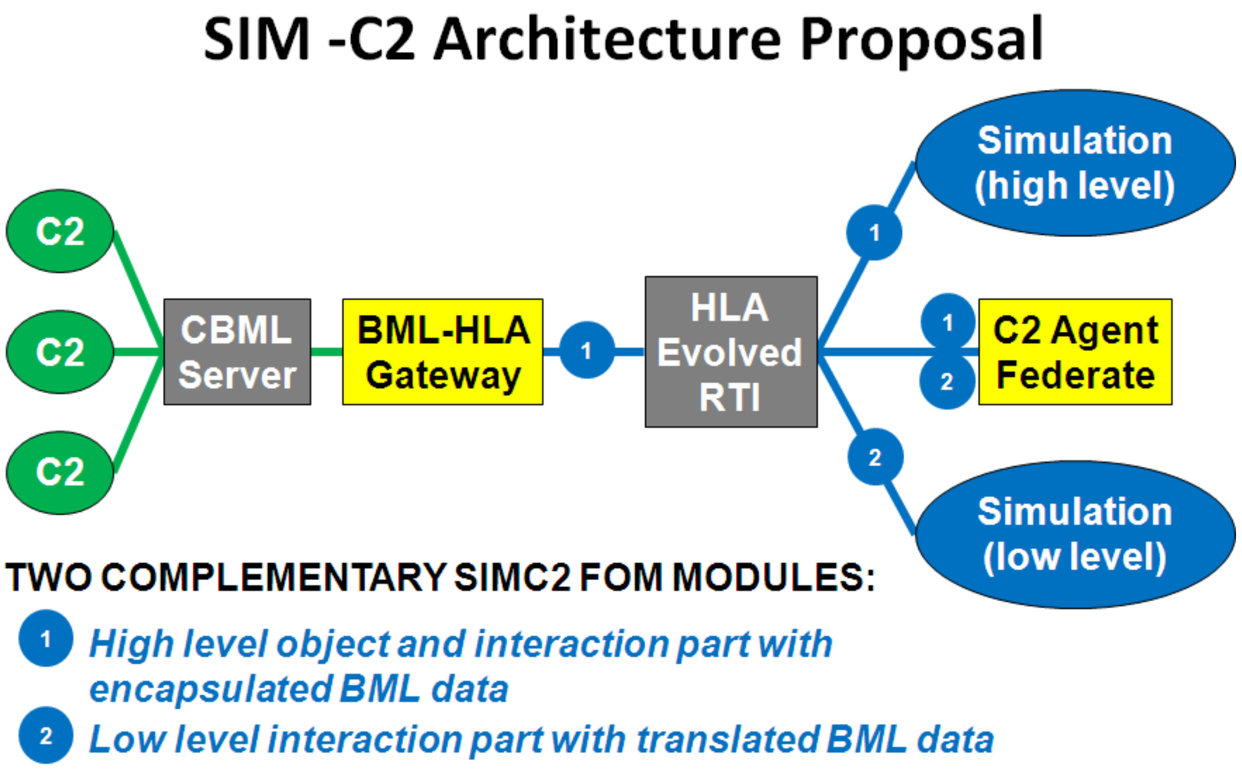


Figure A-8: NETN SIM-C2 Concept.


SIM -C2 Architecture Proposal

TWO COMPLEMENTARY SIMC2 FOM MODULES:

① *High level object and interaction part with encapsulated BML data*

② *Low level interaction part with translated BML data*

Figure A-9: NETN SIM-C2 Architecture.

A.5.5.8 Recommended Information Products

Documentation of protocols supported by simulation systems and C2 systems. Interface control documents for adaptors that define mapping between protocols.

A.5.5.9 Examples/Prototypes of Selected Information Products

Interface Control Document describing the modeling of a C2SimProxy setup.

A.6 ISSUE GROUP "ORGANIZATIONAL AND LEGAL ISSUES" (OL)

A.6.1 OL-01 Improper Project Management

A.6.1.1 Problem Definition

The development of a simulation environment is often a big and complex project, therefore proper project management is important to successfully complete such projects.

A.6.1.2 Extended Problem Description

Below are some examples of improper project management for distributed simulation projects:

- Wrong assumptions of project management about technical solutions might lead to problems. An example of this could be the assumption that HLA is a silver bullet and then blaming HLA when things don't go as expected.
- Different organisations can use a different process to manage a project, for example an agile versus a waterfall approach. This can lead to problems when those organisations have to cooperate.
- Lack of communication with stakeholders, users and engineers.
- Lack of risk management and therefore not taking actions to mitigate risks.
- An imbalance between the budget and the expectations. For example the budget has been reduced during the acquisition phase, but the expected results have not changed.

For most of the other issues in the group OL the solution of these issues also includes better project management. Therefore the other issues in this group can also be seen as examples of problems that project management should address.

A.6.1.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since improper project management may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of "organizational and legal constraints" depicted in the extended LCIM model described in the corresponding section of this report.

A.6.1.4 Connection to DSEEP Steps and Artifacts

Proper project management is of importance in all steps of the DSEEP. Activity 1.3 ("Conduct Initial Planning") is an important activity when it comes to project management.

A.6.1.5 Possible Solution Approaches

The solution to these problems is a structured project management process. The DSEEP provides guidelines for the simulation specific aspects that should be included in such a project management process.

A.6.1.6 Existing Implementations and Their Information Products

There are several (competing) international organisations or communities related to project management. The most prominent ones probably are the IPMA (International Project Management Organisation) and PMI

(Project Management Institute). There communities/organisations provide trainings, background knowledge and certifications and thereby have established "international standards" for project management. An example hereof is the PMBOK book published by the PMI and assigned a standard by ANSI (ANSI/PMI 99-001-2008) and IEEE (IEEE 1490-2011).

A.6.2 OL-02 Missing or Limited Policy, Coordination and Cooperation

A.6.2.1 Problem Definition

Missing or limited coordination and cooperation of participants in a simulation environment engineering process may lead to serious problems. This is especially the case within multi-national cooperation programs or across different (acquisition) programs.

A.6.2.2 Extended Problem Description

Where issue OL-01 focussed on organisational issues within a project, this issue more address organisational issues arising between different projects or over an entire organisation.

Some examples of this issue are:

- During acquisition of a simulation system there might be a discrepancy between the best solution for the project leader and the best solution for the overall organisation. For example, in the long run buying simulation models that can be used by the entire organisation is better, but the project leader has a limited budget and therefore it might be easier for him to buy those models with a more restrictive license that only allows usage in his system.
- Within a multi-organizational production programme there can be issues resulting from different amount of effort put in by the participating members or different levels of quality. For example the geographical data produced by the Multi-national Geospatial Co-Production Program (MGCP) shows a variation in amount of features over the different cells.
- The Task Groups of the NATO Modelling and Simulation Group (NMSG) are a prime example of cooperation in the field of research questions. Within these groups different Nations work together on solutions for common problems. However the fact that different NMSG groups are not always aware of the activities of others shows that more coordination and cooperation would be possible. A solution for this would be to have some central coordination office performing the distribution of research findings.
- Another example of this issue is the adoption of SEDRIS. It has been developed to solve a concrete problem with the exchange of synthetic environment data between different simulation systems. But the usage of SEDRIS is not yet widespread in the simulation community; this is partly due to the lack of a policy that ensures that tools and simulation systems support this standard.
- A solution to share research findings better is coordination, but if there are too many organisations doing this coordination or too many databases to search for research findings this can become a problem as well.
- Within a multi-national context not all Nations might give the same importance or attention to simulation interoperability issues. This lack of coordination means that it is not possible to come up with the same solution for everybody.

A.6.2.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since missing or limited policy, coordination, and cooperation may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of “organizational and legal constraints” depicted in the extended LCIM model described in the corresponding section of this report.

A.6.2.4 Connection to DSEEP Steps and Artifacts

This issue addresses problems with policy and coordination between different programs; this is at the higher level than the DSEEP. Therefore a link to the DSEEP is difficult to make.

However when issues with coordination and cooperation within the development of a simulation environment are addressed these should be identified as constraints or risks, especially in Activities 1.1 and 1.2 of the DSEEP:

- Identify available resources and known development and execution constraints; and
- Assess feasibility and risk, and incorporate into objectives statement.

A.6.3 OL-03 Cultural Aspects

A.6.3.1 Problem Definition

When developing a simulation environment with partners from different companies and/or different countries, cultural aspects can play a big role.

A.6.3.2 Extended Problem Description

Cultural aspects can range from the obvious to the very subtle, but here are a few examples:

- Different public holidays in different countries can have a schedule impact on integration testing: a 5 day week can turn into a 3 day week.
- Similarly, different attitudes can affect willingness to work overtime, especially on public holidays.
- Different attitudes to authority can affect the perceived relative priority of stakeholder demonstrations compared to making progress against the project schedule.
- Language problems can occur because persons from different countries can have a different understanding of the meaning of words. Sometimes the difference is quite subtle. For example, the word “propose” in English means a suggested course of action perhaps open to discussion, but in French it means the action that will be undertaken with the decision already having been made. Words that are the same in different languages actually make the problem worse, because one speaker can think he or she has understood the other when there is a fundamental difference in what they think they have both agreed to. The fact that people do not speak their mother tongue when communicating can enlarge this problem.
- Stereotypes and clichés exist about people from different Nations. These stereotypes and clichés might influence the cooperation between people from different countries.

A.6.3.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since complications regarding cultural aspects may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of “organizational and legal constraints” depicted in the extended LCIM model described in Section 2.2.

A.6.3.4 Connection to DSEEP Steps and Artifacts

In DSEEP Step 1 this issue should be identified as constraints or risks, especially in Activities 1.1 and 1.2:

- Identify available resources and known development and execution constraints; and
- Assess feasibility and risk, and incorporate into objectives statement.

A.6.3.5 Possible Solution Approaches

- Project management should be aware of the different cultural aspects and take actions to try to minimize their impact.
- The project members should be open-minded to each other.
- Special training for project members to learn them how to deal with these cultural aspects before working abroad or being engaged in an intercultural project.

A.6.4 OL-04 Lack of Required Personnel

A.6.4.1 Problem Definition

Simulation environments are complex systems, therefore the development and deployment of such an environment requires personnel with many different areas of expertise. Lack of required personnel can be a serious problem.

A.6.4.2 Extended Problem Description

Some examples of cases where personnel with special expertise are required are:

- Personnel with experience in VV&A to perform the validation and accreditation of the simulation environment;
- Personnel with experience in specific tools used in the simulation environment, for example creating the scenario in the CGF tool or constructing the synthetic environment in the database generation tools; and
- Personnel with experience in interoperability standards, like HLA or DIS, to set up the connection between different systems in the simulation environment.

It is not uncommon that the people with the required expertise are needed for many projects, so these people might not always be available. A good resource planning would help to reduce this problem, ensuring that each project has the required expertise and a good mix between senior and junior personnel.

A.6.4.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since lack of required personal may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of "organizational and legal constraints" depicted in the extended LCIM model (see Section 2.2).

A.6.4.4 Connection to DSEEP Steps and Artifacts

In DSEEP Step 1 this issue should be identified as constraints or risks, especially in Activities 1.1 and 1.2:

- Identify available resources and known development and execution constraints; and
- Assess feasibility and risk, and incorporate into objectives statement.

A.6.5 OL-05 Selection of Incompatible Member Applications

A.6.5.1 Problem Definition

When member applications are selected that are not compatible with each other, this will influence the final utility of the simulation environment for its goal. This issue is related with the organisational part of the selection, not the technical consequences of the incompatible member applications.

A.6.5.2 Extended Problem Description

In many cases existing member applications have to be reused or political reasons require the usage of member applications from different organisations or countries. When these member applications are not compatible with each other (e.g., different time-management models, or a mix of different CGFs), this will influence the utility of the simulation environment for its goal. The process of selection is an organisational issue, but it can also lead to many technical issues. See the issue group Federation Development for more specific technical issues that can occur with incompatible member applications.

A.6.5.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since the selection of incompatible member applications may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of “organizational and legal constraints” depicted in the extended LCIM model (see Section 2.2).

A.6.5.4 Connection to DSEEP Steps and Artifacts

DSEEP Activity 3.1 (“Select member applications”) is where the member applications are selected. An important output of this step is also the list of requirement gaps, which should address incompatible member applications.

A.6.5.5 Possible Solution Approaches

From technical point of view the solution would be to select a different member application. But when that is not possible due to political or organisational reasons the best solution is to make all participants aware of the limitations from the chosen member applications and how that will affect the final usability of the simulation environment.

This issue deals with the organisational aspect of selecting incompatible member applications, in the issue group Federation Development (FD) solution approaches are provided for the technical issues that can result from incompatible member applications.

A.6.6 OL-06 Legal or Political Restrictions Apply

A.6.6.1 Problem Definition

Legal or political restrictions can apply on sharing simulation components or data with other participants of the simulation exercise. This issue can be a show stopper for certain solution approach to interoperability problems.

A.6.6.2 Extended Problem Description

Below some examples are given how this issue can demonstrate itself for the selection and use of the synthetic environment. This problem can also address itself with other simulation components of course:

- The Intellectual Property Rights (IPR) of the synthetic environment or of the data used within the synthetic environment restrict the sharing and reuse of the environment outside of the organisation or country that acquired it. For example a synthetic environment has a license that only allows it usage at one site or in one specific simulation system.
- The choice of which synthetic environment to use in a simulation exercise can also be influenced by political constraints. For a Nation to simulate military operations using a synthetic environment of another real-world Nation can result in political issues and thereby restrict which synthetic environment can be used.
- Different versions of a piece of software might be available. For example of the OneSAF CGF different versions are available with different levels of functionality. Not every users is allowed to use same version.

A.6.6.3 Connection to LCIM Level

This issue cannot be related to a single level of the original LCIM model, since legal or political restrictions may lead to subsequent issues regarding any of the LCIM levels. It is related to the frame of "organizational and legal constraints" depicted in the extended LCIM model (see Section 2.2).

A.6.6.4 Connection to DSEEP Steps and Artifacts

In DSEEP Step 1 these issues should be identified as constraints or risks, especially in Activities 1.1 and 1.2:

- Identify available resources and known development and execution constraints; and
- Assess feasibility and risk, and incorporate into objectives statement.

A.6.6.5 Possible Solution Approaches

- When commercially available data is used for the construction of the simulation component there are in general fewer issues with sharing the resulting work. If other parties want to use the synthetic environment they will need to buy a license for the used commercial data. However for mission specific simulations, the usage of military or intelligence data is sometimes required.
- When acquiring data for simulation it would be useful to acquire it with a "wide" license. In this context wide means that the license does not restrict the data to be used only on the specific system it was acquired for, but would allow the usage of the data within the entire Defence organisation or even within NATO.
- Specifically for synthetic environments, NATO MSG-071 developed a dataset of a fictitious continent in the Atlantic Ocean (called "Missionland"). One of the objectives of this dataset is that it can be freely shared between NATO and PfP countries. The fact that it is a fictitious continent also reduces the political constraints on choosing Missionland as the area for a simulation exercise.

A.6.6.6 Example of Legal Issues

In 2008, the German Armed Forces finished a project to develop a special middleware named PSI-SA (Primary Simulation Interface for Simulation Applications). This middleware is capable of coupling different HLA-based simulation systems regardless of the manufacturer of the underlying RTI. This is achieved by decoupling the simulation model from the transport layer of the RTI while so-called "sockets" provide the exchange of information from and to the RTI (see Figure A-10 for an example).

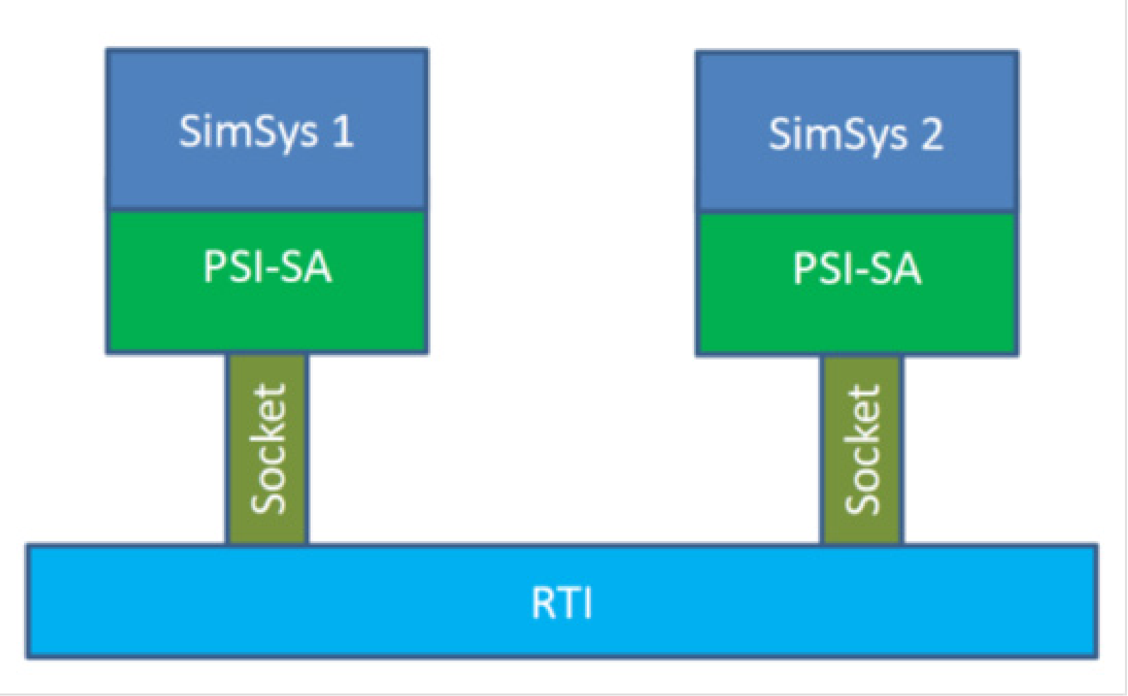


Figure A-10: PSI-SA Principles.

PSI-SA is implemented in the simulation systems and communicates with the HLA-RTI via a socket. With PSI-SA the simulation system is independent of the type of RTI, it is then possible to change the RTI without changing the simulation system – just by replacing the socket.

PSI-SA is available for free to interested users by signing-in on a special website (www.sim-infra.de). By signing-in the user accepts the PSI-SA license agreement which includes, for example, the agreement to report back any changes made on PSI-SA itself.

In 2010, the Johns Hopkins University (JHU) was interested in using PSI-SA for their simulation experiments. As an organization closely affiliated with the US Navy, the JHU planned simulation experiments with classified data. It was not possible for them to accept the PSI-SA license agreement as-is because any changes made could have been traced back to classified data and *would violate US laws*. Therefore, they asked for changes on the PSI-SA license agreement.

First, the German side hesitated to accept changes to the license agreement. The main point was that Germany would, for no apparent benefit, deliver services to another country. This *would violate German budget law*. At the end the German side agreed that the indirect benefits (e.g., improved brand awareness) by providing PSI-SA would outweigh the risk of deliver PSI-SA for no benefit.

A.7 ISSUE GROUP “SCENARIO” (SC)

A.7.1 SC-01 Missing Authoritative Operational Scenarios

A.7.1.1 Problem Definition

Authoritative operational scenarios are required to develop a simulation environment which fulfils the need of the (military) user defining the objectives of the simulation environment. Without authoritative operational scenarios, the results produced by the simulation environment may be useless to the user as they do not answer the question(s) originally posed.

A.7.1.2 Extended Problem Description

Authoritative operational scenarios describe a set of situations (e.g., homeland defence, stabilization operation in northern Africa) which have to be investigated using a simulation environment. Besides defining the objectives of the simulation environment, the user has also to provide such authoritative operational scenarios. These are described in military terms and are not specifically developed for the use in a (distributed) simulation environment. Their importance should not be underestimated: they are the authoritative source of information for all subsequent phases of the simulation environment engineering process, especially for the development of the conceptual model.

If authoritative operational scenarios are missing, this may lead to (interoperability) problems due to misunderstandings when talking about the objectives of the simulation environment. As a result, the conceptual model (and the subsequently developed simulation environment) may not reflect what the user originally wanted. In other words, the simulation environment does not answer the question(s) originally posed, but maybe something different. Therefore, missing authoritative operational scenarios lead to the wrong simulation results.

Authoritative operational scenarios are necessary to prevent this problem, however they are not sufficient. Additionally, the following information is required:

1) Complete and consistent operational scenario descriptions, ideally in a standardized format; and
2) Organizational measures to assure the transition from authoritative operational scenarios to a common problem space and common congruent conceptual model. This activity requires interaction of military SMEs and M&S experts and should be accompanied by a suitable V&V-process.

Missing authoritative operational scenarios can significantly impact the interoperability of simulation systems. As a matter of fact, this is a fundamental reason for interoperability problems of presently existing simulation systems.

A.7.1.3 Related Work

- FEAT: Not directly represented in FEAT schema. FEAT provides only a single “fedagree:scenario”-element which is (implicitly) meant to be used for describing the executable scenario.

A.7.1.4 Connection to LCIM Level

This issue is connected to Level 3 (Semantics), and above.

Authoritative operational scenarios contain definitions of basic terms used by the military user. These definitions help to achieve a common understanding of semantics. They are mandatory to avoid misunderstandings of the involved parties. Furthermore, they serve to define the purpose, scope, context and required features of a simulation environment. Thus they impact pragmatic, dynamic and conceptual interoperability.

A.7.1.5 Connection to DSEEP Steps and Artifacts

Authoritative operational scenarios have to be defined by the user at the very beginning of a simulation environment engineering process. Therefore, this issue is connected with Step 1 of the DSEEP:

> *"Where appropriate, authoritative sources for descriptions of major entities and their capabilities, behavior, and relationships should be identified prior to scenario construction."* [DSEEP, p. 14]

Especially Activities 1.1 "Identify user/sponsor needs" and 1.2 "Develop objectives" are affected by this issue.

A.7.1.6 Possible Solution Approaches

The main task is to make sure that every simulation environment engineering process (like DSEEP or FEDEP) requires the user to develop a thorough definition of objectives as well as a specification of authoritative operational scenarios, ideally in a structured (potentially standardized) way. Ensuring development (or provision) of appropriate scenarios is within the responsibilities of the project lead and the V&V-authority.

The problem of missing authoritative operational scenarios cannot be solved by technical means. This is an organizational problem and has to be tackled by sufficient project management methods and organizational regulations (e.g., binding procedures and guidelines).

A.7.1.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.7.1.8 Recommended Information Products

No recommendations made by MSG-086.

A.7.1.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.7.2 SC-02 Incomplete or Inconsistent Operational Scenario Description

A.7.2.1 Problem Definition

To reach the goal of valid and trustworthy results, objectives and requirements of a simulation environment have to be specified precisely at the very beginning of the simulation engineering process. Besides defining the objectives of a simulation environment, a detailed description of authoritative operational scenarios as basis for answering the question "what has to be represented in the simulation environment?" is also necessary. Incomplete or inconsistent descriptions of operational scenarios may be interpreted differently by the participants resulting in a simulation which is different from the intended purpose.

A.7.2.2 Extended Problem Description

Regarding the necessary descriptions of operational scenarios for a simulation environment, two basic problems are identified:

1) Incomplete description of the operational scenario(s): First of all, it is important that an operational scenario is developed and (depending on size of the simulation environment and time frame) written down. This operational scenario has to be provided by the military user who defines the objectives of the simulation environment. Subject-Matter Experts (SMEs) may help assist the military user during this phase. Therefore, the operational scenario is usually described in terms with which the user/SME is familiar, i.e., military terms. The operational scenario provided by the user does not have to mention (distributed) simulation at all. On the contrary, the user should focus on describing

the problem as well as the operational scenario(s) without having to worry about how this problem will be investigated and analysed. It is important that the description of the operational scenario is complete. *Completeness* means that the description has to contain all necessary information to enable persons in the subsequent process (especially development of the conceptual model) to use the operational scenario meaningfully and extract all information required for their activities.

2) Consistency/Usefulness/Understandability of an operational scenario: Assuming a complete description of a operational scenario is given (see No. 1), there may still be potential for problems:
    a) *Consistency* refers to the "internal" validity of the description (e.g., no unit belongs to more than one force side, initial positions of units are not overlapping); and
    b) *Usefulness/Understandability* refers to the problem that the description has to be written and structured in a way that it is easily accessible by future readers.

Manifold problems may arise if operational scenarios are described incompletely or inconsistently:

- An incomplete operational scenario description may lead to the development of a "wrong" conceptual model. The simulation environment derived consistently (under the same project management) from this conceptual model may be perfectly interoperable, yet the original objectives specified by the user will not be answered (due to wrong understanding of the operational scenario). In this case, incomplete or inconsistent descriptions of operational scenarios are more a V&V-issue, and less an interoperability problem. Under such conditions, this issue is not considered an interoperability problem. Nevertheless, this issue is of utmost importance to ensure that the right simulation environment is developed and the right operational scenario is simulated. Any V&V-activities defined and carried out have to pay attention to this issue.
- Incomplete and/or inconsistent operational scenarios may lead to the development of conceptual models of the systems supposed to be federated which are different in critical parts with respect to simulation interoperability (characteristics of entities, fidelity of representation of entities). The reason for this are for example different federates developed by different companies in different Nations (under distributed and different project managements).

Therefore, the problems due to incompletely or inconsistently described operational scenarios are similar to those described in issue SC-01.

A.7.2.3 Related Work

- FEAT: Not directly represented in FEAT schema. FEAT provides only a single "fedagree:scenario"-element which is (implicitly) meant to be used for describing the executable scenario.

A.7.2.4 Connection to LCIM Level

Level 3 (Semantic Interoperability) and above.

A scenario description has nothing to do with technical or syntactic issues. It is primarily concerned with pragmatic issues (what has to be simulated) and contains also definitions of terms (semantics). Therefore, providing a complete and consistent operational scenario description is connected to Levels 3 and above of the LCIM.

A.7.2.5 Connection to DSEEP Steps and Artifacts

The operational scenario has to be described at the very beginning of the whole process, jointly with defining the objectives. Therefore Step 1 is affected by this issue, especially Activities 1.1 "Identify user/sponsor needs" and 1.2 "Develop objectives".

A.7.2.6 Possible Solution Approaches

An obvious approach to prevent common problems is to define templates for describing operational scenarios. This helps to ensure that all important aspects of an operational scenario are documented. Furthermore, once the user is familiar with the template it is easy to read (and understand!) operational scenario descriptions developed by someone else. Despite the use of templates to ensure completeness of an operational scenario description, it has to be ensured that:

1) Such a description is developed at all and that; and
2) All participants of the simulation engineering process are aware of the description of the operational scenario and use this description as single source in later steps of the engineering process. Of course, if information is missing the description should be updated by the user.

Both aspects are within the responsibility of the project lead and require organizational measures (e.g., binding guidelines, standardized templates).

A.7.2.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.7.2.8 Recommended Information Products

Guideline on Scenario Development for (Distributed) Simulation Environments.

A.7.2.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.7.3 SC-03 Missing Formal Scenario Specification

A.7.3.1 Problem Definition

A missing formal scenario specification (regardless for which type of scenario, operational, conceptual or executable) may lead to interoperability problems as different persons, simulation systems or even simulation environments may interpret a given scenario differently.

A.7.3.2 Extended Problem Description

The interchange of scenarios may lead to problems, if two persons (or simulation systems) have a different understanding of the term scenario and the content of a scenario description. In order to be precise, it is important to understand that one scenario may be represented in many ways (see Table A-2).

All types of scenarios (operational, conceptual or executable) may be represented in the four different ways depicted in Table A-3.

Table A-3: Representation Types of a Scenario (Increasing Maturity from Left to Right).

	Undocumented Scenario	Scenario Documentation	Structured Scenario Documentation	Formal Scenario Specification
Representation of Scenario	Thoughts and ideas within the mind of the military user/ SME	Free text documentation	Free text documentation structured according to some guidelines or templates	Formal specification of a scenario
Available Standard(s)	–	–	Documentation guidelines according to VEVA	MSDL, JTDS
Typical File Formats	–	Word, PowerPoint	Word	MSDL XML Schema, JTDS XML Schema

First it has to be assured that a *Scenario documentation* is created at all. Secondly it has to be assured that this documentation is complete and consistent. By using documentation guidelines or templates, SC-02 tries to make sure that a *Structured scenario description* is developed. This issue treats the case of missing *Formal scenario specifications*. A formal specification of a scenario has at least two benefits:

1) It is less ambiguous than a textual description; and
2) The formal specification can be distributed easily as a file and eventually be interpreted automatically by the simulation systems.

A missing formal specification for scenarios leads to potential problems which are very similar to those described in issue SC-02 regarding operational scenarios: The scenario may be incomplete or inconsistent. This can lead to erroneous and/or inconsistent configuration of the simulation systems which in turn can lead to erroneous simulation results and interoperability problems.

Examples:

- A missing formal specification of the executable scenario may result in the investigation and analysis of a scenario different from the one the user/SME was thinking of. Example: different entities at different locations with different relations and orders. (Theoretically in such a case, even with a “wrong” scenario the simulation systems may be perfectly interoperable, if their conceptual models are congruent).
- A missing formal specification of the executable scenario may provoke differences in the configuration and initialization of different federates of a simulation environment leading to interoperability problems (unfair fight conditions) on the pragmatic and dynamic level (Level 4 and 5). Examples: two federates responsible for the dynamics of the same entity, or a certain entity is not represented at all, or differences in ammunition and/or fuel inventory in a certain type of a tank.

A.7.3.3 Related Work

- [MSG-052], Section B.11.
- FEAT: Partially represented in FEAT schema by the “fedagree:scenario”-element. Unfortunately, the “fedagree:scenario”-element provides just a reference to some external document.

A.7.3.4 Connection to LCIM Level

This issue is mainly related to Levels 2 and 3 (syntactic and semantic interoperability) of the LCIM. Potentially it could also affect Level 4 and 5 (pragmatic and dynamic level).

A formal scenario specification has to define the syntax of a scenario as well as the meaning (semantic) of the syntactic elements. Therefore, it supports interoperability improvements on these levels. Formal scenario specifications may also be helpful in preventing interoperability problems on Level 4 and 5 as unfair fight conditions may be avoided.

A.7.3.5 Connection to DSEEP Steps and Artifacts

Depending on the type of scenario (operational scenario, conceptual scenario, executable scenario) different DSEEP steps are affected:

- With regards to operational scenarios: Activities 1.1 “Identify user/sponsor needs” and 1.2 “Develop objectives” are affected by this issue;
- With regards to conceptual scenarios: Activity 2.1 “Develop” scenario” is affected; and
- With regards to executable scenarios: Activity 4.2 “Establish simulation environment agreements” is affected.

A.7.3.6 Possible Solution Approaches

The important point regarding scenario(s) is that they are completely and consistently specified. To improve on the current situation, SC-02 proposes to specify templates to ensure that a scenario contains all important information.

Regarding the configuration and initialization of simulation systems the current status is that each simulation system uses different configuration file formats. The configuration file formats are tailored for the specific system.

To improve on this situation, it would be helpful to agree on a common formal scenario specification. If such a specification is available, scenarios should be developed according to this specification. Based on such a formal scenario specification, a dedicated file format may be derived which is used and understood by many simulation systems. The need to transform a textually described executable scenario into many different configuration files for all participating simulation systems would be obsolete. Thus the development process would be simplified and a potential source of errors would be eliminated.

A.7.3.7 Existing Implementations and Their Information Products

Existing formal scenario specifications are:

- Military Scenario Description Language (MSDL), provided by SISO;
- JTDS (Joint Training Data Services) 3.0 XML Schema, an executable scenario format used by US Joint Warfighting Center; and
- Turkey uses a specific executable scenario format in some simulation system.

Each of these formal scenario specifications defines also a file format (usually using XML Schema). A formal scenario specification is limited to a specific application domain (and conceptual model). Therefore, it is unreasonable to expect having only one formal scenario specification (and only one file format) in the future which is applicable to all simulation systems respectively all types of scenarios.

A.7.3.8 Recommended Information Products

No recommendations made by MSG-086.

A.7.3.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.7.4 SC-04 Use of Different Formats for Executable Scenarios

A.7.4.1 Problem Definition

Usually different simulation systems need executable scenario configurations in different file formats. The use of different file formats may introduce many problems (e.g., due to different resolution and fidelity supported by the different formats).

A.7.4.2 Extended Problem Description

On the basis of the underlying conceptual model an *executable scenario* specifies all details needed to actually configure and initialize the simulation systems of a simulation environment for execution. Issue SC-04 is a special case of issue “SC-03 Missing formal scenario specification”. Whereas SC-03 treats the case of problems arising due to a missing formal scenario specification and file format, this issue treats the case of problems due to the use of different file formats for executable scenarios.

Usually each member application (federate) of a simulation environment has to be configured using a specific set of configuration files and settings. Currently almost all member applications use a different format of configuration files. Two major problems are identified in this context:

1) Problems arise if the executable scenario cannot be made available in a consistent manner to all partaking member applications, i.e., if the configuration files used for the member applications are substantially different:
    a) Example: Transforming an executable scenario into two configuration files using format A and B may lead to problems if format A supports a higher fidelity than format B. For example, Format A allows the specification of the orientation of the turret of a tank only in four discrete values (north, east, south, west) whereas format B allows to specify the orientation in degrees. In this case an initial orientation of 45 degrees cannot be expressed in format A.
2) Problems arise if the transformation of an executable scenario into configuration files for different systems is erroneous. Currently, the development of configuration files is often done manually. The more configuration files have to be created, the higher is the risk of (unwillingly) introducing errors and creating inconsistent configuration files.

Problem 2 may be addressed by rigorous quality management and sufficient verification activities (which ensure correct transformation of an executable scenario into a configuration file). Therefore, problem 2 is not considered an interoperability issue, but a V&V-issue.

Problem 1 may cause interoperability problems and unfair fight, because of different configuration of the simulation systems due to differences in the file formats.

The origin of the problem is not always the difference in file formats, but may also be a difference in the underlying conceptual models and capabilities of the simulation systems. The file formats used by the simulation systems reflect their conceptual model (e.g., the precision of the simulation system regarding position information). In this case, the different file formats are not the source of an interoperability problem, but make the underlying interoperability problems visible.

A.7.4.3 Related Work

- [MSG-052], Section B.11.
- FEAT: Partially represented in FEAT schema by the “fedagree:scenario”-element. Unfortunately, the “fedagree:scenario”-element provides just a reference to some external document.

A.7.4.4 Connection to LCIM Level

First of all, different file formats imply a different syntax (Level 2). If the formal scenario specifications of two file formats are substantially different, also the semantics (Level 3) may differ a lot. A formal scenario specification is defined for a specific purpose and application domain (e.g., for a training simulation on single entity level, or a decision support simulation on company level). Each application domain imposes different requirements on the simulation system itself as well as on the formal scenario specification and file format used by this system. As the application domain describes the purpose of a simulation system, problems due to different file formats may also be related to the pragmatic level.

A.7.4.5 Connection to DSEEP Steps and Artifacts

This issue is related to Step 4, Activity 4.2 (“Establish simulation environment agreements”) of the DSEEP. In this step, the executable scenario has to be developed and configuration files for the simulation systems have to be derived: “Once all authoritative data sources that will be used in support of the simulation environment have been identified, the actual data stores are used to transition the functional description of the scenario (developed in Step 2; see Figure 4) to an executable scenario instance (or set of instances).” [DSEEP, p. 26]

A.7.4.6 Possible Solution Approaches

Referring to the first problem area mentioned above (different underlying conceptual models), no easy solution is possible. Instead, it has to be ensured that only simulation systems are coupled in a simulation environment which are compatible on a pragmatic level, i.e., with respect to their application domains and conceptual models.

A possible solution for the second problem (different file formats) would obviously be to make sure that all member applications use and understand the same file format. This is possibly impossible to achieve as different simulation systems have different requirements (due to different application domains and different conceptual models).

Taking the specific requirements of the various simulation systems of a simulation environment into account, it has to be ensured that the differences in the file formats are negligible for the specific executable scenario to be represented. The decision whether the differences are negligible can only be made for a specific executable scenario! In order to simplify and accelerate the scenario-specific evaluation, the available file formats can be compared a priori to identify similarities as well as critical aspects. During the scenario-specific evaluation, the main focus may then be on the critical aspects.

A.7.4.7 Existing Implementations and Their Information Products

See SC-03.

A.7.4.8 Recommended Information Products

No recommendations made by MSG-086.

A.7.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.7.5 SC-05 Use of Different Doctrines and ROEs

A.7.5.1 Problem Definition

The problem addressed by this issue is that different Rules Of Engagement (ROE) are used in a simulation environment (either by simulation systems (virtual, constructive) or real systems). Depending on the actual scenario this may lead to Unfair Fight.

A.7.5.2 Extended Problem Description

If systems participating in a simulation environment use different Rules Of Engagement (ROE), unfair fight may result and, in general, the validity and credibility of the results will suffer. Therefore, this issue addresses primarily the quality of a distributed simulation environment and is not per se an interoperability problem.

It could be an interoperability problem though, if the conceptual models of different simulation systems are substantially different with regards to the ROE. In this case, it may not be possible to achieve interoperability on a pragmatic (or dynamic/conceptual) level. First of all, the ROE have to be specified by the military user who is planning the simulation environment and specifying the objectives as well as the authoritative operational scenarios. Depending on the argumentation, ROE may be part of the conceptual model of the simulation environment or of the authoritative operational scenario(s):

- ROE are part of the conceptual model, because ROE define (default) behaviour of entities; and
- ROE are part of the operational scenario(s), because ROE depend highly on the operational scenario(s) which is/are investigated and may change from case to case.

Within a simulation environment, different simulation systems may use different ROE (e.g., for friendly and hostile units). This is perfectly okay and not a problem at all. Problems arise, if different simulation systems use different ROE although they are expected to use the same ROE. In this case, different ROEs in different simulation systems are a result of either:

1) Unspecified ROE;
2) Different capabilities of participating systems; or
3) Incorrect interpretation of ROE by some systems.

The first problem should be addressed during development of the operational scenario(s) or conceptual model. Thus, the first problem may be solved by adequate development guidelines and documentation templates, ensuring that the military user specifies the ROE.

The second problem (different capabilities of the simulation systems) cannot be solved easily; instead it has to be evaluated for each distributed simulation environment whether the differences in capabilities (in this case with regard to ROE) are a problem or not. This evaluation is highly dependent on the purpose and the objectives of the distributed simulation. Therefore, a generally valid statement whether two systems use "compatible" ROE cannot be made.

The third problem (incorrect interpretation of ROE by some systems) may be caused either by misconfiguration of a system (due to human error) or by misinterpretation of an "ROE-file" by a system. Human errors may always occur and are not a specific interoperability problem. Assuming the ROE of an operational scenario are available in form of an "ROE-file" which is distributed to all participating systems, it has still to be ensured that all systems handle and interpret the "ROE-file" in the same way. Ensuring the correct handling and interpretation of such an "ROE-file" is part of the quality management activities during development of the simulation system.

Table A-4 shows the different interpretation of ROEs by different CGFs.

Table A-4: Different Interpretation of ROE by Different Simulation Systems.

ROE Name	VR-Forces Interpretation	OneSAF Interpretation
Weapons Stop	*Not implemented*	Do not fire under any circumstance
Weapons Hold	Do not fire under any circumstance	Fire only if fired upon
Weapons Tight	Fire only if fired upon	Fire on anything positively identified as hostile
Weapons Free	Fire on detection of hostile	Fire on anything **not** positively identified as friendly

The important difference is the different interpretation of a rule like "Weapons Tight". If a scenario just contains an entry <roe = Weapons Tight> the two simulation systems will implement this in different ways resulting in different behaviours during scenario execution. Clearly a definition of what behaviour is expected needs to be specified to ensure consistency across applications.

A.7.5.3 Related Work

- [MSG-052], Section B.11.
- FEAT: Not explicitly represented in FEAT. The FEAT elements "fedagree:conceptualModel" and "fedagree:scenario" may both be used to document ROEs. Unfortunately, the FEAT provides only a reference to some external document.

A.7.5.4 Connection to LCIM Level

4 (Pragmatic Interoperability), and above.

ROE define the behaviour of entities in specific situations; therefore this issue is connected to pragmatic interoperability. Furthermore, ROE specify dynamic behaviour of entities (Level 5) and can be seen as a part of the conceptual model (Level 6).

A.7.5.5 Connection to DSEEP Steps and Artifacts

The DSEEP explicitly mentions ROE in Step 2, Activity 2.2 ("Develop conceptual model"):

> *"For instance, rules of engagement and the political environment [DIS task g)] is defined as being part of DSEEP conceptual analysis activities in Step 2." [DSEEP, p. 51]*

As ROE may also be considered to be part of the scenario instead of the conceptual model, also Activity 2.1 “Develop scenario” is affected.

The DIS standard (IEEE 1278.3) requires the specification of ROE at the beginning of the simulation engineering process (Phase 1 “Plan the exercise and develop requirements”):

> *“As a minimum, the Exercise Manager should consider the following: [...] (g) Rules of engagement and political environment” [DIS 1278.3, p. 7]*

This would correspond to DSEEP Activities 1.1 “Identify user/sponsor needs” and 1.2 “Develop objectives”.

A.7.5.6 Possible Solution Approaches

A solution approach to this issue should include two steps:

1) It has to be ensured, that the military user defining the objectives and the authoritative operational scenarios of the simulation environment also specifies the ROE to be used. This can be supported (and partially enforced) by providing documentation templates which explicitly require the specification of ROE (either as part of the authoritative operational scenario(s) or as part of the conceptual model).
2) In order:
   - To avoid human errors when configuring the participating simulation systems;
   - To allow quicker configuration of the simulation environment; and
   - To improve traceability and reusability.

It would be helpful to formalize the ROE. Similarly as for executable scenarios, the definition of an “ROE format “would be required, e.g., as XML Schema. Once the ROE are available as a file (following the ROE format), the ROE can be distributed to the simulation systems without requiring manual configuration.

A.7.5.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.7.5.8 Recommended Information Products

No recommendations made by MSG-086.

A.7.5.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.7.6 SC-06 Missing or Incomplete Definition of Application Domain

A.7.6.1 Problem Definition

A missing or incomplete definition of the application domain of a simulation environment may lead to the development of a simulation environment that does not satisfy the user’s original requirements.

A.7.6.2 Extended Problem Description

The application domain (as a specific instance of the application space) denotes the purpose for which a single simulation system or a federation (simulation environment) is developed. The application domain of a simulation system or simulation environment is the starting point for the selection and/or elaboration of suitable operational scenarios (see [ScenarioGuideline, ch. 4.2]). If the application domain is not defined at all or incompletely defined, two major problems may arise:

1) Due to missing or incomplete definition of the application domain, chances are high that unsuitable operational scenarios (and corresponding task lists) are selected or elaborated. This in turn leads to an unsuitable (or even wrong) conceptual model of the simulation environment.
2) Due to missing or incomplete definition of the application domain of the simulation environment, selection of appropriate simulation systems is difficult (if not impossible). A common consequence is the selection of simulation systems with inappropriate or incompatible application domains (e.g., training vs. decision support, or single entity level vs. company/brigade level). Such systems are most often not interoperable.

A missing or incomplete definition of the application domain of a simulation system or simulation environment may be caused by a missing standardized way of defining the application domain. One proposal for structuring the application space and defining the application domain was made by MSG-024 and uses a 5-dimensional structure (see [ScenarioGuideline, ch. 4.2.1]).

A.7.6.3 Related Work

- FEAT: partially represented by FEAT elements:
    - “fedagree:domainInfo” which is used for describing the domain. Unfortunately, the FEAT provides only a reference to some external document which shall contain information traceable to authoritative sources; and
    - “fedagree:objectivesAndRequirements” which shall contain: “Measurable objectives for the federation; requirements; constraints, preferences, assumptions and limitations.”

A.7.6.4 Connection to LCIM Level

4-6 (Pragmatic, Dynamic, Conceptual).

The application domain of a simulation system describes for which purpose a simulation system is developed. Thus, it describes pragmatic aspects of the simulation system and reflects the underlying conceptual model of the simulation system.

A.7.6.5 Connection to DSEEP Steps and Artifacts

- Step 2, Activity 2.2 “Develop conceptual model”. The conceptual model of the intended simulation environment has to be matched with the conceptual models of the participating member applications (federates). Therefore, the development of a conceptual model for the simulation environment is a necessary prerequisite for such an evaluation.
- Step 3, Activity 3.1 “Select member applications”. The selection of member applications has to take the application domains of the possible member applications into account as well as the objectives and the conceptual model of the simulation environment.

A.7.6.6 Possible Solution Approaches

In order to allow a standardized definition and quick comparison of application domains of simulation systems and simulation environments, it would be helpful to describe the application domain according to a “standardized” schema. Within MSG-086, the 5-dimensional structure for the simulation application space as described in the Final Report of MSG-024 “M&S Support to Non-Article 5 Operations” (Chapter 3) is used. A broad adoption and use of this terminology within the military (NATO) M&S-community would be helpful.

As the application space as defined by MSG-024 is a high-level proposal, the application space structure (and possibly sub-structures) should furthermore be described in more detail. In context of engineering a specific simulation environment, the following aspects should be taken into account:

1) It has to be ensured that the application domain of the intended simulation environment is defined;
2) Based on the definition of the application domain and the objectives, suitable authoritative operational scenarios have to be selected and/or elaborated; and
3) The selection of member applications (Step 3.1 of DSEEP) of the simulation environment has to take the application domain of the simulation environment into account, and only simulation systems with appropriate application domains should be selected (this evaluation is highly dependent of the actual objectives!). This requires that the application domains and the conceptual models of potential member applications (simulation systems) of the simulation environment are documented thoroughly.

A.7.6.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.7.6.8 Recommended Information Products

DSEEP: Simulation environment requirements.

A.7.6.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.8 ISSUE GROUP “SYNTHETIC ENVIRONMENT” (SN)

A.8.1 SN-01 Synthetic Environment Data is Not Correlated

A.8.1.1 Problem Definition

Synthetic environments do not correlate correctly when different data has been used to construct the synthetic environment. The different data sources can differ in accuracy, resolution or amount of detail. Ignoring the spatial reference system of the data or converting incorrectly between different spatial reference systems can also cause correlation issues.

A.8.1.2 Extended Problem Description

Below some examples are given of this problem:

- One terrain database uses the global VMAP vector features, while another terrain database uses high resolution vector data defining the location of roads. These two terrain databases will then not give a correlated interpretation of the environment.
- A terrain databases uses high resolution satellite imagery as a skin on the terrain, but roads data from the global VMAP vector data will not correlate with this imagery because it defines the roads at much lower resolution.
- In two simulation systems a different radar cross-section value is stored for the same entity type. As a result of this the detection of the entities will differ between the two systems.

- Not all systems work with the same earth model and spatial reference system. Disregarding these differences or using the wrong conversion between such systems results in uncorrelated synthetic environments.

A.8.1.3 Related Work

Federation Engineering Agreements Template (FEAT) has a number of elements which could store information related to the synthetic environment and thereby help reduce correlation issues:

- fedagree:terrain provides information like the area of the synthetic environment, formats used, resolution, conversion and transformations. The GML format is used to specify the bounding box and conversions/transformations. Part of the information is provided as reference to external documents.
- fedagree:environment provides information related to SNE and weather databases, including algorithms and temporal updates. Most of the information is provided as reference to external documents.
- fedagree:coordinateSystems provides information about authoritative coordinate system representations. GML is used to represent this information.

A.8.1.4 Connection to LCIM Level

When different data is used it is Level 3 (semantic). When dealing with equal data sources, the interoperability issues arise from different interpretation of the data and thus Level 4 (pragmatic).

A.8.1.5 Connection to DSEEP Steps and Artifacts

In Step 2 “Perform conceptual analysis” the initial choices for the synthetic environment are made. In all 3 activities there are tasks to perform:

- Define geographic areas of interest (2.1 Develop scenario);
- Using authoritative domain specific documentation, identify the entities, behaviours, and events that need to be represented in the scenario(s) (2.1 Develop scenario);
- Identify and describe all relevant entities within the domain of interest (2.2 Develop conceptual model); and
- Define requirements for natural environment representation (2.3 Develop simulation environment requirements).

In Step 3 “Design simulation environment” it is identified if existing synthetic environments can be re-used and the design of the synthetic environment is laid down. The tasks are performed in Activities 3.1 and 3.2:

- Search existing repositories for existing member applications and/or algorithms and code to be used by a member application (3.1 Select member applications);
- Identify candidate member applications (3.1 Select member applications); and
- Develop design of supporting databases (3.2 Design simulation environment).

In Step 4 “Develop simulation environment” choices are made for which data source to use and the synthetic environment is actually implemented. The tasks are performed in Activities 4.2:

- Decide which databases and algorithms must be common or consistent (4.2 Establish simulation environment agreements); and

- Identify authoritative data sources for both member application and simulation environment databases (4.2 Establish simulation environment agreements).

A.8.1.6 Possible Solution Approaches

The following solution approaches address correlation issues:

- The obvious solution to solve correlation issues is to make sure that all simulation components use the same data sources. In that case interoperability issues can still arise, but these are then caused by the different fidelity used by the systems, as addressed in SN-02. Below there are a number of solutions that can assist in ensuring that the same data sources are used through the entire simulation environment:
    - Providing the needed data as a service, for example a weather server or a database server. With this solution there is one central place that provides the information to all participants that need it. An issue with such services is the lack of standards for them.
    - There is a trend to use GIS data directly in simulation applications. When this is combined with using OGC standards for data distribution, like WMS, WFS or WCS, it enables simulation components to access the underlying geographical data of the synthetic environment from one central place.
    - The Common Database (CDB) standard from Presagis aims to provide data for the synthetic environment in a structured way so that different simulation applications can use it [CDB].
    - Missionland provides a correlated dataset to construct synthetic environments from [NMSG-071].
- Different spatial referencing is not a problem, as long as the conversion is done correctly. Therefore a solution to prevent problems with incorrect conversion between different spatial reference systems is to standardize them. This is what the SEDRIS Spatial Reference Model (SRM) does [SRM].

A.8.1.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.8.1.8 Recommended Information Products

The reduce correlation issues with the synthetic environment it is recommended that information products provide information about the data sources that will be used for the construction of the synthetic environment. This will help different participants in the simulation exercise to use the same data.

A.8.1.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.8.2 SN-02 Different Levels of Fidelity are Used for Synthetic Environment

A.8.2.1 Problem Definition

Different simulation systems simulate the synthetic environment and the interactions going on within this environment at different levels of fidelity. This results in correlation issues when combining these systems in one simulation exercise. These different levels of fidelity can be caused by different technical capabilities of the simulation systems, by different requirements from the user of the system or by different data requirements from the simulation components.

A.8.2.2 Extended Problem Description

Some examples of this problem are given below:

- A legacy simulation system with an older image generator might not be capable to render a terrain with the amount of detail that a modern image generator can handle. As a result this simulation system needs to use a simplified version of the environment.
- A land-based simulation for infantry will require that the houses are modelled including their interior and that these objects can be entered. An aircraft simulator will typically only require the rough shape of the building to be represented.
- In many simulation systems weather effects, like precipitation or clouds, are mainly visual effects. While in other simulation systems a detailed sensor simulation might take into account the physical effects of the weather on the detection. This requires a different fidelity to describe the environment.
- Different data is required by a visual system and a CGF system. The first requires data to generate a visual rendering of the environment, while the latter requires data to make decision about the behaviour of entities. When the data for these different components comes from different sources the correlation is often not guaranteed.
- Not all Computer Generated Forces (CGF) systems use the same fidelity to simulate the entities. For example one might model the radar detection using a probabilistic model, while another CGF actually simulates the beam of the radar. These differences can influence the observability on the radar warning receiver.
- Infrared sensor information about moving models can be store in different ways. Some simulation systems replace the model texture with version that represents the infrared image. Other simulation systems store material information on the polygonal level of the model, so that an infrared image can be generated from this information.
- Many simulation systems are capable to procedurally generation of environment features. For instance placing trees in a certain area by the image generator. When a different system uses another algorithm for this procedural generation of content, the environment will not correlate between the different users.

The synthetic environment related fidelity issues can be seen as a specific case of the more generic fidelity issues covered in the fidelity issue group.

A.8.2.3 Related Work

Federation Engineering Agreements Template (FEAT) has a number of elements which could store information related to the synthetic environment and thereby help reduce fidelity issues:

- fedagree:terrain provides information like the area of the synthetic environment, formats used, resolution, conversion and transformations. The GML format is used to specify the bounding box and conversions/transformations. Part of the information is provided as reference to external documents.

A.8.2.4 Connection to LCIM Level

When the different fidelity results from technical limitations of the simulation systems it is different usage of the same data (Level 4, pragmatic). If the user requirements differ the content of the data varies, but the formats and structure usually remain the same (Level 3, semantic). If the different requirements of simulation components also require a different structure of the data it would be Level 2 (syntactic). Often these issues result from a different conceptual model of the participants.

A.8.2.5 Connection to DSEEP Steps and Artifacts

In Step 2 "Perform conceptual analysis" the initial choices for the fidelity of the synthetic environment are made. In all 3 activities there are tasks to perform:

- Using authoritative domain specific documentation, identify the entities, behaviours, and events that need to be represented in the scenario(s) (2.1 Develop scenario);
- Define environmental conditions of interest (2.1 Develop scenario);
- Define static and dynamic relationships between identified entities (2.2 Develop conceptual model);
- Identify events of interest within the domain, including temporal relationships (2.2 Develop conceptual model);
- Define required behaviours of identified entities and required characteristics of identified events (2.3 Develop simulation environment requirements); and
- Define requirements for natural environment representation (2.3 Develop simulation environment requirements).

In Step 3 "Design simulation environment" it is identified if existing synthetic environments can be re-used and the design of the synthetic environment and member applications interacting with it are laid down. The tasks are performed in Activities 3.1, 3.2 and 3.3:

- Search existing repositories for existing member applications and/or algorithms and code to be used by a member application (3.1 Select member applications);
- Analyse the ability of each candidate member application to represent required entities, event, and phenomena (3.1 Select member applications);
- Develop design of supporting databases (3.2 Design simulation environment); and
- Design member application (3.3 Design member applications).

In Step 4 "Develop simulation environment" choices are made for which data source to use. The tasks are performed in Activity 4.2:

- Decide the behaviour of all objects within the simulation environment (4.2 Establish simulation environment agreements); and
- Decide which databases and algorithms must be common or consistent (4.2 Establish simulation environment agreements).

A.8.2.6 Possible Solution Approaches

The following solution approaches address fidelity issues:

- Providing the needed model as a service within the simulation environment would ensure that the same model is used by all participants. Examples of this are weather servers or a weapon effect server. An issue with such services is the lack of standards for them; and
- Use predefined levels of fidelity for different simulation systems in a distributed simulation in such a way that each simulation systems uses the levels of fidelity for its individual needs *and* these different levels of fidelity do not create unfair or undesirable situations.

A.8.2.7 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.8.2.8 Recommended Information Products

Within the information product describing the synthetic environment it should be clearly described which elements of the synthetic environment need to be correlated over the different participants. Then the effort can be directed to minimizing the problems caused by different levels of fidelity for those elements. For example when an aircraft simulator and a forward air controller simulator are connected for an air-to-ground scenario the target building should be represented the same in both simulations, but other elements of the synthetic environment can differ due to the different requirements of the systems.

A.8.2.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.8.3 SN-03 Difficult to Exchange Synthetic Environments Before Runtime

A.8.3.1 Problem Definition

It is difficult to exchange synthetic environments between different simulation systems, because there is lack of a generally supported formal description of synthetic environments and a format for the exchange of such data. Besides that synthetic environments usually require a lot of storage space, which adds additional challenges to exchanging them.

A.8.3.2 Extended Problem Description

Below some examples are given of this problem:

- Many simulation systems use a proprietary format to store the synthetic environment. This means that other simulation systems cannot read the synthetic environment data directly. Instead they will have to generate the environment themselves from the same source data.
- The meta-data that describes additional attributes of the synthetic environment is not consistent between different simulation systems. For example two different systems might use a different schema to classify the type of a road.
- The fact that synthetic environments usually require a lot of storage space results in challenges when exchanging them. Often the easiest way to exchange them is to put all the data on a hard disk and mail to the other party.

A.8.3.3 Related Work

Within SISO the Reuse and Interoperation of Environmental Data and Processes (RIEDP) PDG started in 2013 to develop two products:

- The Environmental Data Model Foundations product, which will be a SISO Guidance document, and will include the formalization of a Reference Process Model (RPM) and a Reference Abstract Data Model (RADM); and
- The Environmental Detailed Features Description product, which will be a SISO Standard product, and will improve reuse and interoperability between different existing synthetic environment initiatives.

Having these two products would reduce the problems of exchanging the data for the synthetic environment between different simulation systems.

A.8.3.4 Connection to LCIM Level

Incompatible data formats lead to problems at Level 2 (syntactic). Inconsistent meta-data within an agreed data format leads to problems at Level 3 (semantic). Technical difficulties in exchanging the huge quantities of data required lead to problems at Level 1 (technical).

A.8.3.5 Connection to DSEEP Steps and Artifacts

In Activity 3.2 "Design simulation environment" the choices are made for the formats and schemas that are used in the synthetic environment and in this activity this issue should be solved:

- Evaluate the need for bridging technologies; and
- Develop design of supporting databases.

A.8.3.6 Possible Solution Approaches

The following solution approaches address exchange of synthetic environment data:

- To exchange a synthetic environment between different simulation systems a standardized exchange format would be a solution. The SEDRIS Data Representation Model (DRM) and SEDRIS Transmittal Format (STF) have been developed for this purpose [DRM][STF]. However practically the tool support for SEDRIS is still limited at the moment, which means that in reality many systems can't use SEDRIS to exchange their synthetic environment. And often tools that do support SEDRIS only support one specific STF definition, which still restricts the exchange. Having a standardize NATO SEDRIS STF would be a benefit.

- Another trend is to directly use the data from GIS formats in the simulation. For GIS formats there are a number of OGC standards that define how such data can be distributed over the web (for example WMS, WFS or WCS), thereby making the exchange of information easier. It should be noted that using GIS data directly usually implies that the creation of the synthetic environment is done on the fly in the simulation system, which increases the chance of correlation issues if different systems perform this task differently.

- The Common Database (CDB) standard from Presagis aims to provide data for the synthetic environment in a structured way so that different simulation applications can use it [CDB]. According to the CDB design the database could be placed on a central server so that multiple simulation applications can access it. If implemented in that way, the exchange of data is less of a problem.

- Store the attributes of the synthetic environment in a standardized way. The SEDRIS Environmental Data Coding Specification (EDCS) is such a standard defined for the simulation world [EDCS]. In the GIS world other standards are used, mainly developed by the Defence Geospatial Information Working Group (DGIWG). Examples of these are FACC and DFDD [DFDD].

- Describing the features of the synthetic environment in a standardized way would ease the process of creating a correlated synthetic environment for different simulation systems when each system uses different tools to create the synthetic environment. The SISO RIEDP is working on new standards for this. This solution approach does not really make the exchange of data easier, but makes it easier to create correlated results from the same input data [RIEDP_2012].

- A central service to distribute synthetic environments, a database server, can also assist in reducing the problems to exchange the synthetic environment. However this solution mainly works when all systems already support the same formats and an issue with such services is the lack of standards for them.

In the solution approaches above a number of standards are mentioned that at least partly overlap with each other, for example, SEDRIS, CDB and OGC. Each of these initiatives tries to solve problems related with the exchange of synthetic environments. However these solutions cannot always cooperate with each other. So from a situation with many proprietary formats, there is a shift to a situation with a couple of competing standards.

A.8.3.7 Existing Implementations and Their Information Products

Within the German R&D project "SD VIntEL" a Synthetic Environment Service (SES) is developed. The SES uses internally a SEDRIS-compliant database and allows export of synthetic environment data in various data formats (e.g., OpenFlight). Therefore all the simulation systems may be initialized from a common synthetic environment database. Figure A-11 illustrates the SES.

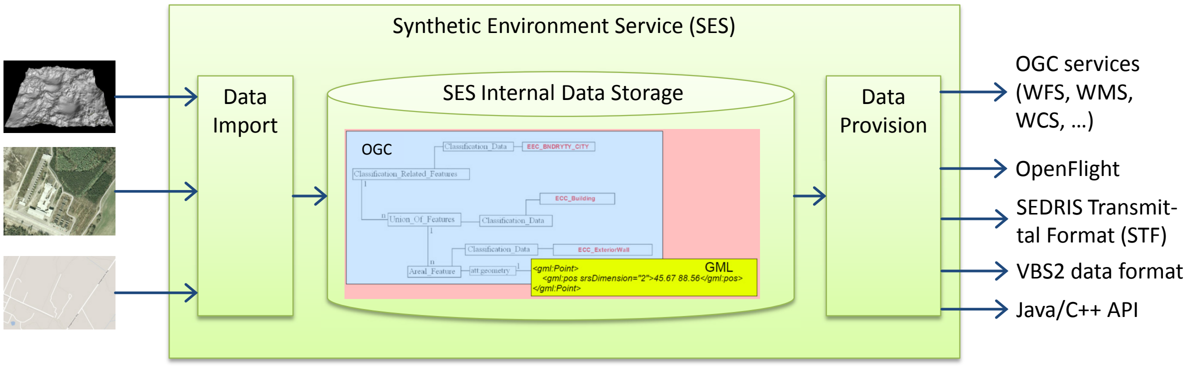


Figure A-11: Basic Idea of DEU Synthetic Environment Service (SES).

Practical experiences show the following results:

- The SES allows centralized standards-compliant storage of correlated synthetic environment data for all simulation systems;
- The SES allows initialization of simulation systems before runtime through various interfaces (OGC services, OpenFlight, etc.); and
- The SES significantly reduces interoperability problems due to uncorrelated synthetic environment data and allows time-saving automated initialization of simulation systems.

More details can be found in [Stueber2012] and [Krueckhans2012].

A.8.3.8 Recommended Information Products

No recommendations made by MSG-086.

A.8.3.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.8.4 SN-04 Difficult to Exchange Synthetic Environment Updates at Runtime

A.8.4.1 Problem Definition

Keeping the different synthetic environments within a simulation exercise synchronized during the execution is difficult. Each simulation component typically determines the consequences of events that influence the environment locally. Therefore the representation of the different synthetic environments might diverge after time.

A.8.4.2 Extended Problem Description

Below some examples of this problem are given:

- When a weapon impacts with the terrain, a detonation event is typically distributed in the simulation exercise. Each synthetic environment will locally determine to what extend the terrain is modified by this impact. As a result the change to the terrain is often not correlated between different synthetic environments. If for example the CGF application does not alter the terrain, the entities simulated at the CGF might not be limited in their manoeuvrability.
- When the location of clouds is not synchronized correctly between different simulation systems this can affect the observability of an entity and thereby result in fair-fight issues.
- Atmospheric conditions, like precipitation, can have an influence on the terrain and thereby on the manoeuvrability of the entities. This kind of interactions is not simulated in a consistent way in most systems.

Issue LC-03, Missing information exchange between real and simulated environment, can be seen as an extreme case of the problems with updating the environment in the context of LVC operations. It deals with the problems of exchanging information between the synthetic environment and a real environment.

A.8.4.3 Related Work

Federation Engineering Agreements Template (FEAT) has a number of elements which could store information related to the synthetic environment and thereby help reduce data exchange issues:

- fedagree:environment provides information related to SNE and weather databases, including algorithms and temporal updates. Most of the information is provided as reference to external documents.

A.8.4.4 Connection to LCIM Level

If the usage of the updates of the synthetic environment is not interpreted in the same way across the different systems it leads to problems at Level 4 (pragmatic). If the updates are interpreted consistently, but the effect of the updates on other elements within the synthetic environment is not clear it lead to problems at Level 5 (dynamic).

This assumes that there are standards to communicate the updates between different simulation systems, else it will lead to problems at a lower level.

A.8.4.5 Connection to DSEEP Steps and Artifacts

In Step 2 “Perform conceptual analysis” the initial choices for the updates that need to be made to the synthetic environment are made:

- Define static and dynamic relationships between identified entities (2.2 Develop conceptual model);
- Identify events of interest within the domain, including temporal relationships (2.2 Develop conceptual model); and
- Define required behaviours of identified entities and required characteristics of identified events (2.3 Develop simulation environment requirements).

In Step 3 “Design simulation environment” the basic data is made for the exchange of synthetic environment updates during runtime:

- Develop design for simulation environment infrastructure, and select protocol standards and implementations (3.2 Design simulation environment); and
- Design member application (3.3 Design member applications).

In Step 4 “Develop simulation environment”, Activity 4.1 “Develop simulation data exchange model” the agreements on how to exchange updates are developed:

- Develop and document the SDEM.

In Activity 4.2 “Establish simulation environment agreements” it is then decided how the data is to be distributed:

- Decide how data is to be distributed across the simulation environment.

A.8.4.6 Possible Solution Approaches

The following solution approaches address exchange of synthetic environment data:

- A service that processes changes to the synthetic environment in one central place would be a solution to make sure these changes happen consistently over different systems. The different systems would then need to request their information from this central service. An issue with such services is the lack of standards for them. An example is a weapon effects server or a weather server. Data could also be distributed to subscribers when there are changes in the synthetic environment. If DDM (HLA – Data Distribution Management) is used could filtering make this distribution effective, subscribers will only get the information that is of interest to them.
- With HLA these kinds of interactions would be captured during FOM design, interaction classes are created.

A.8.4.7 Existing Implementations and Their Information Products

Within the German R&D project “SD VIntEL” a Synthetic Dynamic Service (SDS) is developed. The SDS allows to change/update certain objects and environmental features during runtime (execution) of a simulation environment. This is done via manipulations on the environment database which is distributed to the connected simulation systems (using the SES, see Section A.8.3). While runtime updates of environmental data and features are quite easy for simple objects like buildings (e.g., opening a gate) manipulations on the terrain itself are complex and take some time even for simpler actions (i.e., several seconds for imprinting a crater into the terrain).

A.8.4.8 Recommended Information Products

No recommendations made by MSG-086.

A.8.4.9 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.9 ISSUE GROUP "TIME MANAGEMENT" (TM)

A.9.1 TM-01 Temporal Anomalies Caused by Differences in Precision of Time Representation

A.9.1.1 Problem Definition

- The fidelity of a simulation is affected by the internal time representation and time management in member applications (federates) in a simulation environment (federation execution).
- At the time of implementation of the applications, the requirements on time management and fidelity in the simulation environment may not be clearly defined.
- At the time of integrating member applications into a simulation environment, the member applications are not easily modified to meet time management and fidelity requirement.
- Member applications do not clearly document their time representation and time management capabilities/schemes.
- Synchronization of (internal) time with other member applications requires agreements on common time representation and time management.

When choosing an appropriate time representation, one needs to consider among other things the following factors:

- Deciding the size of the time step; and
- Determining the maximum time span that needs to be used.

Once those two factors are known, it should be possible to select an appropriate time representation that fulfils all the needs of the simulation environment.

A.9.1.2 Extended Problem Description

Most binary representations have a limited precision. An integer representation has a precision of 1. The typical floating-point representations have a limited number of significant digits. Depending on the value of the exponent, the least significant digit may be 1/4 000 000 or 1/2. This value that the least significant digit (lsb) represents if it is 1 is called the Unit of Least Precision (ULP). It can be said that the fixed-storage representations trade precision for range. When the value goes up, the precision goes down.

Exact Values

Binary representations are unable to make an exact representation of some values. For example, a decimal representation can only represent the value 1/3 as an approximation. This is quite intuitive to humans as the same problem occurs in the common decimal system. The value 1/3 generates an infinite number of decimals, 0.33333333..... Less intuitive is the fact that float64 cannot represent the value 0.1 exactly. In the same way as the value 1/3 generates an infinite decimal representation the value 0.1 generates an infinite binary representation.

Most developers would hesitate to use the value 1/3 as time step since it cannot be exactly represented in a decimal form. However, they will use the value 1/10 (0.1) as time step with a float64 representation without realizing that this value cannot be exactly represented in a base 2 format.

Confusing Arithmetic

Adding large and small values using floating-point representations can yield unexpected results since the smaller value can fall below the ULP of the large value and therefore not affect the large value.

Accumulated Errors

Using floating-point arithmetic accumulated errors during calculations may cause rounding errors. For example, 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 + 0.1 in floating-point arithmetics = 0.9999999999999999 while 0.1 * 10 = 1.0.

This discrepancy may cause problems, for example if a federate is planning to perform an action at time 1.0 and is checking the current time to see if it is equal to that value. The values will never be equal and the action will never be performed. It is also very likely that the current time is presented as a rounded value which will be 1.0, thereby hiding the problem and making debugging even more complicated.

Comparing for Equality

Due to the limited precision of floating-point numbers, comparing for equality is unlikely to give the intended result.

```
if (result == expectedValue)
```

A common attempt to solve this problem is to say that the values are considered equal if they are closer than a predefined distance, say 0.00001.

```
if (abs(result - expectedValue) < 0.00001)
```

This predefined distance may work fine during development of a simulation where simulation time never exceeds 1 000, but when the simulation is put into production and it runs for 100 000 steps the predefined distance 0.00001 falls below the precision of the time representation and the comparison fails.

Changing Behavior

In some situations, the result of floating-point calculations can depend on such unexpected factors as compiler flags. For example, the following C program prints one value when compiled in optimized mode, while it prints another value when compiled in standard mode.

```
double foo(double v) {
 double y = v * v;
 return (y / v);
}

main() { printf(“%g\n”, foo(1E308));}
```

The reason for this behavior is that the computation v × v is done in extended precision, with a larger exponent range. In optimized mode, the extended precision result of v × v stored in a register is used in the calculation of y / v. In standard mode, the result of v × v is stored in the double precision variable y as positive infinity due to the overflow. The variable y is then loaded and used in the calculation of y / v to yield positive infinity.

A.9.1.3 Connection to LCIM Level

Concerns mainly LCIM Levels 1 (technical) and maybe 2 (syntactic).

A.9.1.4 Connection to DSEEP Steps and Artifacts

Affected artefacts, this problem may be detected during one of the following steps, sooner is better:

- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment;
- 6.1 Execute Simulation; and
- 7.1 Analyze Data.

This problem could be discovered and managed during:

- 3.1 Select Member Applications;
- 3.2 Design Member Applications; and
- 4.1 Develop Simulation Data Exchange Model.

A.9.1.5 Possible Solution Approaches

There is nothing that prevents an application from using its own internal time representation and then adapting to the requirements of the simulation environment where it participates. It is desirable that applications can support a number of different time representations used in simulation environments.

When selecting an internal time representation for a simulation, there are several factors that need to be taken into consideration. In addition to the ones mentioned above are:

- What time resolution is needed to meet the requirements, both internally in the model and externally in a simulation environment?
- Can this member application use one of the standardized time types that are available?
- Integer or fixed-point time representations are considerably easier to use and debug compared to floating-point time representations. See for example 11S-SIW-049 for details on this topic.

This issue should take into account all these factors.

A.9.1.6 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.9.1.7 Recommended Information Products

No recommendations made by MSG-086.

A.9.1.8 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.9.2 TM-02 Temporal Anomalies Caused by Differences in Time Resolution

A.9.2.1 Problem Definition

When the time resolution at member applications in a simulation environment differs some problems may arise, events from member applications with low resolution (large time step) may for member applications with high resolution (small time step) be regarded as delayed and not received in time.

An event (generated by an event-driven application) will in general not match exactly with the boundary of a time step in a time stepped application, but will occur somewhere within a time step. As a consequence, there will remain a slight mismatch in time between the applications.

A.9.2.2 Extended Problem Description

If member applications in a simulation environment have large difference in their time step size:

- Reaction on events at member application with a low time resolution may, for member applications with high time resolution, be delayed and delivered to late.
- A member application with high time resolution will during most of its time steps not receive any update for subscribed entities from member applications with low time resolution. This may cause that a member application have defective estimation of subscribed entities.
- Member applications with low time resolution will during one time step receive a number of updates with different time stamp for each external entity from member applications with high time resolution. The member application must in its logic manage the changes for external entities that occurs during a time step.

A.9.2.3 Connection to LCIM Level

Concerns mainly LCIM Levels 1 (technical) and maybe 2 (syntactic). This issue could also be categorized at Level 6 (conceptual).

A.9.2.4 Connection to DSEEP Steps and Artifacts

Affected artefacts, this problem may be detected during one of the following steps, sooner is better:

- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment;
- 6.1 Execute Simulation; and
- 7.1 Analyze Data.

This problem could be discovered and managed during:

- 2.2 Develop Conceptual Model;
- 2.3 Develop Simulation Environment Requirements;
- 3.1 Select Member Applications; and
- 3.2 Design Member Applications.

A.9.2.5 Possible Solution Approaches

Applications should be configurable to use a time resolution that is acceptable for application itself and supports the purpose of the simulation environment (see ref. for an example where this solution is model-based).

A member application with a high time resolution can use Dead Reckoning to extrapolate spatial information of entities. DR means, that the receiver of updates makes a guess/extrapolation on the senders state. The receiver, which uses this extrapolated information for his own model computations, must keep in mind, that the extrapolated information resembles the exact state of the sending entity only in the (rare) case, where the equations of motions are identical to the DR equations, i.e., the position data at future timestamps can be computed from the known state (position, velocity, acceleration) at the present timestamp. This is not the case for any sophisticated model for object motion. The mismatch between the extrapolated and the true state information can be minimized by choosing a different DR algorithm or decrease the error threshold level, when new updates are to be sent by the sending entity. Therefore the level of complexity of the dead reckoning algorithms and the thresholds levels shall be set to meet the requirements of the simulation environment.

A member application with a low time resolution can downsample the number of updates with Update Rate Reduction in HLA Evolved. The selection of attributes for down sampling shall not inflict on the subscribing applications performance and behavior, e.g., do not down sample some state changes but entity position can in many applications be down sampled if only the latest position on external entities is of interest in a time frame. It should be kept in mind however, that, e.g., the position is a continuous function over time described by some equations of motion chosen by the model, which is sampled and transmitted by the sending entity at a certain sampling frequency normally high enough for the receiver to reconstruct the position information to the required accuracy at any point in time. Sampling however introduces a frequency window in the spectral properties of the position data, i.e., the frequency spectrum of the positional data is “mirrored” at the Nyquist frequency (which is half the sampling frequency), resulting in high frequency components to be “folded back” to lower frequencies. As this first sampling takes place at the sending entity, one can normally be sure, that the sampling frequency is chosen (by the sending model) to be high enough to prevent these effects from occurring. However, further downsampling the transmitted data (e.g., due to Update Rate Reduction by the HLA RTI) reduces the Nyquist frequency of the sampled data thus folding higher frequency components to lower frequencies thereby producing well-known aliasing artifacts in the resulting downsampled positional data. These artifacts can be avoided by low-pass filtering the data before downsampling. This however would require the RTI to interpret the transmitted data and to apply a low-pass filter before downsampling, which is not required by the HLA standard and not implemented in present RTI implementations.

A.9.2.6 Existing Implementations and Their Information Products

The Generic Distribution Model was developed to overcome special timing problems and time step conflicts within a simulation environment. The usage of this model requires the implementation of PSI-SA in the simulation system.

The conceptual core of PSI-SA (PSI-SA: Primary Simulation Interface for Simulation Applications, a middleware that is capable of coupling different HLA-based simulation systems regardless of the manufacturer of the underlying RTI) is the generic distribution model for networked simulation. This model is based on 3 pillars:

- Time;
- State; and
- Operator.

The introduction of operators on a state space as an independent concept, and the implementation of this concept in the PSI-SA core led in practice to robust models for the coupling of simulation systems. A significant advantage of the operator concept is the reproducibility of the simulation results. But prerequisite for the reproducibility of the result is the reproducibility of the initial state. Here PSI-SA is treading new ways for its own concept for the initialization and its implementation in the PSI-SA core. However, the concept of initialization itself can be applied only to the initial state at time zero.

The creation or destruction of objects after initialization would lead to increasing problems because the object could not be initialized reliably and at its first interaction with other objects fell into an undefined or inconsistent condition. Therefore, the operator concept was expanded to specialized operators for the production and destruction of objects. Particularly for transient objects these new operators should produce a deterministic behavior within the simulation environment, because previously the generation of new objects did not match the execution of operators.

To ensure an uniform behavior of all federates in the creation and annihilation of objects the generation and the annihilation operator was implemented in the software design of PSI-SA and in the architecture layer PSI-COS (PSI-COS: PSI-SA Common Object Service, part of PSI-SA that covers all HLA services).

A.9.2.7 Recommended Information Products

No recommendations made by MSG-086.

A.9.2.8 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.9.3 TM-03 Temporal Anomalies Caused by Unsynchronized Time

A.9.3.1 Problem Definition

A simulation environment executing with an unsynchronized perception of time among the member applications, may have problem with time stamped events. This can occur when not using a method for time synchronization such as HLA Time Management and the system clocks are not synchronized or when system clocks are running with different speed.

A.9.3.2 Extended Problem Description

Time stamped events in a time unsynchronized simulation environment may at the receiver get an incorrect interpretation. Unsynchronized time will in a simulation environment with location and movement such as a RPR2-FOM simulation environment cause incorrect spatial conception, including incorrect Dead Reckoning calculations.

In simulation environments that use the Simulation Execution Control State Transition pattern (MSG-052) can unsynchronized time cause problem when the execution shall start/resume, stop/freeze or reset at a specified time.

A.9.3.3 Connection to LCIM Level

Concerns mainly LCIM Level 1 (technical).

A.9.3.4 Connection to DSEEP Steps and Artifacts

Affected artifacts, this problem may be detected during one of the following steps, sooner is better:

- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment;
- 6.1 Execute Simulation; and
- 7.1 Analyze Data.

This problem could be managed during:

- 3.2 Design Simulation Environment;

A.9.3.5 Possible Solution Approaches

- Use HLA Time Management to synchronize the time in the simulation environment.
- Use NTP (Network Time Protocol) to synchronize system clocks.
- Use PTP (Precision Time Protocol) to synchronize system clocks.
- Use GPS (Global Positioning System) to synchronize system clocks.

A.9.3.6 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.9.3.7 Recommended Information Products

No recommendations made by MSG-086.

A.9.3.8 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

A.9.4 TM-04 Temporal Anomalies Caused by Network Latency

A.9.4.1 Problem Definition

In simulation environments with member applications using high update rates can a high network latency cause problems for time stamped events sent between the member applications. In framed-based applications may the events that were supposed to be received in a frame be delayed and delivered in a frame at a later time. This is more commonly in simulation environments with member applications communicating over WAN (Wide Area Network).

A.9.4.2 Extended Problem Description

For analytic and high resolution types of simulation environments that use the Simulation Execution Control State Transition pattern (MSG-052) can high network latency cause problem at Start/Resume if execution start time is “now” and not specified at some time later on. Application members in the simulation environment will in this case start at different time.

The latency between member applications communicating over WAN will normally vary in a higher degree and be more unpredictable than if communication over a LAN (Local Area Network), this implies that higher demands to manage this needs to be put on the member applications if communication over WAN is used in the simulation environment.

A practical example is the ASTA WAN networking for the Eurofighter within the German Air Force. There the problem may arise when simulation systems which are separated several hundred kilometres from each other are close to each other within the simulation. The actions the simulation systems are conducting may result, for example, in stuttering because the actions cannot be synchronized properly due to signal run time and signal processing time.

A.9.4.3 Connection to LCIM Level

Concerns mainly LCIM Level 1 (technical).

A.9.4.4 Connection to DSEEP Steps and Artifacts

Affected artefacts, this problem may be detected during one of the following steps, sooner is better:

- 5.2 Integrate Simulation Environment;
- 5.3 Test Simulation Environment;
- 6.1 Execute Simulation;
- 7.1 Analyze Data; and
- 4.1 Develop Simulation Data Exchange Model.

This problem could be discovered and managed during:

- 3.2 Design Simulation Environment; and
- 4.1 Develop Simulation Environment.

A.9.4.5 Possible Solution Approaches

Use HLA Time Management to synchronize the time in the simulation environment.

If the Simulation Execution Control State Transition pattern (MSG-052) is used, do not set the start time to "now".

A.9.4.6 Existing Implementations and Their Information Products

No existing implementations identified by MSG-086.

A.9.4.7 Recommended Information Products

No recommendations made by MSG-086.

A.9.4.8 Examples/Prototypes of Selected Information Products

No examples/prototypes identified by MSG-086.

Annex B – DROPPED ISSUES

This annex contains all issues identified by ET-027 which were not considered in detail by MSG-086 (due to different reasons that are explained in the following sections).

B.1 ISSUES REGARDING "APPLICATION SERVICES"

Identified issues from ET-027 list which are dropped:

- 17 Application services (e.g., weapon effect server);
- 18 Weapon Effects Server;
- 40 Communication Effects Server; and
- 59 Leverage of Commercial Games.

To assure fair fight conditions between different simulation systems, one has to agree upon procedures that are used by all simulation systems, e.g., for the evaluation of weapon effects. One possible solution is the use of centralized services that are either called by the simulation systems (e.g., web services) or publish their results in the federation execution (e.g., HLA). Currently, there are no accepted standards for application services (e.g., their functionality and interfaces).

The ET-027 issues refer to the use of a centralized resource to resolve common simulation problems, e.g., weapon effects, damage, communications effects (such as terrain masking). Such an approach can help ensure fair fight.

However, application services are primarily considered to be a solution, not a problem (although problems may arise from the adoption of these solutions, such as a lack of standardized interfaces for servers).

MSG-086 decided to drop these issues and to describe a new issue regarding lack of standards for application services (see issue CM-05).

B.2 ISSUES REGARDING "SECURITY"

Identified issues from ET-027 list which are dropped:

- 52 security systems, measures and IPR, varying levels of classification; and
- 53 security classification (per se).

Security issues are a barrier to interoperability as they can prevent the free exchange of simulation data.

As the area of security is being addressed by MSG-080 the decision of MSG-086 was to drop these issues so as not to duplicate the work of MSG-080.

B.3 ISSUES REGARDING "SYNTHETIC ENVIRONMENT"

Identified issues from ET-027 list which are dropped:

- 31 Environment server; and
- 32 Terrain DB server.

Within the issue group Synthetic Environment (SN) two issues from the original ET-027 list of issues have been dropped. Both of these issues are not considered to be an issue, but they are a possible solution approach to other issues. These solution approaches have been included in the possible solution approaches section of the appropriate issues within the SN issue group.

B.4 ISSUES REGARDING "LEGACY SYSTEMS"

Identified issues from ET-027 list which are dropped:

- 45 Legacy systems interoperability.

MSG-086 decided to drop this issue because the interoperability issues with legacy systems are not fundamentally different from the interoperability issues with more modern systems. For legacy systems certain issues might be felt stronger, but the issues encountered are still of the same nature.

Examples for issues regarding legacy systems are the following:

- Data issues:
  - "For compatibility with legacy systems, it is recommended that Marking Text is limited to 11 characters." (from: *M&S Support to Assessment of Extended Air Defence C2 Interoperability Final Report of Task Group MSG-039, NATO Research and Technology Organization, December 2008.*)
  - Example issue(s) from US DoD: A-10 DTOC system cannot utilize current DIS Enumerations.
- Hardware and software problems:
  - "It may take the vendor a long time (if ever) to support the application on a newer Operating System. Even if the technology becomes available, according to the Meritalk/Unisys survey, adoption can take up to 3 years." (from: *The Danger of legacy Systems at* http://www.mousesecurity.com/?p=220.)
  - "Legacy systems often run on obsolete (and usually slow) hardware, and spare parts for such computers may become increasingly difficult to obtain." (from: http://en.wikipedia.org/wiki/Legacy_system.)
  - "Integration with newer systems may also be difficult because new software may use completely different technologies. The kind of bridge hardware and software that becomes available for different technologies that are popular at the same time are often not developed for differing technologies in different times …" (from: http://en.wikipedia.org/wiki/Legacy_system.)
  - "HLA/RTI Version: The TG recommends HLA IEEE-1516 … However, due to legacy federate capability, we decided to use HLA RTI V1.3 NG." (from: *M&S Support to Assessment of Extended Air Defence C2 Interoperability Final Report of Task Group MSG-039, NATO Research and Technology Organization, December 2008.*)
  - "There are several ways in which the separation event can be handled in an SE, but all of them have potential problems with legacy simulations:
    - Ignore it; just fly the 'whole missile' object along the BM trajectory:
      - Sensors may assign too large a signature to the missile post-separation and detect the target when they should not.
    - Keep the same object, but change its enumeration from Whole Missile to Warhead:
      - This breaks the rules. Many systems cache type information when they first see an entity, so may not notice the change.

      - It is possible to add new objects to represent debris and the dead booster, but this can lead to an overestimate of the combined signature.
    - Delete the Whole Missile and create new components:
      - A remote federate may see the deletion before any new entities are created. This may cause sensor models to drop any existing track on the missile, so the new components will not be associated with the old track.
    - Create new components then delete the Whole Missile:
      - Remote federates may see the new and old entities together, potentially resulting in an anomalously high sensor return.
    - Leave the Whole Missile flying, but flag it as destroyed:
      - Updates to a destroyed entity may confuse some federates. Some sensors may still 'see' the destroyed entity.
    - Use attached parts to model the sub-components:
      - This approach is currently under discussion within the DIS/HLA SISO community, but it is not yet mature enough to be a viable option. This is likely to be incompatible with many legacy systems and there may also be issues with translating between DIS and HLA." (from: *M&S Support to Assessment of Extended Air Defence C2 Interoperability Final Report of Task Group MSG-039, NATO Research and Technology Organization, December 2008.*)
  - Example issue(s) from US DoD:
    - Interoperability issues related to outdated DIS versions; A-10, EP3, F15 (DIS 2.0.6), B-52 (DIS 2.0.4).
    - Interoperability issues related to entity limits; A-10 DTOC limited to 16, B-52 limited to 2 (itself and refueler or JDAM).
    - Configuration challenges; AWACS – Difficult to configure radios, B-52 – Unable to reconfigure ASTi radio, EP3 MAST – must use JSAF for emitters.

- Inadequate documentation:
  - "Lack of good documentation (e.g., problem statement, requirement definitions, conceptual model, design products, data descriptions, documented code) makes modifying a legacy simulation difficult." A Practitioner's Perspective on Simulation Validation.

- Dissimilarities in intended use.

- Inadequate knowledge base:
  - "These systems can be hard to maintain, improve, and expand because there is a general lack of understanding of the system; the staff who were experts on it have retired or forgotten what they knew about it, and staff who entered the field after it became "legacy" never learned about it in the first place. This can be worsened by lack or loss of documentation. A regional airline fired its CEO in 2004 due to the failure of an antiquated legacy crew scheduling system that ran into a limitation not known to anyone in the company." http://www.cio.com/article/112103/Comair_s_Christmas_Disaster_Bound_To_Fail as referenced at http://en.wikipedia.org/wiki/Legacy_system.

- Unforeseen consequences:
    - Example issue(s) from US DoD: Interoperability issues related to heartbeat flexibility (or inflexibility); AWACS – heartbeat of constructive A/C caused problems for AWACS radar, FMSD – High update rates required for radar.
- Cost:
    - “Many legacy, and even some new, simulations will not transition to using a different architecture, unless there are compelling incentives to do so.” (from: A. Henninger, et al. “Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report” Institute for Defense Analyses, September 2008.)
    - “Integration with newer systems may also be difficult because new software may use completely different technologies. The kind of bridge hardware and software that becomes available for different technologies that are popular at the same time are often not developed for differing technologies in different times, because of the lack of a large demand for it and the lack of associated reward of a large market economies of scale …” (from: http://en.wikipedia.org/wiki/Legacy_system.)
    - “In general it is not trivial to change the FOM of a legacy federate. Often the cost will be significant and only the owners/developers of a legacy system have the proper source code and knowledge to make changes to the FOMs.” (from: “M&S Support to Assessment of Extended Air Defence C2 Interoperability Final Report of Task Group MSG-039”, NATO Research and Technology Organization, December 2008.)

Many of the examples given above are described in the issue group federation development already. They are the result of using multiple architectures (FD-01), using different versions (FD-02) or the result of lack of documentation (FD-07).

Issues like additional cost, both in terms of time and resources, are not interoperability issues per se. In many cases these are the consequences of the interoperability issues encountered with legacy systems.

B.5 ISSUES REGARDING “FEDERATION DEVELOPMENT”

Identified issues from ET-027 list which are dropped:

- 42 Traceability and relationship between process artifacts;
- 51 Implementation Processes; and
- 60 Responses to rapidly changing requirements.

The above issues are dropped because they do not reflect original interoperability problems.

B.6 OTHER ISSUES

Identified issues from ET-027 list which are dropped:

- 56 Extended SE standards (CGFs, etc.); and
- 61 Future systems.

These issues are dropped as MSG-086 was unable to obtain clarification or expansion of the issues.

Annex C – TRACEABILITY TO ET-027 ISSUES

The following table provides detailed traceability of issues from the original ET-027 to the final list of issues as analysed and documented by MSG-086.

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
1	Scenario description.	DEU, CAN, NLD	Scenario	SC-02	Incomplete or inconsistent operational scenario description.	
2	Scenario structure.	DEU	Scenario	SC-04	Use of different formats for executable scenarios.	
3	Scenario interchange.	DEU	Scenario	SC-03	Missing formal scenario specification.	
4	Problem space / mission space structure and derivation from authoritative scenarios.	DEU, CAN	Scenario	SC-01	Missing authoritative operational scenarios.	
5	Application space structure and sub-structures.	DEU	Scenario	SC-06	Missing or incomplete definition of application domain.	
6	Formalized specification of conceptual models for a given problem space and application (application sub-space) –> ontologies, meta descriptions.	DEU, FRA, NLD	Conceptual Model	CM-02	Missing standards/templates/guidelines for development of a conceptual model for distributed simulation environments.	
7	Formalized transition from conceptual models to data models / object models.	DEU	Conceptual Model	CM-04	Lack of standard methodologies, techniques and tools for automated transition from conceptual models to Simulation Data Exchange Models (SDEMs) or (automated) comparison and verification between conceptual models and SDEMs.	

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
8	Formalized description of characteristics and relation of different aggregation state entities.	DEU, CAN	Fidelity / Aggregation	FI-01		
9	Agreed levels of entity aggregation/ fidelity/resolution → fidelity inconsistencies.	DEU, GBR	Fidelity / Aggregation	FI-02	Fidelity Inconsistencies (agreed levels of entity aggregation, fidelity, resolution).	Changed to "Fidelity Inconsistencies (agreed levels of entity aggregation, fidelity, resolution)".
10	Data fidelity classification.	GBR	Fidelity	FI-03		
11	Fidelity of LINK 16 systems.	GBR	Fidelity	FI-04	Fidelity of LINK systems.	Changed to "Fidelity of LINK systems".
12	Formalized transfer/mediation functions between different aggregation state or different fidelity or different resolution entities.	DEU	Fidelity	FI-05		
13	Entity (systems, humans) behavior (dynamics) agreed description; interactions.	DEU, CAN	Fidelity	FI-06		
14	Areas of critical dynamic behaviors (with respect to fair fight) and corresponding algorithms.	DEU, GBR	Fidelity	FI-07	Lack of agreed critical behaviours and corresponding algorithms.	
15	Unrealistic model performance effects.	GBR	Fidelity	FI-07	Lack of agreed critical behaviours and corresponding algorithms.	
16	Inconsistent interpretation of fire and detonate messages.	GBR	Fidelity	FI-07	Lack of agreed critical behaviours and corresponding algorithms.	
17	Application services (e.g., weapon effect server).	DEU	–	Dropped		Considered as a solution approach, not a problem.

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
18	Weapon effects server.	DEU	–	Dropped		Considered as a solution approach, not a problem.
19	Formalized doctrine / ROE descriptions.	DEU	Scenario	SC-05	Use of different doctrines and ROEs.	Add new issue "Missing specification of doctrine/ ROE", either to group "Scenario" or group "Conceptual Model".
20	Different time advancement schemes.	DEU	Time Management	TM-01, TM-02, TM-03	Distributed simulation member application time management. Distributed simulation time management. Scenario time management.	Related to all three TM issues.
21	Time management / time synchronization.	CAN, GBR	Time Management	TM-01, TM-02, TM-03	Distributed simulation member application time management. Distributed simulation time management. Scenario time management.	Related to all three TM issues.
22	Human Machine Interface.	CAN	Fidelity	FI-08	Inconsistent Human Machine Interfaces.	
23	Infrastructure/frameworks/tools.	CAN, GBR, FRA	Infrastructure and Tools	IN-04	Common Infrastructure Framework and Tools.	
24	Infrastructure Services (e.g., initialization server).	DEU	Infrastructure and Tools	IN-03	Common Infrastructure Services.	

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
25	Tool incompatibility.	GBR	Infrastructure and Tools	IN-01, IN-02		Covered partly in IN-01 and IN-02.
26	External Interface.	CAN	LVC-Interoperability	LC-01	Limitations due to integration of Live Systems.	Generalization of 11, 38, 46, 47, 49.
27	Execution Management (Configuration, Control and Monitoring).	CAN, DEU, NLD, FRA	Infrastructure and Tools	IN-02	Distributed Simulation Initialization and Execution Management.	
28	After Action Review; data logger, early understanding of logging and replay requirements.	CAN, GBR, FRA	Infrastructure and Tools	IN-01	Data Logging for After Action Review and Analysis.	
29	Agreed levels of synthetic environment fidelity.	DEU	Synthetic Environment	SN-02	Different levels of fidelity are used for synthetic environment.	
30	Agreed environment databases, database correlation and exchange mechanisms.	DEU, NLD	Synthetic Environment	SN-01, SN-03	Synthetic environment data is not correlated. Difficult to exchange synthetic environments before runtime.	SN-01 and SN-03 are the two more general issues that cover this issue.
31	31 Environment server.	FRA	Synthetic Environment	Dropped		Considered as a solution approach, not a problem.
32	32 Terrain DB server.	DEU	Synthetic Environment	Dropped		Considered as a solution approach, not a problem.
33	Agreed environmental data models.	CAN	Synthetic Environment	SN-02, SN-03	Different levels of fidelity are used for synthetic environment. Difficult to exchange synthetic environments before runtime.	SN-02 and SN-03 are the two more general issues that cover this issue.

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
34	Agreed earth model and coordination systems.	CAN, GBR	Synthetic Environment	SN-01	Synthetic environment data is not correlated.	SN-01 is a more general issue, which includes this issue.
35	Agreed Environmental conditions.	CAN	Synthetic Environment	SN-02	Different levels of fidelity are used for synthetic environment.	This is a special case of issue 29.
36	Agreed natural environment database formats.	CZE	Synthetic Environment	SN-03	Difficult to exchange synthetic environments before runtime.	SN-03 is a more general issue, which includes this issue.
37	Adaptor services (e.g., link C2 sys to Simsys).	DEU	LVC and C2-Sim coupling	LC-05	Limited simulation awareness of C2 systems.	
38	Link to C2.	CAN, CZE, GBR, FRA	LVC and C2-Sim coupling	LC-05	Limited simulation awareness of C2 systems.	
39	Communication Model; devices performance and degradation effects; electronic warfare.	CAN	Fidelity	FI-09		
40	Communication effects server.	DEU	–	Dropped		Considered as a solution approach, not a problem.
41	Process (e.g., FEDEP) artifacts standards; FADs.	NLD	Federation Development	FD-07	Missing standards and templates for artifacts.	
42	Traceability/relationship between process artifacts.	NLD	Federation Development	Dropped		No direct influence on interoperability.
43	Documentation templates.	NLD	Federation Development	FD-07	Missing standards and templates for artifacts.	

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
44	Visual systems, visual model formats and fair fight.	NLD, GBR, FRA	Fidelity	FI-08	Inconsistent Human Machine Interfaces.	Special case of 22.
45	Legacy systems interoperability.	NLD	Legacy Systems	Dropped		Legacy systems face same problems as more current systems.
46	FOM versions.	NLD	Federation Development	FD-03, FD-04	Different FOM or SDEM versions. Incompatible FOM modules.	
47	DIS/HLA Versions.	NLD	Federation Development	FD-02	Different HLA versions.	
48	Heterogeneous SEs.	GBR	Federation Development	FD-01	Multi-Architecture Simulation Environments.	
49	SE-Real Systems; integration of live and simulated components.	GBR	LVC and C2-Sim coupling	LC-01	Limitations due to integration of Live Systems.	
50	Overarching governance across acquisition programmes.	GBR, NLD	Organizational	OL-02	Missing or limited policy, coordination and cooperation.	
51	Implementation processes (Engineering, VV&A, Environment, Security, etc.).	GBR, FRA	Federation Development	Dropped		No direct influence on interoperability.
52	Security systems/measures and IPR; varying levels of classification.	GBR	Security	Dropped		Discussed in detail by MSG-080.
53	Security classification (per se).	GBR	Security	Dropped		Discussed in detail by MSG-080.

No.	Issue Title from ET-027	Sponsor	Issue Group	New No.	New Title	Comment
54	Multi organizational production/ cooperation programmes.	GBR, NLD, FRA	Organizational	OL-02	Missing or limited policy, coordination and cooperation.	
55	Centrally co-ordinated research findings.	GBR	Organizational	OL-02	Missing or limited policy, coordination and cooperation.	
56	Extended SE standards (CGFs, etc.).	GBR	?	Dropped		Unable to obtain clarification or expansion of this issue, but it's possible it may be related to (or a restatement of) issue 47 where proprietary extension to standards can impair interoperability.
57	Architectures; technology independent architecture, extensible architecture.	GBR, FRA	Federation Development	FD-06	Missing comprehensive reference architectures.	
58	Policy (promote cooperation).	GBR	Organizational	OL-02	Missing or limited policy, coordination and cooperation.	
59	Leverage of commercial games.	GBR	–	Dropped		Considered as a solution approach, not a problem.
60	Responser to rapidly changing requirements.	GBR	Federation Development	Dropped		Not a specific interoperability issue for simulation systems.
61	Future systems.	GBR	?	Dropped		Unable to obtain clarification or expansion of this issue.
62	Network loading and packet loss.	GBR	Infrastructure and Tools	IN-02	Data Overflow.	
63	Technical personnel behaviour and attitude.	GBR	Organizational	OL-03	Cultural aspects.	

Annex D – DISSEMINATION

D.1 PAPERS AND CONFERENCE PRESENTATIONS

The following papers and conference presentations contain material and/or discussion results of MSG-086:

- R. Siegfried, M. Bertschik, M. Hahn, G. Herrmann, J. Lüthi and M. Rother, "Effective and Efficient Training Capabilities through Next Generation Distributed Simulation Environments", 2013 NMSG Multi-Workshop, Sydney, Australia, Paper 7.
- R. Siegfried, H. Oguztüzün, U. Durak, A. Hatip, G. Herrmann, P. Gustavson and M. Hahn, "Specification and Documentation of Conceptual Scenarios Using Base Object Models (BOMs)", 2013 SISO Spring SIW, San Diego, CA, USA, Paper 13S-SIW-019.
- R. Siegfried, J. Lüthi, M. Rother and M. Hahn, "Limiting Issues for True Simulation Interoperability: A Survey Done by MSG-086 and Resulting Recommendations", 2013 SISO Spring SIW, San Diego, CA, USA, Paper 13S-SIW-035.
- R. Siegfried, J. Lüthi, M. Rother, D. Steinkamp and M. Hahn, "Limiting Issues for True Simulation Interoperability: A Survey Done by MSG-086 and Resulting Recommendations", 2012 NMSG Conference, Stockholm, Sweden, STO-MP-MSG-094.
- R. Siegfried, "Results of MSG-086 on Simulation Interoperability and Recommendations for a Scenario Development Process", Presentation at NATO CAX Forum 2012, Rome, Italy.
- R. Siegfried, A. Laux, M. Rother, D. Steinkamp, G. Herrmann, J. Lüthi and M. Hahn, "Components and Reuse in Scenario Development Processes for Distributed Simulation Environments", SISO 2012 Fall SIW, Orlando, FL, USA, Paper 12F-SIW-046.
- R. Siegfried, A. Laux, M. Rother, D. Steinkamp, G. Herrmann, J. Lüthi and M. Hahn, "Scenarios in Military (Distributed) Simulation Environments", SISO 2012 Spring SIW, Paper 12S-SIW-014.

D.2 POSTERS

A poster of MSG-086 was displayed regularly at ITEC, I/ITSEC and SISO events.

D.3 TRI-FOLDS

Tri-folds of MSG-086 were distributed regularly at ITEC, I/ITSEC and SISO events.

REPORT DOCUMENTATION PAGE

1. Recipient's Reference	**2. Originator's References**	**3. Further Reference**	**4. Security Classification of Document**
	STO-TR-MSG-086-Part-I AC/323(MSG-086)TP/562	ISBN 978-92-837-0201-2	PUBLIC RELEASE

5. Originator	Science and Technology Organization North Atlantic Treaty Organization BP 25, F-92201 Neuilly-sur-Seine Cedex, France
6. Title	Simulation Interoperability
7. Presented at/Sponsored by	Final Report of Task Group MSG-086.

8. Author(s)/Editor(s) Multiple	**9. Date** January 2015
10. Author's/Editor's Address Multiple	**11. Pages** 198

12. Distribution Statement	There are no restrictions on the distribution of this document. Information about the availability of this and other STO unclassified publications is given on the back cover.
13. Keywords/Descriptors	DSEEP Levels of conceptual interoperability model Simulation Interoperability

14. Abstract

Simulation interoperability is most often considered to be a technical problem (e.g., "Does every system support the same HLA standard and which RTI should we use?") or semantic problem (e.g., "How do we map different object models onto each other?"). Yet, reliably achieving high-quality distributed simulations requires to also consider more abstract questions regarding pragmatic, dynamic and conceptual issues (e.g., "How is aggregation and disaggregation treated by different simulation systems?", "Do different simulation systems interpret Rules of Engagement in the same way?").

In 2009, the Exploratory Team ET-027 identified 63 major issues which limit "true" simulation interoperability. In 2010, MSG-086 "Simulation Interoperability" was tasked to analyze the ET-027 interoperability issues in order to recommend and prototype information products augmenting the DSEEP to mitigate or obviate the identified issues.

MSG-086 analyzed 46 interoperability issues in total and categorized these issues into nine issue groups. Detailed descriptions of all interoperability issues have been developed specifying the original problem, its influence on simulation interoperability as well as possible solution approaches and already existing work in the specific problem area. Based on the identified interoperability issues, MSG-086 developed a "Guideline on Scenario Development for (Military) Simulation Environments" to overcome scenario-related interoperability issues.

NORTH ATLANTIC TREATY ORGANIZATION

SCIENCE AND TECHNOLOGY ORGANIZATION

BP 25
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@cso.nato.int

DIFFUSION DES PUBLICATIONS
STO NON CLASSIFIEES

Les publications de l'AGARD, de la RTO et de la STO peuvent parfois être obtenues auprès des centres nationaux de distribution indiqués ci-dessous. Si vous souhaitez recevoir toutes les publications de la STO, ou simplement celles qui concernent certains Panels, vous pouvez demander d'être inclus soit à titre personnel, soit au nom de votre organisation, sur la liste d'envoi.

Les publications de la STO, de la RTO et de l'AGARD sont également en vente auprès des agences de vente indiquées ci-dessous.

Les demandes de documents STO, RTO ou AGARD doivent comporter la dénomination « STO », « RTO » ou « AGARD » selon le cas, suivi du numéro de série. Des informations analogues, telles que le titre est la date de publication sont souhaitables.

Si vous souhaitez recevoir une notification électronique de la disponibilité des rapports de la STO au fur et à mesure de leur publication, vous pouvez consulter notre site Web (http://www.sto.nato.int/) et vous abonner à ce service.

CENTRES DE DIFFUSION NATIONAUX

ALLEMAGNE
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7, D-53229 Bonn

BELGIQUE
Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research
for Defence, National STO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30, 1000 Bruxelles

BULGARIE
Ministry of Defence
Defence Institute "Prof. Zvetan Lazarov"
Blvd "Totleben" 34
1606 Sofia

CANADA
DSIGST 2 – Gérant des publications (S et T)
Recherche et développement pour la défense Canada
Ministère de la Défense nationale
101 Colonel By Drive, 6 CBS – F002
Ottawa, Ontario K1A 0K2

DANEMARK
Danish Acquisition and Logistics Organization
(DALO)
Lautrupbjerg 1-5
2750 Ballerup

ESPAGNE
SDGTECIN (DGAM)
C/ Arturo Soria 289
Madrid 28033

ESTONIE
Estonian Ministry of Defence
Estonian National Coordinator for NATO STO
Sakala 1
Tallinn 15094

ETATS-UNIS
Defense Technical Information Center
8725 John J. Kingman Road
Fort Belvoir, VA 22060-6218

FRANCE
O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc
BP 72
92322 Châtillon Cedex

GRECE (Correspondant)
Defence Industry & Research General
Directorate, Research Directorate
Fakinos Base Camp, S.T.G. 1020
Holargos, Athens

HONGRIE
Hungarian Ministry of Defence
Development and Logistics Agency
P.O.B. 25
H-1885 Budapest

ITALIE
Centro Gestione Conoscenza
Secretariat General of Defence
National Armaments Directorate
Via XX Settembre 123/A
00187 Roma

LUXEMBOURG
Voir Belgique

NORVEGE
Norwegian Defence Research
Establishment
Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

PAYS-BAS
Royal Netherlands Military
Academy Library
P.O. Box 90.002
4800 PA Breda

POLOGNE
Centralna Biblioteka Wojskowa
ul. Ostrobramska 109
04-041 Warszawa

PORTUGAL
Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

REPUBLIQUE TCHEQUE
Vojenský technický ústav s.p.
CZ Distribution Information Centre
Mladoboleslavská 944
PO Box 18
197 06 Praha 9

ROUMANIE
Romanian National Distribution
Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6
061353 Bucharest

ROYAUME-UNI
Dstl Knowledge and Information
Services
Building 247
Porton Down, Salisbury SP4 0JQ

SLOVAQUIE
Akadémia ozbrojených síl gen.
M.R. Štefánika, Distribučné a
informačné stredisko STO
Demänová 393
031 06 Liptovský Mikuláš 6

SLOVENIE
Ministry of Defence
Central Registry for EU & NATO
Vojkova 55
1000 Ljubljana

TURQUIE
Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi
Başkanlığı
06650 Bakanliklar – Ankara

AGENCES DE VENTE

The British Library Document
Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
ROYAUME-UNI

Canada Institute for Scientific and
Technical Information (CISTI)
National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa, Ontario K1A 0S2
CANADA

Les demandes de documents STO, RTO ou AGARD doivent comporter la dénomination « STO », « RTO » ou « AGARD » selon le cas, suivie du numéro de série (par exemple AGARD-AG-315). Des informations analogues, telles que le titre et la date de publication sont souhaitables. Des références bibliographiques complètes ainsi que des résumés des publications STO, RTO et AGARD figurent dans le « NTIS Publications Database » (http://www.ntis.gov).

NORTH ATLANTIC TREATY ORGANIZATION

BP 25
F-92201 NEUILLY-SUR-SEINE CEDEX • FRANCE
Télécopie 0(1)55.61.22.99 • E-mail mailbox@cso.nato.int

SCIENCE AND TECHNOLOGY ORGANIZATION

DISTRIBUTION OF UNCLASSIFIED STO PUBLICATIONS

AGARD, RTO & STO publications are sometimes available from the National Distribution Centres listed below. If you wish to receive all STO reports, or just those relating to one or more specific STO Panels, they may be willing to include you (or your Organisation) in their distribution.

STO, RTO and AGARD reports may also be purchased from the Sales Agencies listed below.

Requests for STO, RTO or AGARD documents should include the word 'STO', 'RTO' or 'AGARD', as appropriate, followed by the serial number. Collateral information such as title and publication date is desirable.

If you wish to receive electronic notification of STO reports as they are published, please visit our website (http://www.sto.nato.int/) from where you can register for this service.

NATIONAL DISTRIBUTION CENTRES

BELGIUM
Royal High Institute for Defence – KHID/IRSD/RHID
Management of Scientific & Technological Research for Defence, National STO Coordinator
Royal Military Academy – Campus Renaissance
Renaissancelaan 30
1000 Brussels

BULGARIA
Ministry of Defence
Defence Institute "Prof. Zvetan Lazarov"
Blvd "Totleben" 34
1606 Sofia

CANADA
DSTKIM 2 – S&T Publications Manager
Defence Research and Development Canada
Department of National Defence
101 Colonel By Drive, 6 CBS – F002
Ottawa, Ontario K1A 0K2

CZECH REPUBLIC
Vojenský technický ústav s.p.
CZ Distribution Information Centre
Mladoboleslavská 944
PO Box 18
197 06 Praha 9

DENMARK
Danish Acquisition and Logistics Organization (DALO)
Lautrupbjerg 1-5
2750 Ballerup

ESTONIA
Estonian Ministry of Defence
Estonian National Coordinator for NATO STO
Sakala 1
Tallinn 15094

FRANCE
O.N.E.R.A. (ISP)
29, Avenue de la Division Leclerc – BP 72
92322 Châtillon Cedex

GERMANY
Streitkräfteamt / Abteilung III
Fachinformationszentrum der Bundeswehr (FIZBw)
Gorch-Fock-Straße 7
D-53229 Bonn

GREECE (Point of Contact)
Defence Industry & Research General Directorate, Research Directorate
Fakinos Base Camp, S.T.G. 1020
Holargos, Athens

HUNGARY
Hungarian Ministry of Defence
Development and Logistics Agency
P.O.B. 25
H-1885 Budapest

ITALY
Centro Gestione Conoscenza
Secretariat General of Defence
National Armaments Directorate
Via XX Settembre 123/A
00187 Roma

LUXEMBOURG
See Belgium

NETHERLANDS
Royal Netherlands Military Academy Library
P.O. Box 90.002
4800 PA Breda

NORWAY
Norwegian Defence Research Establishment, Attn: Biblioteket
P.O. Box 25
NO-2007 Kjeller

POLAND
Centralna Biblioteka Wojskowa
ul. Ostrobramska 109
04-041 Warszawa

PORTUGAL
Estado Maior da Força Aérea
SDFA – Centro de Documentação
Alfragide
P-2720 Amadora

ROMANIA
Romanian National Distribution Centre
Armaments Department
9-11, Drumul Taberei Street
Sector 6
061353 Bucharest

SLOVAKIA
Akadémia ozbrojených síl gen M.R. Štefánika, Distribučné a informačné stredisko STO
Demänová 393
031 06 Liptovský Mikuláš 6

SLOVENIA
Ministry of Defence
Central Registry for EU & NATO
Vojkova 55
1000 Ljubljana

SPAIN
SDGTECIN (DGAM)
C/ Arturo Soria 289
Madrid 28033

TURKEY
Milli Savunma Bakanlığı (MSB)
ARGE ve Teknoloji Dairesi Başkanlığı
06650 Bakanliklar – Ankara

UNITED KINGDOM
Dstl Knowledge and Information Services
Building 247
Porton Down, Salisbury SP4 0JQ

UNITED STATES
Defense Technical Information Center
8725 John J. Kingman Road
Fort Belvoir, VA 22060-6218

SALES AGENCIES

The British Library Document Supply Centre
Boston Spa, Wetherby
West Yorkshire LS23 7BQ
UNITED KINGDOM

Canada Institute for Scientific and Technical Information (CISTI)
National Research Council Acquisitions
Montreal Road, Building M-55
Ottawa, Ontario K1A 0S2
CANADA

Requests for STO, RTO or AGARD documents should include the word 'STO', 'RTO' or 'AGARD', as appropriate, followed by the serial number (for example AGARD-AG-315). Collateral information such as title and publication date is desirable. Full bibliographical references and abstracts of STO, RTO and AGARD publications are given in "NTIS Publications Database" (http://www.ntis.gov).

ISBN 978-92-837-0201-2