Development of an Air Operation eXtension with the (future) C2SIM Standard

Magdalena Dechand, Lukas Sikorski, Irmtrud Trautwein
Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE
Fraunhofer Str. 20 53343 Wachtberg
GERMANY

magdalena.dechand@fkie.fraunhofer.de
lukas.sikorski@fkie.fraunhofer.de
irmtrud.trautwein@fkie.fraunhofer.de

Bruno Gautreau
Airbus Defence and Space
1 boulevard Jean Moulin
78996 Elancourt CEDEX
FRANCE

bruno.gautreau@airbus.com

Eric Bouvier
DIGINEXT
22 avenue Galilée
92350 Le Plessis Robinson
FRANCE

eric.bouvier@diginext.fr

Lionel Khimeche
DGA
16 bis, avenue Prieur de la Côte d'Or
94114 Arcueil
FRANCE

lionel.khimeche@dga.defense.gouv.fr

Keywords: Interoperability, command and control, simulation, C2SIM, air operation

ABSTRACT:

The MSG 145 group is working on the Operationalization of Standardized C2-Simulation

Interoperability, side-by-side with the Simulation Interoperability Standards Organization (SISO) Product Development Group (PDG) which is developing the C2 System to Simulation System Interoperation (C2SIM) standard. The C2SIM standard will improve upon and replace the currently existing combination of Military Scenario Definition Language (MSDL) and Coalition Battle Management Language (C-BML) SISO standards, with a modular approach enabling users of different domains to define their needs in extensions. The main elements of the C2SIM standard are a core logical data model (created as ontology), an extension *mechanism, and several extension ontologies (e.g., Land Operation eXtension). The C2SIM standard also supports the transformation of the exchanges (tasking/reporting messages, initialization of C2 and simulation systems) into Extensible Markup Language (XML) schemata.*

This paper describes the work done by a French-German subgroup of the MSG-145 to implement a new C2SIM extension for an Air operation domain. The work includes the operational analysis of an Air operation scenario, the design of the logical model with ontologies, the generation of the XML schemata, and their implementation in the systems. This paper concludes with lessons learned, and contributes to the evaluation of the operationalization and the extension of the C2SIM standard.

1.0 INTRODUCTION

The involvement of modeling and simulation systems (M&S systems) into training and eventually into operations would add value to both, it would increase the realism and complexity of training sessions, and it would offer enhanced tools for mission planning, for decision making, and for after action analysis. A Simulation Interoperability Standards Organization (SISO) Product Development Group (PDG) is developing a standard for Command and Control System to Simulation System Interoperation (C2SIM) [13] in order to support this approach and to make it possible. The SISO efforts, as evoked by [5], have always been supported by NATO activities [7], currently by the NATO MSG-145 "Operationalization of Standardized C2-Simulation Interoperability". As it has already been proposed in preceding papers [3, 11], this C2SIM standard will replace and improve upon the currently existing Military Scenario Definition Language (MSDL; SISO-STD007-2008) [17, 19] and Coalition Battle Management Language (C-BML; SISO-STD-011-2014) [16] SISO standards. As also has been presented and discussed, the current work in the PDG is specifying a core logical data model ontology, in short a core ontology [2, 11], and a tool to transform the relevant parts of that ontology into an Extensible Markup Language (XML) schema [3]. That schema will serve as a medium for the exchange of messages among systems that constitute C2SIM coalitions. The schema has to support initialization as well as military information exchange during a simulation run, e.g. the exchange of orders to command simulated and real units or the exchange of reports sent by those units.

The paper at hand describes work done by a French-German subgroup of the NATO MSG-145 to implement a C2SIM extension that covers the Air operation domain. The work includes the operational analysis of an Air operation scenario, the design of the ontology extension, the evaluation of a schema generation process when applied to military messages specific for Air operations, and the implementation of the messages within C2 systems and simulation systems. In the following, we will first explain how our work is embedded in the general context and the SISO PDG efforts (section 2). In particular, we will discuss ontologies (subsection 2.1) and their role in the C2SIM context (subsection 2.2). From that, we will discuss automatic schema generation out of ontology representations (subsection 2.3). In a second part (section 3), we will discuss the specifics of military communications in the Air operation domain (subsection 3.1), how these specifics can be represented in an ontology extension (subsection 3.2), and the consequences this has for schema generation (subsection 3.3). The paper continues with the collection of lessons learned from this work (section 4) and concludes with a summary and with the presentation of projected follow-on activities (section 5).

2.0 THE C2SIM CONTEXT

A collection of C2 systems and simulation systems constitutes a C2SIM coalition if the systems can seamlessly interact with each other using the C2SIM standard. For example, a system of two C2 systems and two simulation systems might be used in a training session in which a trainee uses one of the C2 systems to command a simulated force. That force might be simulated on one of the simulation systems. The second C2 system then can be used by the trainer to command opposing simulated forces whose actions are simulated on the second simulation system. In that scenario, trainee and trainer send orders to their respective forces and expect to get reports back. Therefore, the C2 systems need to interact with the respective simulation systems, and the simulation systems need to interact in order to show both, the trainee and the trainer, pictures of the same situation. Of course, the pictures will differ for both sides. In particular, the trainee will not see the whole ground truth of the opposing forces (this is not a chess game). The work by the SISO PDG and the supporting NATO research group, in general, and the work by the French-German subgroup, in particular, focuses on the interaction between the C2 side and the simulation side. It is about the exchange of military communication. A correct exchange might easily be granted if the participating C2 system(s) and simulation system(s) have been developed by the same maker or at least are systems run by the same nation. However, there are many nations in NATO, and each nation has its own systems. Thus, the challenge is interoperable exchange across multiple diverse systems developed by multiple nations. As it has already been mentioned, such an interoperable exchange had been granted by the C-BML standard (SISO-STD-011-2014) and the interoperable initialization had been granted by the MSDL standard (SISO-STD-007-2008). However, the cooperation of MSDL and C-BML turned out to be difficult. In contrast, the current C2SIM provides an integrated solution. Besides, C2SIM defines an extensible standard. In addition, it had adopted ontology, and more specifically the Web Ontology Language (OWL) [20], as the C2SIM main representation. The latter leads us to the question: what are the benefits of an ontological representation?

2.1 Ontologies and the Web Ontology Language (OWL)

"Ontology", according to the Encyclopædia Britannica, is "the theory or study of being as such; i.e., of the basic characteristics of all reality." Rudolf Goclenius the elder coined that philosophical term in the 17th century for what had originally been "metaphysics" in the ancient times [10].

In information science, ontologies are used to represent knowledge. According to the well-known dictum by Tom Gruber, "[a]n ontology is an explicit specification of a conceptualization" [6]. In short, in information science, ontologies are used to unambiguously define the vocabulary of a domain and to express (semantic) relations among the defined terms. Since in information science it is possible to establish ontologies for different domains, the term "ontology" can be used as a plural. This is in contrast to the philosophical "ontology" which is a singular-only noun.

The World Wide Web Consortium (W3C) offers a large palette of techniques to describe and define different forms of ontologies in a standard format. These include Resource Description Framework (RDF), RDF Schemata, and Web Ontology Language (OWL). An OWL ontology consists of a set of axioms which place constraints on sets of individuals (called "classes") and the types of relationships permitted between them. The data described by an ontology, or an "OWL expression", is interpreted as a set of "individuals" and a set of "property assertions" which relate these individuals to each other. Furthermore, because the OWL axioms provide semantics and have mathematical "roots" in set theory (with axioms for union, intersection, and complement of classes, i.e. sets of individuals); use of OWL allows systems to infer additional information based on the data explicitly provided. The automated reasoning capabilities of OWL are based on this inference mechanism.

2.2 Ontologies in the C2SIM Context

The idea of using ontology to support C2SIM and the interoperability among participating systems dates back at least to 2005, cf. [1], "but was postponed for the benefit of first developing an exchange language based on formal syntax (CBML)" [3]. After the C-BML standard had passed, the idea was revived in order to examine its potential, cf. [2]. In principle, there are two domains from which knowledge, when formally represented in an ontology, may provide benefits in the C2SIM context. These are military operations as well as military communication. First, knowledge about military operations accessible by a C2 system can improve for example automated decision support. Therefore, it also can improve C2SIM applications like staff training. For example, if logistic states for the units of one's own forces would be known and exploitable for a C2, the system could warn a commander of possible overreaches when tasking forces. Second, there is the domain of military communication. Military communication follows rules and formats recorded in specific regulations and instructions. These rules and regulations can be represented within a domain ontology covering military communication; e.g., for every kind of message it can be represented by a Boolean value whether the receipt of the message has to be acknowledged. The benefit would be an increase of realism in C2SIM; e.g., simulation systems could generate acknowledgements if a simulated unit receives a message from the user of the connected C2 system of a kind that is marked to be answered by an acknowledgement. The latter idea has already been proposed in Singapogu et al. [15]. Therefore, the PDG's core ontology provides the class "Message" with some subclasses where knowledge about military communication can be stored. In general, it must be acknowledged that we have barely scratched the surface on the use of ontologies for knowledge representation, automated reasoning, information linking, and information query. But, in C2SIM, we opened the door in the secure knowledge that there are numerous open source and commercial tools available that will enable rapid adoption and exploitation of ontologies in the military domain.

2.3 From Ontologies to Schemata

Ontologies represent knowledge. Schemata allow the exchange of messages. From a linguistic point of view, good ontologies have to represent the necessary knowledge correctly. They constitute semantics. In contrast, good schemata transmit their messages in a correct form. They constitute syntax. Nevertheless, the receiver of a message should understand the message's meaning as intended by the sender. Therefore, schemata need to refer back to ontologies. In C2SIM, we consider this insight: we generate the schemata automatically out of the ontology. Blais et al. presented the respective tool. Their paper [3] provides details about schema generation using the tool. With the paper at hand, we will focus on the details and the specifics that hold for the Air operation domain and its specific military operations. However, before we will unfurl this topic in the following section, we would like to stress that automatic schema generation out of an ontology does not mean that everything that is represented in the ontology will be reflected in the schemata adjusted for specific applications. The schemata only need to enable initialization and message exchange. Thus, only those ontological representations that are needed for these processes need to appear in the schemata. All other ontological representations, may they be mandatory for other ontological processes or not, can be omitted.

3.0 AIR OPERATIONS IN C2SIM

Most C2SIM demonstrations of the past focused on land operations, with sometimes enhancements and complements to deal with maritime [14] and/or air operation [4, 8, 18]. Beyond the mentioned contributions, modelling air operations in C2SIM needs further efforts. In order to cover air operations' aspects, we first have to consider the specifics of air operations in general and the specifics of military communication in air operations in particular (subsection 3.1). Having identified those specifics, we have to represent them in an ontology extension for air operations (subsection 3.2). Last but not least, we have to take a look at the automated schema generation (subsection 3.3). It needs some add-ons, so that the resulting schemata allow air operations' specific communications.

3.1 The Specifics of Air Operations

Air-mobile forces excel in tactical mobility and speed. They are characterized by their dynamics. Therefore, communication with air-mobile forces differs from communication on the army side. For example, the army command can track ground forces if these forces send position reports providing their current positions, e.g., in the form of coordinates, periodically. In contrast, due to the already mentioned high dynamics, aircraft are tracked by automated position updates, generated e.g. by an airborne radar picket system, like an AWACS.

The resulting tracks are branded with track numbers. These numbers then allow referring to an air-mobile force or even to a single aircraft in further communication. The communication in air operations, i.e. the exchange of messages including orders, requests, and reports, uses data link standards such as the NATO standards TDL (Tactical Data Link) 11 and TDL 16, also called “Link 11” and “Link 16”, respectively. Link 11 is described in STANAG 5511 and the US MIL-STD-6011 and will be replaced by Link 22. Link 16 is described in STANAG 5516 and the US MIL-STD 6016. In our approach for air operations, the focus is on having generic messages for TDL networks in general, as suggested in Svensson et al. [18].

The scenario is based on a fictional operation executed by the Future Combat Air System, to destroy hostile air defense (Destroy Enemy Air Defense / DEAD scenario). For such operations, the fighters, the drones swarms, and the AWACS are all connected through a TDL network (Link 16, or maybe a new one in the future), and they exchange messages for situation awareness (tracks, own positions, targets) and for command and control. Our experimentation involves the simulation DirectCGF (DIGINEXT) and the C2 surrogate C2LG (FKIE). DirectCGF simulates all the entities on the battlefield. C2LG plays the command and control function of the AWACS on a subset of the entities. All other AWACS tasks, e.g. surveillance and the command and control of other entities, are simulated by DirectCGF.

In order to create realism in C2SIM also in the air operation context, TDL messages have to complement the standard message portfolio as used in C2SIM. In addition, the secure data network that allows the exchange of TDL messages in air operations needs to be considered in the simulation. Consequently, we represented the TDL messages as well as the network, ontologically, as explained in the following subsection.

3.2 Air Operations and Air Operation Communication Represented Ontologically

In order to represent air operations, ontologically, we built a respective ontology extension to the ontology that already has merged the C2SIM core with its standard military extension (SMX). The core ontology provides a taxonomy with classes subsumed under the upper classes “C2SIMContent”, “InitializationConcept”, and “MessageConcept”. These split up into various subclasses that inherit properties from their superclasses as restrictive definitions of them.

In the following, we explain how we have built the air operation extension by discussing the construction of three example classes all of which are crucial for air operations. On the platform side, no class enhancements are needed to add to our air operation extension. The various platform types will be defined by new instances of the C2SIM classes “DISEntityType” and “APP6-SIDC” in the scenario initialization data.

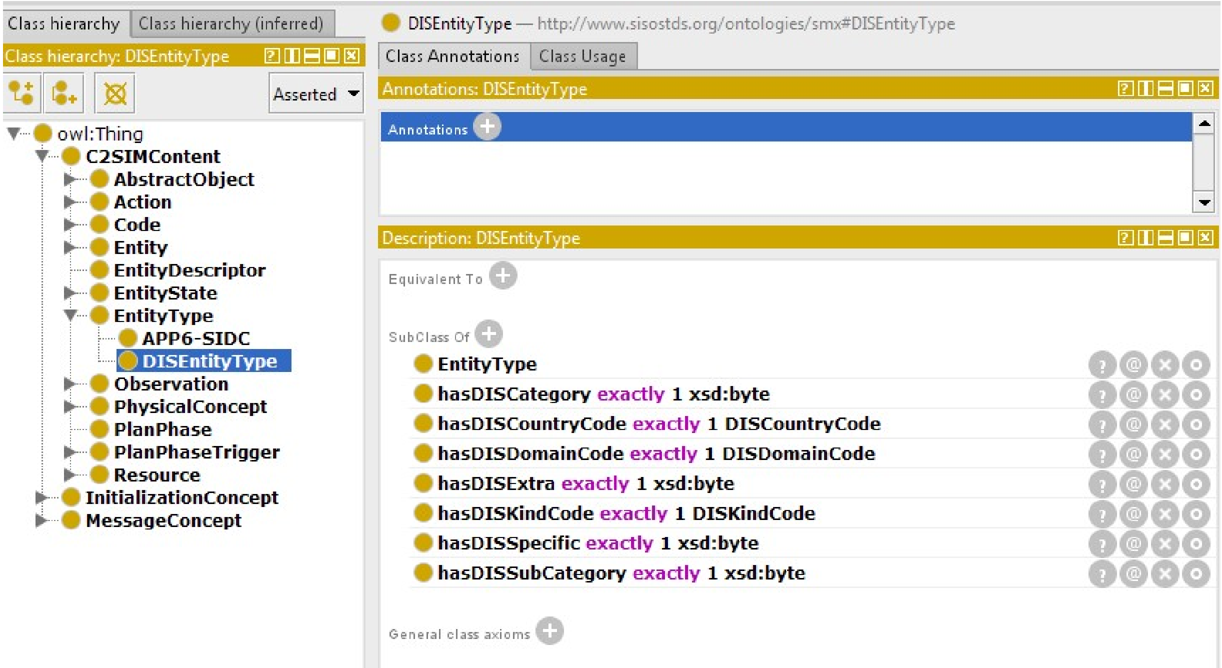

FIGURE 1: Snapshot of the part of the ontology's taxonomy that is about entity types; the snapshots are taken from the implementation of the ontology in Protégé [9]

The C2SIM core ontology already has the class "CommunicationNetwork". It can be used to represent a Tactical Data Link network in the scenario. During initialization, an instance of that class is generated by applying what is represented in the class "TDLNetworkParticipantDefinition", subclass of "InitializationConcept", cf. figure 2. "TDLNetworkParticipantDefinition" has the property "hasCommunicationNetworkUUID" to refer to a network instance and the properties "hasSubjectEntity" and "hasTDLTrackNumber" to assign an entity together with its track number to the network.

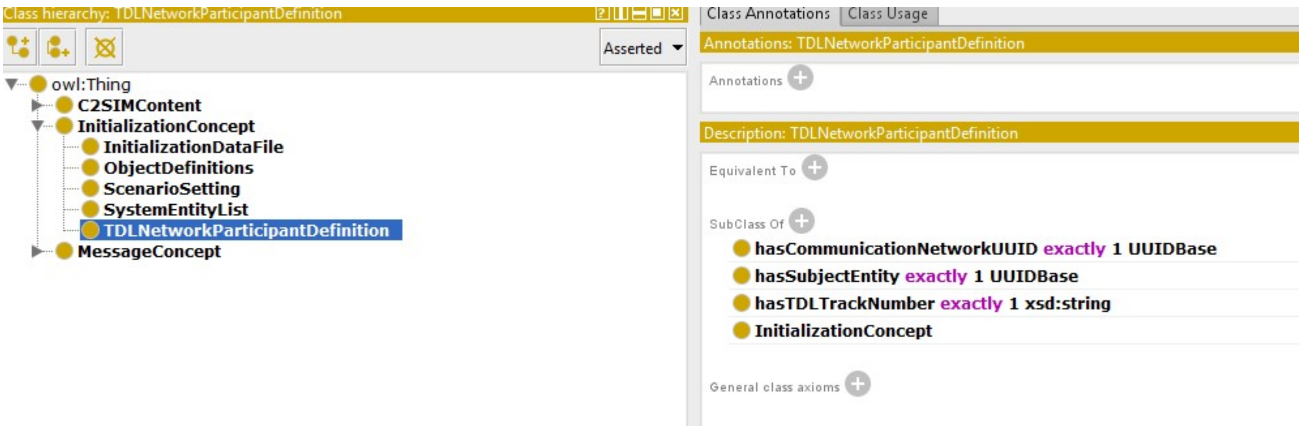

FIGURE 2: Snapshot of the part of the ontology's taxonomy that represents knowledge to be applied during initialization

Since C2SIM is an approach to substitute C-BML (besides MSDL), messaging remains crucial. The C2SIM core ontology provides a general structure of messages that allows adding subclasses for specific messages for different domains.

Specific messages come with restrictions, e.g. how orders and reports should be structured and which elements or pieces of information they should include. In the French-German air operation extension, we paid attention to abide by the structure as provided by the core. As TDL includes specific information and extend the domain formerly covered by C-BML, we introduced the class "TDLHeader" as well as specific classes under "ReportContent", "RequestBody", and "OrderBody"

(in the core ontology, as published by the PDG, "ReportBody" refers to "ReportContent"). In the following, we will illustrate our approach by examples. Our first example is the TDL PPLI. Its representation in the ontology is given in figure 3.

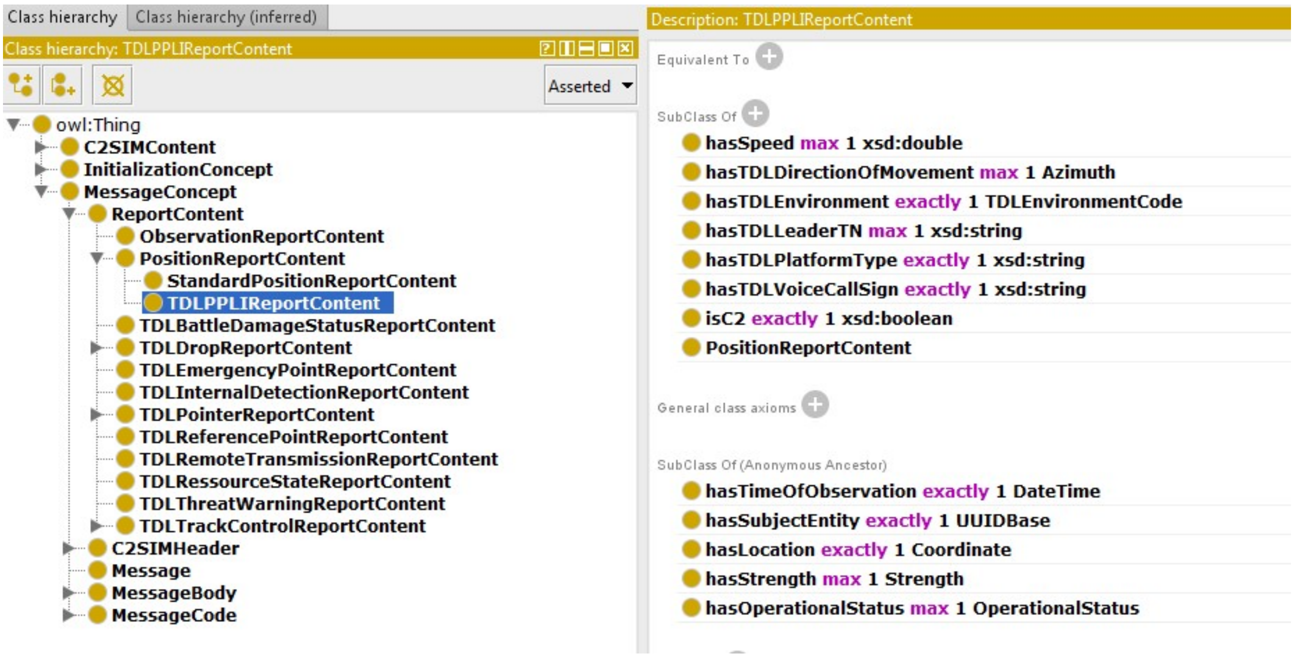

FIGURE 3: Snapshot of the class "TDLPPLIReportContent"

As can be seen on the left side of figure 3, TDL PPLI is represented as a position report. PPLI is the abbreviation for "precise participant location and identification" meaning that the sender reports about its own position, with additional information when necessary. The class for the body and thus for the content of a TDL PPLI ("TDLPPLIReportContent") is a subclass of "PositionReportContent" because it needs more property restrictions than "PositionReportContent" from the core to be defined. However, this expansion leads then to an artificial construction of "StandardPositionReportContent" as a complement to "TDLPPLIReportContent" as all classes need at least one sister class with at least one additional property to fulfill ontology principles. "TDLPPLIReportContent" (just as its sister class) inherits all properties of "PositionReportContent", including mandatory statements about the subject whose position is reported, of the responding location and of the respective point in time (figure 2, right side, lower part). "TDLPPLIReportContent" has some properties that are specific and not inherited from "PositionReportContent". These properties derive from the TDL PPLI's fields and are named respectively. For example, the property that derives from the TDL PPLI's field "Environment" is named "hasTDLEnvironment". In general, following conventional ontological naming, a property's name starts with “has” or “is”. In order to indicate that the property has been added for/by this extension, “TDL” is included in the name followed by the respective field’s name. If there was already a property as part of the C2SIM core ontology (or existing extensions) that corresponds to the field, that property would have been used. In these cases, no “TDL” tag would be in the name. The properties that derive from the TDL PPLI include the subject’s speed (“hasSpeed”), its direction of movement (“hasTDLDirectionOfMovement”), and the track number of its leader (“hasTDLLeaderTN”) whereas its own track number is provided in the report’s header. Some more properties are in the body. They can be used to provide more information about the sender, but they can be seen as superfluous since the subject’s id and track number are given in the report’s header. Nevertheless, the properties are included so that all fields of the TDL are covered. The superfluous properties are “isC2”, “hasTDLPlatformType”, and “hasTDLVoiceCallSignal”. Another property is the already mentioned

“hasTDLEnvironment”. It always bears the value “AIR” in our context.

As a second TDL message example, we would like to discuss the “Mission Assignment” message of the TDL domain. In the C2SIM context, it’s semantically the same thing as an order message, sent from a commander to the subordinated aircraft, with a single task containing the details of the “air mission”. So, we decided to create the class “TDLMissionTask”, as a subclass of the C2SIM “Task”, and use the C2SIM order message.

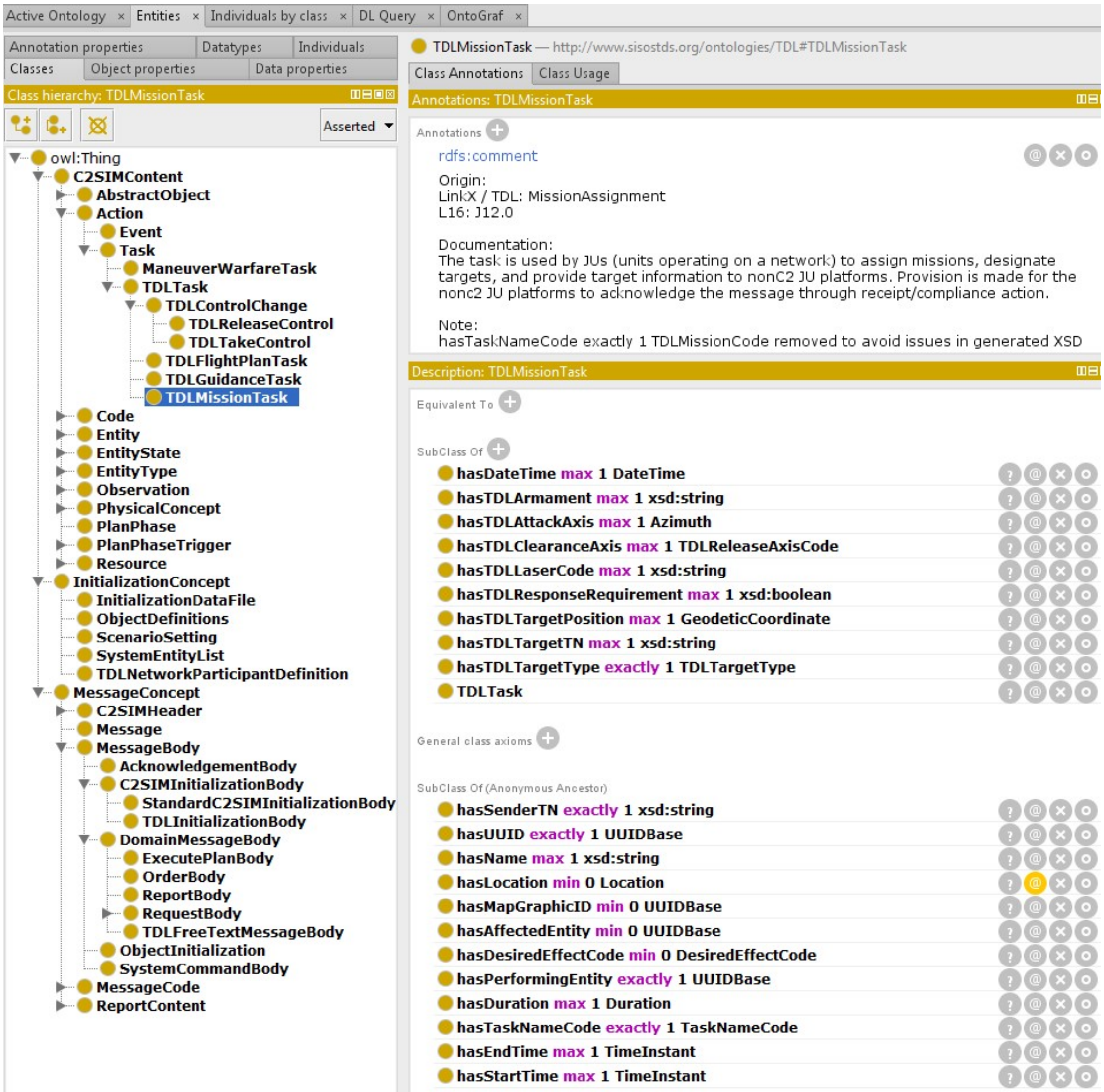

FIGURE 4: Snapshot of the class “TDLMissionTask”

As depicted in Figure 4, “TDLMissionTask” is not directly a subclass of “Task”. We’ve added also the “TDLTask” class, in order to have a common root for all kind of tasks exchanged over TDL networks. The other TDL tasks are used for guidance commands (take a new direction, take a new altitude, speed, etc.), or to order to follow a flight plan (points with dates/times). “TDLMissionTask” inherits all the properties of the C2SIM “Task” class, which includes a task UUID (property “hasUUID”), a reference to the taskee (property “hasPerformingEntity”), and the mission verb (property “hasTaskNameCode”). “TDLMissionTask” is defined with new property restrictions: an instance of “TDLMissionTask” is an instance having one property “hasTDLTargetType” (which should define the type of the target), and having maximum one property “hasTDLTargetTN” (which should define the Track Number of the target), and having maximum one property “hasAttackAxis” (which should define the direction of the aircraft before the aircraft launches its missiles), and so on. “TDLMissionTask” represents all TDL missions with verbs like e.g. “attack”, “escort” or “cover”. As mentioned before, “TDLMissionTask” inherits the property “hasTaskNameCode” and the corresponding range from its superclass “Task”. However, the values that “TDLMissionTask” permits for TDL tasks constitute only a subset of all the values of

"TaskNameCode". We took into consideration restricting the range of "hasTaskNameCode" for "TDLTask" to those codes that refer to a TDL task. However, this would lead to a double occupancy of that property resulting from inheritance, meaning "TDLMissionCode" would be defined by having a TaskNameCode exactly 1 TaskNameCode and exactly 1 TDLTaskCode. And this assertion would be wrong. Therefore, we did not restrict this property further.

3.4 Exchanging TDL Messages: From Ontology to Message

```
<?xml version="1.0" encoding="UTF-8"?>
<Message xmlns="http://www.sisostds.org/schemas/C2SIM/1.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.sisostds.org/schemas/C2SIM/1.1 file:///C:/C2SIM/Comelec/Airbus/C2SIM_SMX_LOX_TDL.xsd">
	<C2SIMHeader>
		<StandardC2SIMHeader>
			<CommunicativeActTypeCode>Inform</CommunicativeActTypeCode>
			<MessageID>cd1d895f-ea3e-4858-906d-6abc93907a5d</MessageID>
			<Protocol>C2SIM</Protocol>
			<ProtocolVersion>1.1</ProtocolVersion>
			<SendingTime>2019-07-30T13:36:30:Z</SendingTime>
			<FromSendingSystem>C2LG</FromSendingSystem>
			<ToReceivingSystem>DirectCGF</ToReceivingSystem>
			<ConversationID>d10f9a43-e4ea-4edc-9ede-ebde7d5e1eb2</ConversationID>
		</StandardC2SIMHeader>
	</C2SIMHeader>
	<MessageBody>
		<DomainMessageBody>
			<OrderBody>
				<FromSender>00000000-0001-0001-1000-000000000000</FromSender>
				<ToReceiver>00000000-0000-0001-2000-000000000000</ToReceiver>
				<Task>
					<TDLTask>
						<TDLMissionTask>
							<TaskNameCode>ATTACK</TaskNameCode>
							<PerformingEntity>00000000-0000-0001-2000-000000000000</PerformingEntity>
							<UUID>cd1d895f-ea3e-4858-906d-6abc93907a5e</UUID>
							<Name>TDLMissionTask_1</Name>
							<TDLMissionType>ATTACK</TDLMissionType>
							<TDLTargetType>AIR_DEFENSES</TDLTargetType>
							<TDLAttackAxis>
								<Angle>90</Angle>
							</TDLAttackAxis>
							<TDLClearanceAxis>LEFT</TDLClearanceAxis>
							<TDLTargetPosition>
								<Latitude>50.397138</Latitude>
								<Longitude>9.862633</Longitude>
							</TDLTargetPosition>
							<TDLArmament>missiles</TDLArmament>
							<TDLTargetTN>501</TDLTargetTN>
						</TDLMissionTask>
					</TDLTask>
				</Task>
				<IssuedTime>
					<DateTime>
						<IsoDateTime>2019-07-30T13:36:30:Z</IsoDateTime>
					</DateTime>
				</IssuedTime>
				<OrderID>cd1d895f-ea3e-4858-906d-6abc93907a5f</OrderID>
			</OrderBody>
		</DomainMessageBody>
	</MessageBody>
</Message>
```

LISTING 1: TDL mission order as completed XML structure conforming to the generated schema

Blais et al. [3] describe how schemata are generated from ontological representations. This approach starts with merging the C2SIM core with all required extensions. The schema is then transformed out of the merged ontology. In our case, this process has been applied to the C2SIM core ontology enhanced by the air operation extension. The result is an extensive schema that covers all kinds of messages, those for initialization as well as those for the different kinds of military communication. Since the ontology has the air operation extension incorporated, this includes TDL messages. In order to exchange information among the systems of a C2SIM coalition, the respective part of the general schema is filled in.

Listing 1 shows an example TDL mission order. In order to express it, the ontological representation of such an order / a task has been processed as described in [3]. During that process, the prefixes “has” and “is” that are used for properties in the ontology are clipped so that for example the ontological property “hasPerformingEntity” is transformed into the tag

“PerformingEntity”. The resulting schema then has been completed with dummy entries.

The listing of the TDL mission order assigns the UUID “00000001-0000-0001-2000-000000000000” to the order, the verb “ATTACK” to the mission, the aircraft UUID “00000000-0000-0001-2000-000000000000” that will perform the mission, the track number “501” of the target, the type “AIR_DEFENSES” of the target, etc.

4.0 LESSONS LEARNED

In this section, we point to some of the lessons we have learned when we developed the Air Operation extension for the C2SIM standard. The lessons refer to different steps from the process of designing an ontology extension that is to abide by the C2SIM structure and, in addition, refer to issues that evolved when we tuned the ontology extension so that the transformation process results in a valid schema.

- *Enumerations (how to deal with them)*

If an ontology, such as the C2SIM core ontology, defines classes being subclasses of others, an extension not always adds details like additional properties but also may limit existing properties in range. An example for that is “TaskNameCode” inherited form class “Task” to class “TDLTask”. “TaskNameCode” among others allows the value “withdraw” which, however, does not name a TDL task (cf. section 3). Most commonly, the value of a TDL field is one element out of a fixed set of an enumeration. The enumeration is provided by the TDL experts to support various data links such as Link 16 [18]. The elements can even correlate with classes from different fields in the core. To keep the TDL elements together building a single enumeration type, they are “subordinated” to one single class. We decided to model the enumeration values as individuals. This allows pointing directly to these specific enumeration values (the individuals) from different perspectives and classes. It is also possible to declare two individuals as being equivalent.

- *Property ranges, restrictions and leaf classes*

If C2SIM users add subclasses to the core superclasses, it is because they need to add more properties or to specify existing properties of a class.

Creating a new subclass with new properties to specify a class requires building a new sister class as well. This is the case in the “PPLIReportContent” that needs a new sister class “StandardPositionReportContent”. If we would have added only “TDLPPIReportContent” without this additional class “StandardPositionReportContent”, only “TDLPPIReportContent” could be used in the schema, meaning that it would have been impossible to exchange only a “PositionReportContent” element.

If a class inherits a property from its superclass, it might happen that the range for that property is restricted more closely in comparison to the property’s range as defined for the superclass. For example, the class “Task” has the property

“hasTaskNameCode”. Its range is an enumeration of the codes for all assumed tasks. The class “Task” has the class “TDLTask” as subclass. “TDLTask” inherits “hasTaskNameCode” but only some of the codes that constitute the enumeration list for “hasTaskNameCode” are codes for TDL tasks. Therefore, we would like to restrict the range of “hasTaskNameCode” for “TDLTask” to those codes that refer to a TDL task. However, if we do this, the schema generated out of the ontology will show two lines for “hasTaskNameCode” (first issue), one for the original enumeration and one for the restricted enumeration, and both having the same range (second issue).

The second issue arises because the range of the property restriction is not taken into account in the schema generation process. Only the range of the property definition is used. We get around this issue by creating several properties with different range definition, and use the appropriate property for the property restriction.

The first issue is a general problem, at least in the C2SIM context. We think that the schema needs not necessarily to be restricted as narrow as possible in the general case, at least not for message exchange. However, for future reasoning, restricting the ranges as narrow as possible seems to be expedient. Currently, in order to achieve a working prototype, we have not added restricted restrictions to respective subclasses such as “TDLTask”. Instead, we trust the users to generate meaningful messages, e.g. only to use codes for TDL tasks if the task is a TDL task.

- *XSD errors*

The last issue identified with the C2SIM schema generation process is that the generated schema (XSD) is not always compliant with the W3C standard. Sometimes, there are some errors in the generated schema that can be seen when the schema is loaded in an editor checking the XSD syntax (Eclipse editor, XML Spy). These errors are all about a name (example “TDLEnvironmentCode”) that cannot be resolved to a(n) ‘element declaration’ component within the XSD file. Currently, the solution is to add manually some instructions in the generated schema, such as:

```
<xs:schema […]>
      <xs:element name="TDLEnvironmentCode" type="TDLEnvironmentCodeType"/>
      […]
</xs:schema>
```

The reasons for these errors have not been evaluated, yet. We assume that this could be due to some OWL definitions in the modeling of the Air Operation extension that have not been implemented yet in the generation process. There is work in progress on this issue at the SISO C2SIM PDG, too.

5.0 FUTURE WORK

This paper discusses the process of developing an air operation ontology extension to the C2SIM core ontology so that the transformation of the entire ontology (core plus extension) results in a valid schema. As the work is still in progress, updated versions may be forthcoming. Even more, not all ontology capabilities have been exploited, yet. In particular, the PDG is planning to use reasoning capabilities of OWL and increase the realism of C2SIM. This also may entail further adjustments to the ontology.

In the paper, we argued that ontologies increase the realism of C2SIM. Two domains have been identified from which knowledge can be represented ontologically for such a purpose. The first domain is military operations. In order to exploit knowledge about military operations, reasoning is needed. We did not explore the chances of such an approach so far. In the current state, operational realism is granted by the simulation systems that are incorporated in a coalition of a C2SIM demonstration. But because we couldn’t use some of the OWL axioms related to the set theory (union, intersection, complement of) and some design constructs (subclass of multiple classes, equivalences, etc.) we cannot assess the reasoning capabilities that the current C2SIM standard will foster.

The second domain is military communication. With respect to this domain, the realism of C2SIM can be increased easily by implementing the ideas proposed by Singapogu et al. [15]. This can be done by adding two Boolean properties to the class “Message” coined “needsAcknowledgement” and “needsAnswer”. These properties come with the default value “false”. For the subclass “Order” the property “needsAcknowledgement” is set to “´true” and for the class “Request” both properties are set to “true”. Then procedures can be incorporated in the participating simulation systems’ procedures by which they react to the reception of a message so that they generate an acknowledgement whenever they receive a message that shows the value “true” for “needsAcknowledgement”. In addition, they will generate an appropriate committing answer whenever they receive a message that shows the value “true” for “needsAnswer”. As a result, simulated forces would react on receiving orders and requests like real forces do. But, this enhancement is not related to the use of the ontology and

OWL; it is related to the definition of a “new” property in the model. For example, this particular feature/property (needsAcknowledgement) already existed in CBML, with the XSD attribute “AcknowledgementRequested” in a CBML order.

Furthermore, and the SISO C2SIM PDG is aware of that, because some of the object properties have been replaced by data properties (with the UUID of the related object), most of the OWL reasoning features (inference of additional information based on the data provided in an “OWL expression”) might be unusable.

6.0 CONCLUSION

The paper at hand describes the first steps of the development of an Air Operation extension for C2SIM. Guided by a scenario of air operations, an extension of the C2SIM ontology had been fleshed out. The transformation tool provided by SISO’s C2SIM PDG (cf. [3]) had been applied to the result of the ontological work to generate a schema. The resulting schema is supposed to be valid for initialization and message exchange during C2SIM demonstrations that include air operations. We recognize that the current transformation tool is preliminary and is expected to be refined and further developed as C2SIM early-adopters work with the emerging standard and apply lessons learned. In this vein, we identified some suggested improvements in this paper from our investigation into an Air Operation extension. We also have noticed that close cooperation is mandatory among different specialists who are knowledgeable in many domains (OWL, XSD, C2SIM XSD generation, and TDL for our Air Operation scenario). But we also recognize that the C2SIM ontology extension design process is preliminary, and is expected to be refined and further developed, which could facilitate the C2SIM future-adopters work.

When the paper had been finished, the validity of the generated schema had been evaluated in test runs but when the paper is due for presentation, a French German demonstration will have revealed additional pros and cons.

7.0 REFERENCES

[1] Blais, C., Galvin, K. & Hieb, M.R.: “Coalition Battle Management Language (C-BML) Study Group Report,” Fall Simulation Interoperability Workshop, Orlando, FL, 2005.

[2] Blais, C., Reece, D. & Singapogu, S.: “From Information Description to Information Understanding: The Role of Ontology in Emerging SISO Standards,” Winter Simulation Innovation Workshop, Orlando, FL, February 2019.

[3] Blais, C., Schade, U., Sikorski, L., Wolski, M., Singapogu, S. & Gautreau, B.: “A Transformation Process for Generating an Extensible Markup Language (XML) Schema from a Formal Ontology for

Practical Application in C2SIM Implementations," Winter Simulation Innovation Workshop, Orlando, FL, February 2019.

[4] Bowers, A., Blais, C., Galvin, K., Gupton, K., Heffner, K., Khimeche, L. & Savasan, H.: "A Systems Engineering Approach to M&S Standards Development: Application to the Coalition Battle Management Language," Spring Simulation Interoperability Workshop, Orlando, FL, 2013.

[5] Carey, S., Kleiner, M., Hieb, M.R. & Brown, R.: "Standardizing Battle Management Language – A Vital Move towards the Army Transformation," Fall Simulation Interoperability Workshop, Orlando, FL, 2001.

[6] Gruber, T.R.: "A Translation Approach to Portable Ontology Specifications," in *Knowledge Acquisition*, 5/2 (1993), 199-220. 1993.

[7] Heffner, K., Brook, A., de Reus, N., Khimeche, L., Mevassvik, O.M., Pullen, M., Schade, U., Simonsen, J. & Gomez-Veiga, R.: "NATO MSG-048 C-BML Final Report Summary," Fall Simulation Interoperability Workshop, Orlando, FL, 2010.

[8] Khimeche, L., Bouvier, E. & Mounet, L.: "Tactical Data Links in a C2SIM Environment," Fall Simulation Innovation Workshop, Orlando, FL, 2018.

[9] Musen, M.A.: "The Protégé Project: A Look Back and a Look Forward. AI Matters," Association of Computing Machinery Specific Interest Group in Artificial Intelligence, 1(4), 2015.

[10] Prechtl, P. & Burkard, F.-P. (Eds.), *Metzler Philosophie-Lexikon: Begriffe und Definitionen*. 2nd Edition. Metzler, Stuttgart, DE, 1999.

[11] Pullen, J. M., Corner, D., Blais, C., Reece, D., Ruth, J. & Singapogu, S.: "Command and Control System to Simulation System Interoperation: Development of the C2SIM Standard," Winter Simulation Innovation Workshop, Orlando, FL, February 2019.

[12] Pullen, J.M., Corner, D., Wittman, R., Brook, A., Mevassvik, O.M. & Alstad, A.: "MSDL and C-BML Working Together for NATO MSG-085," Spring Simulation Interoperability Workshop, Orlando, FL, 2012.

[13] Pullen, J.M. & Khimeche, L.: "Advances in Systems and Technologies toward Interoperating Operational Military C2 and Simulation Systems," 19th International Command and Control Research and Technology Symposium *(ICCRTS)*, Alexandria, VA, 2014.

[14] Savasan, H., Caglayan, A., Yildiz, F., Mevassvik, O.M., Sletten, G., Schade, U., Haarmann, B. & Heffner, K.: "Towards a Maritime Domain Extension to Coalition Battle Management Language," Spring Simulation Interoperability Workshop, San Diego, CA, April 2013.

[15] Singapogu, S., Gupton, K. & Schade, U.: "The Role of Ontology in C2SIM" 21st International Command and Control Research and Technology Symposium (ICCRTS), London, UK, 2016.

[16] SISO: Standard for Coalition Battle Management Language (C-BML) Phase 1, SISO-STD-011-2014, April 14, 2014.

[17] SISO: Standard for Military Scenario Definition Language (MSDL), SISO-STD-007-2008, May 11, 2015.

[18] Svensson, P., Sandberg, S., Antelius, F., Ferriani, L. & Mounet, L.: "Evolving Standards for Tactical Data Link Aware Simulators," Fall Simulation Innovation Workshop, Orlando, FL, 2016.

[19] Wittman, R.L.: "Defining a Standard: The Military Scenario Definition Language Version 1.0 Standard," 2009 Spring Simulation Multiconference, San Diego, CA, 2009.

[20] W3C: OWL 2 Web Ontology Language, Structural Specification and Functional-Style Syntax (Second Edition), https://www.w3.org/TR/2012/REC-owl2-syntax-20121211/, December 11, 2012.

ACKNOWLEDGEMENTS

We thank our colleagues, Curt Blais and Mark Pullen, for offering good advice, helpful comments and encouragement, and the C2SIM PDG for providing C2SIM guidelines and products.

AUTHOR BIOGRAPHIES

BRUNO GAUTREAU is technical expert and leader at Airbus Defence and Space. He has contributed to the development of operational experimentations with CBML and MSDL through MSG-085 activities (Standardization for C2-Simulation Interoperation). He has about 15 years of experience in simulation area for the French Army / the DGA (Brigade and Battalion planning tool) and for the civil domain (Air Traffic Generator).

MAGDALENA DECHAND is a research fellow with Fraunhofer FKIE. She is an expert in spoken and written language analysis with a strong focus on syntax. Since 2015, as member of the C2SIM Product Development Group and the NATOMSG-145, she is working on knowledge representations and on how to exploit ontologies in the C2SIM context.

LUKAS SIKORSKI is a research fellow at Fraunhofer FKIE. He holds a diploma in Applied Computer Science. His primary focus is on developing a C2- system for manned-unmanned teaming. In the C2SIM context, he, together with Dr. Trautwein, works on automatic schema generation and the development of ergonomic user interfaces for C2 systems.

DR. IRMTRUD TRAUTWEIN is a research fellow at Fraunhofer FKIE. She holds a Ph.D. in geology. In the C2SIM context, she, together with Mr. Sikorski, works on automatic schema generation and the development of ergonomic user interfaces for C2 systems.

DR. ERIC BOUVIER is the head of the Business Development department at Diginext. He has about 20 years of experience in the development of simulation software particularly for the defence sector. Eric obtained his Ph.D. in computer science in 1997 at University Paul Sabatier in Toulouse developing algorithms and models for crowd simulation.

LIONEL KHIMECHE is the Head of M&S department at DGA (Direction Générale de l'Armement) Center for Technico-Operationnal Defense Analysis. He is in charge to execute the M&S for acquisition roadmap consisting mainly in providing for technical DGA activities, along the acquisition process, the M&S guidelines and the required tools as called common technical simulation suites (STCS) to ease the development of simulation based systems. His department is also in charge to support Program of records in developing customized M&S strategy. His main topic of research deals with C2 to Simulation Interoperability and Scenario definition to foster reuse. He co-chairs the NATO Technical Activity MSG-145 regarding C2SIM operationalization.