Knowledge Systems for Coalition Operations KSCO 2016

September 6 – 8, 2016

London, UK

Title: **Applying OntoClean for the Evaluation of the MIP Information Model**

Authors:

Names: **Hans-Christian Schmitz**
Michael Gerz

Organization: Fraunhofer FKIE

Address: Fraunhoferstraße 20
53343 Wachtberg-Werthhoven, Germany

Phone: +49 228 9435 414

E-Mail: {hans-christian.schmitz|michael.gerz}@fkie.fraunhofer.de

Applying OntoClean for the Evaluation of the MIP Information Model

Hans-Christian Schmitz, Michael Gerz

Abstract

The Multilateral Interoperability Programme (MIP) is a multinational military standardization committee that develops interoperability specifications for Command and Control Information Systems (C2IS). A key product is the MIP Information Model (MIM). It serves as a standard for information exchange for multiple echelons in joint and combined operations. The MIM harmonizes information elements from a variety of data sources and communities of interest (COIs). It is under continuous development for enabling interoperability under changing operational requirements.

In the construction of information models and ontologies, such as the MIM, inconsistencies can occur and usually do occur. To avoid them, it is advisable to guide the model construction from the very beginning and to identify and resolve inconsistencies early. We recently tested and applied an evaluation method called *OntoClean*, which has been proposed as a universal evaluation method based on insights from philosophical ontology. OntoClean defines meta-properties that are applied to the concepts defined within a specific information model. With reference to these meta-properties, subsumption constraints are defined that can be tested in an automated manner. If the constraints are fulfilled, the model is considered "ontoclean".

It turned out that the MIM can be considered "ontoclean" but the evaluation revealed that the further specification of concepts and the introduction of allegedly plausible relations might lead to semantic problems. Admittedly, the OntoClean annotation remains a challenge: an annotation experiment, conducted with military experts, revealed significant differences between annotations and proofed that the application of the method is not trivial.

1 The MIP Information Model

The Multilateral Interoperability Programme (MIP) is a multinational military standardization committee with participants from 24 member nations, EDA, and NATO. It develops interoperability specifications for Command and Control Information Systems (C2IS) and conducts and supports conformance and interoperability tests. The operational focus of MIP is on information exchange for multiple echelons in joint and combined operations, primarily addressed from a land perspective.

1 The MIP Information Model

The MIP4 Information Exchange Specification (MIP4IES) comprises exchange mechanisms, information definitions (message definitions), test specifications, and reference implementations.[1] The MIP4IES is focussed on the exchange of the current operational picture, which is considered as the main concern of the MIP COI. The MIP4IES provides the means to exchange semantically well-defined messages that represent objects on a battlefield. However, further development by MIP towards pragmatic interoperability solutions can be expected. Moreover, it is under investigation to what extent the requirements of neighbouring COIs can be met.

A key product of MIP is the *MIP Information Model* (*MIM*)[2]. The MIM harmonizes information elements from a variety of data sources and is under continuous development for enabling interoperability under changing operational requirements. It seeks to close the gap between the domain expert on the one hand and the software implementer on the other hand by enabling model-driven software development.

The MIM has been derived by more than 30,000 changes from its predecessor, the *Joint Consultation, Command, and Control Information Exchange Data Model* (JC3IEDM) (cf. Gerz and Bau 2012). It fixes many known technical and operational issues of the JC3IEDM and improves its comprehensibility both for operational and technical users. Unlike the JC3IEDM, the MIM solely serves as a semantic reference. Thus, it can be considered a model that is decoupled from nationally implemented databases. It focuses on describing semantic concepts rather than mandating a specific technical implementation. It has been designed with regard to readability, modularity, extensibility, semantic strictness, and model consistency (cf. Gerz et al. 2015).

Scope of the MIM: the MIM is composed of a few basic concepts, namely Object, Action, Location, Capability, Address and Information Group (cf. Figure 1). Each of these concepts spans its own taxonomy. The concepts and sub-concepts are further characterized by attributes and related to each other by means of associations. In total, the MIM defines approximately 2,300 types of objects, about 500 different actions, approximately 400 code lists, and more than 100 different associations across its classes (the exact number of associations depends on the way they are counted: some associations are attributed and can be unrolled into many individual associations).

Technical implementation: technically, the MIM is based on UML, extended by so-called UML profiles that constitute the MIM meta-model. A UML profile defines stereotypes that extend meta-classes such as Class and Attribute. A stereotype that extends Class can be applied to any UML class. Stereotypes themselves can have attributes. When assigning a stereotype to a model element, the stereotype attributes turn into tags of the respective model element. Values should be provided at modelling time to describe properties specific to elements. Stereotypes provide guidance on how a model element is supposed to be interpreted and used. Within the MIM, stereotypes are used for all kinds of model elements (classes, data types, enumerations, attributes, literals, and associations).

The MIM comes with a number of artefacts, including class diagrams (sub-views), OCL constraints, documentation, and examples. The latter are represented by UML object models and

[1] At the time of writing this paper, the MIP4IES is undergoing a one-year validation phase.
[2] The MIM is freely available via the MIM Portal at `https://www.mimworld.org`.

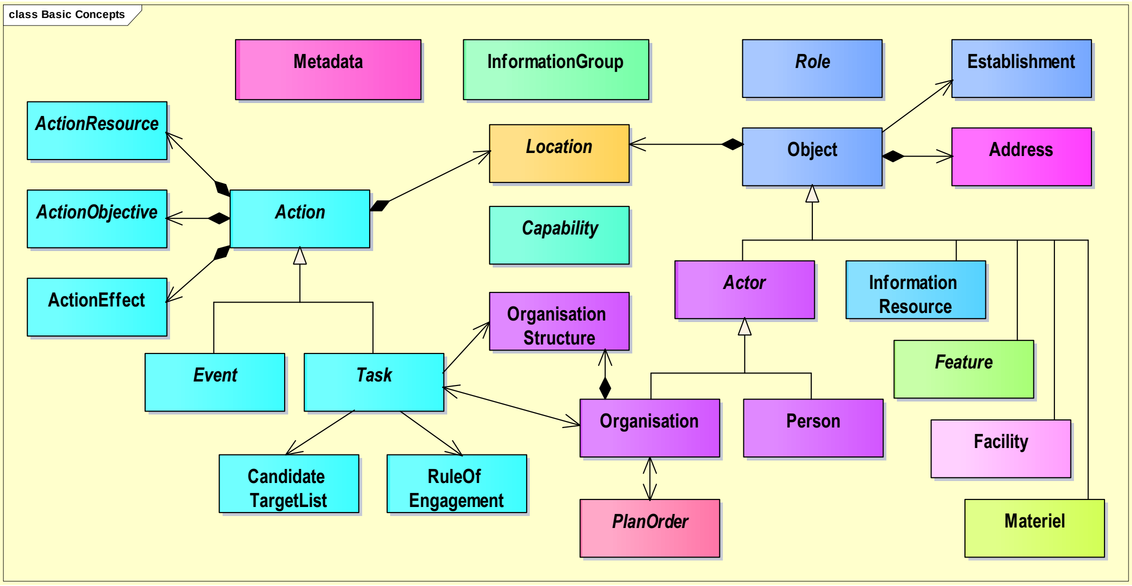

Figure 1: Core Elements of the MIM

shown in terms of UML object diagrams.

Roles: roles are an important (and in the context of this study particularly interesting) but not yet fully expanded concept within the MIM. Roles are based on the observation that, during their lifetime, objects can fulfil different functions. An organisation, e.g., can serve as a resource for performing a task or as a reporter of an incident. In these different contexts, the organisation plays different roles and it behaves according to its respective functions. Modelling roles is useful if the properties and the behaviour of real-world objects depend on the operational context and evolve over time. Which information on an object is relevant and should be specified depends on the role the object plays: (i) instances of roles may have access to only some properties of the object playing the role, and (ii) a role may come with its own properties that may be different from the ones of the class of the object that plays the role. (Such classes are henceforth called *natural classes*.)

Example: only if an object plays the role of a target, its priority as a target becomes relevant and thus should be added. At contrary, other attributes, e.g., the alternate name of the object, become irrelevant and should be disregarded. That is, a role can both extend and restrict the properties and associations of an object.

The MIM supports two different approaches to model roles. They differ with regard to complexity and expressiveness: (i) roles can be modelled as ends of associations. Both ends of a UML association include information such as the role name, the multiplicity, and the navigability. This approach is simple and does not require UML extensions. However, it does not allow specifying properties for specific roles only. Instead, all properties are introduced for the natural class that potentially plays the role. Ultimately, this results in natural classes that combine properties of many different roles. (ii) Roles can be represented by separate classes that are derived from class Role. A specific role class and the natural class whose objects can play the respective role are associated by means of an association with stereotype isRoleOf. This enables the definition of role-specific attributes external to natural classes.

So far, only two roles have been defined in the MIM, namely the role of an observer and that of a reporter. Still, the role concept is considered fruitful. Most probably, it will be extended in future versions of the MIM, with the aim of extending the possibilities of the role-specific and thus context-specific characterisations of objects.

MIM tool suite and model management: the MIM employs state-of-the-art modelling techniques and tools based on open standards and industry best practices. These tools support model management, model tailoring, model transformation, generation of exchange schemas, and model exploration.[3] The further revision of the model is an on-going process. Continuous changes impose technical challenges for configuration management. All artefacts are integrated into and linked with each other in a single model repository (in Sparx Enterprise Architect format). Specialised validation tools perform semantic analyses and check whether the artefacts are sound and consistent.

Linked data and MIM ontology: the MIM supports the notion of linked data. A Linked Data Server (LDS) facilitates the exploration of the MIM, providing information in various formats and powerful search capabilities. Among the transformations that have been defined for the MIM is a transformation to OWL2. By providing the MIM as an OWL ontology, MIP enables the integration of other conceptual schemas and semantic web data. Moreover, it provides a formal model that can be applied in expert systems involving formal reasoning. The provision of the MIM ontology is rather an investment in the future. However, the MIM ontology is already applied in a few projects, e.g., by NATO.

Let us take stock: the MIM is a UML-based information model that serves as a standard for information exchange for multiple echelons in joint and combined operations, primarily addressed from a land perspective. It refers to various legacy data models, in particular the JC3IEDM, and is under continuous development for enabling interoperability under changing operational requirements. Several transformations of the model have been defined, among them a transformation to OWL that makes the model a proper ontology. To ensure the high quality of the model even under development, it has to be continuously evaluated. To this end, procedures for model management and a respective toolbox have been developed. One aim is to improve the toolbox and include further methodologies for checking consistency of the model.

2 Ontology Evaluation with OntoClean

2.1 OntoClean

We can presuppose that inconsistencies may occur and most probably will occur during the construction of an information model or ontology. In order to avoid and mitigate inconsistencies it suggests itself (i) to guide the ontology construction from the beginning and (ii) to evaluate

[3] The MIM tool suite is available as open source software via `https://www.mimworld.org`.

and eventually correct existing parts of the ontology. To these ends, criteria for guidance and evaluation are needed. Ideally, these criteria are so general that they can be applied to any arbitrary ontology.

OntoClean (Guarino and Welty 2004, among other papers) is a general methodology for the evaluation of ontologies, in particular taxonomies. Following OntoClean, classes and other concepts (in OntoClean terms: *properties*) are further specified by meta-classes/meta-properties. With reference to these meta-properties, subsumption constraints are defined. It can be tested whether the subsumption hierarchies within the given ontology meet these constraints. If this is the case, then the ontology can be considered "ontoclean". Otherwise, the ontology is conceptually inconsistent and has to be re-worked.

That is, OntoClean defines evaluation criteria for ontologies. These criteria do not necessarily apply for an entire ontology but only for those parts that stand in subsumption relations. Accordingly, OntoClean cannot provide an exhaustive, concluding evaluation. However, OntoClean claims to be general and applicable to arbitrary ontologies and information models, because the methodology rests exclusively on fundamental, domain-independent insights of *philosophical* ontology (in particular mereology, cf. Simons 1987 and Varzi 2015). As such, in principle, it could meet the demand for guiding the further development and evaluation of the MIM.

In the following, we give a concise introduction to the core concepts of OntoClean, namely *subsumption* and the four meta-properties *rigidity*, *identity*, *unity*, and *external dependence*.

2.1.1 Subsumption

A class A subsumes a class B if and only if all elements of B are always also elements of A. In a class model, subsumption is more or less reducible to the subclass relation. Every class subsumes at least itself and all of its sub-classes.

2.1.2 Rigidity

A class is *rigid* if and only if the membership to this class is essential, that is, always necessary for all of its elements. We distinguish rigid (+R) from non-rigid (-R) classes. The non-rigid classes are further divided into anti-rigid (~R) and semi-rigid classes. Person ("human being") can be an example for a rigid class: if a person ceases to be a member of the class of persons, it ceases to exist; all members of the class Person are always necessarily persons. In contrast, membership to an anti-rigid class is not essential for any member of the class. An example is "Employee of FKIE" (or any other organisation). A person can be employed by FKIE without being always employed by FKIE, and when the person retires, she does not necessarily cease to exist. Anti-rigid classes denote *roles* that objects can play but need not play forever. Membership in a semi-rigid class is essential for some members but not all. Consider the class Weapon: some tanks are essentially weapons, others are not (they might carry a canon but need not necessarily do so).

Subsumption constraint for rigidity: an anti-rigid class (~R) must not subsume a rigid class. Such a subsumption would lead to a contradiction: let a non-empty, rigid class B be given. Because B is rigid, all members of B are necessarily members of B. Let also an anti-rigid class A be given. Because A is anti-rigid, no member of A is necessarily a member of A. If A subsumes B, then all members of B are necessarily also members of A. Therefore, since they are necessary members of B, they are also necessary members of A, which contradicts the assumption of anti-rigidity.

2.1.3 Identity

A class carries the meta-property *identity* if and only if there is a single criterion by which all elements of the class can be stably identified and thus distinguished from each other. Concepts that carry the identity meta-property (+I) are usually expressed by nouns. Examples are Person, Vehicle, or Obstacle. Concepts that do not carry the identity meta-property (-I) are usually expressed by adjectives. Examples are Red and Fast. Of course, it can be possible to identify and distinguish all red objects. However, there is no *common* criterion by which all red objects can be distinguished from each other. Therefore, the class of red objects does not fulfil the condition for identity. Identity-criteria are inherited from super-classes, that is, down a given subsumption hierarchy.

Let us discuss an example of Guarino and Welty (2004): we distinguish between different time spans of 30 Minutes, one hour, etc. The TimeSpan class carries an identity criterion: time spans can be identified and distinguished by their lengths. In addition, we define the class of time intervals. Time intervals have a starting point and an end point (Oct 10th, 2015, 12:00-12:30 GMT; Oct 11th, 2016, 13:00-13:30 GMT; etc.) One might find it reasonable to define that time intervals basically *are* time spans and that, accordingly, class TimeInterval is to be subsumed under the TimeSpan class. However, this would mean that the TimeInterval class inherits the identity criterion of the TimeSpan class and that time intervals are identifiable just by their lengths, like time spans. If this were the case, then Oct 10th, 2015, 12:00-12:30 GMT and Oct 11th, 2016, 13:00-13:30 GMT were the same time intervals (because they are the same time spans). This is obviously not true, and therefore time spans cannot subsume time intervals. It is not the case that time intervals *are* time spans, they just *have* time spans.

Subsumption constraint for identity: a class with an identity criterion (+I) cannot subsume a class without an identity criterion (-I) because the subsumed class inherits the identity criterion from the upper class.

Following Lowe (1989), one can assume two further constraints:

- Every object must be a member of a class with +I so that all objects can somehow be distinguished from each other. We call this *sortal individuation.*
- If an object belongs to several classes that do not stand in subsumption relations, then it must also be a member of a super-class that implements a common identity criterion. We call this *sortal expandability.*

Let us discuss a further example of Guarino and Welty (2004), regarding the relation of groups, groups of people and social entities: Guarino and Welty define a group as an unstructured finite collection of entities. Groups, i.e. instances of the Group class, are defined by their extensions, that is, their members. The extension of a group is its identity criterion, Group carries +I. A group of people is a specific kind of group. The respective class GroupOfPeople inherits +I from its super-class Group. A social entity, finally, consists of people who come together for some social reason like, e.g., playing cards. Since there can be very different kinds of social entities, Guarino and Welty (2004) argue that they "can't imagine a common identity criteria" (p. 162) for the entire class SocialEntity, which therefore carries -I. (Note that the members of a social entity can change over time and that the social entity can therefore not be defined extensionally. Moreover, a social entity can be informal and not always defined by its purpose and structure. It differs in this respect from an organisation.) Classes with +I cannot subsume classes with -I and, therefore, a social entity cannot be defined as a group of people. A social entity *consists* of a group of people but it *is not* a group of people.

2.1.4 Unity

Unity is not defined that easily. Let us define the meta-property in three steps (largely following Guarino and Welty 2000b). We first define the concept of an *integral whole*, then the concept of an *intrinsic integral whole*, and finally the *unity* meta-property.

Integral whole: An integral whole is either an atomic object that cannot be further divided into parts or it is an object that can be exhaustively divided into parts that stand in a unifying relation to all other parts but nothing else (cf. Simons 1987). Note that

- The unifying relation must be an equivalence relation. Therefore, distinct integral wholes cannot overlap. As a consequence, overlapping sets or groups do not count as integral wholes.
- We do not specify the notion of a unifying relation any further. Such relations can be of very different kinds, among them topological, morphological and functional relations (cf. the following examples by Varzi (2015): "The handle is part of the mug"; "The remote control is part of the stereo system"; "The left half is your part of the cake"; "The cutlery is part of the tableware"; etc.).
- Objects can be integral wholes over restricted time spans. A piece of clay, e.g., can be considered a topologically unified integral whole. However, if one puts the piece on a larger piece of clay, its parts stand in the same topological relation to the parts of the larger piece. Therefore, the original piece of clay ceases to be an integral whole for itself. It was just a *contingent* integral whole.

Intrinsic integral whole: an intrinsic integral whole under a unifying relation R is an object that is an integral whole under R for all time of its existence. Intrinsic integral wholes are not just contingent but necessary wholes. Amounts of matter of any kind (pieces of clay, etc.) are not intrinsic integral wholes. In addition, sets or groups with mere membership as the unifying relation do not count as intrinsic integral wholes.

Unity: a class carries the meta-property *unity* (+U) if and only if there exists a common unifying relation by which each element of the class can be seen as an intrinsic integrated whole. Common unifying relations are inherited from super-classes, that is, down a given subsumption hierarchy. Classes not carrying the *unity* meta-property are *non-unity* classes (-U). They might contain elements that count as intrinsic integral wholes but not under a *common* unifying relation. Anti-unity classes (~U) do not contain intrinsic integral wholes at all.

The *unity* meta-property mainly serves to distinguish objects (+U) from substances or amounts of matter (~U). An example for a class carrying +U is Person. An example for a class carrying ~U is ChemicalMateriel. As a rule of thumb, +U classes are denoted by count nouns while ~U classes are denoted by mass nouns. This rule has exceptions, however: "cutlery" and "furniture" are examples for mass nouns denoting classes of objects (+U); "paper" is an example for an ambiguous noun having both a +U and a ~U reading (article vs. material). An example for a class carrying -U is Actor: actors are intrinsic integral wholes but the class of actors is diverse – containing at least persons and organisations – so that its members cannot be considered integral wholes under the same unifying relation (see below for a discussion of the Actor concept).

Note that an object may have fuzzy boundaries and, thus, may be only vaguely definable. Yet, it may count as an integral whole. That is, we can consider mountains or oceans integral wholes, though we cannot precisely determine their borders.

Subsumption constraints for unity: (i) a class with a unity criterion (+U) cannot subsume a class without a unity criterion (-U) because the subsumed class inherits the unity criterion from the super-class. (ii) Furthermore, an anti-unity class (~U) subsumes only anti-unity sub-classes (~U): if a class does not contain integral wholes then its sub-classes do neither.

Let us discuss an example of Guarino and Welty (2004): in a given ontology, let ocean be defined as an amount of water and, accordingly, let the class Ocean be a sub-class of AmountOfWater. However, AmountOfWater carries ~U while Ocean carries +U. Therefore, the ontology violates the second subsumption constraint for unity. In order to fix the ontology, we should replace the sub-class relation with a consistsOf relation: an ocean *is* not an amount of water; it *consists* of an amount of water.

Conceptually, the *unity* meta-property seems to be the one that is most difficult to understand. Its definition is also disputable. Guarino and Welty refer to the mereological system of Simons (1987). However, there are competing systems on the market (cf. Varzi 2015). It is not self-evident that sets cannot be considered wholes: is a goalkeeper part of a football team, or does a football team not count as an integral whole at all? What is the relation between parthood and composition: are cola and rum parts of Cuba Libre? Finally, the status of vague objects with fuzzy boundaries is not that clear, although we stated above that they can be considered integral wholes: is this true for clouds as well? Different mereological systems might give different answers, so that it is hard for OntoClean to justify the claim that it implements only most fundamental, indisputable principles.[4]

[4] "[W]e do believe one aspect of the success of OntoClean has been its relative neutrality with respect to basic ontological choices [. . .]." (Welty and Anderson, 2005) This might actually be questionable.

2.1.5 External Dependence

A class A is externally dependent on another class B if and only if for each instance of A there must be a corresponding instance of B. Dependent classes (+D) are externally dependent on other classes. An example given by Guarino and Welty (2000a) is Parent and Child. Classes without an external dependency carry -D. External dependencies are inherited from superclasses, that is, down a given subsumption hierarchy.

Subsumption constraints for external dependence: A class carrying the *dependence* meta-property (+D) cannot subsume a class without the *dependence* meta-property (-D) because dependence is inherited by the subsumed class.

Let us take stock: OntoClean defines the meta-properties rigidity, identity, unity and external dependence that are assigned to the concepts of a given model (in the case of a class model, in particular to its classes). OntoClean defines inheritance relations and subsumption constraints for the meta-properties. The subsumption constraints have to be met in order to consider a model as being "ontoclean".

2.2 Annotation and Evaluation

The process of analysing an existing information model or ontology with OntoClean can be described as follows:

1. View all classes separately and tag them regarding the OntoClean meta-properties.

   In case difficulties occur:

   a) Do you think that different, incompatible taggings are possible? The reason could be that the class under consideration is ambiguous. You might be able to solve the problem by dividing the class into two different classes that are tagged separately.

   b) If you do not consider different, equally plausible taggings possible but your uncertainty of how to tag a class is more fundamental (you do not have a clue of how to tag at all), then it might be that your understanding of the class is not deep enough. Rather than choosing an arbitrary tag, you should leave the OntoClean annotation underspecified until you reach a better understanding.

2. Now view the entire class hierarchy. Do the subsumption hierarchies meet the subsumption constraints imposed by the tagging? Mark problematic cases that lead to inconsistencies.

3. Correct the problematic cases:

   a) If a problematic class is superfluous, remove it.

   b) If you can solve inconsistencies by redefining problematic classes without changing their intended meaning, do so.

c) Remove the remaining subsumption relations that do not meet the constraints. Connect classes that are now disconnected by relations that do not imply a subsumption (like consistsOf, has, . . .)

For steps (2) and (3), Guarino and Welty (2004) propose to view and correct the "backbone taxonomy" consisting of rigid classes and their subsumption relations first. They argue that the rigid classes represent the invariant domain aspects and, therefore, are the most important ones to be analysed in the first place. Only after the backbone taxonomy has been made "ontoclean", the non-rigid classes should be considered to "'flesh out' the backbone taxonomy" (p. 165).

Approaches to automating both fundamental steps of an OntoClean evaluation – tagging of concepts and evaluating subsumption constraints – have been proposed: *OntOWLClean* (Welty, 2006) specifies a meta-ontology in which the meta-properties (*rigid*, *non-rigid*, *anti-rigid* etc.) are defined as classes and the subsumption constraints are defined as relations between these classes. Classes of an ontology that is to be evaluated have to be transformed into object instances of the OntoClean classes. A sub-class relation for the objects has to be defined as well, so that the subsumption constraints can be evaluated by an automatic reasoner. AEON (Völker et al. 2008, cf. also Hicks and Herold 2009), in contrast, focusses on the automatic annotation of ontologies. The authors define linguistic contexts that give evidence for the OntoClean meta-properties of concepts. They perform web or corpus searches to investigate whether a given concept occurs in such contexts. Example: if a concept occurs very often in contexts like "is no longer (a|an)? CONCEPT", "became (a|an)? CONCEPT" or "while being (a|an)? CONCEPT", then this would count as *negative* evidence for its rigidity.

For evaluating UML class models like the MIM, the OntoClean meta-properties could just be added as attribute-value pairs (also known as tagged values) to the classes and data types of the model. It would be fairly straightforward to implement a test for automatically detecting violations of the subsumption constraints.

3 Evaluation of the MIM

For an OntoClean evaluation of the MIM, we tagged the entire object taxonomy by hand, meaning that approx. 2,300 classes were annotated with the OntoClean meta-properties. We then checked, also by hand, for apparent violations of the subsumption constraints. The initial tagging was done rather superficially; in questionable cases, decisions were taken quickly. After the initial tagging, we did not determine violations of the subsumption constraints and consequently considered the MIM to be "ontoclean". The MIM team can take this as evidence that their product is generally conceptually coherent.

However, we were also pointed to cases that might demand further investigation. In the following, we will discuss two cases that concern the alleged conflation of objects and roles, namely the modelling of obstacles and actors.

3.1 Obstacles

Let us start with an intuitive approach towards modelling obstacles. An obstacle hinders the movement of an actor. It can be a natural, geographical obstacle, like a river, or an object that has been essentially built or established as an obstacle, like an anti-tank barrier or a minefield. A river rather *plays the role* of an obstacle than it *essentially is* an obstacle: it can be an obstacle but it need not always be one. An anti-tank barrier, on the contrary, is designed as an obstacle. Thus, one could argue that the anti-tank barrier *essentially is* an obstacle. However, one could also argue that we have to distinguish between two different concepts of *obstacle*: an obstacle as an object and an obstacle as a role. An anti-tank barrier is an obstacle object that can play the obstacle role. The object is a rigid concept while the role is an anti-rigid concept. The role concept cannot subsume the object concept.

The MIM treats obstacles in the following way: first, it defines the class MilitaryObstacle (definition: "a facility designed to stop, impede, or divert movement of amphibious or ground forces") as a sub-class of Facility (definition: "an object that is built, installed or established to serve some particular purpose and is identified by the service it provides rather than by its content"). Both classes can be considered rigid. Second, the MIM defines Obstacle as a sub-class of ControlFeature which is in turn a sub-class of Feature. A feature is "an object that encompasses meteorological, geographic, or control features of military significance" ("Features can be either natural features that may influence operations or artificial features representing administrative, political, or tactical constraints to be taken into account.") A Control Feature is "a non-tangible feature of military interest that is administratively specified, may be represented by a geometric figure, and is associated with the conduct of military operations." ("A control feature is an abstract object created or assigned by military authorities for the purposes of planning and coordination, especially in operational areas. It is a non-permanent point (e.g., start point for a road move, or a reserved demolition), line (e.g., a main supply route or no fire line), area (e.g., a slow-go area), or volume (e.g., an air corridor) that may be overlaid on a map. A control feature would normally be drawn on a map overlay, traced, or superimposed onto digitised map data and assigned a descriptive title, symbol, or name (e.g., line of departure, corps boundary).") Finally, Obstacle as a sub-class of ControlFeature is defined as "an object that blocks one's way or prevents or hinders progress." The classes Feature, ControlFeature and Obstacle can be considered rigid, too.[5]

That is, the MIM contains two different rigid concepts of obstacle. Both kinds of obstacles – facilities and features – are subsumed under Object. This is neither intuitively plausible nor elegant. It would be desirable to change the model and introduce the obstacle role concept. The current MIM approach can be considered "ontoclean" but we are pointed to an alternative (better) approach, including obstacle roles, that can also be "ontoclean", as long as we keep the rigidity constraint in mind.

[5]This could be questioned.

3.2 Actors

From a naïve point of view, Actor can be considered a role as well: a person or an organisation can be an actor but need not always be so. In the MIM, however, Person and Organisation are sub-classes of and thereby subsumed by Actor (cf. Figure 2). Therefore, either Actor must not be a role but rigid, or Person and Organisation must be roles as well. It is rather implausible to consider Person a role. It therefore seems to be likely to interpret Actor as rigid, which is in accordance with the MIM definition: "a person or a group of persons that *is able* to perform actions" [emphasis by the authors]: if a person or group of persons is no longer able to perform actions it ceases to exist within the MIM domain.

For an organisation, the situation is not that clear: sub-classes of Organisation are, among other sub-classes, GovernmentOrganisation and GroupOrganisation. A government organisation, on the one hand, is defined as "an organisation that controls and administers public policy either under a national or international mandate". GovernmentOrganisation includes MilitaryOrganisation. A group organization, on the other hand, is defined as "an organisation that is non-formal in nature and classes together its members due to mutual or common circumstances". A guerrilla organisation is not a governmental organisation and must be defined as a group organisation. However, what happens if a former guerrilla organisation becomes a government organisation (military organisation)? Consider cases like the Front Patriotique Rwandais (FPR) or the Frente Sandinista de Liberación Nacional (FSLN) in Nicaragua. Did they become, due to political changes, different objects or did they just change their role? If we follow the MIM and consider Organisation to be rigid, then we have to assume that they became different objects. This is indeed both in accordance with the OntoClean principles and possibly unproblematic for the operational purpose of the MIM. If, however, one still finds this implausible and rather wants to interpret types of organisations as roles, one runs into the trouble that Actor must become a semi-rigid and, thus, dubious/potentially incoherent concept.

Again, OntoClean did not detect an inconsistency in the MIM but pointed to a conceptual intricacy.

Let us take stock: the MIM can be considered "ontoclean" and thus conceptually coherent. However, the OntoClean evaluation pointed us to potential sources of errors for the further development of the model, in particular regarding the transformation of objects into roles. Since the introduction of new roles and the respective alignment of the model are upcoming issues, we must be aware of the potential conceptual conflicts that might occur.

4 Experiment on OntoClean Tagging

An OntoClean evaluation demands that the classes of the information model or ontology are tagged with values for the OntoClean meta-properties. As mentioned before, experiments on the automatic tagging have been conducted by Völker et al. (2008). The authors compared the automatic tagging with a manual tagging of OntoClean experts. The evaluation of the manual tagging, however, revealed that the human annotators only came to a very low agreement,

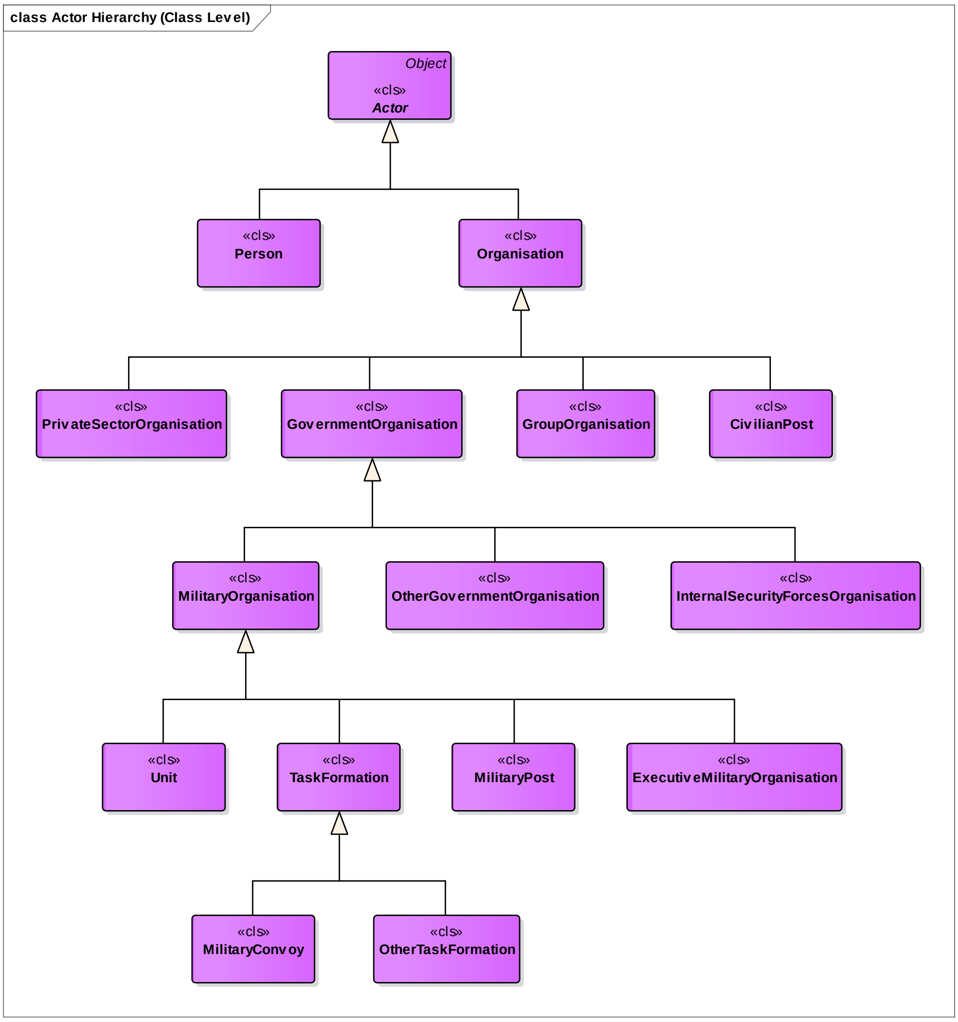

Figure 2: Actor Class Hierarchy

sometimes close to the random baseline. This result is not encouraging and creates doubt on the actual applicability of the OntoClean methodology. Therefore, we performed an annotation experiment ourselves to test to what extend domain experts are capable of applying the OntoClean concepts consistently.

The experiment: Völker and colleagues chose three OntoClean experts as human annotators. We, in contrast, chose seven subject matter experts who can be considered MIM experts – they are either MIM developers or military operational experts – but have been naïve concerning the OntoClean methodology. That is, we selected domain experts instead of methodology experts. We gave them a concise introduction into OntoClean and provided them with definitions and examples of the OntoClean meta-properties (thereby, mostly referring to Guarino and Welty 2004). We randomly selected 30 classes from the Facility hierarchy of the MIM 3.0. The selected classes, including their definitions, are listed in Table 1; the upper levels of the Facility hierarchy is shown in Figure 3. (Note that the hierarchy in Figure 3 contains elements that are not in

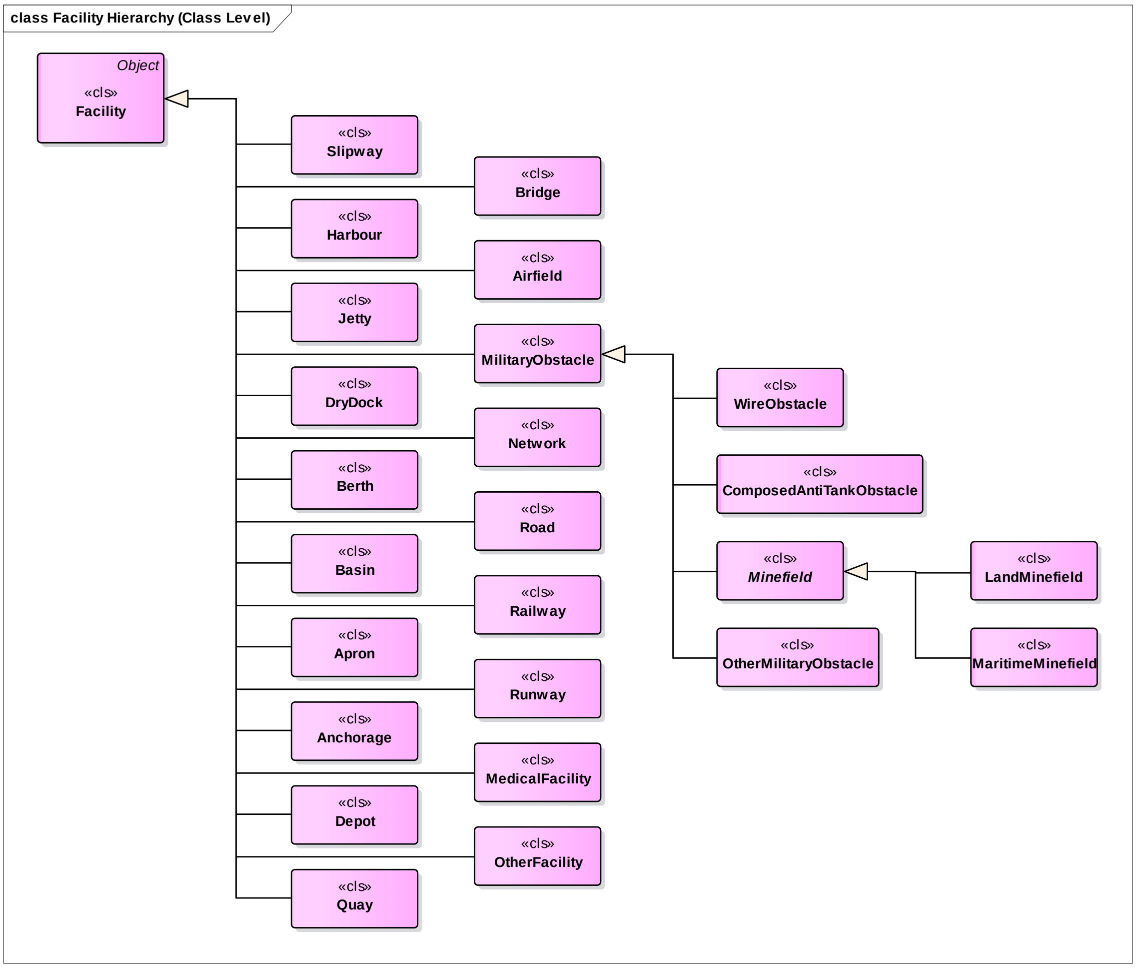

Figure 3: Facility Class Hierarchy

Table 1 and that Table 1 contains a few sub-elements that are not in the part of the hierarchy depicted in Figure 3.) We presented the selected classes without their hierarchical relationships to the test subjects (i.e., the test subjects saw Table 1 but not Figure 3) and asked them to annotate the classes with respect to the meta-properties rigidity, identity, unity, and external dependence. We let them spend approximately 45 minutes to perform this task.

The test subjects were placed together in a room where they annotated the MIM classes by pen and paper on printed tables formatted for this purpose. A slide with definitions of the OntoClean meta-properties was projected to the wall. The subjects were asked to complete the annotations for themselves, without talking, what they did. The annotations were conducted anonymously. (This is not necessarily an advantage. Retrospectively, it would have been interesting to discuss cases in which the annotators inserted a question mark instead of a proper value (+, -, ˜) and ask them why they did so.)

Five of the seven test subjects filled out their annotation sheet completely. They inserted a question mark instead of a valid value only in few cases. The other two subjects completed

Class	Definition
AntiPersonnelMinefield	An obstacle made by laying mines of anti-personnel type laid with or without pattern.
AntiTankDitch	A facility that is a ditch obstacle designed to stop tanks.
AntiTankMinefield	An obstacle made by laying mines of anti-tank type laid with or without pattern.
AntiTankWall	A wall-like obstacle capable of stopping tanks.
ArtilleryLocatingSite	A facility containing equipment employed for locating artillery.
BeamPostObstacle	A squared-off log or a large, oblong piece of timber, metal, or stone inserted in the ground to obstruct movement.
Building	A relatively permanent structure, roofed and usually walled and designed for some particular use.
BuiltUpArea	A facility containing a concentration of buildings and other structures.
ComposedAntiTankObstacle	A MilitaryObstacle (other than Minefield) that is designed or employed to disrupt, fix, turn or block the movement of tanks and that is made of modular, possibly prefabricated, components. Typically, it consists of regular spaced concrete or metal barriers (tetrahedrons or dragon's teeth) laid in single or multiple rows to prevent vehicle movement.
DemolitionDebrisObstacle	Debris obtained from the demolition of an object in order to be used as an obstacle.
Depot	A facility for the receipt, classification, storage, accounting, issue, maintenance, procurement, manufacture, assembly, research, salvage or disposal of material.
Facility	An Object that is built, installed or established to serve some particular purpose and is identified by the service it provides rather than by its content. Remarks: Categorisations of Facilities (and GeographicFeatures) are derived from Digital Geographic Exchange Standard (DIGEST) [DIGEST 2001] (Now referred to as AGeoP-3. See also [STANAG 7074 2001]). DIGEST is a multinational effort by NATO nations to reach agreement on standards for geographic products. Volume 4 of DIGEST (Feature and Attribute Coding Catalog) provides a list of feature types, attributes, and agreed domain values.
FallingBlockObstacle	A structure that is maintained in an elevated position and can be dropped to form an obstacle.
LandMinefield	A Minefield realised on or under the ground.
MainRoad	The specific Road is a main road, highway or federal road.
MilitaryObstacle	A Facility designed to stop, impede, or divert movement of amphibious or ground forces.
Mine	A facility where materials are extracted from the ground.
Minefield	A MilitaryObstacle that is a set of mines distributed across an area or volume.
MixedMinefield	A minefield made by laying mines of both anti-personnel and anti-tank type laid with or without pattern.
OtherMilitaryObstacle	A MilitaryObstacle for which no further information is given other than its category.
PedestrianRoad	The specific Road is a pedestrian road.
Railway	A track or set of tracks made of steel rails along which trains run.
Railway	The specific Road is a railway road.
RefugeeHoldingArea	A facility where refugees are assembled for classification, sorting or further movement to other facilities or installations.
Road	A path or way with a specially prepared surface that vehicles can use.
Runway	A specifically prepared surface along which aircraft take-off and land.
Track	A rough path or road, typically one beaten by use rather than constructed.
TransloadingFacility	Enables transfer of materiel from one mode of transportation to another or between the same modes of transportation.
TransportFacility	A facility that is used to support transport functions.
Tunnel	An underground or underwater passage, open at both ends, and usually containing a road or railway.

Table 1: Classes and Definitions

	Rigidity	Identity	Unity	Dependence	Total
Total agreement	13%	0%	30%	43%	22%
Tendency towards a specific value	37%	17%	60%	23%	34%
Disagreement	50%	83%	10%	33%	44%

Table 2: Total Agreement

far less than 50% of their sheets. We did not consider their sheets for the evaluation of the experiment. Therefore, we considered four annotations (rigidity, identity, unity, dependence) of 30 MIM classes assigned by five annotators, that is, 600 annotations in total.

Evaluation 1: Each test subject was asked to annotate 30 classes with respect to the four meta-properties. In 112 of these 120 cases, all test subjects provided an annotation. In the remaining eight cases, at least one subject has inserted a question mark instead of a proper annotation value. For the meta-properties *identity* and *external dependence*, two distinct values could be specified, namely + and - (non-identity/dependence). For *rigidity* and *unity*, three distinct values exist, namely +, - and ˜ (anti-rigidity/unity). For 16 classes, the anti-rigidity value was assigned by at least one subject. There is no case, in which a subject has assigned the ˜-value to unity (anti-unity). In order to reach a higher level of inter-annotator agreement, we follow Völker et al. (2008), reduce ˜ to - and thus treat anti- and non-rigidity uniformly. We therefore only compare + annotations with - annotations.

With five subjects, we have the following three cases of inter-annotation agreement:

1. Total agreement: all subjects who specify a proper value agree in their specification.
2. Tendency towards a specific value: the specification of at most one subject deviates from the others.
3. Disagreement: half of the subjects specify one value, the other half the other value. In case all five subjects specify a proper value, three specify a + and two a - (or the other way round).

Results 1: The results of the first evaluation are given in Table 2. As can be seen, inter-annotator agreement is low in particular for identity and rigidity. It is higher for unity and dependence.

Evaluation 2: we evaluated the inter-annotator agreement by computing the pair-wise inter-annotator agreements and Fleiss' Kappa for each meta-property. (Fleiss' Kappa is a standard measure for the reliability of interrater agreement.) For Fleiss' Kappa, we only considered cases, in which each annotator has specified a proper value (different from ?).

Results 2: the pair-wise inter-annotator agreements are depicted in Table 3. The results are comparable to those of Völker et al. (2008). Note that since we only have two values for each meta-property, the random agreement level is 50%. The inter-annotator agreement of this experiment can thus be considered low.

The Fleiss' Kappa values for the meta-properties can be considered low as well, showing only slight agreement (only for *dependence*, the value can be regarded acceptable but not good). The values are as follows:

- Rigidity: $\varkappa = 0.004$

	A1/A2	A1/A3	A1/A4	A1/A5	A2/A3	A2/A4	A2/A5	A3/A4	A3/A5	A4/A5	Avg.
Rigidity	69,2%	53,8%	46,2%	58,3%	44,8%	65,6%	57,1%	41,4%	66,7%	25,0%	52,8%
Identity	6,7%	31,0%	50,0%	96,7%	69,0%	56,7%	3,3%	41,4%	27,6%	46,7%	42,9%
Unity	96,7%	93,1%	86,7%	43,3%	89,7%	90,0%	40,0%	86,2%	37,9%	36,7%	70,0%
Depend.	93,3%	48,3%	83,3%	56,7%	55,2%	90,0%	63,3%	58,6%	79,3%	73,3%	70,2%
Avg.	66,5%	56,6%	66,6%	63,8%	64,7%	75,6%	40,9%	56,9%	52,9%	45,4%	55,2%

Table 3: Pair-Wise Agreement

- Identity: $\varkappa$ = -0.151
- Unity: $\varkappa$ = -0.042
- Dependence: $\varkappa$ = 0.434

Interpretation of the results 1 and 2: since the inter-annotator agreement is very low, the annotations cannot be considered reliable. The annotators are domain experts and, therefore, predestined for the annotation and interpretation of MIM classes. However, regarding OntoClean, they have been naïve and not trained properly. Instead, they only received a short introduction into the method of approx. 45 minutes. One can assume that the lack of inter-annotator agreement can be traced back to an uncertainty of how to apply the OntoClean method. If this was true then, by better training, it should be possible to enhance inter-annotator agreement and improve the reliability of the annotations. This optimistic assumption is questioned, however, by the fact that Völker et al. (2008) conducted their experiment with OntoClean experts and still came to very similar results.

In principle, there can be two reasons for the low levels of inter-annotator agreement:

1. The OntoClean meta-properties are unclear to the annotators so that they are uncertain on how to apply them even if they have a clear and unambiguous interpretation of the classes that are to be annotated.
2. The classes that are to be annotated are ambiguous and the annotators do not agree on a single interpretation. In this case, inter-annotator disagreement is to be considered positive since it leads to a discussion and possibly to a clarification of the information model. The subjects of our experiment, however, who are MIM experts, did not diagnose the MIM classes as ambiguous. Therefore, this explanation seems to be rather implausible.

Evaluation 3: do the OntoClean annotations done by the subjects reveal violations of the subsumption constraints?

Results 3: regarding the rigidity constraint (~R must not subsume +R), one annotator tagged both MilitaryObstacle and ComposedAntiTankObstacle, which is a sub-class of MilitaryObstacle, with ~R. However, she or he tagged Minefield and OtherMilitaryObstacle, both sub-classes of MilitaryObstacle as well, with +R.

Regarding identity (+I must not subsume -I), three annotators tagged the Facility class with +I while they tagged several sub-classes with -I. One annotator tagged MilitaryObstacle with +I but the sub-classes Minefield and OtherMilitaryObstacle with -I.

Regarding unity (+U must not subsume -U), three annotators tagged Facility with +U but several sub-classes with -U. One annotator tagged MilitaryObstacle with +U but the sub-classes Minefield and OtherMilitaryObstacle with -U. Finally, one annotator tagged MilitaryObstacle with +U but specific sub-classes of the sub-class OtherMilitaryObstacle (like FallingBlockObstacle) with -U.

Violations of the dependence constraint have not been indicated.

Interpretation of results 3: the violations indicated by the test subjects‘ annotations do not reveal serious inconsistencies of the MIM. They can rather be attributed to an uncertainty of the proper meaning of the meta-properties and how to apply the OntoClean method. A correction of the OntoClean annotations seems to be more plausible than a correction of the MIM.

5 Conclusions and Outlook

We conducted an OntoClean evaluation of the MIP Information Model (MIM) with the result that the MIM can be considered “ontoclean” and thus conceptually coherent. However, we were pointed to possible intricacies regarding the modelling of roles that we should keep in mind for the further development of the model. The OntoClean methodology itself proofed to be difficult to understand and hard to apply. The results of an annotation experiment we conducted support this view.

Völker et al. (2008, p. 60) conclude their study as follows: “Despite the fact that ontology evaluation is a critical task for ontology engineering there currently exist only few approaches. OntoClean is the most well-known approach that takes into account the intension of the concepts when checking the taxonomic structure of the ontology. First, applying OntoClean per se helps ontology engineers to better understand an ontology. Second, applying OntoClean allows for an evaluation of the formal properties of an ontology to detect potential misconceptualisations.” Despite the difficulties in applying OntoClean, this statement is most probably still true, which was our motivation to apply OntoClean to the MIM.

Guarino and Welty (2004, p.p.151f) claim that OntoClean also serves a rhetorical function: “Alan Rector, a seasoned veteran of ontological analysis in the medical domain, said of OntoClean, ”‘... what you have done is reduce the amount of time I spend arguing with doctors that the way I want to model the world is right ... ’ [...] A similar comment came from CYC people attending our AAAI-2000 tutorial, ‘You showed why the heuristic choices we adopted were right.’ Most experienced domain modellers can see the correct way to, e.g. structure a taxonomy but are typically unable to justify themselves to others. OntoClean has provided a logical basis for arguing against the most common modelling pitfalls, and arguing for what we called ‘clean ontologies’.”

In summary, we consider the effort and difficulties of applying OntoClean to the MIM rather high. In contrast, the outcomes of such an evaluation are rather low. We would expect a significant slow-down of progress if OntoClean were considered for all future model changes. Accordingly, we do not recommend integrating OntoClean into the MIM tool suite. However, OntoClean may provide guidance when the modellers of the MIM cannot agree on the proper representation of concepts. The OntoClean constraints will thus be kept in mind as guidance for the further development of the model.

References

Gerz, M. and Bau, N. (2012). A Platform-Independent Reference Data Model for a Future Interoperability Solution. In *17th International Command and Control Research and Technology Symposium (17th ICCRTS)*, Fairfax, Virginia. Command and Control Research Program (CCRP).

Gerz, M., Mulikita, M., Bau, N., and Gökgöz, F. (2015). The MIP Information Model – A Semantic Reference for Command & Control. In *ICMCIS 2015*. IEEE Explore.

Glimm, B., Rudolph, S., and Völker, J. (2010). Integrated Metamodeling and Diagnosis in OWL 2. In Patel-Schneider, P. F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J. Z., Horrocks, I., and Glimm, B., editors, *The Semantic Web - ISWC 2010 - 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part I*, pages 257–272, Heidelberg. Springer.

Guarino, N., Carrara, M., and Giaretta, P. (1994). An Ontology of Meta-Level Categories. In *Proceedings of KR94*, San Mateo. Morgan Kaufmann.

Guarino, N. and Welty, C. (2000a). A Formal Ontology of Properties (Preliminary Version). In Benjamins, R., Gomez-Perez, A., Guarino, N., and Uschold, M., editors, *Proceedings of the ECAI-2000 Workshop on Applications of Ontologies and Problem-Solving Methods*, Berlin.

Guarino, N. and Welty, C. (2000b). Identity, Unity, and Individuality: Towards a Formal Toolkit for Ontological Analysis. In Horn, W., editor, *European Conference on Artificial Intelligence (ECAI-2000)*, pages 219–223, Amsterdam. IOS Press.

Guarino, N. and Welty, C. (2002). Evaluating Ontological Decisions with OntoClean. *Communications of the ACM*, 45(2):61–65.

Guarino, N. and Welty, C. (2004). An Overview of OntoClean. In Staab, S. and Suder, R., editors, *Handbook on Ontologies*, pages 151–172. Springer, Berlin, Heidelberg.

Hicks, A. and Herold, A. (2009). Evaluating Ontologies with Rudify. In *Proceedings of KEOD*.

Kaplan, A. (2001). Towards a Consistent Logical Framework for Ontological Analysis. In *Proceedings of FOIS 01*, New York. ACM.

Lowe, J. (1989). *Kinds of Being: A Study of Individuation, Identity, and the Logic of Sortal Terms*. OUP, Oxford.

Simons, P. (1987). *Parts. A Study in Ontology*. OUP, Oxford.

Tahko, T. and Lowe, J. (2015). Ontological Dependence. In *Stanford Encyclopedia of Philosophy*. `http://plato.stanford.edu/entries/dependance-ontological/`.

van Inwagen, P. and Sullivan, M. (2014). Metaphysics. In *Stanford Encyclopedia of Philosophy*. `http://plato.stanford.edu/entries/metaphysics/`.

Varzi, A. (2015). Mereology. In *Stanford Encyclopedia of Philosophy*. `http://plato.stanford.edu/entries/mereology`.

Völker, J., Vrandečić, D., Sure, Y., and Hotho, A. (2008). AEON – An Approach to the Automatic Evaluation of Ontologies. *Applied Ontology*, (3):41–62.

Welty, C. (2006). OntOWLClean: Cleaning OWL ontologies with OWL. In Bennet, B. and Fellbaum, C., editors, *Proceedings of FOIS 2006*, Amsterdam. IOS Press.

Welty, C. and Anderson, W. (2005). Towards OntoClean 2.0: A Framework for Rigidity. IBM Research Report RC23754, IBM.

Welty, C. and Guarino, N. (2001). Supporting ontological analysis of taxonomic relationships. *Data & Knowledge Engineering*, 39(1):51–74.

Welty, C., Mahindru, R., and Chu-Carroll, J. (2004). Evaluating Ontology Cleaning. In *Knowledge Representation & Reasoning*, pages 311–316.