Appl Intell (2017) 46:285–307
DOI 10.1007/s10489-016-0834-7

Multi-layer ontology based information fusion for situation awareness

Fang-Ping Pai[1] · Lee-Jang Yang[2] · Yeh-Ching Chung[1]

Published online: 22 August 2016

Abstract Originated from the military domain, Situation Awareness (SAW) is proposed with the aim to obtain information superiority through information fusion and thus to achieve decision superiority. It requires not only the perception of the environment, but also the reasoning of the implicit or implicated meaning under the explicit phenomenon. The principal goal of this paper is to exploit the semantic web technologies to enhance the situation awareness through autonomous information fusion and inference. Recently, ontology has played a significant role in the representation and integration of domain knowledge for high-level reasoning. The multi-level ontology merging paradigm is followed in this work for the conceptual modeling and knowledge representation. Firstly, Military Scenario Ontology (MSO) and Battle Management Ontology (BMO) are defined according to corresponding reputable standards as the domain ontology. We propose the Situation Awareness Ontology (SAO) as the core ontology to integrate MSO, BMO and even other publicly defined ontology for higher-level information fusion. The SAO is composed of objects representations, relations and events that are necessary to capture the information for further cognition, reasoning and decision-making about the situation evolving over time. Military doctrines and domain knowledge are expressed as Horn clause type rules for reasoning and inference. Multi-layered semantic information fusion that integrates ontologies, semantic web technologies and rule-based reasoning can therefore be conducted. An experimental scenario is presented to demonstrate the feasibility of this architecture.

✉ Fang-Ping Pai
fppai118@gmail.com

Lee-Jang Yang
calsasrd@hotmail.com

Yeh-Ching Chung
ychung@cs.nthu.edu.tw

[1] Department of Computer Science, National Tsing Hua University, 101, Section 2, Kuang-Fu Road, Hsinchu City 300, Taiwan

[2] Aeronautical Systems Research Division, National Chung-Shan Institute of Science and Technology, P.O. Box 90008-11, Taichung 407, Taiwan

Keywords Situation awareness · Information fusion · Ontology · MSDL · BML

1 Introduction

In this Net-centric era, information superiority is a critical factor that permits the decision makers to conduct correct judgments and thus to cope with the rapidly changing environment. Originated from the military domain, Situation Awareness (SAW) aims to provide information superiority through information fusion and thus to achieve decision superiority. The firmly established and widely accepted definition of SAW is proposed by Endsley [1] as "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future" It requires not only the perception of the dynamically changing state of the environment but also the reasoning of the implicit or implicated meaning under the explicit phenomenon. Related approaches have been applied to many non-military and security-related domains like transportation management, energy management, environmental control, health care, disaster management and financial markets.

Situation awareness needs agents to autonomously collect situation information, derive propositions, and make decisions to satisfy their goals such that they can react to the dynamically changing environment adequately. Even more, the agents can learn to evolve adaptively based on the information and the effects induced by their judgment. All the information should be given well-defined format and meaning as a foundation to better enable computers and people to work in cooperation. The first and second level of situation awareness, i.e., perception and comprehension require the Data and Information Fusion (DIF) capability Fundamentally, DIF involves the collecting and integrating separated data created by heterogeneous and independent sources like devices (i.e., sensor information), humans (in the form of text) or software agents. These data are described following different or even specially defined protocols It is required to define a common model that can provide knowledge representation syntactically and semantically. Moreover, the purpose of information fusion is to derive a holistic picture about the situation. It is necessary that the implicit relationships under the explicit phenomenon can be reasoned, and the critical hidden facts can be inferred through fusion of uncertain and segmented data.

According to [2], the so-called JDL (Joint Directors of Laboratories) model defines different levels of information fusion as the following:

- Level 0, Sub-Object Data Assessment: estimation and prediction of signal/object observable states by pixel/signal level data association and characterization.
- Level 1, Object Assessment: estimation and prediction of entity states on the basis of inferences from observations.
- Level 2, Situation Assessment: estimation and prediction of entity states on the basis of inferred relations among entities.
- Level 3, Impact Assessment: estimation and prediction of effects on the situation of planned or estimated/predicted actions by the participants.
- Level 4, Process Refinement: adaptive data acquisition and processing to support mission objectives.

This paper presents an architecture that supports level1 and level 2 information fusion functions and provides certain level 3 and level 4 processing capabilities. The main challenge of implementing these kinds of highlevel fusion is the construction of a formal descriptive structure of the entities (things), attributes, and relations among them for reasoning about the situation and threats. How to provide a mechanism to standardize the incoming information syntactically and semantically, and thus to orchestrate the autonomous agent works is the major concern of this study. It is also the issue that the semantic web technologies proposed to cope.

Semantic Web, understood as “the web of meanings” or “a web of data” was formally introduced by Berners-Lee in 2001 [3] as “an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.” A whole suite of standards, technologies, and related tools are developed around this concept to allow machines (software agents) understand the meaning (semantics) of information on the Web and, therefore, information can be captured and processed automatically by software agents.

Among the semantic web technologies, ontology provides the semantic basis for interoperability among domain entities. In general, ontologies concentrate on defining classes/subclasses and characterizing the relationships among them and their instances. In this study, the ontology based concepts and methodology, originally designed for semantic web applications, are applied to attach machine (agent)-understandable meanings to the objects in an operation scenario. We follow the multi-level ontology merging paradigm to define Military Scenario Ontology (MSO) and Battle Management Ontology (BMO) according to corresponding standards as the domain ontology firstly. The Situation Awareness Ontology (SAO) that integrates the MSO BMO and other publicly defined ontologies is constructed as the core ontology. SAO is composed of objects, relations and events descriptions that are necessary to capture the information for further cognition, reasoning and decision making about the evolving situation.

Moreover, in the semantic web architecture, the rules focus on defining a general mechanism for discovering and generating new relationships based on existing ones. The logic and rules should be expressed on the basis of ontology such that the agents can derive implicated relations and make queries. In this study, military doctrine and domain knowledge are expressed as Horn clause type rules for reasoning and inference.

As depicted in Fig. 1, our data fusion model is a full RDF-based model with the information annotations according to our defined and publicly defined ontology. The source data are processed and transformed into RDF based description.

The rest of this paper is structured as follows. In Section 2, the relevant studies are reviewed and compared. The proposed ontology architecture and the multi-layered semantic information fusion framework are introduced in Sections 3 and 4. Experimental results are provided and discussed in Section 5. Finally, we concluded in Section 6 with some final remarks and future works.

2 Related works

One trend that has become prevalent recently is using the ontology-based approach as a paradigm to develop computer-based information fusion and thus achieve situation awareness. Most of these approaches define ontologies

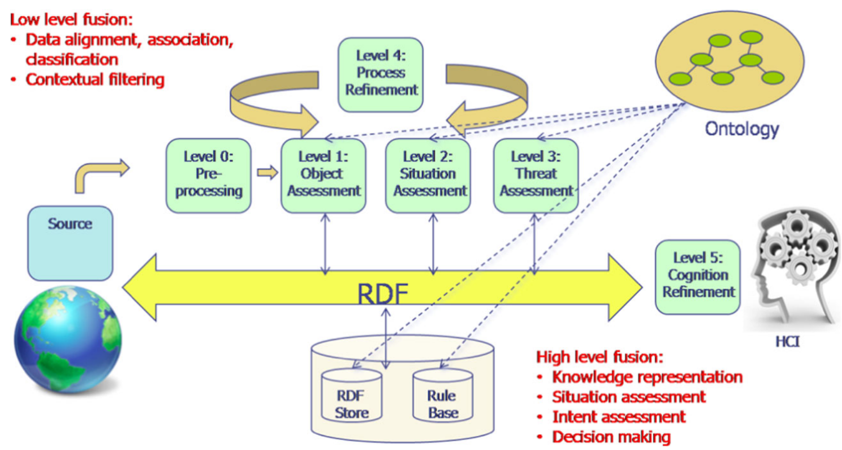


Fig. 1 RDF Based Data Fusion Model

for a particular context and put together the contextually related information to make it semantically richer and machine understandable. Barrachina et al. [4] and Golestan [5] use ontological approaches to describe contextual information and provide situation awareness for the vehicles on the road to avoid traffic accidents. Barrachina et al. propose VEhicular ACcident ONtology (VEACON) to improve safety by combining the accident information and the General Estimation System (GES) database. The VEACON ontology consists of four classes of Accident, Environment, Vehicle, and Occupant, which are interconnected through semantic links.

BeAware! [6, 7] represents another framework for ontology-driven, rule-based situation awareness systems. An action-awareness ontology is designed as domain-independent situation awareness core ontology. The ontology introduces concepts that can describe primitive spatial-temporal relations among the observed real-world objects. Situation types can be defined explicitly using these primitive relations. Moreover, since other approaches assumed the relation individuals are already asserted in the ontology, BeAware! tries to address the problem about how to derive relation individuals during situation assessment. The situation assessment is achieved by searching for interrelated objects that match given situation and relation type definitions. It provides predictions of evolving situation based on the qualitative spatial-temporal relations. The applicability of the framework is demonstrated using a real-world road traffic monitoring scenario.

Sheth et al. [8] describe a semantic sensor web (SSW) in which sensor data is annotated with semantic metadata to provide enhanced descriptions and meanings. RDFa [9] is employed to add semantic annotations of spatial, temporal and thematic information to sensor data. A suite of four existing ontologies including temporal ontology, geospatial ontology, weather ontology and sensor ontology was applied to formally represent the domain knowledge. Furthermore, SWRL (Semantic Web Rule Language) based rules defined over OWL ontology is used to deduce new assertions from known instances. As a proof of concept, two prototype applications are implemented. One of them, the Semantic Sensor Observation Service (SSOS) uses the SSW framework to enable complex queries over weather data. This service produces collections of long-term weather readings annotated with ontological concepts from a public government website. Complex queries can be achieved by a composition of conditions represented as rules.

In [10], Roda et al. present an ontology-based framework to support intelligent data analysis (IDA) of sensed data. It takes advantages of semantic technologies to achieve high-level qualitative descriptions about the state or condition of a dynamic process. A novel knowledge model that integrates four featured ontologies: TAO (Temporal Abstraction Ontology), SSN (Semantic Sensor Network ontology), SWRLTO (SWRL Temporal Ontology) and DUL (DOLCE Ultra-Lite) is constructed. The two existing domain ontology (SSN, SWRLTO) and TAO are aligned using DUL as the upper ontology. SWRLTO provides temporal modeling and reasoning. TAO has been designed to capture the semantic of temporal abstractions. SSN conceptualizes sensor measurements, thus enable a full integration with semantic sensor web (SSW) technologies. The proposed framework can monitor dynamic processes by means of temporal modeling and reasoning (i.e. SWRLTO). Qualitative temporal patterns are formulated to handle temporal relations (e.g. before, during, overlap, etc). These temporal patterns can be placed in both rules and queries, and can be interpreted by the SWRL-enabled reasoner. The presented example illustrates the application of their approach for supervising a chemical process and how critical condition (e.g. a fault in an industrial plant) can be inferred by tracking multivariate qualitative temporal patterns.

A process for building formal ontology to capture various sorts of complex relation types with the purpose to achieve higher-level information fusion is presented by Little and Rogova [11]. Basic Formal Ontology (BFO) [12] methodology is employed in upper ontology construction. In the demo post-disaster scenario, situations can be characterized by items include physical objects, aggregates, processes, events, elementary situations (ES) and combinations of ES. These elements can be categorized into two ontologies respectively, SNAP for *continuent* items of interest and SPAN for *occurrent* items like processes and events. The relations connecting ontological layers of SNAP and SPAN can be decomposed into inter-class and intra-class types. They show that BFO can adequately capture the complex and intermingled situated items needed for higher-level fusion. In our approach, the MSO and BMO can be considered analogous to SNAP and SPAN respectively. Moreover, the SAO we proposed to describe the situational relations can be deemed to be correspondent with the BFO.

Using multi-agent system (MAS) approaches with ontologybased knowledge representation to provide situation assessment is a developing direction of research. Laclavik et al. [13] propose an eventbased model suitable for applications where an agent needs to search for information or knowledge in environment evolving in time. They develop the AgentOWL library to create agents with OWL knowledge model and JENA semantic library. In SACoSS (Semantic Agent Based System for Cloud Service) [14], semantic agents use a cloud service ontology based on OWL-S to extract the knowledge about the services and produce a list of SaaS level and IaaS level cloud services as suggestion according to the consumer requirements. AESOP [15] addresses the problem of inferring threats in urban environments by the so-called BDI-SA agent system based on the general BDI (Belief, Desire and Intention) model for situation awareness. The fundamental concept of this model is about event correlation; this process takes into account temporal, causal, spatial and other domain-specific relations. The external events received, and the events generated by agents are correlated into compound high-level synthetic ones. The synthetic events will be compared with patterns of abstract situation stored in the library to recognize and instantiate specific ones. Fuzzy event correlation and fuzzy situation assessment are incorporated in the approach.

Kokar et al. [16] try to construct a “unifying framework” to integrate various research efforts in the field of situation awareness. They present the Situation Theory Ontology (STO) to capture the situation theory of Barwise [17] in terms of OWL-form ontology. Such ontology allows the expression of situations in a commonly supported language with machine-interpretable semantics. Automatic logical inference using the formal description of the situation and the ontology is also demonstrated.

Matheus et al. [18] propose the so-called Situation Awareness Core (SAW-CORE) ontology to model the concepts of situation awareness and support high-level reasoning. SAW-CORE ontology formalizes the knowledge representation of objects, relations and their temporal evolutions to enhance decision making and achieve good performance. In the SAW-CORE ontology, *Situation* class is proposed as a collection of *Goals*, *SituationObjects* and *Relations*. The *SituationObject* class represents entities in a situation, they can have properties (i.e., *Attributes*) and can participate in *Relations*. The *Relations* combine pairs of situation objects are associated with a truth value triggered by the rules that define the relations.

In [19], Matheus et al. use SWRL and OWL to capture domain knowledge for a situation awareness application applied to a supply logistics scenario. In their study, the rules are firstly presented in an abstract syntax based on n-ary predicates. These predicates are then converted into representation complies with the binary and unary predicates represented using SWRL. They also demonstrate the application of SWRL rules in their general purpose situation awareness assistant (SAWA) [20, 21]. SAWA uses OWL and SWRL to represent domain knowledge and then employs inference engine to reason about the specific evolving situation.

Bowman et al. [22], Boury-Brisset [23] and Smart [24] start the investigation of using ontologies for situation awareness with a focus on the military domain. These studies provide preliminary conceptual and methodological approaches. A methodology used to dynamically represent context knowledge with ontology to detect and evaluate anomalous situations in a harbor surveillance scenario is demonstrated by Gometz-Romero et al. [25]. Their architecture includes two processing levels the first exploit rule-based reasoning to classify objects according to predefined rules and the second tier predicts the threat level of situations involving objects that are not compliant to the normality-model. Farinelli et al. [26] use the multi-agent approach to provide situation assessment in a maritime scenario. The agents cooperatively share local information to reach a shared and coherent assessment of the situation. They build an ontology of situations for the maritime domain to represent situation types and their relationships, and use description logic inference to reason on situations. Event assessment and intent inference are the focus of their study.

Migueláñez et al. [27] provide a semantic knowledge-based framework to improve the situation awareness for autonomous underwater vehicles. An integrated hierarchical framework of ontologies to represent the required knowledge is proposed. They use Standard Ontology for Ubiquitous and Pervasive Applications (SOUPA) [28] as core ontology and introduce two application ontologies. These

Table 1 Existing Ontology-based Event/Situation related Applications

‘Applocations	CoreOntology	Domain Ontology	Demonstration Domain
[5]		VEACON	Road Accidence Avoidance
BeAware! [6, 7]	SAW Core Ontology		Road traffic monitoring
SSOS [8]	SSW	Temporal ontology, geospatial ontology, weather ontology sensor ontology	Semantic sensor web
IDA[10]	DUL	TAO,SSN, SWRLTO	Intelligent data analysis
[11]	BFO	SNAP,SPAN	Post disaster threat assessment
AgentOWL [13]		AgentOWL	Semantic knowledge model and agent architecture
SACoSS [14]		Cloud service ontology	Semantic Agent Based System for Cloud Service
AESOP [15]			Threat inference in urban environment
[16]		STO	Automated situation awareness.
[18]		SAW-CORE	Automated situation awareness.
SAWA [20]	SAW Core Ontology	Event Ontology	Supply Logistics
[25]		Vessel Ontology	Harbor Surveillance
SAAUV [27]	SOUPA	SMAO,MPO	SAW of Autonomous Underwater Vehicles
[29]	CYC Upper Ontology	CONON	Smart home application
[33]	DOLCE	Event-Model-F	Emergency Response
This research	SAO	MSO, BMO	Situation Awareness

two ontologies, i.e. Status Monitor Application Ontology and Mission Planning Ontology are analogous to our MSO and BMO respectively.

Pervasive computing is another important application field of situation awareness or context awareness. The agents (entities) in the pervasive computing environments require the capabilities to be aware of the situation changes and then to dynamically adapt their behavior to satisfy the goals. Wang et al. [29] propose OWL encoded CONtext ONtology (CONON) as the upper ontology for the modeling of context in the pervasive computing environment and logic based context reasoning. CONON describes physical and conceptual objects including Person, Activity, Computational Entity and Location and is demonstrated with a smart home application. The situation ontology by Yau and Liu [30] incorporates situations as contexts origin from pervasive computing. Situations are classified into atomic and composite ones that can be directly or indirectly represented by contexts. Context Broker Architecture (CoBrA) [31, 32] is a broker-centric agent architecture for supporting context awareness systems in smart spaces with SOUPA as the core ontology.

Events representation is critical for modeling of the situation. Scherp et al. [33] propose Event-Model-F as a formal model and core ontology for events based on DOLCE+Dns Ultralite (DUL) as upper ontology. The authors also make a systematic comparison about event and situation related ontologies, and argue that the existing models substantially lack in supporting the mereological, causal and correlative relationships. Event-Model-F provides comprehensive

Table 2 Comparison of Existing Ontologies for Situation Awareness

Ontology	Spatial Relation	Temporal Relation	Situation Types	Situation as Objects	Event Representation	Attributes	Cause Relation
SOUPA [28]	X	X			X	X	
Situ. Ontology [30]							
STO [16]				X		X	
CONON [29]	X					X	
EventModelF [33]	X	X			X	X	X
SNAP/SPAN [11]	X	X	X	X			X
SAWA [20]		X		X	X	X	X
SAW Core [6, 7]	X	X	X	X	X	X	X
SAO (This Study)	X	X	X	X	X	X	X

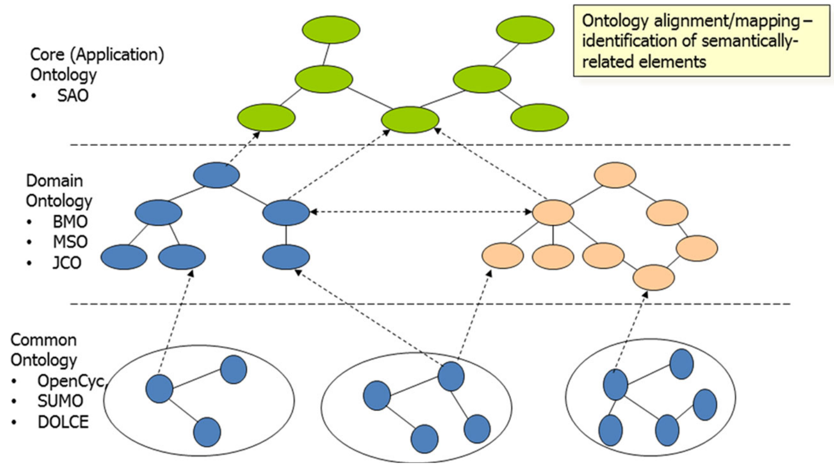


Fig. 2 Multi-level Ontology Integration

support to present time and space, and the inter-relations between events. This model can represent arbitrary occurrences in the real world and formally model different relations of events. A command and control system for emergency response is presented to demonstrate the feasibility of this model. All the applications surveyed above are summarized in Table 1.

In [7] Baumgartner et al. also compare related situation awareness frameworks according to Endsley's levels of SAW i.e., the perception, comprehension and projection layers respectively Baumgartner and Retschitzegger [34] propose an evaluation framework and examine four existing domain-independent upper/core ontologies for situation awareness. The criteria they use can be categorized into three dimensions: top-level concepts, situation awareness distinct concepts and modeling characteristics.

A survey of existing core ontologies for situation awareness is listed in Table 2. Our evaluation focuses on the temporal, spatial, event and relation representation. The works surveyed include SOUPA [28, 31] CONON [29] Situation Ontology [30] SAWA [20, 21], STO [16], the BeAware! SAW Core Ontology [7] and our Situation Awareness

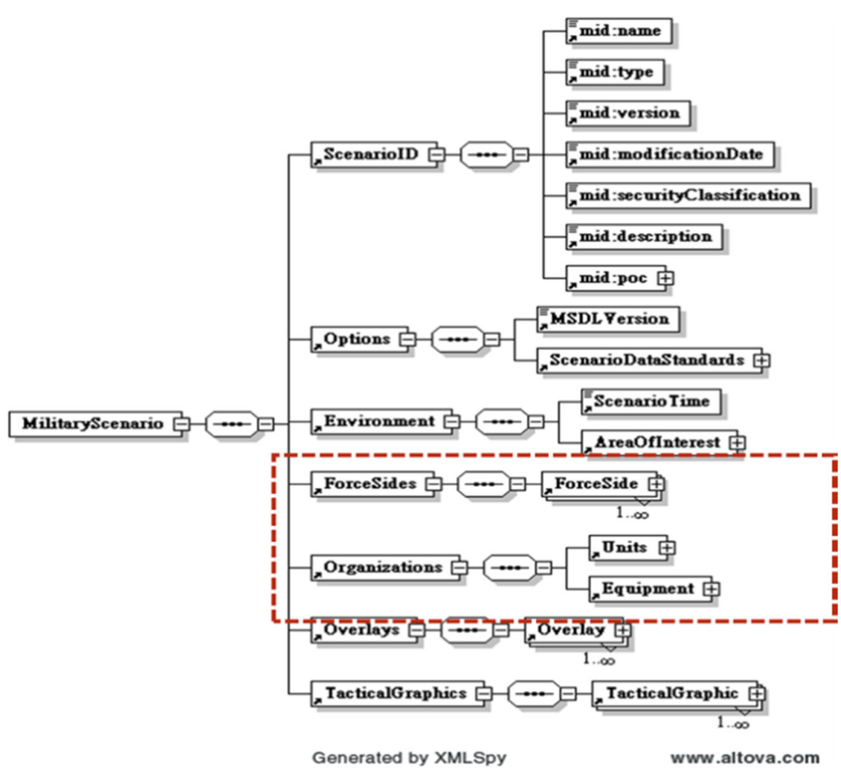


Fig. 3 MSDL Compositions

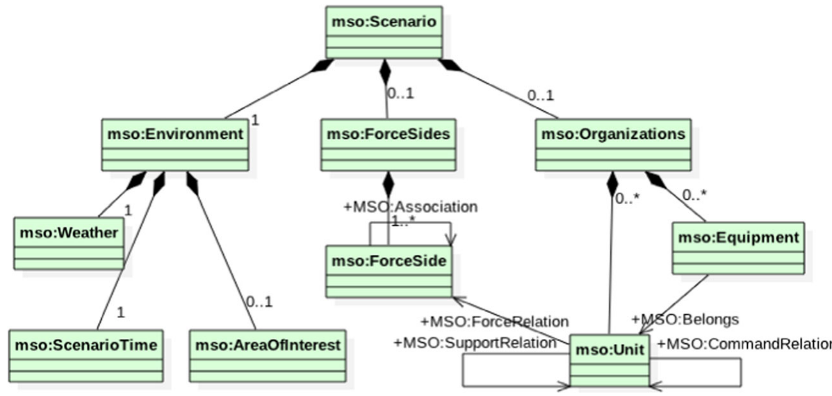


Fig. 4 The Military Scenario Description Ontology (MSO)

Ontology (SAO). The analysis shows that most of the existing systems and models substantially lack in supporting causal and correlation at relationships.

SOUPA provides spatial-temporal representation to cover concepts like time instant, intervals, movable spatial things and geographical entity. It defines a set of ontologies for expressing time and temporal relations, including *TimeInstant* and *TimeInterval* classes and the corresponding *InstantThing* and *IntervalThing*. SOUPA also defines properties to describe time order relations. SOUPA space ontology is designed to support reasoning spatial relations, and parts of the ontology terms are adopted from OpenCyc ontology. It defines *SpacialTemporalThing* as the intersection of *SpatialTemporalThing* and *Event*. These expressions may be adequate for context awareness for pervasive computing applications but not feasible to handle situation awareness.

The issues of representing relations and attributes that evolve over time need to be addressed for a situation awareness ontology. SAW Core ontology introduces *EventNotice* and *PropertyValue* classes with a *Time* member variable to provide time-dependent functions. An instance of *EventNotice* will be generated at time *T* by some situation objects, and the *EventNotice* object can affect the attributes or relations by associating it with value as an instance of *PropertyValue*. In this manner, the *EventNotice* controls the time progressing, it affects *Relation*s classes and controls the life duration of a *PropertyValue*.

Snidaro et al. [35, 36] employ Markov Logic Networks (MLN) to fuse uncertain knowledge and evidence to provide event recognition and anomaly detection for maritime situational awareness. They leverage both the expressive power of first-order logic and the probabilistic uncertainty reasoning supported by MLN. Rules are expressed using first-order logic (FOL), and MLN is applied to encode uncertainties of the knowledge. Observed (non-static) and contextual (static) evidences are fed into the inference engine and deduce results based on the evidences and reasoning on incomplete data. Unobservable complex events and probabilities of occurrence are provided. Comparisons are also made to show that the reliability of situation assessment can be raised after the introduction of contextual information.

3 Multi-layered ontology integration

Ontology has recently been introduced into higher level information fusion as a formal structure to describe entities (things), attributes, relations and general domain theories. In the fields of artificial intelligence, information retrieval, natural language processing, knowledge engineering and

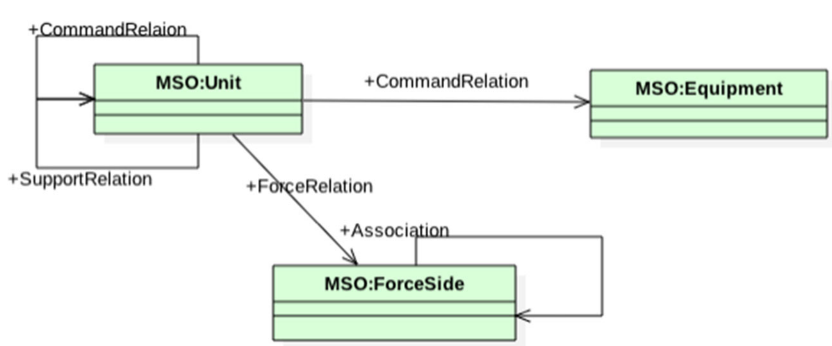
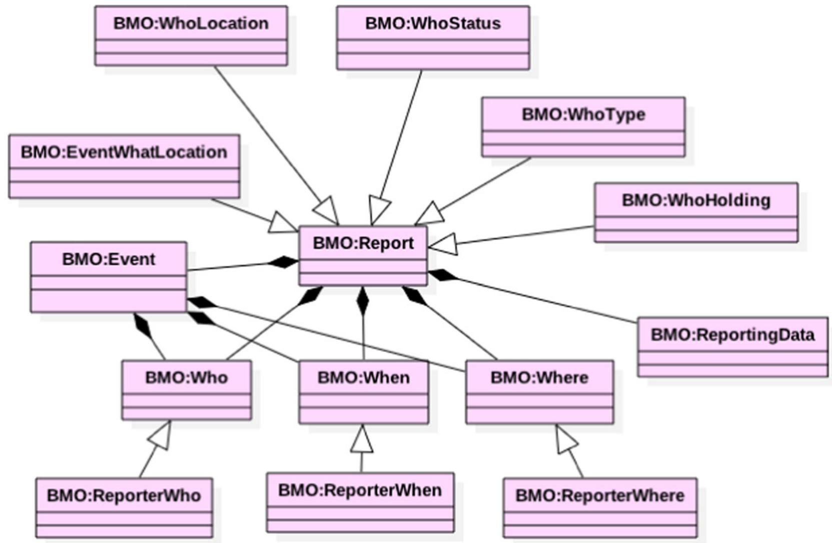


Fig. 5 MSO Relations

Fig. 6 Battle Management Ontology (BMO)



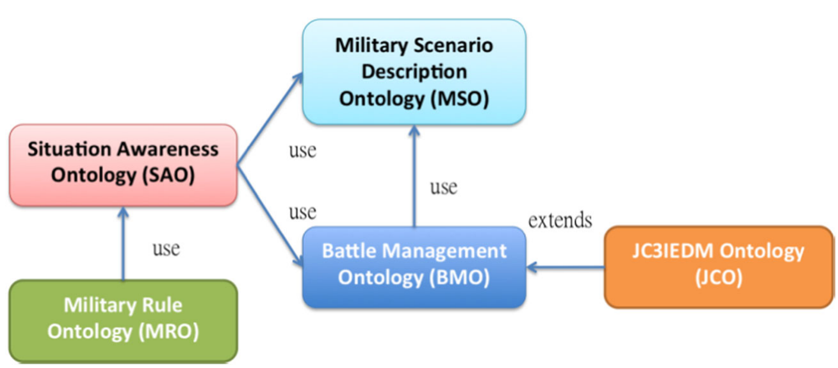
especially e-commerce systems, ontology is utilized to address interoperability problems occurred due to different configurations and communication standards when attempting to share information.

One important feature of ontology is that it can dynamically import and extend any data model by integrating various ontologies directly Moreover, this kind of integration can be done in a distributed way. According to [37], the ontologies used for agent applications have been classified as follows:

- Knowledge representation ontology: it captures the representation primitives (classes, relations, attributes, etc.) used to formalize knowledge under a given Knowledge Representation (KR) paradigm.
- General or Common ontology: it represents common sense knowledge to be reused among domains. The ontology vocabulary contains terms related to things, events, time, space, etc.
- Top–level/Upper–level ontology: it describes very general concepts as well as providing general notions under which all root terms in existing ontology should be linked. However, the existing top–level ontologies provide different criteria to classify the most general concepts.
- Domain ontology: it is an ontology reusable in a given specific domain (medical, engineering, enterprise, etc.).
- Task ontology: it describes the vocabulary related to a generic task or activity by specializing the terms in the top–level ontology.
- Domain-task ontology: it is a task ontology reusable in a given domain but not across domains.
- Method ontology: it gives definitions of relevant concepts and relations applied to specify a reasoning process to achieve a particular task.
- Application ontology: it contains all the definitions needed to model the knowledge required for a particular application.

In our interested domain, ontologies for scenario description, Command and Control (C2) data elements and doctrinal rules are necessary for information fusion and ontology-based inference. We define Military Scenario Ontology (MSO) and Battle Management Ontology (BMO) as the domain ontology respectively MSO addresses the issue

Fig. 7 Ontology Integration Model



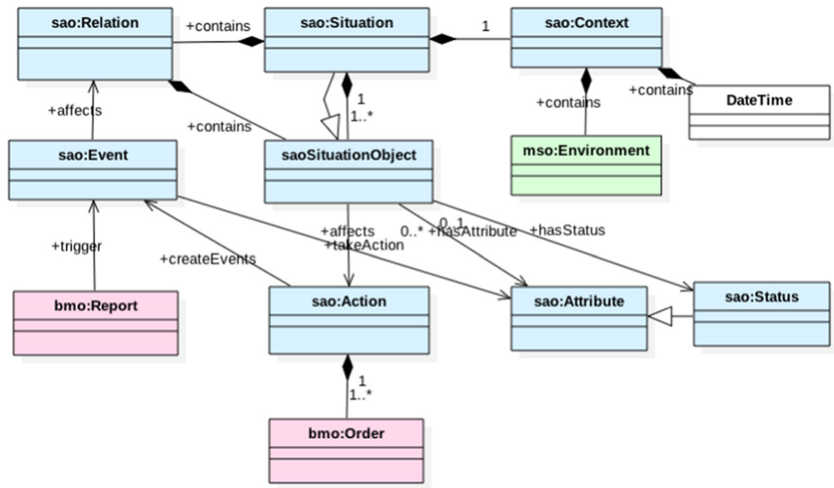


Fig. 8 SAO Core Composition

of providing necessary information to describe scenarios. BMO refers to the general approach of the unambiguous information exchange across command and control (C2), simulation and robotic systems.

To formalize the concepts of situation awareness, integration of the ontologies to describe the military scenario (MSO) and runtime command/control orders and reports (BMO) are required MSO and BMO should be implemented simultaneously to complement one another We follow the mapping and alignment concept to interconnect these two ontologies. Furthermore, complying with the multi-level ontology merging paradigm We construct a Situation Awareness Ontology (SAO) as core ontology to capture the knowledge associated with situation awareness that supports high- level reasoning by importing the MSO and BMO. Furthermore, we also construct rules upon the expressions of MSO, BMO and SAO to express the domain doctrine rules for inference (Fig. 2).

3.1 Military scenario description ontology (MSO)

The Military Scenario Definition Language (MSDL) [38] is designed by SISO (Simulation Interoperability Standards Organization) to specify military scenarios that can be shared and interoperated between simulations and C2 systems consistently. MSDL is an XML-based language that defines a standard format to describe the composition of a scenario, its schema is depicted in Fig. 3. The most important part of an MSDL description is about the force organization compositions including sides, units, equipment and the relations among them.

In our approach, MSO is defined based on MSDL to structurally and formally represent the domain knowledge for the military scenario description. The dynamically changing situations induced by reported information is reflected as an MSO-described scenario. The most important part of MSO is about the organization composition such

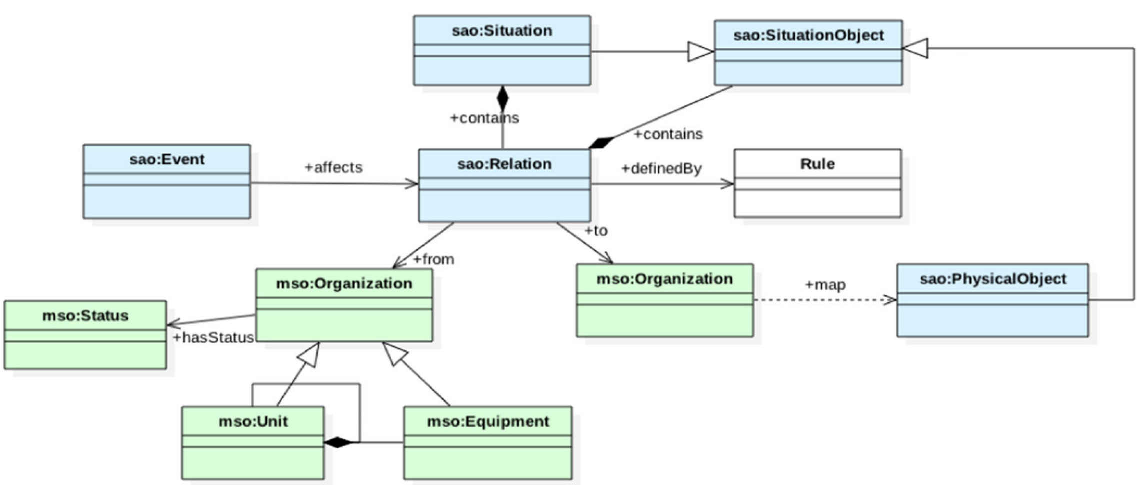


Fig. 9 *SAO:Relation* Class Diagram

as sides, units, equipment and the relations among them. The compositions of MSO include:

- Classes: *ScenarioID, Environments, ForceSides, Organizations, Overlays* and *TechnicalGraphs* are defined to describe the compositions of a scenario. The subclass relations can be defined using *rdfs:subClassOf* property.
- Data Properties: data properties are used to define class attributes of predefined datatypes, the following shows the *CountryCode* attribute of *ForceSide* with the type of integer using OWL data type property expression.

```
<owl:DatatypeProperty
  rdf:about="http://www.semanticweb.org/fppai/ontologies/MSO#CountryCode">
   <rdfs:domain
     rdf:resource="http://www.semanticweb.org/fppai/ontologies/MSO#ForceSide"/>
   <rdfs:range rdf:resource="&xsd;int"/>
</owl:DatatypeProperty>
```

- Object Properties: object properties are used to define relations between *ForceSides*, *Units* and *Equipment* classes. The following shows the *Allegiance* relation between *ForceSide* classes represented using object property.

```
<owl:ObjectProperty
  rdf:about="http://www.semanticweb.org/fppai/ontologies/MSO#Allegiance">
  <rdf:type rdf:resource="&owl;TransitiveProperty"/>
  <rdfs:range
    rdf:resource="http://www.semanticweb.org/fppai/ontologies/MSO#ForceSide"/>
  <rdfs:domain
    rdf:resource="http://www.semanticweb.org/fppai/ontologies/MSO#ForceSide"/>
</owl:ObjectProperty>
```

MSO is depicted using UML in Fig. 4. The most important part of MSO is about the organization composition such as sides, units, equipment and the relations among them.

This study will emphasize on the information fusion of the organization relationships. It is required to use ontological methodology to represent different kinds of relations like *ForceRelation, CommandRelation, SupportRelation* and *OrganicRelation* among sides, units and equipment. The relations among organizations are illustrated in Fig. 5.

3.2 Battle management ontology (BMO)

Battle Management Language (BML) and all its derivatives like C-BML (Coalition Battle Management Language) [39–41] are open standards used to express orders, reports and requests among C2 systems, simulation systems and real units. The primary goal of C-BML is to allow C2 systems to be able to task constructive simulations directly through a well-defined standard interface and to allow for simulation systems to report back to C2 systems through the same interface. It is intended to be an unambiguous and formal language for machine-to-machine communications that will eventually facilitate interoperation and allow automated processing.

The BML statements can be categorized into three types, i.e., orders, reports and requests. Each kind of statement focuses on the 5Ws: Who, What, Where, When and Why, which are associated with the verbs in the natural language text describing the action or event occurred in a military scenario. According to Blais et al. [42], C-BML ontology is needed to formalize the definition and meaning of common terms and formalize the doctrinal rules for *Orders* and *Reports* It can also ease the interoperability because of shared vocabulary and meaning, and allow the performing of convincing reasoning on operational semantics. The ontology should be constructed from C-BML with the associate introduction of JC3IEDM [43] ontology. JC3IEDM is a model that aims to enable the interoperability of systems and projects required to share C2 information and the information exchange mechanism.Valiente et al. [44] present an ontology-based situation awareness model that integrates SAW-CORE ontology into JC3IEDM OWL ontology using mapping to provide high-level information fusion with reasoning capabilities for C4ISR systems. They present a conceptual and preliminary results of mapping between JC3IEDM ontology and SAW-CORE ontology. However, there are obvious level-of-representation differences existing between these two ontology, we propose to introduce the battle management level ontology as an intermediate ontology to remove the conceptual gap.

We build a Battle Management Ontology (BMO) based on BML to define a structural description of orders and

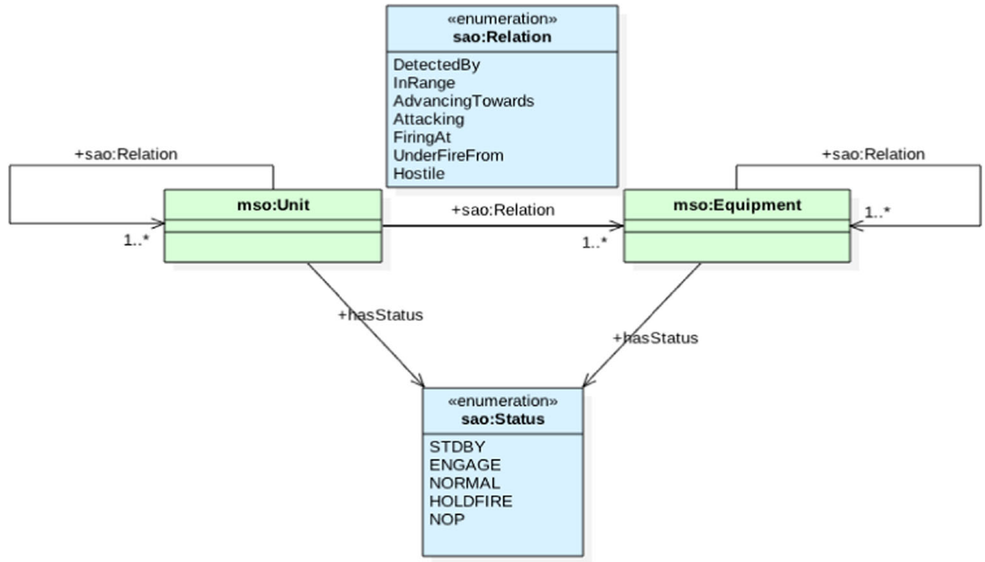


Fig. 10 Enumeration of *SAO:Relation* and *SAO:Status*

reports. This ontology formalizes the definition and meaning of common terms to allow reasoning on operational semantics. The structure of BMO report is depicted using UML as Fig. 6.

The BML report includes *ReporterWho*, *ReportedWhen* and the *ReportingData* (concerns about the information quality) and the content. A series of report type is defined in BML. In this research we will take the *EventWhatType*, *WhoAssociationType*, *WhoAffiliationType*, *WhoHoldingType*, *CandidateTargetListType* and *OrganizationStructureType* to convey the situation input information.

Pullen et al. [45] have tried to address the MSDL and C-BML integration issues about task organization definition and task definition to achieve a basic level of compatibility. In our approach, MSO not only is used to define the initial condition of the scenario, it also contains persistent organization information and sustained status that BMO description lacks. All the BMO *tasker* and *taskee* instances in the orders and reports should be consistent with this information defined in MSO. A mapping between the MSO *ObjectHandle* and BMO *OID* must be created and used to provide mutual referencing. In the other hand, orders and

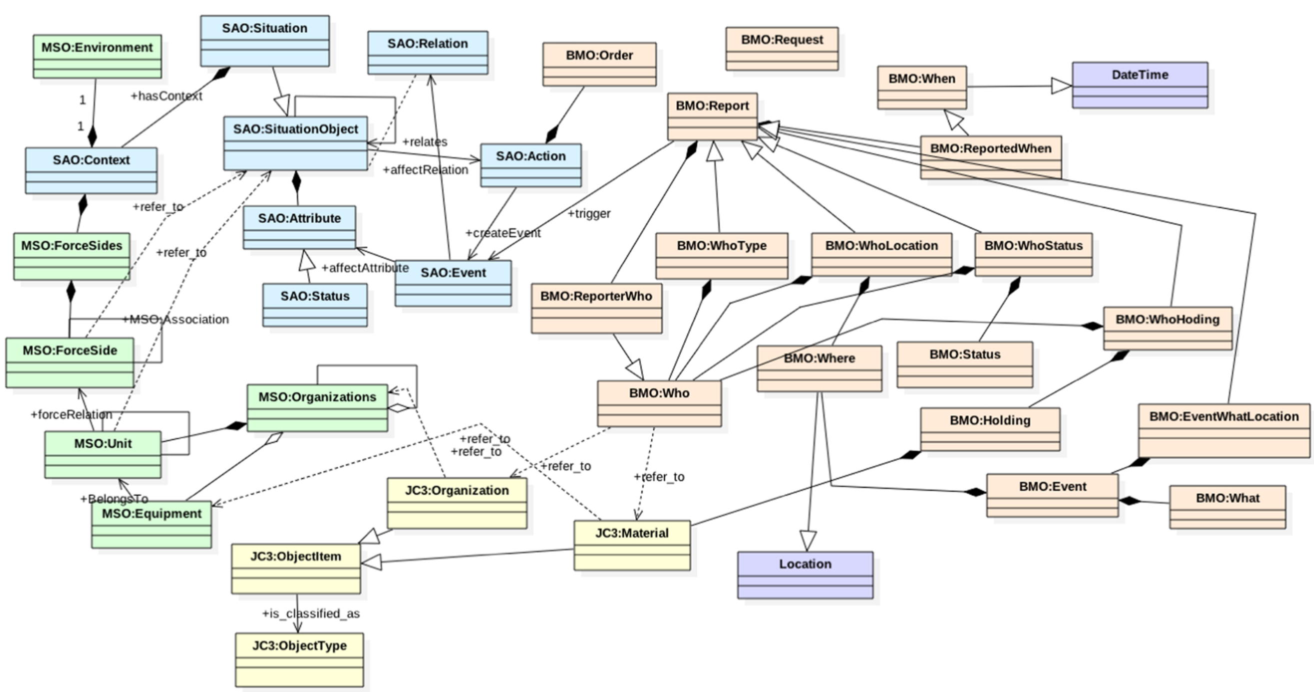


Fig. 11 Situation Awareness Ontology (SAO)

Table 3 SAO Elements

Name	Type	Description	Source
Situation	Class	The class to describe the holistic situation, it will be derived after inference using relations.	SAO
Relation	Class	The component used to carry situation information including situational relations and status of all the entities.	SAO
hasStatus	Data Property	The status description of entities.	SAO
Event	Class	The event description about the relations between entities occurred.	SAO
SituationObject	Class	A container to store lists of objects like targets, engaging units, threat units etc.	SAO
Equipment	Class	Information on all weapon platforms used in the scenario.	MSO
ForceSide	Class	Forces and sides for the scenario.	MSO
Organization	Class	The scenario specified organizations structure. It covers all units and equipment involved and their relations.	MSO
Unit	Class	Information of the units specified in a scenario.	MSO
ForceRelation	Object Property	Mission specific command relationship between unit and its commanding (higher) unit or force side.	MSO
CommandRelation	Object Property	The commanding relationship of units/equipment with their commanding unit.	MSO
SupportRelation	Object Property	The supporting relationship of units/equipment with their commanding unit.	MSO
ObjectItem	Class	Used to implement specific instances of objects described in C-BML messages. This object will refer to the units or equipment defined in MSDL using an UUID.	BMO
Report	Class	The situation reporting data generated during operation.	BMO
Order	Class	The order assigned to the organizations after decision making.	BMO
Who	Class	Subject and object description in C-BML reports.	BMO
Where	Class	Location description.	BMO
When	Class	Date and time description.	BMO

reports cannot be handled by MSO, we should design mechanisms to deal with the continuous flow of these kinds of data described using BMO.

3.3 Situation awareness ontology (SAO)

To formalize the concepts of situation awareness, the integration of the ontologies to describe the military scenario (MSO) and runtime command/control orders and reports (BMO) are required. We construct a Situation Awareness Ontology (SAO) to capture the knowledge associated with situation awareness that supports high- level reasoning by importing the MSO and BMO. BMO and MSO are considered as the domain ontology, and, furthermore, we also construct rules description upon the expressions of MSO,

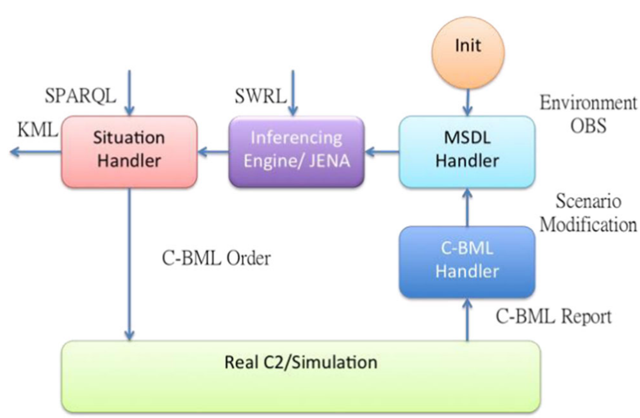


Fig. 12 Data flow of the Multi-layered Semantic Information Fusion Framework

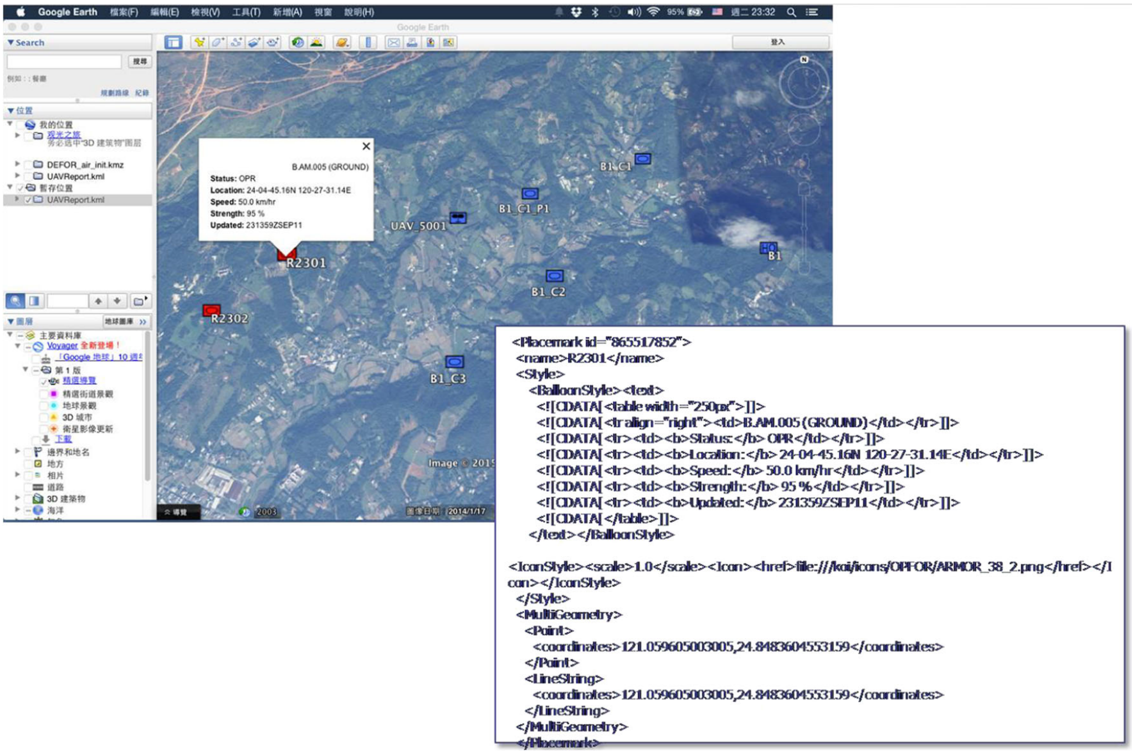


Fig. 13 Situation Display using Google Earth via KML data representation

BMO and SAO to express the domain doctrine rules for inference. The relations among onlogies are depicted in Fig. 7.

The semantic knowledge model designed for discrete events proposed in AgentOWL [13] and the SAWA ontology [20] are extended in this study to build core situation awareness ontology. The action-situation pairs produce events, and the situation changes can be entirely captured by discrete events in this model. The core situation awareness knowledge model comprises elements like *Situation(Si), SituationObjecst(So), Relations(Rel), Context(Cx), Actions(Ac), Events(Ev)* and *Attributes(Attr)*. The situation ontology is constructed based on this knowledge model as a 7-tuple:

$$SAO ::= (Si, So, Rel, Cx, Ac, Ev, Attr)$$

Basically, the *SituationObject* class describes physical entities involved, and they are directly mapped to the organizations including units and equipment defined using MSO. The physical objects can have attributes such as ID, type, location and status. The *SituationObject* class can also be used to handle the threat and target relation between organizations. The situations are objects derived after inference following the rules involving objects attributes and relations. Since there are causal and hierarchical relationships between situations, we use a *situation-as-object* approach to cover the ideas completely. The situation itself is a *SituationObject* instance and can also participate in relations.

$$SAO : Situation \sqsubseteq SAO : SituationObject$$

The information sources provide their information, e.g. reports, in an asynchronous manner since the information

Fig. 14 Excerpted Rule Expression

```
@prefix MSO: <http://www.semanticweb.org/fppai/ontologies/MSO#>.
[rule1: (?a MSO:CommandRelation ?b) (?b MSO:ForceRelation MSO:BLUFOR)->
   (?a MSO:ForceRelation MSO:BLUFOR)]
[rule2: (?a MSO:CommandRelation ?b) (?b MSO:ForceRelation MSO:OPFOR)->
   (?a MSO:ForceRelation MSO:OPFOR)]
[rule3: (?a MSO:CommandingSuperiorHandle ?b) (?c MSO:hasHandle ?b)
   (?c MSO:ForceSideRelation ?d) -> (?a MSO:ForceSideRelation ?d)]
[rule4: (?a MSO:CommandingSuperiorHandle ?b) (?c MSO:hasHandle ?b)
   (?c MSO:CommandingSuperiorHandle ?d)(?e MSO:hasHandle ?d)->
   (?a MSO:CommandRelation ?e)]
```

```
------------------------------------
| Unit             | Name          |
====================================
| MSO:ecf56cce0272 | "BN1_CO1_PL3" |
| MSO:3e868fd6d209 | "BN1_CO1_PL1" |
| MSO:67c564d1e9ac | "BN1_CO1_PL2" |
------------------------------------
```

Fig. 15 Inference and Query Result

are event triggered. Events will affect the attributes of situation objects and the relations among them or instantiate new situations. The descriptions of events including the 5W are all expressed using *BMO:Report* instances. The core ontology for situation awareness is depicted in Fig. 8.

The situation objects can correlate to each other with a *Relation* object. The relations between situation objects can be categorized into class relations (TBox) and instance relations (ABox). Moreover, the organizational relations among physical objects like command relation, support relation and force relation are described using the corresponding elements in MSO. The situational relations are described using SAO. Moreover, the new situation instances can be inserted into the ABox for further inference or processing. The situation relation class is illustrated using UML in Fig. 9.

We also define some primitive relations for situation expression like *FiringAt*, *Attacking*, *AdvancingTowards*, *Facing*, *UnderFireFrom*, *DetectedBy* and *InRegion* etc to define the relevant rules hence to generate new situation instances according to the rules during inference. The enumerations of *SAO:Relation* and *SAO:Status* are depicted in Fig. 10.

The significance of information has always been a joint function of the nature of the information and the context it is interpreted. We define the *Context* class to include ever-changing environmental descriptions. *Context* can be considered as a set of constraints in the reasoning process about a situation. It is described using the environmental descriptions in MSO and can also import external environmental descriptions.

The Situation Awareness Ontology is illustrated using UML in Fig. 11. It includes MSO, BMO, JC3IEDM ontology (JCO) and even some general ontology for the environment, date time and location description. As the operation progresses, the ever-changing scenario including force structure and their relations will be described according to MSO to provide a common picture. The reports following BMO will be generated when a certain event occurs. The agents can query their information provider for updates and, after knowledge processing like reasoning and inference, create reports for their information consumers according to their roles. Certain situation object will be generated based on the result of inference.

Protégé (http://protege.stanford.edu/) is used to generate and edit the Situation Awareness Ontology. Part of the classes and relations defined in SAO are listed in Table 3 about their descriptions and sources.

3.4 Rules

Definition 1 (Domain Knowledge Representation)

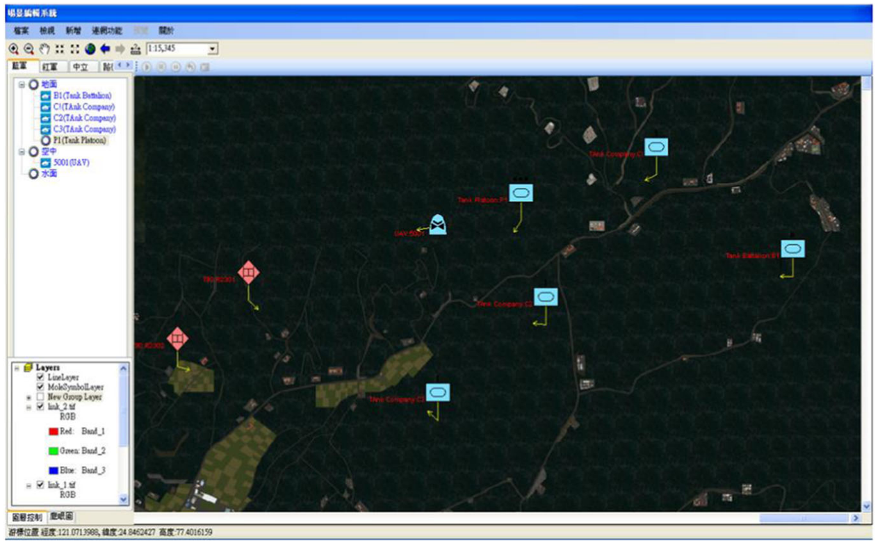
The Knowledge Representation K is a tuple **K=<O,R>**, where

- O denotes Ontology.
- R denotes Rules.

Fig. 16 DBpedia Query Result

Query	PREFIX p: <http://dbpedia.org/property/>
	PREFIX dbx: <http://dbpedia.org/resource/>
	PREFIX dbp: <http://dbpedia.org/property/>
	PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
	SELECT ?Power ?Arm WHERE
	{
	dbx:T-72 dbp:enginePower ?Power .
	dbx:T-72 dbp:secondaryArmament ?Arm .
	}

Results	(?Arm = "* 7.62 mm PKT coax machine gun \n* 12.7 mm NSVT antiaircraft machine gun"@en) (?Power = "780.0"^^<http://dbpedia.org/datatype/horsepower>) ?UnitA sao:ThreatUnitName ?ThreatUnitName . }
Results: DBpedia SNORQL	




```
<CGF Index="5" Name="T90" CallSign="R2301" ForceID="Hostile" Code_2525B="SH6PEVATL---CH6">
  <Position Lon="121.0596050030005" Lat="24.8483604553159" Alt="0" />
  <Attitude Pitch="0" Roll="0" Yaw="130.531891" />
  <EntityType Kind="1" Domain="1" Country="222" Category="1" Subcategory="9" Specific="0" Extra="0" />
  <Velocity>0</Velocity>
  <Consumable>
    <Fuel>0</Fuel>
    <Weapon Type="Cannon" Amount="20" />
    <Weapon Type="MachineGun" Amount="500" />
  </Consumable>
  <Combat Fire="1" ShootingRange="1000" ShootingInterval="100" DesignatedTarget="1" />
  <Settings OnGround="0" RoadRoute="1" CollisionOverride="0" KillOverride="0" SkillLevel="1" Personality="Active" ActivationDelay="0" />
</CGF>
```


Fig. 17 Scenario Editor

The domain knowledge required for situation awareness are of two types: (1) knowledge about what classes or objects, attributes and relations are possibly relevant and (2) what conditions must exist among the objects and their attributes for a given relation to hold true. We propose the use of OWL ontology for the first requirement – a choice that also provides a data representation in terms of instance annotations. The rules constructed on the basis of the ontologies are defined to satisfy the second requirement.

Ontology, augmented with more complex concepts like rules, can provide not only ontologybased reasoning but also the rulebased inference. OWL can represent simple implication such as subsumption, but it has no mechanism for defining arbitrary, multi-element antecedents. The realization of a monotonic rule layer on top of the ontology layer can drive works on hybrid reasoning, combining description logics with Horn logic. In our interested domain, there are concepts in the form of logical rules that OWL cannot express. These rules can be formalized using a computer understandable rule language compatible with OWL. We express military doctrine into Horn-like rules and encode these rules using rule language. The conceptual rules

Table 4 Original Scenario Description

Item	Description
1	The scenario include two sides : BLUFOR side and OPFOR side. The BLUFOR side and OPFOR side are in hostile relation.
2	BLUFOR 1st Battalion (BA1) is the top commanding unit.
3	BA1 is in command and control of Company 1 (CO1), Company 2 (CO2) and Company 3(CO3)
4	Company 1 (CO1) is in command and control of UAV platoon 1(PL1), anti-armor platoon 2(PL2) and armor platoon 3(PL3)
5	BA1 CO1 PL1 is equipped with UAV of type RQ-1A Predator named as RQ-1A-1, and platoon PL1 controls this UAV.
6	1st Battalion defines an organization structure based on associations

```
<ForceSide>
  <ObjectHandle>aa66cb1a11a6</ObjectHandle>
  <ForceSideName>BLUFOR SIDE</ForceSideName>
  <AllegianceHandle>aa66cb1a11a6</AllegianceHandle>
  <Associations>
    <Association>
      <AffiliateHandle>a3588a4b3cda</AffiliateHandle>←OPFOR
      <Relationship>HO</Relationship>
    </Association>
    <Association>
      <AffiliateHandle>e5e9f834031c</AffiliateHandle>
      <Relationship>NEUTRL</Relationship>
    </Association>
  </Associations>
</ForceSide>
```

Fig. 18 ForceSide Expression

embedded in OWL/RDF expression and externally custom defined rules can be combined and handled by the inference engine to derive additional RDF assertions. The *GenericRuleReasoner* provided by Apache Jena [46] is used in this study to produce situation information entailed from context description and doctrine rules.

4 Multi-layered semantic information fusion framework

Information fusion can be achieved by exchanging ontology based machine-understandable information. The proposed Multi-layered Semantic Information Fusion Framework is designed as a threelayered architecture:

- Input layer: the source data like MSDL scenario description, C-BML report, and even the external linked data for auxiliary entity description will be collected and processed in this layer. The heterogeneous data will be transformed into RDF based description according to MSO and BMO in this layer.
- Middle layer: this layer provides functions such as reasoning, inferencing, query and prediction. The MSO, BMO and SAO are used to inference and generate situation awareness information. Moreover, these pieces of information could be fed back into the system as a loop for further inference.
- Output layer: this layer transforms the information generated in the middle layer into the situation awareness system (such as common operation picture) using Keyhole Markup Language (KML) or generate orders according to BML standards.

4.1 Data Processing Procedure

The data flow of the multi-layered semantic information fusion framework is illustrated in Fig. 12. C-BML reports are continuously fed into the C-BML handler and then to update the status of the RDF based scenario description. The information is processed using the Jena inference engine to discover the implicated facts according to the rules. All the explicit and implicit facts are fed into the situation processing unit to generate a common operation picture or furthermore, the suggested plan. The following procedures are executed in a cyclic manner:

1. Read and parse scenario file expressed in MSDL and, as the operation goes, process the received C2 messages to update the situation periodically.
2. Scenario MSDL and C2 messages after processing are integrated and converted into RDF instance document. Physical computations are also performed to determine the distance, range, heading relations. In this phase, the algorithm of sensor data fusion (level 1), clustering and force structure recognition (level 2) used in [47] are implemented to generate situation descriptions. Some relations like *In-Range, AdvancingForwards* may also be produced in this phase and related RDF descriptions are also produced.
3. Jena's *GenericRuleReasoner* is called to perform reasoning using custom doctrine rules and the instance RDF document Details of the inference process are described in Section 4.2.
4. The query criteria are defined using SPARQL (SPARQL Protocol and RDF Query Language) [48] to retrieve and manipulate the in-memory triples after inference.
5. Since KML[1] (keyhole markup language) has been widely accepted as a standard for visualization of geospatial information we output the situational data using KML such that the information can be visualized using open and standard 3D platforms like Google Earth. The Java API for KML (JAK[2]) is exploited to transform situation objects into KML format. All the situational elements, like unit and equipment, are described and depicted using the *PLACEMARK* element of KML. Each object has the corresponding symbol display using the *<IconStyle>* to define the icon image following MilStd 2525C[3] symbology. We also use the *CDATA* element to convey the situation information. *Google Earth* is then used as the visualization platform to display the whole situation and update the information as a common operation picture (COP) as illustrated in Fig. 13.

[1] https://developers.google.com/kml/

[2] http://labs.micromata.de/projects/jak.html

[3] http://www.dtic.mil/doctrine//doctrine/other/ms_2525c.pdf

Fig. 19 Unit Expression

```
<Unit>
  <ObjectHandle>3e868fd6d209</ObjectHandle>
  <SymbolIdentifier>S-G-UH-----C---</SymbolIdentifier>
  <UnitSymbolModifiers>
    <UniqueDesignation>BN1 CO1 UAV PL1</UniqueDesignation>
    <HigherFormation>BN1 CO1</HigherFormation>
    <Echelon>PLATOON</Echelon>
    <CombatEffectiveness>GREEN</CombatEffectiveness>
  </UnitSymbolModifiers>
  <Disposition>
        …
  </Disposition>
  <Relations>
    <ForceRelation>
      <ForceRelationChoice>UNIT</ForceRelationChoice>
      <ForceRelationData>
       <CommandRelation>
            <CommandingSuperiorHandle>5f30aba1d664</CommandingSuperiorHandle>
          <CommandRelationshipType>ORGANIC </CommandRelationshipType>
         </CommandRelation>
      </ForceRelationData>
    </ForceRelation>
  </Relations>
</Unit>
```

4.2 Inference and Query

Jena inference engine is employed to perform reasoning and query using custom rules, OWL ontology definitions, instance RDF assertions and SPARQL queries The custom rules are defined according to domain doctrines The following demonstrates unit relations as an example:

- Rule1: If unit A is under command by unit B, and unit B belongs to the BLUFOR side, then unit A also belongs to the BLUFOR side.
- Rule2: If unit A is under command by unit B, and unit B belongs to the OPFOR side, then unit A also belongs to the OPFOR side.
- Rule3: If the commanding superior handle of unit A is B, unit C has Handle B and C belongs to force side D, then A belongs to force side D.
- Rule4: If the commanding superior handle of unit A is B, unit C has Handle B and the commanding superior handle of unit C is D and unit E has handle D, then unit A is under command by unit E.

These rules expressed in Jena Rule Form are listed in Fig. 14.

We can use SPARQL to query the result after inference according to the rules listed above. For example, we want to obtain the handle of units whose echelon are 'platoon' and belong to the BLUFOR side. Initially, only the units at the battalion echelon have their force side defined in the

Table 5 Sequence and Contents of C-BML Reports

Sequence Number	Report Expression
1	Company CO1 orders UAV Platoon (PL1) to do a tactical air reconnaissance task at a given location.
2	UAV platoon continuously reports the location, status and fuel level of the UAV at a given rate.
3	UAV platoon reports the discovery of Vehicle_1026 with hull number R2301 of type T-72(Type_1025) using *WhoTypeType* report.
4	Battalion BN1 received intelligence report from an external source about existence of 2 enemy organizations: Organization 2001 of type 100 (Company) with name RC1, Organization 2002 of type 100 (Company) with name RC2
5	BN1 received intelligence report about Armor Platoon 2001 owns vehicle R2301 and Platoon 2001 owns R2302.
6	UAV Platoon PL1 continuously reports location of Vehicle_1026at a given rate.
7	UAV Platoon PL1 reports 2 more T-72 Vehicles and in Wedge formation.
8	UAV Platoon reports Armor Vehicle (Vehicle_1026) in OPR status
9	UAV Platoon reports an estimated time of contact

Fig. 20 Report Expression

```
<Report xsi:type="WhoTypeType">
   <ReporterWho>
     <OrganisationRef xsi:type="UnitRef">
       <OID>3e868fd6d209</OID>
     </OrganisationRef>
   </ReporterWho>
   <ReportedWhen xsi:type="ReportedWhenAbsoluteTimingType">
     <ReportingDatetime>20100525163000.000</ReportingDatetime>
     <EffectiveStartDatetime>20100525163000.000</EffectiveStartDatetime>
   </ReportedWhen>
   <ReportingData>
     <OID>1044</OID>
     <ReportingDataCategoryCode>REP</ReportingDataCategoryCode>
     <CredibilityCode>RPTFCT</CredibilityCode>
     <ReliabilityCode>A</ReliabilityCode>
   </ReportingData>
   <Who>
     <ObjectItem xsi:type="OtherMateriel">
       <OID>1026: Vehicle</OID>
       <NameText>Vehicle_1026</NameText>
       <HullNumberText>R2301</HullNumberText>
     </ObjectItem>
   </Who>
   <ObjectTypeRef xsi:type="VehicleTypeRef">
     <OID>1025: T-72 Tank</OID>
   </ObjectTypeRef>
</Report>
```

MSDL description file. After the inference, we can have all the hierarchical organizations with their force-side defined according to our rules listed above. The results of inference and query are listed below (Fig. 15).

The above test demonstrates the feasibility of our framework using only MSO. Section 5 provides complete situation awareness demonstration cases.

4.3 DBPedia query

The ontology-based information fusion can be amplified to include more publicly defined ontologies. We can use SPARQL to query the semantic relationships and properties associated with Wikipedia resources via DBpedia Specific attributes of target entity or the geographical information can be obtained to assist the estimation of target movements, owing to our ontology/semantic web based design.

The DBpedia [49] project leverages large knowledge source of Wikipedia by extracting structured information from Wikipedia and making this information accessible on the Web. The SPARQL query can also be used to access the public DBPedia dataset via the SPARQL endpoint, i.e., http://dbpedia.org/sparql. The resulting data can be automatically retrieved and fed into the situation inference mechanism to increase the accuracy of the prediction and judgment. For example, we can query the engine power and secondary armament of T-72 tank and the result can be input into our inference engine to assess the situation and predict the actions of the targets. Snapshot of the result and DBpedia SNORQL public online query explorer (http://dbpedia.org/snorql) is depicted in Fig. 16.

5 Experimental results

A UAV detection scenario is presented to demonstrate how the proposed methodology and concept model can be used to perform information fusion and then to provide situation awareness. The UAV is equipped with sensors, when targets are in the detection range, a report including the amount, position, direction of moving and velocity will be generated and then sent to the command center agent. The agent of the command center will make identification

Fig. 21 Excerpted Rule Expression

```
[rule12: (?a sao:AdvancingTowards ?b) (?a sao:Hostile ?b) ->
   (?b sao:hasStatus "STDBY")]
[rule13: (?b sao:InRange ?a) (?a sao:hasStatus "STDBY") ->
   (?a sao:hasTargetUnit ?b)]
[rule15: (?a mso:CommandRelation ?c) (?b mso:CommandRelation ?c) ->
   (?a mso:SupportRelation ?b)]
[rule16: (?a mso:SupportRelation ?b) (?b sao:hasStatus "ENGAGE") ->
   (?a sao:hasStatus "STDBY")]
[rule17: (?a sao:AdvancingTowards ?u) (?a sao:Hostile ?u) makeTemp(?x) ->
   (?x rdf:type sao:Situation) (?x sao:hasType "ALERT") (?x sao:ID "1011")]
```

Fig. 22 Case 1 Results

Rules:	`[rule10:(?a bmo:HullNumberText ?b) (?c mso:EquipName ?b) (?c mso:UnitOwnerHandle ?d) (?e mso:ObjectHandle ?d) (?f sao:Hostile ?e) -> (?f sao:hasThreatUnit ?e)]` `[rule11:(?a sao:hasThreatUnit ?b) (?b mso:UnitName ?c) -> (?a sao:ThreatUnitName ?c)]`
Query:	`SELECT ?Name ?ThreatUnitName WHERE { ?UnitA mso:UnitName ?Name. ?UnitA sao:ThreatUnitName ?ThreatUnitName . }`
Results:	Name / ThreatUnitName: "BN1" "RC2"; "BN1" "RC1"

```
Rules:
[rule10:(?a bmo:HullNumberText ?b) (?c mso:EquipName ?b)
   (?c mso:UnitOwnerHandle ?d) (?e mso:ObjectHandle ?d)
   (?f sao:Hostile ?e) -> (?f sao:hasThreatUnit ?e)]
[rule11:(?a sao:hasThreatUnit ?b) (?b mso:UnitName ?c) ->
   (?a sao:ThreatUnitName ?c)]

Query:
SELECT ?Name ?ThreatUnitName
WHERE
{
   ?UnitA mso:UnitName ?Name.
   ?UnitA sao:ThreatUnitName ?ThreatUnitName .
}

Results:
-------------------------------
| Name     | ThreatUnitName    |
===============================
| "BN1"    | "RC2"             |
| "BN1"    | "RC1"             |
-------------------------------
```

and evaluation about the situation. It will also predict the motivation and intention of the tanks. If the attacking tendency is confirmed, the judgment about how to react is made. The commander agent will decide to assign a unit to engage depending on the firing range, status of readiness, sector of responsibility and the scale of the invader. The engagement command will be produced to define the engagement unit, target assignment, and weapons. A protective action order will be sent to the commanding headquarter of the engaging unit, and prepare-to-engage order will be directed to the units that have supportive or supply relations with the unit to standby or maneuver.

First of all, we develop a scenario editing application that provides a GUI for users to generate the scenario. The hierarchical organization of the forces/units/equipment and their deployment positions are described using MSDL Fig. 17 demonstrates the GUI of the scenario editor.

5.1 Scenario description

The original scenario are described in Table 4. It contains only the BLUFOR organization and equipment compositions. The OPFOR compositions will be built up according to the reports generated as the simulation progresses (Table 4).

The above organizations and their relationships can be expressed using MSDL (Figs. 18 and 19), all units and equipment are identified with a unique object handle.

- ForceSide: description of the BLUFOR side, with the expression of its hostile relation to the OPFOR side using *Association, AffiliateHandle* and *Relationship* elements.
- Unit: Platoon 1 is subordinate to Company 1 then to Battalion 1.
- Equipment: UAV RQ-1A-1 is controlled by platoon 1.

5.2 C-BML report

In our experiment, various types of BML reports are generated continuously as the simulation progresses to represent the situation evolutions. Table 5 describes the sequence and content expressions of the reports.

Fig. 23 Case 2 Result

```
Rules:
[rule12:(?a sao:AdvancingTowards ?u) (?a mso:EquipName ?b)
   (?c bmo:HullNumberText ?b) (?c bmo:Status "OPR") ->
   (?u sao:hasStatus "STDBY")]

Query:
SELECT ?Name ?Status
WHERE
{
   ?UnitA mso:UnitName ?Name .
   ?UnitA sao:hasStatus ?Status .
}

Results:
--------------------------
| Name         | Status    |
==========================
| "BN1_CO3"    | "STDBY"   |
--------------------------
```

The *WhoTypeType* report about the discovery of Vehicle_1026 are listed in Fig. 20.

5.3 Rules

As the exercise evolves, the situation awareness can be obtained through continuous information fusion and inference following doctrine rules. The commanding and supporting relations are reasoned from the scenario description file and the status will change as the operation progresses.

Specific situation objects may be instantiated during situation assessment. We use Jena forward chaining inference and built-in function (e.g. *makeTemp*) to create blank node individuals of situations dynamically in the conclusion of Jena rules. Excerpted rule expression are listed in Fig. 21 using Jena rules format.

- If unit A is advancing towards unit B *(SAO:Relation)*, unit A is hostile to unit B *(SAO:Relation)*, then unit B has to raise its status to STANDBY *(SAO:Status)*.
- unit B is in firing range of unit A *(SAO:Relation)*, and unit A has been in the STANDBY status *(SAO:Status)*, then unit A will take unit B into its target list *(SAO:Relation)*.
- If unit A has supportive relation to unit B *(MSO:Relation)*, and unit B has been in the ENGAGE status *(SAO:Status)*, unit A will have to be in the STANDBY status *(SAO:Status)*.
- If unit A is advancing towards unit U *(SAO:Relation)*, and unit A are hostile to unit U *(SAO:Relation)*, then a situation instance will be created with type ALERT and with a given ID *(SAO:Situation)*.

5.4 Situation awareness results

There are three cases provided to demonstrate the situation awareness results derived from our framework according to the JDL model. The related rules, SPARQL query expressions and results are listed respectively.

Case 1 Sub-Object Data Assessment: Tank Vehicle_1026 and Vehicle_1027 are detected and identified. The deduced fact is that OPFOR platoon RC1and RC2 are detected and they are conceived as threat units of BLUFOR BN1. This fact is derived from the existence of the two tanks, and these two tank entities belong to company RC1 and RC2 respectively The rules, SPARQL query and result are illustrated in Fig. 22.

Case 2 Object Assessment: After a computation considering the relative distance and direction of movement, it is reported that OPFOR Vehicle_1026 is moving towards BLUFOR Company 3 and its status is “OPR.” In the rule listed in Fig. 23, the *sao:AdvancingTowards* object property is used to express the moving towards behavior between equipment and units. As a result, according to the rule, the Company 3 has to be in the status of “STDBY”.

The results are illustrated in Fig. 23. In this case, the observed status of the OPFOR is expressed using *bmo:Status* property, and the *sao:hasStatus* property is used to represent the derived status after inference.

Case 3 Situation Assessment: The Vechicle_1026 is reported to be “*InRange*” of BLUFOR Company 3, the

Fig. 24 Case 3 Result

Rules:
```
[rule12:(?a sao:AdvancingTowards ?b) (?a sao:Hostile ?b) ->
   (?b sao:hasStatus "STDBY")]
[rule13:(?b sao:InRange ?a) (?a sao:hasStatus "STDBY") ->
   (?a sao:hasTargetUnit ?b)]
[rule14:(?a sao:hasTargetUnit ?b) -> (?a sao:hasStatus "ENGAGE")]
[rule15:(?a mso:CommandRelation ?c) (?b mso:CommandRelation ?c) ->
   (?a mso:SupportRelation ?b)]
[rule16:(?a mso:SupportRelation ?b) (?b sao:hasStatus "ENGAGE") ->
   (?a sao:hasStatus "STDBY")]
```

Query:
```
SELECT ?Name ?Status
WHERE
{
   ?UnitA mso:UnitName ?Name .
   ?UnitA sao:hasStatus ?Status .
}
```

Results:
```
-----------------------------
| Name          | Status     |
=============================
| "BN1_CO2"     | "STDBY"    |
| "BN1_CO1_PL1" | "STDBY"    |
| "BN1_CO1_PL2" | "STDBY"    |
| "BN1_CO1"     | "STDBY"    |
| "BN1_CO3"     | "ENGAGE"   |
-----------------------------
```

Company 3 enters "ENGAGE" status. The results are illustrated in Fig. 24.

From the above, we can also obtain that not only BLUFOR Company 3 enters the status of *"ENGAGE"*, but also the Company 1 and Company 2 and their subordinates also upgrade its status to *"STDBY"* due to their support relation with Company 3. The result demonstrates that as the situation progresses, the status of each unit should change according to their role and the command or support relations.

6 Conclusion and future works

To the best of our knowledge, this paper is the first study about the feasibility of using ontology based semantic web technologies to fuse information following different specifications to address the military situation awareness issues. We propose MSO for the descriptive architecture of scenario and situation based on MSDL, and BMO for the handling of orders and reports refers to C-BML. The Situation Awareness Ontology (SAO) that integrates MSO, BMO, JCO and publically defined ontologies are proposed as core ontology. Horn clause type rules are used to express military rules. A multi-layered semantic information fusion framework is developed to integrate situation descriptions based on the above ontologies, semantic web technologies and rule-based reasoning to achieve autonomous information fusion. Full RDF-based expressions are constructed to improve interoperability and reusability. The experimental results demonstrate the correctness and power of the methodology.

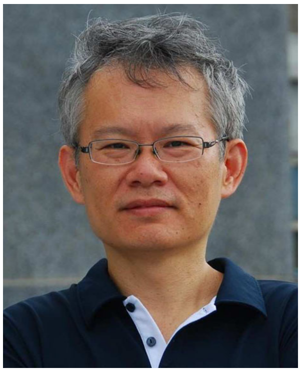
The proposed architecture is a flexible middleware solution that provides a mechanism to allow users define inference rules and queries according to their requirements. Tactically significant operational concepts can be introduced to conduct contextual intent assessment and fed into our framework for further decision making. Current system just proves the feasibility of the concept of using ontological framework to perform information fusion. It needs further implementations to realize automatic processing and the goal is to achieve the generation of plans or orders.

For future works, an information fusion framework that integrates ontology-based knowledge representation model, semantically enhanced agent communication infrastructure and multi-agent architecture is under development. The semantically enhanced communication infrastructure we propose in [50] is implemented to improve interoperability in a multi-agent environment. The ontology-based knowledge framework introduced in this study is adopted not only for the agent's internal knowledge representation but also for the external communication. The agents orchestrate following JDL model to share ontology-enabled domain information, thus to build a consistent, holistic picture of the situation. Both the Jena semantic web toolkit and JADE [51] agent framework are exploited to implement a multi-agent information fusion framework with interoperability enhancement owing to semantic web technologies. Fuzzy event correlation and fuzzy situation assessment will also be incorporated. Fuzzy markup language (FML) related methodology used in [52, 53] to describe the knowledge base and rule base of the fuzzy inference will be studied. Moreover, humans are another important source of information in the situation awareness system, the natural language based document or oral report should be taken into integration. The methodologies for query document retrieval [54] or ontology-based speech-act identification [55] have provided solid theoretical bases and can be considered for further implementation.

References

1. Endsley MR (1995) Toward a theory of situation awareness in dynamic systems. Human Factors: The Journal of the Human Factors and Ergonomics Society 37.1
2. Steinberg AN, Bowman CL, White FE (1999) Revisions to the JDL data fusion model. In: AeroSense'99. International Society for Optics and Photonics
3. Berners-Lee T, Hendler J, andLassila O (2001) The Semantic Web – A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Sci Am 284(5):34–43
4. Javier B et al (2012) VEACON: A Vehicular Accident Ontology designed to improve safety on the roads. J Netw Comput Appl 35.6:1891–1900
5. Dongli Y, Suihua W, Ailing Z Traffic accidents knowledge management based on ontology. In: Sixth international conference on fuzzy systems and knowledge discovery, 2009. FSKD'09. Vol. 7. IEEE, p 2009
6. Baumgartner N, Gottesheim W, Mitsch S, Retschitzegger W, Schwinger W (2010) BeAware!, - Situation Awareness, the Ontology-Driven Way. Data Knowl Eng 69:1183–1193
7. Baumgartner N et al (2014) A tour of BeAware–A situation awareness framework for control centers. Information Fusion 20:155–173
8. Amit S, Henson C, Sahoo SS (2008) Semantic sensor web. Internet Computing IEEE 12.4:78–83
9. Adida B et al (2008) RDFa in XHTML: Syntax and processing. Recommendation W3C
10. Roda F, Musulin E (2014) An ontology-based framework to support intelligent data analysis of sensor measurements. Expert Systems with Applications 41.17:7914–7926
11. Little EG, Rogova GL (2009) Designing ontologies for higher level fusion. Information Fusion 10.1:70–82
12. Smith B (2002) Basic Formal Ontology. http://ontology.buffalo.edu/bfo
13. Laclavik M et al (2012) Agentowl: Semantic knowledge model and agent architecture. Comput Inf 25.5:421–439

14. Balaji BS, Prabhu TN, Karthikeyan NK (2012) SACoSS-semantic Agent Based System for Cloud Service Suggestion Using Cloud Service Ontology. Jmynal Comput Inf Syst 8.21:8763–8773
15. Lewis L, Buford J, Jakobson G (2009) Inferring threats in urban environments with uncertain and approximate data: an agent-based approach. Appl Intell 30.3:220–232
16. Kokar MM, Matheus CJ, Baclawski K (2009) Ontology-based situation awareness. Inf fusion 10.1:83–98
17. Barwise J, Perry J (1983) Situations and Attitudes. MIT Press
18. Matheus CJ, Kokar MM, Baclawski K (2003) A core ontology for situation awareness. In: Proceedings of the Sixth International Conference on Information Fusion. Vol. 1
19. Matheus CJ et al (2005) Using SWRL and OWL to capture domain knowledge for a situation awareness application applied to a supply logistics scenario. Rules and Rule Markup Languages for the Semantic Web. Springer Berlin Heidelberg:130–144
20. Matheus CJ et al (2005) SAWA: An assistant for higher-level fusion and situation awareness. Defense and Security. International Society for Optics and Photonics
21. Matheus CJ et al (2005) Lessons learned from developing SAWA: A situation awareness assistant. In: 2005 8th International Conference on Information Fusion, Vol. 2. IEEE
22. Bowman M, Lopez A, Tecuci G (2001) Ontology development for military applications. In: Proceedings of the Thirty-ninth Annual ACM Southeast Conference. ACM Press. Vol. 36
23. Boury-Brisset A-C Ontology-based approach for information fusion. In: Proceedings of the Sixth International Conference of Information Fusion, 2003. Vol. 1. IEEE, p 2003
24. Smart PR et al (2005) Knowledge-based information fusion for improved situational awareness. In: 2005 8th International Conference on Information Fusion, Vol. 2. IEEE
25. Gómez-Romero J et al (2015) Context-based multi-level information fusion for harbor surveillance. Inf Fusion 21:173–186
26. Farinelli A et al (2012) Cooperative situation assessment in a maritime scenario. Int J Intell Syst 27.5:477–501
27. Migueláñez E et al (2011) Semantic knowledge-based framework to improve the situation awareness of autonomous underwater vehicles. IEEE Trans Knowl Data Eng 23.5:759–773
28. Chen H et al (2004) Soupa: Standard ontology for ubiquitous and pervasive applications. In: The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. IEEE
29. Wang XH et al (2004) Ontology based context modeling and reasoning using OWL. In: Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, 2004. IEEE
30. Yau SS, Liu J (2006) Hierarchical situation modeling and reasoning for pervasive computing. In: The Fourth IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, 2006 and the 2006 Second International Workshop on Collaborative Computing, Integration, and Assurance. SEUS 2006/WCCIA 2006. IEEE
31. Chen H, Finin T, Joshi A (2003) An ontology for context-aware pervasive computing environments. Knowl Eng Rev 18.03:197–207
32. Chen H, Finin T, Joshi A (2003) An intelligent broker architecture for context-aware systems. PhD proposal in computer science, University of Maryland. USA, Baltimore
33. Scherp A et al (2012) A core ontology on events for representing occurrences in the real world. Multi Tools Appl 58.2:293–331
34. Norbert B, Retschitzegger W (2006) A survey of upper ontologies for situation awareness. In: Proc. of the 4th IASTED International Conference on Knowledge Sharing and Collaborative Engineering, St. Thomas, US VI
35. Snidaro L, Visentini I, Bryan K (2015) Fusing uncertain knowledge and evidence for maritime situational awareness via Markov Logic Networks. Inf Fusion 21:159–172
36. Snidaro L, Garcia J, Corchado JM (2015) Context-based information fusion: A survey and discussion. Inf Fusion 25:16–31
37. Bermejo–Alonso J, Sanz R., López I (2006) A Survey on Ontologies for Agents. From Theory to Practice. ASLab-ICEA-R-2006, vol 2
38. MSDL MSDL - Military Scenario Definition Language, 2010. Available from: http://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=30830
39. Blais C, Galvin K, Hieb M (2005) Coalition Battle Management Language (C-BML) Study Group Report. IEEE Fall Simulation Interoperability Workshop, Orlando FL
40. Blais C et al (2009) Coalition Battle Management Language (C-BML) Phase 1 Specification Development: An Update to the M&S Community. Fall Simulation Interoperability Workshop
41. Blais C, Galvin K, Hieb M (2011) Coalition Battle Management Language (C-BML) Study Group Report
42. Blais C, Turnitsa C, Gustavsson P (2006) A strategy for ontology research for the coalition battle management language (C-BML) product development group. Fall Simulation Interoperability Workshop, Orlando
43. JC3IEDM Joint C3 Information Exchange Data Model (JC3IEDM), 2010; Available from: http://www.mip-site.org/publicsite/04-Baseline_3.0/JC3IEDM-Joint_C3_Information_Exchange_Data_Model/
44. María-Cruz V et al (2011) An ontology-based integrated approach to situation awareness for high-level information fusion in C4ISR. Advanced Information Systems Engineering Workshops. Springer Berlin Heidelberg
45. Pullen M et al (2012) MSDL and C-BML Working Together for NATO MSG-085. In: Proceedings of the 2012 Spring Simulation Interoperability Workshop
46. JENA (2007) Jena – A Semantic Web Framework for Java
47. Sycara K et al (2009) An integrated approach to high-level information fusion. Inf Fusion 10.1:25–50
48. Prud'Hommeaux E, Seaborne A (2008) SPARQL query language for RDF. W3C recommendation
49. Bizer C et al (2009) DBpedia-A crystallization point for the Web of Data. Web Semantics: science, services and agents on the world wide web 7.3
50. Pai F-P, Hsu I-C, Chung Y-C (2016) Semantic web technology for agent interoperability: a proposed infrastructure. Appl Intell 44.1:1–16
51. Bellifemine F, Poggi A, Rimassa G (1999) JADE–A FIPA-compliant agent framework. In: Proceedings of PAAM. Vol. 99. No. 97-108
52. Wang M-H et al (2010) Ontology-based multi-agents for intelligent healthcare applications. J Ambient Intell Humaniz Comput 1.2:111–131
53. Lee C-S, Wang M-H (2011) A fuzzy expert system for diabetes decision support application. IEEE Trans Syst Man Cybern Part B Cybern 41. 1:139–153. Cybernetics
54. Yeh J-F, Wu C-H, Chen M-J (2008) Ontology-based speech act identification in a bilingual dialog system using partial pattern trees. J Am Soc Inf Sci Technol 59.5:684–694
55. Yeh J-F et al (2008) Extended probabilistic HAL with close temporal association for psychiatric query document retrieval. ACM Trans Inf Syst 27.1:4



Fang-Ping Pai received his MS degree in Aerospace Engineering from National Chung-Kung University in 1989. He is now a PhD student in the Department of Computer Science at National Tsing-Hua University. His research interests include High Level Architecture (HLA), Military modeling and simulation, Multi-agent Systems and Semantic web technologies.

Lee-Jang Yang received his M.S. degree from the Institute of Applied Mechanics, National Taiwan University (NTU) in 1991, and his Ph.D. degree in Computer Science from National Chung-Hsing University in 2015. He has been working at National Chung-Shan Institute of Science and Technology in the area of intelligent computing technologies since 1991. His current research aims at the creation and study of Semantic Web for Linked Data, Big Data and information modeling.

Yeh-Ching Chung received a B.S. degree in Information Engineering from Chung Yuan Christian University in 1983, and the M.S. and Ph.D. degrees in Computer and Information Science from Syracuse University in 1988 and 1992, respectively. He joined the Department of Information Engineering at Feng Chia University as an associate professor in 1992 and became a full professor in 1999. From 1998 to 2001, he was the chairman of the department. In 2002, he joined the Department of Computer Science at National Tsing Hua University as a full professor. His research interests include parallel and distributed processing, cloud computing, and embedded systems. He is a senior member of the IEEE computer society.