Ontologies for Distributed Command and Control Messaging

Duc N. NGUYEN [a], Joseph B. KOPENA [a], Boon Thau LOO [b], William C. REGLI [a,1]
[a] *Drexel University, Philadelphia, PA, USA*
[b] *University of Pennsylvania, Philadelphia, PA, USA*

Abstract. This paper presents the development of a set of ontologies for use in messaging systems within military and emergency first responder command and control applications. These ontologies are used to annotate and subscribe to messages within a content delivery and service discovery middleware connecting sensors, services, and agents on the network. Ontologies serve an important role in this middleware in providing for flexible addressing and querying that increases the decoupling between senders and receivers afforded by the middleware. This is particularly important in the target applications, which feature heterogenous components and software versions spanning development organizations and deployment time. Discussed here are application uses, development methodology, evaluation criteria, and other notes on these ontologies and their creation. The goal of this paper is to provide useful input to others engaged in similar development efforts.

1. Introduction

Although arguably tantalizingly close to real world, commonplace use, computerized, handheld, mobile command and control systems for dismounted warfighters—troops on the ground—as well as for emergency first responders—police, fire, and emergency medical personnel—are still largely open areas of research and development. These applications aim to improve collaboration and situational awareness among such users by providing tools for geopositioning, map annotation, messaging, plan development and monitoring, media sharing, and other tasks. Many prototype systems have been developed and even deployed, but many hurdles remain to be overcome.

Among these are the challenging, dynamic environments in which these systems must operate, with minimal maintenance, reconfiguration, or deployment overhead. Great emphasis in developing such systems has been placed on the disruption prone nature of the wireless communication technology needed to meet mobility and platform requirements. High latency, low bandwidth, and frequent link disruptions are commonplace and require continual adaptation. Systems must also tolerate frequent hardware and other failures, as well as incorporation of dynamically introduced nodes and components.

Less commonly addressed but as important are issues of interoperability. Particularly in the case of first responder applications, it must be expected that systems must incorporate and work alongside components and systems developed by a number of ven-

[1] dn53@drexel.edu, tjkopena@cs.drexel.edu, boonloo@cis.upenn.edu, regli@drexel.edu

dors and configured and deployed by a variety of authorities, all with differing structure, policies, and procedures. Similarly, in many cases systems must conceivably interoperate with components developed and deployed well before the lifetime of current elements, such as sensor or communication infrastructure emplaced years before use.

The authors' OntoNet [9] project aims to address many of these issues through content delivery and service discovery middleware. By providing for message publishing and subscription, OntoNet intends to provide a flexible, abstract platform for connecting sensors, services, and agents on the network. Such middleware eases system development by reducing the need for application developers to directly address and develop protocols and techniques for networking in disruption prone environments.

OntoNet also promotes flexibility via a knowledge-based approach to message addressing and matching. Messages are tagged with metadata in the Resource Description Framework (RDF) [11], while subscriptions are registered and matched as Web Ontology Language (OWL) [5] class expressions. Sets of ontologies are used to structure that metadata and queries, providing supports for the messages, services, and other entities of particular application domains.

This paper describes the development of ontologies for use in OntoNet for military and first responder command and control applications. These applications are briefly presented, along with development methodology, evaluation criteria, and other notes on these ontologies and their creation. The goal of this paper is to provide useful input to others engaged in similar development efforts.

2. Motivating Scenarios

Although many elements of the OntoNet project are fairly generic, particular focus is given to supporting mobile, handheld command and control (C2) systems. Both settings share many similarities in terms of feasible computing platforms, challenging networking environments, and other aspects. Although different, they also share many core structures in their underlying data and messaging patterns, and consequently the ontologies developed to support messaging in those arenas. Both settings are briefly described here.

2.1. Military Command and Control

According to the Military Decision Making Process [8], the military planning process typically starts with a commander issuing orders to staff to gather data, intelligence, and issuing warning orders to lower level units involved in the operation. During this information gathering phase, the commander incorporates data and other information sources such as unit strength and readiness, terrain information, enemy capabilities, civilian areas, and past enemy activities. This usually requires collaboration among command staff and lower echelons along with querying external data sources. Upon completion of the planning phases, the orders are disseminated to subordinate units to be executed.

During operations monitoring, commanders monitor the situation from information flowing up from subordinate units and other information sources. As the operation evolves, commanders monitoring the situation collaborate between the echelons and issue updates and new orders. An alert such as an IED detection or enemy ambush issued from individual dismount soldiers is sent up to their superior commanders. Such alerts may necessitate a change in the plan and commanders will update subordinates accordingly. Events from external sources and higher echelons such as missile reports also get filtered down to lower echelons and may cause changes to plans.

Similarly, lower echelon units may exchange their own collaboration messages and situation update information among themselves. This may be as simple as automated, GPS-based position updates and vital statistics reports, or as complex as ad hoc plan annotations on common map displays, shared media such as video feeds,

2.2. Emergency First Responders

Many organizations maintain response teams for chemical, biological, or radiological (CBR) incidents, such as industrial or transport accidents. Unfortunately, current best practices largely use non-integrated and redundant tools applied via manual processes. Even among well provisioned teams, critical data collection is frequently performed via clipboard and pen—a difficult task in full protective suits. Automating these processes may significantly improve team performance and efficiency but faces several challenges.

Among these are basic network connectivity and management issues. A mix of mobile and fixed nodes may be in use, fielded by a number of different organizations and potentially using low power, low cost radios. Importantly, team members must be able to operate and work together before infrastructure may be deployed, as well as in environments where disconnections are frequent and they must operate independently. These constraints may imply complex, heterogenous network topologies spanning organizations and employing mobile ad hoc and mesh networking techniques as necessary while also utilizing infrastructure such as base stations as available to gain higher performance and wider resource accessibility.

2.3. Command and Control Relationship

A key aspect of OntoNet's approach is that sensors or software generate messages with associated metadata which is used to deliver those messages to the appropriate destination(s). This functionality can be used to support many tasks such as data and alert dissemination, service requests, and content retrieval. By addressing messages and destinations via declarative, ontology driven languages, the above interoperability challenges may be met with flexible, extensible, formal reasoning mechanisms. Applying that query based messaging paradigm in distributed, robust middleware also decouples message senders and receivers to a great degree, promoting rapid system integration and improved manageability.

3. Background and Related Work

section details related work on the development of ontologies for their respective domains. Military Ontologies have been created and incorporated into upper and mid-level ontologies. Sensor network ontologies are being developed to augment existing data formats for better automated inference in support for discovery, matching, composition, and search. These domains, while different, have similar qualities with regard to the command hierarchy, communications infrastructure, and data flows.

3.1. Military Ontologies

Anderson *et. al*[1] present an ontology of the basic categories and relationships needed to represent modern military organization. The key features of the modern military organization ontology is the echelon structure and the subordination properties associated with a modern military organization. This ontology was merged into the Suggested Upper Merged Ontology (SUMO) [12].

The typical *is_a* relationship implies a hierarchy of concepts. When applied to a modern military organization, the colloquial usage does not apply for creating hierarchies of military echelons. Military echelons are related using a *subEchelon* relationship. The *subEchelon* relationship state one object is a sub-echelon of another, for example, a Company is a *subEchelon* of a Battalion. Furthermore, there are military units that are considered equivalent levels in the military echelon. For example, Regiments (typically found in French brigades) are typically in the same echelon as Battalions, i.e., regiments and battalions are both sub-echelons of a Brigade. There are also differences in terminology; e.g., a Russian infantry division is different in strength than a US infantry division.

The resulting knowledge-base created from Bowman [4] *et. al*to aid commanders in developing courses of action with respect to principles of war and military operations. The goal for this work was to construct a full ontology to determine the focal point of an opposing force, termed the "center-of-gravity" of the enemy. This is accomplished with a large enough ontology such that sufficient inferences can be made. The goal of OntoNet is to provide reasoning capabilities that balance complexity and robustness of queries.

3.2. First Responder Formats

There are several standardization efforts to create a data format for inter-agency emergency messages that span local, state, federal and non-governmental organizations. EDXL[2] is a suite of OASIS standards for emergency response messaging between multiple juristictions and organizational levels. The suite includes standards for distribution (EDXL-DE[3]), message standards (EDXL-RM[4]), hospital status (EDXL-HAVE), and alerting protocols (CAP). The US Department of Justice is sponsoring an effort to provide a common interchange format between jurisdictions with the Global Justics XML Data Model[5]. These messaging formats were created so that agencies have a common standard for interoperability. These message standards do not, however, enable automated understanding and composition of the messages without a richer ontology backing the formatting.

The OpenGIS Sensor Model Language (SensorML) [3] is a specification for geometric, dynamic and observational characteristics of sensors and sensor systems. While the goal is to provide a data format for specifying sensors, sensor systems, and processes, SensorML can be encoded for the Semantic Web.

---

[2] http://xml.coverpages.org/edxl.html
[3] http://docs.oasis-open.org/emergency/edxl-de/v1.0/EDXL-DE_Spec_v1.0.pdf
[4] http://docs.oasis-open.org/emergency/edxl-rm/v1.0/EDXL-RM-SPEC-V1.0.pdf
[5] http://it.ojp.gov/default.aspx?area=nationalInitiatives&page=1013

3.3. Sensor Network Ontologies

The interchange formatting community is not in the dark about backing data formats with descriptive meaning. There are efforts that aim to incorporate a backing ontology or including expressivity into the data models.

Leveraging the descriptive power of ontologies in a sensor networks and systems dates dates back to 2003 with IrisNet [6]. This work envisioned heterogeneous sensors being deployed worldwide and accessed via the web. The key feature of this architecture is the ability to collect and query data received from the sensors from a single interface.

More recently, Sheh *et. al* [14] are working to combine the efforts of the Open Geospatial Consortium and the W3C's Semantic Web activities to create a Semantic Sensor Web. This integration effort adds RDF annotations to sensor data which is then consumed and visualized in a single application interface.

4. Methodology

This section describes the process for developing the ontologies used for OntoNet's applications. The goal is to provide an ontology that balances expressivity and complexity with a bias towards reducing expressivity (where possible) in favor of minimizing complexity. The approach to developing these ontologies roughly follows the procedures outlined by Grüninger and Fox [7] and Noy and McGuinness [13].

4.1. Competency Questions

The competency questions defined in [7] and [13], provide a starting point for developing an ontology. These questions could be considered *use cases* for the application of the ontologies. The scenarios described above in Section 2 describes the application domain of the ontologies for OntoNet. From these scenarios two generic competency questions arise pertaining to the delivery of messages and the organizational structure of the source and destinations of the messages:

- Given the characteristics of the contents of the message, where should these messages be delivered?
- Given the characteristics of the host and the contents of the message, what messages should the host subscribe to?

Domain dependent, specific questions focus on the characteristics of the messages and the senders and recipients of the messages. During an operation, units at the lower echelons (*e.g.*platoons and dismounted soldiers) provide updates of their current position and status to their peers and superior units (*e.g.*battalion command). These position reports and messages are sent as multicast messages where units interested in these types of messages would subscribe to messages of this type. A corresponding competency question for battalion command to subscribe to these messages would be:

- What position reports are from my subordinate units?

Additionally, battalion command and company commanders may request to receive messages of higher priority such as those alerting of events such as ambushes or course corrections. Questions corresponding to these requests would be:

- Which alert messages are about attacks on units part of the First Battalion?
- Which alert messages are about improvised explosive device detections?

The above competency questions cover subscribing to classes of messages from certain units. There are situations where a dismount soldier would issue a request for specific services such as a medevac for wounded soldiers or field maintenance on damaged vehicles. Competency questions for these situations would revolve around describing the event and the units issuing the request and the descriptions of the units at the destination.

- Which medevac units can evacuate wounded soldiers?
- What unit can provide artillery support for location X?
- What kind of unit can provide field maintenance to damaged vehicles?

From these competency questions, the ontologies developed for OntoNet will include information on different of source and destinations based on characteristics of the units such as the roles they play and their capabilities. Furthermore, the ontologies will include information on the characteristics of the messages being transmitted throughout the system.

4.2. Terminology

The classes of the ontologies in OntoNet describe the kinds of messages and the units (the source and destinations of these messages). Instances of these classes correspond to the specific entities in the applications.

Senders and receivers in the military domains correspond with units along the military hierarchy. The military mid-level ontologies in SUMO provide terms and descriptions for units in the military hierarchy including: *Division*, *Brigade*, *Battalion*, *Company-Military*, *Platoon*. The units will also have specific roles and functionality assigned to them such as infantry, mechanized , engineers, medical support, field maintenance, etc.

The messages are classified by type of message: *Orders*, *reports*, *Requests*, and *Alerts*. The properties of the messages include information about the message such as the time, the sender, and a description of the message. The descriptions of the messages are individual events such as ambush, artillery fire, wounded soldiers, evacuation requests, etc.

5. Ontology

The terminology used in the OntoNet ontologies leverage existing upper and mid-level ontologies: SUMO and OpenCyc [6]. Not only does this approach simplify development of the ontology, but it also aids in integration with other products using the same ontologies. The mid-level ontologies provide an abstract subsuming representation for the domain dependent ontologies and instances. The structure of the entire ontology advances the notion of a *background ontology* and a *specific ontology*.

The background ontology is the ontology that specifies the basic unit hierarchy and its roles. The OntoNet military ontology has the concepts, brigade, battalion, company,

[6] http://opencyc.org/

platoon, squad, etc. which are subsumed by the Unit concept. This ontology is based on the military ontology incorporated into SUMO.

The specific ontology refers to the individuals and instances of units used in the application. For example, the 4th Infantry Brigade is a sub-echelon of the 7th Division, and the Medical Company is a sub-echelon of the 235 Support Battalion are instances of Brigade and Battalion.

5.1. Concepts and Concept Relationships.

The unit concepts and relationships uses the ModernMilitaryOrganization ontology developed in [1] and subsequently included in SUMO. The OWL version of the SUMO defines the subEchelon as a sub property of subOrganization. The subOrganization property is classified as a TransitiveProperty in OWL. The subEchelon relationship defines a unit as a subOrganization of another.

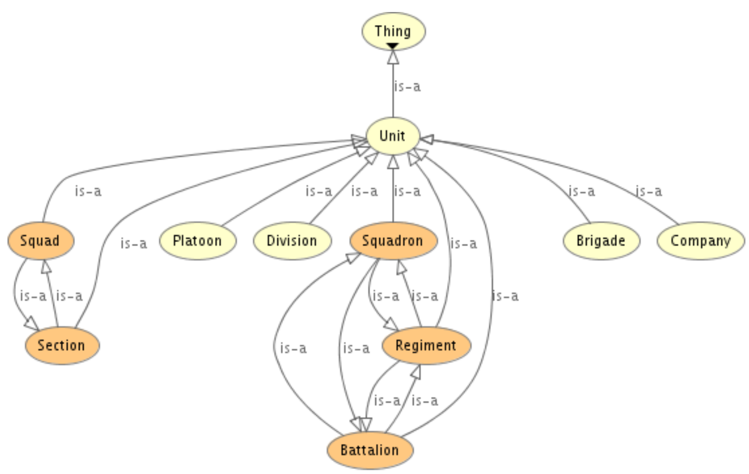

Figure 1. The unit hierarchy

Unit Ontology: This organizational classification defines the hierarchy of the oranization based on scenario specific characteristics, *e.g.*military chain of command, communications chain of command. This characterization can be used for defining source and/or destination of messages. MilitaryUnits, depicted in Figure 1, represent the ontology of units from Division all the way down to Squad. The units are related to one another via the *subEchelon* Transitive Property.

Unit Type Ontology: This is a taxonomy of the types of units in the organization. Rather than defining the structure of communications, this type ontology is used to describe the units themselves. *e.g.*A unit can be a medical support unit in a military organization or a HazMat team for emergency responders organization.

Unit Capabilities Ontology: This ontology specifies the capabilities of units. It provides a structure for reasoning over specific capabilities of units. For example, a recon unit in a military organization can provide lazing for fire support and a medevac helicopter can evacuate an injured person to a hospital.

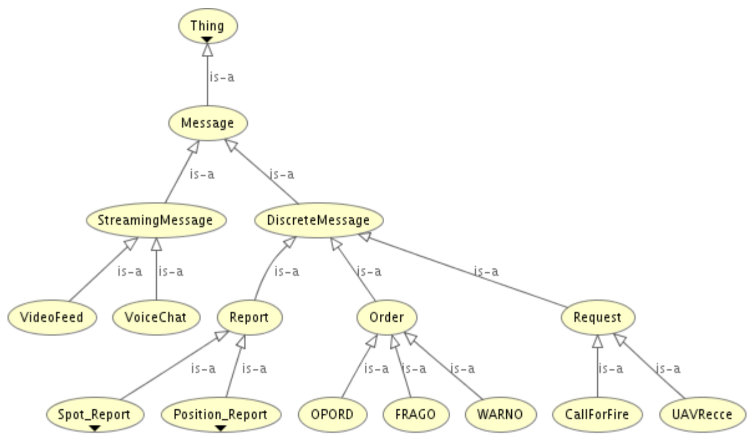

Figure 2. The message hierarchy

Message Type Ontology: This ontology describes the various types of messages sent and received in the system. For example, a request for fire or position or spot report. The classes of messages (Figure 2) are broken down into discrete and streaming messages, and further into the types of messages: Orders, Requests, and Reports.

Message Contents Ontology: This ontology describes the contents of the messages, *e.g.*position information, the CBR payloads, enemy movement events. Events are part of the message type and message contents ontologies. They describe the type of event that the message is reporting about. The messages have associated descriptions. In general, all messages will have a sender, intended receiver (if a direct message), a description of the type of message, a description of the message contents.

Messages and destinations are described in OntoNet using the Semantic Web's Resource Description Framework (RDF) [11] and Web Ontology Language (OWL) [5]. RDF is an XML syntax for first order models restricted to binary relations—labeled graphs. OWL is an RDF language for encoding ontologies, taken here as formal, machine interpretable, logical specifications of domain structure. OWL is defined by description logic [2], an object oriented subset of first order logic. Both are well documented, open standards endorsed by the World Wide Web Consortium (W3C).

5.2. Queries

With the ontology described above, we can now formally state the comptetancy questions in the form of description logic queries. As stated in previous work [10], OntoNet supports multiple matching models based on the three components: Messages, destinations, and ontologies. *Receiver querying* is the typical message model where destinations publish queries that are matched against message descriptions. An example query based on the military command and control ontologies would enable battalion headquarters to monitor the locations and status of subordinate units at the company and platoon levels. The following description matches position reports from direct subordinates, *i.e.*the companies directly below the `1-44 Battalion` and not from the platoons.

$$BNPosRptQuery \equiv PosRpt \sqcap$$

$$\exists sender.(Company \sqcap \exists subEchelon.\{\mathsf{1\text{-}44\_IN\_BN}\})$$

Battalion command units would also subscribe to Spot Report messages about various events. The most general would be subscribing to receive all Spot Reports, *i.e.*from companies as well as platoons.

$$SpotRpt\_ArmoredVehicle \equiv Spot\_Report \sqcap$$
$$\exists subEchelon.\mathsf{1\text{-}44\_IN\_BN}\})$$

On the other hand, a medevac unit interested in being alerted of wounded soldiers in the field would subscribe to messages where units are attacked.

$$SpotRpt\_Attacks \equiv Spot\_Report \sqcap$$
$$\exists activityAbout.EnemyEvent \sqcap$$

Likewise, a field maintenance unit with the role of reparing and retrieving damaged vehicles is only interested in receiving Spot Reports about attacks on armored vehicles:

$$SpotRpt\_ArmoredVehicle \equiv Spot\_Report \sqcap$$
$$\exists activityAbout.EnemyEvent \sqcap$$
$$\exists sender.(\exists hasRole.ArmoredArtilleryRole)$$

Conversely, with *message querying* the messages have include queries that match the descriptions of the destinations. There are situations where a specific message would not be covered by a receiver query. For example, a message calling for a medevac might not be a spot report but rather, just a position report with metadata to deliver the message to units that can provide medical evacuations. The metadata describing the receiver would look like this:

$$Receiver \equiv Unit \sqcap \exists hasRole.MedicalRole$$

Likewise, specific requests would need to be delivered to units with certain capabilities. An example spot report message of a company from the `1-44 Battalion` contains an alert of an ambush by the enemy. This message query states the message needs to be delivered to superior units and also units that can provide fire support.

$$FireSupportReqMsgQuery \equiv Unit \sqcap$$
$$\exists hasRole.ArmoredArtilleryRole$$

6. Application Examples

To demonstrate the use of the OntoNet ontologies consider the following application for Command and control during a military patrol. The command hierarchy consists of a battalion with 4 infantry companies, a reconnaissance troop, a squadron of tanks, a medical support company, and a field maintenance crew. This hierarchy is defined as a set

```
<Battalion rdf:about="#1-44_IN_BN">
  <rdf:type rdf:resource="&owl;Thing"/>
  <subEchelon rdf:resource="#4_IBCT_US"/>
  <hasRole rdf:resource="#InfantryRole"/>
</Battalion>
```

Figure 3. An example 1-44 Infantry Battalion description.

of OWL individuals. For example, an abbreviated description of the battalion is shown in OWL in Figure 3.

During the planning phase of the operation the commander of the battalion will need to gather readiness and capabilities of units under his command. Commander constructs standing queries to be alerted of events: Spot Reports, position reports.

The support companies also construct standing queries to receive reports such, for example, the company medic request spot reports for the wounded and the field maintenance company requests spot reports for attacks on vehicles.

Once the planning is complete and the communications infrastructure is in place, the operation commences. During the patrol, the units in the field automatically send continuous status updates in the form of position reports. These updates are broadcasted to their unit commander, *e.g.*the platoons send their positions to the company leader which are combined to form a single position report for the company to be delivered to the battalion CP.

A unit will issue a Spot Report for events such as attacks or changes in course. The spot report would contain metadata describing the sender, receiver, and content of the message. The OntoNet middleware will deliver the messages to units subscribed to those message classes. The messages are also delivered to the destinations described in the receiver. An example full Spot Report of a unit being attacked is shown in Figure 4. This message will be delivered to the battalion CP, and the company medic because those units subscribed to those messages. The message is also delivered to any unit that has the role of being armored artillery as specified by the `receiver` property.

7. Conclusion

OntoNet's knowledge-based approach to message addressing and matching significantly increases the potential decoupling between message senders and receivers enabled by such middleware. This is particularly important in the target military and first responder settings, which may include many diparate components and ad hoc interactions over time. Knowledge-based reasoning does incur additional computational costs over simpler addressing schemes, but this downside is more than matched by improvements in rapid reconfiguration, system extension, and interoperability that this approach supports. Future work in OntoNet includes determining the exact extent of these costs, particularly on hardware platforms typical for these applications, and possibilities in leveraging the structure of both the core ontologies and reasoning requirements to optimize reasoning.

References

[1] W. Anderson and B. Peterson. An ontology of modern military organizations and their structure. In *Workshop on the IEEE Standard Upper Ontology (IJCAI01)*, 2001.

```
<Spot_Report rdf:about="#Msg_SpotRpt_ArmoredInd">
  <rdf:type rdf:resource="&owl;Thing"/>
  <sender rdf:resource="#SP_BTRY_4-4_FA_BN_4_IBCT_US_"/>
  <activityAbout rdf:resource="#EnemyAmbushEvent_i"/>
  <receiver rdf:resource="#RcvrDesc"/>
  <hasLocation rdf:resource="#position3"/>
</Spot_Report>

<owl:Thing rdf:about="#RcvrDesc">
  <rdf:type>
    <owl:Class>
      <owl:intersectionOf rdf:parseType="Collection">
        <rdf:Description rdf:about="#Unit"/>
        <owl:Restriction>
          <owl:onProperty rdf:resource="#hasRole"/>
          <owl:someValuesFrom rdf:resource="#ArmoredArtilleryRole"/>
        </owl:Restriction>
      </owl:intersectionOf>
    </owl:Class>
  </rdf:type>
</owl:Thing>
```

Figure 4. An example Spot Report with a description of the intended destinations.

[2] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider. *The Description Logic Handbook*. Cambridge Univ. Press, 2003.

[3] M. Botts. OpenGIS sensor model language (SensorML) implementation specification. Technical Report 07-000, Open Geospatial Consortium, Jul 2007.

[4] M. Bowman, A. Lopez Jr, and G. Tecuci. Ontology Development for Military Applications. In *Proceedings of the SouthEastern Regional ACM Conference, Atlanta, GA*. Citeseer, 2001.

[5] M. Dean, G. Schreiber, et al. OWL Web Ontology Language reference. World Wide Web Consortium (W3C), February 2004. `http://www.w3.org/TR/2004/REC-owl-ref-20040210/`.

[6] P. B. Gibbons, B. Karp, Y. Ke, S. Nath, and S. Seshan. Irisnet: An architecture for a worldwide sensor web. *IEEE Pervasive Computing*, 2(4), October–December 2003.

[7] M. Grüninger and M. Fox. Methodology for the Design and Evaluation of Ontologies. In *The Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.

[8] Headquarters. Army planning and orders production. Technical Report FM 5-0, Department of US Army, January 2005.

[9] J. B. Kopena and B. T. Loo. Ontonet: Scalable knowledge based networking. In *In Proceedings of the 4th International Workshop on Networking Meets Databases*, 2008.

[10] J. B. Kopena, W. C. Regli, and B. T. Loo. Message models and aggregation in knowledge based middleware for rich sensor systems. In *In Proceedings of the 6th International Workshop on Data Management for Sensor Networks*, 2009.

[11] O. Lassila, R. Swick, et al. Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium (W3C), 1999. `http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/`.

[12] I. Niles and A. Pease. Towards a standard upper ontology. *International Conference on Formal Ontology in Information Systems*, October 2001.

[13] N. F. Noy and D. L. McGuinness. Ontology development 101: A guide to creating your first ontology. Technical Report KSL-01-05, Stanford University Knowledge Systems Laboratory, March 2001.

[14] A. Sheth, C. Henson, and S. S. Sahoo. Semantic sensor web. *IEEE Internet Computing*, 12(4):78–83, July–August 2008.