An ontology-based model for vehicular ad-hoc networks

Conference Paper · July 2014

DOI: 10.1109/INES.2014.6909346

3 authors:

Adrian Groza
Universitatea Tehnica Cluj-Napoca

Anca Marginean
Universitatea Tehnica Cluj-Napoca

Vlad Muresan
Universitatea Tehnica Cluj-Napoca

An Ontology-Based Model for Vehicular Ad-hoc Networks

Adrian Groza
Intelligent Systems Group
Department of Computer Science
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
Email: Adrian.Groza@cs.utcluj.ro

Anca Marginean
Intelligent Systems Group
Department of Computer Science
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
Email: Anca.Marginean@cs.utcluj.ro

Vlad Muresan
Department of Automation
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
Email: Vlad.Muresan@cs.utcluj.ro

***Abstract*—The main body of research in vehicular ad-hoc networks (VANETs) has focused on solving network communication aspects and regulatory norms and standards, with much work remaining at the application level. Our hypothesis is that, semantic middleware platforms to exchange vehicular knowledge are essential in order to reach the next level of VANET applications. We propose a vehicular network ontology with the goal to facilitate the interoperability at the application level. As vehicular networks is a multi-facet domain, the developed ontology is modular and represents various aspects like communication, types of applications, traffic hazards, events, localisation. We also show how geospatial and temporal reasoning can be performed on top of our ontology.**

***Index Terms*—vehicular networks, ontology engineering, geospatial reasoning.**

I. Introduction

Vehicular Ad-hoc Networks (VANETs) is one of the most promising applications of mobile ad-hoc networks. VANETs use vehicles as mobile nodes able to self-configure in order to create a communication network via wireless links. The communication takes place both between vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I), aiming to enable *automated* cooperation between different vehicles on the road.

Vehicle-to-x technology has just solved low level aspects with respect to ad-hoc networks or regulatory norms and standards. Thus, much work remains at the application layer, to facilitate the integration of the newly developed services [6]. In this line, innovative communication and cooperative techniques are needed to model different interaction patterns, interoperability issues, or high level understanding of micro-traffic events.

Our hypothesis is that, VANET-related applications could benefit from the current advances in knowledge representation and ontology engineering. The benefits of providing a vehicular ontology include: i) facilitates interoperability between components from different manufacturers; ii) provides a human oriented view of microscopic traffic events; iii) facilitates information understood by all the entities involved in the transportation system; iv) provides explanation to the driver about specific traffic event or situation. We argue that a semantic middleware platform to exchange vehicular knowledge is essential in order to reach the next level of intelligent in-car systems [4].

In this paper we propose a vehicular network ontology with the goal to facilitate the interoperability at the application level. The ontology was developed in the KRSS native syntax of the RacerPro tool [7]. Geospatial and temporal reasoning capabilities were exploited to model various aspects of vehicular networks.

The remaining of the paper is organised as follows: Section II browses related ontologies developed for the vehicular domain. Section III briefly introduces description logic as the main technical instrumentation used in our approach. Section IV presents our ontology-based model for vehicular networks. Section V shows the usage of the ontology in a lane-change cooperative scenario. Finally, section VI highlights the contribution of the paper.

II. Related work

Several related ontologies in the automotive domain have been developed [3], [12], [2], [11], [4]. They have focused on modeling specific facets of a vehicular network, but no one had the goal to cover the entire vanets domain. However, part of the following related ontologies were considered for re-used during the engineering of our vehicular network ontology.

An ontology of vehicular networks security has been modeled in [3]. The aim has been to classify the vulnerabilities based on the impact of the intrusion and functionality affected in routing protocols. The top level concepts are *Attack*, *Consequences*, *Vulnerabilities*, and *Actor*. The attacks are either active (like relaying attacks, injection of malicious code, modifying of packets) or passive (sybil, illegal eavesdropping). The identified vulnerabilities are: high mobility, the cooperative relationships, shared wireless medium. Among the consequences modeled by the ontology, there are: degradation of vehicular network performance, road congestion, insulation of nodes or even road accidents.

The ontology in [12] aims to determine the autonomy layer of an automated vehicle. The main use case is at self-assessment of the perception system to monitor co-driving. The module designed for situation assessment formalises knowledge such as: environment conditions, moving obstacles, driver state, navigable space, which are also relevant concepts for the vehicular network domain.

The CAOVA (Car Accident lightweight Ontology for VANETs) structures information from two sources: i) collected from vehicle sensors when an accident occurs, or ii) imported from the the General Estimates System accidents database [2]. Due to the private character of data formalised for the driver concept (blood, allergies, illness, medication, pregnant, weight, age, sex), the ontology has been encrypted using the Advanced Encryption Standard. The top level concepts are: *Vehicle*, *Accident*, *Environment*, and *Occupant*. The use cases of the ontology regard various car safety application, with the goal to increase the interoperability among emergency services, authorities, or other vehicles.

Having the main objective to facilitate vehicle selling, several automotive ontologies have been designed to be used in combination with the GoodRelations [8] commercial oriented vocabulary. In this line, from the Volkswagen Vehicles Ontology[1] or Vehicle Sales Ontology[2] some concepts are also relevant in the context of vehicular communication: model, dimensions of the vehicle, engine, type of the vehicle (such as van, truck, etc.).

A vehicular ontology is proposed in [11] having the role of a dynamic middleware between two vanets. The ontology focuses on defining packets and their features (MFRBroadcastPacket, PositionBasedPacket, ClusterBasedPacket, etc). The axioms of these packets are used to infer the meaning of a packet and to classify it under the most appropriate routing strategy. The work in [11] can be integrated in the larger context of semantic middleware solutions, as it employs ontologies to achieve communication protocol interoperability.

III. Description Logic

In the description logic $\mathcal{ALC}$, concepts are built using the set of constructors formed by negation, conjunction, disjunction, value restriction, and existential restriction [1], as shown in table I. Here, C and D represent concept descriptions, while r is a role name. The semantics is defined based on an interpretation $I = (\Delta^I, \cdot^I)$, where the domain Δ^I of I contains a non-empty set of individuals, and the interpretation function $\cdot^I$ maps each concept name C to a set of individuals $C^I \in \Delta^I$ and each role r to a binary relation $r^I \in \Delta^I \times \Delta^I$. The last column of table I shows the extension of $\cdot^I$ for non-atomic concepts.

An ontology consists of terminologies (or TBoxes) and assertions (or ABoxes).

Definition 1: A terminology $TBox$ is a finite set of terminological axioms of the form *(equiv C D)* or *(implies C D)*.

Example 1 (Terminological box): In the tbox `Vanet` in Fig. 1, the vehicles are partitioned into private (belonging to individuals or private companies in line 2) and public (i.e. buses, police in lines 5-6). The axiom 7 specifies that a `PublicVehicle` should belong only to public agencies. In line 8, road side units can belong to the government or private service operators.

```
1.(in-tbox Vanet)
2.(define-primitive-role belongsTo
     :domain Vehicle
     :range (or Individual Company PublicAgency))
3.(implies PrivateVehicle Vehicle}
4.(implies PublicVehicle Vehicle)
5.(implies Bus PublicVehicle)
6.(implies Police PublicVehicle)
7.(implies PublicVehicle
               (all belongsTo PublicAgency))
8.(implies LocalTransportAgengy PublicAgency)
9.(implies RoadSideUnit (some belongsTo
     (or PublicAgency PrivateServiceOp)))
```

Fig. 1. Modeling VANETs-related knowledge in description logics.

```
10. (in-abox vanet-brno Vanet)
11. (instance b1 Bus)
12. (instance lta-brno LocalTransportAgengy)
13  (instance rsu1 RoadSideUnit)
14. (related b1 lta-brno belongsTo)
15. (related rsu1 lta-brno belongsTo)
```

Fig. 2. Modeling assertions in VANETs.

Definition 2: An assertional box $ABox$ is a finite set of concept assertions *(instance a C)* or role assertions *(related a b r)*, where C designates a concept, r a role, and a and b are two individuals. Usually, the unique name assumption holds within the same $ABox$.

Example 2 (Assertional box): The assertional box `vanet-brno` makes use of the terminologies in the `Vanet` tbox (line 10 in Fig. 2). The bus `b1` (line 11) belongs to the local transportation agency `lta-brno` (line 14). Similarly, the road side unit `rsu1` (line 13) operates under the same public agency `lta-brno` (line 15).

A concept C is satisfied if there exists an interpretation I such that $C^I \neq \emptyset$. The concept D subsumes the concept C, represented by `(implies C D)` if $C^I \subseteq D^I$ for all interpretations I. Constraints on concepts (i.e. `disjoint`) or on roles (`domain`, `range` of a role, `inverse` roles, or `transitive` properties) can be specified in more expressive description logics[3].

IV. Engineering the vehicular network ontology

To develop the vanet ontology, we follow the methodology in [10] and we also enact various ontology design patterns [13]. The engineering steps presented in this section are: i) defining competency questions, ii) re-using other ontologies, iii) defining main concepts and roles, iv) populating the ontology, and v) ontology debugging and evaluation.

Competency questions. The competency questions (CQs) help to define the limits of the domain to be modeled and also to identify the main concepts and roles of the ontology. Examples of CQs for VANET domain are listed in table II.

[1] http://www.volkswagen.co.uk/vocabularies/vvo/ns

[2] http://www.heppnetz.de/ontologies/vso/ns

[3] We provide only some basic terminologies of description logics in this paper to make it self-contained. For a detailed explanation about families of description logics, the reader is referred to [1].

TABLE I
KRSS SYNTAX AND SEMANTICS OF $\mathcal{ALC}$.

Constructor	*Syntax*	*Semantics*
negation	`(not C)`	$\Delta^I \setminus C^I$
conjunction	`(and C D)`	$C^I \cap D^I$
disjunction	`(or C D)`	$C^I \cup D^I$
existential restriction	`(some r C)`	$\{x \in \Delta^I \mid \exists y : (x, y) \in r^I \wedge y \in C^I\}$
value restriction	`(all r C)`	$\{x \in \Delta^I \mid \forall y : (x, y) \in r^I \rightarrow y \in C^I\}$
individual assertion	`(instance a C)`	$\{a\} \in C^I$
role assertion	`(related a b r)`	$(a^I, b^I) \in r^I$

These CQs help to identify the concepts (i.e. `Lane`, `Speed`, `Overtake`, `MultiHop`, `EmergencyVehcile`) and roles (i.e `isLocated`, `hasSpeed`, `nearby`).

TABLE II
SAMPLE OF COMPETENCY QUESTIONS FOR THE VEHICULAR NETWORK ONTOLOGY.

No	*Competency question*
CQ_1	Which are the vehicles on the same lane within a specific area?
CQ_2	Which data is available about the closest vehicle in front/behind?
CQ_3	Which is the closest vehicle approaching from opposite direction?
CQ_4	Which is the average speed for the next 5km?
CQ_5	Is it safe to change lane?
CQ_6	Is it safe to overtake the vehicle in front?
CQ_7	Which vehicles in the VANET can perform multi-hop routing?
CQ_8	Are there any emergency vehicles in the nearby?

Reusing other ontologies. Two categories of ontologies can be reused: vehicular related domain ontologies and general ontologies. The domain ontologies considered to be reused were presented in section II. The general knowledge required to model various facets of vanets include: spatial, temporal, situation awareness.

Aiming to facilitate the identification of reusable knowledge, we designed the ontology to contain several modules. An ontology module represents a shared, domain-independent conceptualization intended to be used for different tasks and applications. From this perspective, ontology modules are comparable to software libraries in the software engineering domain. In this way, different traffic scenarios can enact only knowledge relevant for the application domain.

Defining main concepts and roles. The main elements of the ontology are organised on modules like: communication, vehicular, traffic hazards, etc, as follows:

The *communication module* defines basic communication patterns and messages that take place in vanets-enabled applications. We start by classifying the main application types (Fig. 3) by enacting the taxonomy ontology design pattern [13]. The applications of vehicular communication [14] can be split into three subcategories: safety, resource efficiency and infotainment (axiom 22). Note for instance that, the overtaking maneuver implies both lane changing and collision avoidance safety issues (line 27).

The communication regimes are classified according to the transmission scheme into bidirectional and position based (line 34 in Fig. 5). The bidirectional regime (or unicast) enables connection between two vehicles by performing four

```
21. (in-tbox Communication)
22. (implies (or SafetyApplication Infotainment
       ResourceEfficiency) Application)
23. (implies (or Warning PassiveSafety ActiveSafety
       ProActiveSafety) SafetyApplication)
24. (implies (or QuickWarningAlerts
       NormalWarningAlerts) Warning)
25. (implies CollisionAvoidance ProActiveSafety)
26. (implies LaneChanging ProActiveSafety)
27. (implies Overtaking (and LaneChanging
       CollisionAvoidance))
28. (implies NormalTrafficAlerts ResourceEfficiency)
29. (implies AutonomousSystems ResourceEfficiency)
30. (implies GreenLightWave NormalTrafficAlerts)
31. (implies EnhancedRouteGuidance
                                  NormalTrafficAlerts)
31. (implies CooperativPlatooning AutonomousSystems)
33. (implies AdHocServices Infotainment)
```

Fig. 3. Top level taxonomy of vehicular applications.

phases: discovery, connection, data, and ending (lines 35-37). The position based regime (or geocast) in lines 38-39 simultaneously conveys information one way to a group of vehicles in a specified geographical area.

After discovering the vehicles in the area of interest, the information tagged with the geographical area is sent in the flooding phase. The acknowledgment is skipped in the position based regime (line 40), where `bottom` is the empty concept, while the control channel is eliminated in the fast bidirectional sub-regime (axiom 42). With axioms 43-44, both the single hop and multi hop protocols are classified as position based. If the area is large, the multi-hop routing mechanism is activated when the information needs to travel form one vehicle to another to reach all the targeted vehicles.

Different types are messages (alerts, beacons, normal messages) are conveyed in vanets applications (line 51 in Fig. 6). Permanent beacons and alert messages are sent using the position based communication regime (axiom 52). In line 53, the messages have a priority between 0 (highest) and 4 (lowest). The messages are also characterised by a specific update rate which leads to a reception probability known as *packet delivery ratio* (`PDR`).

For safety applications a minimum value of 0.95 is required for the `PDR` parameter (axiom 54). To guarantee the `PDR` value, multiple messages should be sent within the so-called time-to-live (`TTL`) of the message. However, in some safety applications, the message should be sent in real time (`RT` in line 55). Also relevant, the latency parameter represents the

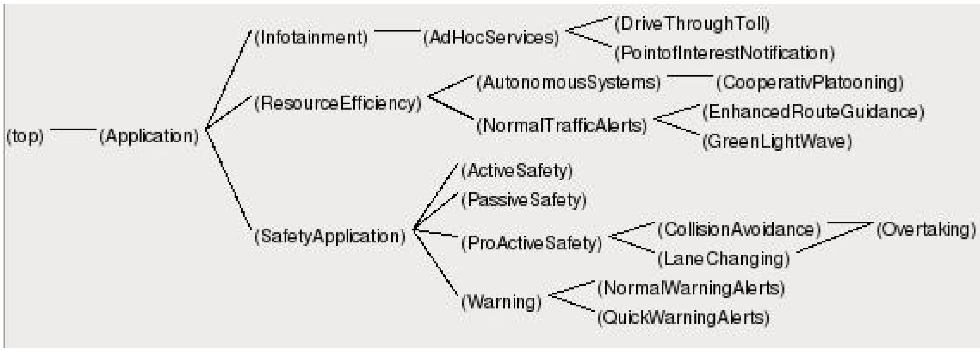

Fig. 4. View on the taxonomy of vehicular applications.

```
34. (implies CommunicationRegimes
      (or Bidirectional PositionBased))
35. (implies Bidirectional
      (and (=1 hasTarget (or Vehicle RoadSideUnit))
36.     (some hasPhase Discovery)
        (some hasPhase Connection)
37.     (some hasPhase Data)
        (some hasPhase  Ending)))
38. (implies PositionBased
      (and OneWay (some hasTarget VehicleGroup)
39.     (some hasPhase Discovery)
        (some hasPhase Flooding)
40.     (some hasAcknowledgement bottom)))
41. (equiv VehicleGroup (and
       (> 2 hasVehicle Vehicle)
       (all hasArea GeoRegion)))
42. (implies FastBidirectional (and Bidirectional
       (some hasControlChannel bottom)))
43. (implies SingleHop PositionBased)
44. (implies MultiHop PositionBased)
```

Fig. 5. Communication regimes.

```
51. (implies (or Alert Beacons Normal) MessageType)
52. (implies Beacons (some hasCommunicationRegime
      PermanentBased))
53. (equiv Priority (one-of 0 1 2 3 4))
54. (implies SafetyApplication (> PDR 0.95))
55. (implies (or TTL RT) TimeCritical)
56. (implies LaneChanging (< Latency 100))
57. (implies (or V2V V2I) TransmissionType)
58. (disjoint V2V V2I)
59. (implies (or T2V D2V V2B) V2V)
60. (implies V2RSU V2I)
```

Fig. 6. Message features in vanets communication.

time delay from sending and receiving a packet. For instance, lane changing application requires a latency below 100ms (line 56).

Two disjoint transmission types exist in vehicular communication: vehicle-to-vehicle (`V2V`) or vehicle-to-infrastructure (`V2I`) (axioms 57-58). Different sub-types of `V2V` transmission can be envisaged: train to vehicle (`T2V`), drone to vehicle (`D2V`), vehicle-to bike (`V2B` in line 59). Road side units (`RSU`) represent the main infrastructure available for `V2I` (definition 60).

Normal warning alerts signal an event by sending multi-hop messages in a time window to other vehicles (Fig. 7).

```
61. (equiv NormalWarningAlerts (and Alert
62.   (some hasCommunicationRegime
            MultiHopPositionBased)
63.   (some hasApplicationType Warning)
64.   (some hasTransmissionType (or V2V V2RSU))))
65. (implies RailCollisionWarning
                NormalWarningAllerts)
66. (implies SlowVehicleWarning
                NormalWarningAllerts)
67. (implies LimitedAccessWarning
                NormalWarningAllerts)
68. (implies WorkingAreaWarning
                NormalWarningAllerts)
69. (implies PostCrashWarning NormalWarningAllerts)
70. (implies HazardousLocationNotification
      NormalWarningAllerts)
71. (implies TrafficJamAheadWarning
      NormalWarningAllerts)
72. (implies (or Pit SlipperyRoadWay WaterOnLane
      OilOnLane) Hazard)
```

Fig. 7. Classifying warning alerts in vanets.

```
73. (in-abox hazard-location-notification)
74. (instance hln (and HazardousLocationNotification
      (= atPos p1))
75. (instance p1 (and Position (= hasLat 49.19205)
      (= hasLong 16.6131)))
76. (instance law (and LimitedAccesWarning
      (< height 2.4) (= atPos p1))
```

Fig. 8. Assertions in the vanet ontology.

For instance, the rail collision warning is sent by the rail to the vehicles nearby, when the train is approaching a level-crossing area. Hazardous notification sends warning about possible hazards detected by vehicle sensors, road side units or driver. The *hazard module* in our ontology was designed to support these warning messages, as axiom 72 bears out. For instance, the ESP sensor of the vehicle may detect a slippery location or a pit on the roadway and warn other drivers behind about these possible dangerous situations. In our approach, the identified slippery locations represent specific instances that populate the ontology.

Populating the ontology. Fig. 8 illustrates an instantiation of the warning message `HazardousLocationNotification`. The individual `hln` is an instance of

```
81. (in-tbox lane-changing)
82. (equiv LaneChanging (and ProActiveSafety
      (some hasMessageType Beacons)
83.   (some hasTransmissionType V2V)
84.   (some hasCommunicationRegime SingleHop)
85.   (some hasTimeContraints RealTime)
86.   (= hasPriority 0)
      (> hasPDR 95)
      (< hasLatency 100)))
```

Fig. 9. Requirements for lane changing-related applications.

`HazardousLocationNotification` concept and it also specifies the position `p1` where the hazard was identified (line 74). The individual `p1` is of type `Position` constrained by two feature roles `hasLat` and `hasLong`. The `LimitedAccesWarning law` signals other vehicles a limitation of 2.4 meter height at the same position `p1`.

Ontology debugging and evaluation. From the semantic point of view, the ontology was checked against consistency and coherence. After iterative repair steps, the ontology is consistent and the cycles in the concepts removed. The domain coverage was checked by validating the ontology against the CQs defined initially (recall table II), thus assuring that the ontology is able to provide answers to the CQs specified.

Graph based evaluation metrics were analysed in order to avoid structured defects such as unbalanced subsumption branches, lazy or uninstantiated concepts, concepts with only one sub-concept, concepts with more than 25 sub-concepts and other elements considered worst practice in ontology engineering [15]. From the engineering perspective, the ontology uses several ontology design patterns: n-ary relationship, partition, universal-existential macro, modular design pattern, taxonomy, agent role, etc. [13].

V. Running scenario

Vanet-related application heavily rely on temporal and spatial reasoning. We exploit the RacerPro [7] capabilities for temporal reasoning and geospatial reasoning to faccilitate semantic-based real time intelligent decisions.

Lane change assistance scenario. The requirements for lane changing application [14] are formalised in our ontology by enacting the *n-ary relations* ontology design pattern [13]. The aim of the pattern is to model n-ary relationship in an ontology, given that description logic has been designed to express binary relations only.

In Fig. 9 the `LaneChanging` is a `ProActiveSafety` application. The lane changing message is not targeted to a specific vehicle, but the message is beaconed to all vehicles nearby (lines 81-82). The `V2V` transmission type is needed to avoid sending messages to road side units. The `SingleHop` communication regime suffices (line 84), due to the current specifications of the IEEE 802.11p standard [9]. The `RealTime` constraint is introduced in line 85 to signal that the message is valid only for the current instance of time. In line 86, the highest priority `0` is used, a `PDR` value above 95%, with a latency of maximum `100ms` [14].

```
87. (instance c1 Skoda)
88. (instance c2 Dacia)
89. (define-event-assertion
            ((hasLocation c1 l1) 0 .1))
90. (define-event-assertion
            ((hasLocation c1 l2) .1 .2))
91. (define-event-assertion
            ((hasLocation c1 l3) .2 .3))
92. (define-event-assertion
            ((hasLocation c2 l1) .3 .4))
93. (define-event-assertion
            ((hasLocation c2 l2) .4 .5))
94. (instance l1 (and (= hasLat 49.19205)
                       (= hasLong 16.6131)))
95. (instance l2 (and (= hasLat 49.19206)
                       (= hasLong 16.6131)))
96. (instance l3 (and (= hasLat 49.19207)
                       (= hasLong 16.6132)))
97. (equiv Lane1 (and (< hasLat 49.19206)
                       (> hasLat 49.19204)
                       (= hasLong 16.6131)))
98. (equiv Lane2 (and (< hasLat 49.19206)
                       (> hasLat 49.19204)
                       (= hasLong 16.6132)))
```

Fig. 10. Geospatial and temporal reasoning.

Detecting inconsistencies. The reasoning services of description logic can be used to validate different messages at the application level. For instance, a specific message sent through the `MultiHop` communication regime will not be classified as a `LangeChanging` message according to the definition in Fig. 9. Moreover, given the abox:

```
ABox={  (instance ln LaneChanging),
        (related ln single hasCommRegime)}
```

the RacerPro signals an inconsistency relative to the tbox `lane-changing`. The query `(instantiators ln)` will print all the concepts to which the individual `ln` is an instance of.

Assertions about vehicles are valid only within a certain time interval. In Fig. 10, `c1` and `c2` are vehicles (axioms 87-88). Between time steps `[0.0, 0.1)ms` the individual `c1` is known to have location `l1`, between `[0.1,0,2)` location `l2`, and between `[0.2, 0.3)` location `l3` (lines 89-91). The locations are characterized by longitude and latitude coordinates, as transmitted through vehicular communication. From GIS maps, the definitions of `Lane1` and `Lane2` can be obtained (axioms 97-98). Based on these assertions, the system is able to deduce that locations `l1` and `l2` belong to the concept Lane1, while location `l3` to the concept `Lane2`, as the following RacerPro queries bear out:

```
? (concept-instances Lane1) -> (l1, l2)
? (concept-instances Lane2) -> (l3)
```

The event rule in Fig. 11 is used to recognise that an individual `?v` changes the lane, and also the instance of time when this event takes place. The rule signals that vehicle `?v` changes `Lane1` with `Line2` sometime between `?t1` and `?t2`. The variable `?v` is matched against objects of type `Vehicle` (line 102), `?l1` against locations that satisfy the

```
101. (define-event-rule ((laneChange ?v ?l1 ?l2)
                          ?t1 ?t2)
102.    ((?v vehicle) ?t0 ?tn)
103.    ((?l1 Lane1) ?t0 ?tn)
104.    ((?l2 Lane2) ?t0 ?tn))
105.    ((?v ?l1 hasLocation)) ?t0 ?t1)
106.    ((?v ?l2 hasLocation)) ?t2 ?t3)
```

Fig. 11. Lane chaning event recognition.

definition of `Lane1` (line 103), respectevely `?l2` against locations within the constraints of the concept `Lane2` (line 104). Consider that the vehicle `?v` was related to the location `?l1` via the role `has-location` within the time interval `[?t0, ?t1]` (line 105). The event is detected if the same vehicle `?v` appears in a different location `?l2` in a time interval starting with `?t2` (line 106).

The rule is fired in RacerPro engine and detects that `c1` has performed a lane changing maneuver, given the assertions in Fig. 10.

VI. Conclusions

We developed an ontology for modeling the vehicular networks domain. The ontology was built based on current engineering methodologies and several ontology design patterns have be enacted. As vehicular networks is a multi-facet domain, the proposed ontology is modular, addressing aspects like communication, types of applications, traffic hazards, events, localisation, etc. The standard reasoning services of description logic (subsumption reasoning, satisfiability, consistency, instance retrieval) are complemented with geospatial and temporal reasoning, with the help of the RacerPro reasoning engine.

Such an ontology is a step towards the integration of multi-agent technology in the vehicular networks domain, subject which we are currently investigating [5].

Acknowledgments

This research is supported by the Technical University of Cluj-Napoca, Romania, through the internal research project "GREEN-VANETS: Improving transportation using Car-2-X communication and multi agent systems".

References

[1] Franz Baader. *The description logic handbook: theory, implementation, and applications*. Cambridge university press, 2003.

[2] Javier Barrachina, Piedad Garrido, Manuel Fogue, Francisco J Martinez, J-C Cano, Carlos T Calafate, and Pietro Manzoni. Caova: A car accident ontology for vanets. In *Wireless Communications and Networking Conference (WCNC)*, pages 1864–1869. IEEE, 2012.

[3] M Erritali, B El Ouahidi, B Hssina, B Boukhalene, and A Merbouha. An ontology-based intrusion detection for vehicular ad hoc networks. *Journal of Theoretical & Applied Information Technology*, 53(3):410–414, 2013.

[4] Michael Feld and Christian Müller. The automotive ontology: managing knowledge inside the vehicle and sharing it between cars. In *3rd International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 79–86. ACM, 2011.

[5] Adrian Groza, Bogdan Iancu, and Anca Marginean. A multi-agent approach towards cooperative overtaking in vehicular networks. In Rajendra Akerkar, Nick Bassiliades, John Davies, and Vadim Ermolayev, editors, *WIMS*, page 48. ACM, 2014.

[6] Adrian Groza, Anca Marginean, and Bogdan Iancu. Towards improving situation awareness during emergency transportation through ambulance-2-x communication and semantic stream reasoning. In *4th International Conference on Advancements of Medicine and Health Care through Technology (MediTech2014), 5-7 June*, 2014.

[7] Volker Haarslev, Kay Hidde, Ralf Möller, and Michael Wessel. The RacerPro knowledge representation and reasoning system. *Semantic Web*, 3(3):267–277, 2012.

[8] Martin Hepp. Goodrelations: An ontology for describing products and services offers on the web. In *Know. Eng.: Practice and Patterns*, pages 329–346. Springer, 2008.

[9] Daniel Jiang and Luca Delgrossi. IEEE 802.11 p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology*, pages 2036–2040. IEEE, 2008.

[10] Natalya F Noy, Deborah L McGuinness, et al. Ontology development 101: A guide to creating your first ontology, 2001.

[11] Vatsala Nundloll, Paul Grace, and Gordon S Blair. The role of ontologies in enabling dynamic interoperability. In *Distributed Applications and Interoperable Systems*, pages 179–193. Springer, 2011.

[12] Evangeline Pollard, Philippe Morignot, and Fawzi Nashashibi. An ontology-based model to determine the automation level of an automated vehicle for co-driving. In *FUSION*, pages 596–603. IEEE, 2013.

[13] Jeffrey T Pollock and Ralph Hodgson. Ontology design patterns. *Adaptive Information: Improving Business through Semantic Interoperability, Grid Computing, and Enterprise Integration*, pages 145–194.

[14] Radu Popescu-Zeletin, Ilja Radusch, and Mihai Adrian Rigani. *Vehicular-2-X Communication: State-of-the-Art and Research in Vehicular Ad hoc Networks*. Springer, 2010.

[15] Steffen Staab and Rudi Studer. *Handbook on ontologies*. Springer, 2009.