Semantically enabling IoT trust to ensure and secure deployment of IoT entities

Konstantinos Kotis*

Department of Digital Systems,
University of Piraeus,
Karaoli & Dimitriou 80, Piraeus, Greece
Email: kotis@aegean.gr
*Corresponding author

Iraklis Athanasakis

School of Science and Technology,
Hellenic Open University,
Parodos Aristotelous 18, 26-335, Patra, Greece
Email: std106292@ac.eap.gr

George A. Vouros

Department of Digital Systems,
University of Piraeus,
Karaoli & Dimitriou 80, Piraeus, Greece
Email: georgev@unipi.gr

Abstract: Semantics for the IoT domain have been already introduced for the (semi-)automated deployment of heterogeneous entities. Depending on the level of interoperability and the ability of dynamic expansion of the IoT environment, an application may have to 'decide' (and then select) which devices in that environment are trustworthy for ensuring and securing effective deployment. In the open and distributed IoT, where a large number of heterogeneous entities will be registered, the need to ensure and secure their selection and deployment tasks is highly important. In this paper, an effective modelling approach towards supporting the selection and deployment of IoT entities is presented, based on the notion of trust semantics. Using fuzzy ontologies as an enabler of trust semantics in IoT, this work demonstrates that such semantics, when seamlessly integrated in IoT ontologies, serve as a secure selection key to an IoT application (or service) for selecting, among the available entities, the one(s) that the application should trust for its effective deployment in the specific environment/context.

Keywords: IoT trust; semantic interoperability; trust semantics; fuzzy semantics; IoT entities deployment; trust ontology; fuzzy ontology; trust interoperability; IoT entities matchmaking; IoT entities selection.

Reference to this paper should be made as follows: Kotis, K., Athanasakis, I. and Vouros, G.A. (2018) 'Semantically enabling IoT trust to ensure and secure deployment of IoT entities', *Int. J. Internet of Things and Cyber-Assurance*, Vol. 1, No. 1, pp.3–21.

Biographical notes: Konstantinos Kotis received his BSc in Computation from the University of Manchester (UMIST), UK and PhD in Knowledge Representation from the University of the Aegean, Greece. He worked as an Adjunct Lecturer at the University of the Aegean (2005–2010) and as an ERCIM fellow at VTT in Finland (2011–2012). He is currently a Post-doc Research Associate at the University of Piraeus and an IT Manager at NARA, Greece. His research interests include knowledge engineering, semantic web and IoT. He has published more than 50 peer-reviewed papers, and served as a PC member in international journals and conferences.

Iraklis Athanasakis received his Diploma in Electrical and Computer Engineering from the University of Patras. He has more than ten years of experience as an IT/technology transfer consultant for SMEs in the North Aegean Region. He is currently employed in the North Aegean Regional Administration as an IT Manager while pursuing his Masters degree in Engineering of Pervasive Computing Systems MSc at the Hellenic Open University.

George A. Vouros received his BSc in Mathematics and PhD in Artificial Intelligence all from the University of Athens, Greece. Currently, he is a Professor in the Department of Digital Systems in the University of Piraeus. He has done research in the areas of expert systems, knowledge management, collaborative systems, ontologies, and agents and multi-agent systems. He has extensively served as program chair, chair and member of organising committees of national and international conferences on related topics, member of steering committees, and Chair of the Hellenic A.I. Society board. Further details can be found in http://ai-group.ds.unipi.gr/georgev/.

1 Introduction

Entities (applications, devices, sensors, humans, gateways, etc.) that ‘live’ in open, distributed and heterogeneous IoT environments, need to be consistently, explicitly and formally represented and managed (registered, aligned, composed, and discovered) through suitable abstraction technologies i.e., ontologies (Hachem et al., 2011; Wang et al., 2013). Such a representation and management capability enables their seamless integration in different application domains, such as smart home, ambient assisted living, transportation, etc., in a way that deployment of generic applications and third-party devices in non-expert end-users’ IoT settings is performed (semi-)automatically, with minimum human involvement.

Depending on the level of interoperability in the IoT environment and the ability of its dynamic expansion, an entity may have to ‘decide’ which other entities in that environment are trustworthy, and then map its individual security policies with those trustworthy entities in order to avoid critical ‘misunderstandings’. This decision requires the ability of a generic application or third-party device to distinguish and consider an entity as a *trustworthy* one. In the open and distributed IoT, where a large number of generic applications and third-party devices will be registered in different available registries, the need to ensure and secure deployment of heterogeneous entities is highly important. To achieve this, there is a need to extend existing IoT-related semantic interoperability approaches (Kiljander et al., 2014; Galov et al., 2015; Amarnath et al., 2016; Gyrard et al., 2015) with *trust semantics*. When seamlessly integrated in IoT ontologies, trust can serve as a *secure selection key* of a generic IoT application/service to choose, among the available third-party registered devices, the one(s) that should be selected for its effective deployment in a specific environment/context.

On the other hand, data in IoT is provided by different data sources. Trustworthiness of sources can be represented by *trust semantics* that describe quality and trust-related attributes for their providers and the sources themselves. Semantics can play an important role for defining trust and reliability attributes (Barnaghi et al., 2012). In addition, the high level of heterogeneity in IoT is expected to magnify security threats during the interaction of humans, machines, and robots, in any combination (Sicari et al., 2014). Furthermore, multiple heterogeneous IoT entities operating in different contexts exchange information with each other, and this complicates the design and deployment of efficient, interoperable and scalable security mechanisms. The size and heterogeneity of the IoT affects (Roman et al., 2013; Yan et al., 2014):

a trust in the interactions between entities

b trust in the system, from the users' perspective.

In a distributed and open IoT, there is uncertainty in both the interactions with the data providers and the interactions with the service providers. The most relevant and trustworthy data providers must be discovered and selected for trustworthy service delivery. A distributed and open infrastructure makes the management of trust more complicated, in terms of selecting and sharing the most appropriate methods/algorithms for calculating trust values or in terms of choosing the most suitable trust model (e.g., ontology).

There are open trust-related issues that the state of the art in IoT needs to address, such as managing trust without the existence of central authorities, and those issues require clear and simple semantics towards solving interoperability as a first step (before going into 'deeper' security issues). Trust management mechanisms have been widely studied in various research fields (Ruohomaa and Kutvonen, 2005; Viljanen, 2005; Yan et al., 2014). However, current research has not comprehensively investigated how to manage trust in IoT in a holistic manner (Yan et al., 2014). Seamless integration and cooperation of trust management mechanisms for achieving a holistic trust management in IoT is needed. The definition of a distributed and dynamic approach suitable for the scalable and openIoT context is still missing (Sicari et al., 2014). The introduction of a well-defined trust negotiation language supporting the semantic interoperability of IoT context is still an open IoT-trust management issue (Sicari et al., 2014; Mahmud Hossain et al., 2015).

The aim of this paper is to semantically enable trust in distributed and open IoT in order to ensure and secure the selection and deployment of heterogeneous IoT entities, without the existence of central trust authorities. By providing a degree of trustworthiness between heterogeneous IoT entities at a higher level of abstraction it is possible to *ensure* that the deployment of heterogeneous entities in the open IoT will be performed (by selecting entities with the higher trust value). In addition, such a way the deployment is *secured* since it involves the most trustworthy registered IoT entities from the available (matched) ones.

Towards this aim, the paper presents a simple but effective approach with the following contributions:

a propose a novel method for easy extension of any IoT ontology, introducing simple and extensible semantics related to trust between IoT entities

b reuse *trust semantics* from existing trust models/ontologies (Huang and Fox, 2006; Viljanen, 2005)

c define *trust semantics* using the existing framework of *FuzzyOwl2* that uses current standard languages and resources, a fuzzy extension of OWL 2

d propose a context-based method for computing trust, extending state-of-the-art well-defined and evaluated work on dynamic trust management for community-based social IoT environment, with no centralised trusted authority.

The paper is structured as follows: Section 2 provides related work on trust modelling for IoT, and Section 3 briefly discusses the main background concepts of semantic interoperability in IoT, fuzzy semantics and trust. Section 4 presents the proposed modelling approach along with a working scenario, and Section 5 introduces a context-based method for computing trust. Finally, Section 6 concludes the paper with future plans highlighted.

2 Related work

In Huang and Fox (2006), an ontology of trust is defined, specifying two types of trust, trust in belief (trust based on an agent believing in what another agent believes) and trust in performance (trust based on believing that another agent will perform an activity correctly). The 'trustor' (object property) is the agent performing the trusting and the 'trustee' is the agent that is being trusted. A 'degree of trust' is a number between 0 and 1 that signifies the degree to which the trustor trusts the trustee. A working ontology is available at http://ontology.eil.utoronto.ca/trust.owl. This related work focuses on the transitivity of trust in social networks.

In Viljanen (2005) authors conducted an extensive survey and classified 13 computational trust models by trust decision input factors. Their analysis is used to propose a new ontology for trust to facilitate interaction between business systems, focusing its utilisation in digital business. A working ontology file in the related paper's corresponding URL (http://www.cs.helsinki.fi/u/viljanen/trust.owl) was not accessible (broken link) during the preparation of this paper.

In Ceolin et al. (2014) authors introduce an ontology for modelling trust which extends with recent trust theories another existing model (Alnemr et al., 2010) that focuses on the computational part of trust, rather than on social and agent aspects. Although the presented model is an updated extension of other efforts towards modelling trust, it focuses on the specific issue of trusting (web) data. A working ontology file is not available (at least, not mentioned in the related paper).

In the presented work, the related trust ontologies have been studied and their common semantics have been reused as well as those that were aligned to the aim of the proposed model. To the best of our knowledge, there is not any related effort of integrating *trust semantics* in IoT ontologies towards supporting interoperability, aiming to ensure and secure automated deployment of IoT entities in specific IoT environments where a centralised trusted authority is not present.

3 Background

The presented work is mainly based on the topic of semantic interoperability in IoT as well as on fuzzy semantics and trust in IoT. The following sections provide basic background knowledge on these topics.

3.1 Semantic interoperability in IoT

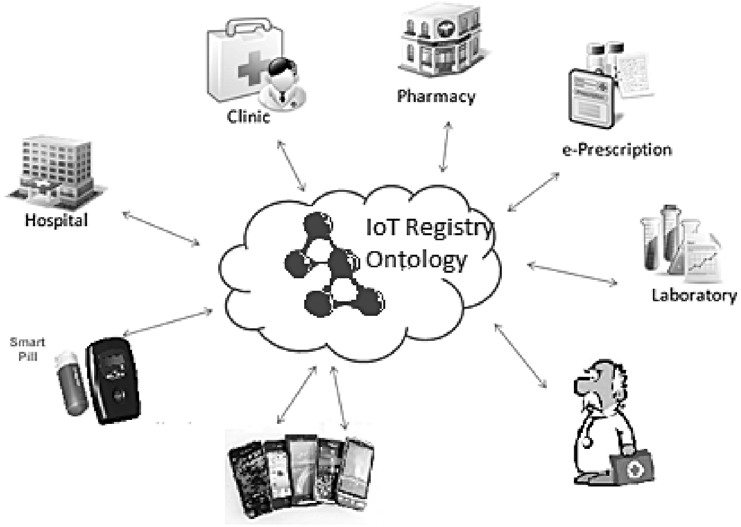
In previous work (Kotis and Katasonov, 2013), authors focus on the use of semantic technologies for the automated deployment of heterogeneous and distributed IoT entities, supporting the following three distinct tasks:

a semantic registration of IoT entities

b alignment of IoT entities' metadata and use of these alignments for their matchmaking

c alignment of the semantics of the messages' data that are exchanged between these IoT entities during device-to-application communication.

Figure 1 IoT ontology as a semantic registry of entities



Such a work considers ontologies as a key technology to solve the problem of automating the deployment of applications in heterogeneous IoT environments, allowing any IoT entity to unambiguously convey the meaning of data/information they 'carry'. The aim of the presented IoT ontology as an abstraction technology is to hide heterogeneity of IoT entities, acting as a mediator between IoT application providers and consumers, and to support their semantic matchmaking. Acting as a mediator, the ontology objective is to be used by the interested stakeholders independently, as a registry for the semantic registration of IoT entities (Figure 1), by the IoT application providers/developers that will register their software, as well as by the IoT application. The IoT-ontology proposed in this work is mainly reusing the semantic sensor network (SSN) ontology and the upper ontology DUL, supporting the IoT-semantic smart gateway framework (SSGF) framework by representing different types of IoT entities that are fundamental parts of the IoT domain. A formal and explicit representation of all types of IoT entities and their associations is required in order to serve as the semantic registry of the real-world entities (Figure 1).

An example of using the ontology is provided in this paper, borrowed from this research line, demonstrating the registration of a smart room and a smart lamp entity in a smart room scenario (a lamp is switched on when motion is detected). The reuse of SSN (*ssn* prefix) and DUL (*dul* prefix) ontologies and the use of new IoT concepts (*iot* prefix) can be observed in Turtle-syntax examples provided below, but details on the actual definitions can be found in Kotis and Katasonov (2013).

'Smart room' example description:

```
: E023 a iot: Room.
: SmartRoom a iot: SmartEntity;
    ssn: featureOfInterest: E023;
    dul: includesObject: MotionDetector;
    dul: isConceptualisedBy [
        a iot: SoftwareAgent;
        iot: providesService                    : DetectionService
        ].
```

'Smart lamp' example description:

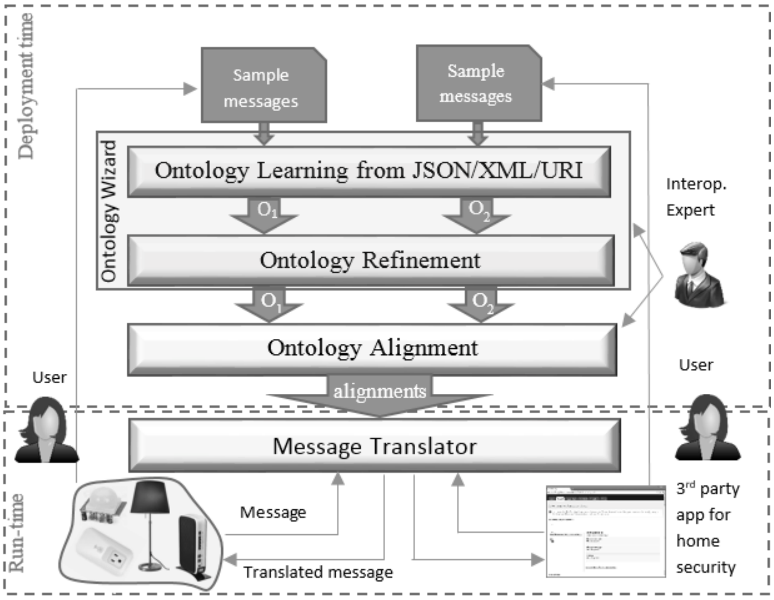
```
: Lamp a dul: DesignedArtifact,: LampType.
: LampType a owl: Class; rdfs: label 'Light'@en.
: Switch a iot: Actuator, iot: ActuatingDevice.
: SmartLamp a iot: SmartEntity;
    ssn: featureOfInterest: Lamp;
    dul: includesObject: Switch.
```

Lets now assume that a generic application has been developed, implementing the function 'switch a light when a movement is detected in the room'. This application will be registered in the IoT ontology (by the IoT service provider and application developer) as an application that provides some light service and conceptualises a control entity. The instantiation of the specific service that the IoT service provider (application developer) provides are described in detail in Kotis and Katasonov (2013), however here the definition of a control entity that provides a light service is presented:

```
: Control a iot: ControlEntity;
    dul: isConceptualisedBy: Application.
    : Application a iot: Application;
    iot: providesService: LightService.
```

As it is depicted in Figure 2, the execution of a third party generic application developed for home security is utilising a set of devices, communicating with them via a gateway box and a message translator that semantically aligns messages from both parts, i.e., the application part and the devices. But how these semantics have been uncovered? This has been previously performed at the deployment time using a set of sample messages.

Figure 2 The smart proxy architecture instantiation for 'smart room' scenario



An 'ontology wizard' component (Figure 2) is responsible for transforming IoT device's and application's messages exchanged between each other or via a gateway (e.g., ThereGate) from JSON or XML or URI format to ontological definitions of OWL classes and properties, as well as to refine those sets using some heuristic rules (e.g., to handle structural issues). The two sets of ontology definitions, one set for the device and one for the application, are then processed by an 'ontology alignment' component in order to obtain their similarities and compute alignments between them. These alignments (computed at the deployment time) are then used by the 'message translator' component at run-time for a bi-directional translation of messages.

The work presented in this paper is based on the previous work of smart proxy and SSGF (Kotis and Katasonov, 2013) and reuses the proposed IoT ontology as an example of extending IoT semantics with *trust semantics*. By providing a degree of trustworthiness between heterogeneous IoT entities at the higher level of abstraction it is possible to ensure that the deployment of heterogeneous entities in the open IoT will be performed by selecting the entities with the higher trust values.

3.2 Fuzzy semantics

The introduction of trust in terms of confidence values in the interval of [0, 1] for relations between concepts and properties has been extensively explored in *fuzzy ontologies*. The probability of an IoT entity (e.g., an app) to be related with another IoT entity (e.g., a device) through a particular semantic relation (e.g., Application × trusts Device y) can be used in open environments to avoid unauthorised/untrustworthy communication between 'foreign' entities as well as to play the role of a secure *selection* *key* for automated deployment, in environments with no central trust authority. In this work the use of fuzzy ontologies as a *semantic enabler* for trust in IoT is presented.

The presented work considers a *fuzzy formula* (or fuzzy axioms) of the form $\varphi \geq \alpha$ or $\varphi \leq \beta$, where φ is a fuzzy proposition and $\alpha, \beta \in [0, 1]$. This imposes that the degree of truth of φ is *at least* α (resp. *at most* β). For example, the proposition '*x is a reliable temperature sensor* $\geq$ 0.9' says that we have a rather reliable temperature sensor (the degree of truth of x being a reliable temperature sensor is at least 0.9). In addition, a (binary) *fuzzy relation* R over two countable classical sets X and Y is a function $R: X \times Y \rightarrow [0, 1]$.

The presented work integrates the fuzzy extension of OWL 2, fuzzyOwl2 (Bobillo and Straccia, 2011a). The use of annotation properties in this formalism allows

a to use current OWL 2 editors for fuzzy ontology representation

b OWL 2 reasoners to discard the fuzzy part of a fuzzy ontology, producing almost the same results as if it would not exist.

Fuzzy OWL 2 assumes three alphabets of symbols, for *fuzzy concepts*, *fuzzy roles* and *individuals*. In fuzzy OWL 2, fuzzy concepts denote fuzzy sets of individuals and fuzzy roles denote fuzzy binary relations.

3.3 Trust

An attempt to produce a general definition and conceptual analysis of trust (and of the related idea of trustworthiness) has been recently made by O'Hara (2012). According to this report, *trust is an attitude that one takes to the trustworthiness of another; in turn, the other's trustworthiness is a property that they have*. Trustworthiness can be expressed as a quadruple:

Tw < Y, Z, R(A), C >

Y and Z are agents, R is a representation of behaviour aimed at an audience A, and C is a context. This states that Y is trustworthy, assuming that there is some context for Y's trustworthiness. The context C is some type of relevant restriction of the circumstances in which Y is claimed to be willing, able and motivated to conform to R (in our work, R is considered to be always equivalent with the behaviour of 'being reliable'). Furthermore, if Y is trustworthy in all (or most) specific contexts where she has a duty, or is claimed, to be trustworthy, then she is generally trustworthy.

Trust is an attitude toward the trustworthiness of an individual. If X trusts Y, then X has a positive view of Y's trustworthiness. If an agent's attitude toward another agent is considered to be a belief about that agent, then: 'X trusts Y' is equal to the definition that 'X believes that Y is trustworthy', where X and Y are agents.

4 Enabling trust in IoT ontologies

To support the demonstration of the proposed approach, let us consider a use case scenario where an entity A trusts an entity B (as *being reliable*) with a trust degree equals to 0.8 and entity A trusts another entity C with a trust degree equals to 0.2. Entities A, B and C are heterogeneous ('foreign' to each other) IoT entities that share however the same environment/context at a specific time interval, and all three are registered in a common publicly available IoT registry (high level IoT layer, at the information layer, e.g., an IoT ontology operating as a registry service).

Let us now explicate the scenario; placing the entities in the specific context of a smart room i.e., if motion is detected in the room then room's lamp is switched on. In this scenario, A is a smart application and B, C are motion detection sensors, and A must be deployed in their common environment (the room) where B and C have already been deployed. There might be also the case where other entities of the same or different type (e.g., other smart lamps), have also being deployed. In such a case, entity A cannot 'decide' which one of the matched (based on the smart proxy computation of alignments of their specifications) entities is most appropriate to use for the execution of its functionalities i.e., in this example, which motion detection sensor to select. For this reason, an automated deployment of the application cannot be ensured (if decision cannot be made). However, by providing a degree of trustworthiness between IoT entities at a higher level of abstraction it is possible to *ensure* that the deployment of IoT entities in such scenarios will be performed: entity A will select to utilise the entity with the higher trust value among all matched entities in its context. Such value/degree of trust may be computed using a function that takes into account environmental/context information as well as other related information e.g., who the provider and owner of the entity is, what are the security policies of this entity, what are the previous deployment statistics of the entity, etc. Such a trustworthy deployment, can be considered also a *secure deployment,* since it involves the most trustworthy entities selected from the available (matched) ones within the deployment environment/context.

As an alternative scenario, let us consider a conference room context where a third-party generic broadcasting application is 'searching' for the most trustworthy recording devices (microphones, cameras, smart phones with embedded capabilities) of registered visitors, before deploying itself in the environment, or a smart city IoT application running on Alice's smartphone for air pollution detection and a number of air pollution sensor devices, all available in the city area of interest of Alice.

So the problem one is facing in those settings is basically a selection problem, i.e., an application needs to decide, among the feature-based matched devices, which one to select for its deployment, or better, which one to trust more than the others. It is actually a two-level selection problem, a feature-based one and a trust-based one. In the first level, among a wide set of IoT entities (both applications and devices), the problem is how to select devices that fulfil applications' I/O requirements. So, a motion detection application for room security, running simple logic saying that 'when a movement is detected in the room switch on a lamp', will select devices such as motion detection sensors and smart lamps (binding of smart switch and lamp).In the second level, among the narrow set of IoT entities that has been selected for the motion detection app, the problem is how to select devices that are more trustworthy than others, thus more reliable for the deployment of the specific application at the specific context e.g., in a room. The work presented in the paper is clearly focusing on the second level of selection, proposing a trustworthiness framework for:

a for the representation of trustworthiness of IoT entities during their deployment

b the computation of trust values.

For the demonstration of the proposed modelling approach, the IoT ontology and the automated deployment process of IoT entities presented in Kotis et al. (2012) will be used. As already stated, this work extends IoT ontologies with *trust semantics*, and this is achieved by reusing only the main class of any IoT ontology i.e., 'IoT-entity' class.

4.1 The smart room scenario

In a smart room context, the following IoT entities have been registered (in the IoT-ontology):

- a smart room application (SmartRoomApp) which is capable of controlling lights in a room, based on the sensing of a motion detector
- two motion detection sensors provided by different vendors, A and B (using different namespaces for specifying their semantics) and owned by different agents (namely, 'me' and 'her')
- two smart lamps (a lamp attached to a smart switch) also provided by those two providers and owned by the same agents.

According to previous work (Kotis and Katasonov, 2013), the task of matchmaking of entities' specifications, as part of the overall smart proxy solution in the SSGF, should align and match the semantics of the registered entities, facilitating such way the communication of the application (via message translation) with the appropriate entities. However, in the presented scenario more than one entity of the same type available for this application to function is considered, and assumed that all entities of the same type have the same matching score in the specifications' matchmaking output of the smart proxy. For further details on the matchmaking task please refer at Kotis et al. (2012). In the open IoT, where a large number of applications and devices will be registered in different publicly available IoT registries, such a scenario is more than likely to be seen in a quite larger scale (possibly hundreds of devices of the same type and functionality can be possibly used by a generic third-party application within the same environment/context).

So, the question to answer in this scenario, which is the main concept of our work, is: which of the matched entities (motion detection devices in this case) the application must use to execute its logic? Authors conjecture that the key to this answer is 'trustworthiness', as in real life, where humans based on who they trust more or less, choose to be coupled only with a subset of those who they possibly match with, or choose to buy only from a specific seller among those who provide exactly the same products and prices.

In the following paragraphs a solution based on the notion of adding *trust semantics* in IoT domain is presented. Such semantics are provided as the key to an IoT application/service to select, among the available devices, the most suitable ones for deployment, i.e., the ones that the application trusts more than others. How this degree of trust is obtained in such a fuzzy environment will be introduced in Section 5.

For demonstration reasons specific example namespaces at the following ontology IRIs have been used:

- IRI of IoT ontology: http://purl.org/IoT/iot, prefix: iot
- IRI of IoT trust ontology: http://purl.org/IoT/iot-trust, prefix: iot-trust
- IRI of IoT application example domain ontology: http://purl.org/IoT/iot-app, prefix: iot-app
- IRI of IoT device provider A: http://purl.org/IoT/iot-provA, prefix: iot-provA
- IRI of IoT device provider B: http://purl.org/IoT/iot-provB, prefix: iot-provB.

4.2 The simple IoT trust model

As already stated, in this work IoT ontologies are extended with *trust semantics*, and this is simply achieved by reusing only the main (and most common) class of any IoT ontology i.e., ‘iot: IoT-entity’. Our simple model introduces a binary relation between two IoT entities (‘iot: IoT-entity’) using the object property ‘iot-trust: trusts’ and its inverse (‘owl: inverseOf’) property ‘iot-trust: trustedBy’. In addition, the model introduces such a property as a non-taxonomy fuzzy associative relationship, using *fuzzyOwl2* semantics.

This simple definition of trust semantics are provided below (in Manchester syntax):

```
ObjectProperty: trusts
    Annotations:
        fuzzyLabel “<fuzzyOwl2 fuzzyType=\”role\”>
            <Role type=\”modified\”modifier=\”trustModifier\” base=\”topObjectProperty\”/>
            </fuzzyOwl2>“
    Domain:
        IoT-entity
    Range:
        IoT-entity
    InverseOf:
        trustedBy
```

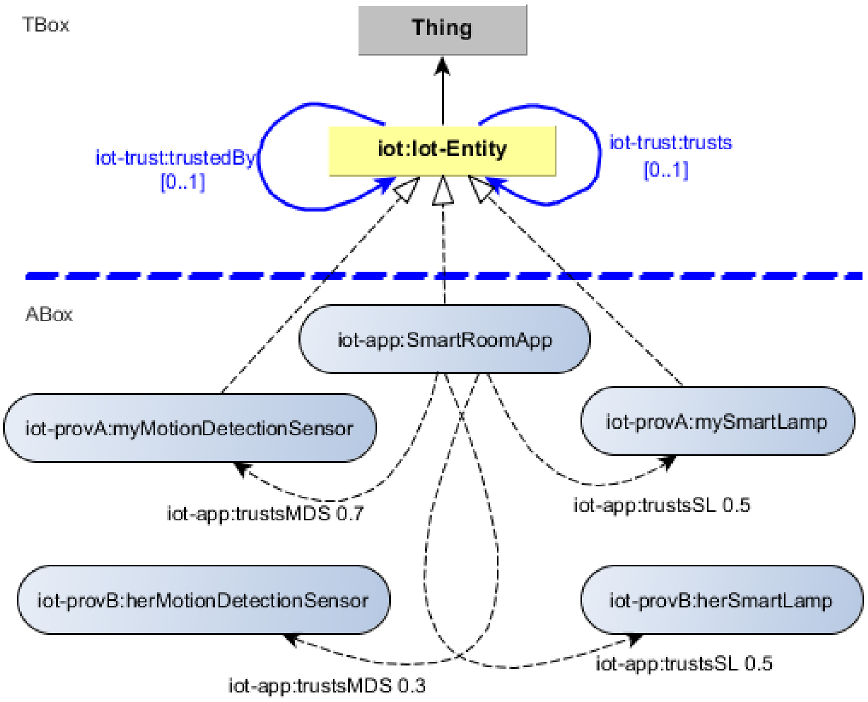
A graphical representation of the model is depicted in Figure 3, instantiated with IoT entities taken from our smart room scenario. Specific instantiations of the trust property i.e., trusts motion detection sensor (‘iot-app: trustsMDS’) and trusts smart lamp (‘iot-app: trustsSL’) can be defined as sub-properties (more specific properties) of iot-trust: trusts. The actual proposed model’s semantics are depicted in the TBox area of Figure 3, where the instantiations of the model’s semantics using the example scenario are placed in the area of ABox (the two areas are distinguished with a dashed horizontal line across the figure). Specific degrees of trust between IoT entities are specified at the specific sub-properties of ‘iot-trust: trust’ property, i.e., at the ‘iot-app: trustsSL’ and ‘iot-app: trustsMDS’ respectively.

Please notice that for the shake of demonstration simplicity as well as to align with the aim of this work (focusing on the deployment of applications in environments where devices have been already registered and deployed), the paper only depicts relations between the application and the devices, however other relations representing trust may also exist (e.g., between the devices themselves).

As stated, we've used fuzzy semantics representation of the 'iot-trust: trusts' object property in order to capture the degree of confidence in the interval of [0, 1]. Using the fuzzyOwl2 plugin of protégé ontology engineering environment, it was possible to translate the instantiated example model in the well-known FuzzyDL (Bobillo and Straccia, 2011b) representation language. Such a representation is provided below:

```
(define-modifier trustModifier linear-modifier(1.0))
(define-primitive-concept IoT-entity *top*)
(inverse trustedBy trusts)
(domain trustedBy IoT-entity)
(domain trusts IoT-entity)
(range trusts IoT-entity)
(range trustedBy IoT-entity)
(related SmartRoomApp herSmartLamp trustsSL 0.5)
(related SmartRoomApp mySmartLamp trustsSL 0.5)
(related SmartRoomApp herMotionDetectionSensor trustsMDS 0.3)
(related SmartRoomApp myMotionDetectionSensor trustsMDS 0.7)
(implies-role trustsMDS trusts 1.0)
(implies-role trustsSL trusts 1.0)
```

Figure 3 The simple trust model for IoT entities (see online version for colours)



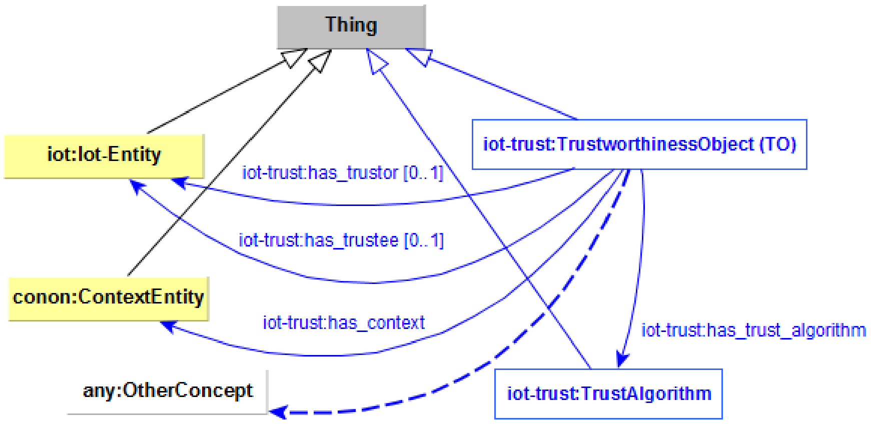
The example ontology assertion (related SmartRoomApp mySmartLamp trustsSL 0.5) is presented below in RDF/OWL syntax:

```
<ObjectPropertyAssertion>
<Annotation>
    <AnnotationProperty IRI="#fuzzyLabel"/>
        <LiteraldatatypeIRI="&rdf;PlainLiteral">
            <fuzzyOwl2fuzzyType="axiom">
                <Degreevalue="0.5"/>
            </fuzzyOwl2>
        </Literal>
            </Annotation>
    <ObjectProperty IRI="iot-trust#trustsSL"/>
    <NamedIndividual IRI="iot-trust#SmartRoomApp"/>
    <NamedIndividual IRI="iot-trust#mySmartLamp"/>
    </ObjectPropertyAssertion>
```

4.3 An extensible IoT trust model

To provide a simple but extensible representation of our proposed model, we've introduced the class 'iot-trust: TrustworthinessObject' (TO) as the domain class of two also introduced fuzzy object properties 'iot-trust: has_trustor' and 'iot-trust: has_trustee'. Both properties have 'iot: IoT-Entity' as range class. Based on this flexible definition of TO, additional properties can be defined towards extending the model such as, the context ('conon: ContextEntity') of the trustworthiness (via the 'iot-trust: has_context' object property) or the trust algorithm ('iot-trust: TrustAlgorithm') used to compute the trust values (via the 'iot-trust: trust_algorithm' object property). Any other possibly useful property related to the trustworthiness of an IoT entity pair may be easily added by specifying 'iot-trust: TrustworthinessObject' as its domain class. A graphical representation of the extensible model is provided in Figure 4.

Figure 4 The simple extensible IoT trust model (see online version for colours)


Thing
iot:Iot-Entity
iot-trust:TrustworthinessObject (TO)
iot-trust:has_trustor [0..1]
iot-trust:has_trustee [0..1]
conon:ContextEntity
iot-trust:has_context
iot-trust:has_trust_algorithm
any:OtherConcept
iot-trust:TrustAlgorithm

As a context-related namespace the context upper ontology CONON (Wang et al., 2004) (prefixconon) has been used. The model in FuzzyDL representation is provided below:

```
(define-modifier trustModifier linear-modifier(1.0))
(define-primitive-concept IoT-entity *top*)
(define-primitive-concept TrustAlgorithm *top*)
(define-primitive-concept TrustworthinessObject *top*)
(domain has_trustee TrustworthinessObject)
(domain has_trustor TrustworthinessObject)
(domain trust_algorithm TrustworthinessObject)
(range has_trustor IoT-entity)
(range has_trustee IoT-entity)
(range trust_algorithm TrustAlgorithm)
```

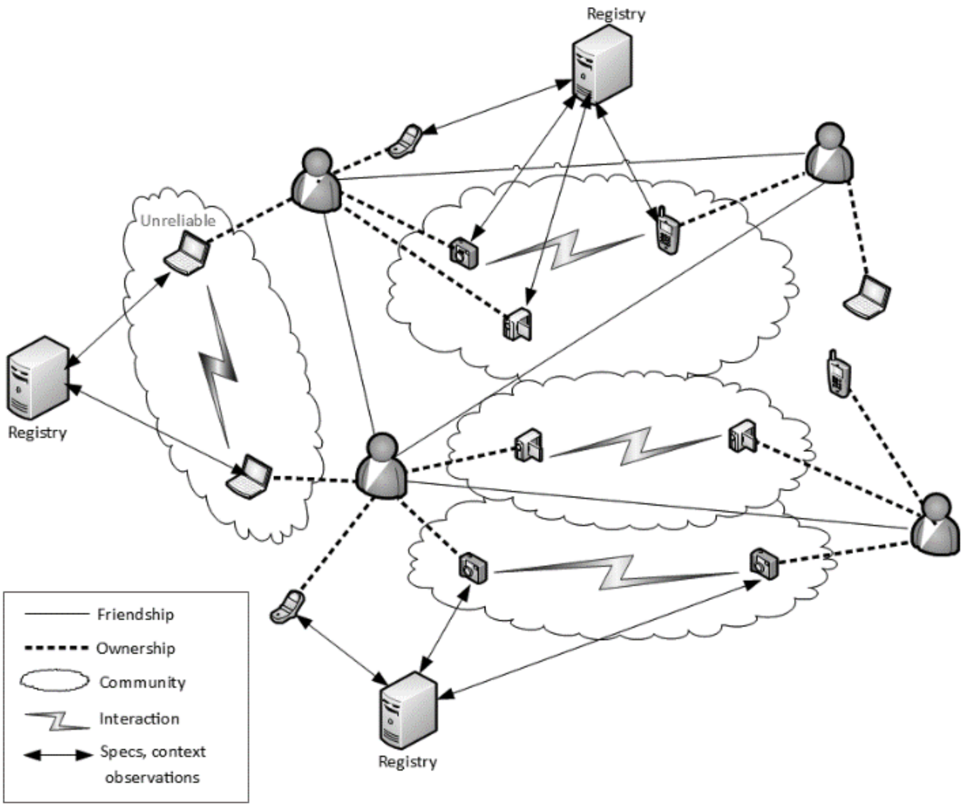
The latest working version of the extended IoT-trust ontology in OWL and FuzzyDL serialisation, as well as the instantiated simple one introduced in Section 4.2, can be accessed at http://ai-group.ds.unipi.gr/kotis/ontologies/IoT-trust-ontology.

5 Computing trust values

The degree of trust, the degree to which X believes that Y is trustworthy, or the confidence that X has in his attitude, is an important parameter of trust. Such a degree (trust value) can be estimated by a computational function in several ways. As stated by O'Hara (2012), one could apply a number of disciplines depending on one's purposes, but nothing very complex is required. One of the ways is by sketching a model like the one presented in this paper, modelling the degree of confidence in a fine-grained way as a real number between 0 and 1. In any case, following O'Hara's definitions on context-depended trustworthiness, the paper accentuates the local (context-depended) range of trust value computation as highly significant for IoT environments and smart spaces, focusing less on the general trustworthiness of IoT entities. Translating this into a more pragmatic statement, a device (e.g., a video camera, a microphone, a phone with embedded camera and microphone) may be more reliable (thus more trustworthy) in a specific environment C (e.g., in a conference room) than in another (e.g., outside spot under sun, near sea and traffic) based on environmental conditions (e.g., sun, noise, moisture levels) that affect the function and consequently the reliability of the device and the performance of the application (unreliable and misbehaving devices minimise applications' performance).

In related work (Bao and Chen, 2012a, 2012b; Chen et al., 2015), a dynamic trust management for a community-based social IoT environment by considering multiple social relationships among device owners is proposed. In this work, a Social IoT (Atzori et al., 2010) environment with no centralised trusted authority is considered (see Figure 5, slightly changed version of the one depicted in Bao and Chen, 2012a), introducing social relationships such as ownership, friendship, community.

Figure 5 Social structure of IoT



The trust value is a real number in the interval [0, 1], where X are trust properties (honesty, cooperativeness, or community-interest), and *i, j* are nodes of the social network. Such a protocol takes the three social relationships into account and advocates the use of the three trust properties to evaluate trust. The approach deals with misbehaving-unreliable nodes (devises) whose status or behaviour may change dynamically, influencing (minimising) applications' performance.

In the line of this research, we propose a similar approach, extending the computation of the degree of trust by a context-depended property i.e., *capacity*. Capacity *cap* is defined as the ability of an IoT entity (a device or an application) to function within specific context requirements (e.g., environmental properties (handled as requirements to be met) such as light, noise, temperature). Such requirements are specified in the IoT ontology at the context level definition, and matched against devices' and applications' specs (also specified in the IoT ontology during their registration in the semantic registry). Such a matching task results to a capacity signature *cap* of an IoT entity *E* for a specific context *C*, i.e., to a capacity value for each device per context. Such a signature then is taken into consideration for the computation of trust value between two IoT entities.

In the well-defined and evaluated work of dynamic trust management protocol of Bao and Chen (2012a) and Chen et al. (2015), the estimation of between nodes i, j (via direct observation) is computed in every time instance for the three X properties (honesty, cooperativeness, community-interest), and takes into consideration the effect of past events/experiences. Honesty represents whether or not a node is honest. In an IoT system, a malicious node can be dishonest when providing services or trust recommendations. A dishonest node can severely disrupt trust management and service continuity of an IoT application. Cooperativeness represents whether or not the trustee node is socially cooperative with the trustor node. A node can evaluate the cooperativeness property of other nodes based on social ties. Community-interest trust represents whether or not the trustor and trustee nodes are in the same social communities/groups or have similar capabilities. Two nodes with a degree of high community-interest trust have more chances and experiences in interacting with each other.

In the presented approach this protocol is extended by adding the additional (fourth) property capacity (*cap*), which represents the degree of satisfaction of the dynamic environmental/context-conditions/factors from technical specs of a node, at a particular time instance, i.e., a matching score related to how well a device can function within specific environmental conditions given their specific capacity. *Capacity* is a value in the interval [0, 1], and for each node *k* it is computed via direct observation according to the following:

$$D_k^{Capacity} = \sum_{n=1}^{f} w_n \cdot cap_n$$

where

f the number of environmental factors that are taken into consideration during the specification of the context

w_n the weight factor of every environmental factor, with respect to the context.

w_n specifies how important is the role of an environmental factor within a specific context. For instance, day light measurement is more important than humidity in the context of a video streaming service. For w_n the following applies:

$$\sum_{n=1}^{f} w_n = 1$$

cap_n: the degree of satisfaction (from technical specs of a node) of the specific requirements that result from the analysis of the environmental factors that are present in the specific context. Cap_n is a number in the interval [0, 1], computed by a mapping function f: $S_1 \rightarrow S_2$, where S_1 kai S_2 are signatures of the ontological specifications that describe environmental conditions and devices specs respectively. As stated in related work (Kotis et al., 2012), the problem of computing mappings between ontologies can be formally described as follows: Given two ontologies $O_1 = (S_1, A_1)$, $O_2 = (S_2, A_2)$ (where S_i denotes the signature and A$_i$ the set of axioms that specify the intended meaning of terms in S_i) and an element (class or property) E_i^1 in the signature S_1 of O_1, locate a corresponding element E_i^2 in S_2, such that a mapping relation (E_i^1, E_j^2, r) holds between them. *r* can be any relation such as the equivalence (≡) or the sub-sumption (⊑) axiom or any other semantic relation e.g., meronym. For any such correspondence a mapping method may relate a value *γ* that represents the preference to relating E_i^1 with E_j^2 via *r*. If there is not such a preference, it is assumed that the method equally prefers any such assessed relation for the element E_1. The correspondence is denoted by $(E_i^1, E_j^2, r, \gamma)$. The set of computed mapping relations produces the mapping function f: $S_1 \rightarrow S_2$ that must preserve the semantics of representation: i.e., all models of axioms A_2 must be models of the translated A_1 axioms: i.e., $A_2 \vDash \mathrm{f}(A_1)$.

Summarising the presented approach for computing context-based trust, a new trust property, *capacity*, is introduced in the dynamic trust management for community-based social IoT environments. Such property, although a dynamic one, is depended on objective characteristics such as the environmental conditions' measurements and the registered devices' tech specs of each node (IoT entity). Thus, errors in the computations maybe inherited mainly from measurements errors than from malicious behaviours of nodes.

6 Conclusions and future work

This paper presents a simple but extensible trust model that is seamlessly integrated in IoT ontologies, towards semantically enabling IoT trust for ensuring and securing IoT entities' effective deployment in specific contexts. The work presented in this paper is focusing on IoT trust modelling, but also introduces a method for computing context-based trust with no centralised trusted authority, extending state-of-the-art well-defined and evaluated approach on dynamic trust management for community-based social IoT environment. The extension of this particular computation model is under implementation using the NS-3-based[1] simulation system provided to us by its developers (Bao and Chen, 2012a).

Future work includes overall evaluation of the proposed *trustworthiness framework* and further investigation of the context-based trust computation, taking into account information such as who the provider and owner of the entity is, what are the security policies of this entity, what are the previous deployment statistics of the entity, etc.

Nevertheless, the work presented in this paper contributes the following:

a propose a novel method for easy extension of any IoT ontology, introducing simple and extensible semantics related to trust between IoT entities

b reuse trust semantics from existing trust models/ontologies

c defines trust semantics using the existing framework of *FuzzyOwl2* that uses current standard languages and resources

d introduces an under-development extension to a well-defined and evaluated method for dynamically computing context-based trust with no centralised trusted authority.

The presented work is expected to advance automation in IoT entities' deployment as well as interoperability in IoT trust, with clear implications in the advancement of trust-based IoT interoperability in general.

References

Alnemr, R., Paschke, A. and Meinel, C. (2010) 'Enabling reputation interoperability through semantic technologies', *Proceedings of the 6th International Conference on Semantic Systems (I-SEMANTICS'10)*, ACM, pp.1–9.

Amarnath, P., Durgaprasad, K. and Pasupuleti, S. (2016) 'Semantic internet of things', *IEEE Tenth International Conference on Semantic Computing (ICSC)*, pp.91–95.

Atzori, L., Iera, A., Morabito, G. and Nitti, M. (2010) 'The social internet of things (SIoT) – when social networks meet the internet of things: concept, architecture and network characterization', *Computer Networks*, Vol. 56, No. 16, pp.3594–3608.

Bao, F. and Chen, I. (2012a) 'Dynamic trust management for internet of things applications', *Proceedings of the 2012 International Workshop on Self-Aware Internet of Things (Self-IoT '12)*, ACM, New York, NY, USA, pp.1–6.

Bao, F. and Chen, I. (2012b) 'Trust management for internet of things and its application to service composition', *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM*, pp.1–6.

Bao, F. and Guo, J. (2015) 'Trust-based service management for social internet of things systems', *IEEE Transactions on Dependable and Secure Computing*, No. 99.

Barnaghi, P., Wang, W., Henson, C. and Taylor, K. (2012) 'Semantics for the internet of things: early progress and back to the future', *Int. J. Semant. Web Inf. Syst.*, Vol. 8, No. 1, pp.1–21.

Bobillo, F. and Straccia, U. (2011a) 'Fuzzy ontology representation using OWL 2', *International Journal of Approximate Reasoning*, Vol. 52, No. 7, pp.1073–1094.

Bobillo, F. and Straccia, U. (2011b) *Syntax and Semantics of Fuzzy DL* [online] http://www.umbertostraccia.it/cs/software/fuzzyDL/download/old/documents/syntax.pdf (accessed 2 July 2016).

Ceolin, D., Nottamkandath, A., Fokkink, W.J. and Maccatrozzo, V. (2014) 'Towards the definition of an ontology for trust in (web) data', *Proc. 10th Workshop on Uncertainty Reasoning for the Semantic Web – URSW'14*, Riva del Garda, CEUR Workshop Proceedings No. 1259, pp.73–78.

Chen, I., Bao, F. and Guo, J. (2015) 'Trust-based service management for social internet of things systems', *IEEE Transactions on Dependable and Secure Computing*, Vol. PP, No. 99, pp.1–1.

Galov, I.V., Lomov, A.A. and Korzun, D.G. (2015) 'Design of semantic information broker for localized computing environments in the internet of things', *17th Conference of Open Innovations Association (FRUCT)*, pp.36–43.

Gyrard, A., Datta, S.K., Bonnet, C. and Boudaoud, K. (2015) 'A semantic engine for internet of things: cloud, mobile devices and gateways', *9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, pp.336–341.

Hachem, S., Teixeira, T. and Issarny, V. (2011) 'Ontologies for the internet of things', *Proceedings of the 8th Middleware Doctoral Symposium (MDS '11)*, ACM, New York, NY, USA, Article 3, 6pp.

Huang, J. and Fox, M.S. (2006) 'An ontology of trust – formal semantics and transitivity', *Proceedings of the International Conference on Electronic Commerce*, Association of Computing Machinery, pp.259–270.

Kiljander, J., D'elia, A., Morandi, F., Hyttinen, P., Takalo-Mattila, J., Ylisaukko-Oja, A., Soininen, J-P. and Cinotti, T.S. (2014) 'Semantic interoperability architecture for pervasive computing and internet of things', *IEEE Access*, Vol. 2, pp.856–873.

Kotis, K. and Katasonov, A. (2013) 'Semantic interoperability on the internet of things: the semantic smart gateway framework', *International Journal of Distributed Systems and Technologies (IJDST)*, Vol. 4, No. 3, pp.47–69.

Kotis, K., Katasonov, A. and Leino, J. (2012) 'Aligning smart and control entities in IoT', *The 5th Conference on Internet of Things and Smart Spaces, LNCS 7469*, St. Petersburg, RU, Springer, pp.39–50.

Mahmud Hossain, M., Fotouhi, M. and Hasan, R. (2015) 'Towards an analysis of security issues, challenges, and open problems in the internet of things', *IEEE Services Visionary Track on Internet of Things*, New York, USA.

O'Hara, K. (2012) *A General Definition of Trust*, Technical Report, University of Southampton.

Roman, R., Zhou, J. and Lopez, J. (2013) 'On the features and challenges of security and privacy in distributed internet of things', *Computer Networks*, Vol. 57, No. 10, pp.2266–2279.

Ruohomaa, S. and Kutvonen, L. (2005) 'Trust management survey', *Proceedings of iTrust – Trust Management: Third International Conference*, Paris, France, 23–26 May.

Sicari, S., Rizzardi, A., Grieco, L.A. and Coen-Porisini, A. (2014) 'Security, privacy and trust in internet of things: the road ahead', *Computer Networks*, Vol. 76, pp.146–164.

Viljanen, L. (2005) 'Towards an ontology of trust', *Trust, Privacy, and Security in Digital Business*, Vol. 3592 of the series Lecture Notes in Computer Science, pp.175–184, DOI: 10.1007/11537878_18.

Wang, W., De, S., Cassar, G. and Moessner, K. (2013) 'Knowledge representation in the internet of things: semantic modelling and its applications automatika', *Journal for Control, Measurement, Electronics, Computing and Communications*, Vol. 54, No. 4, pp.388–400.

Wang, X.H., Zhang, D., Gu, T. and Pung, H.K. (2004) 'Ontology based context modeling and reasoning using OWL', *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops*, pp.18–22.

Yan, Z., Zhang, P. and Vasilakos, A. (2014) 'A survey on trust management for internet of things', *J. Netw. Comput. Appl.*, Vol. 42, No. 2, pp.120–134.

Notes

[1] https://www.nsnam.org/ns-3-13/.