ETSI TR 103 608 V1.1.1 (2019-07)

TECHNICAL REPORT

SmartM2M;
SAREF publication framework reinforcing the engagement of its community of users

Reference

DTR/SmartM2M-103608

Keywords

framework, IoT, SAREF, semantic

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Copyright Notification

Contents

Intellectual Property Rights 4
Foreword 4
Modal verbs terminology 4
1 Scope 4
2 References 4
2.1 Normative references 4
2.2 Informative references 5
3 Definition of terms, symbols and abbreviations 6
3.1 Terms 6
3.2 Symbols 6
3.3 Abbreviations 6
4 Specification of the SAREF publication framework 7
5 Related initiatives 7
6 Actors and use cases 8
6.0 Introduction 8
6.1 Development actors 8
6.2 Steering actors 9
6.3 Community actors 10
7 Technical requirements 11
7.0 Introduction 11
7.1 Ontological requirements management 11
7.2 Ontology implementation 11
7.3 Ontology documentation 12
7.4 Ontology publication 12
7.5 Ontology Search 12
7.6 Ontology Maintenance 13
7.7 Project Management 13
7.8 Community and social 13
8 Requirements and guidelines based on best practices 14
8.0 Introduction 14
8.1 General requirements for the SAREF ontology and its extensions 14
8.2 Requirements for the publication of the SAREF ontology on the portal 15
8.2.1 Different ontology documents with different versions 15
8.2.2 Namespace and IRIs 15
8.2.3 Requirements for usability and referencing 16
8.3 Guidelines for the development of the SAREF ontology 16
8.3.0 Introduction 16
8.3.1 Naming convention 16
8.3.2 Metadata 17
8.3.2.0 Introduction 17
8.3.2.1 Ontology metadata 17
8.3.2.2 Term metadata 17
8.3.3 Reusing existing ontologies 18
8.3.3.0 Introduction 18
8.3.3.1 Direct import 18
8.3.3.2 Reusing terms of existing ontologies 18
8.3.3.3 Alignment 18
History 20

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Smart Machine-to-Machine communications (SmartM2M).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document specifies the functional requirements for the publication framework of the SAREF ontology and its extensions. The publication framework provides a uniform documentation website for SAREF and its extensions such that (1) the documentation is designed for domain experts and software developers apart from documentation for ontology engineers, (2) industries can interact with the content, provide useful feedback to SAREF developers and suggest additions/modifications. Additionally, SAREF developers would get to know more about the SAREF community of users. Requirements for the publication framework aim at enabling industries to implement solutions with SAREF faster, and to reinforce the engagement of the community of users such that the SAREF developers can plan new evolutions of the current and future extensions.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI TS 103 264 (V1.1.1): "SmartM2M; Smart Appliances; Reference Ontology and oneM2M Mapping".

[i.2] ETSI TS 103 264 (V2.1.1): "SmartM2M; Smart Appliances; Reference Ontology and oneM2M Mapping".

[i.3] ETSI TS 103 410-1 (V1.1.1): "SmartM2M; Smart Appliances Extension to SAREF; Part 1: Energy Domain".

[i.4] ETSI TS 103 410-2 (V1.1.1): "SmartM2M; Smart Appliances Extension to SAREF; Part 2: Environment Domain".

[i.5] ETSI TS 103 410-3 (V1.1.1): "SmartM2M; Smart Appliances Extension to SAREF; Part 3: Building Domain".

[i.6] ETSI TS 103 410-4 (V1.1.1): "SmartM2M Extension to SAREF Part 4: Smart Cities Domain".

[i.7] ETSI TS 103 410-5 (V1.1.1): "SmartM2M; Extension to SAREF; Part 5: Industry and Manufacturing Domains".

[i.8] ETSI TS 103 410-6 (V1.1.1): "SmartM2M; Extension to SAREF; Part 6: Smart Agriculture and Food Chain Domain".

[i.9] ETSI TR 103 411 (V1.1.1): "SmartM2M; Smart Appliances; SAREF extension investigation".

[i.10] Alobaid, A., Garijo, D., Poveda-Villalón, M., Santana-Perez, I., Fernández-Izquierdo, A., Corcho, O.:"Automating ontology engineering support activities with OnToology. Journal of Web Semantics". 2018. In press. ISSN 1570-8268.

NOTE: Available at: https://doi.org/10.1016/j.websem.2018.09.003.

[i.11] Halilaj, L., Petersen, N., Grangel-González, I., Lange, C., Auer, S., Coskun, G., & Lohmann, S. (2015). "VoCol: An Integrated Environment to Support Collaborative Vocabulary Development with Version Control Systems". In 20th International Conference on Knowledge Engineering and Knowledge Management (EKAW'16).

[i.12] M. Lefrançois, J. Kalaoja, T. Ghariani, A. Zimmerman: "The SEAS Knowledge Model", ITEA2 12004 Smart Energy Aware Systems Deliverable 2.2, Jan 2017.

[i.13] Tim Berners-Lee: "Cool URIs don't change", W3C Note, W3C, 1998.

[i.14] Leo Sauermann and Richard Cyganiak: "Cool URIs for the Semantic Web", W3C Note, W3C, December 03 2008.

[i.15] W3C Design issue: "Linked data", 2005.

NOTE: Available at http://www.w3.org/DesignIssues/LinkedData.html.

[i.16] W3C OWL Working Group: "OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition)", W3C Recommendation 11 December 2012. Technical report, W3C, 2012.

[i.17] Pierre-Yves Vandenbussche and Bernard Vatant: "Metadata recommendations for linked open data vocabularies", Web document, 2012.

[i.18] W3C OWL Working Group: "OWL 2 Web Ontology Language Document Overview (Second Edition)", W3C Recommendation 11 December 2012. W3C, 2012.

[i.19] Martin, Philippe. 2000. Propositions of Conventions for RDF. May 28, 2000.

NOTE: Available at http://www-sop.inria.fr/acacia/personnel/phmartin/RDF/conventions.html.

[i.20] D. Beckett, T. Berners-Lee, E. Prud'hommeaux, G. Carothers: "RDF 1.1 Turtle, Terse RDF Triple Language", W3C Recommendation 25 February 2014, W3C, 2014.

[i.21] P. Y. Vandenbussche, G. Atemezing, M. Poveda-Villalón, B. Vatant: "Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web". Semantic Web, 8(3), 437-452. 2017.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following term applies:

ontology: formal specification of a conceptualization, used to explicitly capture the semantics of a certain reality

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

EUREKA	European Research Coordination Agency
HTML	HyperText Markup Language
IoT	Internet of Things
IRI	Internationalized Resource Identifier
ITEA	Information Technology for European Advancement
JSON-LD	Javascript Simple Object Notation for Linked Data

OEG/UPM	Ontology Engineering Group/Universidad Politécnica de Madrid
OGC	Open Geospatial Consortium
OOPS	OntOlogy Pitfall Scanner
OPT	Option
OWL	Web Ontology Language
RDF	Resource Description Framework
REQ	Requirement
SAREF	Smart Applications REFerence ontology
SEAS	Smart Energy Aware Systems
SOSA	Sensor, Observation, Sample, and Actuator
SSN	Semantic Sensor Networks
STF	ETSI Specialist Task Force
TB	Technical Body
TR	Technical Report
TS	Technical Specification
URL	Universal Resource Locator
W3C	World Wide Web Consortium
XML	Extensible Markup Language

4 Specification of the SAREF publication framework

SAREF V2.1.1 [i.2] is a reference ontology for the IoT developed by ETSI SmartM2M in close interaction with the industry. SAREF contains core concepts that are common to several IoT domains and, to be able to handle specific data elements for a certain domain, dedicated extensions of SAREF have been created, for example SAREF4ENER [i.3], SAREF4ENVI [i.4], SAREF4BLDG [i.5], SAREF4CITY [i.6], SAREF4INMA [i.7], SAREF4AGRI [i.8]. Each domain can have one or more extensions, depending on the complexity of the domain. As a reference ontology, SAREF serves as the means to connect the extensions in different domains. The earlier document ETSI TR 103 411 [i.9] specifies the rationale and methodology used to create, publish and maintain the SAREF extensions.

The value of SAREF is strongly correlated with the size of its community of users; therefore the SAREF ontologies should be available on the Web. As such, SAREF users and the industry actors need to be attracted to SAREF with clear documentation and a clear indication about how to provide their input and the kind of input that they can provide.

The ETSI members that contribute to SAREF will therefore be able to get benefit from feedback coming from its open community of industrial users, to better plan new evolution of the current and future extensions, and to reduce the costs of developing these extensions. That being said, the development and monitoring of SAREF lies in ETSI's hands to ensure that high quality standards are met, and users that provide feedback have to understand the implication in terms of IPR. The publication and/or use of such feedback has to therefore be controlled by ETSI, but the possibility to provide feedback will be open to the world.

The present document has been developed in the context of the STF 556 (https://portal.etsi.org/STF/STFs/STFHomePages/STF556.aspx), which was established with the goal to consolidate SAREF and its community of industrial users based on the experience of the EUREKA ITEA 12004 SEAS (Smart Energy Aware Systems) project. The present document specifies the SAREF publication framework to reinforce the engagement of its community of users and to enable them to implement solutions with SAREF faster.

5 Related initiatives

In this clause, some of the main related initiatives in terms of modelling reference ontology patterns for the IoT, and using these ontology patterns to develop ontologies, are reviewed:

- **EUREKA ITEA 12004 SEAS:** The SEAS ontology is a modular and versioned ontology with all the terms it defines having the same namespace (https://w3id.org/seas/). It contains a core of SEAS reference ontology patterns that can be instantiated to create the SEAS ontology itself with a homogeneous and predictable structure for the modelling and the description of any kind of engineering-related data/information/systems. These design patterns and some of their instances fill some of the representational gaps that were identified in SAREF.

- **OnToology:** OnToology [i.10] is an on-line application developed and maintained by OEG/UPM that exploits GitHub capabilities to ease collaborative ontology (or OWL vocabulary [i.18]) development focusing on ontology documentation (HTML, diagrams, or JSON-LD context generation), evaluation and publication (using permanent URLs). OnToology reuses existing software to carry out the above-mentioned activities orchestrating the transition between activities and centralizing the resource exchange in GitHub rather than exporting and importing files in a number of different systems. OnToology integrates existing services such as Widoco for ontology documentation, AR2DTool for diagram generation and OOPS! for ontology evaluation. For ontology publishing, OnToology provides support for publishing ontologies using w3id permanent IRLs and also for generating the content negotiation files needed to deploy an ontology in local servers. It is worth noting that OnToology does not require any installation process to be carried out by users as it is provided as an online application that can be accessed and used with a GitHub account. OnToology is available at http://ontoology.linkeddata.es/.

- **Vocol:** The VoCol system [i.11] aims at supporting collaborative vocabulary development, inspired by agile software and content development methodologies, and using Git repositories to maintain the vocabulary-related artefacts. VoCol provides support for project management, quality assurance, documentation and visualization components. It also provides a complete encapsulated framework to publish ontologies and their documentation, relying on the user to deploy them.

6 Actors and use cases

6.0 Introduction

The following list shows the different actors and the use cases that each actor could carry out through the ontology development platform. The actors are organized into the following categories: Development actors, Steering actors and Community actors.

6.1 Development actors

Developer: A developer is a member of the ontology development team who has high knowledge about ontology development and rights to modify the ontology and interact in the development cycle.

The use cases of the developer are the following:

- Create ontology
- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Update ontological requirements
- Manage requirements (accept, discard, prioritize, plan, etc.)
- Update ontology (commit)
- Generate ontology documentation: HTML, diagrams, examples, requirements
- Configure ontology publication
- Evaluate ontology (requirement testing and bad practices detection)
- Visualize existing ontologies
- Search ontology terms in existing extensions
- Plan development sprints
- Report change request
- Access ontology development status (through metrics)

Reviewer: A reviewer is a member of the ontology development team who has knowledge about ontology development and the ontology needs for a given project. This role has decision rights about what contributions can be included in the ontology.

The use cases of the reviewer are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Review and discuss contributions (in terms of change requests on the ontology artefacts)
- Approve contributions (and update the corresponding artefacts)

Validator: A validator is a member of the ontology development team who has domain knowledge about the ontology needs for a given project. This role provides new requirements to the ontology and validates whether they are satisfied or not when implemented.

The use cases of the validator are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Insert ontological requirements
- Validate ontological requirements

Domain expert: A domain expert is an expert in the domains covered by the ontology. This role does not need to be knowledgeable about ontology development.

The use cases of the domain expert are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Insert ontological requirements
- Validate ontological requirements
- Review and discuss contributions

Project leader: A project leader is the person in charge of the ontology project who carries out the project management tasks. This actor usually has experience in ontology development projects.

The use cases of the project leader are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Set up project configuration
- Manage (accept, assign, discard, etc.) issues
- Manage requirements (accept, discard, prioritize, plan, etc.)
- Plan development sprints
- Add ontology to the SAREF community portal
- Update the SAREF community portal
- Generate release of the ontology (publish ontology)

6.2 Steering actors

Steering board member: A steering board member belongs to the group of persons in charge of steering the ontology development, the community involvement and the underlying infrastructure.

The use cases of the steering board member are the following:

- Monitor project
- Access ontology development status (metrics)
- Approve project proposal
- Identify ontologies overlap
- Access ontology users list

6.3 Community actors

Ontology user: An ontology user is a potential end user of the ontology. This actor also includes software developers that will make use of the ontology within their applications.

The use cases of the ontology user are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Access user oriented documentation (tutorials, guidelines, etc.)
- Access available conformance results for the standards
- Ontology suggestion based on ontological requirements
- Search ontology terms in the ontology and its existing extensions
- Report change request
- Register as user of the ontology
- Report usage of the ontology

Contributor: A contributor is a person external to the project who is knowledgeable about the ontology domain and proposes contributions.

The use cases of the contributor are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Propose contribution
- Report change request

Interested party: An interested party is an individual related to the domain of the ontology who could be, among others, industry stakeholders, researches, domain experts, etc.

The use cases of an interested party are the following:

- Access ontology development artefacts (ontology code, documentation, tests, etc.)
- Report change request
- Access ontology users list
- Access ontology project metrics (includes contributors, number of sprints, requirements, ontology metrics, users, analytics, etc.)
- Subscription to notifications and news

Project proposer: A person interested in proposing a new ontology project.

The use cases of the project proposer are the following:

- Propose new project

7 Technical requirements

7.0 Introduction

The present clause classifies the previously-identified use cases and defines for each of the use cases the corresponding high level requirements. This enables maintaining traceability from the requirements to the use cases they support.

7.1 Ontological requirements management

The present clause lists technical requirements for ontological requirements management in Table 1.

Table 1: Technical requirements for ontological requirements management

Use case	Actors	Requirements
Insert ontological requirements	• Validator • Domain expert	The system should allow the creation and storage of ontological requirements.
Update ontological requirements	• Developer	The system should allow the modification of ontological requirements.
Validate ontological requirements	• Validator • Domain expert	The system should allow the validation of ontological requirements.
Manage ontological requirements (accept, discard, prioritize, plan, etc.)	• Developer • Project leader	The system should provide support for the ontological requirements life cycle: • Set ontological requirements status. • Prioritize ontological requirements.

7.2 Ontology implementation

The present clause lists technical requirements for ontology implementation in Table 2.

Table 2: Technical requirements for ontology implementation

Use case	Actors	Requirements
Create ontology	• Developer	The system should allow the creation and storage of ontology files.
Access ontology development artifacts (ontology code, documentation, tests, etc.)	• Developer • Reviewer • Validator • Domain expert • Project leader • Ontology user • Contributor • Interested party	The system should allow the retrieval of ontology files, ontology documentation files, tests implementations, etc.
Update ontology (commit)	• Developer	The system should allow the modification of ontology files.
Evaluate ontology (requirement testing and bad practices detection)	• Developer	The system should allow the execution of ontology evaluation tools and the execution of tests.

Use case	Actors	Requirements
Propose contribution	• Contributor	The system should allow the creation of contribution proposals which would be reviewed before being modifying the ontology files.
Review contributions	• Reviewer • Domain expert	The system should allow the visualization of contribution proposals.
Approve contributions	• Reviewer	The system should provide functionality to approve contribution proposals which would result in modifications in the ontology files.

7.3 Ontology documentation

The present clause lists technical requirements for ontology documentation in Table 3.

Table 3: Technical requirements for ontology documentation

Use case	Actors	Requirements
Generate ontology documentation: HTML, diagrams, examples, requirements	• Developer	The system should provide support for generating ontology documentation artifacts in a semi-automatic way. The system should provide support to import and store ontology documentation artifacts.
Access user oriented documentation (tutorials, guidelines, etc.)	• Ontology user	The system should allow the retrieval of user oriented documentation as tutorials, guidelines, etc.

7.4 Ontology publication

The present clause lists technical requirements for ontology publication in Table 4.

Table 4: Technical requirements for ontology publication

Use case	Actors	Requirements
Configure ontology publication	• Developer	The system should allow the definition of different ways of publication, for example by using third party permanent URL systems or by using namespaces owned by the organization.
Visualize existing ontologies	• Developer	The system should include ontology visualization tools.
Generate release (publish ontology)	• Project leader	The system should allow the generation of releases according to the publication.
Access available conformance results	• Ontology user	The system should allow the retrieval of the conformance results of the existing ontologies.

7.5 Ontology Search

The present clause lists technical requirements for ontology search in Table 5.

Table 5: Technical requirements for ontology search

Use case	Actors	Requirements
Search ontology terms in the ontology and its existing extensions	• Developer • Ontology user	The system should index and allow searching for specific ontologies or for terms within the ontologies.
Identify ontologies overlap	• Steering board	The system should include tools for detecting overlaps in the conceptualizations between ontologies.
Ontology suggestion based on ontological requirements match	• Ontology user	The system should provide mechanisms to suggest ontologies from those stored in the system according to the requirements included for a new extension.

7.6 Ontology Maintenance

The present clause lists technical requirements for ontology maintenance in Table 6.

Table 6: Technical requirements for ontology maintenance

Use case	Actors	Requirements
Report change request	• Developer • Ontology user • Contributor • Interested party	The system should allow the creation and storage of change requests (issues, new features, etc.).
Manage (accept, assign, discard, etc.) issues	• Project leader	The system should provide support for the issues life cycle: accept, assign to a person, discard, close, etc. The system should support discussions about issues.

7.7 Project Management

The present clause lists technical requirements for project management in Table 7.

Table 7: Technical requirements for project management

Use case	Actors	Requirements
Plan sprints	• Developer • Project leader	The system should provide support for the management of sprints, requirements and backlog
Access ontology development status (metrics)	• Developer • Steering board • Interested party	The system should calculate, store and retrieve ontology development metrics related to sprints, requirements, issues, development effort, etc. The system should allow the definition and retrieval of a subset of open metrics for external actors related to the ontology characteristics, the ontology development project and the ontology community metrics.
Set up project configuration	• Project leader	The system should provide support to create new projects, associate roles and members and define access rights.
Add ontology to SAREF community portal	• Project leader	The system should provide support to include an ontology to the SAREF community portal.
Update SAREF community portal	• Project leader	The system should provide support to update the SAREF community portal.
Monitor project	• Steering board	The system should provide support to monitor a project by accessing ontology development metrics and internal project development metrics. The system should allow the generation of custom dashboards to visualize metrics information.
Propose new project	• Project proposer	They system should provide functionality to propose new ontology development projects.
Approve project proposal	• Steering board	They system should provide functionality to approve or discard ontology development project proposals.

7.8 Community and social

The present clause lists technical requirements for community and social in Table 8.

Table 8: Technical requirements for community and social

Use case	Actors	Requirements
Ontology use registration	• Ontology user	The system should allow the registration of users of a given ontology.
Report usage of the ontology	• Ontology user	The system should allow to report where the ontology has been used.
Access ontology users list	• Steering board • Interested party	The system should allow the storage and retrieval of the users of a given ontology.
Subscription to notifications and news	• Interested party	They system should provide functionality to subscribe to notifications and news about ontology projects and the SAREF community portal.

8 Requirements and guidelines based on best practices

8.0 Introduction

The SAREF ontology and its different extensions are specified in a set of versioned ETSI Technical Specifications that may normatively reference to each other. They account for numerous use cases that need knowledge representation means for very different domains (e.g. Building, Energy, Environment, Smart City, Automotive).

Great effort has been made in the Semantic Web community to define best practices for the definition and the publication of ontologies. The following clauses discuss general requirements. They then discuss devise design choices to guide the development of the SAREF ontologies and the SAREF ontology portal.

The list of requirements and the solutions considered took into consideration the EUREKA ITEA2 12004 SEAS project deliverable on the SEAS Knowledge Model [i.12]

8.1 General requirements for the SAREF ontology and its extensions

Each ontology and the server that exposes them on the Web should satisfy the following best practices:

REQ 1. The ontologies should be valid OWL ontologies, and satisfy the best practices for OWL documents as defined in [i.16] , clause 3.

REQ 2. Each ontology should be versioned, using the mechanism described in [i.16], clause 3.3.

REQ 3. The ontologies should rely on the importing mechanism described in [i.16], clause 3.4, when appropriate.

REQ 4. The ontology should be accessible by looking up its IRI.

REQ 5. IRIs should not change, and should conform to the Semantic Web standard [i.13] and [i.14].

REQ 6. The set of terms defined by an ontology should be defined in the namespace of the ontology.

REQ 7. The description of a concept should be accessible by looking up its IRI [i.15], item 3.

REQ 8. The ontologies should satisfy the best practices for linked vocabularies [i.17].

REQ 9. Different representations of an ontology should be served depending on what is requested for [i.13]. It should be possible to obtain representations of the ontology using common ontology engineering tools.

REQ 10. A reference to SAREF and each of its extensions should be accessible from the SAREF portal.

8.2 Requirements for the publication of the SAREF ontology on the portal

8.2.1 Different ontology documents with different versions

Each of the SAREF ontology should be a valid OWL ontology described in an OWL document (REQ 1) and have different versions that correspond to those of the associated ETSI Technical Specification (REQ 2).

A Technical Specification should normatively reference another Technical Specification if and only if its corresponding ontology imports the other ontology (REQ 3). Otherwise, the reference should be informative. In any case, ontologies should be imported in a specific version and references should be of specific versions.

An ontology series O is identified by an IRI, and each of the ontology versions O_i are also identified by IRIs. Requirements REQ 1 and REQ 4 imply that: an ontology version O_i identified by IRI v_i in an ontology series O identified by IRI u should be accessible via the IRI v_i. Furthermore, if O_i is the latest version of the ontology series O, then it should also be accessible via IRI u.

For instance, the SAREF core ontology has two versions: V1.1.1 [i.1] and V2.1.1 [i.2], the latter being the current latest version. Each of these version should have a IRI and be served when looking up that IRI. Also, the SAREF core ontology series should have its IRI, and the version V2.1.1 should be served when looking up that IRI.

8.2.2 Namespace and IRIs

Requirement REQ 5 imposes that resource IRIs are designed with simplicity, stability and manageability in mind. Currently the SAREF ontologies have IRIs under the w3id.org domain, which redirect to the servers of the individual institutions that expose the individual ontologies. All the SAREF ontologies and SAREF terms should have a IRI under the domain saref.etsi.org, and should not change in the future.

The set of terms defined by an ontology should be defined in the namespace of the ontology (REQ 6). Namespaces are usually shortened using prefixes. SAREF ontologies may either share a common namespace, or have different namespaces. Furthermore, namespaces may end with a hash, or with a slash. The two main options that do not violated requirements REQ 4 and 6 are:

OPT 1. Common namespace that ends with a slash, for example https://saref.etsi.org/ shortened by the prefix saref:.

OPT 2. Different namespaces that end with a hash, for example https://saref.etsi.org/saref# for SAREF, shortened by the prefix saref: https://saref.etsi.org/saref4ener# for SAREF4ENER [i.3], shortened by the prefix s4ener:, etc.

The rest of this clause overviews the pros and cons of each of these options.

If a term defined in an extension (for example Deployment in SAREF4AGRI) is to be promoted to the main SAREF ontology, then using option OPT 1 is the only way its IRI can be preserved (REQ 5). However, with both OPT 1 and OPT 2 there may be human-decisions that imply a IRI change. For example, if Bulb is initially defined in its agricultural meaning, but SAREF developers then prefer to assign it the meaning of a light bulb.

If a term relevant to an ontology (for example Building in SAREF4BLDG [i.4]) is already defined in another ontology (for example SAREF), then the lengthy but rigorous lifecycle of extensions makes it too complex to dissociate Building to SAREF and include it in SAREF4BLDG with option OPT 1. On the other hand, relying on option OPT 2 leads to the definition of Building in both SAREF and SAREF4BLDG. OPT 2 allows for different extensions to define terms having the same local name but different semantics. For example, Bulb may have a different semantics in SAREF4BLDG and SAREF4AGRI.

The already existing SAREF ontology and extensions use different hash-based namespaces. Redirections will be required from the old namespace to the new namespace. The redirection can point to the new ontology document, or to a separate document that defines equivalences between the old terms and the new terms. With option OPT 2 the redirection can point to either in the transition phase. Option OPT 1 requires the redirection to point to a separate document.

With option OPT 1, SAREF extension experts need to remember or use tools to check for the pre-existence of terms in other extensions. This may lead to naming collision.

With option OPT 2, SAREF users need to remember or use tools to check in which namespace each term is defined. This may lead to interoperability issues.

With option OPT 1, one may acquire usage analytics at the granularity of terms. While only at the granularity of ontologies for the option OPT 2.

The current draft of the SAREF ontology portal uses different hash-based namespaces for SAREF (OPT 2) and each of its extensions. The namespaces and prefixes are therefore:

- The namespace for SAREF is https://saref.etsi.org/saref# with prefix saref.
- The namespace for an extension SAREF4ABCD is http://saref.etsi.org/saref4abcd# with prefix s4abcd.

8.2.3 Requirements for usability and referencing

Looking up a concept's IRI should retrieve the most recent version of the extension where this concept is defined (REQ 2, 6, 7). Each module version should be available at least in the Turtle, RDF/XML, and HTML formats, with server content negotiation, reference to a unique canonical IRI for each representation, and hint for a filename to use if the browser downloads the file.

Each of the ontology namespaces and prefixes should be registered at the well-known service http://prefix.cc/, which is used by automatic completion tools in ontology engineering software.

In order for the SAREF ontologies to be better referenced, each ontology should be registered on the Linked Open Vocabularies index http://lov.linkeddata.es/ [i.21].

8.3 Guidelines for the development of the SAREF ontology

8.3.0 Introduction

The present clause reports on best practices for associating metadata to the SAREF ontology modules and the terms they define, and about the rationale and the options to reuse other existing ontologies.

Let be the following prefixes and namespaces, defined using the RDF 1.1 Turtle syntax [i.20]:

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix vann: <http://purl.org/vocab/vann/> .
@prefix voaf: <http://purl.org/vocommons/voaf#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
```

8.3.1 Naming convention

Some convention should be adopted to name terms in the SAREF ontology. Since spaces need to be URL-encoded in IRIs and underscores take space as opposed to mixed case formats, typically language specific conventions are preferred with Java being one of the most common. The SAREF ontology and its extensions should use the following naming conventions:

- Classes with CamelCase
- Properties with mixedCase
- Individuals with CamelCase

This is compatible with what is considered as good practices in the Semantic Web [i.19]. With the generalized use of ontology patterns in SAREF, conventions should also be defined for the morphology of analogous concepts. For example, if the link between a device and the property it can measure is named `measuresProperty`, then an analogous link between a device and the property it can control can be named `controlsProperty`.

8.3.2 Metadata

8.3.2.0 Introduction

Every SAREF ontology and every concept defined in those ontologies should have a consistent set of associated metadata.

8.3.2.1 Ontology metadata

The SAREF ontologies are modularized and versioned. Following the recommendations in [i.16], the following metadata should be associated to each of the ontologies:

- type `owl:Ontology`;
- an `owl:versionIRI`, with the IRI of the version;
- an `owl:versionInfo`, with information about the version;
- potentially an `owl:priorVersion`, that points to the IRI of the previous module version;
- potentially one or more `owl:imports` that point to other ontologies (or ontology modules) to import, in their specific version.

The SAREF ontologies also use recommended metadata for linked vocabularies [i.17]:

- type `voaf:Vocabulary;`
- a `dcterms:title,` with a language tag;
- a `dcterms:description`, with a language tag;
- a `dcterms:issued`, which is a literal with datatype xsd:date;
- a `dcterms:modified`, which is a literal with datatype xsd:date;
- a `dcterms:creator`: the main contributors of the ontology version;
- zero or more `dcterms:contributor`: other contributors of the ontology version or the repository issues;
- a `dcterms:publisher`, which points to ETSI;
- a `dcterms:license`, which points to the location of the ETSI licence for the reuse of the SAREF ontologies;
- a `vann:preferredNamespacePrefix`;
- a `vann:preferredNamespaceURI.`

8.3.2.2 Term metadata

Following the recommended metadata for linked vocabularies [i.17], every term is annotated at least with the following metadata:

- a `rdfs:label`, with a language tag;
- a `rdfs:comment`, with a language tag.

8.3.3 Reusing existing ontologies

8.3.3.0 Introduction

The Semantic Web philosophy encourages the reuse of existing ontologies when appropriate. SAREF use cases may require models that are common to numerous other projects (e.g. Provenance, Time instants and intervals, Quantities and Units of Measure), or that are not actual subdomains of the IoT domain (e.g. products and offers). There exist some ontologies for some of these domains, with variable institutional statuses. It is worth recommending the use of such existing ontologies instead of developing a whole new ontology that covers every domain. Clauses 8.3.3.1 to 8.3.3.3 discuss the three main ways to reuse existing ontologies, which have different implications.

- Direct import. Should only be used to import SAREF core ontology or SAREF extensions ontologies.
- Simple reuse of terms. Should be used to reuse any other ontology, such as OWL Time.
- External alignment document. Should be used to define how other existing standards such as OGC&W3C SOSA/SSN or oneM2M can be used in combination with SAREF or one of its extension. This should only be used for Technical Specifications with alignment sections.

8.3.3.1 Direct import

The importing mechanism described in [i.16], clause 3.4, consists of using the `owl:imports` ontology annotation. When an OWL processor encounters this annotation, it looks up the IRI of the imported ontology, and loads all the entity declarations and all the axioms of this ontology. This is further recursive.

However, some existing ontologies have a great number of axioms, are inconsistent, or show a high computational complexity. Importing such ontologies would result in unpractical reasoning in the SAREF ontology. This is the case of the W3C Time ontology for instance. Directly importing the W3C Time ontology would mean that a processor conformant with the SAREF ontologies should deal with the level of complexity of the W3C Time ontology, which is not necessarily appropriate in constrained devices.

Also, the SAREF ontology indirectly endorse the external ontology by importing it directly.

For these reasons, direct import should only be used to import SAREF core ontology or SAREF extensions ontologies.

8.3.3.2 Reusing terms of existing ontologies

A more flexible mechanism to reuse an existing ontology is to declare and use the terms of another ontology without importing that ontology.

For instance, the SAREF core ontology should simply reuse some terms from the OWL Time ontology without importing it. Users would therefore have the choice to import SAREF, or both SAREF and OWL Time if they want to lead complex reasoning tasks related to time.

8.3.3.3 Alignment

Ontologies which aim is not to extend SAREF in a given domain but to provide a set of alignments with existing ontologies (for example the oneM2M base ontology) should be defined in external modules, that import both the relevant SAREF ontologies and the existing ontology. Alignments may be formalized by means of the following properties, among other:

- `owl:equivalentClass` - asserts that the two classes are equivalent;
- `owl:equivalentProperty` - asserts that the two object or datatype properties are equivalent;
- `rdfs:subClassOf` - asserts that all individuals in the first class are also individuals of the second;
- `rdfs:subPropertyOf` - asserts that if two individuals are linked through the first property, then they are also linked through the second property;
- `owl:sameAs` - asserts that two IRIs identify the same OWL named individual.

Users would therefore have the choice to import SAREF, the aligned ontology, or both ontologies with the defined alignments.

Note that these alignment mechanisms could also be used within a given SAREF extension, for example to restrict the concept Building for a particular domain like agriculture. In this case, developers should stablish an `rdfs:subClassOf` relation between the more specific concept and the concept defined in SAREF.

History

Document history		
V1.1.1	July 2019	Publication