ETSI TS 103 410-1 V2.1.1 (2024-10)

SmartM2M;
Extension to SAREF;
Part 1: Energy Domain

Reference

RTS/SmartM2M-103410-1v211

Keywords

data sharing, IoT, M2M, oneM2M,
ontology, 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 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are 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 Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, 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.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

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

The present document is part 1 of a multi-part deliverable covering SmartM2M; Extension to SAREF, as identified below:

Part 1: "Energy Domain";

Part 2: "Environment Domain";

Part 3: "Building Domain";

Part 4: "Smart Cities Domain";

Part 5: "Industry and Manufacturing Domains";

Part 6: "Smart Agriculture and Food Chain Domain";

Part 7: "Automotive Domain";

Part 8: "eHealth/Ageing-well Domain";

Part 9: "Wearables Domain";

Part 10: "Water Domain";

Part 11: "Lift Domain";

Part 12: "Smart Grid Domain";

Part 13: "Maritime Domain".

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**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 presents SAREF4ENER V2.1.1, the SAREF extension for energy.

2 References

2.1 Normative 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.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference/.

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 necessary for the application of the present document.

[1] ETSI TS 103 264: "SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping".

[2] ETSI TS 103 548: "SmartM2M; SAREF reference ontology patterns".

[3] Void.

[4] EN 50631-1:2023: "Household appliances network and grid connectivity - Part 1: General requirements, generic data modelling and neutral messages", (produced by CEN).

[5] EN 50631-2:2023: "Household appliances network and grid connectivity - Part 2: Product specific mappings, details, requirements and deviations", (produced by CEN).

[6] EN 50631-3-1:2023: "Household appliances network and grid connectivity - Part 3-1: Specific Data Model Mapping: SPINE and SPINE-IoT", (produced by CEN).

[7] EN 50631-4-1:2023: "Household appliances network and grid connectivity - Part 4-1: Communication Protocol Specific Aspects: SPINE, SPINE-IoT and SHIP", (produced by CEN).

[8] EN 50491-12-2:2022: "General requirements for Home and Building Electronic Systems (HBES) and Building Automation and Control Systems (BACS) - Part 12-2: Smart grid - Application specification - Interface and framework for customer - Interface between the Home / Building CEM and Resource manager(s) - Data model and messaging", (produced by CEN).

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] TNO, EEBus, Energy@Home: "SAREF4EE: The extension of SAREF for EEBus and Energy@Home".

[i.2] Mandate M/490 for smart grids (March 2011): "Standardization Mandate to European Standardisation Organisations (ESOs) to support European Smart Grid deployment".

[i.3] IEC TR 62746-2:2015: "Systems interface between customer energy management system and the power management system - Part 2: Use cases and requirements".

[i.4] ETSI TR 103 411: "SmartM2M Smart Appliances SAREF extension investigation".

[i.5] Mente Konsman and Ewoud Werkman in collaboration with CLC/TC 205 WG 18 members: "S2 White paper", 2023.

[i.6] S2-ws-json: "A WebSockets and JSON based protocol implementing the EN 50491-12-2 'S2' standard for home and building energy management".

[i.7] Open Geospatial Consortium: "OGC Abstract Specification Topic 20: Observations, measurements and samples", 2023.

[i.8] European Commission: "Horizon 2020 project InterConnect".

[i.9] European Commission (JRC and DG ENER): "Code of Conduct on energy management related interoperability of Energy Smart Appliances (V.1.0)", 2024.

[i.10] Daniele, L., Solanki, M., den Hartog, F., Roes, J. (2016): "Interoperability for Smart Appliances in the IoT World". In: Groth, P., et al. The Semantic Web - ISWC 2016. ISWC 2016. Lecture Notes in Computer Science, vol 9982. Springer, Cham. .

[i.11] H2020 InterConnect project Deliverable 2.3: "Interoperable and Secure Standards and Ontologies", December 2021.

[i.12] Guitart L., Damas Silva C., Liverani S., Daniele L., Rua D.: "A DSO Standard Interface to support grid management", H2020 InterConnect project white paper (2024).

[i.13] ETSI TR 103 781: "SmartM2M; Study for SAREF ontology patterns and usage guidelines".

[i.14] ETSI TS 103 673: "SmartM2M; SAREF Development Framework and Workflow, Streamlining the Development of SAREF and its Extensions".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TS 103 673 [i.14] and the following apply:

Customer Energy Manager (CEM): central component that optimizes the energy usage according to a predefined set of criteria based on the flexibility provided by a set of devices

energy flexibility: ability to optimize the energy usage based on the production and consumption of all devices in a network without a significant effect on comfort

Energy Management System (EMS): central system that optimizes the energy usage according to a predefined set of criteria based on the flexibility provided by a set of devices

energy smart device: device that can expose information about its energy flexibility, state, and/or current readings

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

resource manager: component that arranges the communication between a smart device and external actors, such as primarily the customer energy manager

3.2 Symbols

Void.

3.3 Abbreviations

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

BACS	Building Automation Control Systems
CEM	Customer Energy Manager
CENELEC	European Committee for or Electrotechnical Standardization
CLC/TC	CENELEC Technical Committee
CoC	Code of Conduct
DSO	Distribution System Operator
EC	European Commission
EMS	Energy Management System
ESA	Energy Smart Appliances
EV	Electric Vehicle
HBES	Home and Building Electronic Systems
IoT	Internet of Things
OWL	Web Ontology Language
PV	PhotoVoltaic
RDF	Resource Description Framework
RM	Resource Manager
SAREF	Smart Applications REFerence ontology
SHIP	Smart Home Internet Protocol
SPINE	Smart Premises Interoperable Neutral-message Exchange
SPINE-IoT	Smart Premises Interoperable Neutral-message Exchange for IoT
TC	Technical Committee
TNO	Netherlands Organization for Applied Scientific Research
TR	Technical Report
TS	Technical Specification
UML	Unified Modelling Language
XSD	W3C® XML Schema Definition

4 SAREF4ENER ontology and semantics

4.1 Introduction and overview

The present document is a technical specification of SAREF4ENER, an OWL-DL ontology that extends SAREF [1] for the energy domain.

The present document was created based on the CENELEC standards EN 50631:2023, parts 1 [4], 2 [5], 3-1 [6] and 4-1 [7], and EN 50491-12-2 [8], in collaboration with the Horizon 2020 project Interconnect [i.8], and with industry associations such as EEBUS(http://www.eebus.org/), Energy@Home and the S2 consortium (https://s2standard.org/#consortium), which includes KNX (https://www.knx.org/) and the Flexible power Alliance Network (FAN, https://flexible-energy.eu/).

The EN 50631 "Household appliances network and grid connectivity" series, produced by the CENELEC/Technical Committee (CLC/TC) 59X on "Performance of household and similar electrical appliances", defines the information exchange between smart appliances and management systems in homes and buildings including energy management. Part 1 of the standard (EN 50631-1 [4]) defines the data models for interoperable connected household appliances that are derived from a logical decomposition of use cases into functional blocks. Part 2 (EN 50631-2 [5]) maps the generic use cases, use case functions, and generic data definitions to categories of appliances (e.g. washer, dishwasher, water heater, HVAC devices). Part 3-1 (EN 50631-3-1 [6]) maps the generic use case functions and data models defined in Part 1 to specific languages such as SPINE and SPINE-IoT. Part 4-1 (EN 50631-4-1 [7]) defines the Communication Protocol Specific Aspects in terms of SPINE, SPINE-IoT and Smart Home IP (SHIP).

NOTE 1: SPINE and SPINE-IoT are used by manufacturers from the EEBUS association.

NOTE 2: Since it is based on mappings to the SPINE language and protocol, EN 50631:2023, parts 3-1 [6] and 4-1 [7], is often referred to directly as "SPINE". The EEBUS SPINE specification used in EN 50631 is available free of charge at https://www.eebus.org/media-downloads/.

The EN 50491 "General requirements for Home and Building Electronic Systems (HBES) and Building Automation and Control Systems (BACS)" standard is produced by the CLC/TC 205 on HBES. EN 50491-12-2 [8] specifies the fundamental aspects of interoperability for the S2 interface and the related data exchange between a CEM and the Resource Managers within the home and building premises. It defines a communication standard for energy flexibility and energy management, which helps to optimize the use of energy of smart devices in homes and buildings. Energy flexibility, which can be offered by (a combination of), for example, PV systems, EV chargers, batteries, and (hybrid) heat pumps, is the ability to alter the use of energy without a significant impact on the user's comfort. Energy flexibility plays an important role in the digital energy transition.

NOTE 3: S2 is the customer interface between the CEM in the home/building premises and the Resource Manager(s), as defined in the European Smart Grid Architecture developed in Mandate 490 of the European Commission [i.2]. Therefore, EN 50491-12-2 [8], which defines the S2 interface, is often called the "S2 standard".

NOTE 4: The S2 standard is endorsed by the KNX and FAN industry associations. Concerning its usage, the S2 standard has been recently published (2022) and it is gaining momentum in the industry among heat pump manufactures.

NOTE 5: Additional information about the S2 standard is available at https://s2standard.org/, including a white paper [i.5] and an S2 implementation in JSON [i.6].

The SAREF4ENER extension should be used to annotate (or generate) a neutral (protocol-independent) set of messages, for example by energy smart appliances manufacturers, to exchange information at the home/buildings premises among smart appliances, their Resource Manager (RM) and a Customer Energy Manager (CEM) to efficiently optimize energy consumption and production within the constraints set by the user.

NOTE 6: An alternative term used for CEM is Energy Management System (EMS).

NOTE 7: SAREF4ENER, together with SAREF core, have been adopted as common semantic model in the Code of Conduct (CoC) for Energy Smart Appliances (ESA) manufactures [i.9] promoted by the European Commission [i.8] (EC). The CoC has the goal to increase the number of interoperable ESA that are placed on the European Union market. The current CoC V1.0 has been officially launched by the EC in April 2024 and 10 manufacturers producing appliances have signed this first version, namely Arçelik, Clivet, Daikin, Electrolux, Miele, Mitsubishi Electric, Panasonic, Vaillant Group, Vestel and Viessmann. The manufacturers have committed to develop interoperable connected products within a year. In addition, a Home Energy Management System manufacturer, GEO, has committed to support compliant ESA through their products. This first CoC version includes examples of mappings from EN 50631 (SPINE) to SAREF4ENER V1.2.1. Future releases of the CoC intend to include also examples of mappings from EN 50491-12-2 [8] (S2) to SAREF4ENER. SAREF4ENER V1.1.1 was primarily based on the power profiles as defined in EN 50631:2023 parts 1 [4], 2 [5], 3-1 [6] and 4-1 [7] (SPINE). The subsequent version, SAREF4ENER V1.2.1, added new SPINE concepts not previously covered and introduced the most important concepts from EN 50491-12-2 [8] (S2), with the explicit goal of harmonizing and providing interoperability between the SPINE and S2 standards. In the meantime, a new version of SAREF core, V4.1.1, was released. Therefore, the present document, SAREF4ENER V2.1.1, aligns to the latest SAREF core release 4.1.1, yet preserving the important changes made in SAREF4ENER V1.2.1. The history of the main changes in SAREF4ENER can be found in Annex D.

The application of SAREF4ENER focuses on demand response scenarios, in which customers can offer energy flexibility to the Smart Home and Smart Grid. Energy smart devices and energy managers communicate with each other to achieve the best possible result. Energy smart devices can express their demand/production and flexibility, energy managers are responsible to find the most optimal measure between energy consumption and energy production of energy smart devices based on the customer's chosen configuration and the characteristics of the devices. Next to self-consumption optimization, the Smart Grid can influence the quantity or patterns of use of the energy consumed by customers when grid-energy-supply systems are constrained, e.g. during peak hours.

This can be realized by connecting a smart home device with an Energy Management System (EMS) (see EN 50631:2023, parts 1 [4], 2 [5], 3-1 [6] and 4-1 [7]) or by means of a Resource Manager (RM) (see EN 50491-12-2 [8]).These scenarios involve (but are not limited to) the following use cases. The SAREF4ENER parts applicable per use case are primarily decided by the types of devices that are involved:

- **Use case 1:** flexible start of smart appliances. Smart energy management should be able to (re-)schedule appliances in certain modes and preferred times using power profiles to optimize energy efficiency and accommodate the customer's preferences. The user should be able to decide on a preferred interval within which the energy manager computes the starting time that optimizes the energy usage. Interruption options, such as pausing a task, can further optimize the energy usage.

- **Use case 2:** monitoring and control of the start, status, and power consumption of the appliances. It is essential for an energy manager to be aware of the power consumption of all devices it optimizes for, including devices that are not smart.

- **Use case 3:** reaction to special requests from the Smart Grid, for example, incentives to consume more or less depending on current energy availability, or emergency situations that require temporary reduction of the power consumption.

- **Use case 4:** limitation of power consumption. This use case covers power limits that are sent by the energy manager, as well as power limits set by the manufacturer in the case of a lost connection (fail-safe limits), as well as contractual and nominal power limits.

- **Use case 5:** incentive table. This use case aims to influence the energy usage via a set of incentives that the energy consumer and energy manager negotiate about.

- **Use case 6:** description of the energy flexibility capabilities of any type of device in a (smart) home/building in the information exchange of such devices with resource and energy managers. Energy flexibility, which can be offered by (a combination of), for example, PV systems, EV chargers, batteries, and (hybrid) heat pumps, is the ability to alter the use of energy without a significant impact on the user's comfort [i.5].

- **Use case 7:** Interoperable data exchange between a Distribution System Operator (DSO) and other market parties, such as services providers and flexibility aggregators to support:

    i) network flexibility that can be used by network operators to manage the electricity grids more efficiently and to provide a potential additional revenue stream to the consumers without limiting their consumer behaviour patterns [i.12]; and

    ii) increased grid observability for home appliances that are mostly passive and not actively monitored, enabling a faster identification of potential outages and electricity quality issues [i.12].

These use cases are associated with the user stories described in [i.3], which include, among others, the following examples:

- User wants to do basic settings of his/her devices;

- User wants to know when the washing machine has finished working;

- User wants the washing done by 5:00 p.m. with least electrical power costs;

- User likes to limit his/her own energy consumption up to a defined limit;

- User allows the EMS to reduce the energy consumption of his/her freezer in a defined range for a specific time, if the grid recognizes (severe) stability issues;

- Grid related emergency situations (blackout prevention).

The prefixes and namespaces used in SAREF4ENER and in the present document are listed in Table 1.

Table 1: Prefixes and namespaces used throughout the present document

Prefix	Namespace
s4ener	https://saref.etsi.org/saref4ener/
saref	https://saref.etsi.org/core/
dcterms	http://purl.org/dc/terms/
foaf	http://xmlns.com/foaf/0.1/
owl	http://www.w3.org/2002/07/owl#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
om	http://www.ontology-of-units-of-measure.org/resource/om-2/
qudt	http://qudt.org/vocab/unit/
xsd	http://www.w3.org/2001/XMLSchema#
time	http://www.w3.org/2006/time#

4.2 SAREF4ENER

4.2.1 General Overview

The main addition that SAREF4ENER provides on top of SAREF Core is a set of `saref:Profiles` that describe the energy flexibility capabilities of a device (see clause 4.2.3). These profiles are defined according to the SPINE/SPINE IoT [6] and the S2 [8] data models, with some profiles occurring in both [6] and [8], while some other profiles occurring either in [6] or [8]. For example, the Power Profile flexibility type is described in both S2 and SPINE/IoT, thus is merged into a single representation in SAREF4ENER (see clause 4.2.3.1). The Power Envelope defined in S2 and Power Limits defined in SPINE also present similarities which are therefore specified in SAREF4ENER using several shared concepts (see clause 4.2.3.6). The remaining types of flexibility are unique to either S2 or SPINE, namely, Incentive Tables are defined only in SPINE, whereas Operation Mode, Fill Rate Based, and Demand Driven energy flexibility are control types defined only in S2 [i.5].

The SAREF4ENER extension additionally describes flexibility instructions (see clause 4.2.5) separately from the flexibility profiles. These instructions describe the communication taking place between a device and the EMS to decide on the energy flexibility plan, such as offers from the device and requests from an EMS. A real-time check on the monitoring of power consumption is facilitated via the reuse of the main SAREF module and the load control use case (see clause 4.2.4). Finally, the SAREF4ENER extension provides a modelling approach for data points and time series (see clause 4.2.6), which is necessary for modelling the various forecasts and data elements involved.

An overview of the SAREF4ENER (V2.1.1) ontology is provided in Figure 1. In the image, classes are represented as rectangles. Relationships (object properties) between entities are represented as arrows. Arrows are additionally used to represent some RDF, RDF-S and OWL constructs, more precisely: plain arrows with white triangles represent the rdfs:subClassOf relation between two classes. The origin of the arrow shall be considered as the subclass of the entity at the destination of the arrow. Dashed arrows accompanied by the expression rdf:type are used to indicate that the individual at the origin of the arrow is an instance of the class placed at the end of the arrow. Datatype properties and class restrictions are presented as plain text and positioned within the boxes of the rectangles. The green colour is used to distinguish SAREF core entities. The blue colour is used for highlighting the classes and properties already existing in the previous version of SAREF4ENER (V1.1.2). The white colour is used to denote the classes and properties that have been added in the SAREF4ENER version specified in the present document (V2.1.1). Note that Figure 1 aims at showing a global overview of the main classes of SAREF4ENER and their mutual relations. More details on the different parts of Figure 1 are provided in the rest of the present document .

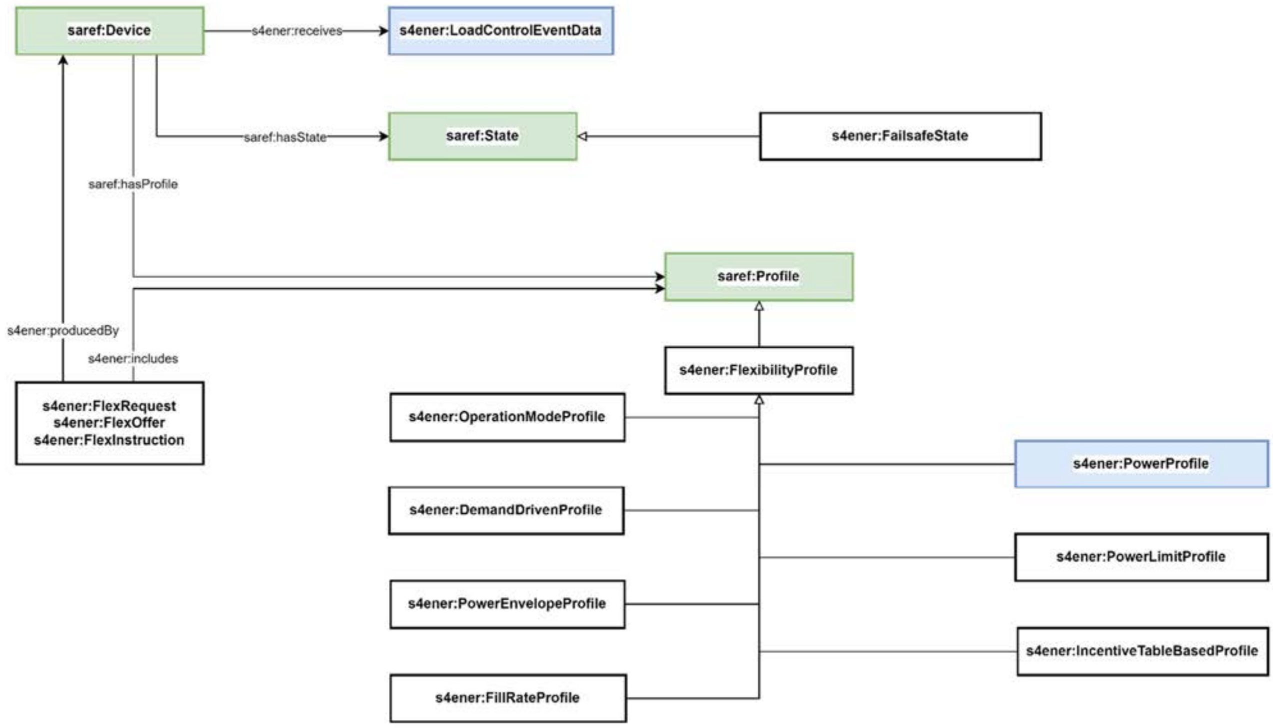

Figure 1: SAREF4ENER overview

4.2.2 Device

This extension adds several properties to the existing `saref:Device` which may be used to describe additional device details to the basic properties already defined in SAREF core.

Table 2: Properties of a Device

Property	Definition
s4ener:receives	A relationship between a device (e.g. an appliance or a smart meter) and a load control event.
s4ener:brandName	The name of the brand of a device. Useful where the name of the brand and the vendor differs.
s4ener:deviceCode	Device code for the device as defined by the manufacturer.
s4ener:deviceName	Name of the device as defined by the manufacturer.
s4ener:hardwareRevision	Hardware revision of the device as defined by the manufacturer.
s4ener:manufacturerDescription	A description for the device as defined by the manufacturer.
s4ener:manufacturerLabel	A short label of the device as defined by the manufacturer.
s4ener:manufacturerNodeIdentification	A node identification for the device as defined by the manufacturer. This could be used for the identification of a device, even if it was removed from the network and re-joined later with changed node address.
s4ener:powerSource	The power source of a device. Possible values are s4ener:MainsSinglePhase, s4ener:Mains3Phase, s4ener:Battery, and s4ener:DC.
s4ener:serialNumber	Serial number of a device as defined by the manufacturer. Usually the same as printed on the case.
s4ener:softwareRevision	Software revision of a device as defined by the manufacturer.
s4ener:vendorCode	Code for the vendor of the device as defined by the manufacturer.
s4ener:vendorName	Name of the vendor of the device as defined by the manufacturer.

4.2.3 Flexibility Profile

4.2.3.0 Foreword

The SAREF4ENER extension defines different energy flexibility profiles that can be offered by a `saref:Device`. They are: `s4ener:PowerProfile`, `s4ener:PowerLimitProfile`, `s4ener:DemandDrivenProfile`, `s4ener:OperationModeProfile`, `s4ener:FillRateBasedProfile`, `s4ener:IncentiveTableBasedProfile`, and `s4ener:PowerEnvelopeProfile`. They are all subclasses of `s4ener:FlexibilityProfile` which is in turn a subclass of `saref:Profile`.

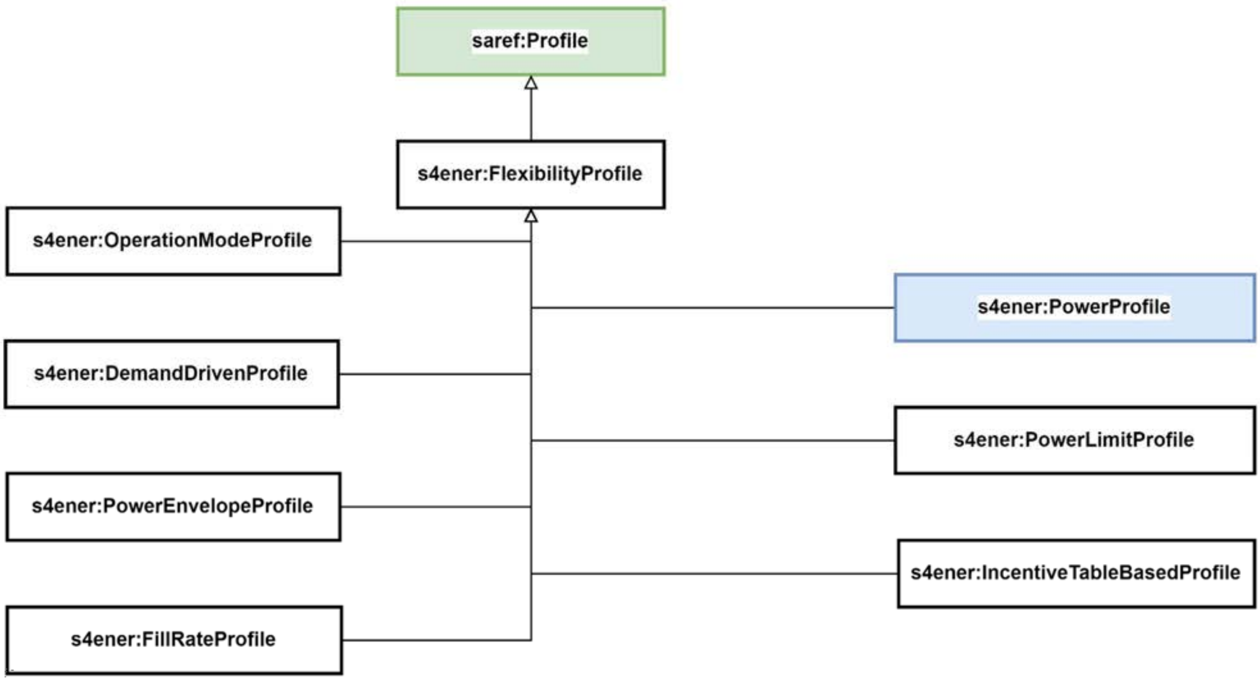

Figure 2: SAREF4ENER Flexibility Profiles

4.2.3.1 Power Profile

A `s4ener:PowerProfile` describes the power usage of a particular task of a device that can be known or predicted beforehand, such as in the case of white goods like a washing machine. The `s4ener:PowerProfile` is used by a `saref:Device` to expose the power sequences that are potentially relevant for the energy manager. A `saref:Device` can expose a `s4ener:PowerProfile`, which consists of one or more alternative plans (`s4ener:AlternativesGroup` class). A `s4ener:AlternativesGroup` consists of one or more power sequences (`s4ener:PowerSequence` class), and a `s4ener:PowerSequence` consists of one or more slots (`s4ener:Slot` class). Inversely, a `s4ener:Slot` belongs to only and exactly one `s4ener:PowerSequence`, which, in turn, belongs to only and exactly one `s4ener:AlternativesGroup`, which, in turn, belongs to only and exactly one `s4ener:PowerProfile`. A `s4ener:PowerProfile` belongs to only and exactly one `saref:Device`.

The `s4ener:AlternativesGroup` consists of one or more power sequences (`s4ener:PowerSequence`) and, inversely, a `s4ener:PowerSequence` belongs to only and exactly one `s4ener:AlternativesGroup`. The `s4ener:PowerSequence` consists of one or more slots (`s4ener:Slot`) and, inversely, a `s4ener:Slot` belongs to only and exactly one `s4ener:PowerSequence`.

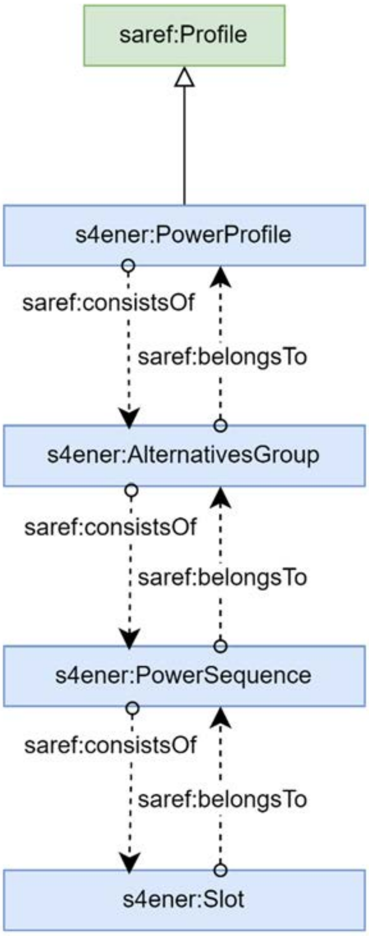

Figure 3: Power Profile Overview

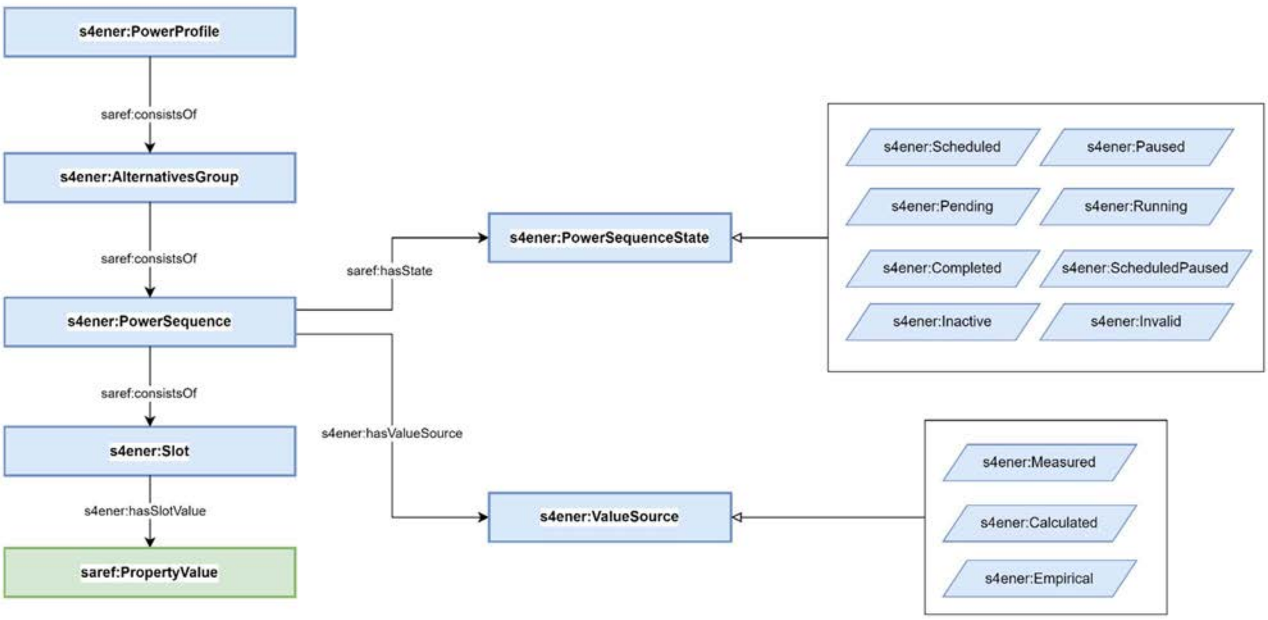

Figure 4: Power Profile and Power Sequence

Table 3: Properties of a Power Profile and an AlternativesGroup

Property	Definition
s4ener:alternativesCount	Number of "alternatives" groups provided by a power profile.
s4ener:nodeRemoteControllable	Whether the device is configured for remote control by the EMS. This refers to the selection chosen by the user on the remote control feature of the device.
s4ener:supportsReselection	Whether the device restricts the number of sequence re-selections by the EMS. If set to TRUE, there is no restriction, i.e. within a given alternative the EMS may first choose one sequence, alter the selection by configuring another sequence later on, then alter the selection again, etc. If set to FALSE, the device permits the EMS to select a sequence of an alternative only one time.
s4ener:supportsSingleSlotSchedulingOnly	Whether the device permits the modification of more than one slot per configuration command. If set to TRUE the device does NOT permit this modification.
s4ener:totalSequencesCountMax	Total number of sequences supported by the device, i.e. the sum of all power sequences across all alternatives.

Table 4: Properties of the PowerSequence

Property	Definition
s4ener:isStoppable	If the power sequence is stoppable by the EMS, this element is TRUE. Otherwise it SHALL be omitted.
s4ener:isPausable	If the power sequence is pausable by the EMS, this element is TRUE. Otherwise it SHALL be omitted.
s4ener:taskIdentifier	Used by a device that wants to uniquely identify reoccurring types of power sequences. For example, specific types of washing cycles with specific parameters SHOULD have the same s4ener:taskIdentifier value every time they are offered using power sequences.
s4ener:activeRepetitionNumber	The current repetition of the sequence of slots. SHALL be present if s4ener:repetitionsTotal is present and has a value > 1. Otherwise, it SHALL be absent.
s4ener:activeSlotNumber	If s4ener:PowerSequenceState is set to "running" or "paused" this element SHALL contain the currently active slot. Otherwise it SHALL be omitted.
s4ener:cheapest	If present and set to TRUE, the SHALL try to apply a configuration that minimizes the user's energy bill for this power sequence. Absence of this element is equal to the presence with value FALSE.
s4ener:greenest	If present and set to TRUE, the SHALL try to optimize the configuration towards the maximum availability of renewable energy. Absence of this element is equal to the presence with value FALSE.
s4ener:maxCyclesPerDay	The maximum amount of starts that the device allows per day.
s4ener:repetitionsTotal	If a power sequence repeats its sequence of slots, the element is present and contains the total number of repetitions. Absence of the element is equal to a presence with a value of 0 (zero). SHALL be absent if the value is 1.
s4ener:sequenceRemoteControllable	Whether the sequence is modifiable (if value is TRUE) or not (if value is FALSE). Modifiability is required to configure power sequences and slots. It is also required to change a power sequence state.
s4ener:valueSource	The source (origin/foundation) of the forecasted values for this power sequence. If absent, the source is undefined.
s4ener:hasEnergy	The additional energy the device will consume before resuming its normal operation (after a pause). This is only an estimated value which will not be added to the value stated in any slot value information.
saref:hasPrice	The additional costs for the resumption of a device to its normal operation (after a pause).
saref:hasState	The current state of the power sequence. It can assume one of the following values: s4ener:Running, s4ener:Paused, s4ener:Scheduled, ss4ener:ScheduledPaused, s4ener:Pending, s4ener:Inactive, s4ener:Completed or s4ener:Invalid.
s4ener:hasActiveDurationMax	The active maximum duration the power sequence can run without interruption.

Property	Definition
s4ener:hasActiveDurationMin	The active minimum duration the power sequence can run without interruption.
s4ener:hasActiveDurationSumMax	The active maximum duration the power sequence can run in total (summation of all active times).
s4ener:hasActiveDurationSumMin	The active minimum duration the power sequence runs in total (summation of all active times).
s4ener:hasStartTime	The start time of the power sequence. SHALL be present.
s4ener:hasEarliestStartTime	SHALL state the earliest possible start time for the whole power sequence.
s4ener:hasEndTime	The end time of the power sequence. If the value is available, it SHALL be denoted here. Otherwise the element SHALL be omitted.
s4ener:hasLatestEndTime	The latest possible end time for the whole power sequence.
s4ener:hasElapsedSlotTime	If the power sequence state is set to 'running' or 'paused' AND the slot is determined, this element CAN contain the time the slot has already been in 'running' state (this also means the value remains constant during a 'paused' state). Otherwise it SHALL be omitted.
s4ener:hasRemainingSlotTime	If the power sequence state is set to 'running' or 'paused' AND the slot is determined, this element SHALL contain the time the slot still needs to be in 'running' state (this also means the value remains constant during a 'paused' state). Otherwise it SHALL be omitted.
s4ener:hasPauseDurationMax	The maximum duration the power sequence can pause after the end of an activity.
s4ener:hasPauseDurationMin	The minimum duration the power sequence can pause after the end of an activity.

Table 5: Properties of a Slot

Property	Definition
s4ener:optionalSlot	It is set to TRUE if the slot can be omitted, otherwise the element SHALL be omitted or set to FALSE (see note 1).
s4ener:slotActivated	If the slot is optional, i.e. s4ener:optionalSlot is set to TRUE, this element reflects the current status of the slot (TRUE = the slot will be executed, FALSE = the slot will not be executed). If the slot is not optional, this element SHALL be absent.
s4ener:hasValueType	The type of energy or power (subclasses of saref:Energy and saref:Power). The energy can be of type s4ener:EnergyMin, s4ener:EnergyMax, s4ener:EnergyExpected, s4ener:EnergyStandardDeviation or s4ener:EnergySkewness. The power can be of type s4ener:PowerMin, s4ener:PowerMax, s4ener:PowerExpected, s4ener:PowerStandardDeviation or s4ener: Power Skewness.
s4ener:hasDefaultDuration	The duration of the slot (in case of 'determined slot'). If the slot has a configurable length, this element SHALL reflect the currently configured length.
s4ener:hasMaxDuration	The maximum supported configuration (if the slot has a configurable duration).
s4ener:hasMinDuration	The minimum supported configuration (if the slot has a configurable duration) (see note 2).
s4ener:hasDurationUncertainty	The uncertainty of the duration given in the s4ener:defaultDuration property.
s4ener:hasStartTime	The start time of the slot. SHALL be present.
s4ener:hasEarliestStartTime	SHALL state the earliest possible start time for the slot.
s4ener:hasEndTime	The end time of the slot. The following equation SHALL apply: endTime - startTime = defaultDuration.
s4ener:hasLatestEndTime	The latest possible end time for the slot.
s4ener:hasRemainingPauseTime	The duration that the current slot permits being paused. This element SHALL ONLY be present if the power sequence is interruptible (pausable), i.e. saref:isInterrupionPossible has value TRUE.
NOTE 1: This element applies to every repetition of the slot number. NOTE 2: This element applies to the first repetition of the slot number only.	

4.2.3.2 Demand Driven Profile

The `s4ener:DemandDrivenProfile` can be used for devices that can consume different types of energy resources such as electricity or natural gas, but that lack a way of buffering that energy. This may for example be a hybrid heat pump that is powered using either electricity of gas. The power demand is determined by the device, but the customer energy manager can choose how to generate that power.

The profile contains a set of `saref:Actuators` that describe the various ways that the demanded energy can be provided. These actuators may be (part of) the actual `saref:Device` that offers this profile. The forecast of the average demand rate (i.e. the amount of energy, heat, and any other resource that needs to be produced by a device in the near future) can be expressed by defining time series (`s4ener:TimeSeries`).

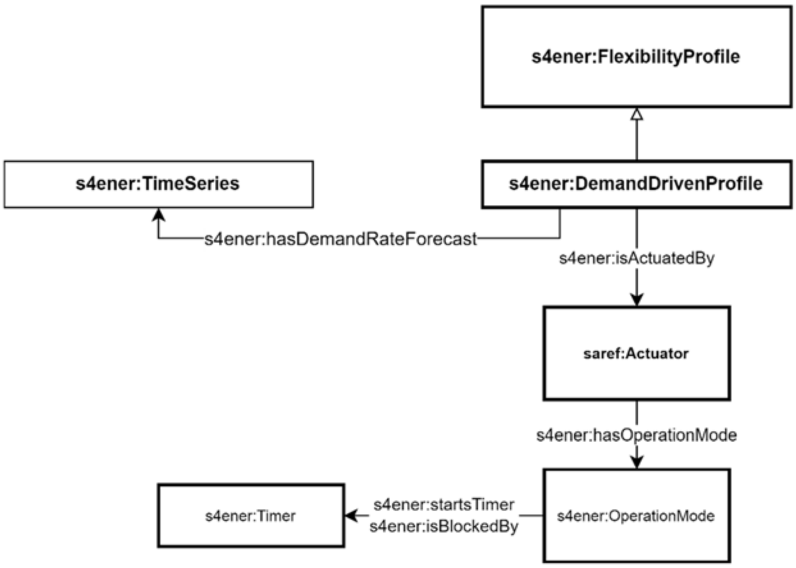

Figure 5: Demand Driven Profile

Table 6: Property of Demand Driven Profile

Property	Definition
s4ener:hasDemandRateForecast	The relationship between the demand driven profile and the time series that indicates the forecasted average demand rate.
s4ener:isActuatedBy	A reference to an (external) actuator that can activate this profile.
s4ener:hasDemandRate	The present demand rate that needs to be satisfied by the device.
s4ener:hasEarliestStartTime	The moment from which the profile is valid.

Table 7: Actuator of a Demand Driven Profile

Property	Definition
s4ener:supportsCommodity	A reference to all commodities by this actuator.
s4ener:hasOperationMode	This property indicates the `s4flex:OperationModes` that can be used by this actuator.
s4ener:hasTransition	The transitions between various `s4flex:OperationModes` that the Actuator can support.
s4ener:hasTimer	The set of timers that are available in this actuator.
s4ener:hasActiveOperationMode	A reference to the Operation Mode that is presently active.
s4ener:hasOperationModeFactor	The number indicates the factor with which the actuator is configured.
s4ener:hasPreviousOperationMode	The previous operation mode this actuator was in.
s4ener:hasTransitionTimestamp	Time at which the transition from the previous operation mode was initiated.

Table 8: Operation Mode of a Demand Drive Profile

Property	Definition
s4ener:hasPowerRange	The range of power that can be produced or consumed via this operation mode. The start of the range is associated with operation mode factor 0, the end of the range is associated with operation mode factor 1.
s4ener:hasSupplyRange	The Supply Range this operation mode can deliver. The start of the range is associated with operation mode factor 0, the end of the range is associated with operation mode factor 1.
s4ener:hasRunningCosts	Additional costs per second associated with this operation mode.
s4ener:abnormalConditionOnly	Indicates if this element can only be used during an abnormal condition.

4.2.3.3 Fill Rate Based Profile

The `s4ener:FillRateBasedProfile` can be used for devices that can store energy (`s4ener:Storage`), such as heat pumps with a buffer, EVs, batteries, and even fridges and freezers. The `saref:Actuators` associated with this fill rate based profile can consume energy to fill the buffer. The information regarding the leakage behaviour of the storage and its fill level (i.e. a measure expressing how full the storage is) can respectively be defined through the classes `s4ener:LeakageBehaviour` and `saref:Observation` via the properties `s4ener:hasLeakageBehaviour` and `s4ener:presentFillLevel`, respectively. The `s4ener:LeakageBehaviour` is always associated with an *element* detailing the leakage behaviour of the storage (`s4ener:LeakageBehaviourElement`). Ultimately, certain storage devices might have a fill-level target profile (`s4ener:FillLevelTargetProfile`) with its associated `s4ener:FillLevelTargetProfileElement`.

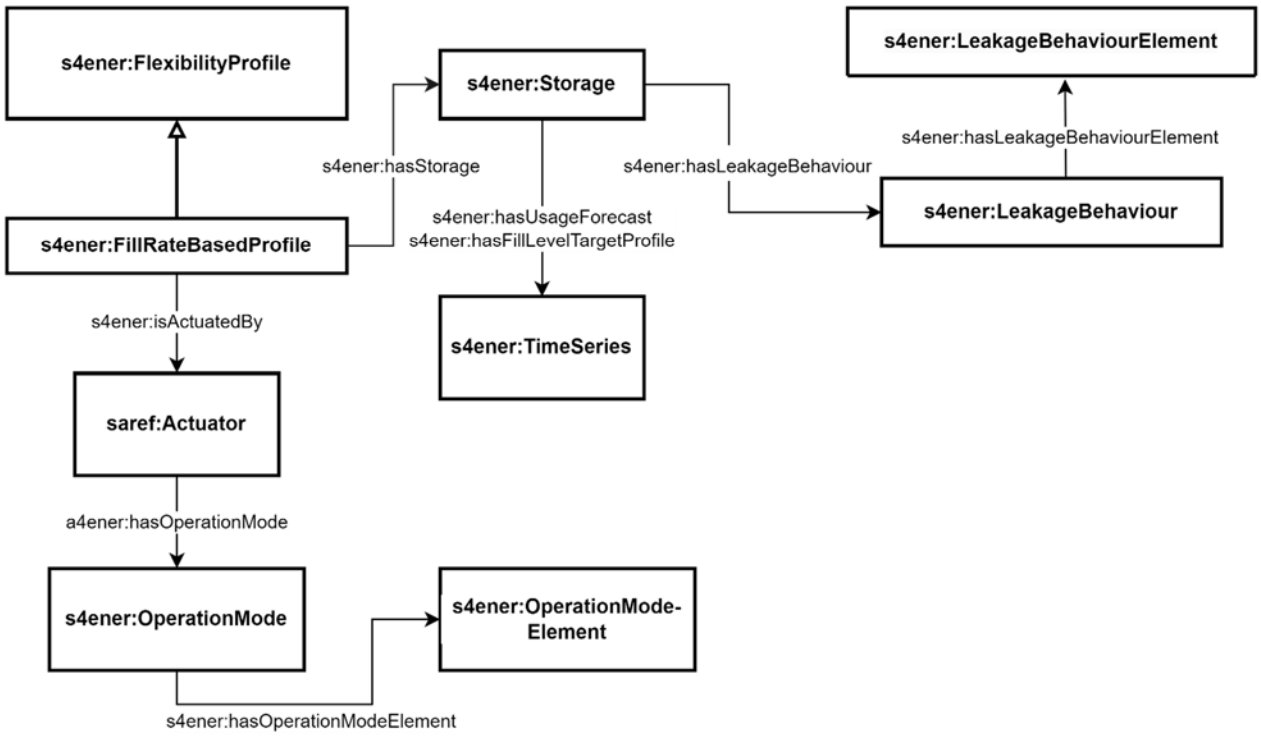

Figure 6: Fill Rate Based Profile

Table 9: Properties of Fill Rate Based Profile

Property	Definition
s4ener:hasStorage	The relationship between the fill rate based profile and the storage that describes energy storage details.
s4ener:isActuatedBy	A reference to an actuator that can activates this profile.
s4ener:hasEarliestStartTime	The moment this fill rate based profile becomes valid.

Table 10: Properties of Storage

Property	Definition
s4ener:hasLeakageBehaviour	The relationship between the storage and its associated leakage behaviour.
s4ener:presentFillLevel	The property that connects the storage to a data point with a percentage value indicating the storage fill level.
s4ener:hasFillLevelTargetProfile	The property that connects the fill rate based profile to the current fill level target profile, represented as a timeseries, that the profile should accommodate for.
s4ener:hasFillLevelRange	The range in which the fill level of the storage should remain.
s4ener:hasUsageForecast	Indicates a timeseries containing the usage forecast for this fill rate based profile.

Table 11: Properties of Leakage Behaviour

Property	Definition
s4ener:hasLeakageBehaviourElement	The property that relates the leakage behaviour to the leakage behaviour element(s).
s4ener:hasStartTime	The moment from which this leakage behaviour is valid.

Table 12: Properties of Leakage Behaviour Element

Property	Definition
s4ener:hasFillLevelRange	The property that connects the leakage behaviour element with the range for which this leakage behaviour is applicable.
s4ener:leakageRate	Indicates how fast the fill level decreases in this particular range.

The Actuator of a Fill Rate Based Profile is identical to Table 7.

Table 13: Operation Mode of a Fill Rate Based Profile Actuator

Property	Definition
s4ener:hasOperationModeElement	References to the Operation Mode Elements contained within this Operation Mode.
s4ener:abnormalConditionOnly	Indicates if this element can only be used during an abnormal condition.

Table 14: Operation Mode Element of a Fill Rate Base Profile Operation Mode

Property	Definition
s4ener:hasFillLevelRange	The range of the fill level for which this Operation Mode Element applies.
s4ener:fillRate	Indicates the change in fill level per second. The lower_boundary of the Power Range is associated with an operation mode factor of 0, the upper boundary is associated with an operation mode factor of 1.
s4ener:hasPowerRange	The power produced or consumed by this operation mode.
s4ener:hasRunningCosts	Additional costs per second (e.g. wear, services) associated with this operation mode.

4.2.3.4 Incentive Table Based Profile

The `s4ener:IncentiveTableBasedProfile` can be used to describe an incentive table, compiled of incentive table slots (`s4ener:IncentiveTableSlot`) as well as a power plan (`s4ener:PowerPlan`). Both are used to negotiate the allocation of upcoming energy usage of a device between the energy manager and the device. The incentive table is used by the energy manager to express the availability of energy via real and/or artificial incentives or costs over time. The device itself uses the table to negotiate the own demand and request the allocation by sending the resulting power plan to the energy manager.

Incentive types can be expressed in the form of relative costs (`s4ener:RelativeCost`), absolute costs (`s4ener:AbsoluteCost`), CO_2 emissions (`s4ener:CO2Emission`), and renewable energy percentage (`s4ener:RenewableEnergyPercentage`). An incentive table also defines a scope type (`s4ener:ScopeType`) to indicate whether it is a preliminary (`s4ener:Preliminary)` or committed version (`s4ener:Committed`).

An incentive table consists of a number of slots (`s4ener:IncentiveTableSlot`) where each slot may contain a series of incentives (`s4ener:Incentive`) representing various tiers (`s4ener:Tier`). Each tier may be linked to a particular energy source, such as the grid, solar panels, or surplus power. Each incentive describes the cost, expressed as a unit applicable to the `s4ener:IncentiveType`, for that power source in the particular (time) slot. The lower and optional upper boundary (`s4ener:DataPoint`) describe for each incentive at which level of power consumption it becomes applicable.

The power plan of a device is defined by a series of sets of data points (`s4ener:TimeSeries`). Each set of data points contains a time interval (`time:Interval`), a relation to a property (`s4ener:Power`), a binding to a minimum (`s4ener:Minimum`), average (`s4ener:Average`) or maximum (`s4ener:Maximum`) value and the value itself (`saref:Observation`). Finally, it also contains a scope type (`s4ener:ScopeType`) to indicate whether it is a preliminary (`s4ener:Preliminary`) or committed value (`s4ener:Committed`).

An incentive table based profile can be used with any type of device.

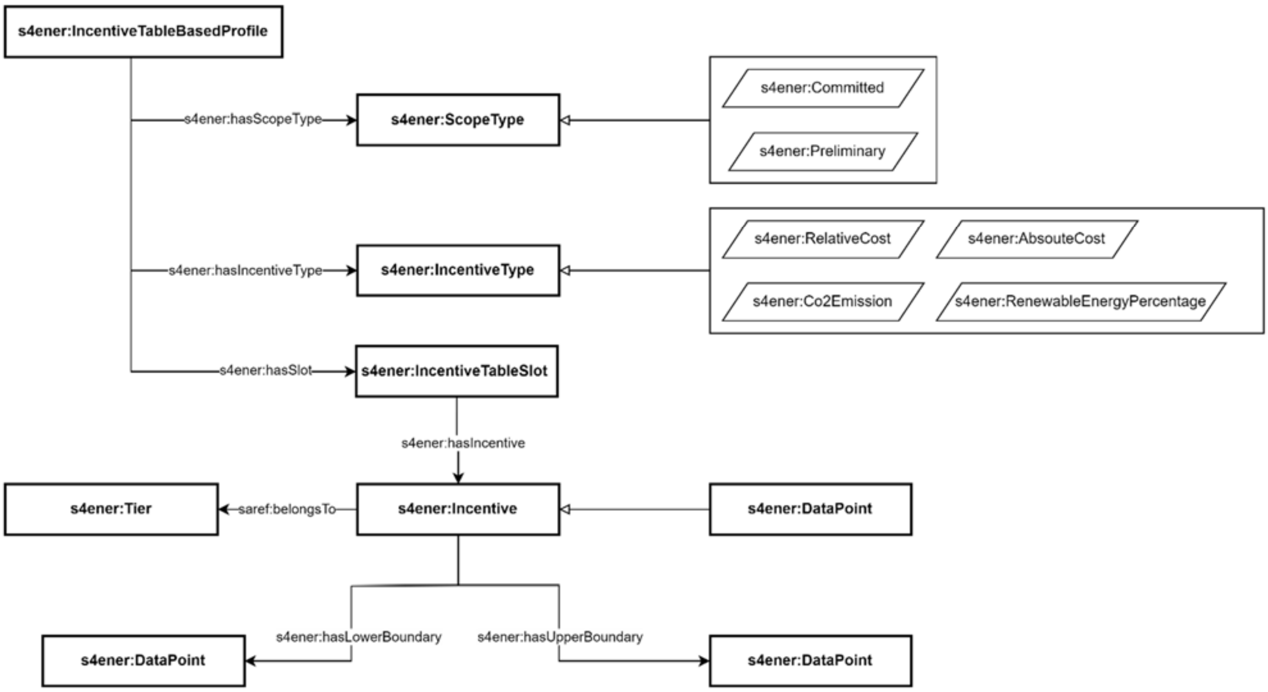

Figure 7: Incentive Table Based Profile

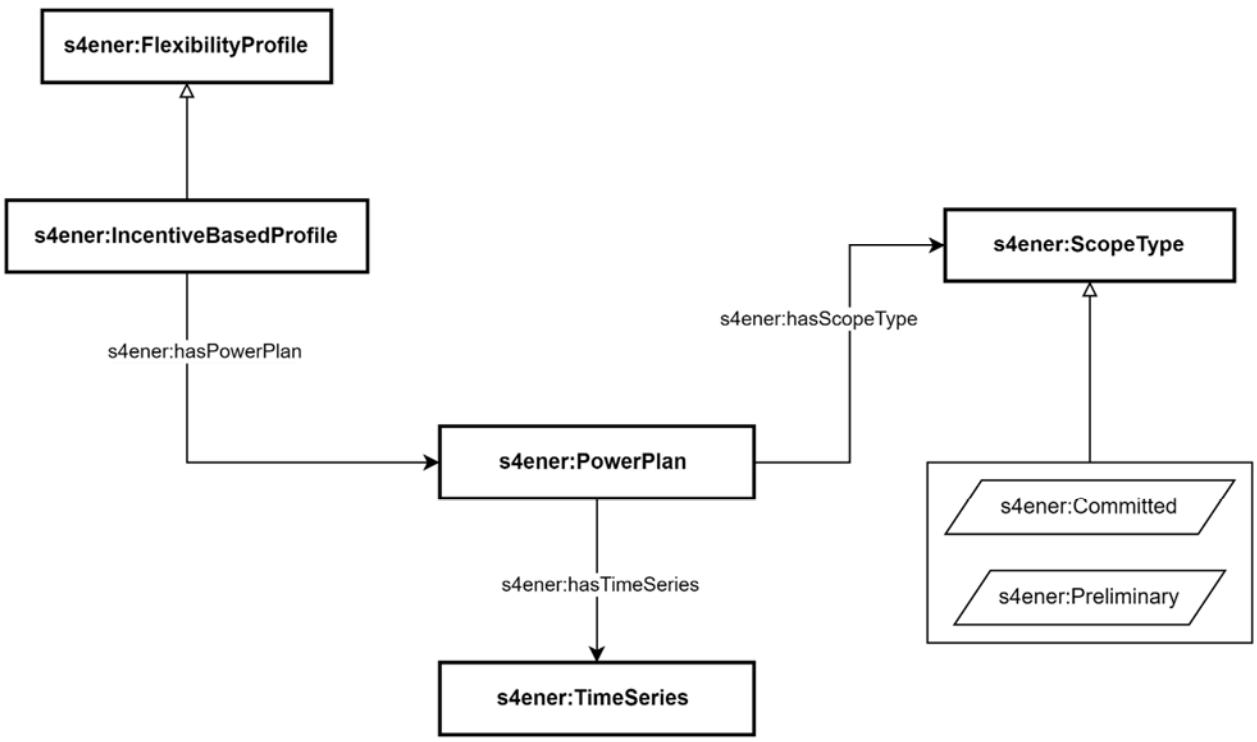

Figure 8: Power Plan associated with an Incentive Table

Table 15: Properties of Incentive Table Based Profile

Property	Definition
s4ener:isChangeable	Boolean indicating whether this incentive table is fixed or can be changed (see note).
s4ener:requiresUpdate	Boolean indicating whether the incentive table requires an update (see note).
s4ener:hasSlot	The slots that this incentive table consists of.
s4ener:hasIncentiveType	A reference to the Incentive Type like relative or absolute costs, CO_2 emission or Renewables percentage.
s4ener:hasScopeType	The scope type indicates whether the incentive table is preliminary or committed.
s4ener:hasPowerPlan	A reference to a power plan indicating the preliminary or committed usage of the energy smart device for a specific period.
NOTE: The properties `s4ener:isChangeable` and `s4ener:requiresUpdate` can be applied on any incentive table element of clause 4.2.3.4.	

Table 16: Properties of Incentive Table Slot

Property	Definition
s4ener:hasIncentive	The incentive indicates the tier, boundary, value, and type of value per incentive.
s4ener:hasEffectivePeriod	The time interval of the incentive slot.

Table 17: Properties of Incentive

Property	Definition
saref:belongsTo	Reference to the tier.
saref:isMeasuredIn	The unit of measure that is applicable for this value, which may be `s4ener:EuroPerKilowattHour`.
saref:hasValue	The incentive value for this boundary.
s4ener:hasLowerBoundary	A reference to an `s4ener:DataPoint` indicating the lower boundary for this particular boundary. The value and unit of measure of the lower boundary are found in the data point object.
s4ener:hasUpperBoundary	An optional reference to an `s4ener:DataPoint` indicating the upper boundary for this particular boundary. If omitted, the value of the next lower boundary is assumed.

Table 18: Properties of Power Plan

Property	Definition
s4ener:hasTimeSeries	The TimeSeries that a Power Plan consists of. There usually are three, namely one with usage of minimum values, one for expected values, and one for maximum values.
s4ener:isWritable	This Boolean value indicates whether this power plan is writable or fixed.

4.2.3.5 Operation Mode Profile

Devices that offer the `s4ener:operationModeProfile` can control the amount of power that they generate and/or consume, such as diesel generators and variable electrical resistors. The states in which devices fall in, such as "running at reduced power" or "running at full power", can be described as operation modes (`s4ener:OperationMode`). These operation modes have therefore been modelled as subclasses of `saref:State`. Transitions between operation modes can be defined as `s4ener:Transition` with associated timers (`s4ener:Timer`) that specify the minimum duration of a particular operation model.

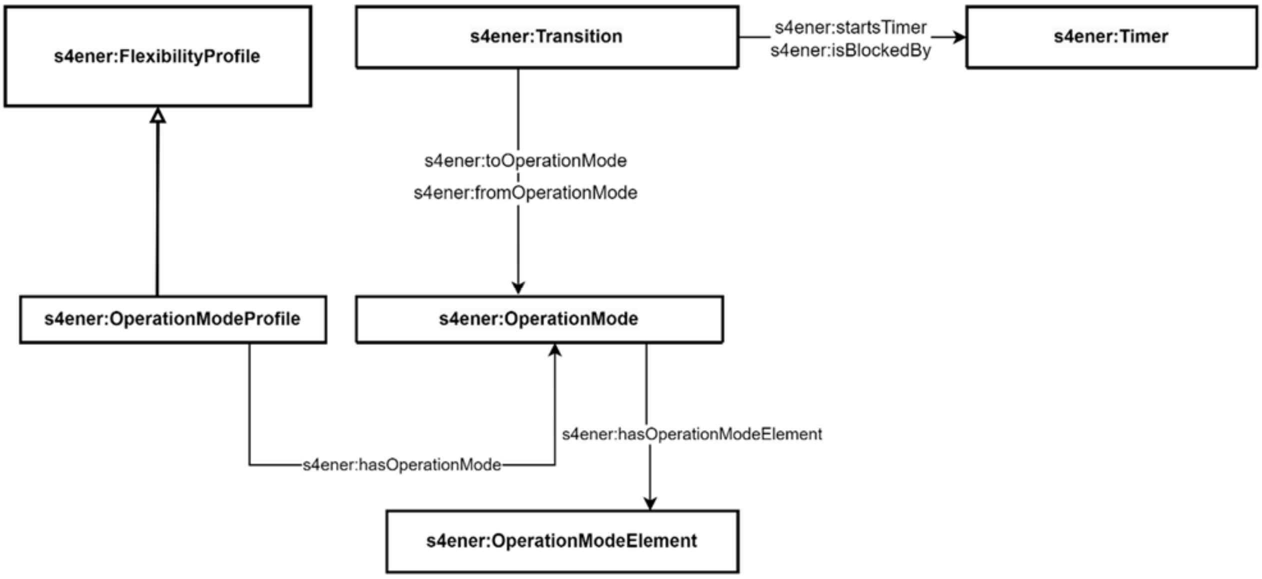

Figure 9: Operation Mode Profile

Table 19: Property of Operation Mode Profile

Property	Definition
s4ener:hasOperationMode	The relationship between the operation mode profile and the various operation mode it offers.
s4ener:hasTransition	The transitions between various Operation Modes that this OperationModeProfile can support.
s4ener:hasTimer	The set of timers that are available in this OperationModeProfile.
s4ener:hasStartTime	The moment this Operation Mode Profile becomes valid.
s4ener:hasActiveOperationMode	A reference to the OperationMode that is presently active.
s4ener:hasOperationModeFactor	The number indicates the factor with which the Operation Mode should be configured.
s4ener:hasPreviousOperationMode	The previous operation mode this device was in.
s4ener:transitionTimestamp	Time at which the transition from the previous Operation Mode was initiated.

Table 20: Properties of Operation Mode

Property	Definition
s4ener:abnormalConditionOnly	The relationship between the operation mode and the boolean datatype value indicating whether the operation mode has abnormal condition.
s4ener:hasPowerRange	The power produced or consumed by this operation mode. The start of the range is associated with operation mode factor 0, the end of the range is associated with operation mode factor 1.
s4ener:hasRunningCosts	Additional costs per second (e.g. wear, services, or money) associated with this operation mode.

Table 21: Properties of Timer

Property	Definition
s4ener:isFinishedAt	The relationship between the timer and its date-time datatype value.
s4ener:hasDuration	The time it takes for the timer to finish after it has been started.

Table 22: Properties of Transition

Property	Definition
s4ener:abnormalConditionOnly	The relationship between the transition and the boolean datatype value indicating whether the transition can only take place with an abnormal condition.
s4ener:hasTransitionCosts	The relationship between the operation mode and the decimal datatype value indicating the transition costs from a particular operation mode to another.
s4ener:toOperationMode	The relationship between the transition and the ID of the operation mode that will be switched to.
s4ener:fromOperationMode	The relationship between the transition and the ID of the operation mode that will be switched from.
s4ener:startsTimer	The relationship between the transition and the IDs of timers that will be (re)started when this transition is initiated.
s4ener:isBlockedBy	The relationship between the transition and the IDs of timers that block this transition if not finished.
s4ener:hasTransitionDuration	Indicates the delay between the initiation of this Transition, and the time at which the device behaves according to the Operation Mode.

4.2.3.6 Power Limitation

4.2.3.6.1 Power Envelope Profile

A saref:Device offers a `s4ener:PowerEnvelopeBasedProfile` when the device is operating within a minimum and maximum amount of power for energy production and/or consumption per time block, but the production or consumption cannot be directly regulated by the energy manager. A PV panels inverter is a typical example, because the energy produced is dependent on the amount of sunshine. The EMS may constrain the power production of the PV panels below its potential to lower a peak.

The minimum and maximum amount of power that can be generated and/or spent by a device in a certain timespan can be set by instantiating the `s4ener:PowerEnvelope` and its corresponding `s4ener:PowerConstraint`. Power constraints are always bound to the allowed power limit ranges of a device (`s4ener:AllowedLimitRange`). The energy level of the `s4ener:PowerEnvelope` can be defined by using `s4ener:TimeSeries`. The type of the *allowed limit ranges* of a device (i.e. upper limit or lower limit) can be defined through the class `s4ener:PowerEnvelopeLimitType`. Commodity quantities *relating to* `s4ener:PowerEnvelope` can be described through the class `s4ener:CommodityQuantity`.

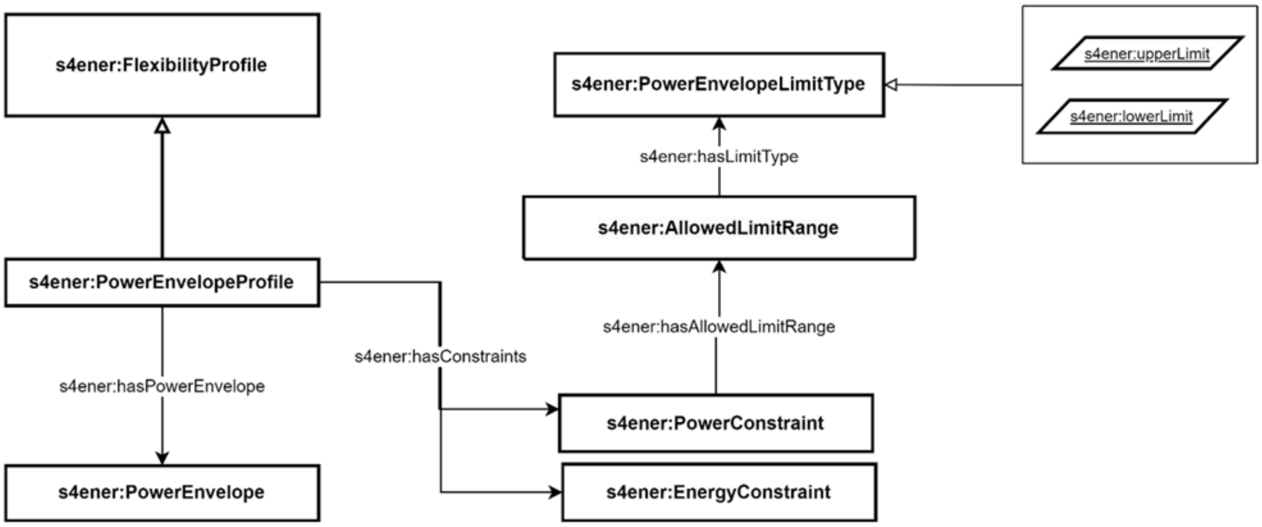

Figure 10: Power Envelope Profile

Table 23: Property of Power Envelope Profile

Property	Definition
s4ener:hasPowerEnvelope	The Power Envelope is a specification received by a resource from an energy manager. This specification is received dynamically and conforms to the power constraints and energy constraints of the resource.
s4ener:hasConstraints	A reference to the sets of constraints that a power envelope consists of. Each set of constraint concerns either power constraints or energy constraints. A device has to have at least one set of power constraints.

Table 24: Properties of Power Envelope

Property	Definition
s4ener:relatesToCommodity Quantity	The relationship between the power envelope and the commodity quantity this envelope constraints.
s4ener:hasEnvelope	The relationship between the power envelope and the time series indicating the actual values of the envelope.

Table 25: Properties of Power Constraint

Property	Definition
s4ener:allowedLimitRange	The relationship between the power constraint and the allowed limit range of this constraint.
s4ener:hasStartTime	The moment this constraint becomes valid.
s4ener:hasEndTime	The moment until which this constraint is valid.
s4ener:hasConsequenceType	The consequence type of this power constraint, which can be Vanish or Defer. This indicates whether surplus power is wasted (`s4ener:Vanish`) or stored (`s4ener:Defer`).

Table 26: Properties of Energy Constraint

Property	Definition
s4ener:rangeBoundary	The range of the average consumed power during this time period.
s4ener:hasStartTime	The moment this constraint becomes valid.
s4ener:hasEndTime	The moment until this constraint is valid.
s4ener:relatesToCommodityQuantity	The commodity quantity this energy constraint applies on.

Table 27: Property of Allowed Limit Range

Property	Definition
s4ener:abnormalConditionOnly	The relationship between the power constraint and the boolean datatype value indicating whether the power constraint has an abnormal condition.
s4ener:limitType	The relationship between the allowed limit range and the power envelope limit type, which can be upper or lower limit.
s4ener:relatesToCommodityQuantity	The commodity quantity this allowed limit range relates draws its energy or power from.
s4ener:rangeBoundary	The range indicating the boundaries of this allowed limit range.

4.2.3.6.2 Power Limit Profile

SAREF4ENER further specifies allowed limit ranges through the classes `s4ener:ContractualPowerLimit`, `s4ener:NominalPowerLimit`, and `s4ener:FailsafePowerLimit`. They are all subclasses of `s4ener:PowerLimit` which is the general upper-class of power limits. Power limits can be toggled active or inactive via the `s4ener:isActive` property. A device has nominal power consumption and/or production values (`s4ener:NominalPowerLimit`) when the manufacturers define quantifiable and measurable limits that has not to be exceeded. The failsafe values provided by the manufacturers has to be given as instances of `saref:Observation`. In case the communication between a device and the energy manager is interrupted, the device enters a fail-safe state (`s4ener:FailsafeState`). Fail-safe values (`s4ener:FailsafePowerLimit`) apply until the communication is re-established, with an optional minimal duration of the fail-safe state given in the `s4ener:hasFailsafeDuration`. Ultimately, a `saref:Device` is always *bound to* a `s4ener:ContractualPowerLimit` (which is defined in a specification by the manufacturers) and *limited by* a `s4ener:FailsafePowerLimit`.

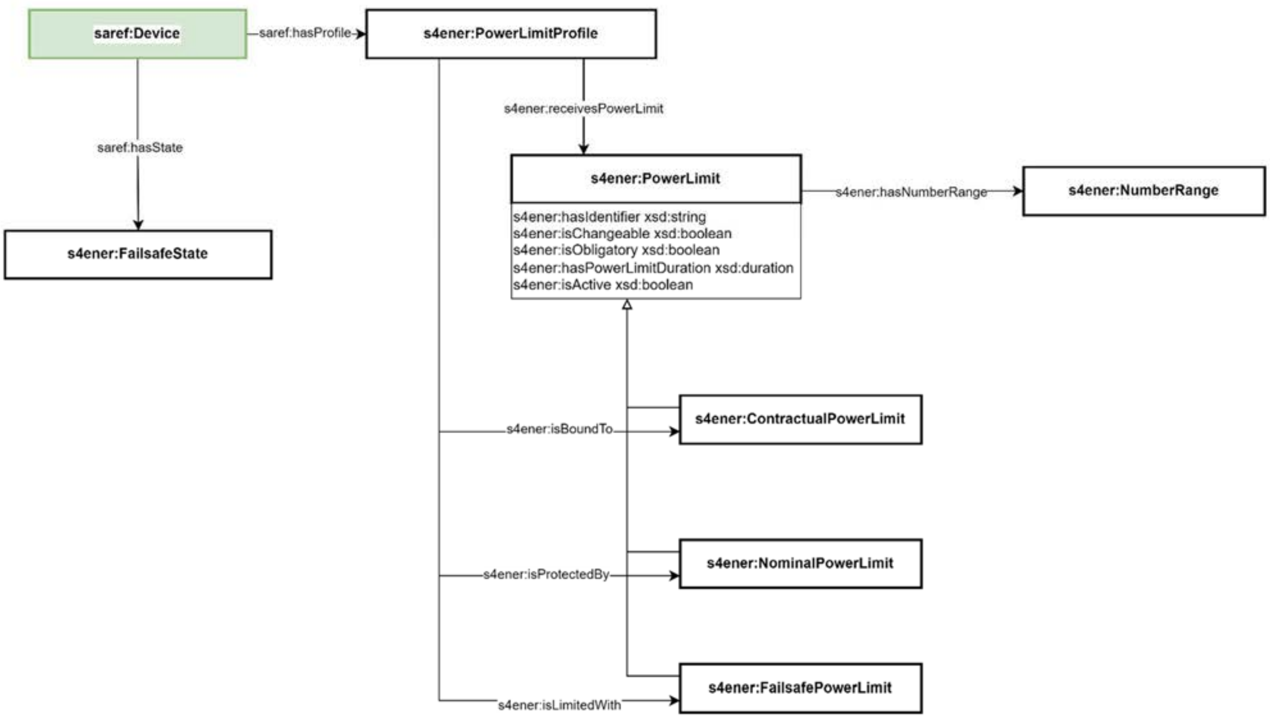

Figure 11 Power Limit Profile

Table 28: Power Limit Profile

Property	Definition
s4ener:receivesPowerLimit	The Power Limit received by the device from an energy manager that conforms to the other power limits of the device.
s4ener:isBoundTo	A contractual power limit the device is bound to.
s4ener:isProtectedBy	A nominal power limit the device is protected by.
s4ener:isLimitedWith	The power limit that the device is limited with when it is in a fail-safe state.

Table 29: Power Limit

Property	Definition
s4ener:isChangeable	The relationship between the power limit and the boolean datatype value indicating whether the power limit is changeable.
s4ener:isObligatory	The relationship between the power limit and the boolean datatype value indicating whether the power limit is obligatory.
s4ener:hasDuration	The duration of the power limitation.
s4ener:isActive	The relationship indicating whether this Power Limit is currently active.
s4ener:hasNumberRange	The Power Limit can have a number range indicating the range of this limit.

Table 30: Failsafe State

Property	Definition
s4ener:hasFailsafeDuration	Indication of the minimum duration a device should stay in the failsafe state once entered.

4.2.4 Load control

This clause presents the part of SAREF4ENER that defines how to model events used in, for example, a direct load management or power curtailing scenario (e.g. use case 3 in clause 4.1). The classes of interest are `s4ener:LoadControlEventData`, `s4ener:LoadControlEventAction`, `s4ener:LoadControlStateData` and `s4ener:LoadControlState`.

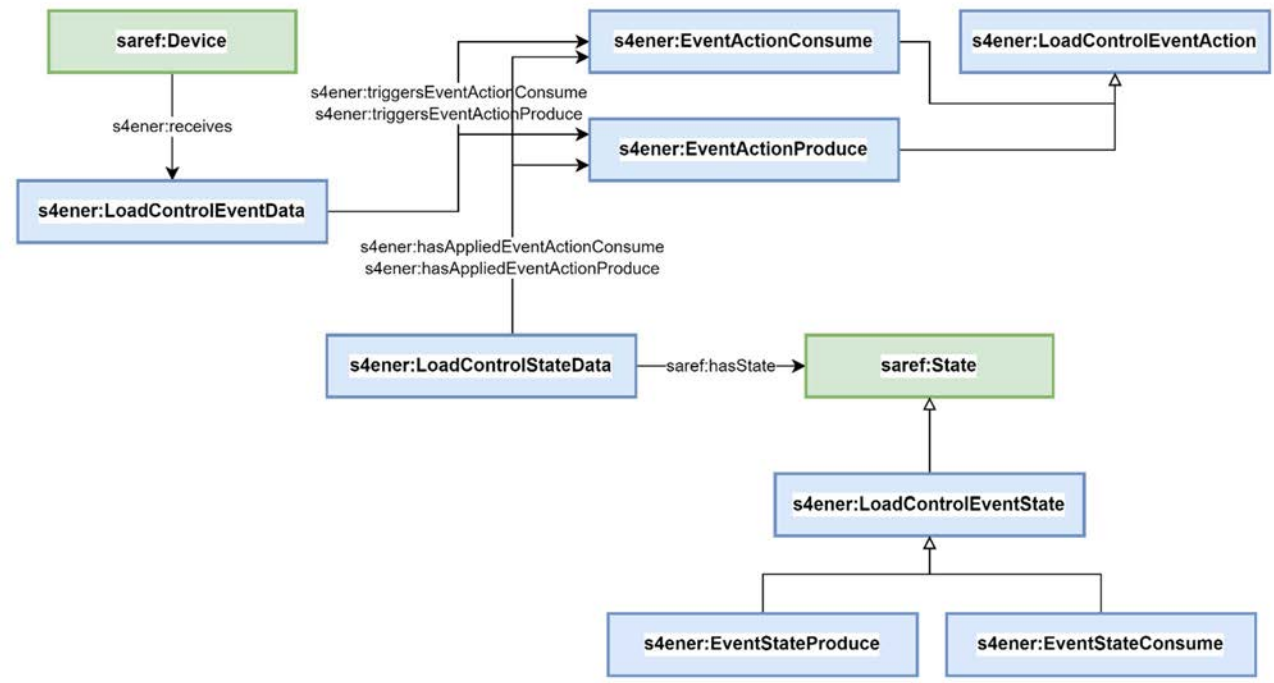

Figure 12: Load Control

The `s4ener:LoadControlEventData` class is used to represent overload warning severity level and related load control commands to a device. It is characterized by an event ID and a timestamp that represents the time the event information instance was created or received, and the time period that denotes the period of validity of the event. For example, 5 minutes ago an event was received which says that it shall take effect tomorrow from 14:00 to 15:30. In this event the timestamp is "5 minutes ago" and time period is "tomorrow from 14:00 to 15:30".

The `s4ener:LoadControlEventAction` class expresses the type of actions to be performed as a consequence of a load control event. A `s4ener:LoadControlEventAction` can be of type "consume" or "produce" to denote consumption or production of energy or power. Values for both consume and produce actions can be `s4ener:emergency`, `s4ener:increase`, `s4ener:normal`, `s4ener:pause`, `s4ener:reduce`, `s4ener:resume`.

The `s4ener:LoadControlStateData` class expresses the data about the state of an event and is characterized by the same event ID used in the `s4ener:LoadControlEventData` class, as well as a timestamp, and it is associated to the class `s4ener:LoadControlState`, which can be of type "consume" or "produce" - analogously to a load control event action - and expresses the possible states of a load control event. Values for both consume and produce load control states can be `s4ener:eventAccepted`, `s4ener:eventStarted`, `s4ener:eventStopped`, `s4ener:eventRejected`, `s4ener:eventCancelled`, or `s4ener:eventError`.

4.2.5 Flexibility Communication

4.2.5.1 Flexibility Request

This clause presents how flexibility requests can be modelled in SAREF4ENER (e.g. use case 7 in clause 4.1). This message can be sent by an EMS to a device to inquire for the flexibility it can offer. A flexibility requests can be defined by using the `s4ener:FlexRequest` class. Flexibility requests can *include* a `s4ener:IncentiveTable`, `s4ener:FlexibilityProfile`, `s4ener:TimeSeries` and `s4ener:Datapoint`. An `s4ener:FlexRequest` can be produced by an agent (`foaf:Agent`) or device (`saref:Device`) and be sent to either an agent or device. An `s4ener:FlexRequest` always has an effective period and a creation time expressed through the Time ontology.

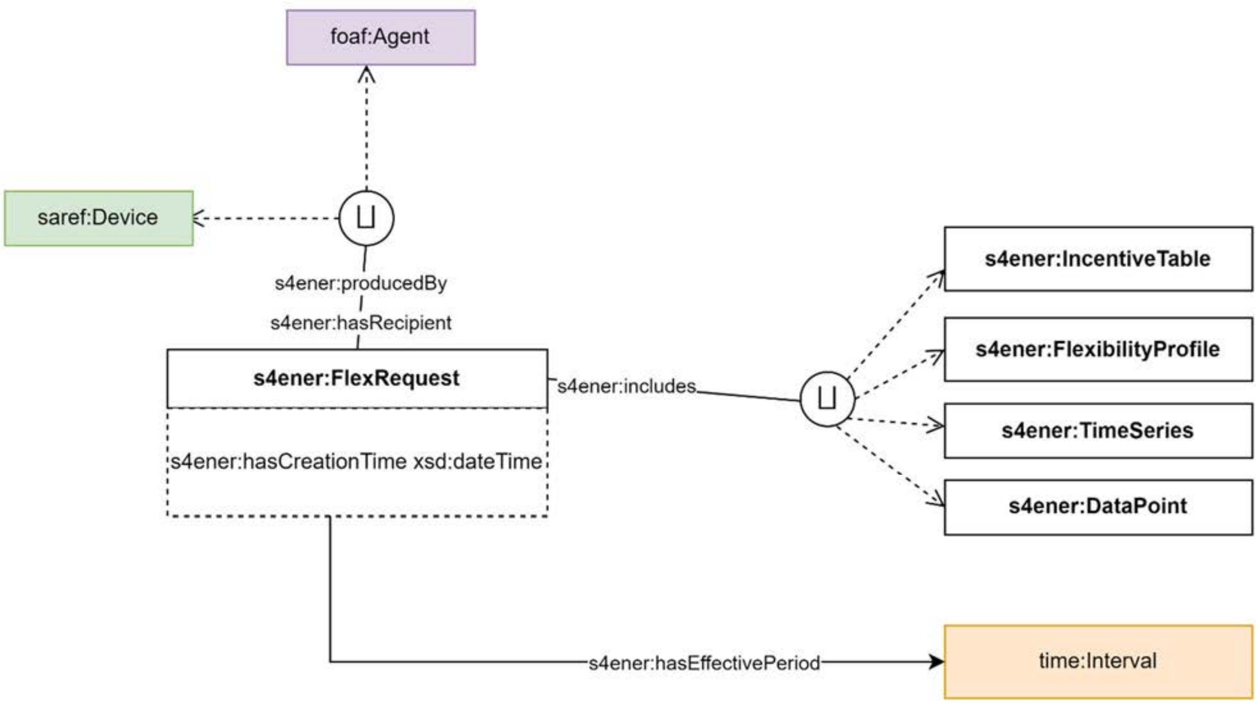

Figure 13: Flexibility Request

Table 31: Flexibility Request

Property	Definition
s4ener:producedBy	The relationship between the flexibility request and the `foaf:agent` or `saref:Device` that produced the flexibility request.
s4ener:hasRecipient	The relationship between the flexibility request and the `foaf:agent` or `saref:Device` to which the flexibility request is directed.
s4ener:includes	The relationship between the flexibility request and the incentive table, flexibility profile, timeseries and datapoint included in the flexibility offer.
s4ener:hasCreationTime	The relationship between the flexibility request and its creation time.
s4ener:hasEffectivePeriod	The relationship between the flexibility request and its creation time.

4.2.5.2 Flexibility Offer

This clause presents how flexibility offers can be modelled in SAREF4ENER (e.g. use case 7 in clause 4.1). This message can be sent by a device to the EMS as a response to a Flexibility Request, indicating the device's flexibility potential. Flexibility offers can be defined by using `s4ener:FlexOffer`. Flexibility offers can *include* a `s4ener:IncentiveTable`, `s4ener:FlexibilityProfile`, `s4ener:TimeSeries` and `s4ener:Datapoint`. A `s4ener:FlexOffer` can be produced by an agent (`foaf:Agent`) or device (`saref:Device`) and it can be sent to either an agent or device. Flexibility offers relate to flexibility requests (`s4ener:FlexRequest`). A `s4ener:FlexOffer` always has an effective period and a creation time expressed through the Time ontology.

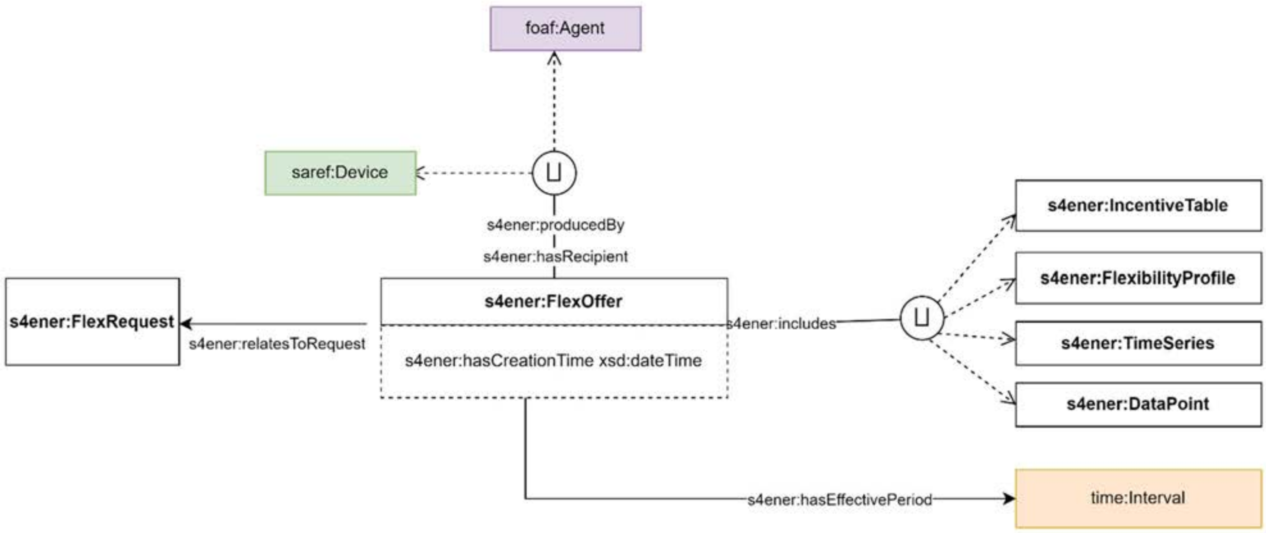

Figure 14: Flexibility Offer

Table 32: Flexibility Offer

Property	Definition
s4ener:relatesToRequest	The relationship between the flexibility offer and the flexibility request.
s4ener:producedBy	The relationship between the flexibility offer and the `foaf:Agent` or `saref:Device` that produced the flexibility offer.
s4ener:hasRecipient	The relationship between the flexibility offer and the `foaf:Agent` or `saref:Device` to which the flexibility offer is directed.
s4ener:includes	The relationship between the flexibility offer and the incentive table, flexibility profile, timeseries and datapoint included in the flexibility offer.
s4ener:hasCreationTime	The relationship between the flexibility offer and its creation time.
s4ener:hasEffectivePeriod	The relationship between the flexibility offer and its creation time.

4.2.5.3 Flexibility Instruction

This clause presents how a flexibility instruction can be modelled in SAREF4ENER (e.g. use case 7 in clause 4.1). This class describes the instruction that an EMS sends to a device about how it should operate according to the EMS optimization plan. Flexibility instruction can be defined by using `s4ener:FlexibilityInstruction`. Flexibility instructions have an activation plan expressed in time-series (`s4ener:TimeSeries`) and a cost defined as a datapoint (`s4ener:DataPoint`). A `s4ener:FlexInstruction` can have an execution time, period of validity, instructionID defined as datatype values. The operation mode factor and the presence of abnormal condition can be specified through the datatype properties `s4ener:abnormalConditionOnly` and `s4ener:operationModeFactor`. Flexibility instructions relate to flexibility requests (`s4ener:FlexRequest`). An `s4ener:Flexinstruction` can be produced by an agent (`foaf:Agent`) or device (`saref:Device`) and be sent to either an agent or device. An `s4ener:FlexInstruction` always has an effective period and a creation time expressed through the Time ontology.

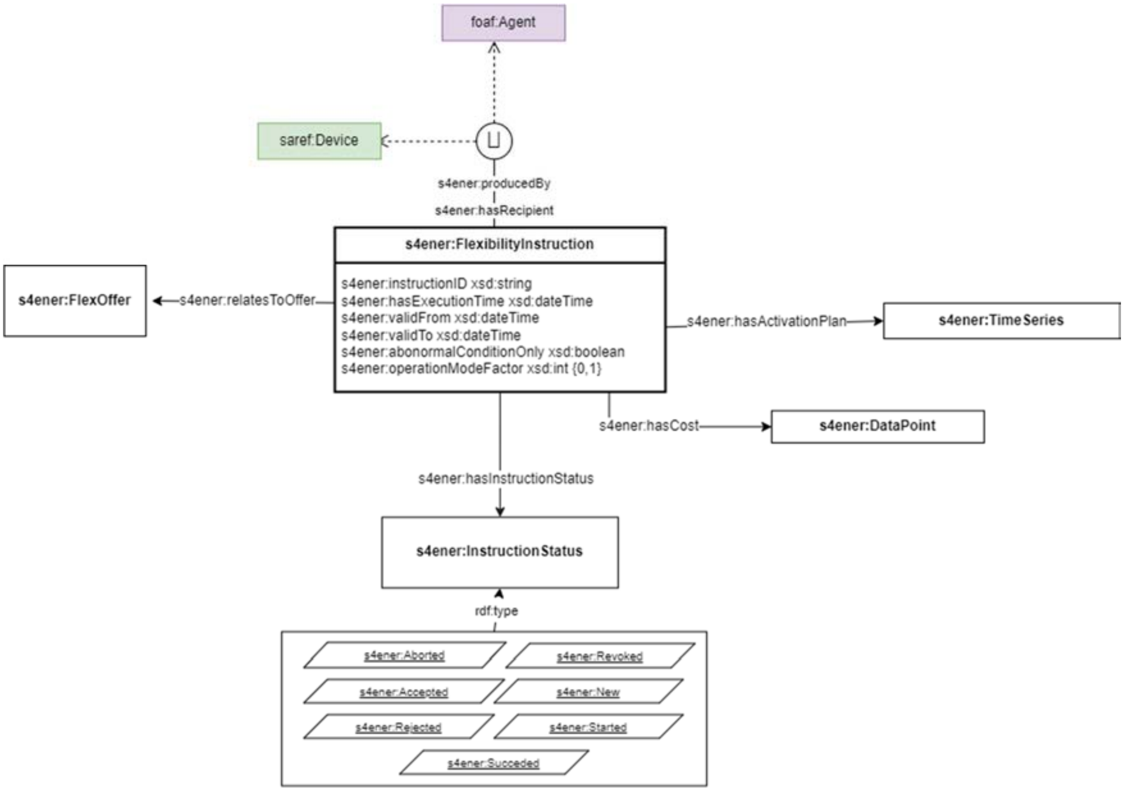

Figure 15: Flexibility Instruction

Table 33: Flexibility Instruction

Property	Definition
s4ener:producedBy	The relationship between the flexibility instruction and the `foaf:Agent` or `saref:Device` that produced the flexibility instruction.
s4ener:hasRecipient	The relationship between the flexibility instruction and the `foaf:Agent` or `saref:Device` to which the flexibility request is directed.
saref:hasPrice	The relationship between the flexibility instruction and the cost expressed as a datapoint.
s4ener:hasInstructionStatus	The relationship between the flexibility instruction and its instruction status.
s4ener:hasExecutionTime	The relationship between the flexibility instruction and the dateTime datatype value of its execution time.
s4ener:hasStartTime	The relationship between the flexibility instruction and the dateTime datatype value expressing the starting time of its validity.
s4ener:hasEndTime	The relationship between the flexibility instruction and the dateTime datatype value expressing the ending time of its validity.
s4ener:abnormalCondition	The relationship between the flexibility instruction and the boolean datatype value indicating whether the power constraint has an abnormal condition.
s4ener:hasOperationModeFactor	The relationship between the flexibility instruction and the integer datatype value expressing its operation mode factor.
s4ener:relatesToOffer	The relationship between the flexibility instruction and the flexibility offer.

4.2.6 Data Points and Time Series

The `s4ener:DataPoint` is an atomic piece of information about a certain observable quantity in nature that can contain a numerical value and a corresponding unit of measure [i.7]. An `s4ener:TimeSeries` is related to the `s4ener:Datapoint` class via the `s4ener:hasDataPoint` property and is defined as an ordered sequence of data-points of a quantity that is observed at spaced (not necessarily equally spaced) time intervals.

The `s4ener:DataPoint` is defined as a subclass of `saref:Observation` and, as such, inherits the `saref:hasResult` and `saref:hasTimestamp` properties. Therefore, if the combination of a result and timestamp is sufficient to represent a datapoint, then the SAREF concepts for observation can be directly reused. However, it can be noticed that often, especially when representing timeseries of datapoints in a forecast, a number of additional properties are needed.

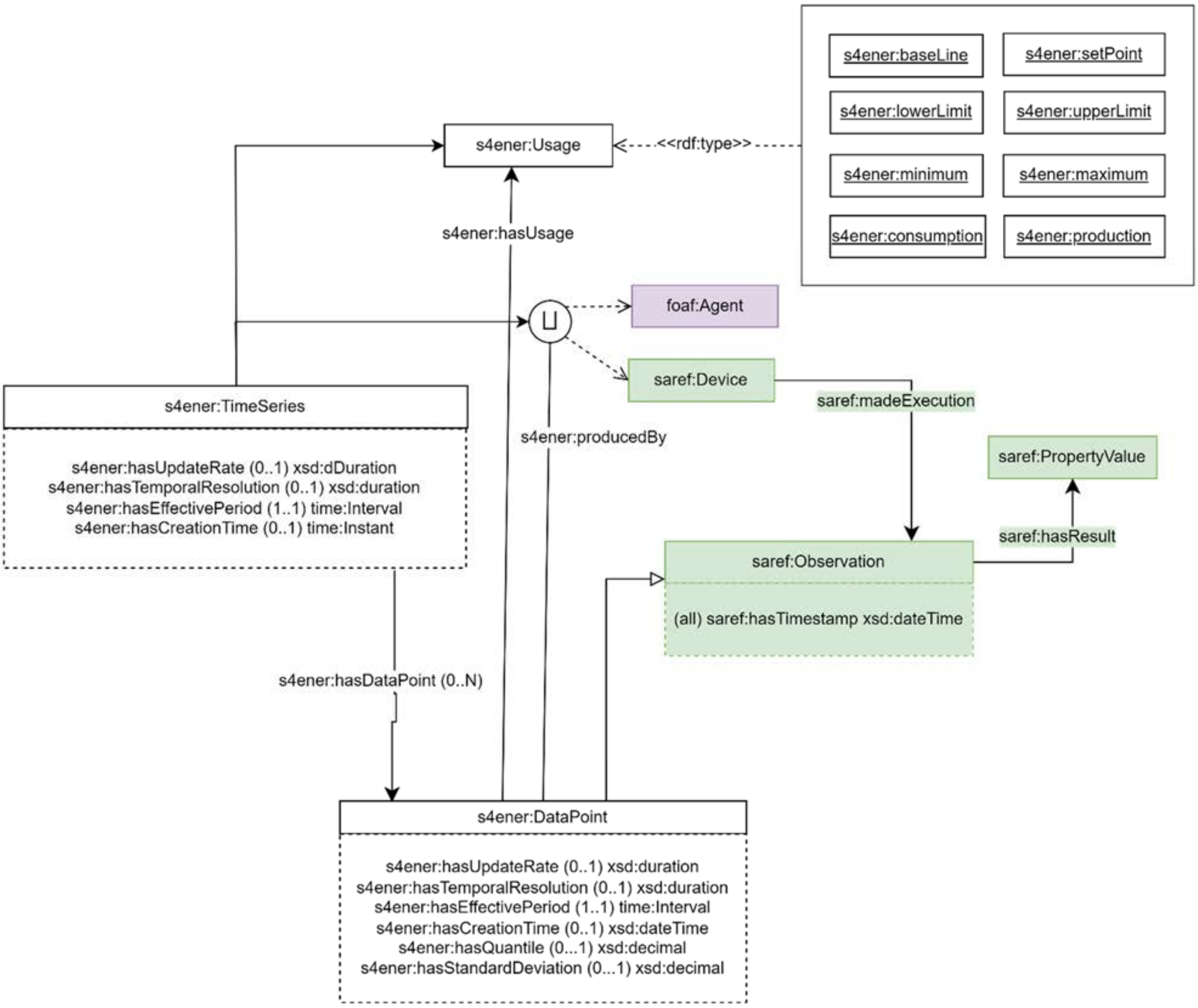

Figure 16: Time Series and Data Point

Table 34: Time Series

Property	Definition
s4ener:hasUpdateRate	Defines the rate at which a data point or timeseries is being updated.
s4ener:hasTemporalResoultion	Defines the distance between two data points measured at different times.
s4ener:hasEffectivePeriod	Connects to the interval (with a beginning and an end) in which the data point was, is, or will be in effect.
s4ener:hasCreationTime	Defines the instant in which a data point or timeseries has been created.
s4ener:hasUsage	Gives some additional information about the usage of a data point, i.e. to define for which purpose the datapoint or timeseries is used.
s4ener:hasDataPoint	Refers to the various data points that this timeseries consists of.

Table 35: Data point

Property	Definition
s4ener:belongsToTimeSeries	Relates a data point to the timeseries it may belong to.
s4ener:hasQuantile	This property assigns to the data point the percentage of values that are below this value. In other words, a data point with quantile 90 indicates that 90 % of other data points are (estimated to be) lower.
NOTE: This class contains the same properties as Timeseries, except for `s4ener:hasDataPoint`.	

Table 36: Gaussian data point

Property	Definition
s4ener:hasStandardDeviation	This is a mandatory property for Gaussian forecast data points. The standard deviation (i.e. the square root of the average of the squared deviations of the values subtracted from their average value) can be described with this property.

4.3 Exemplifying SAREF4ENER

Clause 4.3 provides a set of examples to show the usage of the various SAREF4ENER clauses. The following are available on the SAREF Labs repository for SAREF4ENER:

- Clause 4.2.3.1: Power Profile Example
  - Following S2
  - Following SPINE
- Clause 4.2.3.2: Demand Driven Profile
- Clause 4.2.3.3: Fill Rate Based Profile
- Clause 4.2.3.4: Incentive Table Based Profile
- Clause 4.2.3.5: Operation Mode Profile
- Clause 4.2.3.6.1: Power Envelope Profile
- Clause 4.2.3.6.2: Power Limit Profile
- Clause 4.2.6 is exemplified throughout the flexibility profile examples mentioned above.

4.4 Discussion

The present document is a major revision of the SAREF4ENER ontology extension, developed, using updated reference ontology patterns specified in ETSI TS 103 548 [2] to solve the harmonization needs identified in ETSI TR 103 781 [i.13], with updated development framework and tools defined in ETSI TS 103 673 [i.14].

As all the SAREF ontologies, SAREF4ENER is a dynamic semantic model that is meant to evolve over time. Therefore, the stakeholders in the energy domain are invited to use, validate and provide feedback on SAREF4ENER, collaborating with the SAREF ontology experts to improve and evolve SAREF4ENER in an iterative and interactive manner, so that changes and additions can be incorporated in future releases of the present document. The first extension of SAREF for the energy domain was originally called SAREF4EE [i.1], since it was created for the Energy@Home and EEBus industry associations with the purpose to make semantically interoperable their different data models, as described in [i.10]. However, once it was brought to ETSI to make it an official specification, that initial SAREF4EE extension has been renamed to SAREF4ENER, as in the present document, according to the naming convention for SAREF extensions adopted in ETSI TR 103 411 [i.4] (i.e. SAREF4XXXX, where XXXX are letters that describe the domain for which the extension was created).

The first version of SAREF4ENER (V1.1.1) was based on SAREF core V2.1.1. The subsequent version SAREF4ENER V1.2.1 was based on SAREF core V3.1.1. The SAREF4ENER V2.1.1 specified in the present document is based on SAREF core V4.1.1.

Annex A (informative): Approach versions 2.1.1 and 1.2.1

The present document specifies the new SAREF4ENER V2.1.1 that incorporates - in addition to EN 50631:2023, parts 1 [4], 2 [5], 3-1 [6] and 4-1 [7] already incorporated in SAREF4ENER V1.1.1 - also EN 50491-12-2 [8] (called "S2") and the inputs on energy flexibility from the Horizon 2020 InterConnect project [i.8].

The InterConnect project [i.8] was carried out from October 2019 to March 2024 in the context of the Horizon 2020 programme and involved more than 50 European partners from industry, research and academia to develop and demonstrate advanced solutions for connecting and converging digital homes and buildings with the electricity sector. The project pioneered cross-domain semantic interoperability without a centrally hosted facilitator leveraging the SAREF framework of ontologies. Large scale deployment of the SAREF-based solutions was carried out in seven connected large-scale test-sites in Portugal, Belgium, Germany, the Netherlands, Italy, Greece and France. Additional SAREF-based deployments were carried out as a result of a first open call (which funded 10 SMEs with 150 000 € each) and a second open call (which funded 7 SMEs with 150 000 € each). These open call projects extended the SAREF-based deployments to additional countries, such as Spain, UK, Ireland, Denmark, Croatia, Slovenia, Estonia, Latvia and Lithuania.

NOTE 1: The work in the InterConnect project - which resulted in SAREF4ENER V1.2.1 - is based on SAREF core V3.1.1.

The H2020 Interconnect project created strong synergies among different (and often competing) industrial partners and also provided a standardization success story:

i) the technical work to create the SAREF4ENER V1.2.1 was conducted with the Interconnect stakeholders in 2020-2021;

ii) the pre-standardization activities in collaboration with ETSI were carried out in 2022; and

iii) the official ETSI standardization workflow was finalized in 8 months in 2023, resulting in the publication of SAREF4ENER V1.2.1 in November 2023, which subsequently evolved in SAREF4ENER V2.1.1 specified in the present document, according to the new version of SAREF core V4.1.1.

NOTE 2: The present document - which specifies SAREF4ENER V2.1.1 - is based on SAREF core V4.1.1.

In order to develop the extension defined in the present document, the ontology engineering methodology adopted by ETSI for the development of the SAREF ontologies and documented in ETSI TR 103 411 [i.4] was used in the context of the InterConnect project. This methodology consists of the activities of ontology requirements specification, implementation and maintenance, and further specifies the inputs, outputs and involved actors in these activities. Further information on how this methodology has been applied in the InterConnect project can be found in [i.11]. During these activities, TNO conducted a series of workshops with the InterConnect partners to collect their input, validate some intermediate results and iteratively collect new requirements to improve and complete the existing SAREF suite of ontologies, according to the following timeline:

1) A first stakeholders' workshop took place in October 2020 in which the InterConnect ontology development team led by TNO collaborated with the InterConnect domain experts and software developers to devise together the requirements to realize the InterConnect ontologies as extensions of the ETSI SAREF suite of ontologies, making use of the use cases defined in the InterConnect project and the domain documentation provided in terms of API specifications. Relevant standards, such as EN 50491-12-2 [8] (i.e. the S2 standard used in the present document), were also taken into account already in this early stage.

2) The requirements collected in step 1, were used in Q4 2020 as input to produce an initial conceptualization and formalization in RDF/OWL of the InterConnect ontologies.

3) The subsequent implementation phase further produced three drafts of the InterConnect ontologies (i.e. early, stable and final draft) that were presented, discussed and validated with the InterConnect partners in three stakeholders' workshops that were held, respectively, in February, April and June 2021.

4) These drafts were used in the meantime to create a broader stakeholders basis for SAREF4ENER by successfully harmonizing the different approaches from the EEBUS, KNX and FAN industry associations in close collaboration with them.

5) During the summer 2021, the InterConnect ontologies were then finalized and made available to be used by the InterConnect pilots in Portugal, Belgium, Germany, the Netherlands, Italy, Greece and France as of September 2021. A close collaboration started then with the pilots in order to iteratively improve the InterConnect ontologies with their early feedback. The process resulted in the InterConnect ontologies presented in [i.11] and available in the InterConnect Wiki ontology page.

6) During the maintenance phase in 2022, the InterConnect ontologies were improved with the feedback provided by the InterConnect pilots based on their concrete usage of the ontologies.

7) Three preparatory standardization meetings were then conducted by InterConnect with ETSI and relevant industry associations, like KNX and EEBUS, respectively in March, April and September 2022 in which a common standardization strategy was agreed between ETSI SmartM2M TC and CEN/CENELEC TC 205 WG19 in order to incorporate the most important and mature exploitable results of the InterConnect ontologies into a new official version of SAREF4ENER.

NOTE 3: The explicit consensus achieved between ETSI SmartM2M TC and CENELEC TC 205 established that a new version of SAREF4ENER based on the InterConnect ontologies could be published by ETSI (as it usually happens with all SAREF standards) and afterwards receive an official endorsement by CENELEC TC 205 WG19.

8) The official standardization process followed in ETSI and was finalized in 8 months as follows:

- March 2023: opening of Work Item for the new SAREF4ENER V1.2.1.
- June 2023: approval of SAREF4ENER V1.2.1 early draft.
- September 2023: approval of SAREF4ENER V1.2.1 stable draft.
- October 2023: approval of SAREF4ENER V1.2.1 final draft.
- November 2023: publication of new SAREF4ENER V1.2.1.

9) In the period January 2024 - September 2024 the version of SAREF4ENER V1.2.1 that resulted from InterConnect has been aligned with the latest version of SAREF core V4.1.1, resulting in theSAREF4ENER V2.1.1 of the present document.

NOTE 4: The final step of endorsement of the present document by CEN/CENELEC (CLC TC 205 WG19) is ongoing at the time of writing.

Annex B (informative): Approach version 1.1.1

For the sake of completeness, this clause describes the approach that was followed back in 2016 to create the first SAREF4EE (which in 2017 was published by ETSI as SAREF4ENER V1.1.1 - now superseded by the SAREF4ENER V2.1.1 in the present document) for the Energy@Home and EEBus industry associations with the purpose to make semantically interoperable their different data models, as described in [i.10]. The approach was a combination of bottom-up and top-down steps, as shown in Figure B.1. The (bottom-up) starting point was given by the two existing data models to be harmonized, namely the E@H data model (an UML class diagram) [i.2] and the EEBus data model (an XSDs specification). These two data models represented similar concepts, such as the concept of "power profile", but using different terminologies. For example, E@H defined "power profiles", "modes" and "phases", while EEBus referred to these concepts as "power sequences", "alternatives" and "slots". In order to converge to a shared terminology, experts of EEBus and E@H preliminarily defined a common specification [i.1] that was subsequently used by TNO as basis for creating SAREF4ENER.

The preliminary phase was followed by a kick-off workshop in which the experts of EEBus and E@H presented the details of their individual data models, i.e. EEBus (XSDs) and E@H (UML), and also their common data model, the EEBus & E@H (UML+XSDs) model.

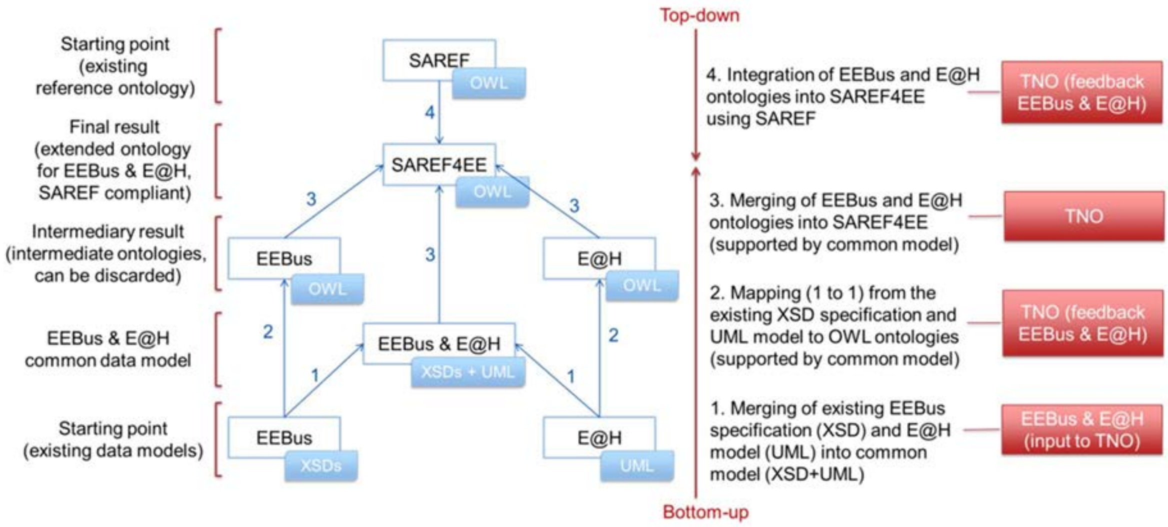

Figure B.1: Approach

Since the existing EEBus and E@H data models were expressed in different formats, i.e. XSD and UML, and SAREF4ENER had to be expressed in OWL as an extension of SAREF, these data models were first translated into corresponding OWL versions that could be used as intermediate ontologies towards the creation of SAREF4ENER. The transformations UML → OWL and XSD → OWL were performed manually, but existing tools can be used to automate this step (for example, TopBraid Composer™ Maestro Edition). The outcomes of these transformations were the EEBus (OWL) and E@H (OWL) intermediate ontologies in Figure B.1. The reason to create these two separate intermediate ontologies was practical. The common EEBus & E@H data model is a merged model whose parts could be straightforwardly identified as coming either from the EEBus or the E@H data model. Given that the EEBus and E@H experts were not yet (completely) acquainted with ontologies and OWL, their review process was facilitated by separating the generation of an OWL version in two parts. In this way, these experts could focus on their own part, namely EEBus or E@H, instead of having to deal with a single, large and more complex ontology. Moreover, these intermediate ontologies can be reused individually by the two associations if they decide to make use of an OWL version of their own data model in the future.

After receiving and incorporating the feedback from EEBus and E@H experts, the two intermediate ontologies were merged into a first version of SAREF4ENER, as depicted by step 3 in Figure B.1. Since this initial SAREF4ENER version was obtained by making a one-to-one mapping of existing data models that were implementation-driven rather than conceptual specifications, it was necessary to:

1) cleanse unnecessary redundancy, e.g. redundancy of data type properties carrying the same semantics, especially when expressing time-related information and unit of measures; and

2) create axioms that were absent in the original data models. While doing so, a top-down approach starting from SAREF was taken, as depicted by step 4 in Figure B.1. SAREF contains concepts that are rather high-level and needed further specialization into a finer-grained level of detail to accommodate the specific requirements of the EEBus and E@H use cases.

Therefore, when creating SAREF4ENER, classes and properties of SAREF V1.1.1. were reused and specialized where possible, while SAREF V1.1.1 was extended with new classes and properties where it did not suffice for the purpose.

In particular:

- Only a subset of concepts defined in SAREF V1.1.1 was reused, i.e. saref:Device, saref:Profile, saref:State, saref:Energy, saref:Power, saref:UnitOfMeasure and saref:Time.

- The saref:Device and saref:Profile classes were specialized in the more specific s4ener:Device and s4ener:PowerProfile subclasses, respectively. Devices and power profiles in SAREF4ENER V1.1.1 had specific properties for EEbus and E@H that did not apply to all SAREF V1.1.1 devices and profiles.

Annex C (informative): Bibliography

- ETSI TS 103 267: "SmartM2M; Smart Applications; Communication Framework".
- ETSI TS 102 689: "Machine-to-Machine communications (M2M); M2M Service Requirements".
- ETSI TS 118 101: "oneM2M; Functional Architecture (oneM2M TS-0001)".
- ETSI TS 118 102: "oneM2M; Requirements (oneM2M TS-0002)".
- ETSI, European Commission and TNO: "Study on Semantic Assets for Smart Appliances Interoperability", final report, April 2015.
- Spatial Data on the Web Interest Group: "Extensions to the Semantic Sensor Network Ontology", W3C® Working Draft, 16 January 2020.

Annex D (informative): Change history

Date	Version	Information about changes
January 2017	V1.1.1	• Introduced the Power Profile, Alternatives Group, and associated classes • Introduced the Load Control as a class together with associated classes for its use case
May 2020	V1.1.2	• Changed namespace and prefix
November 2023	V1.2.1	• Added Flexibility Profiles Clause as a superclass for all flexibility profiles • Extended the coverage of the flexibility profiles to include Demand Driven, Fill Rate based, Incentive Table based, Operation Mode, Power Envelope, and Power Limit types of flexibility • Adapted the Power Profile to be more generic • Deprecate terms that will be removed in version 2.1.1 follow the changes in SAREF v3.2.1. • Added Flexibility Communication Clause, including flexibility offer, flexibility request, and flexibility instruction • Change the datetime properties to refer to xsd:dateTimeStamp and xsd:duration instead of Time ontology concepts
May 2024	V2.1.1	• This major version primarily constitutes a revision to align with the changes introduced in SAREF v3.2.1 • Remove classes and properties deprecated in the previous version • Change Data Point superclass from saref:Measurement to saref:Observation • Change Energy properties to PropertyKinds

History

Document history		
V1.1.1	January 2017	Publication
V1.1.2	May 2020	Publication
V1.2.1	November 2023	Publication
V2.1.1	October 2024	Publication