Draft ETSI TS 103 410-1 V1.1.3 (2023-07-05)

TECHNICAL SPECIFICATION

SmartM2M;
Extension to SAREF;
Part 1: Energy Domain

Disclaimer: This DRAFT is a working document of ETSI TC SmartM2M. It is provided for information only and is still under development within ETSI TC SmartM2M.

ETSI and its Members have no liability for any current or further use/implementation of the present DRAFT.

Non-published TC SmartM2M DRAFTS stored in the "Open Area" are working documents, these may be updated, replaced, or removed at any time.

Do not use as reference material.

Do not cite this document other than as "work in progress."

Any draft approved and PUBLISHED shall be obtained exclusively as a deliverable via the ETSI Standards search page at:

http://www.etsi.org/standards-search

Reference

RTS/SmartM2M-103410-1v121

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

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.
The content of the PDF version shall not be modified without the written authorization of ETSI.
The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2023.
All rights reserved.

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.

DRAFT

Contents

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 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”.

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 V1.2.1, the SAREF extension for (Blank)

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] (to fill)

NOTE: Available at (to fill)

[2] ETSI TS 103 264 (V3.1.1): "SmartM2M; Smart Applications; Reference Ontology and oneM2M Mapping".

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] (to add)

NOTE: Available at (to add)

[i.2] (to add)

NOTE: Available at (to add)

[i.3] (to add)

NOTE: Available at (to add)

[i.4] (to add)

DRAFT

3 Definition of terms, symbols and abbreviations

3.1 Terms

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

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

3.2 Symbols

Void.

3.2 Abbreviations

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

BACS	Building Automation Control Systems
CEM	Customer Energy Manager
HBES	Home and Building Electronic Systems
OM	Ontology of units of Measure
OWL	Web Ontology Language
RM	Resource Manager
SAREF	Smart Applications REFerence ontology
SASS	Single Application Smart System
TNO	Netherlands Organization for Applied Scientific Research
TR	Technical Report
TS	Technical Specification
UML	Unified Modeling 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 extension of SAREF [2] for the energy domain that was created in collaboration with industry associations such as EEBus (http://www.eebus.org/en), Energy@Home (http://www.energy-home.it) and KNX (https://www.knx.org/).

SAREF4ENER should be used to annotate (or generate) a neutral (protocol-independent) set of messages to be directly adopted by the various manufacturers, or mapped to their domain specific protocols of choice. These messages can be exchanged by energy smart appliances with a Customer Energy Manager (CEM) and/or a to efficiently control energy consumption and production.

SAREF4ENER is an OWL-DL ontology that extends SAREF with x classes, xobject properties and x data type properties. SAREF4ENER focuses on demand response scenarios, in which customers can offer energy flexibility to the Smart Grid to manage their smart home devices by means of a Resource Manager (RM) and/or a Customer Energy Manager (CEM)The RM is responsible to accommodate user preferences and maintain the proper functioning of the devices. It is a software component that can physically run either on smart devices or on the cloud services of the manufacturers. The CEM is a software component that is responsible to find the most optimal measure between energy consumption and energy production of devices based on the customer's chosen configuration, the characteristics of the devices and the grid. The CEM can run on a gateway device or in the (cloud) environment of the manufacturer. Moreover, the Smart Grid can influence the quantity or patterns of use of the energy consumed by customers when energy-supply systems are constrained, e.g. during peak hours. These scenarios involve (but are not limited to) the following use cases:

- **Use case 1:** configuration of devices that want to connect to each other in the home network, for example, to register a new dishwasher to the list of devices managed by the CEM;
- **Use case 2:** smart energy management/ (re-)scheduling appliances in certain modes and preferred times using power profiles to optimize energy efficiency and accommodate the customer's preferences;
- **Use case 3:** monitoring and control of the start and status of the appliances;
- **Use case 4:** 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 5:** delayed start of smart appliances
- **Use case 6:** power limits
- **Use case 7**: incentive table

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 CEM 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 within the SAREF4ENER V1. 2. 1 ontology

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.wurvoc.org/vocabularies/om-1.8/
xsd	http://www.w3.org/2001/XMLSchema#
time	http://www.w3.org/2006/time#

4.2 SAREF4ENER

4.2.1 General Overview

An overview of the SAREF4ENER V1. 2. 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, RDFS 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 color is used to distinguish SAREF core entities. The blue color is used for highlithing the classes and properties already exisiting in the previous version of SAREF4ENER (V 1.1.2). The white color is used to denote the classes and properties that have been added in the SAREF4ENER version specified in the present document (V1.2.1). 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 from clause 4.2.2 to clause 4.2.x.

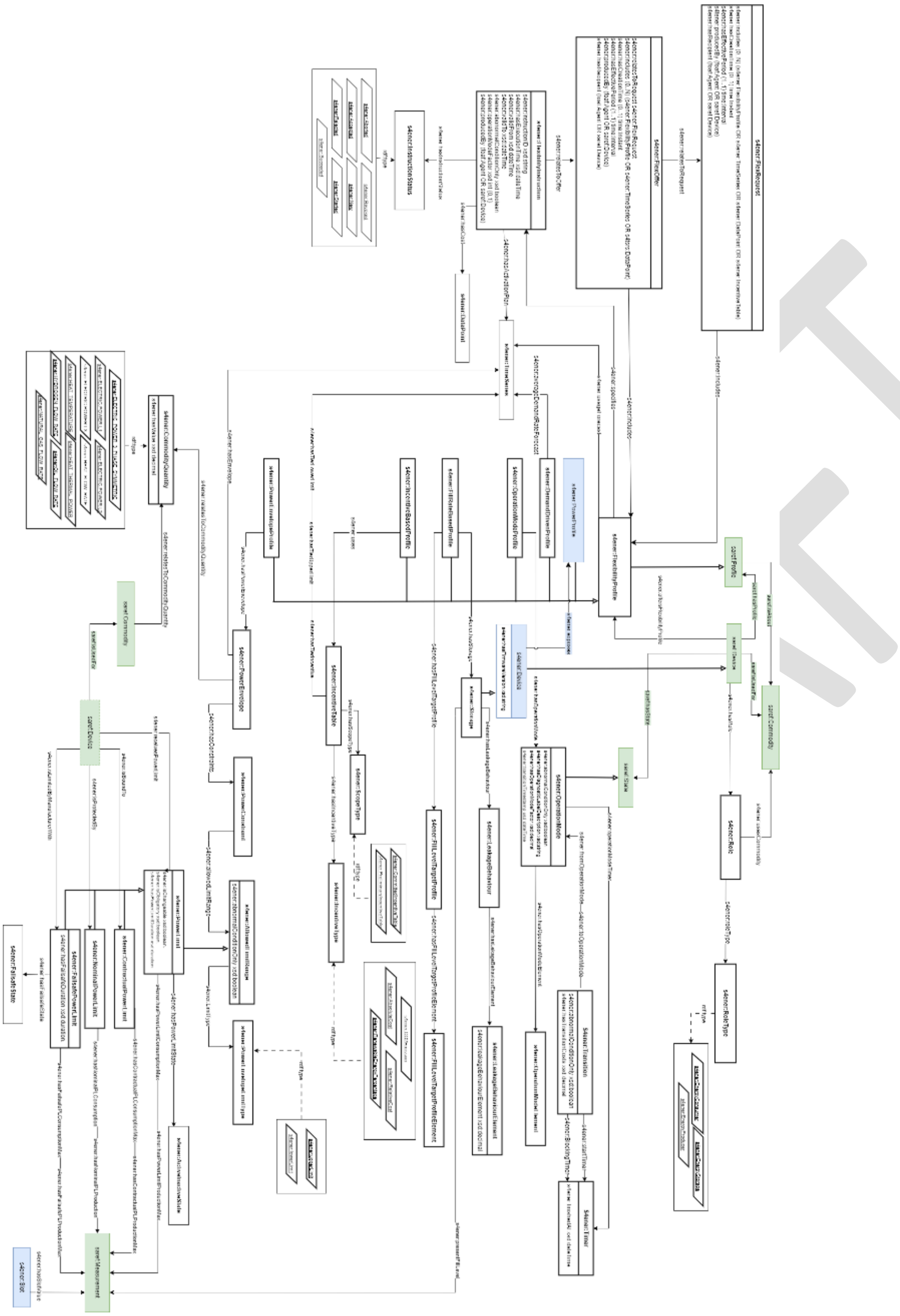


Figure 1: SAREF4ENER overview

4.2.2 Device

A s4ener:Device is a subclass of a saref:Device, i.e. it inherits the properties of the more general saref:Device and extends it with additional properties that are specific for SAREF4ENER. The s4ener:Device class is shown in Figure 2.

Table 2 summarizes the properties that characterize a s4ener:Device.

Table 2: Properties of a Device

Property	Definition
s4ener:exposes **min** 0 s4ener:FlexibilityProfile	A relationship between a device and its flexibility profiles.
s4ener:receives **min** 0 s4ener:LoadControlEventData	A relationship between a device (e.g. an appliance or a smart meter) and a load control event.
s4ener:brandName **max** 1 xsd:string	The name of the brand of a device. Useful where the name of the brand and the vendor differs.
s4ener: deviceCode **max** 1 xsd:string	Device code for the device as defined by the manufacturer.
s4ener: s4ee:deviceName **max** 1 xsd:string	Name of the device as defined by the manufacturer.
s4ener: hardwareRevision **max** 1 xsd:string	Hardware revision of the device as defined by the manufacturer.
s4ener:manufacturerDescription **max** 1 xsd:string	A description for the device as defined by the manufacturer.
s4ener:manufacturerLabel **max** 1 xsd:string	A short label of the device as defined by the manufacturer.
s4ener:manufacturerNodeIdentification **max** 1 xsd:string	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 **min** 0 xsd:string	The power source of a device. Possible values are {"unknown", "mainsSinglePhase", "mains3Phase", "battery", "dc"}.
s4ener:serialNumber **max** 1 xsd:string	Serial number of a device as defined by the manufacturer. Usually the same as printed on the case.
s4ener:softwareRevision **max** 1 xsd:string	Software revision of a device as defined by the manufacturer.
s4ener:vendorCode **max** 1 xsd:string	Code for the vendor of the device as defined by the manufacturer.
s4ener:vendorName **max** 1 xsd:string	Name of the vendor of the device as defined by the manufacturer.

DRAFT

4.2.3 Flexibility ProfileSAREF4ENER defines different energy flexibility profiles that can be *offered* by a `saref:Device`. They are: `s4ener:DemandDrivenProfile, s4ener:OperationModeProfile, s4ener:FillRateBasedProfile`, `s4ener:IncentiveBasedProfile`, and `s4ener:PowerEnvelopeProfile`. They are all subclasses of `s4ener:FlexibilityProfile` which is, in turn, a subclass of `saref:Profile`. Figure 2 displays the taxonomy of the flexibility profiles defined in SAREF4ENER V1. 2. 1.

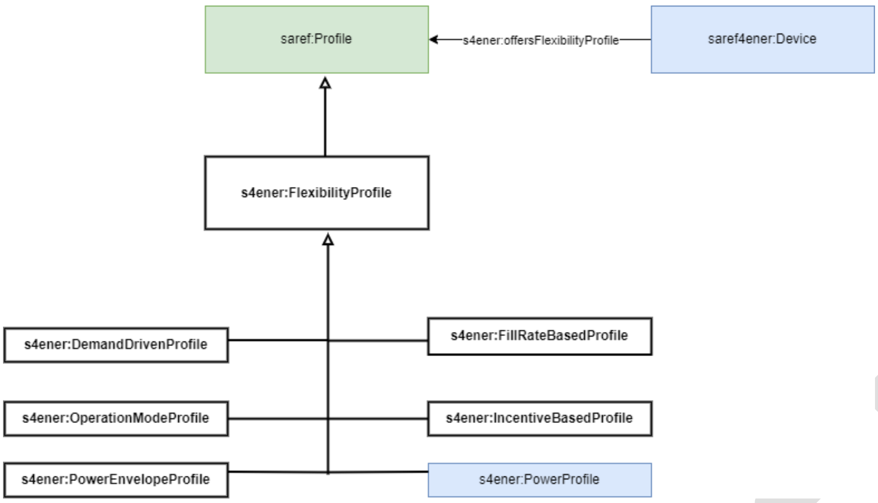


Figure 2: SAREF4ENER Flexibility Profiles

4.2.3.1 Demand Driven Profile

`S4ener:DemandDrivenProfile` can be used for devices that can consume different types of energy resources such as electricity or natural gas. 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`) as presented in figure 3.

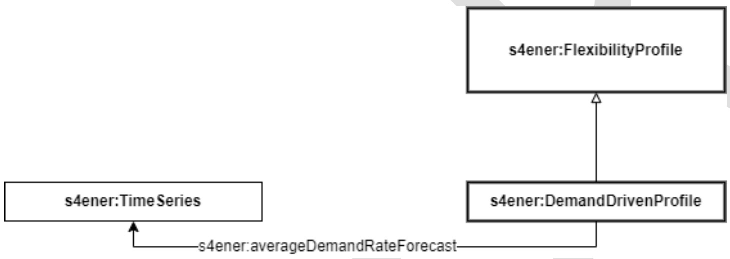


Figure 3: Demand Driven Profile

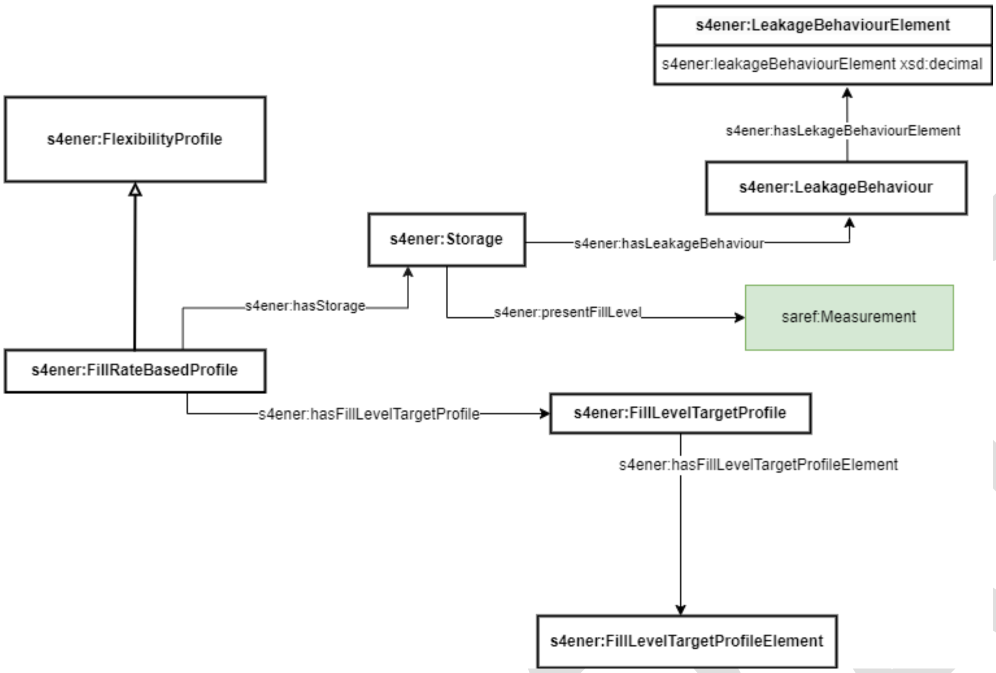
Table 3 displays the property that characterize the `s4ener:DemandDrivenProfile`.

Table 3: Property of Demand Driven Profile

Property	Definition
s4ener:avarageDemandRateForecast	The relationship between the demand driven profile and the time series

4.2.3.2 Fill Rate Based Profile

`S4ener:FillRateBasedProfile` can be used for devices that can store energy (`s4ener:Storage`). 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:Measurement` via the properties `s4ener:hasLeakageBehaviour` and `s4ener:presentFillLevel`. `S4ener:LeakageBehaviour` is always associated with an *element* detailing the leakage behaviour of the storage (`s4ener:LeakageBehaviourElement`). The decimal value of `s4ener:LeakageBehaviourElement` can be expressed through the datatype property `s4ener:leakageBehaviourElement`. Ultimately, certain storage devices might have a fill-level target profile (`s4ener:FillLevelTargetProfile`) with its associated `s4ener:FillLevelTargetProfileElement`. Figure 4 displays the way in which `S4ener:FillRateBasedProfile` relates with the afore presented entities.



Tables 4-8 summarize the properties that characterize a `s4ener:FillRateBasedProfile` (table 4), `s4ener:Storage` (table 5), `s4ener:LeakageBehaviour` (table 6), `s4ener:LeakageBehaviourElement` (table 7), and `s4ener:FillLevelTargetProfile` (table 8).

Figure 4: Fill Rate Based Profile

Table 4: Properties of Fill Rate Based Profile

Property	Definition
s4ener:hasFillLevelTargetProfile	The property that connects the fill rate based profile to the current fill level target profile that tries to accomodate for
s4ener:hasStorage	The relationship between the fill rate based profile and the storage

Table 5: 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 measurement with a percentage value indicating the storage fill level

Table 6: Properties of Leakage Behaviour

Property	Definition
s4ener:hasLeakageBehaviourElement	The property that relates the leakage behaviour to the leakage behaviour element(s)

Table 7: Properties of Leakage Behaviour Element

Property	Definition
s4ener:leakageBehaviourElement xsd:decimal	The property that connects the leakage behaviour element with its decimal datatype value

Table 8: Properties of Fill Level Target Profile

Property	Definition
s4ener:hasFillLevelTargetProfileElement	The property that connects the fill level target profile to the fill level target profile element

4.2.3.3 Incentive Based Profile

`S4ener:IncentiveBasedProfile` can be employed in devices that *use* an incentive table (`s4ener:IncentiveTable`). The *incentive table* allows monitoring different types of incentives (`s4ener:IncentiveType`) used by devices. Incentive types can be expressed in the form of relative costs (`s4ener:RelativeCost`), absolute costs (`s4ener:AbsoluteCost`), CO2 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:PreliminaryIncentiveTable)` or committed version (s4ener:CommittedIncentiveTable). Tier upper limits, tier lower limits and tier incentive values can be expressed as time series (s4ener:TimeSeries) as displayed in figure 5.

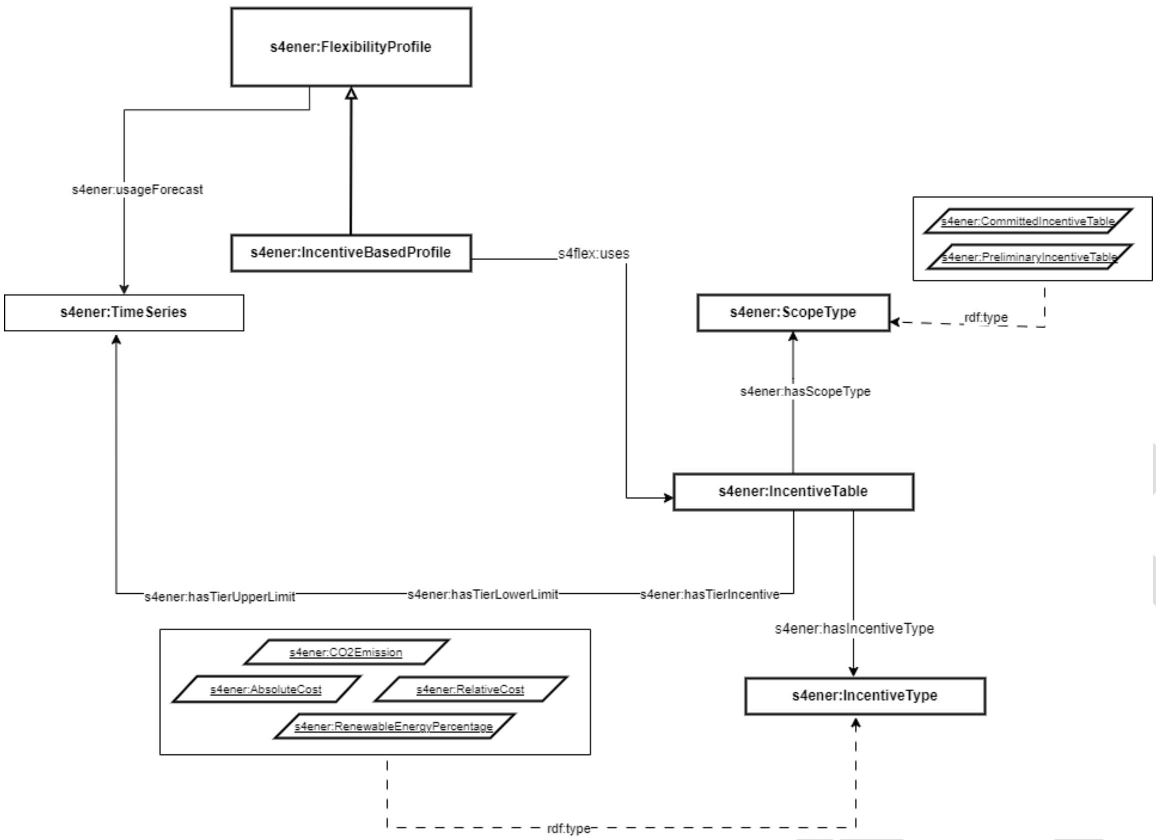


Figure 5: Incentive Based Profile

Tables 9-10 summarize the properties that characterize a s4ener:IncentiveBasedProfile (table 9) and s4ener:IncentiveTable (table 10).

Table 9: Properties of Incentive Based Profile

Property	Definition
s4ener:uses	The relationship between the incentive based profile and the incentive table

Table 10: Properties of Incentive Table

Property	Definition
s4ener:hasScopeType	The relationship between the incentive table and the scope type
s4ener:hasIncentiveType	The relationship between the incentive table and the incentive type
s4ener:hasTierIncentive	The relationship between the incentive table and the tier incentive time series
s4ener:hasTierLowerLimit	The relationship between the incentive table and the tier lower limit time series
s4ener:hasTierUpperLimit	The relationship between the incentive table and the tier upper limit time series

4.2.3.4 Operation Mode Profile

Devices that offer the `s4ener:operationModeProfile` can control the amount of power that they generate and/or consume. 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:OpeartionMode`). Transitions between operation modes can be defined as `s4ener:Transition` with associated timing constraints (`s4ener:Timer`) for transitioning between states. Specific datatype properties can be used to indicate (A) whether an abnormal condition (`s4ener:abnormalConditionOnly`) applies to the `s4ener:OperationMode` and/or the `s4ener:Transition`, (B) the diagnostic label description (`s4ener:hasDiagnosticLabelDescription`) of a `s4ener:OperationMode`, (C) the operation mode factor (`s4ener:hasOpeartionModeFactor`) of a `s4ener:OperationMode`,(D) the transition timestamp (`s4ener:transitionTimestamp`) of a `s4ener:OperationMode`, (E) the transition costs of a `s4ener:Transition`, and (F) the datetime value indicating when the `s4ener:Timer` finishes (`s4ener:finishedAt`). Figure 6 displays the `s4ener:operationModeProfile` and its related entities.

Tables 11-14 summarize the properties that characterize a `s4ener:OperationModeProfile` (table 11), `s4ener:OperationMode` (table 12), `s4ener:Timer` (table 13), and `s4ener:Transition` (table 14).

Table 11: Property of Operation Mode Profile

Property	Definition
s4ener:hasOperationMode	The relationship between the operation mode profile and the operation mode

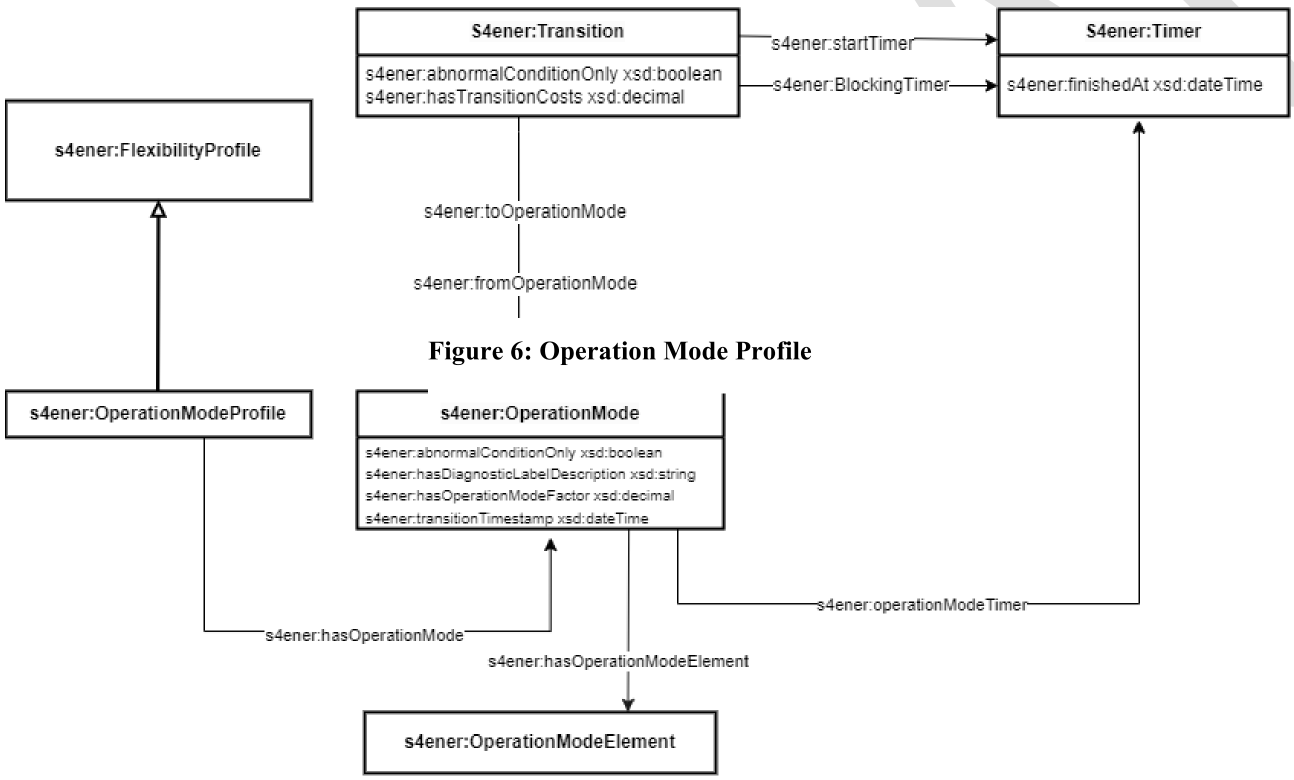


Figure 6: Operation Mode Profile

Table 12: 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:hasDiagnosticLabelDescription	The relationship between the operation mode and the string datatype value of the description of the diagnostic label
s4ener:hasOperationModeFactor	The relationship between the operation mode and the decimal datatype value of the operation mode factor
s4ener:transitionTimestamp	The relationship between the operation mode and the date-time datatype value of the operation mode factor
s4ener:hasOperationModeElement	The relationship between the operation mode and the operation mode element
s4ener:operationModeTimer	The relationship between the operation mode and the timer

Table 13: Propery of Timer

Property	Definition
s4ener:finishedAt	The relationship between the timer and its date-time datatype value

Table 14: Properties of Transition

Property	Definition
s4ener:abnormalConditionOnly	The relationship between the transition and the boolean datatype value indicating whether the transition has abnormal condition
s4ener:hasTransitionCosts	The relationship between the operation mode and the decimal datatype value indicating the transition costs
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:startTimer	The relationship between the transition and the IDs of timers that will be (re)started when this transition is initiated
s4ener:BlockingTimer	The relationship between the transition and the IDs of timers that are bloked

4.2.3.5 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. 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:PowerEnevelope` can be described through the class `s4ener:CommodityQuantity` as presented in figure 7.

SAREF4ENER further specify *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 through the hierarchy of `s4ener:ActiveInactiveState`. A device has nominal power consumption and/or production values (`s4ener:NominalPowerLimit`) when the manufacturers define quantifiable and measurable limits that must not be exceeded. The failsafe values provided by the manufacturers must be given as instances of `saref:measurement`. 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. Ultimately, a `saref:Device` is always *bound to* a `s4ener:ContractualPowerLimit` (which is defined in the contract with the energy provider), *protected by* a `s4ener:NominalPowerLimit` (which is defined in the

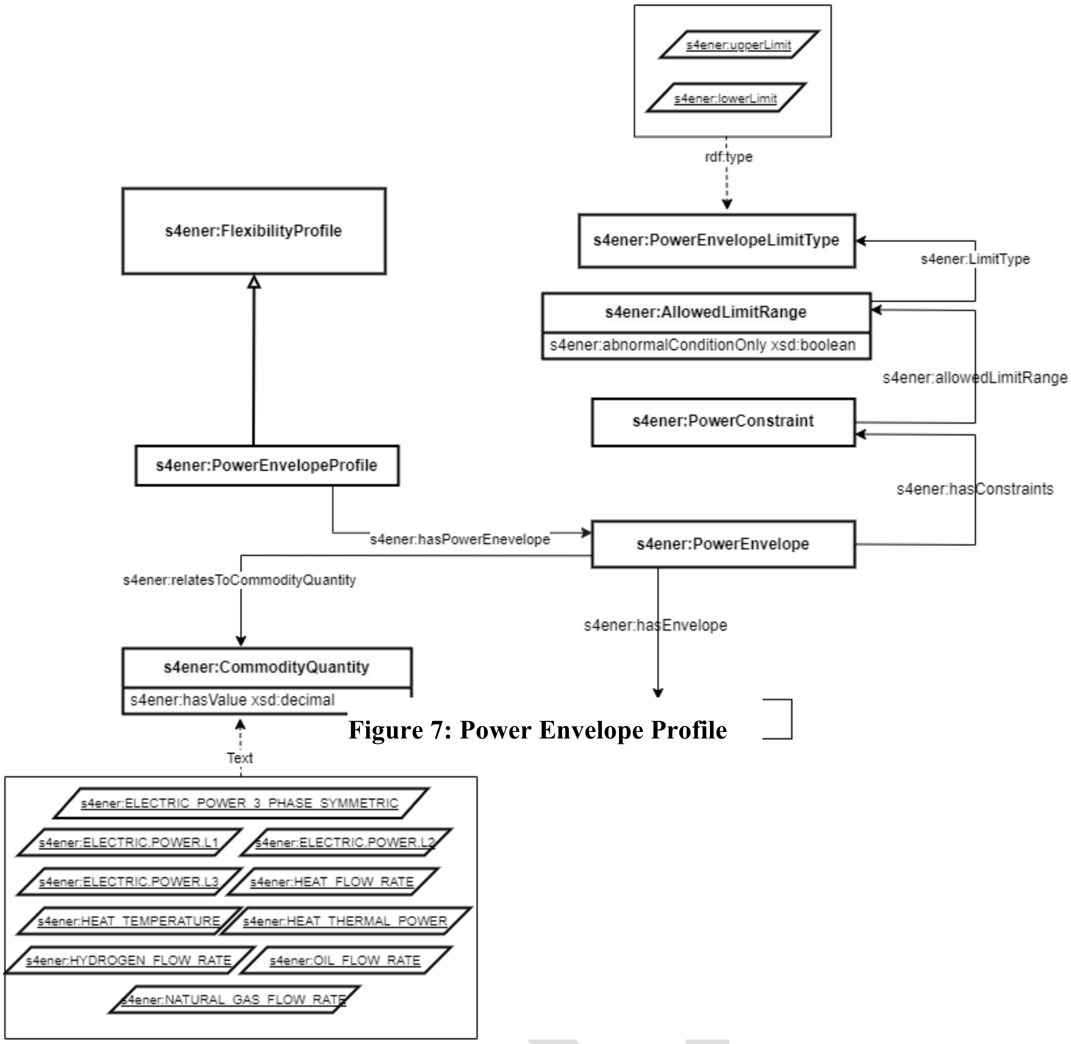


Figure 7: Power Envelope Profile

specification by the manufacturers) and *limited by* a `s4ener:FailsafePowerLimit`. Figure 8 displays the taxonomy and relationships of power limits.

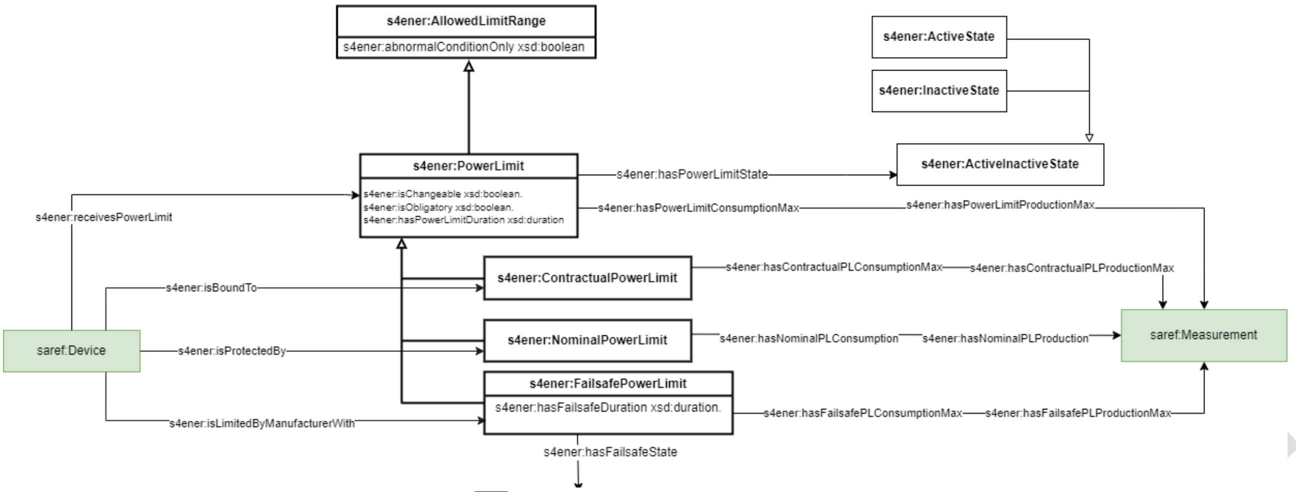


Figure 8: Power Limit(s)

Tables 15- 22 summarize the properties that characterize a `s4ener:PowerEnvelopeProfile` (table 15), `s4ener:PowerEnvelope` (table 16), `s4ener:PowerConstraint` (table 17), and `s4ener:AllowedLimitRange` (table 18), `s4ener:PowerLimit` (table 19), `s4ener:ContractualPowerLimit` (table 20), `s4ener:NominalPowerLimit` (table 21), `s4ener:FailsafePowerLimit` (table 22). Table 23 displays the relationships between `saref:device` and the power limits defined in SAREF4ENER.

Table 15: Propery of Power Envelope Profile

Property	Definition
s4ener:hasPowerEnvelope	The relationship between the power envelope profile and the power envelope

Table 16: Properties of Power Envelope

Property	Definition
s4ener:relatesToCommodityQuantity	The relationship between the power envelope and the commodity quantity
s4ener:hasEnvelope	The relationship between the power envelope and the time series
s4ener:hasConstraints	The relationship between the power envelope and the power constraint

Table 17: Properties of Power Constraint

Property	Definition
s4ener:allowedLimitRange	The relationship between the power constraint and the allowed limit range

DRAFT

Table 18: 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

Table 19: 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:hasPowerLimitDuration	The relationship between the power limit and the duration datatype value indicating the duration of the power limit
s4ener:hasPowerLimitState	The relationship between the power limit and the state (active or inactive) of the power limit
s4ener:hasPowerLimitConsumptionMax	The relationship between the power limit and its maximum consumption value expressed as saref:Measurement
s4ener:hasPowerLimitProductionMax	The relationship between the power limit and its maximum production value expressed as saref:Measurement.

Table 20: Contractual Power Limit

Property	Definition
s4ener:hasContractualPLConsumptionMax	The relationship between the contractual power limit and its maximum consumption value expressed as saref:Measurement
s4ener:hasContractualPLProductionMax	The relationship between the contractual power limit and its maximum production value expressed as saref:Measurement

Table 21: Nominal Power Limit

Property	Definition
s4ener:hasNominalPLConsumption	The relationship between the nominal power limit and its maximum consumption value expressed as saref:Measurement
s4ener:hasNominalPLProduction	The relationship between the nominal power limit and its maximum production value expressed as saref:Measurement

Table 22: Failsafe Power Limit

Property	Definition
s4ener:hasFailsafeDuration	The relationship between the failsafe power limit and the datatype value indicating its duration
s4ener:hasFailsafeState	The relationship between the failsafe power limit and the failsafe state
s4ener:hasFailsafePLConsumptionMax	The relationship between the failsafe power limit and its maximum consumption value expressed as saref:Measurement
s4ener:hasFailsafePLProductionMax	The relationship between the failsafe power limit and its maximum production value expressed as saref:Measurement

DRAFT

Table 23: Relationships between saref:Device and Power Limits

Property	Definition
s4ener:receivesPowerLimit	The relationship between saref:Device and s4ener:PowerLimit
s4ener:isBoundTo	The relationship between saref:Device and s4ener:ContractualPowerLimit
s4ener:isProtectedBy	The relationship between saref:Device and s4ener:NominalPowerLimit
s4ener:isLimitedByManufacturerWith	The relationship between saref:Device and s4ener:FailsafePowerLimit

4.2.3.6 Power Profile Flexibility

4.2.3.6.1 Power Profile

This clause presents the classes of interest for smart energy management. These classes are used to schedule devices in certain modes and preferred times using power profiles to optimize energy efficiency and accommodate the customer's preferences (i.e. use case 2). These classes are `s4ener:PowerProfile`, `s4ener:Alternative`, `s4ener:PowerSequence` and `s4ener:Slot`, which are shown in Figure 4.

A `s4ener:PowerProfile` is a subclass of a `saref:Profile`, i.e. it inherits the properties of the more general `saref:Profile` extending it with additional properties that are specific for SAREF4ENER. The `s4ener:PowerProfile` is used by a `s4ener:Device` to expose the power sequences that are potentially relevant for the CEM. A `s4ener: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 `s4ener:Device`.

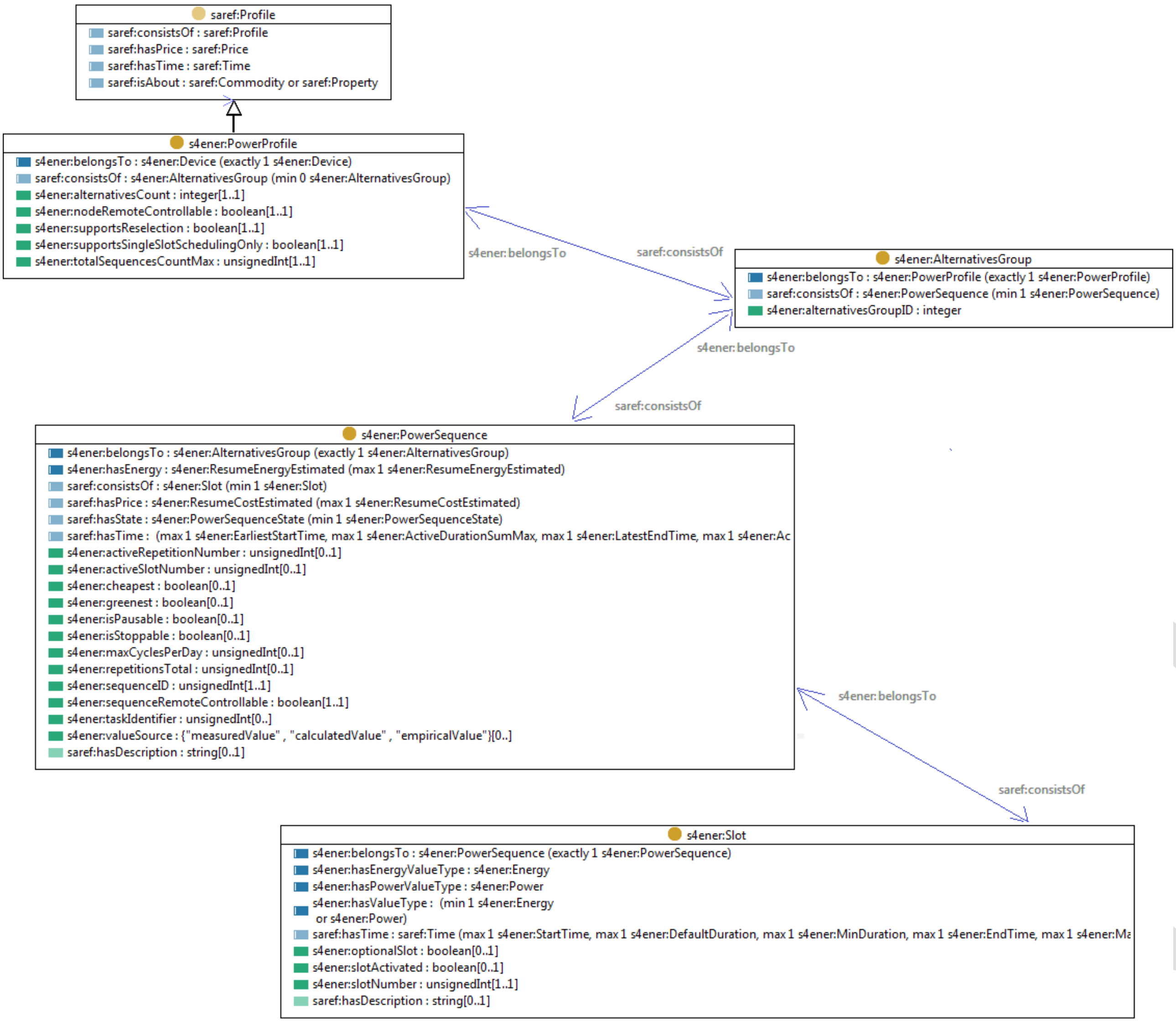


Figure 9: Power Profile and Alternatives Group

Table 3 summarizes the properties that characterize a `s4ener:PowerProfile` and an `s4ener:AternativesGroup`.

Table 3: Properties of a Power Profile and an AlternativesGroup

Property	Definition
s4ener:alternativesGroupID **exactly** 1 xsd:unsignedInt	The endpoint-wide unique identifier for the alternatives group instances provided by a power profile.
s4ener:alternativesCount **exactly** 1 xsd:integer	Number of "alternatives" groups provided by a power profile.
s4ener:nodeRemoteControllable **exactly** 1 xsd:boolean	Whether the device is configured for remote control by the CEM. This refers to the selection chosen by the user on the remote control feature of the device.
s4ener:supportsReselection **exactly** 1 xsd:boolean	Whether the device restricts the number of sequence re-selections by the CEM. If set to TRUE, there is no restriction, i.e. within a given alternative the CEM 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 CEM to select a sequence of an alternative only one time.
s4ener:supportsSingleSlotSchedulingOnly **exactly** 1 xsd:boolean	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 **exactly** 1 xsd:unsignedInt	Total number of sequences supported by the device, i.e. the sum of all power sequences across all alternatives.

4.2.3.6.2 Power Sequence

The `s4ener:AlternativesGroup` described in clause 4.2.3 consists of one or more power sequences (`s4ener:PowerSequence` class) and, inversely, a `s4ener:PowerSequence` belongs to only and exactly one `s4ener:AlternativesGroup`. Figure 5 shows the details of the `s4ener:PowerSequence` class.

DRAFT

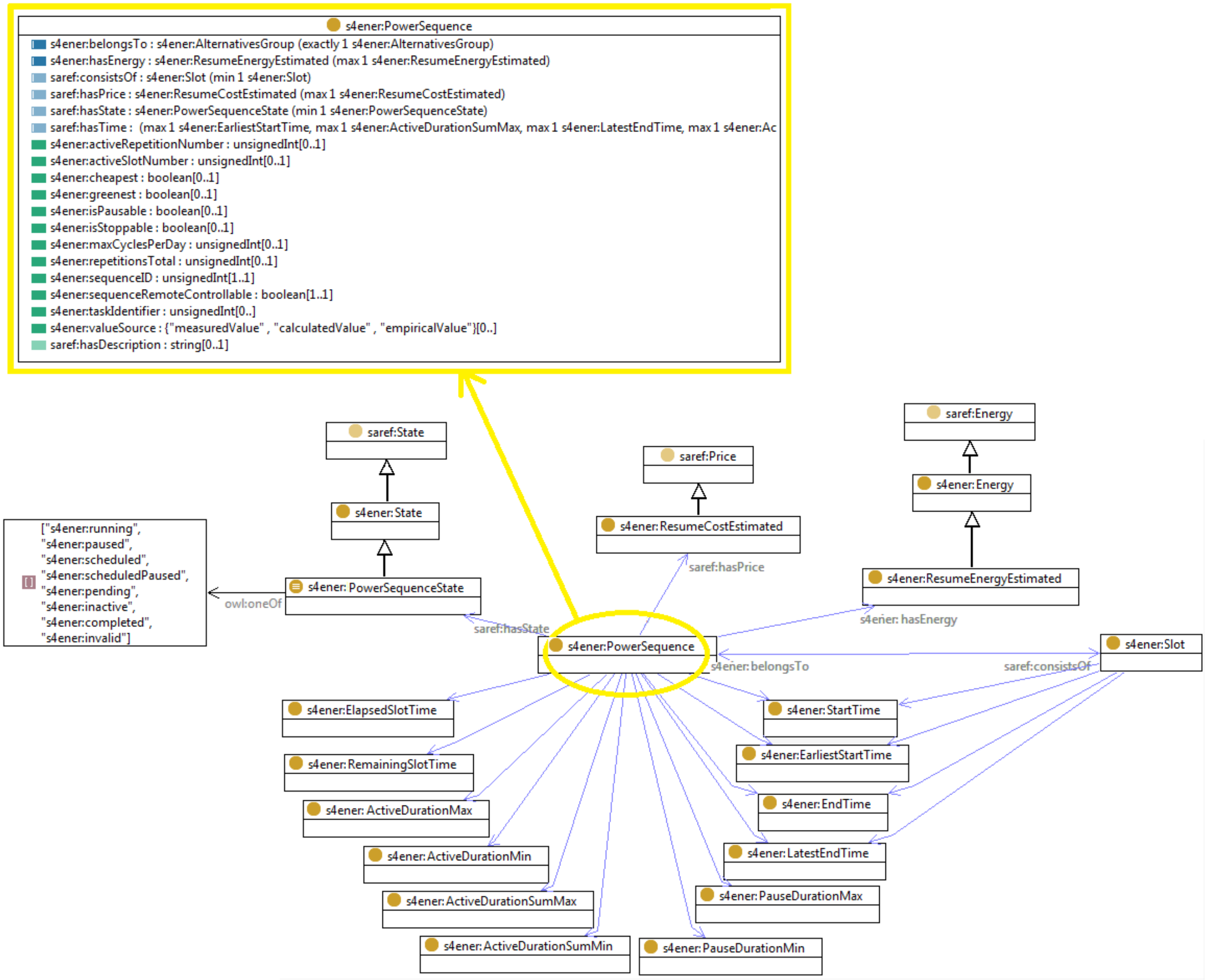


Figure 10: Power Sequence

Table 4 summarizes the properties that characterize a `s4ener:PowerSequence`.

Table 4: Properties of the PowerSequence

Property	Definition
s4ener:sequenceId **exactly** 1 xsd:unsignedInt	An endpoint-wide unique sequence identifier.
saref:hasDescription **max** 1 xsd:string	Textual description for the power sequence.
s4ener:isStoppable **max** 1 xsd:boolean	If the power sequence is stoppable by the CEM, this element is TRUE. Otherwise it SHALL be omitted.
s4eef:isPausable **max** 1 xsd:boolean	If the power sequence is pausable by the CEM, this element is TRUE. Otherwise it SHALL be omitted.
s4ener:taskIdentifier **min** 0 xsd:unsignedInt	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 **max** 1 xsd:unsignedInt	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 **max** 1 xsd:unsignedInt	If s4ener:PowerSequenceState is set to "running" or "paused" this element SHALL contain the currently active slot. Otherwise it SHALL be omitted.
s4ener:cheapest **max** 1 xsd:boolean	If present and set to TRUE, the CEM 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.

Property	Definition
s4ener:greenest **max** 1 xsd:boolean	If present and set to TRUE, the CEM 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 **max** 1 xsd:unsignedInt	The maximum amount of starts that the device allows per day.
s4ener:repetitionsTotal **max** 1 xsd:unsignedInt	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 **exactly** 1 xsd:boolean	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 **min** 0 {"measuredValue", "calculatedValue", "empiricalValue"}	The source (origin/foundation) of the measurement forecasted values for this power sequence. If absent, the source is undefined.
s4ener:hasEnergy **max** 1 s4ener:ResumeEnergyEstimated	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 **max** 1 s4ener:ResumeCostEstimated	The additional costs for the resumption of a device to its normal operation (after a pause).
saref:hasState **min** 1 s4ener:PowerSequenceState	The current state of the power sequence. It can assume one of the following values: 'running', 'paused', 'scheduled', 'scheduled paused', 'pending', 'inactive', 'completed' or 'invalid'.
saref:hasTime **max** 1 s4ener:ActiveDurationMax	The active maximum duration the power sequence can run without interruption.
saref:hasTime **max** 1 s4ener:ActiveDurationMin	The active minimum duration the power sequence can run without interruption.
saref:hasTime **max** 1 s4ener:ActiveDurationSumMax	The active maximum duration the power sequence can run in total (summation of all active times).
saref:hasTime **max** 1 s4ener:ActiveDurationSumMin	The active minimum duration the power sequence runs in total (summation of all active times).
saref:hasTime **min** 1 s4ener:StartTime	The start time of the power sequence. SHALL be present.
saref:hasTime **max** 1 s4ener:EarliestStartTime	SHALL state the earliest possible start time for the whole power sequence.
saref:hasTime **max** 1 s4ener:EndTime	The end time of the power sequence. If the value is available, it SHALL be denoted here. Otherwise the element SHALL be omitted.
saref:hasTime **max** 1 s4ener:LatestEndTime	The latest possible end time for the whole power sequence.
saref:hasTime **max** 1 s4ener:ElapsedSlotTime	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.
saref:hasTime **max** 1 s4ener:RemainingSlotTime	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.
saref:hasTime **max** 1 s4ener:PauseDurationMax	The maximum duration the power sequence can pause after the end of an activity.
saref:hasTime **max** 1 s4ener:PauseDurationMin	The minimum duration the power sequence can pause after the end of an activity.

4.2.3.6.3 Slot

The `s4ener:PowerSequence` described in clause 4.2.4 consists of one or more slots (`s4ener:Slot` class) and, inversely, a `s4ener:Slot belongs` to only and exactly one `s4ener:PowerSequence`. Figure 6 shows the details of the s4ener:Slot class.

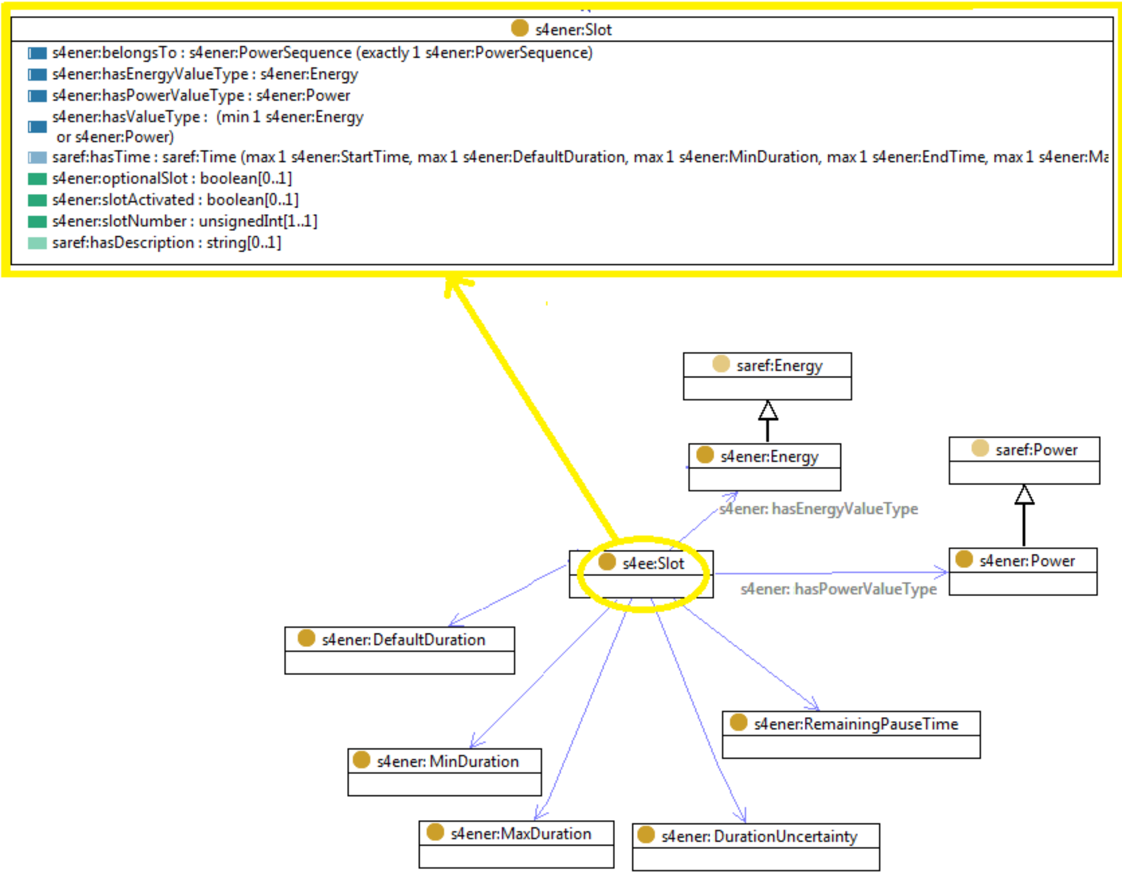


Figure 11: Slot

Table 5 summarizes the properties that characterize a `s4ener:Slot`.

Table 5: Properties of a Slot

Property	Definition
s4ener:slotNumber **exactly** 1	A power sequence Id-wide unique slot identifier.
saref:hasDescription **max** 1	Textual description for the slot.
s4ener:optionalSlot **max** 1	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 **max** 1	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 **min** 1 (s4ener:Energy **or** s4ener:Power)	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.
saref:hasTime **max** 1 s4ener:DefaultDuration	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.
saref:hasTime **max** 1 s4ener:MaxDuration	The maximum supported configuration (if the slot has a configurable duration).
saref:hasTime **max** 1 s4ener:MinDuration	The minimum supported configuration (if the slot has a configurable duration) (see note 3).
saref:hasTime **max** 1 s4ener:DurationUncertainty	The uncertainty of the duration given in the s4ener:Duration class.
saref:hasTime **max** 1 s4ener:StartTime	The start time of the slot. SHALL be present.
saref:hasTime **max** 1 s4ener:EarliestStartTime	SHALL state the earliest possible start time for the slot.
saref:hasTime **max** 1 s4ener:EndTime	The end time of the slot. The following equation SHALL apply: EndTime - StartTime = DefaultDuration.
saref:hasTime **max** 1 s4ener:LatestEndTime	The latest possible end time for the slot.
saref:hasTime **max** 1 s4ener:RemainingPauseTime	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. NOTE 3: This element applies to the first repetition of the slot number only.	

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 and power curtailing scenarios (i.e. use case 4). The classes of interest are `s4ener:LoadControlEventData`, `s4ener:LoadControlEventAction`, `s4ener:LoadControlStateData` and `s4ener:LoadControlState`, as shown in Figure 7.

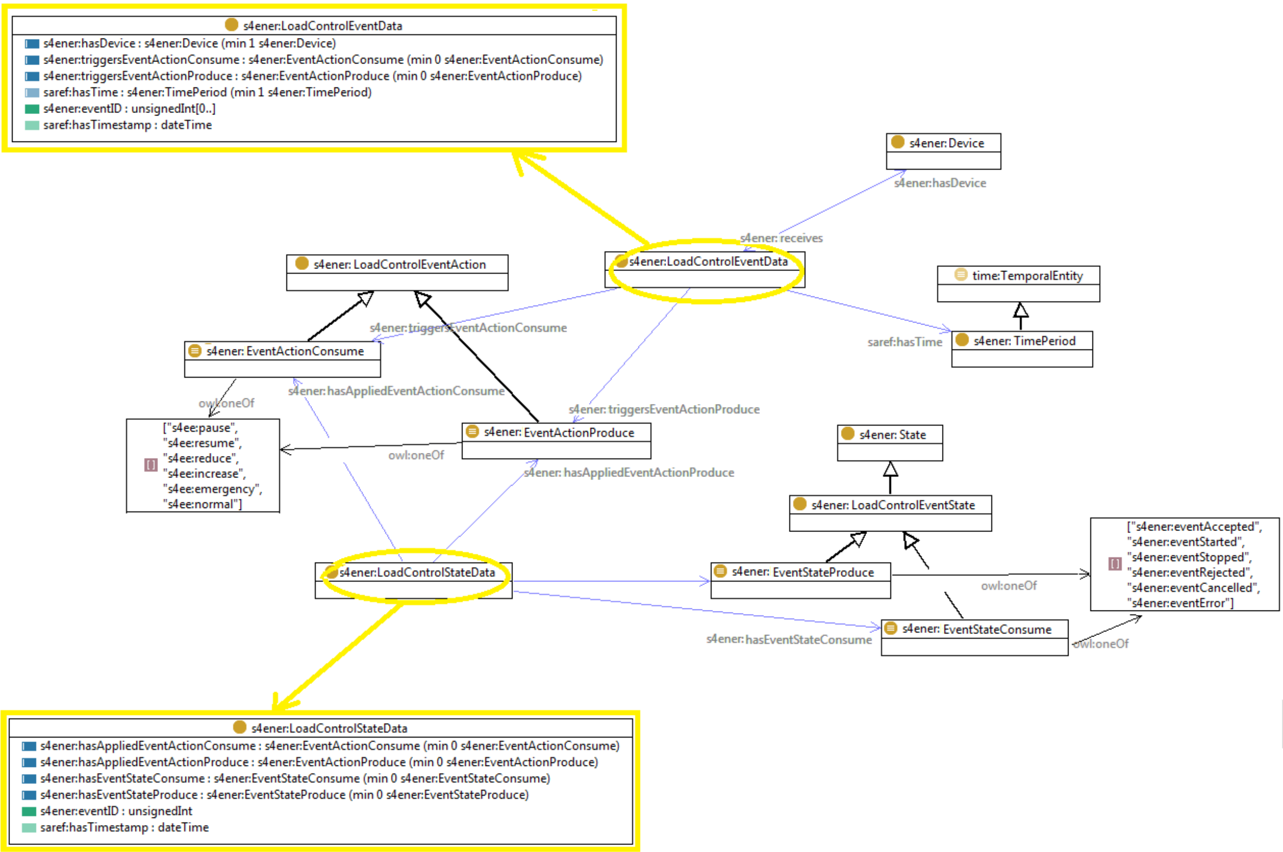


Figure 12: Load Control

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 section presents how flexibility requests can be modelled in SAREF4ENER. Flexibility requests can be defined by using `s4ener:FlexRequest`. Flexibility requests can *include* a `s4ener:IncentiveTable`, `s4ener:FlexibilityProfile`, `s4ener:TimeSeries` and `s4ener:Datapoint`. `s4ener:FlexRequest` can be produced by an agent (`foaf:Agent`) or device (`saref:Device`) and be sent to either an agent or device.

s4ener:FlexRequest always has an effective period and a creation time expressed through the Time ontology as presented in figure 10.

Figure 13: Flexibility Request

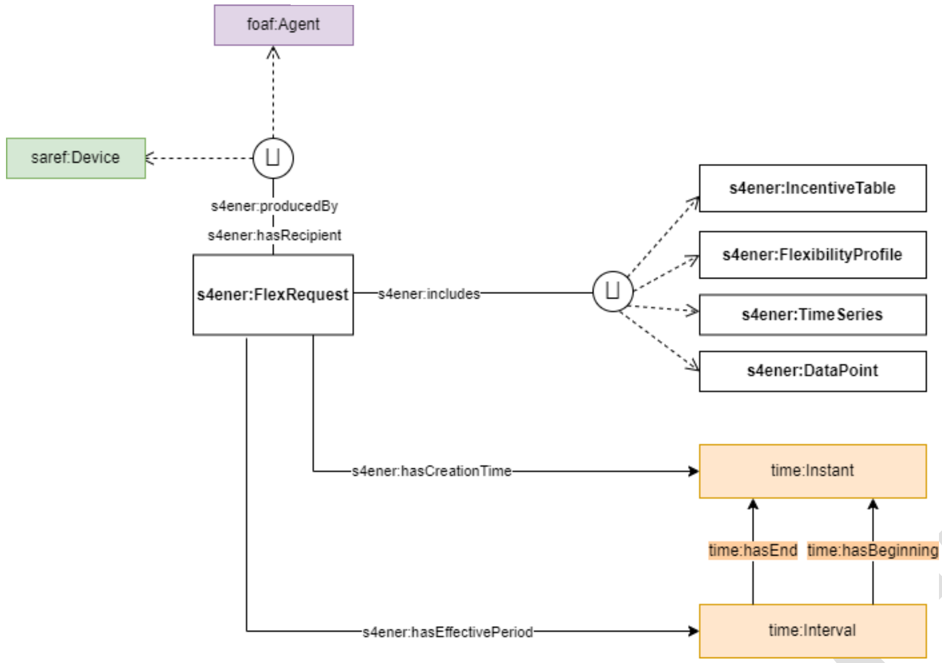


Table 25 summarizes the properties that characterize a s4ener:FlexRequest.

Table 25: 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, time-series and datpoint included in the flexibility offer
s4ener:hasCreationTime	The relationship between the flexibility request and its creation time expressed as time:istant
s4ener:hasEffectivePeriod	The relationship between the flexibility request and its creation time expressed as time:interval

4.2.5.2 Flexibility Offer

This section presents how flexibility requests can be modelled in SAREF4ENER. Flexibility offers can be defined by using s4ener:FlexOffer. Flexibility offers can *include* a s4ener:IncentiveTable, s4ener:FlexibilityProfile, s4ener:TimeSeries and s4ener:Datapoint. s4ener:FlexOffer can be produced by an agent (foaf:Agent) or device (saref:Device) and be sent to either an agent or device. Flexibility offers relate to flexibility requests (`s4ener:FlexRequest`). `s4ener:FlexOffer` always has an effective period and a creation time expressed through the Time ontology as displayed in figure 9.

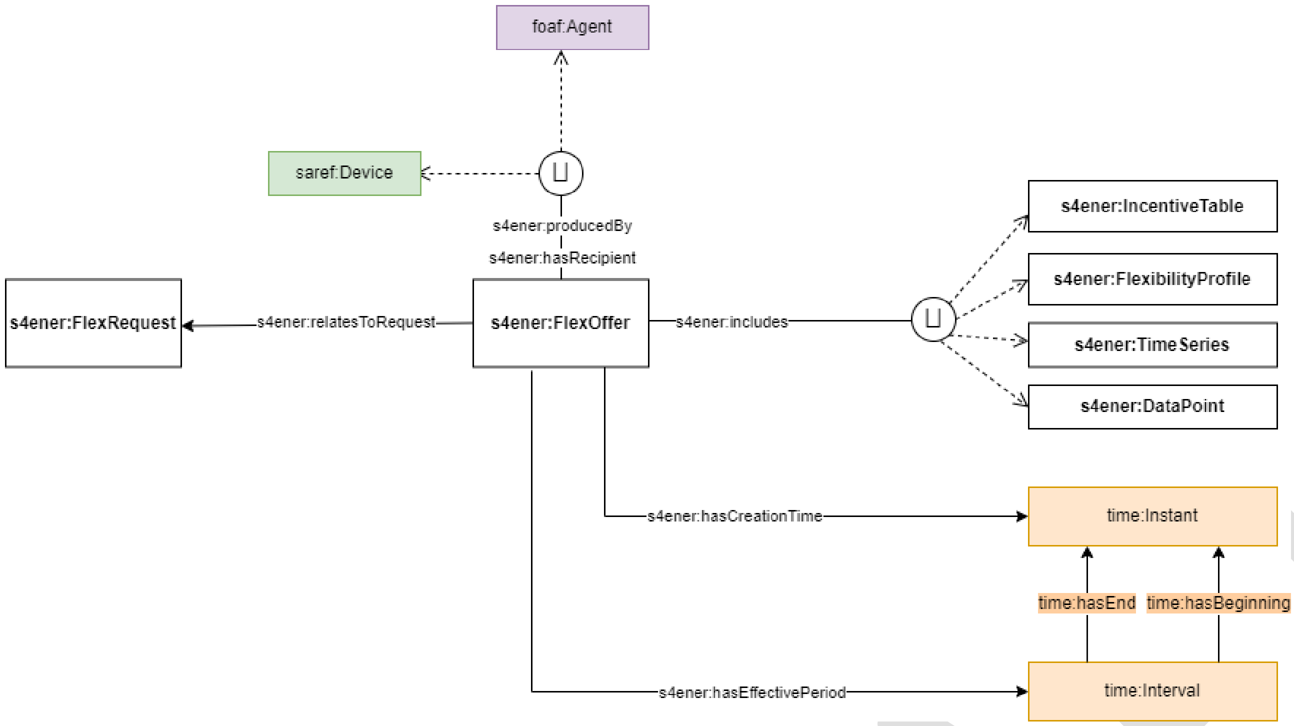


Figure 14: Flexibility Offer

Table 24 summarizes the properties that characterize a `s4ener:FlexOffer`.

Table 24: 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, time-series and datpoint included in the flexibility offer
s4ener:hasCreationTime	The relationship between the flexibility offer and its creation time expressed as time:istant
s4ener:hasEffectivePeriod	The relationship between the flexibility offer and its creation time expressed as time:interval

4.2.5.3 Flexibility Instruction

This section presents how flexibility instruction can be modelled in SAREF4ENER. 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`). `s4ener:FlexInstruction` can have an execution time, period of validity, istructionID defiened 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`). `s4ener:Flexinstruction` can be produced by an agent (`foaf:Agent`) or device (`saref:Device`) and be sent to either an agent or device. `s4ener:FlexInstruction` always has an effective period and a creation time expressed through the Time ontology. Every flexibility instruction has an instruction status as presented in figure 11.

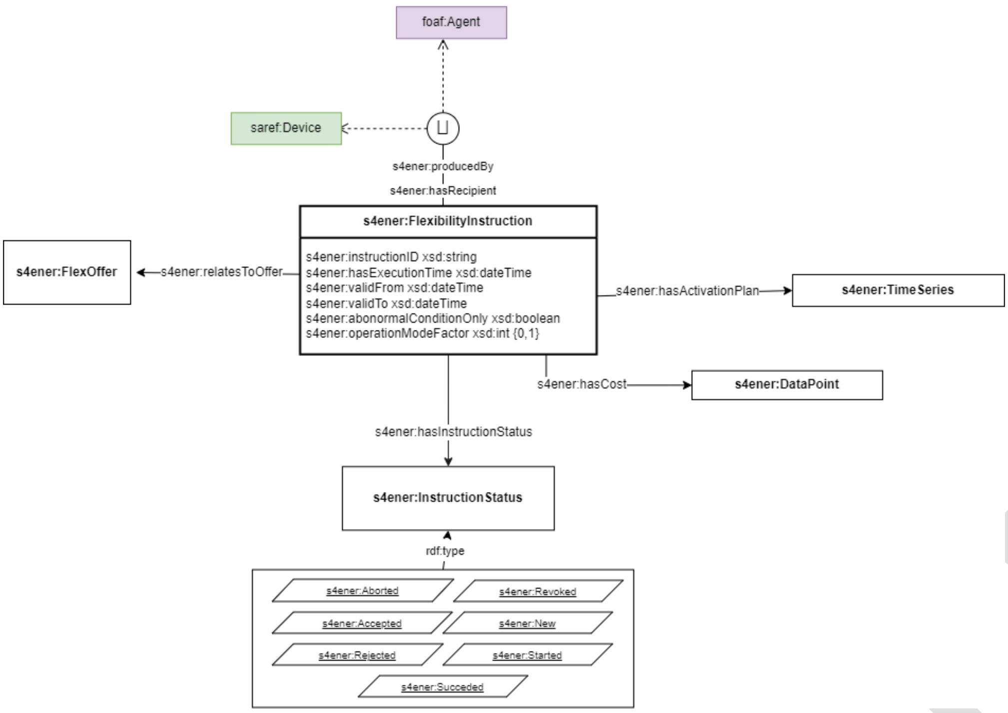


Figure 15: Flexibility Instruction

Table 26 summarizes the properties that characterize a `s4ener:FleInstruction`.

DRAFT

Table 26: 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 flexibilityinstruction and the foaf:agent or saref:Device to which the flexibility request is directed
s4ener:hasCost	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:instructionID	The relationship between the flexibility instruction and the string datatype value of its ID.
s4ener:hasExecutionTime	The relationship between the flexibility instruction and the dateTime datatype value of its execution time.
s4ener:validFrom xsd:dateTime	The relationship between the flexibility instruction and the dateTime datatype value expressing the starting time of its validity.
s4ener:validTo xsd:dateTime	The relationship between the flexibility instruction and the dateTime datatype value expressing the ending time of its validity.
s4ener:abnormalConditionOnly	The relationship between the flexibility instruction and the boolean datatype value indicating whether the power constraint has an abnormal condition
s4ener:operationModeFactor	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

4.3 Instantiating SAREF4ENER

We will add here examples of how to instantiate with real data the various flexibility profiles and flexibility request/offer/instruction. We will likely use some subclauses for describing different examples. 4.4 Observations

The extension for the energy domain presented in the present document was originally called SAREF4EE, since it was created for the Energy@Home and EEBus associations [i.1]. However, in the present document the extension has been renamed to SAREF4ENER according to the naming convention for SAREF extensions adopted in the ETSI TR 103 411 [i.4] (i.e. SAREF4XXXX, where XXXX are letters that describe the domain for which the extension was created).

The present document describes the concepts for the use case 2 (smart energy management), and the use case 4 (representation of events in case of direct load management and power curtailing) elaborated in clause 5.1.1 in ETSI TR 103 411 [i.4]. However, the present document does not include the concepts for the use case 1 (exchange configuration information of devices in order to connect to each other) and the use case 3 (monitor and control the start and status of the appliances), since these concepts at the time of publication are under discussion between the Energy@Home and EEBus associations and NOT yet included in the SPINE specification in [1]. For the sake of completeness, these concepts are included in the annex B of the present document as informative, but they are NOT part of the current release of SAREF4ENER.

Annex A (informative): Approach

The approach that was followed to create SAREF4ENER is a combination of bottom-up and top-down steps, as shown in Figure A.1. The (bottom-up) starting point is given by the two existing data models of E@H (an UML class diagram) and EEBus (the XSDs specification). These two data models focus on similar concepts, such as the concept of "power profile", but they use different terminologies. For example, E@H defines "power profiles", "modes" and "phases", while EEBus refers 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 [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.

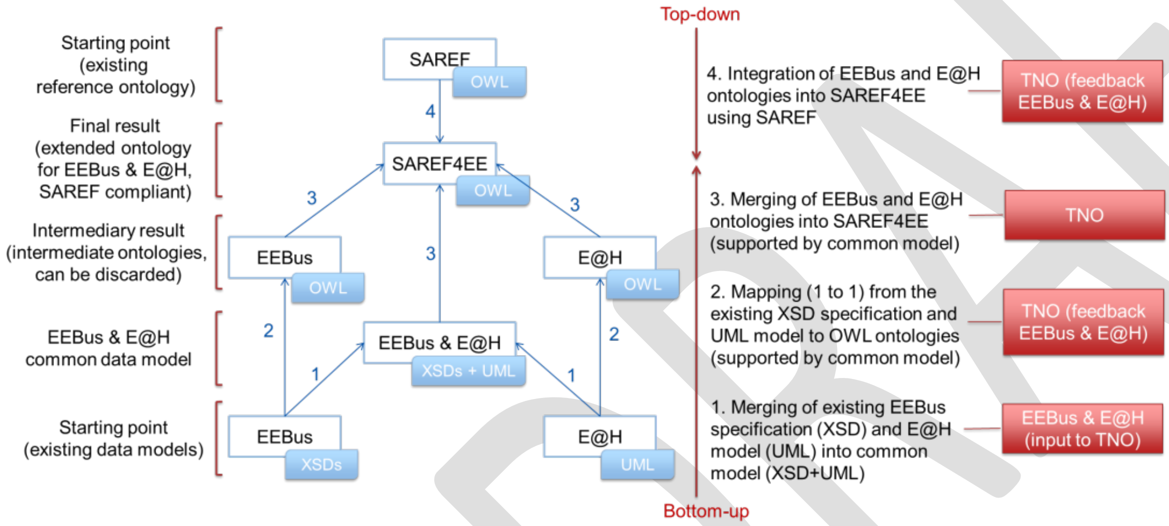


Figure A.1: Approach

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 A.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.

NOTE: TopBraid Composer Maestro Edition™ is an example of a suitable product available commercially. This information is given for the convenience of users of the present document and does not constitute an endorsement by ETSI of this product.

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 A.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 A.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 were reused and specialized where possible, while SAREF 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 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 have specific properties for EEbus and E@H that do not apply to all SAREF 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.

NOTE: Available at https://sites.google.com/site/smartappliancesproject/deliverables.

DRAFT

Annex D (informative): Change History

Date	Version	Information about changes
May 2023	V1.1.3	• Added Flexibility Profiles section • Added Flexibility Communication section

History

Document history		
V1.1.1	January 2017	Publication
V1.1.2	May 2020	Publication
V1.1.3	May 2023	Early draft

DRAFT