

SMART FACTORY WEB ONTOLOGY SPECIFICATION

Version 1.0

SFW ONTOLOGY SPEC

Version 1.0

February 23, 2023

Authors

Dr. Florian Patzer

Boris Schnebel

Felix Schöppenthau

Dr. Kym Watson

Fraunhofer-Institut für Optronik, Systemtechnik und Bildauswertung, IOSB
in Karlsruhe

Inhalt

1 SFW Capability Model Top Level – normative (2.0.5) **5**

- 1.1 Concepts
5
- 1.1.1 Entity
5
- 1.1.2 Asset
5
- 1.1.3 Process
6
- 1.1.4 Capability
7
- 1.1.5 ProductionResource
7
- 1.1.6 Product
8
- 1.1.7 ProductApplication
8
- 1.1.8 Factory
9
- 1.1.9 PhysicalFactory
9
- 1.1.10 VirtualFactory
9
- 1.1.11 Enterprise
10
- 1.1.12 Property
10
- 1.1.13 SemanticReference
11
- 1.1.14 ProductClass
11
- 1.1.15 User
12
- 1.2 Object Properties
12

2 SFW Capability Model Mid-Level – normative (0.2.1) **15**

- 2.1 Concepts
15
- 2.1.1 HumanResource
15
- 2.1.2 Machine
16
- 2.1.3 Supply Chain
16
- 2.1.4 Supply Chain Element
17
- 2.1.5 ProductPassport
17
- 2.1.6 tlm:Product
17

2.2 Properties
18

3 SFW Geo – normative (0.1.0)
20

3.1 Concepts
20

3.1.1 Location
20

An Asset is a physical or logical object (Entity) owned by or under the custodial duties of an organization, having either a perceived or actual value to the organization [IEC62443-1-1]. *Inherits relations from Entity.*

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
contains	Asset	An asset can contain other assets or even consist of them.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

1.1.3 Process

A Process is a set of interacting operations in a system by which matter, energy or information is transformed, transported or stored¹. A Process may contain ProcessEntity's as input and output. A Process is a set of process steps or a single process step. *Inherits relations from Entity.*

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
hasInput	Product	A Process has an input Product or ProductApplication (e.i product + quantity and possibly other Properties).
hasOutput	Product	A Process has an output Product or ProductApplication (e.i product + quantity and possibly other Properties).
realizes	Capability	A realizes B, if A is a more concrete or realistic version of A.
requires	Capability	A Process requires a Capability, if the Capability has to be provided in order to conduct the process.
uses	ProductionResource	A Process uses a ProductionResource, if the Production Resource is utilized during the conduction of the Process.
contains	Process	A Process can contain another process as sub process
containedIn	Process	A Process can be contained in another process as a sub process
containedIn	Enterprise	A Process must be contained in an enterprise.
containedIn	Factory	A Process must be contained in a factory.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:comment	rdf:PlainLiteral	Each Process can have a human readable description (at least in English (language=en)).

¹ IEV ref351-42-33 (aus PI 4.0 Capability Workinggroup)

1.1.4 Capability

A *Capability* is an Entity which represents a designated function to achieve an effect in the physical or virtual world. *Inherits relations from Entity.*

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
generalizes	Capability	A Capability generalizes or specializes another capability to build a terminological hierarchy.
specializes	Capability	C.f. generalizes.
realizedBy	Process	A is realized by B, if B is a more concrete or realistic version of A.
providedBy	ProductionResource	A Capability is provided by a ProductionResource, if the ProductionResource is capable to perform actions the capability describes and implies.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each Capability must have a human readable label (at least in English (language=en)) if it is not linked to a SemanticReference.
rdfs:comment	rdf:PlainLiteral	Each Capability must have a human readable description (at least in English (language=en)) if it is not linked to a SemanticReference.

1.1.5 ProductionResource

A function unit needed to perform required operations. *Inherits relations from Entity and Asset.*

A ProductionResource is an Asset that provides the Capabilities which are required for performing a particular process step on a product.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
provides	Capability	A ProductionResource provides a capability, if the ProductionResource is capable to perform actions the capability describes and implies.
usedBy	Process	A ProductionResource is used by a Process, if the Process utilizes the ProductionResource during the conduction of the Process.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each ProductionResource requires a human readable label (at least in English (language=en)).

rdfs:comment	rdf:PlainLiteral	Each ProductionResource requires a human readable description (at least in English (language=en)).
--------------	------------------	--

1.1.6 Product

A Product is a material good or an (immaterial) service offering which is an outcome (OutputProduct) or an input (InputProduct) of a Process. *Inherits relations from Entity and Asset.*

Data Properties		
Name	Range	Description
Object Properties		
Name	Range	Description
specializes	ProductClass	Relation to describe that a Product belongs to a ProductClass.
contains	ProductApplication	A Product can contain a ProductApplication as part of its bill of materials.
contains	Product	A Product can contain a Product derived from its bill of materials.
containedIn	Product	A Product can contain a Product derived from its bill of materials.
Annotations		
Name	Range	Description
rdfs:label	rdf:PlainLiteral	Each Product requires a human readable label or name (at least in English (language=en)).
rdfs:comment	rdf:PlainLiteral	Each Product can have a human readable description (at least in English (language=en)).

1.1.7 ProductApplication

A ProductApplication is a container for a product, a quantity and possible other properties. It describes meta data of a specific product usage, e.g. as input product for a process or as part of the bill of materials of another product (but never both!). *Inherits relations from Entity.*

Data Properties		
Name	Range	Description
Object Properties		
Name	Range	Description
has	Product	A product application must contain a product. (has is used to omit transitive expansion)
containedIn	Product	A Product can contain a ProductApplication as part of its bill of materials.
Annotations		
Name	Range	Description

1.1.8 Factory

A Factory is either a physical or a virtual production plant acting as container for ProductionResource's and Products. It has the child concepts VirtualFactory and PhysicalFactory. *Inherits relations from Entity and Asset.*

Data Properties		
Name	Range	Description
Object Properties		
Name	Range	Description
contains	ProductionResource	Relation describing the affiliation of a factory to a production resource.
contains	Product	Relation describing the affiliation of a factory to a product offered or produced there.
contains	Process	Relation describing the affiliation of a factory to a process conducted in that factory.
Annotations		
Name	Range	Description
rdfs:label	rdf:PlainLiteral	Each Factory can have a human readable label or name (at least in English (language=en)).
rdfs:comment	rdf:PlainLiteral	Each Factory can have a human readable description (at least in English (language=en)).

1.1.9 PhysicalFactory

A PhysicalFactory is a physical production plant acting as container for ProductionResource's and Products. *Inherits relations from Entity, Asset and Factory.*

Data Properties		
Name	Range	Description
Object Properties		
Name	Range	Description
Annotations		
Name	Range	Description
rdfs:label	rdf:PlainLiteral	Each Factory can have a human readable label or name (at least in English (language=en)).
rdfs:comment	rdf:PlainLiteral	Each Factory can have a human readable description (at least in English (language=en)).

1.1.10 VirtualFactory

A VirtualFactory is a container for ProductionResource's and Products which cannot be associated to a physical Factory. *Inherits relations from Entity, Asset and Factory.*

Data Properties		
Name	Range	Description

Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

1.1.11 Enterprise

An Enterprise is an organization, a company or a business containing Factory's. *Inherits Properties from Entity and Asset.*

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
contains	Factory	Relation describing the affiliation of an enterprise to a factory.
contains	ProductionResource	Relation describing the affiliation of a factory to a production resource.
contains	Product	Relation describing the affiliation of a factory to a product offered or produced there.
contains	Process	Relation describing the affiliation of a factory to a process conducted in that factory.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each Enterprise requires a human readable label or name (at least in English (language=en)).
rdfs:comment	rdf:PlainLiteral	Each Enterprise can have a human readable description (at least in English (language=en)).

1.1.12 Property

A Property is an attribute of an Entity which might - but does not need to have - data.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
value	any	A property can contain data. Data consists of the actual value and a reference to its semantic. The value and the semantic reference are part of the property individual. If BAMB is used to describe the data's semantic, a semantic reference of the property refers to the respective aspect model. Note that the property can contain other semantic references as well.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
hasSemantic	SemanticReference	C.f. value. This relation is used to refer to an external semantic for a Property.
Annotations		

<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each Property must have a human readable label (at least in English (language=en)) if it is not linked to a SemanticReference.
rdfs:comment	rdf:PlainLiteral	Each Property must have a human readable description (at least in English (language=en)) if it is not linked to a SemanticReference.

1.1.13 SemanticReference

A reference to an arbitrary globally uniquely identifiable semantic.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
identifier	xsd:string	A semantic reference needs an identifier. This would be an iri within an ontology, an eClass identifier or any other kind of globally unique semantic identifier.
sourceUri	xsd:AnyUri	The URI of the endpoint the (sub-)semantic can be found at.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each semantic reference can have a human readable label or name which should be used in frontends to represent the semantic in one word (at least in English (language=en)).
rdfs:comment	rdf:PlainLiteral	Each semantic reference can have a human readable description (at least in English (language=en)).

1.1.14 ProductClass

A ProductClass is a group of products which share characteristics.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
hasSemantic	SemanticReference	This relation is used to refer to an external semantic for a ProductClass.
generalizes	Product	This relation is used to express the affiliation of a Product to a ProductClass
generalizes	ProductClass	This relation expresses that a ProductClass is a parent of another ProductClass.
specializes	ProductClass	This relation expresses that a ProductClass is a child of another ProductClass.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

rdfs:label	rdf:PlainLiteral	Each product class must have a human readable label (at least in English (language=en)) if it is not linked to a SemanticReference.
rdfs:comment	rdf:PlainLiteral	Each product class must have a human readable description (at least in English (language=en)) if it is not linked to a SemanticReference.

1.1.15 User

A user within the SFW.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
gender	xsd:byte	The gender of a person: 0=male, 1=female, 2=diverse
firstname	xsd:string	The user's firstname.
surname	xsd:string	The user's surname.
mail	xsd:string	The user's mail address.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

1.2 Object Properties

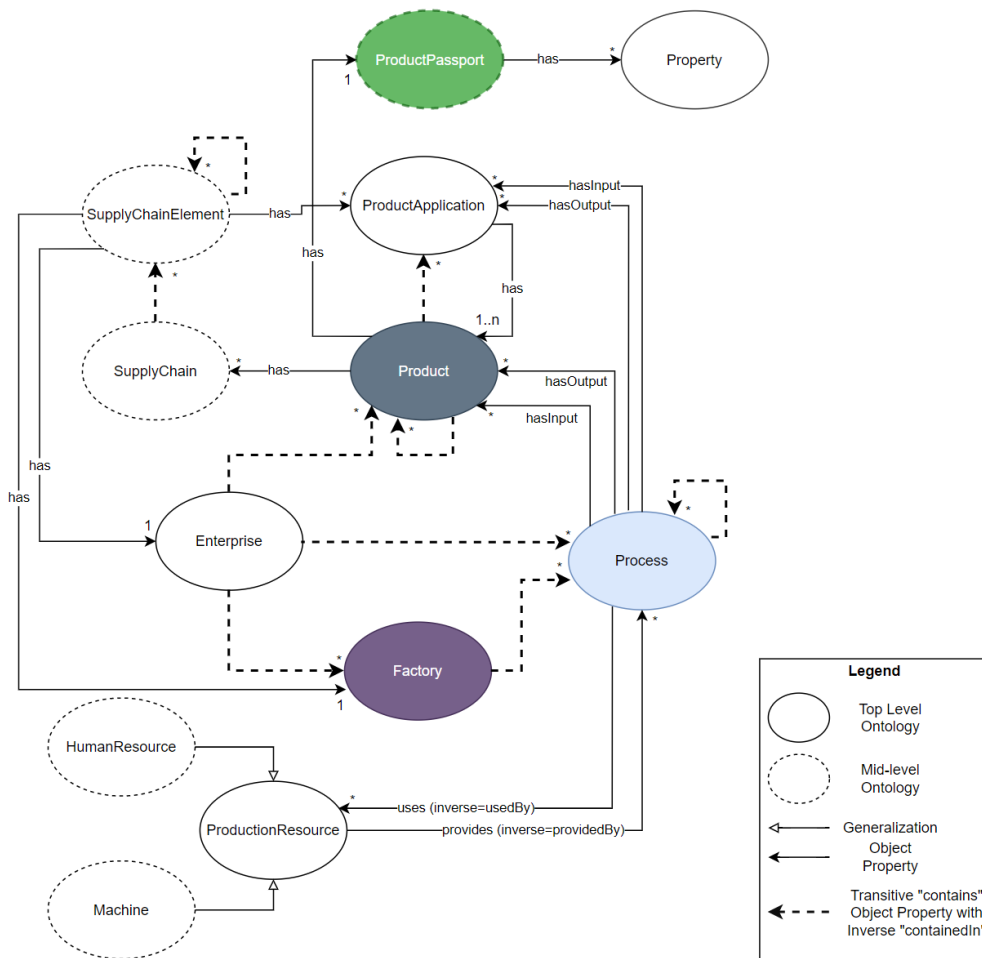
All Object Properties				
<i>Name</i>	<i>Domain</i>	<i>Range</i>	<i>Description</i>	<i>Characteristics</i>
generalizes	any	any	Generalization on individual level.	transitive, inverseOf=specializes
specializes	any	any	Specialization on individual level.	transitive, inverseOf=generalizes
has	any	any	General relation for a thing has a thing. Thus, to define a directed association.	
contains	any	any	A contains B if B is a part of A (or A is a container for B).	transitive, inverseOf=containedIn
containedIn	any	any	A is contained in B if A is a part of B (or B is a container for A)	transitive, inverseOf=contains

hasInput	Process	Product or ProductApplication	A Process has an input Product or ProductApplication.	
hasOutput	Process	Product or ProductApplication	A Process has an output Product or ProductApplication.	
hasSemantic	Entity or Property or ProductClass	SemanticReference	This relation is used to refer to an external semantic for an Entity or Property.	
provides	ProductionResource	Capability or Process	A ProductionResource provides a capability, if the ProductionResource is capable to perform actions the capability describes and implies. A ProductionResource provides a Process, if the ProductionResource provides a Capability which is realized by that Process.	inverseOf=providedBy
realizedBy	Capability	Process	A is realized by B, if B is a more concrete or realistic version of A.	inverseOf=realizes
realizes	Process	Capability	A realizes B, if A is a more concrete or realistic version of A.	inverseOf=realizedBy
requires	Process	Capability	A Process requires a Capability, if the Capability has to be provided in order to conduct the process.	

provided By	Capability or Process	ProductionResource	A Capability is provided by a ProductionResource, if the ProductionResource is capable to perform actions the capability describes and implies. A Process is provided by a ProductionResource, if the ProductionResource provides a Capability which is realized by that process.	inverseOf=provided
uses	Process	ProductionResource	A Process uses a ProductionResource, if the ProductionResource is utilized during the conduction of the Process.	inverseOf=usedBy
usedBy	ProductionResource	Process	A ProductionResource is used by a Process, if the ProductionResource is utilized during the conduction of the Process.	inverseOf=uses

SFW Capability Model Top Level –
normative (2.0.5)

2 SFW Capability Model Mid-Level – normative (0.2.1)



References Top Level Model with Prefix *tlm*. The ontology adds, amongst others an externalIdentifier Property which is used by applications to add their internal ids to be able to map between the individuals and their internal data representations. This property can be used for all individuals, however it will probably sufficient to use it for Entities, since semantic references can be used for that purpose as well, e.g. for product classes or capabilities.

2.1 Concepts

2.1.1 HumanResource

A HumanResource is a human ProductionResource and inherits all properties from *tlm*:ProductionResource.

Data Properties		
Name	Range	Description

certificate	tlm:Property	A HumanResource can have a certificate to express a certain expertise.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each HumanResource requires a human readable label or name e.g. a human resource identifier (at least in English (language=en)).

2.1.2 Machine

A Machine is a ProductionResource which represents a physical system using power to apply forces and control movement to perform an action¹ and inherits all properties from ProductionResource

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
externalIdentifier	xsd:string	A machine can have an arbitrary external identifier.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:label	rdf:PlainLiteral	Each Machine requires a human readable label or name e.g. a machine identifier (at least in English (language=en)).

2.1.3 Supply Chain

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
externalIdentifier	xsd:string	A supply chain can have an arbitrary external identifier.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
tlm:contains	SupplyChainElement	A supply chain must have at least one supply chain element.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:comment	rdf:PlainLiteral	Each supply chain can have a human readable description (at least in English (language=en)).

¹ <https://en.wikipedia.org/wiki/Machine>

2.1.4 Supply Chain Element

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
externalIdentifier	xsd:string	A supply chain element can have an arbitrary external identifier.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
tlm:has	tlm:Enterprise	A supply chain element is related to an enterprise.
tlm:has	tlm:PhysicalFactory	A supply chain element must be related to exactly one physical factory.
tlm:contains	SupplyChainElement	A supply chain element can have child supply chain elements to reference the sub supply chain.
tlm:has	Product	A supply chain element can have an arbitrary amount of products indicating they all come from the supply chain element's factory.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>
rdfs:comment	rdf:PlainLiteral	Each supply chain element can have a human readable description (at least in English (language=en)).

2.1.5 ProductPassport

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
tlm:identifier	xsd:string	Unique product identifier as required by regulation https://environment.ec.europa.eu/publications/proposal-ecodesign-sustainable-products-regulation_en
tlm:has	tlm:Property	A ProductPassport can have arbitrary properties.
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

2.1.6 tlm:Product

Refines tlm:Product.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Object Properties		

<i>Name</i>	<i>Range</i>	<i>Description</i>
Tlm:has	ProductPassport	A Product can have a ProductPassport.
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>

SFW Capability Model Mid-Level –
normative (0.2.1)

2.2 Properties

All Data Properties				
<i>Name</i>	<i>Domain</i>	<i>Range</i>	<i>Description</i>	<i>Characteristics</i>
externalIdentifier	any	xsd:string	An identifier which is used for arbitrary external use cases, such as an asset id from an asset management tool of a supplier, which helps the supplier to link the onboarded information of an asset with the information in the asset management tool.	
All Object Properties				
certificate	any	tIm:Property	A HumanResource can have a certificate to express a certain expertise.	
hasAuxiliaryMaterial	tIm:Process	tIm:Product or tIm:ProductApplication	A process can have an arbitrary number of auxiliary materials as input.	subProperty Of= tIm:hasInput
hasOperatingMaterial	tIm:Process	tIm:Product or tIm:ProductApplication	A process can have an arbitrary number of operating	subProperty Of= tIm:hasInput

			materials as input.	
hasPreliminaryProduct	tlm:Process	tlm:Product or tlm:ProductApplication	A process can have an arbitrary number of preliminary products as input.	subPropertyOf= tlm:hasInput
hasRawMaterial	tlm:Process	tlm:Product or tlm:ProductApplication	A process can have an arbitrary number of raw materials as input.	subPropertyOf= tlm:hasInput
hasByProduct	tlm:Process	tlm:Product or tlm:ProductApplication	A process can have an arbitrary number of by-products as output.	subPropertyOf= tlm:hasOutput
hasEndProduct	tlm:Process	tlm:Product or tlm:ProductApplication	A process can have an arbitrary number of end-products as output.	subPropertyOf= tlm:hasOutput
hasWasteProduct	tlm:Process	tlm:Product or tlm:ProductApplication	A process can have an arbitrary number of waste products as output.	subPropertyOf= tlm:hasOutput

3 SFW Geo – normative (0.1.0)

3.1 Concepts

3.1.1 Location

A Location is a container for location relevant information like addresses and geo coordinates. It is used to be referenced from an Asset.

Data Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
city	xsd:string	
country	xsd:string	
street	xsd:string	
streetNumber	xsd:string	
zipcode	xsd:decimal	
latitude	xsd:double	
longitude	xsd:double	
Object Properties		
<i>Name</i>	<i>Range</i>	<i>Description</i>
Annotations		
<i>Name</i>	<i>Range</i>	<i>Description</i>