SLOVENSKI STANDARD
SIST-TP CEN/TR 17859:2022

01-september-2022

Jezik za storitve modeliranja

Service Modelling Language

Sprache zur Modellierung von Dienstleistungen

Langage de Modélisation de Service

Ta slovenski standard je istoveten z: **CEN/TR 17859:2022**

ICS:

35.060	Jeziki, ki se uporabljajo v informacijski tehniki in tehnologiji	Languages used in information technology

SIST-TP CEN/TR 17859:2022 **en,fr,de**

TECHNICAL REPORT

RAPPORT TECHNIQUE

TECHNISCHER REPORT

CEN/TR 17859

July 2022

ICS 35.240.50

English Version

Service Modelling Language

Langage de Modélisation de Service

Sprache zur Modellierung von Dienstleistungen

This Technical Report was approved by CEN on 10 July 2022. It has been drawn up by the Technical Committee CEN/TC 310.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Türkiye and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

Ref. No. CEN/TR 17859:2022 E

Contents

Page

European foreword

This document (CEN/TR 17859:2022) has been prepared by Technical Committee CEN/TC 310 "Advanced automation technologies and their applications", the secretariat of which is held by BSI.

During its preparation, contributions have been received from the European FP7 Integrated Project Manufacturing Service Ecosystem (MSEE) and from the H2020 project PYMBIOSYS.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN shall not be held responsible for identifying any or all such patent rights.

Any feedback and questions on this document should be directed to the users' national standards body. A complete listing of these bodies can be found on the CEN website.

Introduction

It is generally considered that Manufacturing Enterprises will progressively migrate from traditional product-centric business to product-based, service-oriented virtual enterprises and ecosystems. This is a long and complex process that needs to be carefully assessed, prepared and planned. In particular, it would be necessary for a company that decides to pursue a manufacturing servitization project, to know clearly where it is (the current position) and where to go (the target position) so that strengths, weaknesses and needed investments can be identified. Enterprise Modelling offers the basic concepts for models, methods and tools to support this servitization process.

Benefits for the user result from a coordinated use of a common modelling language in the design and operation of service system. This leads to considerable quality improvement in the design process and cost reduction in the system operation.

A standardized Service Modelling Language (SML) is expected to foster the development of more compatible products in enterprise service modelling and hence reduce problems in the interoperation of such ICT products. SML will facilitate not only the modelling of services and service systems but will also support the development of interoperable software among co-operating organizations.

In addition, the SML will have a positive impact on improving interoperability of model based, service-oriented products, enabling European virtual factories and enterprises to adapt to the future internet infrastructure.

The SML with its associated meta-model reduces costly and fragmented development in this domain. SML focuses on modelling of manufacturing services that a company can develop to support its products.

The main added value of the proposed SML will be threefold:

i. Identification of the language constructs needed to define services needed by the business user.

ii. Integration of existing modelling languages constructs into one coherent meta-model.

iii. Definition of an MDSEA framework based on MDI/MDA to host the language and offer methods of model transformation between the modelling levels.

Three categories of enterprises are considered for this SML Technical report:

a) SMEs or large companies active in model based servitization or in the process of designing business models that include servitization aspects. The benefits of an SML standard are seen in a contribution to ease enterprise interoperability between organisations without the need of strong (re)engineering.

b) National/Regional projects or IT consultancies willing to integrate an SML standard in their project/domain. This business model together with the MSEE Toolbox (as described in Annex D) will enable the transfer of MSEE technology to other projects beyond those already existing. Four use cases deploying the MSEE technology in SMEs of the industry sectors Machine Tools, Observation, Furniture and Textile elaborated in the European PSYMBIOSYS project are presented in Annex D. The use cases demonstrate the applicability and the benefits of (SML) standard-based solutions

c) Large industrial enterprises, who need business models aimed at offering IT assets to other large industrial partners who are looking for standards-based solutions. A standard for language modelling will ease enterprise interoperability between the partners of such enterprise network and create measurable value.

This specification applies to manufacturing enterprises but can also apply to other classes of enterprises. It is intended for use by system engineers, IT and research specialists who are concerned with developing and deploying product related services in VMEs and Ecosystems. The constructs specified in this document are also intended to be used by those business users who are making decisions based on business rather than technical concerns. For this reason, many of the details are simplified or omitted compared to their equivalents (where they exist) in ISO 19440:2020.

Compared to ISO 19440:2020, SML employs fewer constructs and has a simpler structure. The SML can be considered as a derivation (but not a formal specialization) of the more general modelling language proposed in ISO 19440:2020. The modelling constructs of this proposed Technical Report are complementary to those constructs and support the design and implementation of future enterprise systems providing extended products (products + services) to the market.

NOTE Where SML constructs have the same name as those in IS19440, their meaning is the same, but their attributes are simplified (and sometimes rephrased) to include only those relevant to service modelling.

This document specifies a Service Modelling Language defined according to a Model Driven Service Engineering Architecture (MDSEA), to support the design and implementation of the service system in a virtual manufacturing enterprise environment. It also specifies a set of constructs for a Service Modelling Language.

Five annexes are provided addressing the basics concepts of service modelling, service modelling languages, tools and MDSEA, an example of service modelling at BSM level, industrial pilots to validate the SML and the progression between MDSEA levels.

The MDSEA is derived from [1] with necessary adaptation and extension to cover the modelling of service (and its system) in its most general forms. The modelling language addressed in this document is specified only at the Business Service Modelling (BSM) level of MDSEA.

1 Scope

This document specifies constructs for modelling and specifying product-related service systems in general business terms, recognising the service environment and the product lifecycle. The constructs and their meta-model are consistent with the Model Driven Service Engineering Architecture (MDSEA). They are intended for use by business users to address their business concerns and decision-making, and by systems engineers and IT/researchers using a model-driven engineering approach in the design, development and deployment of service systems in Virtual Manufacturing Enterprises (VMEs), business ecosystems and other application areas.

2 Normative references

There are no normative references in this document.

3 Terms, definitions, abbreviated terms and construct labels

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at https://www.electropedia.org/

— ISO Online browsing platform: available at https://www.iso.org/obp

3.1 Terms and definitions

3.1.1
construct-based modelling language
set of constructs and rules for valid groupings, which define the syntax of the modelling language

3.1.2
construct template
common structure that allows the identification and description of distinct modelling language constructs and the assignment of their properties

3.1.3
extended product
product enriched with associated product related services

3.1.4
manufacturing service ecosystem
manufacturing system of products bundled with associated services

3.1.5
service modelling language
set of constructs and rules for valid groupings of enterprise services, which define the syntax of the modelling language

3.1.6
servitization
process in a manufacturing enterprise to augment the value of products with services to better satisfy customer needs and sustainability

[SOURCE: ISO 19440:2020, modified]

3.2 Abbreviated terms

BSM Business System Modelling

GRAI Graphs with Results and Actions

ICT Information and Communication Technology

IT Information Technology

MDA Model Driven Architecture

MDI Model Driven Interoperability

MDSEA Model Driven Service Engineering Architecture

MSEE Manufacturing Service Ecosystem

SME Small- and/or Medium-size Enterprise

SML Service Modelling Language

TIM Technology Independent Modelling

TSM Technology Specific Modelling

UML Unified Modelling Language

VMA Virtual Manufacturing Enterprise

3.3 Construct labels

CUST Customer

DECN Decision

FUNC Functionality

ORG Organization

PERS Person

PI Performance Indicator

PR Product

PROC Process

RE Resource

SLA Service Level Agreement

SRV Service

STK Stakeholder

SVPR Service Provider

SVVN Service Vendor

USER User

VAL Value

4 Service Modelling Architecture and Language

4.1 Overview

A modelling language is defined by a set of modelling constructs. Construct(s) are represented, as illustrated in Figure 1, by a graphical representation, a template description and text. A template contains a header part to identify a construct instance, and a body part to describe the particular instance with descriptive and relationship properties. The proposed service modelling language is consistent with the enterprise modelling language representation concepts defined in ISO 19440:2020 [3].

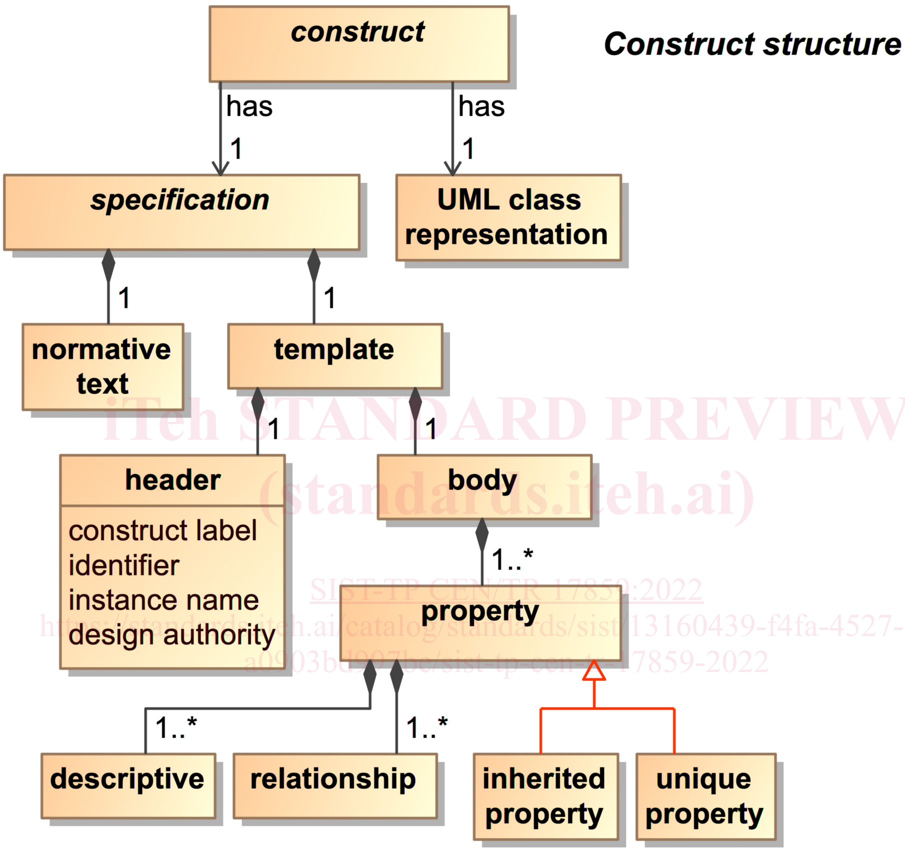

Figure 1 — Modelling language constituents (from ISO 19440:2020)

Using this template, this document specifies in Clause 5 a set of core Business Service Modelling constructs to model Service System at BSM level (which is the first level of the MDSEA).

4.2 Model Driven Service Engineering Architecture (MDSEA)

A Model Driven Service Engineering Architecture (MDSEA) is elaborated on the basis of [1] and [2] to support modelling of the three types of service system components (IT, human and physical means). This MDSEA is considered as an adaptation and extension of MDA and MDI to the engineering of product related services in virtual manufacturing enterprise environment.

As in MDA and MDI, the proposed MDSEA defines three abstraction levels as illustrated in Figure 2.

a) Business System Modelling (BSM), which specifies the models, at the global level, describing the running of the enterprise or set of enterprises as well as the links between these enterprises.

b) Technology Independent Modelling (TIM), which are the models at a more detailed level of abstraction, identifying needed resources and the capability independent from the technology used to implement the system.

c) Technology Specific Modelling (TSM) that combines the specification in the TIM model with details that specify how the system uses a particular type of technology (such as for example IT platform, physical means or organization with particular human profiles).

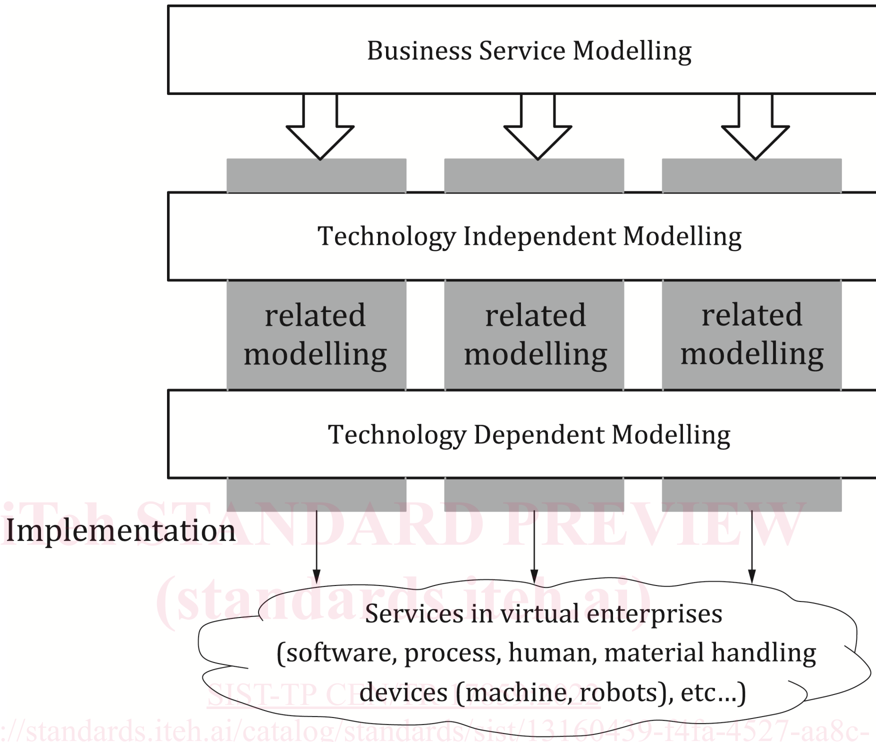

Figure 2 — MDSEA architecture

One result of Technology Specific Modelling is technical specifications of the three types of components (resources) to use to build the required service system: IT type for information handling, Machine type for material handling, and Human type for both information and material handling as well as the organisation. One example is the servitization project carried out in 2018 by the Renault car manufacturer in Paris Ville called Renault Mobility car service. In collaboration with Paris Municipality and other service providers in a framework of a virtual enterprise, a network of automated Renault car renting stations has been implemented in Paris. This service system is built with the three types of components:

— car information management system implemented by IT type components (computers, software applications and sensors, etc.);

— parking/Recharging stations implemented with Machine type components (robots, recharging facilities, etc.); and

— car maintenance system (repairing, cleaning, etc.) implemented with Human type components (cleaning workers, technicians, etc.).

The relationship between the MDSEA modelling levels (BSM, TIM, TSM) and the Service System lifecycle phases (user-requirements, design and implementation) have been established. One of the important innovations in MDSEA is to define the integration between domain components at the BSM level in order to ensure that these integration aspects will be addressed at the other levels.

4.3 Service modelling language (SML)

The concepts identified in 4.2 are adopted as modelling constructs to represent a Service System at BSM level. Figure 3 is an overview illustrating the Service concept and relationships with other concepts. This figure is elaborated to show all SML constructs in Figure 5. In Clause 5, each construct is further described by a template, and text to explain specific concerns and details.

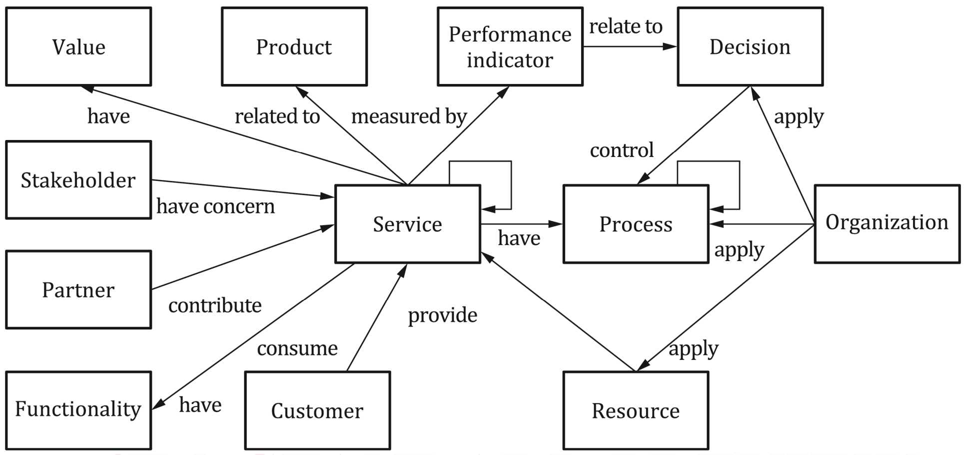

Figure 3 — Modelling constructs and relationships at BSM level

The kind of service considered in this document is that relating to a manufactured product. To develop such a service, it is necessary to build a Service System. Various entities are involved in such a development, ranging from the intended Customer who will consume the service to the Service Vendors and the Service Providers. Other entities such as technical centres, research centres and banks could be involved. All these other entities are called Stakeholders and all of them may express their specific concerns. A Service is used by Customers. A Service System provides functions that are utilities to fulfil customer's needs. The provided Service can be evaluated by a set of Performance Indicators, which are related to a set of decisions that control the Service System, i.e., are related to a set of objectives and decision variables.

4.4 Actor construct representing an Organization's or a Person's role

Persons and Organizations can have several roles and a role can generally be undertaken by either an Organization or a Person. Construct relationships (and their underlying class diagrams) need to cover this requirement without proliferating relationship links between Person/Organization and each role that they can possibly assume. This is achieved by using an abstract 'Actor' class model as illustrated in Figure 4.

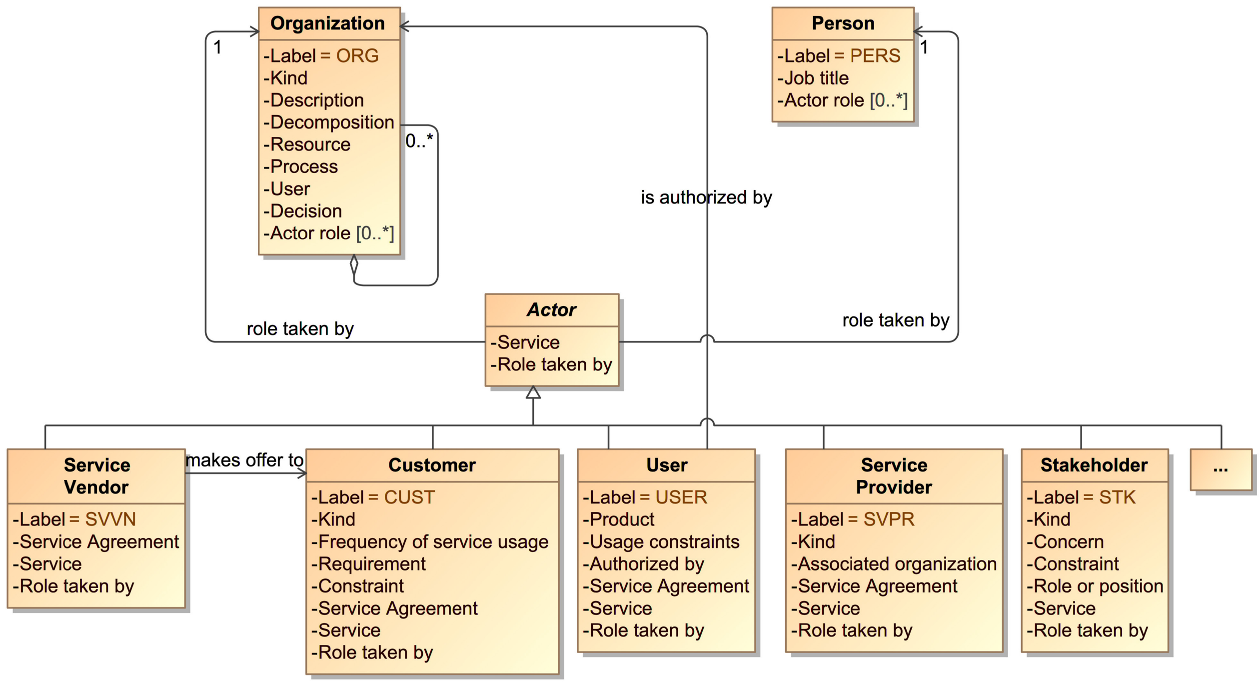

Figure 4 — Organization, Person and Actor roles

NOTE 1 Attributes which concern a particular role of an Organization or Person, e.g., usage of a service, have now migrated to attributes of the Actor specialization, e.g., to the User construct. All Actor role specializations then have the associated attributes <Service> and <Role taken by>. Whether a role is undertaken by an Organization or Person is determined by the Identifier and Name of the Actor specialization.

NOTE 2 Figure 4 is a view on a UML model and since the <Service> and <Role taken by> attributes are inherited from Actor they do not need to be shown explicitly as attributes for the Actor specializations. They are shown here (and in Figures 7 and 8) for consistency with the templates and completeness (since Actor is abstract and so not included in Clause 5).

4.5 SML Use Case Example

NOTE This clause is an extract adapted with permission from [9].

In today's service development, it is usual to have combination of services provided by different provides working together. The use case considered covers different kinds of services, approaches and solutions in the context of Industry 4.0, Internet of Things, Cyber Physical Systems, Cloud Computing and integration/federation of services of enterprise applications such as SCM, ERP, MES.

A central component of this use case is a matching service, which provides opportunities to create partnerships like value or supply chains but also purchasing pools and other networks within the hyper-connected ecosystem. A matching service running in a cloud infrastructure and using other services has been selected as one focus for the use case. The complete use case scenario considers cloud infrastructures, a matching service, a service to provide partner information to different service providers, a pump producer and a product service provider for water supply as well as a list of part suppliers (as illustrated in Figure 5).