ENAV20-9.24

Service Design Description for the xxx Service

<xyz Technology>

Contents

Table of figures

List of tables

1 Introduction

The bulk of work on this document, has been made as a deliverable for the EfficienSea2 project co-funded by the European Commission.

1.1 Purpose of the Document

This template shall support the service architects in creating a technical design description of the services (put down in writing), following the guidelines given in [1]. The template provides for each section descriptive instructions for the intended content. Formally, such instructions are written in blue italic font – they shall be deleted when writing the actual service design description document. In addition, some parts of this template provide suggested text fragments that may be directly re-used in the service design description document. Such proposed text fragments are given in black normal font.

The purpose of the service design description document is to write down the results of service technical design activity. The aim is to document how the service shall be realised by using a certain technology. The service design description document contains

- *identification and summary of the service design*
  - *reference to the service specification*
  - *identification of the service design*
- *identification and summary of chosen technology*
- *detailed description of how to realize each service interface and service operation*
  - *mapping of interfaced to the chosen technology*
  - *mapping of operations to the chosen technology*
  - *mapping of the message exchange patterns to the chosen technology*
- *detailed description of the physical data model*
  - *mapping to the service data model of the service specification.*

Note that a service design description document usually describes the technical aspects of one dedicated service specification. In theory, however, it is possible to elaborate a service design that realises more than one service specification.

The purpose of this service design description document is to provide a technology-specific description of how to realise a service specified by a service specification. The service design description document describes a well-defined baseline of the service design and clearly identifies the service design version. In this way it supports the configuration management process.

Note that the service design description is intended to complement the technology-agnostic service specification. The purpose of the service design description document is to describe in detail the actual realisation of a service with a dedicated technology.

This section should be replaced by a suitable description of the purpose. For instance:

The purpose of this service design description document is to provide a detailed description of the *<XYZ>* service (see *[3]*), realized by using the *<ABC>* technology, according to the guidelines given in [1]. It describes a well-defined baseline of the service design by clearly identifying the service design version.

The aim is to document the key aspects of the *XYZ* service technical design. This includes:

- identification and summary of the service design
  - reference to the service specification
  - identification of the service design
- identification and summary of chosen technology
- detailed description about the realization of each service interface and service operation
  - mapping of interfaces to the chosen technology
  - mapping of operations to the chosen technology
  - mapping of the message exchange patterns to the chosen technology
- detailed description of the physical data model
  - mapping to the service data model of the service specification.

1.2 Intended Readership

This service design description template is intended to be read by service architects and designers who shall produce service technical designs.

This section shall describe the intended readers of the service design description document. E.g.:

This service design description document is intended to be read by service architects, designers, system engineers and developers in charge of designing and developing an instance of the *XYZ* service.

Furthermore, this service design description is intended to be read by service architects, information architects, system engineers and developers in pursuing architecting, design and development activities of other related services.

1.3 Inputs from Other Projects

This section lists previous work on the subject covered by this document.

Special emphasis shall be put on what has been reused from other (already finished) projects.

This section provides an overview of projects, which are dealing with similar topics and lists already finished ones that provided inputs to this activity.

2 Service Design Identification

The purpose of this chapter is to provide a unique identification of the service design and describe where the service is in terms of the engineering lifecycle.

The table~~s~~ below shall be completed.

Name	*Service Design Name*
ID	*Unique identity of service design*
Version	*Version of the XYZ service design*
Technology	*Indication of the technology for which this design is intended (for example REST or SOAP).*
Service Specification ID	*Reference to the service specification*
Service Specification Version	*Reference to the service specification*
Description	*Description of the XYZ service design*
Keywords	*Keywords that can be used to find the service design in the service registry*
Architect(s)	*Name of service architects/designers and their organisation*
Status	*Status of the service design in the engineering lifecycle – either "Provisional", "Released", "Deprecated" or "Deleted".*[1] *"Provisional": the service design is (partly) available, but not yet officially released.* *"Released": the service design is ready to be used.* *"Deprecated": service design is announced to become invalid in near future.* *"Deleted": service design is not valid any more.*

[1] If more elaborated governance rules for the service design process would become available, additional status values could be envisaged in the future: e.g., Validated, Verified.

3 Technology Introduction

The technology introduction section contains a basic background about the chosen technology. In most cases this will be a short description of basic technology aspects accompanied with appropriate references to standard documents and best practice descriptions.

The template does not provide further details for the structure of this section. The actual structure is left to the author's choice.

4 Service Design Overview

This chapter provides an overview of the main elements of the service design and a mapping of the design elements to the service specification elements.

This chapter aims at providing an overview of the main elements of the service design and a mapping of the design elements to the service specification elements. The elements in this view are all usually created by a UML modelling tool.

Architectural elements applicable for this description are:

- *Service:*
  the element representing the service in its entirety.
- *Service Interfaces:*
  the mechanisms by which a service communicates. Defined by allocating service operations to either the provider or the consumer of the service.
- *Service Operations:*
  describe the operations used to access the service.
- *Service Operations Parameter Definitions:*
  identify data structures being exchanged via Service Operations.

Above elements may be depicted in one or many diagrams. Which and how many diagrams are needed depends on the chosen architecture description framework, the chosen technology, and the complexity of the service.

If the structure of the service design follows the service specification to a great extent, then it is not necessary to repeat identical diagrams here in this section; in this case, this section shall contain references to the service specification document. However, it is assumed that in many cases, depending on the chosen technology, the actual interface and/or operation names (and structuring) are not 100% identical to the abstract definition given in the service specification.

4.1 Service Interfaces

Describe the interfaces of the service design and their mapping to the interfaces defined in the service specification. Furthermore describe how the specified Message Exchange Patterns (MEP) are realised with the chosen technology.

An example diagram and corresponding table is given below.

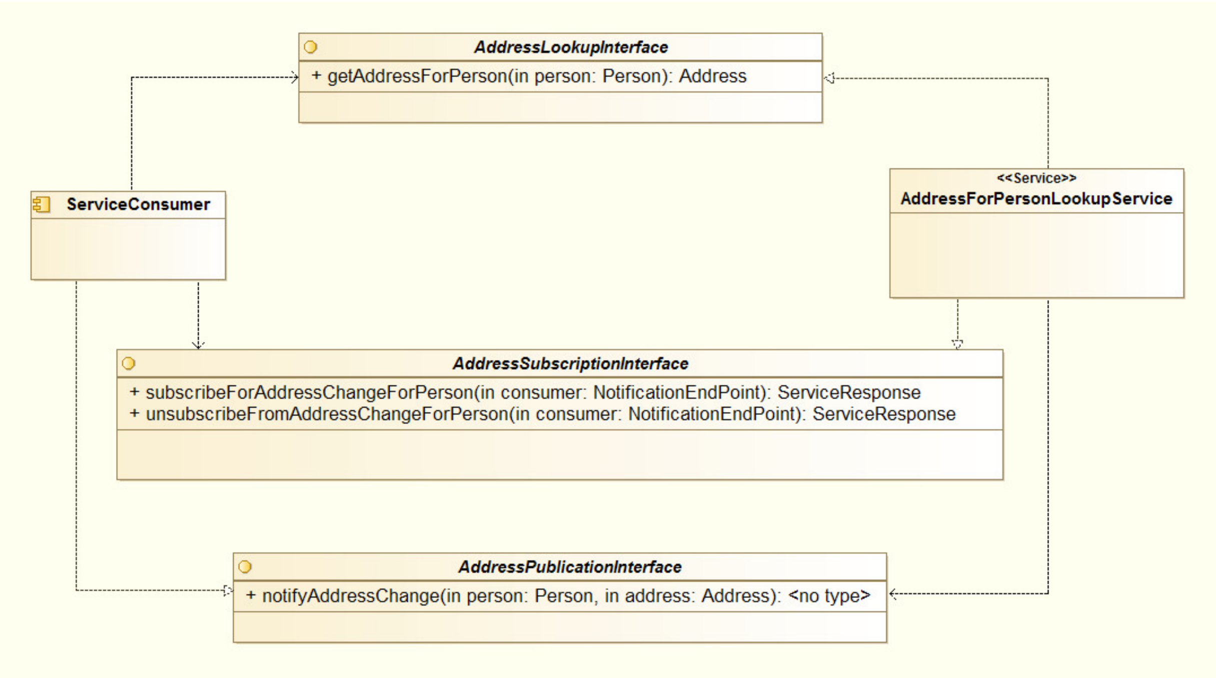
AddressLookupInterface
+ getAddressForPerson(in person: Person): Address
ServiceConsumer
<<Service>>
AddressForPersonLookupService
AddressSubscriptionInterface
+ subscribeForAddressChangeForPerson(in consumer: NotificationEndPoint): ServiceResponse
+ unsubscribeFromAddressChangeForPerson(in consumer: NotificationEndPoint): ServiceResponse
AddressPublicationInterface
+ notifyAddressChange(in person: Person, in address: Address): <no type>

Figure 1: <Service Name> **Interface Definition diagram**

Table 1: Service Interface Mapping

Service Specification		Service Design	
ServiceInterface	**Service Operation**	**Service Interface**	**Service Operation**
AddressLookupInterface	getAddressForPerson	AddressLookup (see WSDL file [x])	findAddress()
subscribeForAddressChangeForPerson	subscribeForAddressChangeForPerson	WebService Notification interface specified by WSDL file [y]	Standard WS-N subscribe()
	unsubscribeFromAddressChangeForPerson		Standard WS-N unsubscribe()
AddressPublicationInterface	notifyAddressChange	WebService Notification interface specified by WSDL file [y]	Standard WS-N notify()

The table above (in this example for service design using SOAP) shall provide the mapping of service design to service specification, as well as references to the formal descriptions of the service interfaces and operations (these references are symbolised by [x], [y] in the table above). These may be references to external documents (e.g., standards) or to other sections in this document (e.g., to subsection of section 6).

5 Physical Data Model

This section describes in detail the data structures to be exchanged between providers and consumers of the service.

This chapter provides a detailed description of the data structures exchanged between service provider and service consumer. This description shall also include a mapping of the data structures to the service data model provided in the service specification.

The service design description template does not prescribe a detailed format for this section. Allowed presentations of the physical data model include

- *UML diagrams representing the data structures including detailed physical data type descriptions at attribute level;*
- *XML/XSD files describing the data structures;*
- *Tabular presentations.*

Any mixture of the above formats is allowed.

Example of an UML diagram:

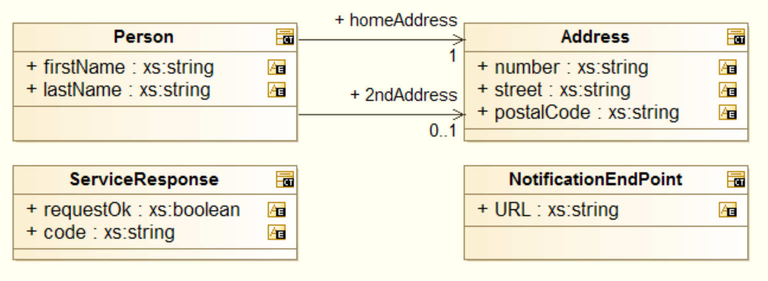

Figure 2: <Service Name> Service Data Model diagram

It is mandatory to give a description of each entity item (class), its attributes and the associations between entity items. The data type of each attribute must be provided, appropriate to the chosen technology.

If the physical service data model is related to an external data model (e.g., being a subset of a standard data model, e.g. based on an S-100 specification), then this section shall refer to it: each data item of the physical data model shall be mapped to a data item defined in the external data model. This mapping may be added in the same table that describes the data items and their attributes and associations.

The table below is an example for describing a service data model including traces to an external model.

Element Name	Description

Person		Describe here the “Person” structure.	
	Attribute Name	Type	Description
	firstName	String	Description of firstName goes here.
	Tracing Information	Value	
	Spec.data model trace	Trace into the service specification data model for firstName	
	External model trace	Trace into the external data model for firstName	
	Attribute Name	Type	Description
	lastName	String	Description of lastName goes here.
	Tracing Information	Value	
	Spec.data model trace	Trace into the service specification data model for lastName	
	External model trace	Trace into the logical or physical model for the lastName	
	Attribute Name	Type	Description
	homeAddress	Address	The main home address of Person
	Tracing Information	Value	
	Spec.data model trace	Trace into the service specification data model for homeAdd	
	External model trace	Trace into the logical or physical model for the homeAddress	
	Attribute Name	Type	Description
	2ndAddress	Address	Any second address of Person (optional)
	Tracing Information	Value	
	External model trace	Trace into the logical or physical model for the 2ndAddress	
Element Name		Description	
Address		Describe here the Address structure.	
	Attribute Name	Type	Description
	number	String	Description of number goes here.
	Tracing Information	Value	
	Spec.data model trace	Trace into the service specification data model for the number attribute	
	External model trace	Trace into the logical or physical model for the number attribute	
	Attribute Name	Type	Description
	…		

An XML schema for this data model is included in the formal service design xml file attached in Appendix A.

5.1 Service Internal Data Model (optional)

Optionally, this section may provide a description of the internal data model, as it seems appropriate to the service provider and/or the service consumer side. Such description might be helpful for the understanding as it provides additional information of how the service might be built. However, it has to be seen as exemplary only – it is not an authoritative part of the service design description.

6 Service Interface Design

This chapter describes the details of each service interface. One sub-chapter is provided for each Service Interface.

The Service Interface design covers the static design description while the dynamic design (behaviour) is described in chapter 7.

The static interface description is vital since it describes how the interfaces shall be constructed. The structure of this section is identical to the structure of the Service Interface Specifications section in the service specification document. This section may be limited to references to the service specification document, if all of the following conditions are fulfilled:

- *the service design reflects the service interfaces in a 1:1 manner,*
- *the service interfaces are sufficiently described in the service specification,*
- *the physical data model (section 5) contains an unambiguous mapping of all payload data items of the service specification to the detailed physical data items.*

Architectural elements applicable for this description are:

- *Service Interfaces*
- *Operations*
  Function or procedures which enable programmatic communication with a service via a service interface.
- *Parameters*
  Constants or variables passed into or out of a service interface as part of the execution of an operation

A Service may have one or more service Interfaces. Please describe each in separate sections below.

6.1 Service Interface <Interface Name>

Please explain the purpose, messaging pattern and architecture of the Interface.

A Service Interface supports one or several service operations. Each operation in the service interface shall be described in the following sections.

6.1.1 Operation <Operation Name>

Give an overview of the operation: Include here a textual description of the operation functionality. In most instances this will be the same as the operation description taken from the UML modelling tool.

Operation Functionality

Describe here the functionality of the operation, i.e., how does it produce the output from the input payload.

Operation Parameters

Describe the logical data structure of input and output parameters of the operation (payload) by using an explanatory table (see below) and optionally UML diagrams (which are usually sub-sets of the service data model described in previous section above).

Below is an example of a UML diagram (subset of the service data model, related to one operation):

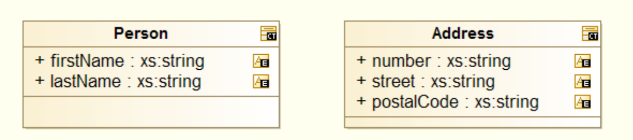

Figure 3: <Service Name> Interface Parameter Definition diagram for <operation name>

It is mandatory to provide a table with a clear description of each service operation parameter and the information about which data types defined in the service data mode are used by the service operation in its input and output parameters.

Note: While the descriptions provided in the physical data model shall explain the data types in a neutral format, the descriptions provided here shall explicitly explain the purpose of the parameters for the operation.

Below is an example operation parameter description table.

Table 2: Payload description of <operation name> operation

Parameter Name	Direction	Data Type	Description
person	*Input*	*Person*	*The "person" parameter specifies the person for which the address is being looked for.*
<none>	*Return*	*Address*	*The return value provides the address of the person.*

6.1.2 Operation <Operation Name>

Repeat previous section for every operation defined in the service interface definition operation.

6.2 Service Interface <Interface Name>

Repeat previous section for each interface

7 Service Dynamic Behaviour

This chapter describes the interactive behaviour between service interfaces (interaction specification) and, if required, between different services (orchestration). Architectural elements applicable for this description are:

- *Service Interaction Specifications*
- *Service State machines*
- *Service orchestration*

Following types of views and UML diagrams can be used to describe the dynamic behaviour[2]:

- *Sequence diagrams*
- *Interaction diagrams*
- *State machine diagrams*

This section is especially relevant, if the service design structure (see section 4) differs from the service structure introduced in the service specification. If designed service interfaces and operations are equivalent to those of the service specification, and if the dynamic behaviour is sufficiently described in the service specification, then this section may be limited to references to the service specification document.

7.1 Service Interface <Interface Name>

Include some information about the dynamic aspects of the service interface; each operation should be exposed on at least one diagram.

An example sequence diagram is given below.

[2] e.g., in NATO Architectural Framework (NAF), state model and interaction specification (NAF3.1) or NSOV-5 Service constraints, state model could be used.

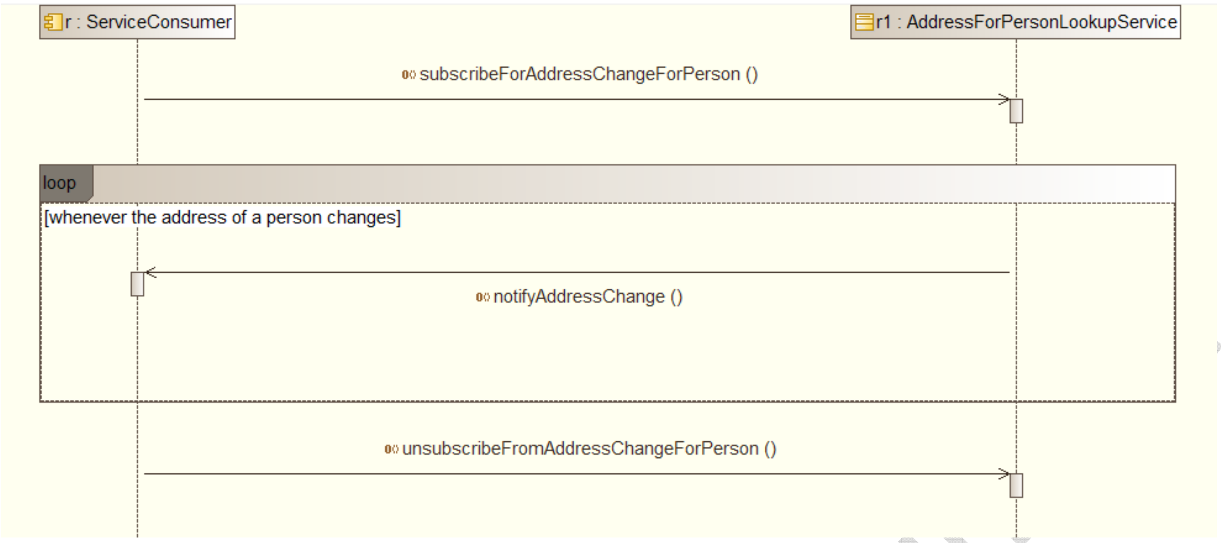
r : ServiceConsumer
r1 : AddressForPersonLookupService
subscribeForAddressChangeForPerson ()
loop
[whenever the address of a person changes]
notifyAddressChange ()
unsubscribeFromAddressChangeForPerson ()

Figure 4: <Service Name> Operation Sequence Diagram

7.2 Service Interface <Interface Name>

Replicate previous section for each service interface

8 References

This chapter shall include all references used when designing the service. Specifically, the service specification document as well as standard documents describing the chosen technology and documents describing any external data models (if applicable) shall be listed.

Nr.	Version	Reference
[1] Service Documentation Guidelines	01.00	SG_Annex_A_Service_Documentation_Guidelines
[3] XYZ Service Specification	*xx.yy*	*Service Specification for the XYZ service.*

9 Acronyms and Terminology

9.1 Acronyms

Term	Definition
API	Application Programming Interface
MC	Maritime Cloud
MEP	Message Exchange Pattern
NAF	NATO Architectural Framework
REST	Representational State Transfer
SOAP	Simple Object Access Protocol
SSD	Service Specification Document
UML	Unified Modelling Language
URL	Uniform Resource Locator
VTS	Vessel Traffic Service
WSDL	Web Service Definition Language
XML	Extendible Mark-up Language
XSD	XML Schema Definition

9.2 Terminology

Term	Definition
External Data Model	Describes the semantics of the "maritime world" (or a significant part thereof) by defining data structures and their relations. This could be at logical level (e.g., in UML) or at physical level (e.g., in XSD schema definitions), as for example standard data models, or S-100 based data produce specifications.
Message Exchange Pattern	Describes the principles two different parts of a message passing system (in our case: the service provider and the service consumer) interact and communicate with each other. Examples: In the Request/Response MEP, the service consumer sends a request to the service provider in order to obtain certain information; the service provider provides the requested information in a dedicated response. In the Publish/Subscribe MEP, the service consumer establishes a subscription with the service provider in order to obtain certain information; the service provider publishes information (either in regular intervals or upon change) to all subscribed service consumers.
Operational Activity	An activity performed by an operational node. Examples of operational activities in the maritime context are: Route Planning, Route Optimization, Logistics, Safety, Weather Forecast Provision, …
Operational Model	A structure of operational nodes and associated operational activities and their inter-relations in a process model.
Operational Node	A logical entity that performs activities. Note: nodes are specified independently of any physical realisation. Examples of operational nodes in the maritime context are: Maritime Control Center, Maritime Authority, Ship, Port, Weather Information Provider, …
Service	The provision of something (a non-physical object), by one, for the use

	of one or more others, regulated by formal definitions and mutual agreements. Services involve interactions between providers and consumers, which may be performed in a digital form (data exchanges) or through voice communication or written processes and procedures.
Service Consumer	A service consumer uses service instances provided by service providers. All users within the maritime domain can be service customers, e.g., ships and their crew, authorities, VTS stations, organizations (e.g., meteorological), commercial service providers, etc.
Service Data Model	Formal description of one dedicated service at logical level. The service data model is part of the service specification. Is typically defined in UML and/or XSD. If an external data model exists (e.g., a standard data model), then the service data model shall refer to it: each data item of the service data model shall be mapped to a data item defined in the external data model.
Service Design Description	Documents the details of a service technical design (most likely documented by the service implementer). The service design description includes (but is not limited to) a service physical data model and describes the used technology, transport mechanism, quality of service, etc.
Service Implementation	The provider side implementation of a dedicated service technical design (i.e., implementation of a dedicated service in a dedicated technology).
Service Implementer	Implementers of services from the service provider side and/or the service consumer side. Anybody can be a service implementer but mainly this will be commercial companies implementing solutions for shore and ship.
Service Instance	One service implementation may be deployed at several places by same or different service providers; each such deployment represents a different service instance, being accessible via different URLs.
Service Instance Description	Documents the details of a service implementation (most likely documented by the service implementer) and deployment (most likely documented by the service provider). The service instance description includes (but is not limited to) service technical design reference, service provider reference, service access information, service coverage information, etc.
Service Interface	The communication mechanism of the service, i.e., interaction mechanism between service provider and service consumer. A service interface is characterised by a message exchange pattern and consists of service operations that are either allocated to the provider or the consumer of the service.
Service Operation	Functions or procedure which enables programmatic communication with a service via a service interface.
Service Physical Data Model	Describes the realisation of a dedicated service data model in a dedicated technology. This includes a detailed description of the data payload to be exchanged using the chosen technology. The actual format of the service physical data model depends on the chosen technology. Examples may be WSDL and XSD files (e.g., for SOAP services) or swagger (Open API) specifications (e.g., for REST services). If an external data model exists (e.g., a standard data model), then the service physical data model shall refer to it: each data item of the service physical data model shall be mapped to a data item defined in the external data model.

	In order to prove correct implementation of the service specification, there shall exist a mapping between the service physical data model and the service data model. This means, each data item used in the service physical data model shall be mapped to a corresponding data item of the service data model. (In case of existing mappings to a common external (standard) data model from both the service data model and the service physical data model, such a mapping is implicitly given.)
Service Provider	A service provider provides instances of services according to a service specification and service instance description. All users within the maritime domain can be service providers, e.g., authorities, VTS stations, organizations (e.g., meteorological), commercial service providers, etc.
Service Specification	Describes one dedicated service at logical level. The Service Specification is technology-agnostic. The Service Specification includes (but is not limited to) a description of the Service Interfaces and Service Operations with their data payload. The data payload description may be formally defined by a Service Data Model.
Service Specification Producer	Producers of service specifications in accordance with the service documentation guidelines.
Service Technical Design	The technical design of a dedicated service in a dedicated technology. One service specification may result in several technical service designs, realising the service with different or same technologies.
Service Technology Catalogue	List and specifications of allowed technologies for service implementations. Currently, SOAP and REST are envisaged to be allowed service technologies. The service technology catalogue shall describe in detail the allowed service profiles, e.g., by listing communication standards, security standards, stacks, bindings, etc.
Spatial Exclusiveness	A service specification is characterised as "spatially exclusive", if in any geographical region just one service instance of that specification is allowed to be registered per technology. The decision, which service instance (out of a number of available spatially exclusive services) shall be registered for a certain geographical region, is a governance issue.

Appendix A Service Design Description XML

This appendix contains the formal definition of the service design description.

It is up to the author whether the service design description xml file (which includes the technology dependent definition of the physical data model) is presented in full text or just as an embedded file.