XXX

Service Description

Abstract
This is the template for Service Description (SD document) according to the Eclipse Arrowhead documentation structure.

Contents

1 Overview

This document describes the [XX] service, which is enables [a description of service capabilities excluding implementation details].

The rest of this document is organized as follows. In Section 2, we describe the abstract message operations provided by the service. In Section 3, we end the document by presenting the data types used by the mentioned operations.

1.1 Significant Prior Art

Describe significant prior art which provides the foundation for the service - May be omitted for simple services

1.2 How This Service Is Meant to Be Used

Describe intended usage of the service

1.3 Important Delimitations and Dependencies

Provide delimitations and dependecies of the provided service

2 Service Interface

This section describes the operations of the [XX] service. In particular, each subsection names an *abstract operation*, an *input type* and an *output expression*, in that order. The input type is named inside parentheses, while the output expression is preceded by a colon. There is currently no formal grammar for how to express output expressions. A simple such would just be to state one type, or none at all. A more complicated example is given for the Subscribe() example operation, defined in this section. Input and output types are only denoted when accepted or returned, respectively, by the interface in question.

All operations listed in this section are examples of such that could be part of a service. None of them is in any way mandatory. All abstract data types named in this section are defined in Section 3.

Figure 1: EXAMPLE: SysML block description diagram of the ServiceDiscovery and its interface

The following interface operations are available.

2.1 operation A(AA): Response

The AA operation is used to ..... The services will contain various metadata as well as a physical endpoint. The various parameters are representing the necessary service input information.

2.2 operation B(BB): AA or Error

The BB operation is used to ....

2.3 operation C(CC)

The CC operation enable .....

2.4 operation Echo(): StatusCodeKind

The Echo operation provides an is alive response from the XX service.

2.5 operation Subscribe(): repeat StatusUpdate until Unsubscribe() or shutdown

The subscribe operation initiates a flow of status updates, which continues until the Unsubscribe() operation is invoked by the same consumer or the provider shuts down.

An operation may trigger any kind of messaging pattern, not just the request-response pattern that is most convenient to express with the notation we have currently adopted. There is, at the time of writing, no formal grammar for expressing these kinds of response behaviors in Arrowhead documentation.

2.6 operation Unsubscribe()

Cancels any subscription initiated by the same consumer via Subscribe().

3 Information Model

Here, all data objects that can be part something the XX Service provides to the hosting System are listed in alphabetic order. Note that each subsection, which describes one type of object, begins with the *struct* keyword, which is used to denote a collection of named fields, each with its own data type. As a complement to the explicitly defined types in this section, there is also a list of implicit primitive types in Section 3.5, which are used to represent things like hashes and identifiers.

EXAMPLE data object types are illustrated in Figure 2.

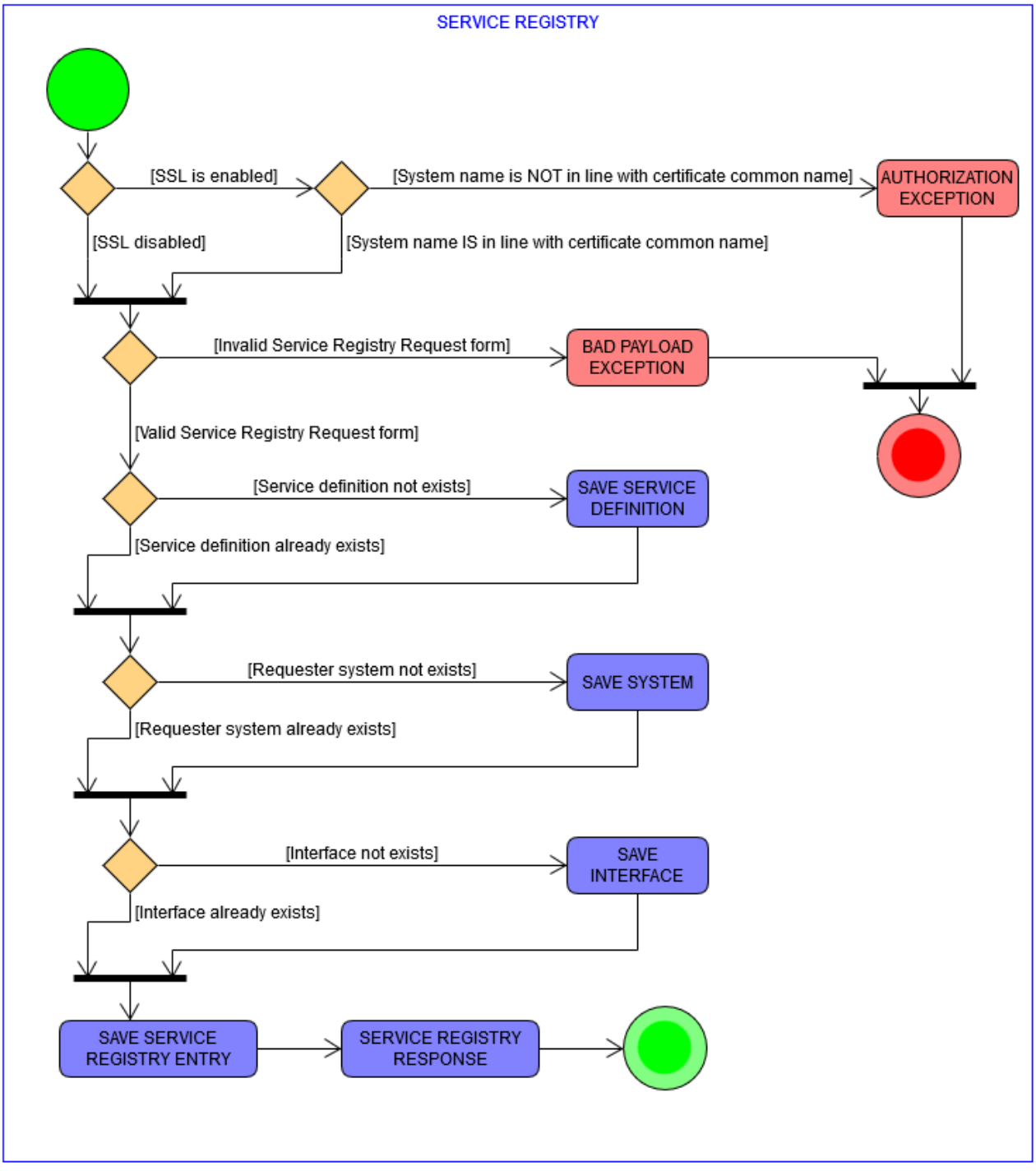

Figure 2: EXAMPLE: Information model as a UML activity diagram. Describes the process of service registration.

3.1 struct AA

EXAMPLE: This structure is used to register a service offering into the Service Registry.

Field	Type	Description
endofValidity	DateTime	Service is available until this UTC timestamp.
interfaces	Array<Interface>	List of interfaces the service supports.
metadata	Metadata	Metadata
providerSystem	Name	Name of the provider system.
secure	SecureType	Type of security the service uses.
serviceDefinition	Name	Service Definition.
serviceUri	URI	URI of the service.
version	Version	Version of the service.

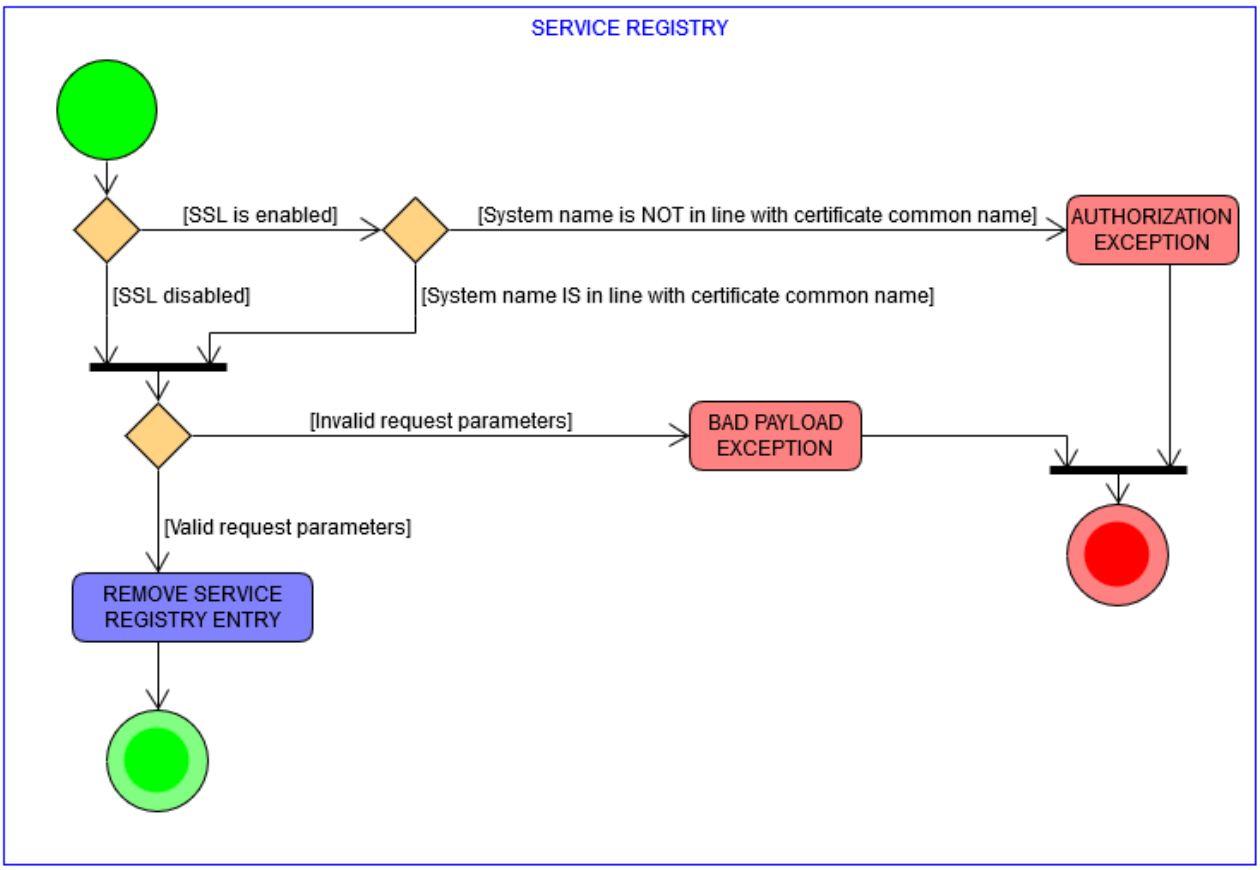

Figure 3: EXAMPLE: Information model as a UML activity diagram. Describes the process of service unregistration.

3.2 struct BB

EXAMPLE: This structure is used to register a service offering into the Service Registry. Please also refer to the activity diagram in Figure 3

Field	Type	Description
address	Address	Address of the provider systems.
port	PortNumber	Port of the provider system.
system_name	Name	System name of the provider system
service_definition	Name	Service Definition of the unregistered service.

3.3 struct CC

EXAMPLE: This structure is used to query service offering registered in the Service Registry. Please also refer to the activity diagram in Figure 4

Object Field	Value Type	Description
interfaceRequirements	Array<Interface>	List of the required interfaces.
maxVersionRequirement	Version	Maximum version.
minVersionRequirement	Version	Minimum version.
metadataRequirements	Metadata	Metadata.
pingProviders	Boolean	Checks the availability of the providers if true
securityRequirements	Name	Type of security.
serviceDefinitionRequirement	Name	Service Definition.
versionRequirement	Version	Version of the service.

3.4 struct StatusUpdate

EXAMPLE: An event propagated to systems having invoked the Subscribe() operation.

Object Field	Value Type	Description
timestamp	DateTime	Time at which this status update was generated.
subscriptionCount	Integer	The number of active subscriptions.

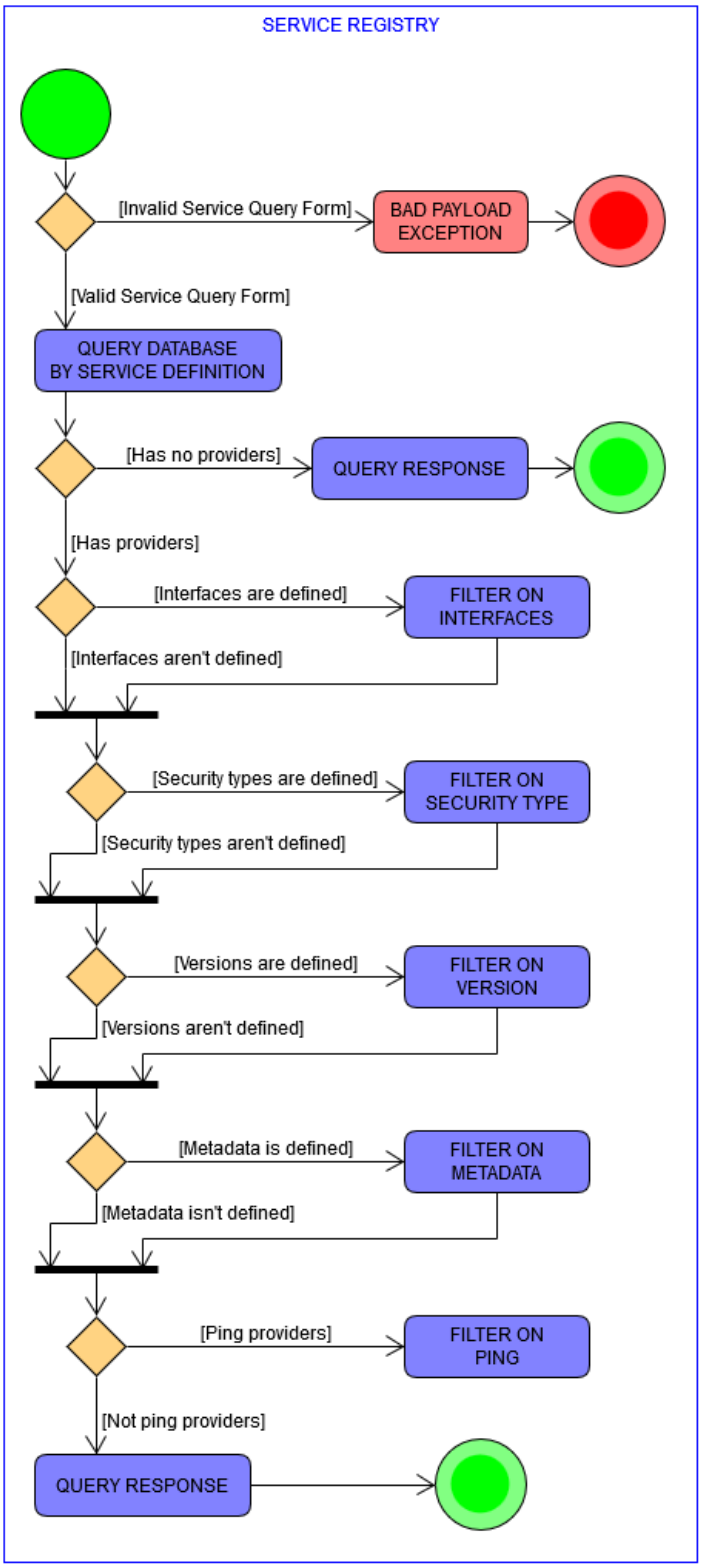

Figure 4: EXAMPLE: Information model as a UML activity diagram. Describes the process of service querying.

3.5 Primitives

Types and structures mentioned throughout this document that are assumed to be available to implementations of this service. The concrete interpretations of each of these types and structures must be provided by any IDD document claiming to implement this service.

Type	Description
Address	A string representation of the address
Array<A>	An *array* of a known number of items, each having type A.
Boolean	One out of `true` or `false`.
DateTime	Pinpoints a specific moment in time.
Error	An error indicator, the exact details of which are chosen by the implementor of the service.
Integer	An integer.
Interface	Any suitable type chosen by the implementor of the service.
Name	A string identifier that is intended to be both human and machine-readable.
PortNumber	Decimal number in the range of 0-65535
Response	Any suitable type chosen by the implementor of the service.
StatusCodeKind	A string status code.
Version	Specifies a service version.

4 Revision History

4.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2020-12-05	X.Y.Z		Tanyi Szvetlin
2	2021-07-14	X.Y.Z	Minor updates	Jerker Delsing
3	2022-01-10	X.Y.Z	Minor updates	Jerker Delsing

4.2 Quality Assurance

No.	Date	Version	Approved by
1	2022-01-10	X.Y.Z	