D3.7 Technical Specification

Project no.	636329
Project acronym:	EfficienSea2
	EFFICIENSEA2 – efficient, safe and sustainable traffic at sea
Funding scheme:	Innovation Action (IA)
Start date of project:	1 May 2015
End date of project:	30 April 2018
Duration:	36 months
Due date of deliverable:	31.10.2016
Actual submission date:	28.10.2016
Organisation in charge of deliverable:	OFFIS

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

Document Status

Authors

Name	Organisation
Sören Schweigert	OFFIS
Benjamin Weinert	OFFIS
André Bolles	OFFIS

Document History

Version	Date	Initials	Description
1.0	31.10.2016	SoS	Initial version

Review

Name	Organisation
Tomas Groth Christensen	DMA
Hubert Künig	Frequentis
Adam Lipka	National Institute of Telecommunications

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

Contents

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

1 Introduction

The Maritime Cloud is an initiative towards providing infrastructural functions such as supporting the concept of a digital maritime identity.

The concept behind the Maritime Cloud has been derived as “A communication framework enabling efficient, secure, reliable and seamless electronic information exchange among all authorized maritime stakeholders across available communication systems”, based on the IMO e-navigation strategy. This concept shall be enabled through an open and vendor-neutral platform and will allow for interconnecting heterogeneous software systems on board of various ship types, offshore structures or on shore.

Thereby the Maritime Cloud infrastructure does not aim to be used by mariners or ship owners directly, but to serve as a framework, providing standardized protocols and functions for service discovery, identity and role management and efficient messaging in a geographic context. Thus enables easy development of new and innovative solutions for mariners and shore based end users.

For more information about the vision of the Maritime Cloud, the conceptual model behind and an informational user story, refer to the Deliverable “D3.2 Conceptual Model” [1]

1.1 Objectives of the document

The main goal of this document is to present a common understanding of the core components of the Maritime Cloud (MC), between project partners in EfficienSea 2 as well as other related projects like STM[1].

Besides the identification of the Maritime Cloud core components, this also includes the description and definition of how the Service Registry (SR) and Identity Registry (IR) work, how they interact with each other and how to use them in new maritime applications.

For this purpose the document takes over different perspectives onto the MC, related to its usage:

- The perspective of end users, like mariners,
- The perspective of service providers, who wish to publish their new developed services and
- The perspective of administrators in maritime organizations, who manage the identity of their employees or associated stuff.

[1] STM Website: http://stmvalidation.eu/

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

2 Overview

The Maritime Cloud is a service-oriented architecture that is composed of four core components, namely the Service Registry (SR), the Identity Registry (IR), an optional Communication and Messaging Component and the Maritime Demonstrator Component (MCDC). In addition a fifth component called Almanac serves as offline copy of parts of the SR and IR, to allow access to relevant information in the absence of an internet connection.

1) **Service Registry (SR):**
   The Service Registry is one of the Maritime Cloud core components (MC³) and is supposed to serve as a central reference point to provide and find services and thus to improve the visibility and accessibility of available information and services in the maritime domain. It's best compared with a yellow pages phone book.

2) **Identity Registry (IR):**
   The Identity Registry has the objective to enable an authentication of all maritime stakeholders in the context of the Maritime Cloud and thus increasing the security and reliability of communication. This goal is archived by providing a trustworthy infrastructure for identity authentication of maritime entities like human actors, services and devices.

3) **Communication and Messaging Component:**
   The optional communication and messaging component shall ease the communication between the user, the Maritime Cloud Core Components and external services. This is done, by providing the capability to support seamless roaming on IP and non IP based communication channels and advanced message distribution, for example the Roaming Device from EfficienSea 2 (D2.7 Conecpt and specification for seamless roaming).

4) **Maritime Cloud Demonstrator Component (MCDC):**
   The Maritime Cloud Demonstrator Component serves as a reference implementation on how to use the other Maritime Cloud core components and can further be used by application developer (for example provider of ship equipment) to ease the usage of MC functionality. For this purpose the MCDC realizes some convenience functions, and takes the Roaming Device into account, to ensure a maximum of connectivity.

5) **Almanac:**
   The Almanac is an offline version of parts of the Service and Identity Registry, to be used if no stable internet connection is available for lookup in the online versions of SR and IR and thus to always allow access to the most relevant information during a journey.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

The following Figure 1 gives an overview over the components and how users and services interact with the Maritime Cloud.

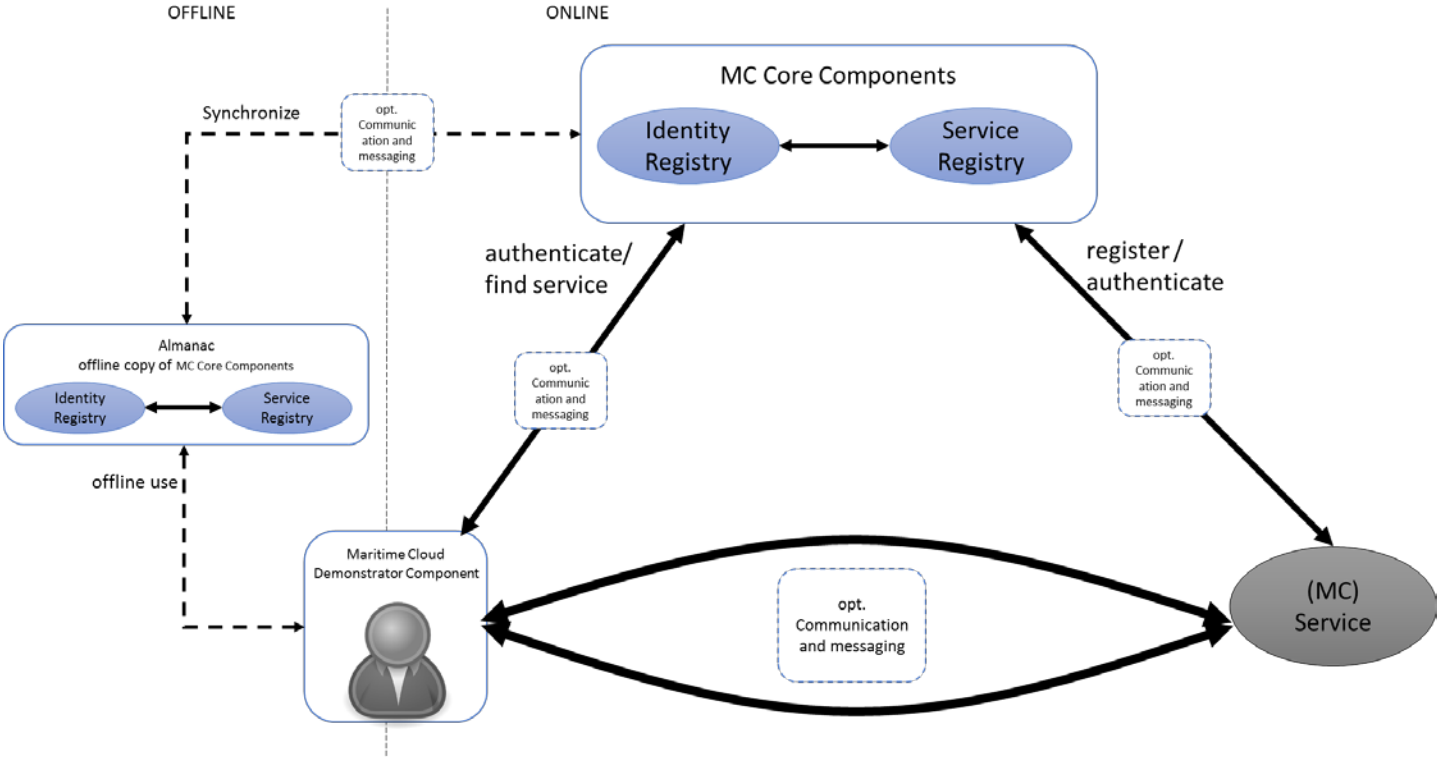

OFFLINE
ONLINE
MC Core Components
Identity Registry
Service Registry
Synchronize
opt. Communic ation and messaging
authenticate/ find service
register / authenticate
Almanac
offline copy of MC Core Components
Identity Registry
Service Registry
opt. Communic ation and messaging
opt. Communic ation and messaging
offline use
Maritime Cloud Demonstrator Component
opt. Communication and messaging
(MC) Service

Figure 1 Overview over the Maritime Cloud components and their interconnection

From the actor's perspective, the actor uses the Maritime Cloud Core Components, to authenticate himself within the Maritime Cloud. After he has been authenticated by the Identity Registry, he can use the Service Registry to discover descriptions and endpoints of services.

On the right side of Figure 1, a service can be registered within the Service Registry, by its provider. After a Maritime Cloud user has established a connection, the service can use the Identity Registry to verify the identity of that user.

While the Maritime Cloud in general assumes an IP based connection, such an IP connection may not be available at any time, especially on board of an offshore vessel. In such a case, the Maritime Cloud provides an optional Communication and messaging component, which is able to utilize different types of communication channels (e.g. by using the Roaming Device).

In case the user is not able to establish a stable internet connection, even with the help of the communication and messaging component, the actor also got the opportunity to lookup services and identities of other maritime actors, using the Almanac component, as indicated on the most left part of Figure 1. This offline copy of the Maritime Cloud Core Component, will be synchronized, if a stable internet connection is available.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

2.1 Communication and messaging

As described above, the Maritime Cloud supports the MC users, service user as well as service implementers with means for communication, using the optional Communication and messaging component.

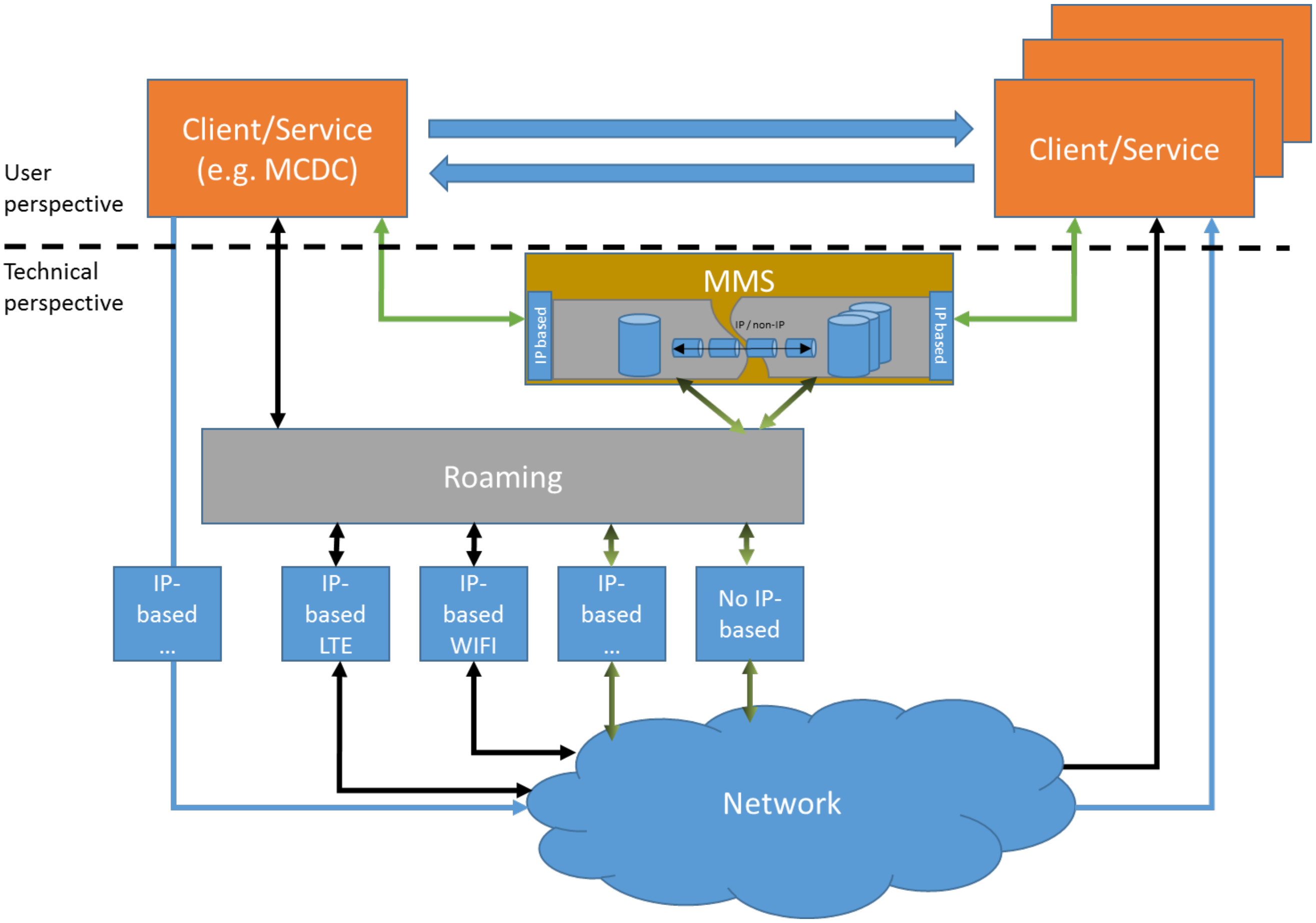


Figure 2 Logical perspective of communication and messaging in the Maritime Cloud

Figure 2 describes a logical view on the communication and messaging inside the Maritime Cloud. Thereby the upper part of the figure describes the user's perspective of the communication, whereas the lower part describes the technical perspective that is hidden as much as possible by the Maritime Cloud Demonstrator Component.

The communication infrastructure supports three types of communication scenarios, indicated by the arrow color in Figure 2.

1) IP-based communication (blue lines in Figure 2):
   The connection between client and services is instantiated using an existing internet connection.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

2) Roaming communication (black lines in Figure 2):
   The connection between client and service is done, using a roaming device which routes the requests for connections to the maritime cloud, via a suitable IP or non-IP connection. This solution also participates from a QoS that can be provided by the roaming component.
3) Maritime Messaging Server (green lines in Figure 2):
   Client (e.g. MCDC) uses the Maritime Messaging Service (MMS) to access a service in the Maritime Cloud. The MMS regulates the access request and connects to available communication channels with a defined QoS. The communication channel either is IP-based or benefits from the Roaming Device (black-green lines in Figure 2). In addition the MMS is able to queue incoming and outgoing messages for the case, no stable connection could be established with any of the available devices.

2.2 Maritime Cloud structure

This section gives a holistic view on the Maritime Cloud approach using the Maritime Architecture Framework (MAF) to show the structure and relation between intended functions of the Maritime Cloud in context with Information- and Data models as well as with used protocols and physical- and software components.

The Maritime Architecture Framework is a framework to support the development of system architectures by representing technical, functional and organizational aspects of a systems structure. It uses a structural framework to enable a visualization of those aspects in relation to each other and with the structure of the maritime domain [2].

2.2.1 Maritime Cloud components

The Maritime Clouds overall goal is to support a safe and reliable seafaring by facilitating information exchange across various communication channels, either IP-based or non IP-based.

The development of this approach within the EfficienSea2 project focuses mainly on the development of the key components Service Registry and Identity Registry (see Figure 3 below). With this technical services, the users are enabled to access any kind of service, which is registered in the Service Registry either via the Maritime Cloud Demonstrator Component (MCDC) as a “add-on” component to maritime systems (e.g. an ECDIS) or through a shore based application (e.g. the web-platform MC Portal[2] for managing services and identity’s of organizations in the Maritime Cloud, developed in an external project).

The communication between ship and shore is granted via the Roaming Device, which is currently specified in D2.7 [3] and D2.8 [4] in combination with existing components (AIS-Receiver and LAN-/WLAN-router etc.). The roaming device can be configured by the users

[2] https://portal.maritimecloud.net

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

through the MCDC to establish a specific communication link either via LAN / WLAN or other communication channels such as a radio waves.

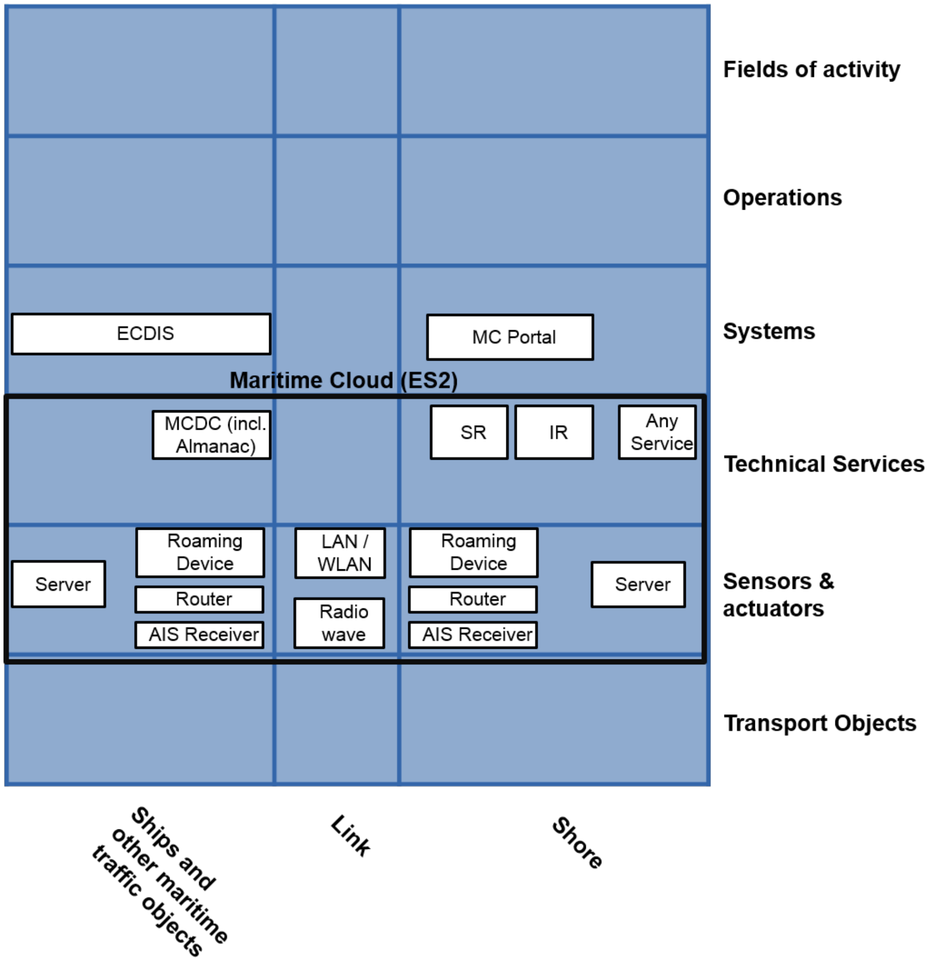
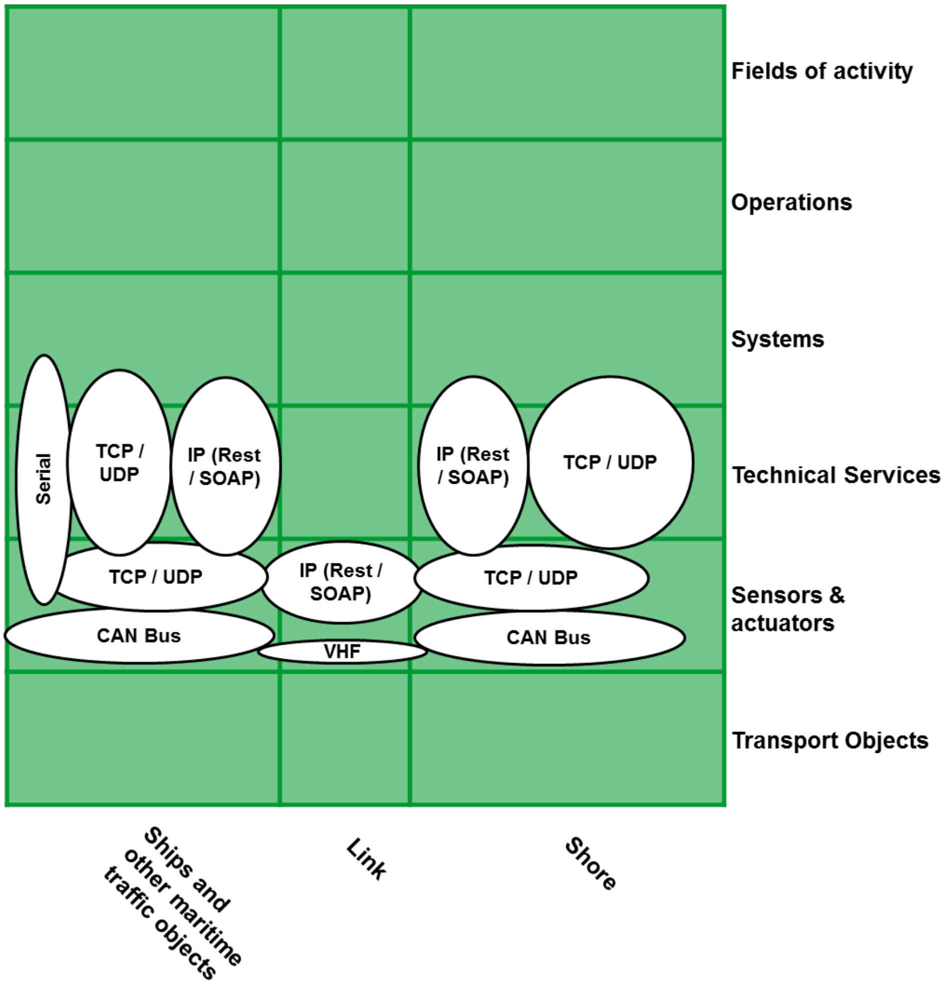


Figure 3 Maritime Cloud components displayed in the MAF

2.2.2 Maritime Cloud communication

The communication between the listed components in the previous subsection is established by using the following communication methods (see Figure 4). While using VHF for communication via AIS-Receiver, the common “Maritime Cloud” communication, either to the Service Registry or Identity Registry or other services, is done via TCP / IP based approach with web service technologies such as SOAP and REST. Furthermore, a Roaming Device enables the ship-side infrastructure to offer an additional serial interface to maritime components such as an ECDIS.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.



,

Figure 4. Communication within the Maritime Cloud displayed in the MAF

2.2.3 Maritime Cloud information exchange

The various Maritime Cloud components are used with different data models (such as data formats or APIs). As displayed in Figure 5, the Service Registry and Identity Registry provides their own API and data formats (sketched in section 3.2) to communicate with any kind of maritime cloud services or with the MCDC as ship-side Maritime Cloud component. Router and Roaming Device as well as AIS-Receiver are “routing” the information flow via TCP / UDP or CAN Bus to specific data facilitating technical services. The communication means of the different components enables the use of custom data formats of not yet defined Maritime Cloud services via REST / SOAP.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

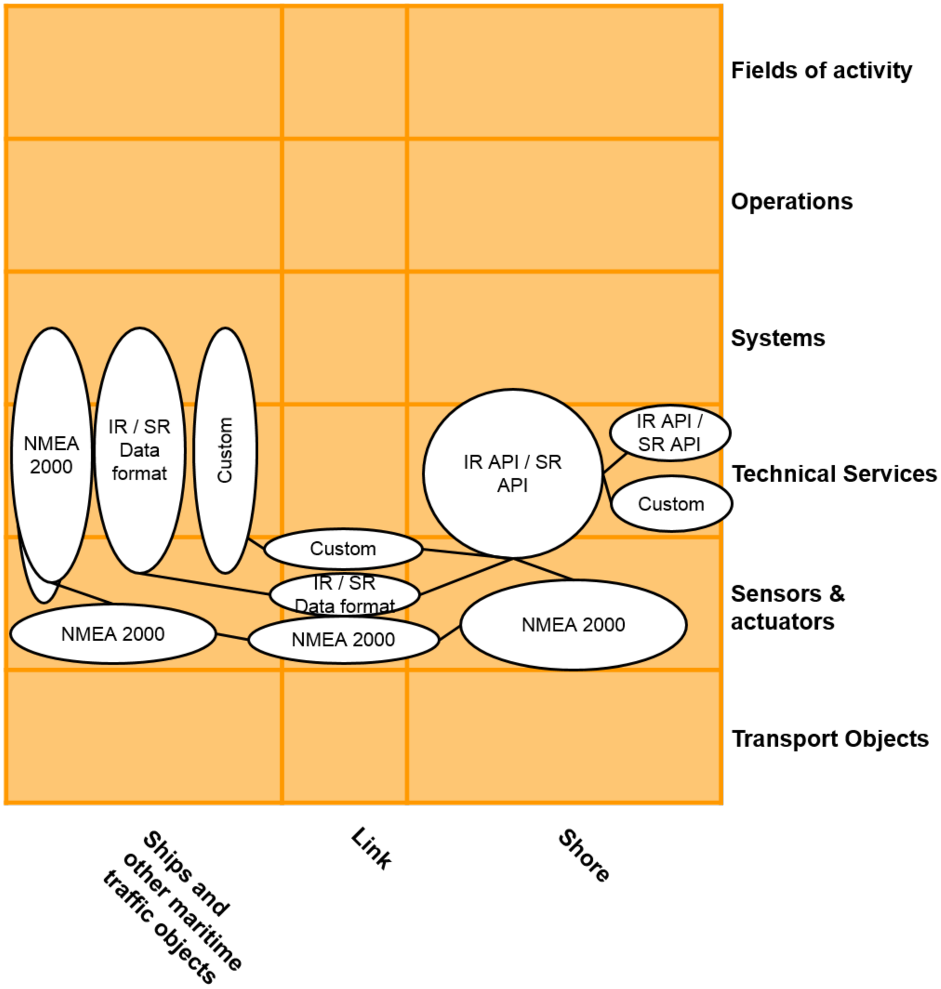


Figure 5 The data formats used in the Maritime Cloud

2.2.4 Conclusion

This subsection introduces a holistic view on the Maritime Cloud based on the sub-sections. The figure below takes the components, information aspects and communication methods as listed in further subsections and establish a layer model to contextualize them to each other. Furthermore, the top-layer shows the intended functions for the items in the underlying layers. This enables the view from perspectives on the Maritime Cloud structure (see Figure 6).

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

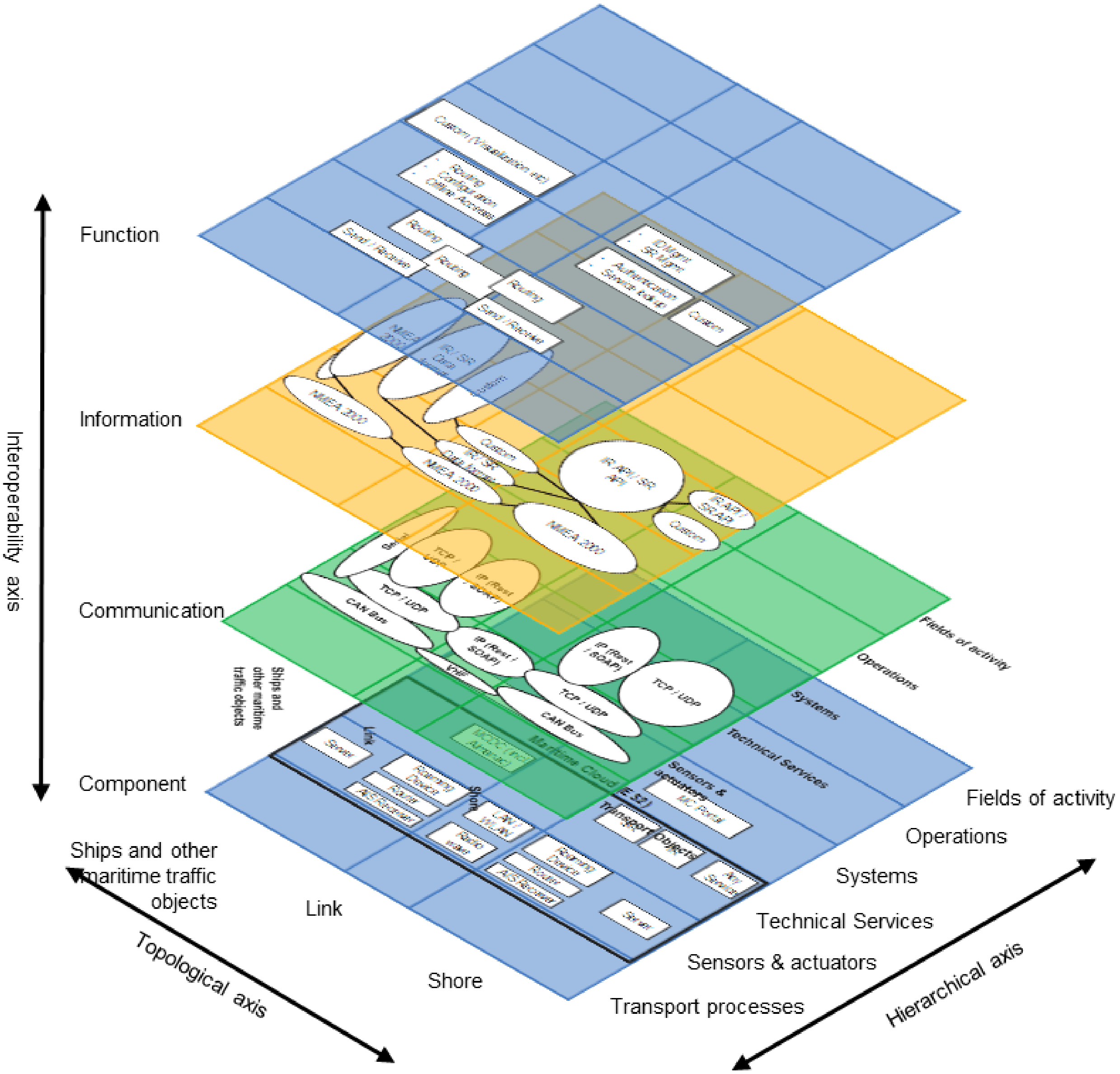


Figure 6 The Maritime Cloud structure

This deliverable describes in detail the different aspects of the Service Registry, Identity Registry as well as the MCDC in relation with the holistic maritime structure as described above, in the sections below. It starts with a conceptual description containing use cases as well as intended data models for the key components (Service Registry and Identity Registry) before the sections 3.4 and 3.5. concentrates on the dedicated functionalities of those components. The technical specification ends up with a description of the functional design in section 4.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

3 Conceptual Description

3.1 User and Use Cases

This section will describe the functionality of the Maritime Cloud core components (MC^3) from the perspective of an MC user.

Table 1 Possible user roles for the Maritime Cloud

Role	Description	Rights
Service User	Regular users of the Service Registry.	Users of this role are allowed to lookup the service registry for service specifications, designs, instances. They can also be found (and authenticated) within the Identity Registry.
Service Administrator	Role for administration of services in the Service Registry.	Users of this role are allowed to register service specifications, designs, instances and to modify such registrations.
Organization Administrator	Role for administration of identities in the Identity Registry	Users of this role are allowed to register and modify entities (User, Services, Devices) in the Identity Registry.
Service Registry Administrator	Role for administration of the service Registry.	Users of this role are allowed to administrate user roles and access rights in the service registry. Note that this only covers the access rights for accessing the Service Registry. Not to be confused with the rights for accessing other services: such access rights administration for other services is not performed in the Service Registry, but is in the responsibility of the other services themselves.
Identity Registry Administrator	Role for the administration of the Identity Registry	Users of this role are allowed to administrate user roles and access rights in the Identity Registry
Service Registry Super User	Local super user of the Service Registry	This user is not published. Allows to manually create *Service Registry Administrator* users.

Since both, the Identity Registry as well as the Service Registry can be seen as a service, the *Organization Administrator* and the *Service Administrator* inherit the roles of a Service User. In addition, they have to perform Service User tasks, in order to fulfill the administration tasks (see section 3.3.2). The *Service Registry Super User* and the *Identity Registry Super User*

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

are local administrators, which are only relevant in the company, hosting either the Service or Identity Registry. Thus their roles will not be further considered.

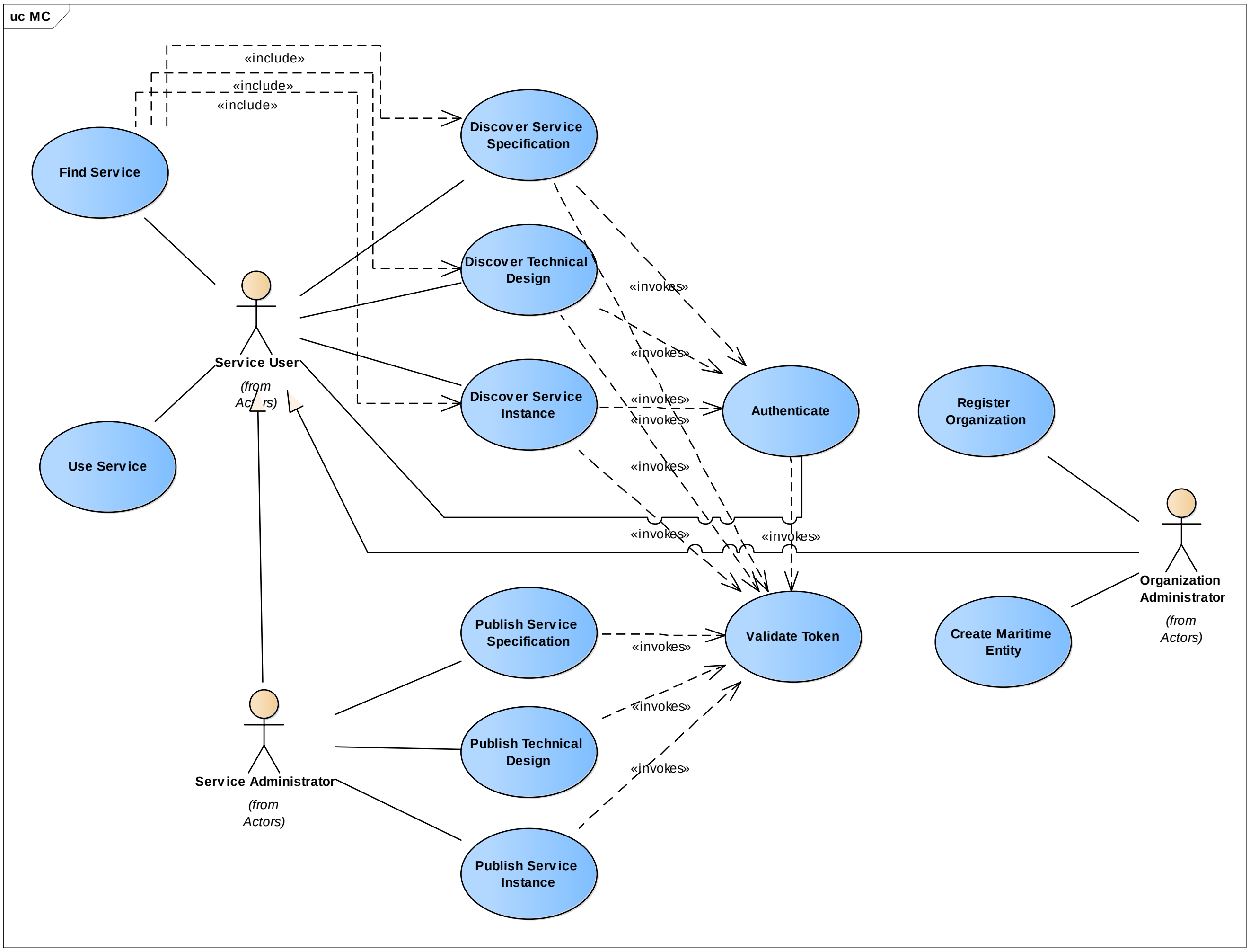


Figure 7 identified use cases for the Identity and Service Registry

Figure 7 describes the identified use cases for the MC components IR and SR, while

Table 2 gives a short description as well as a link to those sections that handle the use case in more detail.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

Table 2 Use Case summary for the Maritime Cloud

ID	Name	Description	Section
1	Find Service	Find a service in the service registry	3.3
	Discover Service Specification	Find one or more service specifications.	3.3.1, 4.2.1, 4.3.4
3	Discover Technical Design	Find one or more technical designs, which realize a service specification.	3.3.1, 4.2.1, 4.3.4
4	Discover Service Instance	Find one or more service instances for a technical specification	3.3.1, 4.2.1, 4.3.4
5	Authenticate User	Authenticate an actor	4.3.1, 4.3.3
6	Validate Token	Check if an provided token is valid and thus if the the calling identity is trustworthy.	4.3.1,
7	Publish Service Specification	Publish a service specification, according to the Service Documentation Guideline inside of the Service Registry	3.3.2, 4.2.2, 4.3.5
8	Publish Technical Design	Register a techncial design for one or more service specifications, inside the Service Registry	3.3.2, 4.2.2, 4.3.5
9	Publish Service Instance	Register a service instance for one or more technical designs inside of the Service Registry	3.3.2, 4.2.2, 4.3.5
10	Register Organization	Register a new organization in the Identity Registry.	3.4.1, 4.2.2, 4.3.2
11	Register Maritime Entity	Register a maritime entity (human user, device, vessel or service) inside the Identity Registry	3.4.1, 4.2.2, 4.3.2
12	Use Service	Using an external (IP-based) service through the MCDC	3.5

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

3.2 Information Model

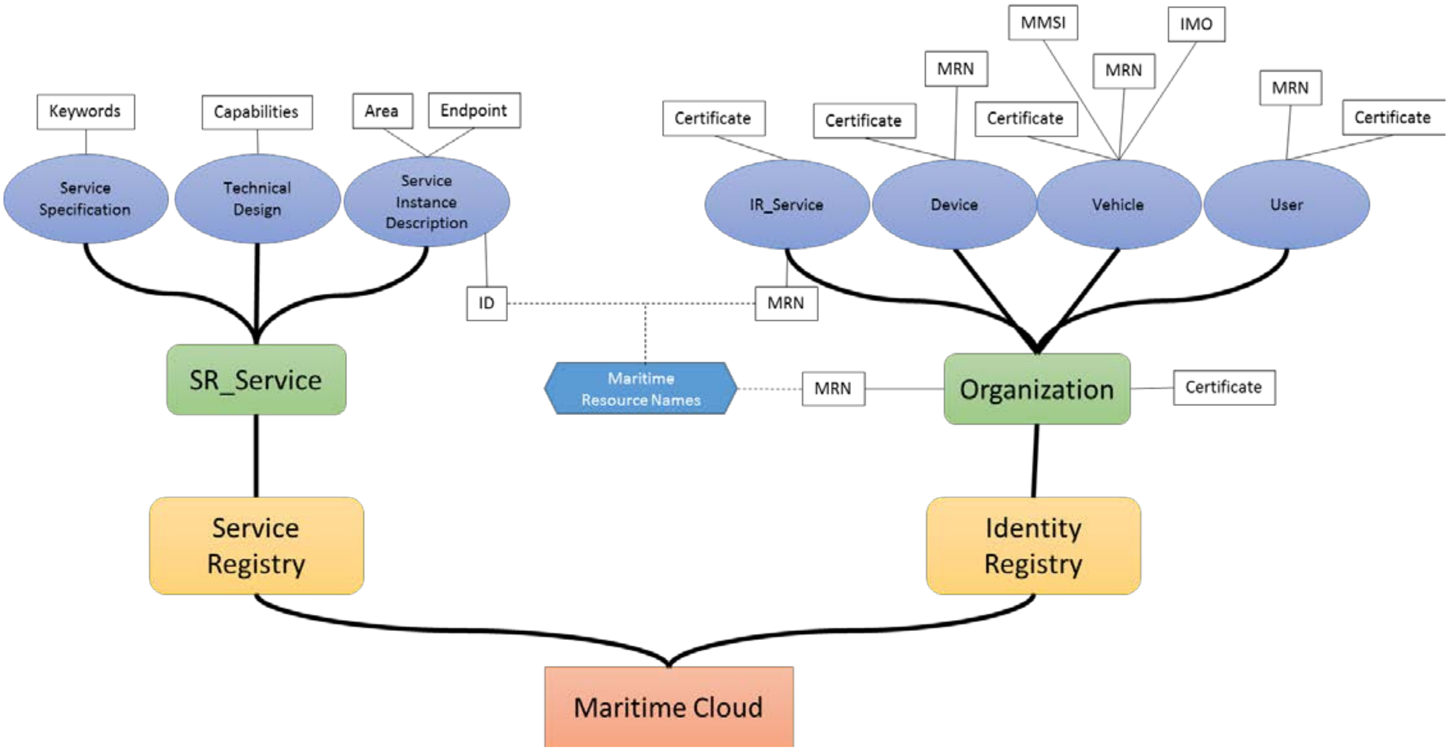


Figure 8 Mind map for the data model of the Maritime Cloud

Figure 8 gives an overview about the most relevant information, stored in the Service and Identity Registry.

3.2.1 Service Registry

As it can be seen on the left side of the picture, the Service Registry contains a list of “SR_Services”, which should not be confused with the “IR_Service” of the Identity Registry. Such an SR Service is further subdivided into a Service Specification, a Technical Design and a Service Instance Description.

1) **Service Specification**

   The service specification describes one dedicated service at logical level in a technology-agnostic manner.
   Among other, the Service Specification contains a list of freeform keywords that describes the service.

2) **Technical Design**

   The technical design describes the details of a specific service specification. Thereby the technical design provides information about the actual realization of the service with a dedicated technology decision (depicted in Figure 8 as “Capabilities”).

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

3) **Service Instance Description**

A service implementation (implemented according to a given technical design) may be deployed at different locations by different service providers. For each such service instance a service instance description shall be provided.
Among others, the service instance description contains the endpoint for the service implementation and thus how to access the service. In addition the description optionally contains an area of application, which is the area the specific service implementation is valid for. For example: A VTS – Service like Jade-Traffic that is only valid inside the Jade-fairway.

The relations among these three elements are visualized in Figure 9.

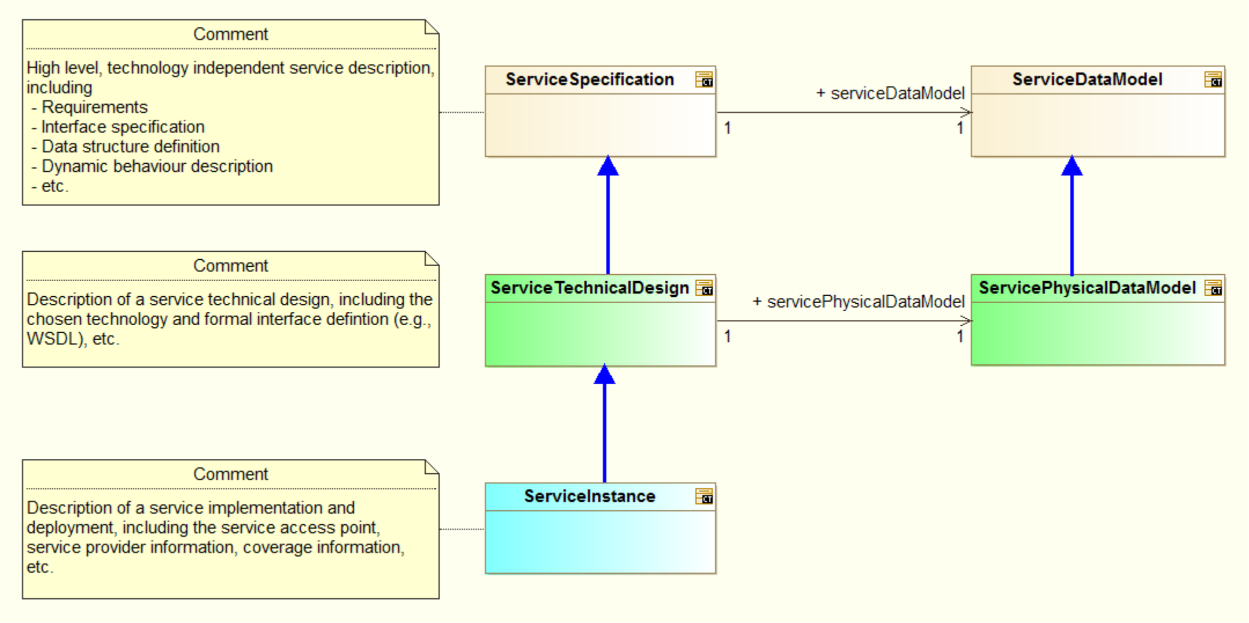

Comment
High level, technology independent service description, including
- Requirements
- Interface specification
- Data structure definition
- Dynamic behaviour description
- etc.
ServiceSpecification
+ serviceDataModel
1
1
ServiceDataModel
Comment
Description of a service technical design, including the chosen technology and formal interface defintion (e.g., WSDL), etc.
ServiceTechnicalDesign
+ servicePhysicalDataModel
1
1
ServicePhysicalDataModel
Comment
Description of a service implementation and deployment, including the service access point, service provider information, coverage information, etc.
ServiceInstance

Figure 9 Distinction between Service Specification, Service Technical Design and Service Instance [5]

For more information about the structure of the Service Specification, Technical Design and Service Instance Description, please refer to Deliverable "D3.6 - Service Documentation Guidelines" [5].

3.2.2 Identity Registry

The Identity Registry, as shown in Figure 8 is composed of multiple organizations. Each organization can contain different types of maritime entities, such as human users, vessels, devices and services.
At this point, it's important to know that the organization a maritime entity belongs to does not necessarily is the owner (or employer) of the corresponding entity, but need to be able to prove its identity.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

Each of these maritime entities contains a maritime resource name (MRN), as described in [6] as unique identifier.

1 Organization

In the Maritime Cloud an *organization* is an entity, such as an institution, company or an association that has a collective goal and is linked to an external environment. Examples, include international organizations such as IMO, IALA, and IHO. National authorities such as US Coastguard, Swedish Maritime Administration. Local authorities such as VTS-Oeresund, Port of Rotterdam, Hong Kong SAR. Or commercial companies such as Transas or Maris.

2 Users

Users mainly refers to human users. Human users differ from other actors in that they typically use a username/password to login which implies a different interaction pattern with the identity registry then say communication between vessels.

3 Services

Services refers to digital services, as described above. For example, a weather service that is available to other services for machine to machine communication. Services needs to be registered in such a way that it can successfully authenticate users.

4 Vessels

Vessels describes any floating object used for the carriage of people or goods. The main need for registering vessels in the Maritime Cloud is so that digital authentication certificates can be issued for them. Thereby enabling secure communication between vessels as well as digitally signing of documents. Users might also use these authentication certificates for other purposes. The important thing is that the functionality is there. As part of the authentication certificate of a vessel its name, MMSI number, IMO number, call sign and possible other attributes is included in the header of the authentication certificate.

5 Devices

Devices can be any number of entities that is not covered by the other entity types. It could for example be a lighthouse, an ECDIS or a server that needs to be able to authenticate itself. For example an ECDIS system.

As both, the SR and the IR contain a “Service” element, the Identity Registry service represents an identifiable service that links to a service implementation (e.g. a service instance description) in the Service Registry. In this case, both service representations share

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

the same maritime resource name (MRN). On the other hand it's not required that a service is registered within the Identity Registry to be registered in the Service Registry.

3.3 Functionalities of Service Registry

The functionality of the service registry is separated into two areas 1) the discovery of services and 2) the management of those.

3.3.1 Service Discovery

Main objective of the Service Registry is to enable the discovery of maritime services for a large number of users. Those users may be human actors, like a nautical officer or artificial devices, an ECDIS or a weather service.

As described in the following Figure 10, the process of finding service instances is split into 3 successive steps, together with one optional step, performed on the user's side.

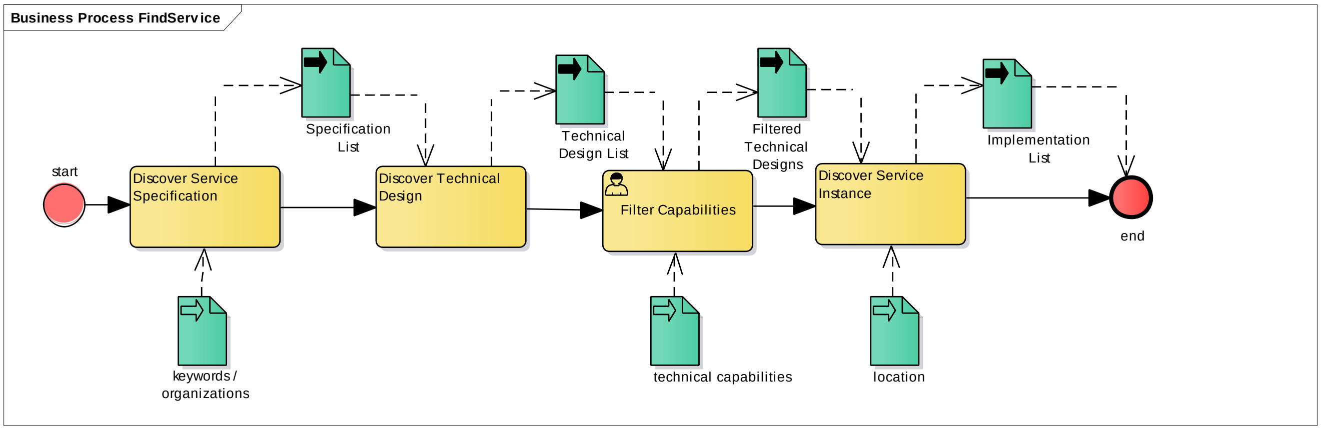


Figure 10 Procedure to find one or more specific services

As indicated through the different input values for the discovery activities, services may be searched for different criteria's, like keywords, locations, organizations or combinations of those. The result of each discovery activity is a list of service documentations, which serve as input for the next step. In addition the used technical equipment can perform filtering ("filter compatibility" in Figure 10) with respect to its own capabilities. Those capabilities may contain, but are not restricted to, the communication means (REST / SOAP).

Figure 10 describes an uninformed search, e.g. there is no additional information present at the beginning. If on the other hand prior knowledge about for example the Service Specification is available, the "Discover Service Specification" step may be omitted. Same goes for the Technical Design.

Taking into account that the Service Registry may be used either by humans or machines, the Service Registry offers two approaches to discover services, a manual discovery for human users and a machine to machine discovery process, as shown in Figure 11 and Figure 12.

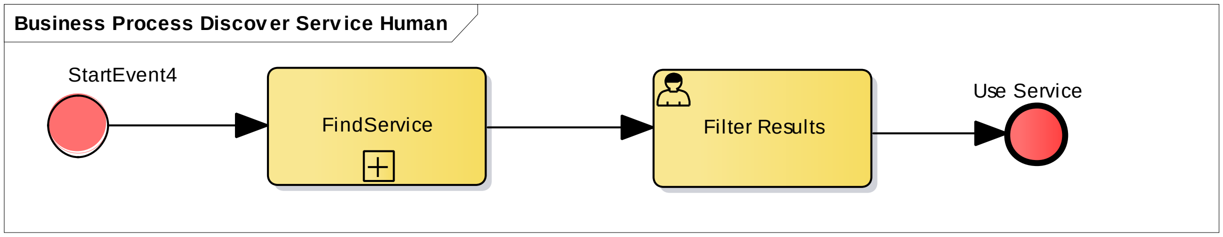


Figure 11 Process to find a service as a human user

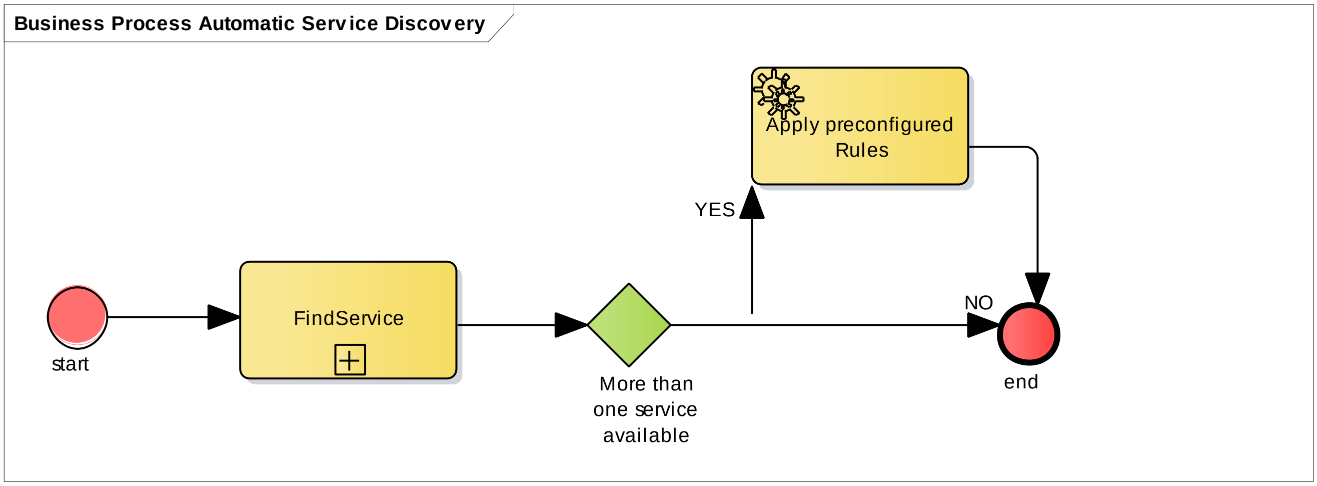

Business Process Automatic Service Discovery
Apply preconfigured Rules
YES
NO
start
FindService
More than one service available
end

Figure 12 Process to find a service as a machine

Both processes use the sub process “FindService” (as described in Figure 10) that contains the actual realization of the Service Registry’s functionality.

The main difference between the manual and the automatic service discovery process is the need of unique answers from the service registry, in case of automatic discovery. This is archived through a “spatial exclusive” flag that is, for this service specification there is only one combination of service specification and technical design, valid within a defined area / location. To ensure the uniqueness of services (e.g. who is allowed to set the exclusive flag) appropriate governance processes have to be established in the future. If no such governance process exists or does not apply for a specific service, the filtering through the preconfigured rules has to ensure that exactly one service is selected.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

3.3.2 Service Management

The following Figure 13 shows the activities that have to be performed, when a Service Specification, a Technical Design or a Service Instance Description shall be registered within the Service Registry.

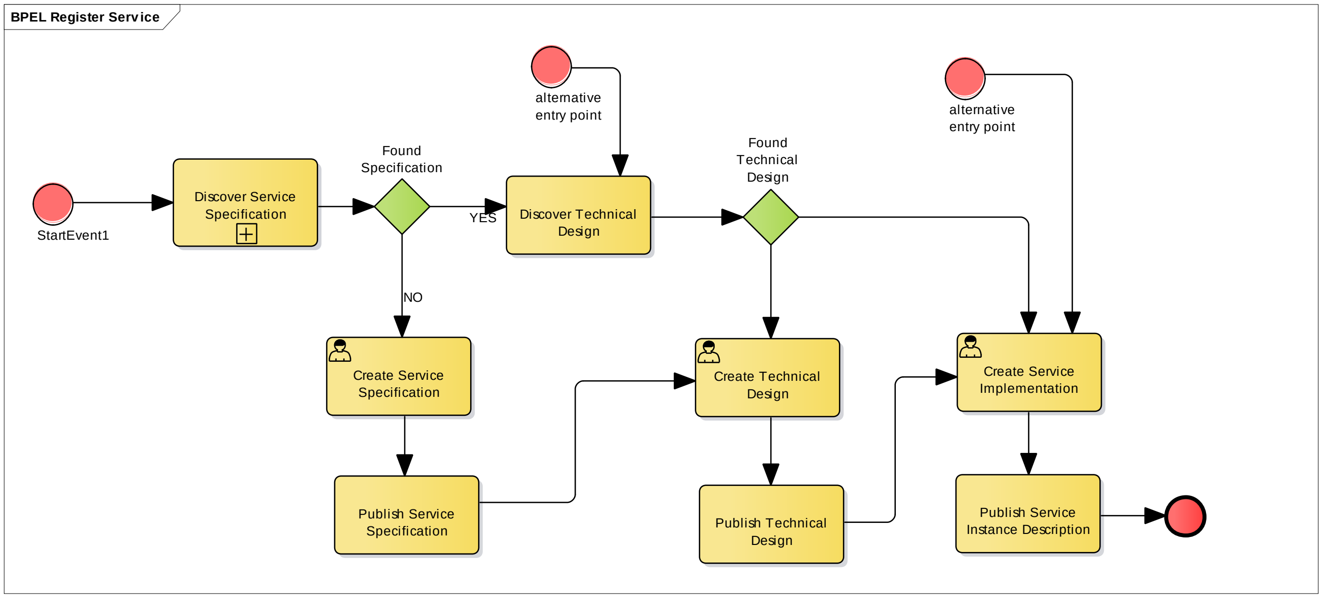

BPEL Register Service
StartEvent1
Discover Service Specification
Found Specification
YES
NO
alternative entry point
Discover Technical Design
Found Technical Design
alternative entry point
Create Service Specification
Publish Service Specification
Create Technical Design
Publish Technical Design
Create Service Implementation
Publish Service Instance Description

Figure 13 Activity Diagram for management of services in the service registry. Green activities are manual activities, not handled here.

Here the “user – activities” (marked through the user icon) describe the creation of the required documents, which is not further considered here. Furthermore the user may enter the described workflow at different entry points, depending on the available information. That is, if the technical design to be implemented is already known, the service implementation may be created and published without the previous steps.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

3.4 Functionalities of Identity Registry

Besides discovery and management of services, within the Maritime Cloud, there is a need to authenticate all maritime resources before providing or using a service. This is done by the Identity Registry.

As the Service Registry the Identity Registry can be subdivided into the two sections as well. This two sections include the management of identities and the authentication of maritime resources.

3.4.1 User Federation

The Maritime Cloud Identity Registry uses a federated system of identity providers, to manage the maritime identities.

Federation is the means of linking distinct identity management systems to a person's electronic identity and attributes. For example, a shipping company might expose all their users in LDAP or Active Directory to the Maritime Cloud in such a way as they appear as Maritime Cloud users. Thereby bypassing the need to manage their users directly in the Maritime Cloud. This also means that the Maritime Cloud is not responsible for management of users. In practical terms, federation means that users asked to authenticate in the Maritime Cloud will be redirected to a login webpage supplied by their organization where they can login using their organizational id.

An identity provider has to be compliant with the OpenID Connect[3] protocol, to be compliant with the Identity Registry.

3.4.1.1 Security

The federation approach allows each registered organization to manage their users according to their own security and privacy policies. This significantly increases for example the security of user data.

However the identity broker, provided by the Maritime Cloud Identity Registry, serves as "man in the middle" and thus becomes an issue regarding the reliability of the Identity Registry. If the identity broker is not available for any reason, none of the users or services within the Maritime Cloud would be able to authenticate themselves.

Within the EfficienSea 2 testbeds this issue will not be addressed at first. For a later, operational operation, the usage of several redundant identity brokers is considered, to accomplish a failsafe authentication.

[3] http://openid.net/

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

3.4.2 Identity Management

As shown in Figure 8 each maritime entity is bound to a trusted organization. This has to be considered in the process of creating new identities (see Figure 14).

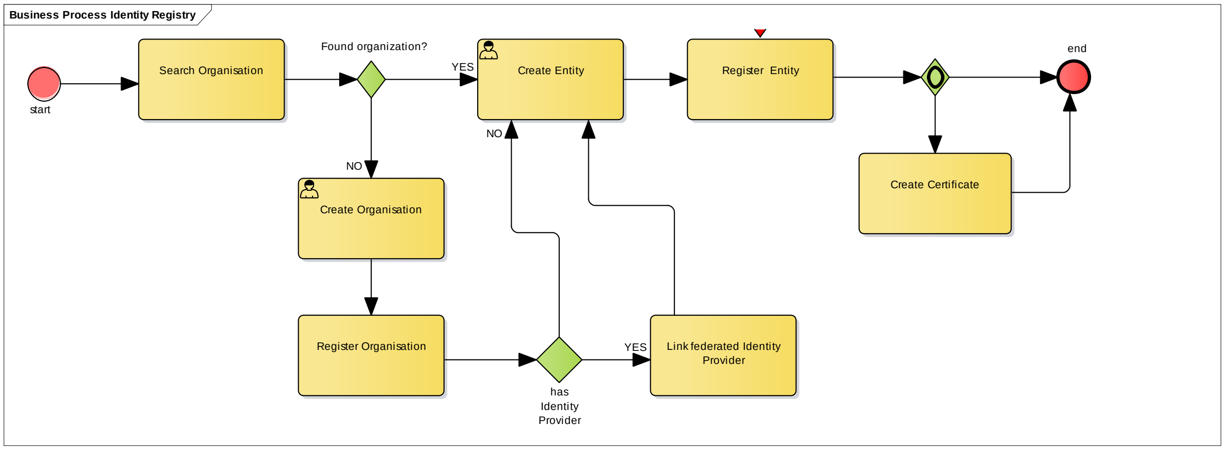


Figure 14 Process to create a new maritime identitity in the Identity Registry

Figure 14 is using the same graphical notation as Figure 13, in terms of “user- activities”. E.g. the processes of creating an organization, or creating an entity is the collection of all required information, to be feed into the Identity Registry and is done by an Identity Registry Administrator (see Table 1).

Like in case of the Service Registry, may different starting points can be used, if prior knowledge is available. If for example the organization is known, its identifier has not to be determined using the “Search Organization” activity.

3.5 Maritime Cloud Demonstrator Component

The Maritime Cloud Demonstrator Component (MCDC) is meant as a reference implementation, to demonstrate the correct usage of the identity and service registry, as well as the offline copy of those. For this purpose it will provide an easy to use interface for application developers, who want to interface their products with the Maritime Cloud as described in Figure 15.

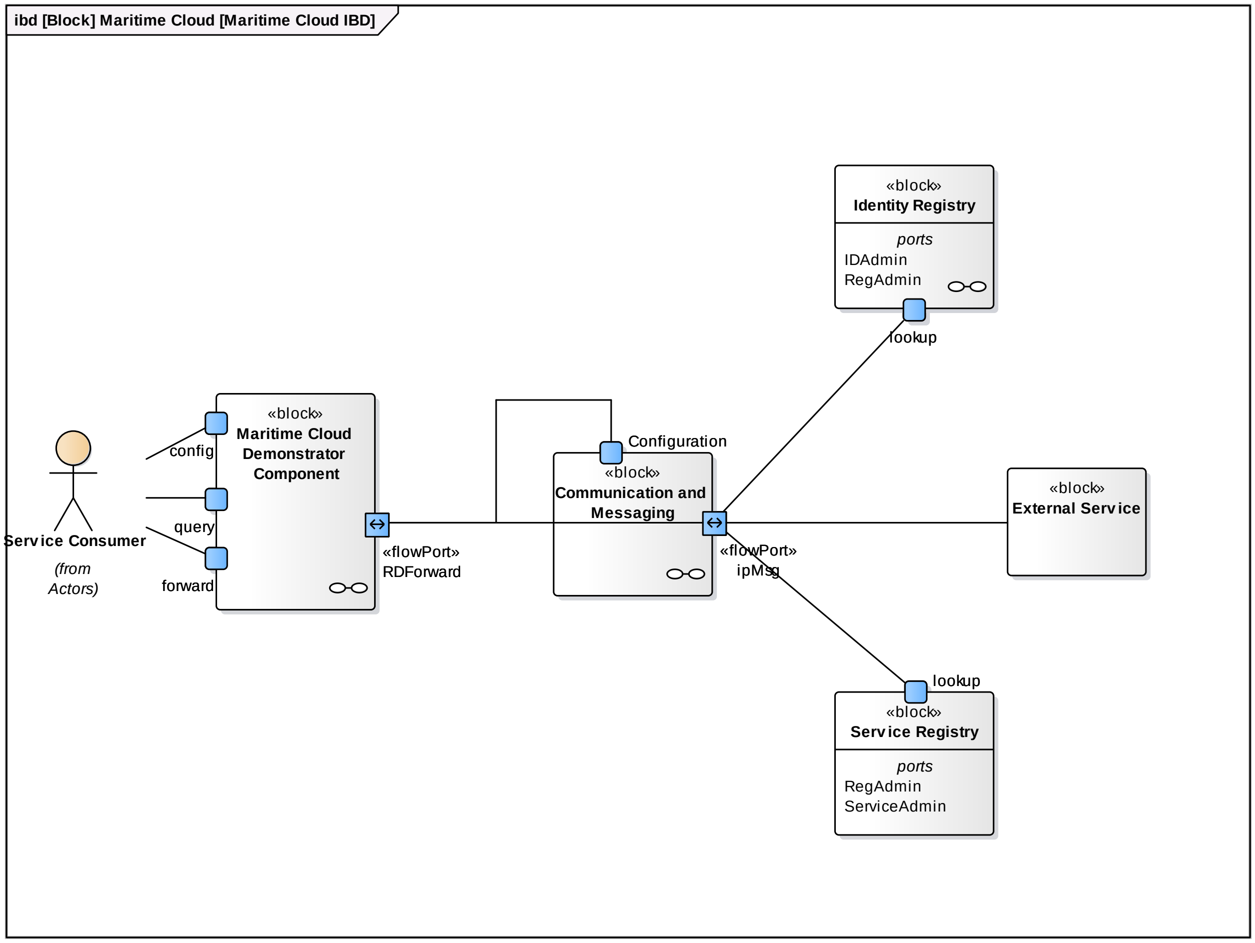


Figure 15 Technical overview over the components of the maritime cloud and the interfaces for service users.

The following Figure 16 shows an early consideration, for the internal structure of the MCDC, which can be decomposed into the following five components.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

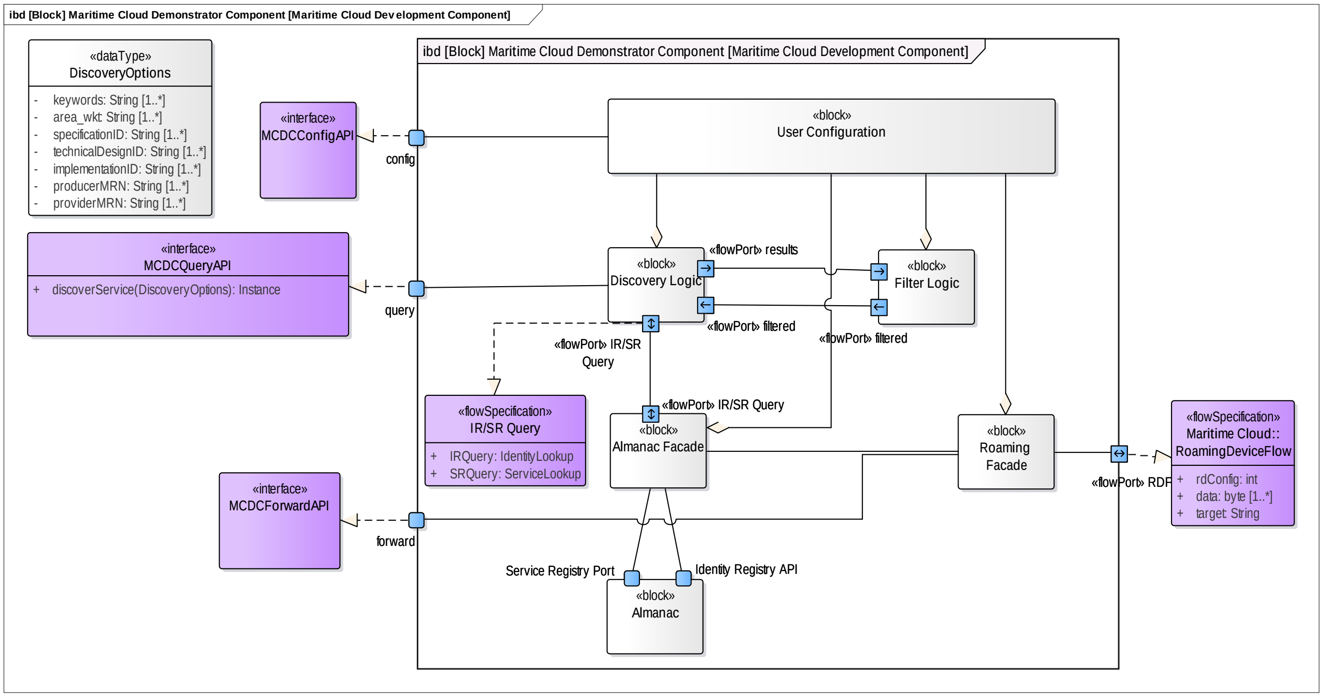


Figure 16 internal structure of the Maritime Cloud Demonstrator Component

- **User Configuration:**
  This component stores user preferences on how to access the maritime cloud. These user preferences, can be manipulated by the user and contain for example information about service provider, the user (or his organization) has contacts with.
- **Discovery Logic:**
  The discovery logic interacts with the Service and Identity Registry, to discover service instances for the user. Thereby the Discovery Logic calls the Service Registry API in the correct order, for the current discovery operation, taking into account preliminary knowledge like already known service specification ids (see below).
- **Filter Logic:**
  The Filter Logic is used to filter out results, from the Service Registry, which are of no interest for the user. At this point the Filter Logic interacts with the user preferences as well as with discovery options.
- **Almanac Facade:**
  The Almanac Facade is used to decide whether to use the online or the offline version (Almanac) of the Service and Identity Registry. Depending on the result the request is either forwarded to the Almanac component or to the external service / identity registry using the Roaming Facade.
- **Roaming Facade:**
  The Roaming Facade is used to communicate with the outer world with the assistance of the Roaming Device, developed in EfficienSea 2 (Work package 2). For this purpose requests to external sites will be passed through the Roaming Facade to the

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

Roaming Device. In addition the Facade is able to configure the Device according to the user’s requirements, by making use of the User Configuration component.

3.6 Almanac

The Almanac is an offline copy of the service and identity registry, to be used, if no stable internet connection could be established. It will provide the same interface as the real Service and Identity Registry but runs on a local server on board of a vessel.
At the time, writing this document, the Almanac has not been fully specified.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

4 Functional Design

The following section will provide an overview over the available interfaces and interactions for the Maritime Cloud Core Components, namely Service and Identity Registry.

Thereby Figure 17 shows a subset of public available interfaces of the core components, as well as an assignment to the user roles defined in Table 1. The interfaces are described in more detail in the following sections 4.2.1(SR) and 4.2.2 (IR).

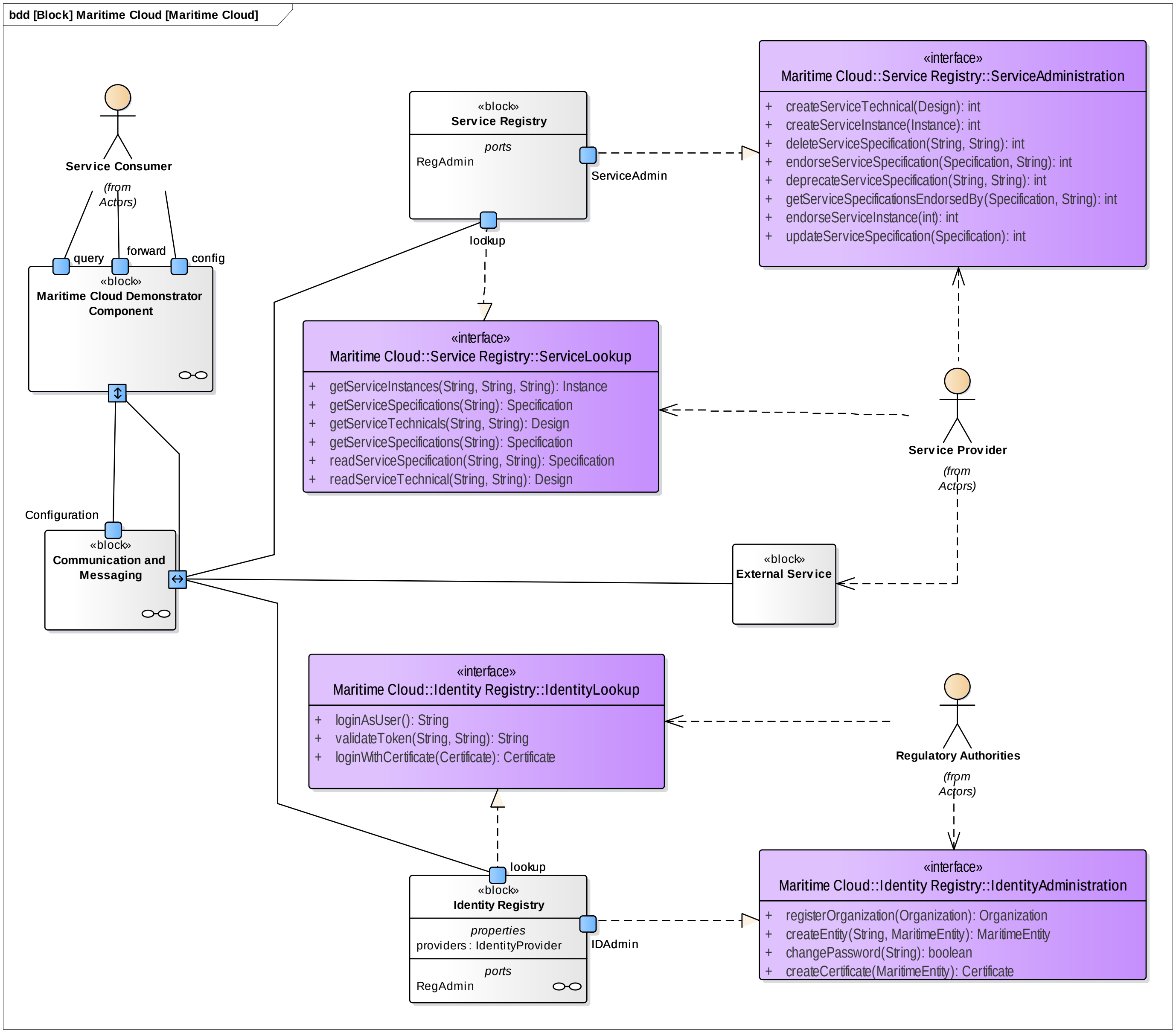

bdd [Block] Maritime Cloud [Maritime Cloud]
Service Consumer
(from Actors)
query
forward
config
«block»
Maritime Cloud Demonstrator Component
«block»
Service Registry
ports
RegAdmin
ServiceAdmin
lookup
«interface»
Maritime Cloud::Service Registry::ServiceAdministration
+ createServiceTechnical(Design): int
+ createServiceInstance(Instance): int
+ deleteServiceSpecification(String, String): int
+ endorseServiceSpecification(Specification, String): int
+ deprecateServiceSpecification(String, String): int
+ getServiceSpecificationsEndorsedBy(Specification, String): int
+ endorseServiceInstance(int): int
+ updateServiceSpecification(Specification): int
«interface»
Maritime Cloud::Service Registry::ServiceLookup
+ getServiceInstances(String, String, String): Instance
+ getServiceSpecifications(String): Specification
+ getServiceTechnicals(String, String): Design
+ getServiceSpecifications(String): Specification
+ readServiceSpecification(String, String): Specification
+ readServiceTechnical(String, String): Design
Service Provider
(from Actors)
Configuration
«block»
Communication and Messaging
«block»
External Service
«interface»
Maritime Cloud::Identity Registry::IdentityLookup
+ loginAsUser(): String
+ validateToken(String, String): String
+ loginWithCertificate(Certificate): Certificate
Regulatory Authorities
(from Actors)
lookup
«block»
Identity Registry
properties
providers : IdentityProvider
ports
RegAdmin
IDAdmin
«interface»
Maritime Cloud::Identity Registry::IdentityAdministration
+ registerOrganization(Organization): Organization
+ createEntity(String, MaritimeEntity): MaritimeEntity
+ changePassword(String): boolean
+ createCertificate(MaritimeEntity): Certificate

Figure 17 Public interfaces of the Maritime Cloud

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

4.1 Data types

The following to Figures (Figure 18 and Figure 19) give a brief overview about important information, stored within the Service and Identity Registry.

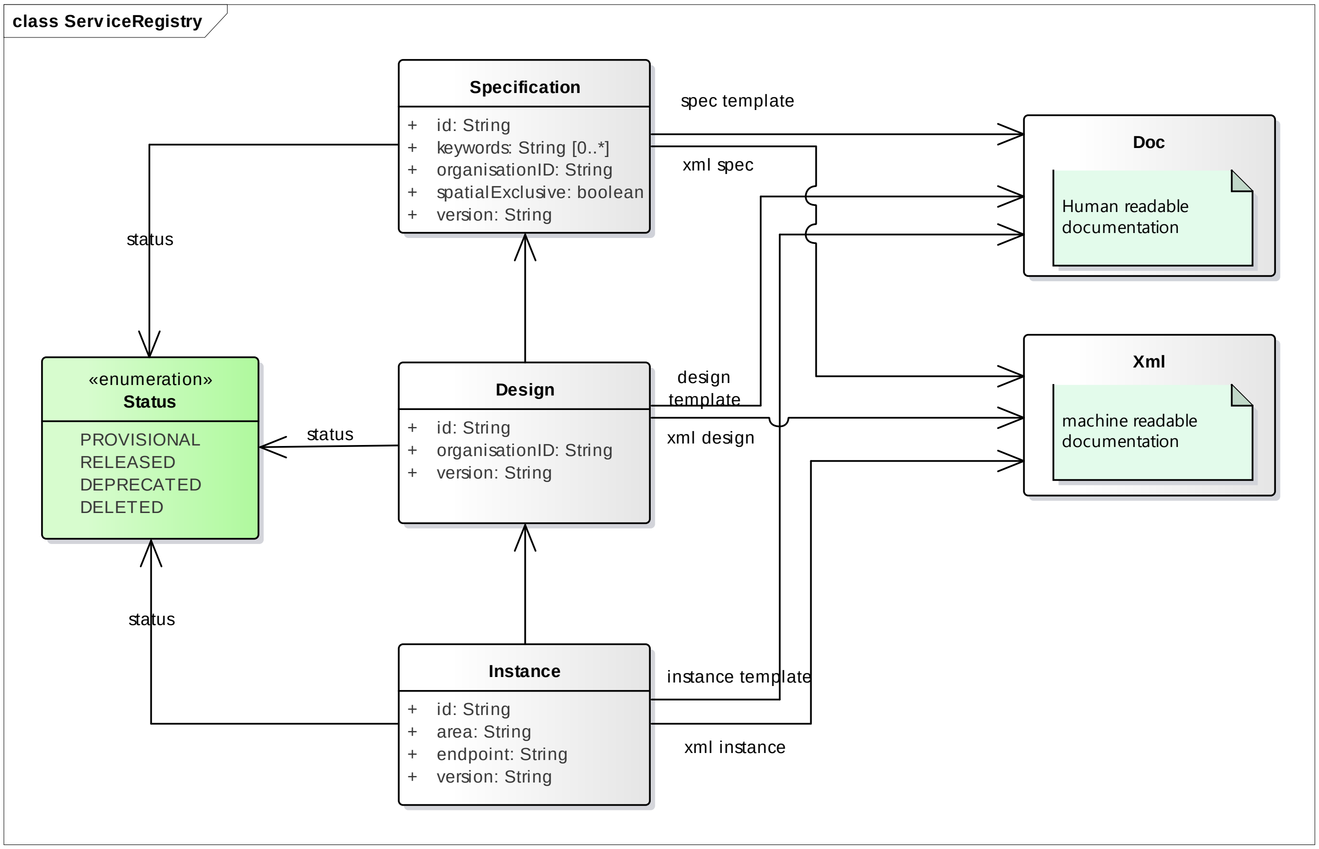


Figure 18 Data types for Service Registry

The Service Registry contains the Service Specification, Technical Design and Instance Description, as described in Deliverable D3.6 (Service Documentation Guidelines [5]). Those documents shall be available in a human readable and a machine readable format. For a complete description of the Service Registry Data Model, refer the Service Documentation Guideline.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

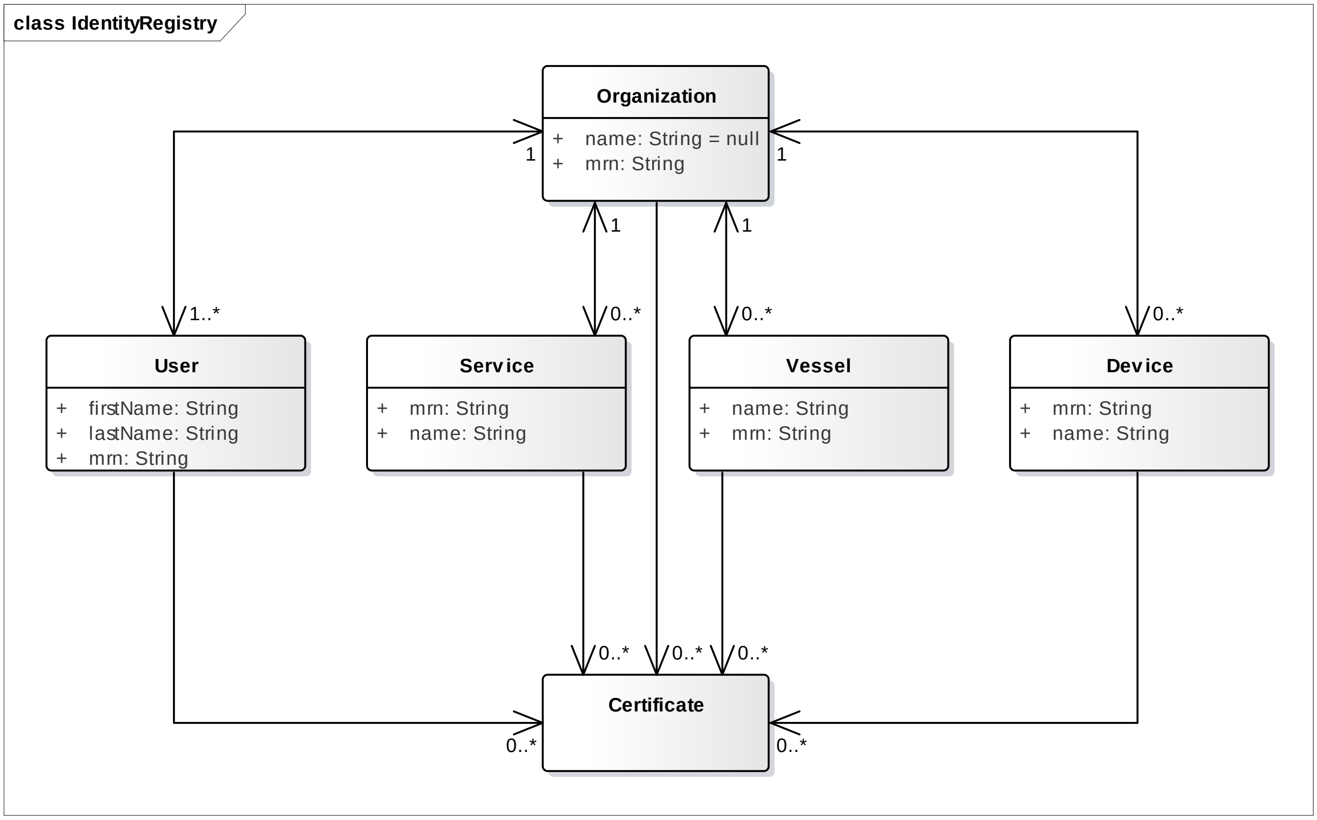


Figure 19 Data types for Identity Registry

The Identity Registry is organized around an organization. Each maritime entity has to be associated to exactly one organization which is able to proof its identity. In addition each maritime entity as well as each organization can own certificates, used for authentication. Certificates used within the Maritime Cloud are based on the X.509 standard for M2M communication. For more information about the used certificates refer to the article Identity Management and Cyber Security [7][4].

[4] Also available: https://maritimecloud.net/docs/Identity%20Management%20and%20Cyber%20Security.pdf

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

4.2 Interfaces

4.2.1 Service Registry API

The service registry supports the following API.

The first column of the table contains the related area of the SR that is modified with the API call. The second column contains the API call, whereas the third column contains the desired result.

Table 3 Public Service Registry API

Area	Function	Result
Service Specification	*createServiceSpecification(serviceSpecification)*	*Status*
Service Specification	*getServiceSpecifications(organisationId)*	*list of service specifications including some metadata, e.g., lastupdated, lastupdatedby, etc.*
Service Specification	*getServiceSpecifications(keywords)*	*list of service specifications*
Service Specification	*readServiceSpecification(serviceSpecificationId, versionID)*	*ServiceSpecification (including human readable documents)*
Service Specification	*updateServiceSpecification(serviceSpecification)*	*Updated ServiceSpecification*
Service Specification	*deleteServiceSpecification(serviceSpecificationId, versionID)*	*Deleted ServiceSpecification (logical deletion)*
Service Specification	*deprecateServiceSpecification(serviceSpecificationId, versionID)*	*Deprecated ServiceSpecification*
Service Specification	*endorseServiceSpecification(serviceSpecificationId, endorsingOrganizationId)*	*Status*

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

Service Specification	*revokeEndorsementServiceSpecification(serviceSpecificationId, endorsingOrganizationId)*	*Status*
Service Specification	*getServiceSpecificationsEndorsedBy(organisationId)*	*list of service specifications*
Service Technical Design	*createServiceTechnical(serviceTechnicalDesign)*	*Status*
Service Technical Design	*getServiceTechnicals(organisationId | serviceSpecificationId)*	*list of service Technicals Designs, including some metadata, e.g., lastupdated, lastupdatedby, etc.*
Service Technical Design	*readServiceTechnical(serviceTechnicalDesignId, versionID)*	*ServiceTechnicalDesign (including human readable documents)*
Service Technical Design	*updateServiceTechnical(serviceTechnicalDesign)*	*Updated ServiceTechnicalDesign*
Service Technical Design	*deleteServiceTechnical(serviceTechnicalDesignId)*	*Deleted ServiceTechnicalDesign*
Service Technical Design	*deprecateServiceTechnical(serviceTechnicalDesignId, versionID)*	*Deprecated ServiceTechnicalDesign*
Service Technical Design	*endorseServiceTechnical(serviceTechnicalDesignId, endorsingOrganizationId)*	*status*
Service Technical Design	*revokeEndorsementServiceTechnical(serviceTechnicalDesignId, endorsingOrganizationId)*	*status*
Service Technical Design	*getServiceTechnicalsEndorsedBy(organisationId, [serviceSpecificationId])*	*list of service Technical Designs*
Service Instance	*getServiceInstances(<[serviceSpecificationId]|[technicalDesignId]>)*	*list of service Instances*

Service Instance	*getServiceInstances(serviceSpecificationId, technicalDesignId, simpleOGCgeometrytypes)*	*list of service Instances*
Service Instance	*createServiceInstance(serviceInstance)*	*Status*
Service Instance	*readServiceInstance(serviceInstanceId, versionID)*	*ServiceInstance*
Service Instance	*updateServiceInstance(serviceInstance)*	*Updated ServiceInstance*
Service Instance	*deleteServiceInstance(serviceInstanceId, versionID)*	*Deleted ServiceInstance*
Service Instance	*endorseServiceInstance(serviceInstanceId, endorsingOrganizationId)*	*Status*
Service Instance	*revokeEndorsementServiceInstance(serviceInstanceId, endorsingOrganizationId)*	*Status*
Service Instance	*getServiceInstancesEndorsedBy(organisationId, [serviceSpecificationId])*	*list of service Instances*

4.2.2 Identity Registry API

Table 4 Public Identity Registry API

Area	***Function***	**Result**
Organizations	*apply(organizationDetails)*	Status
Organizations	*readOrganization(organizationId)*	organizationDetails
Organizations	*updateOrganization(organizationDetails)*	Status
Organizations	*getOrganizations()*	list of all organizations
Services	*getServices(organizationId)*	list of service for organization
Services	*createService(serviceDetails)*	Status
Services	*readService(serviceId)*	Service details
Services	*updateService(serviceDetails)*	status
Services	*deleteService(serviceId)*	Status
Services	*createServiceCertificate(serviceId)*	certificate and public and

		private keys
Services	*revokeServiceCertificate(serviceId, CertificateId, revokeReason)*	Status
Vessels	*getVessels(organizationId)*	list of vessel for organization
Vessels	*createVessel(vesselDetails)*	Status
Vessels	*readVessel(vesselId)*	Vessel details
Vessels	*updateVessel(vesselDetails)*	Status
Vessels	*deleteVessel(vesselId)*	Status
Vessels	*createVesselCertificate(vesselId)*	certificate and public and private keys
Vessels	*revokeVesselCertificate(vesselId, CertificateId, revokeReason)*	Status
Users	*getUsers(organizationId)*	list of user for organization
Users	*createUser(userDetails)*	Status
Users	*readUser(userId)*	User details
Users	*updateUser(userDetails)*	Status
Users	*deleteUser(userId)*	Status
Users	*createUserCertificate(userId)*	certificate and public and private keys
Users	*revokeUserCertificate(userId, CertificateId, revokeReason)*	Status
Devices	*getDevices(organizationId)*	list of device for organization
Devices	*createDevice(deviceDetails)*	Status
Devices	*readDevice(deviceId)*	Device details
Devices	*updateDevice(deviceDetails)*	Status
Devices	*deleteDevice(deviceId)*	Status
Devices	*createDeviceCertificate(deviceId)*	certificate and public and private keys
Devices	*revokeDeviceCertificate(deviceId, CertificateId, revokeReason)*	Status
Certificates	*getRevokedCertificates()*	list of revoked certificates
Certificates	*verifyCertificate(certificateId)*	certificate status

4.3 Interactions

4.3.1 Authentification

One of the main functionalities of the Identity registry is the authentication of users. The following Figure 20 shows the interaction between a human user and the responsible parts within the Identity Registry. The figure also contains the required interaction to validate a security token, as it is provided as result of a login activity.

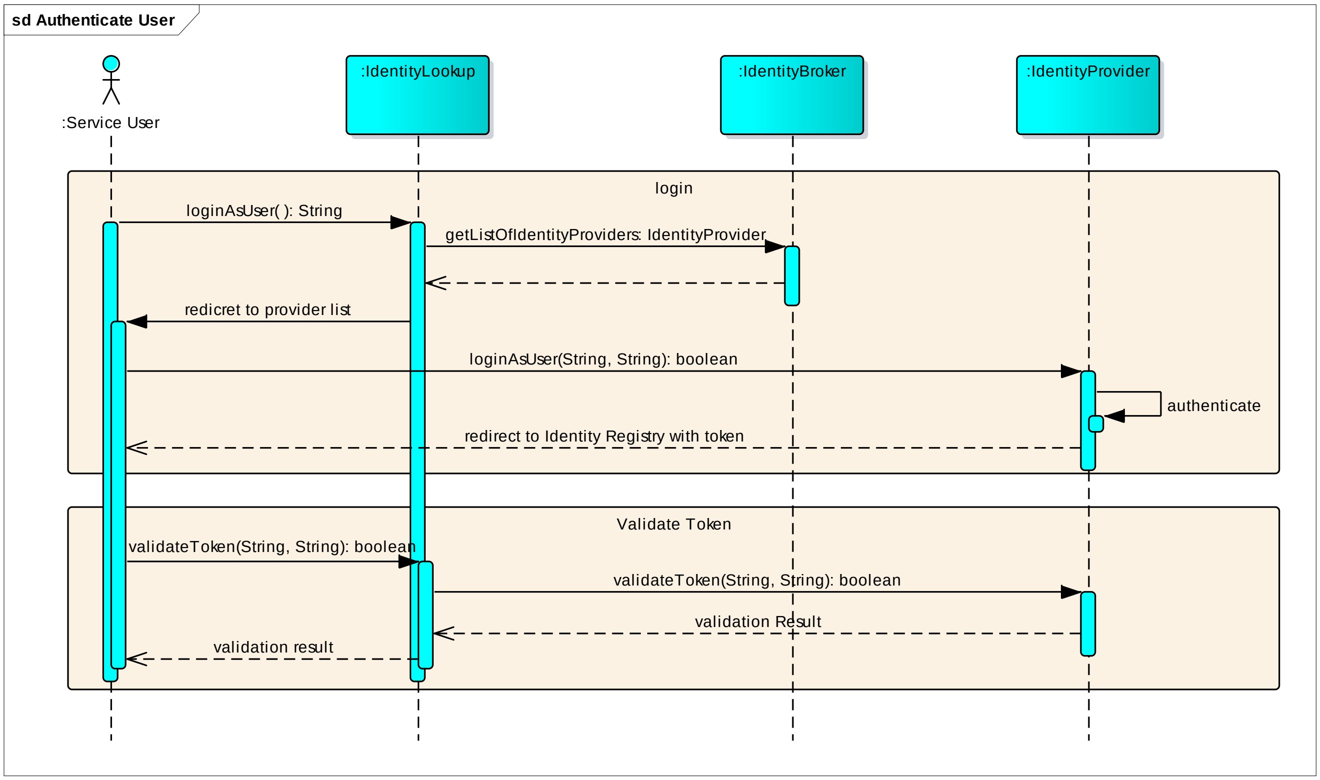

sd Authenticate User
:Service User
:IdentityLookup
:IdentityBroker
:IdentityProvider
login
loginAsUser( ): String
getListOfIdentityProviders: IdentityProvider
redicret to provider list
loginAsUser(String, String): boolean
authenticate
redirect to Identity Registry with token
Validate Token
validateToken(String, String): boolean
validateToken(String, String): boolean
validation Result
validation result

Figure 20 Sequence diagram for the user authentication and token validation

The Identity Registry uses a federated system approach, where the IdentityBroker is used as centralized mediator. The Broker knows all registered organizations and can redirect the login request to a specific IdentityProvider. The authentication is then performed, depending on the capabilities of the used identity provider. This centralized approach is intended to ease the usage of the Identity Registry, by providing only one entry point for Identity Management and authentication.

Besides the login with username and password (as shown in Figure 20), the Identity Broker also supports the login, using a certificate. The corresponding sequence diagram is shown in Figure 21. When using a certificate for login, the Identity Broker validates the provided certificate and returns an OpenID token. Thus an external service do not have to implement two validation methods; token based authentication and certificate authentication.

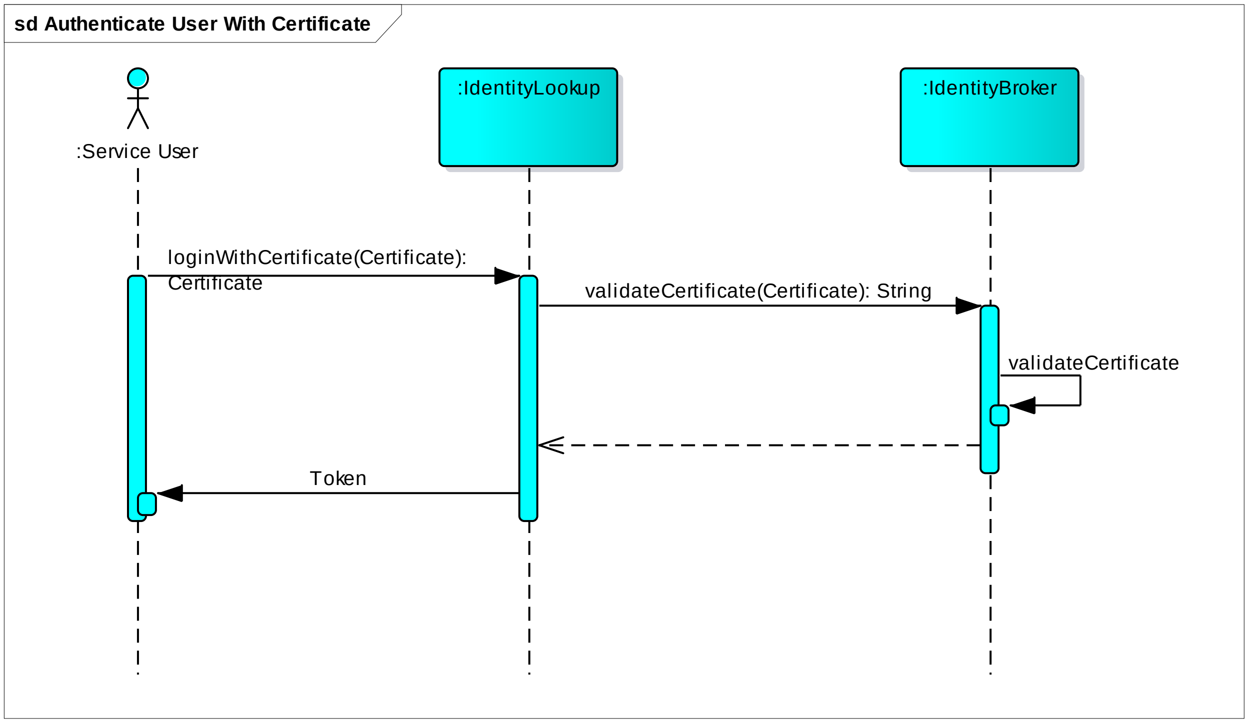

sd Authenticate User With Certificate
:Service User
:IdentityLookup
:IdentityBroker
loginWithCertificate(Certificate): Certificate
validateCertificate(Certificate): String
validateCertificate
Token

Figure 21 Sequence diagram for the user authentication using a certificate

4.3.2 Management

Within the Maritime Cloud Identity Registry, each maritime entity (Vessel, User, Device or Service) is associated to an organization that is able to verify the identity of the entity.
The creation of a new organization is currently done manually by a not yet fully defined authority (for the EfficienSea 2 testbed, the DMA takes over this part). The authority is responsible to check the trustworthiness of the requesting organization and register it within the Identity Registry. The corresponding sequence diagram is shown in Figure 22.

In Figure 22, the Identity Registry performs some additional steps, like checking the authorization of the current user, who has to be an Identity Registry administrator. If the authorization has been confirmed, the internal structures for the organization will be created. In addition an organization administrator (see Table 1) for the new organization is created. This organization administrator will be able to register additional maritime entities for the new organization, as described in the following section (section 4.3.3).

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

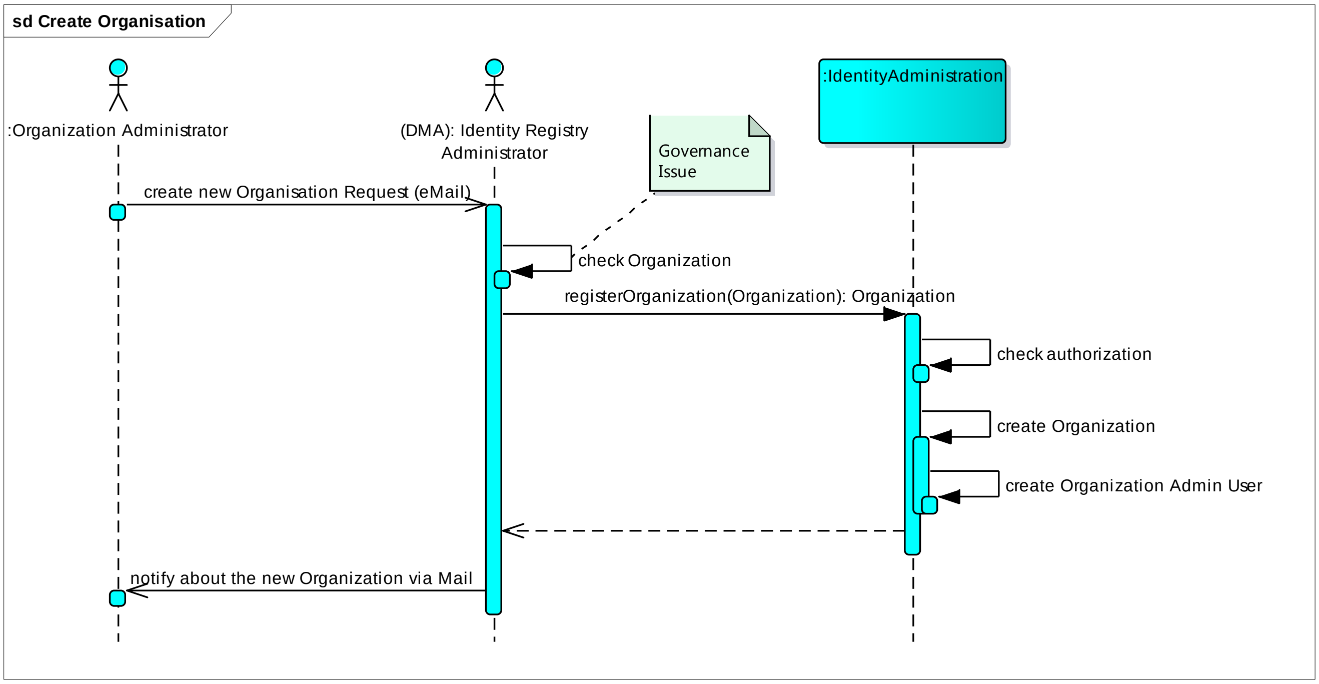

sd Create Organisation
:Organization Administrator
(DMA): Identity Registry Administrator
:IdentityAdministration
Governance Issue
create new Organisation Request (eMail)
check Organization
registerOrganization(Organization): Organization
check authorization
create Organization
create Organization Admin User
notify about the new Organization via Mail

Figure 22 Sequence diagram to create a new Organization within the Identity Registry

4.3.3 Create Maritime Entity

The interaction to create a new maritime entity (User, Service, Device or Vessel) within the Identity follows always the same procedure, therefore the following interaction diagram Figure 23 can be used for all entities.

This step becomes obsolete, if the organization uses its own identity provider to manage their users. In this case the procedures of the individual identity provider has to be used. However, other maritime entities like vessels, devices and services have to be registered inside the Identity Registry to create certificates (see Figure 24) and to use the broker's capability to evaluate certificates (see Figure 21).

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

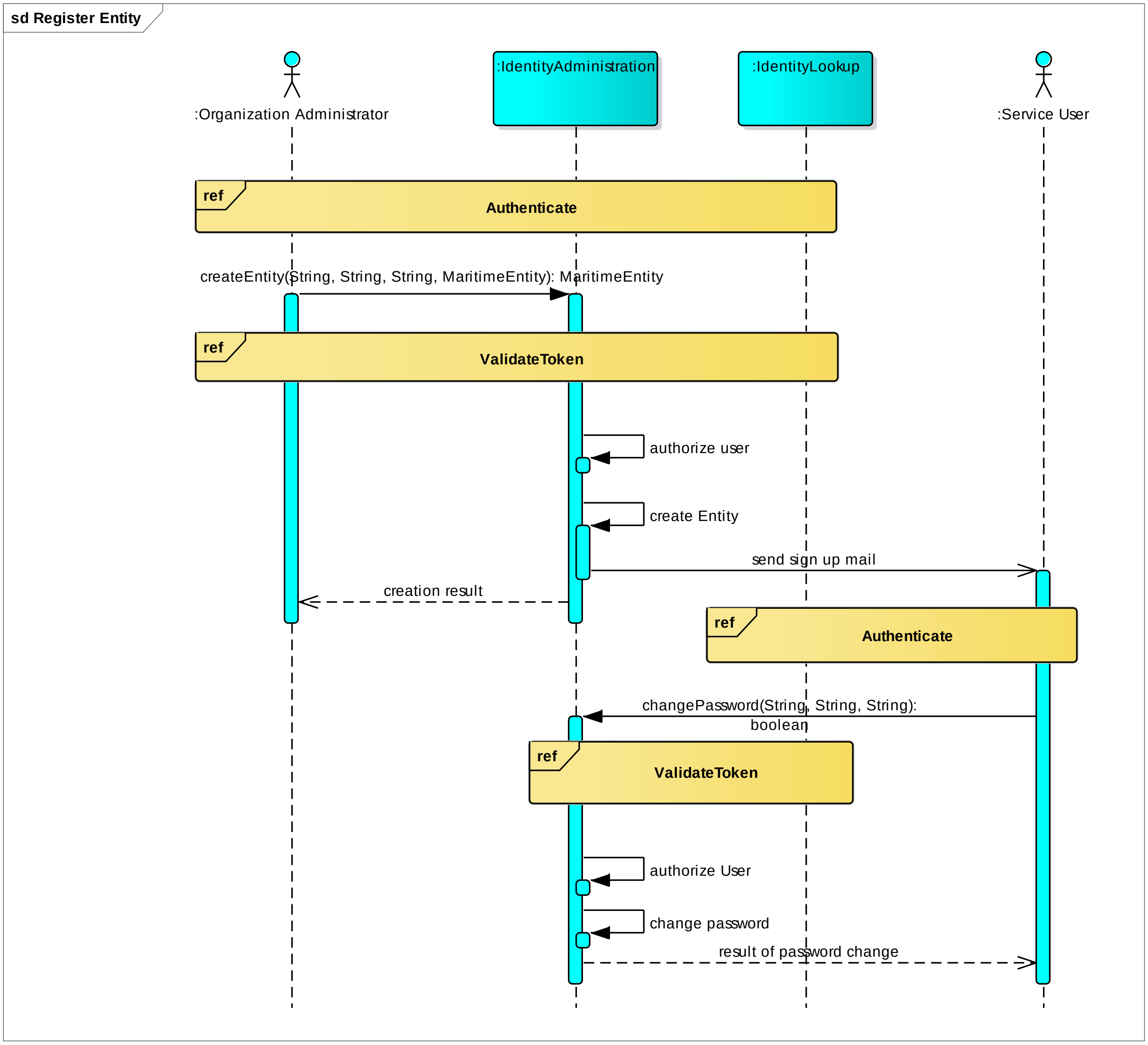


Figure 23 Sequence Diagram to create a new maritime entity

If a maritime entity has been created, the Identity Registry, allows to create certificates for this entity. The sequence diagram to create a new certificate is shown in Figure 24. This step requires the user, to have the role of an organization administrator, within the Identity Registry (see. Table 1). As indicated through the optional "Create Entity" reference, this step can be performed on any existing maritime entity.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

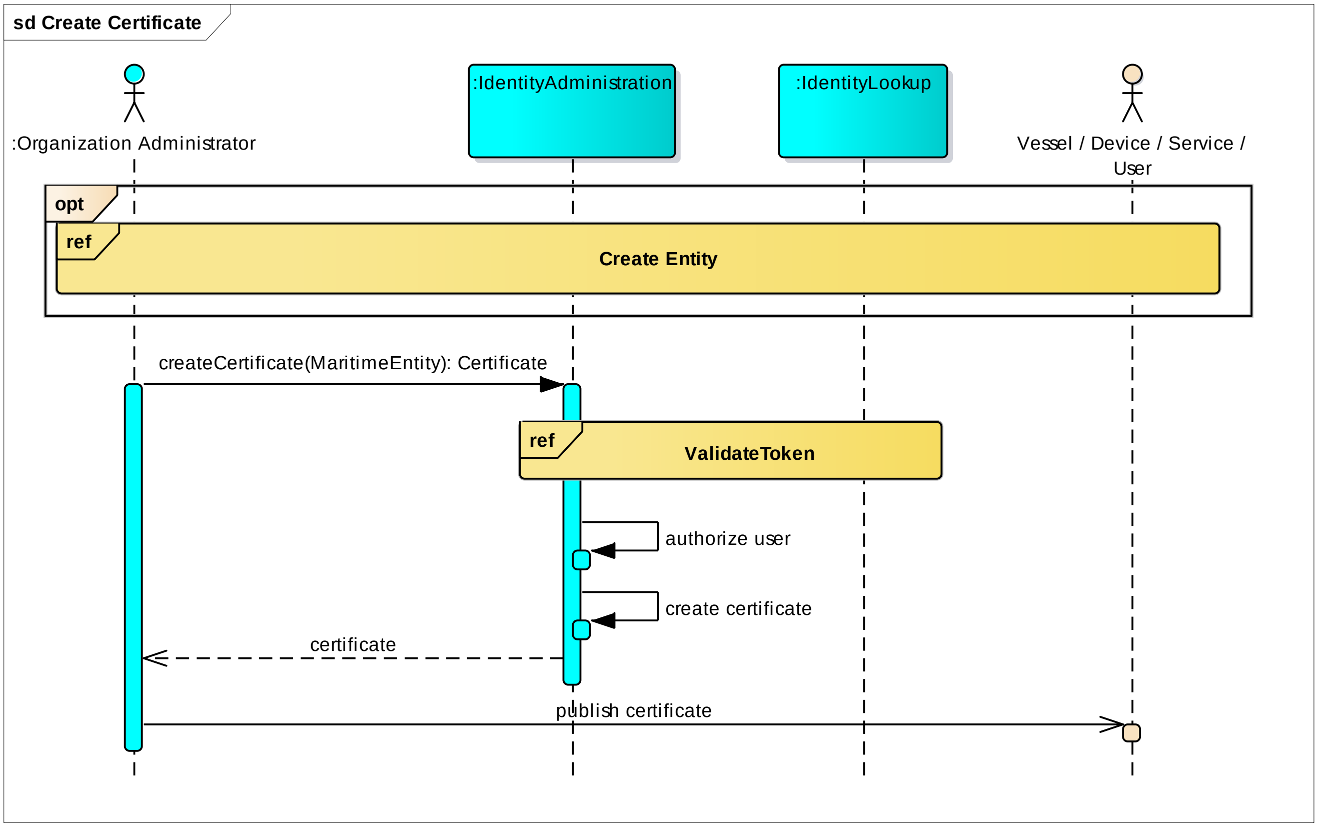


Figure 24 Sequence diagram to create a certificate

At this point, it’s worth to note, that the private keys, created together with the new certificate, can only be accessed while creating the certificate, for security reasons.
The last sequence in Figure 24 (publish certificate) is a manual task, that has to be done by the organization administrator. This could for example mean, to copy the certificate to an USB device and manually install it on a vessel.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

The following section describes the interactions to either discover service specifications, technical designs or service instance descriptions as well as the management of those.
The corresponding sequence diagrams uses an optional reference to the “Authenticate User” sequence diagram (see Figure 20). This optional step has only to be performed on the first lookup, until the service user has its OpenID token.

4.3.4 Service discovery

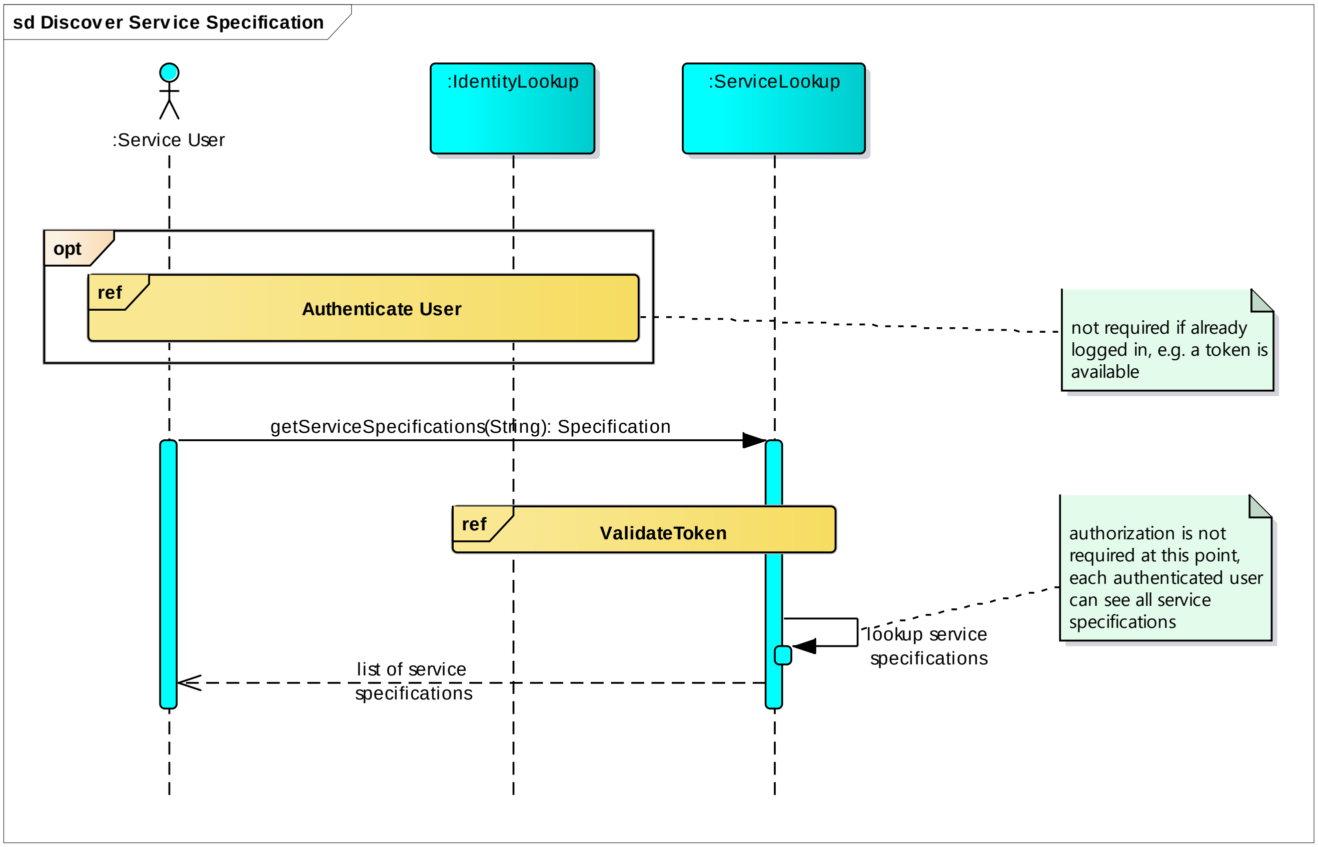

sd Discover Service Specification
:Service User
:IdentityLookup
:ServiceLookup
opt
ref
Authenticate User
not required if already logged in, e.g. a token is available
getServiceSpecifications(String): Specification
ref
ValidateToken
authorization is not required at this point, each authenticated user can see all service specifications
lookup service specifications
list of service specifications

Figure 25 interaction pattern to discover one or more service specifications

As described in Table 3 (SR API) the “getServiceSpecifications” method can take a search query for keywords, which have been provided, when the service specification has been published.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

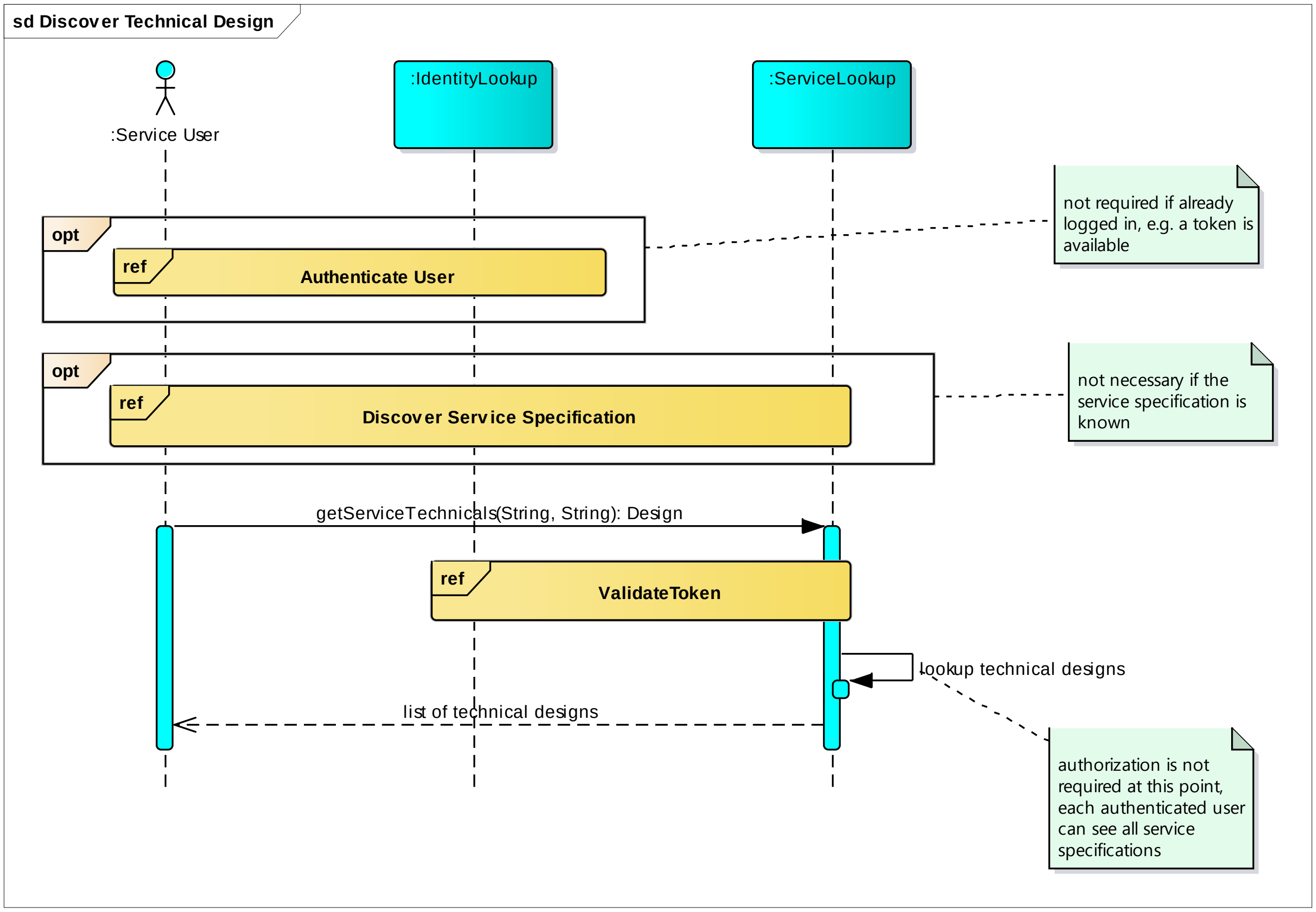

sd Discover Technical Design
:Service User
:IdentityLookup
:ServiceLookup
opt
ref
Authenticate User
not required if already logged in, e.g. a token is available
opt
ref
Discover Service Specification
not necessary if the service specification is known
getServiceTechnicals(String, String): Design
ref
ValidateToken
lookup technical designs
list of technical designs
authorization is not required at this point, each authenticated user can see all service specifications

Figure 26 Interaction pattern to get one or more technical designs for a service specification

To get a technical design, the realized service specification id has to be known by the service user. If the specification id is already known, the optional “Discover Service Specification” step in Figure 26 may be omitted.
The second argument in the “getServiceTechnicals” represents the version of the Service Specification. If the second argument is left empty, the latest version is returned by the Service Registry.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

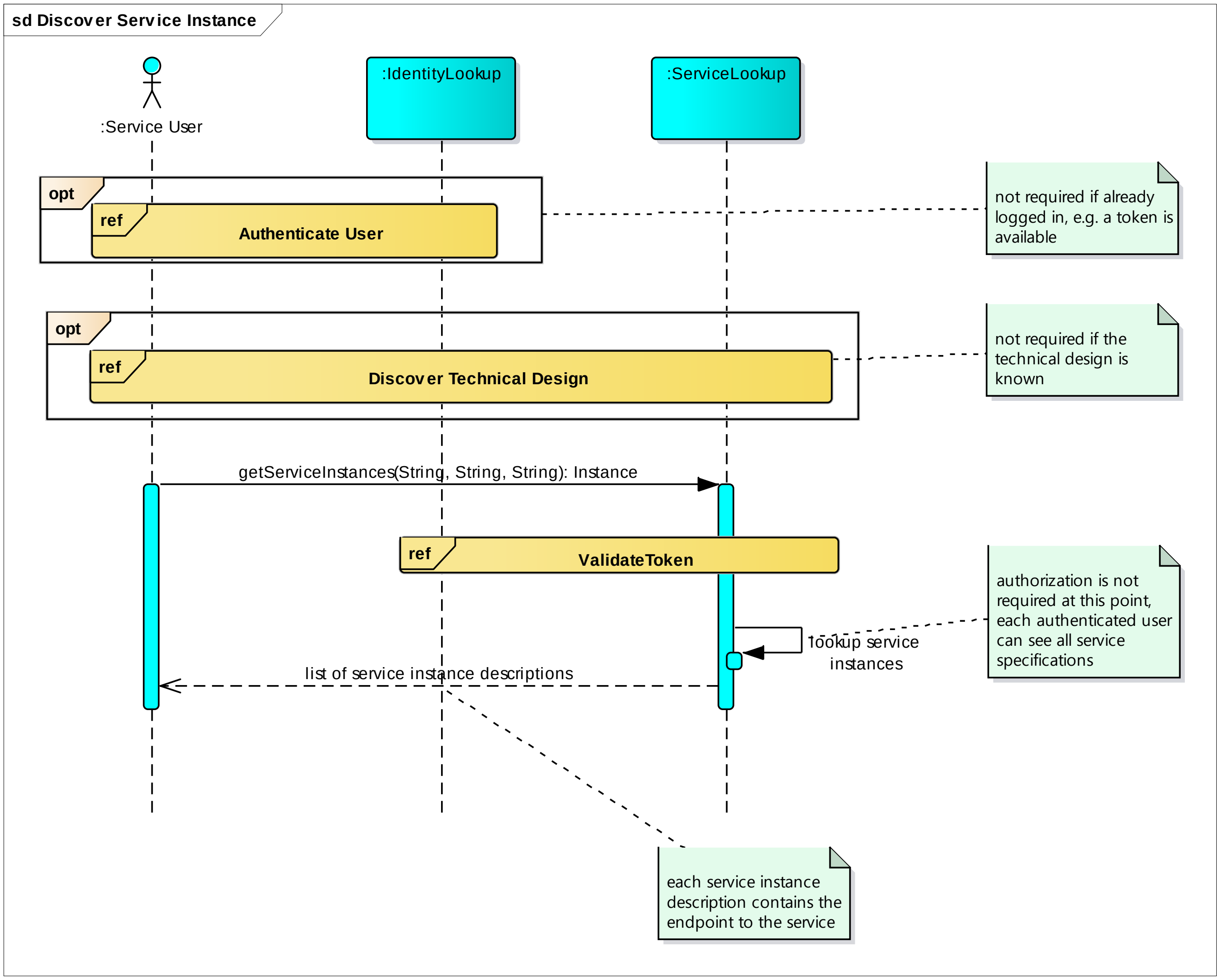

sd Discover Service Instance
:Service User
:IdentityLookup
:ServiceLookup
opt
ref
Authenticate User
not required if already logged in, e.g. a token is available
opt
ref
Discover Technical Design
not required if the technical design is known
getServiceInstances(String, String, String): Instance
ref
ValidateToken
lookup service instances
authorization is not required at this point, each authenticated user can see all service specifications
list of service instance descriptions
each service instance description contains the endpoint to the service

Figure 27 Interaction pattern to get one or more service instance descriptions for a technical design

Searching for a service instance description requires the service specification id and id of the technical design, implemented by that service. In addition an additional geographic location can be provided. The geometry has to be provided, using the Well Known Text (WKT) format, for example *Polygon((53.57 9.89, 53.57 10.06, 53.49 10.06, 53.49 9.89, 53.57 9.89))* if a list of service instances in the area of Hamburg (Germany) should be found.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

4.3.5 Service publishing

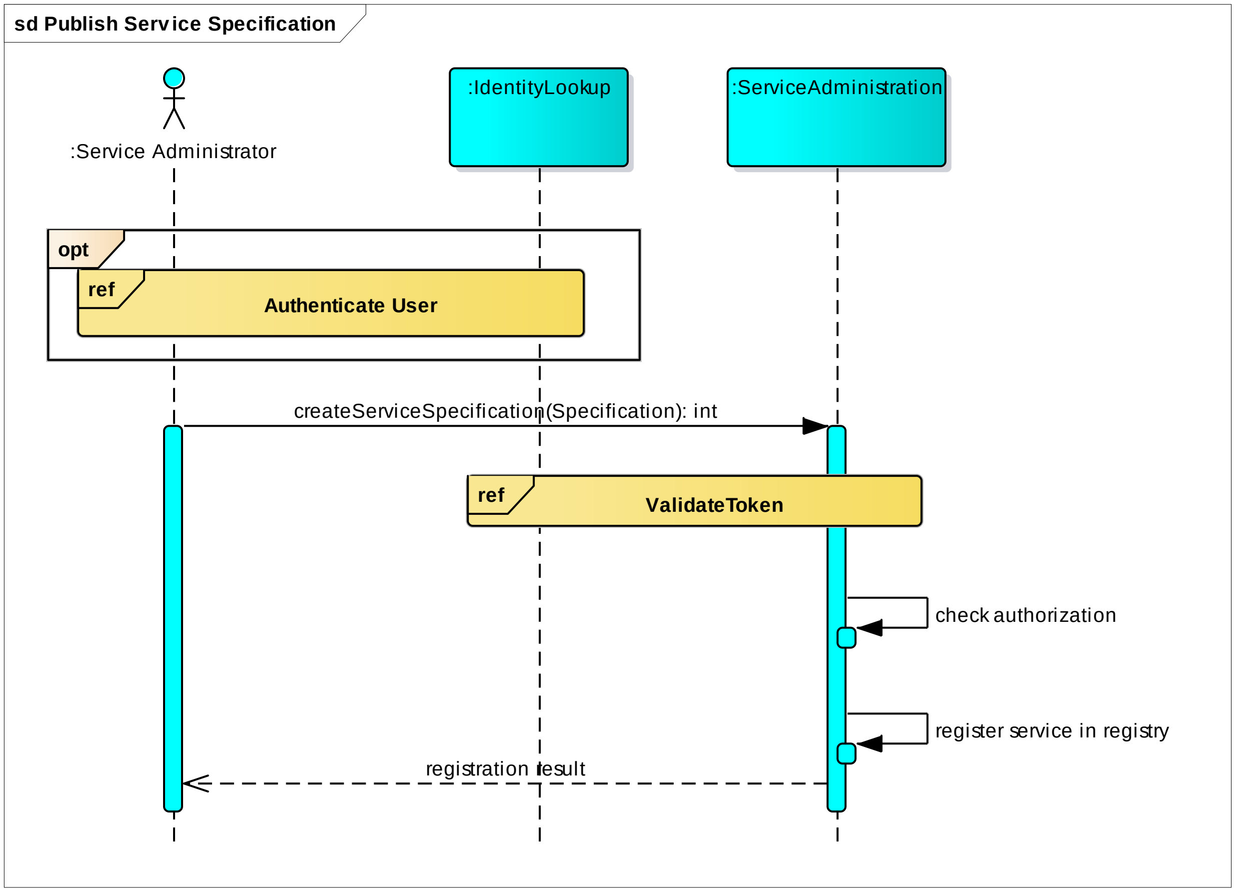


Figure 28 Required interactions to register a service specification within the service registry

The service specification that should be registered, contains different Meta information that should help to find the service, as described in Deliverable D3.6 (Service Specification Guidelines). This Meta information contains, among others, a list of keywords describing the service as well as the filled service specification template, as described in Deliverable D3.6.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

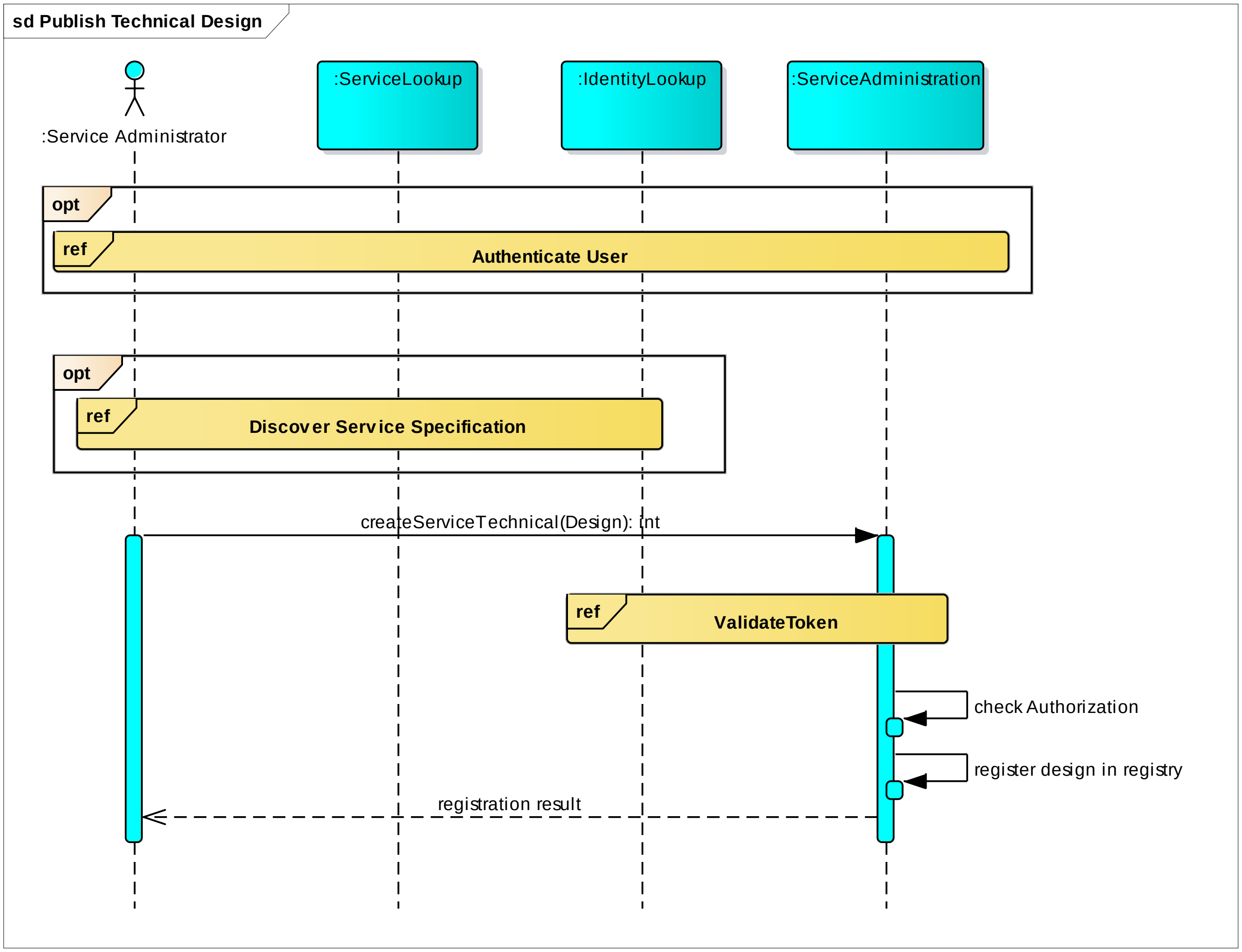


Figure 29 Sequence diagram to publish a technical design in the Service Registry

To register a technical design, implementing a service specification, the corresponding service specification has to be known. If the specification (incl. version of service specification) is not known, the specification can be received using the optional "Discover Service Specification" sequence in advance of uploading the design document.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

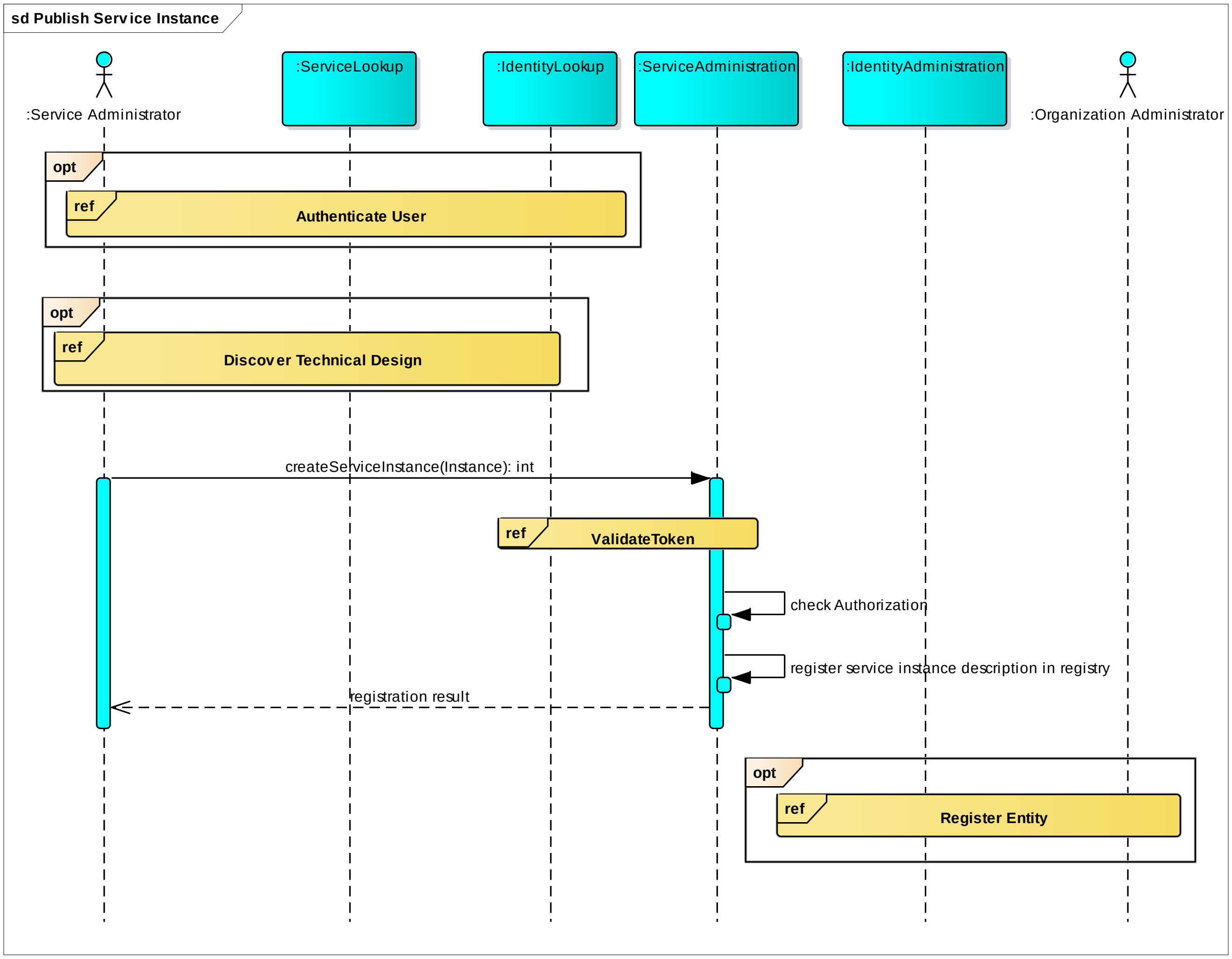


Figure 30 Sequence diagram to publish a service instance description

The sequence to register a service instance is similar to those to register a service specification and a technical design. However, if the service should be able to authenticate itself he can be registered in the Identity Registry as well. For this purpose the “Register Entity” Sequence has to be instantiated with a service as maritime entity.
To create a connection between the service within the Service Registry and the service within the Identity Registry, they have to be registered using the same Maritime Resource Name.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

4.4 Deployment

The following Figure 31 shows one possible deployment diagram for the described Maritime Cloud components.

The deployment can be separated into 3 different types of deployment environments.

1) Maritime Cloud Environment
   The Maritime Cloud environment contains the Service and Identity Registry, as central parts of the Maritime Cloud architecture. Both may be hosted on different servers, with direct internet connection.
2) Client Environment
   The client environment may be located onboard of a vessel or onshore, on a standard computer, hosting the client application (e.g. an ECDIS or any other application, using the Maritime Cloud).
3) Service Environment
   External Services as well as external identity provider shall be hosted in their own environments, depending on their needs, e.g. a virtual server for an instance of Keycloak identity provider.

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

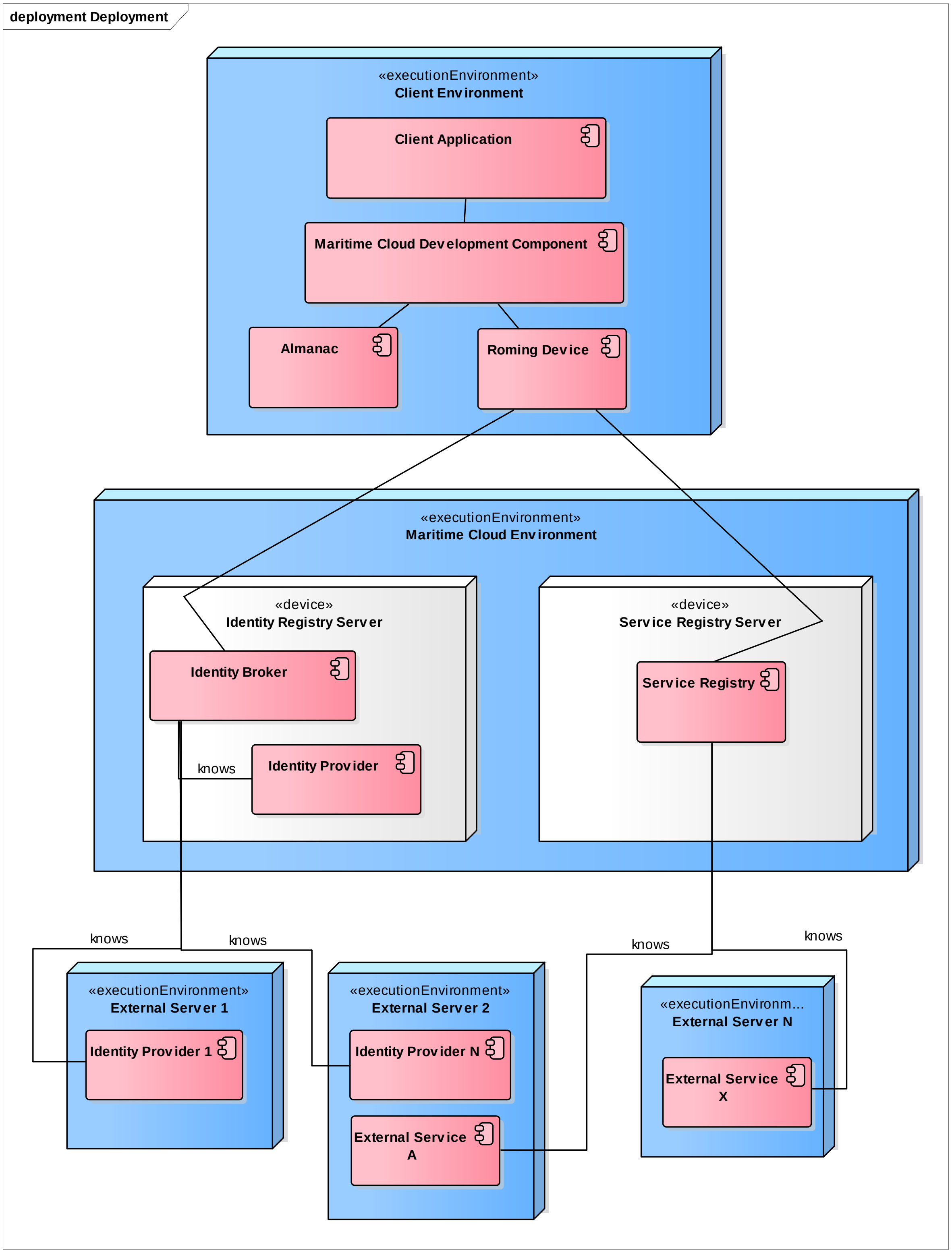


Figure 31 Deployment diagram for the described maritime cloud components

"This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 636329".

References

[1] C. Rihacek, T. Lutz, B. Weinert, A. . Bolles, K. . Nielsen and J. Jensen, "D3.2 Conceptual Model," 2016.

[2] B. Weinert, A. Hahn and O. Norkus, "A domain-specific architecture framework for the maritime domain," *Lecture Notes in Informatics,* 2016.

[3] K. Bronk, A. . Lipka, R. Niski, B. Wereszko and K. Wereszko, "D2.7 Concept and specification for seamless roaming," 2016.

[4] K. Bronk, A. Lipka and K. Werezko, "D2.8 Specification of the interface to Maritime Cloud," 2016.

[5] C. Rihacek, H. Künig and T. Lutz, "D3.6 Service Documentation Guidelines," 2016.

[6] ENAV Committee, "Maritime Resource Names," 2016.

[7] Maritime Cloud Developer Forum, "Identity Management and Cyber Security," 2016.

“This project has received funding from the *European Union’s Horizon 2020 research and innovation programme* under grant agreement No 636329”.

Appendix <enter number here>. Review procedure

No°	Reviewer Initials	Reference in document (General or Paragraph, Figure …)	Type (editorial, structural, formulation, error)	Reviewer's Comments, Question and Proposals	Editor’s action on review comment.