| | Document title | Document type |
|---|---|---|
| | **query** | **SD** |
| | Date | Version |
| | **2022-10-25** | **4.6.0** |
| | Author | Status |
| | **Tamás Bordi** | **RELEASE** |
| | Contact | Page |
| | **tbordi@aitia.ai** | **1 (11)** |

ARROWHEAD

# query

# Service Description

## Abstract

This document provides service description for the **query** service.

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**2 (11)**

# Contents

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**3 (11)**

# 1  Overview

This document describes the **query** service, which enables autonomous service look up, therefore it is an integral part of the implementation of service discovery requirements in Service Registry Mandatory Core System. Examples of this interaction is a system that has the capability to consume some kind of public core service. In order the system will be aware, where and how the service could be consumed, it needs to query the service details from Service Registry Core System.

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

| Document title | Version |
| --- | --- |
| **query** | **4.6.0** |
| Date | Status |
| **2022-10-25** | **RELEASE** |
| | Page |
| | **4 (11)** |

## 1.1   How This Service Is Meant to Be Used

The given system is required to being aware how to consume the **query** service from Service Registry Core System in order to look for the other public core services such as orchestration from the Orchestrator Core System. Figure 1 describes the processing of query data.
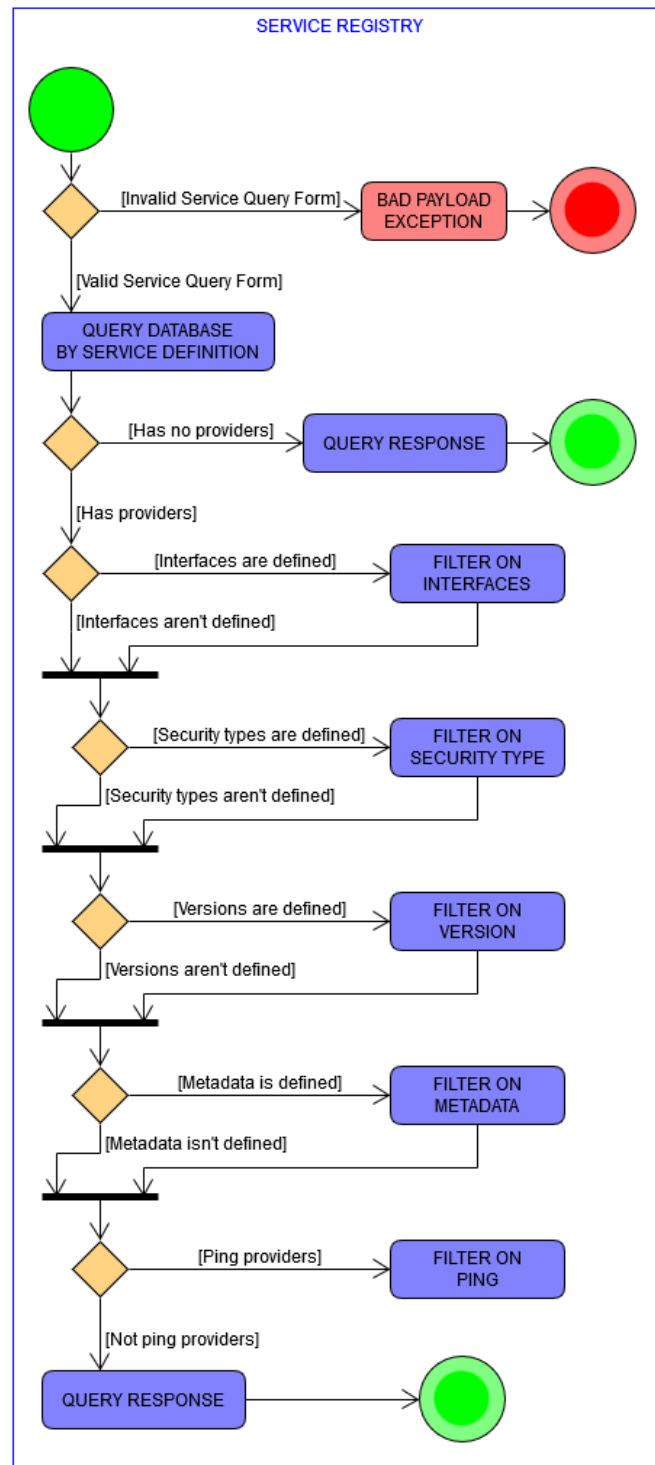


Figure 1: UML activity diagram of query process.

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**5 (11)**

## 1.2   Important Delimitations

The query data must meet the following criteria:

- Service definition can contain maximum 63 character of letters (english alphabet), numbers and dash (-), and has to start with a letter (also cannot ends with dash).

- Interface names have to follow the `Protocol-SecurityType-MimeType` format.

- Security types could be only `NOT_SECURE`, `CERTIFICATE` or `TOKEN`.

## 1.3   Access policy

Available for anyone within the local cloud, but in case of application systems, the system is allowed to query only for public core services. Other Core Systems are allowed to query only for their own services.

*Exception:* Translator Supporting Core Sytem is allowed to query for other services too.

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**6 (11)**

# 2  Service Interface

This section describes the interfaces to the service. The **query** service is used to looking for services. A service could contain various metadata as well as a physical endpoint. The various parameters are representing the necessary system and service input information. In particular, each subsection names an interface, an input type and an output type, in that order. The input type is named inside parentheses, while the output type is preceded by a colon. Input and output types are only denoted when accepted or returned, respectively, by the interface in question. All abstract data types named in this section are defined in Section 3.

The following interfaces are available.

## 2.1   interface HTTP/TLS/JSON (ServiceQueryForm) : ServiceQueryResult

| Profile ype | Type | Version |
|---|---|---|
| Transfer protocol | HTTP | 1.1 |
| Data encryption | TLS | 1.3 |
| Encoding | JSON | RFC 8259 [1] |
| Compression | N/A | - |

Table 1: HTTP/TLS/JSON communication details.

| | Document title | Version |
| --- | --- | --- |
| | **query** | **4.6.0** |
| | Date | Status |
| | **2022-10-25** | **RELEASE** |
| | | Page |
| | | **7 (11)** |

ARROWHEAD

# 3   Information Model

Here, all data objects that can be part of the **query** 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 3.3, which are used to represent things like hashes and identifiers.

## 3.1   struct ServiceQueryForm

| Field | Type | Mandatory | Description |
| --- | --- | --- | --- |
| interfaceRequirements | Array<Interface> | no | Names of the required interfaces. If specified at least one of the interfaces must match for having result(s). |
| maxVersionRequirement | Number | no | Required maximum version of the service. If specified version must be equals or lower or having result(s). Ignored if *versionRequirement* is specified. |
| metadataRequirements | Metadata | no | Service metadata requirements. If specified the whole content of the map must match for having result(s) |
| minVersionRequirement | Number | no | Required minimum version of the service. If specified version must be equals or higher or having result(s). Ignored if *versionRequirement* is specified. |
| pingProviders | Boolean | no | Whether or not the provider should be pinged. If *true* only the responding providers will comply. |
| securityRequirements | Array<SecureType> | no | Types of the required security levels. If specified at least one of the types must match for having result(s). |
| serviceDefinitionRequirement | Name | yes | Identifier of the service. |
| versionRequirement | Number | no | Required version of the service. If specified version must match for having result(s) |

### 3.1.1   struct Metadata

A JSON Object which maps String key-value pairs

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**8 (11)**

ARROWHEAD

## 3.2 struct ServiceQueryResult

| Field | Type | Description |
|---|---|---|
| serviceQueryData | Array<ServiceRegistryResponse> | List of service instance results |
| unfilteredHits | Number | Number of service instances not included by *serviceQueryData* due to the specified filters. |

### 3.2.2 struct ServiceRegistryResponse

| Field | Type | Description |
|---|---|---|
| createdAt | DateTime | Service instance record was created at this UTC timestamp. |
| endOfValidity | DateTime | Service is available until this UTC timestamp. |
| id | Number | Identifier of the service instance |
| interfaces | Array<Object> | List of interfaces the service supports. |
| provider | Object | Descriptor of the provider system record. |
| secure | SecureType | Type of security the service uses. |
| serviceDefinitionResponse | Object | Descriptor of the serviceDefinition record. |
| serviceUri | URI | URI of the service. |
| metadata | Metadata | Service metadata |
| updatedAt | DateTime | Service instance record was modified at this UTC timestamp. |
| version | Version | Version of the service. |

### 3.2.3 struct interfaces

| Field | Type | Description |
|---|---|---|
| createdAt | DateTime | Interface instance record was created at this UTC timestamp. |
| id | Number | Identifier of the interface instance |
| interfaceName | Interface | Specified name of the interface. |
| updatedAt | DateTime | Interface instance record was modified at this UTC timestamp. |

| Document title | Version |
| --- | --- |
| **query** | **4.6.0** |
| Date | Status |
| **2022-10-25** | **RELEASE** |
| | Page |
| | **9 (11)** |

ARROWHEAD

### 3.2.4 struct provider

| Field | Type | Description |
| --- | --- | --- |
| address | String | Network address. |
| authenticationInfo | String | Public key of the client certificate. |
| createdAt | DateTime | System instance record was created at this UTC timestamp. |
| id | Number | Identifier of the system instance |
| metadata | Metadata | Metadata |
| port | PortNumber | Port of the system. |
| systemName | Name | Name of the system. |
| updatedAt | DateTime | System instance record was modified at this UTC timestamp. |

### 3.2.5 struct serviceDefinition

| Field | Type | Description |
| --- | --- | --- |
| createdAt | DateTime | Service definition instance record was created at this UTC timestamp. |
| id | Number | Identifier of the service definition instance |
| serviceDefinition | Name | Name of the service definition. |
| updatedAt | DateTime | Service definition instance record was modified at this UTC timestamp. |

## 3.3 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.

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**10 (11)**

| Type | Description |
|------|-------------|
| Address | A string representation of the address |
| Boolean | One out of `true` or `false`. |
| DateTime | Pinpoints a specific moment in time. |
| Object | Set of primitives and possible further objects. |
| Interface | Any suitable type chosen by the implementor of service |
| List<A> | An *array* of a known number of items, each having type A. |
| Name | A string identifier that is intended to be both human and machine-readable. |
| Number | Decimal number |
| SecureType | Any suitable type chosen by the implementor of service |
| Version | Specifies a service version. |

# 4 References

[1]  T. Bray, "The JavaScript Object Notation (JSON) Data Interchange Format," RFC 8259, Dec. 2017. [Online]. Available: https://rfc-editor.org/rfc/rfc8259.txt

Document title
**query**
Date
**2022-10-25**

Version
**4.6.0**
Status
**RELEASE**
Page
**11 (11)**

# 5   Revision History

## 5.1   Amendments

| No. | Date | Version | Subject of Amendments | Author |
|-----|------|---------|----------------------|--------|
| 1 | YYYY-MM-DD | 4.6.0 | | Xxx Yyy |

## 5.2   Quality Assurance

| No. | Date | Version | Approved by |
|-----|------|---------|-------------|
| 1 | YYYY-MM-DD | 4.6.0 | |