# orchestration-service-by-id

## Service Description

**Abstract**
This document provides service description for the **orchestration-service-by-id** service.

# Contents

# 1 Overview

This document describes the **orchestration-service-by-id** service that provides runtime (late) binding between application systems. It is a special version of **orchestration-service**. Its primary purpose is to provide application systems with orchestration information: where they need to connect to. The outcome of the service includes data that will tell the application system what service provider system(s) it should connect to and how (acting as a service consumer). Such orchestration data include:

- Accessibility information details of a service provider (e.g network address and port),
- Details of the service instance within the provider system (e.g. base URL, IDD specification and other metadata),
- Authorization-related information (e.g. access token and signature),
- Additional information that is necessary for establishing connection.

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.

## 1.1 How This Service Is Meant to Be Used

The given system should consume the Service Registry Core System's **query** service to get information about the **orchestration-service-by-id** service. Using this information the system can request the **orchestration-service-by-id** service with its own database id. The service returns the top priority local provider of all services contained by the Orchestrator store database for the consumer system.

### 1.1.1 The orchestration process

Figure 1 describes the orchestration process.

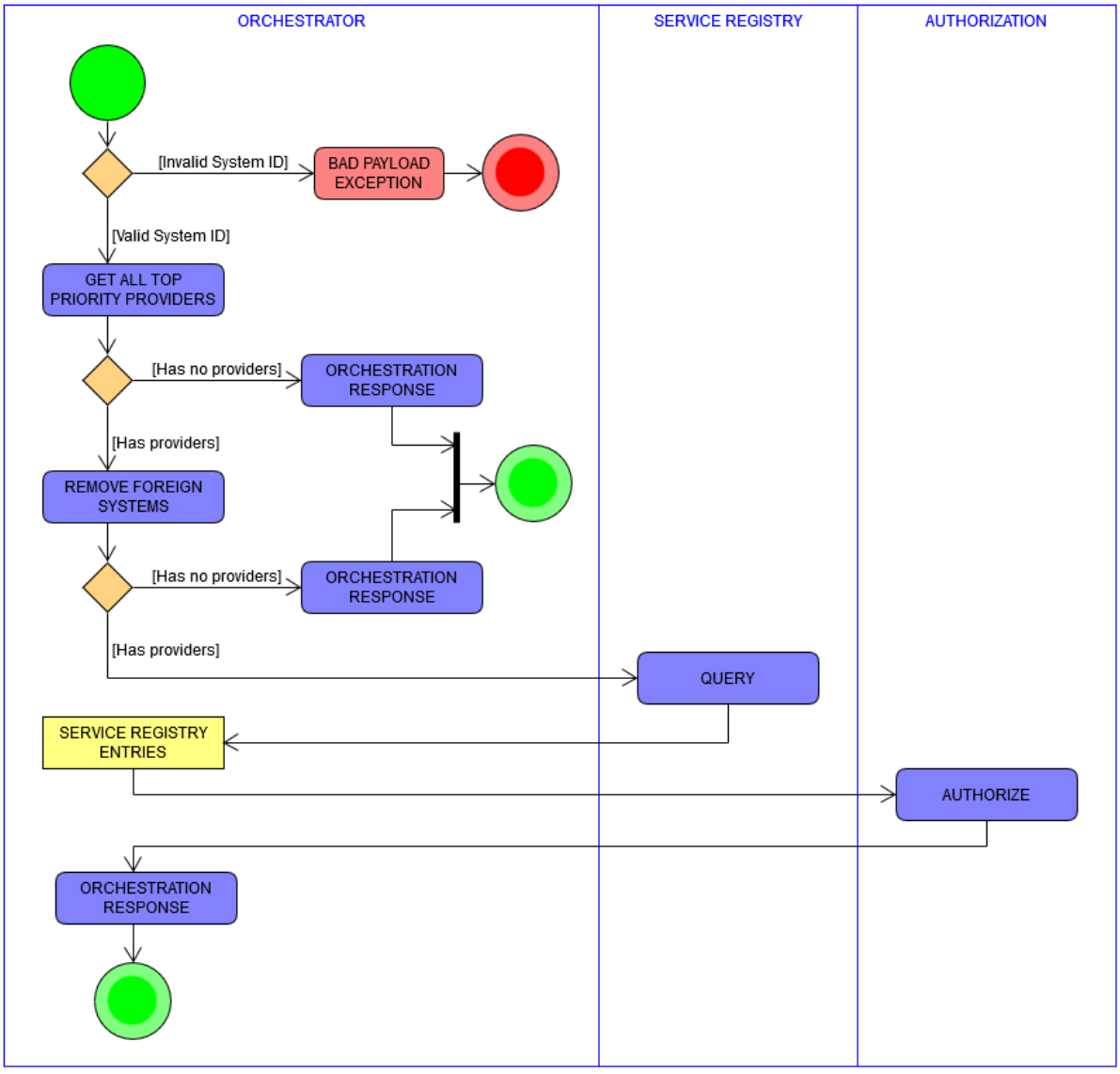


Figure 1: UML activity diagram of the orchestration process.

## 1.2 Important Delimitations

- This service is only supports the fix store orchestration.
- Inter-cloud rules are not supported.
- Orchestration flags are not supported.
- Quality-of-Service requirements are not supported.
- Provider reservation is not supported.

## 1.3 Access policy

Available for anyone within the local cloud.

# 2 Service Interface

This section describes the interfaces to the service. The **orchestration-service-by-id** service is used to looking for matching providers. 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 (Number) : OrchestrationResultList

| Profile type | 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.

# 3 Information Model

Here, all data objects that can be part of the **orchestration-service-by-id** 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 3.7, which are used to represent things like hashes and identifiers.

## 3.1 struct OrchestrationResultList

| Field | Type | Description |
|---|---|---|
| response | List<OrchestrationResult> | List of orchestration results. |

## 3.2 struct OrchestrationResult

| Field | Type | Description |
|---|---|---|
| authorizationTokens | Metadata | Tokens to use the service instance (one for every supported interface). Only filled if the security type is `TOKEN`. |
| interfaces | List<ServiceInterfaceRecord> | List of interfaces the service instance supports. |
| metadata | Metadata | Service instance metadata. |
| provider | SystemRecord | Descriptor of the provider system record. |
| secure | SecureType | Type of security the service instance uses. |
| service | ServiceDefinitionRecord | Descriptor of the service definition record. |
| serviceUri | String | Path of the service on the provider. |
| version | Version | Version of the service instance. |
| warnings | List<OrchestratorWarning> | List of warnings about the provider and/or its service instance. |

## 3.3 struct Metadata

An Object which maps String key-value pairs.

## 3.4 struct ServiceInterfaceRecord

| 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. |

## 3.5 struct SystemRecord

| Field | Type | Description |
|---|---|---|
| address | Address | Network address of the system. |
| authenticationInfo | String | X.509 public key of the system. |
| createdAt | DateTime | System instance record was created at this UTC timestamp. |
| id | Number | Identifier of the system instance. |
| metadata | Metadata | Additional information about the system. |
| port | PortNumber | Port of the system. |
| systemName | Name | Name of the system. |
| updatedAt | DateTime | System instance record was modified at this UTC timestamp. |

## 3.6 struct ServiceDefinitionRecord

| 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. |

## 3.7 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. |
| 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 |
| OrchestratorWarning | A potentially interesting information about a provider and/or its service instance. |
| PortNumber | A Number between 0 and 65535. |
| SecureType | Any suitable type chosen by the implementor of service. |
| String | A chain of characters. |
| 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

# 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 | |