

SysD SystemRegistry v1.2

25 september 2019

Contents

1	System Description Overview	2
2	Use-cases	3
3	System services	4
3.1	Produced Services	4
3.2	Consumed Services	4
4	Security	5
4.1	Security Objectives	5
4.2	Assets	5
4.3	Non-technical Security Requirements	5
5	Revision history	5
5.1	Amendments	5

Document title	Document type
SysD SystemRegistry - Black Box Design	v1.2
Date	Version
September 25, 2019	1.2
Author	Status
Silia Maksuti	Draft
Contact	Page
silia.maksuti@fh-burgenland.at	2(5)

1 System Description Overview

The SystemRegistry system is used to provide a local cloud storage holding the information on which systems are registered in a local cloud, meta-data of these registered systems and the services these systems are designed to consume. The SystemRegistry system holds for the local cloud unique system identities for systems deployed within the Arrowhead Framework local cloud.

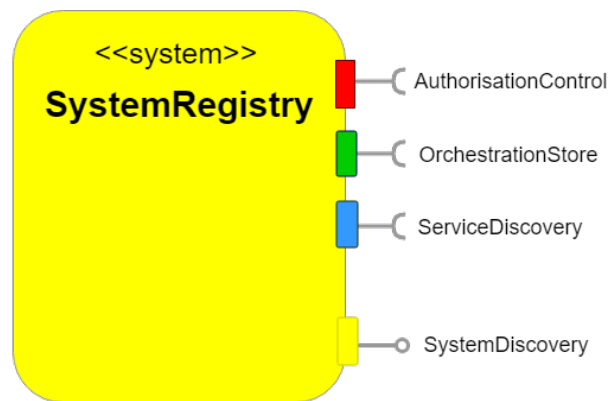


Figure 1: The SystemRegistry system, which produces the SystemDiscovery service and consumes the three mandatory core services

The SystemRegistry provides SystemDiscovery service for system registration, de-registration and lookup based on a Java-based REST interface, which uses a MySQL database. It consumes the three mandatory core services, AuthorisationControl, OrchestrationStore and ServiceDiscovery.

All Arrowhead Framework Systems within a local cloud shall register within the SystemRegistry by using the SystemDiscovery service. As such SystemDiscovery is regarded as a well known service, and shall be accessible using a multitude of SOA protocols like e.g. REST, CoAP, MQTT. The current implemented protocols for the SystemRegistry system are documented in the IDD document.

This registry in combination with the DeviceRegistry is necessary to create a chain of trust from a hardware device to a hosted software system and its associated services.

Document title	Document type
SysD SystemRegistry - Black Box Design	v1.2
Date	Version
September 25, 2019	1.2
Author	Status
Silia Maksuti	Draft
Contact	Page
silia.maksuti@fh-burgenland.at	3(5)

2 Use-cases

This section provides use cases that represent the actors and their interaction with SystemRegistry system.

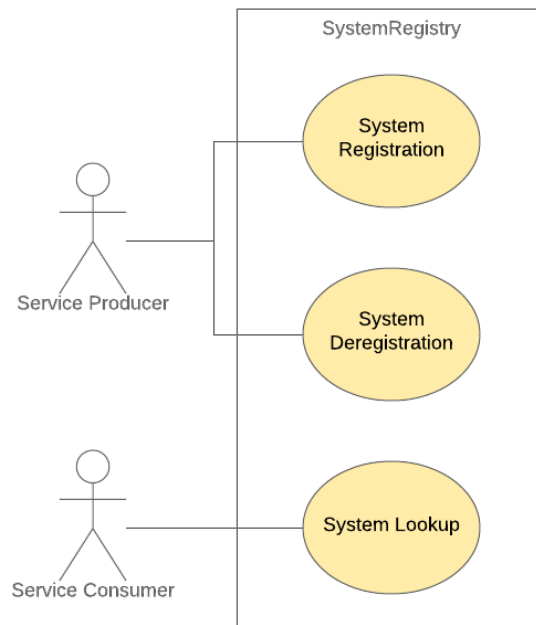


Figure 2: SystemRegistry use cases

System Lookup The use case System Lookup is used by the service consumer to get the list of registered systems in the SystemRegistry system, e.g. searches for a specific system instance with a specific ID (primary key).

System Registration The use case System Registration is used by the service producer to register its system in the SystemRegistry system, e.g. stores a system instance in the database (instances must not contain IDs, but embedded entities can have IDs if they already exist).

System De-registration The use case System De-registration is used by the service producer to delete its system from the SystemRegistry system, e.g. removes a system instance from the database.



Document title	Document type
SysD SystemRegistry - Black Box Design	v1.2
Date	Version
September 25, 2019	1.2
Author	Status
Silia Maksuti	Draft
Contact	Page
silia.maksuti@fh-burgenland.at	4(5)

3 System services

The SystemRegistry system produces one service, as shown in Table 1, and consumes three services, as shown in Table 2. All documents and code related to SystemRegistry system can be found in the repository of the Arrowhead Framework project, https://forge.soa4d.org/docman/?group_id=58.

3.1 Produced Services

The SystemDiscovery service is used to publish and unpublish systems, their produced services and their hosting devices, as well as to find systems among the registered systems in the SystemRegistry system.

Table 1: Produced services by the SystemRegistry system

Service	IDD Document Reference
SystemDiscovery	https://forge.soa4d.org/docman/view.php/58/177/Arrowhead_IDD_SystemDiscovery.pdf

3.2 Consumed Services

The SystemRegistry system consumes the three mandatory core services, briefly described below.

- The *AuthorisationControl* service provides the possibility of enabling fine grained access control to any resource/service for external requests; also provides customized information about the external consumer.
- The *OrchestrationStore* service provides functionality for storing and retrieving orchestration requirements, which is a set of rules for describing the ideal service required by a consuming system.
- The *ServiceDiscovery* service is used to register and unregister services, as well as find services among the registered serviced in the ServiceRegistry system.

Table 2: Consumed services by the SystemRegistry system

Service	IDD Document Reference
AuthorisationControl	https://forge.soa4d.org/docman/view.php/58/215/auth-40.zip
OrchestrationStore	https://forge.soa4d.org/docman/view.php/58/216/Orch-40.zip
ServiceDiscovery	https://forge.soa4d.org/docman/view.php/58/217/sr-40.zip



Document title	Document type
SysD SystemRegistry - Black Box Design	v1.2
Date	Version
September 25, 2019	1.2
Author	Status
Silia Maksuti	Draft
Contact	Page
silia.maksuti@fh-burgenland.at	5(5)

4 Security

4.1 Security Objectives

Registration of a system to the SystemRegistry of a local cloud shall allow for

- Authentication of systems requesting system registration
- Authentication of systems requesting system lookup
- Authentication of system requesting system de-registration
- Authorization of system registration
- Authorization of system lookup
- Authorization of system de-registration

4.2 Assets

No defined yet.

4.3 Non-technical Security Requirements

No defined yet.

5 Revision history

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2017-09-10	1.0	Initial Version	Silia Maksuti
2	2017-11-15	1.1	Updated Version	Silia Maksuti
3	2019-09-23	1.2	Updated Version	Silia Maksuti, Mario Zsilak