

Service Registry System System Description (SysD)

Abstract

The Service Registry System is a component that handles service registration and lookup in an Arrowhead local cloud.

Table of Contents

Table of Contents	2
1. Overview	3
1.1. Significant prior art	3
1.2. How this system is meant to be used	3
1.3. System functionalities and properties	4
1.4. Important delimitations	4
2. Services	4
2.1. Produced services	4
2.2. Consumed services	4
3. Security	4
3.1. Security Model	4
4. Revision history	5
4.1. Amendments	5
4.2. Quality Assurance	5

1. Overview

1.1. Significant prior art

This system description is derived from the Service Registry systems used in version 1, 2 3 and 4 of the Arrowhead Framework.

In the Swedish Network Based Defence project the information exchange platform, named OpenSIS, contained a Service Registry based on a Corba implementation. This Service Registry had the function of storing references to all producers that were available in the network. Every system that was integrated in the platform had an instance of the Service Registry executing. A master Service Registry was negotiated and every following Service Registry instance became a slave that would cache the SR content. All slave SR:s could be promoted to a master if the master SR would become unavailable. A service instance that were registered would have a lease time during the SR would store the information and the service was obliged to renew the lease when the time had expired.

UDDI has been proposed as a standard for Service Registration interface.
(https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=uddi-spec)

Multicast DNS (Apple Bonjour) has been used to handle Service Registry in local networks and the proposed standard DNS-SD (<http://www.dns-sd.org/> and described in RFC 6763). This standard also has been used in early versions of Arrowhead.

1.2. How this system is meant to be used

The Service Registry is used as the source of service presence, as all services that will be exposed in the Arrowhead network should be registered in this system. Any service that wants to connect to another service should be able to look other services up in this system.

The Service Registry is not depending on any other system to be operational and is considered a mandatory Core Arrowhead system.

A system that intends to register a producer service should call the Service Registration service and provide information about logical name, access endpoint and necessary metadata that the Service Registry requires. Additionally, if the SR uses lease of service registrations, a Time to Live for the service.

A system that intends to consume a service in the Arrowhead network should perform a lookup of the service/services in order to find the most suiting producer. This task can either be performed manually or via Orchestration.

The Service Registry system should implement a service for administration activities, such as manual registration of services and manual removal of stale services.

1.3. System functionalities and properties

The system should be configurable whether it should use “Time to Live”-settings for services or if the responsibility of cleaning up “dead” services relies on any system out of bounds of the Service Registry.

In order to find the Service Discovery service all systems will have to get this endpoint information out of bounds of the Service Registry, may it be options such as a DNS entry or a hardcoded property of each system.

1.4. Important delimitations

None defined.

2. Services

The Service Registry should produce services for service Discovery and Service Registry Administration.

2.1. Produced services

The Service Registry produces the following services:

Service name	SD
Service Discovery	SD Service Discovery
Service Registry Administration	SD Service Registry Administration

2.2. Consumed services

None

3. Security

3.1. Security Model

The service registry services can be controlled via Authorisation. Typically, all services should be allowed to register and lookup services.

4. Revision history

4.1. Amendments

No.	Date	Version	Subject of Amendments	Author
1	2022-04-29	0.1	First version for AH 5.0.	Per Olofsson
2				

4.2. Quality Assurance

No.	Date	Version	Approved by
1			
2			