System Description (SysD) – Arrowhead Zwave System Demonstrator

**Abstract**

This document provides the System Description of Zwave Compliant Arrowhead System. The system provides the services to control the zwave devices, such as Wall Plug and Thermostat using secure HTTP.

Table of Contents

[Table of Contents 2](#_Toc375649363)

[1. System Description Overview 2](#_Toc375649364)

[2. Diagrams 3](#_Toc375649366)

[3. Application services 3](#_Toc375649367)

[3.1. Produced Services 3](#_Toc375649368)

[3.2. Consumed Services 3](#_Toc375649369)

[4. Security 3](#_Toc375649370)

[5. References 3](#_Toc375649371)

[6. Revision history 3](#_Toc375649372)

[6.1. Amendments 3](#_Toc375649373)

[6.2. Quality Assurance 3](#_Toc375649374)

1. System Description Overview:

The Arrowhead Zwave System Demonstrator is a SOS with a Z-wave Controller, a Z-wave compliant Dasnfoss Thermostat Valve and a Z-wave compliant Fibaro Wall Plug which all behave as service providers in the arrowhead framework and the consumer connect to these devices and consumes their services.

* The Zwave controller provider system get the metadata and services available of the live zwave devices in the network. It performs first exclusion of all the devices then inclusion to remove all the dead devices and only get the metadata and services of live devices. Then it registers these devices and corresponding services into the service registry.
* The Consumer will first send the orchestration request for getting the z-wave device list service to get all the available devices connected to z-wave controller.
* After the list of devices with unique device ids and their types is received from zwave controller service provider, it will use the device ids and device types to request for different kind of services.
* The consumer request get-setpoint-thermo from Thermostat provider after getting the orchestration response for get-setpoint-thermo service to receive the current setpoint value from Thermostat Valve while the Thermostat provider also stores the value with the timestamp into its inherent DataManager.
* The consumer then requests get-set-point-history from Thermostat provider after getting the orchestration response for get-setpoint-history service to receive the records of setpoint value history stored in the Thermostat Provider inherent DataManager.
* Then the consumer requests switch-plug-state from Plug provider after getting the orchestration response for it to First Turn OFF the Switch then Turn it ON after 5 seconds.

Following is the Hardware used for this Project

* + - Raspberry Pi 4.0
    - Z-Wave hat
    - Fibaro Z-Wave electrical Outlet PLUG
    - Danfoss Z-Wave radiator valve Thermostat

Figure 1 Arrowhead and Zwave Setup



Service Registry

Orchestrator

Authenticator

**Zwave Consumer**

**Plug Provider**

**Zwave Controller Provider**

**Thermostat Provider**

1. Behaviour Diagrams

Timeline

Description automatically generated

Figure 2 Behavior Diagram showing the interaction of Zwave devices Provider systems with the Arrowhead core systems and Consumer System

1. Application services

This system provides four services:

# Produced Services

Table 1 Pointers to IDD documents

|  |  |
| --- | --- |
| Service | IDD Document Reference |
| live-zwave-devices | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_ZwaveController.docx |
| register-zwave-devices | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_ZwaveController.docx |
| get-state | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_WallPlugServices.docx |
| set-state | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_WallPlugServices.docx |
| get-power | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_WallPlugServices.docx |
| get-energy-consumption | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_WallPlugServices.docx |
| set-energy-counter-reset | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_WallPlugServices.docx |
| get-setpoint | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_ThermostatServices.docx |
| set-setpoint | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_ThermostatServices.docx |
| get-battery-level | IDD Interface Design Description\_Arrowhead\_Zwave\_System\_Demonstrator\_ThermostatServices.docx |

# Consumed Services

Table 2 Pointers to IDD documents

|  |  |
| --- | --- |
| Service | IDD Document Reference |
| Authorization Core Service | Arrowhead AuthorizationControl Service G4.0 IDD.docx |
| Orchestration Core Service | Arrowhead Orchestration Service G4.0 IDD.docx |
| Service Registry Core Service | Arrowhead ServiceDiscovery Service G4.0 IDD.docx |
|  |  |

1. Security

The system is using the HTTPS-SECURE-JSON security interface. Each system both consumer and providers are using their corresponding client certificate for secure communication. Authorization core system is responsible for the access verification and token generation during the provider and consumer interaction.

1. References

<https://github.com/arrowhead-f/core-java-spring>

<https://github.com/arrowhead-f/sos-examples-spring>

1. Revision history

# Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 2021-01-6 | 0.1 | First draft | Salman Javed |
|  | 2021-03-18 | 0.2 | Second Draft | Salman Javed |

# Quality Assurance

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Date | Version | Approved by |
| 1 |  |  |  |
| 2 |  |  |  |