Interface Design Description (IDD) – Monitorable

**Abstract**

This document defines the template for the Interface Design Description of Arrowhead compliant Interfaces.

It provides a detailed description of how the Monitorable service is implemented to fetch the status of the application system.

All Arrowhead Interface Designs should be specified using this template and stored on a common repository (available on the SVN server), in order to document and formalize the pilot demonstrators and the common Arrowhead framework.

Table of contents

[1. Interface Design Description Overview 2](#_Toc377455180)

[2. Interfaces 3](#_Toc377455181)

[3. Information Model 3](#_Toc377455182)

[4. References 3](#_Toc377455183)

[5. Revision history 4](#_Toc377455184)

[5.1. Amendments 4](#_Toc377455185)

[5.2. Quality Assurance 4](#_Toc377455186)

1. Interface Design Description Overview

This document describes the HTTP/{TLS}/JSON variant of the Monitorable service with REST interface. This allows for arbitrary Arrowhead Framework systems to check that the application system (fischertechnik Indexed line with two machining stations controlled with a Siemens S7-1500 PLC as asset via an OPC-UA server) is alive and responding to service requests.

1. Service Interfaces

This section describes the interfaces that must be exposed by Monitorable service. In particular, the below subsection first names the HTTP method and path used to call the interface. The interface is expected to respond with HTTP status code 200 OK for all successful calls.

1. **GET {baseURL}/services**

* **Interface: getAllServicesInfo**
* **Output: Sensor JSON Object**

Called to check all the available registered service information of the fischertechnik system in the service registry.

Example of valid invocation:

GET /services HTTP/1.1

Accept: String

Response: application/json

Example of valid response:

HTTP/1.1 200 OK

Content-Length:

Content-Type: application/json

[ {

"ServiceId": "150",

"SystemId": "17",

"ServiceDefinition": "SensorValue"

},

{

"ServiceId": "151",

"SystemId": "17",

"ServiceDefinition": "ActuatorValue"

},

]

1. **GET {baseURL}/services/{ServiceDefinition}**

* **Interface: getServiceInfo**
* **Output: Sensor JSON Objects**

Called to check the availibity of the particular registered service information of the fischertechnik system in the service registry.

Example of valid invocation:

GET /SensorValue HTTP/1.1

Accept: String

Response: application/json

Example of valid response:

HTTP/1.1 200 OK

Content-Length:

Content-Type: application/json

{

"ServiceId": "150",

"SystemId": "17",

"ServiceDefinition": "SensorValue"

}

1. **GET {baseURL}/echo**

* **Interface: Monitor**
* **Output: Any valid JSON object**

Called to check that the application system is alive and responding to service requests.

Example of valid invocation:

GET /echo HTTP/1.1

Accept: application/json

Example of valid response:

HTTP/1.1 200 OK

Content-Length:

Content-Type: application/json

{

“Got it”

}

1. Information Model

Here, all the data objects that can be part of Monitorable service calls are listed in alphabetic order.

**Service**

JSON object with the following fields.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Field** | **Type** | **Description** | **Mandatory** | **Default** |
| ServiceId | String | The Id of the individual service | True |  |
| SystemId | String | The Id of the application system the particular service is registered to. | True |  |
| ServiceDefinition | String | Definition/name of the service. | True |  |

1. Revision history

# Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 2020-04-15 | 0.1 | First Draft | Aparajita Tripathy |

# Quality Assurance

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