Interface Design Description (IDD) Template

**Abstract**

This document describes the IDD of the Windmill service’s.

This document describes the interfaces, communication profiles and other desirable information about using the Windmill system’s.

This document defines the template for the Interface Design Description, IDD, of Arrowhead compliant Interfaces. The document outlines the structure that should be followed when documenting the design of the interfaces exposed by a concrete realization of an Arrowhead Service Description (SD).

An IDD provides a detailed description of how the service is implemented/realized by including the communication profile and the chosen technologies.

An IDD is distinct from its corresponding SD in that it is a white-box rather than block-box description, outlining how each of the abstract interfaces of its SD are realized using a particular Communication Profile (CP). In other words, an IDD describes its interfaces in terms of a single transport protocol stack, encoding and, potentially, standardized semantics (?), while an SD describes its interfaces in abstract terms.

All Arrowhead Interface Designs should be specified using this template and stored on the common repository (github.com/arrowhead-f), in order to document and formalize the Arrowhead systems.

1. Interface Design Description Overview 3
2. Service Interfaces 3
   1. Interface 1 4
      1. Data Model 4
      2. Status and Error handling 5
      3. Interaction with consumers 5
   2. Interface 2 5
      1. Information Model 6
      2. Status and Error handling 6
3. Security 6
   1. Certificate 6
   2. Payload protection 7
   3. Audit of service exchange 7
4. References 7
5. Revision history 8
   1. Amendments 8
   2. Quality Assurance 8

## Interface Design Description Overview

This describes how to use the system’s wm-data services together with relevant information.

**Table 1 Pointers to SD documents**

|  |  |
| --- | --- |
| **Realised Service Description** | **Location** |
| wm-data | documentation/SD\_wm-data.docx |

This section shall outline the higher-level technical aspects of the interface design that must be known in order to have an Arrowhead system successfully consume it. Examples of such aspects include:

* Protocol
  + HTTPS
* Encoding
  + JSON
* Compression
  + None
* Base path: /wm-data
  + Example path: https://<IP>:<Port>/wm-data

## Service Interfaces

The Windmill system only provides on service **wm-data**. The wm-data service provides sensor data from a connected windmill. The provider utilizes a first in first out queue for the data sent by the windmill.

## Service 1: wm-data

Specifications for this interface:

* The data model is application/json.
* No ontologies are in use.
* No schemas are currently defined.
* No payload encryption is used.

**Table 4 INTERFACE description**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Interface name** | **Relative URL path** | **Method** | **Input** | **Output** |
| wm-data | /wm-data | GET | Param: next | One measurement of sensor data from the windmill. |

## Data Model

After a GET request with the correct parameter the below output is expected.

Output:

{”timeStamp”: <time>, ”speed” : <speed>, ”accelerometer” : <value1> ,…. , ”accelerometer” : <value16384> }

Example of input and output data shall be provided as examples with tables explaining the data parameters used. What fields are included in the tables will vary depending on the concrete encoding used by the interface design described the the IDD. The below tables assumes that JSON is the encoding in question.

**Table 5 Data Object description**

|  |  |  |  |
| --- | --- | --- | --- |
| **Object Field** | **Description** | **Format/limitations** | **Required** |
| timeStamp | Time of measurement | Double | Yes |
| speed | Speed at measurement | Double | Yes |
| accelerometer | Timeseries of Gs measured | List<Double> | Yes |

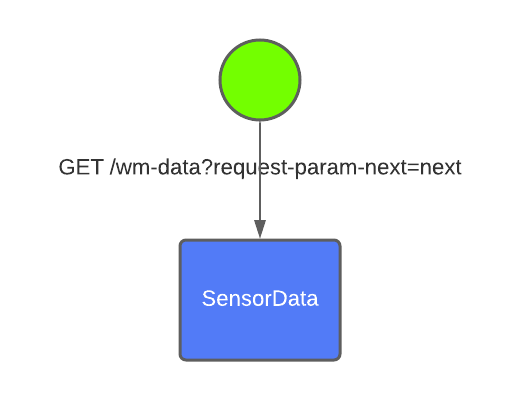
UML or SysML can be used to describe the relation of data format and specifications.

If any metadata is available, they shall be included in this section.

## Status and Error handling

|  |  |  |
| --- | --- | --- |
| **Code** | **Meaning** | **Comment** |
| 200 | Successful request | Succes |
| 401 | Unauthorized | Access Denied |
| 400 | Bad request | Wrong parameters |
| 500 | Internal server error | Error getting sensor data etc |

## Interaction with consumers



## Security

This system runs over TLS using server and client side X.509 certificates



## Certificate

The service uses applications certificates.

## Payload protection

No separate payload protection is used.

## Audit of service exchange

None

## References

## Revision history

## Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 20-2-2015 | 1.0 | Revision of text | Michele Albano / Luis Ferreira |
| 2 | 30-09-2015 | 1.1 | Refinement of the structure | Michele Albano / Luis Ferreira |
| 3 | 2020-06-10 | 2.0 | Major update | Jerker Delsing |
| 4 | 2020-06-15 | 2.1 | More verbose help text | Emanuel Palm |
| 4 | 2020-08-17 | 4.2 | Cleaning and minor update | Jerker Delsing |
| 5 | 2020-01-25 | 4.3 | Added Windmill information | Robin Jonsson |

## Quality Assurance

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