Arrowhead Historian Service IDD

**Abstract**

This document defines interface design of the Historian Service based on REST within Arrowhead generation 4.0.

1. Overview

This document describes the Historian service of G4.0 with the REST interface.

This interface can use HTTP, HTTPS, CoAP or CoAPS, therefore the related CP is valid.

1. Interfaces

As per the SD of this Service, there are five functions implemented. Table 1 describe these.

Table 1 Function description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Function | URL subpath | Method | Input | Output |
| ListServices | “/historian/{systemName}” | PUT | Operation | HTTP CODE:  CREATED, NO\_CONTENT  BAD\_REQUEST |
| CreateService | “/historian/{systemName}” | PUT | Operation | HTTP CODE: CREATED, NO\_CONTENT  BAD\_REQUEST |
| DeleteService | “/historian/{systemName}” | PUT | Operation | HTTP CODE: CREATED, NO\_CONTENT  BAD\_REQUEST |
| Store | “/historian/{systemName} /{serviceName}” | PUT | SensorData GenericFile | HTTP CODE: CREATED, NO\_CONTENT  BAD\_REQUEST |
| Retrieve | “/historian/{systemName} /{serviceName}” | GET | SensorData GenericFile | HTTP CODE: OK, NO\_CONTENT  BAD\_REQUEST |

1. Information Model

The SensorData request payload contains generic sensor data. SensorData normally contains information about the unit, source system, timestamp and metadata. The default payload type is JSON-encoded SenML (RFC 8428). The response to a Store/ Retrieve request is a simple HTTP/CoAP status code (Created/OK – request was success, No Content – request had no effect). The Content-type for Store must be *application/senml+json*. The Content-type for Retrieve is *application/json*.

# 

# 3.1. Operation request ListServices request

This function lists all service instances for a specific system.

{

"op": “list”

}

Response:

# 3.2. Operation request CreateService request

This function creates a new service instance for a specific system.

{

"op": “create”,

"srvName":"tempService1.\_tempSys-1.\_http.\_tcp.\_arrowhead.eu:8000”

}

Response:

# 3.3. Operation request DeleteService

This function deletes a service instance, and all of it’s stored data including files.

{

"op": “delete”,

"srvName":"tempService1.\_tempSys-1.\_http.\_tcp.\_arrowhead.eu:8000”

}

Response:

# 3.4. SensorData Store- SenML request

This function stores a SenML message in a specific service belonging to a system.

[

{"bn":" tempService1.\_tempSys-1.\_http.\_tcp.\_arrowhead.eu:8000",

"bt":1.276020076001e+09, "bu":"Cel","bver":5,

"n":" IndoorTemperature ","u":"Cel","v":22.1},

{"n":" IndoorTemperature ","t":-5,"v":22.2},

{"n":" IndoorTemperature ","t":-4,"v":22.4}

]

Response:

{  
 "x":" 0

}

The x tag (eXception) value of 0 indicates “No error”. A value different than 0 must also be accompanied by an xs tag that in plain text gives a reason for the error.

If the uploaded JSON is incorrectly constructed, then the following example response could be generated.

{  
 "x":" 2,

“xs”: “JSON Parse error”

}

# 3.5. Sensor data Retrieve

This function fetches data for a specific service belonging to a specific system. The output is SenML.

# 3.6. GenericFile

The GenericFile can be any type of file. The extension and mime type together defines the content.

# Revision history

# Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 2018-09-19 | G4.0 | Initial | Jens Eliasson |
| 2 | 2018-10-30 | G4.0 | Text update | Jens Eliasson |
| 3 | 2019-03-15 | G4.0 | Minor text improvements | Jens Eliasson |
| 4 | 2019-03-20 | G4.0 | Switched to RFC8428 | Jens Eliasson |
| 5 | 2019-07-02 | G4.0 | Updated path with systemName/serviceName | Jens Eliasson |
|  |  |  |  |  |

# Quality Assurance

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