

Proxy

Service Description

Service ID: *"proxy"*

Abstract

This document describes an abstract service that provides functions and features to allow data producing systems to temporary store/cache data.



ARTEMIS Innovation Pilot Project: Arrowhead
THEME [SP1-JTI-ARTEMIS-2012-AIPP4 SP1-JTI-ARTEMIS-2012-AIPP6]
[Production and Energy System Automation Intelligent-Built environment and urban infrastructure for sustainable and friendly cities]



ARROWHEAD

Document title
Proxy
Date
2020-12-09

Version
0.2
Status
DRAFT
Page
2 (9)

Contents

1	Overview	3
1.1	Introduction	4
1.2	Status of this Document	4
2	Service Interface	5
2.1	function ListSystems	5
2.2	function ListSystemService s	5
2.3	function Push	5
2.4	function Fetch	5
3	Information Model	6
3.1	struct SensorData	6
3.2	Primitives	7
4	References	8
5	Revision History	9
5.1	Amendments	9
5.2	Quality Assurance	9



ARROWHEAD

Document title
Proxy
Date
2020-12-09

Version
0.2
Status
DRAFT
Page
3 (9)

1 Overview

This document describes an abstract Eclipse Arrowhead service that was designed to allow resource-constrained devices, often battery-powered sleepy sensors, to temporarily store data so that consumers can access the data even if the producing device is sleeping and hence offline. The Proxy service can store the last message from a producer and make it available for consumers. No database is needed.

The rest of this document is organized as follows. In the remainder of this section we consider significant prior art, describe how this service is meant to be used and comment on the status of this document. In Section 2, we describe the abstract interface, in terms of functions invoked by messages, provided by this service. Finally, in Section 3, we present the data types used by those functions.



ARROWHEAD

Document title
Proxy
Date
2020-12-09

Version
0.2
Status
DRAFT
Page
4 (9)

1.1 Introduction

This Arrowhead Core service proposal is a vital part for enabling message and data exchange in an Arrowhead local cloud. The Proxy service provides a simple mechanism for (resource-constrained) data producing systems to make their data available for consumer systems at all times. This is particularly important for battery powered devices that can spend extended time in sleep and thus being offline. The Proxy service allows those type of devices to periodically wake up and update their endpoint in the Proxy service. The data model is currently chosen to be SenML [1].

1.2 Status of this Document

This document represents the current version of the Proxy service. Eclipse Arrowhead, being part of an academic and R&D community is constantly evolving to provide more features and increased performance and stability. More features can therefore be added to the Proxy service in the future. This document will be updated to incorporate any new features or changes.



2 Service Interface

This section lists the *functions* that must be exposed by a Proxy service in alphabetical order. Each function represents one feature the Proxy service can *perform* a task, e.g. list systems or services, or store and fetch data. In particular, each following subsection names an abstract function, an input type and an output type, in that order. The input type is named inside parentheses, while the output type is preceded by a colon. Input and output types are only denoted when accepted or returned, respectively, by the function in question.

All abstract data types named in this section are defined in Section 3.

2.1 function **ListSystems** () : **SystemList**

This function lists all systems that have inserted service data. The output is an array of system names.

2.2 function **ListSystemServices** () : **ServiceList**

This function lists all services for a system that have inserted service data. The output is an array of service names.

2.3 function **Push** (**SensorData**)

This function is used to push a message to the Proxy Service in a Local Cloud through the DataManager Core System.

2.4 function **Fetch** () : **SensorData**

This interface is used to fetch a message from the Proxy Service in a Local Cloud through the DataManager Core System.



3 Information Model

Here, the main data object model that is used by the Proxy service is listed.

3.1 struct **SensorData**

Below is a summary of the most used data entries in an SenML document.

Field	Type	Description
bn	Base name	Base name of a message.
bt	Base time	Base time.
bu	Base unit	Base unit.
bver	Base version	Base version (not used).
n	Name	Name of an data tag.
t	Time	Time a measurement was taken
u	Unit	Unit of a measurement.



ARROWHEAD

Document title
Proxy
Date
2020-12-09

Version
0.2
Status
DRAFT
Page
7 (9)

3.2 Primitives

Please refer to the SenML RFC [1] for more information about the supported data models.

4 References

- [1] C. Jennings and Z. Shelby, "Sensor Measurement Lists (SenML)," RFC 8428, 2018, RFC Editor. [Online]. Available: <https://doi.org/10.17487/RFC8428>



ARROWHEAD

Document title
Proxy
Date
2020-12-09

Version
0.2
Status
DRAFT
Page
9 (9)

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2018-09-17	G4.0 d1	Initial	Jens Eliasson
2	2018-10-30	G4.0 d2	Text updates	Jens Eliasson
3	2019-03-20	G4.0 d3	Updated data models	Jens Eliasson
4	2020-11-17	G4.1.3	Added more data models	Jens Eliasson
5	2020-11-30	G4.1.2	New template	Jens Eliasson

5.2 Quality Assurance

No.	Date	Version	Approved by
1			