

Configuration Service Description

Service ID: *"config"*

Abstract

This document describes an abstract service that provides functions and features to manage configuration files for Arrowhead systems.



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]

Contents

1	Overview	3
1.1	Introduction	4
1.2	Status of this Document	4
2	Service Interface	5
2.1	function getConfig	5
2.2	function getRawConfig	5
2.3	function GetList	5
2.4	function storeConfiguration	5
2.5	function deleteConfiguration	5
3	Information Model	6
3.1	struct ConfigurationRequest	6
3.2	struct ConfigurationResponse	6
3.3	struct ConfigurationListResponse	6
3.4	Primitives	7
4	References	8
5	Revision History	9
5.1	Amendments	9
5.2	Quality Assurance	9



ARROWHEAD

Document title
Configuration
Date
2021-02-05

Version
0.9
Status
DRAFT
Page
3 (9)

1 Overview

This document describes an abstract Eclipse Arrowhead service that was designed to manage configuration and settings for Eclipse Arrowhead systems. The Config service enables an Arrowhead system to fetch its configuration file from a central repository, thus simplifying management of large scale installations.

A configuration can be either black- or white boxed. A black box configuration can be binary or on any format. A white box configuration can be in e.g. JSON that the Config service can understand and validate.

The rest of this document is organized as follows. In the remainder of this section we consider significant prior work, 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
Configuration
Date
2021-02-05

Version
0.9
Status
DRAFT
Page
4 (9)

1.1 Introduction

This Arrowhead Core service [1] proposal is a vital part for enabling message and data exchange in an Arrowhead local cloud. The Config service provides features to manage configurations and settings in a centralized manner.

1.2 Status of this Document

This document represents the current version of the Config 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 Translator service in the future. This document will therefore be updated to incorporate any new features or changes when needed.

2 Service Interface

This section lists the *functions* that must be exposed by a Config service. Each function represents one feature the Config service can *perform*, e.g. list configurations, or store new ones 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 **getConfig** (**systemName**) : **ConfigurationResponse**

Fetches a configuration object.

2.2 function **getRawConfig** (**systemName**) : **Configuration file**

Fetches a black box configuration file of any type.

2.3 function **GetList** () : **ConfigurationListResponse**

Get a List of all configurations.

2.4 function **storeConfiguration** (**systemName**, **ConfigurationRequest**) : **ConfigurationResponse**

Stores a new configuration for an Arrowhead system.

2.5 function **deleteConfiguration** (**systemName**) : **ConfigurationResponse**

Deletes a configuration for an Arrowhead system and returns the deleted object.

3 Information Model

Here, the main data object models that are used by the Config service are listed.

3.1 struct **ConfigurationRequest**

Below is a description of a request to store a configuration for a local cloud system from the administrator.

Field	Type	Description
systemName	Name	System name
filename	String	Name of the uploaded config file
contentType	Content-type	Type of data (text/plain, application/json, etc.)
data	Base64	Base64-encoded configuration data

3.2 struct **ConfigurationResponse**

Below is a description of a the response to fetch a configuration for a local cloud system.

Field	Type	Description
id	Integer	Configuration id
systemName	Name	System name
filename	String	Name of the uploaded config file
contentType	Content-type	Type of data (text/plain, application/json, etc.)
data	Base64	Base64-encoded configuration data
createdAt	Timestamp	Time of creation.
updatedAt	Timestamp	Time of last update.

3.3 struct **ConfigurationListResponse**

Below is a description of the reponse to list all configurations for all local cloud systems.

Field	Type	Description
Count	Integer	Number of entities
[Data]	ConfigurationResponse	Array of stored configuration entities

3.4 Primitives

Types and structures mentioned throughout this document that are assumed to be available to implementations of this service. The concrete interpretations of each of these types and structures must be provided by any IDD document claiming to implement this service.

Type	Description
Id	Unique integer that represents a configuration.
Name	Unique String that points to a Producer or Subscriber.
Base64	Base64 encoded data
Address	String of a host, it contains the IP Address and can include also the port number.
Content-type	String of media data, examples are text/plain, application/json, etc.
Hash	MD5 hash of data
Timestamp	ISO8601 based datetime object.

4 References

[1] J. Delsing, “IoT Automation : Arrowhead Framework,” 2017.



ARROWHEAD

Document title
Configuration
Date
2021-02-05

Version
0.9
Status
DRAFT
Page
9 (9)

5 Revision History

5.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2021-02-03	0.1	Initial	Jens Eliasson
2	2021-02-04	0.5	Updated datamodels	Jens Eliasson
3	2021-02-05	0.9	Updated primitives and functions	Jens Eliasson

5.2 Quality Assurance

No.	Date	Version	Approved by
1			