Monitoring

Service Description

Abstract
Monitoring of any microsystem and its produced and consumed microservices is important in a professional indutrial context. The Monitoring service provides some very basic interface key-pairs and a possibility to for dynamically update it self with modified, new or even removed interface key-pairs. All key-pairs are necessitated to be provided as meta-data to the service registration.

Contents

1 Overview

This document describes the Monitoring service, which enables very basic basic monitoring of a microsystem and its micrcoservices. The Monitoring service provides one very basic interface and mechanisms for dynamically update it self with modified, new or even removed interfaces.

The rest of this document is organized as follows. In Section 2, we describe the abstract message operations provided by the service. In Section 3, we end the document by presenting the data types used by the mentioned operations.

1.1 Significant Prior Art

Monitoring is not a simple service. For Arrowhead v4.6 and earlier individual services may have an echo interface.

1.2 How This Service Is Meant to Be Used

The monitoring service is a mandatory service for all core and support microsystems and a highly recomended service for any application microsystem.

The general idea is that a Monitor microsystem subscribes to all avialable Monitoring service. The subscription can hold conditions like e.g. time interval, battery level below X%, error reports.

The usage scenario is depicted in Figure 1

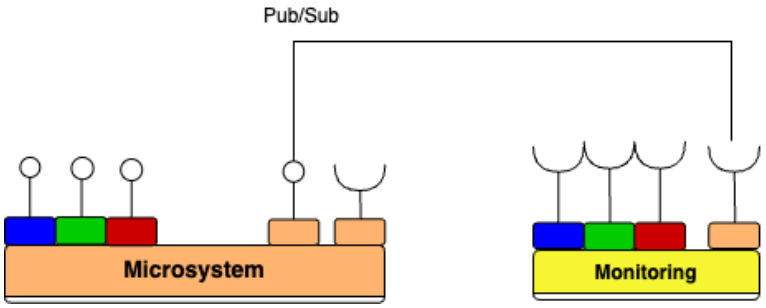

Figure 1: The general architectural usage of the Monitoring service isSysML SD diagram of the Monitoring service and its basic interfaces

This service is intended to provides the capability to monitor status of a microsystem and its associated microservices. Example of statuses are:

- Echo alive
- Power monitoring
- Test consumption of service for which the microsystem has a consumer.
- Internal state of the microsystem, e.g. remaining usefull life, ...
- Trigger a subscription push of data to a consumer
- Update of the monitoring service

1.3 Important Delimitations and Dependencies

The service privides only a very limited basic monitoring interfaces. Dynamically the servcie can be modified with updated, new or rewoked interfaces. A specifc upddate interface will be provided. This one can be protected nad thus limited to only accept systems or user comsumptions with certain certificate level. Specific security concerns need to be addressed while using this Update interface.

2 Service Interface

The Monitoring service has the interface depicted in 2

Figure 2: SysML SD diagram of the Monitoring service and its basic interfaces

The following basic interfaces are available.

2.1 Interface operation Echo (-) : StatusCodeKind

The Echo interface will responde with an TCP respnde code e.g. 100. Thus indicateing things are working as expected.

2.2 Interface operation Power (type,level) : type, level

The Power interface input parameters are: type and level. The responses are: Type=line—battery—harvesting Level=x[V]—xx%—nn[A]

Additional interfaces can be added by the individual producing microsystem. This is done specifiying

- interface: Namn, e.g. FreeMemory
- RequestParameters, e.g. MemoryType:FreeMem
- ResponseParameters, e.g. DRAM:2kB

Additional interfaces need to be provided to the ServiceRegistry when registering using the ServiceDiscovery service.

3 Information Model

3.1 struct echo

This structure is used to get an echo from the Monitoring serevice based on StatuCodeKind.

3.2 struct power

Provide type of power supply e.g. line, battery, harvesting. Provides level of power e.g. %, energy

3.3 struct CC

EXAMPLE: This structure is used to query service offering registered in the Service Registry. Please also refer to the activity diagram in Figure **??**

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.

4 Revision History

4.1 Amendments

No.	Date	Version	Subject of Amendments	Author
1	2024-05-26	5.0.0		Jerker Delsing
2		5.0.0		
3		5.0.0		

4.2 Quality Assurance

No.	Date	Version	Approved by
1		5.0.0	