SendGetData

Service Description

Abstract
This is the template for Service Description (SD document) according to the Eclipse Arrowhead documentation structure.

Contents

1 Overview

This document describes the SendGetData service, which is enables the transfer of data between systems.

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 How This Service Is Meant to Be Used

This service serves only one use - Send numerical data between two different systems. This service is used between multiple different systems such as between training data and preprocessing, pre processing and training model and between test data and evaluation of the training model and when returning the result.

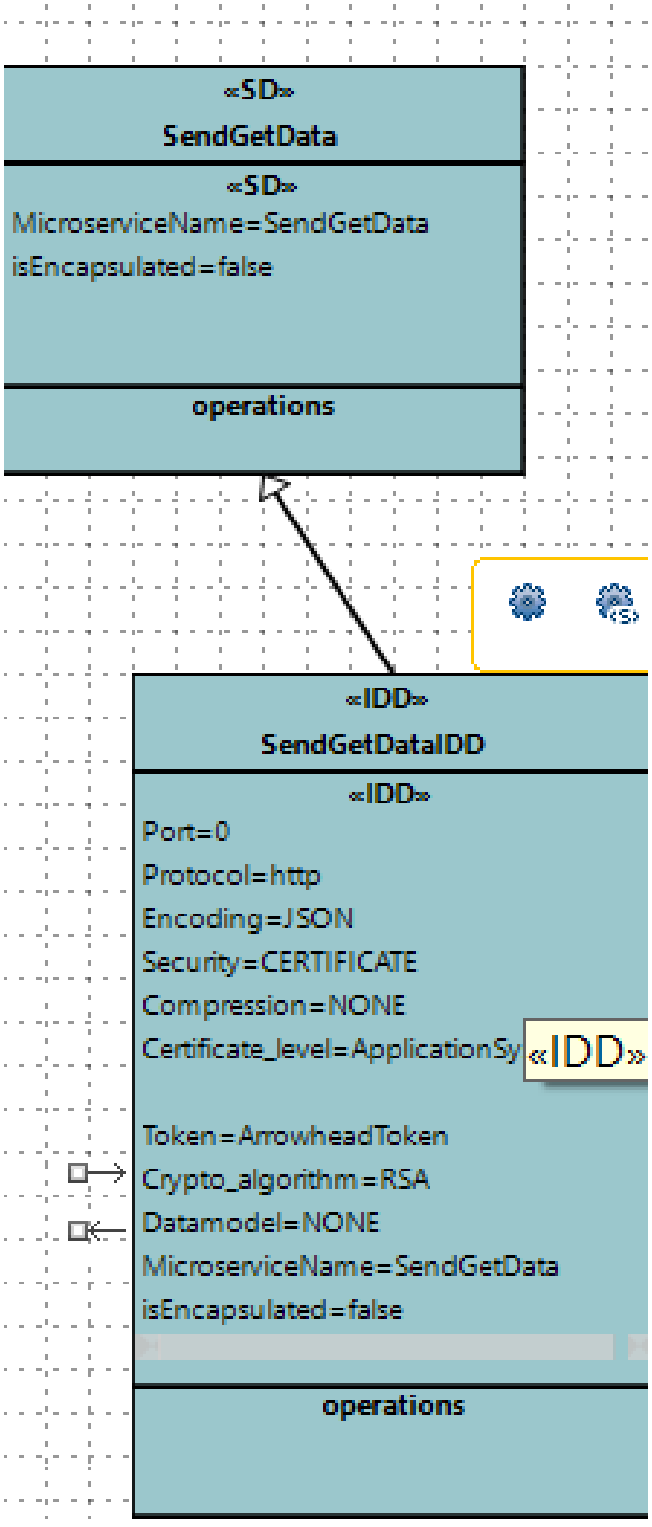

Figure 1: SD service and IDD document

2 Service Interface

This section describes the interfaces to the SendGetData service. In particular, each subsection names an abstract operation, 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 interface in question. All abstract data types named in this section are defined in Section 3. The following interface operations are available.

2.1 operation SendGetData (SendData) (integers) : integers

This operation sends numerical data from one system to another within the SoS. We are sending multiple numerical values as in a dataset.

2.2 operation SendGetData (GetData) (integers) : integers

The service gets numerical data from one system to another.

2.3 operation SendGetData (ResultData) (integers) : integers

Return the data as a result measure output.

2.4 operation Echo () : StatusCodeKind

The Echo operation provides an is alive response from the SendGetData service.

3 Information Model

3.1 struct TrainingRequest

This struct defines the format for a request to start an AI training task within the SoS.

- ‘modelID‘ (string): A unique identifier for the AI model being trained. - ‘datasetID‘ (string): The identifier for the dataset used for training. - ‘systemID‘ (integer): The identifier of the system where the task will be executed. - ‘startTime‘ (DateTime): The time when the training is scheduled to begin. - ‘hyperparameters‘ (map <string, float >): A collection of hyperparameters (e.g., learning rate, batch size) and their values. - ‘priority‘ (integer): An optional field indicating the priority level of the task (e.g., 1 for high priority, 5 for low priority).

As a complement to the explicitly defined types in this section, there is also a list of implicit primitive types in Section 3.2, which are used to represent things like hashes and identifiers.

This structure describes the profile of the service.

Field	Type	Description
modelID	string	A unique identifier for the AI model being trained.
datasetID	string	The identifier for the dataset used for training.
SystemID	integer	The identifier of the system where the task will be executed.
startTime	DateTime	The time when the training is scheduled to begin.
hyperparameters	Map<String, Float>	A collection of hyperparameters (e.g., learning rate, batch size) and their values.
priority	integer	An optional field indicating the priority level of the task (e.g., 1 for high priority, 5 for low priority).

Table 1: Struct: TrainingRequest

3.2 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
String	A sequence of characters, used for identifiers like modelID and datasetID.
Integer	A whole number, used for fields like systemID and priority.
DateTime	A representation of date and time, used to schedule training tasks (e.g., startTime).
Map<String, Float>	A key-value collection, where the key is a string (e.g., hyperparameter name) and the value is a float.

Table 2: Primitives Table for TrainingRequest Struct