Problem Statement

This assignment aims to practice the concepts, and techniques for data models and the communications for resources represented by data models.
The data set is from a Github project, under the directory of Workload Data.
https://github.com/haniehalipour/Online-Machine-Learning-for-Cloud- Resource-Provisioning-of-Microservice-Backend-Systems
The workload data contains the workload generated from two industrial benchmarks NDBench from Netflix and Dell DVD store from Dell. Both benchmarks are deployed on a cluster of cloud VMs on AWS and Azure clouds. The workload has been split to training sets and testing sets for machine learning purpose.
In each of the workload file, the first 4 columns contain the following attributes.
CPUUtilization_Average,NetworkIn_Average,NetworkOut_Average, MemoryUtilization_Average
In this assignment, please develop a client/server program to serve a "workload query" scenario. In this scenario, a client sends a 'Request For Workload (RFW)', and the server replies an 'Response for Data (RFD)' for each conversation.
The client's RFW includes:
1. RFW ID
2. Benchmark Type (such as DVD store or NDBench)
3. Workload Metric (such as CPU or NetworkIn or NetworkOut or Memory)
4. Batch Unit (the number of samples contained in each batch, such as 100)
5. Batch ID (such as the 1st or 2nd or... 5th Batch)
6. Batch Size (such as the how many batches to return, 5 means 5

batches to return)
Batch ID + Batch Size * Batch Unit
The server's RFD reply includes:
1. RFW ID
2. The last Batch ID
3. The samples requested
You are responsible for the design of the data model, and implementation of the data communication. You do not have to develop a full-fledged database system. Data can be stored in files or other types of storage.

Technical Requirement

1. Data Communication
The data should be communicated between the client and server through data serialization/deserialization in **two methods**, namely text based (de)-serialization and binary (de)-serialization. For example,
(1) XML or JSON can be used for text based (de)-serialization. (2) Protocol Buf or Thrift can be used for binary (de)-serialization.
For each method, your program should be able to retrieve the samples requested for each RFW.
2. Programming Language
You can program this application in any language.
3. Application
Your client/server can be a standalone program, or you build on any
software framework that supports client/server. You can choose the protocol your prefer TCP, or HTTP.