

Distributed Programming II

A.Y. 2018/19

Assignment n. 1 – part a)

The material for this assignment is in the *.zip* archive where you have found this file. Extract the archive to an empty directory that you will use as your working area and that we will call `[root]`.

DP2-RNS is a (simplified) distributed system that provides *Road Navigation Services*. The users of this system can get directions about how to reach a given destination place within the road system managed by *DP2-RNS* and their position can be tracked by the system. The concepts and terminology used in the *DP2* assignments are explained in the document file `[root]/intro.pdf`.

The Java interfaces defined in the package `it.polito.dp2.RNS` (available under `[root]/doc` and at <https://pad.polito.it:8080/enginframe/dp2/assignments/lab1/doc/index.html>) give read-only access to some information about the current status of the *DP2-RNS* system. **The javadoc of the interfaces documents the kind of information that can be retrieved.** The main interface, from which all the information can be accessed, is `RnsReader`. The methods in this interface can be used to get the information available about the road system itself (places and their connections) and about the vehicles that are currently tracked in the system. These methods return sets of interfaces, by which all the available information about the *RNS* system and its status can be obtained.

The solution to part a) will be submitted along with the solution to part b).

Assignment description

1. Design an ***XML* application** that can be used to store all the information that can be retrieved by using the *Java* interfaces defined in package `it.polito.dp2.RNS`, starting from interface `RnsReader`. The *XML* format must be such that all the data that can be retrieved using such interfaces can also be obtained from the *XML* document, and redundancies should be avoided. The *XML* application must be specified by means of an *XML Schema*, which must be saved in file `[root]/xsd/rnsInfo.xsd`. The potential of *XML* schemas must be exploited, in order to accurately represent constraints on data types, and keep checks as much as possible within the *XML* processor (rather than in the application).
2. Write a short documentation of your design choices about the schema (max 1 page) and save it as an ASCII file in `[root]/doc.txt`.
3. Write a valid *XML* file that references the designed *schema* locally. The file must be saved as `[root]/xsd/rnsInfo.xml`

Correctness verification

Before submitting your files, please verify their correctness. The submitted solution must at least satisfy the following requirements, in order to be considered acceptable:

- the file `[root]/xsd/rnsInfo.xsd` must be a valid *XML Schema*;
- the file `[root]/xsd/rnsInfo.xml` must be valid with respect to the above schema

and must reference it, assuming it is stored in the same directory;

- the file `[root]/doc.txt` must exist.

The validity of the XML file and of the schema can be checked by any XML validation program. For example, it can be checked by the Eclipse validate command.