CPSC 551 - Distributed Systems - Fall 2019

Note: this is a preliminary draft of the project as of 2019-09-23. Bookmark this page and check back for updates.

Project 1, due October 14

Background

Note 2.3 on pp. 68-70 of the textbook discusses Linda tuple spaces and provides example code using a Python Package named *PyLinda*. Unfortunately,

- The PyLinda package is not not the same as the pylinda package available on PyPI.
- PyLinda was written for Python 2.
- The available code for PyLinda is missing some source files.
- The original author of the code seems to have disappeared from the Internet.

Fortunately Ruby, another common scripting language, includes a module named *Rinda* which implements the Linda distributed computing paradigm. Even better, Chapter 2 of the textbook describes the *adapter* pattern for middleware, so that we can bridge the gap between Python and Ruby.

Starting with Rinda

Write a Ruby server program `tuplespace.rb` to create a Rinda TupleSpace and begin listening on a TCP port for incoming dRuby connections.

Write a Ruby client program `blog.rb` to connect to the Rinda TupleSpace and perform the same operations as the example code fragments in Note 2.3 in the textbook.

Note: the Ruby equivalent of the Python `str` type is `String`, and the `in(t)`, `rd(t)`, and `out(t)` operations are named `take(t)`, `read(t)`, and `write(t)`, respectively.

Adapting Rinda to XML-RPC

According to WIkipedia, "XML-RPC is a remote procedure call (RPC) protocol which uses XML to encode its calls and HTTP as a transport mechanism." XML-RPC support is included in the Python Standard Library and until recently was included in the Ruby Standard LIbrary as well.

Write a Ruby server program `adapter.rb` to act as an XML-RPC server. Connect to the Rinda TupleSpace and add handlers named `_in`, `_rd`, and `_out` to allow the TupleSpace operations to be called via XML-RPC.

Using Rinda from Python

Write a Python client program `blog.py` to connect to the XML-RPC adapter and perform the same operations as the example code fragments in Note 2.3 of the textbook.

Write Python client programs `server.py` and `client.py` equivalent to the example programs for performing arithmetic operations shown on the Wikipedia page for Rinda.

Finally, solve the "exercise to the reader" described at the end of Note 2.3 by writing a Python client program `next.py` that extends the code fragments such that a *next* message will be selected instead of a random one.

Platform

You may use any platform to develop and test the project, but note that per the Syllabus the test environment for projects in this course is a Tuffix 2019 Edition r2 Virtual Machine with Python 3.6.8 and Ruby 2.5.1p57. It is your team's responsibility to ensure that your code runs on this platform.

Libraries

The Python xmlrpc.client and Ruby Rinda modules are included in the language's standard libraries. XML-RPC support was removed from the Ruby standard library starting in version 2.4, but can be installed as a Ruby Gem (similar to a Python Package). To install the `xmlrpc` gem on Tuffix, use the following command:

```
$ gem install --user-install xmlrpc
```

Tips

- For a quick tutorial on Ruby, see Ruby in Twenty Minutes.

Submission

Submit your project by uploading your Python and Ruby source code and any other relevant artifacts to the `project1/` subdirectory of the folder that will be shared with you on Dropbox.

You may work alone, or make a single submission for a team of 2-3 students. If you work in a team, only one submission is required, but for safety consider uploading copies to each team member's submission folder. (Make certain, however, that the copies are the same in each case; the instructor will not attempt to ascertain which is the "real" submission.)

A printed submission sheet will be provided on the due date. To finalize your submission, fill out the sheet with the requested information and hand it in to the professor by the end of class.

Failure to follow any submission instructions exactly will incur a **10%** penalty on the project grade for all team members.