Université Paris-Saclay
Master 2 Data and Knowledge

Professor: Ioana Manolescu
Teaching assistant: Paweł Guzewicz[1]

Big Data Architectures Lab 1

September 23, 2019

Submission deadline: Sunday, October 6, 2019 at 23:59:59.

1 Lab organization[2]

1.1 Key-value stores recall

Please find the presentation at `https://www.lix.polytechnique.fr/Labo/Pawel.Guzewicz/teaching/2019_2020_Big_Data_Architectures/lab_1_slides.pdf`.

1.2 Redis setup

1. Download and install Redis following instructions at `https://redis.io/topics/quickstart`.
2. Launch Redis and check if it works.

If you use Windows as your operating system, please consider installing a virtual machine (e.g. VirtualBox) and a Linux distribution onto it (some popular distros: Ubuntu, Linux Mint, Debian, Fedora/CentOS, Arch Linux) so that you can go through the lab under the virtual environment.

1.3 Redis tutorial

To warm up, start with a tutorial on Redis. Learn some basic commands at `https://try.redis.io`.

1.4 Assignment and questions

Please read the assignment description in the next section. If you have any questions about the lab or on the course material, don't hesitate to ask them during the lab session or via e-mail (don't wait until the last moment).

2 Assignment

With the help of a programming language interface to Redis, write a toy application of a database of books that can be borrowed in a library. Use can use any supported language among the ones listed at `https://redis.io/clients`.

2.1 Application specification

1. All books in the library database have an ISBN, a title, an author and a number of copies.
2. Books may also have other properties, e.g., language, publication year, edition... (up to you).
3. Create a publish-subscribe news system (see `https://redis.io/topics/pubsub`) that:
   (a) on the **publisher** side lets the user add a book to the library, indexes it by the keywords in its description, publishes a channel for each indexed keyword, and emits a news message containing the newly published book ID;

[1] pawel.guzewicz@lix.polytechnique.fr; `https://www.lix.polytechnique.fr/Labo/Pawel.Guzewicz/teaching`
[2] All lab materials based on previous course editions courtesy of Ioana Manolescu and Silviu Maniu (with some changes)

(b) on the **subscriber** side enables the user to subscribe to news channels matching certain keywords, retrieve a book from a news by the ID, and show the full book entry from the database; separately the user can borrow or return a book: the system needs to check if the book is available;

(c) makes books expire after a while (if no one borrows the book): these are no longer available to borrow;

(d) if someone borrows a book (by, e.g., setting a certain field in the database), makes the book refresh its expiry date.

To understand better how to interact with some commands through Redis API, you can look at a tutorial implementing a Twitter clone in a PHP client. It can be found at `https://redis.io/topics/twitter-clone`.

2.2 Report

Write a short report on your implementation. It should include the following elements.

1. Setup information: which programming language did you use, what are the necessary steps to use a Redis client (e.g. for python you need to run `pip install redis` to install Redis package), and, most importantly, **how to run your code**.

2. Supported scenarios: how to publish a book, how to subscribe to a channel, how to borrow/return books, how does the client detect an expired book, etc.

3. An example of the program execution per scenario: terminal screenshots, a list of commands or a script along with their outputs.

2.3 Submission guidelines

Please follow submission rules and guidelines: `https://www.lix.polytechnique.fr/Labo/Pawel.Guzewicz/teaching/2019_2020_Big_Data_Architectures/submission_rules_and_guidelines.pdf`.

Moreover, I encourage you to read my advice on lab sessions and submissions: `https://www.lix.polytechnique.fr/Labo/Pawel.Guzewicz/teaching/2019_2020_Big_Data_Architectures/advice_on_lab_sessions_and_submissions.pdf`