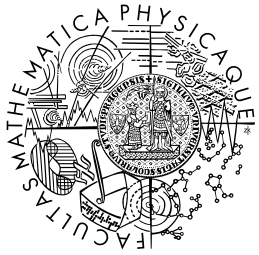

FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

BACHELOR THESIS

Přemysl Šťastný

LSQL language and it's lsql-csv implementation for csv files

Department of Applied Mathematics

Supervisor of the bachelor thesis: Jan Hubička
Study programme: General Information
Study branch: Informatics

Prague 2021

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ....

Author's signature

Dedication.

Title: LSQL language and it's lsql-csv implementation for csv files

Author: Přemysl Šťastný

Department: Department of Applied Mathematics

Supervisor: Jan Hubička, Department of Applied Mathematics

Abstract: LSQL is language, which is new database query language optimilized for simple onetime queries in command line enviroment. lsql-csv is project implementing LSQL for csv files. The thesis is about reason and design of the language and it's lsql-csv implementation.

Keywords: lsql lsql-csv unix kiss unix-philosophy haskell database csv new-language data-analysis data-query

Contents

Introduction

1. Introduction

1.1 What is LSQL? What it is good for?

Why would anyone come with a new query language for flat data, when there are standartized languages for doing so? It's easy.

The widly used standard for querying flat data is SQL. The SQL is designed to make a human (and machine) readable queries, which can be contained in large projects with many people. The readability of written code comes in the first place and therefore it was designed not for the comfort of the programmer, who actually writes the code, but for hunderts of people, who comes after him and tries to find out, what his code does.

This is the opposite of what LSQL is developed for. We try to make a "write-only" language, which will make an unix user more comfortable on his machine. It doesn't care about constraints, try to ignore types as much as possible and is designed by the means of unix philosophy.

Why should we use a poweruser-friendly tool, when we have user-friendly tool to do the same thing, faster[1] and maybe better? Like Excel, Calc or Django admin? Simply put, there are use cases, where user-friendly tools unnecessarly complicate the whole situation, and you want the solution to be as simple as possible. For the sake of your brain, your time, your psyché, the maintability and lifetime[2] of data and scripts and amount of information, you have to remember.

All of these questions will be discussed later in text.

1.2 What is lsql-csv?

lsql-csv is tool implementing LSQL for quering csv files. The main[3] ambition of this project is to get into standartized UNIX ecosystem. It is simple, useful tools coresponding the KISS (keep it simple stupid) and UNIX philosophy (mainly do only one thing and do it right).

Similary of the Java gredo is Write once, run anywhere, the author thinks, the UNIX ecosystem gredo should be Write once, run forever. The reason, why would you want to store data in csv is not only its simplicity and usefulness, but that you can be sure, you can open UTF-8 csv file 30 years later, and you will be probably able to run your UNIX ecosystem scripts.

1.3 Who is this text for?

This text is for people, who wants to understand the usefulness of LSQL, lsql-csv and the reason, why and how they exist. If you want to try the tool without knowing anything deeper about it, you might consider to read README.md instead of this text.

[1] You don't have to learn new language.

[2] Have you noticed, how offen Microsoft Excel or Postgresql are changing the database format and how they are complex, when we compare them to csv?

[3] and unrealistic

2. Use cases and their standard solutions with or without lsql-csv

It was once sayed, that a simple example can be more then thousand words. The author absolutly agrees with this thesis and therefore before any generilezed analysis outcome, he has decided to write down the real world motivation.

2.1 Student subject evidence

2.2 Rapid analysis of statistical data

2.3 Madhouse internet censorship

Conclusion

Bibliography

List of Figures

List of Tables

List of Abbreviations

A. Attachments

A.1 First Attachment