FOM Hochschule für Oekonomie & Management

university location Bonn

Exposé

in the study course Wirtschaftsinformatik

on the subject

Development of a Query Language for Full-Text Search in Relational Databases

by

Sebastian Bunge

Advisor:	Prof. Dr. Peter Steininger
Matriculation Number:	539441
Submission:	July 30, 2022

Contents

1 Problem of the Thesis and Scientific Question

What "problem" do you want to work on or better understand?

Current relational databases go far beyond the transactional processing of data with their functionalities. One of these functionalities is the possibility of a full-text search over data. For this full-text search, powerful extensions of the database query language (SQL) are available, which generate complex query terms by an extensive combination of operators. The goal of this thesis is to generate query terms for the database based on a search engine-like input. The starting point for the input can be e.g. the search operators of Google.
The scientific question may be defined as: In what way can a software component be developed with a custom query language and interpreter to enable full-text search on Microsoft SQL servers?

2 Initial Situation and Classification in the Existing Research

What is the scientific environment of the topic? In which larger context/question would you place the topic?

The environment of grammar design, interpreters and basic understanding of full-text search.

Why is the topic relevant? From what do you deduce that your question is relevant? Who might have an interest in or benefit from the results?

With options such as word and phrase-based searches and the ability to index documents in their native formats for example Office documents and PDFs, full-text search functionality is an increasingly important function in modern databases and document management systems. Being able to operate such a mighty feature with an easy-to-understand grammar instead of learning SQL is just one example of why custom languages generating code are getting in higher demand.

What is already available? What has already been tried?

There are many examples of code generators to either make writing code more accessible or assist experienced developers. This specific example of full-text search is often used in a simplified way by search engines.

3 Approach and Methodology

Do you want to collect data independently (empirical research) or rely on existing data/literature?

Rely on existing literature for concepts and try to implement them. (Prototype implementation)

4 Outline

1. Introduction
2. Theory
    a) Elaboration of the language elements for full-text search in common relational databases (Microsoft SQL Server)
    b) How to build interpreters
3. Prototype
    a) Definition of grammar for full-text search input and schematic description of the grammar using the extended Backus-Naur form
    b) Conversion of the grammar into a software component in a web application
4. Source Code
5. Examples and Usage
6. Conclusion

5 Literature

What literature do you want to draw on? How do you search for and find it?

- Pro full-text search in SQL Server 2008 by Coles
- Crafting Interpreters by Nystrom
- Compilers: Principles, Techniques, and Tools by Aho

What recent articles have you read on the subject?

- Several articles by Holger and Hannah Bast develop tools to use full-text search
- Articles explaining the basics of compilers and code generators (Tokenizer, Lexer etc.)

6 Working Title

Development of a Query Language for Full-Text Search in Relational Databases

7 Objective / Expected Result

- A website with a search box, where search criteria are defined
- Able to use search reflections, search specific question
- SQL code is generated for full-text search
- Datagrid with the tuples of the result

8 Project / Time Schedule

Schedule here