ManBearPig. An attempt at providing native full text search to the mandoc system.

spiros thanasoulas st19@illinois.edu

November 2, 2020

Description

The goal of this project would be to create a report and possibly code improvements towards providing a backend that supports Full Text Search capabilities for the mandoc project (https://mandoc.bsd.lv/).

Background

UNIX system provide their documentation to the user through a set of tools collectively referred to as the Manual Page system. The well known man(1) command exists today on all UNIX systems but even on other platforms like MacOSX and android. Searching efficiently keywords and semantics has been of paramount importance for the user to quickly get to the relevant manual page and the command apropos(1) traditionally served that purpose, meaning doing database lookups. The databases are built with the makewhatis(1) tool.

Project proposal

We will investigate the C source code of the mandoc project, targeted on the modules of searching and database creation. The goal of this project would be to lay a path for full text search capabilities from the apropos command. Currently only certain words of a manual page are indexed and their semantic information stored with them, in a persisted to a file database that on the outerlevel is implemented as a hash map.

To allow for the full text search capabilities we will implement a database based on trigrams keying an inverted index of the full text being contained in a manual page, after it has been parsed from the mdoc parsers and only the content remains.

In detail the goal of the project would be to create the equivalent database of the makewhatis(1) db that is currently created, but which stores the trigrams. Due to lack of time no optimizations for very large databases are going to be implemented and the testing input will be constrained enough to make sure the datastructures will be able to fit in memory.

The generated database will be evaluated by dumping the contents and making sure all the trigrams that should be produced and only those are contained within it. A test harness to ensure that will be provided.

If time permits the search capabilities will be attempted to connect to the database through the apropos command and query for a text string.

note to the reviewer: although i would love to finish the whole thing but it might be unfeasable in around 25hrs that i have budgeted it for it. My intention though is to lay the foundation so that a patch will be eventually merged in the mandoc codebase, not to demo something that noone will ever user

Proposed Workflow

We propose that the analysis and development will be split across 5 6hr man - days of work

Day 1	Code and Documentation analysis. Reviewing the makewhatis utility and the resulting databases it creates examine the relevant code flow and find where to plugin the new functionality.
Day 2	Design of the binary file format that will store to the trigrams data structure, as well as the parsing functions to extract them.
Day 3	Development and Documentation
Day 4	Development, Documentation and test harness
Day 5	Final report

members

st19 / solo project