Software Engineering Project
Progress Report

Xiaodong Cui

Group 1

1149546

Table of Contents

1. Introduction

This report is the progress report for the tasks I have done for the last few weeks. I was assigned the task to look at ticket 138 by group 1. In this report , I will discuss the progress that I have made for this ticket, and also the tasks that is still pending for this ticket.

2. Ticket 138 Description

Radial and Treemap views when combined with filters have very long running queries. We do not use treemap in the current version of earth. This version for this ticket is 0.1. From current version of earth 0.2, it says that Improved speed of radial graph, addresses performance problem described in ticket #138. But this ticket is still open, we may need to enhance the performance.

3. What I Have Done

3.1 Problem Solved

There is a bug when you want to generate radial views together with a filter, it gives the following error,

PGError: ERROR: column reference "bytes" is ambiguous

LINE 1: SELECT SUM(bytes) AS sum_bytes, SUM(blocks) AS sum_blocks, C...

It does not know where to get the data from the “bytes” field, because both “files” and “directory” table contain this “bytes” field, it is ambiguous to know which table should be used here. The query should look for information from “files” table, the aggregate function SUM(bytes) and SUM(blocks) should be changed to SUM (files.bytes) and SUM(files.blocks).

3.2 Ticket 138 Analysis

In order to check the current performance of the radial graph, I checked out the old earth version 1004, and asked it to monitor the same directory, then managed to get response times of generating radial view for both version 1004 and 1179 using ruby-debug, also the response time can be displayed in the browser. The results are listed in Table 1.

Earth 1004 (23385 rows in files table)	Earth 1179 (23385 rows in files table)
Average: 70-80 seconds	Average: 0.73-0.82 seconds

Table 1: Performance Test with Different Version of Earth

From the table, current performace has been greatly improved, but why the ticket is still open? Then I made further tests. I measured the query performance on earth 1179 with different sizes of database (by adding huge number of files in the directory monitored by earth), the results are listed in Table 2.

23385 rows in files table	44696 rows in files table	66022 rows in files table
Average: 0.73-0.82seconds	Average: 3seconds	Average: 7seconds

Table 2: Performance Test with Different Sizes of Database

From 20K rows to 40K rows, the size of the table is increased by 2 times, but the time taken to generate the radial graph is increased by 4 times. Similar, from 20K rows to 60K rows, size is increased by 3 times, but the time taken is increased by 9 times. The result is not scale, we need to improve it further.

If we need to improve the current performance, we need to find where the query spend its time. There is a method find_files_down_to_level method under the model file directory.rb returns all files belonging to sub-directories of the current directory down to the given level. It contains a single query, but it takes almost 60% of the overall running time, so we need to improve this method.

3.3 More Work Need to be Done

We need to optimize the query to reduce the searching time in the database, we may create a meta table to hold all the size information we need, then we can only search in that table, we may also need to re-write the query to get better performace.

4. Discussion

Based on the measurement I have made for this ticket, this ticket still needs to be enhanced. The performance of that particular query which takes most of the running time needs to be improved. There may some other methods affect the running time can be improved as well. Next thing we do for this ticket is to actually enhance performace, to get radial graph with filters with shorter time.