BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC ĐÀ NẴNG

ĐỒNG THỊ ÁNH PHƯỢNG

NGHIÊN CỨU ỨNG DỤNG CẤU TRÚC DỮ LIỆU TRIE CHO TÌM KIẾM CHUỖI KÝ TỰ

Chuyên ngành : **KHOA HỌC MÁY TÍNH**
Mã số : **60.48.01**

TÓM TẮT LUẬN VĂN THẠC SĨ KỸ THUẬT

Đà Nẵng - Năm 2012

Công trình được hoàn thành tại

ĐẠI HỌC ĐÀ NẴNG

Người hướng dẫn khoa học: **PGS.TS. VÕ TRUNG HÙNG**

Phản biện 1: **TS. NGUYỄN THANH BÌNH**

Phản biện 2: **TS. NGUYỄN MẬU HÂN**

Luận văn được bảo vệ tại Hội đồng chấm Luận văn tốt nghiệp thạc sĩ kỹ thuật họp tại Đại học Đà Nẵng vào ngày 03 tháng 03 năm 2012.

Có thể tìm hiểu luận văn tại:

- Trung tâm Thông tin - Học liệu, Đại học Đà Nẵng
- Trung tâm Học liệu, Đại học Đà Nẵng

MỞ ĐẦU

1. Lý do chọn đề tài

Gần đây, hệ thống kho dữ liệu ngày càng được mở rộng và đóng vai trò quan trọng hơn đối với người ra quyết định; hầu hết các truy vấn đối với một kho dữ liệu lớn rất phức tạp và lặp đi lặp lại; khả năng trả lời những truy vấn hiệu quả là một vấn đề mà nhiều hệ thống đang hướng đến. Làm thế nào để tăng tốc độ, cải thiện hiệu suất truy vấn luôn là câu hỏi lớn và không ngừng tìm kiếm lời giải đáp tối ưu. Hiện nay có rất nhiều kỹ thuật được áp dụng nhằm tăng hiệu quả truy vấn, mỗi kỹ thuật đều có những thế mạnh riêng, TRIE là một cấu trúc dữ liệu đang được triển khai sử dụng trong các hệ thống tìm kiếm lớn hiện tại bởi nhiều tính năng ưu việt giúp đẩy nhanh tốc độ và hiệu quả của quá trình truy vấn.

Trước thực trạng đó, tôi chọn nghiên cứu và thực hiện đề tài "*Nghiên cứu ứng dụng cấu trúc dữ liệu TRIE cho tìm kiếm chuỗi ký tự*" dưới sự hướng dẫn của PGS. TS Võ Trung Hùng. Đề tài phát triển sẽ giúp cho sinh viên nói riêng và những người nghiên cứu về Công nghệ thông tin nói chung có thêm tài liệu hỗ trợ triển khai cấu trúc dữ liệu này phục vụ cho công tác tìm kiếm chuỗi ký tự bên cạnh các cấu trúc dữ liệu đang sử dụng hiện nay trong một số hệ quản trị cơ sở dữ liệu lớn, đặc biệt là các hệ quản trị cơ sở dữ liệu mã nguồn mở.

2. Mục tiêu và nhiệm vụ nghiên cứu

Mục tiêu của đề tài là cấu trúc dữ liệu Trie được tìm hiểu và trình bày cụ thể kèm theo việc ứng dụng trong MariaDB.

Nhiệm vụ nghiên cứu bao gồm phần nghiên cứu lý thuyết về các phương pháp tạo chỉ mục và tìm kiếm; tìm hiểu các phương pháp Hash index, Bitmap Index, Btree Index và nghiên cứu cấu trúc dữ liệu Trie, các biến thể của Trie, các thao tác cơ bản trên cấu trúc dữ liệu này. Dựa trên các nghiên cứu lý thuyết đó, đề tài đưa ra được tài liệu Tiếng việt về cấu trúc dữ liệu Trie phục vụ cho việc học tập và nghiên cứu.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài gồm: Cơ sở lý thuyết về các phương pháp tìm kiếm, truy xuất dữ liệu, chỉ mục và các kỹ thuật lập chỉ mục phục vụ tìm kiếm, các giải thuật liên quan đến cấu trúc dữ liệu TRIE.

Phạm vi nghiên cứu về hệ thống tìm kiếm thông tin nói chung và các kỹ thuật lập chỉ mục phục vụ công tác tìm kiếm thông tin (Hash Index, Bitmap Index, Btree Index), trọng tâm đi sâu tìm hiểu cấu trúc dữ liệu TRIE, các biến thể Trie nén và các thao tác căn bản trên Trie, Trie nén.

4. Phương pháp nghiên cứu

Đề tài được triển khai bằng các phương pháp nghiên cứu sau: Phương pháp tài liệu nhằm thu thập, phân tích và tổng hợp tài liệu liên quan đến vấn đề lý thuyết, phương pháp mô hình hóa và phương pháp thực nghiệm.

5. Ý nghĩa khoa học và thực tiễn của đề tài

Kết quả nghiên cứu có thể làm tài liệu tham khảo cho việc tìm hiểu các phương pháp lập chỉ mục phục vụ tìm kiếm và so sánh hiệu quả giữa chúng, đặc biệt là tài liệu về cấu trúc dữ liệu Trie phục vụ tìm kiếm. Ngoài ra, phần nghiên cứu lý thuyết sẽ cung cấp một cách nhìn tổng quát về hệ thống tìm kiếm, các phương pháp tìm kiếm.

6. Bố cục luận văn

Luận văn được trình bày cơ bản bao gồm 3 chương chính.

CHƯƠNG 1: TỔNG QUAN VỀ TÌM KIẾM THÔNG TIN TRÊN VĂN BẢN

CHƯƠNG 2: TRIE - CẤU TRÚC DỮ LIỆU TÌM KIẾM CHUỖI KÝ TỰ

CHƯƠNG 3: TRIE TÌM KIẾM TRÊN CƠ SỞ DỮ LIỆU MARIADB

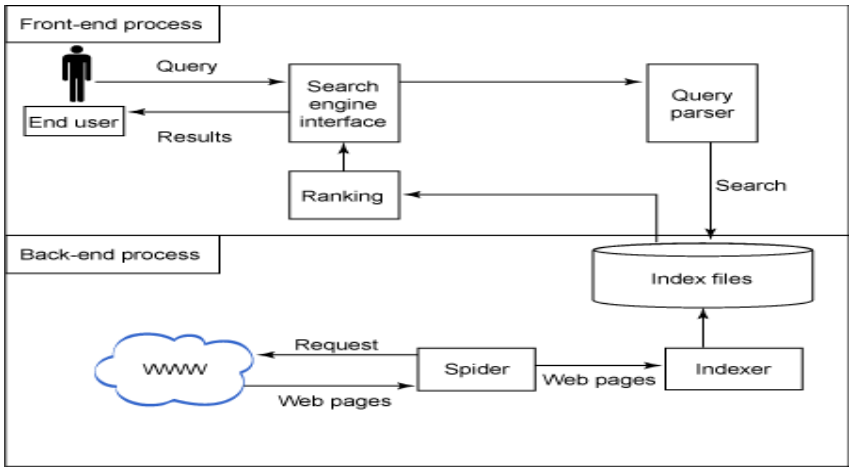
CHƯƠNG 1: TỔNG QUAN VỀ TÌM KIẾM THÔNG TIN TRÊN VĂN BẢN

Trong chương này chúng tôi sẽ trình bày khái quát về tìm kiếm thông tin (*Retrieval Information*) và cấu trúc cũng như phương thức hoạt động của hệ thống tìm kiếm. Bên cạnh đó chúng tôi sẽ giới thiệu một số hệ thống tìm kiếm trên Internet và trên Desktop đang phổ biến hiện nay. Cuối chương, chúng tôi sẽ trình bày một số đánh giá và định hướng cho việc ứng dụng mã nguồn mở.

1.1. TÌM KIẾM THÔNG TIN

1.1.1. Khái quát về tìm kiếm thông tin [1],[2],[3]

1.1.2. Mô hình tìm kiếm [2]



Hình 1.1. Mô hình tìm kiếm [2]

Hình 1.1 mô tả một mô hình tìm kiếm thông tin trong đó "front-end process" là các bước xử lý liên quan đến phần chương trình tương tác trực tiếp với người sử dụng, điều khiển việc giao tiếp với người sử dụng; "back-end process" là các xử lý liên quan đến phần chương trình phụ trợ phía sau, thường được đặt trên máy chủ. Query parser phân tích cú pháp truy vấn và Search engine interface là giao diện của máy tìm kiếm. Hình vẽ này miêu tả nhiệm vụ tương ứng với nhiệm vụ của Bộ thu thập thông tin, Bộ lập chỉ mục và Bộ tìm kiếm thông tin đã được nêu phía trên.

1.2. MỘT SỐ PHƯƠNG PHÁP LẬP CHỈ MỤC CHO TÌM KIẾM CHUỖI KÝ TỰ - VĂN BẢN [3], [4]

1.2.1. Hash Index

1.2.2. Btree Index

1.2.3. Bitmap Index

1.3. MỘT SỐ HỆ THỐNG TÌM KIẾM HIỆN CÓ

1.3.1. Công cụ tìm kiếm trên mạng internet

1.3.2. Công cụ tìm kiếm trên máy tính cá nhân

1.4. KẾT LUẬN VÀ ĐÁNH GIÁ

Trong chương này chúng ta đã tìm hiểu những điểm cơ bản của thông tin và các hình thức lưu trữ của chúng trên máy tính. Chương này cũng nghiên cứu cơ bản các phương pháp tìm kiếm thông tin đã và đang được ứng dụng trong lĩnh vực tài liệu điện tử. Đặc biệt, nghiên cứu các phương pháp lập chỉ mục phục vụ cho tìm kiếm chuỗi ký tự - văn bản, điển hình là phương pháp Hash Index, Btree Index và Bitmap Index. Thông qua việc tìm hiểu các phương pháp lập chỉ mục này, để tìm ra được những hạn chế khi thao tác trên các phương pháp đó, và đề xuất tìm hiểu phương pháp mới khắc phục được các hạn chế ấy nhưng vẫn đảm bảo kế thừa được các đặc tính tích cực đã có. Cấu trúc mới được đề xuất là Trie, được trình bày cụ thể trong chương sau.

Bên cạnh đó còn tìm hiểu cấu trúc của hệ thống tìm kiếm và tìm hiểu một số công cụ tìm kiếm trên Internet và trên Desktop đang phát triển ứng dụng hiện nay. Qua những kiến thức đó, chúng ta cơ bản đã nắm được những lý thuyết về lĩnh vực tra cứu tìm kiếm thông tin, nắm được một số đặc điểm của các ứng dụng tìm kiếm mà các hãng sản xuất lớn đã phát triển. Từ đó, tổng hợp được những lý thuyết cần thiết nhất cho việc xây dựng một ứng dụng tương tự hay kế thừa thư viện mã nguồn mở để phát triển theo đúng quy chuẩn.

CHƯƠNG 2: TRIE - CẤU TRÚC DỮ LIỆU TÌM KIẾM CHUỖI KÝ TỰ

Trong chương 1, chúng ta đã tìm hiểu những kiến thức tổng quát liên quan đến tìm kiếm thông tin trên văn bản và các phương pháp lập chỉ mục đã được sử dụng, mỗi phương pháp có một số ưu điểm và hạn chế nhất định; Trong chương này, chúng ta sẽ tìm hiểu về một cấu trúc dữ liệu mới: Trie. Cấu trúc này được sử dụng kết hợp với các cấu trúc đã được lập chỉ mục trước đây nhằm khắc phục những hạn chế đã nêu.

2.1. CẤU TRÚC DỮ LIỆU TRIE [5], [6]

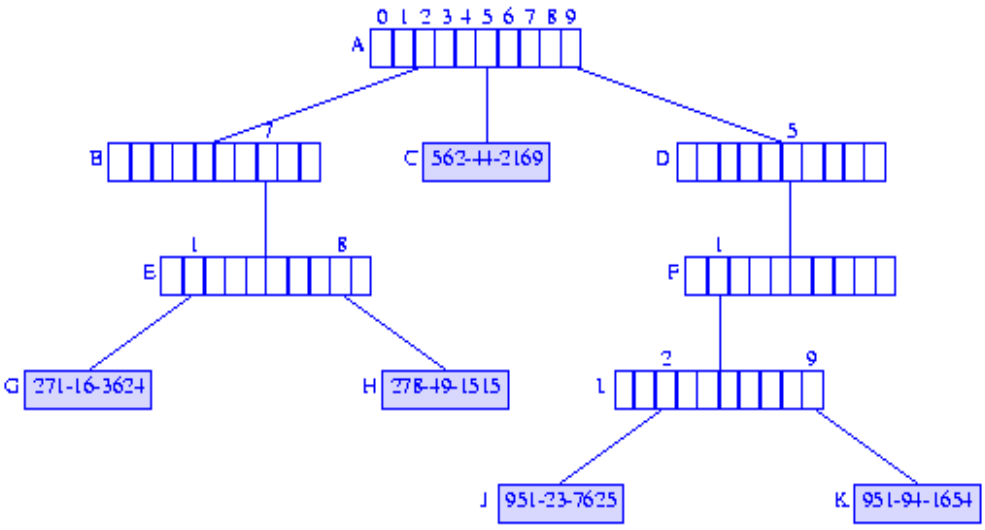
TRIE, phát âm là “try”, là từ xuất phát từ chữ *retrieval,* người phát minh ra nó là Edward Fredkin; là một cấu trúc dữ liệu sử dụng ký số trong khóa để tổ chức và tìm kiếm. Mặc dù trong thực tế chúng ta có thể sử dụng rất nhiều hệ cơ số để phân tích các khóa bên trong các ký số, ví dụ chúng ta có thể chọn các số tự nhiên (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) hoặc các ký tự trong bảng chữ cái Tiếng Anh (a – z, A – Z).

Giả sử các phần tử trong từ điển cần tìm kiếm là hồ sơ Sinh viên có chứa các trường như: Tên SV, chuyên ngành học, ngày sinh, Mã số. Trong đó, trường khóa là “Mã số”, được biểu diễn bằng chín ký số thập phân từ 1 đến 9. Để thể hiện ví dụ này, giả sử rằng từ điển chỉ có năm phần tử. Trường Tên và Mã số của mỗi năm phần tử được hiển thị như hình bên dưới:

TÊN	MÃ SỐ
Hoa	951-94-1654
Thanh	562-44-2169
Anh	271-16-3624
Vy	278-49-1515
Phong	951-23-7625

Hình 2.1. Năm phần tử trong từ điển [6]

Chúng ta phân chia thành 3 nhóm, những phần tử có Mã số bắt đầu bằng ký số "2"; những phần tử bắt đầu bằng ký số "5" và nhóm cuối cùng gồm các phần tử bắt đầu bằng ký số 9. Những nhóm nào có nhiều hơn một phần tử thì sẽ được phân chia sử dụng ký số tiếp theo trong khóa để phân biệt. Việc phân chia này sẽ được tiếp tục cho đến khi mỗi nhóm chỉ còn duy nhất một phần tử. Kết quả quá trình phân chia trên được mô tả trong một Tree có 10 đường nhánh như hình dưới.



Hình 2.2: Trie cho các phần tử ở hình 2.1 [7]

2.2. TÌM KIẾM TRONG TRIE

2.2.1. Khóa có chiều dài giống nhau

Để tìm kiếm một Trie cho một phần tử với khóa của nó, chúng ta bắt đầu từ gốc và duyệt dần xuống phía dưới cho đến khi hết Trie hoặc cho đến khi nút đang duyệt là một nút lá.

2.2.2. Khóa có chiều dài khác nhau

Trong ví dụ trên, tất cả các phím có cùng số ký số, là 9 ký số. Trong các ứng dụng thực tế, có thể sẽ gặp một số trường hợp mà các khóa khác nhau có số ký số khác nhau, thông thường, chúng ta sẽ thêm ký số đặc biệt (thường là #) vào cuối mỗi khóa.

2.3. CHIỀU CAO CỦA TRIE [6], [7]

Trong trường hợp xấu nhất, từ nút gốc đến nút lá, mỗi ký số trong khóa phải duyệt qua một nút nhánh. Khi đó chiều cao của Trie là tối đa, và là số ký số của khóa cộng 1. Trong trường hợp này, Trie mã số được minh họa ở ví dụ trên có chiều cao là 10.

2.4. YÊU CẦU KHÔNG GIAN VÀ CẤU TRÚC NÚT [6]

2.5. CHÈN MỘT PHẦN TỬ VÀO TRIE

Để chèn một phần tử A với khóa là K vào trong một Trie thì việc đầu tiên ta phải tiến hành tìm kiếm trên Trie xem đã có tồn tại phần tử với khóa K này chưa. Nếu trie đã chứa phần tử với khóa đó, ta tiến hành thay thế phần tử hiện hành với phần tử A. Nếu Trie không chứa phần tử A với khóa K, thì phần tử A sẽ được đưa vào Trie bằng cách sử dụng các thủ tục sau:

Trường hợp 1, thủ tục chèn nếu kết quả tìm kiếm cho khóa K kết thúc tại một nút lá X bất kỳ, thì sau đó, khóa của phần tử tại X và khóa K sẽ được sử dụng để xây dựng một nhánh con mới thay thế cho phần tử X.

Trong trường hợp 2, thủ tục chèn khi kết quả tìm kiếm phần tử A với khóa K trong Trie kết thúc bởi việc duyệt hết trie tại một nút nhánh X bất kỳ nào đó và không tìm thấy kết quả. Lúc này chúng ta chỉ thực hiện một thao tác đơn giản là thêm vào một nút con (là một nút lá) tại vị trí nút X. Nút lá được thêm vào chính là phần tử A với khóa K.

2.6. LOẠI BỎ MỘT PHẦN TỬ KHỎI TRIE

Để loại bỏ một phần tử A với khóa K, việc đầu tiên chúng ta phải tìm kiếm phần tử với khóa K này trong Trie. Nếu không có phần tử với khóa K trong Trie thì mọi việc không có gì để thực hiện. Vì thế, ta giả định rằng Trie chúng ta đang thao tác có chứa phần tử A với khóa K. Các nút lá X được tìm thấy chứa phần tử A sẽ bị loại bỏ và chúng ta xét duyệt lại các nút nhánh trên đường dẫn từ nút lá X quay về nút gốc của Trie. Sẽ có một số nút nhánh bị loại bỏ nếu chúng chỉ chứa duy nhất một phần tử trong đó. Việc này được lặp lại cho đến khi chúng ta gặp một nút nhánh không bị loại bỏ hoặc cho đến khi chúng ta đã tiến hành loại bỏ cả nút gốc của Trie. Chúng ta hãy xem ví dụ minh họa tại hình 2.4

2.7. TRIE NÉN VÀ CÁC BIẾN THỂ

Chúng ta quan sát trie của hình 2.2. Trong Trie này, có một vài nút nhánh (đó là B, D, F), các nút này chỉ có một nhánh duy nhất. Chúng ta có thể cải thiện thời gian và không gian lưu trữ của Trie bằng cách loại bỏ tất cả các nút nhánh mà chỉ có duy nhất một nhánh con này. Trie kết quả thu được từ thao tác này được gọi là Trie nén.

Khi các nút nhánh chỉ có một con thì chúng sẽ được di chuyển khỏi Trie. Ta cần lưu trữ lại tất cả những thông tin này để đảm bảo việc tổ chức từ điển được chính xác. Các thông tin này được lưu trữ trong ba loại cấu trúc trie nén được mô tả dưới đây.

2.7.1. Trie nén kiểu số

2.7.2. Trie nén lượt bỏ

2.7.3. Trie nén với thông tin cạnh

2.7.4. Yêu cầu không gian của Trie nén

2.8. TÌM KIẾM TIỀN TỐ VÀ MỘT SỐ ỨNG DỤNG

2.9. KẾT LUẬN

Trong chương này, tôi đã nghiên cứu và tìm hiểu các vấn đề lý thuyết liên quan đến cấu trúc dữ liệu Trie khá cụ thể bao gồm những kiến thức chung tổng quát về cấu trúc Trie và những thao tác cơ bản trên cấu trúc dữ liệu này. Qua việc đánh giá và so sánh với các cấu trúc dữ liệu trước, các phương pháp lập chỉ mục đã được sử dụng, luận văn tìm ra được một số điểm hạn chế của cấu trúc này và đề xuất các biến thể nén của trie.

Qua chương này, người đọc có thể có được kiến thức cơ bản nhất về Trie, hướng dẫn từng bước việc thực hiện các thao tác khi sử dụng cấu trúc này, đánh giá hiệu suất của cấu trúc nói chung và các thao tác nói riêng.

CHƯƠNG 3: ỨNG DỤNG TRIE TÌM KIẾM TRONG MARIADB

Trong nội dung chương 2, chúng ta đã tìm hiểu sơ lược về cấu trúc Trie và những thao tác cơ bản trên Trie. Cấu trúc này hiện đang được ứng dụng ngày càng nhiều trên toàn thế giới, đặc biệt trong việc lưu trữ và xử lý với các kho dữ liệu lớn. Hệ quản trị cơ sở dữ liệu mã nguồn mở ngày càng được nhiều người lựa chọn do nhờ "tính mở" cho người dùng. Trong số đó, không thể không kể đến MySQL với cộng đồng người sử dụng rộng khắp và những tính năng hỗ trợ ưu việt. Từ khi người sáng lập ra MySQL rời bỏ Sun, cộng đồng người sử dụng MySQL trên toàn thế giới bắt đầu tiếp cận với MariaDB, một nhánh rẽ của MySQL với những tính năng kế thừa hoàn toàn của MySQL và được tích hợp thêm nhiều tính năng mới nhằm khắc phục những hạn chế của các phiên bản MySQL trước đó.

MariaDB cũng là một hệ cơ sở dữ liệu mã nguồn mở hoàn toàn miễn phí cho người dùng, ngoài những cấu trúc dữ liệu được bố trí như với MySQL, MariaDB còn kết hợp thêm nhiều cấu trúc mới nhằm tối ưu hóa quá trình truy vấn dữ liệu để phù hợp với những kho dữ liệu lớn. Một trong các cấu trúc được sử dụng là Trie (đã trình bày ở chương 2).

3.1. MÔ TẢ ỨNG DỤNG

Để làm rõ hơn cho những nội dung liên quan đến Trie đã được trình bày trong chương trước, trong chương này, chúng ta sẽ tiến hành nghiên cứu ứng dụng cấu trúc khá mới này(cấu trúc Trie) trong các truy vấn của hệ cơ sở dữ liệu MariaDB. Chọn MariaDB cho việc ứng dụng Trie vì đây là một hệ cơ sở dữ liệu mã nguồn mở hoàn toàn miễn phí, đang được sử dụng rộng rãi và dần thay cho MySQL vốn đã chiếm được rất nhiều tình cảm của cộng đồng người sử dụng trên toàn thế giới. Ngoài ra, để khắc phục một số lỗi hạn chế trong quá trình sử dụng MySQL. MariaDB đã có rất nhiều cải tiến

mới trong tính năng hỗ trợ, tốc độ và khả năng truy vấn dữ liệu mà một trong những phát triển ghi nhận là sử dụng tích hợp cấu trúc dữ liệu Trie hỗ trợ full-text-search.

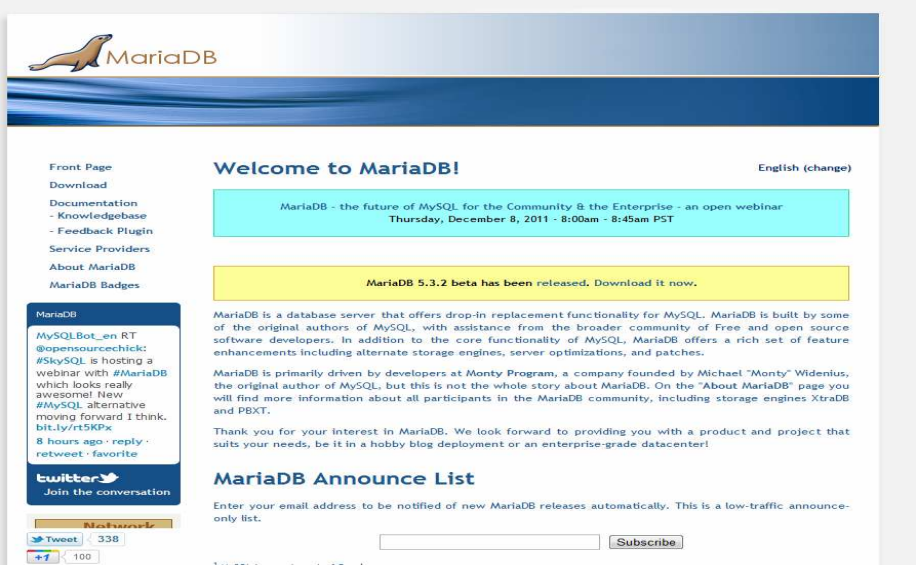
Theo đó, nội dung sau đây sẽ trình bày những kiến thức cơ bản về hệ cơ sở dữ liệu MariaDB, cách cài đặt và lấy mã nguồn từ MariaDB trên môi trường Ubuntu. Để làm rõ hơn cho cấu trúc Trie đã minh họa trong chương 2, sau khi cài đặt thành công, dựa trên việc nghiên cứu mã nguồn của MariaDB, chúng ta sẽ xác định cấu trúc Trie được ứng dụng trong MariaDB như thế nào và cài đặt ra sao để tiến hành tối ưu hóa các thao tác truy vấn ngay trên hệ cơ sở dữ liệu này. Qua đó, chúng ta sẽ biết được cấu trúc Trie được cài đặt như thế nào trong thực tiễn.

3.2. MARIADB [7]

3.2.1. Giới thiệu chung [7]

MariaDB, một nhánh rẽ của MySQL, là một máy chủ cơ sở dữ liệu cung cấp các chức năng thay thế cho MySQL. MariaDB được xây dựng bởi một số các tác giả ban đầu của MySQL với sự hỗ trợ từ cộng đồng các nhà phát triển phần mềm miễn phí và mã nguồn mở. Ngoài các chức năng cốt lõi của MySQL, MariaDB cung cấp một tập hợp phong phú các tính năng cải tiến bao gồm công cụ lưu trữ thay thế, tối ưu hóa máy chủ và các bản vá lỗi.

Phiên bản MariaDB được tung ra hồi tháng 11/2008 bởi Monty Widenius, người đồng sáng lập ra MySQL. Widenius đã công bố sự phát triển của rẽ nhánh MariaDB ngay sau khi rời bỏ chức vụ là người duy trì cho MySQL của Sun. Cùng lúc nhà lập trình này đã sáng lập ra Monty Program AB, một công ty mới để đưa rẽ nhánh này ra thị trường.



Hình 3.1. Trang chủ MariaDB

MariaDB là một nhánh của MySQL, một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới. Từ các dự án nhỏ phát triển trên Web cho một số trang Web nổi tiếng và uy tín, MySQL đã tự chứng minh bản thân một cách vững chắc, đang tin cậy, nhanh chóng và thật sự là một giải pháp hữu hiệu cho việc sắp xếp và lưu trữ dữ liệu.

3.2.2. Các phiên bản phát triển

3.3. TRIE TÌM KIẾM TRONG MARIADB

3.4. CÀI ĐẶT THỬ NGHIỆM

3.4.1. Cài đặt

3.4.1.1. Thu thập mã nguồn

Như đã giới thiệu, MariaDB là một phiên bản được rẽ nhánh từ MySQL, được sáng lập bởi những người một thời là tác giả của MySQL. Người sáng lập hàng đầu là Monty Widenius_người đã sáng lập ra MySQL và Monty Program AB

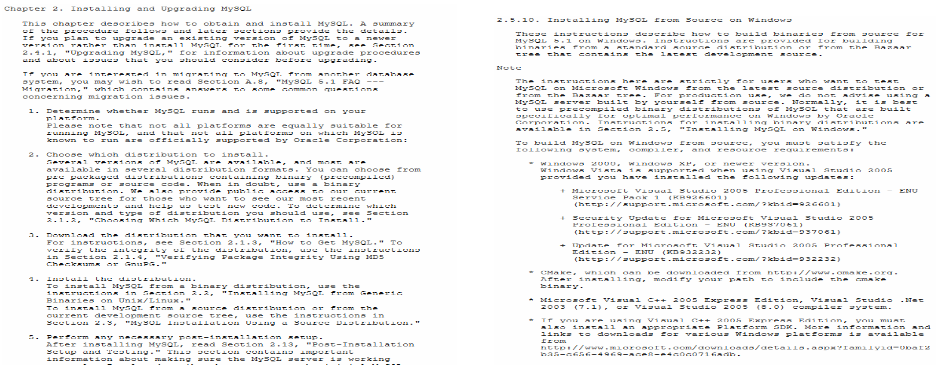
Người dùng ở khắp mọi nơi trên thế giới có thể truy cập vào địa chỉ http://mariadb.org/ để tìm hiểu, học hỏi và tải về bộ cài đặt cũng như mã nguồn của MariaDB cùng với tất cả các phiên bản được phát hành.

Ngoài chức năng hỗ trợ download cái gói cài đặt khác nhau trên các môi trường khác nhau, http://mariadb.org/ còn có nhiều hỗ trợ khác cho người dùng trong quá trình cài đặt và tiếp cận với việc sử dụng, khai thác mã nguồn trên MariaDB.

Hình 3.7. Trang download mariadb

Ở đây, người dùng có thể lựa chọn các phiên bản phát triển phù hợp với yêu cầu của họ, các hệ điều hành hỗ trợ Generic Linux, Linux Package, Solaris, Source code, Windows. Các gói hỗ trợ gồm source tar.gz file, gzipped tar file, MSI Package, Zip file và RPM Package; CPU 64-bit hoặc 32-bit.

Khi đã lựa chọn gói download với hệ điều hành phù hợp, chúng ta tiến hành cài đặt MariaDB, có thể tham khảo mã nguồn để phát triển. Các phiên bản từ được phát triển từ 5.1 đến 5.3 cũng có sự hỗ trợ tương ứng như nhau. Cách cài đặt tương tự như khi sử dụng MySQL, người dùng có thể tham khảo tại file Install-source hoặc install-win-source trong gói được lựa chọn tải về



```
Chapter 2. Installing and Upgrading MySQL

   This chapter describes how to obtain and install MySQL. A summary
   of the procedure follows and later sections provide the details.
   If you plan to upgrade an existing version of MySQL to a newer
   version rather than install MySQL for the first time, see Section
   2.4.1, "Upgrading MySQL," for information about upgrade procedures
   and about issues that you should consider before upgrading.

   If you are interested in migrating to MySQL from another database
   system, you may wish to read Section A.8, "MySQL 5.1 FAQ ---
   Migration," which contains answers to some common questions
   concerning migration issues.

    1. Determine whether MySQL runs and is supported on your
       platform.
       Please note that not all platforms are equally suitable for
       running MySQL, and that not all platforms on which MySQL is
       known to run are officially supported by Oracle Corporation:

    2. Choose which distribution to install.
       Several versions of MySQL are available, and most are
       available in several distribution formats. You can choose from
       pre-packaged distributions containing binary (precompiled)
       programs or source code. When in doubt, use a binary
       distribution. We also provide public access to our current
       source tree for those who want to see our most recent
       developments and help us test new code. To determine which
       version and type of distribution you should use, see Section
       2.1.2, "Choosing Which MySQL Distribution to Install."

    3. Download the distribution that you want to install.
       For instructions, see Section 2.1.3, "How to Get MySQL." To
       verify the integrity of the distribution, use the instructions
       in Section 2.1.4, "Verifying Package Integrity Using MD5
       Checksums or GnuPG."

    4. Install the distribution.
       To install MySQL from a binary distribution, use the
       instructions in Section 2.2, "Installing MySQL from Generic
       Binaries on Unix/Linux."
       To install MySQL from a source distribution or from the
       current development source tree, use the instructions in
       Section 2.3, "MySQL Installation Using a Source Distribution."

    5. Perform any necessary post-installation setup.
       After installing MySQL, read Section 2.13, "Post-Installation
       Setup and Testing." This section contains important
       information about making sure the MySQL server is working
```

```
2.5.10. Installing MySQL from Source on Windows

   These instructions describe how to build binaries from source for
   MySQL 5.1 on Windows. Instructions are provided for building
   binaries from a standard source distribution or from the Bazaar
   tree that contains the latest development source.

Note

   The instructions here are strictly for users who want to test
   MySQL on Microsoft Windows from the latest source distribution or
   from the Bazaar tree. For production use, we do not advise using a
   MySQL server built by yourself from source. Normally, it is best
   to use precompiled binary distributions of MySQL that are built
   specifically for optimal performance on Windows by Oracle
   Corporation. Instructions for installing binary distributions are
   available in Section 2.5, "Installing MySQL on Windows."

   To build MySQL on Windows from source, you must satisfy the
   following system, compiler, and resource requirements:

     * Windows 2000, Windows XP, or newer version.
       Windows Vista is supported when using Visual Studio 2005
       provided you have installed the following updates:

          + Microsoft Visual Studio 2005 Professional Edition - ENU
            Service Pack 1 (KB926601)
            (http://support.microsoft.com/?kbid=926601)

          + Security Update for Microsoft Visual Studio 2005
            Professional Edition - ENU (KB937061)
            (http://support.microsoft.com/?kbid=937061)

          + Update for Microsoft Visual Studio 2005 Professional
            Edition - ENU (KB932232)
            (http://support.microsoft.com/?kbid=932232)

     * CMake, which can be downloaded from http://www.cmake.org.
       After installing, modify your path to include the cmake
       binary.

     * Microsoft Visual C++ 2005 Express Edition, Visual Studio .Net
       2003 (7.1), or Visual Studio 2005 (8.0) compiler system.

     * If you are using Visual C++ 2005 Express Edition, you must
       also install an appropriate Platform SDK. More information and
       links to downloads for various Windows platforms is available
       from
       http://www.microsoft.com/downloads/details.aspx?familyid=0baf2
       b35-c656-4969-ace8-e4c0c0716adb.
```

Hình 3.8. File Install-Source và Install-Win-Source

3.4.1.2. Cài đặt thực thi

Phiên bản MariaDB mới nhất là 5.3 hiện đã có bản beta phát hành vào tháng 7/2011, tuy nhiên chạy ổn định và nhiều tính năng ưu việt, người dùng có thể tải bản mới nhất này tại http://downloads.askmonty.org/mariadb/5.3/ hoặc tải các phiên bản cũ là MariaDB 5.2 tại http://downloads.askmonty.org/mariadb/5.2/ hay phiên bản đầu tiên 5.1 tại http://downloads.askmonty.org/mariadb/5.1/.

File Name	Package Type	OS / CPU	Size	Meta
mariadb-5.2.9.tar.gz	source tar.gz file	Source	25.1 MB	[MD5] Instructions
mariadb-5.2.9-winx64.msi	MSI Package	Windows 64-bit	95.3 MB	Instructions
mariadb-5.2.9-win32.msi	MSI Package	Windows 32-bit	95.8 MB	Instructions
mariadb-5.2.9-win32.zip	ZIP file	Windows 32-bit	152.9 MB	Instructions
mariadb-5.2.9-winx64.zip	ZIP file	Windows 64-bit	152.5 MB	Instructions
mariadb-5.2.9-solaris11-i386.tar.gz	gzipped tar file	Solaris 11 32-bit	65.7 MB	[MD5] Instructions
mariadb-5.2.9-solaris10-i386.tar.gz	gzipped tar file	Solaris 10 32-bit	65.7 MB	[MD5] Instructions
mariadb-5.2.9-Linux-x86_64.tar.gz	gzipped tar file	Linux 64-bit	158.3 MB	[MD5] Instructions
mariadb-5.2.9-Linux-i686.tar.gz	gzipped tar file	Linux 32-bit	151.3 MB	[MD5] Instructions
CentOS 5 amd64 RPMs	RPM Package	CentOS 5 64-bit		Instructions
CentOS 5 x86 RPMs	RPM Package	CentOS 5 32-bit		Instructions
Red Hat Enterprise Linux 5 amd64 RPMs	RPM Package	Red Hat Enterprise Linux 5 64-bit		Instructions
Red Hat Enterprise Linux 5 x86 RPMs	RPM Package	Red Hat Enterprise Linux 5 32-bit		Instructions

Hình 3.9. Các gói cài đặt của phiên bản MariaDB 5.2

Trong mỗi phiên bản đều có nhiều gói để cài đặt, bạn cần chọn lựa gói cài đặt nào phù hợp với yêu cầu của mình

.MariaDB chủ yếu được khai thác trên các hệ điều hành mã nguồn mở, bản thân nó cũng là mã nguồn mở nên không được cài đặt theo kiểu Wizard. Cài Mariadb trên Ubuntu có thể dùng nhóm lệnh sau:

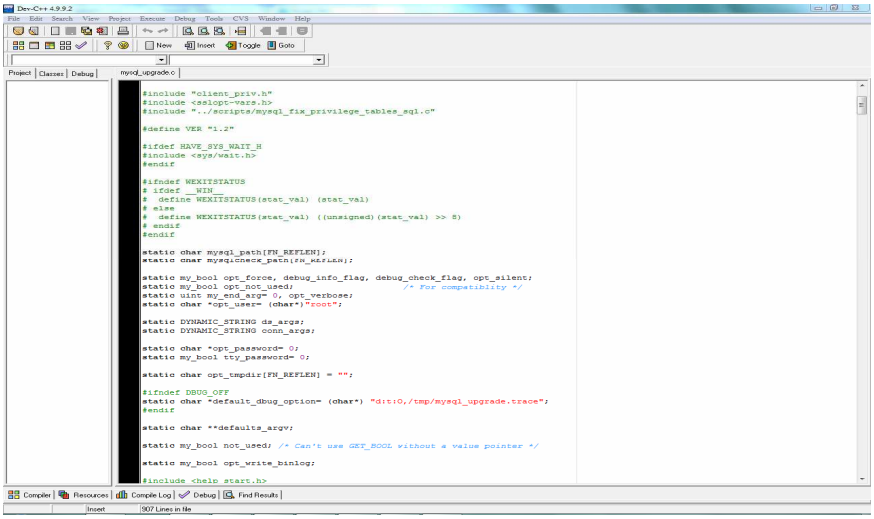
```
shell> groupadd mysql

shell> useradd -g mysql mysql

shell> cd /usr/local

shell> gunzip < /path/to/mysql-VERSION-OS.tar.gz | tar xvf -

shell> ln -s full-path-to-mysql-VERSION-OS mysql

shell> cd mysql
```

```
shell> chown -R mysql .

shell> chgrp -R mysql .

shell> scripts/mysql_install_db --user=mysql

shell> chown -R root .

shell> chown -R mysql data

shell> bin/mysqld_safe --user=mysql &
```

Và tiến hành thay thế phiên bản Mariadb bạn đang sử dụng trong câu lệnh cho phù hợp. Sau khi hoàn tất việc cài đặt, chúng ta có thể can thiệp vào mã nguồn đê nghiên cứu, trích lọc và phát triển chương trình riêng. Với các cấu trúc dữ liệu được tổ chức sử dụng, người dùng sẽ đọc và trích lọc để lựa chọn những đoạn mã cần cho quá trình làm việc của mình.

Chúng ta sẽ tiến hành việc download và cài đặt thử ngiệm đối với phiên bản MariaDB 5.2. Sử dụng máy ảo chạy hệ điều hành Ubuntu phiên bản 10.10 để download và cài đặt như sau:

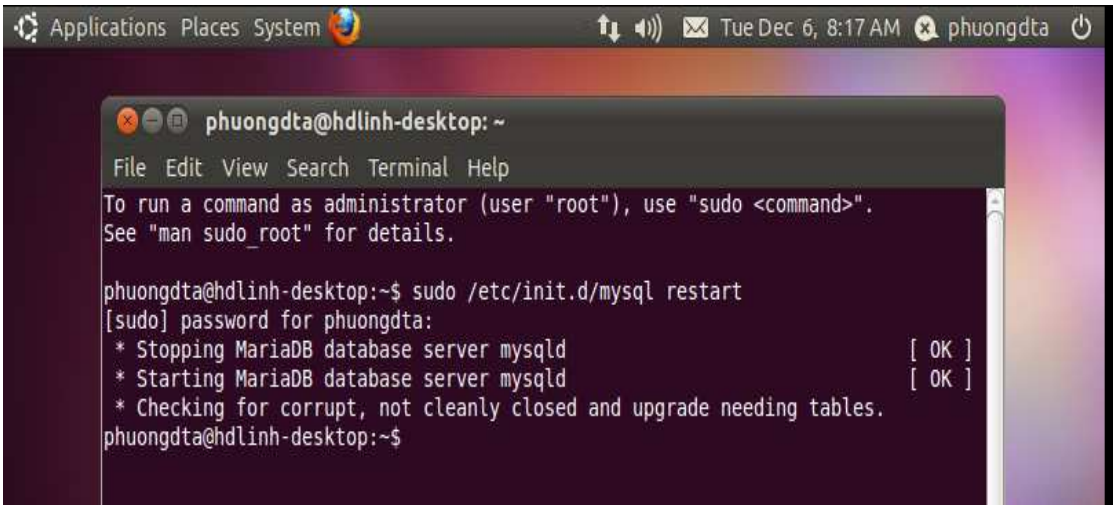
- Dùng lệnh chạy lấy khóa cần thiết (apt –key)
- Thêm lệnh yêu cầu của hệ thống vào file Source.list
- Cập nhập các gói
- Tiến hành cài đặt MariaDB trên ubuntu
- Cài các gói hỗ trợ và khai thác



Hình 3.10. Mã nguồn được mở bằng Dev-C

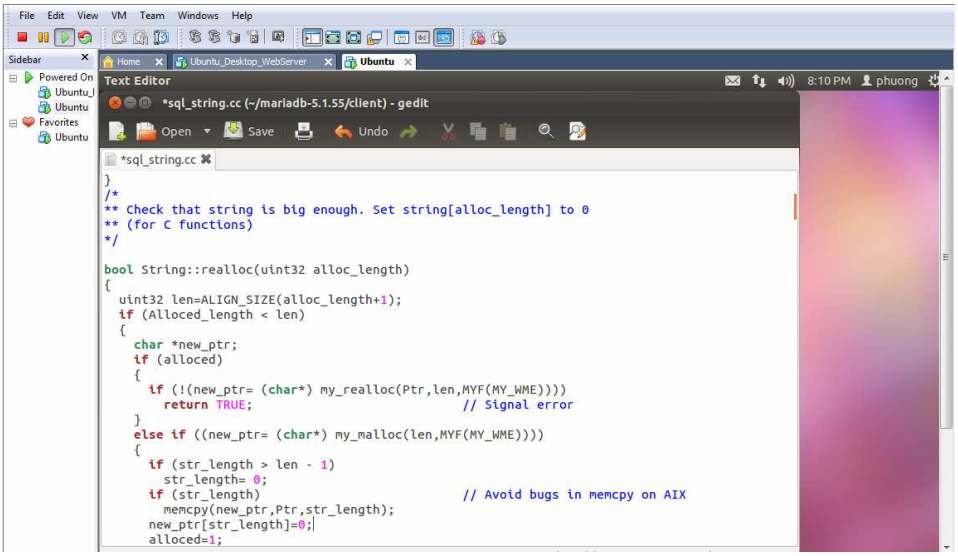
3.4.2. Thử nghiệm

Để thử nghệm, chúng ta tiến hành cài đặt trên hệ điều hành Ubuntu 10.10, mariaDB 5.2. Cấu trúc dữ liệu Trie ở đây được sử dụng như một Trie index và tích hợp trong chức năng Full-Text-Search. Sau khi cài đặt, khởi động: *sudo /etc/init.d/mysql restart,* Màn hình thành công nếu có lệnh báo



Hình 3.11. Khởi động MariaDB

Trong phạm vi luận văn, chỉ dừng lại ở thử nghiệm tìm kiếm một chuỗi trong Trie trong MariaDB, cấu trúc Trie được xây dựng lưu trữ bằng một danh sách các nút liên kết với nhau, chúng ta tiến hành tìm kiếm một chuỗi trong danh sách các nút này.



Hình 3.12. Mã nguồn MariaDB được sử dụng sau khi cài đặt thành công

Sau khi cài đặt thành công và can thiệp vào mã nguồn của MariaDB, ta tiến hành tìm hiểu việc ứng dụng cấu trúc dữ liệu Trie trong MariaDB. Trong hệ quản trị cơ sở dữ liệu MariaDB, cấu trúc dữ liệu Trie được sử dụng tích hợp trong chức năng full-text-search. Việc ứng dụng Trie cơ bản mô tả như sau:

Cấu trúc của node

```
/* Cấu trúc cua node */
typedef struct node
{
  char *word;
  struct node *next;
}node;
```

Khởi tạo Trie

```
int init()
{
  char **c_trie=NULL;
  int i=0;
```

Phân bổ một mảng con trỏ sẽ sử dụng

```
trie_pack = (char **) calloc (trie_pack_capacity, sizeof(char *));
trie_pack_idx=0;
trie_counter=0;
```

Phân bổ 1 nút mới và đánh dấu đó là nút gốc của Trie

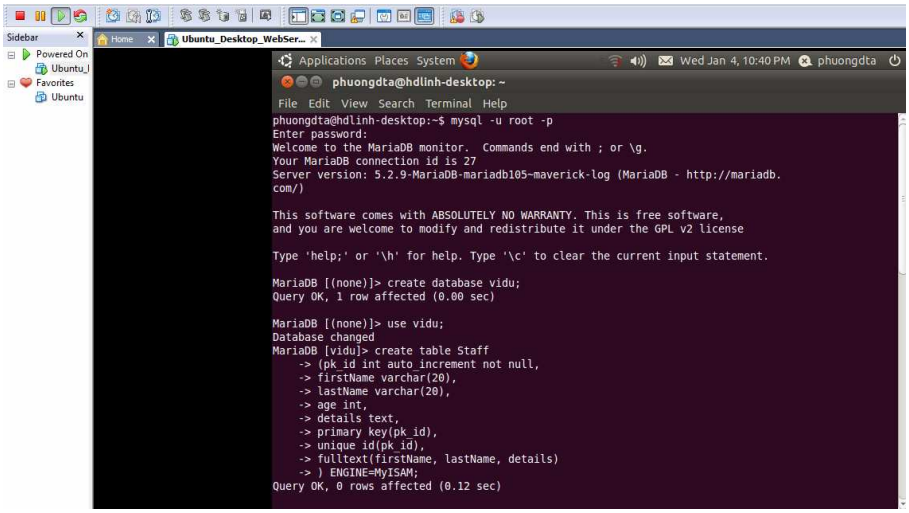
```
*(trie_pack+trie_pack_idx) = calloc(trie_pack_entry_capacity*TRIE_SIZE,
sizeof(char));
root_trie=new_trie();
c_trie = (char **)root_trie;
```

So sánh, đối chiếu chuỗi đang truy vấn và chuỗi tại nút, trả về 1 nếu tìm thấy, ngược lại trả về 0.

```
for (; *query == *array && *query != '\0'; query++, array++);
    if ( *query == '\0' && *array == '\0')
{
   if(previous!=NULL)
   {
      previous->next=traverse->next;
      traverse->next=header->head;
      header->head=traverse;
  mtf_counter++;
   }
   return 1;
}
```

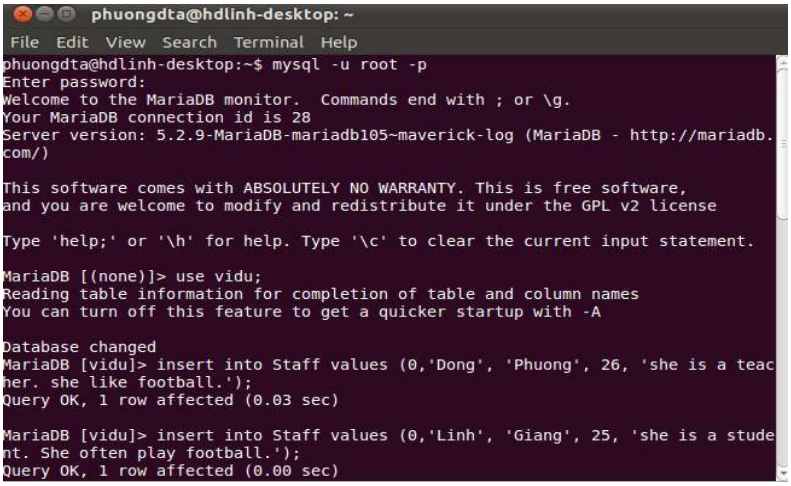
Với cấu trúc được xây dựng cơ bản như trình bày trên, Trie hỗ trợ chức năng full-text-search trong MariaDB, phát huy tên tuổi MySQL được đông đảo người sử dụng trên thế giới sử dụng.

Kết quả thử nghiệm trên MariaDB, với dữ liệu cụ thể. Tạo cơ sở dữ liệu "vidu" với bảng "staff". Staff bao gồm các trường: pk_id, firstname, lastname, age, details.



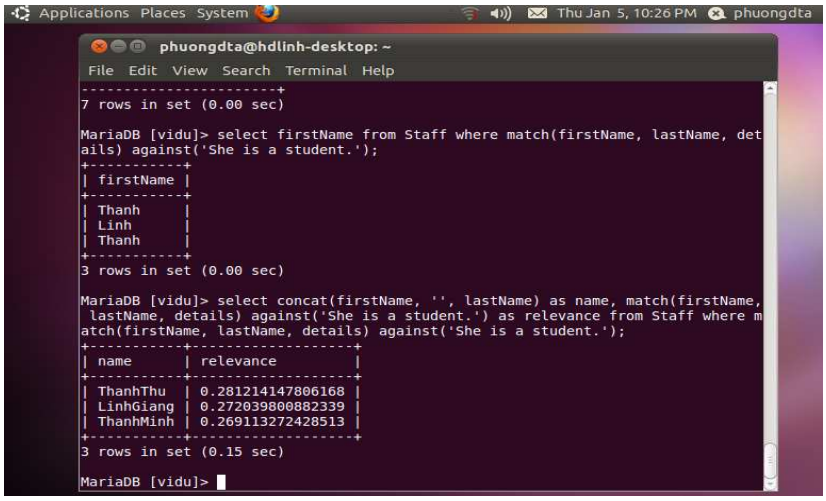
Hình 3.13. Tạo cơ sở dữ liệu và bảng dữ liệu

Đánh chỉ mục, nhập dữ liệu cho bảng. Dữ liệu này sẽ được dùng làm mẫu tìm kiếm.



Hình 3.14. Nhập dữ liệu cho bảng

Sau khi hoàn tất việc nhập dữ liệu ta tiến hành thao tác truy vấn trên bảng dữ liệu vừa nhập.



Hình 3.15. Kết quả truy vấn

3.5. ĐÁNH GIÁ

3.5.1. Kết quả đạt được

MariaDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở có nhiều phiên bản, hỗ trợ nhiều hệ điều hành với nhiều gói cài đặt khác nhau nên phù hợp cho nhiều đối tượng sử dụng.

MariaDB có hiệu suất khá cao khi sử dụng, vấn đề này người đọc có thể tham khảo ý kiến của cộng đồng MySQL và MariaDB trên toàn thế giới. Chương này đã hướng dẫn sơ lược cách tải và cài đặt MariaDB để nghiên cứu ứng dụng của cấu trúc dữ liệu Trie trong hệ quản trị cơ sở dữ liệu này. Người đọc có thể dựa vào các thông tin tại đây để lấy môi trường, công cụ phát triển làm nền nghiên cứu cho ứng dụng thực tế của cấu trúc dữ liệu Trie.

3.5.2. Ưu điểm và hạn chế

MariaDB là một hệ quản trị mã nguồn mở nên tất cả các phiên bản cài đặt và mã của nó đều miễn phí, cùng với MySQL, MariaDB đang có ảnh hưởng lớn trên thế giới. Việc MariaDB đưa vào sử dụng kết hợp cấu trúc dữ liệu Trie cùng với các cấu trúc dữ liệu khác và dùng các phương pháp lập chỉ mục cũ và mới kết hợp là một dấu hiệu đánh dấu việc khẳng định cấu trúc dữ liệu này trong thời gian sắp tới trên thị trường công nghệ.

Tuy nhiên, cũng chính vì sự mới lạ này mà cả MariaDB và cấu trúc dữ liệu Trie đều có những hạn chế nhất định. MariaDb và Trie hầu như còn khá xa lạ và khá ít tài liệu hỗ trợ như những công cụ, cấu trúc khác. Trong MariaDB chỉ mới đưa Trie vào sử dụng khá ít và sử dụng kết hợp với các cấu trúc khác nên người dùng muốn nghiên cứu cấu trúc Trie dựa trên MariaDB phải trích lọc và xây dựng lại hầu như phần lớn.

3.5.3. Kết luận

Nội dung chương đã cài đặt được MariaDB cùng đoạn chương trình được rút trích và hoàn thiện để mô tả cụ thể cho cấu trúc Trie được sử dụng trong MariaDB. Một cấu trúc dữ liệu còn khá mới và chưa được sử dụng đại trà hiện nay; Ngoài ra việc chọn MariaDB để minh họa cho việc sử dụng câu trúc này cũng khá mới vì MariaDB là một lựa chọn tiếp theo cho MySQL và còn khá trẻ tuổi (phiên bản beta mới nhất vừa được phát hành vào tháng 7/2011). Với sự mới mẻ này cùng thời gian và phạm vi hạn chế, hy vọng trong thời gian tới tác giả sẽ phát triển nghiên cứu sâu hơn về cấu trúc Trie và phát triển ứng dụng thực tế cho cấu trúc này.

KẾT LUẬN VÀ KIẾN NGHỊ

Trong luận văn, tôi đã tìm hiểu và nghiên cứu được cấu trúc dữ liệu Trie, một số thao tác cơ bản trên Trie và đặc biệt là cách nén Trie nhằm tăng hiệu quả sử dụng cấu trúc này cùng với các thao tác trên Trie đã được nén như: tìm kiếm, chèn, xóa, sửa,…Bên cạnh đó, luận văn còn tìm hiểu về các kiến thức liên quan đến tìm kiếm thông tin trên văn bản, các phương pháp lập chỉ mục và các cấu trúc dữ liệu đã từng được sử dụng trước đây, so sánh các cấu trúc với nhau, tìm ra những mặt tích cực và những hạn chế.

Trong quá trình thực hiện, tôi đã hiểu cơ bản về cấu trúc dữ liệu này; hiện nay, cấu trúc này chưa được phổ biến ở Việt nam nhưng nó có một số tính năng thật sự hiệu quả không thể phủ nhận, vì thế, luận văn sẽ là tài liệu khá bổ ích cho các bạn sinh viên nói riêng và những người nghiên cứu về công nghệ thông tin nói chung có thêm tài liệu để tham khảo ở bước ban đầu khi tiến hành nghiên cứu, học tập và triển khai xây dựng ứng dụng thực tế trong trường hợp muốn tiếp cận và đi sâu khai thác Trie.

Do thời gian, phạm vi của luận văn và kiến thức cá nhân còn giới hạn cộng thêm cấu trúc dữ liệu này còn khá mới và tuổi của MariaDB còn quá trẻ nên luận văn chỉ dừng lại ở mức tìm hiểu kỹ về mặt lý thuyết liên quan đến cấu trúc TRIE và các thao tác trên Trie; Tìm hiểu cách sử dụng TRIE trong hệ quản trị cơ sở dữ liệu mã nguồn mở MariaDB để chứng minh rằng cấu trúc này hiện đang được sử dụng và ngày càng phổ biến ở nhiều quốc gia trên thế giới. Tuy nhiên, luận văn chưa thể thực nghiệm trực tiếp để so sánh Trie với các câu trúc khác trên MariaDB, đây là một điểm hạn chế và là hướng phát triển trong thời gian sắp đến với mong muốn có được một tài liệu hoàn chỉnh về Trie để cung cấp cho người sử dụng.

Từ những hạn chế và thiếu sót trên, để luận văn được tiếp tục phát triển theo định hướng rất mong nhận được sự đóng góp của các nhà phát triển trong lĩnh vực tìm kiếm thông tin, sự trợ giúp về nguồn thông tin để hoàn thiện phần ứng dụng hệ thống vào thực tế sử dụng phục vụ cho nhu cầu hàng triệu người sử dụng trong nước.