Chương 5

TỐI ƯU HÓA CÂU TRUY VẤN

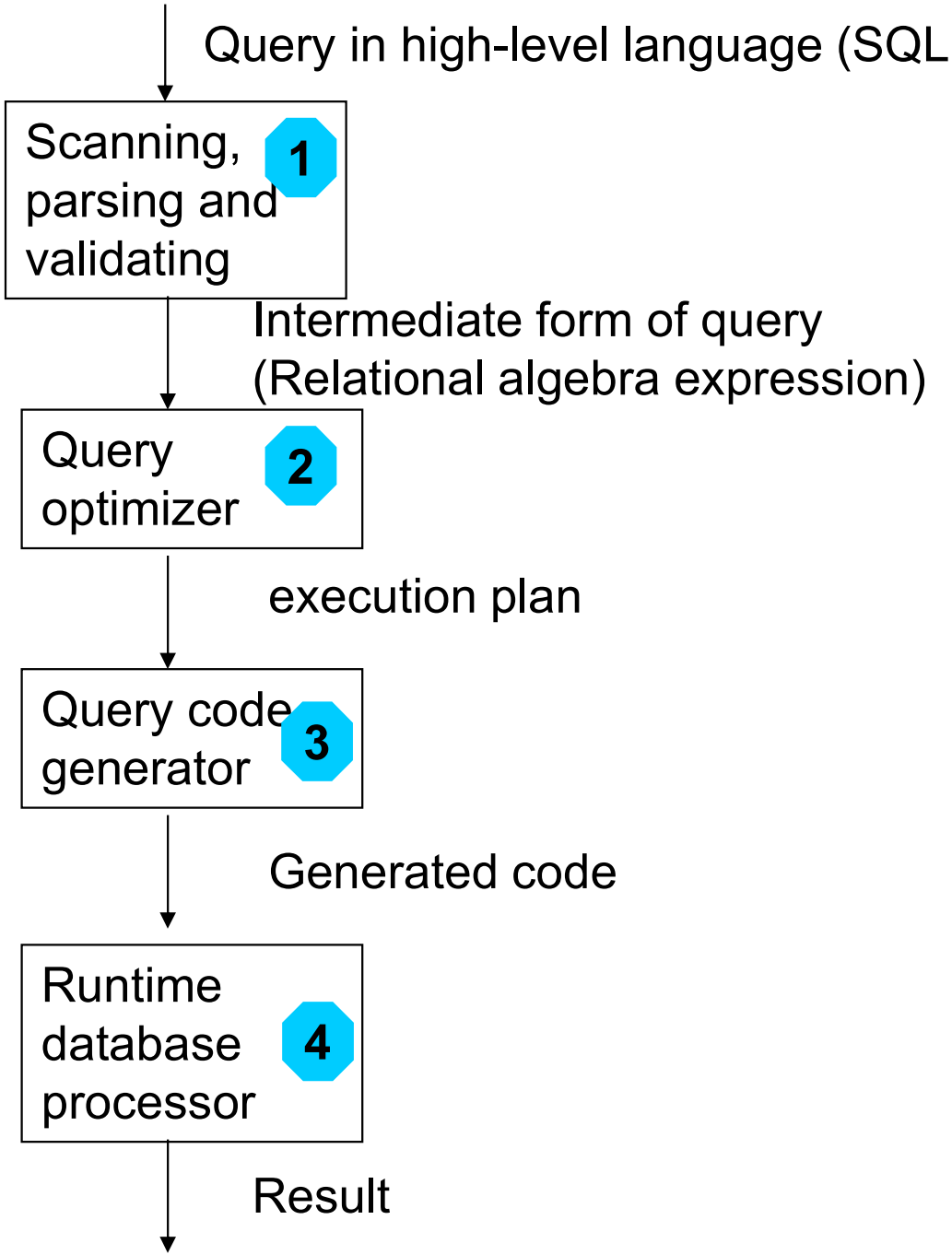
Mục đích

- Tối ưu hóa vấn tin là tiến trình lựa chọn kế hoạch thực thi câu vấn tin một cách hiệu quả nhất.
  - Tốn ít tài nguyên nhất.
  - Hồi đáp nhanh nhất.

Nội dung

1. Tổng quan về xử lý truy vấn
2. Tối ưu hóa truy vấn dùng Heuristics
3. Tối ưu hóa truy vấn dùng phương pháp ước lượng chi phí

Các bước xử lý vấn tin



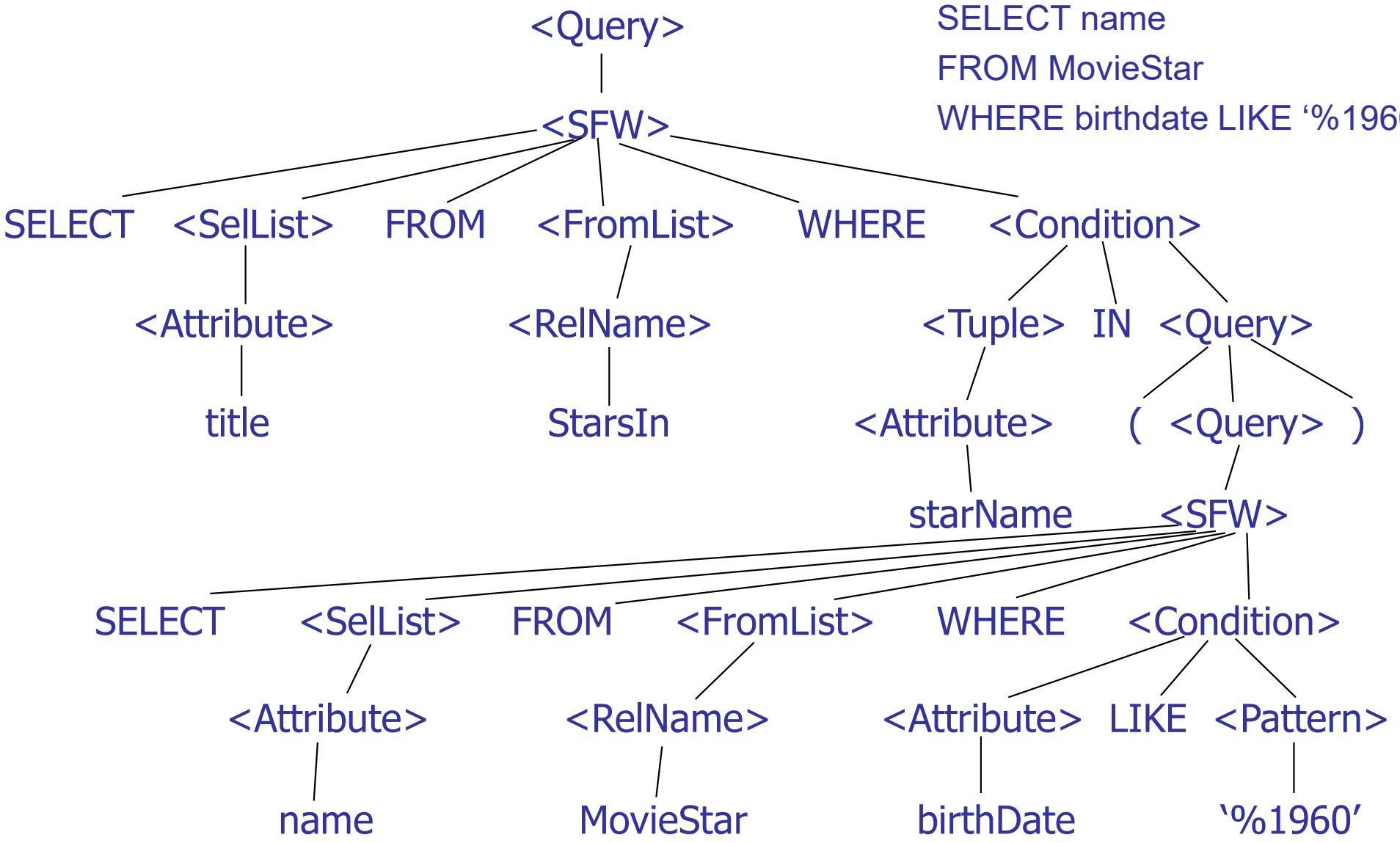
Các bước xử lý vấn tin

- Bước 1
  - Scan
    - Xác định các từ khóa của ngôn ngữ SQL, tên thuộc tính, tên quan hệ.
  - Parse
    - Kiểm tra cú pháp câu truy vấn.
  - Validate
    - Kiểm tra tên thuộc tính, tên quan hệ có trong lược đồ đã khai báo hay không.
    - Không nhập nhằng khi dùng các thuộc tính.
    - Kiểu dữ liệu dùng để so sánh đều hợp lệ.
  - Thể hiện lại câu truy vấn: đại số quan hệ, query tree, query graph.

Parse tree

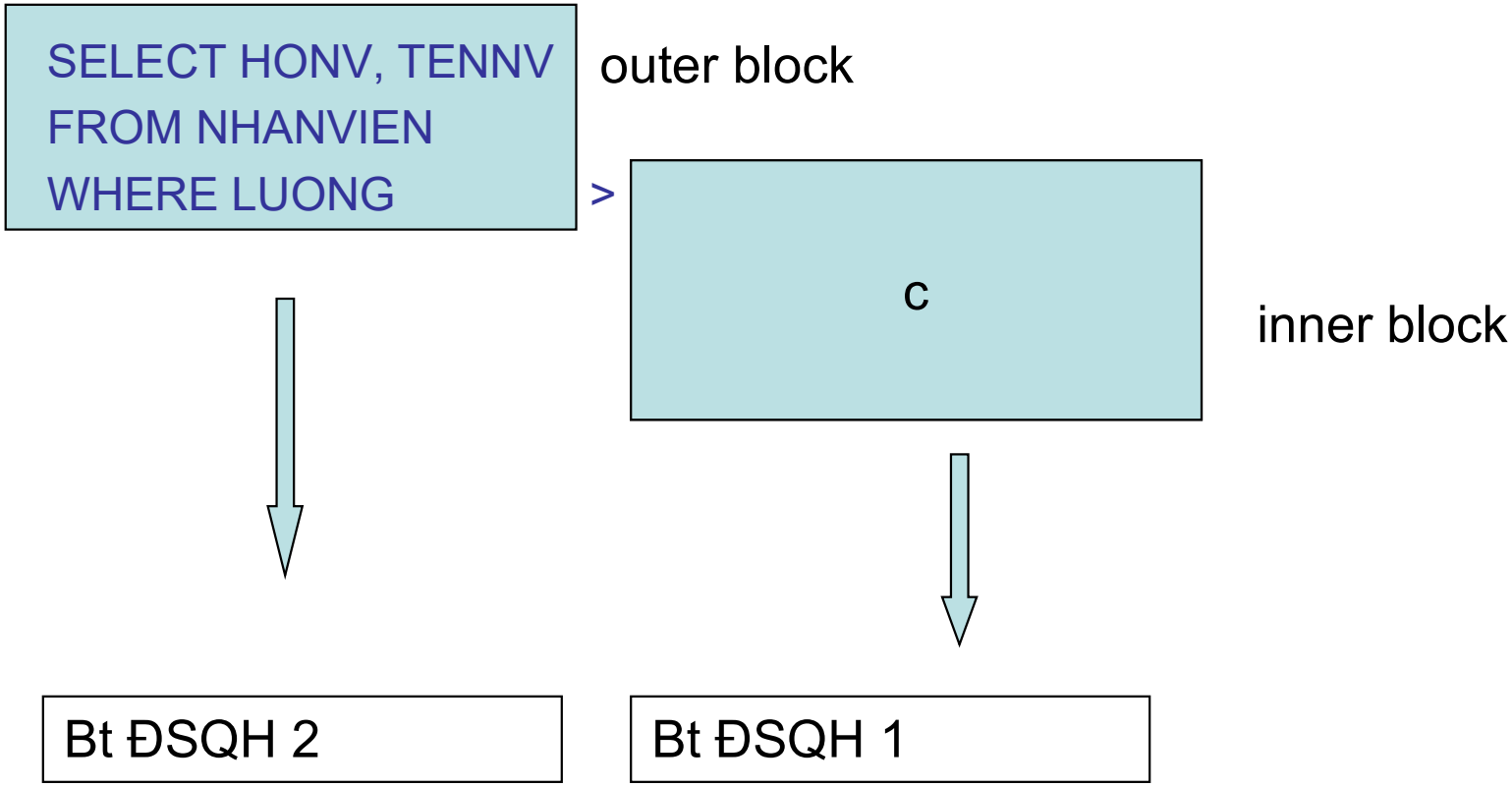
Tìm các bộ phim mà diễn viên sinh vào năm 1960

```
SELECT title
FROM StarsIn
WHERE starName IN (
        SELECT name
        FROM MovieStar
        WHERE birthdate LIKE '%1960');
```



Chuyển Q thành ĐSQH

- Câu truy vấn được phân rã thành các **query block** (QB).
  - QB là đơn vị cơ bản để có thể chuyển sang các biểu thức ĐSQH và tối ưu hóa.
  - Một QB chứa một biểu thức đơn SELEC-FROM-WHERE-GROUP BY – HAVING.
  - Các câu truy vấn lồng trong 1 câu truy vấn là các QB độc lập.
  - Các toán tử gom nhóm (max, min, sum, count) được thể hiện dùng ĐSQH mở rộng.



Bộ tối ưu hóa truy vấn (Query Optimizer - QO) sẽ chọn lựa kế hoạch thực thi cho từng block.

- Bước 2
  - DBMS đề ra kế hoạch thực hiện câu truy vấn phù hợp nhất trong các chiến lược thực thi.
  - Tiến trình này gọi là tối ưu hóa câu truy vấn.
- Bước 3
  - Bộ phát sinh mã sẽ cho ra mã để thực thi câu truy vấn theo chiến lược vừa chọn.
- Bước 4
  - Thi hành mã đã phát sinh.

Sắp xếp ngoài (external sorting)

- Sắp xếp là thuật toán chính dùng khi xử lý truy vấn.Ví dụ ORDER BY.
- Sắp xếp cũng là bước quan trọng dùng cho phép join, union, và bước loại bỏ dòng trùng nhau khi thực hiện phép chiếu.
- Tránh thực hiện sắp xếp nếu dữ liệu đã có chỉ mục cho phép truy cập theo thứ tự.
- Sắp xếp ngoài đề cập đến các thuật toán sắp xếp trên tập tin cơ sở dữ liệu lớn không thể chứa đủ trong bộ nhớ chính.
- Sort-Merge:
  - Thuật toán sắp xếp gồm 2 bước: sorting và merging.
  - Sắp xếp các subfile (runs) của tập tin chính, sau đó trộn các sorted runs, rồi tạo subfile lớn hơn, sắp xếp rồi lại trộn chúng.
  - Kích thước của 1 run và số lượng run khởi đầu nR tùy vào số lượng file blocks b và không gian buffer trống nB.
    - Nếu nB = 5 và b = 1024 blocks thì nR = $\lceil b/nB \rceil$, tức là ban đầu có 205 run. Sau khi sắp xếp, 205 sorted run được lưu trong file tạm trên đĩa.

Phép chọn

- Có nhiều chọn lựa khi thực hiện phép chọn đơn.
  - S1: Tìm tuyến tính: đọc từng mẫu tin và kiểm tra giá trị thuộc tính có thỏa điều kiện chọn hay không.
  - S2: tìm nhị phân: nếu điều kiện chọn là phép so sánh bằng trên thuộc tính khóa dùng để sắp xếp file, thì tìm nhị phân sẽ được áp dụng.
  - S3: Dùng primary index hoặc hash key để đọc 1 mẫu tin nếu phép chọn là so sánh bằng trên thuộc tính khóa đã khai báo là primary index hoặc là khóa băm.
  - S4: Dùng primary index để tìm nhiều mẫu tin: nếu điều kiện so sánh là >, >=, <, <=, trên trường khóa được khai báo là primary index thì dùng index để tìm kiếm trên điều kiện =, sau đó tìm thêm các mẫu tin thỏa điều kiện không bằng.
  - S5: Dùng clustering index tìm nhiều mẫu tin: nếu điều kiện chọn là so sánh bằng trên trường không là khóa và có khai báo clustering index.
  - S6: Dùng secondary index trên điều kiện so sánh bằng để tìm 1 mẫu tin nếu index field là khóa hoặc tìm nhiều mẫu tin nếu indexing field không là khóa. Cách này cũng có thể dùng để tìm kiếm với điều kiện chọn không phải là so sánh bằng.

Phép chọn

- Điều kiện chọn phức nối nhau bởi AND
  - Nếu thuộc tính trong điều kiện chọn phức có liên quan đến các kiểu chọn đơn như đã đề cập thì vận dụng chúng, sau đó kiểm tra kết quả trả về có thỏa điều kiện chọn còn lại trong mệnh đề chọn phức hay không.
  - Nếu điều kiện chọn phức có liên quan đến composite index thì vận dụng chúng trực tiếp.
  - Dùng pp giao các record pointer của từng loại index liên quan đến điều kiện chọn phức nếu index đang dùng gồm có record pointer.

Phép kết $R \bowtie_{A=B} S$

- J1: Nested-loop join: đối với từng mẫu tin t trong R, tìm từng mẫu tin s trong S và kiểm tra xem hai mẫu tin có thỏa t[A] = s[B]?.
- J2: Single-loop join: đối với từng mẫu tin t trong R, dùng cấu trúc chỉ mục truy cập trực tiếp mẫu tin thỏa điều kiện kết ở quan hệ S.
- J3: Sort-merge join: nếu mẫu tin trong R và S đều được sắp xếp vật lý trên A và B thì phép kết diễn ra rất hiệu quả (nếu không thì sắp xếp cả hai trước), cả hai tập tin được duyệt theo thuộc tính kết, so khớp các mẫu tin cùng giá trị A và B.
- J4: Hash join (kết băm):dùng 1 hàm băm để ánh xạ các mẫu tin của R vào các bucket Ri dựa vào giá trị của A. Các mẫu tin của S cũng được ánh xạ vào các bucket Si. Các Ri và Si được duyệt qua để tổ hợp các bộ thuộc Hi và Si thỏa điều kiện kết.

Phép chiếu Π_{dstt} (R)

- Nếu dstt có chứa khóa của R thì số bộ kết quả bằng số bộ của R ban đầu.
- Nếu dstt không chứa khóa của R thì loại bỏ những bộ trùng.
  - Sắp xếp kết quả rồi loại bỏ những bộ trùng.

Phép toán tập hợp

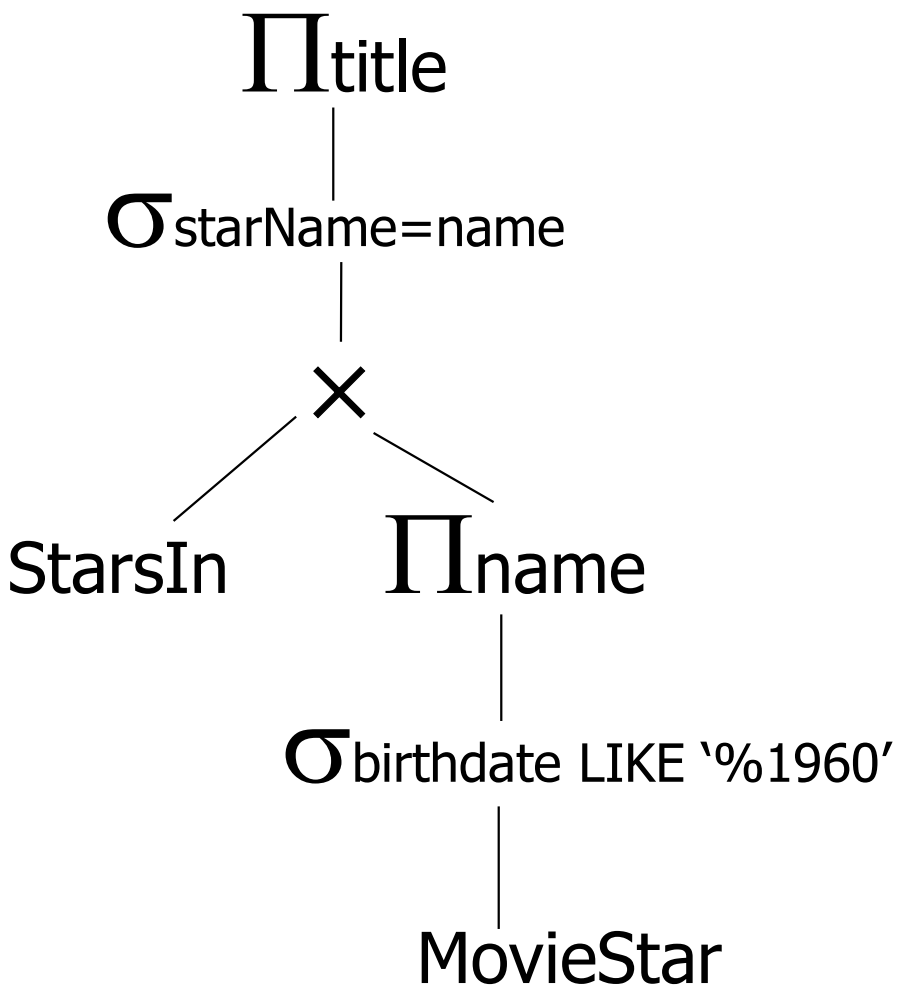
- Phép toán hội, giao, trừ đòi hỏi 2 quan hệ phải khả hợp, thường cài đặt bằng cách sắp xếp chúng theo cùng 1 thuộc tính, sau đó bằng 1 phép duyệt đơn giản lên 2 quan hệ cũng đủ tạo ra quan hệ kết quả.
- Phép tích đề - các tốn rất nhiều chi phí và nên tránh nếu có thể.

Các hàm kết hợp

- Nếu tính trên toàn bảng thì được thực hiện bằng việc duyệt bảng hoặc dùng index nếu có.
- Nếu tính toán trên từng nhóm (có group by) thì việc phân nhóm có thể thực hiện bằng cách:
  - Sắp xếp.
  - Băm.
  - Nếu có clustering index thì chỉ việc tính toán trên từng nhóm có sẵn.

Query tree

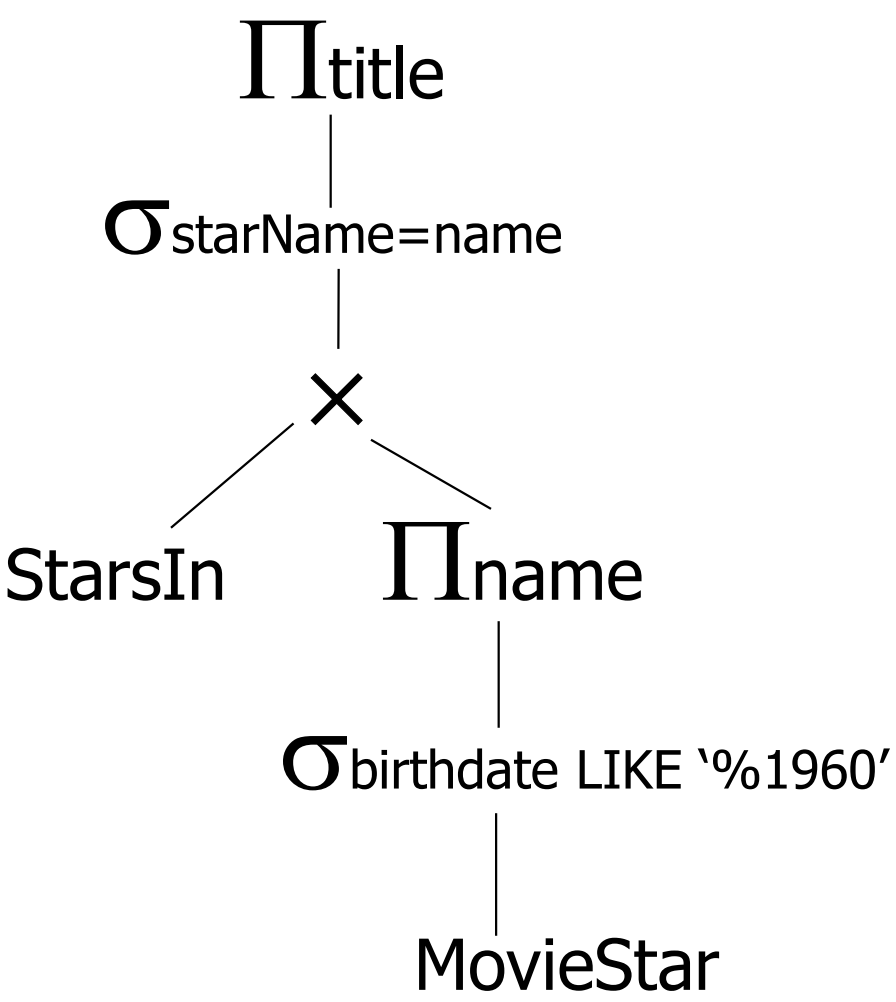
– Là cấu trúc dạng cây tương ứng với một biểu thức đại số quan hệ.

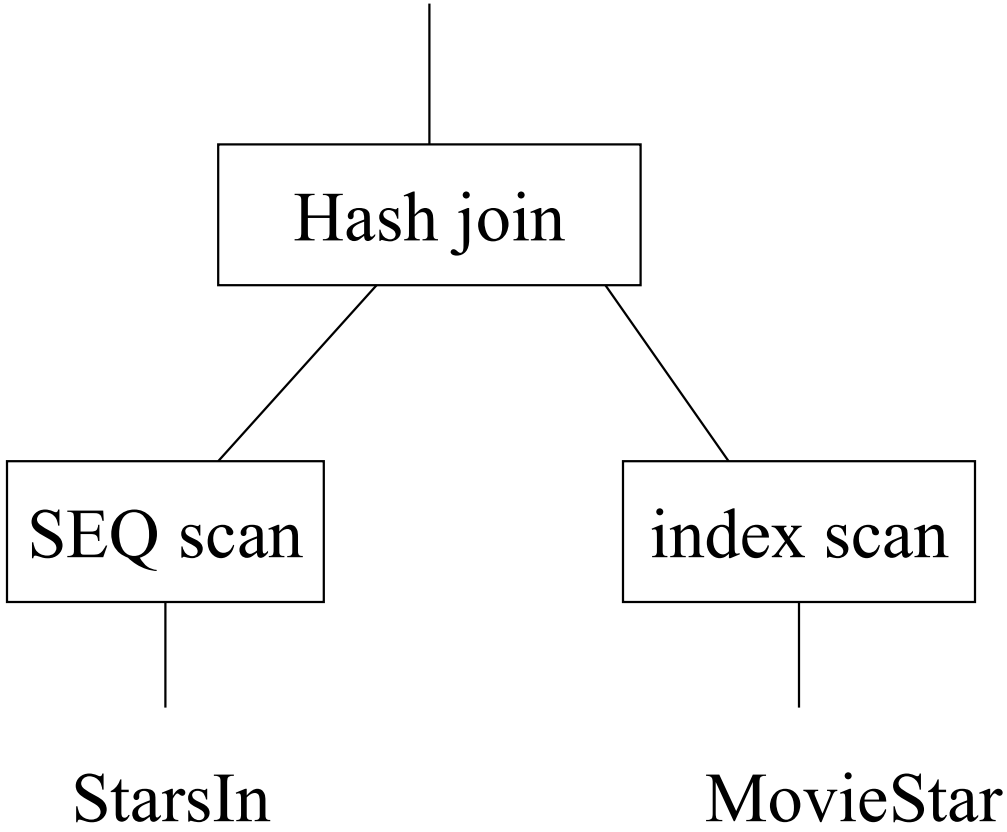


Kế hoạch thực thi truy vấn

- Kế hoạch thực thi mức logic (Logical plan) thể hiện mức cao và dùng đại số, qua cấu trúc ngôn ngữ truy vấn.
- Kế hoạch thực thi mức vật lý (Physical plan) thể hiện cấp thấp và liên quan đến việc thực hiện, qua các phương pháp truy xuất.
- Có nhiều kế hoạch thực thi truy vấn mức vật lý ứng với một kế hoạch thực thi mức logic cho trước.

LP và PP


Ptitle
sstarName=name
×
StarsIn
Pname
sbirthdate LIKE '%1960'
MovieStar


Hash join
SEQ scan
index scan
StarsIn
MovieStar

Các luật biến đổi tương đương

1. $\sigma_{c1 AND\ c2\ AND \ldots AND\ cn}(R) \equiv \sigma_{c1}(\sigma_{c2}(\ldots \sigma_{cn}(R) \ldots))$
2. $\sigma_{c1}(\sigma_{c2}(R)) \equiv \sigma_{c2}(\sigma_{c1}(R))$ giao hoán của σ
3. $\Pi_{L1}(\Pi_{L2}(\ldots(\Pi_{Ln}(R))\ldots)) \equiv \Pi_{L1}(R)$
4. $\Pi_{L1,L2,\ldots,Ln}(\sigma_c(R)) \equiv \sigma_c(\Pi_{L1,L2,\ldots,Ln}(R))$
5. $R_1 \bowtie_c R_2 \equiv R_2 \bowtie_c R_1$ giao hoán của $\bowtie$ và x

   $R_1 x R_2 \equiv R_2 x R_1$
6. $\sigma_c(R_1 \bowtie R_2) \equiv (\sigma_c(R_1)) \bowtie R_2$

   $\sigma_c(R_1 \bowtie R_2) \equiv (\sigma_{c1}(R_1)) \bowtie (\sigma_{c2}(R_2))$ nếu c có thể viết là c1 AND c2, c1 gồm thuộc tính của R1, c2 gồm thuộc tính của R2
7. $\Pi_L(R_1 \bowtie_c R_2) \equiv (\Pi_{A1, A2 \ldots, An}(R_1)) \bowtie_c (\Pi_{B1, B2 \ldots, Bn}(R_2))$ đổi chỗ giữa Π và $\bowtie$ (hoặc x) L = {A1, A2, … An, B1, B2, …, Bn} Ai $\in$ R1, Bi $\in$ R2
8. $\cup$ và $\cap$ có tính giao hoán, nhưng phép – thì không.
9. Tính kết hợp của θ: $\bowtie$, x, $\cup$ và $\cap$: $(R_1 \theta R_2) \theta R_3 \equiv R_1 \theta (R_2 \theta R_3)$
10. $\sigma_c(R_1 \theta R_2) \equiv (\sigma_c(R_1)) \theta (\sigma_c(R_2))$ đổi chỗ σ và θ (gồm $\cup$, $\cap$ và -)
11. $\Pi_L(R_1 \cup R_2) \equiv \Pi_L(R_1) \cup \Pi_L(R_2)$
12. $\sigma_c(R_1 x R_2) \equiv R_1 \bowtie_c R_2$ chuyển σ, x sang $\bowtie$
13. Luật DeMorgan NOT (C1 AND C2) ≡ (NOT C1) OR (NOT C2)

    NOT (C1 OR C2) ≡ NOT (C1) AND NOT(C2)

Các luật biến đổi tương đương

14. $(\sigma_P (R_1 - R_2) \equiv \sigma_P (R_1) - R_2$
15. $\Pi_{A1,\ldots, An}(\sigma_C(R)) \Leftrightarrow \Pi_{A1,\ldots, An}(\sigma_C(\Pi_{A1,\ldots, An,Ap} (R)))$

    $\sigma_{c1}(R_1 \bowtie_{c2} R_2) \equiv R_1 \bowtie_{c1 \wedge c2} R_2$

Giải thuật tối ưu hóa bt ĐSQH dựa trên Heuristics

1. Dùng quy tắc 1, tách các phép chọn đi cùng nhau để có thể tự do di chuyển phép chọn xuống các nhánh của cây.
2. Dùng quy tắc 2, 4, 6, 10 liên quan đến tính giao hóan giữa phép chọn và các phép toán khác để di chuyển phép chọn xuống nhánh của cây.
3. Dùng quy tắc 5 và 9 liên quan đến tính chất giao hoán và kết hợp của các phép 2 ngôi, để sắp xếp lại các nút lá của cây, để các phép chọn được ưu tiên thực hiện trước.
4. Dùng 12, kết hợp tích đề-các và phép chọn thành phép kết, nếu có thể xem điều kiện chọn là điều kiện kết.
5. Dùng 3, 4, 7, 11 để tách và đẩy các phép chiếu xuống các nhánh.
6. Nhận biết từng nhánh biểu diễn cho một nhóm các thao tác có thể thi hành bằng thuật toán đơn.

Tối ưu hóa câu truy vấn dùng việc chọn lựa và ước lượng chi phí

- Ước lượng chi phí thi hành một câu truy vấn cho nhiều chiến lược thực thi khác nhau và chọn ra chiến lược thi hành có chi phí thấp nhất.
- Chi phí cho một chiến lược bao gồm:
  1. Chi phí truy xuất đến nơi lưu trữ thứ cấp (vd: đĩa cứng)
  2. Chi phí lưu trữ dữ liệu kết quả trung gian.
  3. Chi phí tính toán: để thực hiện các thao tác trong bộ nhớ chính.
  4. Chi phí truyền thông.

Tối ưu hóa câu truy vấn dùng việc chọn lựa và ước lượng chi phí

- Để ước lượng chi phí cho các chiến lược truy vấn khác nhau, cần lưu lại thông tin cần thiết trong catalog để bộ tối ưu hóa sử dụng.
  - Số mẫu tin r.
  - Kích thước trung bình của từng mẫu tin R.
  - Số khối b.
  - Hệ số khối bfr.
  - Chỉ mục nếu có loại gì (primary, secondary, clustering): số mức x (nếu là multilevel index), số block ở mức đầu tiên của index b_{I1}
  - …

Hết chương 5.