



Chapter 12: 질의 처리

Database System Concepts, 6th Ed.

©Silberschatz, Korth and Sudarshan

See www.db-book.com for conditions on re-use



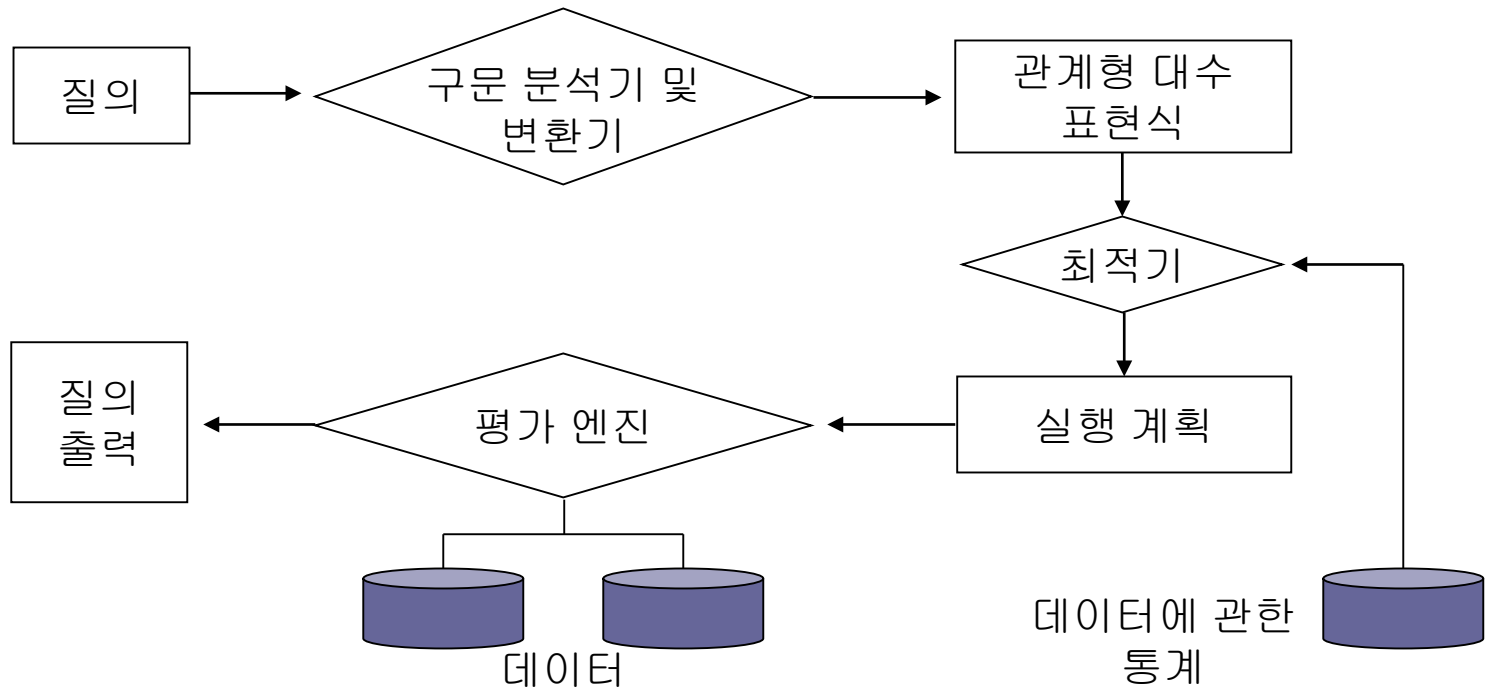
Chapter 12: 질의처리

- 개요
- 질의 비용산정
- 비용 산정을 위한 카타로그 정보
- 선택 연산
- 정렬
- 조인 연산
- 기타 연산
- 표현식의 평가



질의처리의 기본절차

1. 구문 분석 및 변환
2. 최적화
3. 평가





관계대수 연산자 (복습)

| Symbol (Name) | Example of Use |
|---------------------------------|---|
| σ (Selection) | $\sigma_{\text{salary} \geq 85000}(\text{instructor})$ |
| | Return rows of the input relation that satisfy the predicate. |
| Π (Projection) | $\Pi_{ID, salary}(\text{instructor})$ |
| | Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output. |
| \bowtie (Natural Join) | $\text{instructor} \bowtie \text{department}$ |
| | Output pairs of rows from the two input relations that have the same value on all attributes that have the same name. |
| \times (Cartesian Product) | $\text{instructor} \times \text{department}$ |
| | Output all pairs of rows from the two input relations (regardless of whether or not they have the same values on common attributes) |
| \cup (Union) | $\Pi_{name}(\text{instructor}) \cup \Pi_{name}(\text{student})$ |
| | Output the union of tuples from the two input relations. |



선택연산 (Selection of tuples) - 복습

□ Relation r

| A | B | C | D |
|----------|----------|----|----|
| α | α | 1 | 7 |
| α | β | 5 | 7 |
| β | β | 12 | 3 |
| β | β | 23 | 10 |

□ Select tuples with A=B
and D > 5

□ $\sigma_{A=B \text{ and } D > 5}(r)$

```
SELECT *  
FROM r  
WHERE A=B and D>5;
```

| A | B | C | D |
|----------|----------|----|----|
| α | α | 1 | 7 |
| β | β | 23 | 10 |



추출 연산 (Selection of Columns) -복습

□ Relation r :

| A | B | C |
|----------|----|---|
| α | 10 | 1 |
| α | 20 | 1 |
| β | 30 | 1 |
| β | 40 | 2 |

- Select A and C
 - Projection
 - $\Pi_{A, C}(r)$

```
SELECT A, C  
FROM r;
```

| A | C |
|----------|---|
| α | 1 |
| α | 1 |
| β | 1 |
| β | 2 |

=

| A | C |
|----------|---|
| α | 1 |
| β | 1 |
| β | 2 |



자연 조인 (Natural Join) - 복습

□ Relations r, s :

| A | B | C | D |
|----------|-----|----------|-----|
| α | 1 | α | a |
| β | 2 | γ | a |
| γ | 4 | β | b |
| α | 1 | γ | a |
| δ | 2 | β | b |

r

| B | D | E |
|-----|-----|------------|
| 1 | a | α |
| 3 | a | β |
| 1 | a | γ |
| 2 | b | δ |
| 3 | b | ϵ |

s

□ Natural Join

□ $r \bowtie s$

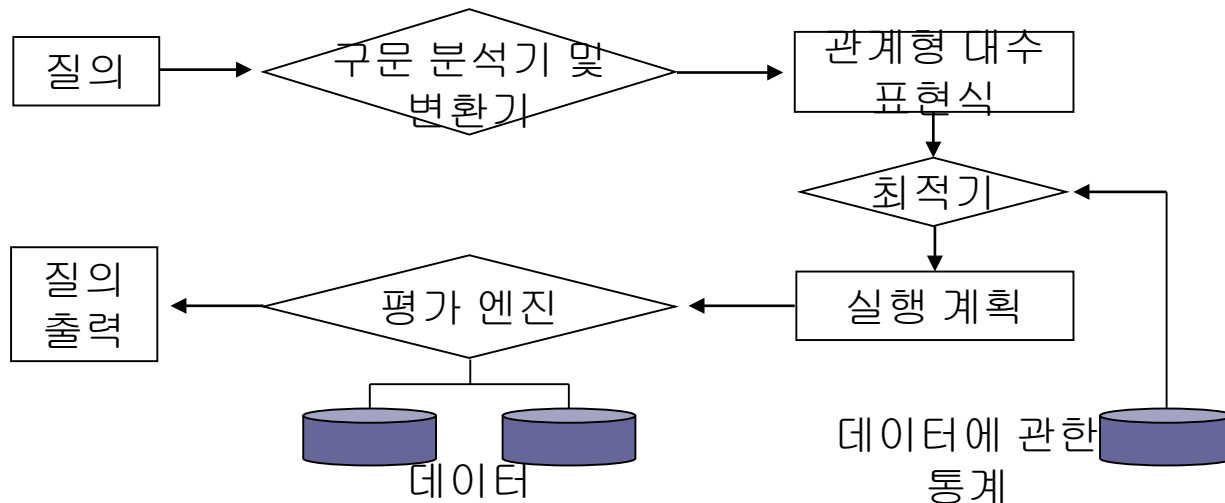
SELECT *
FROM r NATURAL JOIN s ;

| A | B | C | D | E |
|----------|-----|----------|-----|----------|
| α | 1 | α | a | α |
| α | 1 | α | a | γ |
| α | 1 | γ | a | α |
| α | 1 | γ | a | γ |
| δ | 2 | β | b | δ |



질의 처리의 기본 절차 (계속)

- 구문 분석 및 변환
 - 구문 분석기는 문법을 체크하고 릴레이션을 검사한다.
 - 질의를 내부 형식으로 변환한다. 즉, 관계형 대수로 변환된다.
- 평가
 - 질의 실행 엔진은 질의 실행 계획을 취해, 그 계획을 실행하고 결과를 돌려준다.





질의처리의 기본절차 : 최적화

- 주어진 관계형 대수 표현식은 동등한 많은 표현식을 가질 수 있다.

- Select salary from instructor where salary < 75000

```
SELECT salary
FROM instructor
WHERE salary < 75000;
```

| ID | name | dept_name | salary |
|-------|------------|------------|--------|
| 22222 | Einstein | Physics | 95000 |
| 12121 | Wu | Finance | 90000 |
| 32343 | El Said | History | 60000 |
| 45565 | Katz | Comp. Sci. | 75000 |
| 98345 | Kim | Elec. Eng. | 80000 |
| 76766 | Crick | Biology | 72000 |
| 10101 | Srinivasan | Comp. Sci. | 65000 |
| 58583 | Califieri | History | 62000 |
| 83821 | Brandt | Comp. Sci. | 92000 |
| 15151 | Mozart | Music | 40000 |
| 33456 | Gold | Physics | 87000 |
| 76543 | Singh | Finance | 80000 |

- E.g., $\sigma_{salary < 75000}(\Pi_{salary}(instructor))$ is equivalent to $\Pi_{salary}(\sigma_{salary < 75000}(instructor))$

- 각각의 관계형 대수 연산은 많은 서로 다른 방법으로 평가될 수 있다.

- 따라서 임의의 관계형 대수 표현식은 많은 서로 다른 방법으로 평가될 수 있다.

- 상세한 평가 전략을 명시하고 있는 주석이 붙은 표현식을 평가 계획(evaluation-plan)이라 한다.

- 예를 들어, 연봉이 75000불보다 적은 강사를 찾기 위해 salary 인덱스를 사용할 수 있다
- 또는 릴레이션 전체를 검색해 연봉이 75000불 이상인 강사 정보는 고려하지 않을 수 있다.



기본 절차: 최적화 (계속)

- **질의 최적화 (Query Optimization):** 모든 동등한 표현식 중에서 가장 비용이 적게 드는 평가 계획을 선택하도록 한다.
 - 계획의 비용 산정은 DBMS 카타로그내의 통계 정보에 의거한다.
 - ▶ e.g. 릴레이션의 튜플 수, 튜플의 크기 등
- 이 장에서의 학습 내용
 - 질의 비용 산정 방법
 - 관계 표현식 평가 알고리즘
 - 각 알고리즘의 결합/연결 수행 방법
- In Chapter 13
 - 최소의 비용을 가진 평가 계획을 구하는 방법에 대하여 학습함



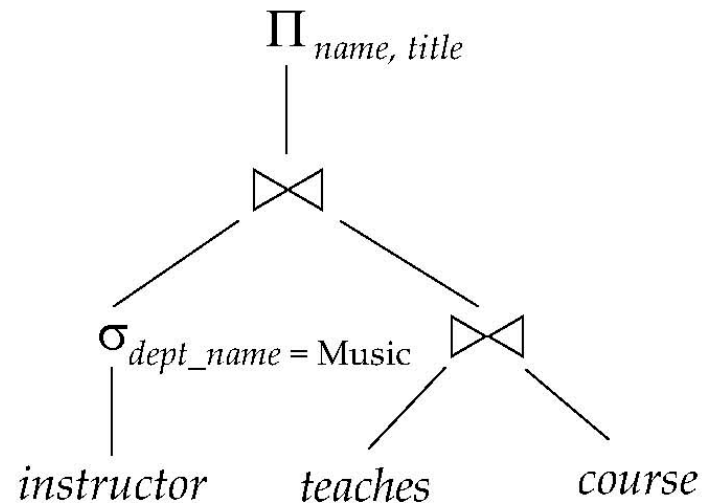
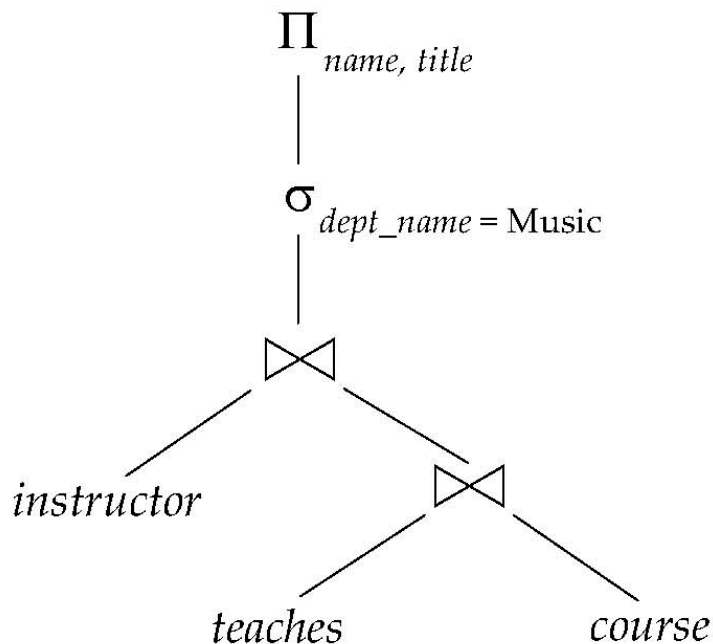
질의 최적화?

음악 대학의 모든 교수 이름과
그 교수들이 가르치는 과목명을
찾으시오.

```
SELECT name, title  
FROM instructor NATURAL JOIN teaches NATURAL JOIN course  
WHERE instructor.dept_name="Music";
```

$$\Pi_{name, title}(\sigma_{dept_name = \text{"Music"}} (instructor \bowtie (teaches \bowtie (course))))$$

- 주어진 질의를 평가하는 서로 다른 방법
 - 동일 표현식
 - 각 연산에 대한 서로 다른 알고리즘을 사용할 수 있음





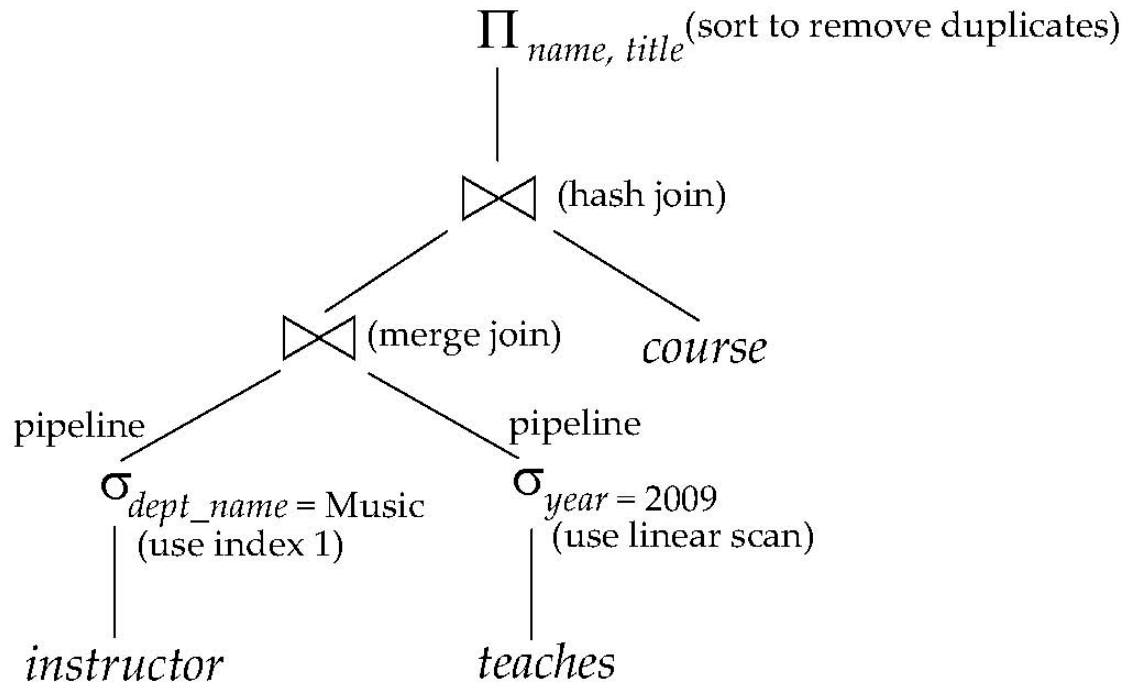
평가 계획(evaluation plan)의 예

2009년에 강의를 한 음악 대학의 모든 교수 이름과 그 교수들이 가르친 과목명을 찾으시오.

```
SELECT name, title
FROM instructor NATURAL JOIN teaches NATURAL JOIN course
WHERE instructor.dept_name='Music' and year=2009;
```

$$\Pi_{name, title}(\sigma_{dept_name = \text{"Music"} \wedge year = 2009} (instructor \bowtie (teaches \bowtie (course))))$$

- 평가 계획은 각 연산에 어떤 알고리즘이 사용되고 연산들의 실행이 어떻게 협력하는지를 정확히 정의한다.



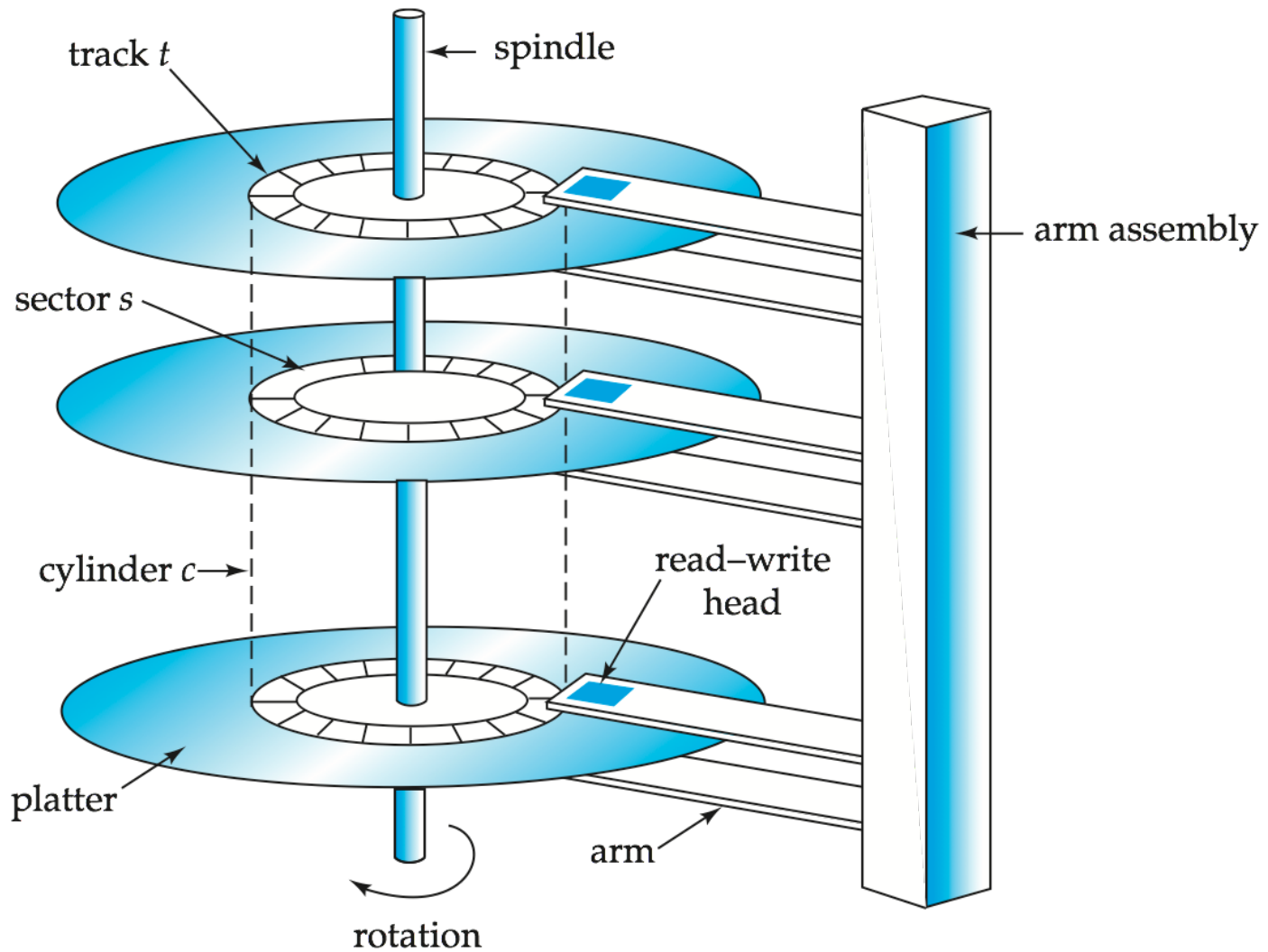


질의 처리 비용 산정

- 비용은 질의에 대한 답을 얻기까지 걸린 총 시간을 나타낸다. 비용을 산정하는 데는 많은 가능한 방법이 있는데, 그 예로는 디스크 액세스, CPU 시간 및 분산 또는 병렬 시스템에서의 통신 비용 등이 있다.
- 디스크 액세스가 일반적으로 가장 유력한 비용이며, 또한 산정하기가 비교적 쉽다. 다음 요소들로 이루어진다.
 - Number of seeks * average-seek-cost
 - Number of blocks read * average-block-read-cost
 - Number of blocks written * average-block-write-cost
 - ▶ 블록 쓰기 연산은 블록 읽기 연산에 비하여 비용이 많이 든다.
 - 데이터 쓰기 연산 후, 쓰기 연산 수행이 성공적으로 수행되었는지를 확인하기 위하여 읽기 연산을 재수행한다.



자기 디스크 구조 (복습)



NOTE: Diagram is schematic, and simplifies the structure of actual disk drives



질의 처리 비용 산정 (계속)

- 일반적으로 디스크로부터의 블록 전송 수(**number of block transfers**)와 탐색 수 (**number of seeks**)가 평가의 실질적인 비용산정으로 사용된다.
 - t_T – time to transfer one block
 - t_S – time for one seek
 - b개의 블록 전송과 S번의 탐색 비용
$$b * t_T + S * t_S$$
- CPU 비용은 일반적으로 무시한다.
 - 실시간 시스템에서는 CPU 비용을 산정하여야 한다.
- 이후, 비용 산정에 있어 디스크에 결과를 쓰기 연산하는 비용을 포함시키지 않는다고 가정한다.



질의 처리 비용 산정 (계속)

- 알고리즘의 비용은 많은 메모리를 가지면 디스크 액세스를 줄일 수 있으므로, 메인 메모리 내 버퍼의 크기에 따른다.
- 따라서, 비용을 산정할 때 메모리의 크기가 파라미터가 되어야 한다 : 흔히 최악의 경우의 비용을 사용한다.



비용 산정을 위한 카타로그 정보

- n_r : 릴레이션 r 내의 튜플의 수
- b_r : r 의 튜플들을 내포하고 있는 블록의 수
- s_r : r 의 한 튜플의 바이트 단위의 크기
- f_r : r 의 블록킹 요인 – 즉, 한 블록에 들어가는 r 의 튜플 수
- r 의 튜플들이 파일내에 물리적으로 함께 저장되면, 다음과 같다.

$$b_r = \left\lceil \frac{n_r}{f_r} \right\rceil$$

- $V(A, r)$: 애트리뷰트 A 에 대해 r 에 나타나는 서로 다른 값의 수 ;
 $\Pi_A(r)$ 의 크기와 같다.
- f_i : B^+ - 트리와 같은 트리 구조 인덱스에 있어, 인덱스 i 의 내부 노드의 평균 전개
- h_i : 인덱스 i 의 계층 수 – 즉, i 의 높이이다.
 - (B^+ - 트리 같은) 인덱스에 있어, $h_i = \lceil \log_{f_i}(V(A, r)) \rceil$
 - 해쉬 인덱스에 있어 h_i 는 1이다.



Thank You