Relational Database: Identities of Relational Algebra; Example of Query Optimization

Greg Plaxton
Theory in Programming Practice, Spring 2005
Department of Computer Science
University of Texas at Austin
Selection Splitting

• For any database relation $R$ and predicates $p$, $q$, we have

$$\sigma_{p \land q}(R) = \sigma_p(\sigma_q(R))$$

• A corollary is that selection is commutative, that is,

$$\sigma_p(\sigma_q(R)) = \sigma_q(\sigma_p(R))$$
Projection Refinement

- For any subsets $a$ and $b$ of a database relation $R$ such that $a \subseteq b$, we have

$$\pi_a(R) = \pi_a(\pi_b(R))$$
Commutativity of Selection and Projection

• For any subset \( a \) of the attributes of a database relation \( R \), and any predicate \( p \), we have

\[
\pi_a(\sigma_p(R)) = \sigma_p(\pi_a(R))
\]
Commutativity and Associativity of Union, Cross Product, Join

- Union and cross product are commutative and associative
- Join is commutative
- For any database relations $R$, $S$, and $T$ such that (1) $R$ and $S$ have at least one common attribute, (2) $S$ and $T$ have at least one common attribute, and (3) no attribute is common to $R$, $S$, and $T$, we have

\[(R \bowtie S) \bowtie T = R \bowtie (S \bowtie T)\]
Selection Pushing

- For any database relations $R$ and $S$, any predicate $p$, and any operator $\circ$ in the set \{∪, ∩, −\}, we have

$$\sigma_p(R \circ S) = \sigma_p(R) \circ \sigma_p(S)$$

- For any database relations $R$ and $S$, any predicate $p$ that depends only on attributes of $R$, and any operator $\otimes$ in the set \{×, ♦\}, we have

$$\sigma_p(R \otimes S) = \sigma_p(R) \otimes S$$
Projection Pushing

- For any database relations $R$ and $S$, and any set of attributes $a$, we have

$$\pi_a(R \cup S) = \pi_a(R) \cup \pi_a(S)$$
Distributivity of Projection over Join

For any database relations $R$ and $S$ with associated sets of attributes $r$ and $s$, respectively, and any sets of attributes $a$, $b$, and $c$ such that $a \subseteq r \cup s$, $b = (a \cap r) \cup d$, and $c = (a \cap s) \cup d$ where $d = r \cap s$, we have

$$\pi_a(R \bowtie S) = \pi_a(\pi_b(R) \bowtie \pi_c(S))$$
An “Unnamed” Identity (to be used later)

• For any database relations $R$ and $S$, and any predicates $p$ and $q$ such that $p$ depends only on attributes of $R$ and $q$ depends only on attributes of $S$, we have

$$\sigma_{p \land q}(R \Join S) = \sigma_p(R) \Join \sigma_q(S)$$

• Proof:
  – By selection splitting and commutativity of join,

$$\sigma_{p \land q}(R \Join S) = \sigma_p(\sigma_q(S \Join R))$$

  – By selection pushing over join and commutativity of join,

$$\sigma_p(\sigma_q(S \Join R)) = \sigma_p(R \Join \sigma_q(S))$$

  – By selection pushing over join,

$$\sigma_p(R \Join \sigma_q(S)) = \sigma_p(R) \Join \sigma_q(S)$$
Query Optimization

• We are given a query in the form of a relational algebra expression $\alpha$

• We could evaluate $\alpha$ directly

• Instead, it might be more efficient to use identities such as the ones presented earlier to obtain an equivalent expression $\beta$ for which a direct evaluation is more efficient
An Example of Query Optimization

• We consider an abstraction of the movie example discussed in the course packet

• For the sake of brevity, we use the letters $A$ through $I$ to refer to the nine attributes of the example

• We have three database relations $R$, $S$, and $T$ with attributes \{$A, B, C, D, E$\}, \{$A, F, G, H$\}, and \{$F, I$\}, respectively

• Let $p$ (resp., $q$) denote a predicate asserting that attribute $B$ (resp., $G$) has a particular given value

• We wish to evaluate $\pi_I(\sigma_{p \land q}(R \bowtie S \bowtie T))$
Example: High Level

• We wish to evaluate

\[ \pi_I(\sigma_{p \land q}(R \bowtie S \bowtie T)) \]

• We will prove that this expression is equivalent to

\[ \pi_I([\pi_A(\sigma_p(R)) \bowtie \pi_{A,F}(\sigma_q(S))] \bowtie T) \]

• Why is the latter expression likely to be more efficient to evaluate directly?

• In what follows we will give a step-by-step proof of the equivalence of the two preceding formulae
Step One

• Claim:

\[ \pi_I(\sigma_{p \land q}(R \Join S \Join T)) = \pi_I(\sigma_{p \land q}[(R \Join S) \Join T]) \]

• This claim follows from the associativity of \(\Join\) (as we have already noted, the required conditions are met)
Step Two

• Claim:

$$\pi_I(\sigma_{p \land q}[(R \bowtie S) \bowtie T]) = \pi_I(\sigma_{p \land q}(R \bowtie S) \bowtie T)$$

• This claim follows from selection pushing over join
Step Three

• Claim:

$$\pi_I(\sigma_{p \land q}(R \bowtie S) \bowtie T) = \pi_I([\sigma_p(R) \bowtie \sigma_q(S)] \bowtie T)$$

• This claim follows from the “unnamed” identity established earlier since $p$ only involves attribute $B$ and $q$ only involves attribute $G$
  – Note that $B$ is an attribute of $R$ and $G$ is an attribute of $S$
Step Four

- Claim:

\[ \pi_I([\sigma_p(R) \bowtie \sigma_q(S)] \bowtie T) = \pi_I(\pi_F[\sigma_p(R) \bowtie \sigma_q(S)] \bowtie \pi_{F,I}(T)) \]

- This claim follows from distributivity of projection over join
Step Five

• Claim:

\[ \pi_I(\pi_F[\sigma_p(R) \bowtie \sigma_q(S)] \bowtie \pi_{F,I}(T)) = \pi_I(\pi_F[\sigma_p(R) \bowtie \sigma_q(S)] \bowtie T) \]

• This claim follows from the observation that \( \pi_{F,I}(T) = T \)
Last Step

- Claim:

\[ \pi_I(\pi_F[\sigma_p(R) \bowtie \sigma_q(S)] \bowtie T) = \pi_I([\pi_A(\sigma_p(R)) \bowtie \pi_{A,F}(\sigma_q(S))] \bowtie T) \]

- This claim follows from distributivity of projection over join
  - Note that the lone common attribute of \( \sigma_p(R) \) and \( \sigma_q(S) \) is \( A \)