**Relational Model**

'set-theoretic' notation:

o Domain: a set of values.

o Cartesian product of domains D1, D2, ... Dk : {(v1, v2, ..., vk) | vi ε Di, 0 <= i <= k}

o k-tuple: (v1, v2, ..., vk).

o A relation is any subset of the Cartesian product of one or more domains.

o Tuples: the members of relations.

o A relation of k domains has *arity* (or *degree*) of k.

o Each component of a k-tuple may has a name called *an attribute*.

'set-of-mapping' notation:

o Tuples as mapping from attribute names to values:

e.g. t = (12345, Houston, 713-200-2000).

t(S\_NO) = 12345

t(CITY) = Houston

t(PHONE) = 713-200-2000

**Relational Algebra**

o Satisfy the closure property: the result is also a relation.

o relational algebra operators can be nested.

o procedural, not descriptive.

**Basic operations:**

1. Select:

SELECT relation-name WHERE condition.

σcond(R) = {t | t ε R and cond}

o Horizontal subset.

o Conditions may include attributes.

o Select from one relation only.

o Selection is *associative*.

2. Project

PROJECT relation-name OVER (col-name, ..., col-name)

πc1, .., cm(R) = {s | Ǝ t ε R (t(ci) = s(ci), for 1 <=i <= m)}

o 'vertical' subset.

o duplicate 'rows' remove. (note: a relation is a set).

o Projection is *associative* if it makes sense.

o Combining projection and selection is *not* associative:

πφ(σΘ(R)) /= σΘ(πφ(R))

3. Union

Relation\_1 UNION Relation\_2

R U S = {t | t ε R or t ε S}

o R1 and R2 must be *union-compatible*, with the *same* structure.

o Example: FEMALE\_EMPLOYEE U MALE\_EMPLOYEE.

UHU\_STUDENT U UHCL\_STUDENT.

o Unions can usually be replaced by *natural joins*.

4. Difference

Relation\_1 DIFFERENCE Relation\_2

R - S = {t | t ε R and not (t ε S)}

o R and S must be union-compatible (with the same structure).

o Example: UHU\_STUDENT - UHCL\_STUDENT.

5. Cartesian Product

R TIMES S

R x S = {t | Ǝ t1 ε R and t2 ε S such that t(R) = t1 and t(S) = t2}

o Combining *all possible* information from more than one tables.

o Name ambiguity may be resolved by using *full* names.

o Costly operation: produce large relations.

o Much redundancy in information.

**Other operations**

1. Intersection:

R1 INTERSECT R2

R1 ∩ R2 = R1 - (R1 - R2)

2. Theta-join:

R1 TIMES R2 WHERE condition

R1 |x|Θ R2

o Cartesian Product with a condition.

o Note the two vertical bars.

o Equi-join: a theta join with an equality condition for *common* attributes.

3. Natural join:

R JOIN S

o Eliminate redundancy in equi-join on common attributes. Keep only one copy of the common attributes.

let C1, C2, ... Cm be the common attributes of R1 and R2.

R |x| S = πA1, A2, .. Al(σR.C1=S.C1,.., R.Cm=S.Cm(RxS)

where A1, A2, ... Al is the list of attributes in RxS except S.C1, S.C2,.. S.Cm.

o Most popular and 'natural' join.

4. Semi-join:

R SEMIJOIN S

o A natural join with attributes only from the first relation.

R |x S = πR(R |x| S)

5. Outerjoin (theta join plus unparticipatory tuple in one relation included, with null values in the attributes of the other relation).

o seldom used.

6. Divide

R DIVIDEBY S

R ÷ S

Let the schemes of R, S and T be dom(R), dom(S) and dom(T) = dom(R) - dom(S) respectively

Condition s is a *proper* subset of r.

R ÷ S = {t | t in dom(T) and for all s ε S, there exist r ε R such that r = st}

**Issues of relational algebra**

o Cannot navigate tuples.

e.g. Show the available total quantities of all parts.

o Cannot deal with recursion.

e.g. Employee(SS#, Supervisor\_SS#, ...)

Find all supervisors (direct or indirect).

**Exercise**

Supplier(SNum, SName, SCity, Status)

Part(PNum, PName, Color, Weight, PCity)

Supply(SNum, PNum, Quantity)

1. Show all information of suppliers in the city Houston.

2. Show all information of parts with the color Red.

3. Show all information of suppliers with a status greater than 5.

4. Show all information of parts with a color of Red and weight more than 5 lbs.

5. Show all information of parts with a color of Red or Blue.

6. List all supplier names.

7. List all part numbers.

8. List all part colors.

9. Show all supplier names in the city Houston.

10. Show all supplier numbers that represent suppliers supplying part P2.

11. Show all supplier numbers that represent suppliers supplying more than 20 of part P2.

12. Show the part numbers and weights of all parts with color Red.

13. Show the part numbers and weights of all parts with weights more than 5 lbs.

14. Show all supplier status of all suppliers in Houston.

15. Show all information of suppliers that supplies part P1.

16. Show all information of parts supplied by supplier S1.

17. Show all information of parts supplied by supplier S1 *or* S2.

18. Show all supplier names of suppliers supplying part P1.

19. Show all information of parts supplied by supplier S1 *and* S2.

20. Show all part numbers representing parts supplied by supplier S3 or S4.

21. Show all cities with suppliers supplying part P3.

22. Show all status with active suppliers (a supplier is active if it supplies at least one part).

23. Show all information of parts that are supplied by a supplier in the city Houston.

24. Show all information of suppliers supplying a red part.

25. Show all information of suppliers with a status of greater than five and supplies a part of weight greater than five.

26. Show all cities that contain inactive suppliers.

27. Show the supplier numbers of all suppliers that supply part P1 but not part P2.

28. Show the supplier numbers of all suppliers that supply red parts but not green parts.

29. Show all information of parts that are supplied by at least one supplier in Houston.

30. Show all information of suppliers that supplies all parts supplied by supplier S1.

31. Show the part names of all parts that are supplied by at least one supplier in Houston with a status of 5 or above.

32. Show the part names of all parts that are supplied by every supplier in Houston with a status of 5 or above.