Outer Join

Teacher (name, rank)
Course (subject, enrollment, quarter, teacher)

```
WITH
    innerjoin(name, rank, subject, enrollment) AS
            (SELECT t.name, t.rank, c.subject, c.enrollment
             FROM teachers AS t, courses AS c
             WHERE t.name=c.teacher AND c.quarter='Fall 96') ,
    teacher-only(name, rank) AS
            (SELECT name, rank
             FROM teachers
             EXCEPT ALL
             SELECT name, rank
             FROM innerjoin) ,
    course-only(subject, enrollment) AS
            (SELECT subject, enrollment
             FROM courses
             EXCEPT ALL
             SELECT subject, enrollment
             FROM innerjoin)
```

Outer Join

```
SELECT name, rank, subject, enrollment
FROM innerjoin
UNION ALL
SELECT name, rank,
        CAST (NULL AS Varchar(20)) AS subject,
        CAST (NULL AS Integer) AS enrollment
FROM teacher-only
UNION ALL
SELECT CAST (NULL AS Varchar(20)) AS name,
        CAST (NULL AS Varchar(20)) AS rank,
        subject, enrollment
FROM course-only ;
```

Some New Features of SQL

- CAST expression
- CASE expression
- Sub-query
- Outer Join
- **Recursion**

Recursion

- If a common table expression uses itself in its definition, this is called recursion. It can calculate a complex recursive inference in one SQL statement.

  FedEmp (name, salary, manager)

- *Find all employees under the management of Hoover and whose salary is more than 100000*

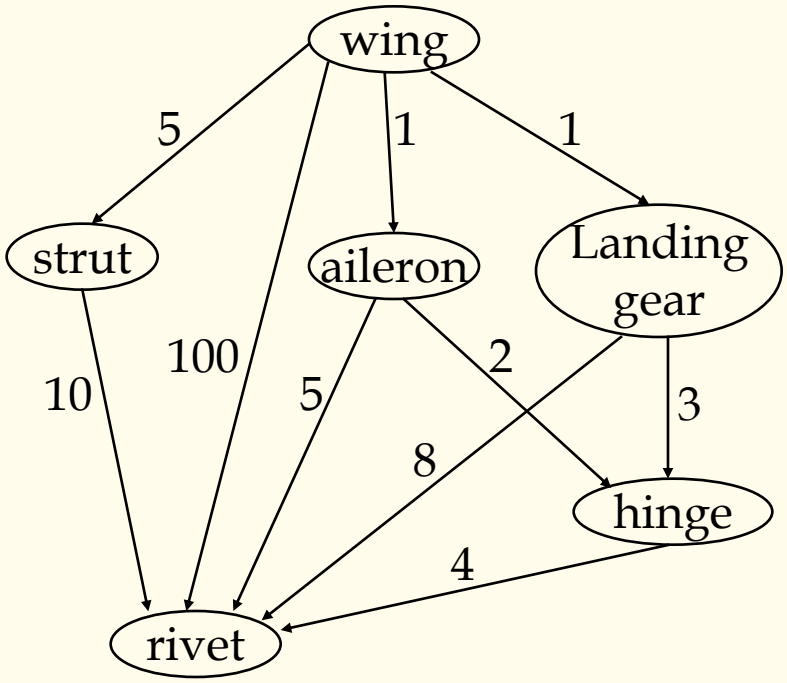
```
WITH agents (name, salary) AS
        ((SELECT name, salary                    --- initial query
          FROM FedEmp
          WHERE manager='Hoover')
        UNION ALL
         (SELECT f.name, f.salary                --- recursive query
          FROM agents AS a, FedEmp AS f
          WHERE f.manager = a.name))
SELECT name                                      --- final query
FROM agents
WHERE salary>100000 ;
```

Recursive Calculation

- A classical "parts searching problem"

Components

Part	Subpart	QTY
wing	strut	5
wing	aileron	1
wing	landing gear	1
wing	rivet	100
strut	rivet	10
aileron	hinge	2
aileron	rivet	5
landing gear	hinge	3
landing gear	rivet	8
hinge	rivet	4



Directed acyclic graph, which assures the recursion can be stopped

Recursive Calculation

- *Find how much rivets are used in one wing?*
- A temporary view is defined to show the list of each subpart's quantity used in a specified part :

```
WITH wingpart (subpart, qty) AS
    ((SELECT subpart, qty                    ---initial query
      FROM components
      WHERE part='wing')
     UNION ALL
     (SELECT c.subpart, w.qty*c.qty  ---recursive qry
      FROM wingpart w, components c
      WHERE w.subpart=c.part))
```

wingpart

Subpart	QTY	
strut	5	Used directly
aileron	1	Used directly
landing gear	1	Used directly
rivet	100	Used directly
rivet	50	Used on strut
hinge	2	Used on aileron
rivet	5	Used on aileron
hinge	3	on landing gear
rivet	8	on landing gear
rivet	8	on aileron hinges
rivet	12	on L G hinges