

# Unit 5

## The Network Model

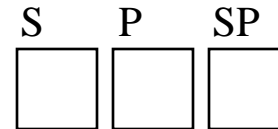
- ❑ 5.1 Data Modeling Issues
- ❑ 5.2 The Network Model
- ❑ 5.3 IDMS

# 5.1 Data Modeling Issue

- Issue: **How to represent entities and relationships?**

- Two major paradigms

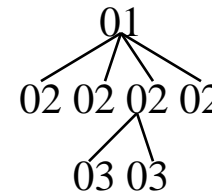
- Relational
- Graph { Hierarchical  
Network



- E.g., a 'pile' of data

- John, 25, NCTU, CS Dept, ...
- Mary, 22, NCTU, IS Dept, ...

..  
 COBOL  
 01 data-record  
   02 name    PIC X(6)  
   02 age     PIC 9(2)  
   02 univ    PIC X(4)  
   02 dept    PIC X(3)



01			
02	02	02	02
		03	03

# Model 1: Relational

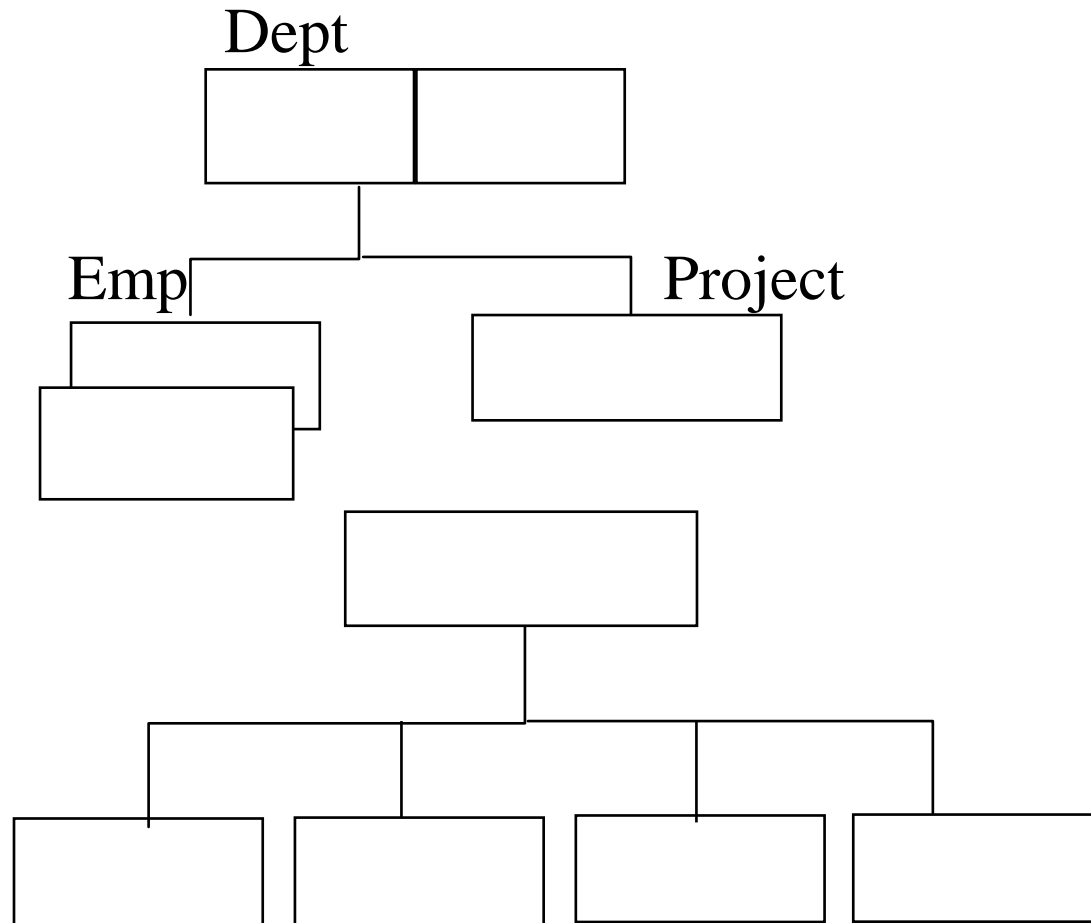
---

- Model 1: Relational
  - Decomposition (normalization issue)

Emp	Dept	Project

# Model 2: Hierarchical

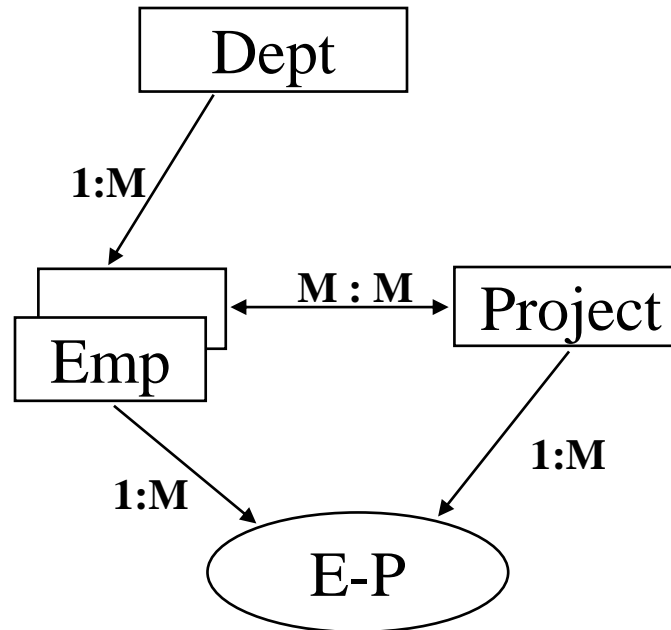
---



# Model 3: Network

---

- Model 3: Network  
proposed by CODASYL



# 5.2 The Network Model

- Data Structure

Consider the 'supplier-and-parts' database

**S**

**P**

**SP**

S#	SNAME	STATUS	CITY
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris
S4	Clark	20	London
S5	Adams	30	Athens

P#	PNAME	COLOR	WEIGHT	CITY
P1	Nut	Red	12	London
P2	Bolt	Green	17	Paris
P3	Screw	Blue	17	Rome
P4	Screw	Red	14	London
P5	Cam	Blue	12	Paris
P6	Cog	Red	19	London

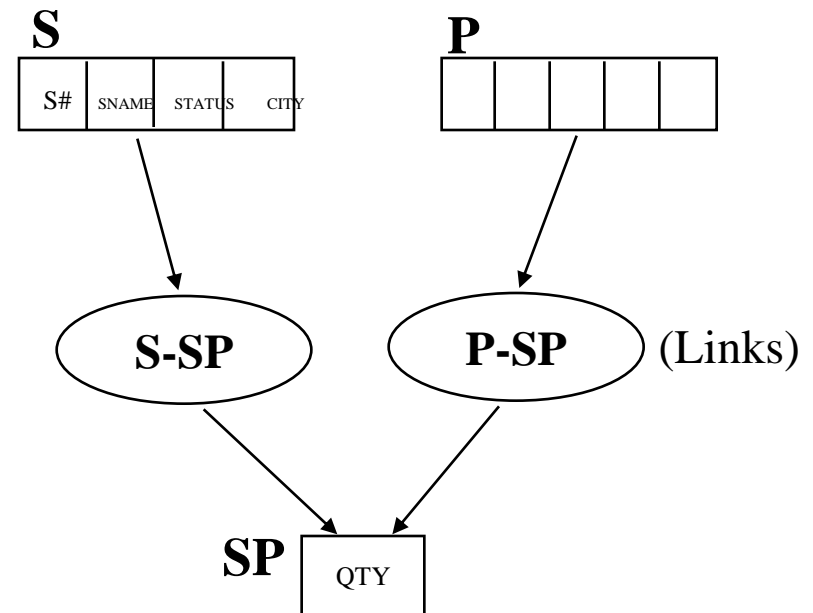
S#	P#	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	300
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

} Structure  
} Sample values

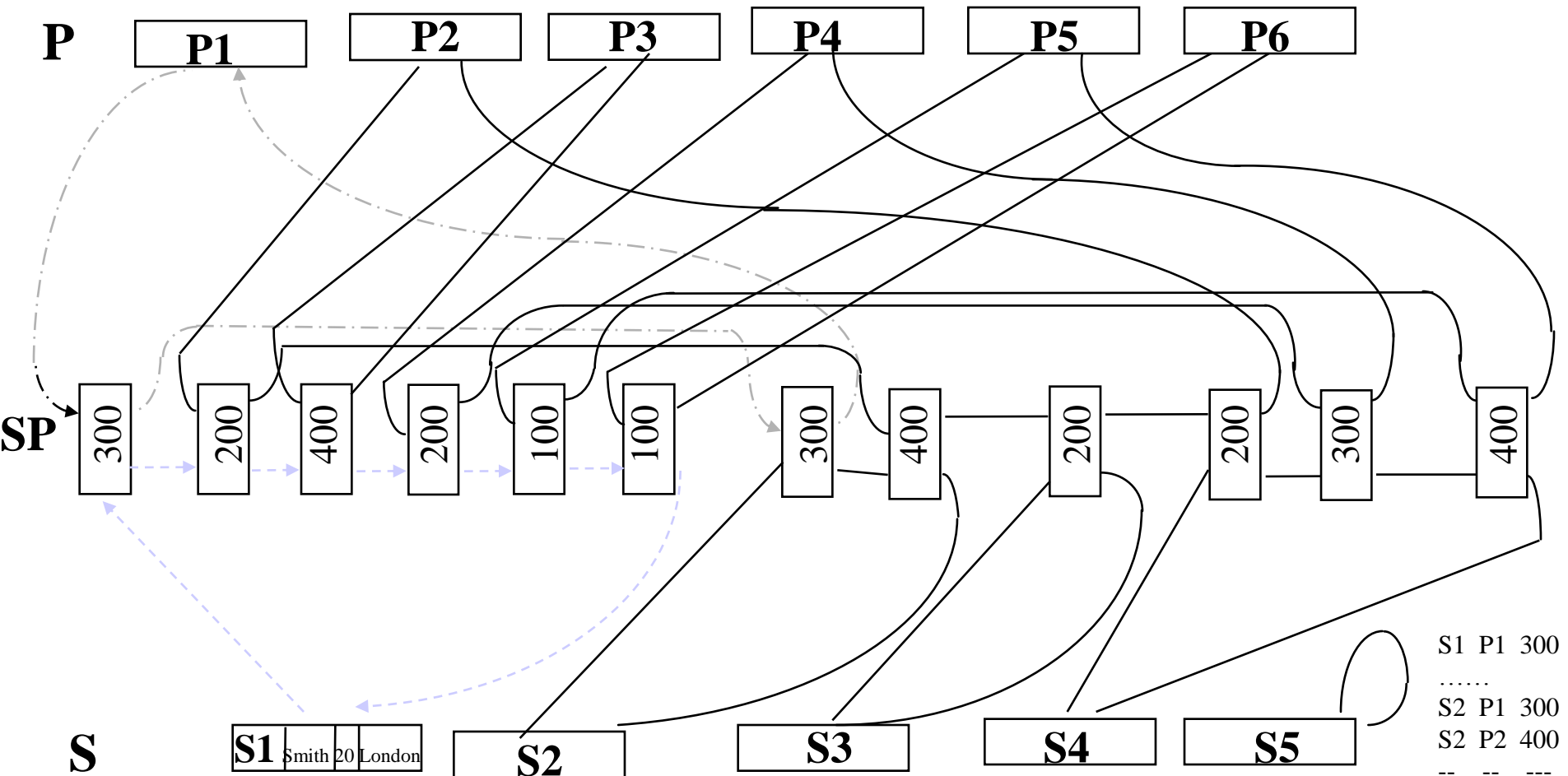
# The Network Model: Sets and Structure

## □ Network Model

- two sets {
  - a set of records (**Record types**): entities
  - a set of links (**Link types**): relationship
- Structure
- Sample values:
  - occurrences, data, records
  - (Ref. next page)
- Note:
  - Three **record types**: S, P, SP
  - Two **link types**: S-SP, p-SP



# The Network Model: Sample Values



S1 P1 300  
 .....  
 S2 P1 300  
 S2 P2 400  
 .. .. .

(p. 2-5)



# The Network Model: Data Manipulation

---

- Data Manipulation

- locate a specific record
- move from a parent to its first child
- move from one child to the next
- 
- 
- 
- create a new record
- delete a new record
- update a new record
- connect a child into a link

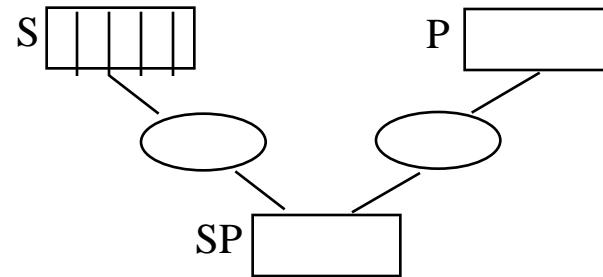
- Data Integrity

**Rule:** A child can not be inserted unless its parent already exists

# 5.3 IDMS: Overview

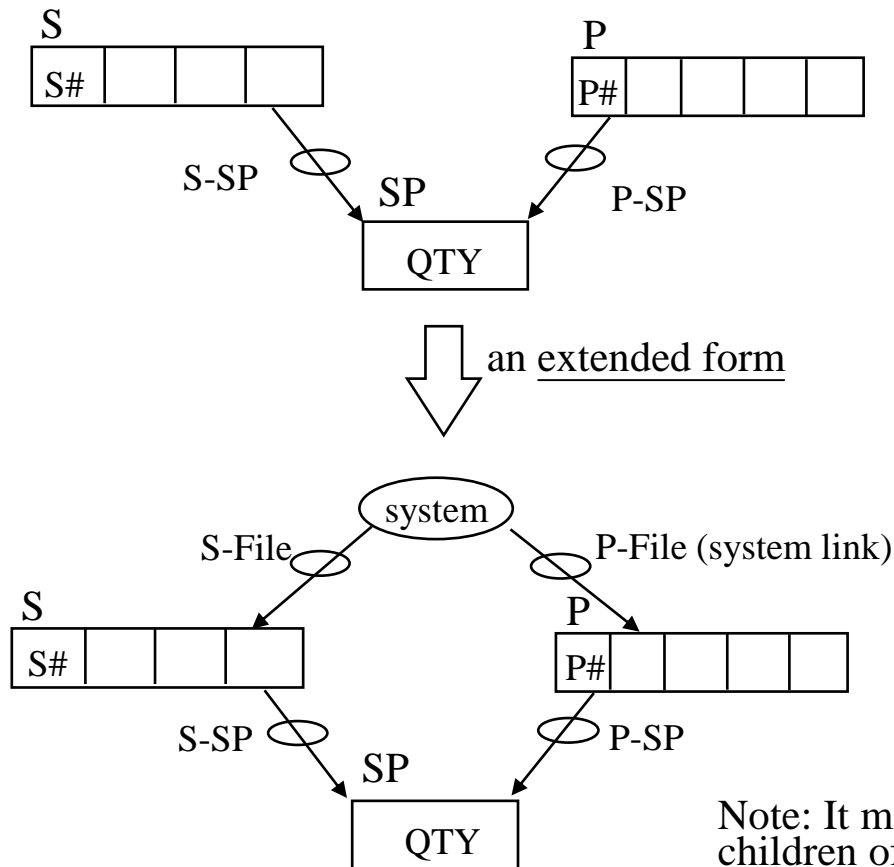
---

- Runs on IBM mainframes; a product of Cullinet Software
- The best known example that based on **DBTG** (The CODASYL DataBase Task Group)
- An **IDMS** database is defined by DMS Schema DDL (*Data Definition Language*)
- Schema defines:
  - (1) records in the database
  - (2) elements (i.e. fields)
  - (3) sets (i.e. links)
    - owner (parent)
    - member (child)



# IDMS: Data Structure

- Consider the data structure of the "suppliers-and-parts" database:



Note: It may be convenient regarding all roots as children of a hypothetical system record.

# IDMS: Schema for Suppliers-and-parts

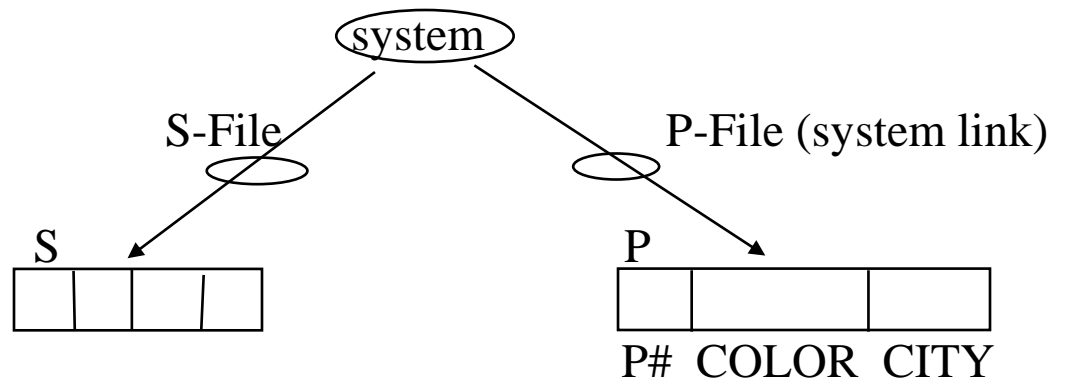
---

1 SCHEMA NAME IS SUPPLIERS-AND-PARTS  
2 RECORD NAME IS S.  
3 LOCATION MODE IS CALC USING S#  
4           DUPLICATES NOT ALLOWED.  
5     02 S#           PIC X(5).  
6     02 SNAME    PIC X(20).  
7     02 STATUS   PIC 999 USAGE COMP-3.  
8     02 CITY     PIC X(15).  
  
9 RECORD NAME IS P.  
10 LOCATION MODE IS CALC USING P#  
11           DUPLICATES NOT ALLOWED.  
12     02 P#           PIC X(6).  
13     02 PNAME   PIC X(20).  
14     02 COLOR   PIC X(6).  
15     02 WEIGHT   PIC 999 USAGE COMP-3.  
16     02 CITY     PIC X(15).  
  
17 RECORD NAME IS SP.  
18 LOCATION MODE IS VIA S-SP SET.  
19     02 QTY       PIC 99999 USAGE COMP-3.

20 SET NAME IS S-SP.  
21 ORDER IS NEXT.  
22 OWNER IS S.  
23 MEMBER IS SP OPTIONAL MANUAL.  
  
24 SET NAME IS P-SP.  
25 ORDER IS NEXT.  
26 OWNER IS P.  
27 MEMBER IS SP OPTIONAL MANUAL.  
  
28 SET NAME IS S-FILE.  
29 ORDER IS SORTED.  
30 OWNER IS SYSTEM.  
31 MEMBER IS S MANDATORY AUTOMATIC  
32                    ASCENDING KEY IS CITY.  
  
33 SET NAME IS P-FILE.  
34 ORDER IS SORTED.  
35 OWNER IS SYSTEM.  
36 MEMBER IS P MANDATORY AUTOMATIC  
37                    ASCENDING KEY IS COLOR.

# IDMS: Subschema “View”

- Subschema name  
↙
- 1 ADD SUBSCHEMA NAME IS **S-AND-P-ONLY**
  - 2       OF SCHEMA NAME IS **SUPPLIERS-AND-PARTS**.
  
  - 3 ADD RECORD **S**.
  - 4 ADD RECORD **P**
  - 5       ELEMENTS ARE
  - 6             **P#**
  - 7             **COLOR**
  - 8             **CITY**.
  
  - 9 ADD SET **S-FILE**.
  - 10 ADD SET **P-FILE**.



# IDMS: Data Manipulation

---

- **Working area:**

COBOL

⋮

record description



}

for each subschema record type to be processed

(i.e. UWA working area in DBTG)

⋮

77 ICB (IDMS Communication Block) (ref 4-9)



(similar to SQLCA in DB2)



error-status

- **Currency indicator:** similar to current of cursor in SQL, and current position in IMS

# IDMS: Data Manipulation (cont.)

---

## ■ Selected commands:

- **Bind:** Associates IDMS record types and control blocks with space
- **Ready:** Prepares database areas for processing
- **Commit:** Effects a checkpoint for recovery procedures
- **Rollback:** Requests recovery of the database
- **Finish:** Releases database areas
- **Find:** Locates a record occurrence in the database
- **Get:** Delivers a record occurrence to variable storage
- **Store:** Adds a record occurrence to the database
- **Modify:** Rewrites a record occurrence in the database
- **Connect:** Links a record occurrence to a set
- **Disconnect:** Dissociates a record occurrence from the database
- **Erase:** Deletes a record occurrence from the database
- **If:** Tests whether a set is empty or whether a record occurrence is a member
- **Keep:** Locks a record occurrence against access or update by another run unit
- **⋮**