CONCEPTUAL DESIGN
aggregation

We can find the aggregate object often as a verb conjugation. For example:

'a student registers for a course' → registration
'a person sells a house on a certain date' → sale
'a company supplies a product on a date' → supply
'a family moves house on a certain date' → removal.

There are cases where the aggregate object cannot be inferred from the conjugation of the word, as in

'a student gets a mark for an examination' → result.

The type definitions for these examples could be:

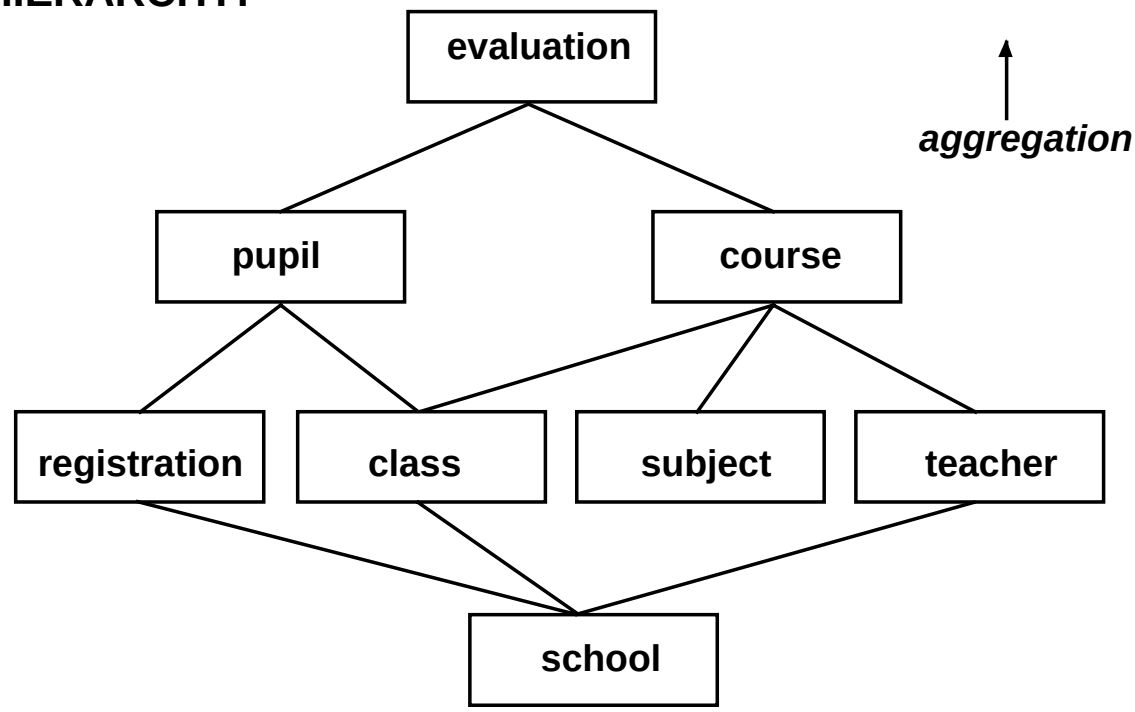
***type* registration = student, course**
***type* sale = person, house, date**
***type* supply = company, product, date**
***type* removal = family, house, date**
***type* result = student, subject, mark.**

AGGREGATION
definition and hierarchy

TYPE DEFINITIONS:

type **pupil**	**= registration, class**
type **teacher**	**= name, address, town, school**
type **class**	**= school, course year**
type **school**	**= name, address, town, headmaster**
type **subject**	**= description, curriculum**
type **evaluation**	**= pupil, course, mark**
type **course**	**= subject, teacher, class**
type **registration**	**= name, address, town, birth_date, birth_place, school.**

HIERARCHY:



CONCEPTUAL DESIGN
generalization

type **vehicle = make, trade mark, wheel base,**
cylinder volume, power, trailer coupling,
trailer brake.

vehicle	make	trade mark	wheel base	cylinder volume	power	trailer coupling	trailer brake
1040	BUICK	APOLLO	283	5730	145	-	-
1170	ROMEO	ALFA 6	260	2492	116	-	-
1002	ATM	OKFAL	750	-	-	pin	no
1004	DAF	FT 1600	265	6170	153	-	-
0401	PKA	KAD 16	525	-	-	ring	yes

This is a source of errors because:

- **the description of real world objects is incomplete if relevant values are not specified;**
- **the description of real world objects is wrong if irrelevant values are specified.**

CONCEPTUAL DESIGN
generalization (continued)

***type* motor vehicle** **= make, trade mark, wheel base, cylinder volume, power**

***type* trailer** **= make, trade mark, wheel base, trailer coupling, trailer brake.**

motor vehicle	make	trade mark	wheel base	cylinder volume	power
1040	BUICK	APOLLO	283	5730	145
1170	ROMEO	ALFA 6	260	2492	116
1004	DAF	FT 1600	265	6170	153

trailer	make	trade mark	wheel base	trailer coupling	trailer brake
1002	ATM	OKFAL	750	pin	no
0401	PKA	KAD 16	525	ring	yes

DISADVANTAGES:

- **RETRIEVALS OF GENERAL VEHICLE CHARACTERISTICS REQUIRE TWO TYPES.**
- **RELATIONSHIPS WITH VEHICLES DIFFICULT TO EXPRESS.**

GENERALIZATION

type **vehicle** **= make, trade mark, wheel base**
type **motor vehicle** **= [vehicle], cylinder volume, power**
type **trailer** **= [vehicle], trailer coupling, trailer brake.**

vehicle	make	trade mark	wheel base
1040	BUICK	APOLLO	283
1170	ROMEO	ALFA 6	260
1002	ATM	OKFAL	750
1004	DAF	FT 1600	265
0401	PKA	KAD 16	525

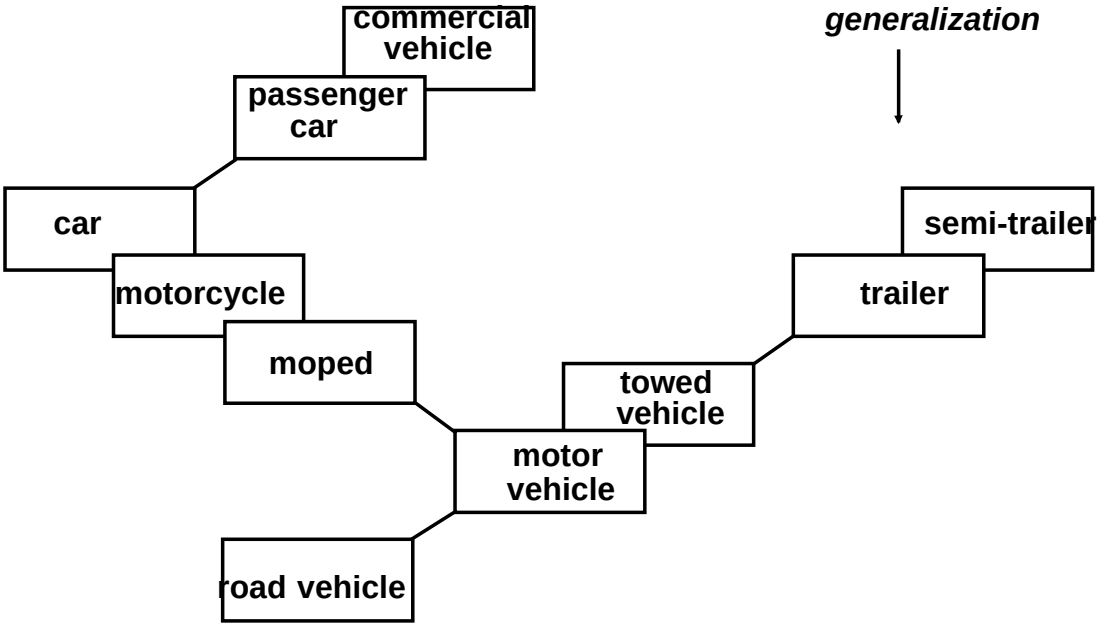
motor vehicle	vehicle	cylinder volume	power
M200	1040	5730	145
M327	1170	2492	116
M410	1004	6170	153

trailer	vehicle	trailer coupling	trailer brake
T108	1002	pin	no
T110	0401	ring	yes

GENERALIZATION
hierarchy

The generalization hierarchy is to be interpreted as follows:

1. **Road vehicle is specialized into motor vehicle and towed vehicle.**
2. **Towed vehicle is specialized into trailer and semi-trailer.**
3. **Motor vehicle is specialized into moped, motorcycle and car.**
4. **Car is specialized into passenger car and commercial vehicle.**

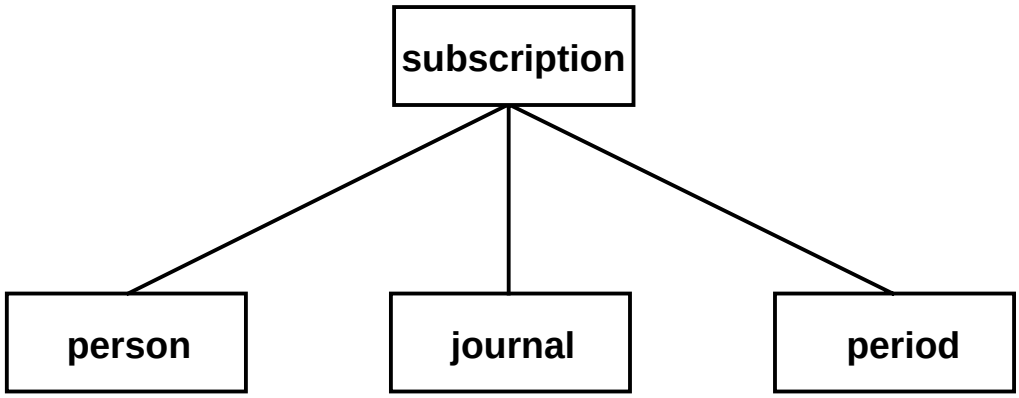


PUBLISHING COMPANY
the problem of decomposition

'persons subscribe to a journal for a certain period'.

Type definition:

***type* subscription = person, period, journal.**



Candidates for decomposition are person and journal:

***type* journal= title, editor, circulation.**
***type* person= name, address, postal code, town.**

New subscriptions require:
address, postal code, town, period and journal.

Necessary identifications are generated.

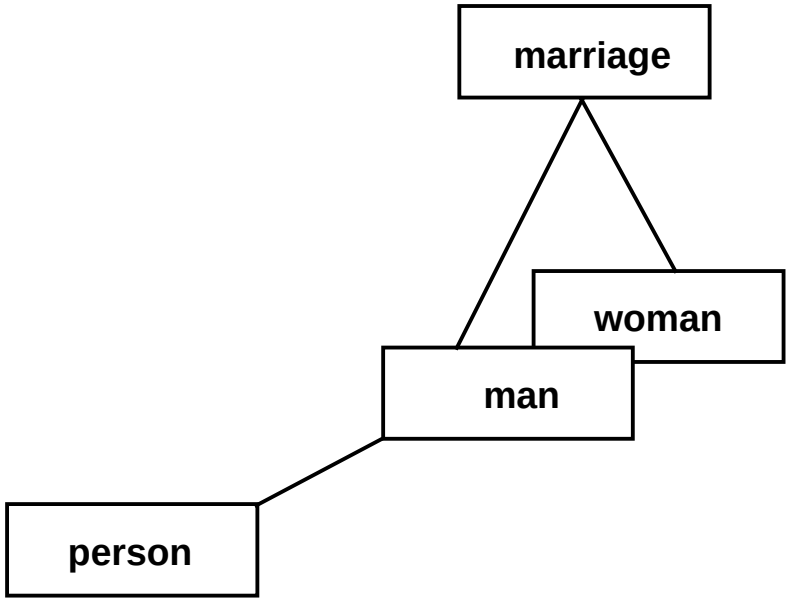
Identification generation and system boundary:
- **Journal within: No problems.**
- **Person beyond: Problems.**

CONCLUSION:

***type* subscription = name, address, postal code, town, period, journal**
***type* journal = title, editor, circulation.**

MARRIAGE
convertibility and generalization

ABSTRACTION HIERARCHY:



TYPE DEFINITIONS:

***type* person** **= name, address, town, birth_date,**
birth_place, sex

***type* man** **= [person]**

***type* woman** **= [person]**

***type* marriage** **= man, woman.**

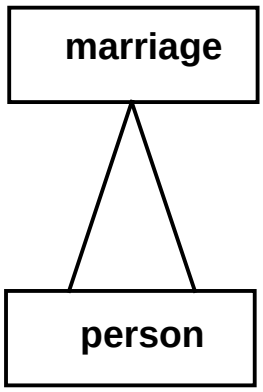
DEFINITIONS ARE NOT CONVERTIBLE !!

MARRIAGE (continued)

SOLUTION:

***type* person = name, address, town, birth_date,**
birth_place, sex
***type* marriage = man_person, woman_person.**

ABSTRACTION HIERARCHY:



EVENTUALLY A STATIC CONSTRAINT:

***assert* marriage *its* admitted (*true*) =**
man_person *its* sex = "m" *and*
woman_person *its* sex = "w".

UNIVERSITY REGISTRATION SYSTEM
system boundary and generalization

TYPE DEFINITIONS:

***type* student** = name, address, town, birth_date, registration_date, faculty

***type* employee** = name, address, town, birth_date, employment_date, department, rank.

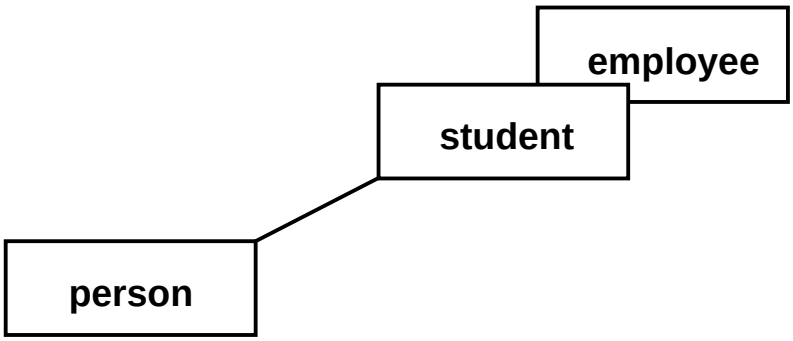
GENERALIZATION:

***type* person** = name, address, town, birth_date

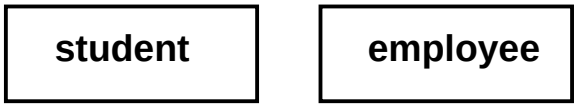
***type* student** = [person], registration_date, faculty

***type* employee** = [person], employment_date, department, rank.

ABSTRACTION HIERARCHY:



SYSTEM BOUNDARIES ARE CROSSED, SOLUTION IS:



ORIGINAL TYPE DEFINITIONS REMAIN UNCHANGED !!

OPEN UNIVERSITY
system boundary and generalization (continued)

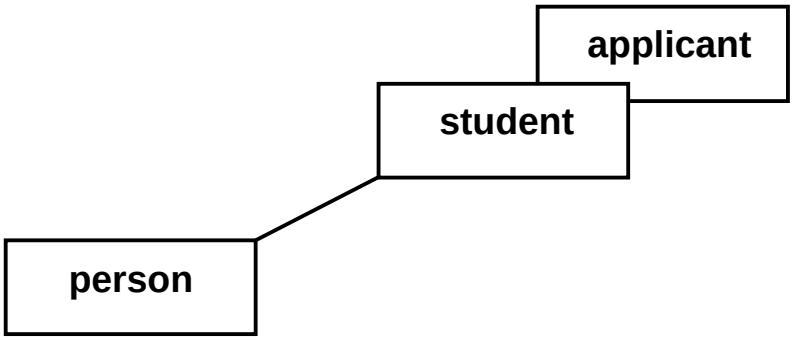
TYPE DEFINITIONS:

***type* student** = name, address, town, birth_date, registration_date, study
***type* applicant** = name, address, town, response_date, study.

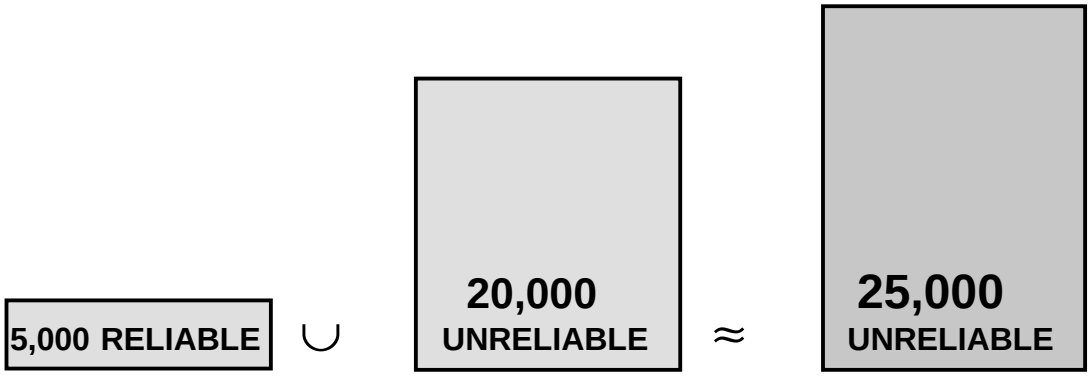
GENERALIZATION:

***type* person** = name, address, town, study
***type* student** = [person], registration_date, birth_date
***type* applicant** = [person], response_date.

ABSTRACTION HIERARCHY:



VULNERABLE DATABASE:



CONCEPTUAL DESIGN
vehicles

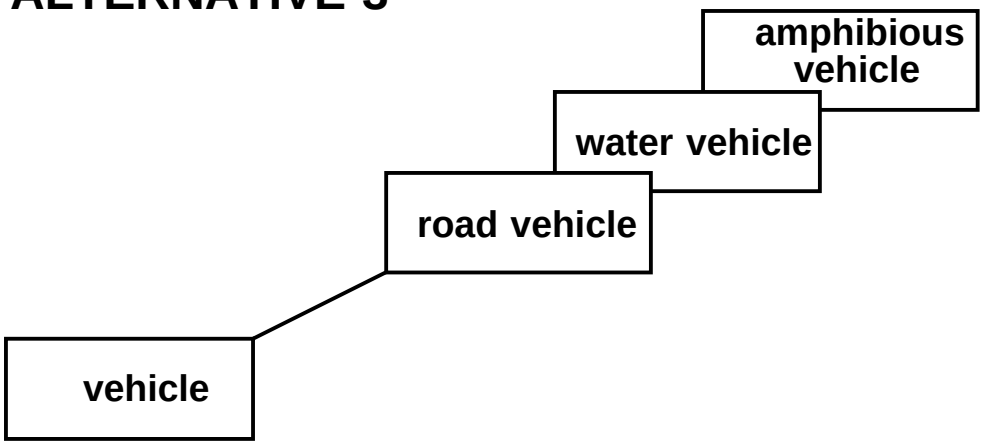
- **ALTERNATIVE 1**

DISADVANTAGE: NULL VALUES.

- **ALTERNATIVE 2**

DISADVANTAGE: NO INTERRELATIONSHIPS.

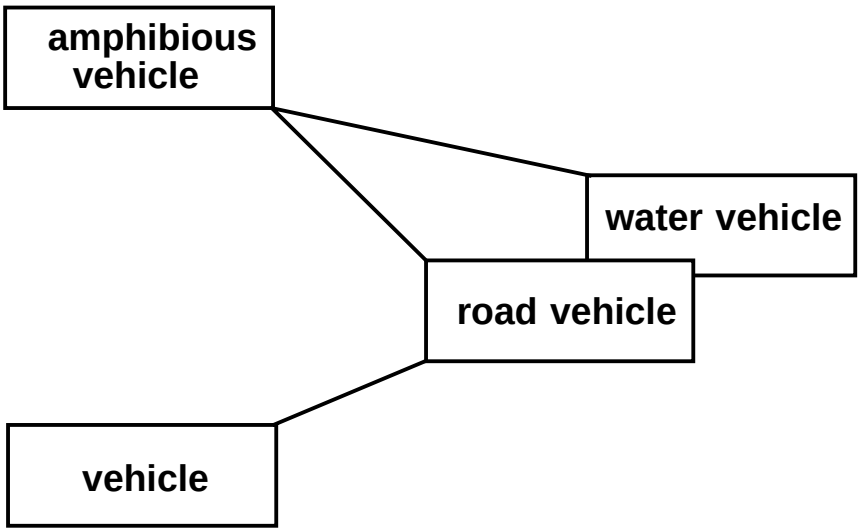
- **ALTERNATIVE 3**



PROBLEM: AMPHIBIOUS VEHICLE IS ROAD VEHICLE AND IS WATER VEHICLE.

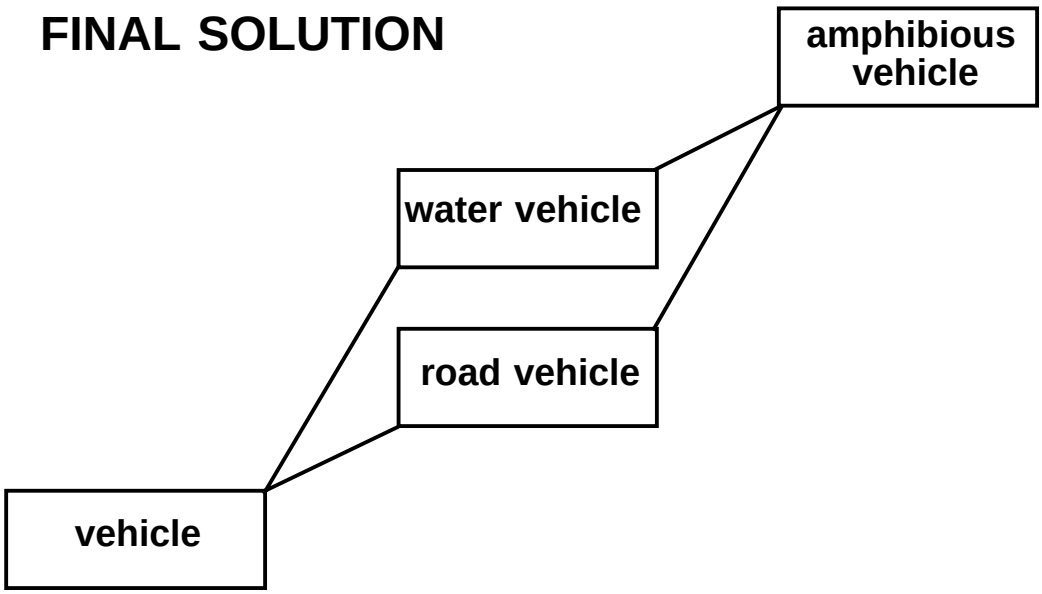
CONCEPTUAL DESIGN
vehicles (continued)

- **ALTERNATIVE 4**



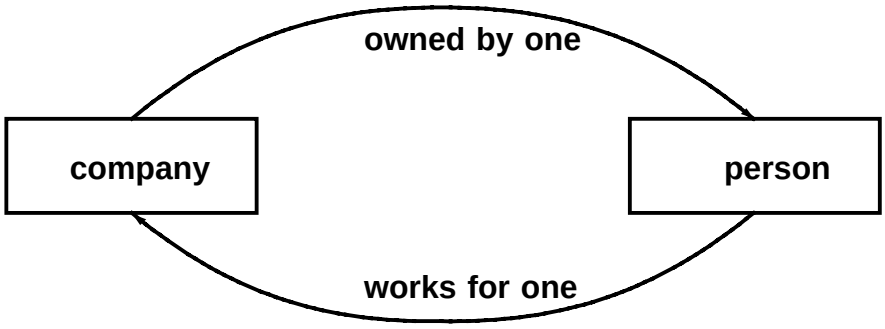
PROBLEM: DISJOINT WATER VEHICLE AND ROAD VEHICLE RESULT IN EMPTY AMPHIBIOUS VEHICLE.

- **FINAL SOLUTION**



CONCEPTUAL DESIGN
exercise: cyclic relationship

Given the cyclic relationships below, analyze the functional relationships and provide a non-cyclic design. Also represent the solution by an abstraction hierarchy.



CONCEPTUAL DESIGN
exercise: training institute

A training institute plans courses by using a database. Courses are presented by lecturers. Assume information on which courses lecturers prepared to give is available. Courses are named and have given durations.

As soon as the venue and date of planned courses have been determined, the information is stored in the database. Students can apply for a planned course from this moment onwards. Names and addresses of participants are recorded, but the training institute has no intention of keeping a student administration. Planned courses are canceled if the number of registrations is below a given minimum. The number of course participants is limited.

Course planning includes the appointment of lecturers. Preference is given to lecturers prepared to conduct courses, but other lecturers my be appointed as well. The ultimate registration is the identity of the lecturer for a given course.

a Provide the semantic model for the institute's database. Provide the type definitions and the corresponding abstraction hierarchy.

b The relationship between planned courses and courses can be indicated by both aggregation and generalization. Provide the requirements for both cases.

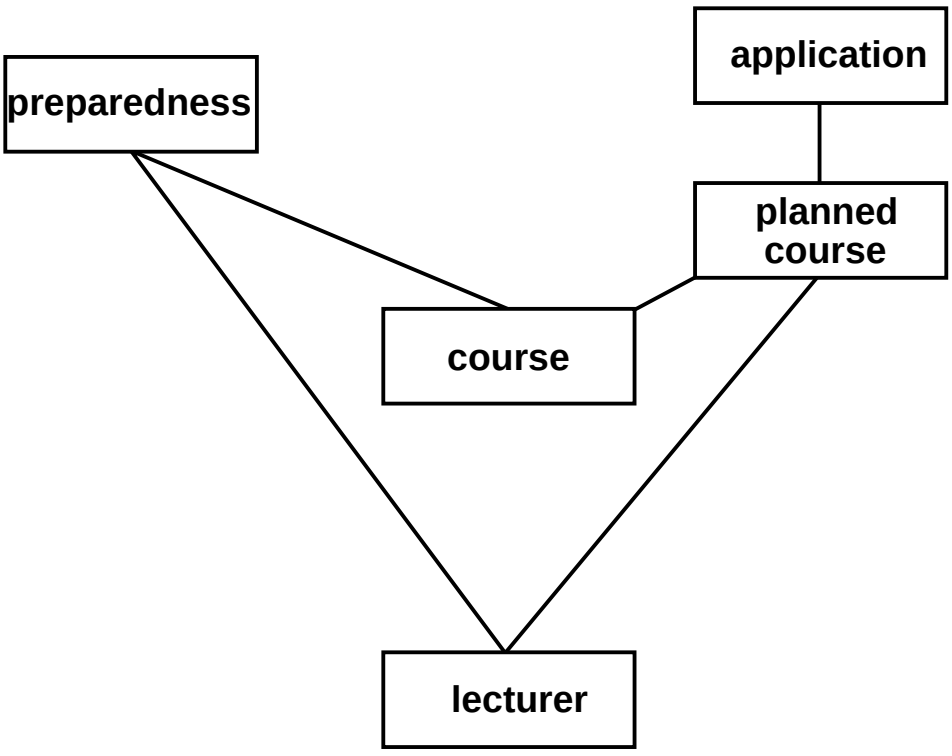
c Assume that only lecturers willing to run a course can be scheduled. Can this limitation be specified by a static or dynamic constraint? Which type definitions and constraints must be given under these circumstances?

CONCEPTUAL DESIGN
model: training institute

TYPE DEFINITIONS:

type **lecturer**	**= name, address, town, salary**
type **course**	**= name, limit**
type **preparedness**	**= lecturer, course**
type **planned course**	**= [course], town, date, minimum, lecturer**
type **application**	**= name, address, town, planned course.**

ABSTRACTION HIERARCHY:



CONCEPTUAL DESIGN
exercise: camping

A camping supervisor offers bungalows and pitches for tents and caravans. He has commissioned an information analyst to design a database for his rental administration.

A pitch can accommodate either a tent or a caravan; the relevant data are surface area in m^2 and daily rental. Bungalow data include location and bungalow category. The surface area of each bungalow (in m^2) and the maximum number of beds is recorded for categories. A weekly rental per person is set for each category, the tariff being season dependent (i.e. week number).

A pitch or a bungalow is rented out to one person during an agreed period, which is defined by start and end dates. One person renting a pitch or a bungalow acts as a contact for the group staying at the camping during the rental period.

Each person renting a pitch or a bungalow for the first time is allocated a rent identification, which is referenced when the person applies for a pitch or bungalow at a subsequent occasion. The renter's name, address and town are registered. The number of adults and children is registered for each group.

An occupancy code is used to determine whether the pitch is occupied by a tent or a caravan during the rental period.

The facilities available at a pitch are indicated by a facility code (e.g. electricity, water, sewage).

a Provide the semantic model for the camping supervisor's database. Provide the type definitions and the abstraction hierarchy. Aggregations should be used only.

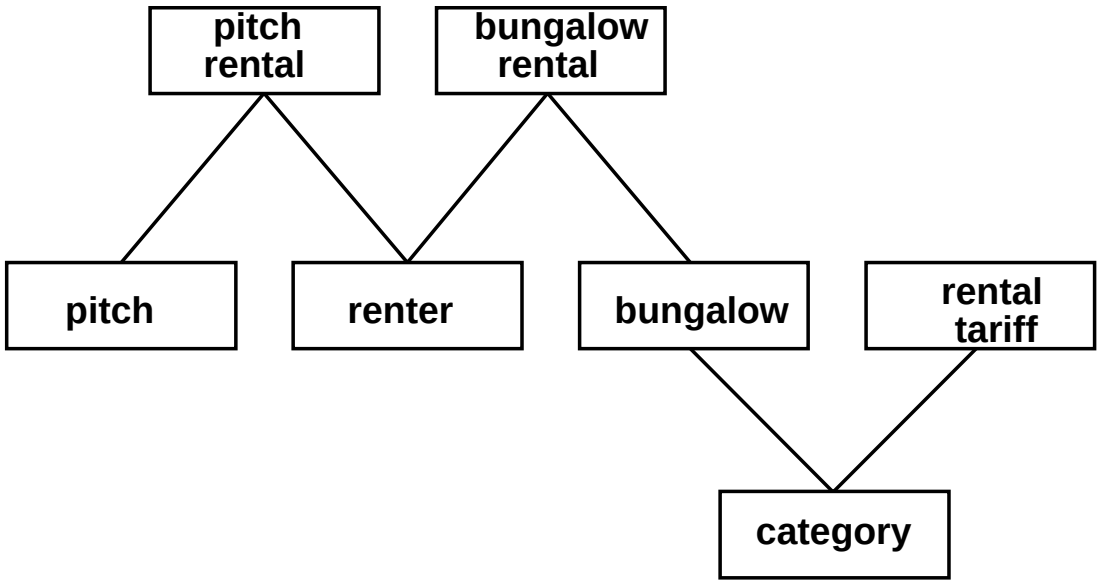
b Would it be meaningful to apply generalizations to this database?

CONCEPTUAL DESIGN
model: camping

TYPE DEFINITIONS:

type **pitch**	**= pitch_surface, rental, facility_code**
type **bungalow**	**= location, category**
type **category**	**= bungalow_surface, max_beds**
type **rental tariff**	**= category, week, tariff**
type **bungalow rental**	**= bungalow, renter, start_date, end_date, child_number, adult_number**
type **pitch rental**	**= location, renter, start_date, end_date, child_number, adult_number, occupancy_code**
type **renter**	**= name, address, town.**

ABSTRACTION HIERARCHY:



CONCEPTUAL DESIGN
exercise: garage

The management of a garage wishes to monitor the sale of cars by their salesmen and the maintenance of the cars. A database is designed, containing data about salesmen, sales and maintenance of sold cars.

Personal data, employment date and the number of cars sold the year to date are recorded for each salesman. The following data are recorded for sold cars: recommended price, actual price, price of the exchange car, sale date, registration, make, type, the buyer (owner) and the salesman. The exchange price is valued zero if there was no exchange car.

The garage is served by a number of workshops for maintenance jobs. The number of mechanics, apprentice mechanics, floor space and data on available technical equipment is recorded. The following data are recorded for sold cars requiring maintenance at a given workshop: date, mileage and repairs carried out. Each repair is identified by a description, average repair time and average price. The cost of certain quantities of materials required for repairs are recorded in the database.

a **Provide the semantic model for the garage's database. Provide the type definitions and the abstraction hierarchy.**

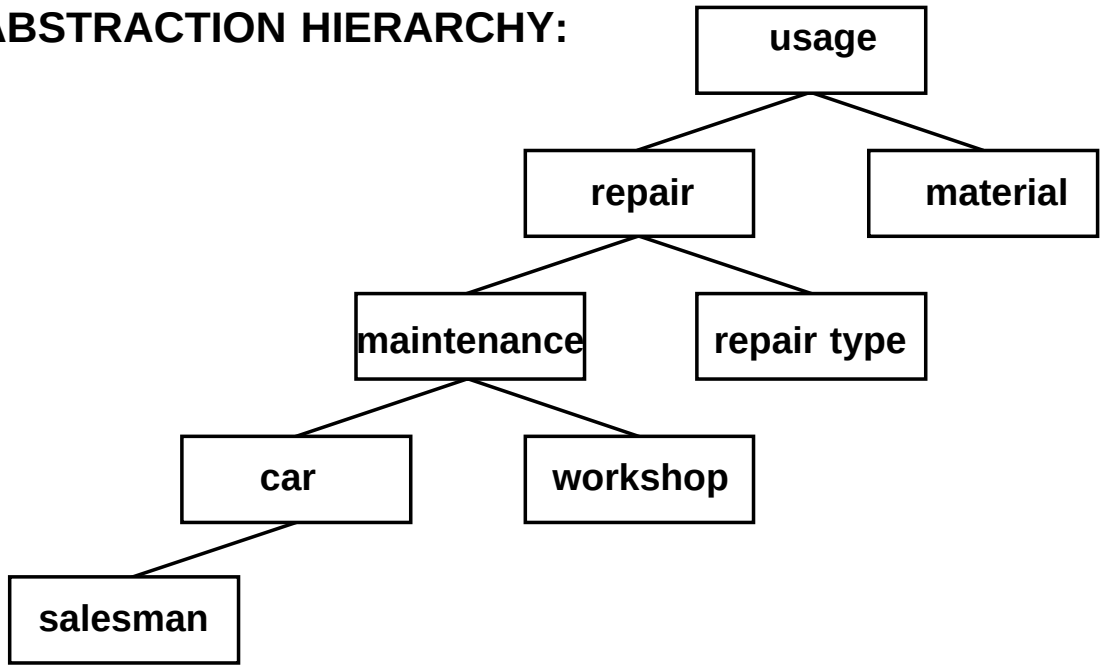
b **Assume that maintenance data on other cars have to be recorded as well. Assume further that of these cars only the registration, type and owner are known. Describe the consequences of the extension to the semantic model for this application. Provide new type definitions and a new abstraction hierarchy.**

CONCEPTUAL DESIGN
model: garage

TYPE DEFINITIONS:

type **salesman**	**= name, address, town, employment_date, sold_number**
type **car**	**= make, type, recommended_price, actual_price, exchange_price, sale_date, owner, salesman**
type **workshop**	**= mechanic_number, apprentice_number, floor_space, technical_equipment**
type **maintenance**	**= car, workshop, date, mileage**
type **repair type**	**= description, avg_price, avg_time**
type **material**	**= description, price**
type **usage**	**= repair, material**
type **repair**	**= maintenance, repair_type, actual_time.**

ABSTRACTION HIERARCHY:



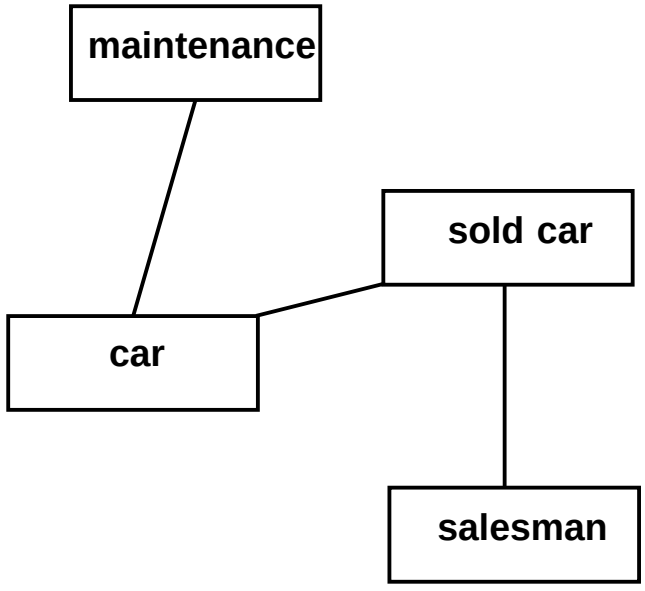
CONCEPTUAL DESIGN
model: garage (continued)

The extension implies a differentiation between cars and sold cars.

MODIFIED TYPE DEFINITIONS:

***type* car = make, type, owner**
***type* sold car = [car], recommended_price, actual_price, exchange_price, sale_date, salesman.**

MODIFIED ABSTRACTION HIERARCHY:



CONCEPTUAL DESIGN
the data dictionary

CONCEPTUAL DATABASE MODEL:

type X = A, B, P_C
type Y = [X], D
type Z = X, Y
type A
type B
type C
type D

composite types

base types

MODEL OF THE DATA DICTIONARY:

type type = name
type attribute = composite_type, type, kind
type role attribute = [attribute], prefix

type	name
1	A
2	B
3	C
4	D
5	X
6	Y
7	Z

attr.	comp. type	type	kind
1	5	1	aggr.
2	5	2	aggr.
3	5	3	aggr.
4	6	4	aggr.
5	6	5	gen.
6	7	5	aggr.
7	7	6	aggr.

role attr.	attr.	prefix
1	3	P

THE DATA DICTIONARY IS SELF DESCRIBING !!

CONCEPTUAL DESIGN
exercise: MOVIE CLUB

A company has a movie club with members within the company (i.e. employees) and outsiders. Internal address and internal telephone number are relevant data for employee members. The board of the movie club consists of a *fixed* number of officials: chairman, secretary, treasurer, editor of the magazine and one ordinary member. Officials are always club members. The board is in charge for a certain period of time. Club members own a number of movies, which can be shown on club meetings. For each movie are relevant: title, owner, genre, duration and system (8 mm or 16 mm).

The secretary needs the name, address and private telephone number of each member. For visits to other clubs it is relevant how many passengers each member can take. The treasurer needs also data about overdue subscription payments.

a. Provide the semantic model for the movie club's database. Give type definitions and abstraction hierarchy.

b. The movie club wishes to collect data about video movies for the VHS system. In what way does the solution need adaptation?

CONCEPTUAL DESIGN
exercise: PURCHASE

A database contains data about suppliers, articles, offerings and supplies. The semantic model of this database is as follows:

***type* article = description, selling_price.**
***type* supplier = name, address, city, credit_limit.**
***type* offering = supplier, article, price.**
***type* supply = supplier, article, unit_price, number, date.**

Definitions of base types (as name, price, ...) have been omitted in the definition above.

a. This database contains redundancy because unit price is derivable.

Is this proposition true?
I. yes
II. no

b. This database leads to update anomalies because supply is related to offering.

Is this proposition true?
I. yes
II. no

c. This database is conform the structural integrity rules.

Is this proposition true?
I. yes
II. no

CONCEPTUAL DESIGN
exercise: RECORD LIBRARY

A record library holds 8000 records for lending out to members. A record is a long-playing record (LP) or compact disk (CD). For both types are required: title, performer (group or individual artist) and a fee (payment for lending). Two different fees are used: one for LP's and one for CD's.

The following type definitions have been designed for the administrative system:

***type* LP = title, performer, fee.**
***type* CD = title, performer, fee.**

a. Why do these type definitions not fulfil the requirements of the semantic data model?

b. Is it meaningful to distinguish specializations? Explain briefly why.

c. Provide the correct model for this situation. Give type definitions and abstraction hierarchy.

d. How many instances according the new model contain a fee?