Unit 11

Case Study
Bus Company Information System

Outlines

Step 1: Join data and functional analysis

Step 2: Analysis important operations

Step 3: High-level logical design

Step 4: Logical design for relational model

Step1:
Join data and functional analysis

1. Drawing the skeleton F-schema
2. Drawing the skeleton D-schema
3. Repeating the refinements of the D-schema and the F-schema to a complete ones

Requirements

The company is subdivided into two major areas: *passenger management* and *service management*. Passenger management deals with reservations of bus seats and with providing information to passengers and travel agencies about schedules and fares. Service management deals with organizing ordinary trips, special trips, managing buses, and bus drivers.

Passenger Management

Seat reservations are made directly by passengers or by travel agencies; they can be made on any bus service (both ordinary and special) and are free. The passengers can ask for a smoking or nonsmoking area. Reservations can be made for segments of the bus trip; thus, the same seat might be made available to different persons on different parts of the same trip. The office holds the trip description sheets, which are prepared about four weeks before the actual beginning of the trip; they include the schedule of the trip, the list of all intermediate stops, and spaces for each seat, where it is possible to write reservations. Trip description sheets are prepared by the service management office and used by the reservation *office* to record the reservations. They are returned to the service management office at least 2 hours before departure; thus, reservations cannot be taken immediately before the departure.

The main office has a reservation desk and some dedicated telephone lines for reservations. Fare information is given at a different office, called the *fares office*, which also has dedicated telephone lines. Fares are reported in a *fare description booklet*, which is prepared monthly; fares are relatively stable. Office employees and operators on either of the telephone lines cannot answer questions concerning the other office; at most, they can redirect clients to the other desk or switch telephone calls. Thus, we distinguish two separate offices for reservations and fares.

Typically, passengers also request information about *schedules* of trips. This information is prepared by the service management office, but it is much more variable; schedules can be modified depending on road conditions or special events. Special trips can be organized for special events. Thus, the collection of information about schedules is centralized within a third office, called the *schedule office*, which supports the other two offices. Questions about schedules from passengers can be answered directly at the reservation and fare offices, because the employees of these offices can use the schedule or the trip description sheets. Modifications to the schedules of trips are communicated to the schedule office, which is responsible for updating the trip description sheets (and some-times for communicating changes to passengers holding reservations).

Passengers can enroll in a *frequent traveler* program; those who do earn a bonus when they accumulate a given number of miles.

Service Management

We distinguish four major offices within the service management area, concerned with ordinary trips, special trips, bus management, and management of bus drivers. The *ordinary trips* management office prepares the trip description sheets and the fare description booklet. Every season, ordinary trips are rearranged to deal with different requirements of travelers. Officers of the road condition service and of the government give information that can be used to modify the trips.

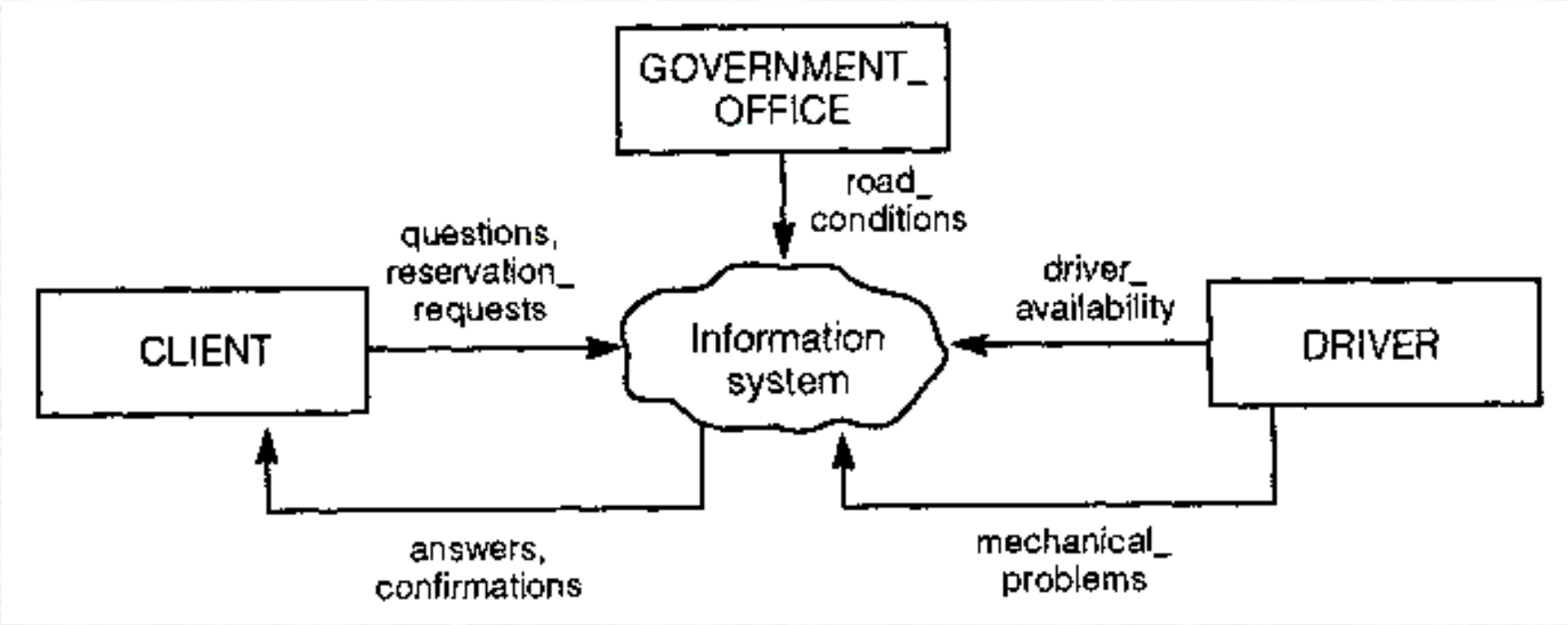
Sometimes, on the basis of external events (such as football games, elections, fairs, special holidays, etc.), the company decides to activate special trips; when this occurs, the *special trip management* office produces a trip description sheet. It then has the responsibility of communicating information on special trips to the reservation and schedule offices.

The *bus management* office has the goal of organizing each individual trip. Specifically, it receives the trip description sheets with passengers' reservations written on them and adds the name of the bus driver, the make and the license number of the bus, and the maximum number of passengers that can be accepted. Fares are also reported for each segment of the trip; thus, all the information needed by the bus driver is recorded on the trip description sheet.

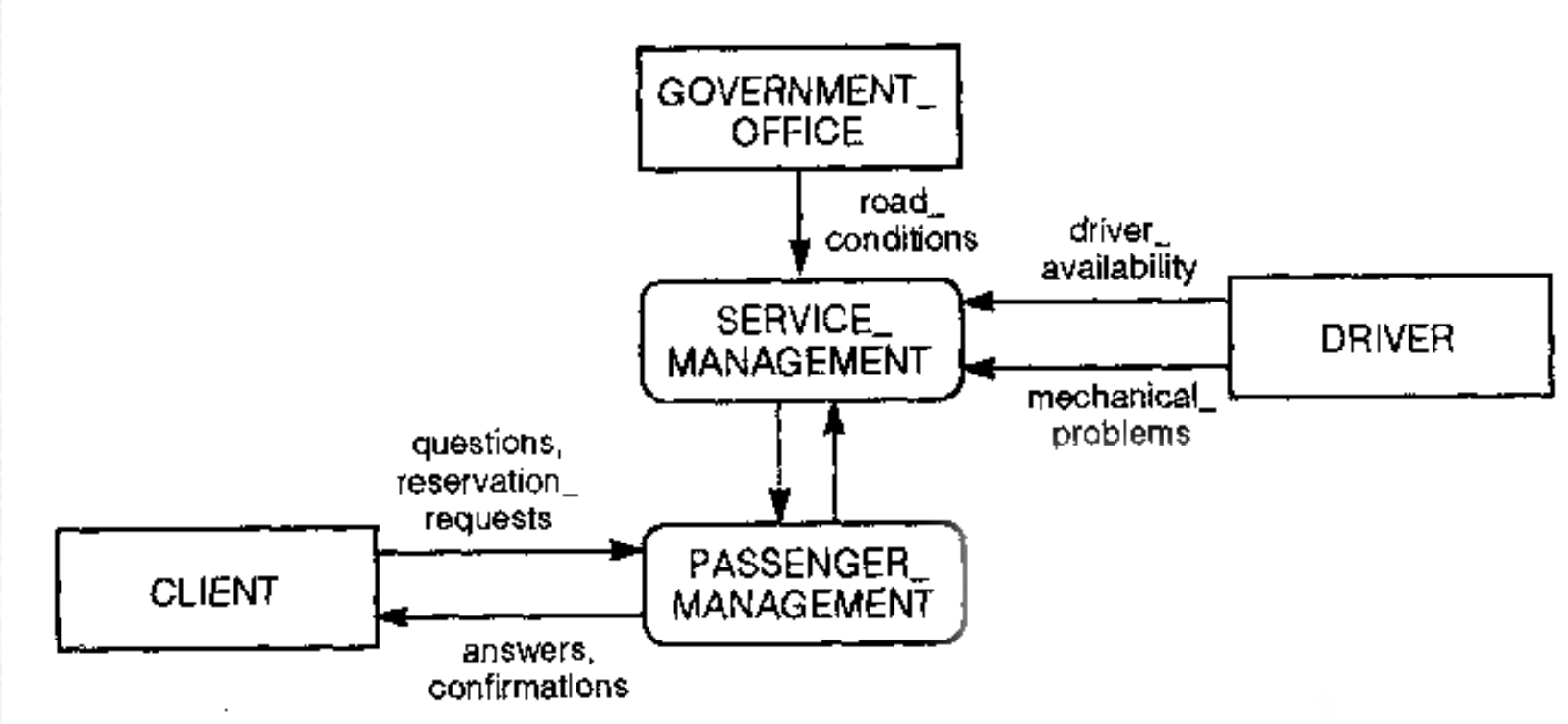
The other activities of bus management include dealing with the mechanical problems of buses, their purchase and sale, the repair of damages, and so on. The office maintains buses' documents and a (paper) file on each bus, recording repairs, maintenance, and usage of the bus.

The *driver management* office keeps a file of individual data on each driver. When a driver is sick, he communicates the absence to this office. Similarly, the driver communicates to this office all the mechanical problems that he discovers on his bus. In practice, this office provides assistance to drivers and controls them.

Context Diagram



Level 1 DFD (F-Schema skeleton)



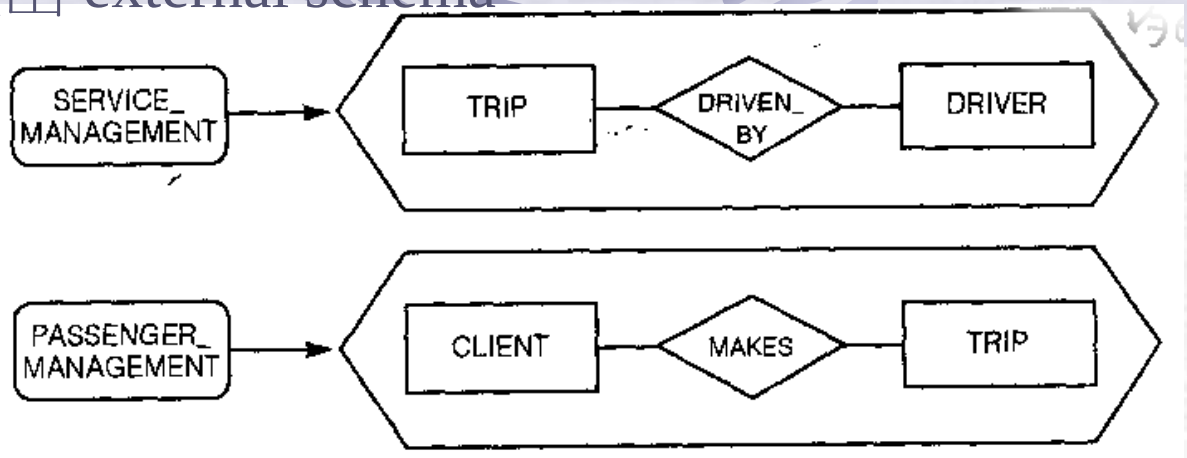
Service management 安排旅程

管理班車與司機.

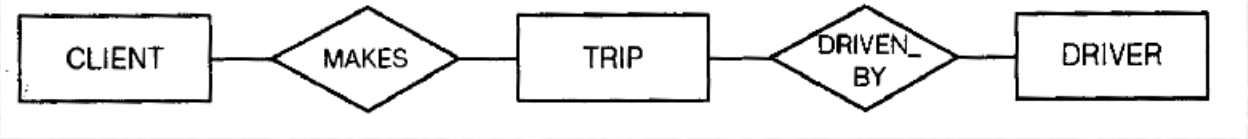
Passenger management 處理預定座位

包括提供時間表與費率.

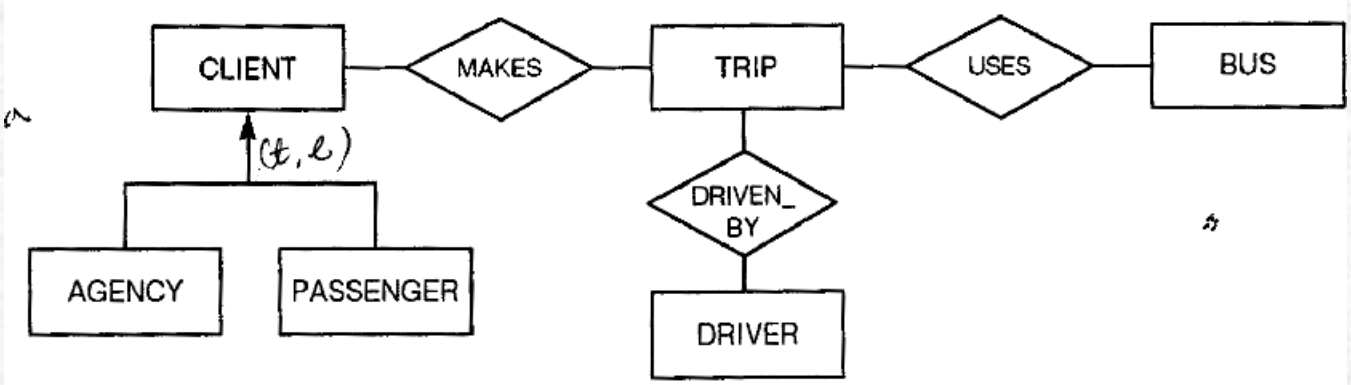
由 process 畫出 external schema



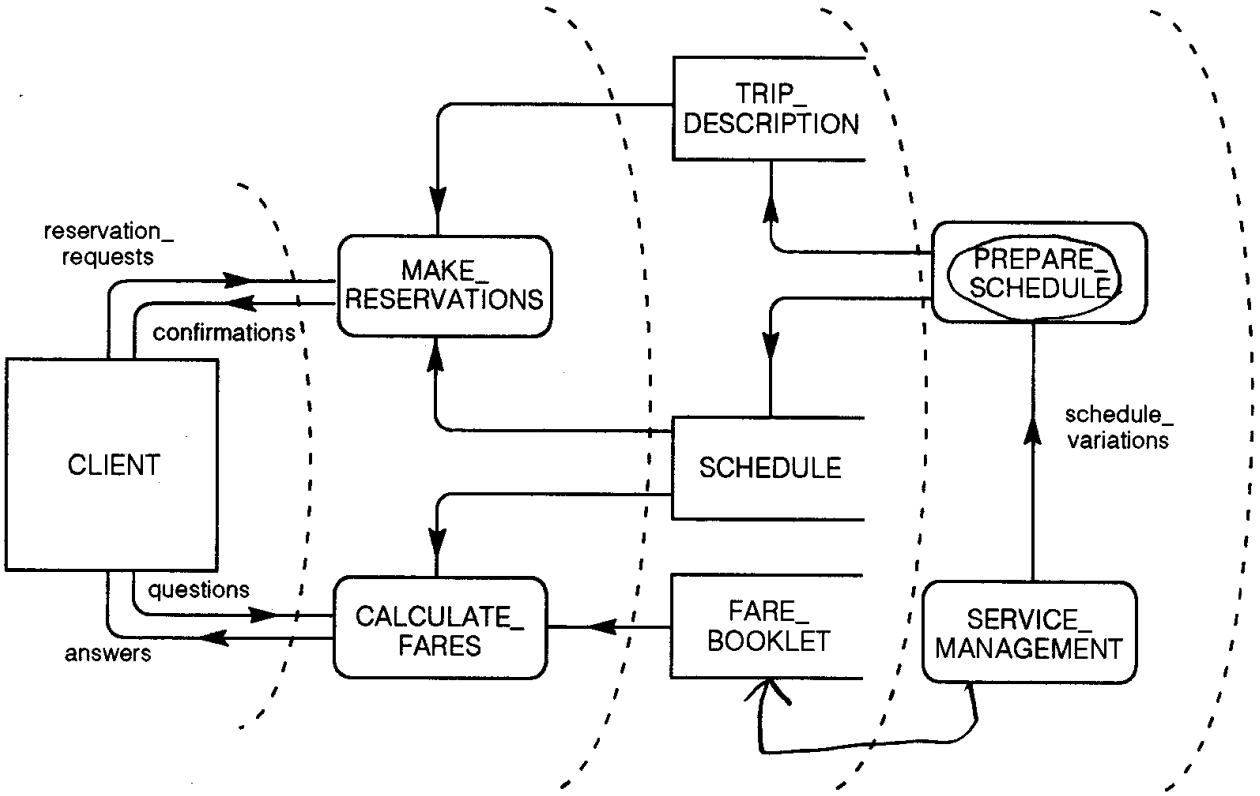
整合 external schema



D-Schema skeleton



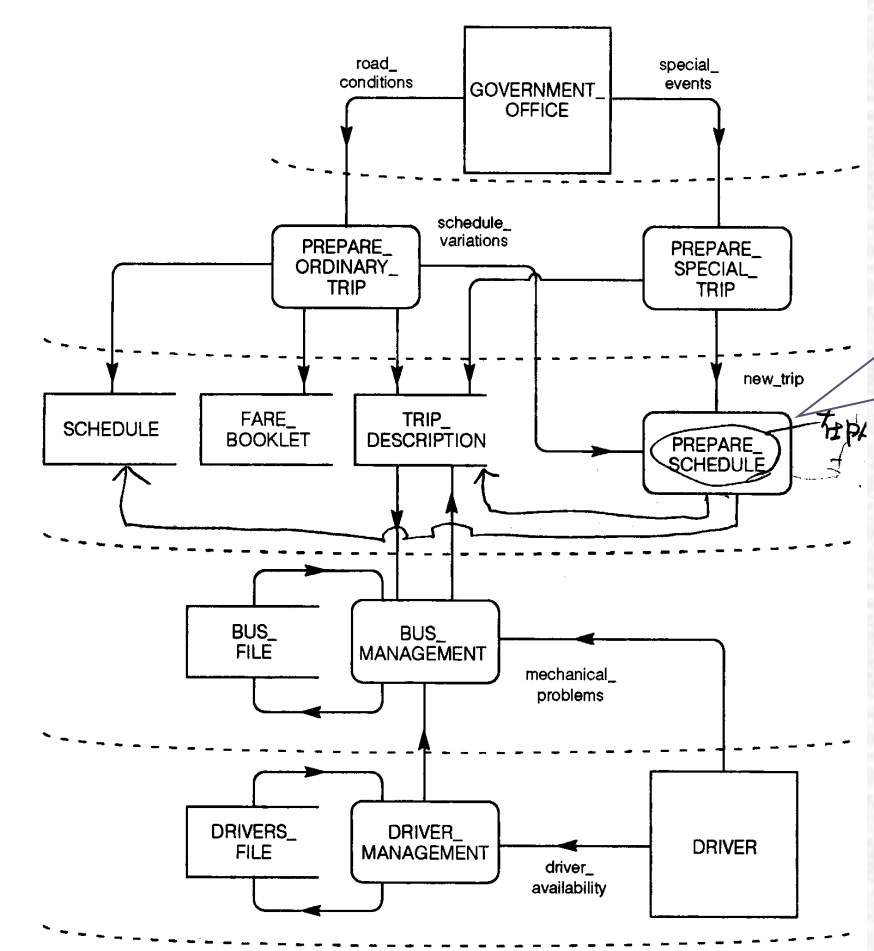
PASSENGER_MANAGEMENT



(a) F-schema for PASSENGER_MANAGEMENT

Figure 10.4 F-schema refinements: First refinement

SERVICE_MANAGEMENT

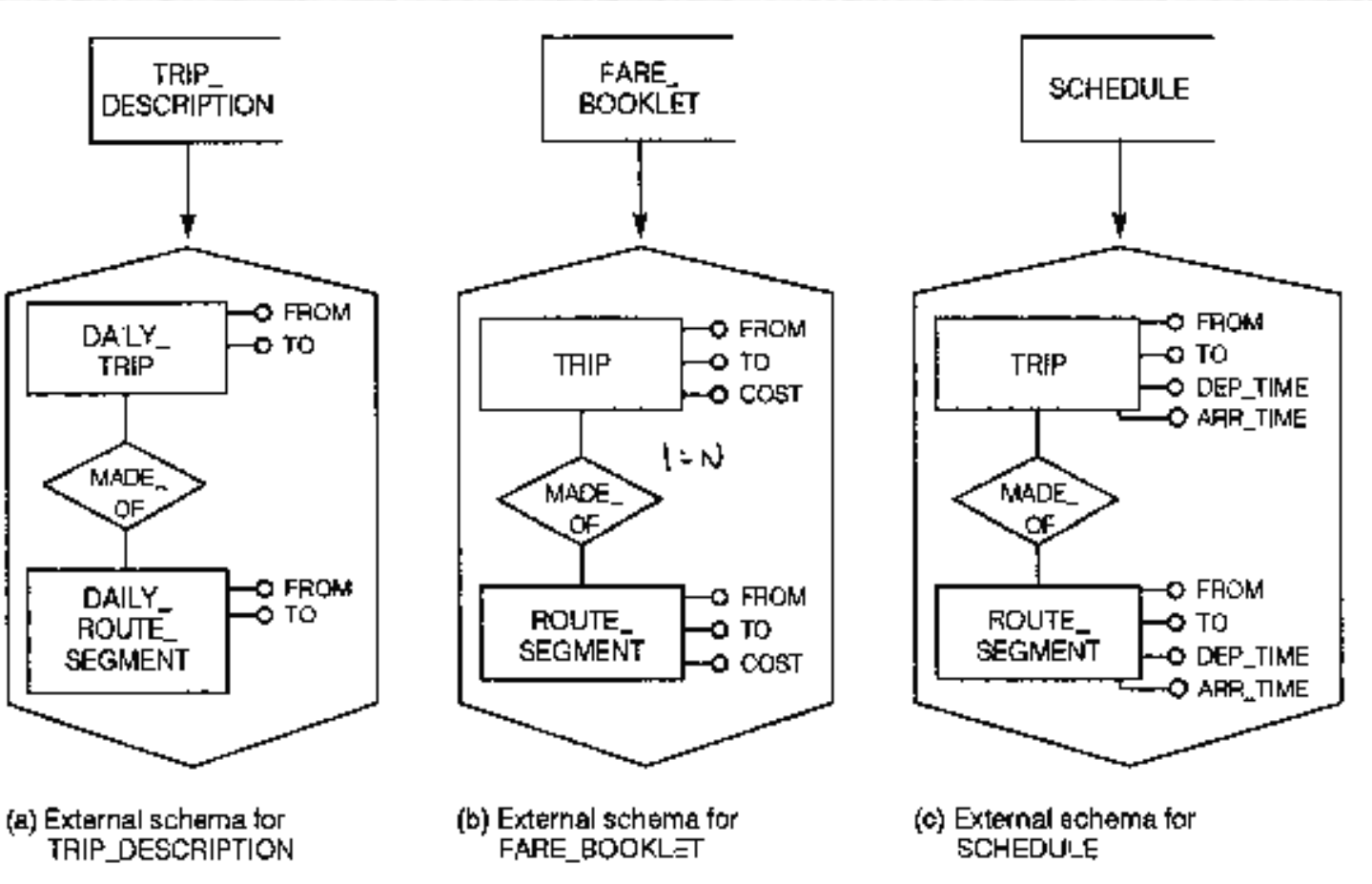


(b) F-schema for SERVICE_MANAGEMENT

Figure 10.4 (cont'd) F-schema refinements: First refinement

在 PASENGER_ MANAGEMENT 中

External Schemas

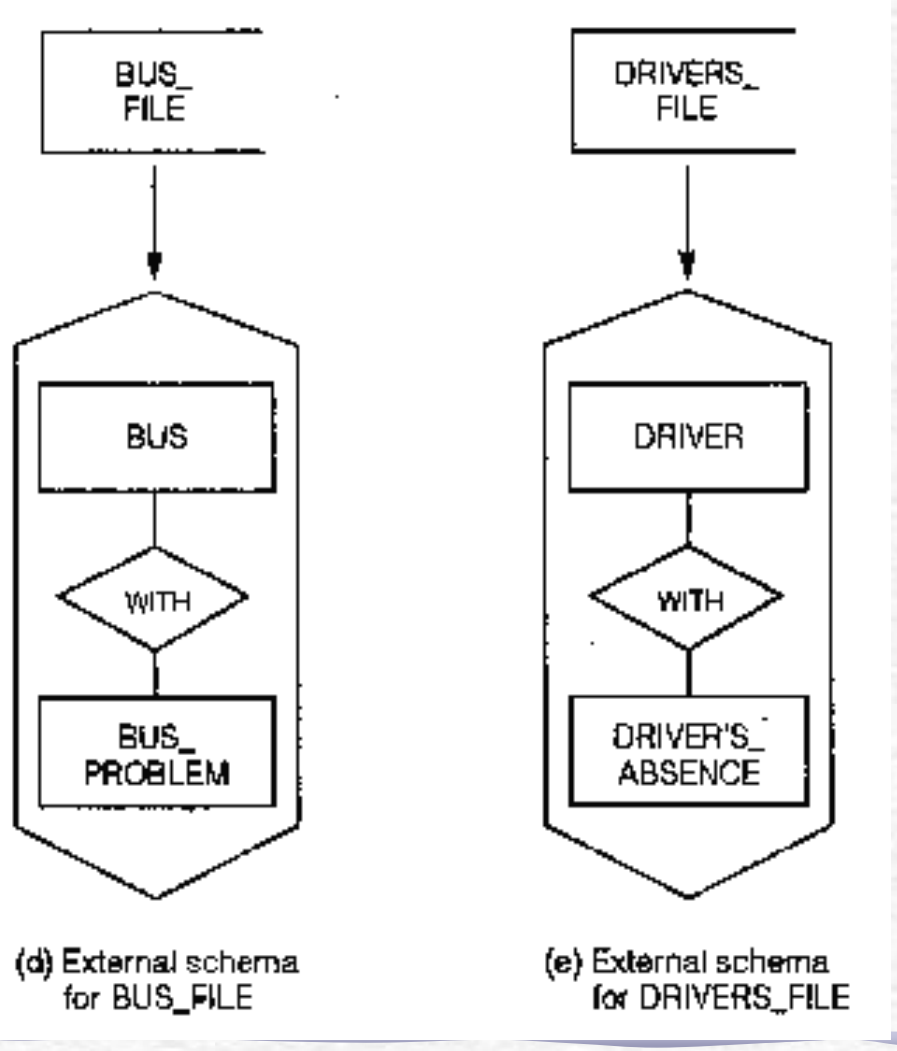


(a) External schema for TRIP_DESCRIPTION

(b) External schema for FARE_BOOKLET

(c) External schema for SCHEDULE

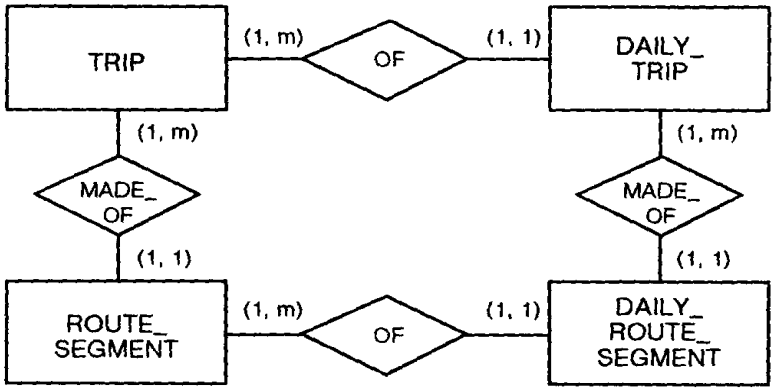
External Schemas



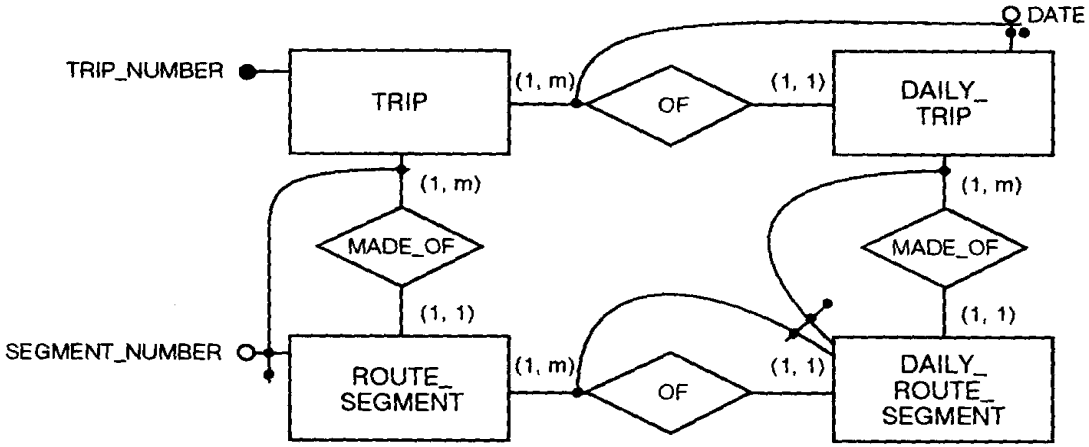
(d) External schema for BUS_FILE

(e) External schema for DRIVERS_FILE

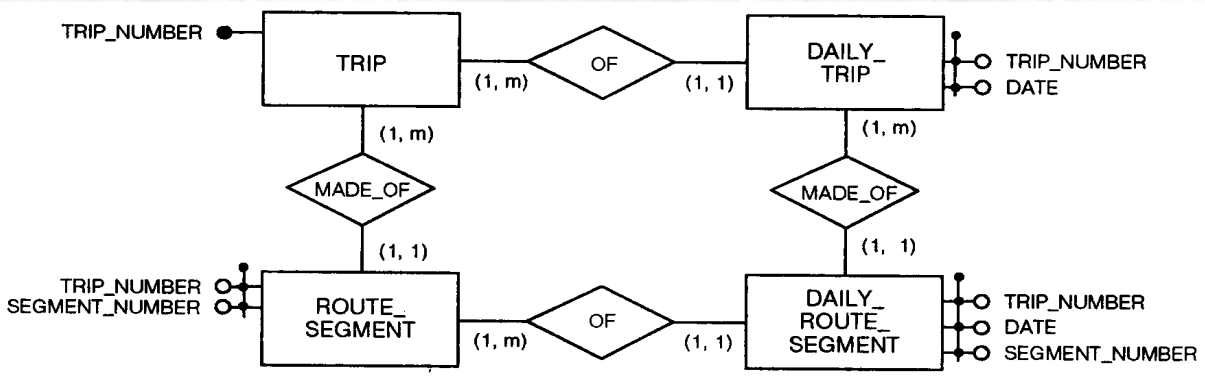
整合 external schema (a) and (b)



(a) Integration of external schemas of TRIP_DESCRIPTION and FARE_BOOKLET

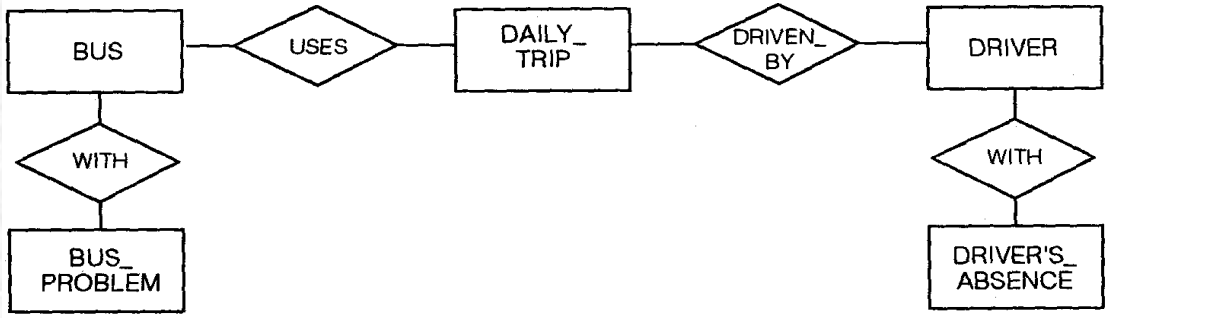


(b) Subschema describing trips and route segments with external identifiers

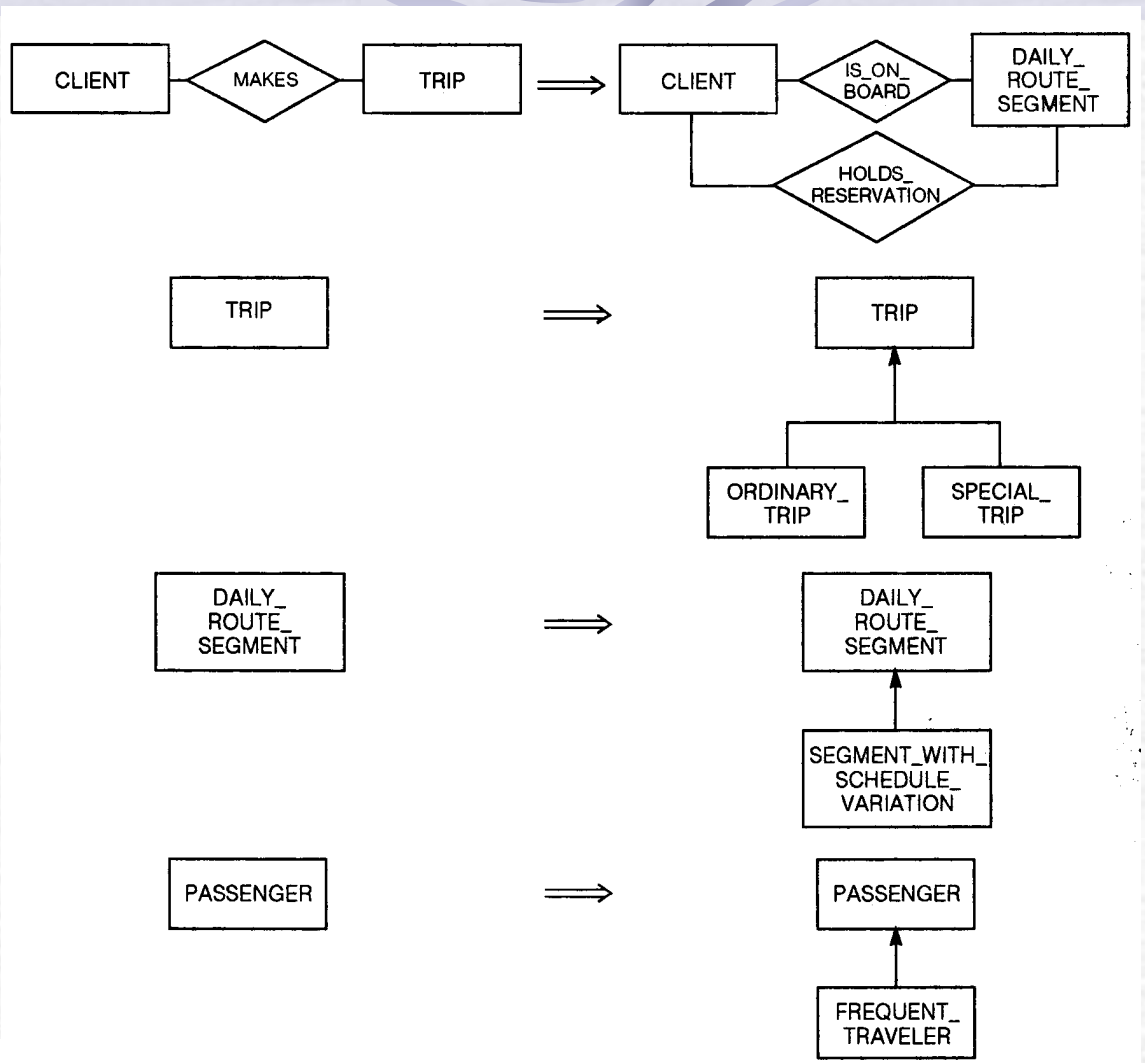


(c) Subschema describing trips and route segments with internal identifiers

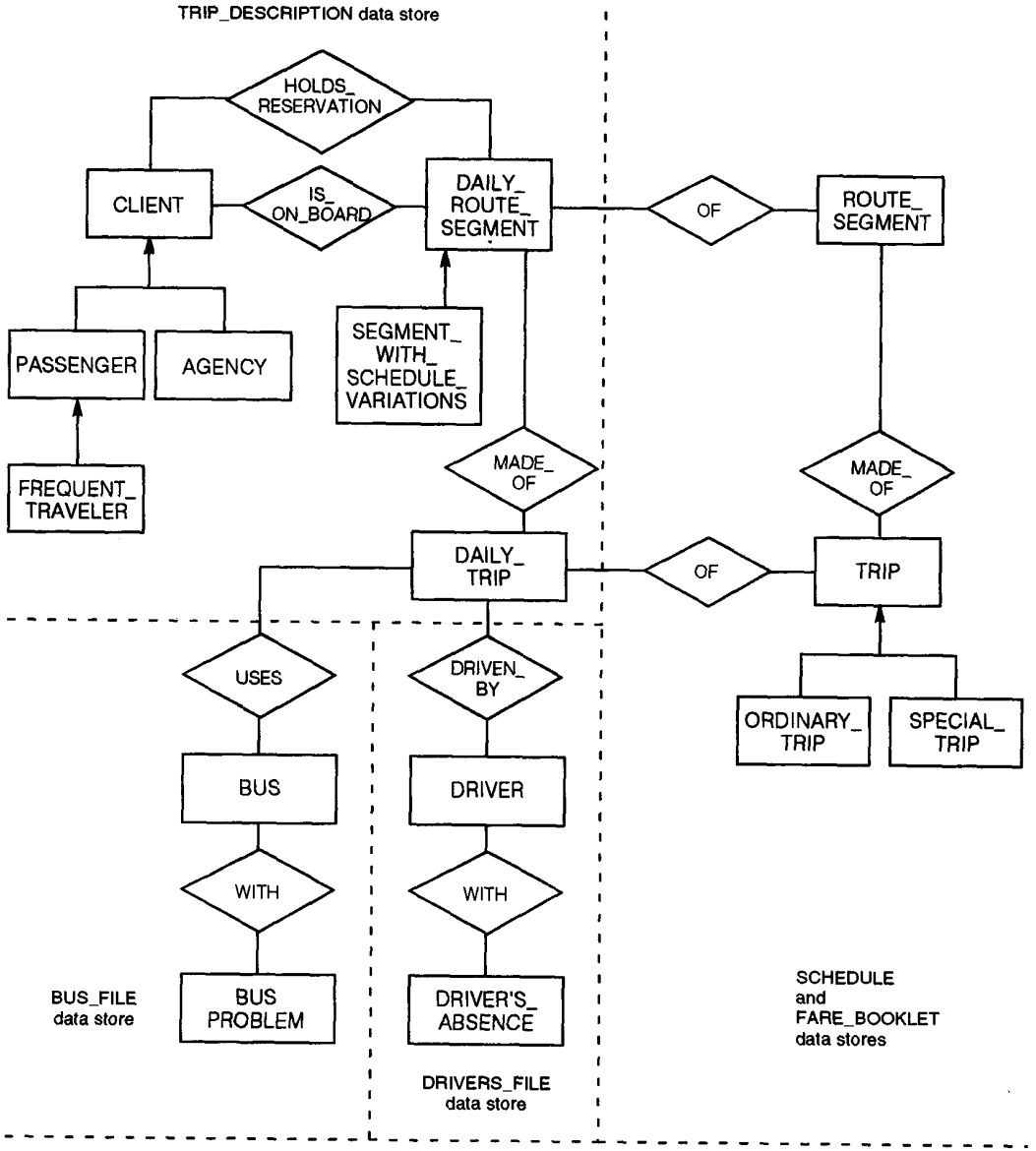
整合 external schema (d) and (e)



(d) Integration of the external schemas for BUS_FILE and DRIVER_FILE with the DAILY_TRIP entity

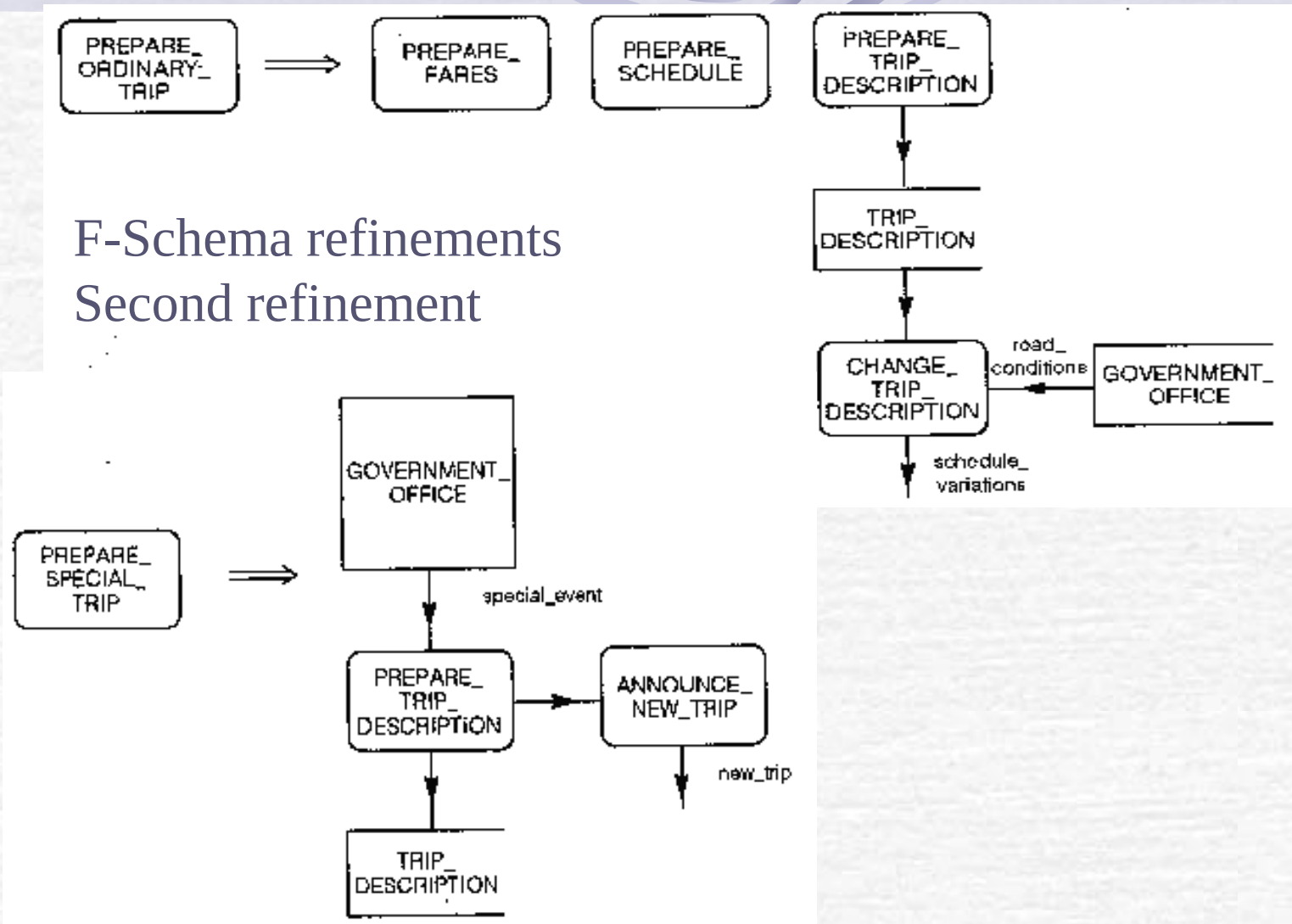


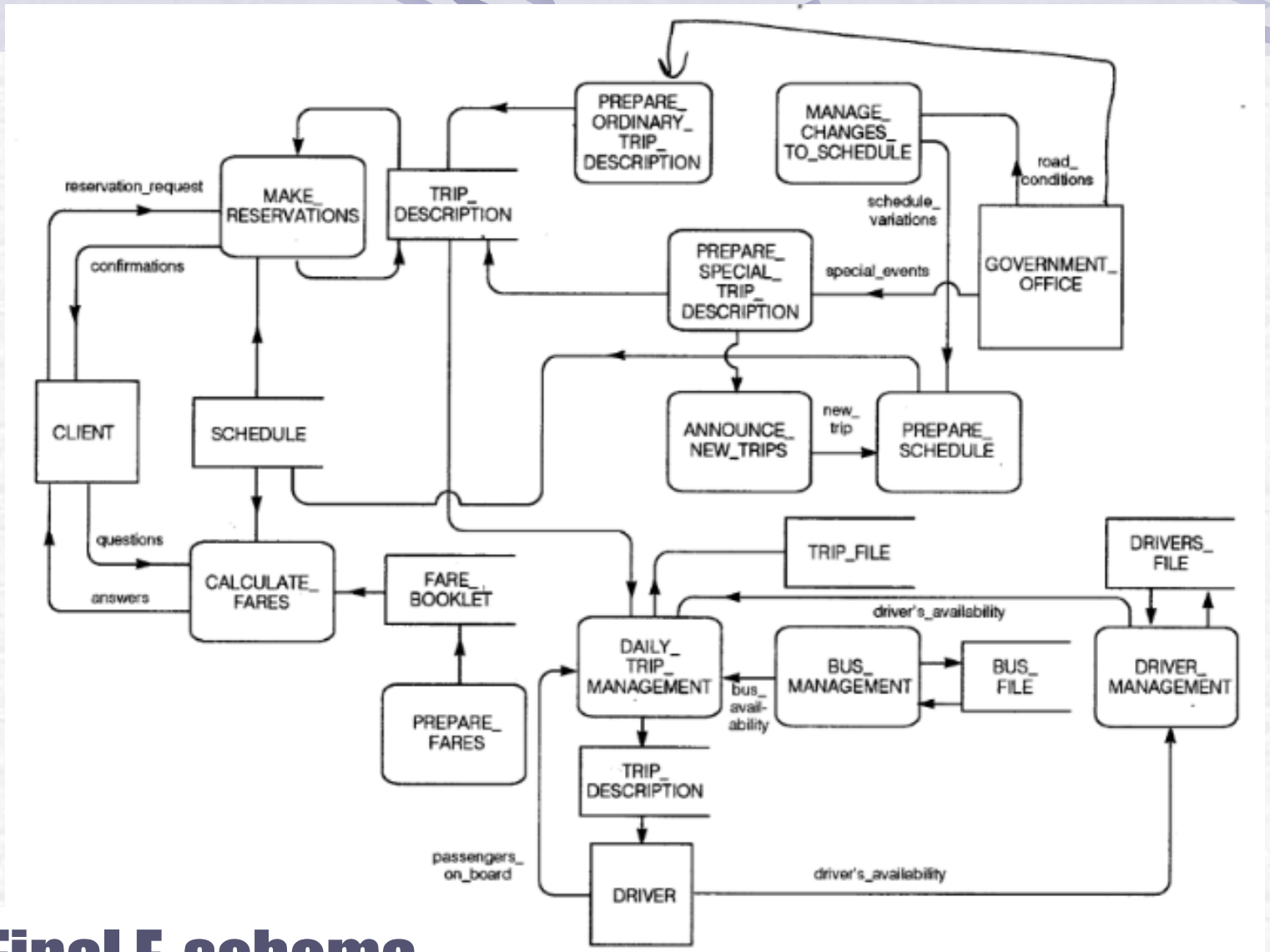
Top-down refinements



Integrated D-schema (First D-schema refinement)

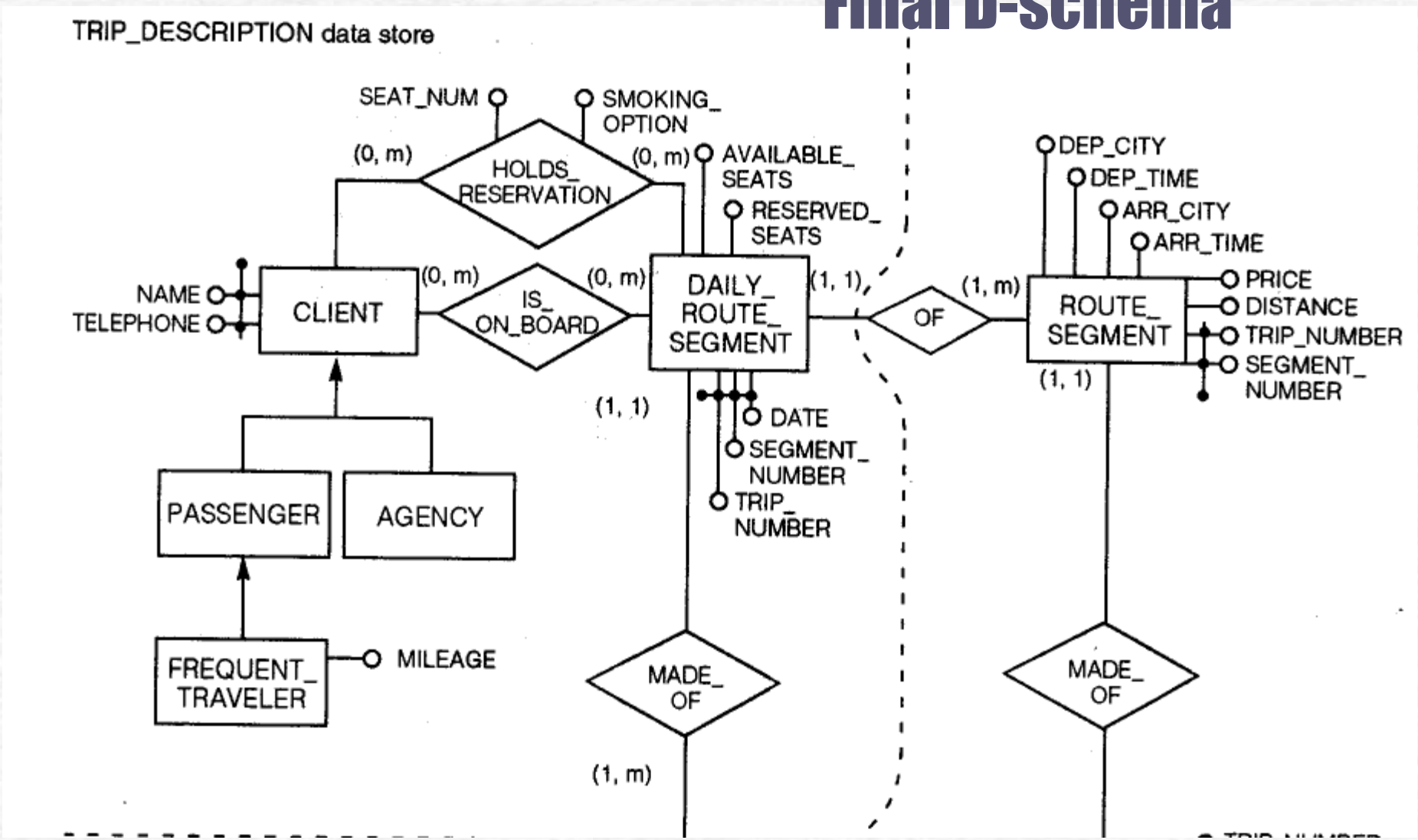
F-Schema refinements
Second refinement

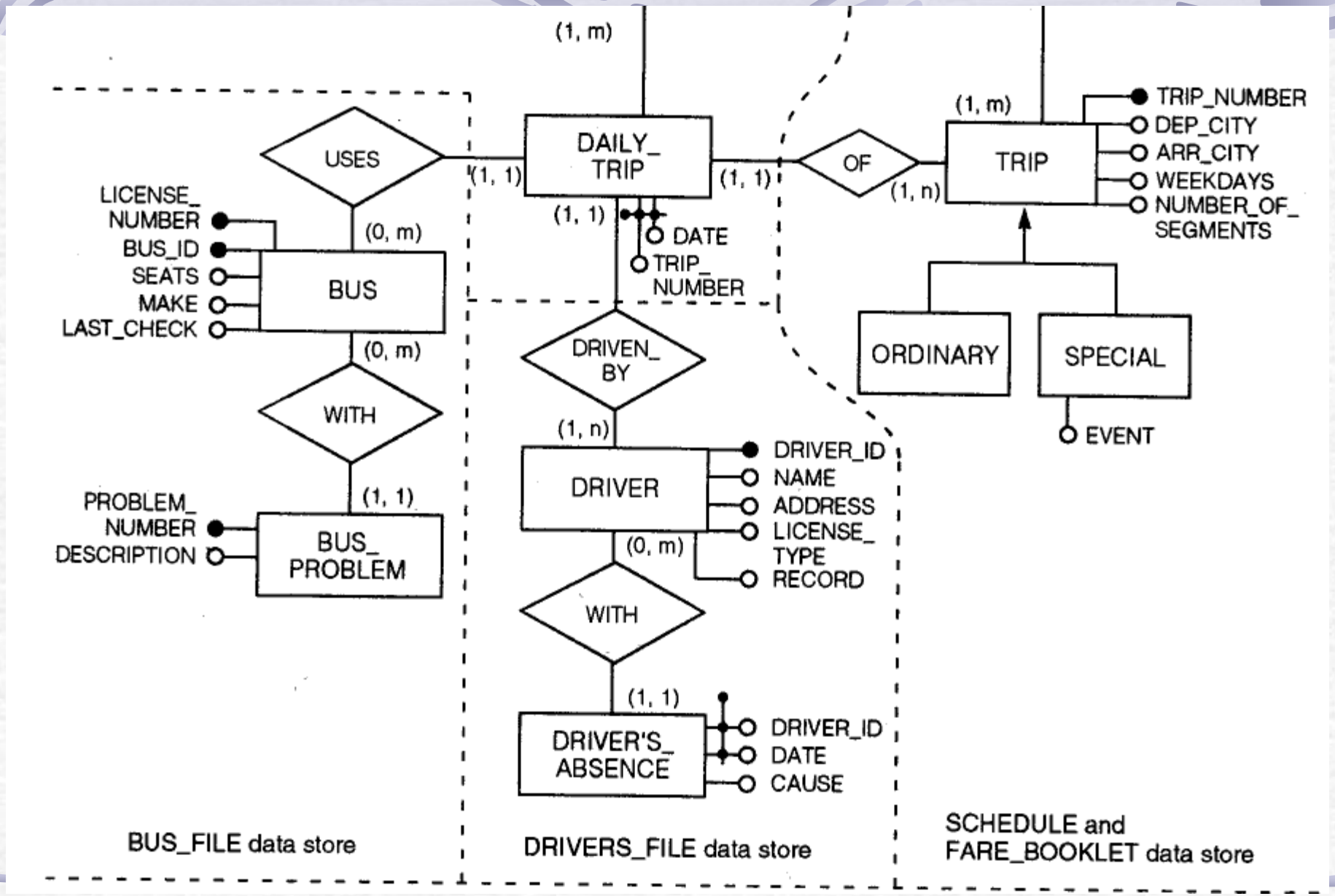




Final F-schema

Final D-schema




(1, m)
USES
(1, 1)
DAILY_ TRIP
(1, 1)
OF
(1, n)
(1, m)
TRIP
TRIP_NUMBER
DEP_CITY
ARR_CITY
WEEKDAYS
NUMBER_OF_ SEGMENTS
LICENSE_ NUMBER
BUS_ID
SEATS
MAKE
LAST_CHECK
(0, m)
BUS
(1, 1)
DATE
TRIP_ NUMBER
DRIVEN_ BY
ORDINARY
SPECIAL
EVENT
(0, m)
WITH
(1, n)
DRIVER
DRIVER_ID
NAME
ADDRESS
LICENSE_ TYPE
RECORD
PROBLEM_ NUMBER
DESCRIPTION
(1, 1)
BUS_ PROBLEM
(0, m)
WITH
(1, 1)
DRIVER'S_ ABSENCE
DRIVER_ID
DATE
CAUSE
BUS_FILE data store
DRIVERS_FILE data store
SCHEDULE and FARE_BOOKLET data store

Completeness Check of the D-Schema

☞ 檢查 F-schema 中每個資料流和 data_store 在 D-schema 中都用資料概念表達出

1. The TRIP_DESCRIPTION is represented by the following cluster of entities and relationships: CLIENT,HOLDS_RESERVATION, IS_ON_BOARD, DAILY_ROUTE_SEGMENT, MADE_OF, DAILY_TRIP, USES, BUS, DRIVEN_BY, and DRIVER.
2. The FARE_BOOKLET and SCHEDULE are represented by the following cluster of entities and relationships: ROUTE_SEGMENT, MADE_OF, and TRIP. In particular, attribute price gives the fare for each section of a trip, and the attributes DEP_TIME and ARR_TIME give the schedule information.
3. The BUS_FILE is represented by the following cluster of entities and relationships: BUS, WITH, **and** BUS_PROBLEM.
4. The DRIVER_FILE is represented by the following cluster of entities and relation-ships: DRIVER, WITH, DRIVER'S_ABSENCE.

Completeness Check of the F-Schema

- ☞ 在 D-Schema 中的每個 entity & relationship, 必須在 F-Schema 存在並使用.

1. The entities CLIENT and DAILY_ROUT_SEGMENT and relationships HOLDS_RESERVATION and IS_ON_BOARD are created, retrieved, and modified by the process MAKE_REVERVATIONS.
2. The entities DAILY_TRIP and DAILY_ROUT_SEGMENT and both relationships OF and MADE_OF are created by the two process PREPARE_ORDINARY_TRIP_DESCRIPTION and PREPARE_SPECIAL_TRIP_DESCRIPTION, and used by the processes MAKE_RESERVATIONS and DAILY_TRIP_MANAGEMENT.
3. The entities TRIP and DAILY_TRIP with the relationship OF between them are created by the processes PREPARE_FARES and PREPARE_SCHEDULE and used by the processes MAKE_RESERVATIONS and PREPARE_FARES.
4. The entities BUS and BUS_PROBLEM and the relationships USES and WITH are created and used by the BUS_MANAGEMENT process.
5. The entities DRIVER and DRIVER'S_ABSENCE and the relationship WITH are created and used by the DRIVER_MANAGEMENT process; the relationship DRIVEN_BY is created and used by the BUS_MANAGEMENT process.

Step2:
Analysis important operations

1. Describing the operations
2. Drawing its navigation schema

Database operations executed by MAKE_RESERVATIONS

1. SEARCH ROUTE_SEGMENT BY DEP_CITY.
2. SEARCH ROUTE_SEGMENT BY BOTH DEP_CITY AND ARR_CITY.
3. SEARCH ALL TRIPS HAVING AN INTERMEDIATE STOP AT A GIVEN CITY.
4. CREATE A NEW CLIENT RECORD.
5. MAKE A RESERVATION (for an existing client).
6. DELETE RESERVATIONS OF A PAST TRIP.
7. DELETE A CLIENT RECORD (provided that he/she is not a frequent traveler or has no reservations).
8. QUALIFY A CLIENT AS FREQUENT TRAVELER.
9. RETRIEVE THE AMOUNT OF MILES EARNED BY FREQUENT TRAVELERS.
10. UPDATE THE MILEAGE OF A FREQUENT TRAVELER.
11. RETRIEVE ALL CURRENT RESERVATIONS FOR A GIVEN TRIP ON A GIVEN DATE.
12. RETRIEVE ALL CURRENT RESERVATIONS OF A GIVEN AGENCY.

Navigation schemas

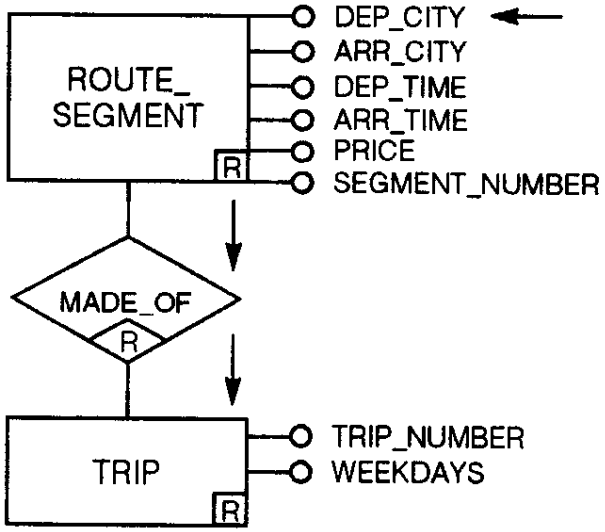


Figure 10.10 Navigation schema for O1

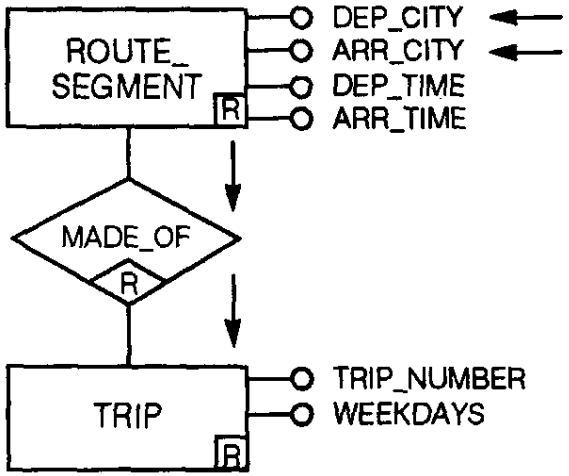


Figure 10.11 Navigation schema for O2 and O3

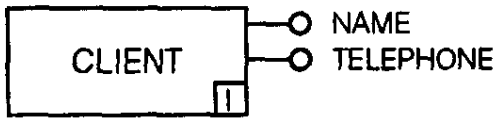
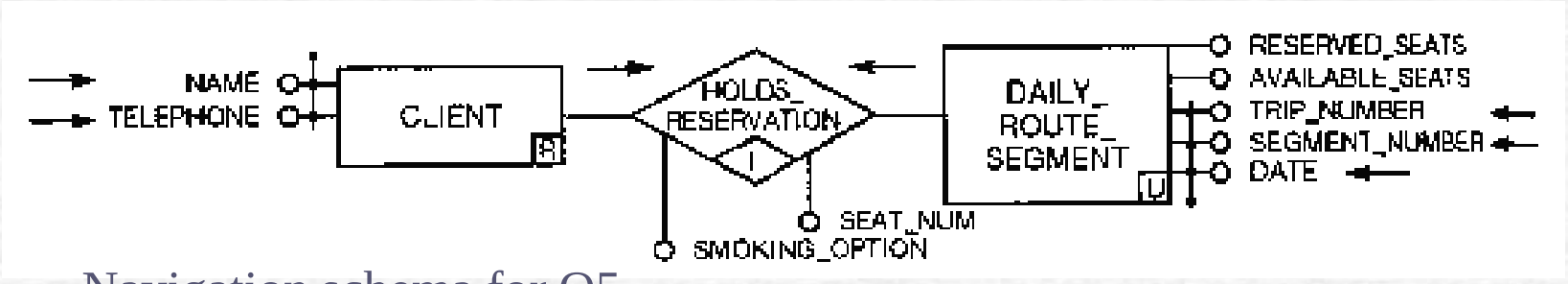
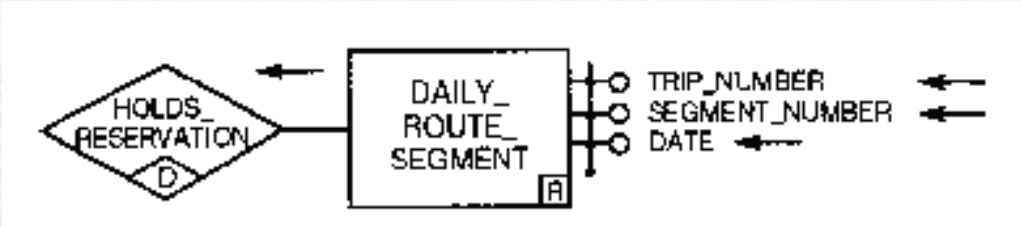


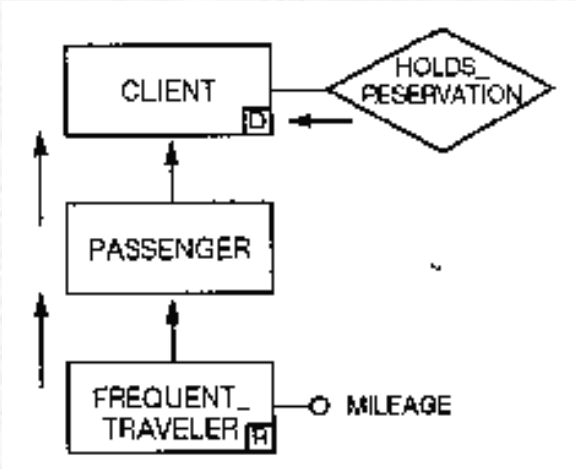
Figure 10.12 Navigation schema for O4



Navigation schema for O5



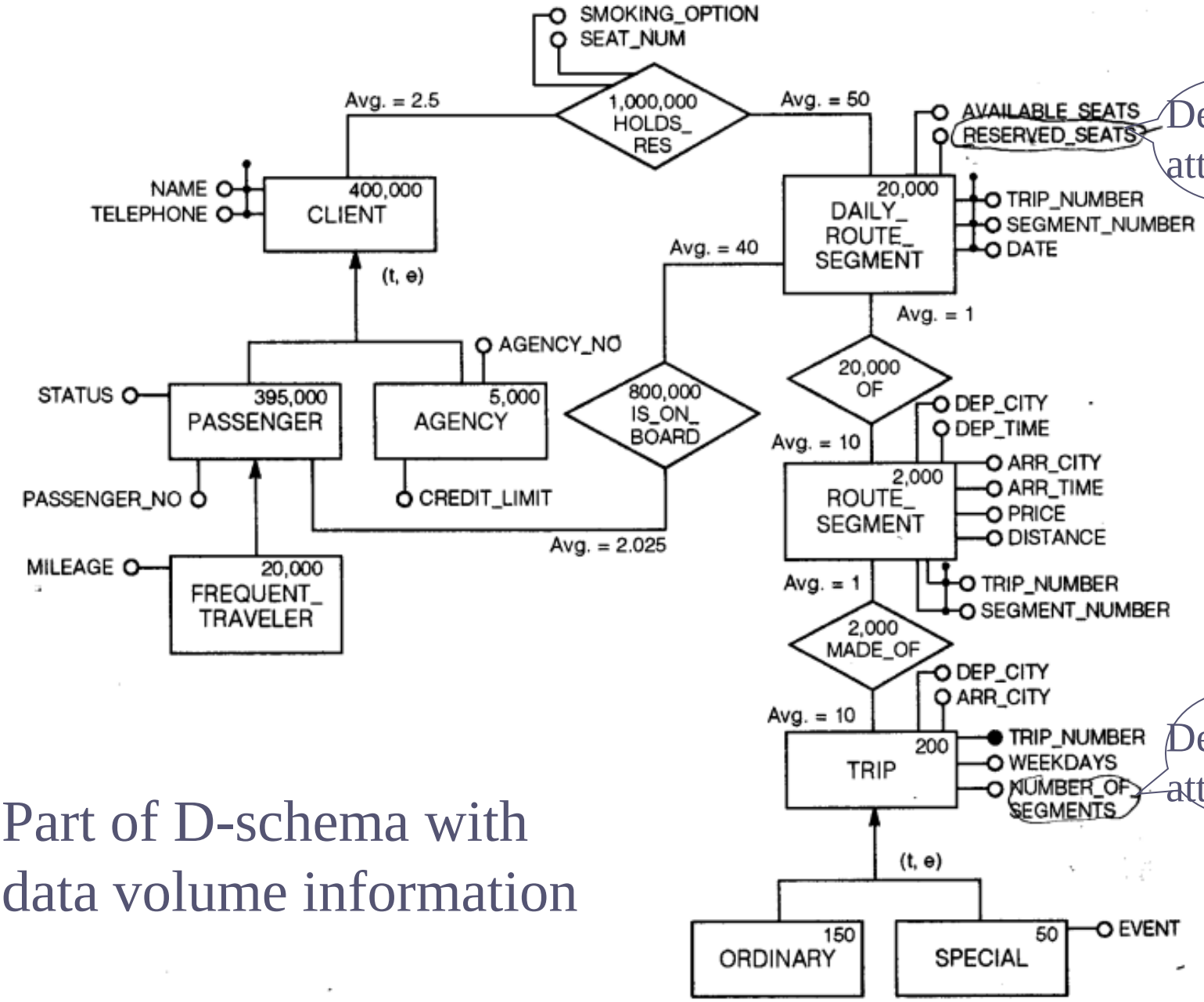
Navigation schema for O6



Navigation schema for O7

Step3: High-level logical design

1. Analysis the database load
2. Decision about derived data
3. Removing generalization hierachies
4. Partitioning of entities
5. Merging entities and relationship
6. Primary key selection



Part of D-schema with data volume information

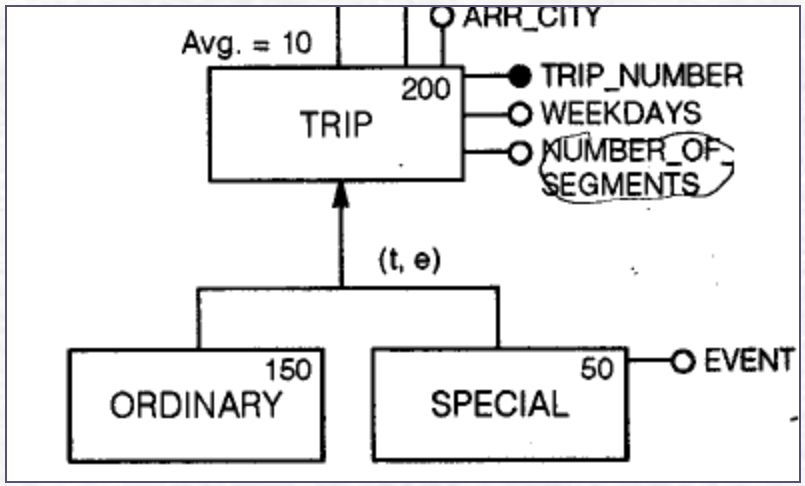
Operation Frequency Table

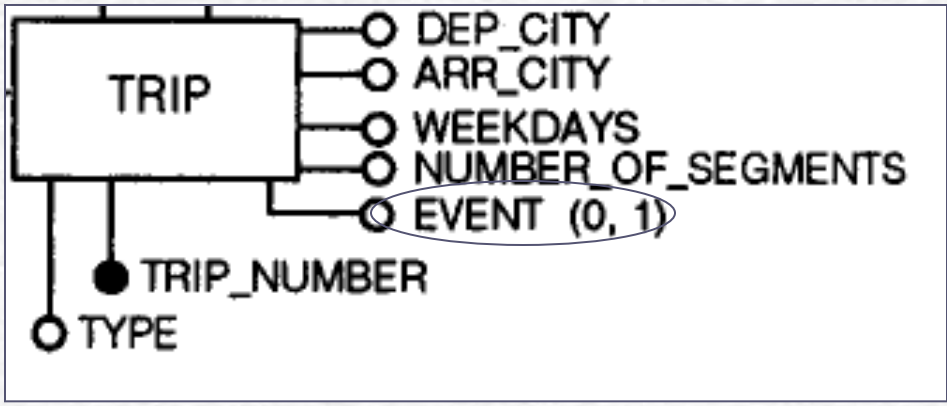
Operation Name/Description	Frequency	Type (On-line/Batch)
01 SEARCH ROUTE_SEGMENT BY DEP_CITY	50 times a day	OL
02 RETRIEVE ROUTE_SEGMENT BY DEP_CITY AND ARR_CITY	200 times a day	OL
03 RETRIEVE TRIPs WITH INTERMEDIATE STOP AT A GIVEN CITY	5 times a day	OL
04 CREATE A NEW CLIENT RECORD	500 times a day	OL
05 MAKE A RESERVATION (EXISTING CLIENT)	20,000 times a day	OL
06 DELETE RESERVATION OF A PAST TRIP	70 times a day	OL
07 DELETE A CLIENT RECORD	1 time a day	OL
08 QUALIFY A CLIENT AS A FREQUENT TRAVELER	10 times a day	OL
09 RETRIEVE THE AMOUNT OF MILES EARNED BY FREQUENT TRAVELERS	Once in a month	B
010 UPDATE THE MILEAGE OF A FREQUENT TRAVELER	Once in a day	B
011 RETRIEVE ALL CURRENT RESERVATIONS OF A GIVEN TRIP ON A GIVEN DATE	100 times a day	OL
012 RETRIEVE ALL CURRENT RESERVATIONS OF A GIVEN AGENCY	10 times a day	OL

Decisions about derived data

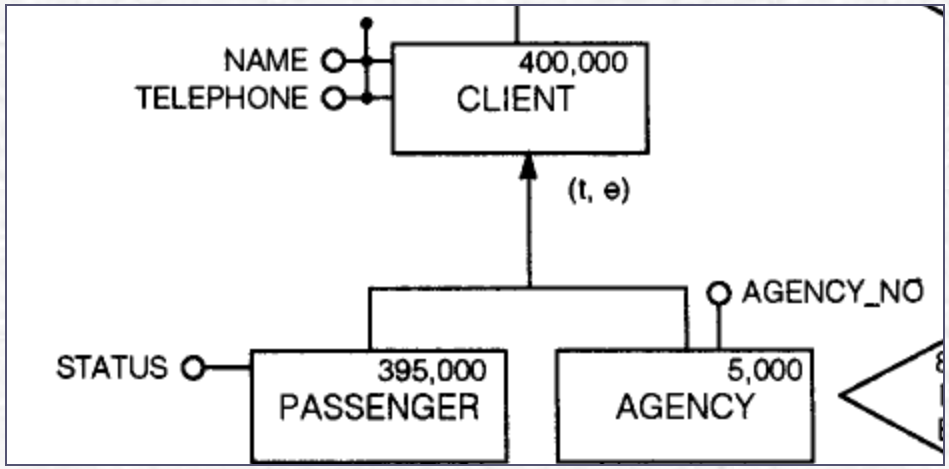
Derived data Item	O1	O2	O3	O4	O5	O6	O7	O8	O9	O10	O11	O12
RESERVED_SEATS	N	N	N	N	**Y**	**Y**	N	N	N	N	N	N
AVAILABLE_SEATS	N	N	N	N	**Y**	**Y**	N	N	N	N	N	N
NUMBER_OF_SEGMENT	N	N	N	N	N	N	N	N	N	N	N	N
DEP_CITY	**Y**	**Y**	**Y**	N	N	N	N	N	N	N	N	N
ARR_CITY	N	**Y**	**Y**	N	N	N	N	N	N	N	N	N

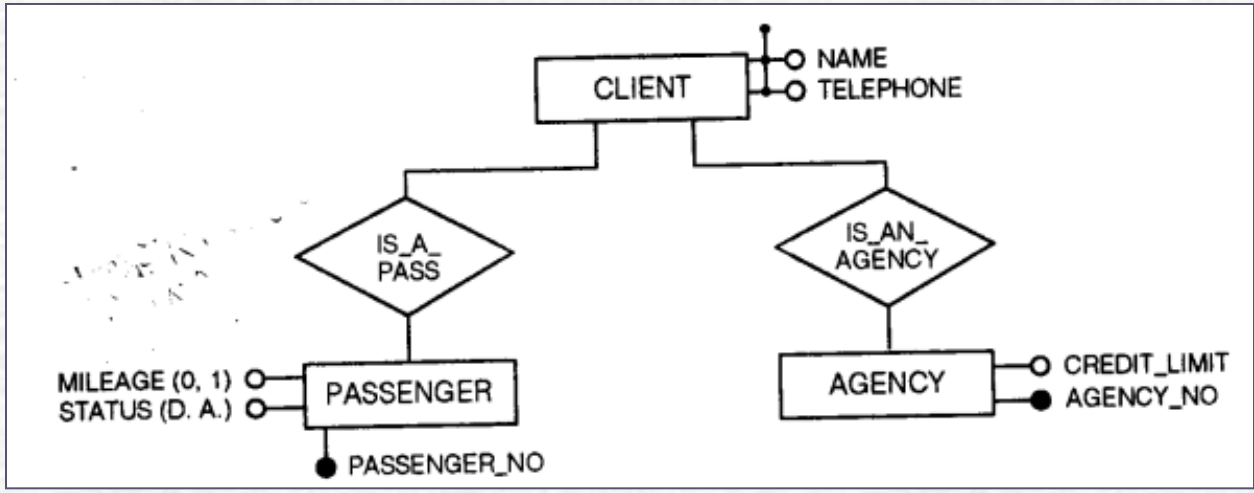
Collapsing the generalization hierarchy



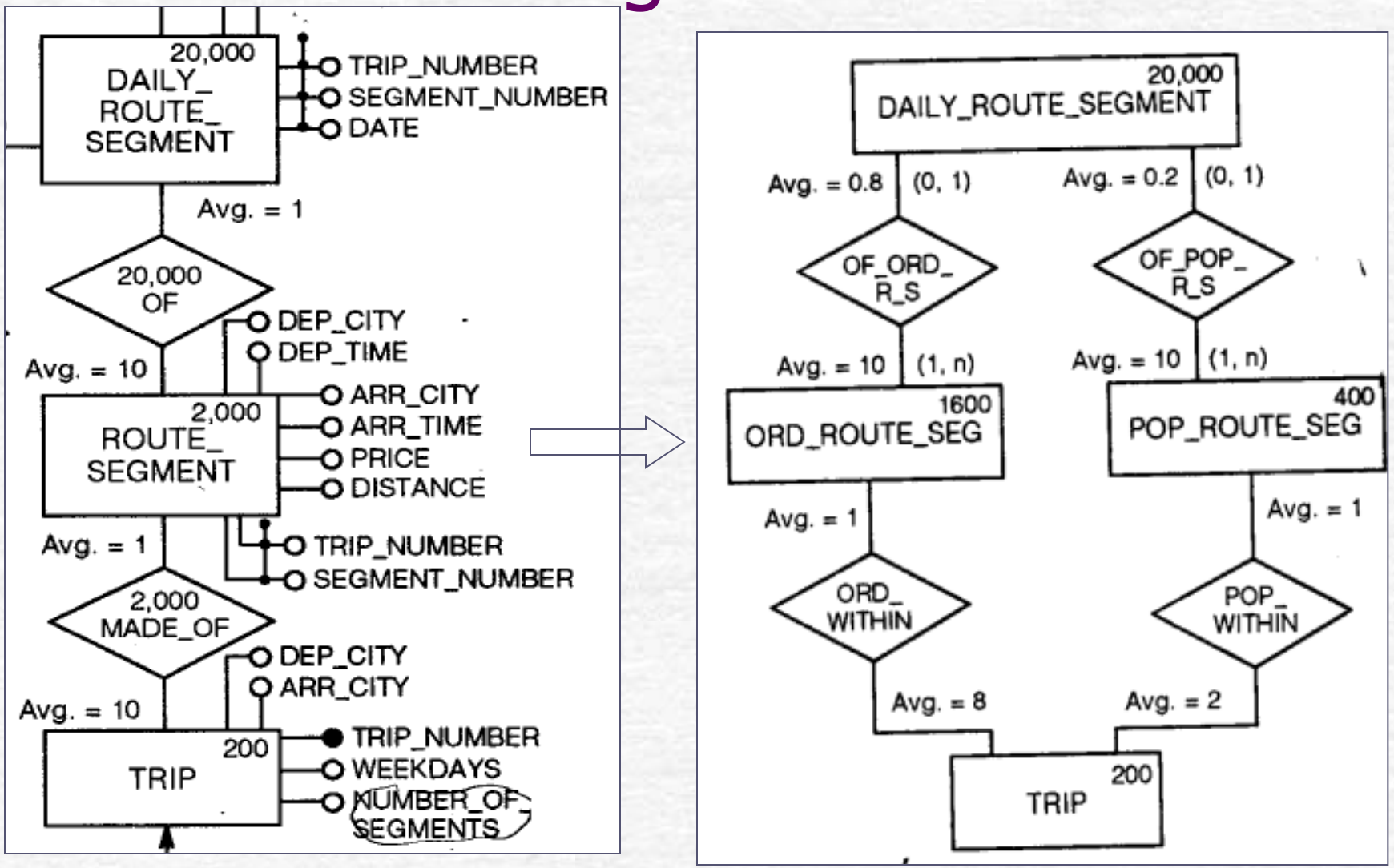


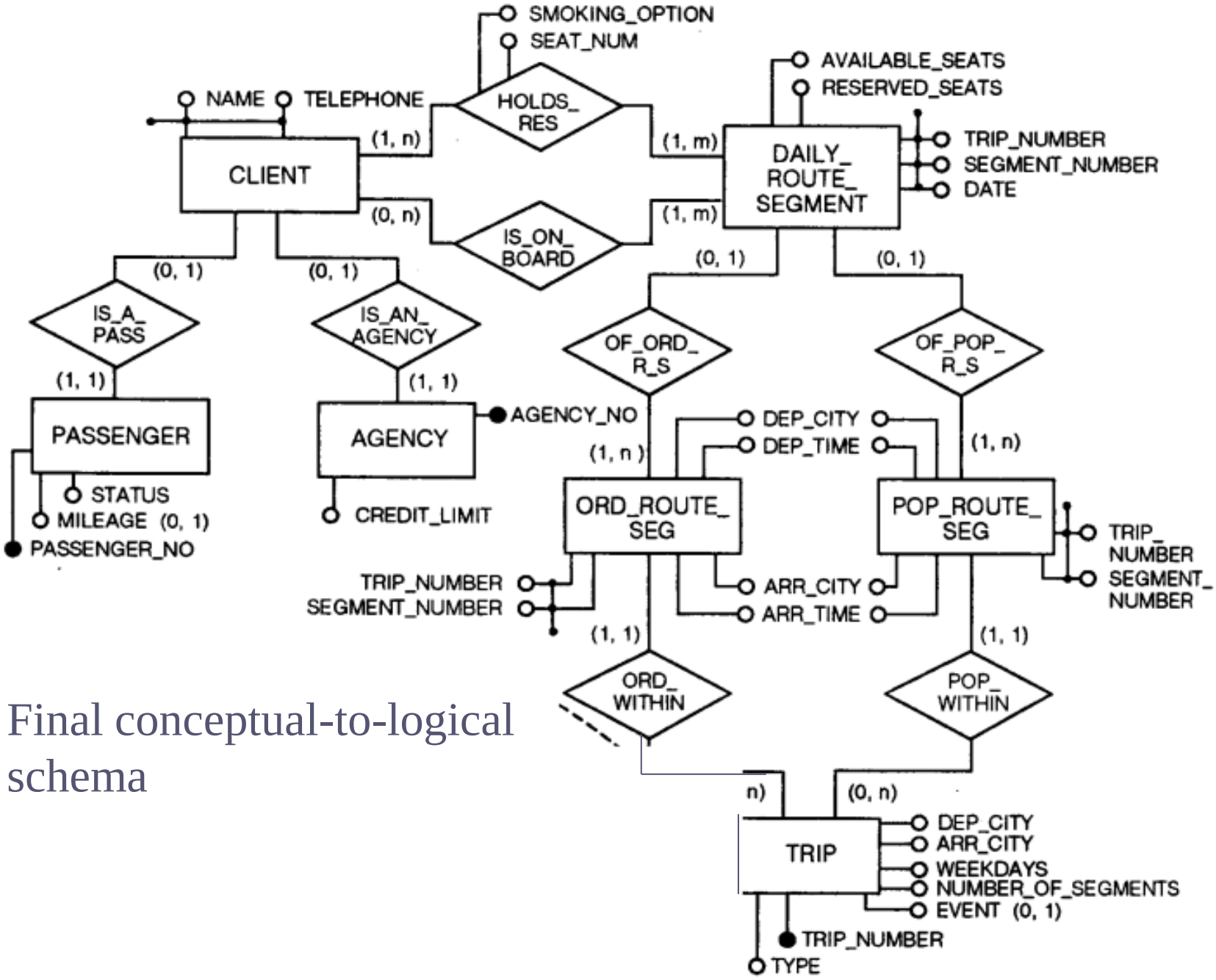
Removing the generalization hierarchy





Partitioning of entities

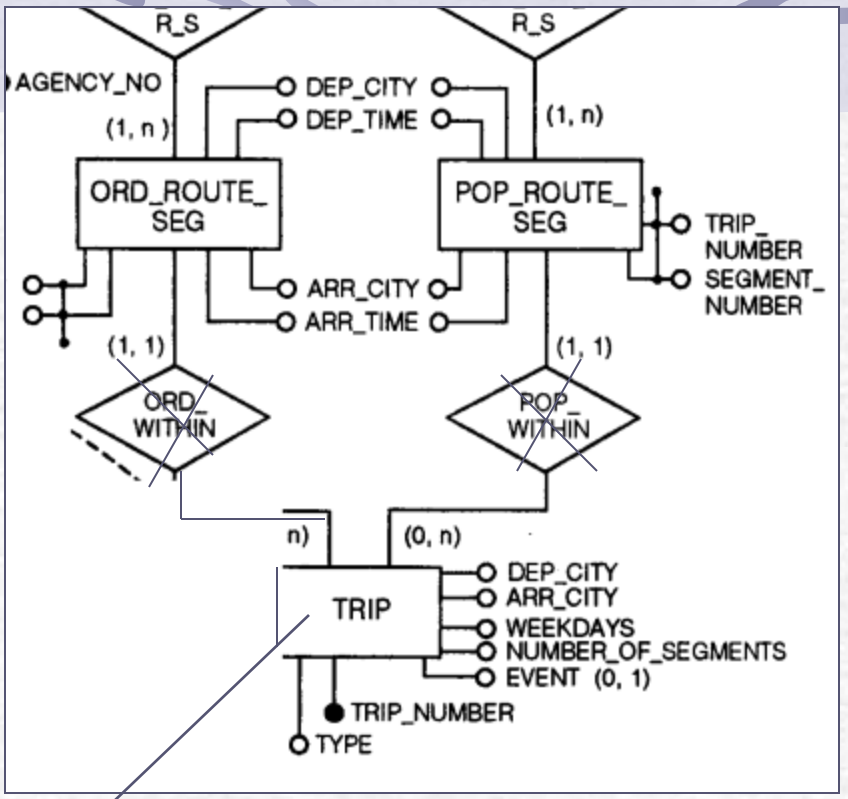




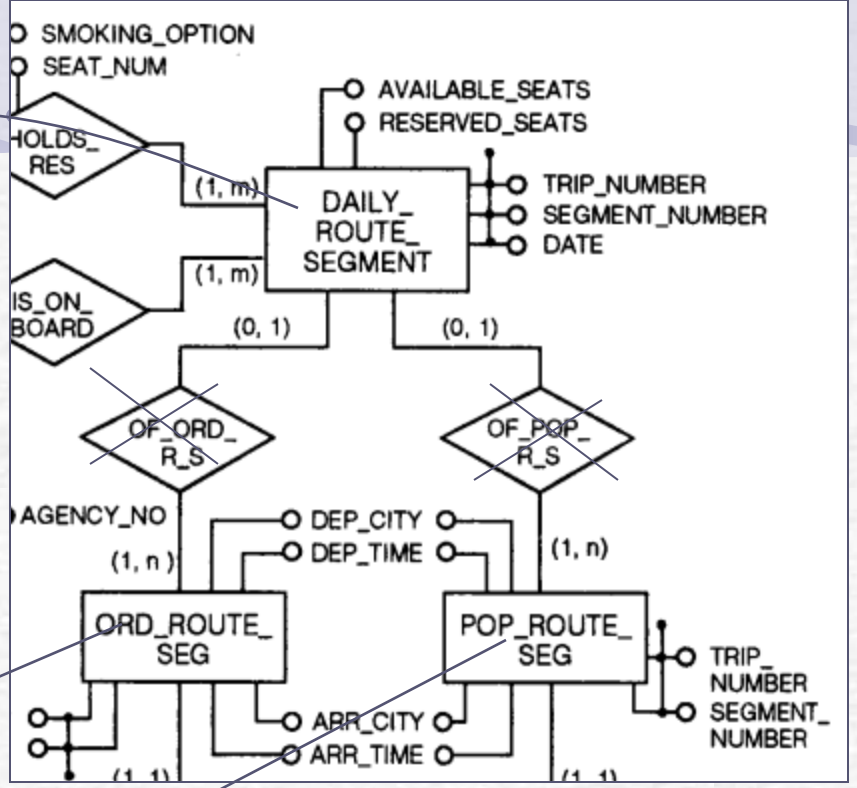
Final conceptual-to-logical schema

Step4: Logical design for relational model

1. Elimination of external identifier
2. Elimination of composite and multi-value attributes
3. Translation of entities
4. Translation of relationships
5. Mapping the operations to SQL



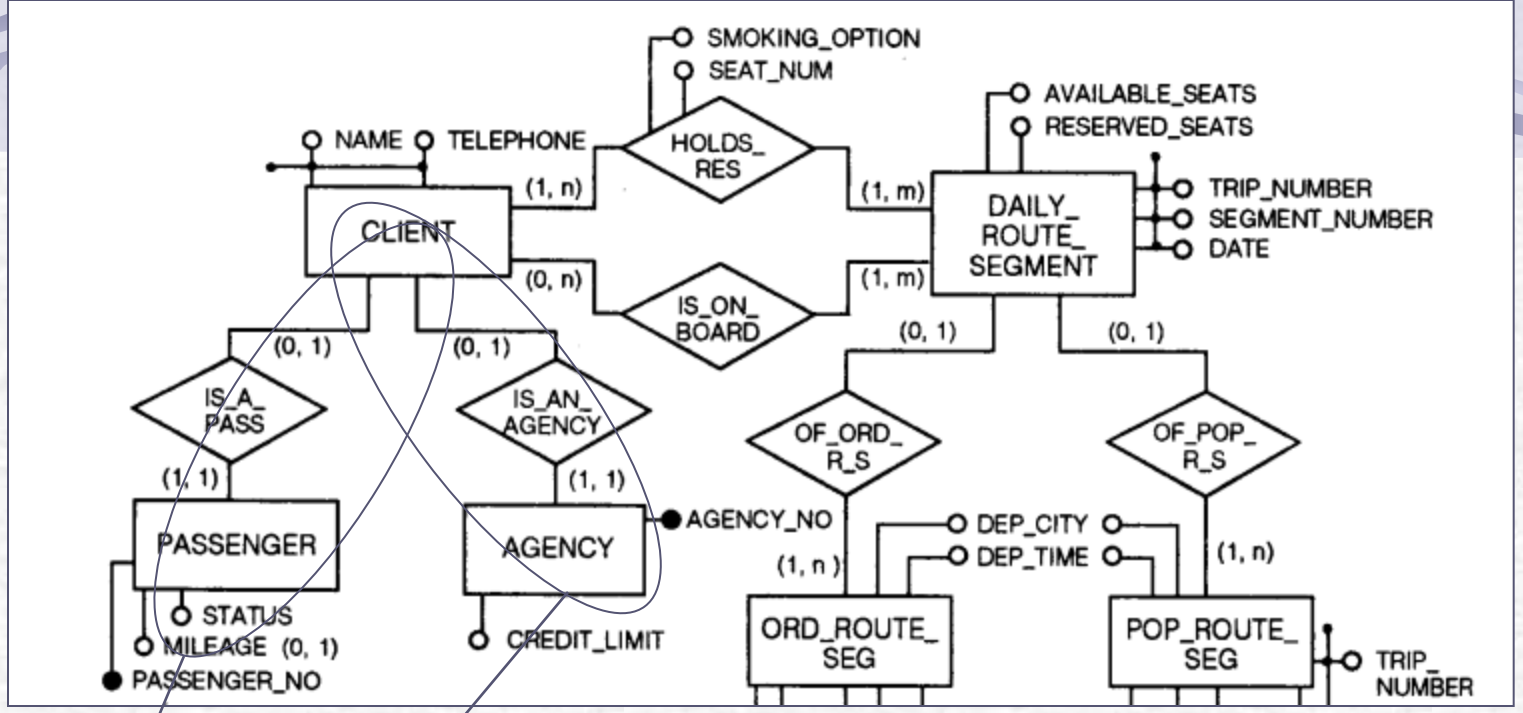
TRIP(TRIP_NUMBER, DEP_CITY, ARR_CITY, WEEKDAYS, TYPE, EVENT)



DAILY_ROUTE_SEGMENT
(TRIP_NUMBER,
SEGMENT_NUMBER,
DATE,
AVAILABLE_SEATS,
RESERVED_SEATS)

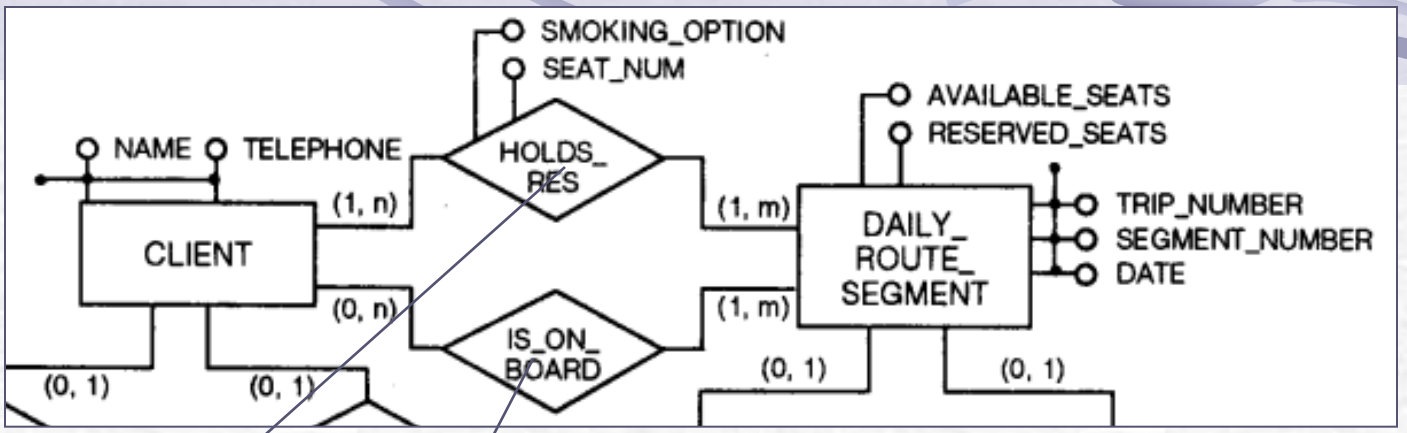
ORDINARY_ROUTE_SEGMENT(TRIP_NUMBER, SEGMENT_NUMBER,
DEP_CITY, DEP_TIME, ARR_CITY, ARR_TIME, PRICE, DISTANCE)

POPULAR_ROUTE_SEGMENT(TRIP_NUMBER, SEGMENT_NUMBER,
DEP_CITY, DEP_TIME, ARR_CITY, ARR_TIME, PRICE, DISTANCE)



PASSENGER(CLIENT_NO, NAME, PHONE, MILEAGE, STATUS)

AGENCY(CLIENT_NO, NAME, PHONE, CREDIT_LIMIT)



HOLDS_RES(CLIENT_NO, TRIP_NUMBER, DATE, SEGMENT_NUMBER, SEAT_NUM, SMOKING_OPTION)

PASSENGER_IS_ON_BOARD(CLIENT_NO, TRIP_NUMBER, DATE, SEGMENT_NUMBER)

Exercise

Use the techniques presented in this case study to design the information system of the library of a computer science department described here. Make further assumptions as needed and state them explicitly. Produce both the D-schema and F-schema at the end, with the list of major operations and their navigation schemas.

In the library of a computer science department, books can be purchased both by researchers and by students. Researchers must indicate the grant used to pay for the book; each student has a limited budget, which is fixed each year by the dean of the college.

Some books are chosen from among those sent periodically for review by publishing companies; each book is requested by filling out a request form. When a book for review is selected, it is kept in the library, and a purchase order is issued; otherwise, books for review are returned after a short period. When an ordered book is not for review, the correctness of the request is checked against the directory of recently published books, and then an order is issued. When the librarian receives the invoice (which comes with the book in the second case), its content is compared against the order, and then the invoice is paid if there is no error. At this point, the librarian asks the requestor for 10 key words and the classification according to ACM categories.

續下頁

Together with books, journals are also kept in the library. Every year, each professor of the department can express ten preferences among currently published journals. The twenty journals with the highest scores are selected.

Books can be loaned to students and professors of the department: professors can also borrow journals. Students cannot keep more than five books at the same time, and each book no longer than one month. When a student keeps a book longer than one month, a letter is sent to him or her. If this happens -two or more times in the current year, new books can be loaned only after the student indicates a warranting professor. Professors of the department can keep any number of books and journals, but if they want to keep a book for more than one month, they have to declare where they keep it and allow the librarian to get it back at any time.

If a student or a professor asks for a book that is on loan, then the requestor is informed about the time by which the book should be returned by the current borrower. The request is stored for statistical purposes: when a book is not available for a large number of requests, a new copy is automatically purchased.

Sometimes books disappear from the library; these are detected periodically by taking an inventory of the books physically present in the library and by merging the list with the books on loan, recorded in the master directory. Every six months, a list of the missing books is published. If they are not returned in one month, they are automatically purchased again.

When a book is 10 years old, the library director decides whether to store the book in the library store, instead of leaving it on the shelves. Journals that have not been used or loaned in a year are no longer purchased in the next year.