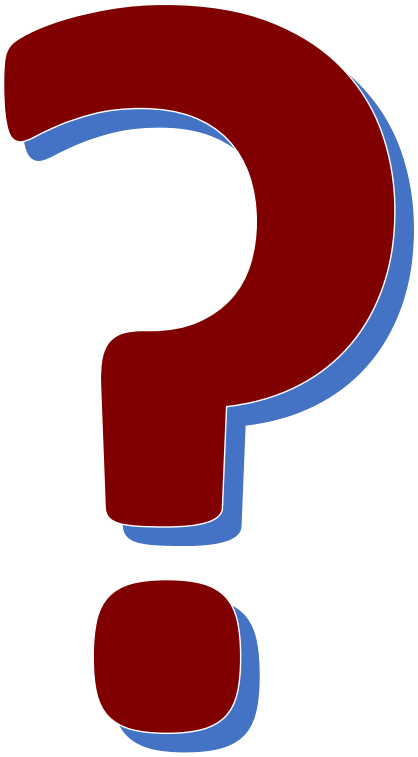
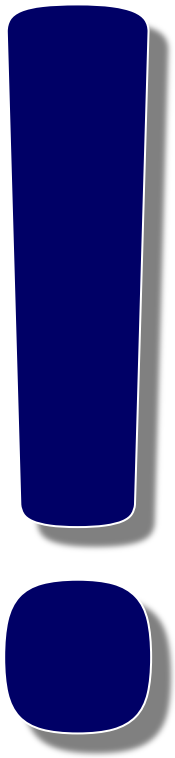
ISE-DMDD

Exercises and Answers for Data Modeling and Rel. Database Structures





General information

Title	ISE-DMDD Exercises and Answers for Data Modeling and Relational Database Structures
Academic year	2019-2020
Training Programs	Information Science, profile Software Development (SD) Information Science, profile Enterprise Software Solutions (ESS) Information Science, profile Information Management & Consultancy (IMC)
Author	Jan Pieter Zwart

The English texts were written by Jan-Pieter Zwart, who also made all the English figures
The exercises are based on original texts by Marco Engelbart.

Contents

Use of this Reader

The exercises are presented first in section 2, arranged by theme, and worked answers to many of them are given in section 3.

Students are encouraged to do the exercises without consulting section 3, and to compare their own results with the worked answers afterwards. Try to explain any differences you find yourself: sometimes two different solutions are both correct, sometimes there really is a mistake.

The teacher will be happy to answer any questions about the solutions presented here, or your own results.

1 Exercises

1.1 Theme 1 Normal forms and Normalization

Exercise 1.1 Movies

Table MOVIE below is used to store data about movies and actors who perform in these films. Only a small part of the population is shown.

MOVIE

Title ←	Year	Director	Actor →
Grease	1978	Randal Kleiser	Olivia Newton-John
Grease	1978	Randal Kleiser	John Travolta
Grease	2016	Bruce Johnson	Justin Bieber
Metro	1997	Thomas Carter	Eddie Murphy
Metro	1997	Thomas Carter	Kim Kiyori
Metro	1997	Thomas Carter	Michael Rapaport
…	…	…	…

Figure E1.1.1 Table MOVIE

The primary key of this table is on the combination of columns (Title, Year, Actor).

Names of directors and actors are considered as a whole: they don't have to be separated in first name, middle name (if any) and surname.

There are movies with the same title, but released in a different year (for example: a remake). These might have the same director and/or one or more same actors. For each film, only one director is stored, and at least one actor.

One of the functional dependencies in this table is:
(Title, Year) → Director

Tasks

a There is redundancy in table MOVIE: some facts are stored more than once. Give a verbalization of all the complete elementary facts that appear more than once in the population shown in figure E1.1.1.

b The cause of the redundancy is that the table in not in second normal form (2NF). Why is this table not in 2NF?

c Give one or more tables in 2NF to store the same information in. For each table, give the primary key, and foreign key references (if any). Add the complete population from figure E1.1.1.

d Account for your design: show that the new tables are indeed in 2NF now.

e Are the new tables also in 3NF or BCNF?

f Verify that the redundancy is now removed.

Exercise 1.2 Theater schedule

Before the start of a new season, a theater registers which performances are scheduled on which dates and times, and what the prices of tickets are. The theater is using the SCHEDULE table in figure E1.2.1 below, together with a small example population.

SCHEDULE

Date ←	→ Time	Show Number	Show Name	Normal Ticket	Season Ticket
5-03-2008	20:15	24	Ballet Don Quichote	40	35
6-03-2008	14:00	25	Cowboy Billie Boem	15	12
6-03-2008	20:15	26	Lebbis & Jansen	25	20
7-03-2008	20:30	27	Aïda	50	45
8-03-2008	20:30	27	Aïda	50	45

Figure E1.2.1 Table SCHEDULE

Every performance features a show, which is identified by a show number.
The functional dependencies are:

Show Number → Show Name
Show Number → Normal Ticket
Show Number → Season Ticket.
(Date, Time) → Show Number

Tasks

a There is redundancy in table SCHEDULE: some facts are stored more than once. Give a verbalization of all the complete elementary facts that appear more than once in the population shown in figure E1.2.1.

b Show that the table is in 2NF, but not in 3NF.

c Give one or more tables in 3NF to store the same information in. For each table, give the primary key, and foreign key references (if any). Add the complete population from figure E1.2.1.

d Account for your design: show that the new tables are indeed in 3NF now.

e Are the new tables also in BCNF?

f Verify that the redundancy is now removed.

Exercise 1.3 Cars

Consider table CAR in figure E1.3.1, used by a service station (garage) for the cars it services.

CAR

License Plate	Chassis Number	Owner	Name	Fuel
PB-72-ZT	SB172ZBN10E005652	123	J de Vries	regular
35-LK-BG	TG367BZN93G000448	667	K Verstappen	diesel

Figure E1.3.1 Table CAR

License Plate: a plate with a unique code, issued to every car in the Netherlands.
Chassis Number: a unique code issued to the chassis of a car by the car manufactuirer.
Owner: the present owner of the car, identified by a unique owner number.
Name: is the name of the owner.
Fuel: is the type of fuel the car takes (only one type is recorded for each car).

Tasks

a Is there redundancy possible in this table? If so, extend the population of the table so it contains redundant facts, and give a verbalization of the complete redundant facts.

b Determine the two candidate keys of this table, and choose one primary key (the other will be the alternative key).

c Determine all the basic functional dependencies (FDs) in this table.
The FDs should meet the following requirements:
R1: The left-hand side and the right-hand side both contain as few attributes as possible.
R2: No transitive FDs:
If we have FDs P → Q and Q → R, then we do not also give the transitive FD P → R.
Because there is an alternative key, you do not have to consider any FDs from the complete alternative key to other attributes. All other FDs must be determined, however.

d In which normal form is table CAR? Carefully account for your answer.

e If table CAR is not in 3NF, then design a table structure that is in 3NF, and fill the table(s) with the same example population as is shown in figure E1.3.1. If you found any redundancy in task a, is this redundancy now removed?

Exercise 1.4 Skating races

Table Tournement Winners contains data about all-round skating tournements for men.

Tournament Winners

Tournament Type	Year	Town	Country	Winner ISU-no	Winner Name	Winner Country
WC	2006	Calgary	Canada	3005	Shani Davis	USA
ECp	2006	Hamar	Norway	2935	Enrico Fabris	Italy
WC	2007	Heerenveen	Netherlands	3064	Sven Kramer	Netherlands
EC	2007	Collalbo	Italy	3064	Sven Kramer	Netherlands

Figure E1.4.1 Table Tournament Winners

Explanation of the table

WC World Championship
EC European Championship
ISU-no International Skating Union number, unique for each skating contestant.
Towns have a unique name.
The name of a skater is not always divided into surname and first name, so the name is regarded as a single data item. A skater name that belongs to an ISU-no cannot be changed.
Skaters can represent different countries in different tournaments.

Facts in this table (verbalizations from the top row)**:**

```
The WC tournament of 2006 was held in Calgary.
Calgary is a town in Canada.
The skater with ISU-no 3005 won the WC tournament of 2006.
The name of skater 3005 is Shane Davis.
In the WC tournament of 2006, the winner represented the USA.
```

Tasks

a Is there redundancy present in the given small population of this table?
If so, give a verbalization of the complete redundant facts.

b Are there other places where redundancy is possible in this table? If so, extend the population of the table so it contains redundant facts in these places as well, and give a verbalization of these complete redundant facts.

c This table is in Second Normal Form (2NF). Show that this is true.

d Is this table also in Third Nornal Form (3NF)? Motivate your answer.

e If the table is not in 3NF, then give one or more tables in 3NF and add the given population. Show that any redundancy in the table above is now removed.

Exercise 1.5 Invoice

Invoice

Invoice No	Cust-omer	Customer Name	Address	Town	Article	Price	Amount	Seller	Seller Name	Sub-total	VAT	Total
121	67	Jansen	Bergstraat 44	Arnhem	2177	3,50	3	1	De Groot	10,50	6%	15,50
121	67	Jansen	Bergstraat 44	Arnhem	3550	2,50	2	1	De Groot	5,00	19%	15,50
122	67	Jansen	Bergstraat 44	Arnhem	2177	3,50	1	2	De Boer	3,50	6%	3,50
123	89	Meurs	Mozartlaan 7	Velp	4609	1,50	4	2	De Boer	6,00	19%	11,00
123	89	Meurs	Mozartlaan 7	Velp	3550	2,50	2	2	De Boer	5,00	19%	11,00

Figure E1.5.1 Table Invoice

Explanation of the table

Every invoice is for exactly one customer.
A customer can receive several invoices.
Several articles can be billed on the same invoice.
The price per article (in Euros) is the same for all customers.
To each article (which is identified by an article number), a VAT (Value Added Tax) charge applies (6% or 19%), which is already included in the price.
Only one seller is responsible for each invoice.
There is no regular association between a customer and a seller (no fixed accounts).
A customer is identified by a customer number, and a seller by a seller number.
Only one address is recorded for each customer. The combination of street + house number is regarded as a single value, and so is the name of a customer.

Tasks

a Show whether this table is in 1NF, 2NF, 3NF or BCNF.

b If the table is not in BCNF, give a relational table structure in BCNF to store the same information, and fill it with the same facts as in the original table.

c Show that this table structure is indeed in BCNF.

1.3 Theme 3 Drawing up an ERM diagram

Exercise 3.1 Verbalizing

This exercise consists of a few verbalization tasks.

Exercise 3.1.1 Choir member borrowings

Consider the following table **Borrowings**. A description of this table is given below.

Borrowings

PieceNo	Title	Copy	Purchased	MemberNo	Name
…	…	…	…	…	…
81	Jesu meine Freude	7	1/10/2002	16	Willemse
81	Jesu meine Freude	8	1/10/2002	17	De Jong
…	…	…	…	…	…
81	Jesu meine Freude	23	7/12/2003	25	Boomsma
81	Jesu meine Freude	24	7/12/2003	--	--
82	Requiem	1	15/10/2002	--	--
…	…	…	…	…	…
99	Requiem	1	1/10/2002	--	--
…	…	…	…	…	…

Figure E3.1.1.1 Table Borrowings

Description

The members of a choir are identified by a unique member number. The surname of each member is also recorded. (Many other data are recorded for members as well, but these are not relevant for this task). A piece of music is identified by a unique number too, and the title is of interest (other data are recorded for pieces of music as well, but these are not relevant for this task). If the choir buys copies of sheet music of a composition, then each copy is numbered sequentially starting with 1 for each piece of music. Usually the choir buys one copy for each member, but sometimes more (if the choir is expected to grow) or less (if choir members own a copy themselves already). Sometimes additional copies of sheet music are bought later (for example for a new performance at a later concert, if the choir doesn't have enough copies for everyone). Then the numbering is simply continued. The purchase date of each copy is recorded, and can therefore be different for different copies. If a member borrows a copy, this is recorded. After a concert, the borrowed copies are returned to the librarian of the choir, and the data of the loan are deleted (only the members that are presently borrowing a copy are recorded).

The task is to verbalize the facts in this table (two facts per fact type will suffice). Make the meaning of the facts as clear as possible.

Exercise 3.1.2 Schedule for the Rio olympics

See the following figure, which should be self-explanatory as it was published without comment.

OC Opening ceremony • Event competitions 1 Gold medal events EG Exhibition gala CC Closing ceremony

August	3 Wed	4 Thu	5 Fri	6 Sat	7 Sun	8 Mon	9 Tue	10 Wed	11 Thu	12 Fri	13 Sat	14 Sun	15 Mon	16 Tue	17 Wed	18 Thu	19 Fri	20 Sat	21 Sun	Gold medal events
Ceremonies (opening / closing)			OC																CC	
Archery				1	1	•	•	•	1	1										4
Athletics										3	5	4	5	5	4	6	7	7	1	47
Badminton									•	•	•	•	•	•	1	1	2	1		5
Basketball				•	•	•	•	•	•	•	•	•	•	•	•	•	•	1	1	2
Boxing				•	•	•	•	•	•	•	•	1	1	1	1	1	1	3	4	13
Canoeing – Slalom					•	•	1	1	2											16
Canoeing – Sprint													•	4	•	4	•	4		
Cycling – Road cycling				1	1			2												18
Cycling – Track cycling									1	2	2	1	1	3						
Cycling – BMX															•	•	2			
Cycling – Mountain biking																		1	1	
Diving					1	1	1	1		•	•	1	•	1	•	1	•	1		8
Equestrian				•	•	•	2	•	•	1		•	1	•	1		1			6
Fencing				1	1	1	1	2	1	1	1	1								10
Field hockey				•	•	•	•	•	•	•	•	•	•	•	•	1	1			2
Football	•	•		•	•		•	•		•	•			•	•		1	1		2
Golf									•	•	•	1			•	•	•	1		2
Gymnastics – Artistic				•	•	1	1	1	1			4	3	3	EG					18
Gymnastics – Rhythmic																	•	1	1	
Gymnastics – Trampolining										1	1									
Handball				•	•	•	•	•	•	•	•	•	•	•	•	•	•	1	1	2
Judo				2	2	2	2	2	2	2										14
Modern pentathlon																•	1	1		2
Rowing				•	•	•	•	2	4	4	4									14
Rugby sevens				•	•	1	•	•	1											2
Sailing						•	•	•	•	•	•	2	2	2	2	2				10
Shooting				2	2	2	1	2	1	2	2	1								15
Swimming				4	4	4	4	4	4	4	4		1	1						34
Synchronized swimming												•	•	1		•	1			2
Table tennis				•	•	•	•	1	1	•	•	•	•	1	1					4
Taekwondo															2	2	2	2		8
Tennis				•	•	•	•	•	•	1	1	3								5
Triathlon																1		1		2
Volleyball – Beach volleyball				•	•	•	•	•	•	•	•	•	•	•	1	1				4
Volleyball – Indoor volleyball				•	•	•	•	•	•	•	•	•	•	•	•	•	•	1	1	
Water polo				•		•	•	•	•	•	•	•	•	•	•	•	1	1		2
Weightlifting				1	2	2	2	2		2	1	1	1	1						15
Wrestling												2	2	2	3	3	2	2	2	18
Total gold medal events				12	14	14	15	20	19	24	21	22	17	25	16	23	22	30	12	306
Cumulative total				12	26	40	55	75	94	118	139	161	178	203	219	242	264	294	306	
August	3 Wed	4 Thu	5 Fri	6 Sat	7 Sun	8 Mon	9 Tue	10 Wed	11 Thu	12 Fri	13 Sat	14 Sun	15 Mon	16 Tue	17 Wed	18 Thu	19 Fri	20 Sat	21 Sun	Gold medal events

Figure E3.1.2.1 Schedule of the 2016 olympic games in Rio de Janeiro

The task is to verbalize the facts in this figure. Make the meaning of the facts as clear as possible.

Exercise 3.2 Exam results

Consider the following tables with information about courses, teachers, students and exams. Teachers are identified by a teacher code, and students by a student number. Course realizations are distinguished by their course name and starting date (several course realizations can start on the same day, but not of the same course). Every course realization has at least two exam opportunities. Students can enroll for several course realizations. Every course realization is given by one teacher, and can be assisted by one or more students, but a student is only allowed to assist one course realization. If a student takes an exam, the result is recorded. (In a more realistic case, there would be many more attributes for students, teachers etc., but for this exercise these are enough). Verbalizations, approved by the domain experts, are given below the tables.

Course	Date	Teacher	As-sis-tent	Exams
SQL	20160104	Wht (White)	S2	20160307 20160412
SQL	20160502	Brw (Brown)	S7	20160620 20160704 20160711
FO-ERM	20160307	Wht (White)	S8 S9	20160509 20160606
…	…	…	…	…

Stu-dent	Name	Courses	Exam results
S1	Jones	SQL 20160104	20160307: 45 20160412: 72
S2	Smith	SQL 20160104 FO-ERM 20160307	-- 20160509: 81
S7	Doyle	FO-ERM 20160307	--
…	…	…	…

Figure E3.2.1 Courses and students

Verbalizations (already sorted in ascending order of the number of components):

```
FT1:
The surname of teacher Wht is White.
 "     "       "     "     Brw  " Brown.
FT2:
The surname of student S1 is Jones.
 "     "       "     "     S2   " Smith.
FT3:
The SQL course realization of 20160104 was taught by teacher Wht.
 "  SQL     "         "        " 20160502  "      "     "     "    Brw.
 "  FO-ERM  "         "        " 20160307  "      "     "     "    Wht.
FT4:
Student S1 has enrolled for the SQL course realization of 20160104.
   "    S2  "       "      "    "  SQL     "         "        " 20160104.
   "    S2  "       "      "    "  FO-ERM  "         "        " 20160307.
FT5:
Student S2 assisted the SQL course realization of 20160104.
   "    S7      "      "  SQL     "         "        " 20160502.
   "    S9      "      "  FO-ERM  "         "        " 20160307.
FT6:
There is an exam for the SQL course realization of 20160104 on 20160307.
  "    "   "    "    "    "   SQL     "         "        " 20160104  " 20160412.
  "    "   "    "    "    "   FO-ERM  "         "        " 20160307  " 20160509.
FT7:
In the exam on 20160307 of the SQL course realization of 20160104, student S1 scored 45 %.
 "  "    "    " 20160412  "  "  SQL     "         "        " 20160104,         S1     "   72 ".
 "  "    "    " 20160509  "  "  FO-ERM  "         "        " 20160307.         S2     "   81 ".
```

Task: Make an ER-model by analyzing the verbalizations, and create the diagram in PowerDesigner.

Hint: In FT6, underline only one segment and treat it as is shown in section 4.4.5 of the Reader DM-RDS. Add predicates and example populations where you think they help to understand the model better (at least for two fact types).

Exercise 3.3 Solar eclipses

Consider the information about solar eclipses in the figures below. Similar information is available for eclipses in other cities and years (not shown). You can omit the columns 'Object', 'Path of the eclipse' and 'Looks like'.

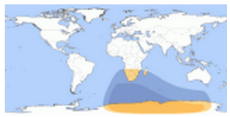
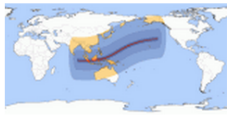
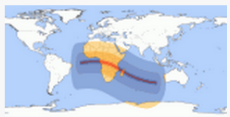
Dates	Object	Type	Looks like	Where is the eclipse visible	Path of the eclipse
2015					
20 maa 2015	Solar	Total		Europe, North/East Asia, North/West Africa, West in North America, Atlantic, Arctic	
13 sep 2015	Solar	Partial		South in Africa, Atlantic, Indian Ocean, Antarctica	
2016					
8 maa / 9 maa 2016	Solar	Total		South/East Asia, North/East Australia, Pacific, Indian Ocean	
1 sep 2016	Solar	Annular		South in Asia, East in Australia, Much of Africa, Atlantic, Indian Ocean, Antarctica	

Figure E3.3.1 Solar Eclipses in 2015 and 2016

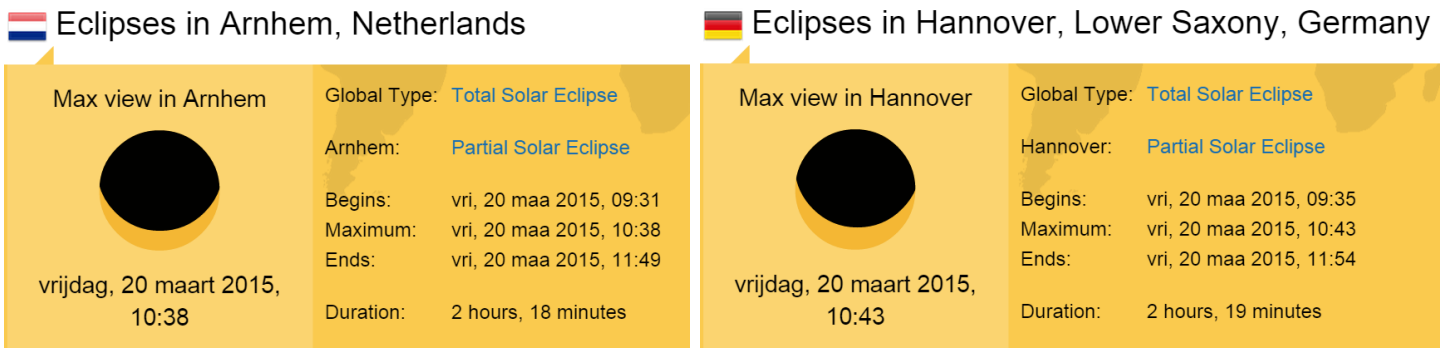


Figure E3.3.2 Details of the solar eclipses of March 20, 2015 in Arnhem and Hannover.

Tasks

a Verbalize the facts on the example documents. Give for each fact type two verbalizations.

b Analyze the verbalizations.

c Derive an ER-model (CDM) from the analysis and enter it into PowerDesigner. Add predicates and an example population to the ER-model for at least one <pi>-Att pair and one RT.

Exercise 3.4 Conference workshops

Participants in a conference on information science can enroll for a number of workshops. Each workshop has a unique code and a title. The workshops take place in four sessions (session 1 runs from 9:15 to 10:45, session 2 from 11:00 to 12:30, etc.). A workshop is held in one of the rooms of the conference center, and can be held in more than one session. There are two or more workshops in each session. All participants receive a unique participant number, and their name is recorded and written on their badge.

Here is the conference schedule, and a few enrollments of participants in the workshops:

Session	Time	Workshop	Title	Room	Participants
1	9.15 – 10.45	A	Model Driven Architecture	Eagle	99 Janny Mulder 12 Jan Verkerk 40 Anke de Vries … ……
		B	Scrum Project Management	Cockatoo	33 Carel Doorman 62 Fred Overbeek 15 Geert van Straaten … ……
2	11.00 – 12.30	C	Extreme Programming	Eagle	33 Carel Doorman 99 Janny Mulder 62 Fred Overbeek … ……
		D	Data Mining	Cockatoo	17 Pieter Carelse 58 Marten Goudsmid 12 Jan Verkerk … ……
		E	LINQ	Toucan	86 John van de Berg 62 Fred Overbeek 15 Geert van Straaten … ……
3	13.30 – 14.00	F	Enterprise Application Integration	Cockatoo	86 John van de Berg 55 Maria Cornelisse 41 Joris Dekker … ……
		G	XML Databases	Toucan	90 Vera van Aartsen 62 Fred Overbeek 15 Geert van Straaten … ……
4	14.15 – 15.45	B	Scrum Project Management	Cockatoo	86 John van de Berg 41 Joris Dekker 99 Janny Mulder … ……
		D	Data Mining	Toucan	90 Vera van Aartsen 33 Carel Doorman 38 Jack Evans … ……
		E	LINQ	Eagle	17 Pieter Carelse 12 Jan Verkerk 40 Anke de Vries … ……

Figure E3.4.1 Conference schedule

The conference schedule and enrollment data are to be stored in a small database. page.

Tasks

a Verbalize the examples given in figure E3.2.1 (one fact per sentence). Give at least two verbalizations for each fact type. Use sentences that make the meaning of the facts as clear as possible.

b Sort and analyze the verbalizations, and build an ERM diagram based on these analyses. Add predicates and example populations to all non-dependent relationship types, and to key-attribute fact types that are not self-evident.

Exercise 3.5 Meal recipes

Consider the general description of relevant data in the context of recipes for meals in figure E3.3.1.

An enthousiastic amateur cook wants to start a well-organized automated administration for the many recipes of the meals he prepares. It concerns a large number of meals from various countries.

A meal consists of several courses. In each course, one dish is served. A course has a sequence number within a meal and a name, which is unique for a course in a meal (the same sequence numbers in different meals do not have to have the same name). For each meal the name (unique), the country of origin and the type of meal (vegetarian, ..) is to be recorded.

A dish has a unique name, and can occur as a course in several meals from the same country. A short method of preparation is included for each dish. To prepare a dish, several ingredients are listed (sometimes the same ingredient is used more than once in the same recipe, in different quantities). For each use of an ingredient in a dish, the required quantity must be recorded. Ingrediënts in a dish also have a sequence number. Because the cook regularly cooks for people on a diet, the number of kilocalories per 100 gram or per 100 millilitre must be known as well, and also the percentage of salt. An ingredient has a unique name.

Sometimes there is an alternative ingredient for a certain ingredient in a dish (like macaroni or rice instead of spaghetti). In such a case the required quantity of the replacing ingredient must be recorded.

Figure E3.5.1 Description of meal recipes

Tasks

a Make concrete examples (in any convenient form) for all the relevant facts: at least two for each fact type. Do not make empty recipes, menus, lists, tables or forms, but invent plausible concrete names for ingredients, meals, etc. What assumptions do you have to make to be able to do this because of missing information in figure E3.5.1? Make plausible assumptions for such missing information and write those down.

b Verbalize your concrete examples. Make the sentences as clear in meaning as you can.

c Sort and analyze your verbalizations, and build an ERM diagram based on these analyses. Add predicates and example populations to all non-dependent relationship types, and to key-attribute fact types that are not self-evident.

Exercise 3.6 Council Elections

In the Netherlands, elections for town councils are held every four years. In every town, political parties present a list of their candidates for the council in that town. In each town, these lists are numbered sequentially starting from 1. The party that has the most members in the present town council gets list number 1, the party with the next greatest number gets list number 2, and so on. For example, in Oss list number 1 was assigned to the party CDA, but in Nijmegen list number 1 was assigned to the party PvdA. Some large parties have a list in every town, but there are also small local parties that run only in a few towns (or even in only one).

On each list, the candidates have a sequence number. This identifies the candidate in the list, and also determines their default priority in the election. For example, if the party wins enough votes to claim five seats in the council, but all votes were cast on candidate number 1, then still the first five candidates are elected and can claim their seat in the council. For each candidate the initials, surname, prefix if any (a short word like 'van' or 'de' that is part of a surname but is not used in ordering surnames alphabetically) and gender are recorded.

A voter in a town can vote for exactly one of the candidates on one of the lists. In each town there are several polling stations where voters can cast their vote. The same set of lists is used in all the polling stations of the same town. A polling station has a unique station number and a name. At the end of the election day, the total number of votes won by each candidate at each polling station is counted and recorded.

Figure E3.6.1 shows a few examples of the election result of one polling station in the town Oss.

Polling station 24 – 't Oude Theater												
1				**2**				**3**				**4**
CDA				**SP**				**VVD**				**…**
1	G Louwers (m)	88		1	LH van Doren (v)	226		1	HG Nijenkamp (m)	29		
2	H Peters (v)	14		2	M Scheepens (v)	19		2	MHN Huiskens (m)	4		
3	M Mensink (m)	9		3	HGM Vervoort (m)	12		3	G de Bruijn (m)	1		
4	S Jansen (m)	11		4	S Veulings (m)	7		4	JF Soulier (m)	27		
5	J Reijmer (v)	5		5	AD Ghouassi (v)	9		5	HMM van Ruberg (v)	1		
..	……	...		…	……	...		…	……	...		

Figure E3.6.1 Part of election result of one polling station

Tasks

a Suppose that Oss is the only town in which we are interested. So only the lists and polling station results from Oss are to be recorded. Verbalize the concrete examples given in figure E3.4.1. It is enough to verbalize two facts for each fact type (two concrete facts of the same kind). Each verbalization should contain only one fact, which is to stated as clearly as possible.

Since only Oss is considered, the name 'Oss' does not have to be used in the verbalizations.

b Sort and analyze the fact expressions from task a, and give the resulting ERM diagram. If the name of the town (Oss) does occur in these fact expressions, then do not treat is as a component, but as fixed (unvarying) text.

c Are there any business rules that cannot be shown graphically in the ERD?

d Extend the verbalizations, the analysis and the model so it will now capture the council elections in all towns in the Netherlands where elections are held. Include a domain list of all towns in the Netherlands with a council in the analysis (a separate ET for towns with councils).l

e Are there new business rules now?

Exercise 3.7 Book list

In the figures below, example documents from a secondary school are shown. They concern the books students will use for subjects treated in class. Excerpts from interviews with domain experts are used to explain the contents.

Book list

On our school, the book list shown in figure E3.7.1 is issued to all our pupils. They have to buy some of these books for the subjects they are taught, but this depends on the class they're in (see figure E3.7.3). For convenience, we use our own unique book number, although books of course also have a unique ISBN, but that is so long and error-prone. For a new book in press the price might not yet be known, which is indicated as '??'. Is the rest of the table clear?.

Boekno	ISBN	Title	Publisher	Price
11	9035622459	Bridge Math A	Liemers	€ 34,50
12	9035656438	Bridge Math B	Liemers	€ 34,50
13	9045327845	Math plus I	Liemers	€ 37,95
14	9045327895	Math plus II	Liemers	€ 38,25
15	9055437856	Math and Matics I	Zaal BV	€ 35,00
…				
34	9056489323	Intro to Geo	Bartels	€ 35,00
35	9056438764	Atlas	Vellinga's	??
…	…	…	…	…

Figure E3.7.1 Book list

Departments and classes

Our school is divided in three departments that have the classes indicated in figure E3.7.2:

Department	Class
Bridge	1, 2
MAVO	3, 4
HAVO	3, 4, 5

Figure E3.7.2 Departments and classes

Subjects and books per class

Figure E3.7.3 shows a few examples of which subjects are taught in how many hours per week to which classes and which books are to be used by the pupils. The numbers between parentheses following the book numbers indicate the edition of the book. Unfortunately, the ISBN does not change for a new edition, even if changes were made: this only happens after a major revision so the book can be regarded as 'new'.

Department	Class	Subject	Hours	Booknos
Brug	1	MA	4	11(4), 12(4)
		GEO	3	34(2), 35(2)
		…	…	…
	2	MA	3	13(3)
		GEO	2	35(2)
		…	…	…
MAVO	3	MA	4	13(2)
		GEO	3	35(1)
…	…	…	…	…

Figure E3.7.3 **Subjects, hours and books**

a Verbalize the examples given in figure E3.7.1 – E3.7.3 (one fact per sentence). Give at least two verbalizations for each fact type. Use sentences that make the meaning of the facts as clear as possible.

b Sort and analyze the verbalizations, and build an ERM diagram based on these analyses. Explore several possibilities for adding domain lists (for class, department or publisher). Choose one possibility, and give reasons for your choice.

c Add predicates and example populations to all non-dependent relationship types, and to key-attribute fact types that are not self-evident.

1.4 Theme 4 Deriving a PDM from a CDM

Exercise 4.1 Employee-department-location

Task a

Figure E4.1.1 shows a complete ERD.

Derive a logical relational schema (LRS) from this ERD. Draw the LRS by hand in the form of a complete PowerDesigner PDM, including all domains, join conditions, cardinalities etc.

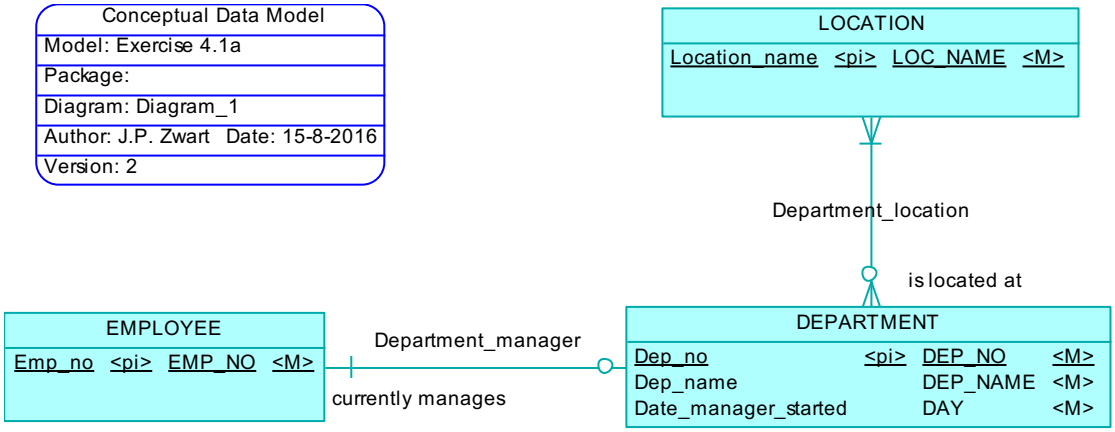


Figure E4.1.1 ERD for employee-department-location

Task b

Figure E4.1.2 shows a complete ERD.

Derive a logical relational schema (LRS) from this ERD. Draw the LRS by hand in the form of a complete PowerDesigner PDM, including all domains, join conditions, cardinalities etc. Add integrity constraints that cannot be shown graphically (if any) in a text block.

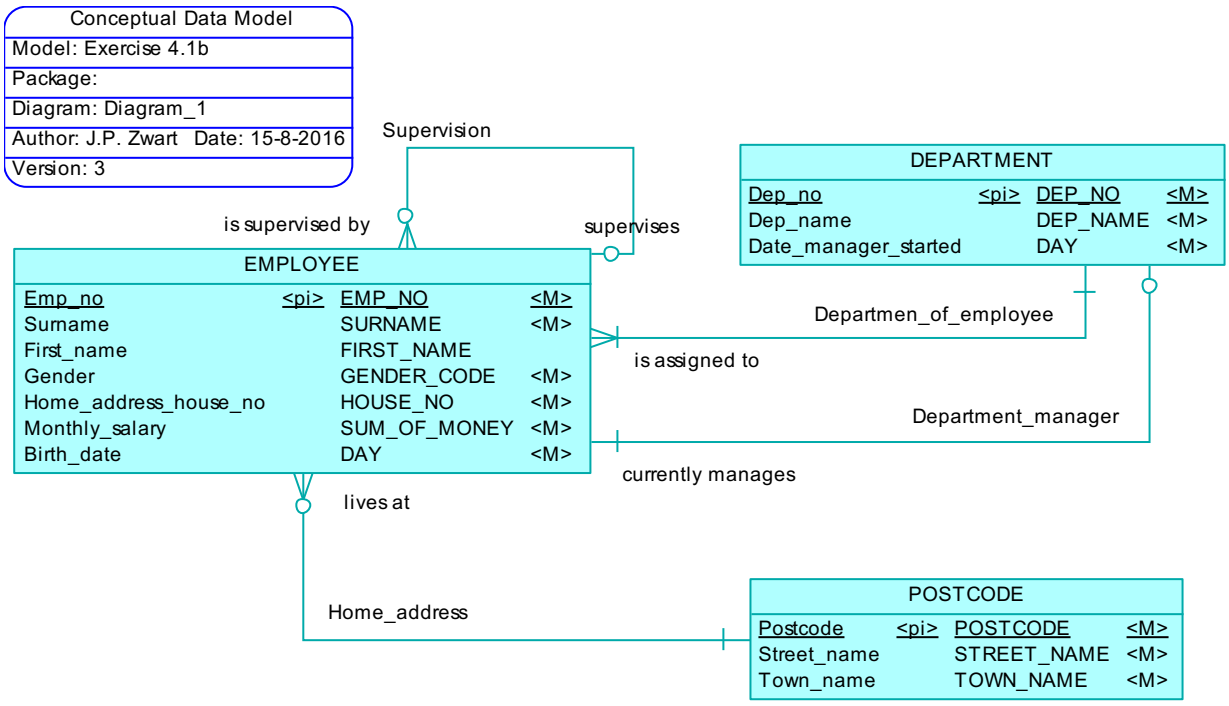


Figure E4.1.2 ERD for employee-department-postcode

Exercise 4.2 Participant-trip-hotel

Consider the ERD in figure E4.2.1. It shows a correct conceptual data model.

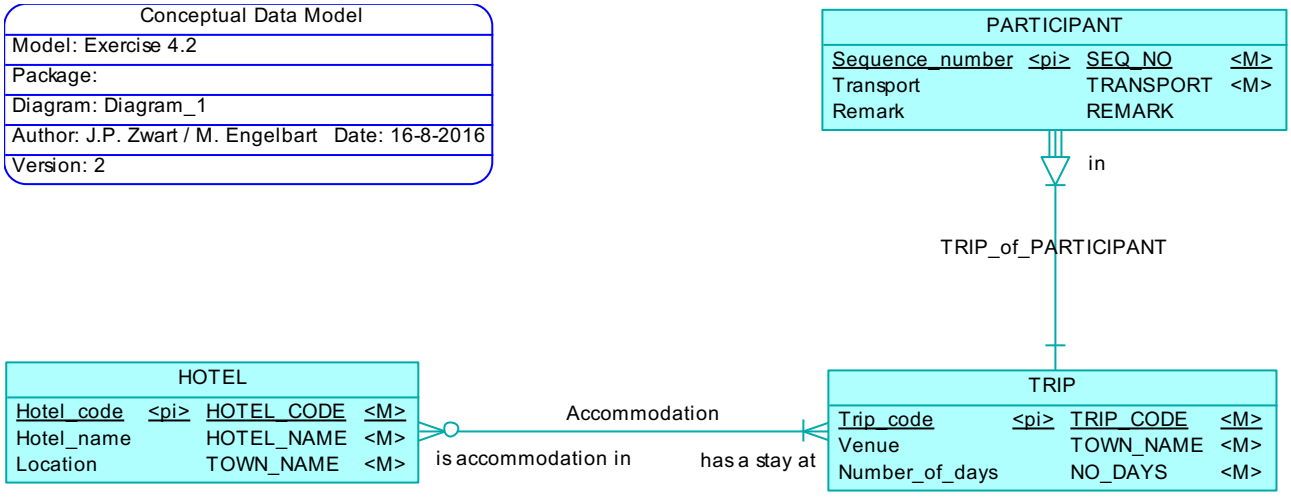


Figure E4.2.1 ERD for participant-trip-hotel

Task a

Are the following statements about this ERD TRUE or FALSE? Mark the appropriate checkbox.

i. The means of transport must be recorded for each participant.
☐ TRUE ☐ FALSE

ii. The identifier of a participant is Sequence_number together with Trip_code.
☐ TRUE ☐ FALSE

iii. Every trip has a stay in at least one hotel.
☐ TRUE ☐ FALSE

iv. A hotel can be registered that is not the accommodation in any trip.
☐ TRUE ☐ FALSE

v. Because Location in HOTEL and Venue in TRIP have the same domain, they will be stored in the same table.
☐ TRUE ☐ FALSE

Task b

Figure E4.2.2 contains a part of the PDM that can be generated from the CDM in figure E4.2.1. Complete the PDM by adding all the missing parts (including join conditions, cardinalities at the foot of references, etc.). Add integrity constraints that cannot be shown graphically (if any) in a text block.

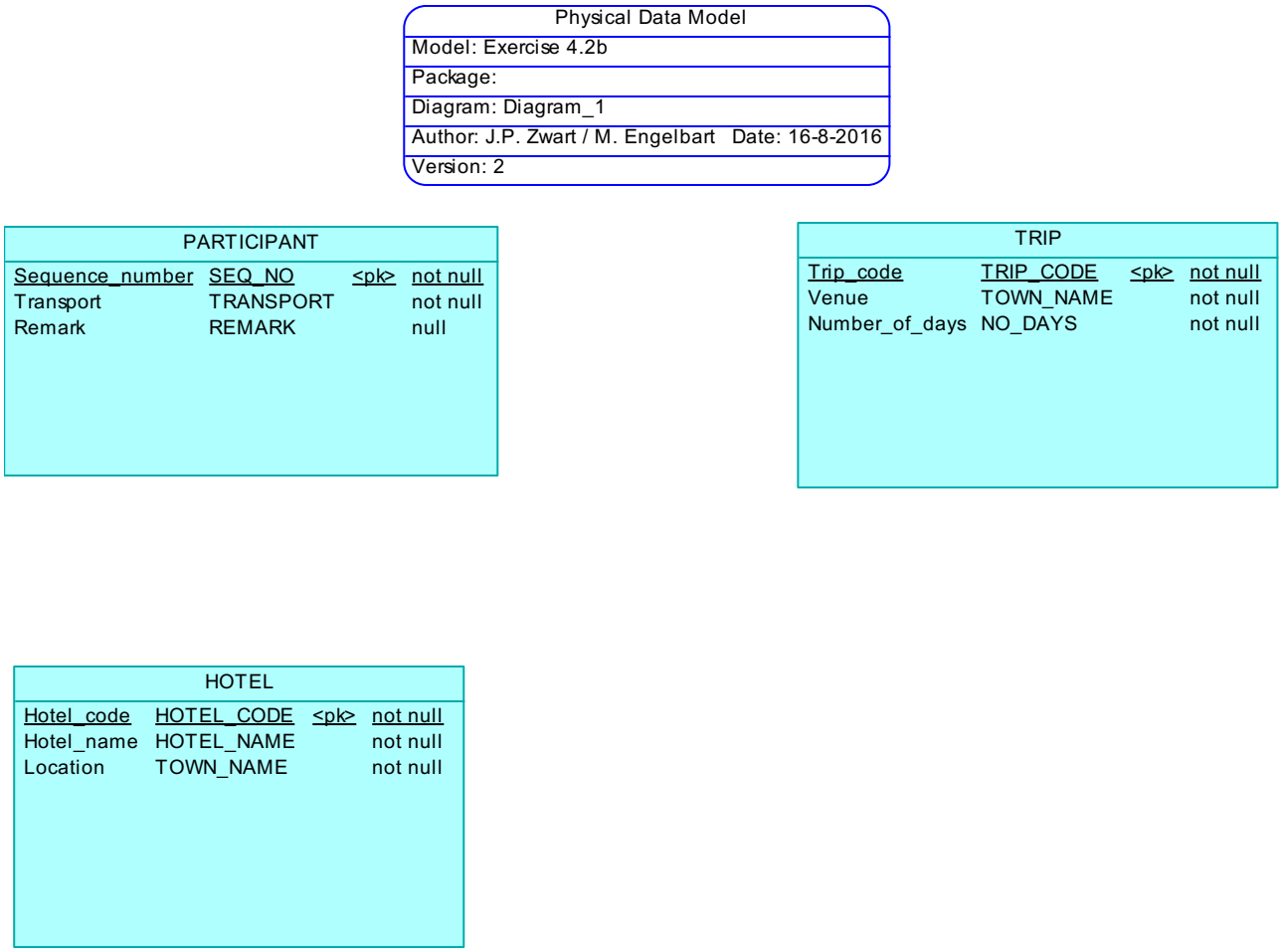


Figure E4.2.2 Part of PDM derived from figure E4.2.1

Exercise 4.3 Employee-car-building

Consider the ERD in figure E4.3.1. It shows a correct conceptual data model.

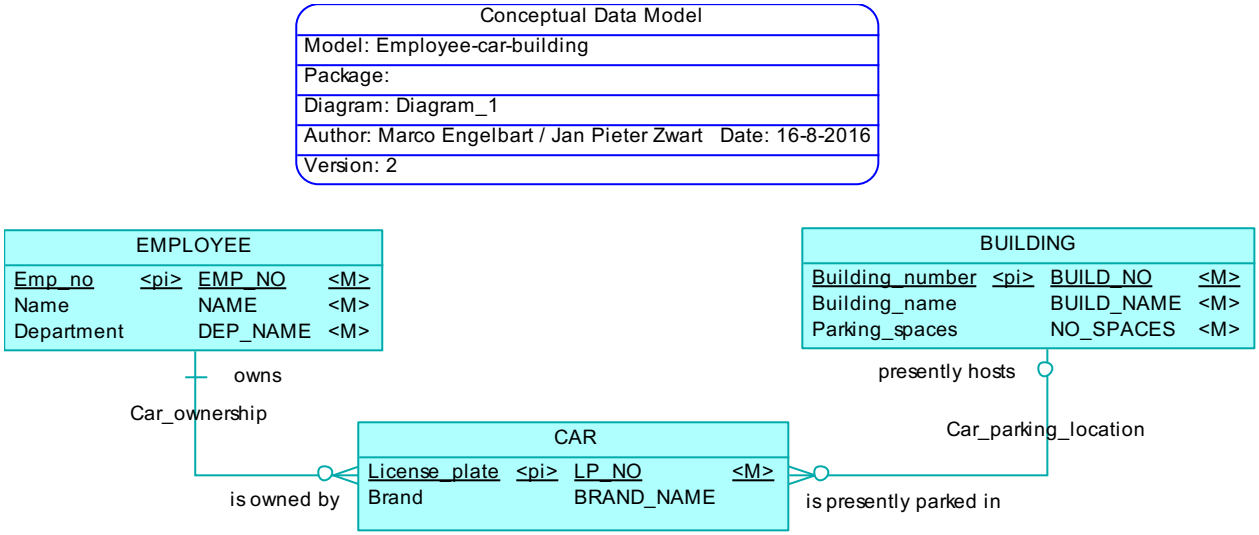


Figure E4.3.1 ERD for employee-building-car

Task a

Can a database built according to this data model supply the answers to the following questions?

- Q1: In which building did employee EM123 park his car?
- Q2: In which parking space in which building is the car with license plate KA-12-DE parked?

Task b

Are the following situations allowed according to this data model?

- S1: An employee can park several cars in the buildings.
- S2: A car can have parking spaces in several buildings.

Task c

Have the following business rules been modeled in this data model?

- B1: A car must always be parked in the same building.
- B2: A building can also host cars of non-employees (guests).

Task d

Transform this CDM into a complete PDM. Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.4 Marriages

Consider the ERD in figure E4.4.1. It shows a correct conceptual data model.

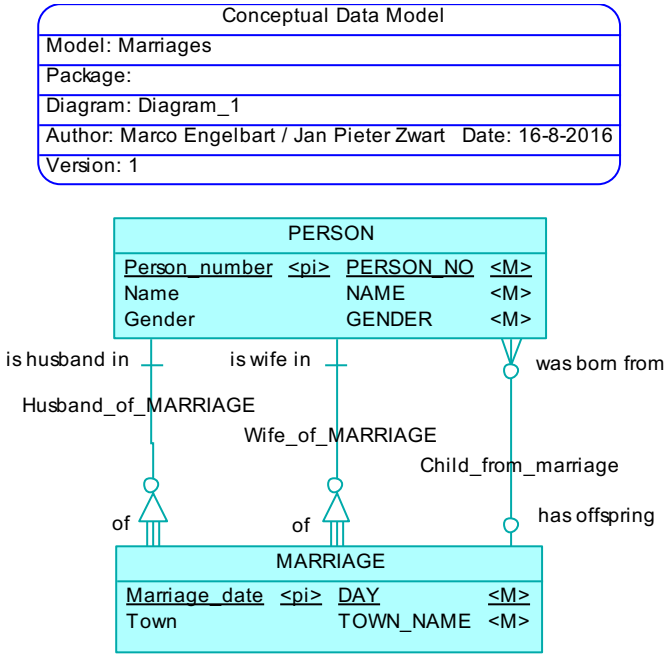


Figure E4.4.1 ERD for marriages

Task a

Can a database built according to this data model supply the answers to the following questions?

- Q1: Who are the parents of person 643?
- Q2: Of how many children is person 555 the father?

Task b

Are the following situations allowed according to this data model?

- S1: A person can be married to someone with the same gender.
- S2: The model is flawed: it allows a person to be married with himself/herself.
- S3 A marriage can be without offspring.
- S4: The model is flawed: it allows a person to marry several other persons on the same day.
- S5: A child can have more than one parent of the same gender.

Task c

Have the following business rules been modeled in this data model?

- B1: Only married persons can be registered.
- B2: Illegitimate children can also be registered.

Task d

Transform this CDM into a complete PDM. Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.5 Exam results

Take the CDM from the answer to exercise 3.2 (either your own version, or one supplied by your teacher), and derive a PDM from it automatically in PowerDesigner. Check it carefully, correct it where needed, and make any changes you would like (give reasons for your alterations). Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.6 Solar eclipses

Take the CDM from the answer to exercise 3.3 (either your own version, or one supplied by your teacher), and derive a PDM from it automatically in PowerDesigner. Check it carefully, correct it where needed, and make any changes you would like (give reasons for your alterations). Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.7 Conference workshops

Take the CDM from the answer to exercise 3.4 (either your own version, or one supplied by your teacher), and derive a PDM from it automatically in PowerDesigner. Check it carefully, correct it where needed, and make any changes you would like (give reasons for your alterations). Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.8 Meal recipes

Take the CDM from the answer to exercise 3.5 (your own version) and derive a PDM from it automatically in PowerDesigner. Check it carefully, correct it where needed, and make any changes you would like (give reasons for your alterations). Add integrity constraints that cannot be shown graphically (if any) in a text block.

Exercise 4.9 Council Elections

Take the CDM from the answer to exercise 3.6 – the version with all towns from figure A3.6.7 - (either your own version, or one supplied by your teacher), and derive a PDM from it automatically in PowerDesigner. Check it carefully, correct it where needed, and make any changes you would like (give reasons for your alterations). Add integrity constraints that cannot be shown graphically (if any) in a text block.

Note: the table ELECTION_RESULT will contain one column too many. Delete the spurious column.

1.5 Theme 5 Subtypes in ERM

Exercise 5.1 Workstations

The ICT department of a business keeps track of all the workstations used by employees. The following geberal description was given by domain experts:

> A workstation consists of a computer and a monitor. Every device (computer or monitor) has a unique device number. The brand and purchase date of each device are recorded as well. For computers, the processor type, hard disk capacity and internal memory capacity are registered. For monitors, the screen size (for example 17 inches) is relevant.

Figure E5.1.1 General description for workstations

From documents with examples of concrete facts (not shown here), an ERD with an entity type DEVICE was drawn up (see figure E5.1.2).

DEVICE			
Device_number	<pi>	DEV_NO	<M>
Brand		BRAND_NAME	<M>
Date_purchased		DAY	<M>
Processor		PROC_TYPE	
Hard_disk_ (GB)		CAPACITY	
RAM_(GB)		CAPACITY	
Diameter_(inch)		SIZE	

Figure E5.1.2 ERD for devices

Task a

Give business rules that specify for which devices the optional attributes are to have values.

Task b

Can the business rules from task a be translated into constraints?

- If so, give these constraints in terms of attributes from figure E5.1.2 and values they have (do not use subtypes yet).
- If not, what information is missing in figure E5.1.2? Add this missing information to the model, then give the constraints (do not use subtypes yet).

Task c

Add an example population to ET DEVICE (you do not have to supply predicates) that is consistent with the constraints (give the values in the same order as the attributes, and use '—' for a NULL value).

Task d

Replace the constraints from task b by subtypes. Give Subtype Derivation Rules and subtype defining fact types in the form of predicates associated with the subtypes. Also indicate whether the subtypes are exclusive and/or complete. Add the same example population as you used in task c.

Task e

Derive two relational schema's from the result of task d: one with a separate table for each subtype, and one with only a table for the supertype. Give equivalent Subtype Derivation Rules in both cases.

Exercise 5.2 Library

Consider the following general description of a domain concerning a library:

"The Book Worm" is a library with a number of branches. The library owns a large number of books that members can borrow. Each branch can own several copies of the same book.

In the library catalog, the ISBN, author(s), title and publisher are stated for each book. The ISBN is used as identifier for a book. Book copies are sequentially numbered within each branch (a book copy is the property of the branch concerned), even if there is only one copy. The date of purchase and the number of times the book was borrowed is recorded as well.

The books can be lent out to the library members. The loan date is recorded, and a member can keep a book for a period of three weeks at most. A member can borrow several book copies at the same time (at most 8). Of course a book copy can only be lent out to one member at a time.

Name and address of each member is recorded, and also the date of birth, because this is used to classify the members as junior (18-), adult or senior (60+) member. The yearly contribution depends on this membership category. Members receive a unique number within the branch they register in (a member can register in only one branch).

A branch has a unique number, and the address and telephone number is recorded as well. Members who have failed to return a book in time receive a reminder by mail (or preferably email if they have an email address). This might have to be repeated a few times.

The loan term for juniors is two weeks, for other members it is three weeks.

For senior members it is recorded whether they would like to receive the newsletter. Hobbies are recorded for junior members, which are taken into account for decisions on acquiring new books.

Task a
From this (inevitably incomplete) general description, draw up an ERD with subtypes for this domain. Record all modeling decisions you need to take (and your reasons to take them as you did) because of incomplete or inconsistent information in the description. Provide a list of constraints that apply to the domain but cannot be seen in the diagram.

Task b
Derive a relational schema (PDM) from your answer to task a, and generate a DDL script for a certain RDBMS. Motivate your choices about the handling of the subtypes. Provide a list of integrity rules that cannot be implemented automatically by the RDBMS and have to be programmed by hand.

2 Answers to Exercises

2.1 Answers to exercises from theme 1: Normal forms and Normalization

Answer to exercise 1.1 Movies

Table MOVIE is used to store data about movies and actors who perform in these films. Only a small part of the population is shown.

MOVIE

Title ←	Year	Director	Actor →
Grease	1978	Randal Kleiser	Olivia Newton-John
Grease	1978	Randal Kleiser	John Travolta
Grease	2016	Bruce Johnson	Justin Bieber
Metro	1997	Thomas Carter	Eddie Murphy
Metro	1997	Thomas Carter	Kim Kiyori
Metro	1997	Thomas Carter	Michael Rapaport
…	…	…	…

Figure A1.1.1 Table MOVIE

a Give a verbalization of all the complete elementary facts that appear more than once in the population shown in figure A1.1.1.

The following facts are stored more than once:
Randal Kleiser is the director of the movie Grease, released in 1978.
Thomas Carter is the director of the movie Metro, released in 1997.

b Why is this table not in second normal form (2NF)?

Definition 4 in the Reader DM-RDS states:
A table is in 2NF
IF
it is in 1NF
AND
all non-key attributes are functionally dependent on the whole primary key.

The table is indeed in 1NF (every cell has a unique value): names of actors and directors are considered as a whole.

There is only one non-key attribute(column): Director. But this column is functionally dependent on only a *part* of the primary key: (Title, Year) → Director. So this table fails to meet the requirements for 2NF in this respect.

c Give one or more tables in 2NF to store the same information in. For each table, give the primary key, and foreign key references (if any). Add the complete population from figure A1.1.1.

The 'wrong' functional dependency can be taken out of table MOVIE, and a separate table MOVIE DIRECTOR can be created for it. The remaining parts of the old table MOVIE form a new table MOVIE ACTOR. The primary keys are indicated. There is a foreign key reference MOVIE ACTOR(Title, Year) → MOVIE DIRECTOR(Title, Year). Note that there is also a reference that goes in the opposite direction: MOVIE DIRCTOR(Title, Year) → MOVIE ACTOR (Title, Year): table MOVIE ACTOR is a mandatory child of table MOVIE DIRECTOR. This is because at least one actor is recorded for each film.

MOVIE DIRECTOR

Title	Year	Director
Grease	1978	Randal Kleiser
Grease	2016	Bruce Johnson
Metro	1997	Thomas Carter
…	…	…

Figure A1.1.2 Table MOVIE DIRECTOR

MOVIE ACTOR

Title	Year	Actor
Grease	1978	Olivia Newton-John
Grease	1978	John Travolta
Grease	2016	Justin Bieber
Metro	1997	Eddie Murphy
Metro	1997	Kim Kiyori
Metro	1997	Michael Rapaport
…	…	…

Figure A1.1.3 Table MOVIE ACTOR

d Account for your design: show that the new tables are indeed in 2NF now.

All cells contain atomic values, so both tables are in 1NF. There is only one non-key attribute in table MOVIE DIRECTOR: Director. The functional dependency (Title, Year) → Director shows that Director does indeed depend on the whole primary key, so table MOVIE DIRECTOR is in 2NF. In table MOVIE ACTOR, there are no non-key attributes, and so it is 2NF as well.

e Are the new tables also in 3NF or BCNF?

Definition 5 from the Reader DM-RDS states: A table is in 3NF if it is in 2NF and all non-key attributes are directly functionally dependent on the primary key. Both tables satisfy these conditions. Table MOVIE DIRECTOR is in 2NF, and column Director is the only non-key attribute, which is directly dependent on the PK, so the table is in 3NF. Table MOVIE ACTOR is in 2NF, and there are no non-key attributes, so it is in 3NF as well.

Definition 7 from the Reader DM-RDS states: A table is in BCNF if every minimal determinant is also a candidate key. Both tables satisfy this condition. In table MOVIE DIRECTOR, there is only one minimal determinant (the left-hand side of a functional dependency), namely: (Title, Year). This is also a candidate key, so the table is in BCNF. In table MOVIE ACTOR, there are no minimal determinants (no dependencies), so it is in BCNF as well.

f Verify that the redundancy is now removed: the facts from 1.1.a are stored only once now.

Answer to exercise 1.2 Theater Schedule

No answer is given here. See the **Answer to exercise 1.1** above for an answer to an analogous exercise: your answer to exercise 1.2 should follow the approach given there closely.
Consult your teacher for feedback on your work on this exercise.

Answer to exercise 1.3 Cars

a Table with redundant fact added:

CAR

License Plate	Chassis Number	Owner	Name	Fuel
PB-72-ZT	SB172ZBN10E005652	123	J de Vries	regular
35-LK-BG	TG367BZN93G000448	667	K Verstappen	diesel
AA-11-22	DFJH46JH45LJ35HU35	667	K Verstappen	LPG

Figure A1.3.1 Table CAR

Verbalization of the redundant fact:
`The name of owner 667 is K Verstappen.`

b Candidate keys:
K1: License Plate
K2: Chassis Number

Chosen primary key: K1 (more convenient than K2).

c Functional dependencies (the three from Chassis Number can be omitted):
License Plate → Chassis Number
License Plate → Owner
Owner → Name
License Plate → Fuel

d The table is in 2NF because it is in 1NF (all values are atomic) and all columns depend on the whole PK. The table is not in 3NF because column Name does not depend on the PK directly but only transitively (License Plate → Owner → Name).

e See the tables in 3NF in figure A1.3.2 below, in which the redundant fact does not occur anymore. **Note:** references (mandatory child): CAR(Owner) ←→ OWNER(Owner Number).

CAR

License Plate (PK)	Chassis Number (AK)	Owner	Fuel
PB-72-ZT	SB172ZBN10E005652	123	regular
35-LK-BG	TG367BZN93G000448	667	diesel
AA-11-22	DFJH46JH45LJ35HU35	667	LPG

OWNER

Owner Number	Name
123	J de Vries
667	K Verstappen

Figure A1.3.2 Tables CAR and OWNER

Answer to exercise 1.4 Skating Races

Tournament Winners

Tournament Type ⟷	Year	Town	Country	Winner ISU-no	Winner Name	Winner Country
WC	2006	Calgary	Canada	3005	Shani Davis	USA
ECp	2006	Hamar	Norway	2935	Enrico Fabris	Italy
WC	2007	Heerenveen	Netherlands	3064	Sven Kramer	Netherlands
EC	2007	Collalbo	Italy	3064	Sven Kramer	Netherlands

Figure A1.4.1 Table Tournament Winners

a Yes, there is redundancy. One fact in the example population is stored redundantly:
`The name of skater 3064 is Sven Kramer.`
Note: there is no redundancy for the combination of attribute values
'Sven Kramer' and 'Netherlands'
because it was stated in the explanation of the table that
"skaters can represent different countries in different tournaments",
so skater 3064 (Sven Kramer) could have represented
a different country in the EC of 2007 than in the WC of 2007.

b There is another place where redundancy is possible: see the first and last tuples (rows) below:

Tournament Winners

Tournament Type ⟷	Year	Town	Country	Winner ISU-no	Winner Name	Winner Country
WC	2006	Calgary	Canada	3005	Shani Davis	USA
ECp	2006	Hamar	Norway	2935	Enrico Fabris	Italy
WC	2007	Heerenveen	Netherlands	3064	Sven Kramer	Netherlands
EC	2007	Collalbo	Italy	3064	Sven Kramer	Netherlands
WC	1964	Calgary	Canada	1111	Jaap Eden	Netherlands

c **Figure A1.4.2 Table Tournament Winners with another redundant fact**

The fact: `Calgary is a town in Canada` is stored twice now.

c Functional dependencies:
(Tournament, Year) → Town
Town → Country
(Tournament, Year) → Winner ISU-no
Winner ISU-no → Winner Name
(Tournament, Year) → Winner Country

The table is in 2NF because it is in 1NF (the values in each cell are atomic) and all columns are dependent on the whole primary key.

d The table is not in 3NF because there are columns (Country and Winner Country) that do not depend directly on the primary key, but only transitively:

(Tournament, Year) → Town → Country
(Tournament, Year) → Winner ISU-no → Winner Name

A 3NF table structure for the same data:
Note:references:
(mandatory child): Tournament Winners(Town) ←→ TOWN(Town Name)
(ordinary FK ref): Tournament Winners(Winner ISU-no) → Skater(Skater ISU-no)

Tournament Winners

Tournament Type ←→	Year	Town	Winner ISU-no	Winner Country
WC	2006	Calgary	3005	USA
ECp	2006	Hamar	2935	Italy
WC	2007	Heerenveen	3064	Netherlands
EC	2007	Collalbo	3064	Netherlands
WC	1964	Calgary	1111	Netherlands

Town

Town Name ←→	Country
Calgary	Canada
Hamar	Norway
Heerenveen	Netherlands
Collalbo	Italy

Skater

Skater ISU-no ←→	Winner Name
3005	Shani Davis
2935	Enrico Fabris
3064	Sven Kramer
1111	Jaap Eden

Figure A1.4.3 3NF tables for Tournament Winners

e The redundant facts in figure A1.4.2, namely:

```
The name of skater 3064 is Sven Kramer.
Calgary is a town in Canada
```

occur only once in the population of figure A1.4.3 now.

Answer to exercise 1.5 Invoice

a The table is in 1NF because each cell contains an atomic value.
For higher Normal Forms, we need to know the keys and functional dependencies.
Keys: no single column can be a candidate key (in each column, duplicate values exist).
The combination Invoice No + Article is the only realistic candidate key,
and therefore also the primary key. Functional dependencies:

Invoice No → Customer, Invoice No → Seller
Invoice No → Total. **Column Total should be calculated by the system**.
Customer → Customer Name, Customer → Address, Customer → Town
Article → Price, Article → VAT
(Invoice No + Article) → Amount
Seller → Seller Name
(Article + Amount) → Subtotal. **Column Subtotal should be calculated by the system**.

The table is not in 2NF, because there are columns (namely Customer, Price, VAT and Total) that do not depend functionally on the whole primary key. So the table is in 1NF.

b In a BCNF set of tables, every minimal determinant (left-hand side of a functional dependency) is a candidate key. Using this, we can place all functional dependencies with the same minimal determinant in one table, and make the minimal determinant the primary key. Other columns are obvious:

Table INVOICE:
- Invoice No → Customer
- Invoice No → Seller
- Invoice No → Total

Table CUSTOMER:
- Customer No → Customer Name
- Customer No → Address
- Customer No → Town

Table ARTICLE:
- Article No → Price
- Article No → VAT

Table SELLER:
- Seller No → Seller Name

Table INVOICE_ITEM:
- (Invoice + Article) → Amount

Table ARTICLE_SUBTOTAL: **Derivable, not very useful, probably not to be inplemented**
- (Article + Amount) → Subtotal

References. Note the mandatory children.

The last reference is neither a FK-reference nor a mandatory child reference.
- INVOICE(Customer) ←→ CUSTOMER(Customer No)
- INVOICE(Seller) → SELLER(Seller No)
- INVOICE_ITEM(Invoice) ←→ INVOICE(Invoice No)
- INVOICE_ITEM(Article + Amount) → ARTICLE_SUBTOTAL(Article + Amount)
- ARTICLE_SUBTOTAL(Article) → ARTICLE(Article No)
- If table ARTICLE_SUBTOTAL is not implemented, the last two references disappear and the following is added: INVOICE_ITEM(Article) → ARTICLE(Article No)

The data are easily entered (not shown here). Note that all redundancy is now gone.

c In each table, the minimal determinant is also a candidate key, so each table is in BCNF

2.3 Answers to exercises from theme 3: Drawing up an ERM diagram

Answer to exercise 3.1

Answer to exercise 3.1.1 Choir member borrowings

Borrowings

PieceNo	Title	Copy	Purchased	MemberNo	Name
…	…	…	…	…	…
81	Jesu meine Freude	7	1/10/2002	16	Willemse
81	Jesu meine Freude	8	1/10/2002	17	De Jong
…	…	…	…	…	…
81	Jesu meine Freude	23	7/12/2003	25	Boomsma
81	Jesu meine Freude	24	7/12/2003	--	--
82	Requiem	1	15/10/2002	--	--
…	…	…	…	…	…
99	Requiem	1	1/10/2002	--	--
…	…	…	…	…	…

Figure A3.1.1.1 Table Borrowings

For a description of the table, see the original exercise.

Verbalizations
Two facts per fact type were verbalized:

```
The title of piece 81 is Jesu meine Freude.
 "    "     "   "    82  " Requiem.

Copy 7 of piece 81 was bought on  1/10/2002.
   " 1  "   "   82  "      "     " 15/12/2002.

Copy  7 of piece 81 is being borrowed by member 16.
   " 23  "   "   81  "   "        "      "    "   25.

The surname of member 16 is Willemse.
   "    "     "    "   25  " Boomsma.
```

Answer to exercise 3.1.2 Schedule for the Rio olympics

No answer is given here. Discuss your verbalizations with a fellow student or consult your teacher for feedback on your work on this exercise. Did you take the following points into account?

- Clear verbs, avoiding ‘has’ as much as possible.
- Clear indications of what kind of persons/things/concepts/ the facts are about (including the subdivisions in the first column in figure E3.1.2.1).
- If the meaning of something is not 100 % clear, try to find background information online.

Answer to exercise 3.2 Exam results

The ERD is given after the analysis of each fact type, to show how it can grow incrementally as each new fact type is analyzed. The diagram are derived directly from the analysis. A domain was determined for each attribute as it was added.

```
FT1:
The surname of teacher Wht is White.
 "     "     "    "     Brw   " Brown.
              ET TEACHER     Att Teacher_surname
              ID: Att Teacher_code

The surname of teacher <Teacher_code> is <Teacher_surname>.
```

Both attributes are mandatory: the <pi> automatically, and the domain expert asserts that the surname must be known for all teachers.

TEACHER			
Teacher_code	<pi>	TEACHER_CODE	<M>
Teacher_surname		SURNAME	<M>

Figure A3.2.1 ER model after analyzing FT1

```
FT2:
The surname of student S1 is Jones.
 "     "     "    "     S2   " Smith.
              ET STUDENT     Att Student_surname
              ID: Att Student_number

The surname of student <Student_code> is <Student_surname>.
```

Both attributes are mandatory: the <pi> automatically, and the domain expert asserts that the surname must be known for all students. No predicates or populations were added to the diagram yet because the meaning is clear enough.

TEACHER			
Teacher_code	<pi>	TEACHER_CODE	<M>
Teacher_surname		SURNAME	<M>

STUDENT			
Student_number	<pi>	STUDENT_NUMBER	<M>
Student_surname		SURNAME	<M>

Figure A3.2.2 ER model after analyzing FT1-FT2

```
FT3:
The SQL course realization of 20160104 was taught by teacher Wht.
 "   SQL     "         "        " 20160502  "     "    "    "    Brw.
 "   FO-ERM  "         "        " 20160307  "     "    "    "    Wht.
ET COURSE_REALIZATION                                  TEACHER
ID: Att Course_name + Att Starting_date                MATCH

RT Staffing between Course_REALIZATION and TEACHER

The <Course_name> course realization of <Starting_date> was taught by teacher
<Teacher_code>.
```

Three components, but the two segments are easily recognized: Course name and starting date together identify a course realization, and TEACHER already exists. Both Atts are mandatory because

they are the <pi>. The domain expert further agrees that every course realization must have exactly one teacher, and a teacher can teach at least none and at most several course realizations. This determines the cardinalities in RT Staffing (see figure A3.2.3a).

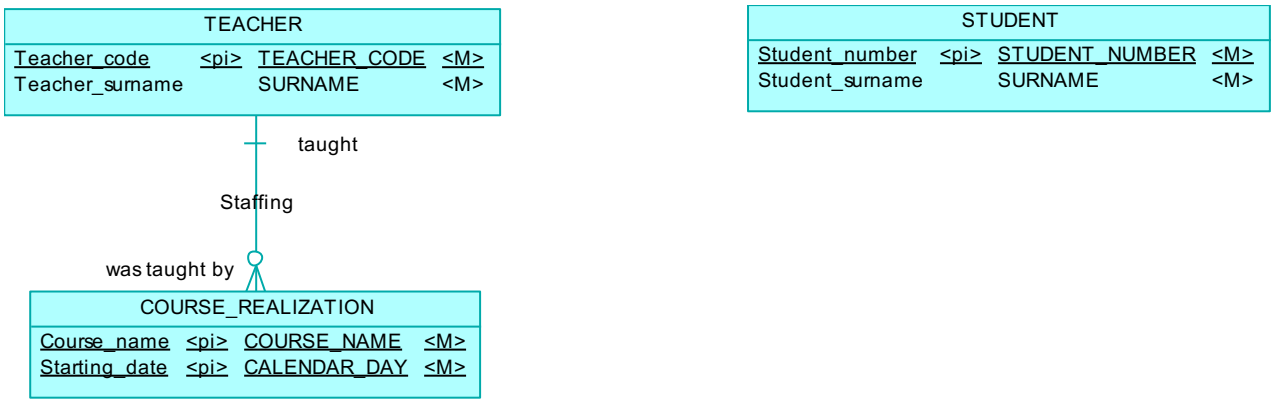


Figure A3.2.3a **ER model after analyzing FT1-FT3**

No predicates or populations were added to the diagram yet because the meaning is clear enough.

An alternative modeling is possible, in which an extra ET COURSE is created. See the analysis below and figure A3.2.3b for this option.

```
FT3 (alternative analysis with extra ET COURSE):
The SQL course realization of 20160104 was taught by teacher Wht.
 "   SQL     "          "        " 20160502   "      "     "     "     Brw.
 "   FO-ERM  "          "        " 20160307   "      "     "     "     Wht.
ET COURSE_REALIZATION                                    TEACHER
ID: ET COURSE + Att Starting_date                        MATCH
    ID: Att Course_name
RT COURSE_in_COURSE_REALIZATION between COURSE REALIZATION(dependent) and COURSE

RT Staffing between Course_REALIZATION and TEACHER

The <Course_name> course realization of <Starting_date> was taught by teacher
<Teacher_code>.
```

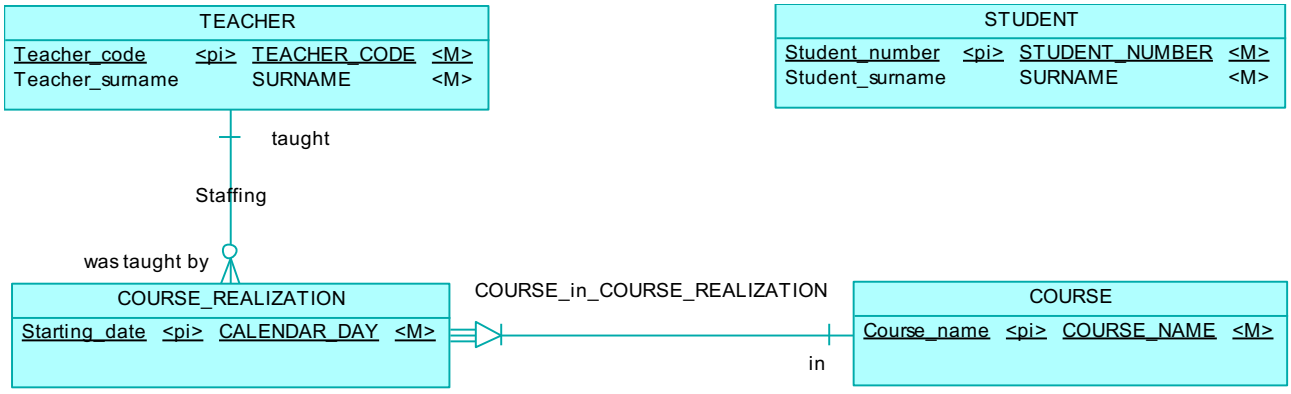


Figure A3.2.3b **Alternative ER model after analyzing FT1-FT3**

Note that COURSE_REALIZATION now has an attribute less (Course_name is now the <pi> of COURSE, and therefore cannot appear in COURSE_REALIZATION as well). Also note that the more complex diagram in figure A3.2.3b models exactly the same three fact types as the simpler one in figure A3.2.3a.

Below, the first alternative is chosen (see figure A3.2.3a), because it is simpler and no other facts about courses are to be modeled in this exercise. In a more realistic example there would probably be more facts, and an ET COURSE might well have other attributes as well (number of EC points for example), but in this exercise the simpler version suffices.

```
FT4:
Student S1 has enrolled for the SQL course realization of 20160104.
   "    S2  "      "     "   "  SQL     "         "       " 20160104.
   "    S2  "      "     "   "  FO-ERM  "         "       " 20160307.
ET STUDENT                  ET COURSE_REALIZATION
MATCH                       MATCH

RT Enrollment between STUDENT and COURSE_REALIZATION

Student <Student_number> has enrolled for the <Course_name> course realization
of <Starting_date>.
```

The domain expert explains that a student can enroll in at least ZERO course_realizations (a new student who hasn't enrolled in anything yet) and in at most MANY. Each course realization can have at least ZERO students enrolled (if the course realization is already scheduled but the enrollment is not yet open), and at most MANY. This settles the cardinalities of RT Enrollment.

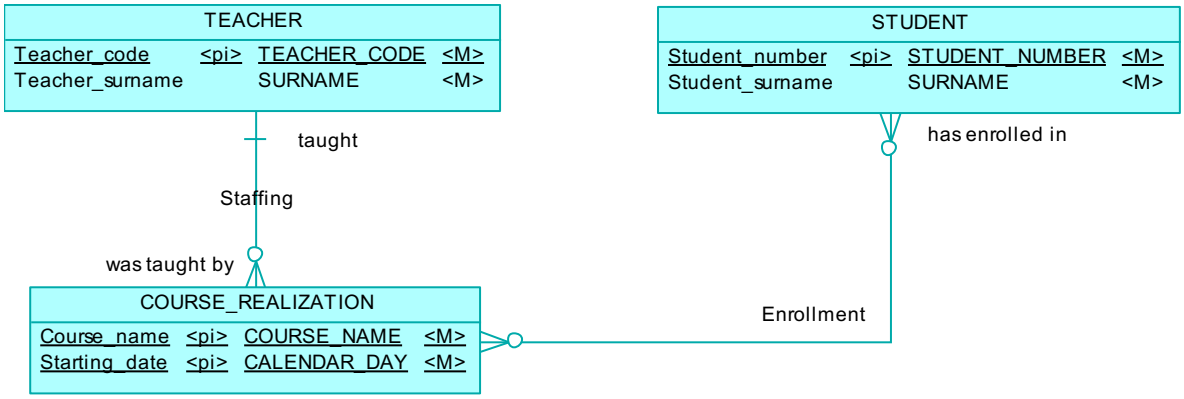


Figure A3.2.4 ER model after analyzing FT1-FT4

No predicates or populations were added to the diagram yet because the meaning is clear enough.

```
FT5:
Student S2 assisted the SQL course realization of 20160104.
   "    S7     "      "   SQL     "         "       " 20160502.
   "    S9     "      "   FO-ERM  "         "       " 20160307.
ET STUDENT              ET COURSE_REALIZATION
MATCH                   MATCH

RT Assistance between STUDENT and COURSE_REALIZATION

Student <Student_number> assisted the <Course_name> course realization of
<Starting_date>.
```

The domain expert explains that a student can assist in at least ZERO course_realizations (actually few students assist, it is considered a privilege to be chosen as assistant) and in at most ONE. Each course realization can have at least ZERO student assistants and at most MANY. This settles the cardinalities of RT Assistance.

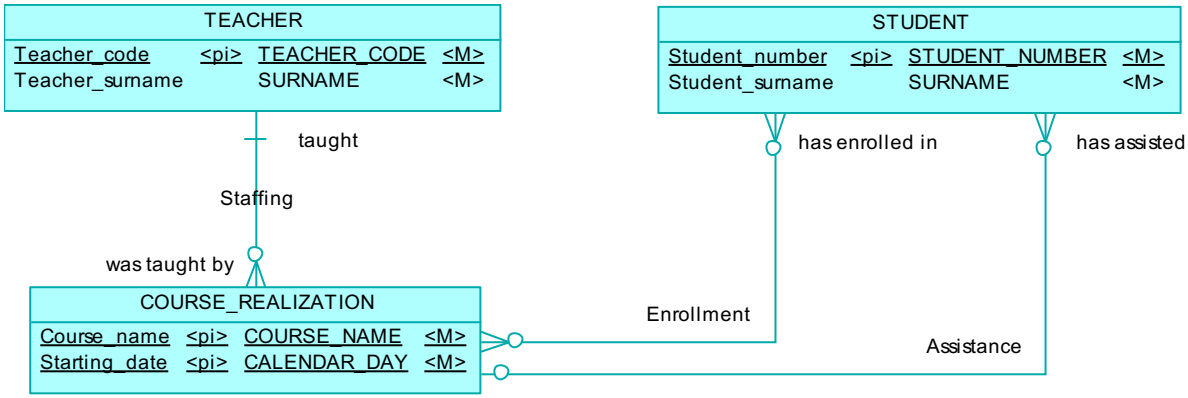


Figure A3.2.5 ER model after analyzing FT1-FT5

No predicates or populations were added to the diagram yet because the meaning is clear enough.

FT6:

Hint: underline only one segment and treat is as is shown in section 4.4.5 of the reader DM-RDS.

The identification of an exam consists of the combination of the exam date and the course realization, which make up all the components in this fact type. So only one segment is underlined.

```
There is an exam for the SQL course realization of 20160104 on 20160307.
  "    "  "   "   "   "   SQL    "         "       " 20160104  " 20160412.
  "    "  "   "   "   "   FO-ERM "         "       " 20160307  " 20160509.
          ET EXAM
          ID: ET COURSE_REALIZATION + Att Exam_date
              MATCH

RT EXAM_of_COURSE_REALIZATION between EXAM(dependent) and COURSE_REALIZATION

There is an exam for the <Course_name> course realization of <Starting_date> on
<Exam_date>.
```

The domain expert explains that every course realization must have at least two exams, and can have more. So the cardinalities of the dependent RT at the side of EXAM are at least ONE (actually at least TWO, but this cannot be shown graphically and must be given as a separate constraint) and at most MANY (the other cardinalities are automatically both ONE because of the dependency).

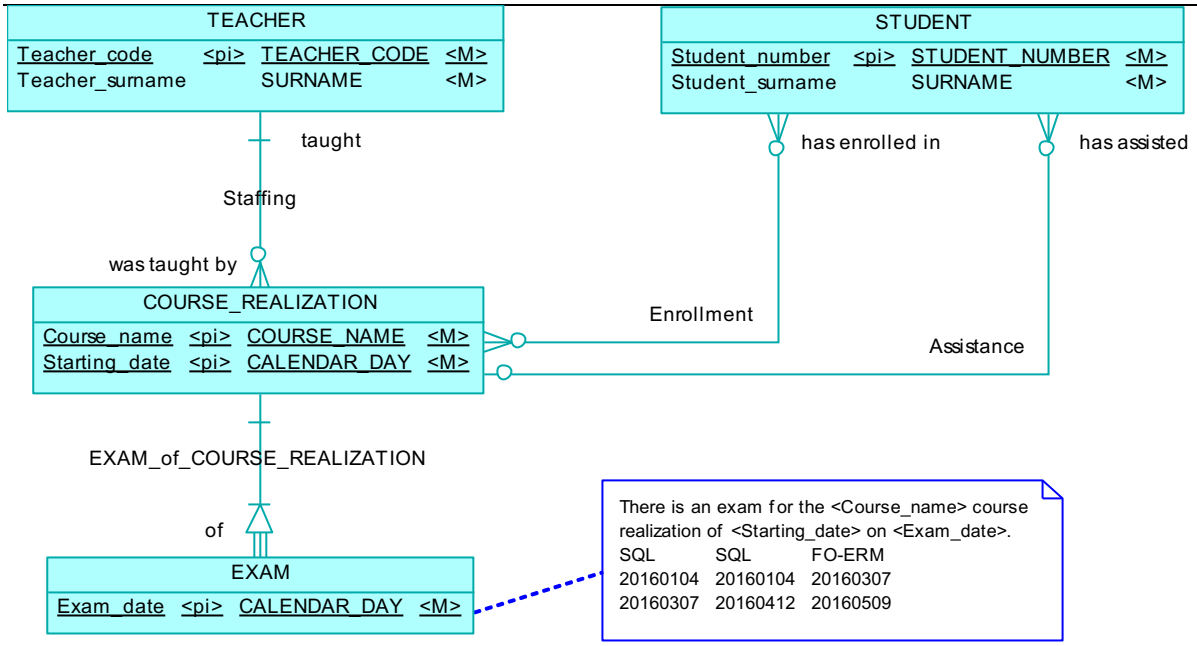


Figure A3.2.6 ER model after analyzing FT1-FT6

The predicate and an example population for FT6 was added to the diagram because the meaning of the fact type is not obvious.

```
FT7:
In the exam on 20160307 of the SQL course realization of 20160104, student S1 scored 45 %.
 "  "    "   " 20160412  "  "  SQL    "        "       " 20160104,         S1    "   72 ".
 "  "    "   " 20160509  "  "  FO-ERM "        "       " 20160307,         S2    "   81 ".
```

Which two segments are to be chosen now? Below, first two wrong segments are shown, and after that the correct segments.

Wrong segments:

```
FT7:
In the exam on 20160307 of the SQL course realization of 20160104, student S1 scored 45 %.
 "  "    "   " 20160412  "  "  SQL    "        "       " 20160104,         S1    "   72 ".
 "  "    "   " 20160509  "  "  FO-ERM "        "       " 20160307,         S2    "   81 ".
```

ET EXAM	ET STUDENT_MARK
MATCH	ID: ??

ET EXAM can be easily recognized in FT7, but the other segment then presents problems. The other two components must now be combined into an ET (there is no other possibility), but what exactly is identified by the combination of a student code and a mark? A student can have the same mark in different exams, so 'student S1 scored 45 %' does not mean anything useful by itself and so an ET STUDENT_MARK cannot be identified by this combination. The only possible solution is the following division into two segments:

Correct segments:

```
FT7:
In the exam on 20160307 of the SQL course realization of 20160104, student S1 scored 45 %.
 "  "    "   " 20160412  "  "  SQL    "        "       " 20160104,         S1    "   72 ".
 "  "    "   " 20160509  "  "  FO-ERM "        "       " 20160307,         S2    "   81 ".
```

```
ET EXAM_RESULT                                                    Att Mark
ID: ET EXAM + ET STUDENT
    MATCH     MATCH

RT EXAM_in_EXAM_RESULT between EXAM_RESULT(dependent) and EXAM
RT STUDENT_in_EXAM_RESULT between EXAM_RESULT(dependent) and STUDENT

In the exam on <Exam_date> of the <Course_name> course realization of <Starting_date>,
student <Student_code> scored <Mark> %.
```

The domain expert agrees that a student can have at least ZERO and at most MANY exam results, and that an exam can yield at least ZERO (if the exam is scheduled but not held yet) and at most MANY exam results. This settles the cardinalities of both dependent RTs.

The final model is given in figure A3.1.7, with the predicate and example population for FT7 added as well.

There is a constraint not shown in the diagram:
C1 The minimum cardinality of RT EXAM_of_COURSE_REALIZATION at the side of EXAM is TWO.
Perhaps there are other constraints as well. Can you think of one?

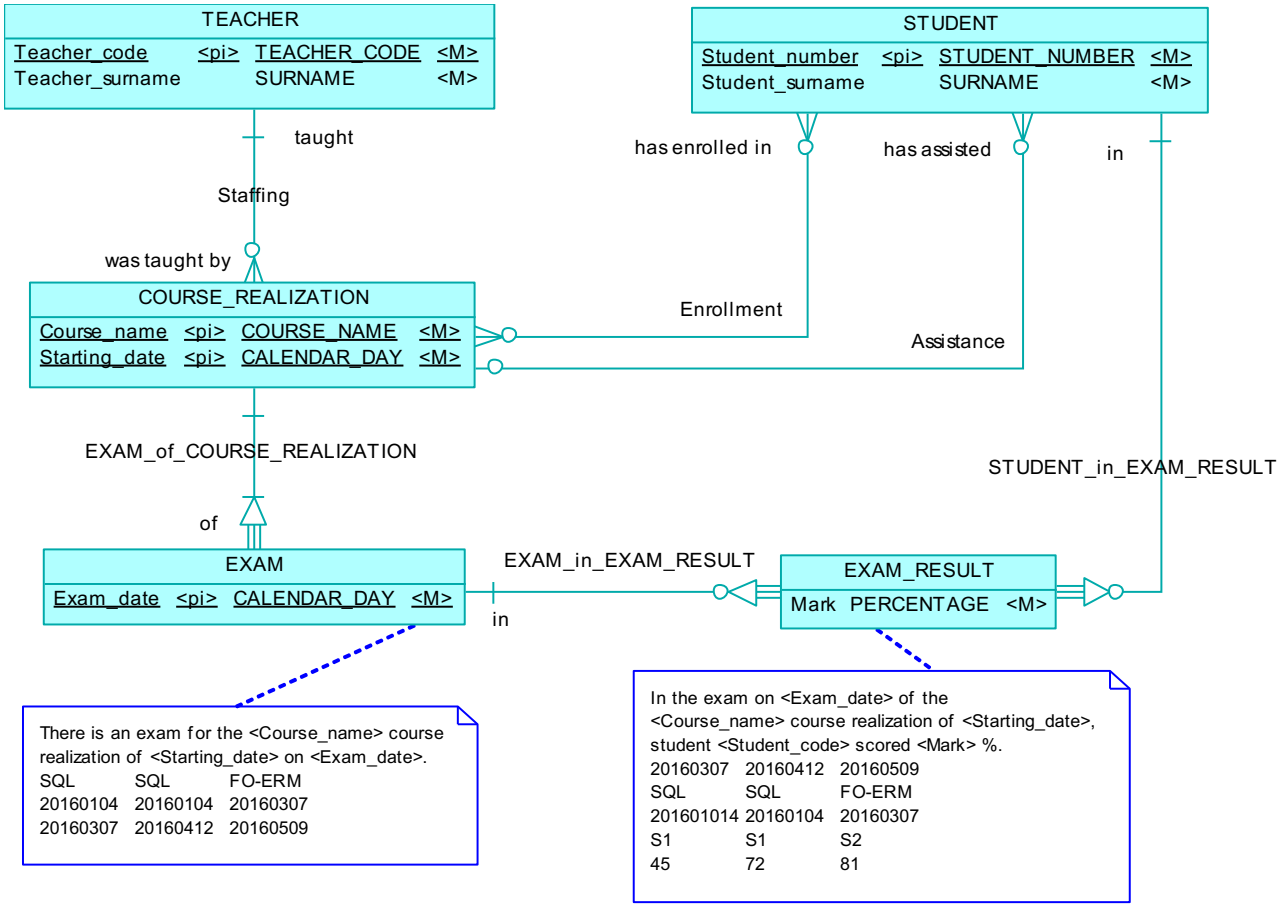


Figure A3.2.7 Final ER model

Answer to exercise 3.3 Solar eclipses

Answer to task a: Verbalizations

```
The global type of the solar eclipse of 20/3/2015 is total.
 "    "      "   " "    "      "     "  1/9/2016  " annular.

The solar eclipse of 20/3/2015 is visible  in Europe.
 "    "      "     "  1/9/2016   "    "     " Australia.
```

Note: The domain expert here preferred to model only the region, not the extra things mentioned in the examples like 'Much of' or 'East in'. Of course, these extra things could be modeled as well, in extra fact expressions like "An additional indication for the visibility of the solar eclipse of 1/9/2016 in Australia is: 'East of'.", but this option was not chosen here.

```
Hannover is located in the state Lower Saxony.
(A second verbalization is not present in the examples, but can be made up.)

Hannover is located in the country Germany.
Arnhem    "    "     " "     "    Netherlands.

The solar eclipse of 20/3/2015 in Hannover is partial.
 "    "      "     " 20/3/2015  " Arnhem    " partial.

The solar eclipse of 20/3/2015 in Hannover begins at 20/3/2015, 09:35h.
 "    "      "     " 20/3/2015 in Arnhem      "    " 20/3/2015, 09:31h.

The solar eclipse of 20/3/2015 in Hannover is maximal at 20/3/2015, 10:43h.
 "    "      "     " 20/3/2015 in Arnhem    "    "     " 20/3/2015, 10:38h.

The solar eclipse of 20/3/2015 in Hannover ends at 20/3/2015, 11:54h.
 "    "      "     " 20/3/2015 in Arnhem     "   " 20/3/2015, 11:49h.
```

Note The duration was not verbalized separately, because it can be easily calculated from the beginning and end times. However, derivable fact types can be modeled as well, but must then be handled completely and exclusively by the information system, which will use an algorithm to (re)calculate the duration on every update of the beginning and end times, and never allow an end user to modify the calculated facts. Verbalization: "The duration of the solar eclipse of 20/3/2015 in Arnhem is 2 hours and 18 minutes.". This option was not chosen by the domain expert, however.

Answers to task b and c: Analysis and ER-model

Note: the mandatory constraints of Atts and the cardinalities of RTs were determined as well, and are briefly indicated in the analyses below: '(M)' for 'Mandatory', '(O)' for 'Optional, and '(0,1)', '(0,n)', '(1,1)' and '(1,n)' for the cardinalities, written at the side of the appropriate ET (as shown in a PowerDesigner CDM). Though this is not required in the analysis (these constraints can also be determined later), it is a convenient way to express them here, which also helps to make sure no constraint is overlooked. Domains are shown in the ER model and not reported separately.

The ERD is given after the analysis of each fact type, to show how it can grow incrementally as each new fact type is analyzed. The diagram are derived directly from the analysis. A domain was determined for each attribute as it was added.

Fact types with two components:

```
FT1:
The global type of the solar eclipse of 20/3/2015 is total.
 "    "      "   "  "    "      "      "  1/9/2016   " partial.
                  ET SOLAR_ECLIPSE                    ET ECLIPSE_TYPE
                  ID: Att Eclipse_date(M)           ID: Att Eclipse_type_name(M)
RT Eclipse_is_of_type between SOLAR_ECLIPSE(0,n) and ECLIPSE_TYPE(1,1)
The global type of the solar eclipse of <Eclipse_date> is <Eclipse_type_name>.
```

NOTE: A model with an Att Eclipse_type instead of ET ECLIPSE_TYPE is also possible.

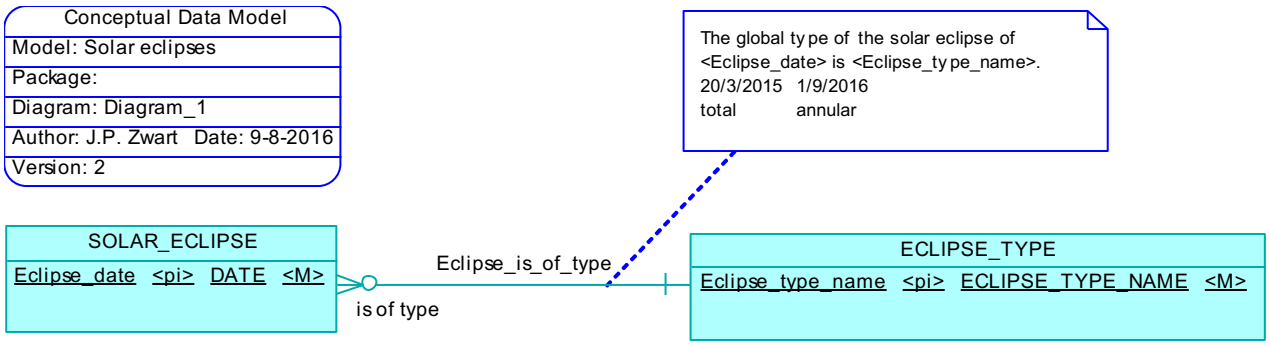


Figure A3.3.1 **ER model after analyzing FT1**

```
FT2:
The solar eclipse of 20/3/2015 is visible  in Europe.
 "    "      "     "  1/9/2016   "    "      " Australia.
ET SOLAR_ECLIPSE                               ET REGION
MATCH                                          ID: Att Region_name(M)
RT Eclipse_visible_in_region between SOLAR_ECLIPSE(0,n) and REGION(1,n)
The solar eclipse of <Eclipse_date> is visible in <Region_name>.
```

NOTE: A model with an Att Region instead of ET REGION is also possible.

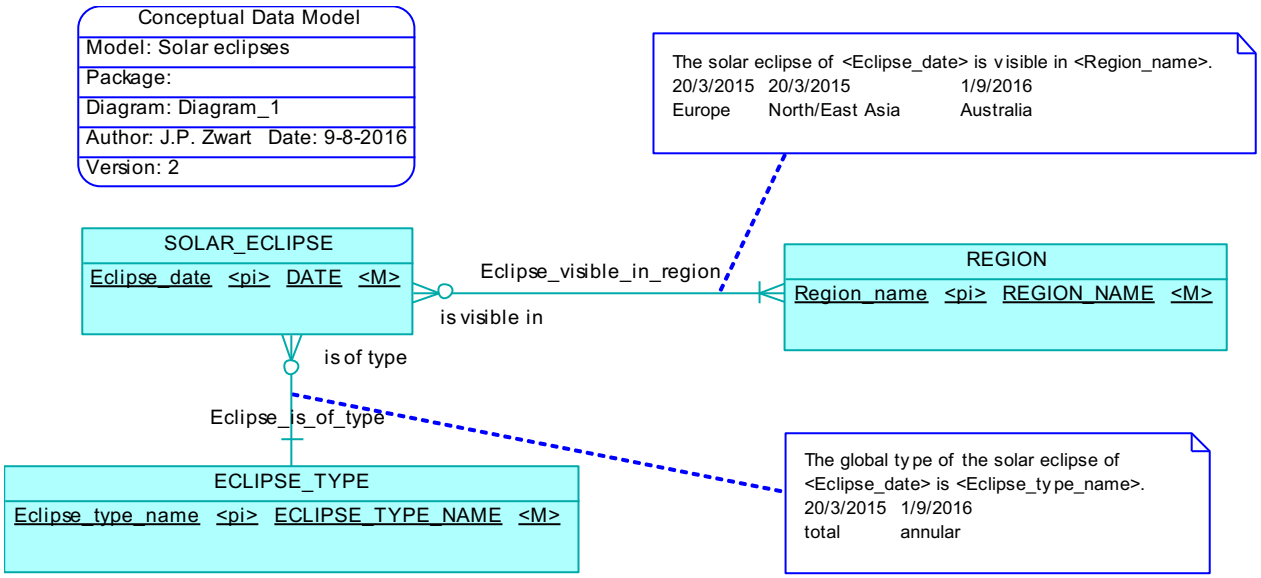


Figure A3.3.2 **ER model after analyzing FT1-FT2**

Because FT3-FT4 are very similar, they are treated in one go, with the ER-model given after the analysis of FT4. No predicates or population are given, these are clear enough from the diagram.

```
FT3:
Hannover is located in the state Lower Saxony.
ET TOWN                 Att State(O)
ID: Att Town_name(M)
<Town_name> is located in the state <State>.
```

NOTE: A model with an ET STATE instead of Att State is also possible.

```
FT4:
Hannover is located in the country Germany.
Arnhem    "     "     "  "      "    Netherlands.
ET TOWN                 Att Country(M)
MATCH
<Town_name> is located in the country <Country>.
```

NOTE: A model with an ET COUNTRY instead of Att Country is also possible.

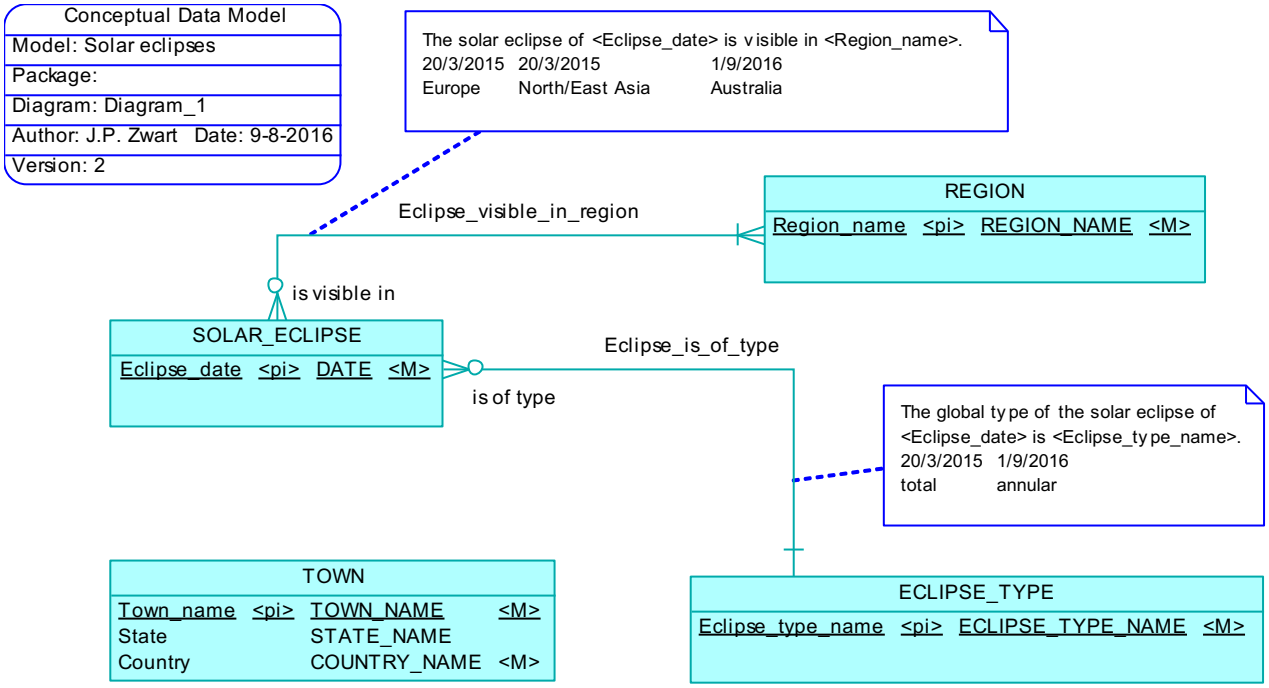


Figure A3.3.3 **ER model after analyzing FT1-FT4**

Fact types with three components:

```
Ft5:
The solar eclipse of 20/3/2015 in Hannover is partial.
 "    "      "     " 20/3/2015  " Arnhem    " partial.
ET ECLIPSE_IN_TOWN                          ET ECLIPSE_TYPE
ID: ET SOLAR_ECLIPSE + ET TOWN              MATCH
    MATCH              MATCH
RT TOWN_in_ECLIPSE_IN_TOWN
   between ECLIPSE_IN_TOWN(dependent, 0,n) and TOWN(1,1)
RT SOLAR_ECLIPSE_in_ECLIPSE_IN_TOWN
   between ECLIPSE_IN_TOWN(dependent, 1,n) and SOLAR_ECLIPSE(1,1)
RT Eclipse_type_in_town between ECLIPSE_IN_TOWN(0,n) and ECLIPSE_TYPE(1,1)
The solar eclipse of <Eclipse_date> in <Town_name> is <Eclipse_type_name>.
```

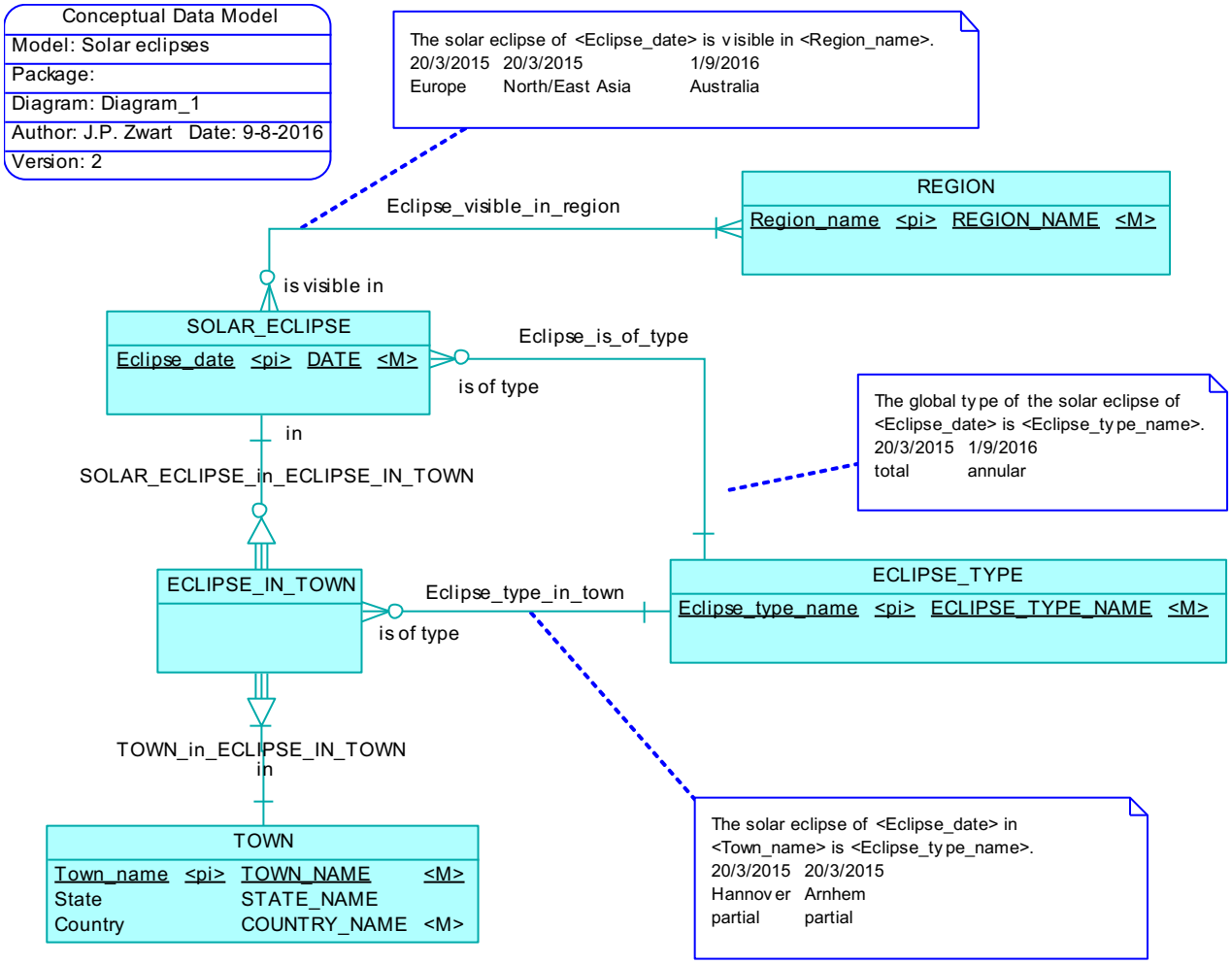


Figure A3.3.4 ER model after analyzing FT1-FT5

Because FT6-FT8 are very similar, they are treated in one go, with the final ER-model given after the analysis of FT8.

```
FT6:
The solar eclipse of 20/3/2015 in Hannover begins at 20/3/2015, 09:35h.
 "    "      "     " 20/3/2015 in Arnhem     "    " 20/3/2015, 09:31h.
ET ECLIPSE_IN_TOWN                          Att Start(M)
MATCH
The solar eclipse of <Eclipse_date> in <Town_name> begins at <Start>h.
```

```
FT7:
The solar eclipse of 20/3/2015 in Hannover is maximal at 20/3/2015, 10:43h.
 "    "      "      " 20/3/2015 in Arnhem    "    "     " 20/3/2015, 10:38h.
ET ECLIPSE_IN_TOWN                                  Att Maximum(M)
MATCH
The solar eclipse of <Eclipse_date> in <Town_name> is maximal at <Maximum>h.


FT8:
The solar eclipse of 20/3/2015 in Hannover ends at 20/3/2015, 11:54h.
 "    "      "      " 20/3/2015 in Arnhem    "    " 20/3/2015, 11:49h.
ET ECLIPSE_IN_TOWN                                  Att End(M)
MATCH
The solar eclipse of <Eclipse_date> in <Town_name> ends at <End>h.
```

The final ERD:

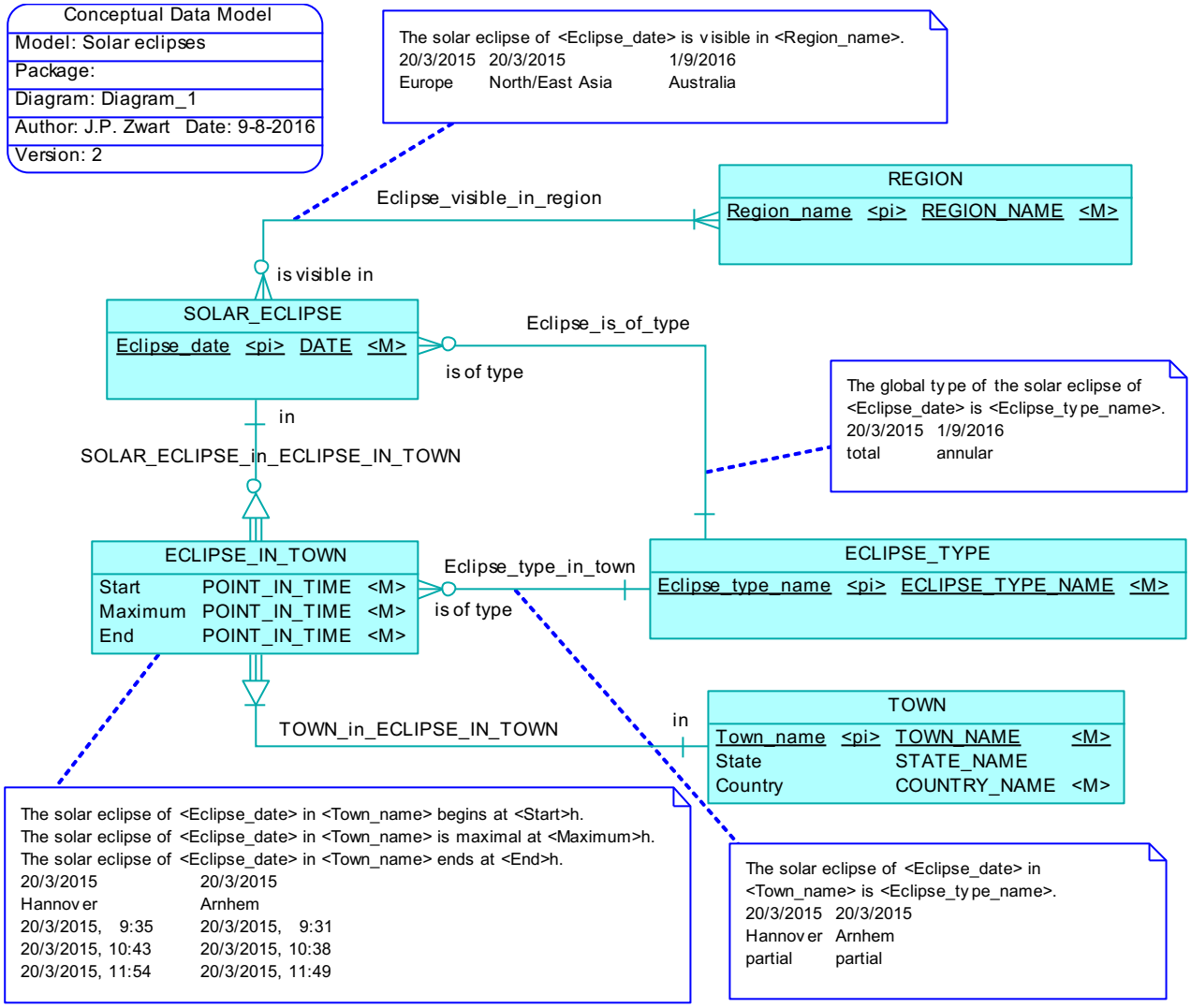


Figure A3.3.5 Final ER model

No predicate or population was given for FT3 and FT4 because the diagram is clear enough for these fact types.

Answer to exercise 3.4 Conference workshops

The answer to tasks a and b is given simultaneously below. For illustration purposes, predicates and example populations are added for all fact types. Ditto marks (") to indicate fixed text parts were not used. Constraints (<M> and cardinalities) were determined by consulting a domain expert.

Verbalizations with two blanks:

```
FT1
Session 1 starts at 9:15 AM.
Session 4 starts at 2:15 PM.
ET SESSION          Att Starts_at
ID: Att Session_number
Session <Session_number> starts at <Starts_at>.
```

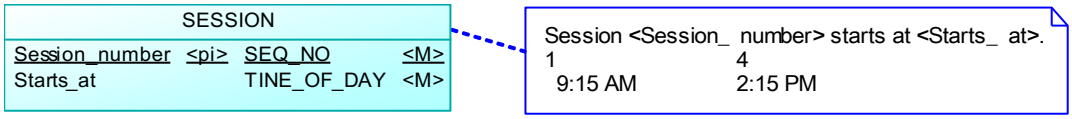


Figure A3.4.1 ERD of FT1

```
FT2
Session 1 ends by 10:45 AM.
Session 4 ends by  3:45 PM.
ET SESSION        Att Ends_by
MATCH
Session <Session_number> ends by <Ends_by>.
```

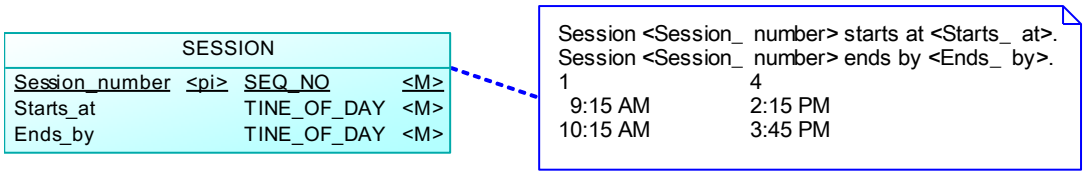


Figure A3.4.2 ERD of FT1 – FT2

```
FT3
Workshop A is titled: Model Driven Architecture.
Workshop E is titled: LINQ.
ET WORKSHOP          Att Title
ID: Att Workshop_code
Workshop <Workshop_code> is titled <Title>.
```

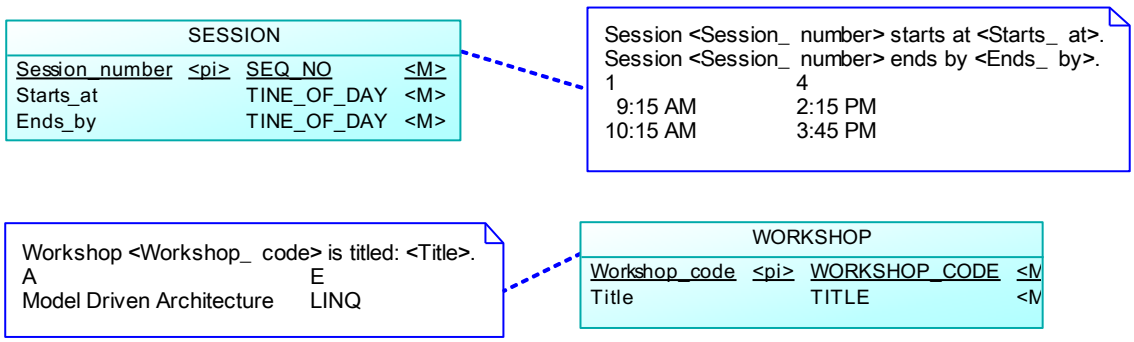


Figure A3.4.3 ERD of FT1 – FT3

```
FT4
The first name of participant 99 is Janny.
The first name of participant 15 is Geert.
                  ET PARTICIPANT    Att First_name
                  ID: Att Participant_number
The first name of participant <Participant_number> is <First_name>.


FT5
The surname of participant 99 is Mulder.
The surname of participant 15 is Straaten, van.
                ET PARTICIPANT    Att Surname(M)
                MATCH
The surname of participant <Participant_number> is <Surname>.
```

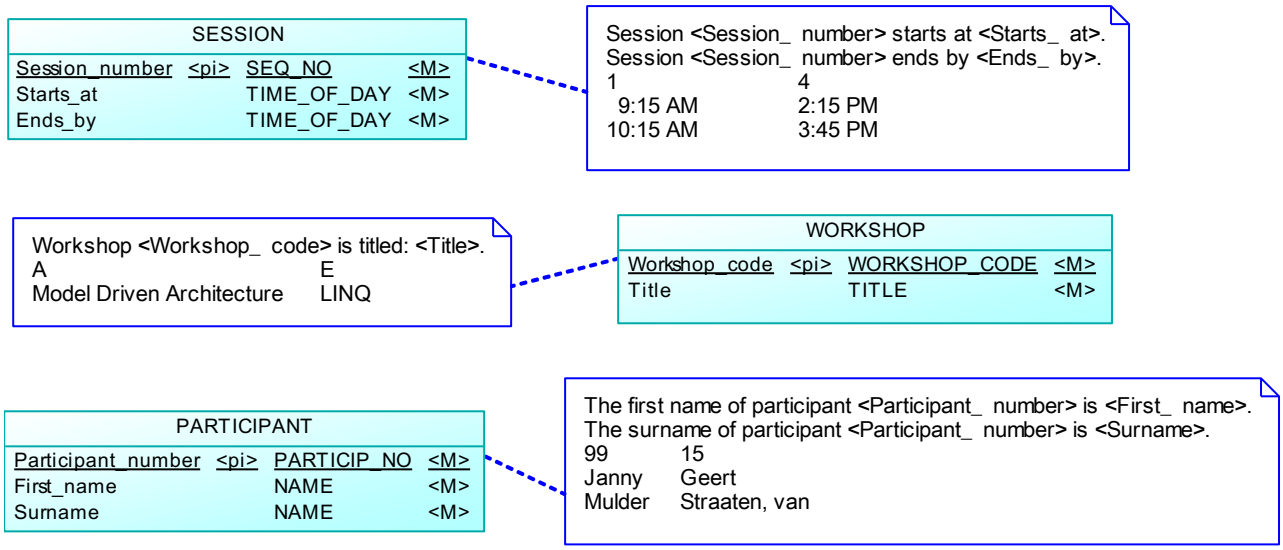


Figure A3.4.4 ERD of FT1 –FT5

Verbalizations with three blanks:

```
FT6
Workshop A in session 1 will be held in room Eagle.
Workshop E in session 2 will be held in room Toucan.
Workshop E in session 4 will be held in room Eagle.
ET SCHEDULE                                   Att Room
ID: ET WORKSHOP + ET SESSION
    MATCH          MATCH
RT WORKSHOP_in_SCHEDULE between SCHEDULE(dependent) and WORKSHOP
RT SESSION_in_SCHEDULE between SCHEDULE(dependent) and SESSION
Workshop <Workshop_code> in session <Session_number> will be held in room <Room>.
```

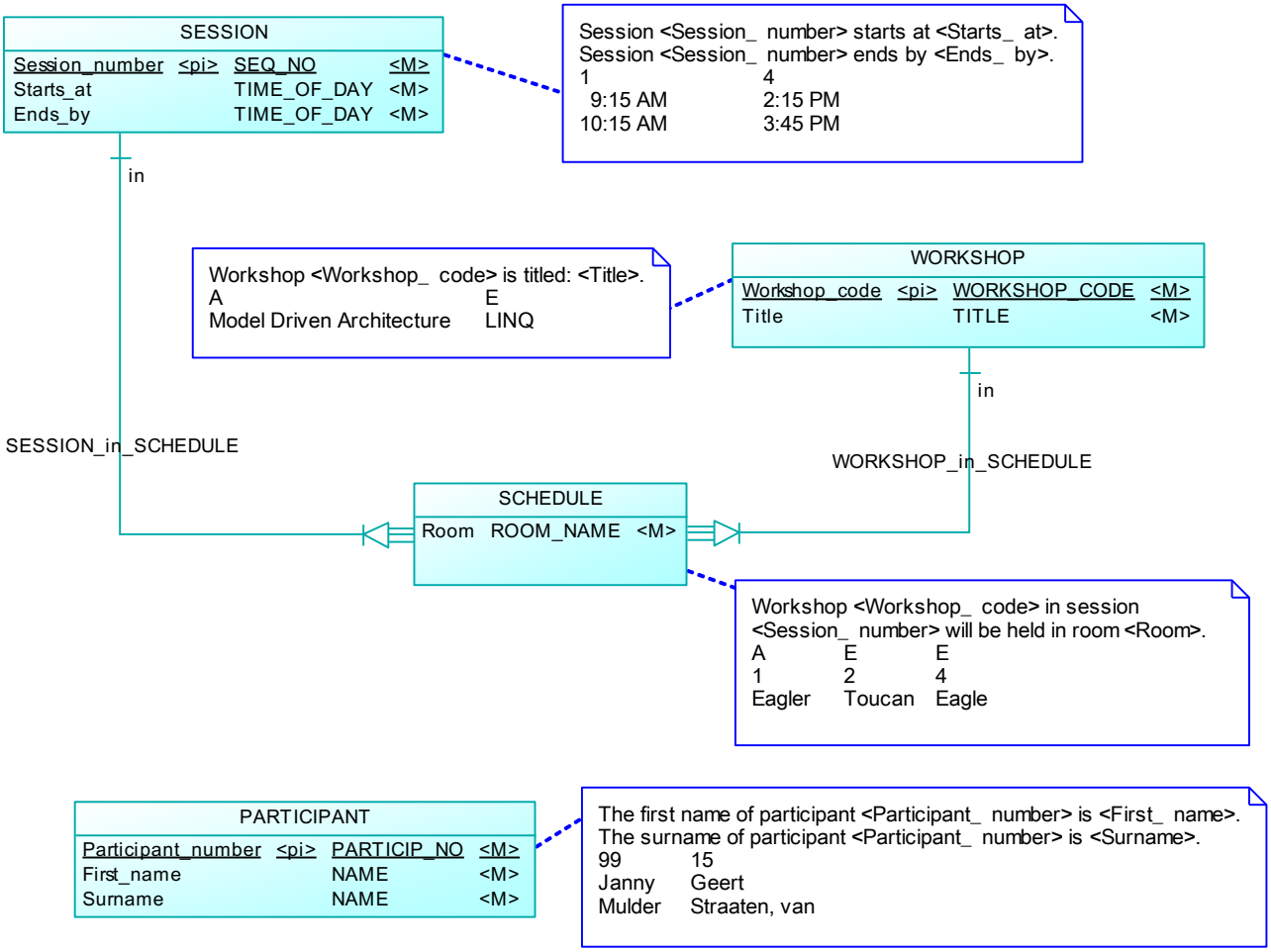


Figure A3.4.5 **ERD of FT1 – FT6**

Note: A room, here modeled as an attribute Room, could also be modeled as an entity type ROOM. This ET would only have one attribute: Room_name. See the alternative analysis and ERD below.

In a Relational schema, this ET would lead to a domain table ROOM, with one column Room_name, containing a list of all available rooms

```
FT6, alternative analysis
Workshop A in session 1 will be held in room Eagle.
Workshop E in session 2 will be held in room Toucan.
Workshop E in session 4 will be held in room Eagle.
ET SCHEDULE                                   ET ROOM
ID: ET WORKSHOP + ET SESSION                  ID: Att Room_name
    MATCH          MATCH
RT WORKSHOP_in_SCHEDULE between SCHEDULE(dependent) and WORKSHOP
RT SESSION_in_SCHEDULE between SCHEDULE(dependentn) and SESSION
RT Room_used_in_schedule between SCHEDULE and ROOM
Workshop <Workshop_code> in session <Session_number> will be held in room
<Room_name>.
```

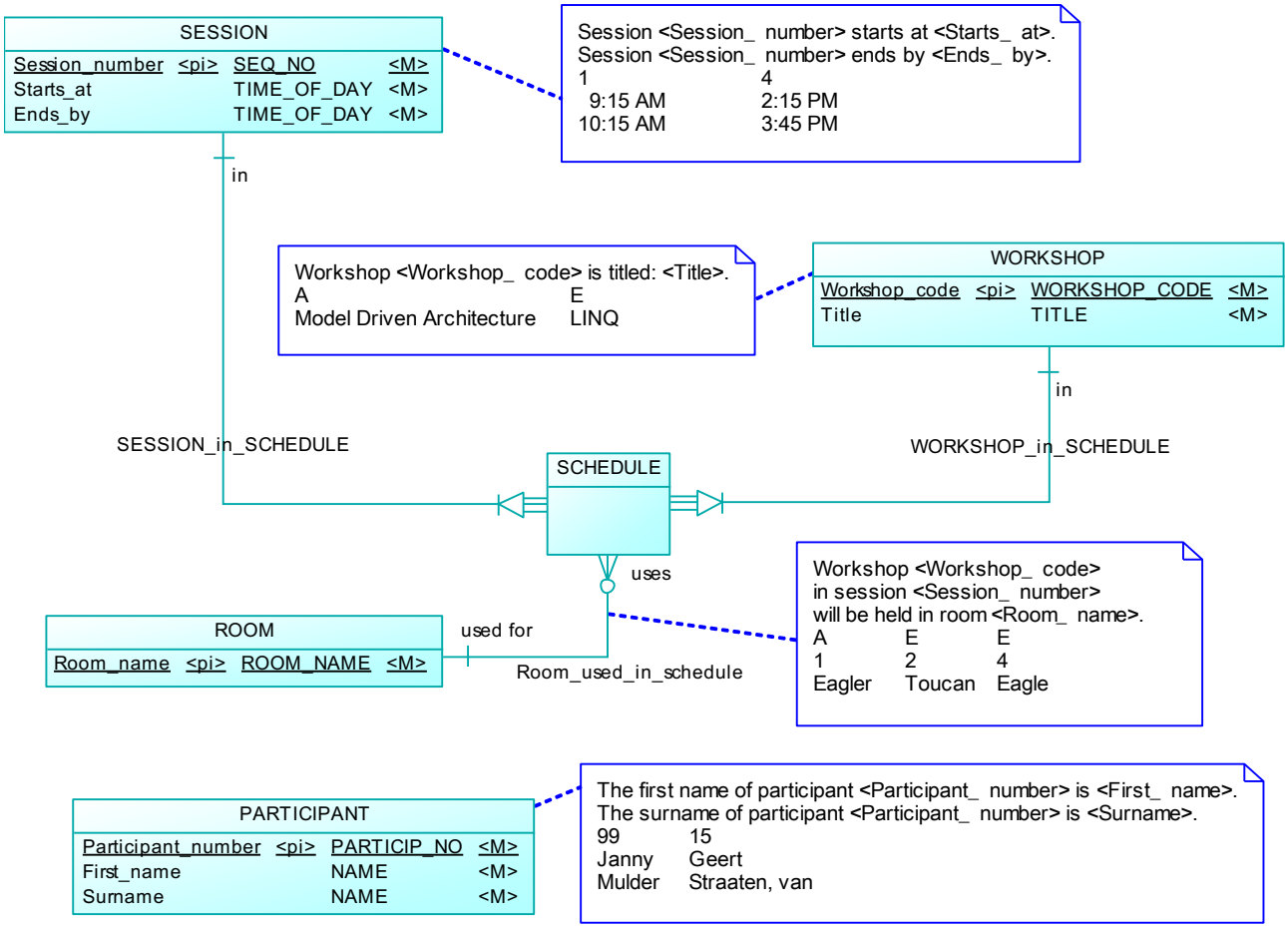


Figure A3.4.6 Alternative ERD of FT1 – FT6

However, because there are no other fact types that contain a room name, and no other attributes for a room are to be recorded, and because such a list is not a primary concern here, it was decided to use the simpler modeling with 'Room' only as an attribute of ET SCHEDULE.

```
FT7
Participant 99 will take part in workshop A in session 1.
Participant 99 will take part in workshop C in session 2.
Participant 15 will take part in workshop E in session 2.
ET PARTICIPANT                    ET SCHEDULE
MATCH                             MATCH
RT Participation between PARTICIPANT and SCHEDULE
Participant <Participant_number> will take part in workshop <Workshop_code> in
session <Session_number>.
```

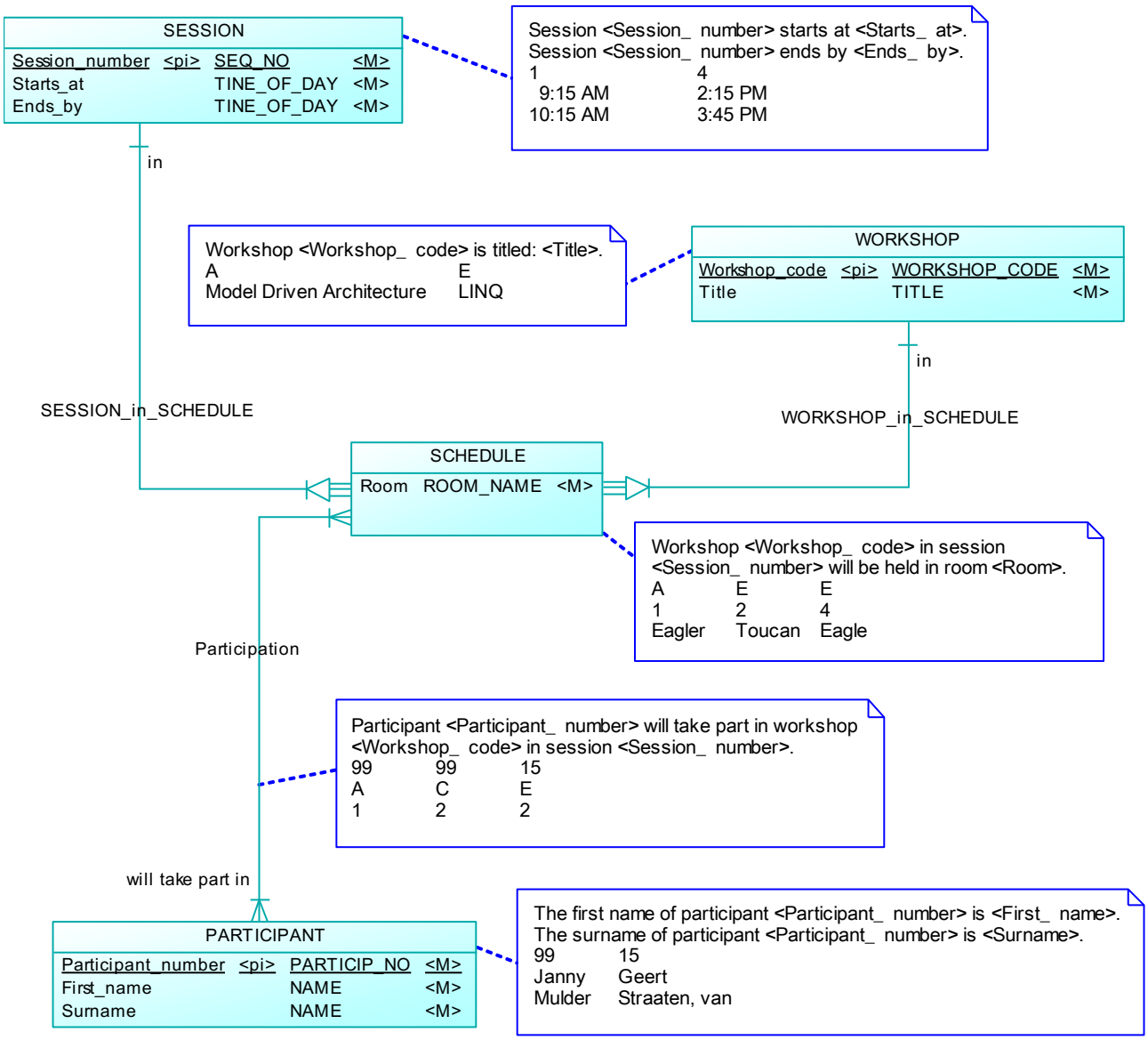


Figure A3.4.7 **Final ERD with all predicates and example populations**

Because many attributes are self-explanatory, not all predicates need to be given. Although adding the complete semantics to the diagram itself does have advantages, this does obscure the overview a bit, so a version with a minimum of predicates and example populations is probably preferred by many modelers. But make sure you always include a few! For such a version, see figure A3.4.8.

In RT Participation, the minimum cardinality 1 at the side of PARTICIPANT means that every schedule must have at least one participant. This is a bit too strict: it means that you cannot enter a schedule first, and add participants later. But if you specify a minimum cardinality of zero, the system will eventually allow a multitude of empty schedules. The problem here is, that the minimum cardinality is to be understood as 'soft': it should allow schedules without participants for a while, but issue reminders if they remain empty too long, because in the end all schedules must have a tleast one participant. More on this matter: see the distinction in the literature between **alethic constraints** (must always apply) and **deontic constraints** (may be violated, temporarily or otherwise).

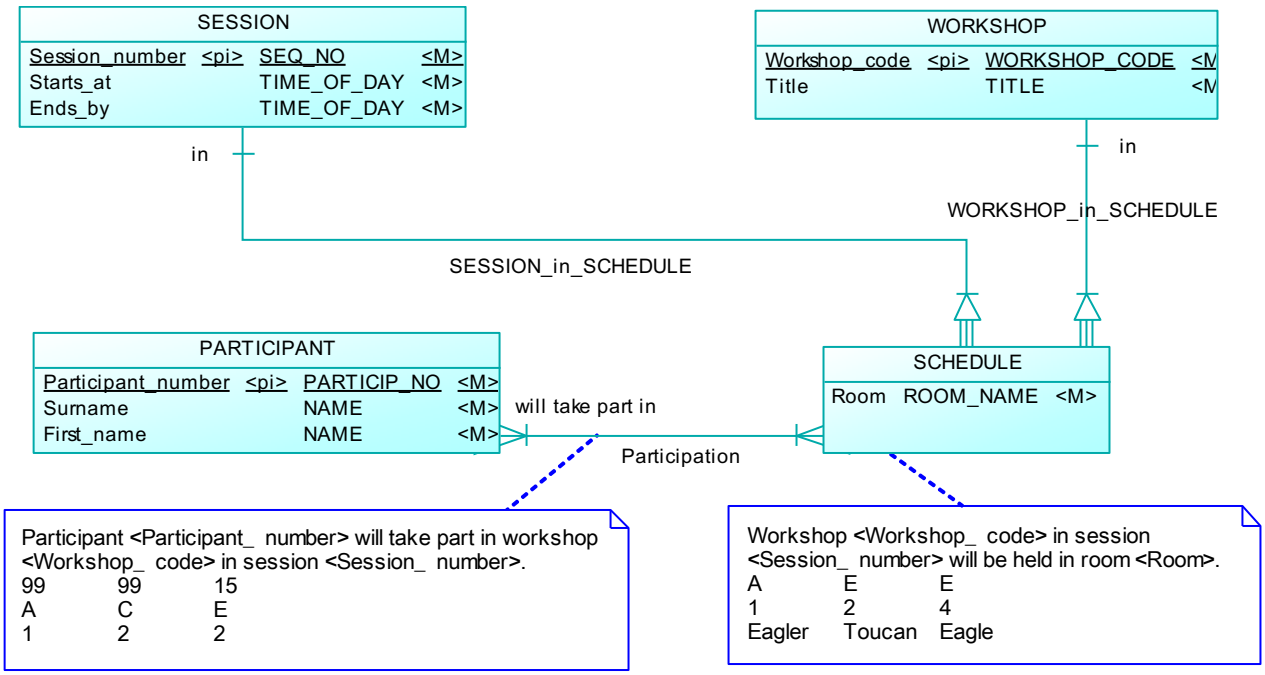


Figure A3.4.8 **Final IGD with a minimum of predicates and example populations**

Answer to exercise 3.5 Meal recipes

No answer is given here. Consult your teacher for feedback on your work on this exercise.

Answer to Exercise 3.6 Council Elections

Figure A3.6.1 shows a few examples of the election result of one polling station in the town Oss.

Polling station 24 – 't Oude Theater												
1				**2**				**3**				**4**
CDA				**SP**				**VVD**				**…**
1	G Louwers (m)		88	1	LH van Doren (v)		226	1	HG Nijenkamp (m)		29	
2	H Peters (v)		14	2	M Scheepens (v)		19	2	MHN Huiskens (m)		4	
3	M Mensink (m)		9	3	HGM Vervoort (m)		12	3	G de Bruijn (m)		1	
4	S Jansen (m)		11	4	S Veulings (m)		7	4	JF Soulier (m)		27	
5	J Reijmer (v)		5	5	AD Ghouassi (v)		9	5	HMM van Ruberg (v)		1	
..	……		...	…	……		...	…	……		...	

Figure A3.6.1 Part of election result of one polling station

Answer to task a

Verbalizations, already sorted and arranged in order of number of components, are given below. Note: the words 'In Oss' can be added everywhere, but should then be treated as fixed text, as in the following example from the first fact type, which has two components (not three):

```
     In Oss, the name of polling station 24 is: 't Oude Theater.
      "  "  ,  "    "   "   "         "     11   "  Cafee Staminee.

FT1
The name of polling station 24 is: 't Oude Theater.
 "    "   "    "        "     11   "  Cafee Staminee.

FT2
List 1 is assigned to the party CDA.
   "  2  "     "       "  "    "   SP.

FT3
Candidate 1 of list 1 is a man.
    "     2  "   "  2  " " woman.

FT4
The surname of candidate 1 of list 1 is Louwers.
 "     "     "      "     2  "   "  2  " Scheepens.

FT5
The initials of candidate 1 of list 1 are G.
 "     "     "      "     3  "   "  2  "  HGM.

FT6
The prefix of candidate 1 of list 2 is: de.
 "     "    "      "     3  "   "  3  "  van.

FT7
In polling station 24, candidate 1 of list 1 got 88 votes.
 "     "        "    24,     "     4  "   "  3  "  27   "   .
```

Answer to task b: Analysis and ERD per fact type of the above given verbalizations.

```
FT1
The name of polling station 24 is: 't Oude Theater.
 "    "   "   "      "      "    11   "  Cafe Staminee.
           ET POLLING_STATION      Att Station_Name
           ID: Att Station_number
The name of polling station <Station_number> is: <Station_name>.
```

All stations must have a station name according to the domain expert, so Att Station_name is <M>. The <pi>-attribute is <M> of course (the <M> for <pi>-attributes will not be explained again below).

POLLING_STATION			
Station_nunber	<pi>	STATION_NO	<M>
Station_name		STATION_NAME	<M>

Figure A3.6.2 **Result of analyzing FT1**

```
FT2
List 1 is assigned to the party CDA.
  "  2  "    "      "  "    "   SP.
ET LIST              Att Party
ID: Att List_number
List <List_number> is assigned to the party <Party>.
```

The second segment could also be seen as an ET PARTY with ID: Att Party_code,
and a RT Party_on_list between LIST and PARTY (all four cardinalities: ONE).
This would model a domain list containing all valid political party codes, and might be preferred by the domain experts, but here the choice for an attribute was made. If other towns than just Oss are considered as well, then perhaps the choice would be different.

All lists must have a party associated with it (otherwise the list is useless and isn't recorded) according to the domain expert, so Att Party is <M>.

LIST			
List_number	<pi>	LIST_NO	<M>
Party		PARTY_CODE	<M>

POLLING_STATION			
Station_nunber	<pi>	STATION_NO	<M>
Station_name		STATION_NAME	<M>

Figure A3.6.3 **Result of analyzing FT2**

```
FT3
Candidate 1 of list 1 is a man.
    "     2  "   "  2   " " woman.
ET CANDIDATE             Att Gender
ID: ET LIST + Att Sequence_number
    MATCH
RT LIST_Of_CANDIDATE between LIST and CANDIDATE(dependent)
Candidate <Sequence_number> of list <List_number> is a <Gender>.
```

The second segment could also be modeled as a domain list (ET GENDER, ID: Att Gender_code).

The gender is recorded as ‘m’ for men or ‘v’ for women, which is added as a constraint to domain GENDER_CODE. If an ET GENDER is modeled, these values can be entered as population to the resulting domain table by the database administrator.

The domain expert agrees that a list must have at least one candidate (indeed it does happen sometimes there is only one candidate), and usually has several. So the minimum and maximum cardinalities of RT LIST_of_CANDIDATE at the side of CANDIDATE are ONE and MANY respectively. The cardinalities at the other side are both ONE because of the dependency (these fixed cardinalities for dependent RTs will not be explained again below).

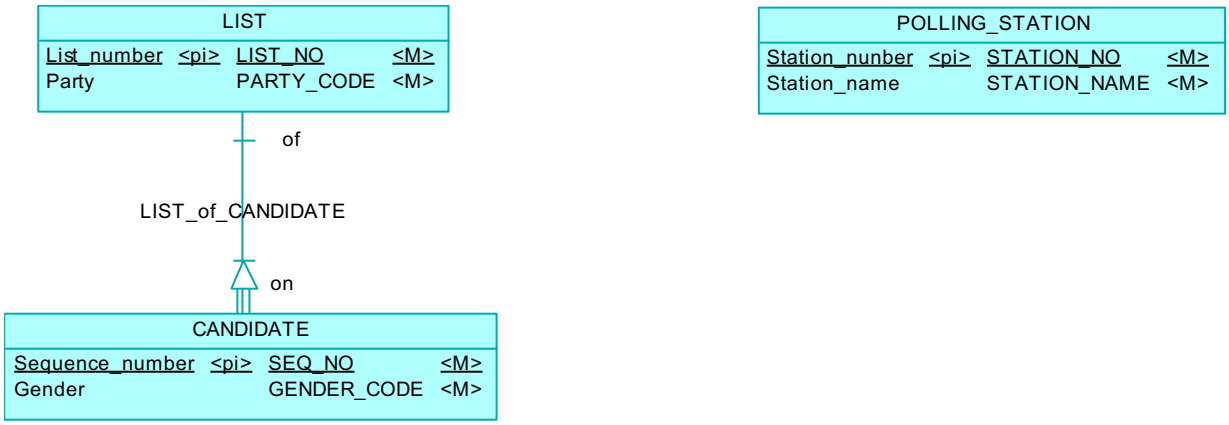


Figure A3.6.4 Result of analyzing FT3

Fact types FT4 – FT6 are easily added: they verbalize simple attributes of ET CANDIDATE. So the growing diagram is only shown after analyzing FT6. For each new attribute it was checked with the domain expert whether it is <M> or not.

```
FT4
The surname of candidate 1 of list 1 is Louwers.
 "      "       "       "      2   "   "  2   "  Scheepens.
               ET CANDIDATE                 Att Surname
               MATCH
The surname of candidate <Sequence_number> of list <List_number> is <Surname>.
```

The domain expert acknowledges that the surname must be known for all candidates: Att is <M>.

```
FT5
The initials of candidate 1 of list 1 are G.
 "      "       "       "      3   "   "  2   "  HGM.
               ET CANDIDATE                 Att Initials
               MATCH
The initials of candidate <Sequence_number> of list <List_number> are <Initials>.
```

The domain expert acknowledges that the initials must be known for all candidates: Att is <M>.

```
FT6
The prefix of candidate 1 of list 2 is: de.
 "      "       "       "      3   "   "  3   "  van.
               ET CANDIDATE                 Att Prefix
               MATCH
The prefix of candidate <Sequence_number> of list <List_number> is: <Prefix>.
```

The domain expert acknowledges that only some candidates have a prefix: Att is optional (not <M>).

Note in figure A3.6.5: the meaning of all attributes is sufficiently clear, so no predicates or example population is needed at this point yet.

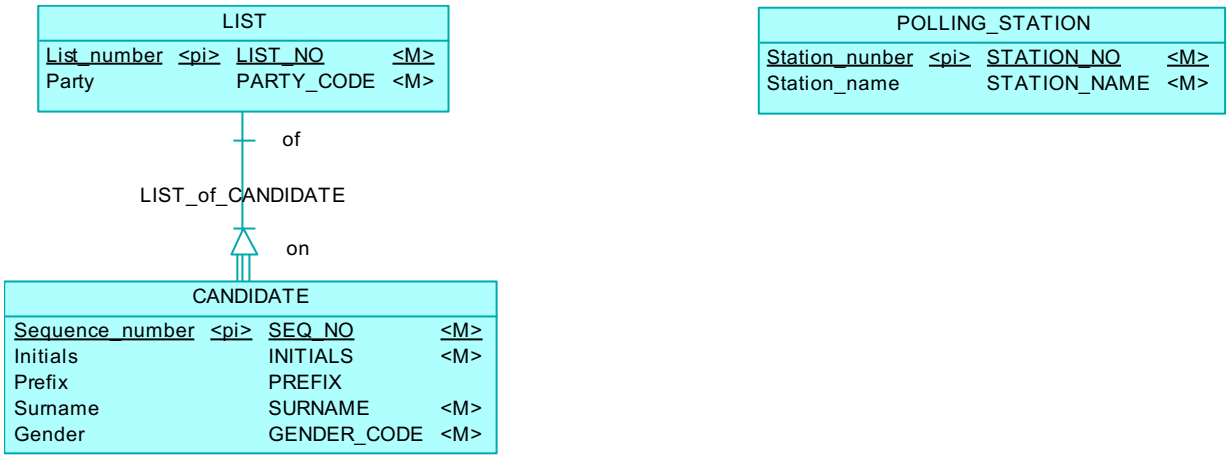


Figure A3.6.5 Result of analyzing FT4 – FT6

```
FT7
In polling station 24, candidate 1 of list 1 got 88 votes.
 "    "       "    24,     "     4  "   "  3  "  27   "  .
```

In FT7, there are four components. We recognize ET POLLING_STATION and ET CANDIDATE, and then there is the number of votes. Since the given numbers of votes are clearly those that *one candidate has won at one particular polling station*, it is natural to consider Number_of_votes as an attribute of a new (weak) ET ELECTION_RESULT (see further below for an alternative analysis approach and why that fails). The new ET has two dependent RTs.

```
FT7
In polling station 24, candidate 1 of list 1 got 88 votes.
 "    "       "    24,     "     4  "   "  3  "  27   "  .
   ET ELECTION_RESULT                              Att Number_of_votes
   ID: ET POLLING_STATION + ET CANDIDATE
       MATCH                MATCH
RT POLLING_STATION_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                          and POLLING_STATION
RT CANDIDATE_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                    and CANDIDATE

In polling station <Station_number>, candidate <Sequence_number> of list
<List_number> got <Number_of_votes> votes.
```

The domain expert assures the following: if a candidate receives no votes at a polling station, the official result is stored as: zero votes for this candidate in this polling station (this is also the default number of votes before the election closes and the votes are counted). So no number of votes can ever be absent for any candidate at any polling station. Therefore the minimum cardinalities of both new RTs are ONE (and the maximum cardinalities are MANY).

Because of the complex identifier for ET ELECTION_RESULT, the predicate and a small example population for this fact type are added to the diagram, to make it easier to read.

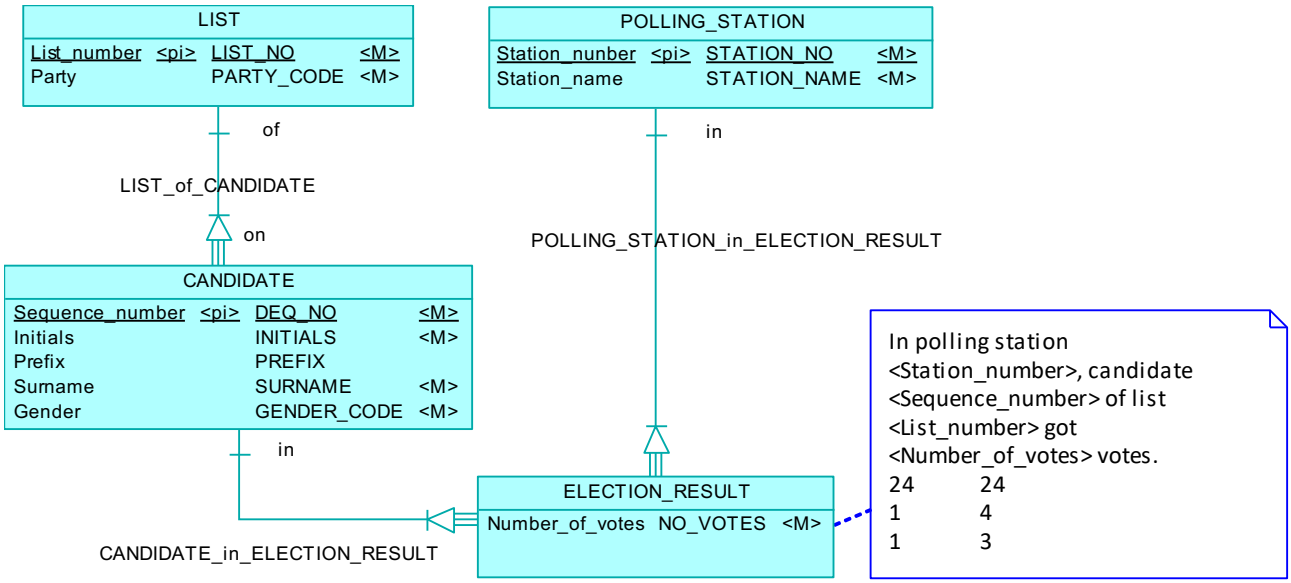


Figure A3.6.6 **Result of analyzing FT7**

Could another analysis have been made? For example, taking the polling station as one segment, and the candidate plus the number of votes as the second?

FT7 (wrong analysis)
In polling station 24, candidate 1 of list 1 got 88 votes.
" " " 24, " " 4 " " 3 " 27 " .
ET POLLING_STATION *ET ELECTION_RESULT*
MATCH *ID: ET CANDIDATE + Att Number_of_votes*
MATCH

The new ET would now be identified by ET CANDIDATE + Att Number_of_votes. But this combination clearly misses a component: the same candidate may have won the same number of votes at a different polling station too, so this combination ('88 votes for candidate 1 of list 1') does not by itself identify anything meaningful and the new ET ELECTION_RESULT would miss an essential part of its primary identifier (there would be a RT between POLLING_STATION and ELECTION_RESULT, but not a dependent one).

The same objection would arise for a combination of POLLING STATION and Number_of_votes (with a slight rearrangement of the verbalizations to clearly show this):

FT7 (wrong analysis)
Candidate 1 of list 1 got 88 votes in polling station 24.
" 4 " " 3 " 27 " " " " 24.
ET CANDIDATE *ET ELECTION_RESULT*
MATCH *ID: Att Number_of_votes + ET POLLING_STATION*
MATCH

The new ET would now be identified by Att Number_of_votes + ET POLLING_STATION. But this combination clearly misses a component: the same candidate may have won the same number of votes at a different polling station too, so this combination ('88 votes in polling station 24') does not by itself identify anything meaningful and the new ET ELECTION_RESULT would miss an essential part of its primary identifier (there would be a RT between CANDIDATE and ELECTION_RESULT, but not a dependent one).

Even if the verbalization is in the order as given directly above, a good analysis can be made, leading to the same figure A3.6.6:

```
FT7
Candidate 1 of list 1 got 88 votes in polling station 24.
    "     4  "   "  3  "  27   "    "    "       " "  24.
ET ELECTION_RESULT
                              _________
                             Att Number_of votes
ID: ET CANDIDATE + ET POLLING_STATION
    MATCH          MATCH
RT CANDIDATE_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                    and CANDIDATE
RT POLLING_STATION_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                          and POLLING_STATION

Candidate <Sequence_number> of list <List_number> got <Number_of_votes> votes in
polling station <Station_number>.
```

Answer to task c: Are there any business rules that cannot be shown graphically in the ERD?

The same candidate cannot be on a list more than once. But there is no way to identify a person in the given examples: two different persons could well have the same surname, initials, prefix and gender. More information would be needed (like the BSN (Burger Service Nummer), which is a unique number for each Dutch citizen. Other business rules (did you find more?):
BR1 The number of votes must be an integral number greater than or equal to zero.

Answer to task d: Extend the verbalizations, the analysis and the model so it will now capture the council elections in all the towns in the Netherlands. Include a domain list of all towns in the Netherlands with a council in the analysis (a separate ET for towns with councils).

We need to include the name of the town in the verbalizations now. This can be easily done by adding 'In <town name>,' at the front of each sentence (or somewhere else, like in FT7 below). Examples from other towns were made up. A verbalization for which towns have councils (for the domain list of towns) might be given as well, but this is discussed later, below figure A3.6.7.

```
FT1
In Oss, the name of polling station 24 is: 't Oude Theater.
 " Elst, "   "    "    "        "     8  "  Onder de Toren.
FT2
In Oss, list 1 is assigned to the party CDA.
 " Elst,  "  2  "    "       "  "    "   VVD.
FT3
In Oss, candidate 1 of list 1 is a man.
 " Elst,    "     3  "    "  6  " " woman.
FT4
In Oss, the surname of candidate 1 of list 1 is Louwers.
 " Elst, "     "     "      "     2  "   "  3  " Halbers.
FT5
In Oss, the initials of candidate 1 of list 1 are G.
 " Elst, "     "     "      "     1  "   "  1  "  JJ.
FT6
In Oss, the prefix of candidate 1 of list 2 is: de.
 " Elst, "     "    "      "     7  "   "  4  "  van.
FT7
In polling station 24 in Oss, candidate 1 of list 1 got 88 votes.
 "    "        "    14  " Elst,    "     7  "   "  4  "  12   "   .
```

The analysis is given below without much comment, with only the final ERD in figure A3.6.7. It includes the domain list for Dutch towns with a council. For <M> and cardinalities: only the new ones are mentioned (see above for all the others).

```
FT1
In Oss, the name of polling station 24 is: 't Oude Theater.
 " Elst, "   "    "    "       "     8   "  Onder de Toren.
ET POLLING_STATION                          Att Station_name
ID: ET TOWN + Att Station_number
    ID: Att Town_name
RT TOWN_in_POLLING_STATION between POLLING_STATION(dependent) and TOWN.
In <Town_name>, the name of polling station <Station_number> is: <Station_name>.
```

RT TOWN_in_POLLING_STATION: Only towns with councils are included in ET TOWN, so all these towns must have elections. Therefore they must all have at least one polling station, and can have several (min. card. ONE, max. card. MANY).

```
FT2
In Oss, list 1 is assigned to the party CDA.
 " Elst,  "  2   "    "      " "    "   VVD.
ET LIST                     Att Party
ID: ET TOWN + Att List_number
    MATCH
RT TOWN_in_LIST between LIST(dependent) and TOWN.
In <Town_name>, list <List_number> is assigned to the party <Party>.
```

RT TOWN_in_LIST: A town must have at least one list (because only towns with elections are recorded), and can have several (min. card. ONE, max. card. MANY).

```
FT3
In Oss, candidate 1 of list 1 is a man.
 " Elst,    "     3  "   "  6   " " woman.
ET CANDIDATE                      Att Gender
ID: Att Sequence_number + ET LIST
                          MATCH
RT LIST_of_CANDIDATE between LIST and CANDIDATE(dependent)
In <Town_name>, candidate <Sequence_number> of list <List_number> is a <gender>.
```

```
FT4
In Oss, the surname of candidate 1 of list 1 is Louwers.
 " Elst, "    "    "     "     2  "   "  3   " Halbers.
ET CANDIDATE                                  Att Surname
MATCH
In <Town_name>, the surname of candidate <Sequence_number> of list <List_number>
is <Surname>.
```

```
FT5
In Oss, the initials of candidate 1 of list 1 are G.
 " Elst, "     "     "     "     1  "   "  1   "  JJ.
ET CANDIDATE                                  Att Initials
MATCH
In <Town_name>, the initials of candidate <Sequence_number> of list <List_number>
are <Initials>.
```

```
FT6
In Oss, the prefix of candidate 1 of list 2 is: de.
 " Elst, "     "    "     "     7  "   "  4   "  van.
ET CANDIDATE                                  Att Prefix
MATCH
In <Town_name>, the prefix of candidate <Sequence_number> of list <List_number>
is: <Prefix>.
```

```
FT7
In polling station 24 in Oss, candidate 1 of list 1 got 88 votes.
 "    "        "    14  " Elst,    "     7  "   "  4   "  12   "  .
   ET ELECTION_RESULT                                    Att Number_of_votes
   ID: ET POLLING_STATION + ET CANDIDATE
       MATCH                MATCH
RT POLLING_STATION_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                          and POLLING_STATION
RT CANDIDATE_in_ELECTION_RESULT between ELECTION_RESULT(dependent)
                                   and CANDIDATE
In polling station <Station_number> in <Town_name>, candidate <Sequence_number>
of list <List_number> got <Number_of_votes> votes.
```

Note: in the last fact expressions, the town name serves two purposes: it is a part of the identifier of a list, but also of a candidate. Of course, this town name must be the same, and the verbalization assures this by having only one blank for a town name (see also the answer to task e below).

The resulting ERD in figure A3.6.7 is almost the same as the one from figure A3.4.6, but with an extra ET and two extra dependent RTs.

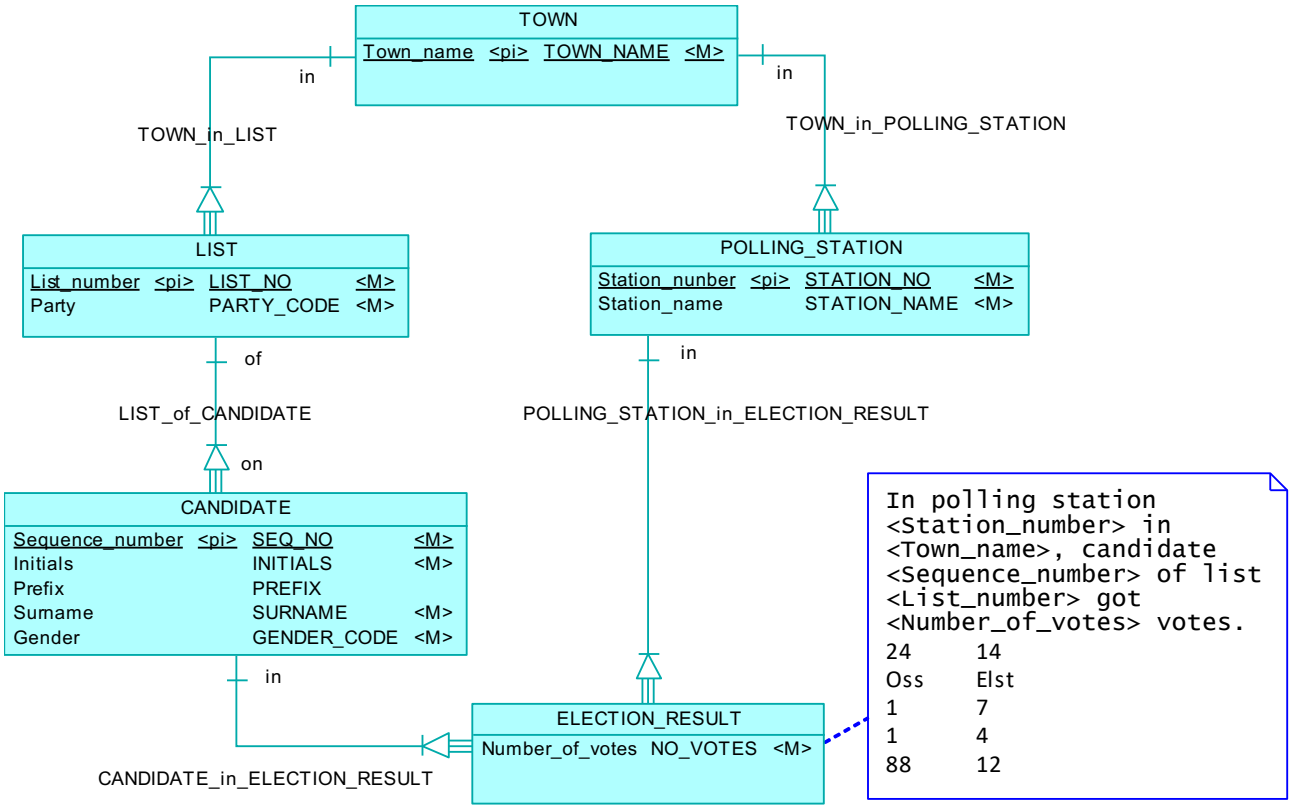


Figure A3.6.7 Result for elections in all towns

A verbalization for towns with councils could also have been given. This would have only one segment, and so should be analyzed as is explained in section 4.4.5 of the Reader DM-RDS:

```
FT0
The town Oss has a council.
 "    "  elst "  "    "   .
ET TOWN
ID: Att Town_name
The town <Town_name> has a council.
```

Then ET TOWN would be the first to be entered into the diagram.

Answer to task e: Are there new business rules now?

There is a loop in the diagram that requires the following *circular constraint* to be added:

BR2: In an election result, the town of the polling station must be same as the town of the candidate.

Note that the verbalization of an election result already ensures this (only one town is present in this verbalization), but in the CDM the constraint still needs to be added.

Answer to exercise 3.7 Book list

In this answer, only one verbalization per fact type is given.

Verbalizations with two blanks
Verbalizations from the example Book List:

```
FT1
Book 11 has ISBN 9035622459.
ET BOOK          Att ISBN(alternative identifier)
ID: Att Book_number
Book <Book_number> has ISBN <ISBN>.
```

Note: Book number and ISBN are both candidate identifiers, and Book number was chosen as <pi>.

```
FT2
The title of book 11 is: Brugwiskunde A.
             ET BOOK     Att Book_title
             MATCH
The title of book <Book_number> is: <Book_title>.
```

```
FT3
Boek 11 costs € 34,50.
ET BOEK       Att: Price
MATCH
Book <Book_number> costs € <Price>.
```

```
FT4
Book 11 is published by Liemers.
ET BOOK                 Att Publisher
MATCH
Book <Book number> is published by <Publisher>.
```

Note: because no other facts about publishers are to be recorded, it was decided to model the publisher of a book as an attribute Publisher, but it is also possible to model the publisher as an entity type PUBLISHER, with a relationship type between BOOK and PUBLISHER:

```
Book 11 is published by Liemers.
ET BOOK                 ET PUBLISHER
MATCH                   ID: Att Publisher_name
RT Publisher_of_book between BOOK and PUBLISHER
Book <Book_number> is published by <Publisher_name>.
```

This would result in a one-column domain table for the names of the publishers, with the usual pros and cons (cons: more tables and constraints; pros: pick list in interface, no typos in publisher names).

Both versions are shown here:

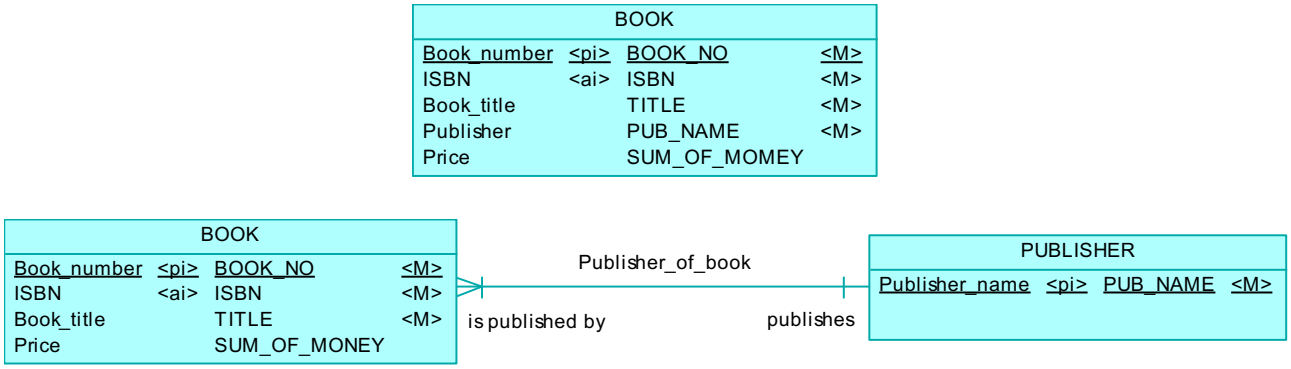


Figure 3.7.1 Two versions to model FT1 – FT4

The simpler model with just one ET was chosen to carry on with.

Verbalizations from the example Departments and Classes:

```
FT5
There is a class 3 in the MAVO department.
There is a class 4 in the MAVO department.
There is a class 3 in the HAVO department.
           Att or ET?     Att or ET?
```

Note 1: 'Class' cannot be an attribute of an ET DEPARTMENT, because a department can have more than one class. So 'CLASS' must be an ET.

Note 2: To identify a class, we need not only the class number but the department name as well: there is a class 3 in the MAVO department, but also a class 3 in the HAVO department.

Note 3: It follows from note 2 that we have two possibilities now:

1. DEPARTMENT as an ET, with a RT between them, dependent on the CLASS side.
2. Department as another identifying attribute of ET CLASS.

Both possibilities are shown below, and each in two ways of analyzing: the first with two segments, the second with only one segment (they yield the same diagram):

Possibility 1a:

```
     There is a class 3 in the HAVO department.
                ET CLASS        ET DEPARMENT
                                ID: Att Department_name
                ID: Att Class_number + ET DEPARTMENT
                                       MATCH
     RT DEPARTMENT_in_CLASS between CLASS(dependent) and DEPARTMENT
     There is a class <Class_number> in the <Department_name> department.
```

Possibility 1b:

```
     There is a class 3 in the HAVO department.
                ET CLASS
                ID: Att Class_number + ET DEPARTMENT
                                       ID: Att Department_name
     RT DEPARTMENT_in_CLASS between CLASS(dependent) and DEPARTMENT
     There is a class <Class_number> in the <Department_name> department.
```

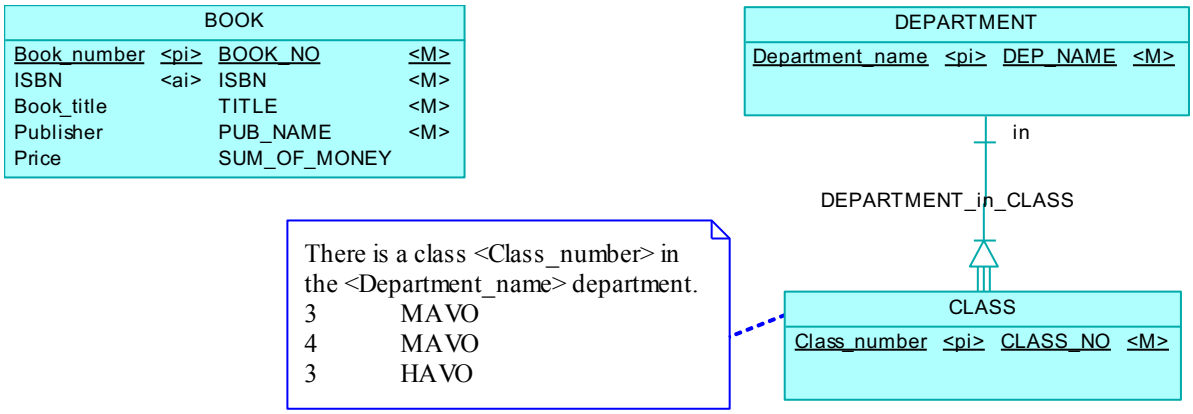


Figure A3.7.2 **First possibility to model FT5**

Possibility 2a:
```
There is a class_3 in the HAVO department.
           ET CLASS        Att Department
           ID: Att Class_number + Att Department
There is a class <Class_number> in the <Department> department.
```

Possibility 2b:
```
There is a class_3 in the HAVO department.
           ET CLASS
           ID: Att Class_number + Att Department
There is a class <Class_number> in the <Department> department.
```

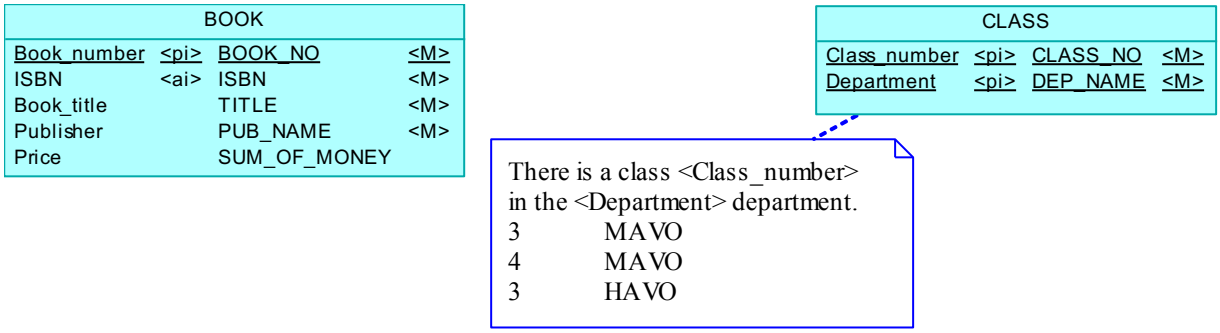


Figure A3.7.3 Second possibility to model FT5

Because no other facts about departments are to be recorded, it was decided to model the department of a class as an attribute Department in this exercise. However, in a more elaborate example with a larger scope, there would probably be other facts about departments, so a modeling with an ET DEPARTMENT would then be chosen instead.

Verbalizations with four blanks:

Verbalizations from the example Departments and Classes:

```
FT6
In class 1 of department Bridge, the subject WI is taught 4 hours per week.
In class 2 of department Bridge, the subject WI is taught 3 hours per week.
In class 3 of department MAVO,   the subject WI is taught 4 hours per week.
   ET SCHEDULE                                          Att Number_of_hours
   ID: ET CLASS + Att Subject
       MATCH
In class <Class_number> of department <Department>, the subject <Subject> is
taught <Number_of_hours> per week.
```

Note 1: Time durations, weights, lengths and other such measured quantities should (almost) always be modeled as an attribute. Therefore the other components together must belong to an ET.

Note 2: For clarity, a verbalization was chosen in which all components of ET SCHEDULE appear in a connected sentence part. This is no requirement however, as the following alternative verbalization illustrates, which yields the same result in the analysis, yet contains a gap in the parts for SCHEDULE:

```
On the schedule for class 1 in the Bridge department, there are 3 hours per week
for the subject WI.
       ET SCHEDULE
       ID: ET CLASS + Att Subject                          Att Number_of_hours
           MATCH
On the schedule for class <Class_number> in the <Department> department, there
are <Number_of_hours> hours per week for the subject <Subject>.
```

Note 3: The components of ET SCHEDULE contain ET CLASS, which was found earlier. In addition, there is a subject. This subject can be modeled as Att or ET. The version with ET SUBJECT is:

```
On the schedule for class 1 in the Bridge department, there are 3 hours per week
for the subject WI.
       ET SCHEDULE
       ID: ET CLASS + ET SUBJECT                           Att Number_of_hours
           MATCH      ID: Att Subject_code
       RT SUBJECT_in_SCHEDULE between SCHEDULE(dependent) and SUBJECT
On the schedule for class <Class_number> in the <Department> department, there
are <Number_of_hours> per week for the subject <Subject>.
```

See diagrams for these two versions for SUBJECT on the next page.

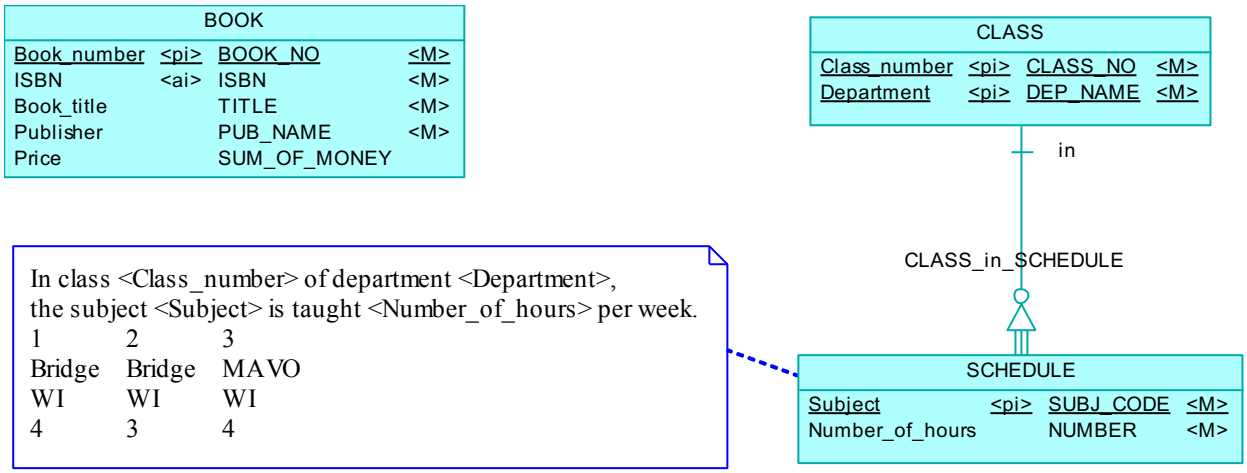


Figure A3.7.4 **First version to model FT6**

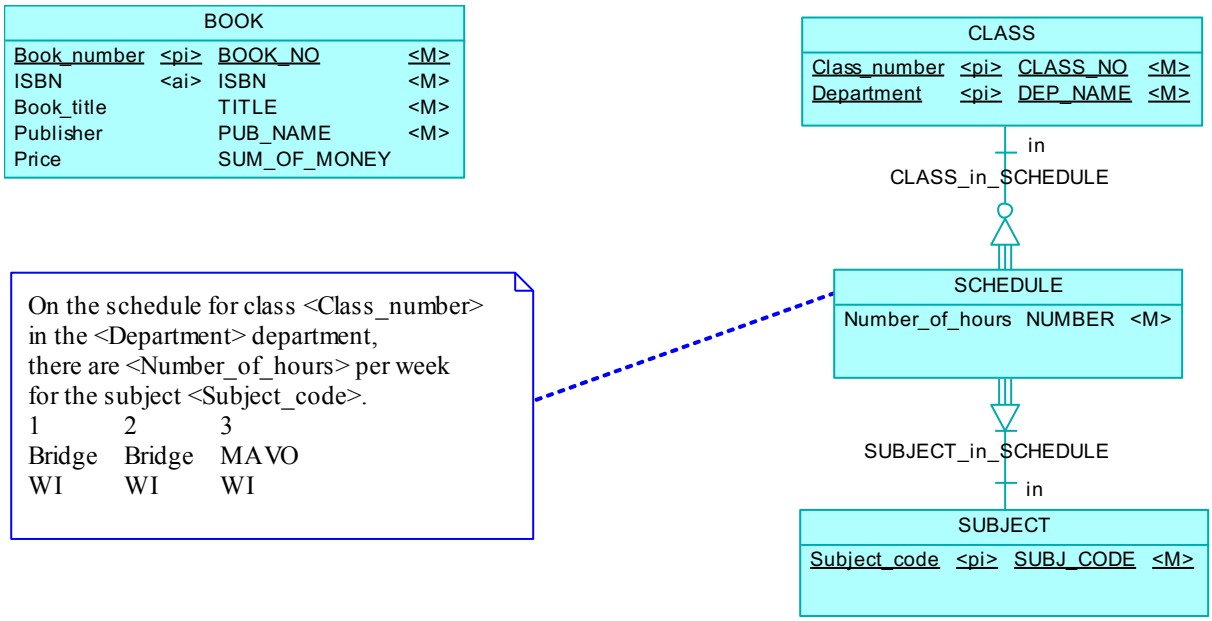


Figure A3.7.5 **Second version to model FT6**

The simpler version in figure A3.7.4 was chosen.

Verbalizations with five blanks:

Verbalizations from the example Departments and Classes

```
FT7
The 4th edition of book 11 is to be used for
                              the subject WI in class 1 of the Bridge department.
The 3rd edition of book 13 is to be used for
                              the subject WI in class 2 of the Bridge department.
The 2nd edition of book 13 is to be used for
                              the subject WI in class 3 of the MAVO department.
The 2nd edition of book 35 is to be used for
                              the subject AK in class 2 of the Bridge department.
  Att of which ET?
```

Clearly, '2nd edition' by itself does not identify any entity, therefore it must be an attribute of some entity type.

Note 1: Only **one** edition of a book can be used for a subject in a class of a department: two or more editions for the same combination of book, subject, class and department cannot occur. This suggests that the edition is an attribute of an entity type TEACHING_MATERIAL that contains the other four components as identifiers. This possible way of analyzing will be explored below first.

Note 2: Another possible way of analyzing is: the sentence part '2nd edition of book 35' identifies a specific book edition, and '2nd edition' is one of the identifying attributes of an entity type BOOK_EDITION. This second way will be explored below later, and shown to lead to a more complicated model that requires an awkward constraint and manual repair in the relational schema.

First way of analysis

The sentence part '2nd edition' is seen as an attribute of an entity type, to which the other four components must then belong. The previously modeled ETs BOOK and SCHEDULE can be easily recognized in these components.

```
The 2nd edition of book 35 is to be used for
                              the subject AK in class 2 of the Bridge department.
    Att Edition    ET TEACHING_MATERIAL
                   ID: ET BOOK + ET SCHEDULE
                       MATCH     MATCH
RT BOOK_in_TEACHING_MATERIAL between TEACHING MATERIAL(dependent)
                                 and BOOK
RT SCHEDULE_in_TEACHING_MATERIAL between TEACHING_MATERIAL(dependent)
                                    and SCHEDULE
The <Edition> edition of book <Book_number> is to be used for the subject <Subject>
in class <Class_number> of the <Department> department.
```

The ERD in figure A3.7.6 shows the final result, with only the non-trivial predicates and population.

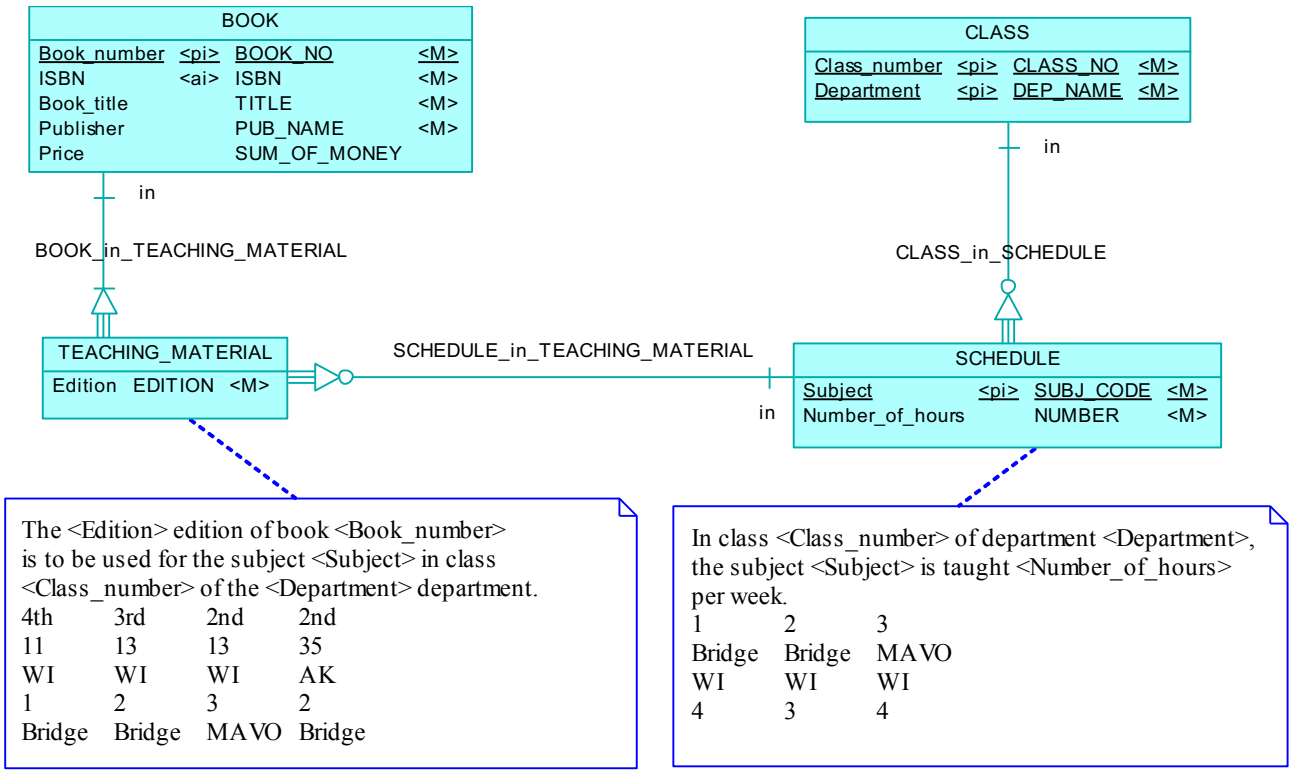


Figure A3.7.6 **Final ERD, first way (recommended)**

The relational model that follows from this model is shown here as well:

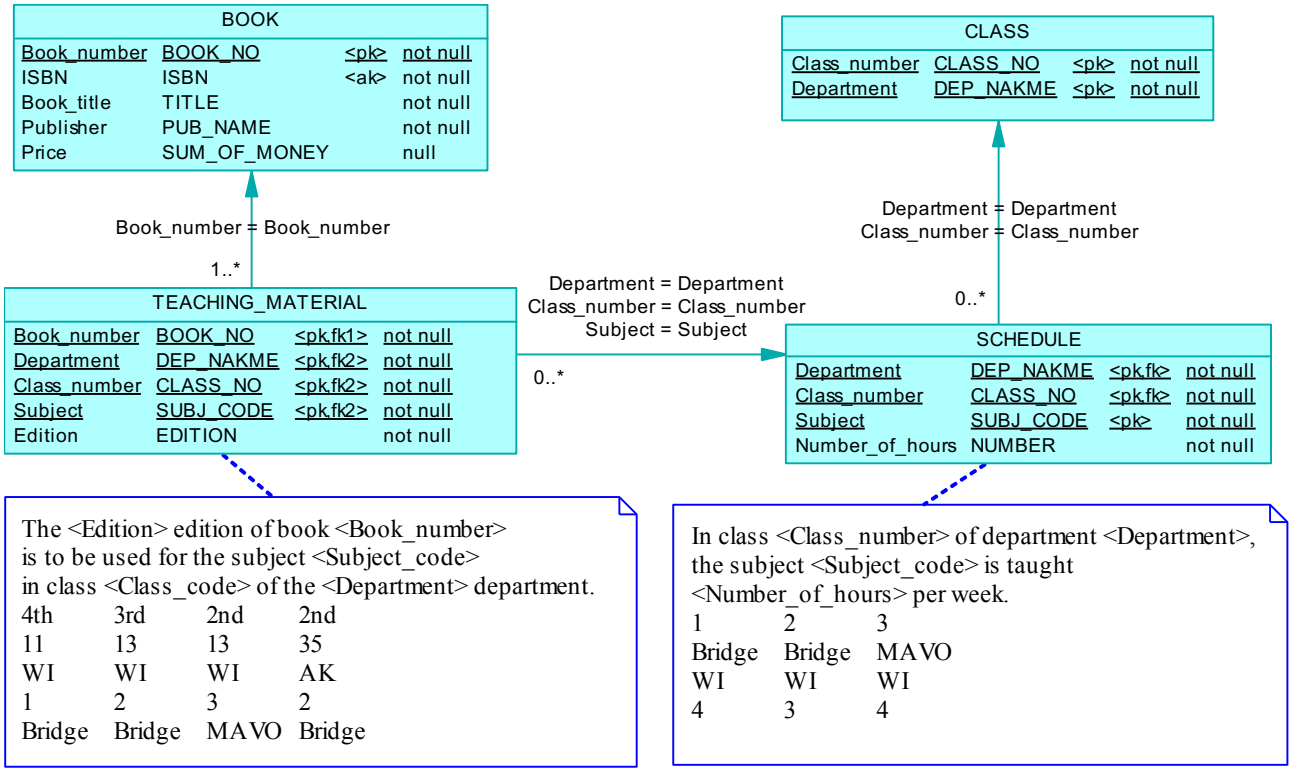


Figure A3.7.7 **Relational schema that follows from the ERD in figure A3.7.6**

Compare this with the relational model in figure A3.7.9 that results from the other model in figure A3.7.8, in particular note the <pk> in table TEACHING_MATERIAL here, and compare it with the <pk> of the same table below.

Second way of analysis

The sentence part '2nd edition' is seen as an identifying attribute of an entity type for book editions, to which also the book number belongs. The previously modeled ETs BOOK and SCHEDULE can be easily recognized.

```
The 4th edition of book 11 is to be used for
                              the subject WI in class 1 of the Bridge department.
The 3rd edition of book 13 is to be used for
                              the subject WI in class 2 of the Bridge department.
The 2nd edition of book 13 is to be used for
                              the subject WI in class 3 of the MAVO department.
The 2nd edition of book 35 is to be used for
                              the subject AK in class 2 of the Bridge department.
    ET BOOK_EDITION          ET SCHEDULE
    ID: ET BOOK + Att Edition     MATCH
        MATCH
RT BOOK_in_BOOK_EDITION between BOOK_EDITION(dependent) and BOOK
RT Teaching_material between BOOK_EDITION and SCHEDULE
The <Edition> edition of book <Book_number> is to be used for the subject
<Subject_code> in class <Class_code> of the <Department> department.
```

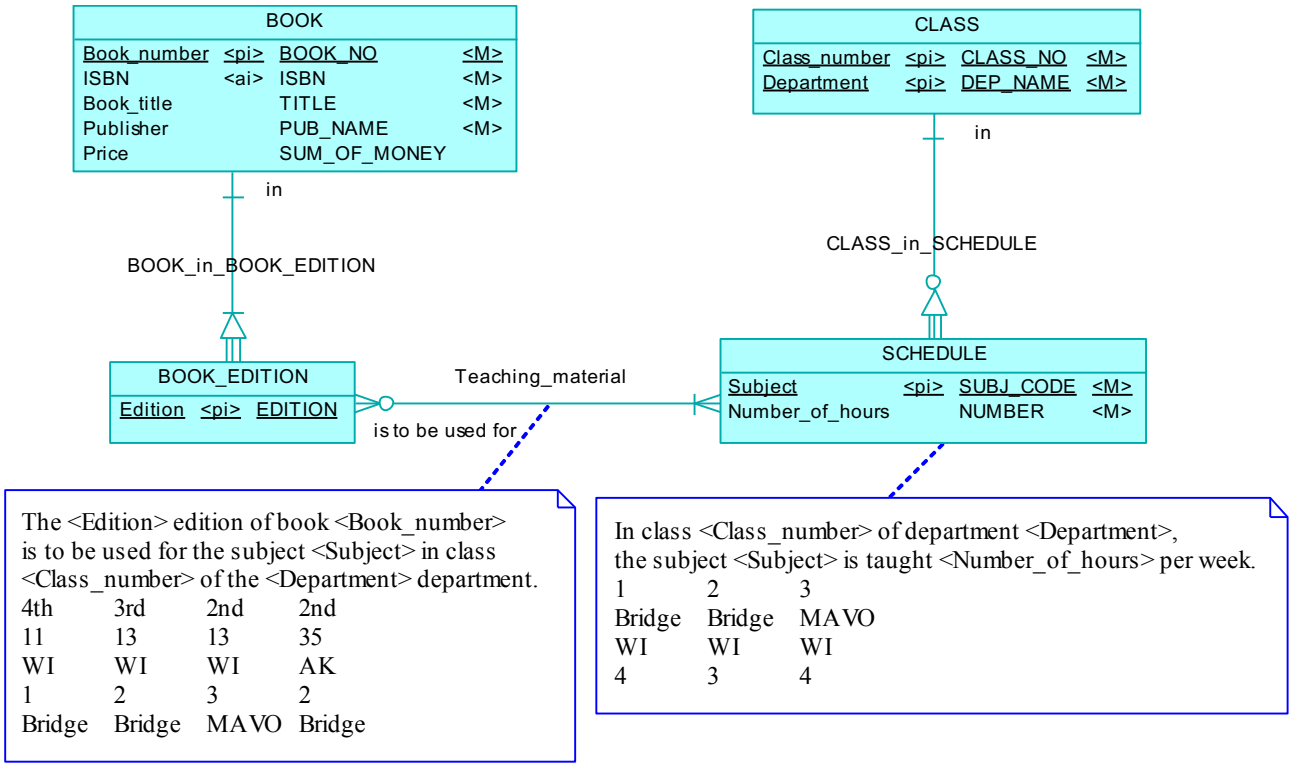


Figure A3.7.8 **Final ERD, second, more awkward way (not recommended)**

At first sight this looks plausible enough, but the rule that only **one** edition of a book can be used for a subject in a class of a department (see **Note 1** above) is not satisfied: two or more editions for the same combination of book, subject, class and department can be entered in the population here. This is perhaps better visible in the resulting relational schema (see figure A3.7.9). Note that the <pk> of table TEACHING_MATERIAL is now too wide: it includes attribute Edition as well, which makes violations of the rule from **Note 1** possible.

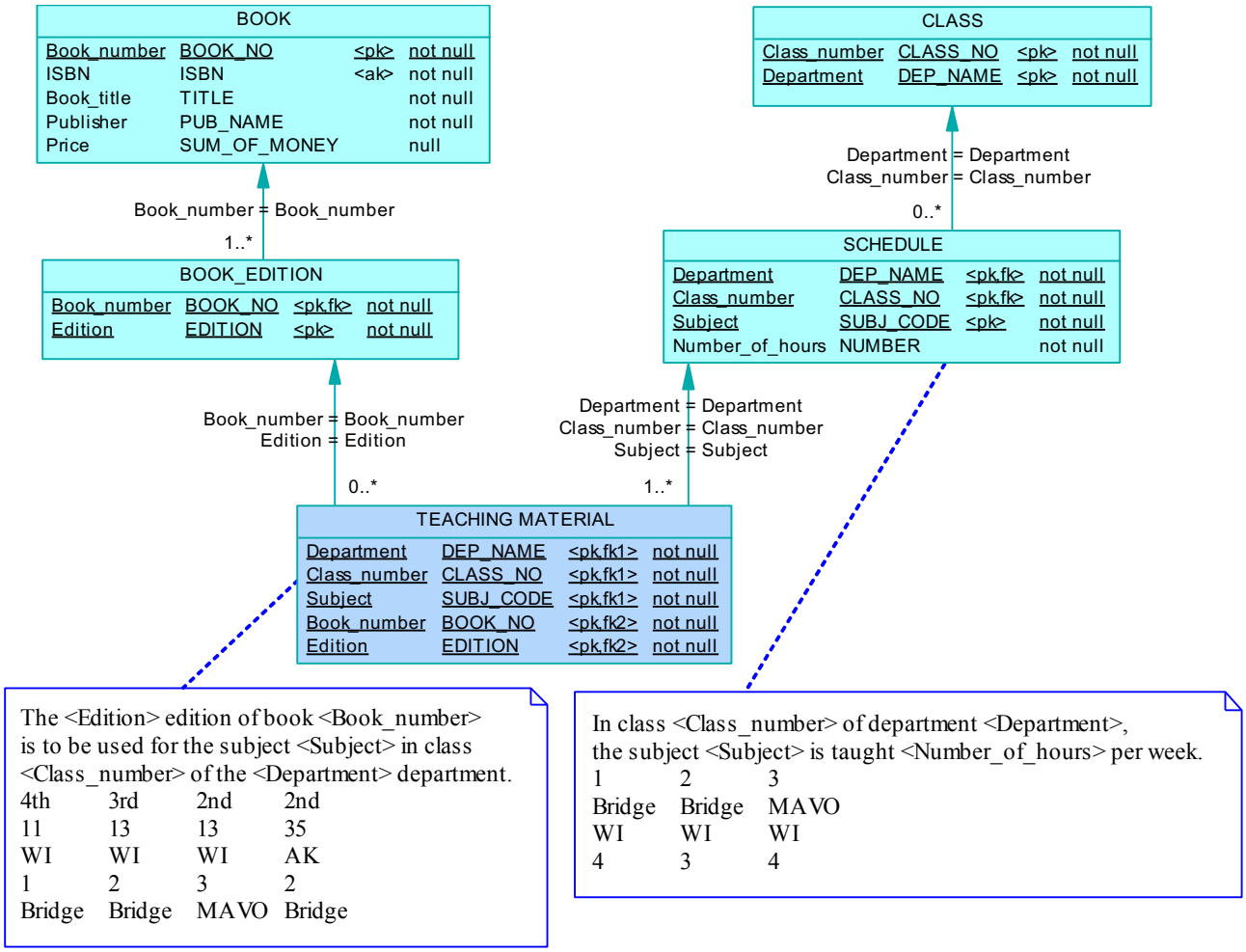


Figure A3.7.9 **Relational schema that follows from the ERD in figure A3.7.8 (to be corrected manually)**

To remedy this, we would have to do two things:

1 In ERM, specify a constraint on RT Teaching_material:
C1: Only one edition can be specified
for each combination of book, subject, class and department.

2 In the generated relational schema, manually change the <pk> in TEACHING_MATERIAL.

The first way of modeling results in fewer tables and less complications, so it was chosen as the final model.

Please note that information modeling can be difficult, even in small examples as in this exercise. Techniques like verbalizing can help a great deal, but in the end it is still an art to draw up a good information model.

2.4 Theme 4: Deriving a PDM from a CDM

Answer to exercise 4.1 Employee-department-location

Answer to task a

Result of substeps 1-7: see figure A4.1.1. Note the dominant side in RT Department_manager, chosen because this is a 1-1 RT with minimum cardinalities ZERO and ONE, so the ONE-side is chosen as the dominant (parent) side (see the Reader DM-RDS for further explanation).

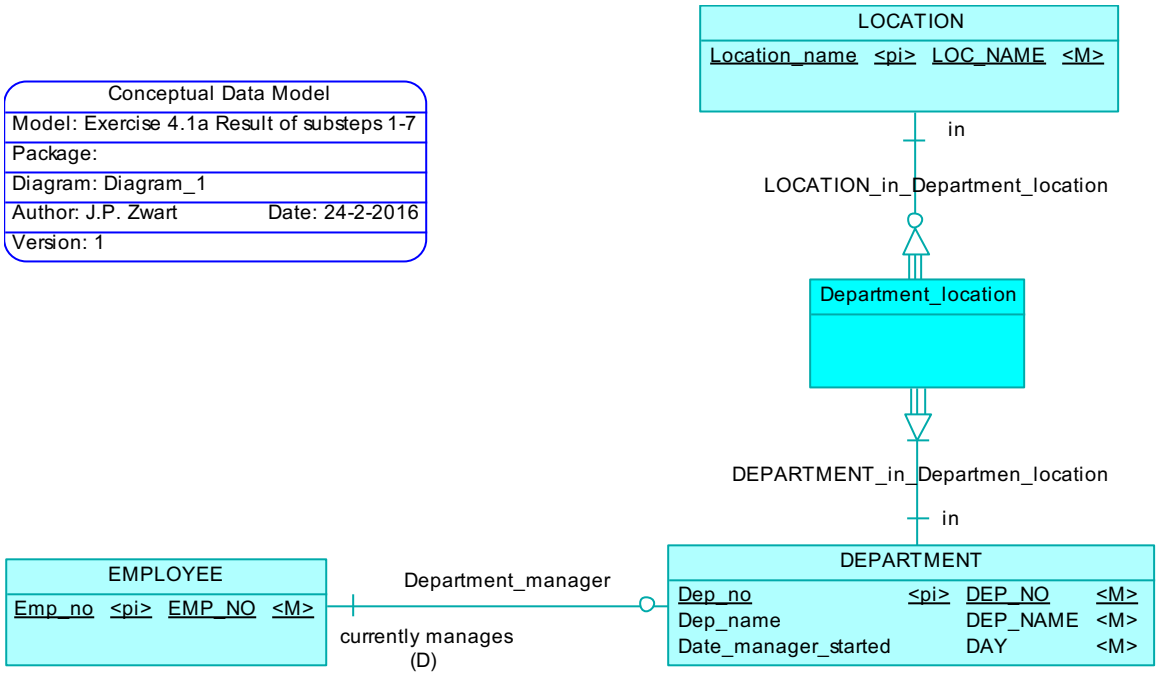


Figure A4.1.1 Result of substeps 1-7

The final result is shown in figure A4.1.2. In table DEPARTMENT, the automatically generated FK-column name was 'EmpNo'. This was changed to the clearer name 'Manager'.

Note: three arrows, but four references!

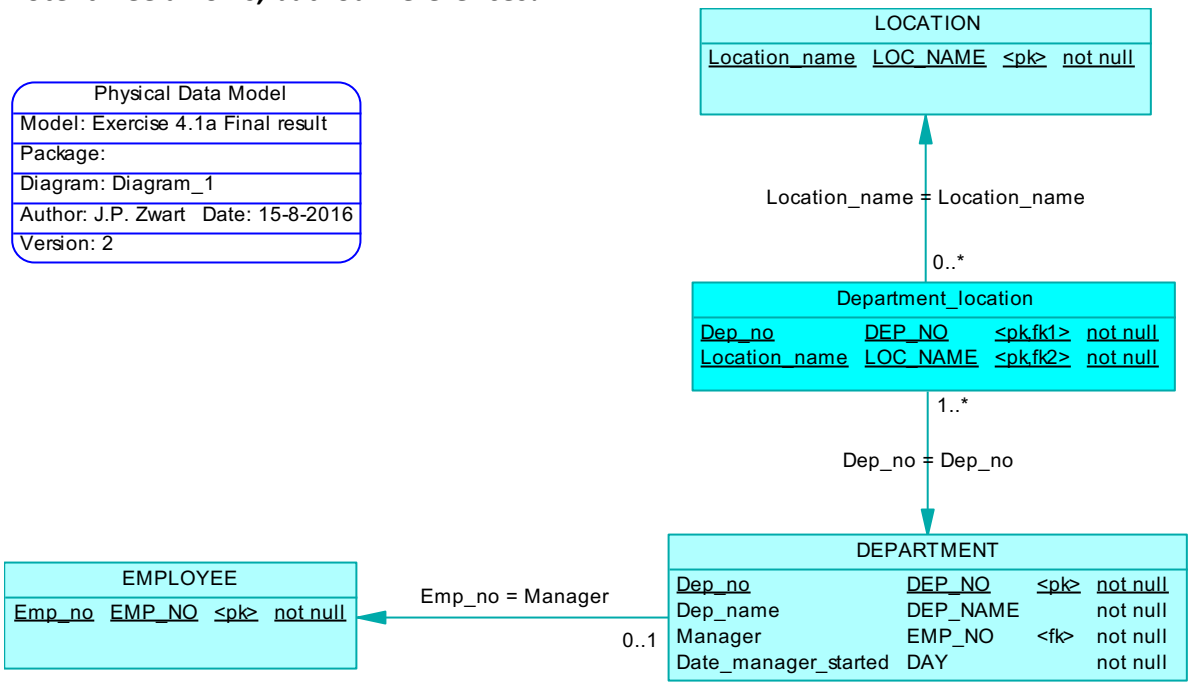


Figure A4.1.2 Final PDM

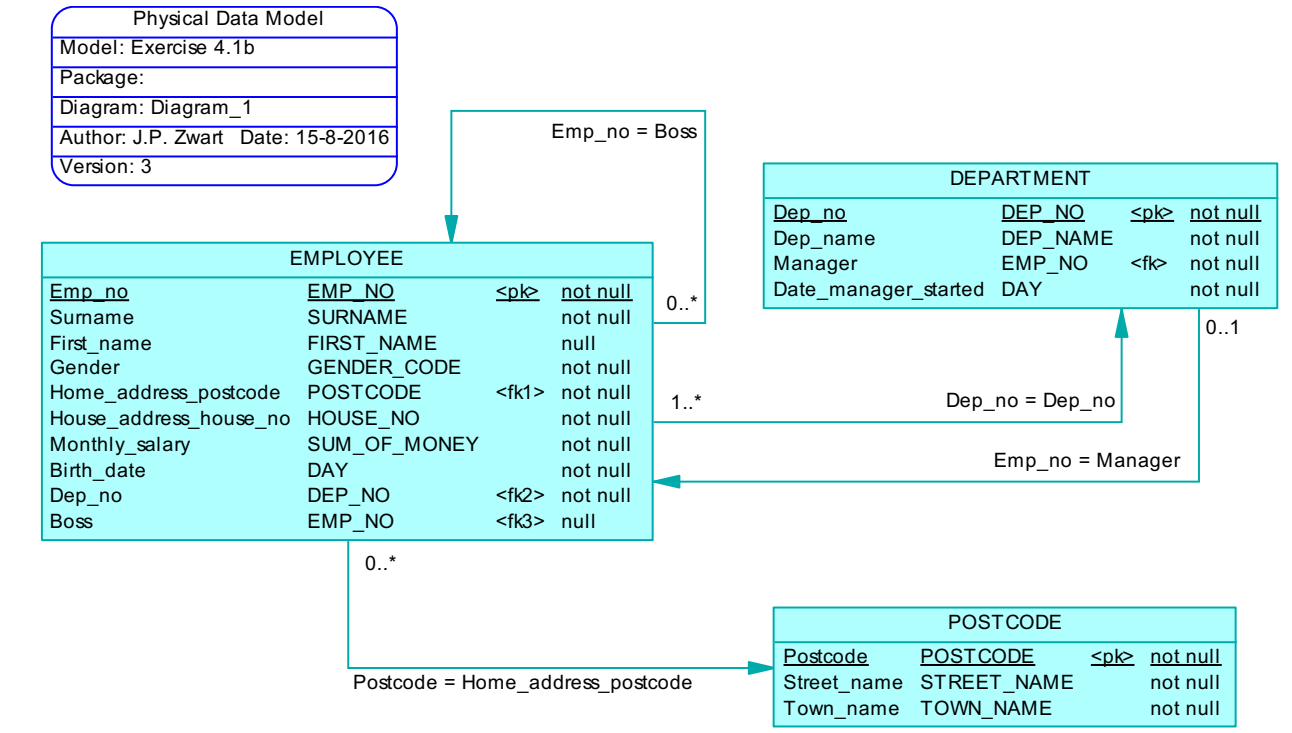
Answer to task b

Figure A4.1.3 shows only the final result (again, the ONE-side of RT Department_manager was chosen as the dominant side). Note the integrity rules.

The following changes were made in the automatically generated column names (to make them clearer) in the generated PDM:

- Table EMPLOYEE:
  - <fk2>: 'Postcode' → 'Home_address_postcode'
  - <fk3>: 'EMP_Emp_no' → 'Boss'
- Table DEPARTMENT:
  - <fk>: 'Emp_no' → 'Manager'.

Note: four arrows, but five references!



IR1 If an employee manages a department, (s)he must be assigned to the same department.
Table EMPLOYEE(Emp_no, Dep_no) and DEPARTMENT(Manager, Dep_no):
If Emp_no = Manager, then the two values of Dep_no must be the same as well.
Examples that together violate this constraint:
Employee 4321234 is assigned to department D3.
The current manager of department D4 is employee 4321234.

IR2 No cycles can exist in supervision: "A supervises B supervises C supervises A" cannot be.
This concerns Table EMPLOYEE(Boss, Emp_no).
Examples that together violate this constraint:
Employee 4321234 supervises employee 1111111.
Employee 1111111 supervises employee 4321234.

Figure A4.1.3 PDM for task B

Answer to exercise 4.2 Participant-trip-hotel

Answer to task a

Are the following statements about this ERD TRUE or FALSE? Mark the appropriate checkbox.

i. The means of transport must be recorded for each participant.
☑ TRUE ☐ FALSE

ii. The identifier of a participant is Sequence_number together with Trip_code.
☑ TRUE ☐ FALSE

iii. Every trip has a stay in at least one hotel.
☐ TRUE ☑ FALSE

iv. A hotel can be registered that is not the accommodation in any trip.
☐ TRUE ☑ FALSE

v. Because Location in HOTEL and Venue in TRIP have the same domain, they will be stored in the same table.
☐ TRUE ☑ FALSE

Answer to task b

Note: Did you get the cardinalities at the foot of the reference-arrows right? PowerDesigner did not and had to be corrected manually.

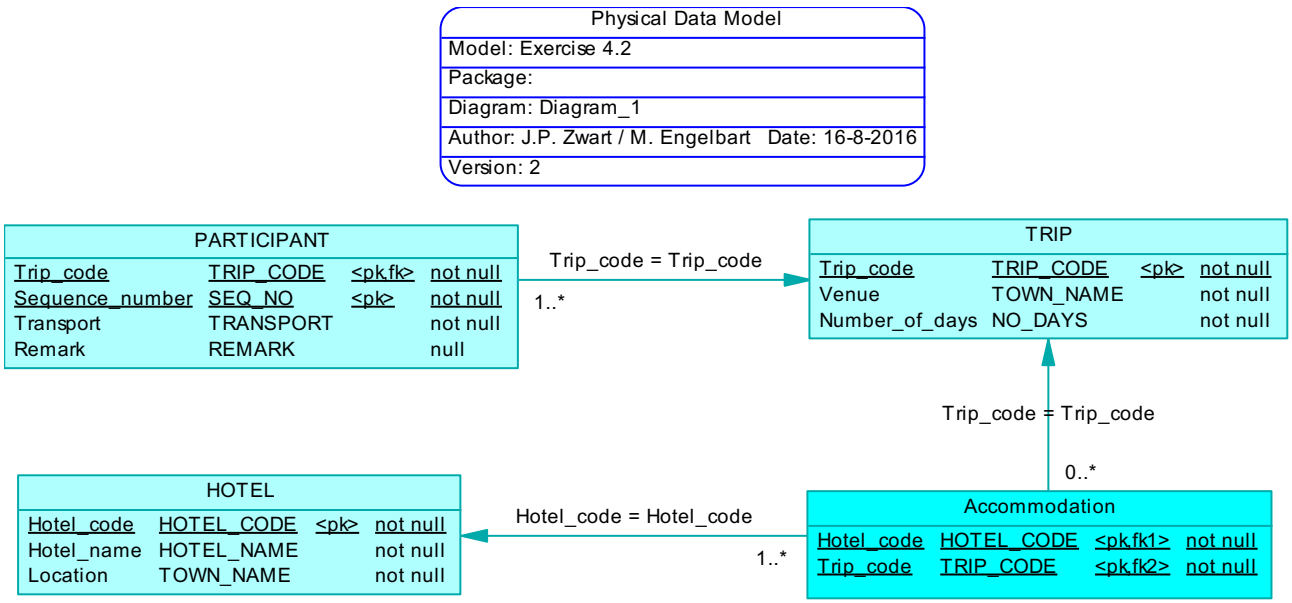


Figure A4.2.1 Complete PDM

Answer to exercise 4.3 Employee-car-building

Answer to task a

Can a database built according to this data model supply the answers to the following questions?

Q1: In which building did employee EM123 park his car?

Q2: In which parking space in which building is the car with license plate KA-12-DE parked?

Q1: Yes (if the relevant facts were indeed entered).

Q2: No, the current parking space of a car is not modeled, only the building the car is in.

Task b

Are the following situations allowed according to this data model?

S1: An employee can park several cars in the buildings.

S2: A car can have parking spaces in several buildings.

S1: Yes: the max. card. of Car_ownership at the side of CAR is MANY, and each car can be parked.

S2: Yes, if all constraints have been given. Then there is no restriction in which building a car is parked, but it can only be parked in one building at a time (max. card. 1 of Car_parking_location).

Task c

Have the following business rules been modeled in this data model?

B1: A car must always be parked in the same building.

B2: A building can also host cars of non-employees (guests).

B1: No. It is only modeled that a car is parked in a building, but there is no restriction which building.

B2: No. It might be true, but is not modeled at all here.

Task d

Transform this CDM into a complete PDM. Add integrity constraints that cannot be shown graphically (if any) in a text block.

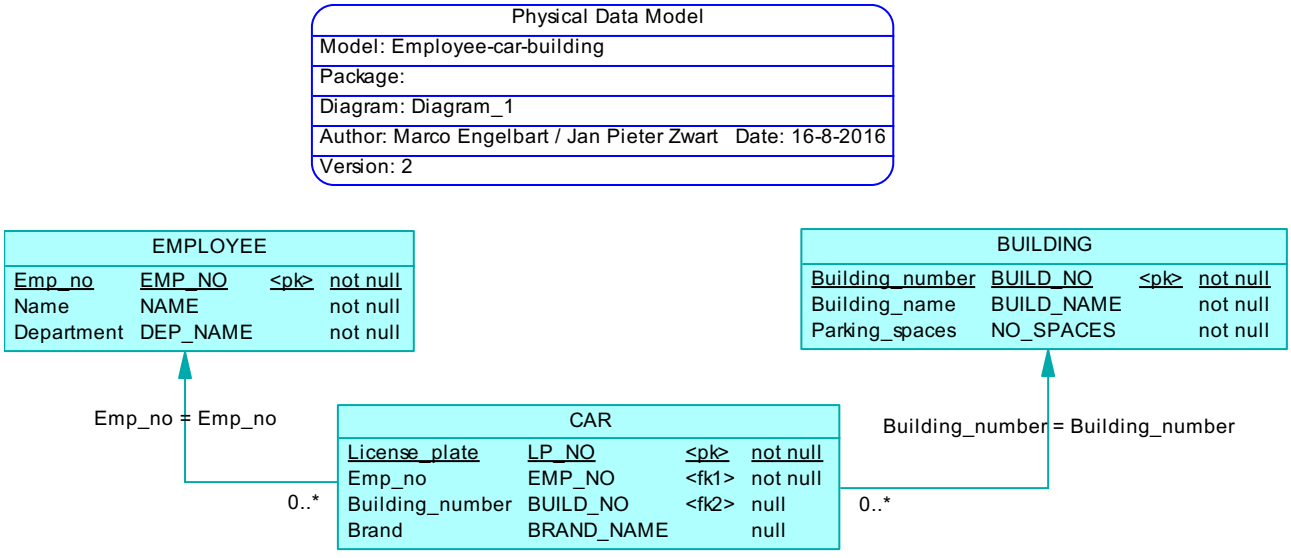


Figure A4.3.1 Complete PDM

Answer to exercise 4.4 Marriages

Ansswer to task a

Can a database built according to this data model supply the answers to the following questions?

Q1: Who are the parents of person 643?
Q2: Of how many children is person 555 the father?

Q1: Yes. Via RT Child_from_marriage the marriage is found, and the parents via the other RTs.
Q2: No, because possible illegitimate children are not modeled. But for each of his marriages the number of children can be counted and summed up.

Answer to task b

Are the following situations allowed according to this data model?

S1: A person can be married to someone with the same gender.
S2: The model is flawed: it allows a person to be married with himself/herself.
S3 A marriage can be without offspring.
S4: The model is flawed: it allows a person to marry several other persons on the same day.
S5: A child can have more than one parent of the same gender.

S1: Yes, unless the business rules that a husband must be male and a wife must be female are enforced (these BRs are not mentioned in the model).
S2: Yes, unless same-sex marriages are not allowed (see S1). Otherwise: it is possible.
S3: Yes (min. card. ZERO of Child_from_marriage at the side of PERSON).
S4: Yes, the model allows this.
S5: See S1.

Answer to task c

Have the following business rules been modeled in this data model?

B1: Only married persons can be registered.
B2: Illegitimate children can also be registered.

B1: No (min. card. ZERO in RTs Husband_of_MARRIAGE and Wife_of_MARRIAGE).
B2: No. Only children from marriages can be registered.

Answer to task d

Transform this CDM into a complete PDM. Add integrity constraints that cannot be shown graphically (if any) in a text block.

Note: The PDM that results from the given CDM is given below. But (apart from being a useful exercise in dealing with multiple RTs between the same two ETs) the CDM is actually a very bad model indeed: it falls far short of modeling marriages in real-life situations (which should at least include the end date and reason for the end of a marriage), allows polygamy, etc. etc.). Therefore no attempt was made to add constraints, of which there are many, for example:

- Same-sex marriages: allowed or not?
- Marriage periods cannot overlap
- …

It would be a challenging exercise to model marriages correctly (try it!).

Awkward names of generated columns were changed manually.

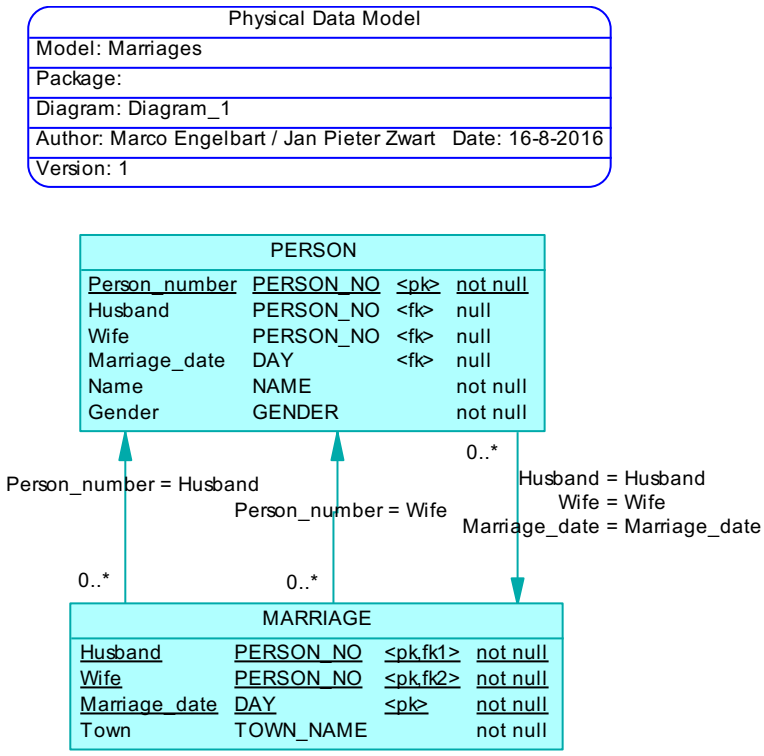


Figure A4.4.1 PDM that follows from the given CDM (but very bad model for marriages)

Answer to exercise 4.5 Exam results

The PDM generated from the CDM in figure A3.2.7 is:

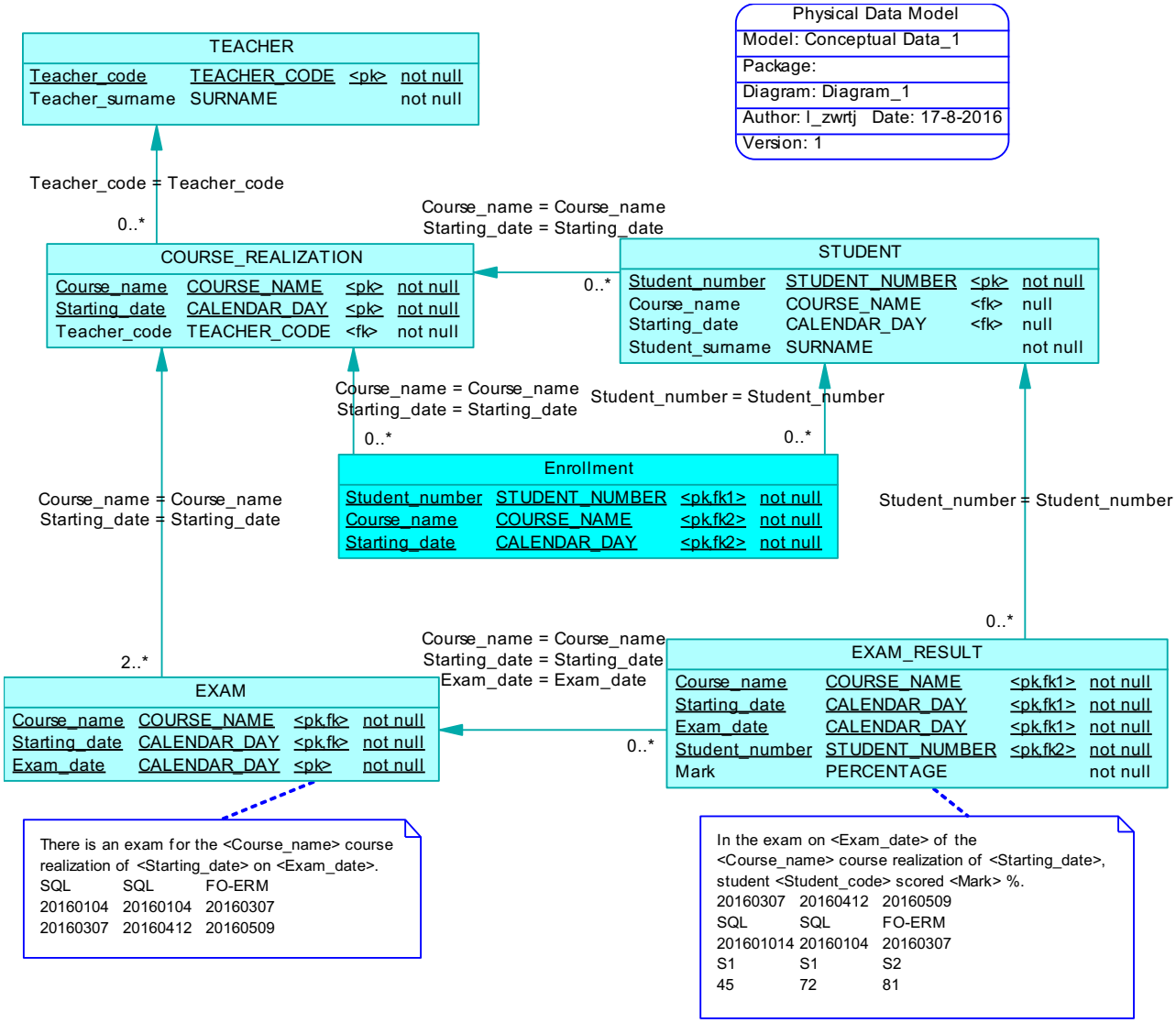


Figure A4.5.1 PDM for Exam results

Answer to exercise 4.6 Solar eclipses

No answer is given here. Consult your teacher for feedback on your work on this exercise.

Answer to exercise 4.7 Conference workshops

The PDM generated from the CDM in figure A3.4.8 is:

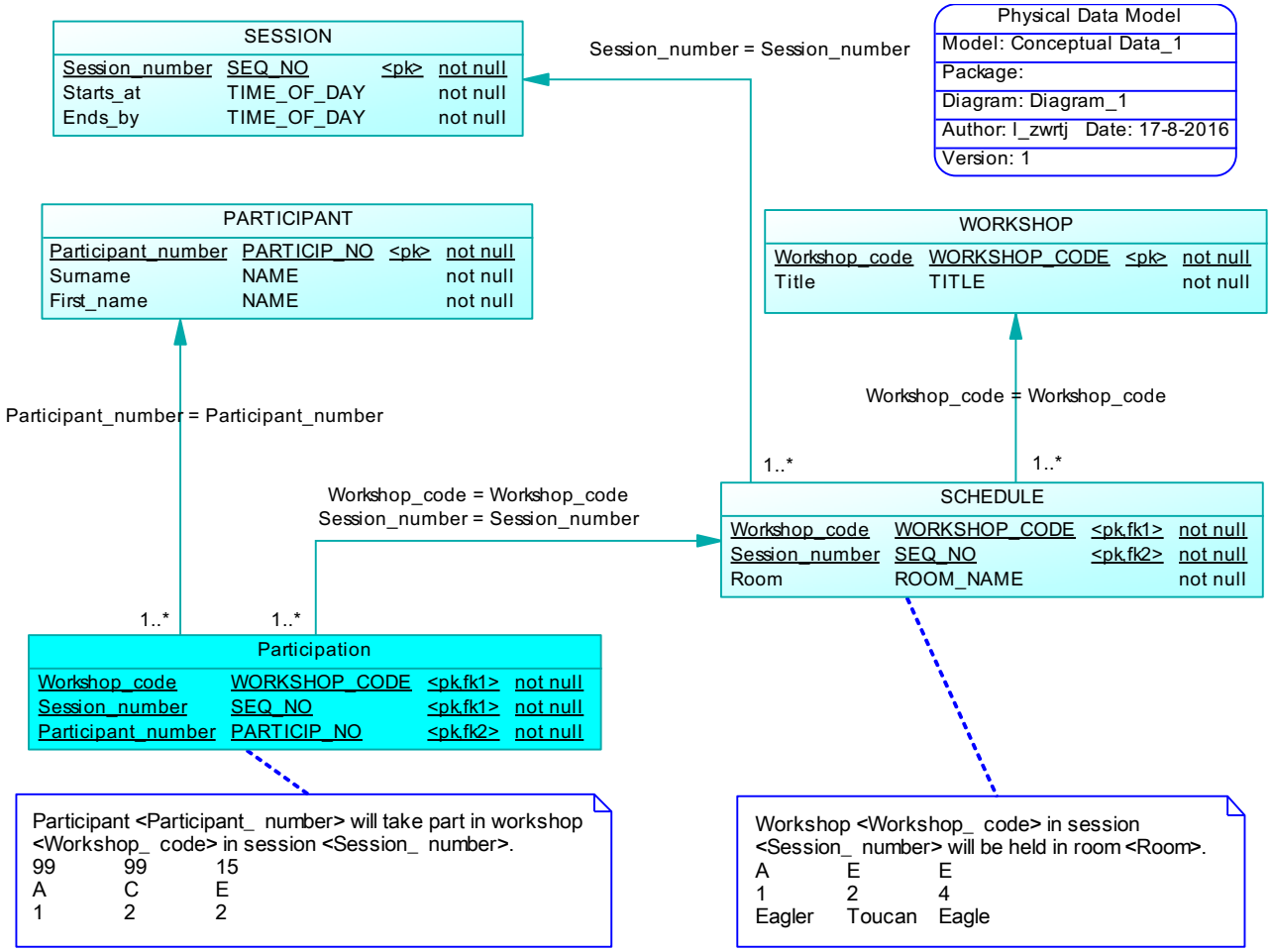


Figure A4.7.1 PDM for conference workshops

Note the many mandatory-child references, which all have to be implemented by hand.
Note also the unchanged predicates and example populations, now all attached to tables.

Answer to exercise 4.8 Meal recipes

No answer is given here. Consult your teacher for feedback on your work on this exercise.

Answer to exercise 4.9 Council elections

When PowerDesigner generated the PDM from the CDM in figure A3.6.7, table ELECTION RESULT contained a sixth column CAN.Town_name, in addition to the other column Town_name. This is because in figure A3.6.7 there are two routes to follow from ET ELECTION_RESULT to ET TOWN, which both generate a FK-column in the resulting table.

However, since the town names must be ths same in both columns, one can be dropped. So column CAN.Town_name was deleted manually from the PDM, and the reference manually changed to refer to the other column Town_name. The final PDM is shown in figure A4.9.1 below.

Note the many mandatory child refeences, which must all be implemented by hand.

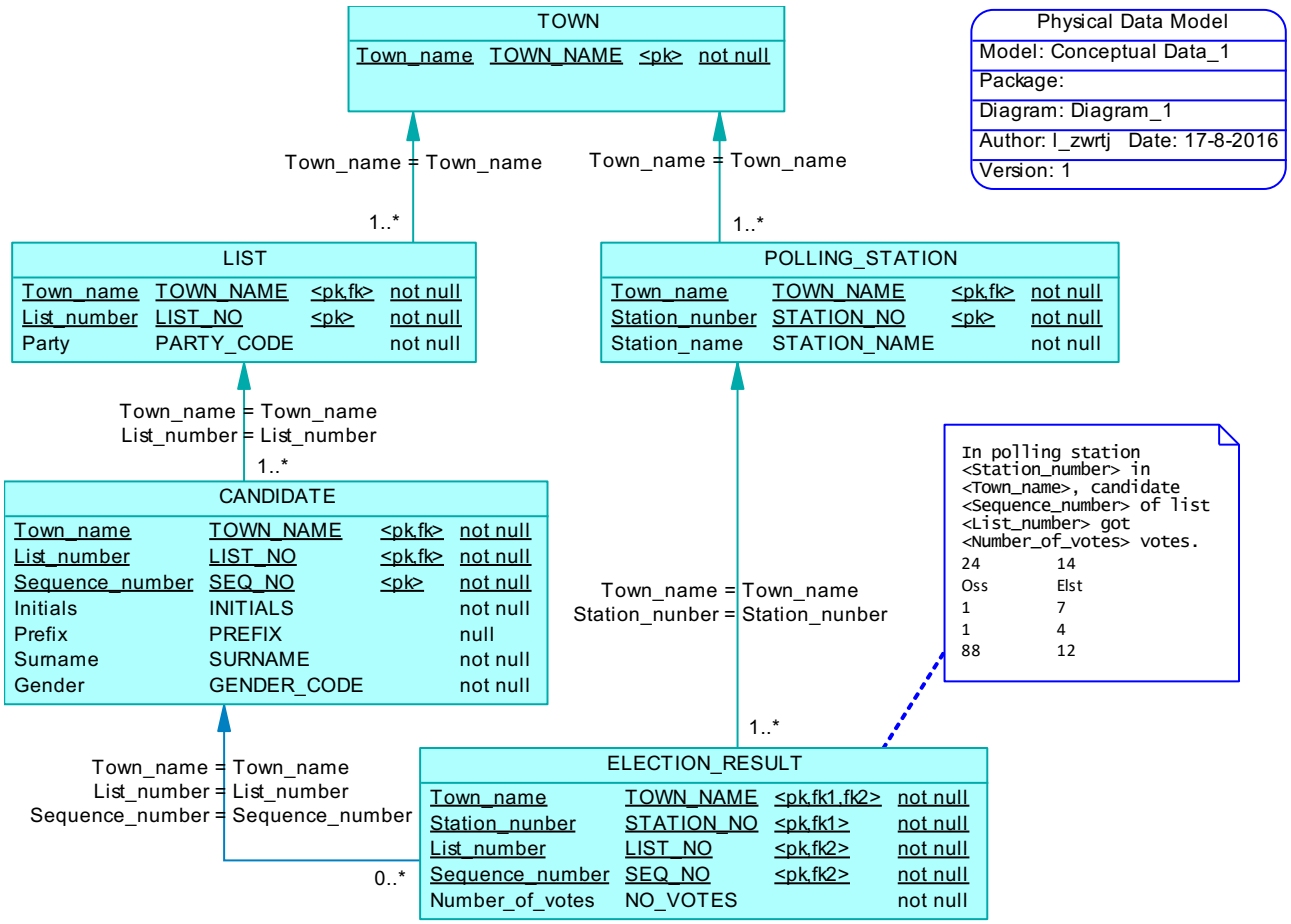


Figure A4.9.1 Final PDM for Council elections

2.5 Theme 5: Subtypes in ERM

Answer to exercise 5.1 Workstations

DEVICE			
Device_number	<pi>	DEV_NO	<M>
Brand		BRAND_NAME	<M>
Date_purchased		DAY	<M>
Processor		PROC_TYPE	
Hard_disk_ (GB)		CAPACITY	
RAM_(GB)		CAPACITY	
Diameter_(inch)		SIZE	

Figure A5.1.1 ERD for devices (same as figure E5.1.2)

Answer to task a

BR1 Processor type, hard disk capacity and internal memory capacity are only to be registered for computers.

BR2 The screen size is only to be registered for monitors.

Answer to tasks b and c

The business rules cannot be translated into constraints yet: information about which device is a computer or monitor is missing. Another attribute Device_type is needed (see figure A5.1.2, which also contains an example population), and then the constraints can be given.

DEVICE			
Device_number	<pi>	DEV_NO	<M>
Device_type		DEV_TYPE	<M>
Brand		BRAND_NAME	<M>
Date_purchased		DAY	<M>
Processor		PROC_TYPE	
Hard_disk_ (GB)		CAPACITY	
RAM_(GB)		CAPACITY	
Diameter_(inch)		SIZE	

D123	D127
Computer	Monitor
Doll	Pholips
23/1/2016	15/10/2013
Inter Cole M6	--
8	--
2	--
--	17

Figure A5.1.2 ERD for devices with extra attribute and example population

C1 In ET DEVICE, attributes Processor, Hard_disk_(GB) and RAM_(GB) must have a value if and only if in the same tuple attribute Device_type has the value ‘Computer’.

C2 In ET DEVICE, attribute Diameter_(inch) must have a value if and only if in the same tuple attribute Device_type has the value ‘Monitor’.

Note that it is not clear from the decription in figure E5.1.1 whether C1 applies to all computers and C2 applies to all monitors, or not. Here, it is assumed this is indeed the case.

Answer to task d

Figure A5.1.3 gives the ERD with subtypes. Note the following points:

- C1 and C2 are now shown in the diagram: the subtypes with their SDRs and <M>-attributes.
- The SDRs only use fact types from the supertype, as is required by the rule in figure 6.5 from the Reader DM-RDS.
- In the example population of the subtypes, the device number (inherited from the supertype) is needed as well, for all the fact types in the subtypes, not only the subtype defining fact types (most predicates are not shown, but they are of course still there).
- The subtype defining fact types are derivable and follow from the SDRs. Still, it is best to show them explicitly, to remind the modeler of their existence.

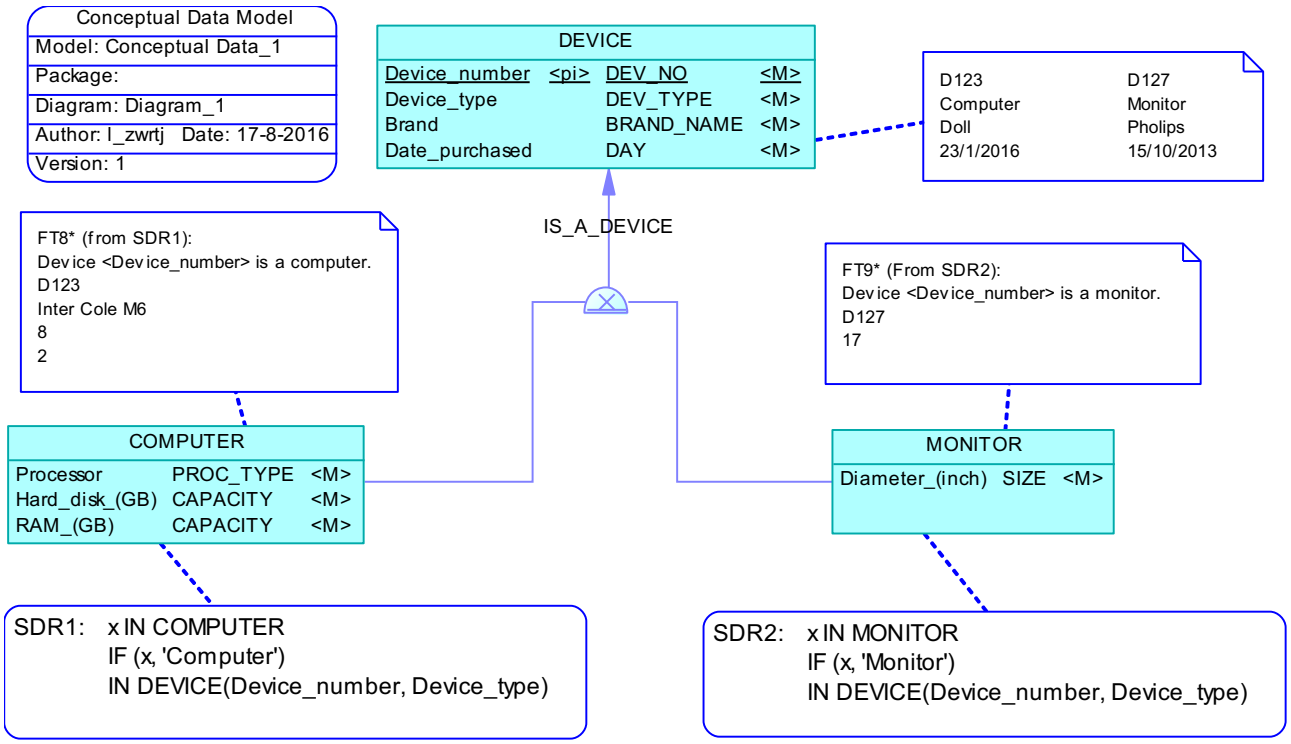


Figure A5.1.3 ERD for devices with subtypes

Answer to task e

A table for each subtype:

- IR1 and IR2 are the simple translation of SDR1 and SDR2 (see the answer to task d above). They must both be implemented manually to enforce business rules BR1 and BR2 (see the answer to task a above).
- The cardinalities of the references between the supertype and the subtypes were wrongly given as '0..*' by PowerDesigner version 16.0 and had to be manually corrected.

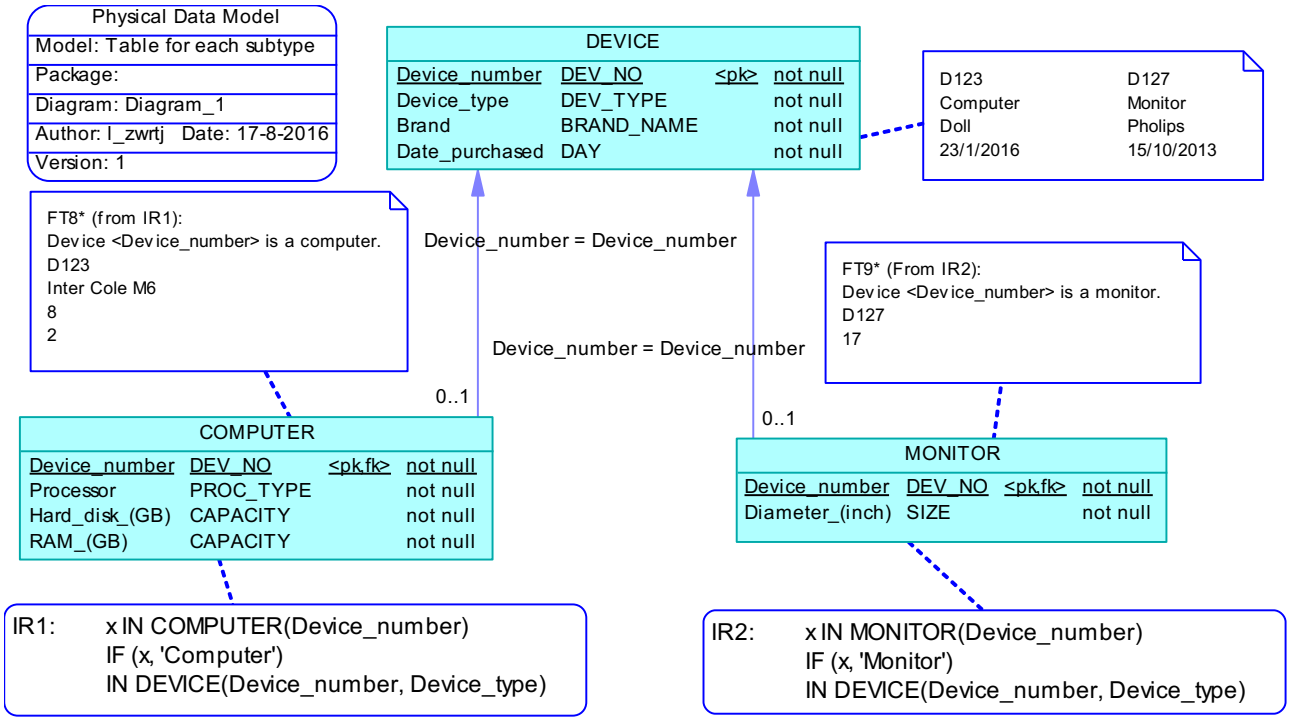


Figure A5.1.4 **PDM with a table for the supertype and for each subtype**

Only a table for the supertype:

- IR1 and IR2 are the elaborate translation of SDR1 and SDR2 (see the answer to task d above). They must both be implemented manually to enforce business rules BR1 and BR2 (see the answer to task a above). Note that the derivable fact types FT8 and FT9 are no longer needed now.

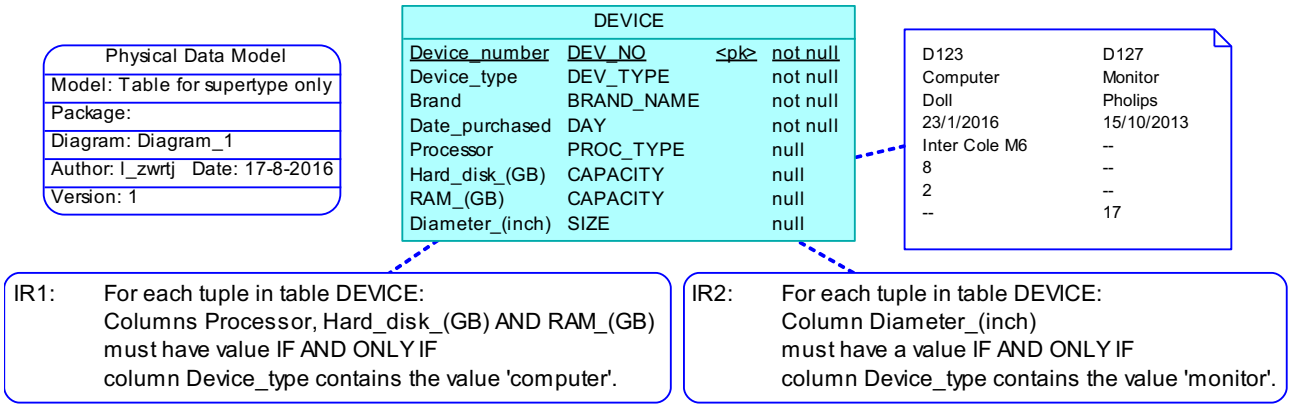


Figure A5.1.5 **PDM with a table for the supertype only**

Answer to exercise 5.2 Library

No answer is given here. Consult your teacher for feedback on your work on this exercise. The exercise contains many points for discussion, so it is worthwhile to compare your answer with answers by other students and/or your teacher.