Politecnico di Torino

Logic design

Example

Example of relational logic design

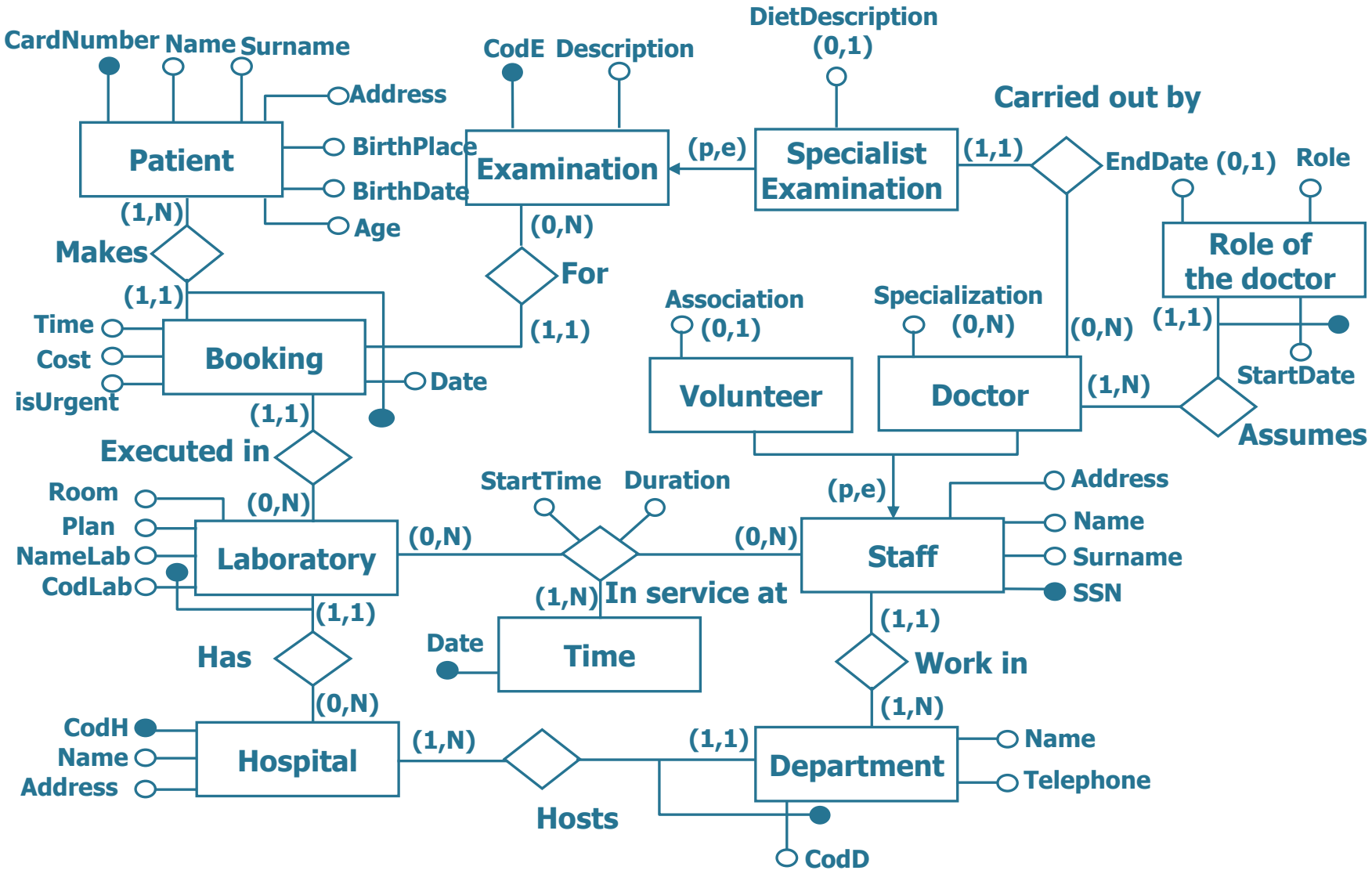
- Introduction
- ER schema restructing
- Translation of the entities without an external identifier
- Translation of the entities with an external identifier
- Translation of the relationships

Politecnico di Torino

Introduction

Example of relational logic design

Starting conceptual model

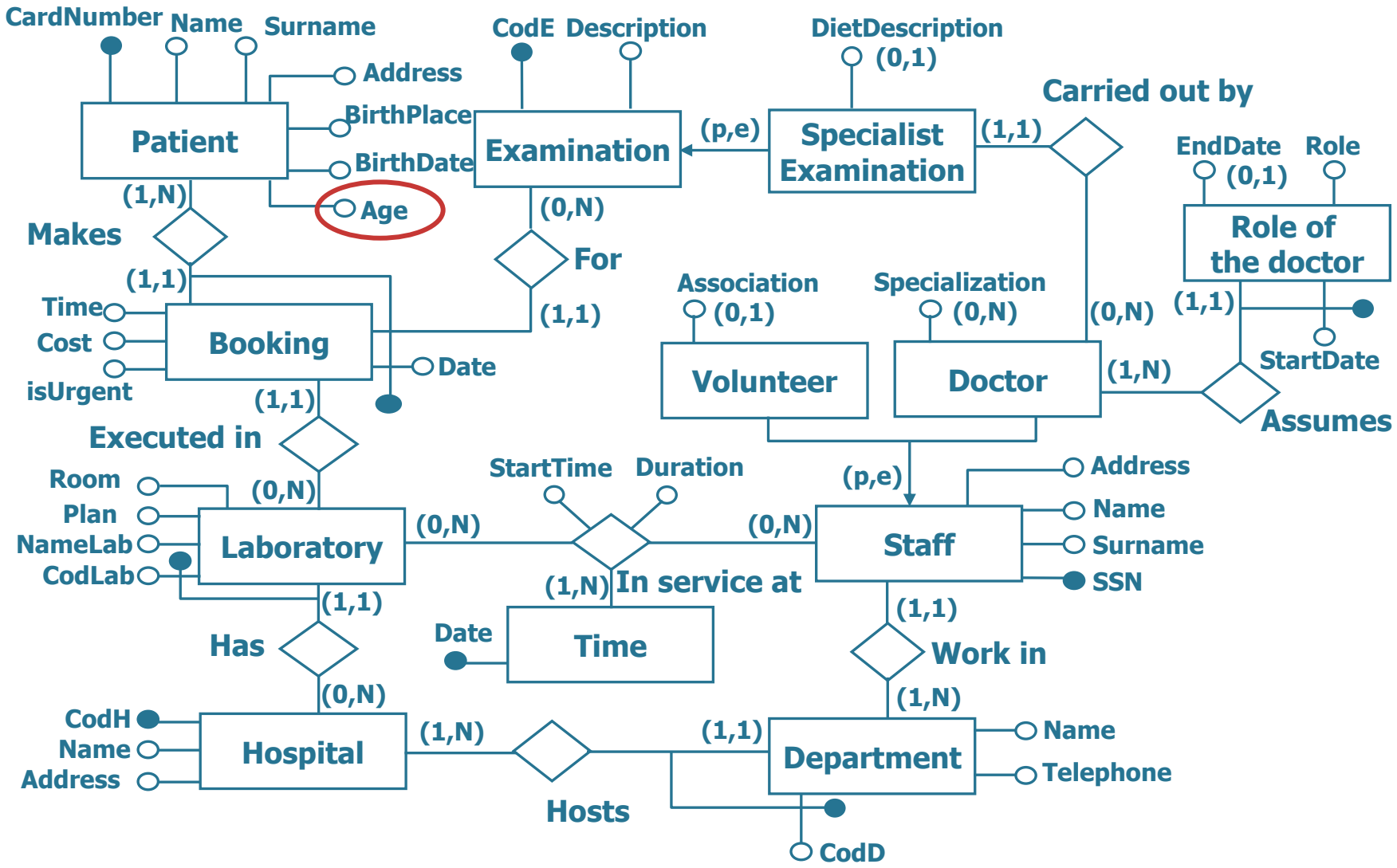


Politecnico di Torino

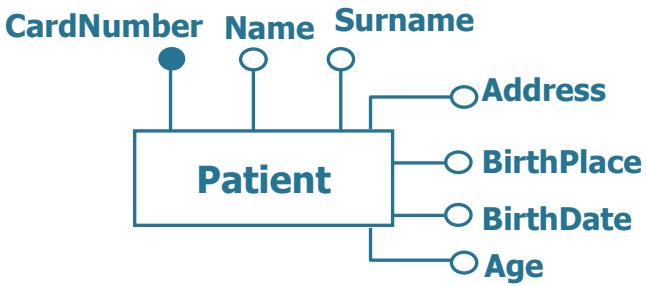
Restructing the ER model

Example of relational logic design

Analysis of the derived attributes

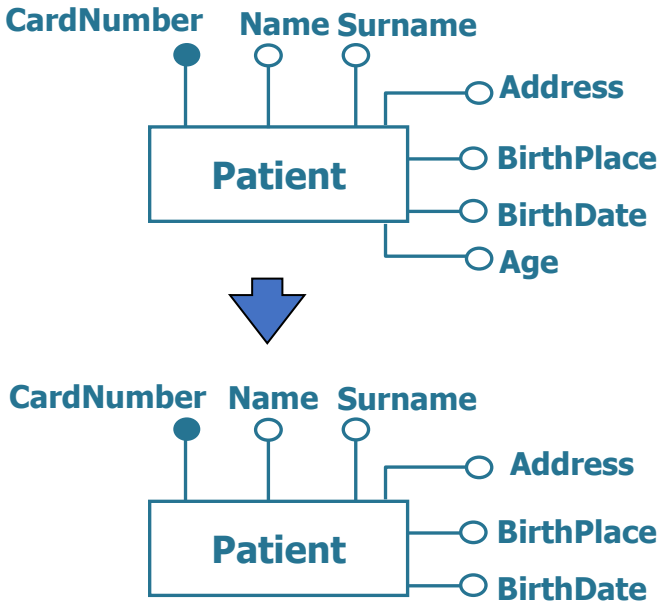


Derived Age attribute

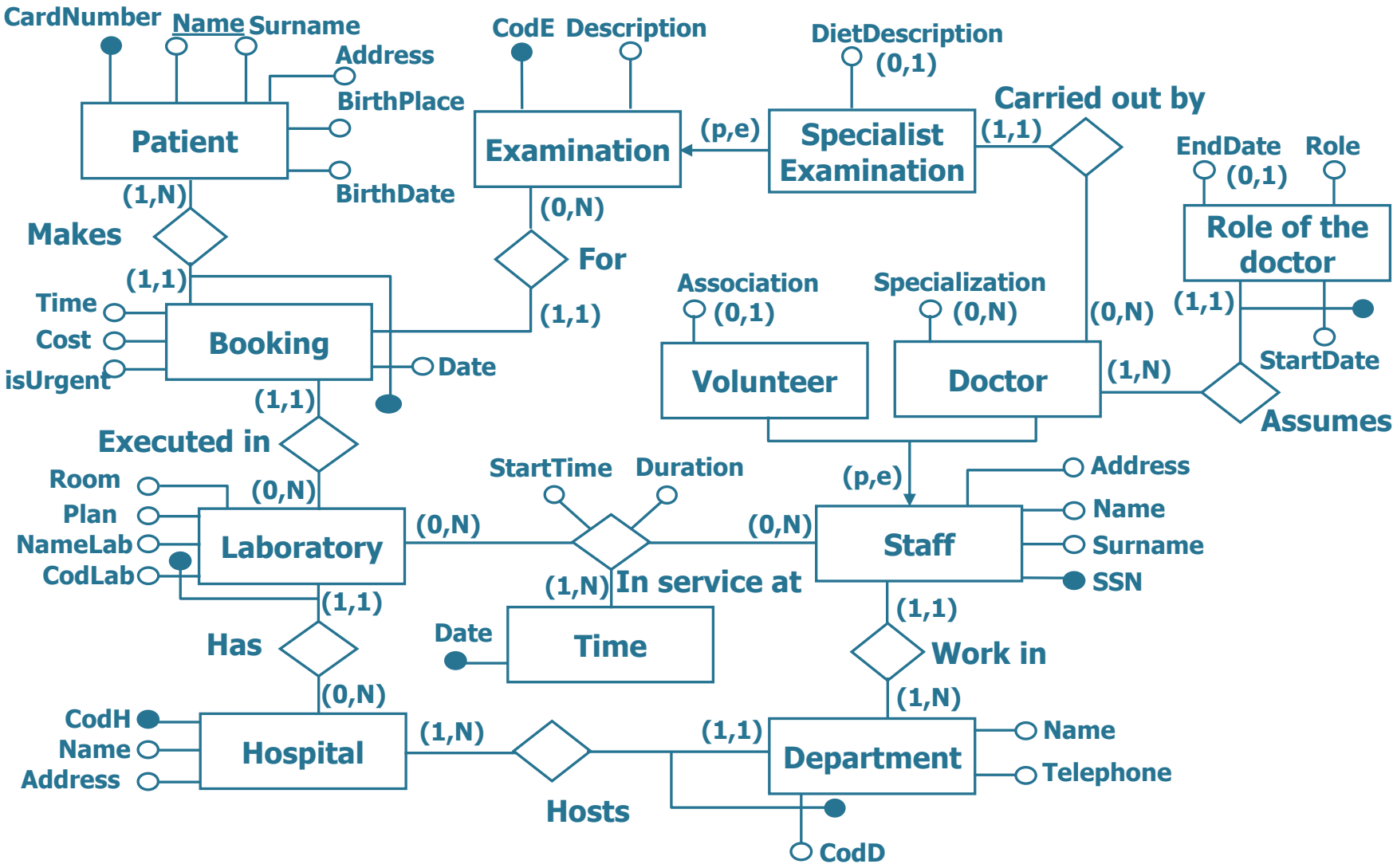


- The Age attribute can be removed since
  - it can be easily calculated from the date of birth (BirthDate)
  - it is not generally present in a query

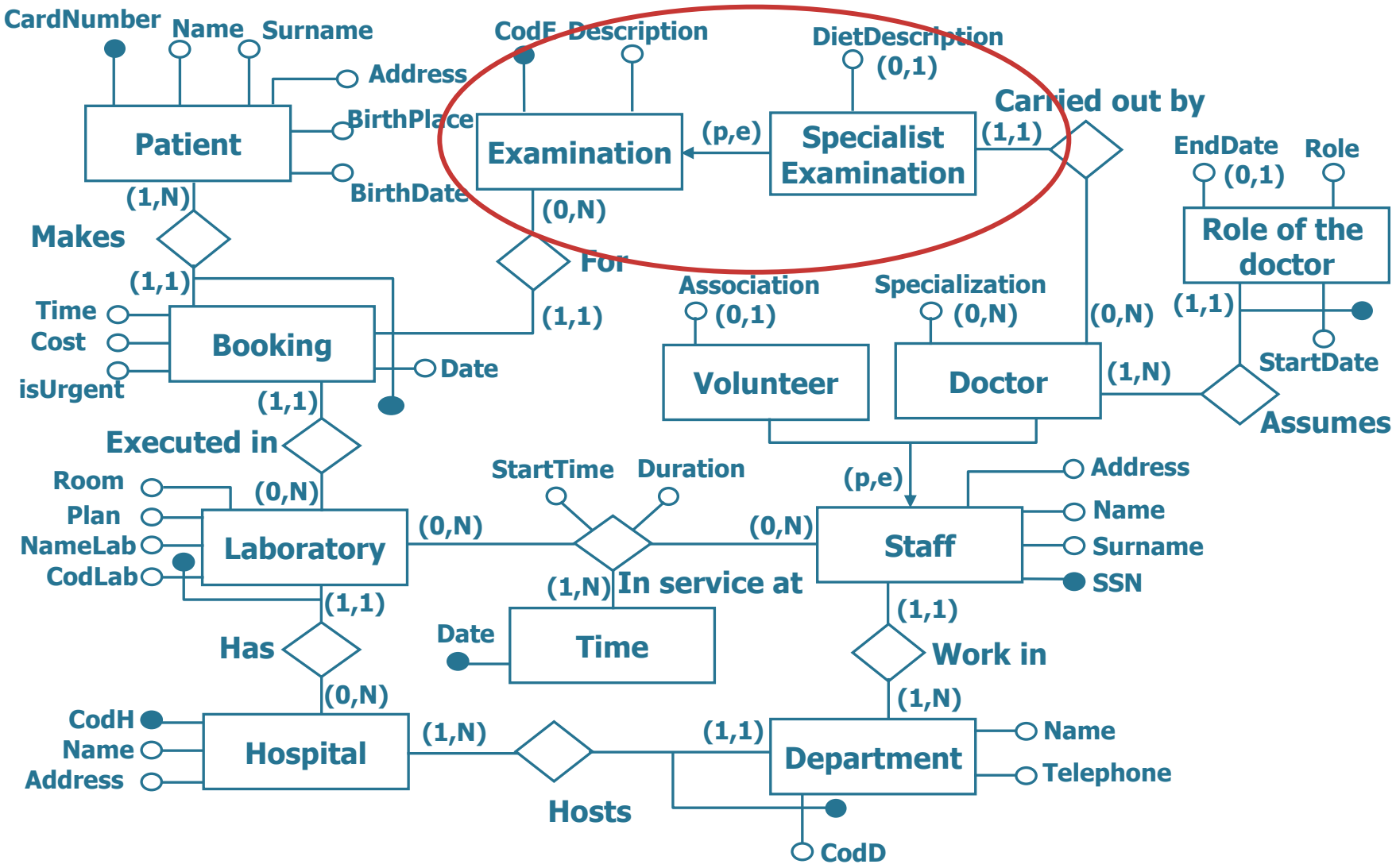
Eliminating the Age attribute



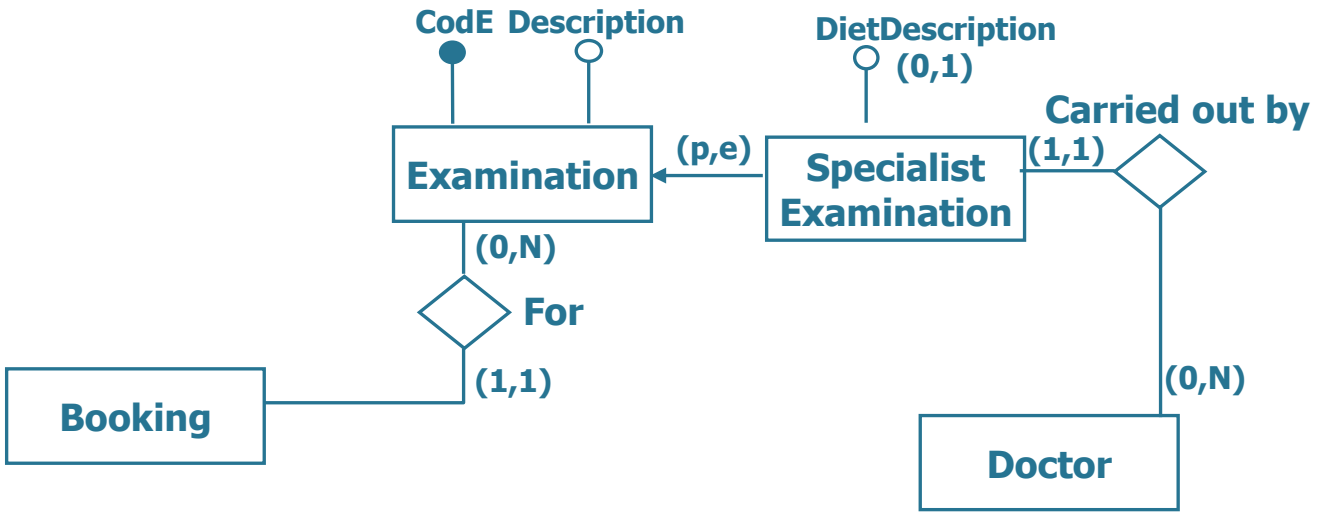
Restructured schema (n. 1)



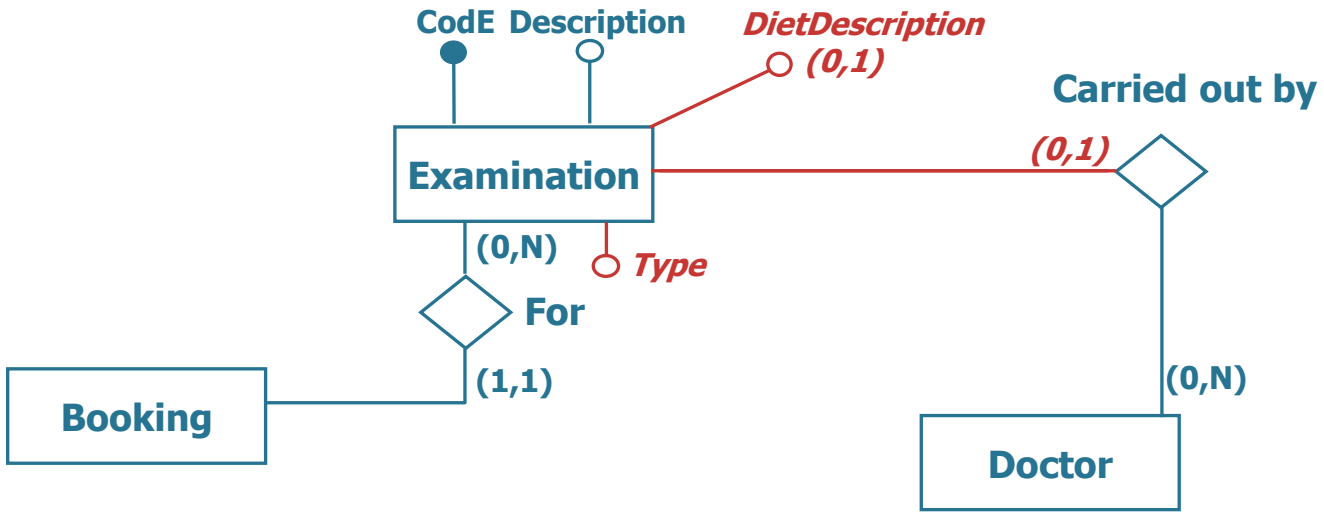
Removing generalizations



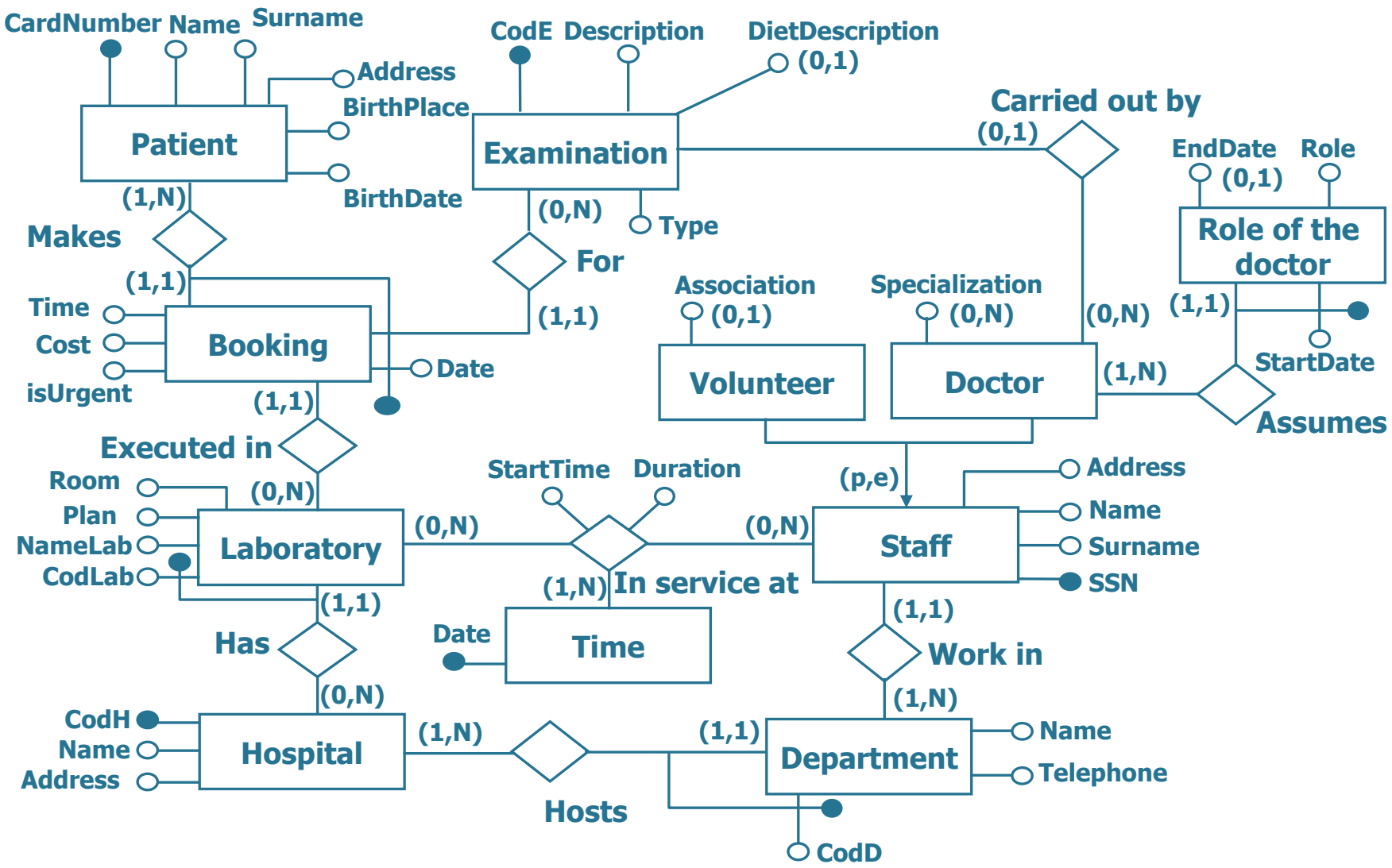
Removing generalizations



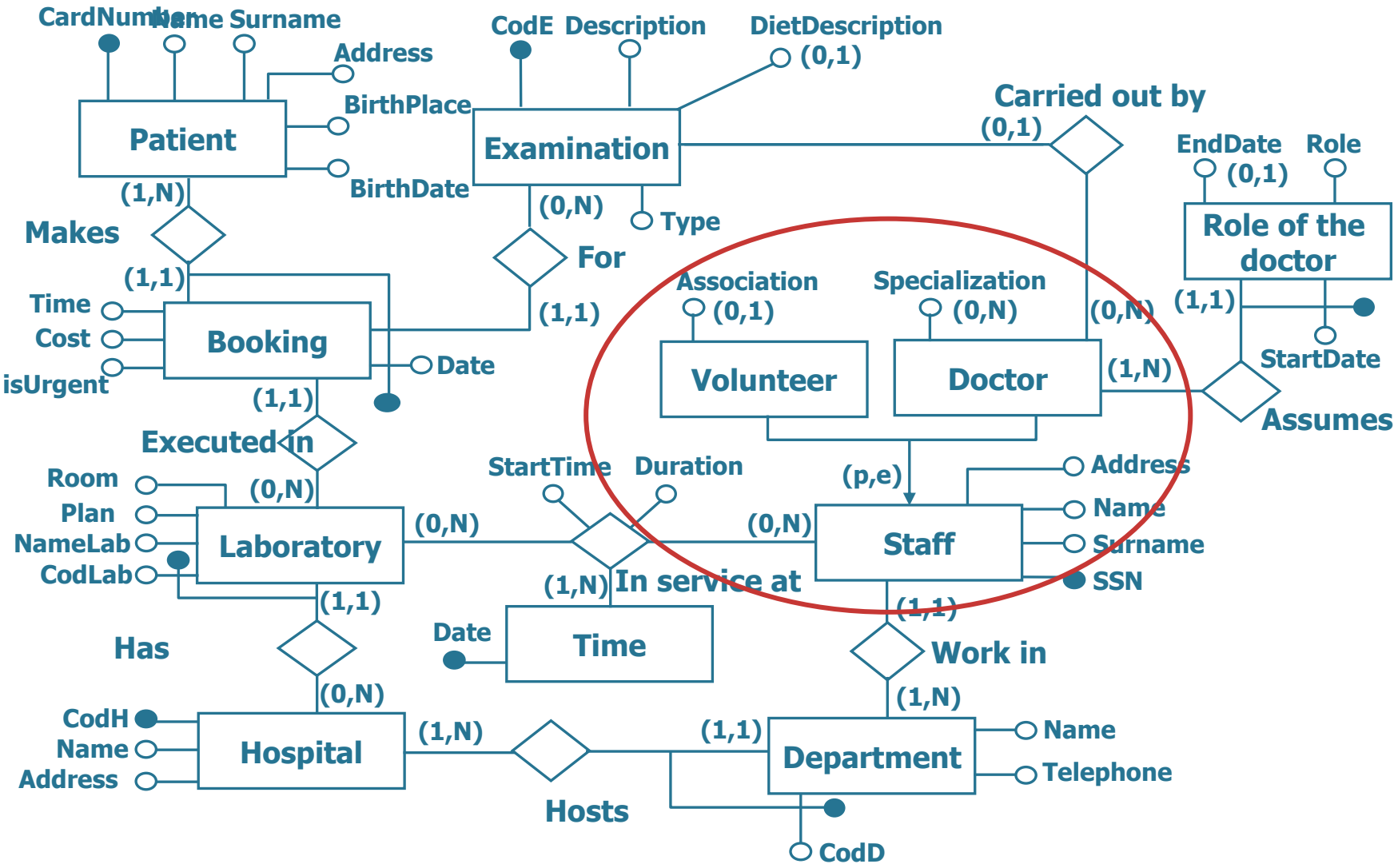
Merge into the parent entity



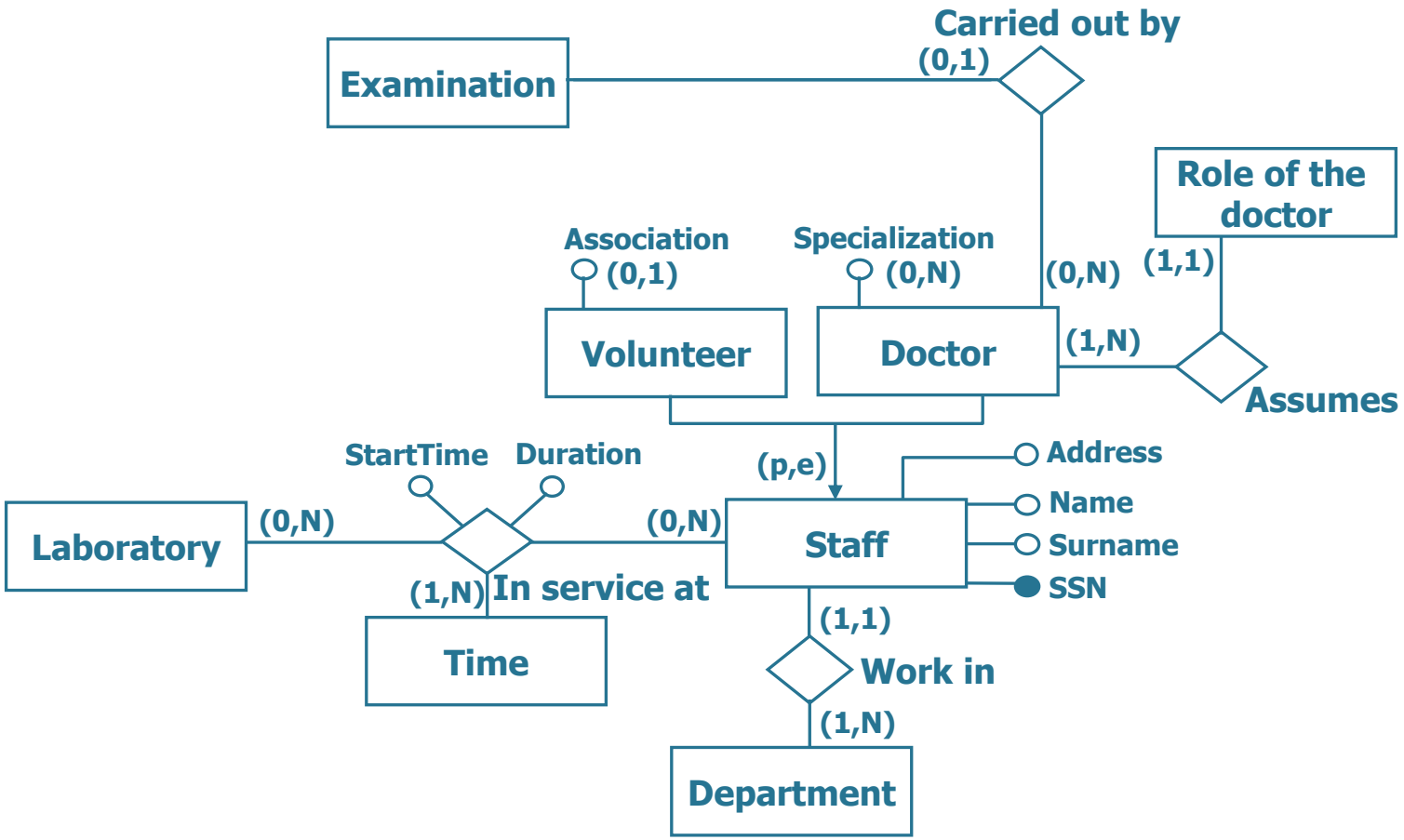
Restructured schema (n.2)



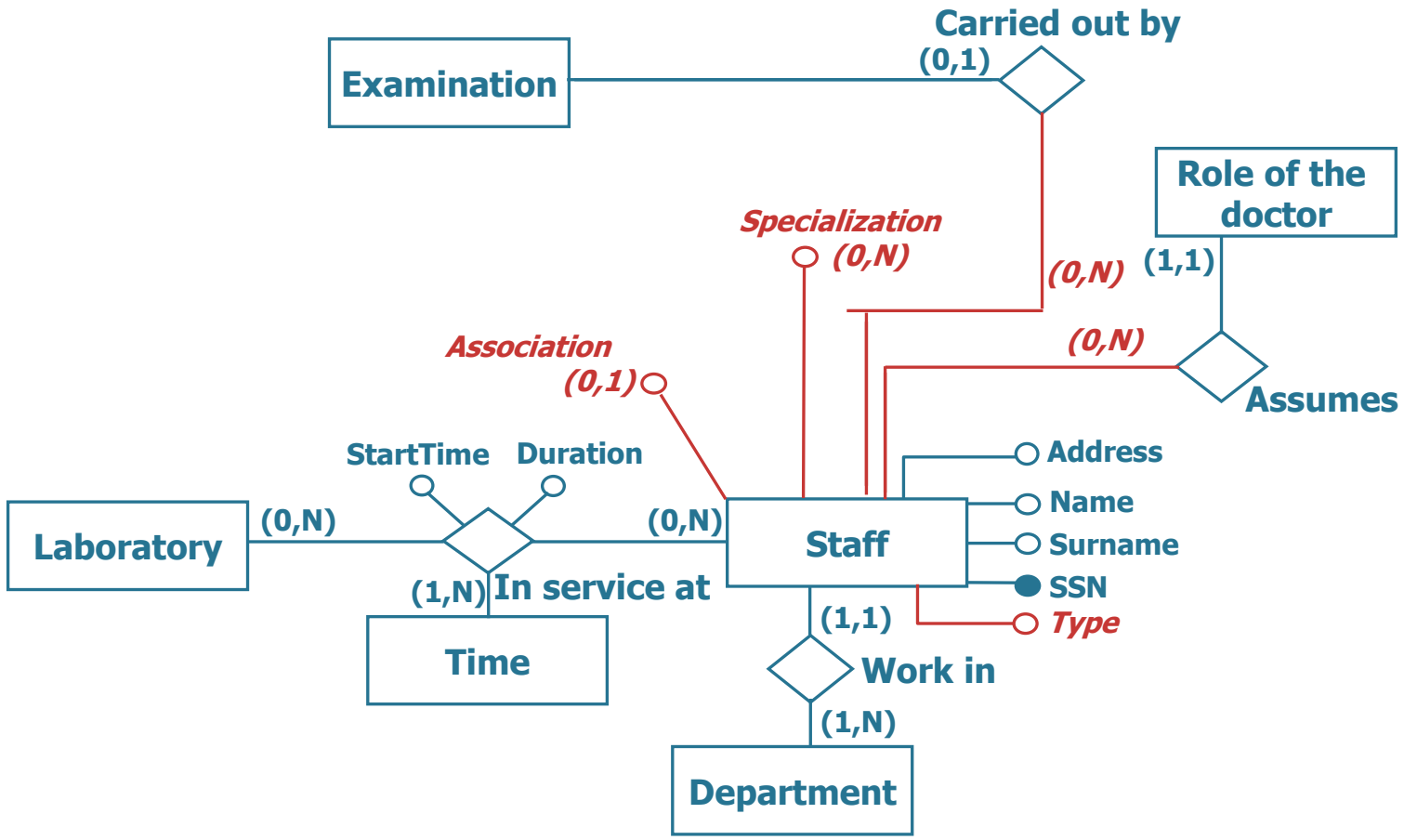
Removing generalizations



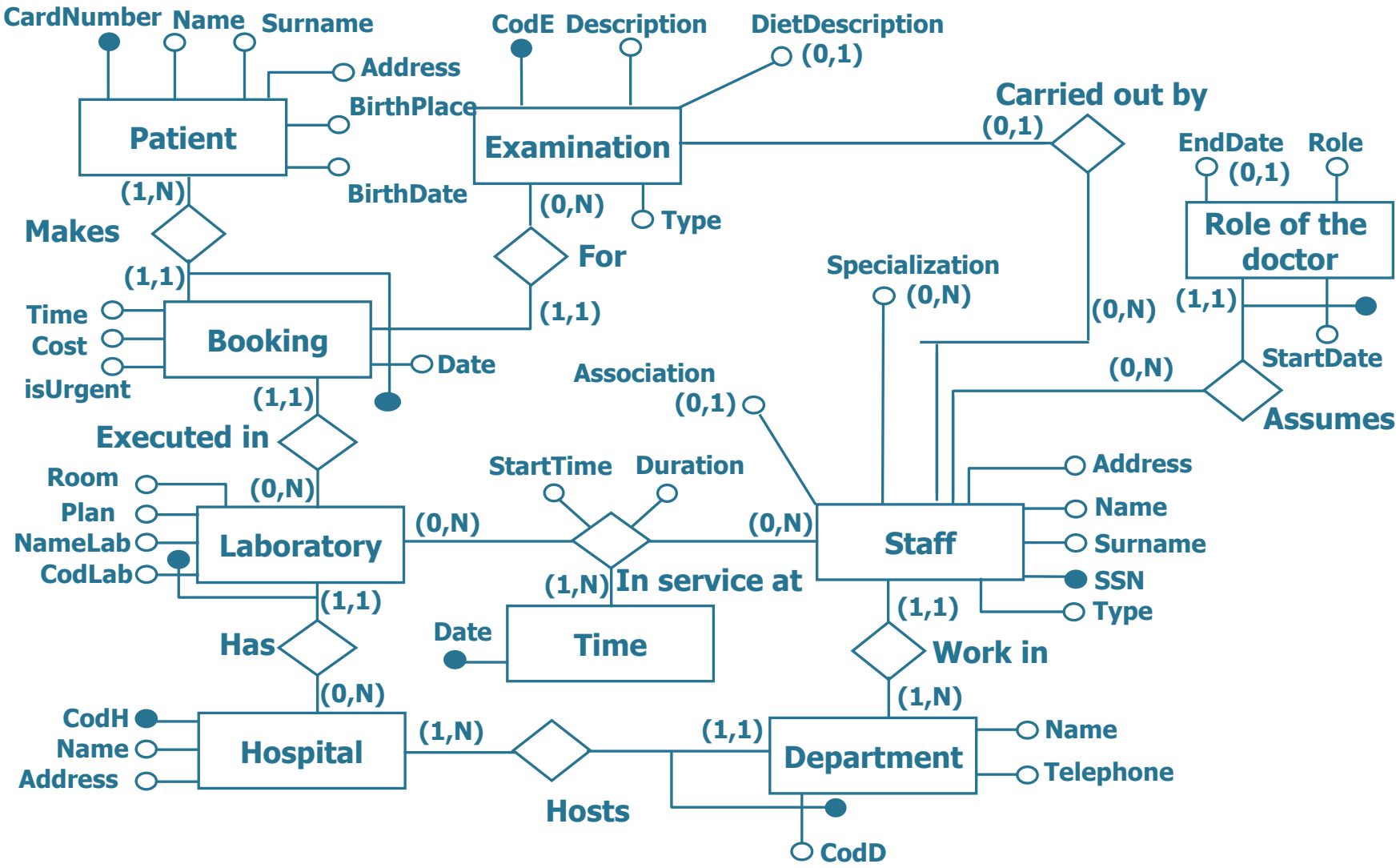
Staff generalization



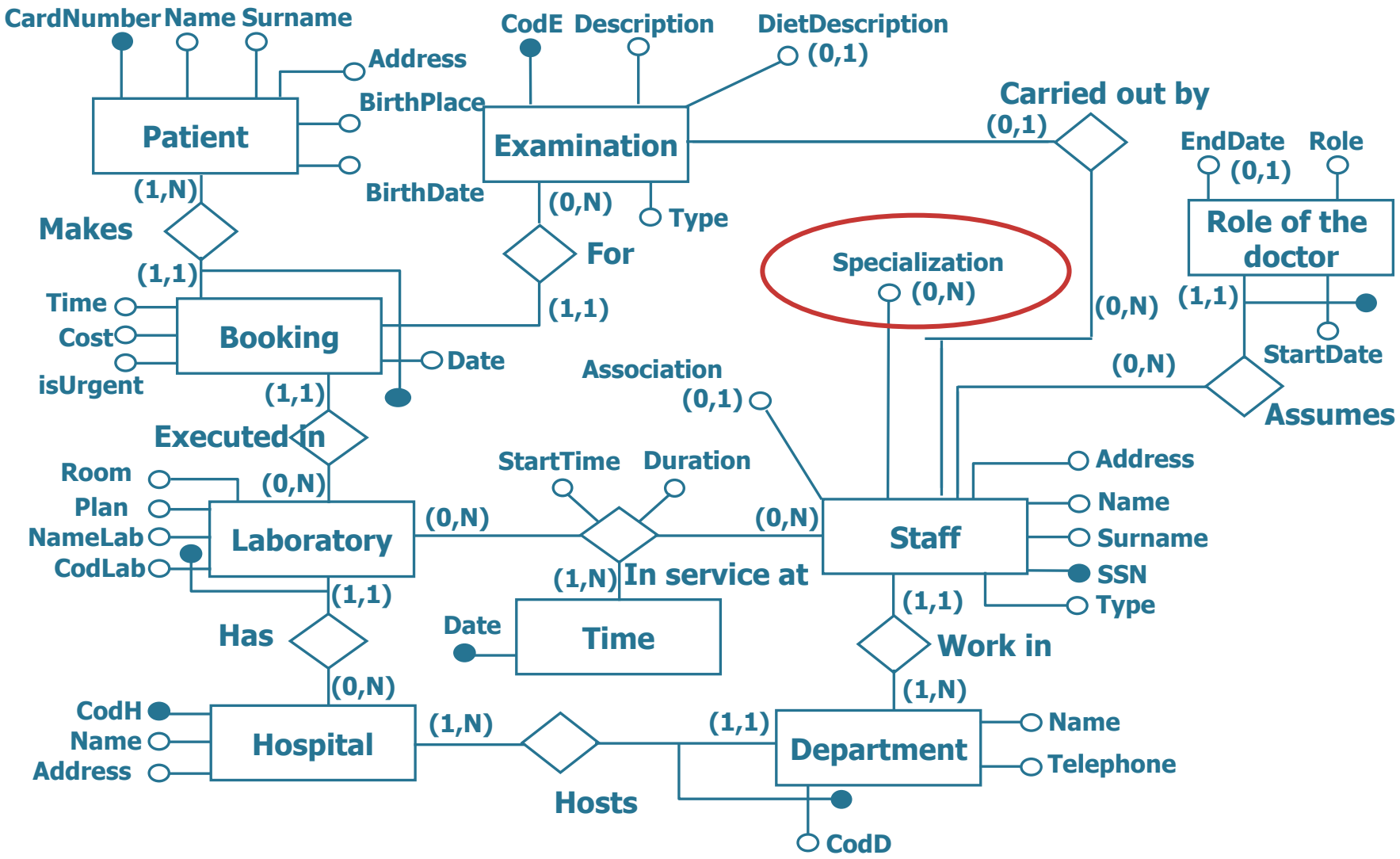
Merge into the parent entity



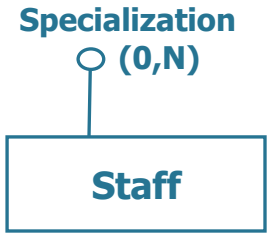
Restructured schema (n.3)



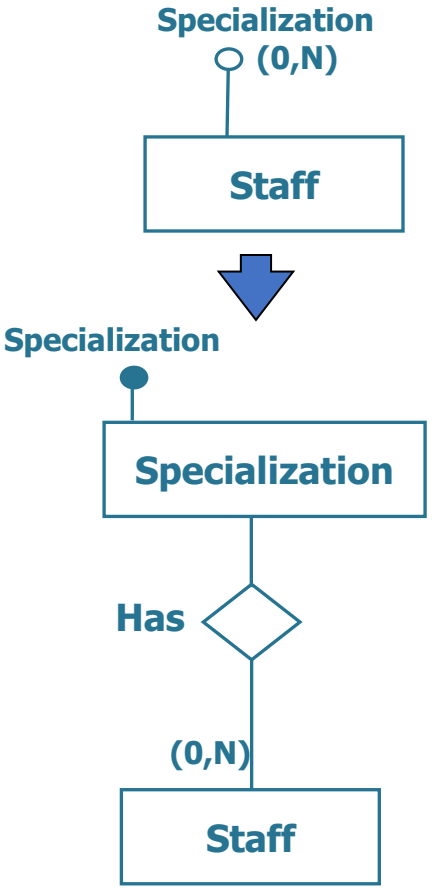
Removing multivalued attributes



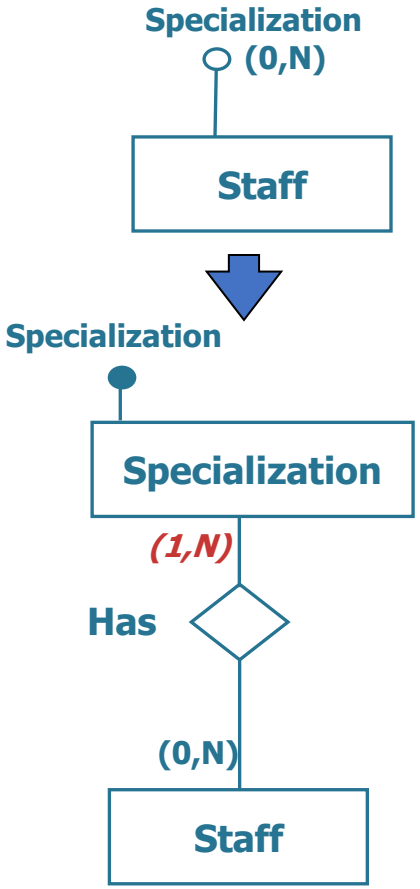
Removing multivalued Spacialization attribute



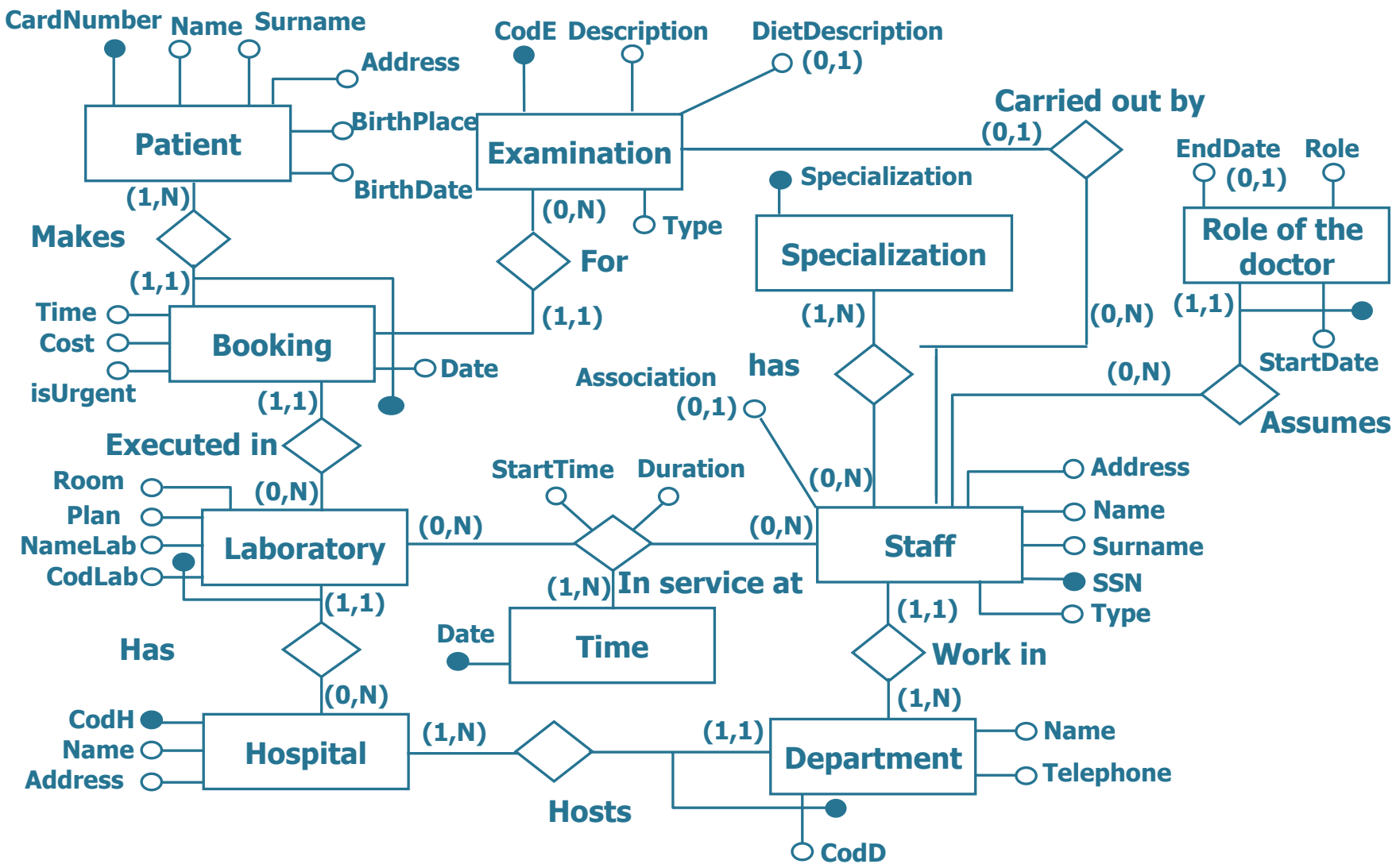
New specialization entity



Cardinality of the *Has* relationship



Restructured schema (final)

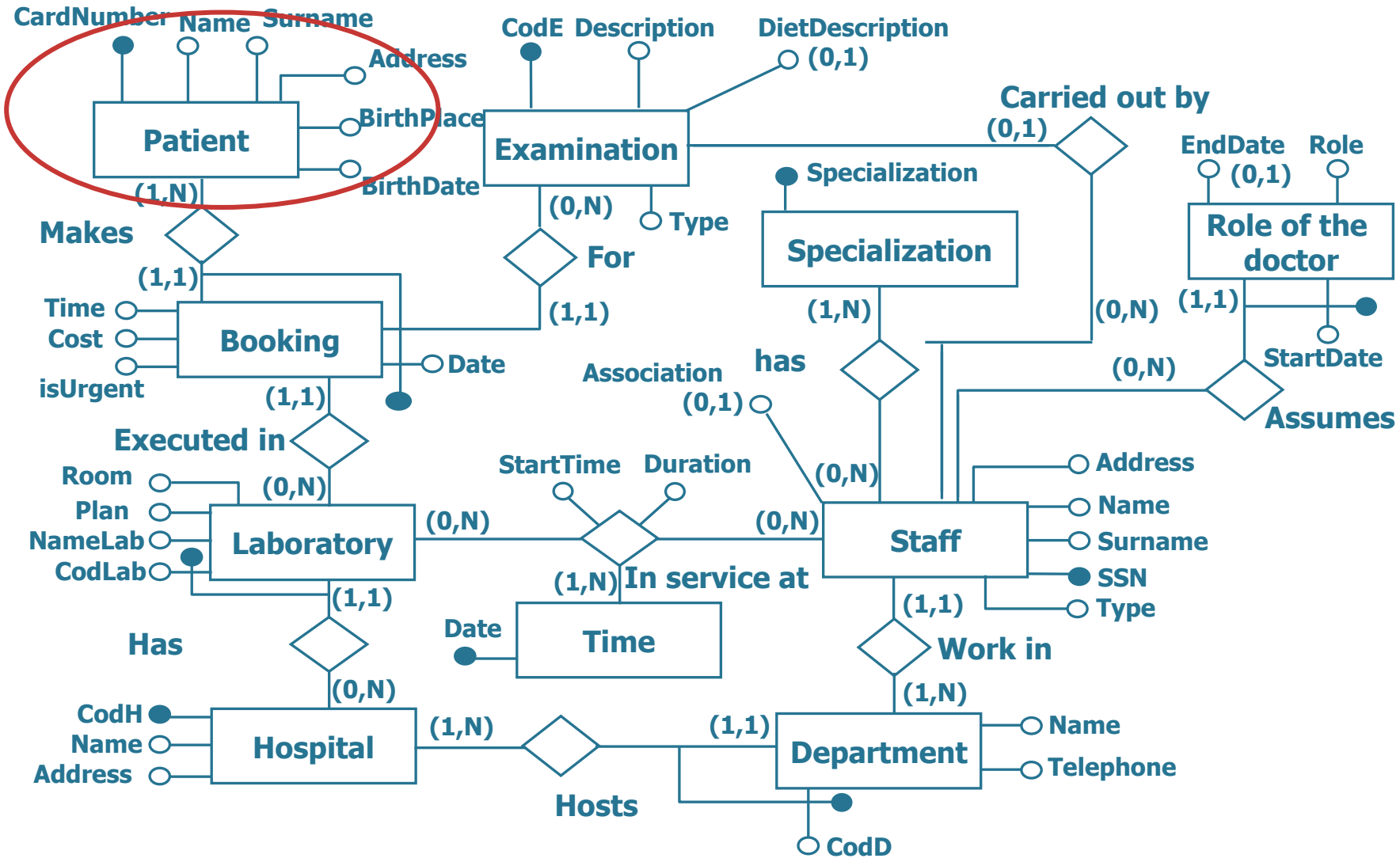


Politecnico di Torino

Traslating entities without an external identifier

Example of relational logic design

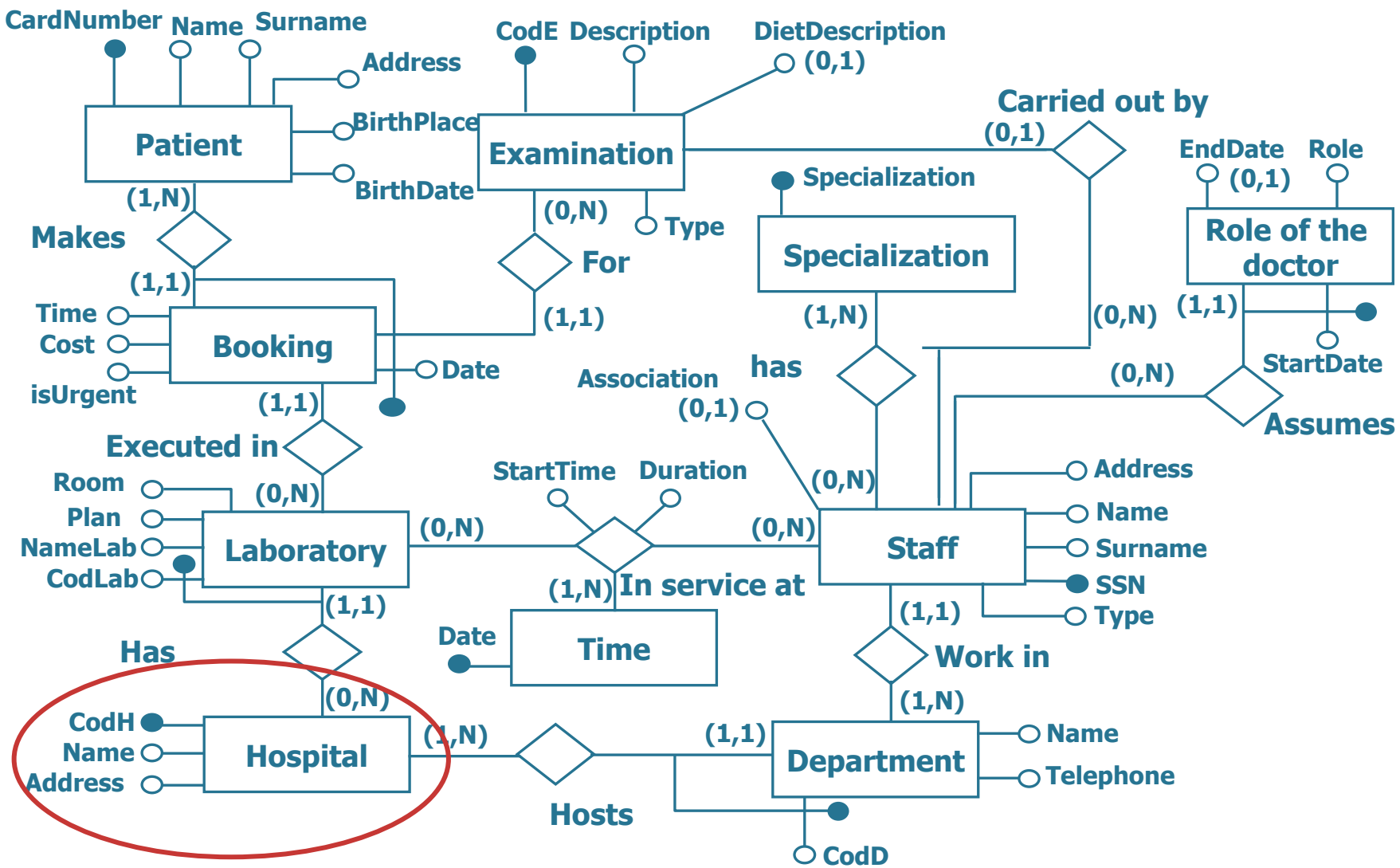
Translation of the Patient entity



Translation of the Patient entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Translation of the Hospital entity

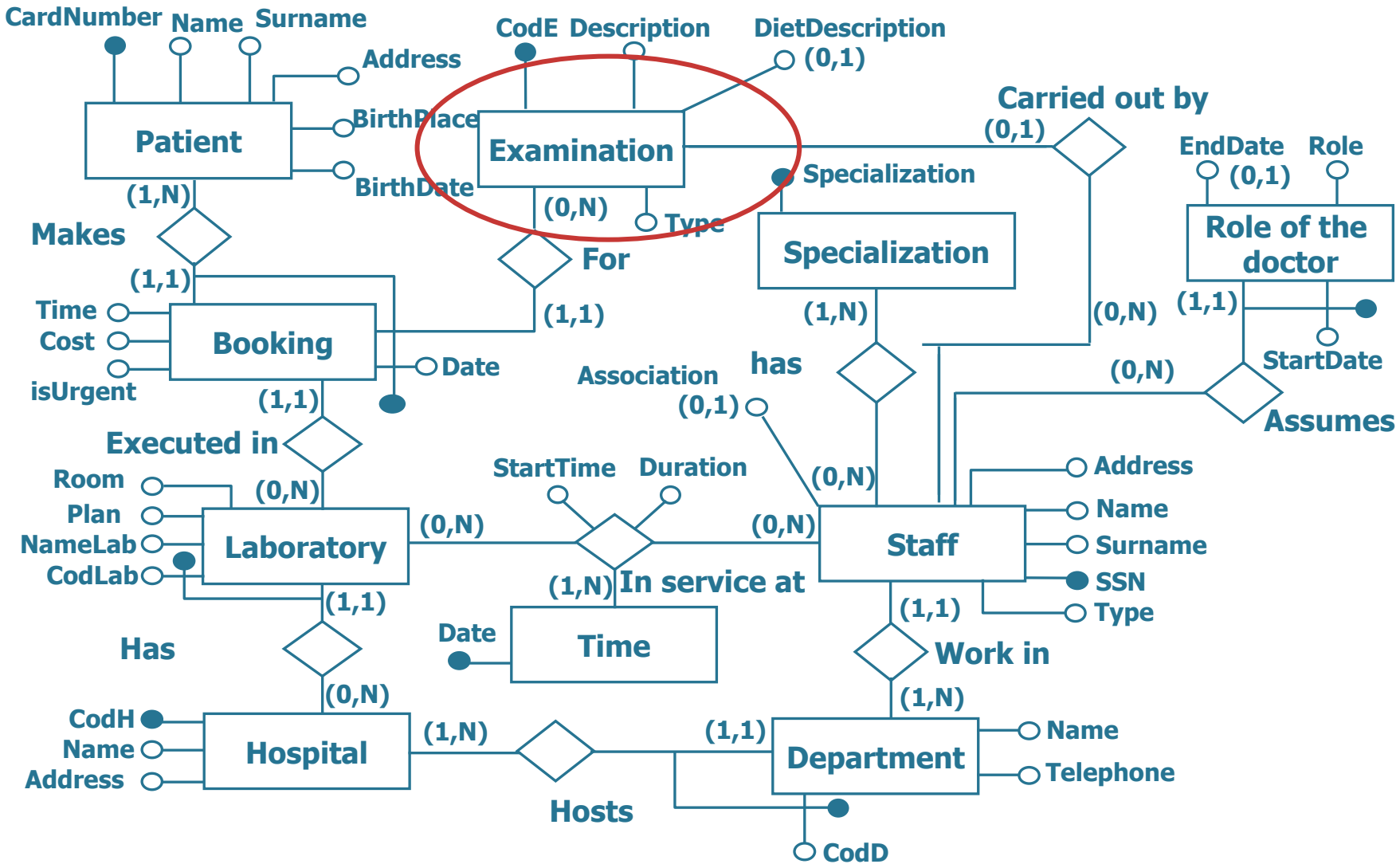


Translation of the Hospital entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Translation of the Examination entity



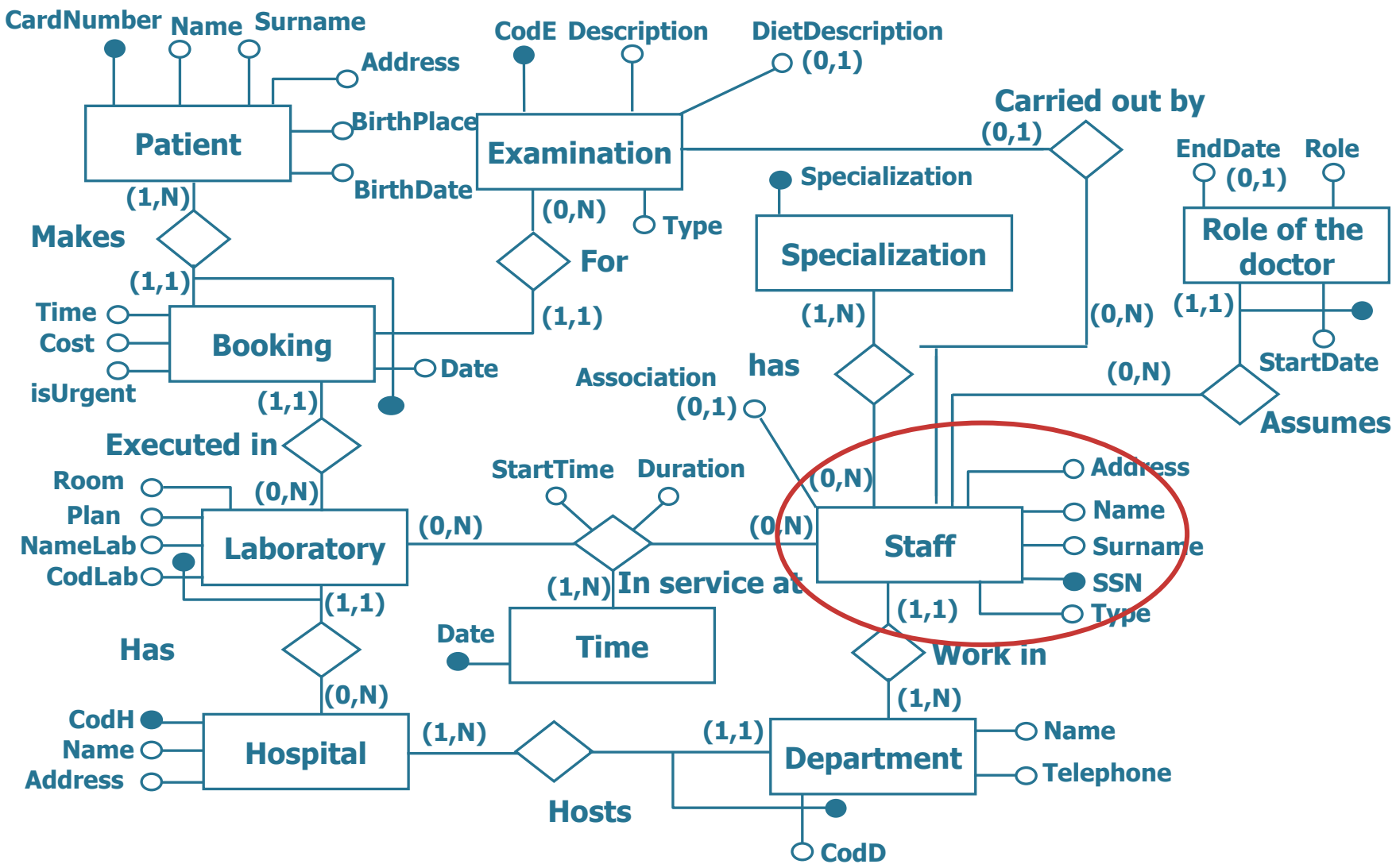
Translation of the Examination entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription, Type)*

Translation of the Staff entity



Translation of the Staff entity

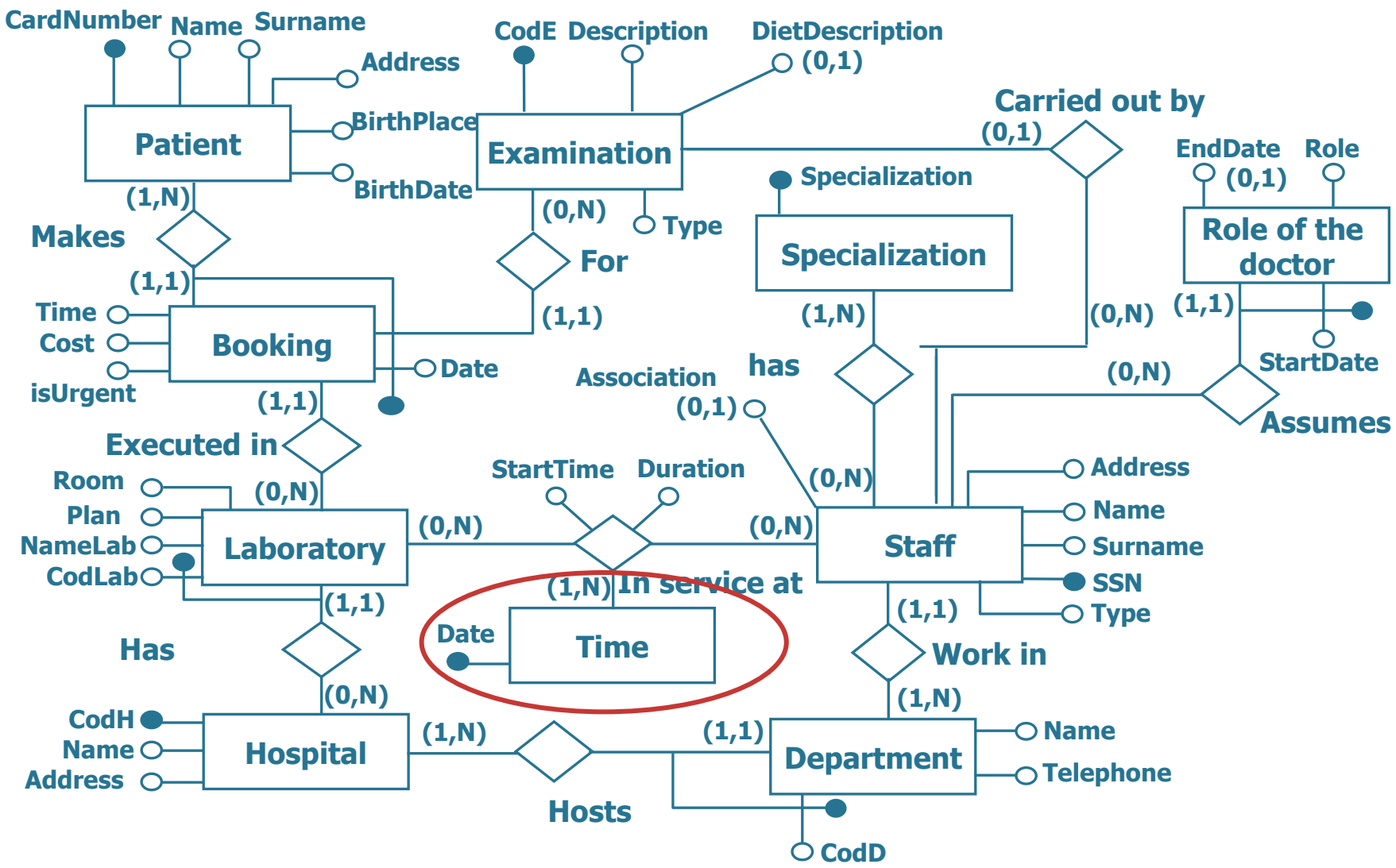
Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association,Type)*

Translation of the Time entity



Translation of the Time entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

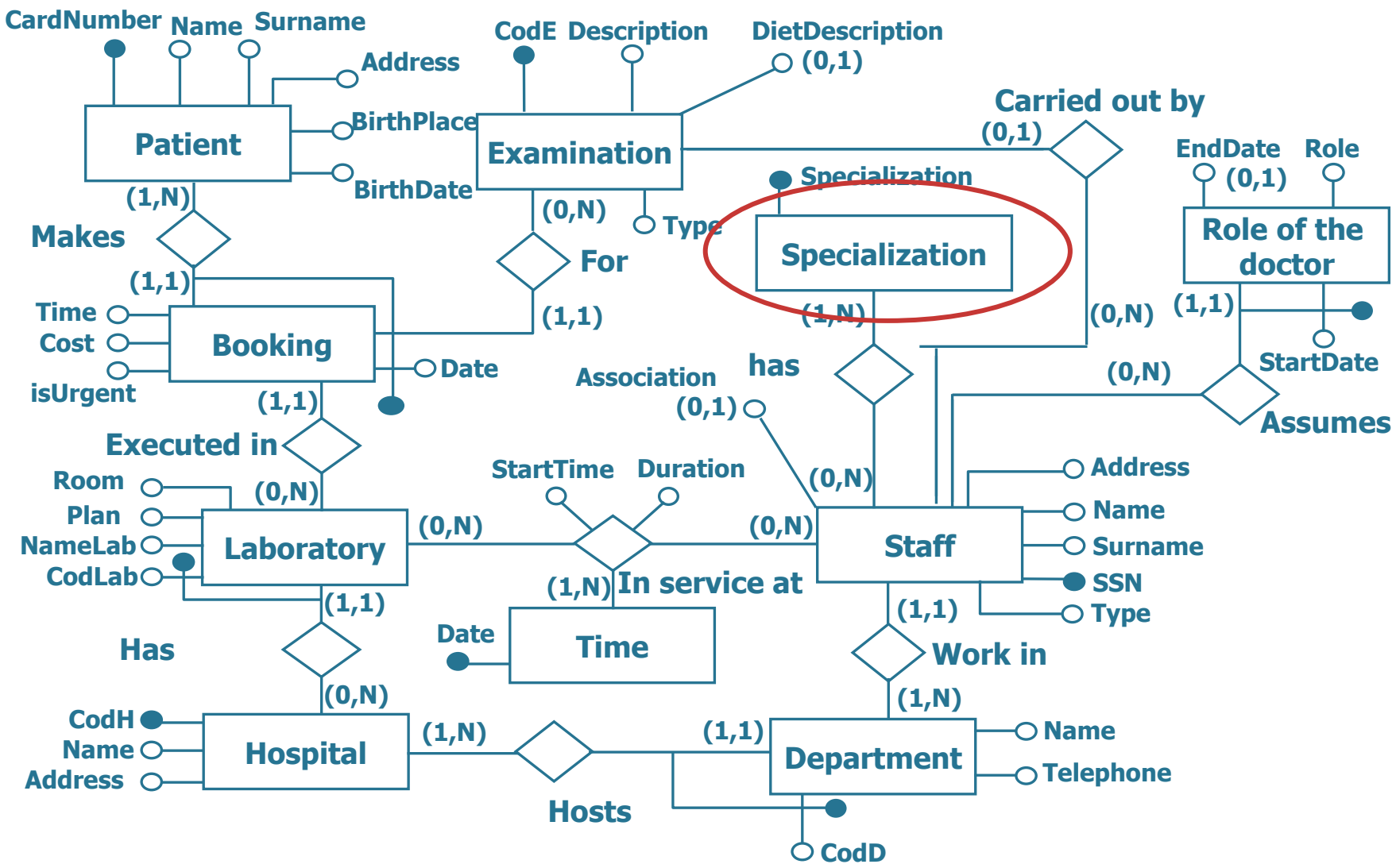
Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association*,Type)

Time(Date)

Translation of the Specialization entity



Translation of the Specialization entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association*,Type)

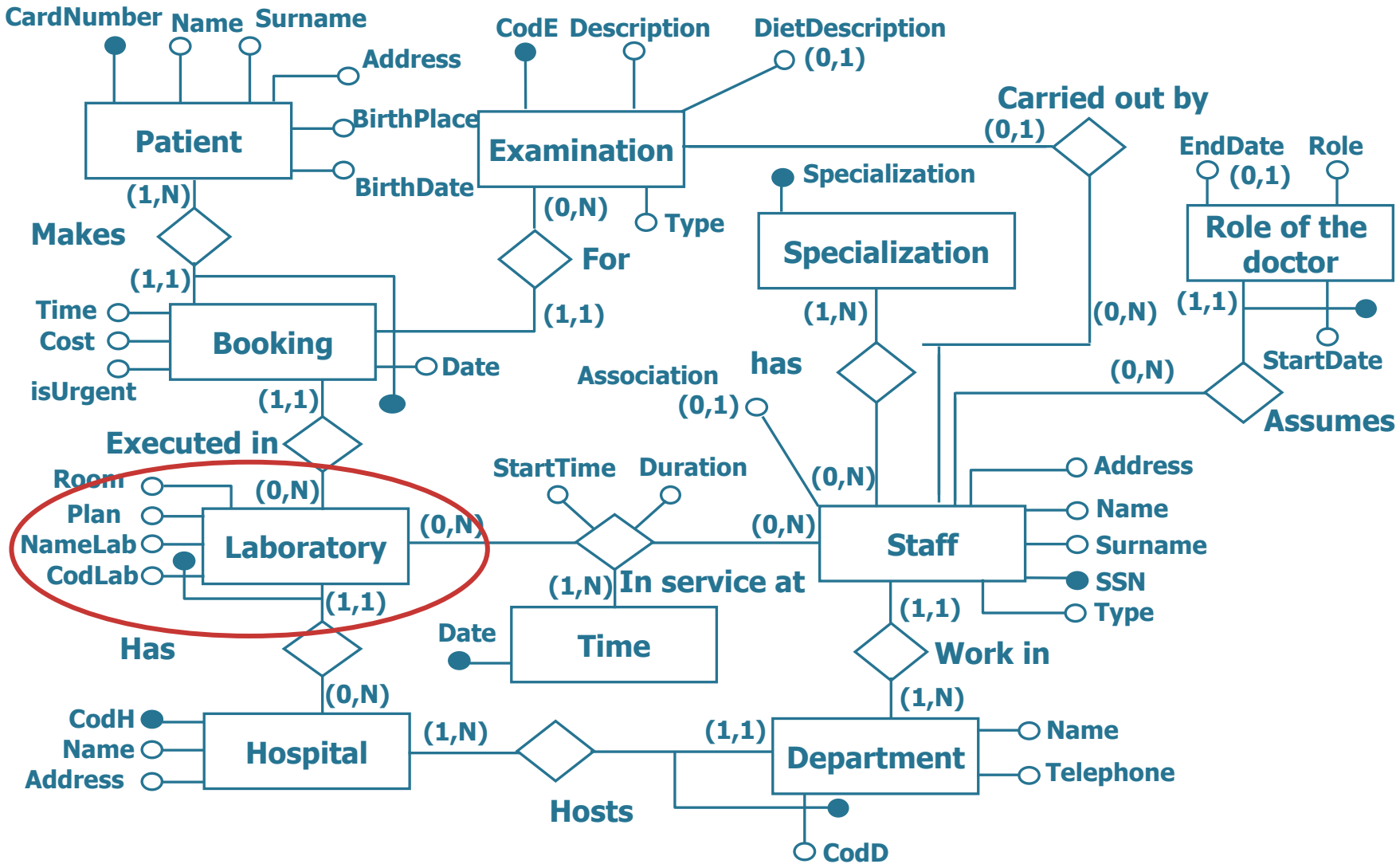
Time(Date)

Specialization(Specialization)

Translating entities with an external identifier

Example of relational logic design

Translation of the Laboratory entity



Translation of the Laboratory entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

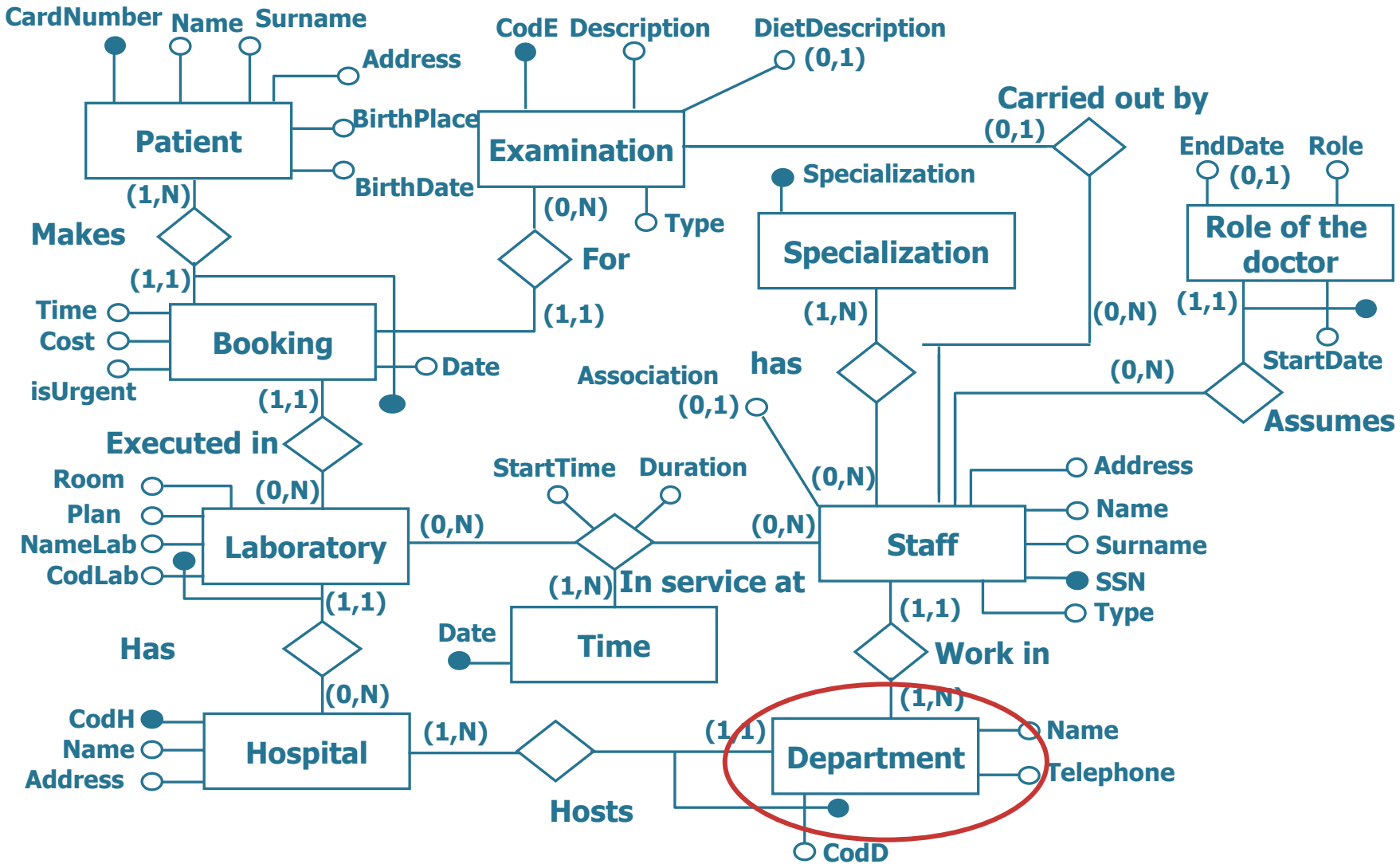
Staff(SSN, Name, Surname, Address, Association*,Type)

Time(Date)

Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Translation of the Department entity



Translation of the Department entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association*,Type)

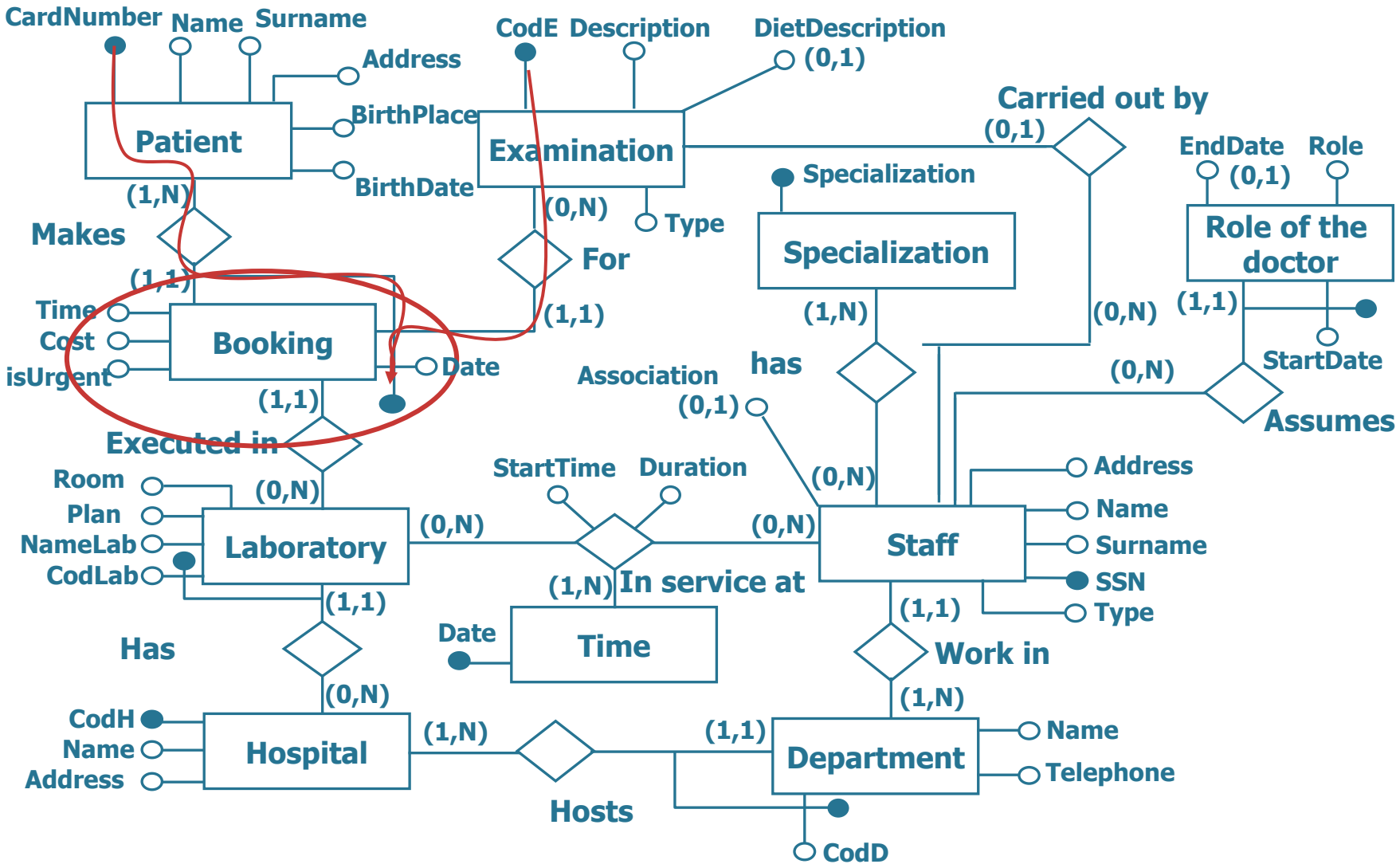
Time(Date)

Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Translation of the Booking entity



Translation of the Booking entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association*,Type)

Time(Date)

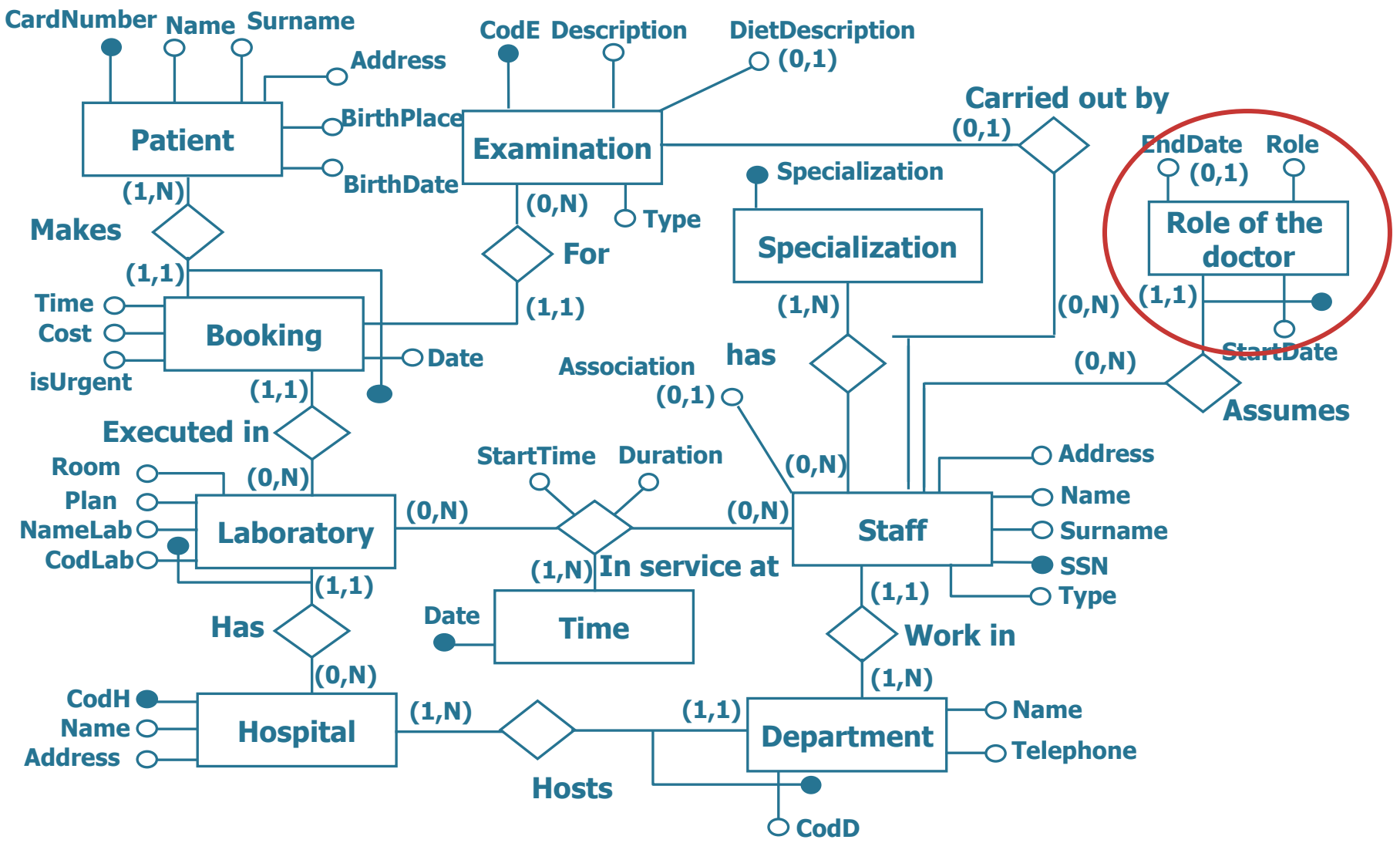
Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent)

Translation of the Role of the doctor entity



Translation of the Role of the doctor entity

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type)

Staff(SSN, Name, Surname, Address, Association*,Type)

Time(Date)

Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

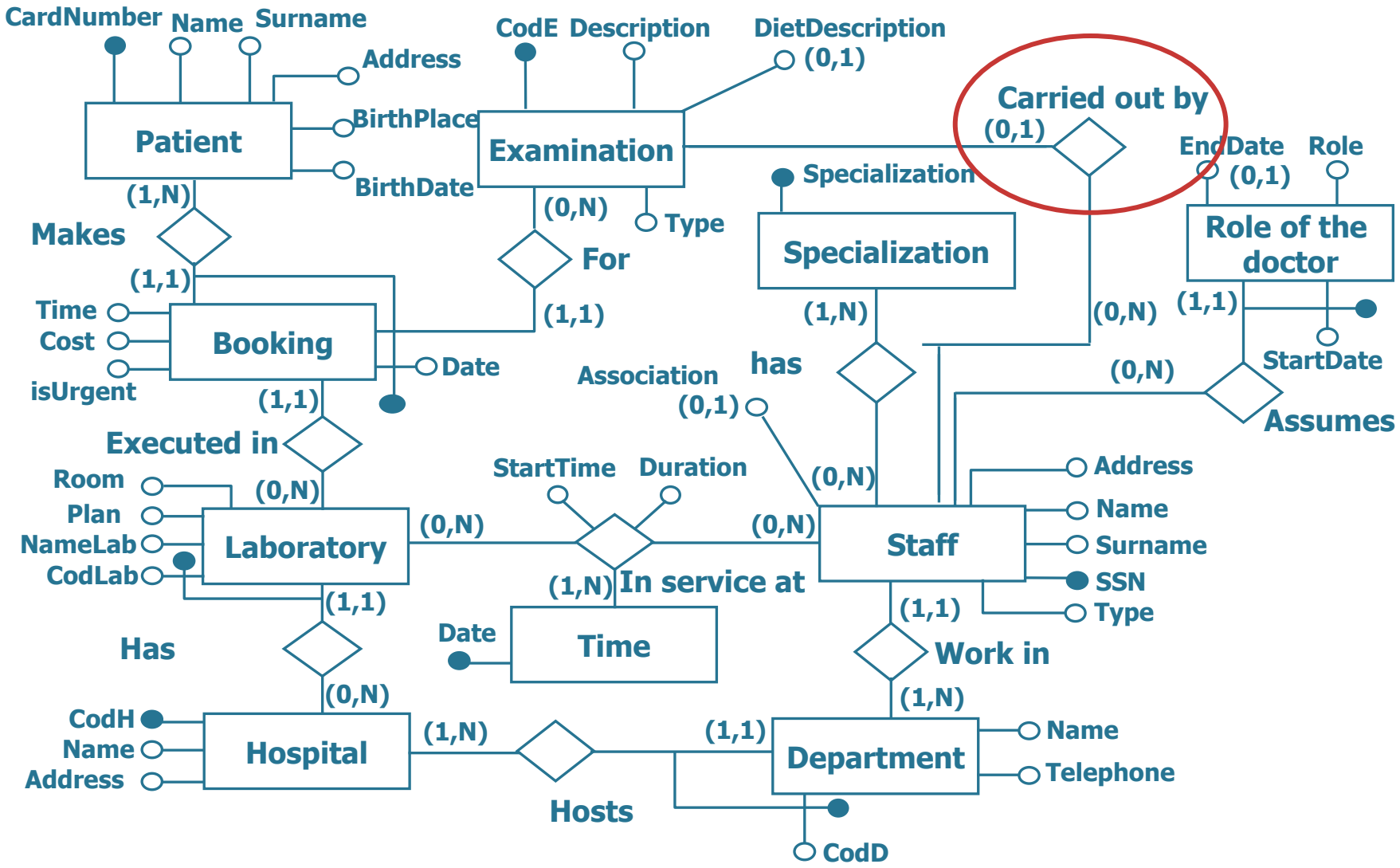
Booking(CardNumber, CodE, Date, Time, Cost, isUrgent)

DoctorRole(SSN, StartDate, EndDate, Role)*

Translating relationships

Example of relational logic design

Binary one-to-many *Carried out by* relationship



Binary one-to-many *Carried out by* relationship

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, *SSN**)

Staff(SSN, Name, Surname, Address, Association*,Type)

Time(Date)

Specialization(Specialization)

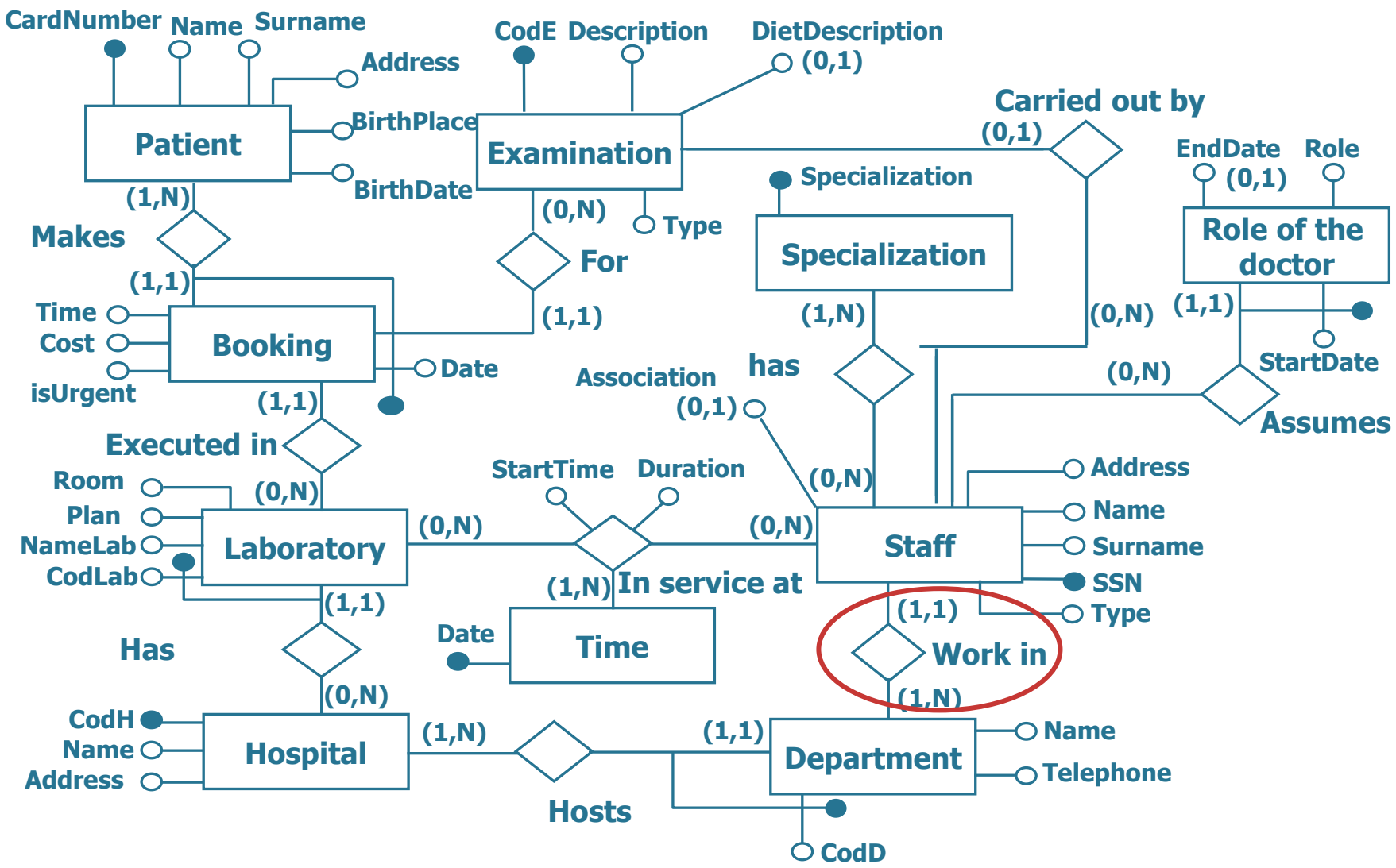
Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent)

DoctorRole(SSN, StartDate, EndDate*, Role)

Binary one-to-many *Work in* relationship



Binary one-to-many *Work in* relationship

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type, *CodD, CodH*)

Time(Date)

Specialization(Specialization)

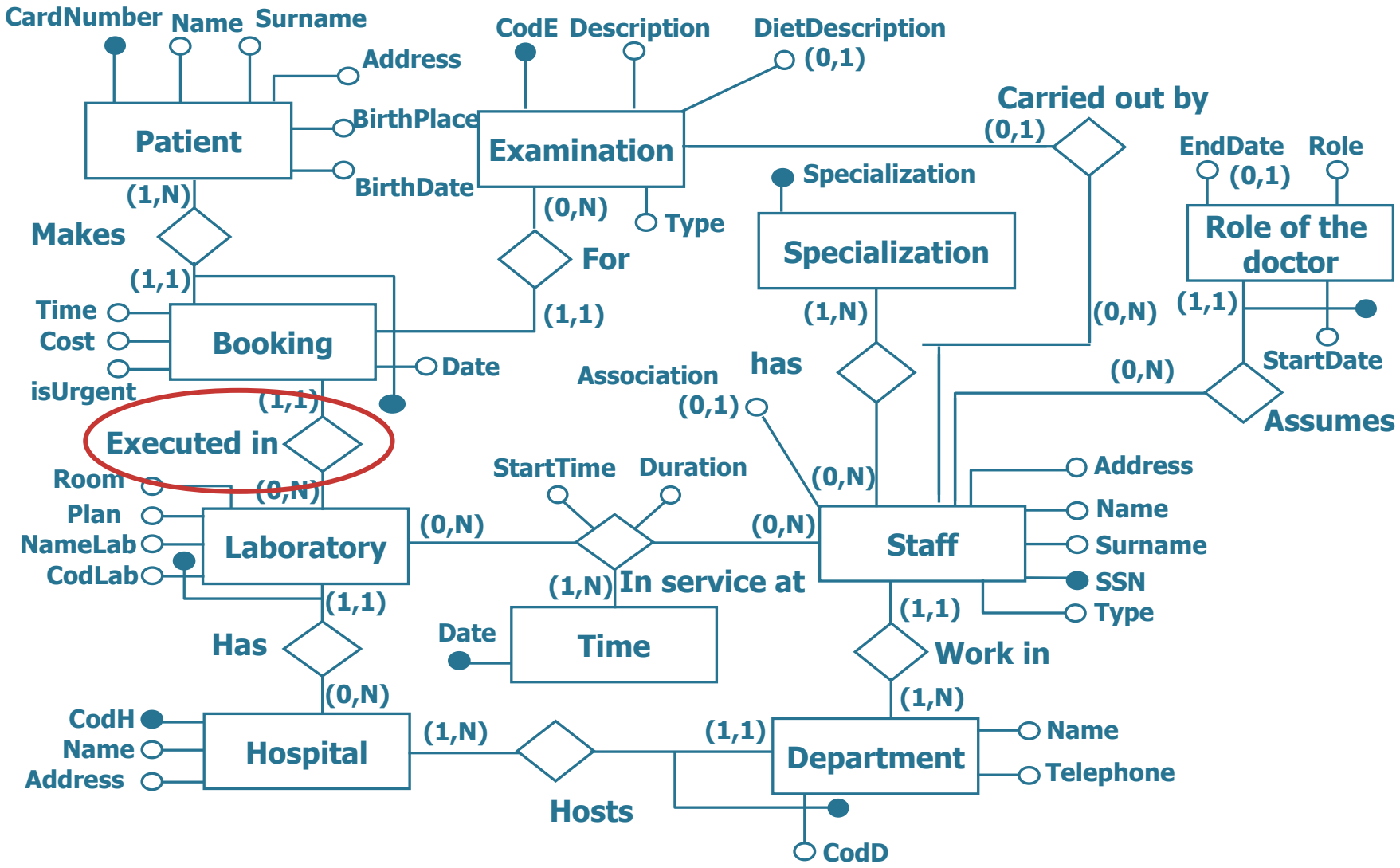
Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent)

DoctorRole(SSN, StartDate, EndDate*, Role)

Binary one-to-many *Executed in* relationship



Binary one-to-many *Executed in* relationship

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

Time(Date)

Specialization(Specialization)

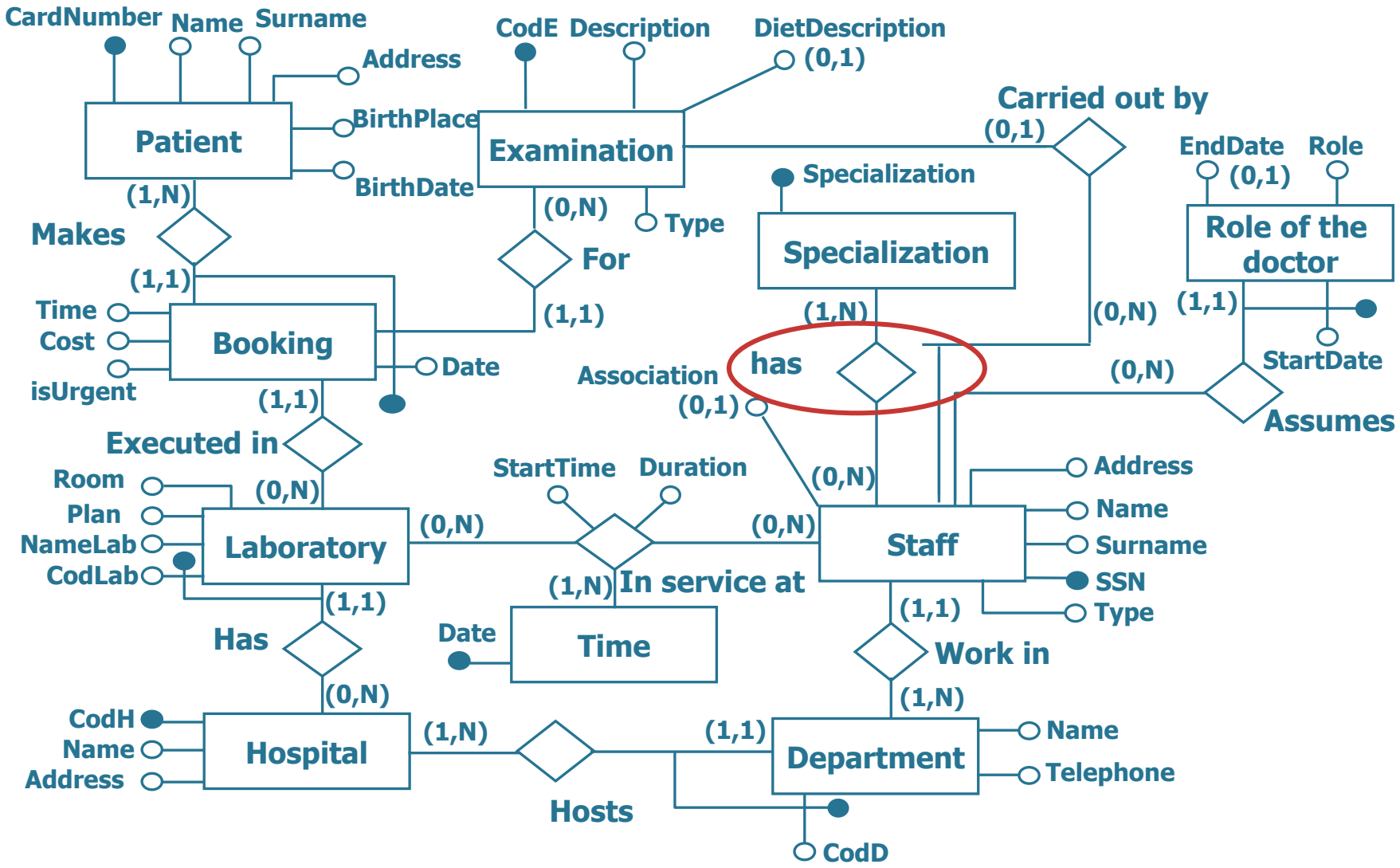
Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, *CodLab, CodH*)

DoctorRole(SSN, StartDate, EndDate*, Role)

Binary many-to-many *has* relationship



Binary many-to-many *has* relationship

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

Time(Date)

Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

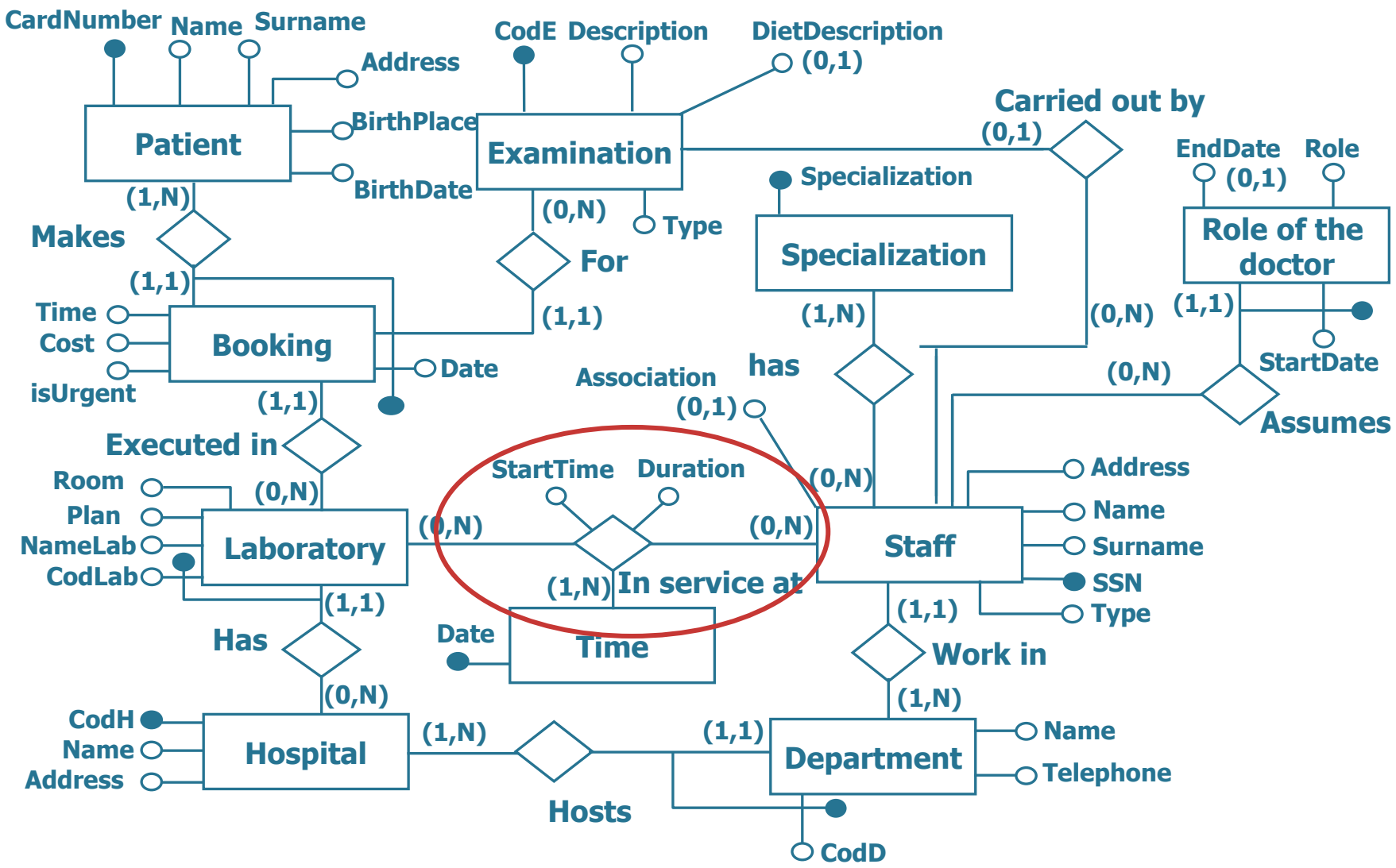
Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

DoctorRole(SSN, StartDate, EndDate*, Role)

HasSpecialization(SSN, Specialization)

Ternary many-to-many *In service at* relationship



Ternary many-to-many *In service at* relationship

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

Time(Date)

Specialization(Specialization)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

DoctorRole(SSN, StartDate, EndDate*, Role)

HasSpecialization(SSN, Specialization)

InServiceAt(SSN, CodLab, CodH, Date, StartTime, Duration)

Eliminating redundant tables

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type,
CodD, CodH)

Time(Date)

~~*Specialization(Specialization)*~~

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

DoctorRole(SSN, StartDate, EndDate*, Role)

HasSpecialization(SSN, Specialization)

InServiceAt(SSN, CodLab, CodH, Date, StartTime, Duration)

Eliminating redundant tables

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

~~*Time(Date)*~~

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

DoctorRole(SSN, StartDate, EndDate*, Role)

HasSpecialization(SSN, Specialization)

InServiceAt(SSN, CodLab, CodH, Date, StartTime, Duration)

Final relational schema

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

Hospital(CodH, Name, Address)

Examination(CodE, Description, DietDescription*, Type, SSN*)

Staff(SSN, Name, Surname, Address, Association*, Type, CodD, CodH)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

Department(CodD, CodH, Name, Telephone)

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

DoctorRole(SSN, StartDate, EndDate*, Role)

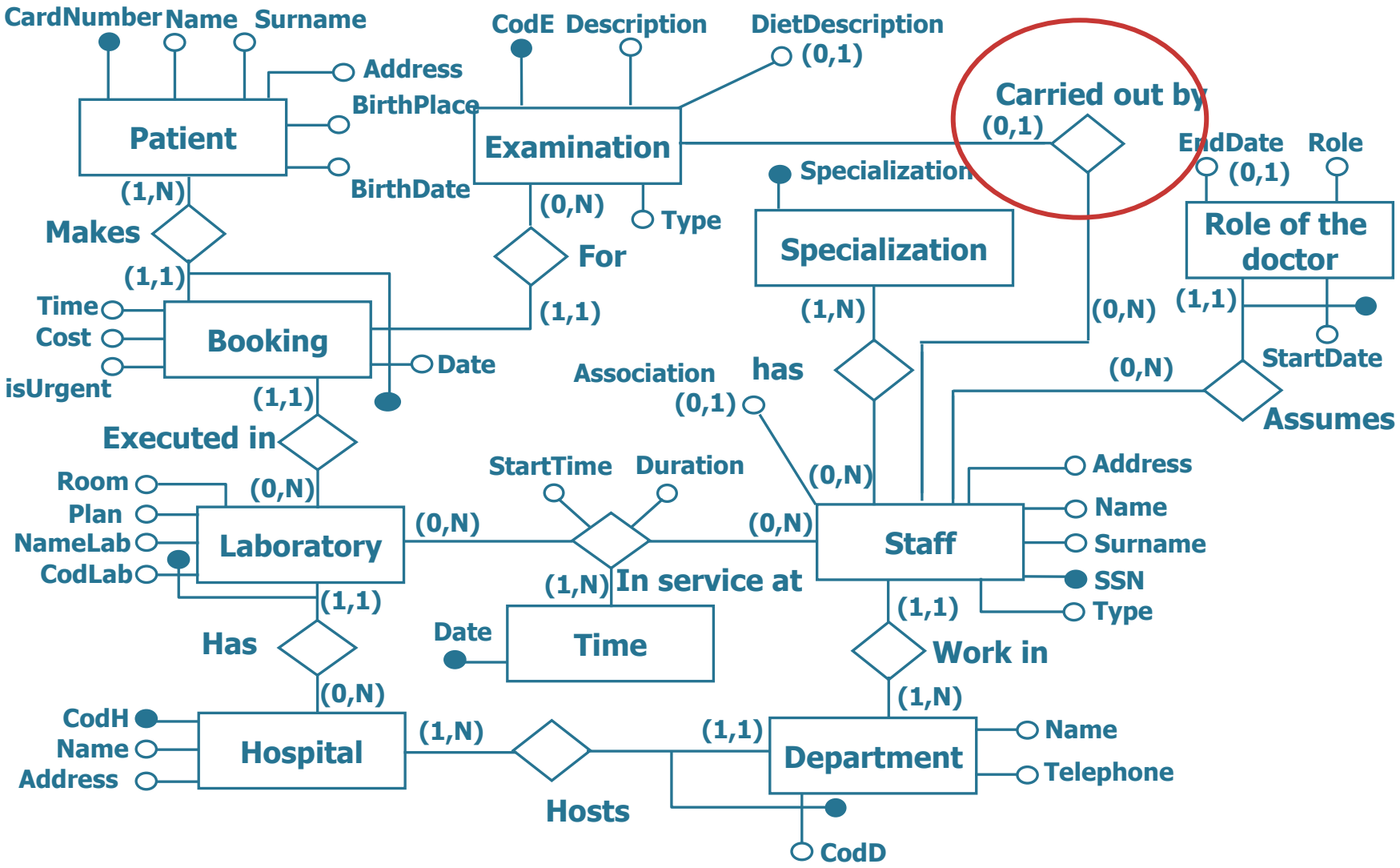
HasSpecialization(SSN, Specialization)

InServiceAt(SSN, CodLab, CodH, Date, StartTime, Duration)

Politecnico di Torino

Referential integrity constraints

Example of relational logic design


CardNumber
Name
Surname
Address
BirthPlace
BirthDate
Patient
(1,N)
Makes
(1,1)
Time
Cost
isUrgent
Booking
Date
(1,1)
Executed in
Room
Plan
NameLab
CodLab
(0,N)
Laboratory
(1,1)
Has
(0,N)
CodH
Name
Address
Hospital
CodE
Description
DietDescription
(0,1)
Examination
(0,N)
Type
For
(1,1)
StartTime
Duration
(0,N)
(0,N)
(1,N)
In service at
Date
Time
Hosts
(1,N)
(1,1)
Department
Name
Telephone
CodD
Carried out by
(0,1)
Specialization
Specialization
(1,N)
Association
(0,1)
has
(0,N)
Staff
Address
Name
Surname
SSN
Type
(1,1)
Work in
(1,N)
EndDate
(0,1)
Role
Role of the doctor
(1,1)
StartDate
(0,N)
Assumes

Referential integrity: *Carried out by* relationship

- Involved tables

Examination(CodE, Description, DietDescription*, Type, *SSN**)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

- Referential integrity constraint

Examination(SSN) REFERENCES Staff(SSN)

Referential integrity: *Work in* relationship

- Involved tables

Staff(SSN, Name, Surname, Address, Association*,Type, *CodD, CodH*)

Department(CodD, CodH, Name, Telephone)

- Referential integrity constraint

Staff(CodD,CodH) REFERENCES Department(CodD,CodH)

Referential integrity: *Hosts* relationship

- Involved tables

Department(CodD, *CodH*, Name, Telephone)

Hospital(CodH, Name, Address)

- Referential integrity constraint

Department(CodH) REFERENCES Hospital(CodH)

Referential integrity: *Has* relationship

- Involved tables

Laboratory(CodLab, *CodH*, NameLab, Plan, Room)

Hospital(CodH, Name, Address)

- Referential integrity constraint

Laboratory(CodH) REFERENCES Hospital(CodH)

Referential integrity: *Makes* relationship

- Involved tables

Booking(*CardNumber*, CodE, Date, Time, Cost, isUrgent, CodLab, CodH)

Patient(CardNumber, Name, Surname, Address, BirthPlace, BirthDate)

- Referential integrity constraint

Booking(CardNumber) REFERENCES Patient(CardNumber)

Referential integrity: *For* relationship

- Involved tables

Booking(CardNumber, *CodE*, Date, Time, Cost, isUrgent, CodLab, CodH)

Examination(CodE, Description, DietDescription*, Type, SSN*)

- Referential integrity constraint

Booking(CodE) REFERENCES Examination(CodE)

Referential integrity: *Executed in* relationship

- Involved tables

Booking(CardNumber, CodE, Date, Time, Cost, isUrgent, *CodLab, CodH*)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

- Referential integrity constraint

Booking(CodLab,CodH) REFERENCES Laboratory(CodLab,CodH)

Referential integrity: *Assumes* relationship

- Involved tables

DoctorRole(*SSN*, StartDate, EndDate*, Role)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

- Referential integrity constraint

DoctorRole(SSN) REFERENCES Staff(SSN)

Referential integrity: *Has* relationship

- Involved tables

HasSpecialization(*SSN*, Specialization)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

- Referential integrity constraint

HasSpecialization(SSN) REFERENCES Staff(SSN)

Referential integrity: *In service at* relationship

- Involved tables

InServiceAt(*SSN*, CodLab, CodH, Date, StartTime, Duration)

Staff(SSN, Name, Surname, Address, Association*,Type, CodD, CodH)

- Referential integrity constraint

InServiceAt(SSN) REFERENCES Staff(SSN)

Referential integrity: *In service at* relationship

- Involved tables

InServiceAt(SSN, *CodLab*, *CodH*, Date, StartTime, Duration)

Laboratory(CodLab, CodH, NameLab, Plan, Room)

- Referential integrity constraint

InServiceAt(CodLab,CodH) REFERENCES Laboratory(CodLab,CodH)

Referential integrity constraints

Examination(SSN) REFERENCES Staff(SSN)

Staff(CodD,CodH) REFERENCES Department(CodD,CodH)

Department(CodH) REFERENCES Hospital(CodH)

Laboratory(CodH) REFERENCES Hospital(CodH)

Booking(CardNumber) REFERENCES Patient(CardNumber)

Booking(CodE) REFERENCES Examination(CodE)

Booking(CodLab,CodH) REFERENCES Laboratory(CodLab,CodH)

DoctorRole(SSN) REFERENCES Staff(SSN)

HasSpecialization(SSN) REFERENCES Staff(SSN)

InServiceAt(SSN) REFERENCES Staff(SSN)

InServiceAt(CodLab,CodH) REFERENCES Laboratory(CodLab,CodH)