UNIVERSITY
OF MIAMI

CSC423 Project

Report 2 - Developing a Logical Data Model

William James Calum Blair
BS Computer Science

wjb102@miami.edu

Department of Computer Science

November 17, 2022

Contents

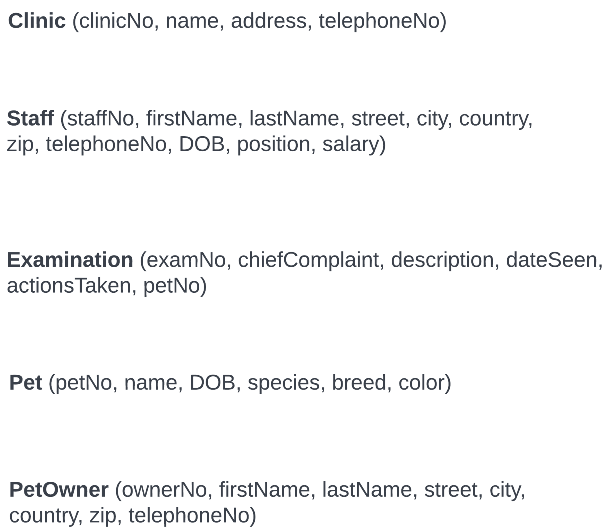
1 Author & Change Control

Author, Change Control

Date	Author	Version	Description
17-Nov-2022	William Blair	1.0	Logical Data Model

2 Deriving relations from the conceptual model

Here are the current tables:



Clinic (clinicNo, name, address, telephoneNo)

Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary)

Examination (examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo)

Pet (petNo, name, DOB, species, breed, color)

PetOwner (ownerNo, firstName, lastName, street, city, country, zip, telephoneNo)

Here we derive the relations for the logical data model to represent entities, relationships, and attributes. We derive relations for the following structures that have appeared in conceptual data model:

- strong entity types;
- one-to-many (1:*) binary relationship types;
- one-to-one (1:1) mandatory relationship types;

2.1 Deriving relations for strong entity types

For each strong entity type in the conceptual data model, we will create a relation that includes all the simple attributes of that entity:

The composition of the **Clinic** relation is:

- Clinic (clinicNo, name, address, telephoneNo)

The composition of the **Staff** relation is:

- Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary)

The composition of the **PetOwner** relation is:

- PetOwner (ownerNo, firstName, lastName, street, city, country, zip, telephoneNo)

The composition of the **Pet** relation is:

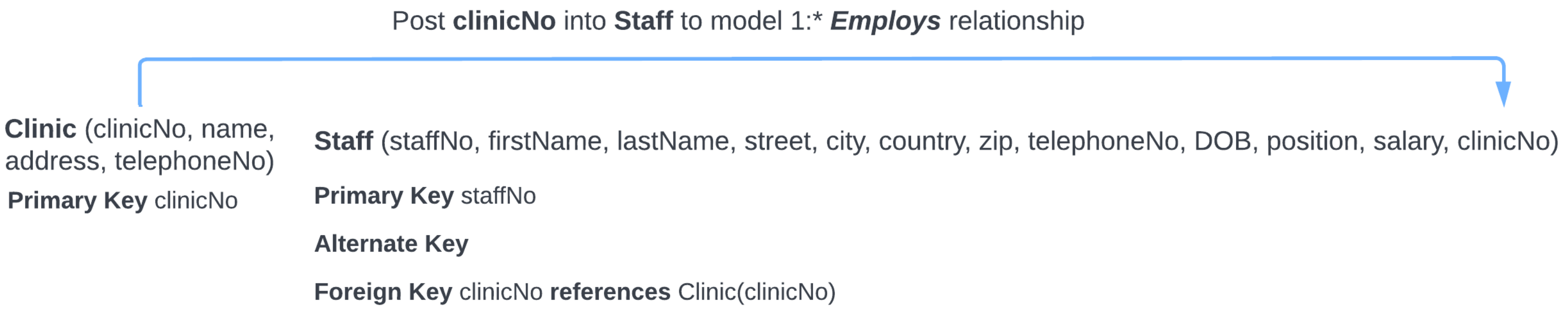
- Pet (petNo, name, DOB, species, breed, color)

The composition of the **Examination** relation is:

- Examination (examNo, chiefComplaint, description, dateSeen, actionsTaken)

2.2 Deriving one-to-many (1:*) binary relationships

The composition of the *Clinic* and *Staff* relation is:



Post **clinicNo** into **Staff** to model 1:* ***Employs*** relationship

Clinic (clinicNo, name, address, telephoneNo)

Primary Key clinicNo

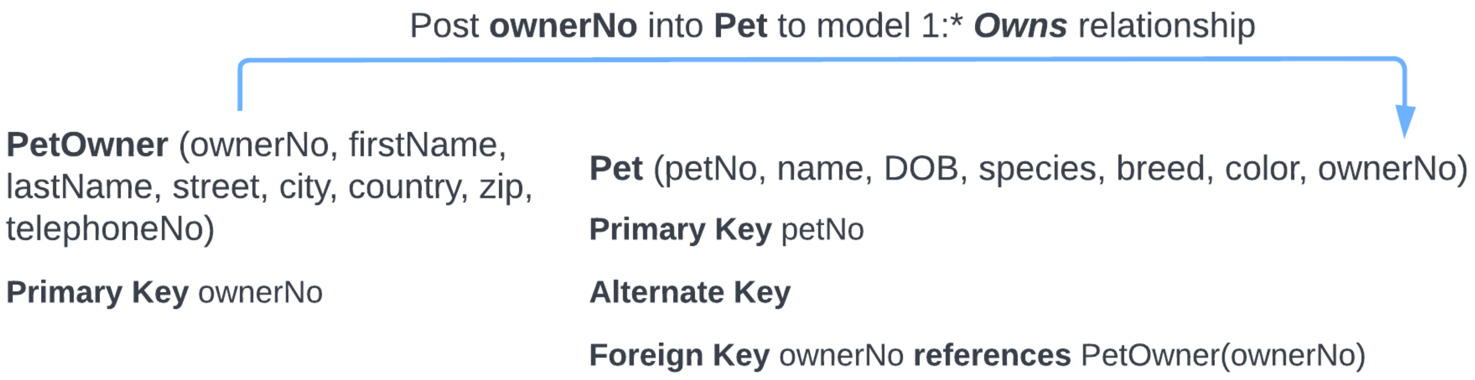
Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary, clinicNo)

Primary Key staffNo

Alternate Key

Foreign Key clinicNo **references** Clinic(clinicNo)

The composition of the *Pet Owner* and *Pet* relation is:



Post **ownerNo** into **Pet** to model 1:* ***Owns*** relationship

PetOwner (ownerNo, firstName, lastName, street, city, country, zip, telephoneNo)

Primary Key ownerNo

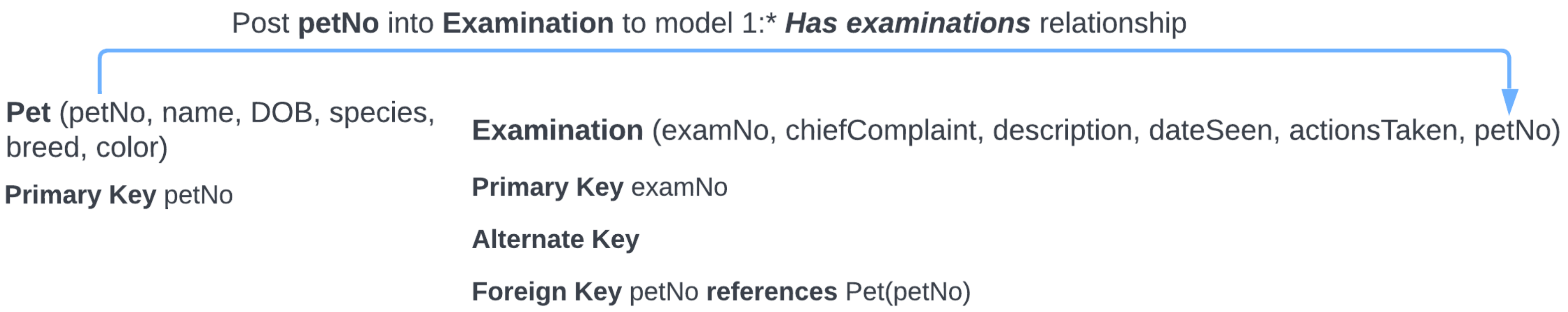
Pet (petNo, name, DOB, species, breed, color, ownerNo)

Primary Key petNo

Alternate Key

Foreign Key ownerNo **references** PetOwner(ownerNo)

The composition of the *Pet* and *Examination* relation is:



Post **petNo** into **Examination** to model 1:* ***Has examinations*** relationship

Pet (petNo, name, DOB, species, breed, color)

Primary Key petNo

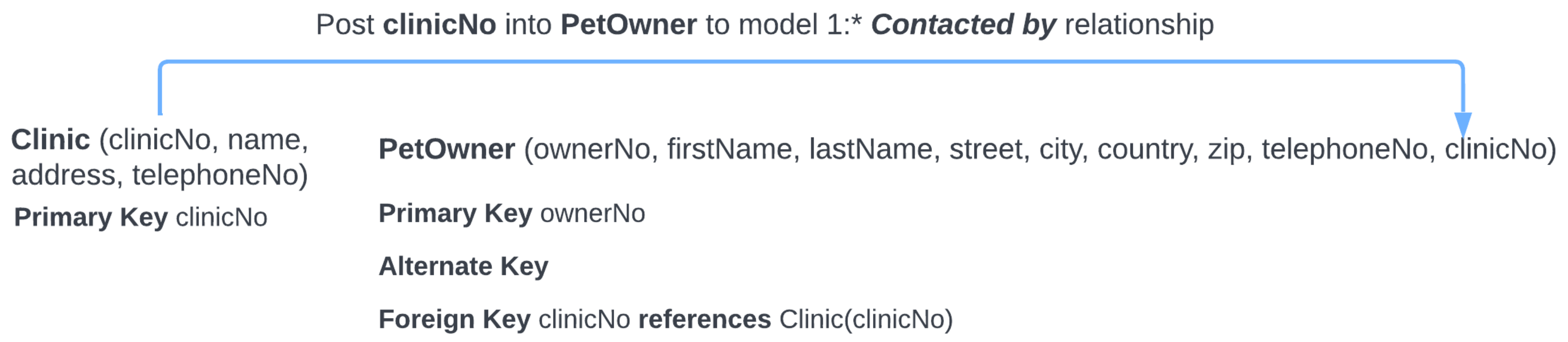
Examination (examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo)

Primary Key examNo

Alternate Key

Foreign Key petNo **references** Pet(petNo)

The composition of the *Clinic* and *Pet Owner* relation is:



Post **clinicNo** into **PetOwner** to model 1:* ***Contacted by*** relationship

Clinic (clinicNo, name, address, telephoneNo)

Primary Key clinicNo

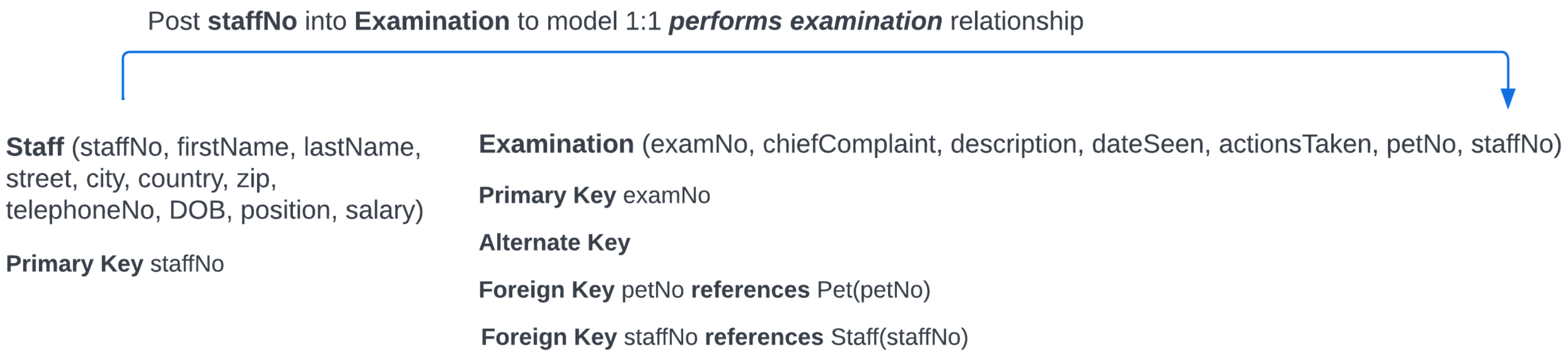
PetOwner (ownerNo, firstName, lastName, street, city, country, zip, telephoneNo, clinicNo)

Primary Key ownerNo

Alternate Key

Foreign Key clinicNo **references** Clinic(clinicNo)

The composition of the *Staff* and *Examination* relation is:



Post **staffNo** into **Examination** to model 1:1 ***performs examination*** relationship

Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary)

Primary Key staffNo

Examination (examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo, staffNo)

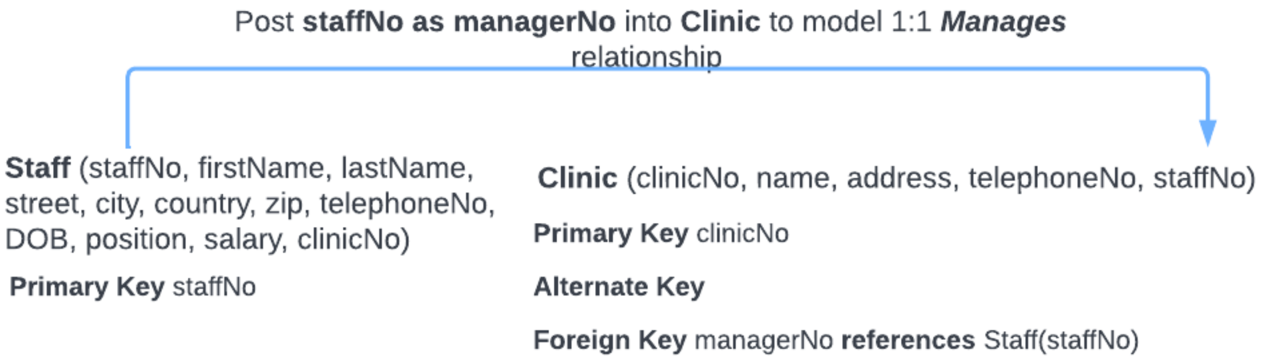
Primary Key examNo

Alternate Key

Foreign Key petNo **references** Pet(petNo)

Foreign Key staffNo **references** Staff(staffNo)

2.3 Deriving one-to-one (1:1) mandatory relationship types



Post **staffNo as managerNo** into **Clinic** to model 1:1 ***Manages*** relationship

Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary, clinicNo)

Primary Key staffNo

Clinic (clinicNo, name, address, telephoneNo, staffNo)

Primary Key clinicNo

Alternate Key

Foreign Key managerNo **references** Staff(staffNo)

Here are the final tables after identifying the relations, with the changes in blue:

Clinic (clinicNo, name, address, telephoneNo, managerNo)

Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary, clinicNo)

Examination (examNo, chiefComplaint, description, dateSeen, actionsTaken, petNo, staffNo)

Pet (petNo, name, DOB, species, breed, color, ownerNo)

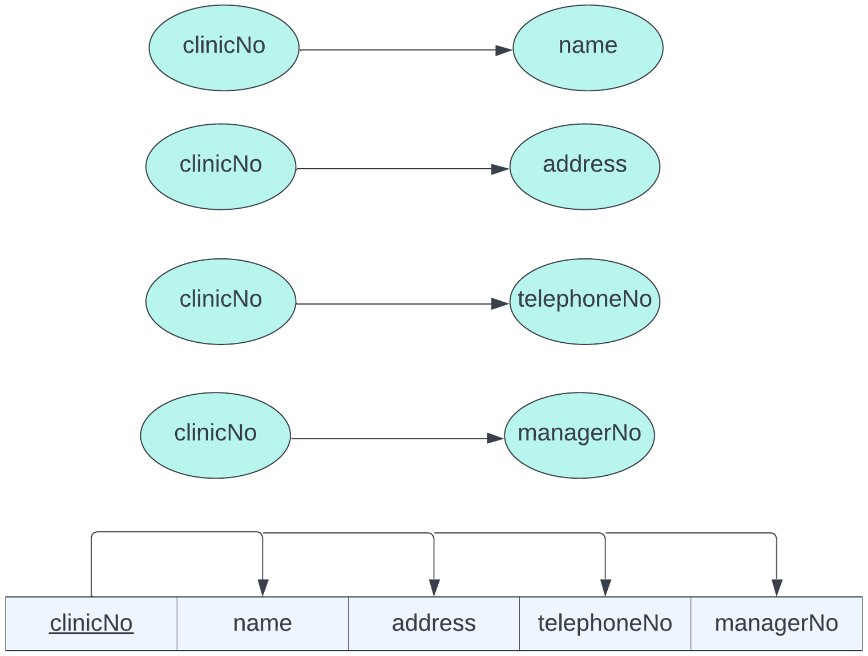
PetOwner (ownerNo, firstName, lastName, street, city,

3 Validating the logical model using normalisation to 3NF

3.1 Identifying functional dependencies

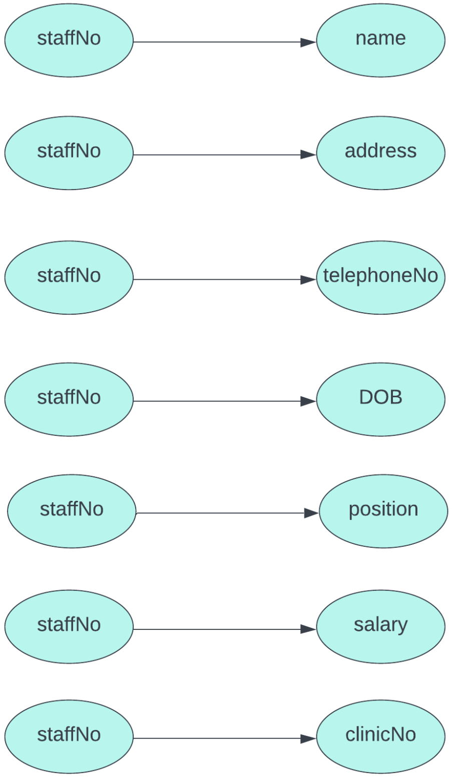
Clinic Functional Dependencies

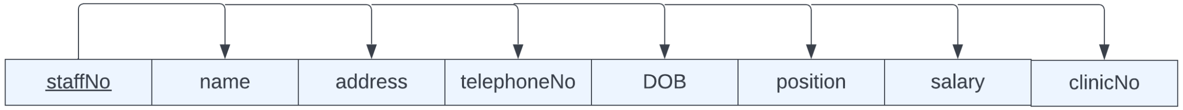
Clinic (<u>clinicNo</u>, name, address, telephoneNo, managerNo)



Staff Functional Dependencies

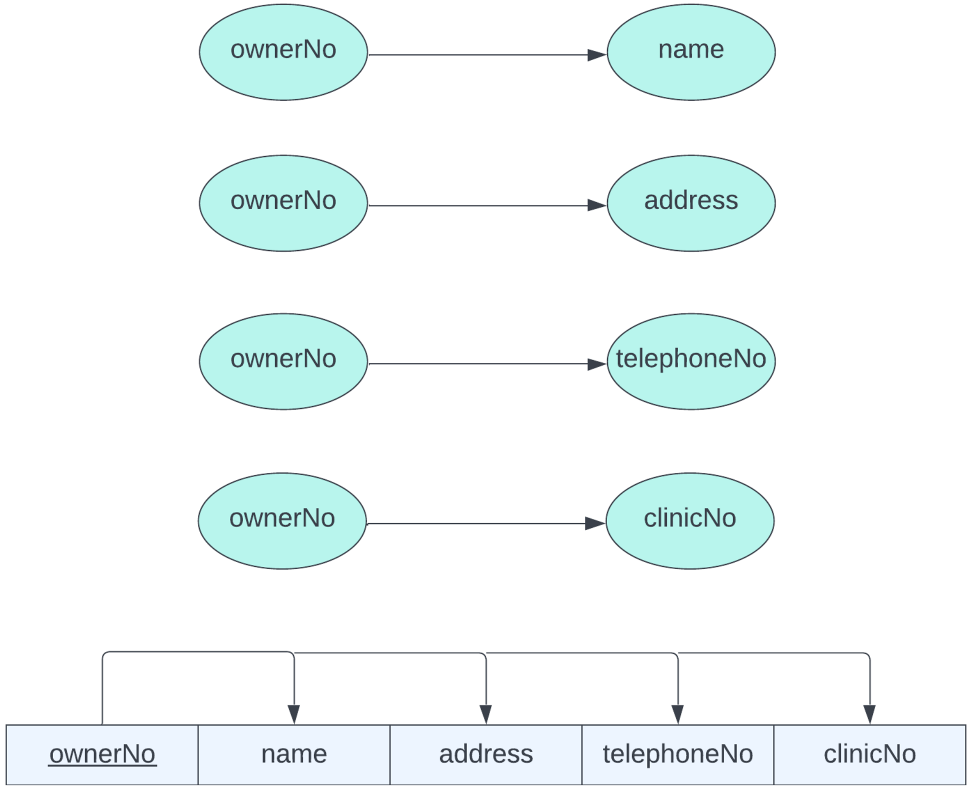
Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary, clinicNo)





PetOwner Functional Dependencies

PetOwner (<u>ownerNo</u>, firstName, lastName, street, city, country, zip, telephoneNo, clinicNo)



ownerNo → name

ownerNo → address

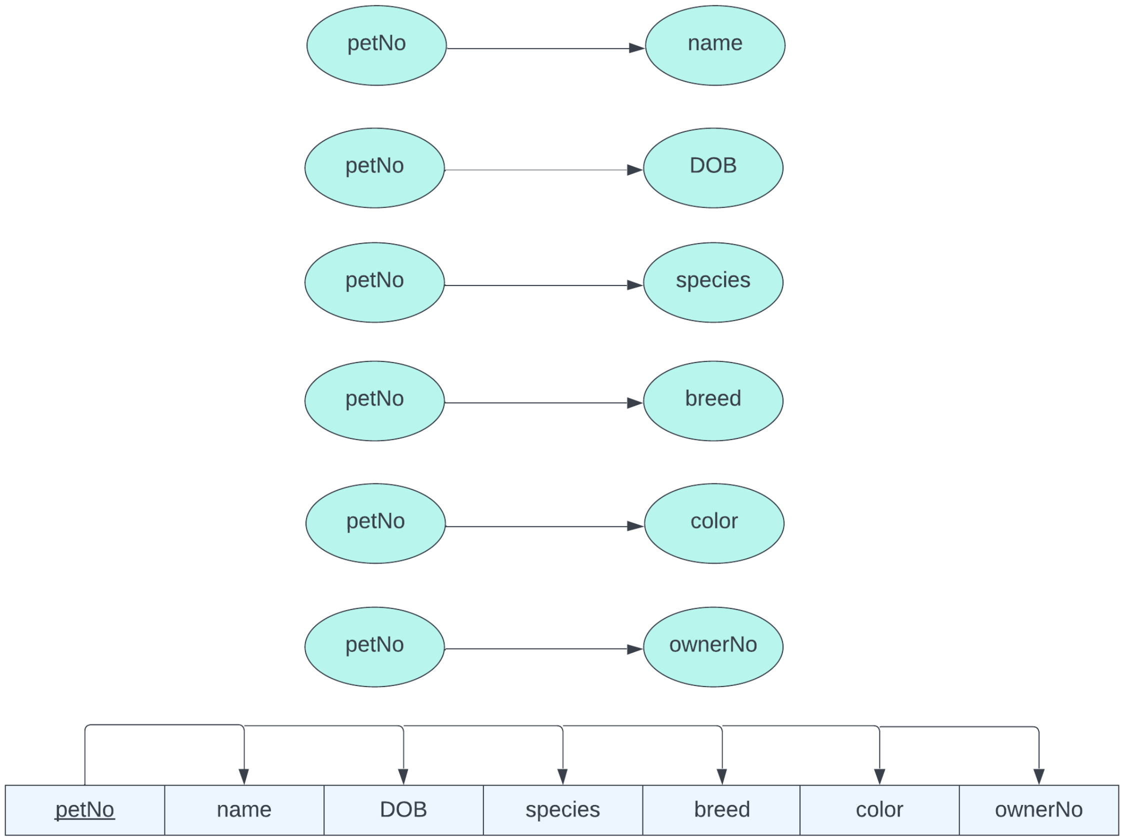
ownerNo → telephoneNo

ownerNo → clinicNo

<u>ownerNo</u>	name	address	telephoneNo	clinicNo

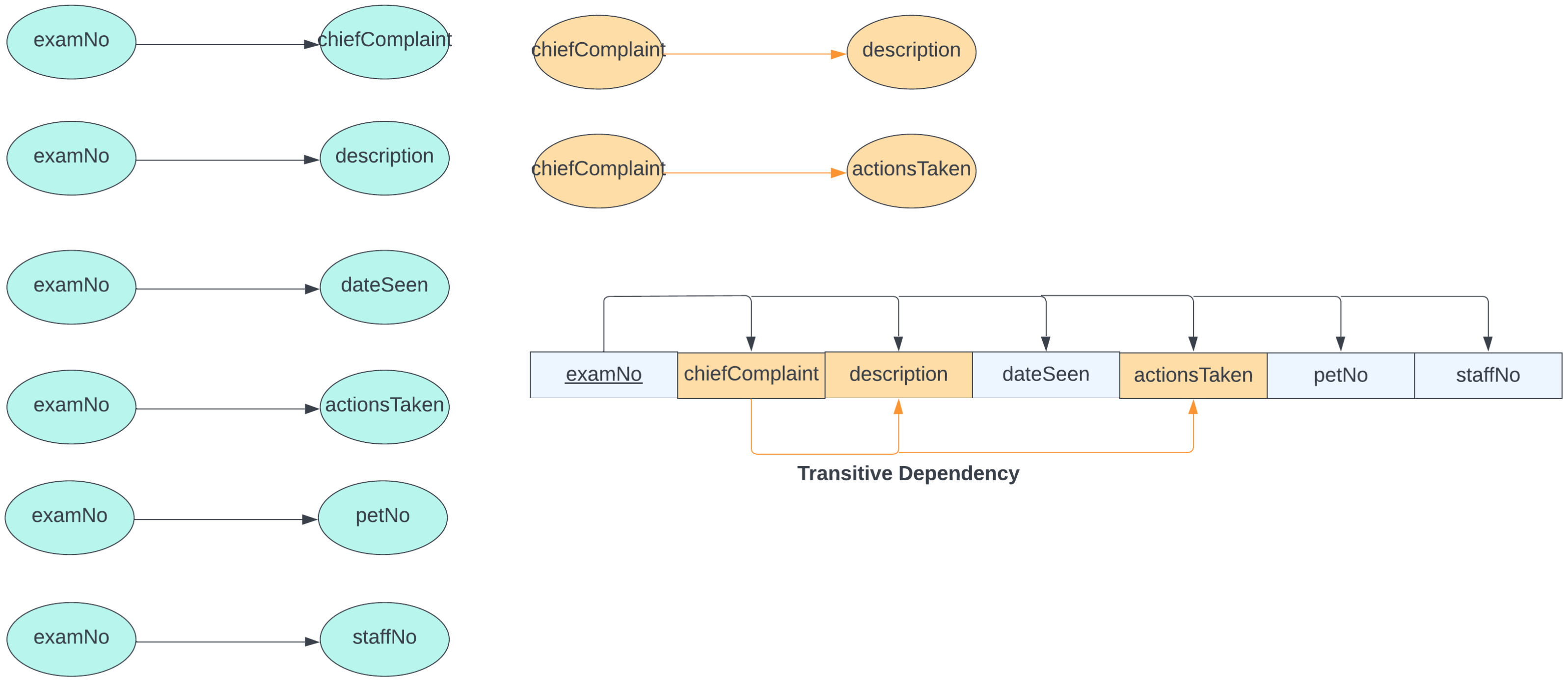
Pet Functional Dependencies

Pet (petNo, name, DOB, species, breed, color, ownerNo)

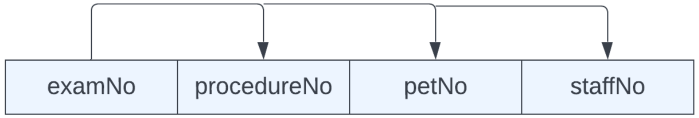


Examination Functional Dependencies

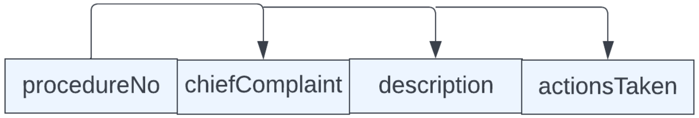
Examination (<u>examNo</u>, chiefComplaint, description, dateSeen, actionsTaken, petNo, staffNo)



Because there is a transitive dependency within the Examination table, we will split it up into a separate relation ’**Procedure**’ that has ’procedureNo’ as the primary key. The Procedure relation represents the procedure taken and the details of the examination, including the chiefComplaint, description of the examination / procedure, and the actionsTaken. Thus all attributes of the resulting relations are fully functionally dependent on the primary key.



Examination (examNo, procedureNo, petNo, staffNo)



Procedure (procedureNo, chiefComplaint, description, actionsTaken)

Clinic (clinicNo, name, address, telephoneNo, managerNo)

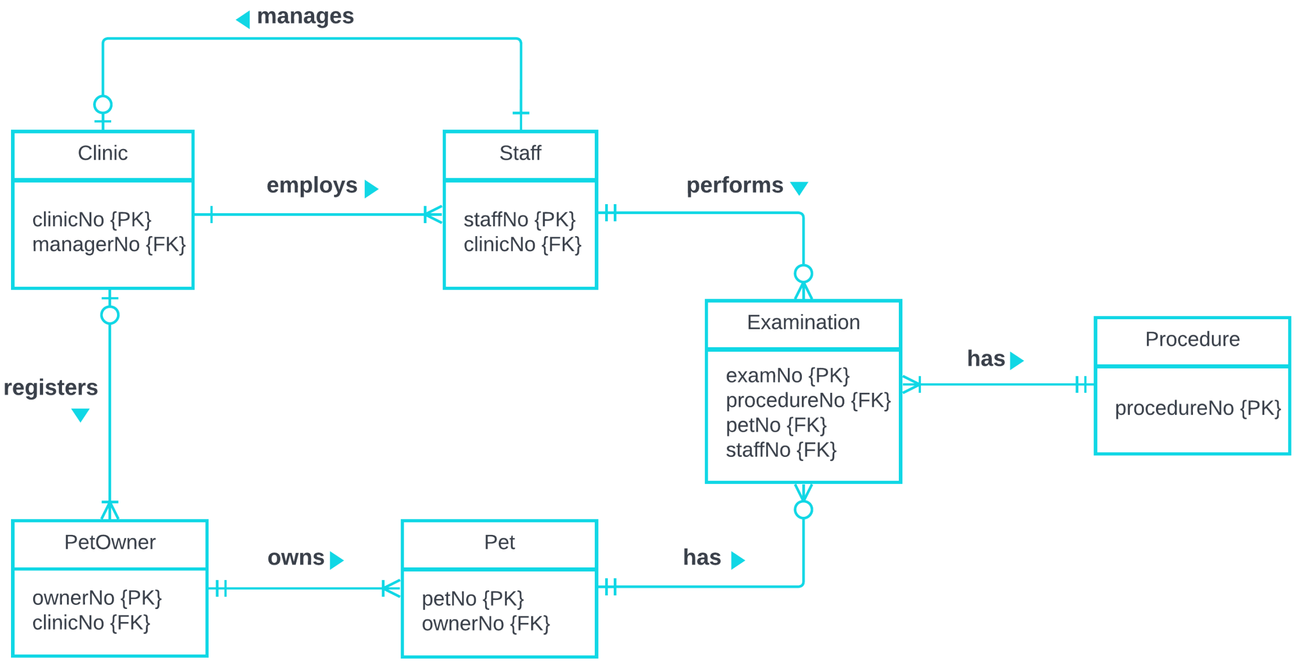
Staff (staffNo, firstName, lastName, street, city, country, zip, telephoneNo, DOB, position, salary, clinicNo)

Examination (examNo, procedureNo, dateSeen, petNo, staffNo)

Procedure (procedureNo, chiefComplaint, description, actionsTaken)

Pet (petNo, name, DOB, species, breed, color, ownerNo)

PetOwner (ownerNo, firstName, lastName, street, city,


manages
Clinic
clinicNo {PK}
managerNo {FK}
employs
Staff
staffNo {PK}
clinicNo {FK}
performs
Examination
examNo {PK}
procedureNo {FK}
petNo {FK}
staffNo {FK}
has
Procedure
procedureNo {PK}
registers
PetOwner
ownerNo {PK}
clinicNo {FK}
owns
Pet
petNo {PK}
ownerNo {FK}
has

4 Validating the logical model against user transactions

5 Defining integrity constraints

5.1 Primary key constraints

For all primary keys:

- No duplicate values in relation; the column assigned as primary key should have UNIQUE values only.
- NO NULL values are present in column with primary key.
- Only one primary key per table.
- No new tuple can be added with a primary key that already exists.

5.2 Referential integrity / Foreign key constraints

Entity	*Foreign Keys*	*Nulls Allowed*	*On Delete*	*On Update*
Clinic	managerNo references Staff(staffNo)	No	SET DEFAULT (next staff member in hierarchical model)	CASCADE
Staff	clinicNo references Clinic(clinicNo)	No	NO ACTION	CASCADE
PetOwner	clinicNo references Clinic(clinicNo)	No	NO ACTION	CASCADE
Pet	ownerNo references PetOwner(ownerNo)	No	CASCADE	NO ACTION
Examination	procedureNo references Procedure(procedureNo) petNo references Pet(petNo) staffNo references Staff(staffNo)	No No No	NO ACTION CASCADE NO ACTION	CASCADE NO ACTION NO ACTION

5.3 Alternate key constraints (if any)

Entity	*Alternate Keys*	*On Delete*	*On Update*
Clinic	name		
Staff	telephoneNo		
PetOwner	telephoneNo		
Pet			
Examination			

5.4 Required data

5.5 Attribute Domain Constraints

5.5.1 Clinic Attribute Domain Constraints

Attribute	*Domain Name*	*Meaning*	*Definition*
clinicNo	ClincNos	The set of all possible clinic numbers.	range from C001 - C999
name	SmallText	The set of all possible small alphabetical text entries	varchar: 50
address	SmallText		
telephoneNo	TelephoneNos	The set of all possible telephone numbers	fixed size of 15: format +1 NXX-NXX-XXXX, N range 2-9, X range 0-9
managerNo	StaffNos		

5.5.2 Staff Attribute Domain Constraints

Attribute	*Domain Name*	*Meaning*	*Definition*
staffNo	StaffNos	The set of all possible staff numbers	range from S0001 - S9999
firstName	SmallText		
lastName	SmallText		
street	SmallText		
city	SmallText		
country	SmallText		
zip	ZipCodes	The set of all possible zip-codes	fixed size 10, format XXXXX-XXXX, X range 0-9
telephoneNo	TelephoneNos		
DOB	DateOfBirths	The set of all possible DOBs	Date, format YYYY-MM-DD value greater than 1900
position	StaffPositions	The set of all staff positions	varchar 20
salary	StaffSalaries	The set of all possible staff salaries	varchar 10
clinicNo	ClincNos		

5.5.3 Examination Attribute Domain Constraints

Attribute	*Domain Name*	*Meaning*	*Definition*
examNo	Exam	The set of all possible examNos	range from E00000001 - E99999999
procedureNo	procedureNos		
dateSeen	CurrentDate	The current date	fixed value: current date, format YYYY-MM-DD
petNo	PetNos		
staffNo	StaffNos		

5.5.4 Procedure Attribute Domain Constraints

Attribute	*Domain Name*	*Meaning*	*Definition*
procedureNo	ProcedureNos	The set of all possible procedureNos	range from P0001 - P9999
chiefComplaint	SmallText		
description	LargeText	The set of all	
actionsTaken	LargeText		

5.5.5 Pet Attribute Domain Constraints

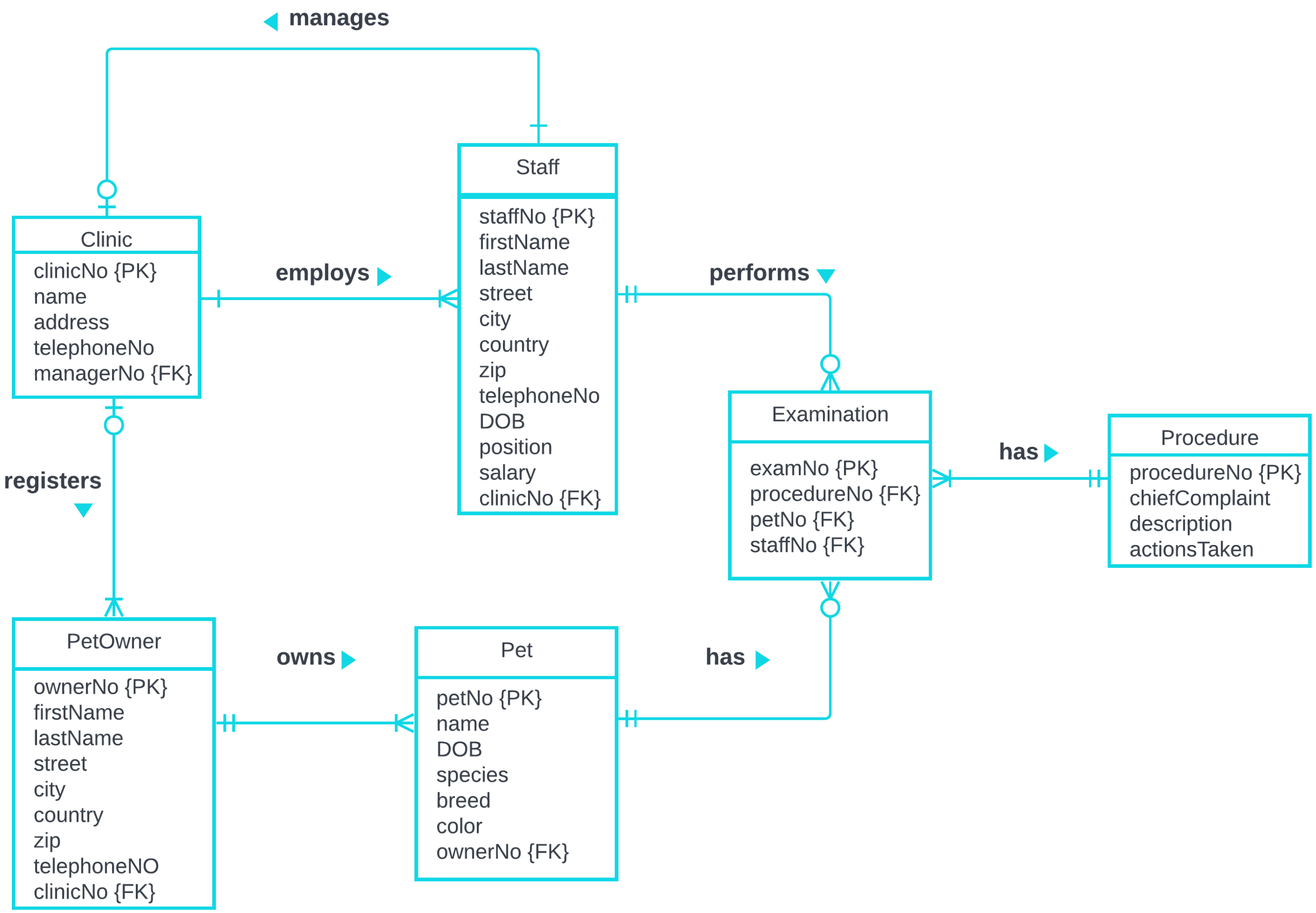
Attribute	*Domain Name*	*Meaning*	*Definition*
petNo	PetNos	The set of all possible petNos	range from P0000-P9999
name	SmallText		
DOB	DateOfBirths		
species	SmallText		
breed	SmallText		
color	SmallText		
ownerNo	OwnerNos		

5.5.6 Pet Owner Attribute Domain Constraints

Attribute	*Domain Name*	*Meaning*	*Definition*
ownerNo	OwnerNos	The set of all possible Pet Owner Nos	range from O00000-O99999
firstName	SmallText		
lastName	SmallText		
street	SmallText		
city	SmallText		
country	SmallText		
zip	ZipCodes		
telephoneNo	TelephoneNos		
clinicNo	ClinicNos		

5.6 General constraints (if any)

6 E-R diagram for logical level (contains FKs as attributes


manages
Staff
staffNo {PK}
firstName
lastName
street
city
country
zip
telephoneNo
DOB
position
salary
clinicNo {FK}
Clinic
clinicNo {PK}
name
address
telephoneNo
managerNo {FK}
employs
performs
Examination
examNo {PK}
procedureNo {FK}
petNo {FK}
staffNo {FK}
has
Procedure
procedureNo {PK}
chiefComplaint
description
actionsTaken
registers
PetOwner
ownerNo {PK}
firstName
lastName
street
city
country
zip
telephoneNO
clinicNo {FK}
owns
Pet
petNo {PK}
name
DOB
species
breed
color
ownerNo {FK}
has