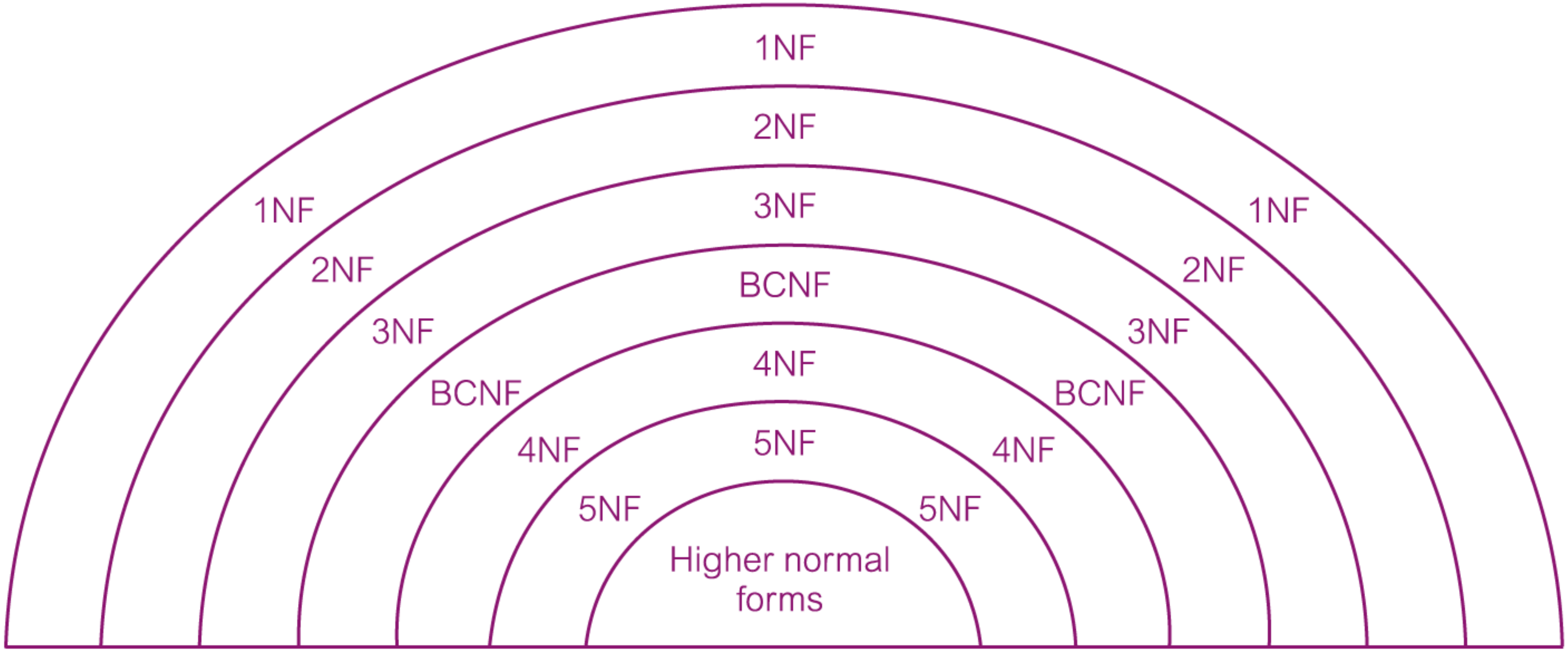
Advanced Normalization

Dr Na Yao

Objectives

- Understand there are normal forms that go beyond Third Normal Form (3NF).
- Understand and be able to identify Boyce-Codd Normal Form (BCNF).
- Be able to decompose relations to BCNF.
- Understand and be able to explain Multi-valued Dependency (MVD)
- Understand and be able to identify Fourth Normal Form (4NF).

The Process of Normalization



Boyce–Codd Normal Form (BCNF)

- Based on functional dependencies that take into account all candidate keys in a relation, however BCNF also has additional constraints compared with the general definition of 3NF.

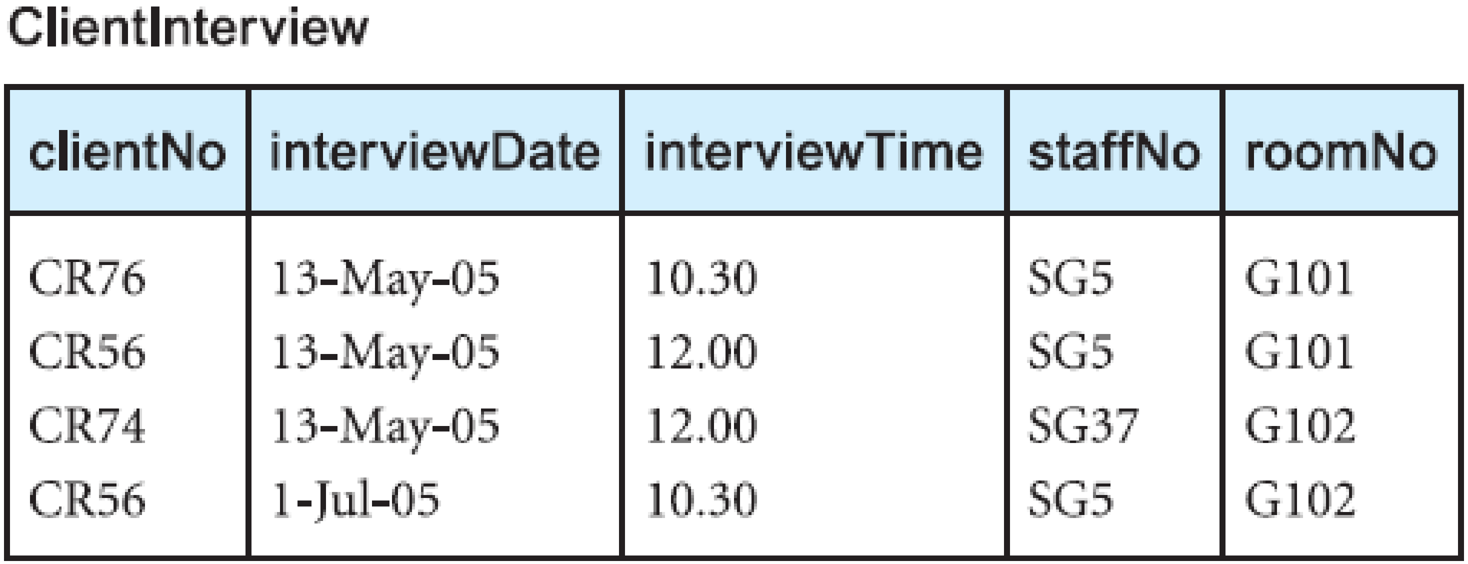
- Boyce–Codd normal form (BCNF)
  - A relation is in BCNF if and only if every determinant is a candidate key.

BCNF example

Example:

- Client is interviewed by members of staff in *DreamHome* case.
- The members of staff involved in interviewing clients are allocated to a specific room on the day of interview.
- However a room may be allocated to several members of staff as required throughout a working day.
- A client is interviewed only once on a given date, but may be requested to attend further interviews at later dates.

BCNF example



ClientInterview

clientNo	interviewDate	interviewTime	staffNo	roomNo
CR76	13-May-05	10.30	SG5	G101
CR56	13-May-05	12.00	SG5	G101
CR74	13-May-05	12.00	SG37	G102
CR56	1-Jul-05	10.30	SG5	G102

- Functional dependencies:

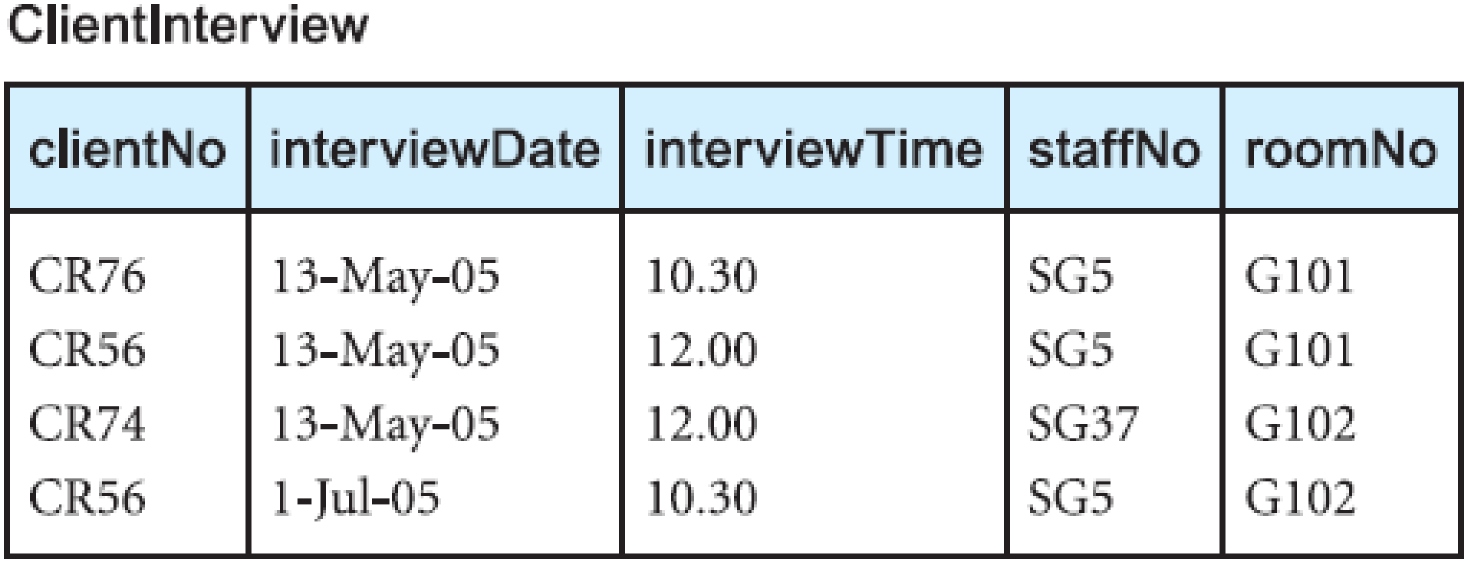
clientNo, interviewDate → interviewTime, staffNo, roomNo (Primary key)

staffNo, interviewDate, interviewTime → clientNo (Candidate key)

roomNo, interviewDate, interviewTime → staffNo, clientNo (Candidate key)

staffNo, interviewDate → roomNo

BCNF example

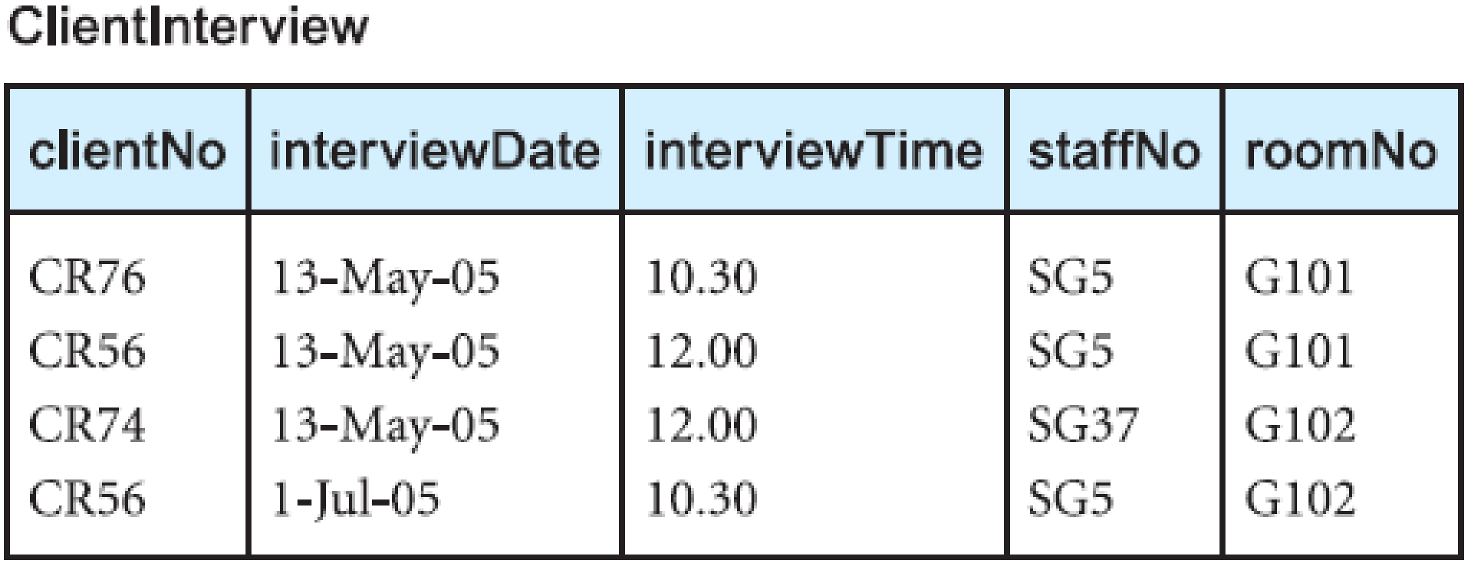


ClientInterview

clientNo	interviewDate	interviewTime	staffNo	roomNo
CR76	13-May-05	10.30	SG5	G101
CR56	13-May-05	12.00	SG5	G101
CR74	13-May-05	12.00	SG37	G102
CR56	1-Jul-05	10.30	SG5	G102

- Candidate keys:
  (clientNo, interviewDate)
  (staffNo, interviewDate, interviewTime)
  (roomNo, interviewDate, interviewTime)

BCNF example



ClientInterview

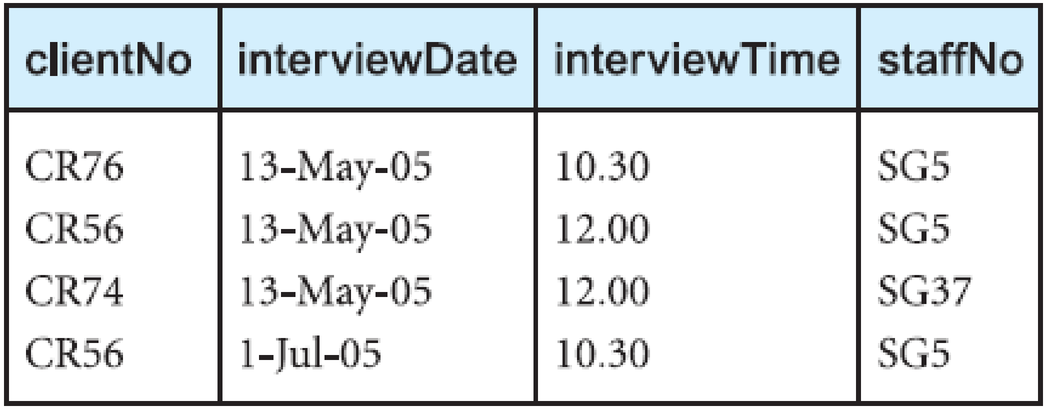
clientNo	interviewDate	interviewTime	staffNo	roomNo
CR76	13-May-05	10.30	SG5	G101
CR56	13-May-05	12.00	SG5	G101
CR74	13-May-05	12.00	SG37	G102
CR56	1-Jul-05	10.30	SG5	G102

- Update anomaly: to change the room number for staff SG5 on 13-May-05, two tuples must be updated.

BCNF example

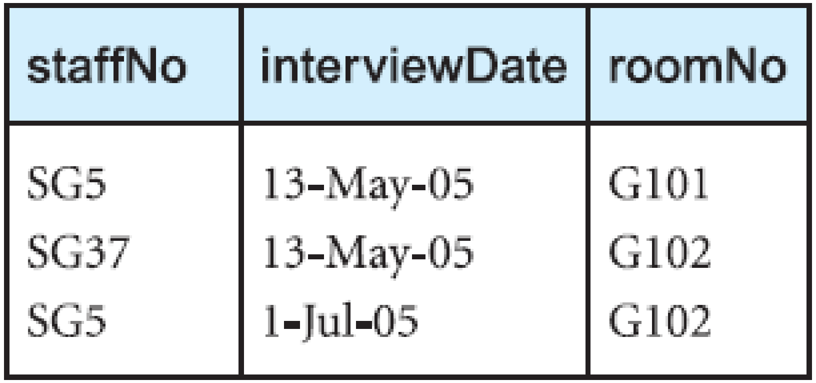
- Interview (clientNo, interviewDate, interviewTime, staffNo)
- StaffRoom (staffNo, interviewDate, roomNo)

Interview



clientNo	interviewDate	interviewTime	staffNo
CR76	13-May-05	10.30	SG5
CR56	13-May-05	12.00	SG5
CR74	13-May-05	12.00	SG37
CR56	1-Jul-05	10.30	SG5

StaffRoom



staffNo	interviewDate	roomNo
SG5	13-May-05	G101
SG37	13-May-05	G102
SG5	1-Jul-05	G102

Boyce–Codd Normal Form (BCNF)

- Every relation in BCNF is also in 3NF. However, a relation in 3NF is not necessarily in BCNF.

- Violation of BCNF is quite *rare*.

- The potential to violate BCNF may occur in a relation that:
  - contains **two (or more)** composite candidate keys;
  - the candidate keys **overlap**, that is have at least one attribute in common.

BCNF exercise

- For the relation

  Apply(SSN, collegeName, state, date, major), suppose college names are unique and students may apply to each college only once, so we have two FDs: collegeName → state and SSN,collegeName → date,major. Is Apply in BCNF?

BCNF exercise

- Relation Z(A, B, C, D, E) has functional dependencies:

  A, B → C, D, E

  B, C → D

  Is Z in BCNF?

Algorithm for decomposing relations into BCNF

- Relation R with FDs
- Compute keys for R
- Repeat until all relations are in BCNF:
    - Pick any R’ with A->B that violates BCNF
    - Decompose R’ into R1(A, B) and R2(A, rest)
    - Compute FDs for R1 and R2
    - Compute keys for R1 and R2

BCNF problem

- Consider following relation StudentLabTime:

Student	courseLab	time
111	Database	9:00
112	Database	9:00
113	Database	11:00
111	Multimedia	13:00
113	Multimedia	15:00

- Each course has several labs
- Only one lab (of any course at all) takes place at any given time
- Each student taking a course is assigned to a single lab for it

BCNF problem

Student	courseLab	time
111	Database	9:00
112	Database	9:00
113	Database	11:00
111	Multimedia	13:00
113	Multimedia	15:00

- FDs:

  Student, courseLab → time

  time → courseLab

- Candidate keys: (Student, courseLab) and (Student, time)

BCNF problem

- To change StudentLabTime to BCNF:

Student	time
111	9:00
112	9:00
113	11:00
111	13:00
113	15:00
111	**11:00**

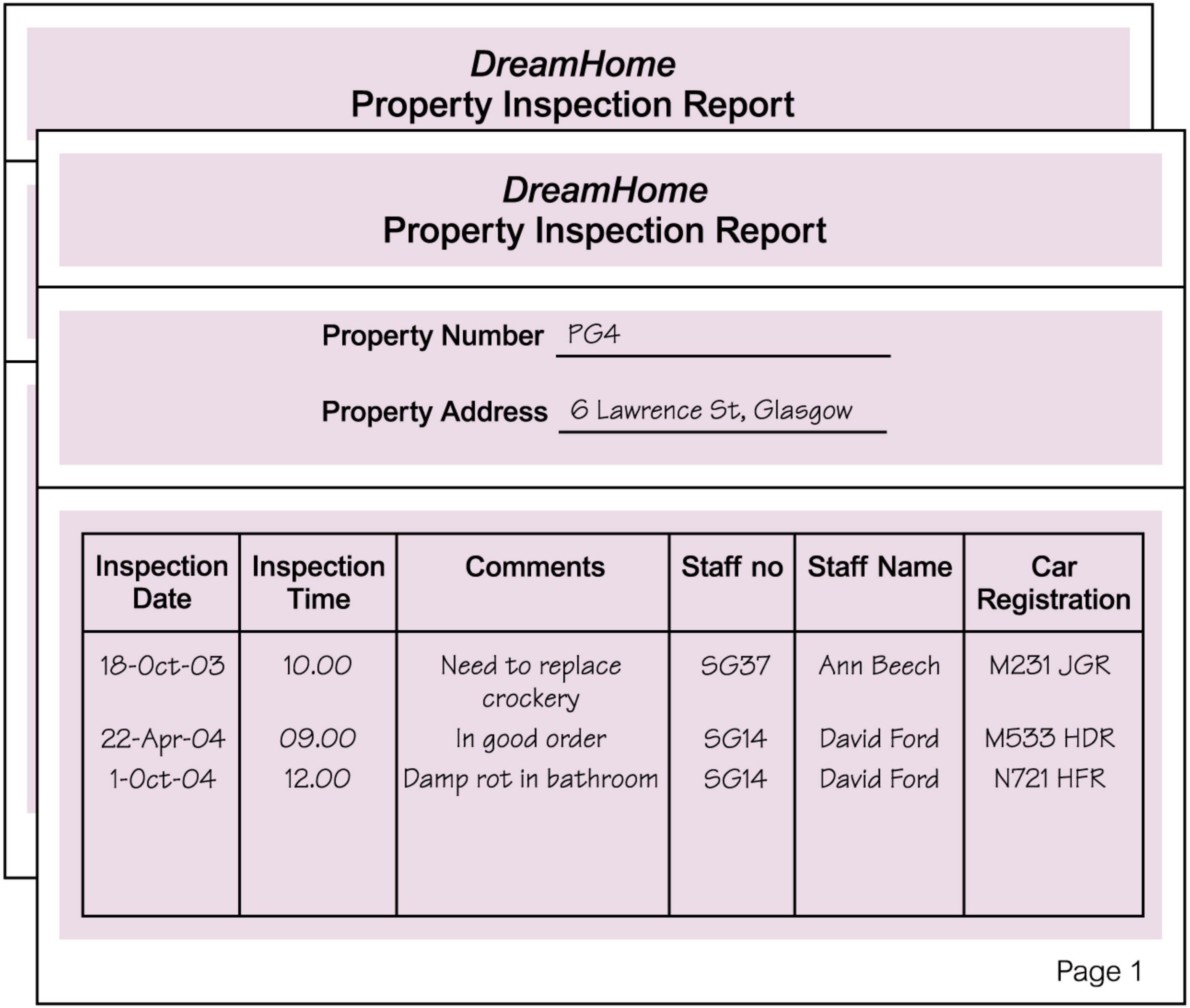
time	courseLab
9:00	Database
11:00	Database
13:00	Multimedia
15:00	Multimedia

However the decomposition is not acceptable because it allows us to record multiple times of the same courseLab against the same student. That is, we have lost the FD: Student, courseLab → Time

BCNF problem

- A set of functional dependencies ***{AB → C, C → B}*** cannot be represented by a BCNF schema
- A design that eliminates all of these anomalies (but does not conform to BCNF) is possible. This design introduces a new normal form, know as Elementary Key Normal Form (EKNF).

Review of Normalization (UNF to BCNF)



DreamHome
Property Inspection Report

Property Number PG4

Property Address 6 Lawrence St, Glasgow

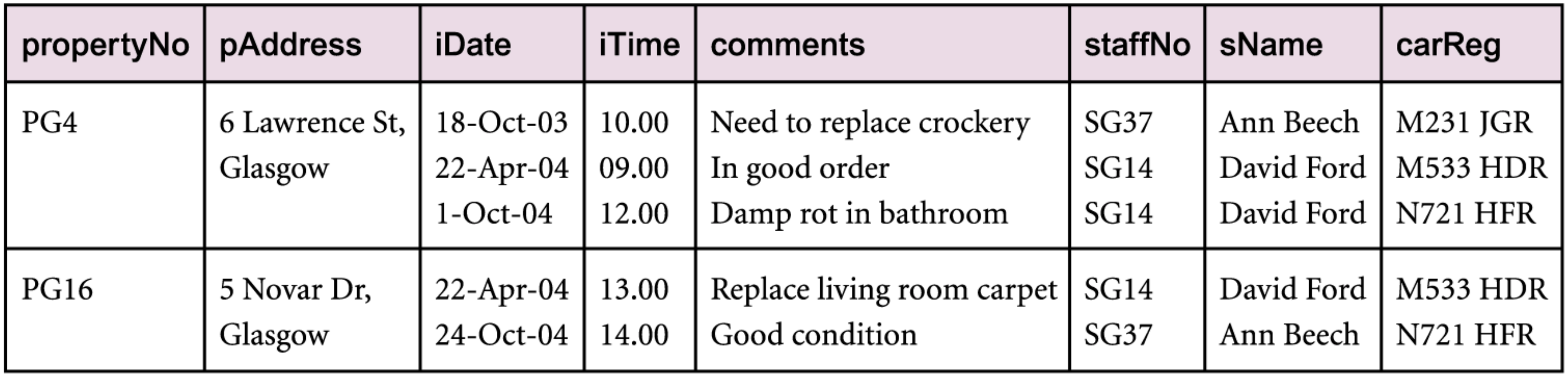
Inspection Date	Inspection Time	Comments	Staff no	Staff Name	Car Registration
18-Oct-03	10.00	Need to replace crockery	SG37	Ann Beech	M231 JGR
22-Apr-04	09.00	In good order	SG14	David Ford	M533 HDR
1-Oct-04	12.00	Damp rot in bathroom	SG14	David Ford	N721 HFR

Page 1

- In this example we extend the *DreamHome* case study to include *property inspection* by members of staff.
- When staff are required to undertake these inspections, they are allocated a company car for use on the day of the inspections.
- However, a car may be allocated to several members of staff as required throughout the working day.
- A member of staff may inspect several properties on a given date, but a property is only inspected once on a given date.

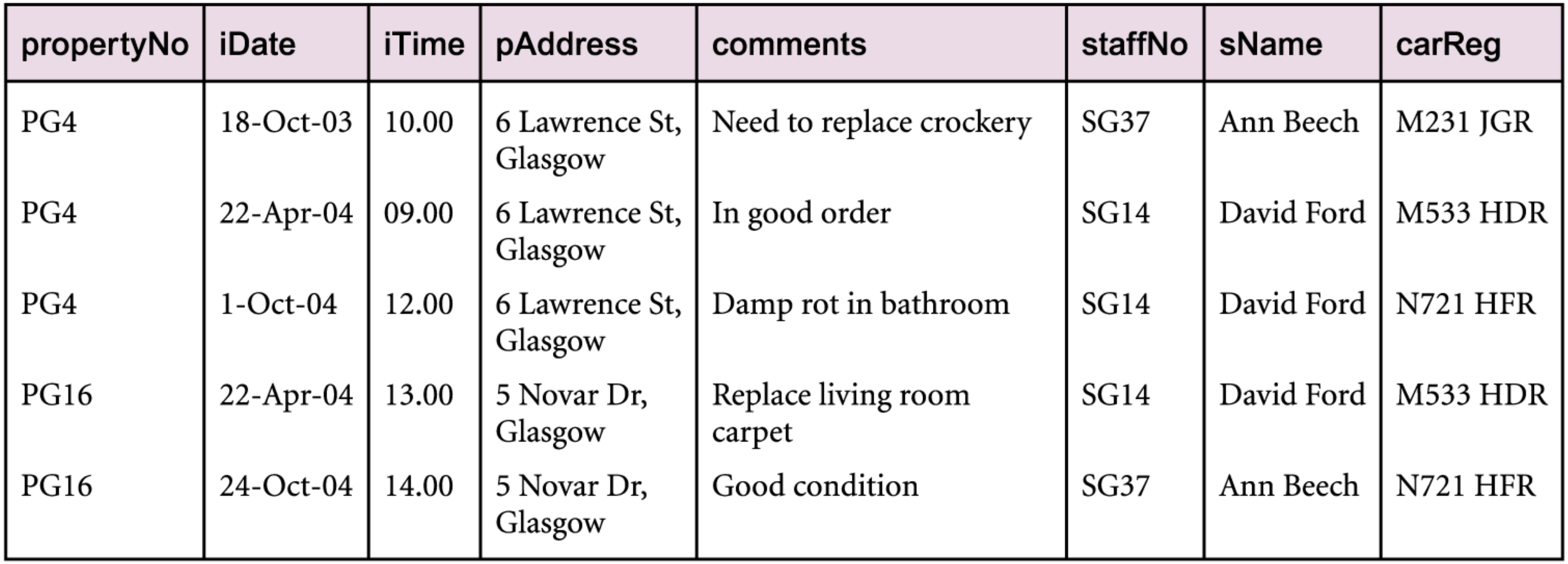
Review of Normalization (UNF to BCNF)

StaffPropertyInspection



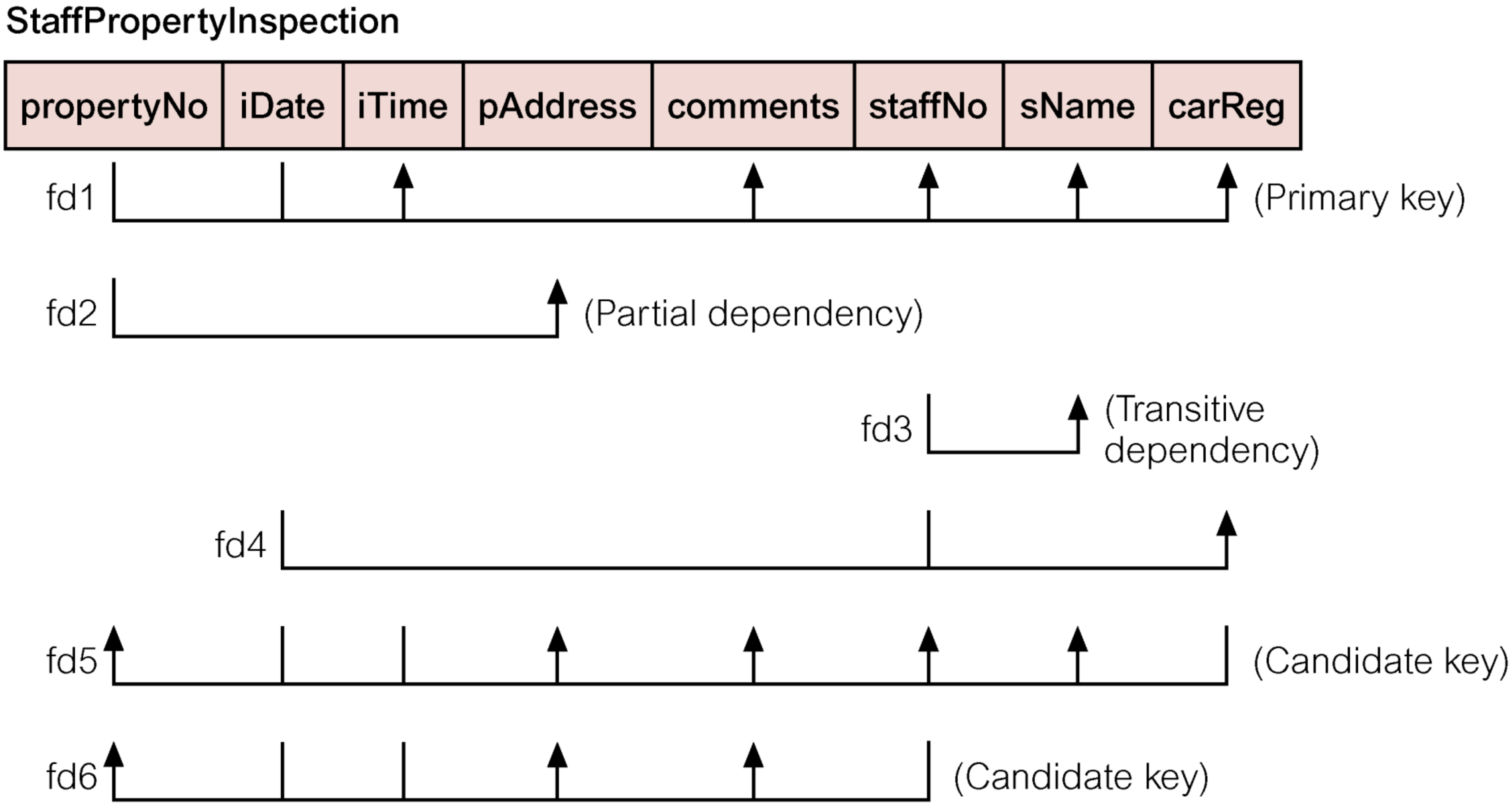
propertyNo	pAddress	iDate	iTime	comments	staffNo	sName	carReg
PG4	6 Lawrence St, Glasgow	18-Oct-03 22-Apr-04 1-Oct-04	10.00 09.00 12.00	Need to replace crockery In good order Damp rot in bathroom	SG37 SG14 SG14	Ann Beech David Ford David Ford	M231 JGR M533 HDR N721 HFR
PG16	5 Novar Dr, Glasgow	22-Apr-04 24-Oct-04	13.00 14.00	Replace living room carpet Good condition	SG14 SG37	David Ford Ann Beech	M533 HDR N721 HFR

StaffPropertyInspection

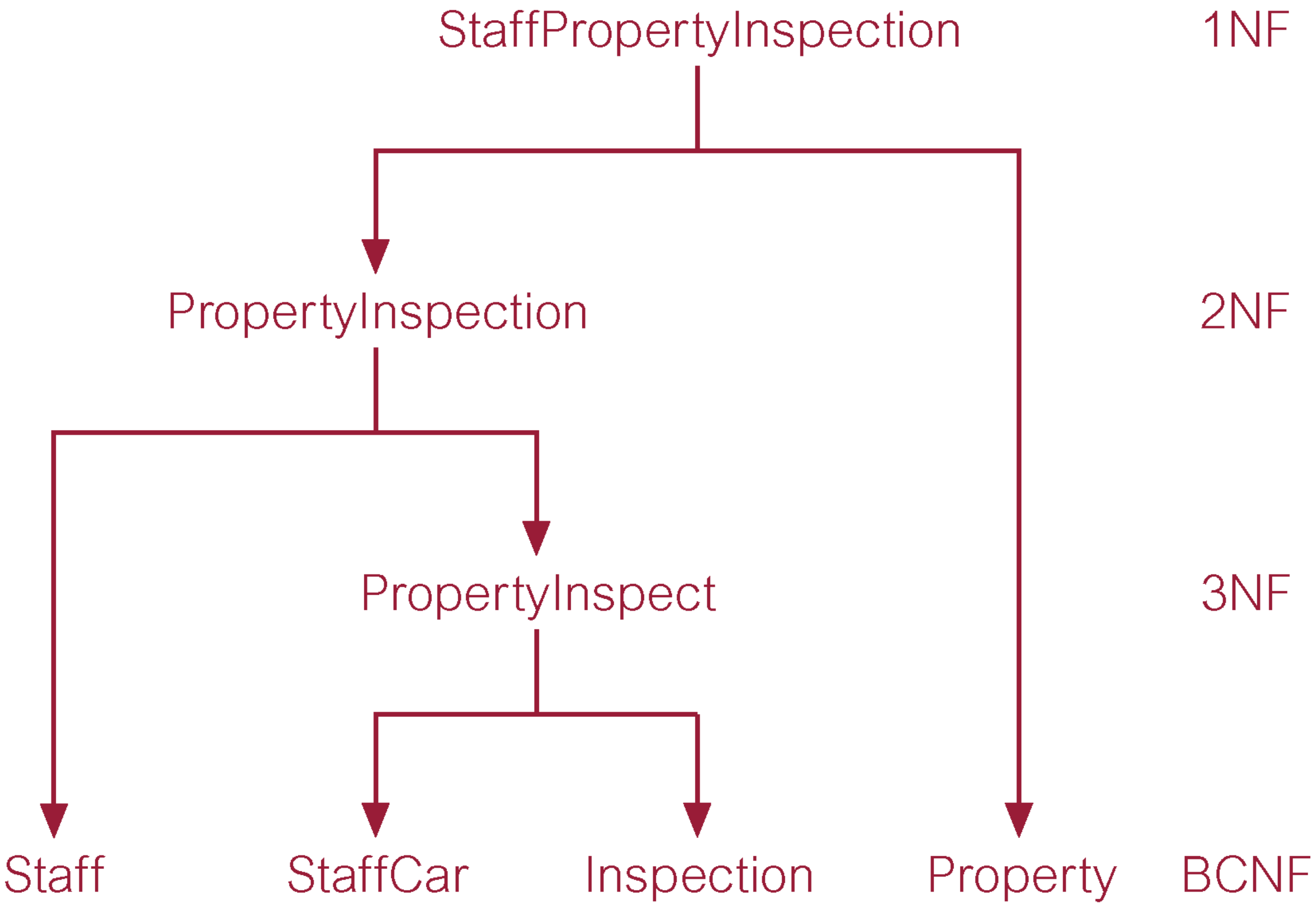


propertyNo	iDate	iTime	pAddress	comments	staffNo	sName	carReg
PG4	18-Oct-03	10.00	6 Lawrence St, Glasgow	Need to replace crockery	SG37	Ann Beech	M231 JGR
PG4	22-Apr-04	09.00	6 Lawrence St, Glasgow	In good order	SG14	David Ford	M533 HDR
PG4	1-Oct-04	12.00	6 Lawrence St, Glasgow	Damp rot in bathroom	SG14	David Ford	N721 HFR
PG16	22-Apr-04	13.00	5 Novar Dr, Glasgow	Replace living room carpet	SG14	David Ford	M533 HDR
PG16	24-Oct-04	14.00	5 Novar Dr, Glasgow	Good condition	SG37	Ann Beech	N721 HFR

Review of Normalization (UNF to BCNF)



Review of Normalization (UNF to BCNF)



StaffPropertyInspection

1NF

PropertyInspection

2NF

PropertyInspect

3NF

Staff

StaffCar

Inspection

Property

BCNF

Fourth Normal Form (4NF)

- Although BCNF removes anomalies due to functional dependencies, another type of dependency called a multi-valued dependency (MVD) can also cause data redundancy.

- Possible existence of multi-valued dependencies in a relation is due to 1NF and can result in data redundancy.

Fourth Normal Form (4NF)

- Multi-valued Dependency (MVD)
  - Dependency between attributes (for example, A, B, and C) in a relation, such that for each value of A there is a set of values for B and a set of values for C. However, the set of values for B and C are independent of each other.
- MVD between attributes A, B, and C in a relation using the following notation:

  A −>> B

  A −>> C

Fourth Normal Form (4NF)

- A multi-valued dependency can be further defined as being trivial or nontrivial.
  - A MVD A −>> B in relation R is defined as being trivial if (a) B is a subset of A *or* (b) $A \cup B = R$.
  - A MVD is defined as being nontrivial if neither (a) nor (b) are satisfied.
  - A trivial MVD does not specify a constraint on a relation, while a nontrivial MVD does specify a constraint.

Fourth Normal Form (4NF)

- Defined as a relation that is in Boyce-Codd Normal Form and contains no nontrivial multi-valued dependencies.

4NF - Example

BranchStaffOwner

branchNo	sName	oName
B003	Ann Beech	Carol Farrel
B003	David Ford	Carol Farrel
B003	Ann Beech	Tina Murphy
B003	David Ford	Tina Murphy

↓

BranchStaff

branchNo	sName
B003	Ann Beech
B003	David Ford

BranchOwner

branchNo	oName
B003	Carol Farrel
B003	Tina Murphy

Student

studentID	sport	subject
45	Football	English
45	Football	Music
45	Tennis	Maths
45	Basketball	Maths
50	Basketball	Maths
50	Tennis	English

- Is student relation in 4NF?

Decomposition Properties

- **Lossless**: Data should not be lost or created when splitting relations up

- **Dependency preservation**: It is desirable that FDs are preserved when splitting relations up

Decomposition Properties

- Normalization to 3NF is always lossless and dependency preserving
- Normalization to BCNF is lossless, but may not preserve all dependencies

- Normalization
  - Removes data redundancy
  - Solves INSERT, UPDATE, and DELETE anomalies
  - This makes it easier to maintain the information in the database in a consistent state

- However
  - It leads to more tables in the database
  - Often these need to be joined back together, which is expensive to do

  - So sometimes (not often) it is worth ‘**denormalizing**’

Denormalization

- You *might* want to denormalize if
  - Database speeds are unacceptable (not just a bit slow)
  - There are going to be very few INSERTs, UPDATEs, or DELETEs
  - There are going to be lots of SELECTs that involve the joining of tables