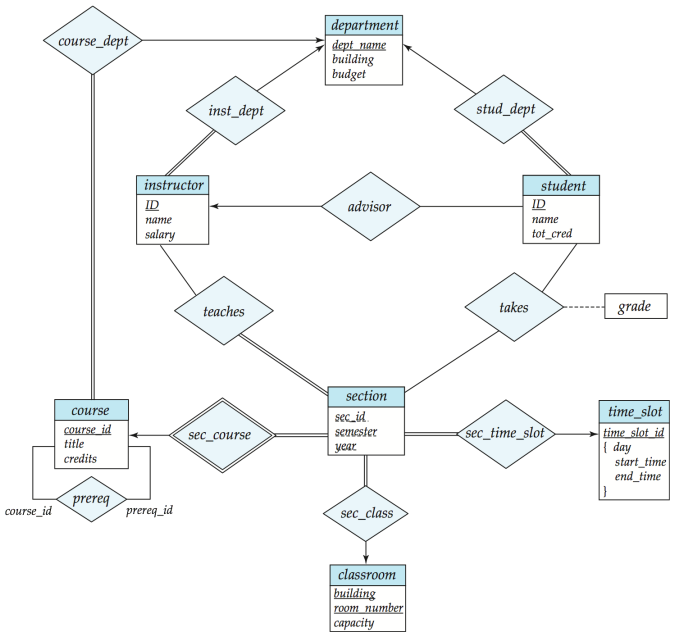
Decompositions

CISC637, Lecture #16

Ben Carterette



Decomposition Example

- Schema:
  - class(course_id, title, credits, sec_id, semester, year, building, room_no, capacity, time_slot_id)
- FDs:
  - course_id → title, dept_name, credits
  - building, room_no → capacity
  - course_id, sec_id, semester, year → building, room_no, time_slot_id
- Candidate keys?
  - (course_id, sec_id, semester, year) is the only candidate key
- Is schema in BCNF?
  - No – 1st & 2nd FDs violate BCNF
- Decomposition based on 1st FD:
  - R1(course_id, title, dept_name, credits)
  - R(course_id, sec_id, semester, year, building, room_no, capacity, time_slot_id)
- Is R1 in BCNF? Yes. Is R in BCNF? No – 2nd FD violates BCNF.
- Decomposition based on 2nd FD:
  - R2(building, room_no, capacity)
  - R(course_id, sec_id, semester, year, building, room_no, time_slot_id)

31

Lossles-Join Test Example

- Schema:
  - class(course_id, title, credits, sec_id, semester, year, building, room_no, capacity, time_slot_id)
- FDs:
  - course_id → title, dept_name, credits
  - building, room_no → capacity
  - course_id, sec_id, semester, year → building, room_no, time_slot_id
- Decomposition based on 1st FD:
  - R1(course_id, title, dept_name, credits)
  - R(course_id, sec_id, semester, year, building, room_no, capacity, time_slot_id)
  - Is this lossless-join?
    - Intersection of R1 and R is course_id
    - course_id is a superkey for R1
- Decomposition based on 2nd FD:
  - R2(building, room_no, capacity)
  - R(course_id, sec_id, semester, year, building, room_no, time_slot_id)
  - Is this lossless-join?
    - Intersection of R2 and R is (building, room_no)
    - (building, room_no) is superkey for R2

32

Dependency-Preserving Test

1. Check if every FD in F holds on either R1 or R2
   - If so, the decomposition is dependency-preserving
   - If not, continue to step 2

2. Run algorithm to check dependency-preserving

```
for each FD f ∈ F:
  let result = left side of f
  do:
    for each Ri in the decomposition:
      let t = (result ∩ Ri)⁺ ∩ Ri
      result = result ∪ t
  until result is unchanged
  if result contains right side of f:
    f is preserved, continue to next FD
  else
    f is not preserved, return failure
```

 33

Dependency-Preserving Test Example

- Schema:
  - class(course_id, title, credits, sec_id, semester, year, building, room_no, capacity, time_slot_id)
- FDs:
  - course_id → title, dept_name, credits
  - building, room_no → capacity
  - course_id, sec_id, semester, year → building, room_no, time_slot_id
- Decomposition based on 1st FD:
  - R1(course_id, title, dept_name, credits)
  - R(course_id, sec_id, semester, year, building, room_no, capacity, time_slot_id)
  - Is this dependency-preserving?
- Decomposition based on 2nd FD:
  - R2(building, room_no, capacity)
  - R(course_id, sec_id, semester, year, building, room_no, time_slot_id)
  - Is this dependency-preserving?

34