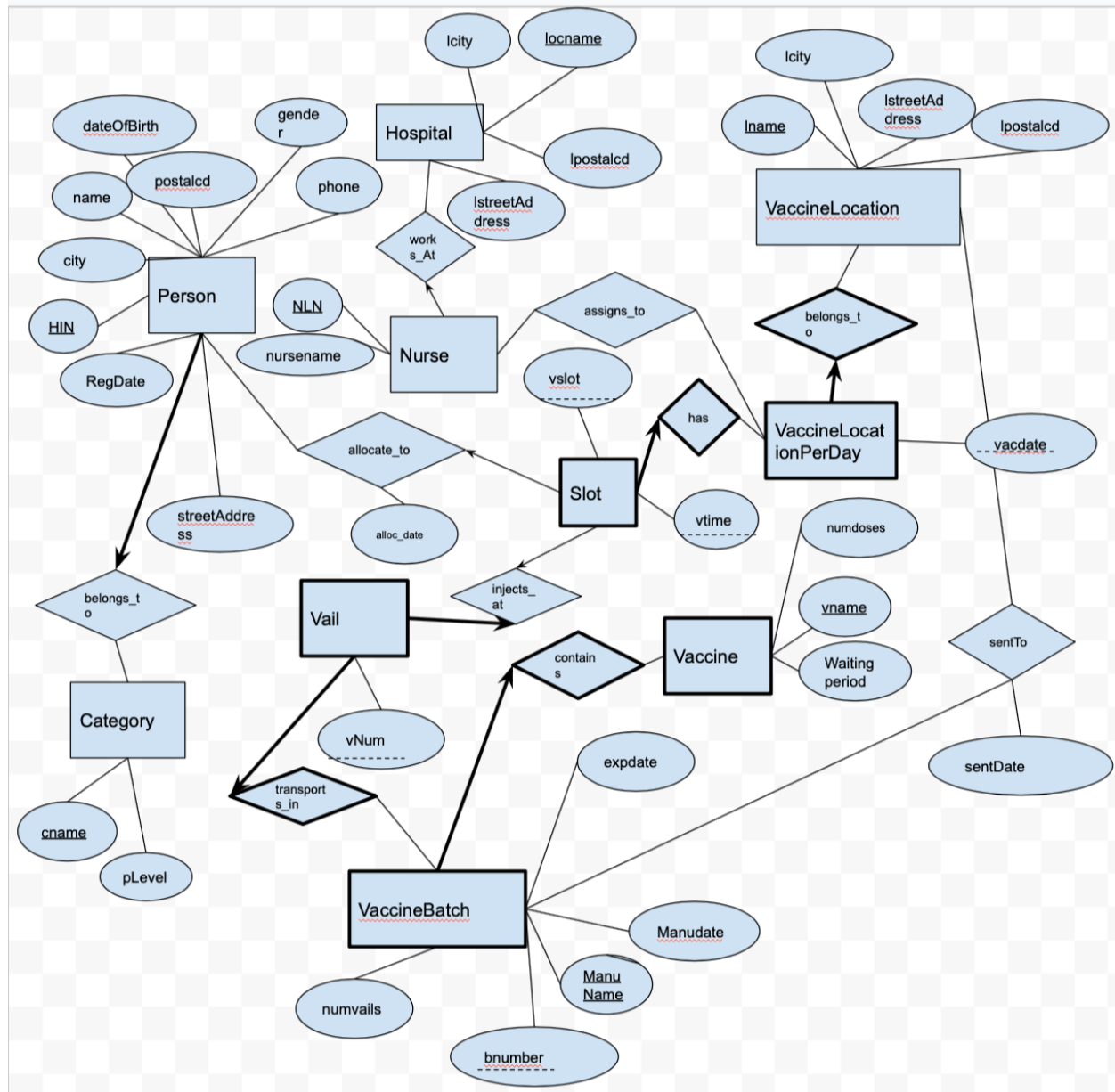


Amelia Cui 260824815 P2

## Part I Relational Model

Updated E/R model:



Updated relational model:

Person (HIN, RegDate, city, name, postalcd, phone, streetAddress, gender, cname)

- cname foreign key referencing relation Category

Category (cname, pLevel)

Nurse (NLN, nursename, locname)

- locname foreign key referencing relation Hospital

Hospital (locname, lcity, lpostalcd, lstreetAddress)

VaccineLocation (lname, lcity, lpostalcd, lstreetAddress)

VaccineLocPerDay (vacdate, lname)

- lname foreign key referencing relation VaccineLocation

Slot (vslot, vtime, vacdate, lname)

- (vacdate, lname) foreign key referencing relation VaccineLocPerDay

Vaccine(vname, waitingPeriod, numdoses)

VaccineBatch (numvails, expdate, bnumber, manudate, vname)

- vname foreign key referencing relation Vaccine

Vail (vnum, bnumber, vname, vslot, vtime, vacdate, lname)

- (bnumber, vname) foreign key referencing relation VaccineBatch
- (vslot, vtime, vacdate, lname) foreign key referencing relation Slot

nurseAssignments (NLN, vacdate, lname)

- NLN foreign key referencing relation Nurse
- (vacdate, lname) foreign key referencing relation VaccineLocPerDay

personAllocation (alloc\_date, HIN, vslot, vtime, vacdate, lname)

- HIN foreign key referencing relation Person
- (vslot, vtime, vacdate, lname) foreign key referencing relation Slot

batchStorage (bnumber, vname, lname, sentDate)

- (bnumber, vname) foreign key referencing relation VaccineBatch
- lname foreign key referencing relation VaccineLocation

## Part II Pending constraints

- When assign a person a category, we have to manually check the category stored is the one with highest priority (pLevel) if a person belongs to several categories
- When a slot is created at a VaccineLocPerDay, the personAllocation must be created based on the pLevel for the category that cname is associated with the person (connected to personAllocation by HIN)
- We have to manually make sure that when a vail injection is created, at least one nurse assignment must exist for the vaccineLocPerDay associated with the slot where the injection happened.
- We have to manually check that the sentDate in batchStorage must be before the vail injection slot date and time that the vails inside each VaccineBatch is injected
- We have to check that the number of vail relations for each VaccineBatch must be less that the numvails attributes in this VaccineBatch
- For the personAllocation relation for each person, if there are more than one slot associated with him/her, the vail injection happened at these slots must belong to same vaccine name, which can be acquired by VaccineBatch relation, and must be less that the numdoses for that vaccine. Additionally, the vacdate difference between these slots must be greater than the waitingPeriod for the vaccine.
- For each vail injection, the date of the slot associated with it must before the expdate of the vaccineBatch associated with it.
- The vail can only be created if the personAllocation for the slot associated with the vail is not null.
- The database system does not prevent 2 people choosing the same slot, which should not be allowed in reality.
- Assume that the person must be successfully injected once personAllocation is created (there cannot be invalid personAllocation); the instance of personAllocation has to deleted from database if a person wants to cancel the appointment

- The regDate for each person must be before the data and time for the slot that is associated with the personAllocation for that person (can be accessed by the HIN for a given person)
- For each slot associated with a given vail injection, the number of personAllocation associated with the vaccineLocPerDay that is associated with the slot must be less than or equal to the number of nurse assignments associated with that vaccineLocPerDay.

### Part III SQL Queries

a) select vslot, vtime, vacdate, lname  
from Slot  
where lname = 'Jewish General' and vacdate = '2021-03-20'  
except  
select vslot, vtime, vacdate, lname  
from personAllocation  
where lname = 'Jewish General' and vacdate = '2021-03-20'  
;  
  
db2 => select vslot, vtime, vacdate, lname  
from Slot  
where lname = 'Jewish General' and vacdate = '2021-03-20'  
except  
select vslot, vtime, vacdate, lname  
from personAllocation  
[where lname = 'Jewish General' and vacdate = '2021-03-20'db2 (cont.) => db2 (con]  
t.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => ;

VSLLOT	VTIME	VACDATE	LNAME
1	13:30:08	03/20/2021	Jewish General
4	13:30:09	03/20/2021	Jewish General

b) select expdate  
from VaccineBatch  
where (bnumber, vname) in (  
select Vail.bnumber, Vail.vname  
from Vail, personAllocation  
where Vail.vacdate = '2021-02-06'  
and Vail.vacdate = personAllocation.vacdate  
and Vail.vtime = personAllocation.vtime  
and Vail.lname = personAllocation.lname  
and Vail.vslot = personAllocation.vslot  
and personAllocation.HIN in(  
select HIN  
from Person  
where name = 'Jane Doe'  
)  
)  
;  
;

```

db2 => select expdate
      from VaccineBatch
        where (bnumber, vname) in (
              select Vail.bnumber, Vail.vname
              from Vail, personAllocation
              where Vail.vacdate = '2021-02-06'
              and Vail.vacdate = personAllocation.vacdate
              and Vail.vtime = personAllocation.vtime
              and Vail.lname = personAllocation.lname
              and Vail.vslot = personAllocation.vslot
              and personAllocation.HIN in(
                select HIN
                from Person
                where name = 'Jane Doe'
              )
            )
;db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.)
.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) =>

EXPDATE
-----
01/04/2028

1 record(s) selected.

```

- c) select COUNT(HIN) as totalNum  
 from personAllocation  
 where vacdate = '2021-02-06'  
 and (vacdate, vslot, vtime, lname) in  
 (  
 select vacdate, vslot, vtime, lname  
 from Slot  
 where Slot.lname in (  
 select lname  
 from VaccineLocation  
 where lcity = 'Montreal'  
 )  
 )  
 ;

```

db2 => select COUNT(HIN) as totalNum
      from personAllocation
      where vacdate = '2021-02-06'
      and (vacdate, vslot, vtime, lname) in
      (
        select vacdate, vslot, vtime, lname
        from Slot
        where Slot.lname in (
          select lname
          from VaccineLocation
          where lcity = 'Montreal'
        )
      )
;db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.)
.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) =>

TOTALNUM
-----
5

1 record(s) selected.

```

- d) select name, phone, HIN  
 from Person  
 where HIN in (  
 select HIN  
 from (  
 select HIN, COUNT(\*) as counts  
 from personAllocation, Vail  
 where personAllocation.vacdate = Vail.vacdate

```

        and personAllocation.lname = Vail.lname
        and personAllocation.vtime = Vail.vtime
        and personAllocation.vslot = Vail.vslot
        and Vail.vname = 'Pfizer-BioNTech'
    group by personAllocation.HIN
    ) as temp
where temp.counts =1
and HIN in (
    select HIN
    from personAllocation
    where vacdate < '2021-02-01'
)
)
;

db2 => select name, phone, HIN
from Person
where HIN in (
    select HIN
    from (
        select HIN, COUNT(*) as counts
        from personAllocation, Vail
        where personAllocation.vacdate = Vail.vacdate
        and personAllocation.lname = Vail.lname
        and personAllocation.vtime = Vail.vtime
        and personAllocation.vslot = Vail.vslot
        and Vail.vname = 'Pfizer-BioNTech'
        group by personAllocation.HIN
    ) as temp
    where temp.counts =1
    and HIN in (
        select HIN
        from personAllocation
        where vacdate < '2021-02-01'
    )
)
;db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.)
.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.)
(cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) =>

NAME                PHONE                HIN
-----
Jesus Poole          7153019183          3917587088
Guadalupe Wallace   4983136624          5025429412

2 record(s) selected.

```

- e) select count(\*) as counts, cname  
from Person  
where HIN in (  
 select distinct HIN  
 from personAllocation  
)

group by cname

;

//assume that the question was asking for all records of person that are injected or planned to be injected for each category. (Or we may need to check the date information for personAllocation to be before current time)

COUNTS	CNAME
--------	-------

7 record(s) selected.

- db2 =>

- Screenshot of selecting 5 records from nurses that are employed at “Jewish General”

Query:

```
select *
from mtlnurses
where locname='Jewish General'
limit 5
```

```
db2 => select *
from mtlnurses
where locname='Jewish General'
limit 5
;db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => ;
```

NLN	NURSENAME	LOCNAME	LCITY	LPOSTALCD	LSTREETADDRESS
5670572612	Faye Parker	Jewish General	Montreal	L4A456	Ashton Loan
3152764765	Mack Brady	Jewish General	Montreal	L4A456	Ashton Loan
3359852020	Victoria Edwards	Jewish General	Montreal	L4A456	Ashton Loan
12345678	Mike Brady	Jewish General	Montreal	L4A456	Ashton Loan
87654321	Victory Edwards	Jewish General	Montreal	L4A456	Ashton Loan

5 record(s) selected.

```
db2 =>
```

- Screenshot of the insert command

```
db2 => INSERT INTO mtlnurses(NLN, nursename, locname, lcity, lpostalcd, lstreetAddress) VALUES ('123', 'testName', 'location', 'montreal', 'H3C', 'streetA');
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0150N The target fullselect, view, typed table, materialized query table,
range-clustered table, or staging table in the INSERT, DELETE, UPDATE, MERGE,
or TRUNCATE statement is a target for which the requested operation is not
permitted. SQLSTATE=42807
db2 =>
```

- Explanation:

As shown above, the insert cannot be successfully executed. It is because the view we created is not a valid table. The view is just created for our convenience to get certain combinations of data in database. It is not created to store data, instead, it is created based on “actual” table in database, therefore, we cannot perform “insert” operation on a view.

## Part V: Check Constraints

- Screenshot of the check constraint

```
db2 => ALTER TABLE VaccineBatch
ADD CONSTRAINT constraintA
CHECK(expdate > manudate);db2 (cont.) => db2 (cont.) => ;
DB20000I The SQL command completed successfully.
db2 =>
```

- Attempt to insert an invalid record

```
db2 => INSERT INTO VaccineBatch (numvails,expdate,bnumber,manudate,vname) VALUES ('8', '2017-01-10', 'E', '2020-12-10', 'Moderna');
DB21034E The command was processed as an SQL statement because it was not a
valid Command Line Processor command. During SQL processing it returned:
SQL0803N One or more values in the INSERT statement, UPDATE statement, or
foreign key update caused by a DELETE statement are not valid because the
primary key, unique constraint or unique index identified by "1" constrains
table "WCUI6.VACCINEBATCH" from having duplicate values for the index key.
SQLSTATE=23505
db2 =>
```

It is a failure because it violates the constraint that the expdate of a VaccineBatch must be past its manudate.