ER-Relational Notes

Eugene Wu
Fall 2015

Read the textbook!

It is actually quite good on this topic!

Relational Constraints

- Gives the DBMS a list of consistency checks
- It is run with respect to whatever data that exists in the database
  - DBMS doesn't understand *anything* about what application wants or intends or "should" have
  - It simply goes through each constraint one by one and checks them against the data in the database

Relational Constraints

Domain constraints
Foreign key constraints
Unique constraints
Primary key constraints
NOT NULL constraints
CHECK constraints

ER → Relational translation

Translate entities and relationships into relational tables
Translate ER constraints into relational constraints
The translation is correct if for *any* database instance:
- Constraints violated in ER are violated in relational
- Constraints violated in relational are violated in ER
- If ER doesn't violate, neither should relational
- If relational doesn't violate, neither should ER

Note: some translations are not possible

What constraints are translatable?

A – ab – B
A → ab – B
A → ab ← B
A ➔ ab – B
A ➔ ab ➔ B

And the mirror images

How to check a translation?

1. Come up with data that satisfies the ER constraints and check that they don't violate relational version
2. For each ER constraint
   1. Come up with data that violated the ER constraint. Does relational version identify the violation given the same data?
3. Vice Versa

Non-exhaustive examples (with bastardized notation!)

And how to check them

A – ab – B

A(
a int primary key,
name text
)

a	name
1	eve
2	bob

ab(
a references A(a)
b references B(b)
)

a	b
1	3
1	4
2	3

B(
b int primary key
)

b
3
4

A → ab – B

A(
a int primary key,
name text
)

a	name
1	eve
2	bob

ab(
a references A(a)
b references B(b)
unique(a)
)

a	b
1	3
2	3

B(
b int primary key
)

b
3
4

This database satisfies ER constraints.
Thus it should satisfy the relational constraints

A → ab – B

A(
a int primary key,
name text
)

a	name
1	eve
2	bob

ab(
a references A(a)
b references B(b)
unique(a)
)

a	b
1	3
2	null

B(
b int primary key
)

b
3
4

This database satisfies ER constraints.
Thus it should satisfy the relational constraints

A → ab – B

A(
a int primary key,
name text
)

a	name
1	eve
2	bob

ab(
a references A(a)
b references B(b)
unique(a)
)

a	b
1	3
1	4
2	3

B(
b int primary key
)

b
3
4

This database violates ER constraints.
Thus, it should violate the relational constraints

A → ab – B

A(
a int primary key,
name text,
b references B(b)
)

B(
b int primary key
)

a	name	b
1	eve	3
2	bob	3

b
3
4

This database satisfies ER constraints.
Thus should satisfy relational constraints

A → ab – B

A(
a int primary key,
name text
)

ab(
a references A(a)
b references B(b)
unique(b)
)

B(
b int primary key
)

a	name
1	eve
2	bob

a	b
1	3
2	3

b
3
4

This database satisfies ER constraints.
BUT it doesn't satisfy the relational constraints. FAIL

A → ab – B

Can this be expressed relationally?

A(
a int primary key,
name text
)

ab(
a references A(a)
b references B(b)
UNIQUE (a)
)

B(
b int primary key
)

a	name
1	eve
2	bob

a	b
1	null
2	3

b
3
4

This database violates ER constraints.
BUT it does satisfy the relational constraints. FAIL

A → ab – B

Can this be expressed relationally?

A(
a int primary key,
name text
)

ab(
a references A(a)
b references B(b) NOT NULL
UNIQUE (a)
)

B(
b int primary key
)

a	name
1	eve
2	bob

a	b
1	null
2	3

b
3
4

This database violates ER constraints.
And it violates relational constraints. OK

A → ab – B

Can this be expressed relationally?

A(
a int primary key,
name text
)

ab(
a references A(a)
b references B(b) NOT NULL
UNIQUE (a)
)

B(
b int primary key
)

a	name
1	eve
2	bob

a	b
2	3

b
3
4

This database violates ER constraints.
BUT it satisfies relational constraints. FAIL.

A → ab – B

A(
a int primary key,
name text,
b references B(b) NOT NULL
)

B(
b int primary key
)

a	name	b
1	eve	3
2	bob	3

b
3
4

This database satisfies ER constraints.
Also satisfies relational constraints. OK

A ➔ ab – B

A(
a int primary key,
name text,
b references B(b) NOT NULL
)

B(
b int primary key
)

a	name	b
1	eve	null
2	bob	3

b
3
4

This database violates ER constraints.
Also violates relational constraints. OK

A ➔ ab – B

A(
a int primary key,
name text,
b references B(b) NOT NULL
)

B(
b int primary key
)

a	name	b
2	bob	3

b
3
4

This database satisfies ER constraints.
Also satisfies relational constraints. OK

Some tips

There are not that many ways to express a relationship between A and B for any combination of constraints

A(…), ab(...), B(...)
AB(...), B(...) // A and ab are merged
A(...), AB(...) // ab and B are merged
AB(...) // all three merged

You should understand…

Why the following cannot be expressed or cannot be expressed without redundancy

A == ab – B
A == ab == B