Information Systems 2
1. Modelling Information Systems I: Databases

Lars Schmidt-Thieme
Artus Krohn-Grimberghe
http://www.ismll.uni-hildesheim.de/

Recap

- Why databases?
- 3 levels of data models
- E/R modeling
- Relational model

Why databases?

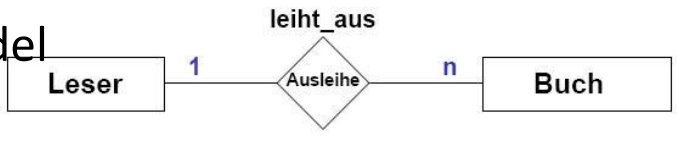
- Persistent data storage
- Integrated (Students, Courses, …)
- Shared
- Integrity
- Data standards can be enforced
- Eliminate reduncancy
- Eliminate inconsistency

Conceptual, Representation, and Physical Data Models

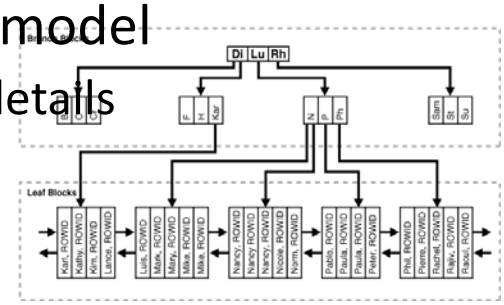
→ Nomenclature varies in literature!

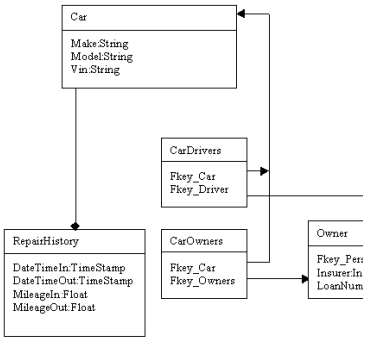
- Conceptual model
  - Business objects
  - Entity Relationship model
- Representation model
  - Relational model
  - SQL
- „Physical“ model
  - Storage details

Customer cust = **new Customer()**;



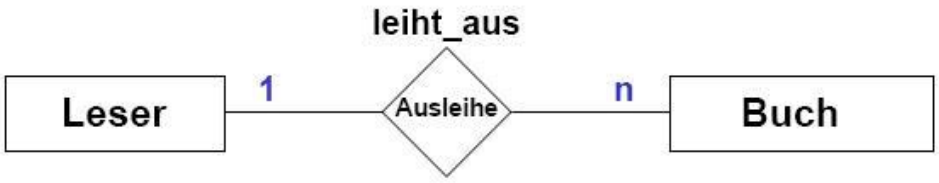
SELECT… FROM… WHERE…



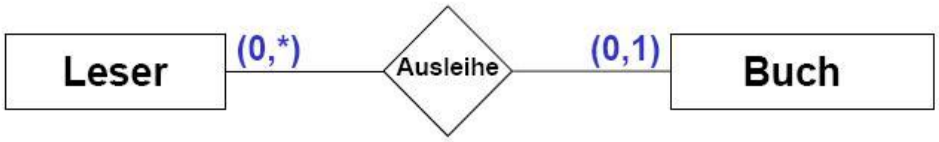


Entity Relationship model

- Entities, weak entities, attributes, keys, relations
- Cardinality
  - 1, n, m notation



  - (min,max) notation



Let‘s move on

- We have seen a way to model our mini-world conceptually, let‘s now think about how to store the belonging data
  - Hierarchic systems – outdated
  - Network systems – outdated
  - Relational model – state of the art
    - Describes data structure (following slides)
    - Data manipulation
    - Data integrity

Basic Concepts

The Relational model organizes data in tables.

Relational model	common sense table
attribute	column
attribute domain	value domain of a column
tuple	row
relation	table
null value	cells without entry (missing values, unappropriate attributes)
key	set of columns which values uniquely identify a row
primary key	key usually used for identifying rows

KUNDE

Kundennr	Name	Geburtstag
1	Frank Müller	20.11.1980
2	Fred Schmidt	6.6.1972
3	Heribert Mayer	11.1.1954
4	Frank Müller	3.7.1978

Basic Concepts / Foreign Keys

Foreign Key:
an attribute (or set of attributes) that contains the key value of another relation.

The value domain of the foreign key must be the same as the value domain of the key of the **referenced relation**.

For each tuple of the **referencing relation** the value of the foreign keys must occur among the values of the key attribute of the referenced relation or be null (**referential integrity**).

KUNDE

Kundennr	Name	Geburtstag
1	Frank Müller	20.11.1980
2	Fred Schmidt	6.6.1972
3	Heribert Mayer	11.1.1954
4	Frank Müller	3.7.1978

BESTELLUNG

Rechnungsnr	Rechnungsdatum	Rechnungsbetrag	Kundennr
1099	12.2.2000	2099,-	2
1100	12.2.2000	589,-	1
1101	13.2.2000	4490,-	3
1102	15.2.2000	3349,-	2
1103	18.2.2000	10500,-	5

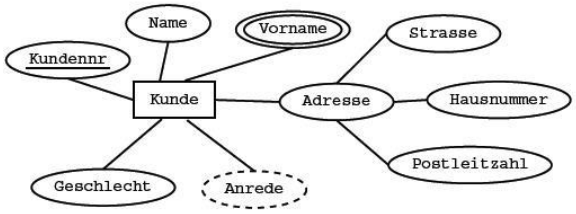
Mapping an ER Model to a Relational Model / Regular Entity

Regular entity with ⟶ relation (**entity relation**) with
- simple attributes ⟶ attribute
- complex attributes ⟶ one attribute for each component (neglect structure)
- key ⟶ primary key (select one)
- keys ⟶ secondary keys (all other)

Multi-valued attributes are mapped to an own relation that contains

- the foreign key of the entity relation and
- the value of the attribute.

Together they define the primary key of the new relation.



KUNDE

Kundennr	Name	Geschlecht	Strasse	Hausnummer	Postleitzah

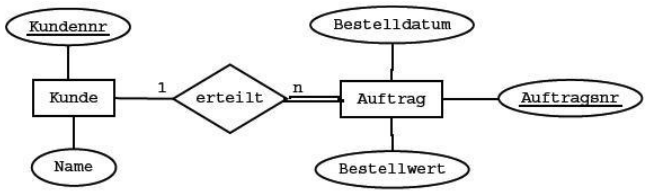
KUNDENVORNAMEN

Kundennr	Vorname

Mapping an ER Model to a Relational Model / Binary Relation 1:n

Binary relation 1:n ⟶ add a foreign key
⟶ referencing the entity relationship on the 1-side
⟶ to the entity relationship on the n-side.

All attributes of the relation are added to the entity relationship on the n-side.



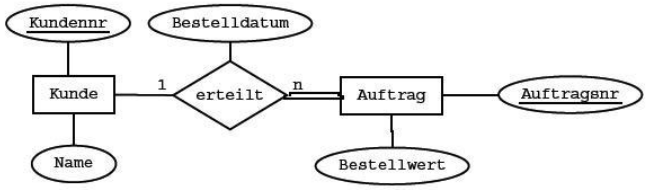
KUNDE

Kundennr	Name

AUFTRAG

Auftragsnr	Bestelldatum	Bestellwert	Kundennr

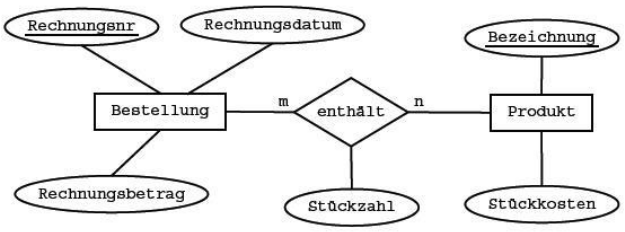
variant:



Mapping an ER Model to a Relational Model / Binary Relation n:m

Binary relation n:m $\longrightarrow$ relation (**relationship relation**) with
- $\longrightarrow$ foreign key to the entity relation on the n-side and
- $\longrightarrow$ foreign key to the entity relation on the m-side
- $\longrightarrow$ (jointly defining the primary key)

All attributes of the relation are mapped to attributes of the relationship relation.



BESTELLUNG

Rechnungsnr	Rechnungsdatum	Rechnungsbetrag

PRODUKT

Bezeichnung	Stückkosten

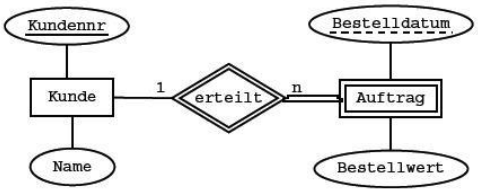
POSTEN

Rechnungsnr	Produktbezeichnung	Stückzahl

Mapping an ER Model to a Relational Model / Binary Relation n:m

Weak entity $\longrightarrow$ relation (**entity relation**) with
- $\longrightarrow$ foreign key to the identifying entity relation
- $\longrightarrow$ (jointly with the partial key defining the primary key)

All attributes are mapped as for regular entities.



KUNDE

Kundennr	Name

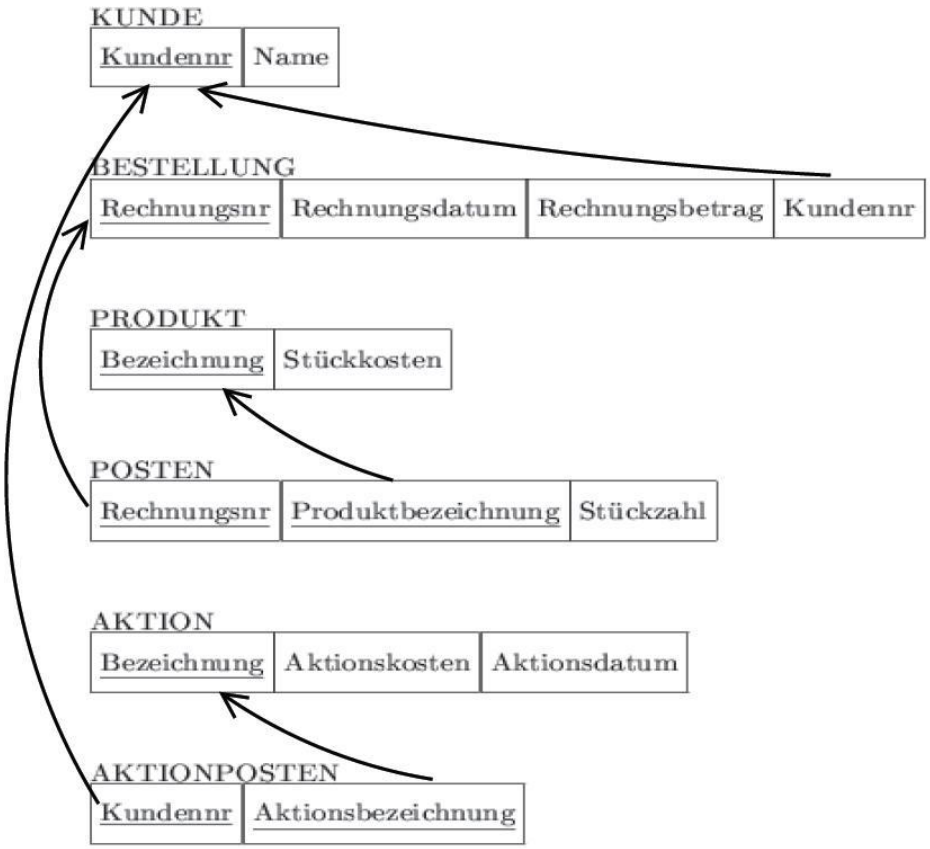
AUFTRAG

Kundennr	Bestelldatum	Bestellwert

Mapping an ER Model to a Relational Model / Summary

ER model	Relational model
entity	relation (**entity relation**)
relation, binary, 1:1	add foreign key to one of the participating entity relations
relation, binary, 1:n	add foreign key to the participating entity relation on the n-side
relation, binary, n:m	relation (**relationship relation**) with 2 foreign keys
relation, n-ary	relation (**relationship relation**) with n foreign keys
attribute, simple	add attribute to relation
attribute, complex	add attributes to relation, one for each component
attribute, multi-valued	relation with foreign key
key	primary or secondary key

Mapping an ER Model to a Relational Model / Example



Relational model – data manipulation

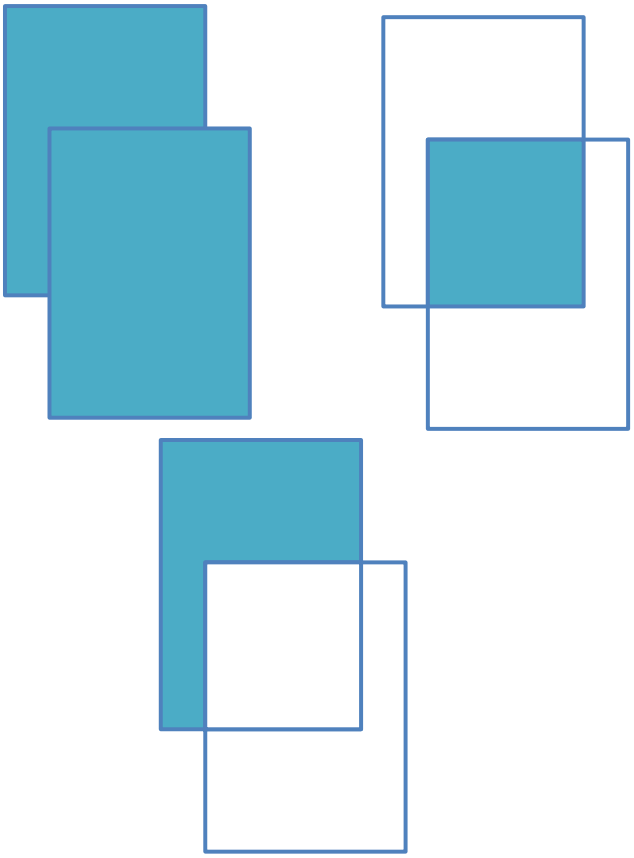
- Two languages defined
  - Relational Algebra
  - Relational Calculus
- Neither is navigating nor record-based but both are set-based and descriptive
- Mathematical foundation
- Relational closure
  - Lecture Diskrete Methoden (first semester)

Relational Algebra

- Collection of operators that take relations as their operands and return a relation as their result
  - Nesting relational operators of arbitrary complexity possible
  - Relation type inference rules (esp. for „header“)

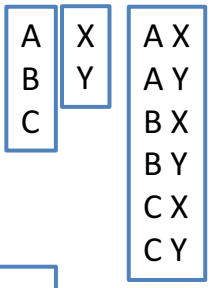
Relational operators

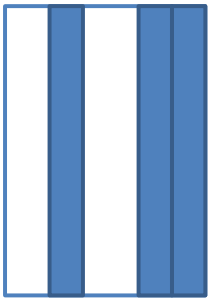
- Union
  - Rel. of same type
  - Remove dupes
  - a UNION b
- Intersect
  - a INTERSECT b
- Difference
  - a MINUS b



Relational operators (cont‘d)

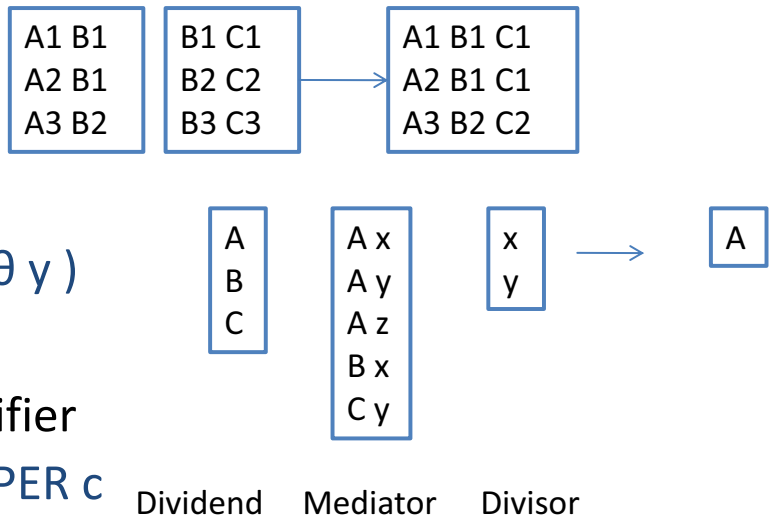
- Product
  - Cartesian product
  - Needs renaming of dupe attribute!
  - a TIMES b
- Restrict
  - Sometimes called Theta-Restrict
  - a WHERE x θ y
- Project
  - „vertical“ subset of relation
  - a {x, y, z}





Relational operators (cont'd)

- (Natural) Join
  - Theta-join
  - (a TIMES b) WHERE x θ y
  - a JOIN b ON (x θ y)
- Divide
  - Universal quantifier
  - a DIVIDED BY b PER c
- Rename
  - (a RENAME City AS aCity)



Buch	InvNr	Titel	ISBN	Autor
	0007	Dr. No	3-125	James B.
	1201	Objektbanken	3-111	Heuer
	4711	Datenbanken	3-765	Vossen
	4712	Datenbanken	3-891	Ullman
	4717	Pascal	3-999	Wirth

- (1) Get Name from Ausleihe

Ausleihe	InvNr	Name
	4711	Meyer
	1201	Schulz
	0007	Müller
	4712	Meyer

- (2) Get InvNr and ISBN from Buch
- (3) Get all data from Buch except where Autor is „Wirth"
- (4) Combine Buch and Ausleihe
- …

Not (yet) covered

- Relational Algebra – integrity
- Relational Calculus
- SQL (tomorrow!)
- XML (in two weeks)
- Impedance mismatch
- Normalization