Relational Model

COURSE 6: Databases

Relational model

Relational model

- Codd rules → Este un DBMS relațional? To what degree?

- **Rule 0.** Un sistem trebuie să gestioneze bazele de date în întregime prin capacitățile sale relaționale.

Relational Integrity constraints

RELATIONS

OPERATORS

Relational model

- Codd rules → Este un DBMS relațional? To what degree?

- Regula 1. Toate informațiile dintr-o bază de date relațională sunt reprezentate explicit la nivel logic, într-un singur mod -- prin valori în tabele.

- Regula 2. Fiecare valoare atomică dintr-o bază de date relațională este garantată a fi accesibilă logic referind o combinație de nume de tabel, valoare cheie primară și nume de coloană.

    - ROWID -- Oracle

ROWID

- The data object number of the object.
- The data block in the data file in which the row resides.
- The position of the row in the data block (first row is 0).
- The data file in which the row resides (first file is 1). The file number is relative to the tablespace.

Relational model

- Regula 3. Valorile nule sunt acceptate în SGBD complet relațional pentru reprezentarea informațiilor lipsă și a informațiilor inaplicabile, într-un mod sistematic, independent de tipul de date.

- Regula 4. **data catalog:** Descrierea bazei de date este reprezentată la nivel logic în același mod ca și datele obișnuite, astfel încât utilizatorii autorizați să poată aplica același limbaj relațional interogării sale ca și datelor obișnuite.

Relational model

- Regula 5. Regula limbajului de acces: Într-un sistem relațional trebuie să existe cel puțin un limbaj de accesare a datelor, care să asigure următoarele operații:
  - definirea tabelelor de bază şi a tabelelor virtuale (vederilor) CREATE, ALTER, DROP,
  - manipularea şi interogarea datelor (atât interactiv cât şi prin program) INSERT UPDATE, DELETE, SELECT
  - definirea restricțiilor de integritate, CONSTRAINT
  - autorizarea accesului la date, ROLES, PRIVILEGES
  - delimitarea tranzacțiilor. COMMIT, ROLLBACK

Relational model

- Regula 6. Toate vizualizările care sunt teoretic actualizabile sunt, de asemenea, actualizate de sistem.

- Regula 7. O relație de bază sau o relație derivată este tratată ca un singur operand Această regulă se aplică nu numai recuperării datelor, ci și inserării, actualizării și ștergerii datelor.

  - insert select, merge

Relational model

- Rule 8. Physical data independence:

  Application programs and terminal activities remain logically unimpaired whenever any changes are made in either storage representations or access methods.

- Rule 9. Logical data independence:

  Application programs and terminal activities remain logically unimpaired when information-preserving changes of any kind are made to the base tables.

- Rule 10. Integrity independence:

  Integrity constraints specific to a particular relational data base must be definable in the relational data sublanguage and storable in the catalog, not in the application programs.

Relational model

- Rule 11. Distribution independence:

The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only.

- Rule 12. The nonsubversion rule:

Dacă un sistem relațional are un limbaj de nivel scăzut (o singură înregistrare la un moment dat), acel nivel scăzut nu poate fi folosit pentru a submina sau ocoli regulile și constrângerile de integritate exprimate în limbajul relațional de nivel superior (înregistrări multiple la un moment dat).

Relational model

Relational Integrity constraints

RELATIONS

OPERATORS

Relational model

- Database = collection of RELATIONS
  - relation in relational model ≠ relationship in ERD.
  - relation in relation model < -- > table with lines and columns

- Relation Schema: A relation schema represents the name of the relation with its attributes.

- Attribute domain – Each attribute has some pre-defined values.

Relational Integrity constraints

RELATIONS

OPERATORS

- Relational schema $R(A_1, A_2, \ldots, A_n)$
- $R \subset D_1 \times D_2 \times \cdots \times D_n$, D_i $domain$
- Example
  - Participant(participant_id, last_name, first_name)
  - A1 - - participant_id D1 - - integer size 6
  - A2 - - last_name D2 - - string, length 20
  - A3 - - first_name D3 - - string, length 20

Relational Integrity constraints

RELATIONS

OPERATORS

- Domain constraints
  - “the value of each attribute must be unique”, specifies data types: integers, real numbers, characters, Booleans; variable length for strings, numbers etc.
- Key constraint
  - Unique + not null -- PK
- Referential integrity constraints
  - the value of a FK is null or it corresponds to the value of a PK.

Relational Integrity constraints

RELATIONS

OPERATORS

- UNION, INTERSECT, PRODUCT, DIFFERENCE

- PROJECT
- SELECT
- JOIN
- DIVISION

Converting ERD into RM

Rules for entities

- Strong entities → independent tables
  - PK doesn’t contain foreign keys.

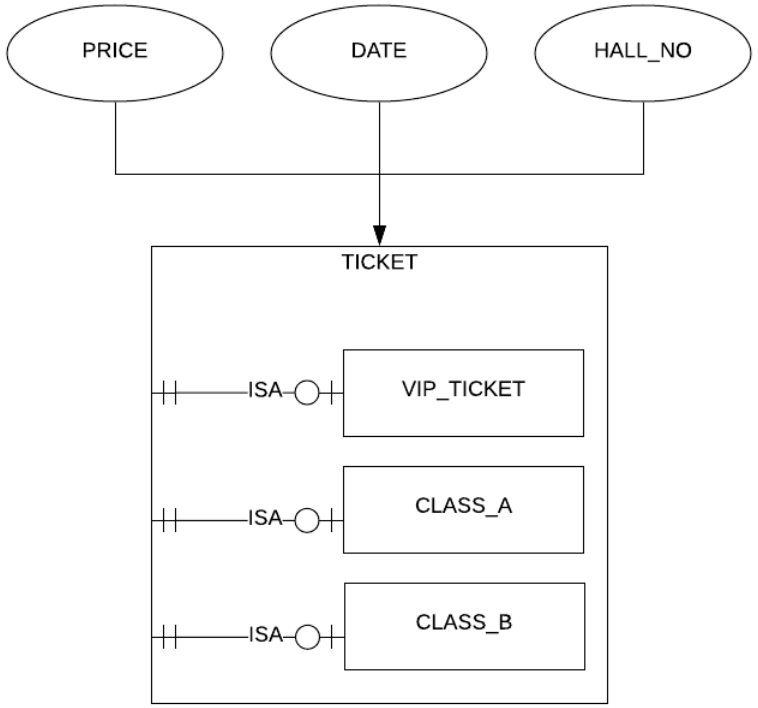
- Weak entities → table
  - PK contains the key of the related strong entity and one or more key attributes.

- Sub-entities → one ore more tables/ Boolean attribute, /type_attribute
  - PK of a subentity may also represent a FK.

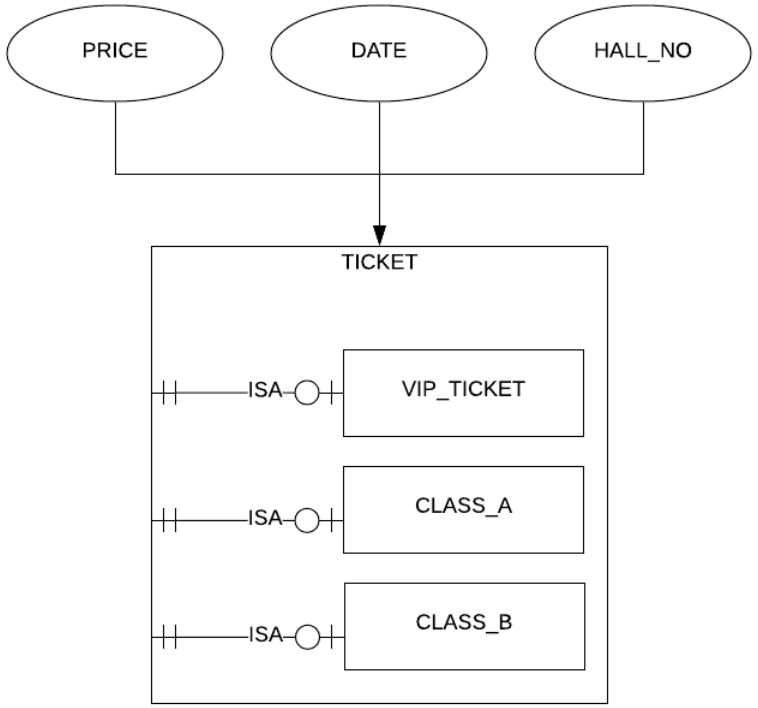
Rules for entities strong – weak entity

AIRPLANE ($airplane\_id$, …)
SEAT ($airplane\_id, seat\_id$, …, observations)

Rules for entities ISA

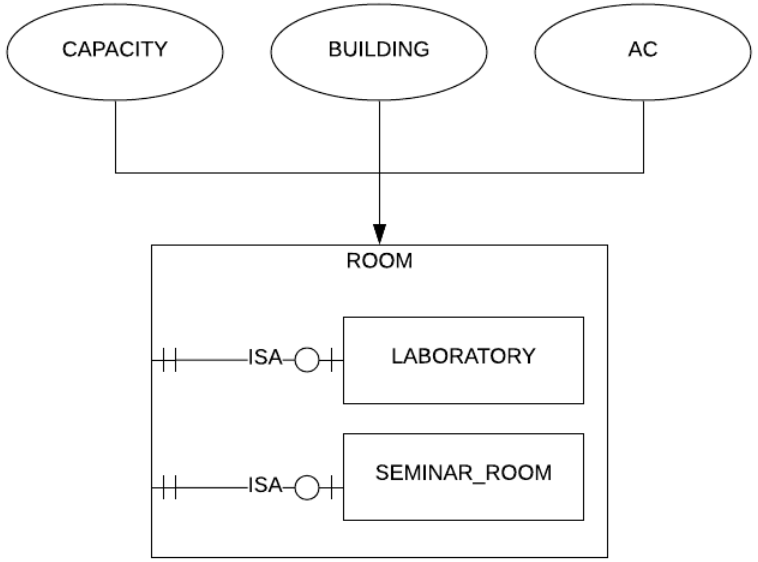


Rules for entities ISA

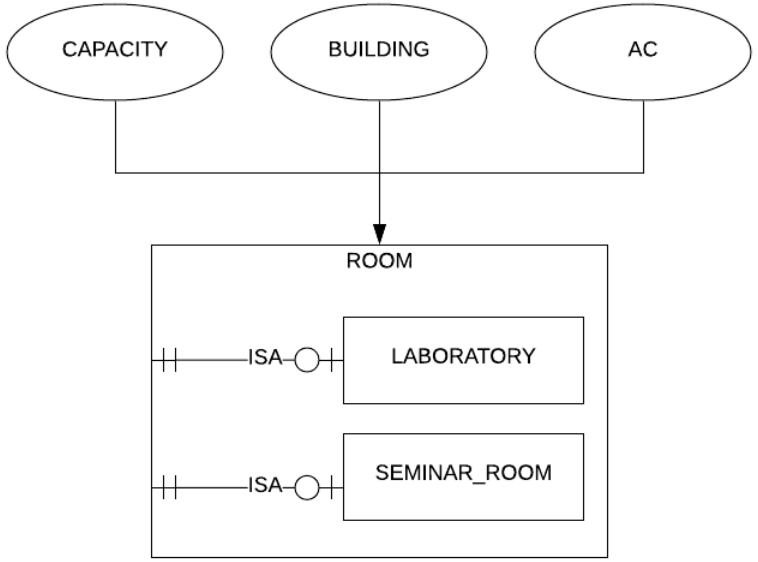


TICKET_ID	PRICE	HALL_NO	DATE	TYPE
1	200	Coliseum	08/03/20	VIP
2	150	Lyttelton	14/04/20	A
3	140	Olivier	01/05/20	A
4	90	Coliseum	04/06/20	B
5	220	Lyttelton	08/03/20	VIP
6	95	Olivier	14/04/20	B
7	210	Coliseum	20/03/20	VIP

Rules for entities ISA

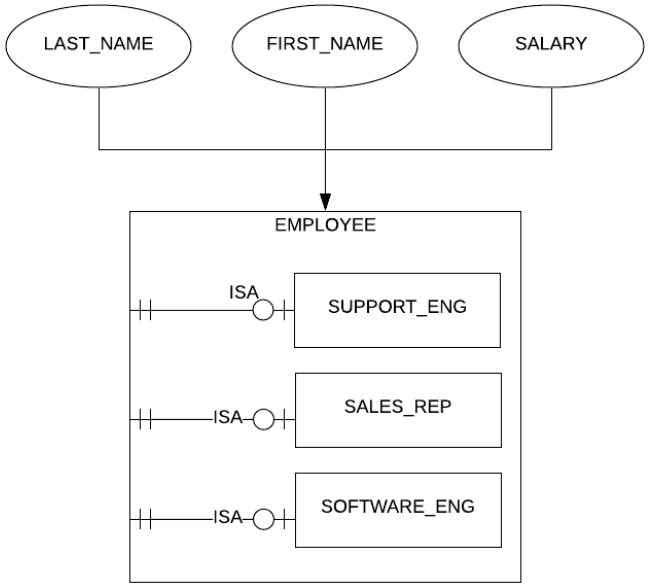


Rules for entities ISA

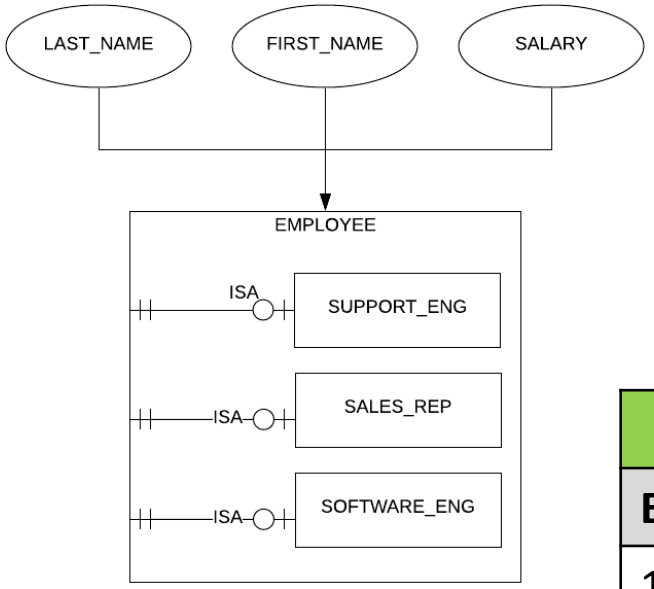


ROOM_ID	CAPACITY	BUILDING	LAB	SEM
1	40	FMI	1	1
2	45	Magurele	1	0
3	30	Geografie	0	0
4	90	FMI	1	0
5	80	FMI	1	0
6	95	Drept	0	1
7	20	FMI	1	1

Rules for entities ISA



Rules for entities ISA



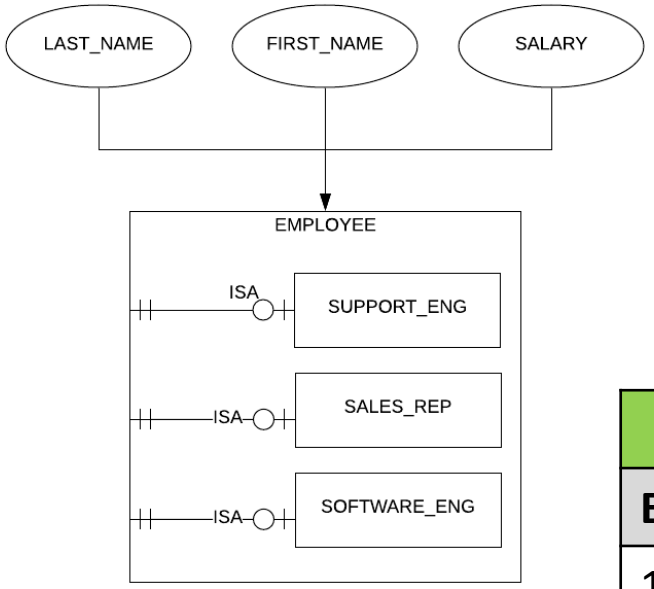
EMPLOYEES			
EMP_ID	**LAST_NAME**	**FIRST_NAME**	**SALARY**
1	Smith	John	2500
2	Grant	Anne	2700
3	Brown	Gregory	2300
…			

SUPPORT_ENG	
EMP_ID	**LEVEL**
1	3
…	…

SALES_REP	
EMP_ID	**TARGET**
2	25
…	…

SOFTWARE_ENG	
EMP_ID	**TEEM**
3	
…	…

Rules for entities ISA



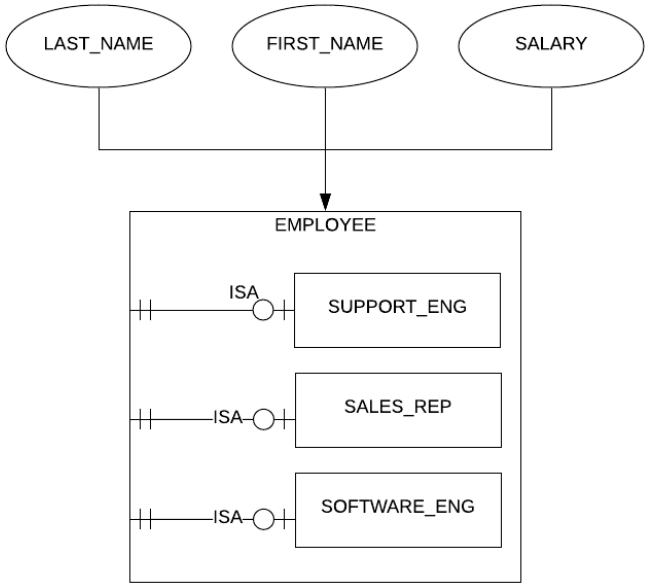
EMPLOYEES			
EMP_ID	**LAST_NAME**	**FIRST_NAME**	**SALARY**
1	Smith	John	2500
2	Grant	Anne	2700
3	Brown	Gregory	2300
…			

SUPPORT_ENG	
EMP_ID	**LEVEL**
1	3
…	…

SALES_REP	
EMP_ID	**TARGET**
2	25
…	…

SOFTWARE_ENG	
EMP_ID	**TEEM**
3	
…	…

Rules for entities ISA



SUPPORT_ENG				
EMP_ID	**LEVEL**	**LAST_NAME**	**FIRST_NAME**	**SALARY**
1	3	Smith	John	2500
…	…			

SALES_REP				
EMP_ID	**TARGET**	**LAST_NAME**	**FIRST_NAME**	**SALARY**
2	25	Grant	Anee	2700
…	…			

SOFTWARE_ENG				
EMP_ID	**TEEM**	**LAST_NAME**	**FIRST_NAME**	**SALARY**
3	3	Brown	Gregory	2300
…	…			

Rules for relationships

- 1 to 1 & 1 to M → foreign keys.
  - 1 (PK) to M (FK)
  - Usually, in 1 to 1 relationship, the FK is placed in the tables with fewer rows.
  - in 1 to many relationship, the PK is places on the ‘M’ side of relationship.

- M to M → associative table.
  - PK contains FKs and additional column.

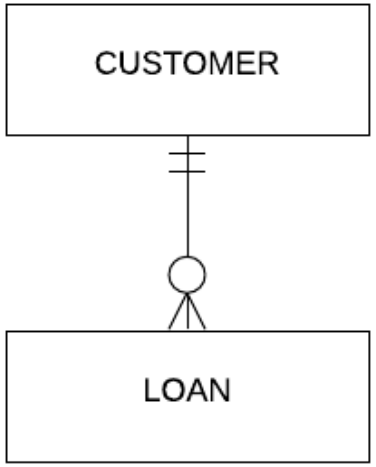
- Ternary relationships → associative table.
  - PK contains FKs and additional column.

One to One

ACCOUNT			
ACCOUNT_ID	**LAST_NAME**	**FIRST_NAME**	**DATE**
10	Snow	John	08/03/20
22	Grant	Anee	14/04/20
300	Brown	Gregory	01/05/20
…	…	…	…

CARD			
CARD_ID	**ACCOUNT_ID**	**CVN**	**DATE**
16897	10	125	18/04/21
24789	22	987	14/04/22
34597	300	875	03/05/21
…	…	…	…

One to Many

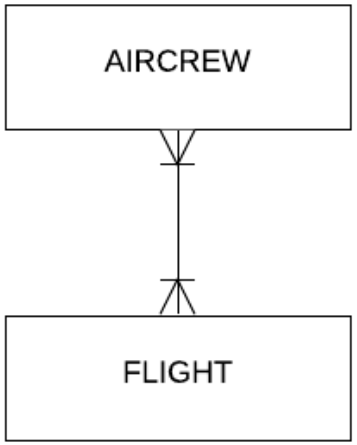


CUSTOMER			
CUSTOMER_ID	**LAST_NAME**	**FIRST_NAME**	**….**
10	Snow	John	….
22	Grant	Anee	….
300	Brown	Gregory	….
…	…	…	…

LOAN			
LOAN_ID	**CUSTOMER_ID**	**VALUES**	**DATE**
16897	10	125000	18/04/21
24789	22	987000	14/04/22
34597	300	87500	03/05/21
…	…	…	…

Many to Many

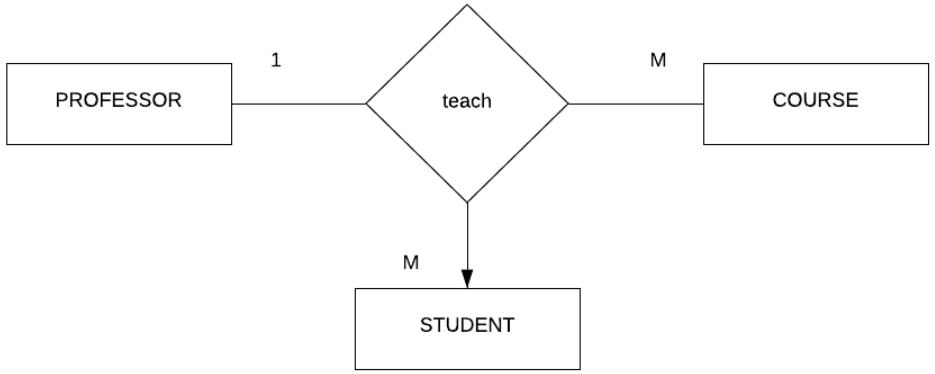
FLIGHT			
FLIGHT_ID	**DEP_AIRPORT**	**DATE**	**….**
1	Gatwick Airport	20/04/21	….
2	Grant	14/05/20	….
…	…	…	…



FLIGHT_CREW		
CREW_ID	**FLIGHT_ID**	**OBSERVATIONS**
10	1	…
22	1	…
10	2	…

AIRCREW			
CREW_ID	**LAST_NAME**	**FIRST_NAME**	**JOB_ID**
10	Snow	John	captain
22	Grant	Anee	first_officer
…	…	…	…

Ternary Relationships

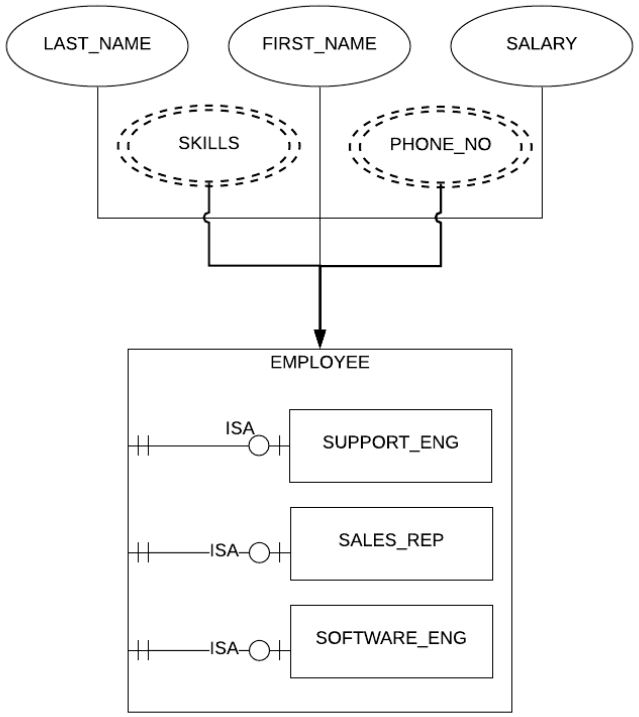


TEACH			
PROFESSOR_ID	**COURSE_ID**	**STUDENT_ID**	**GRADE**
1	BD	1001	9
1	SGBD	1002	10
1	BD	1002	8
2	TAP	1001	8
2	TAP	1002	10
2	AG	1001	5
….	….	….	….

Rules for attributes

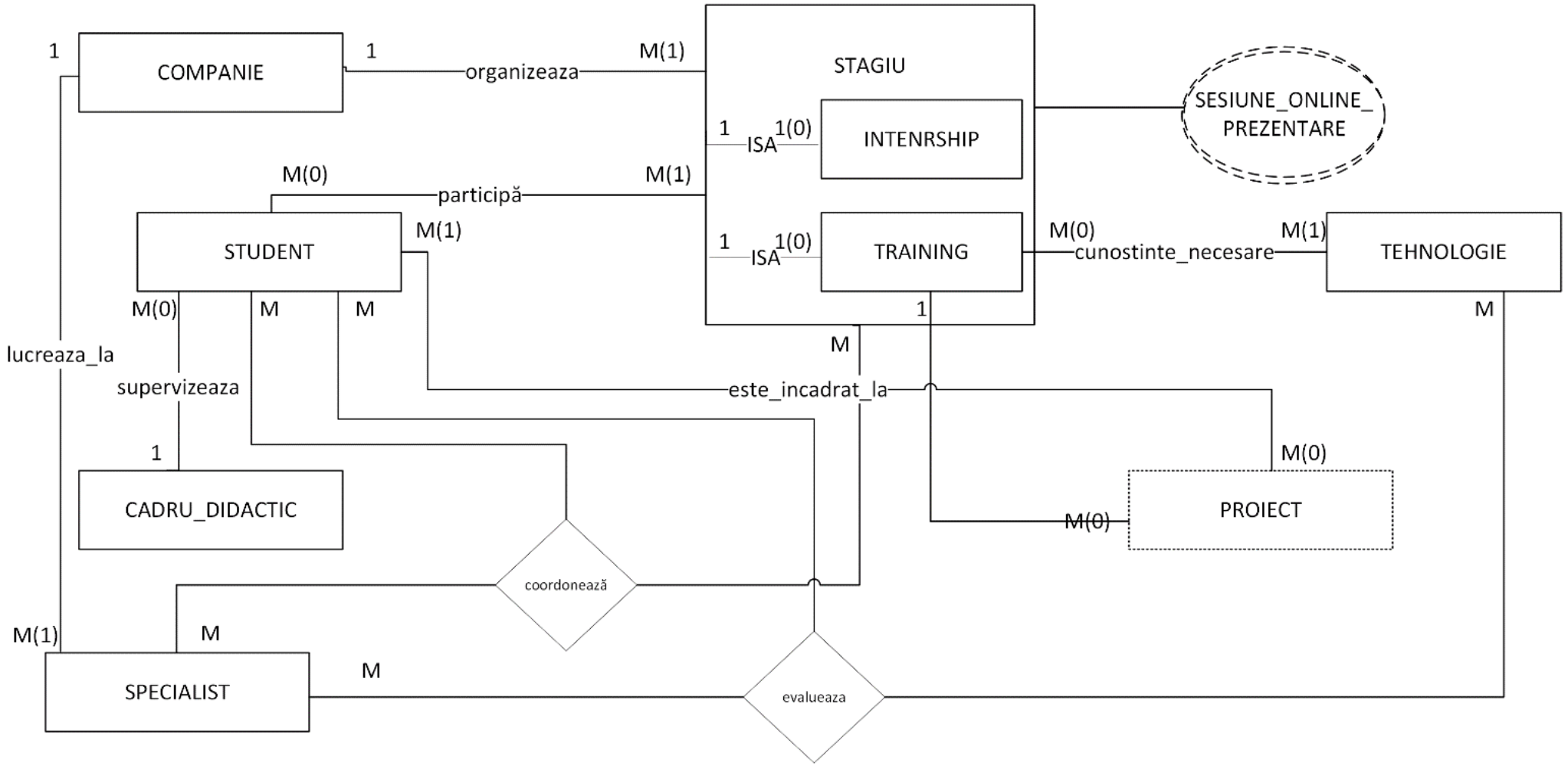
- Simple attribute → column
- Multivalued attributes → weak entity → table
  → set of columns

Rules for entities ISA



EMPLOYEES					
EMP_ID	**LAST_NAME**	**FIRST_NAME**	**SALARY**	**PHONE1**	**PHONE2**
1	Smith	John	2500	0745…	0720…
2	Grant	Anne	2700	07497…	NULL
3	Brown	Gregory	2300	NULL	07458..
…	…	…	…	…	…

EMP_SKILL		
EMP_ID	**SKILL**	**LEVEL**
1	Python	3
1	C++	2
1	NoSql	3
2	SQL	1


COMPANIE
organizeaza
STAGIU
INTENRSHIP
ISA
SESIUNE_ONLINE_
PREZENTARE
participă
STUDENT
TRAINING
cunostinte_necesare
TEHNOLOGIE
lucreaza_la
supervizeaza
este_incadrat_la
CADRU_DIDACTIC
PROIECT
coordonează
SPECIALIST
evalueaza