Generalitat de Catalunya
Departament d'Educació
Institut Thos i Codina

draw.io per a diagrames Entitat-Relació (ER)

Domini Públic (CC0)

Índex

Objectius 1

1. Conceptes previs de modelització ER 1

2. Primers passos 2
Format del fitxer . . . 2

3. Activar la biblioteca d'Entity Relation 2

4. Dibuixar una entitat com a taula (model lògic/físic) 2
Pas a pas . . . 3
Estil recomanat . . . 3

5. Crear relacions amb cardinalitat 4
5.1. Connectar entitats . . . 4
5.2. Triar la notació de cardinalitat (crow's foot) . . . 4
5.3. Exemple: relació 1:N . . . 4
5.4. Exemple: relació N:M amb entitat intermèdia . . . 5

6. Estil de connectors i organització 5

7. Generar entitats a partir de text (SQL) 5

8. Exportar i integrar a la documentació 6
Exportar . . . 6
Incrustar en Markdown . . . 6
Integracions . . . 6

9. Drecreres de teclat útils 7

10. Bones pràctiques 7

11. Recursos 7

Índex de figures 8

draw.io (també anomenat *diagrams.net*) és una aplicació gratuïta de diagramació, de codi obert (llicència Apache 2.0), que no requereix registre i et deixa triar on desar els fitxers. És ideal per dissenyar models de dades abans de crear la base de dades.



Figura 1: draw.io Logo

- Web: https://app.diagrams.net
- Aplicació d'escriptori: https://get.diagrams.net
- Documentació: https://www.drawio.com/docs/

Objectius

1. Obrir draw.io i triar on desar el diagrama.
2. Activar la biblioteca de formes d'**Entity Relation**.
3. Dibuixar entitats amb atributs, claus primàries i claus foranes.
4. Representar relacions amb la cardinalitat correcta (notació *crow's foot* / peu de gall).
5. Generar entitats automàticament a partir de text (SQL).
6. Exportar i integrar el diagrama en documentació.

1. Conceptes previs de modelització ER

Concepte	Descripció	Exemple
Entitat	Objecte del qual volem guardar dades	`ALUMNE`, `MODUL`
Atribut	Propietat d'una entitat	`nom`, `dni`
Clau primària (PK)	Atribut(s) que identifiquen de forma única una fila	`id_alumne`
Clau forana (FK)	Atribut que referencia la PK d'una altra entitat	`id_modul`
Relació	Associació entre entitats	un alumne *cursa* mòduls
Cardinalitat	Quantes ocurrències s'associen: 1:1, 1:N, N:M	un mòdul té molts alumnes

NOTA

Les relacions **N:M** no es poden implementar directament en un model relacional; es resolen amb una **entitat intermèdia** (taula de unió) que conté les dues FK.

2. Primers passos

1. Obre https://app.diagrams.net.
2. draw.io et demanarà **on desar** els diagrames: *Device* (local), Google Drive, OneDrive, GitHub, etc. Les dades les gestiones tu; el servei no hi té accés.
3. Tria **Create New Diagram**, posa un nom (p. ex. `institut-er.drawio`) i selecciona una plantilla en blanc o la categoria **Software / Database** si vols partir d'un exemple.
4. Selecciona el tema de l'editor (*Classic*, *Simple*, *Sketch*…) des de **Extras → Theme** o el menú *View*. Per a diagrames tècnics, *Classic* o *Simple* funcionen bé.

Format del fitxer

- `.drawio`: format natiu (XML), editable.
- `.drawio.svg` o `.drawio.png`: imatge que **conserva el diagrama editable** incrustat. Molt útil per a documentació en Markdown (vegeu l'apartat 8).

3. Activar la biblioteca d'Entity Relation

Al panell lateral esquerre hi ha les biblioteques de formes. Per activar la d'ER:

1. Clica **+ More Shapes** (a la part inferior del panell lateral) o utilitza **Arrange → Insert → …** / *File → Open Library*.
2. A la finestra *Shapes*, marca **Entity Relation** (dins de la categoria *Software* o *Other*, segons la versió).
3. Prem **Apply**.

Apareixerà una secció **Entity Relation** amb formes com: taula, entitat, entitat feble, atribut, relació, i connectors amb notació ER.

CONSELL

També pots activar **Entity Relation** i, si vols treballar amb notació UML, la biblioteca **UML**. Les taules d'UML i la forma *Table* de draw.io són vàlides per a model físic.

4. Dibuixar una entitat com a taula (model lògic/físic)

La manera més pràctica de modelitzar en bases de dades relacionals és utilitzar la forma de **taula** amb files separades per a cada atribut.

Pas a pas

1. A la biblioteca **Entity Relation** (o a **General** → **Table**), arrossega la forma **Table** (taula amb capçalera) al llenç.
2. Fes **doble clic** a la capçalera i escriu el nom de l'entitat: `alumne`.
3. Fes doble clic a cada fila i escriu els atributs. Convenció habitual:

```
PK   id_alumne        INT
     nom              VARCHAR(50)
     cognoms          VARCHAR(100)
     email            VARCHAR(120)
FK   id_cicle         INT
```

4. Per afegir files: selecciona una fila i prem **Ctrl + Enter** (duplica), o fes servir el menú *Arrange* / les fletxes verdes de la taula.
5. Per marcar la clau primària: dins de la taula, la primera columna acostuma a contenir les etiquetes **PK** / **FK**. Pots aplicar **negreta** a la fila de la PK (*Ctrl* + *B*) o subratllar-ne el text (*Ctrl* + *U*).

Estil recomanat

Element	Convenció
Nom d'entitat	Singular, minúscules o `PascalCase`, coherent a tot el diagrama
PK	Negreta o subratllat + etiqueta PK
FK	Cursiva + etiqueta FK
Atributs obligatoris	Marca `NOT NULL` o fes servir un símbol (*)
Colors	Un color per àrea funcional (p. ex. alumnat, mòduls, avaluació)

Per canviar els colors: selecciona la capçalera i, al panell **Style** de la dreta, modifica *Fill* i *Line*. Utilitza **Edit Style** per copiar l'estil a altres entitats (*Copy Style* / *Paste Style* al panell de format).

5. Crear relacions amb cardinalitat

5.1. Connectar entitats

1. Passa el ratolí per sobre d'una entitat: apareixen **fletxes blaves**.
2. Arrossega una fletxa fins a l'altra entitat (millor fins a la **fila de la FK** i la **fila de la PK** per tal que el connector apunti als atributs concrets).
3. Quan el connector es mostri **resaltat en verd/blau** sobre la fila, deixa anar.

5.2. Triar la notació de cardinalitat (crow's foot)

1.- Selecciona el connector.

2.- Al panell **Style** → **Line**, desplega els selectors de **inici** (*Start*) i **final** (*End*) de la fletxa.

3.- Tria les formes de la família **ER**:

Cardinalitat	Forma de connector
Exactament un	`ERmandOne` (dues barres)
Zero o un	`ERzeroToOne`
Un o molts	`ERoneToMany`
Zero o molts	`ERzeroToMany`

Aquestes formes també es poden escriure directament a **Edit Style** (*Ctrl + E*) amb les propietats `startArrow` i `endArrow`:

```
edgeStyle=entityRelationEdgeStyle;fontSize=12;html=1;endArr
↪ ow=ERzeroToMany;startArrow=ERmandOne;
```

4.- Per posar un **nom a la relació** (p. ex. *cursa*), fes doble clic al connector i escriu el text.

5.3. Exemple: relació 1:N

Un **cicle formatiu** té molts **alumnes**; un alumne pertany a un sol cicle:

- Connector des de `alumne.id_cicle` (FK) cap a `cicle.id_cicle` (PK).
- Extrem `cicle`: `ERmandOne`.
- Extrem `alumne`: `ERzeroToMany`.

5.4. Exemple: relació N:M amb entitat intermèdia

Un alumne cursa molts mòduls i un mòdul té molts alumnes. Es resol amb `matricula`:

```
alumne (1) ──< matricula >── (1) modul
```

1. Crea la taula `matricula` amb dues FK: `id_alumne` i `id_modul` (la combinació de les dues és la PK composta).
2. Connecta `alumne` → `matricula` amb `ERmandOne` / `ERzeroToMany`.
3. Connecta `modul` → `matricula` amb `ERmandOne` / `ERzeroToMany`.
4. Pots afegir atributs propis de la relació a `matricula` (p. ex. `nota`, `data_matricula`).

6. Estil de connectors i organització

- **Rectes o ortogonals:** selecciona el connector i, a *Style* → *Connection*, tria *Straight*, *Orthogonal*, *Curved*…
- **Waypoints:** arrossega el punt groc del connector per canviar-ne el traçat.
- **Alinear i distribuir:** selecciona diverses entitats → **Arrange** → **Align / Distribute**.
- **Quadrícula i guies:** activa **View** → **Grid** i **View** → **Guides** perquè els elements encaixin.
- **Layouts automàtics: Arrange** → **Layout** (*Vertical Tree*, *Organic*, *Circle*…). Per a ER, *Organic* sol donar bons resultats d'entrada i després s'ajusta a mà.
- **Capes: Arrange** → **Layers** (*Ctrl* + *Shift* + *L*) permet separar, per exemple, el model conceptual del físic.

7. Generar entitats a partir de text (SQL)

draw.io pot **crear taules ER a partir de codi SQL**, molt útil per documentar una base de dades existent.

1. Ves a **Arrange** → **Insert** → **Advanced** → **From Text…** (o a *Extras* → *Edit Diagram* si vols treballar directament amb el XML).
2. Selecciona el mode **SQL** (o *Entity Relation*, segons la versió) i enganxa el codi:

```
CREATE TABLE cicle (
  id_cicle INT PRIMARY KEY,
  nom VARCHAR(80) NOT NULL
);

CREATE TABLE alumne (
  id_alumne INT PRIMARY KEY,
  nom VARCHAR(50) NOT NULL,
  cognoms VARCHAR(100),
  id_cicle INT,
  FOREIGN KEY (id_cicle) REFERENCES cicle(id_cicle)
);
```

3. Prem **Insert**. S'afegiran les taules al llenç, que després podràs reorganitzar i connectar.

> **NOTA**
>
> L'opció exacta del menú pot variar entre versions. Si no la trobes, cerca *"Text"* o *"SQL"* amb la lupa del menú **Arrange** → **Insert** o consulta la documentació oficial.

També pots anar en sentit contrari: dissenyar al diagrama i escriure manualment el `CREATE TABLE` a partir del model, o fer servir un script d'exportació.

8. Exportar i integrar a la documentació

Exportar

File → **Export as** i tria el format:

Format	Quan l'utilitzes
PNG	Imatge ràpida; activa *Include a copy of my diagram* per mantenir-lo editable
SVG	Vectorial; ideal per a web i documents imprimibles
PDF	Entregues o annexos
HTML / URL	Compartir o incrustar
XML (.drawio)	Còpia de seguretat editable

Opcions útils a la finestra d'exportació: **Transparent Background**, **Border Width**, **Zoom** i **Selection Only** (exportar només els elements seleccionats).

Incrustar en Markdown

La manera més còmoda és exportar com a **`.drawio.svg`** (o `.drawio.png`) i referenciar-lo com una imatge:

```
![Diagrama ER de l'institut](img/institut.drawio.svg)
```

D'aquesta manera, el mateix fitxer es mostra a la documentació **i** es pot reobrir a draw.io per editar-lo. Funciona bé amb pipelines que converteixen Markdown a HTML, PDF o ODT.

Integracions

draw.io s'integra amb Google Drive, OneDrive/SharePoint, Confluence, GitHub, Notion i VS Code (extensió *Draw.io Integration*), entre altres. Pots editar fitxers `.drawio` directament a l'editor de codi si fas servir l'extensió de VS Code.

9. Drecreres de teclat útils

Acció	Dreçera
Duplicar element	`Ctrl + D`
Copiar / enganxar estil	`Ctrl + Shift + C` / `Ctrl + Shift + V`
Editar estil	`Ctrl + E`
Editar dades	`Ctrl + M`
Seleccionar-ho tot	`Ctrl + A`
Agrupar / desagrupar	`Ctrl + G` / `Ctrl + Shift + U`
Desfer / refer	`Ctrl + Z` / `Ctrl + Y`
Cercar a la biblioteca de formes	`Ctrl + Shift + K`
Mostrar/amagar quadrícula	`Ctrl + Shift + G`

A macOS substitueix `Ctrl` per `Cmd` (⌘).

10. Bones pràctiques

1. **Defineix primer el model conceptual** (entitats i relacions) i després afegeix atributs i tipus.
2. **Mantén una convenció de noms** (singular/plural, minúscules, guions baixos) i aplica-la a tot el diagrama.
3. **Marca sempre PK i FK** de forma visible.
4. **Evita creuar connectors**; reorganitza les entitats per minimitzar-ne els encreuaments.
5. **Usa colors amb criteri**: màxim 3-4 colors, un per àrea funcional.
6. **Inclou una llegenda** de la notació (*crow's foot*) si el diagrama es compartirà amb persones que no la coneguin.
7. **Versiona els fitxers `.drawio`** (p. ex. a Git): són XML i els canvis es poden revisar.
8. **Desa una còpia `.drawio.svg`** per a la documentació en lloc d'una captura de pantalla.

11. Recursos

- Pàgina oficial: https://www.drawio.com
- Editor en línia: https://app.diagrams.net
- Documentació: https://www.drawio.com/docs/
- Codi font: https://github.com/jgraph/drawio
- Comunitat: https://github.com/jgraph/drawio/discussions

Índex de figures

1 draw.io Logo 1

Versions d'aquest document

- HTML
- PDF
- ODT
- MD

Domini Públic (CC0)