



**Universidade de Brasília**

Departamento de Ciência da Computação



# Bancos de Dados

CIC0097



**Prof. Pedro Garcia Freitas**

<https://pedrogarcia.gitlab.io/>

[pedro.garcia@unb.br](mailto:pedro.garcia@unb.br)

Universidade de Brasília  
Instituto de Ciências Exatas  
Departamento de Ciências da Computação



Este conjunto de slides não deve ser utilizado ou republicado sem a expressa permissão do autor.

This set of slides should not be used or republished without the author's express permission.



# **Módulo 11**

# **Ferramentas CASE**

**CIC0097/2023.1**  
**T1/T2**



# 1. Objetivos

Esta aula apresenta um exemplo de Ferramenta CASE para modelagem de bancos de dados.



## 2. Ferramentas CASE

- **Computer Aided Software Engineering**
- Uma **ferramenta CASE** é um programa de computador que tem o propósito de dar apoio aos engenheiros de software e projetistas de bancos de dados nas atividades de modelagens.



## 2. Ferramentas CASE

- As ferramentas podem suportar diferentes tarefas envolvidas em todo o processo de desenvolvimento de um software ou de um sistema.
  - Analisar
  - Modelar
  - Documentar



## 2. Ferramentas CASE

- Por exemplo, uma ferramenta CASE pode ser usada para:
  - Diagramação
  - Compilação de resultados e geração de relatórios
  - Ferramentas de análise
  - Repositório central
  - Geração de documentos
  - Geração de código



## 2. Ferramentas CASE

- Algumas ferramentas são projetadas especificamente para suportar uma técnica específica, enquanto outras ferramentas são de natureza mais geral.
- No nosso caso, considerando o escopo da disciplina, veremos uma ferramenta CASE para modelagem de Banco de Dados.





## 2. Ferramentas CASE

- Uma ferramenta **CASE** para modelagem de bancos de dados tem o propósito de suportar as diferentes tarefas que o projetista tem que executar durante o projeto do banco de dados.



## 2. Ferramentas CASE

- Exemplos de tarefas que devem ser suportadas para modelagem de Bds:
  - Criação de **Modelos Entidade-Relacionamento**;
  - Tradução/Criação de **Modelos Relacionais**;
  - Criação automática de **dicionário de dados**;
  - Produção de “scripts” e **esquemas** de bancos de dados para serem usados como entradas de SGBDs;
  - Possibilitar a **engenharia reversa**.



## 2. Ferramentas CASE

- Espera-se que uma ferramenta CASE forneça funcionalidades de verificação das regras/restrições que definem um modelo;



## 2. Ferramentas CASE

- Espera-se que uma ferramenta CASE forneça funcionalidades de verificação das regras/restrições que definem um modelo;
- Dessa forma, a ferramenta CASE deve ser capaz de restringir as ações por parte dos seus usuários (analistas, projetistas, etc) que levem a erros de modelagem.



## 2. Ferramentas CASE

- Portanto, as ferramentas CASE disponibilizam funcionalidades que vão, em até certo nível, **guiar** o trabalho do engenheiro de dados.



## 2. Ferramentas CASE

- Exemplos de ferramentas CASE com recursos para atividades de modelagem de banco de dados:
  - DB-Main: <https://www.db-main.eu/>
  - DBDesigner: <https://www.dbdesigner.net/>
  - erwin Data Modeler:  
<https://www.erwin.com/products/erwin-data-modeler/>

## 3. DB-Main

- DB-Main é uma ferramenta CASE gráfica que suporta a maioria dos processos de engenharia de banco de dados.
- Ele fornece funções para análise de informações conceituais, design lógico, geração de código, integração de esquema, avaliação de qualidade, mapeamento de esquema, engenharia reversa, transformação de esquemas, etc.



## 3. DB-Main

Dessa ferramenta, nos interessa particularmente as funcionalidades de **construção no MER** e os recursos de **mapeamento** automático para o Modelo Relacional e a **criação de esquemas** em SQL de construção do Banco de Dados.





### 3. DB-Main

- A ferramenta é gratuita e pode ser baixada para os sistemas operacionais Windows e Linux em:
  - <https://www.db-main.eu/getit/>



## 3. DB-Main

The screenshot shows a web browser window with the address bar displaying "db-main.eu/getit/". The page content includes a statement that DB-MAIN is free of charge and available for Windows and Linux. Below this, the "Latest Version" section states that by downloading the software, users agree to the following clauses:

1. DB-MAIN can not be redistributed to third parties.
2. The University of Namur declines any responsibility for the installation, use or uninstallation of DB-MAIN.
3. The University of Namur is the exclusive owner of the DB-MAIN software.

A dark blue button labeled "I agree terms and conditions" is highlighted with a red arrow pointing to it from the right. To the right of the arrow, the text "Click in this button" is written in red. Below the button, the text indicates that the latest version is 11.0.2, released on 20 Sept 2021. The "Previous versions" section is also visible at the bottom, with a link to the LIBD research group.



### 3. DB-Main

I agree terms and conditions

- Download DB-MAIN Version 11.0.2 (for Windows 10 64bits)  
(SHA256:  
5FC88F124E033431313D2BC187EA743162D594B2A4BA4C7C  
9B1093D72C30A021)
- Download DB-MAIN Version 11.0.2 (for Windows 10 32bits)  
(SHA256:  
305D8937F2CFA84A79DC7BAAAB3FC5EF7DF7A7C645C40E1  
D0C44F83B13C87974)
- Download DB-MAIN Version 11.0.2 (for Linux 64bits)  
(SHA256:  
8F61E6069337DC734AC6F886B224329EC943CD4DAFBD8886  
11BD806373CB3296)

The latest version is version 11.0.2 released on 20 Sept 2021



### 3. DB-Main

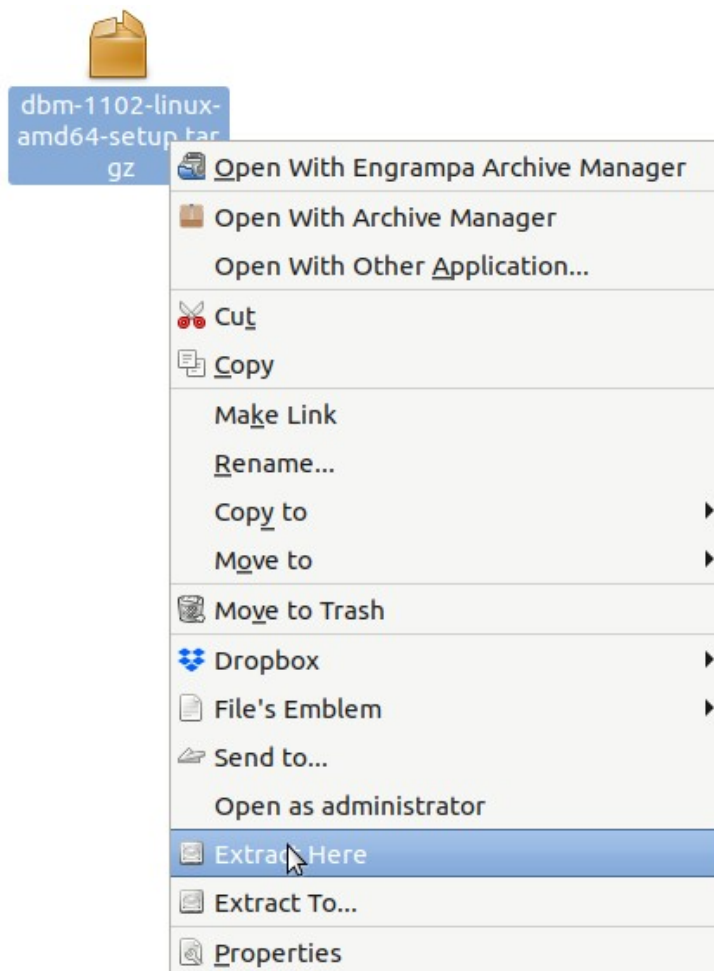
I agree terms and conditions

- Download DB-MAIN Version 11.0.2 (for Windows 10 64bits)  
(SHA256:  
5FC88F124E033431313D2BC187EA743162D594B2A4BA4C7C  
9B1093D72C30A021)
- Download DB-MAIN Version 11.0.2 (for Windows 10 32bits)  
(SHA256:  
305D8937F2CFA84A79DC7BAAAB3FC5EF7DF7A7C645C40E1  
D0C44F83B13C87974)
- Download DB-MAIN Version 11.0.2 (for Linux 64bits)  
(SHA256:  
8F61E6069337DC734AC6F886B224329EC943CD4DAFBD8886  
11BD806373CB3296)

The latest version is version 11.0.2 released on 20 Sept 2021



### 3. DB-Main





## 3. DB-Main

Copie o diretório para um local de destino,  
e.g.:

```
$ mv ./dbm-1102-linux-amd64-setup  
/opt/db-main/
```



## 3. DB-Main

Edite o `~/.bashrc` (ou `~/.profile` em alguns ambientes) e adicione as seguintes linhas:

- `export DB_MAIN_BIN=/opt/db-main/bin`
- `export PATH=$DB_MAIN_BIN:$PATH`
- `export LD_LIBRARY_PATH=$DB_MAIN_BIN:`  
`$DB_MAIN_BIN/../../java/jre/lib/amd64/server:`  
`$LD_LIBRARY_PATH`



## 3. DB-Main

Exemplo:

```
pedro@suse:~ 103x24
GNU nano 6.3 /home/pedro/.bashrc
export CUDA_HOME="/usr/local/cuda"
export PATH="/usr/local/cuda/bin:$PATH"
o4-setup.tar.gz

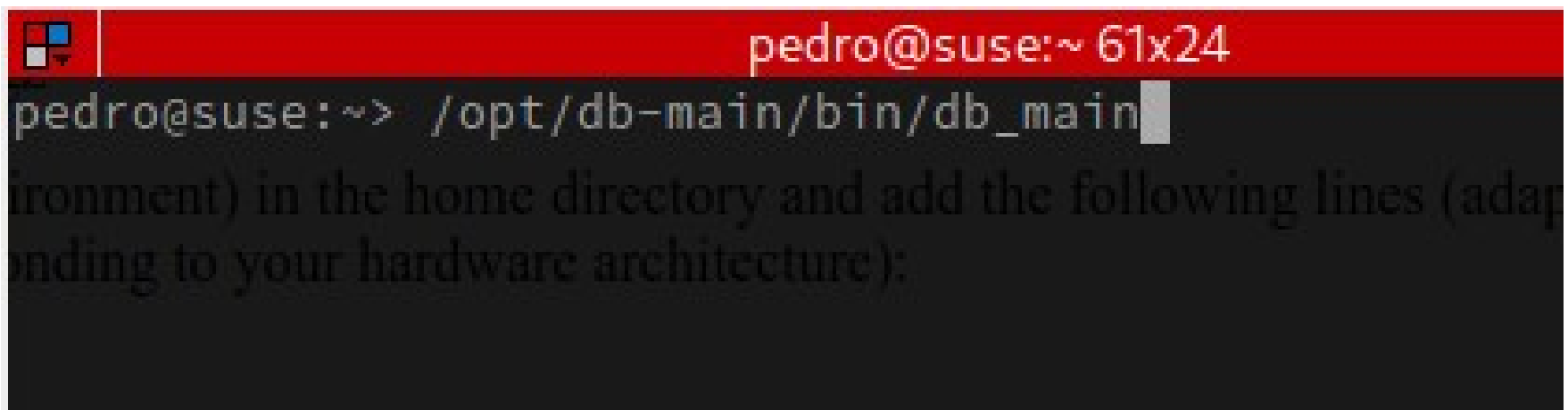
export DB_MAIN_BIN=/opt/db-main/bin and add the following lines (adapt the ~/db-main/bin directory to the one chosen pr
export PATH=$DB_MAIN_BIN:$PATH architecture);
export LD_LIBRARY_PATH=$DB_MAIN_BIN:$DB_MAIN_BIN/../../java/jre/lib/amd64/server:$LD_LIBRARY_PATH
lb-main/bin
BIN:$PATH
# >>> mamba initialize >>>
# !! Contents within this block are managed by 'mamba init' !!
export MAMBA_EXE="/opt/micromamba/micromamba";
export MAMBA_ROOT_PREFIX="/opt/micromamba";
__mamba_setup="$('/opt/micromamba/micromamba' shell hook --shell bash --prefix '/opt/micromamba' 2> /d>
if [ $? -eq 0 ]; then
    eval "$__mamba_setup"
else
    if [ -f "/opt/micromamba/etc/profile.d/micromamba.sh" ]; then
        "/opt/micromamba/etc/profile.d/micromamba.sh"
    else
        i386-setup.tar.gz or
        amd64-setup.tar.gz
    fi
fi

^G Help      ^O Write Out  ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```



## 3. DB-Main

Após configurar o DB-Main, basta executá-lo a partir do diretório onde ele foi instalado. No nosso exemplo:

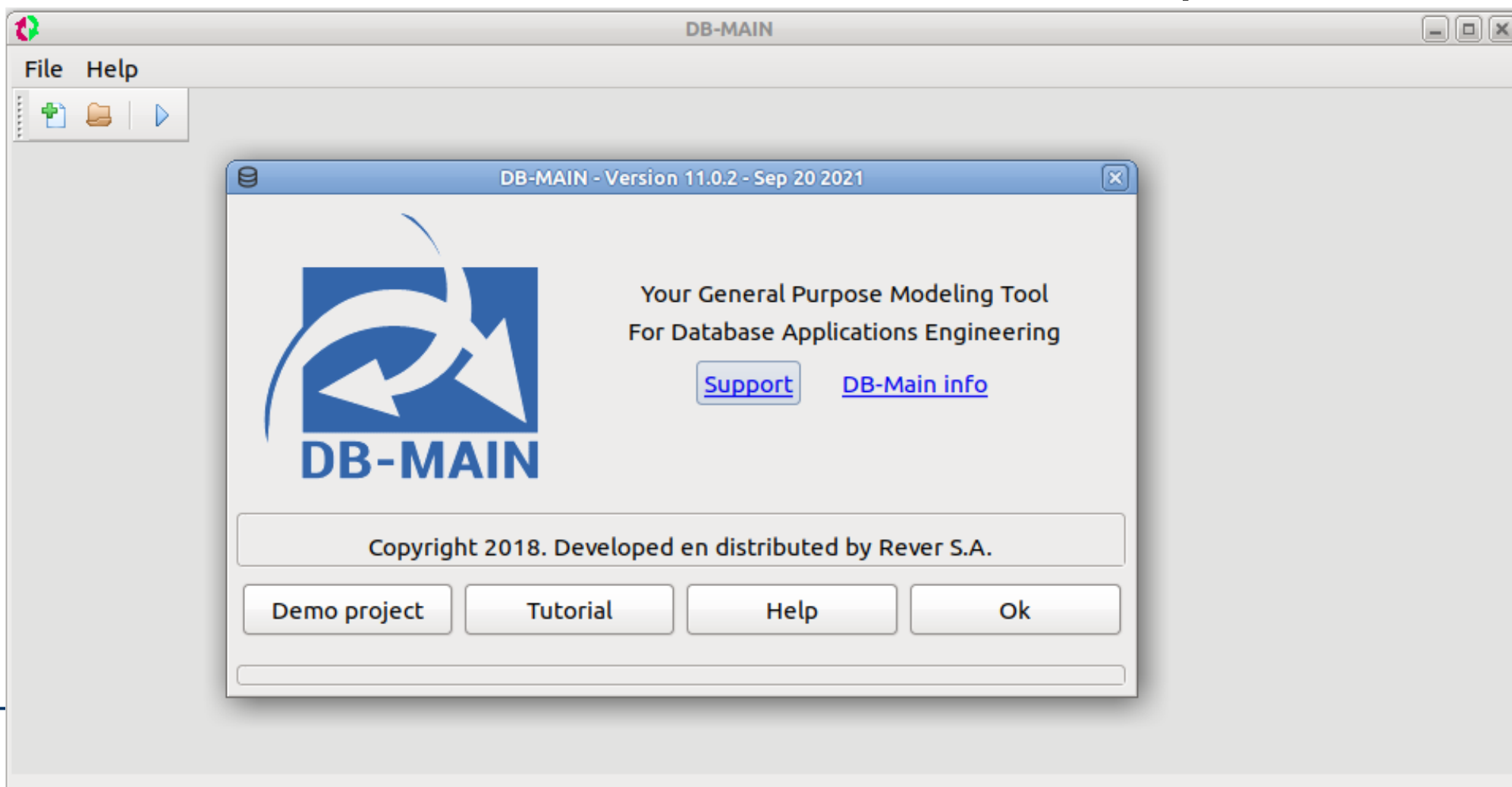


```
pedro@suse:~ 61x24
pedro@suse:~> /opt/db-main/bin/db_main
ironment) in the home directory and add the following lines (adap
onding to your hardware architecture):
```



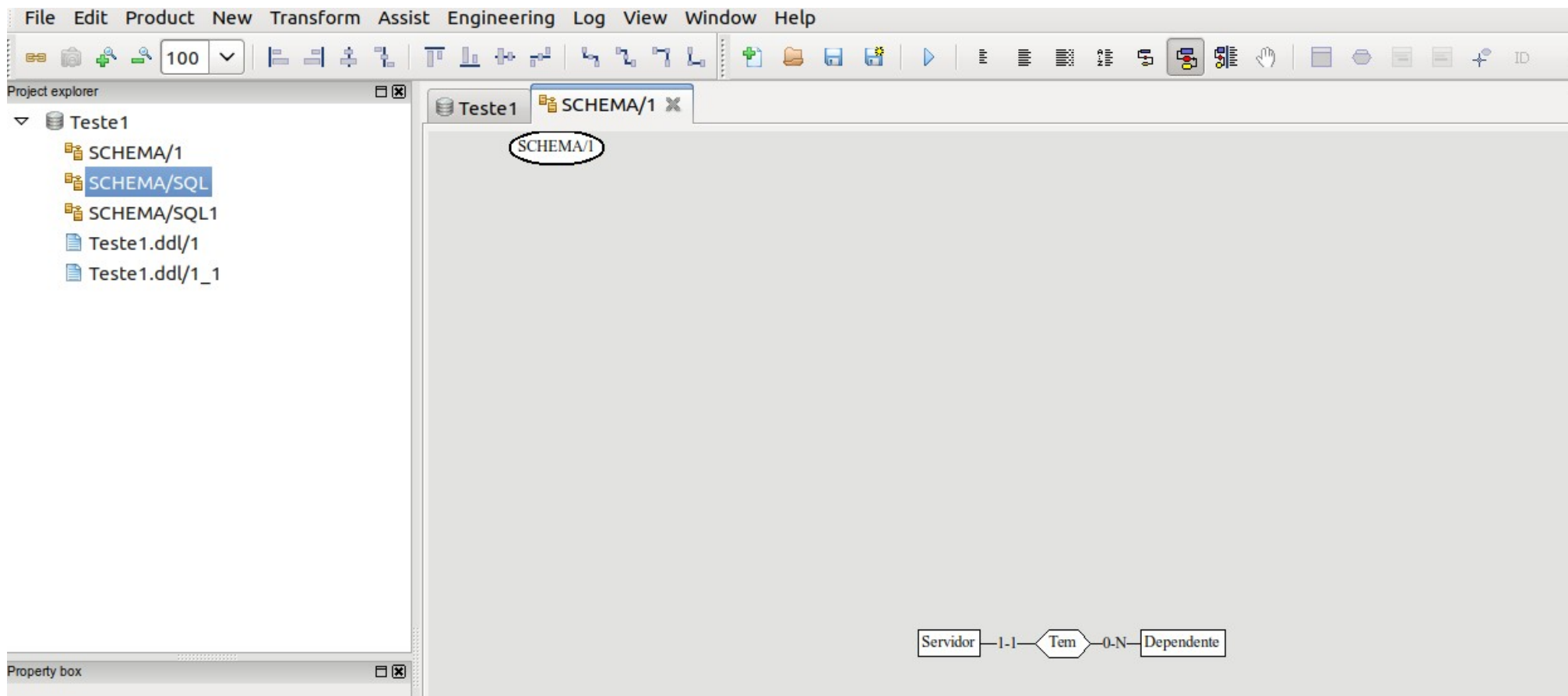
### 3. DB-Main

Se tudo correr bem, essa tela deve aparecer:





## 3. DB-Main



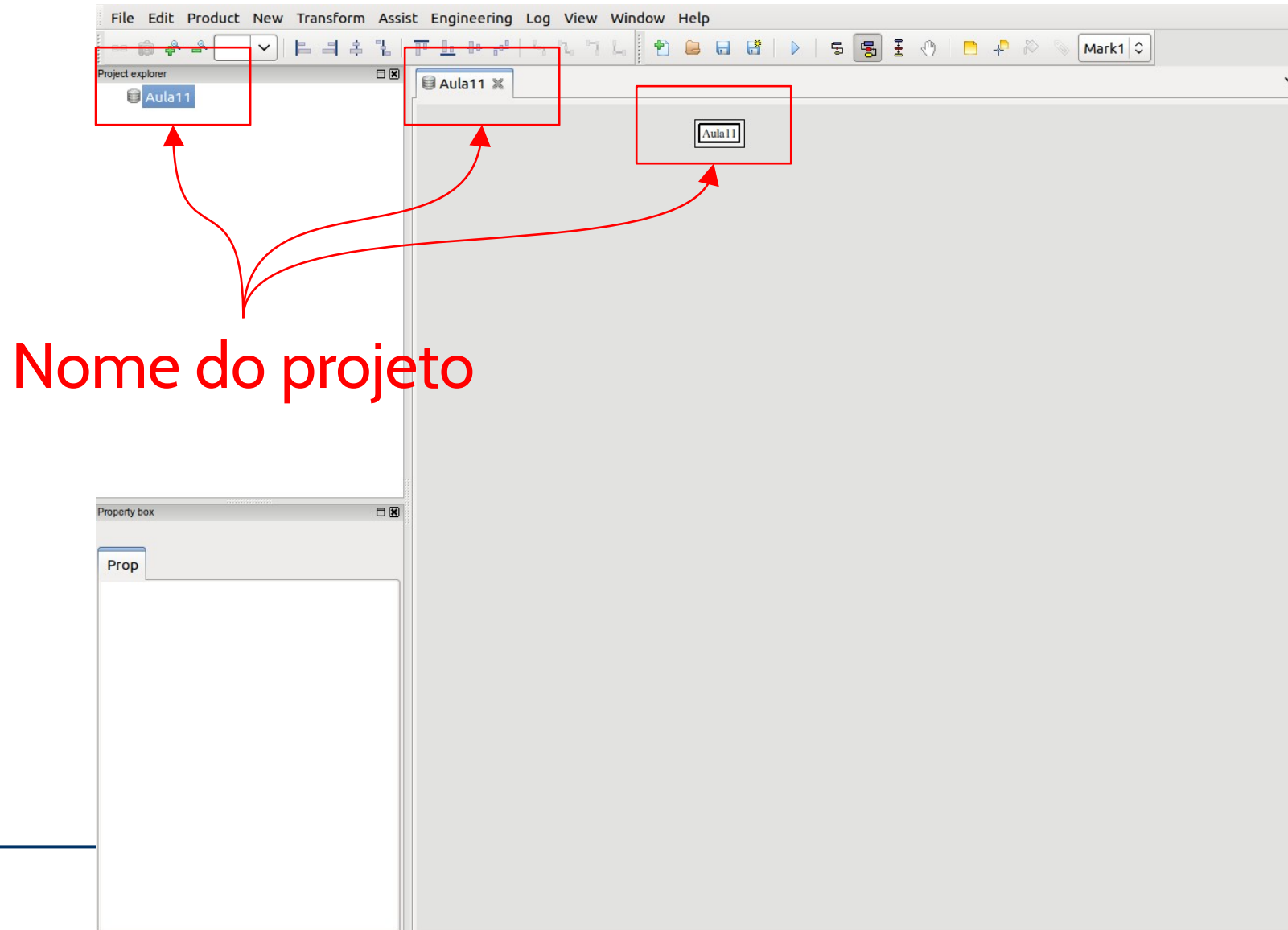


## 3. DB-Main

- Criação do Projeto no DB-Main:
  - Todo modelo criado é associado a um projeto;
  - Um projeto pode ter diferentes esquemas (produtos)



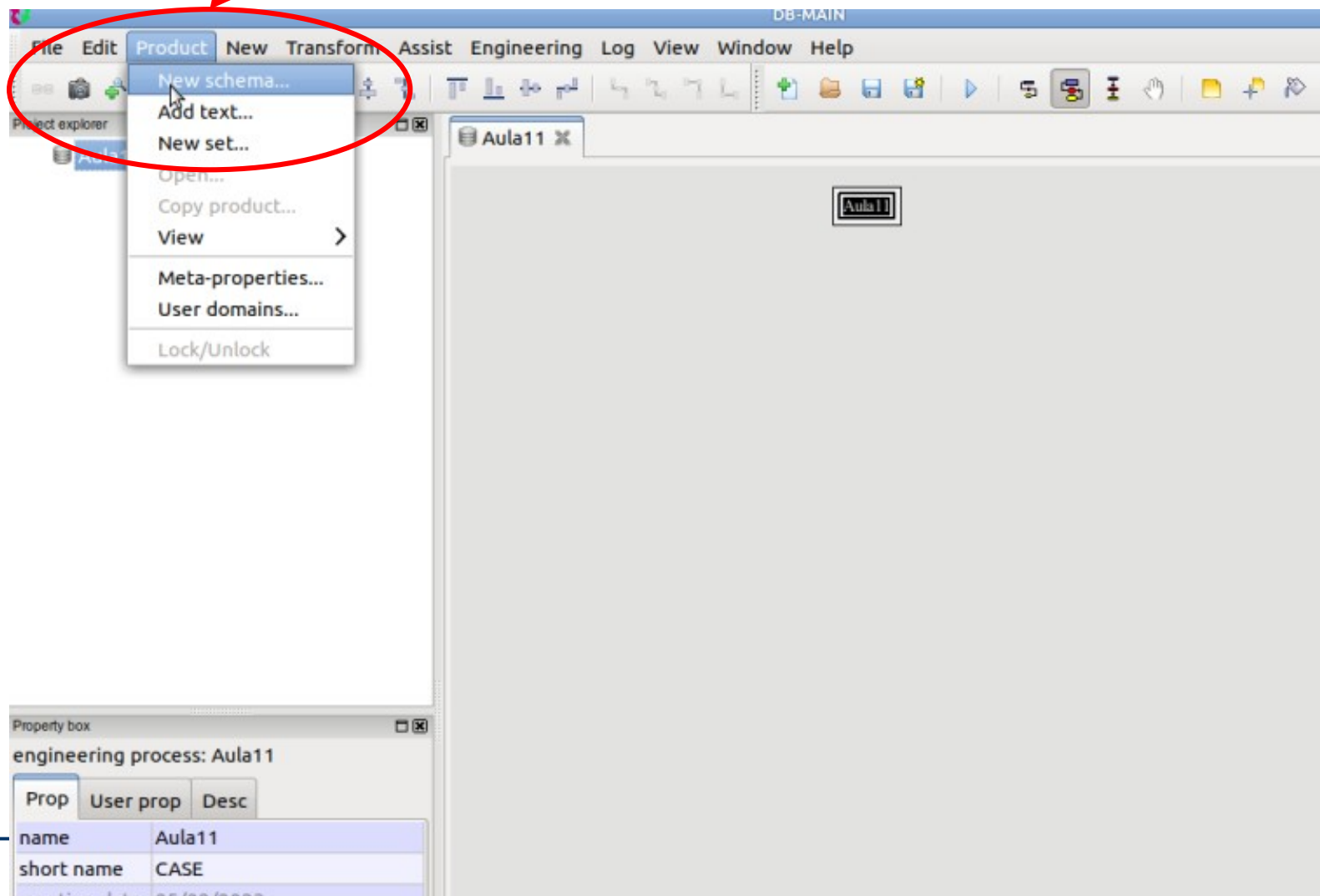
## 3. DB-Main





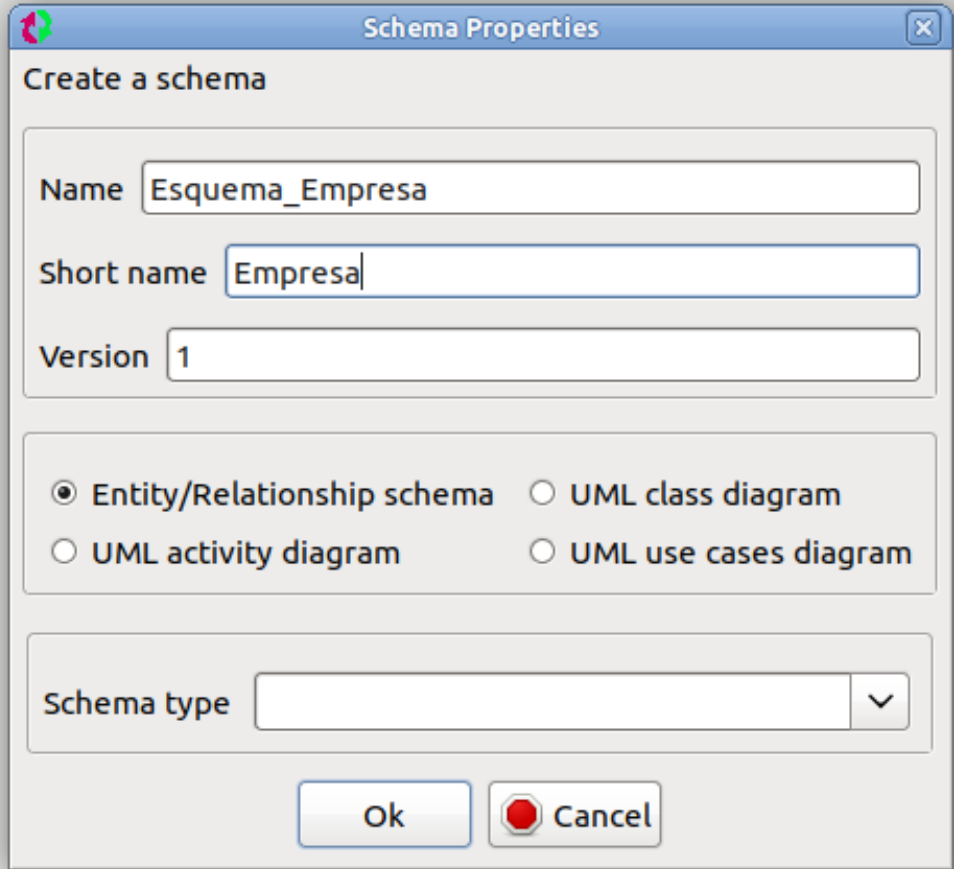
### 3. DB-Main

Criação de esquema dentro do projeto.



## 3. DB-Main

Criação de esquema dentro do projeto.



Schema Properties

Create a schema

Name: Esquema\_Empresa

Short name: Empresa

Version: 1

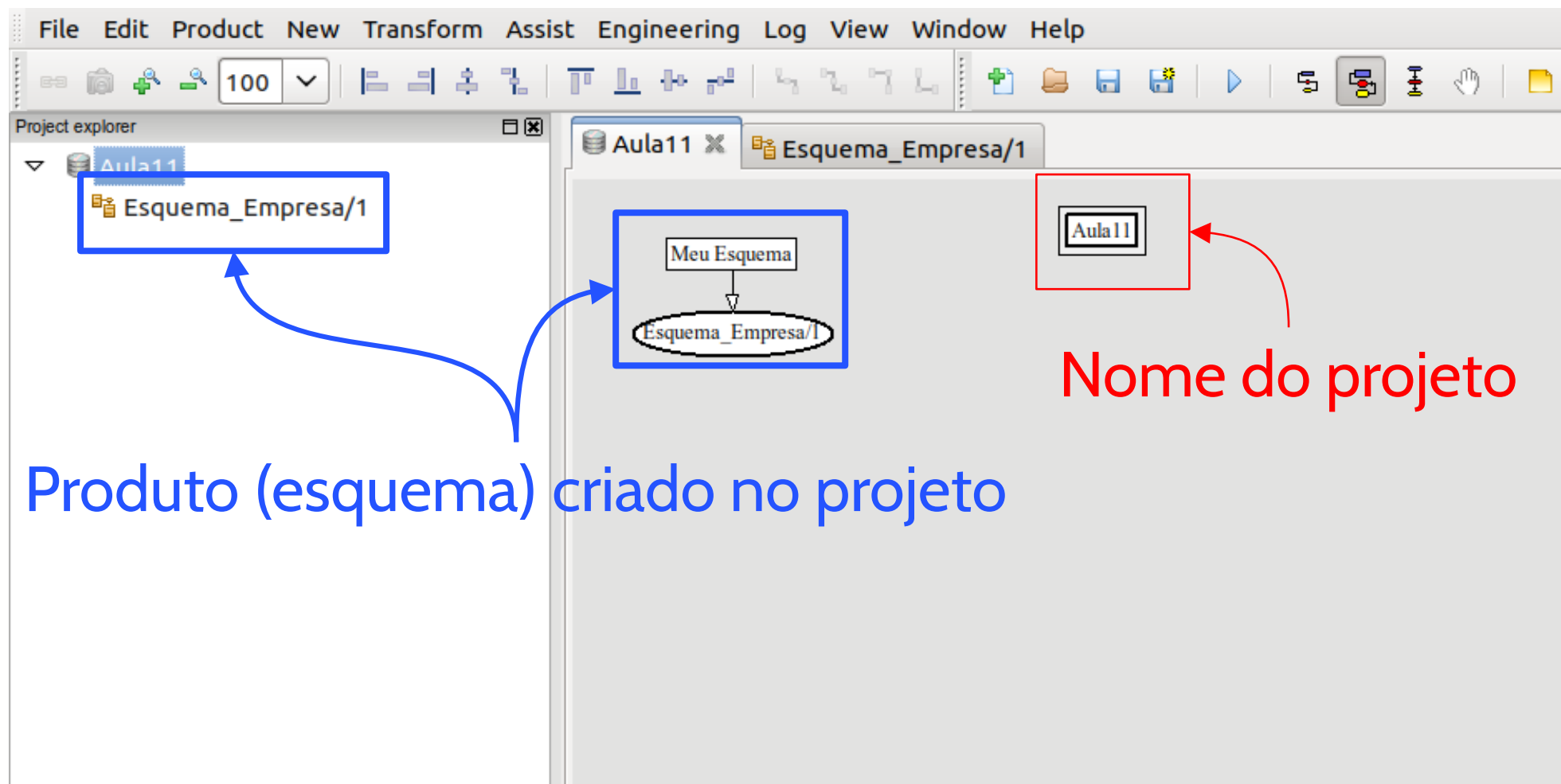
☒ Entity/Relationship schema ☐ UML class diagram  
☐ UML activity diagram ☐ UML use cases diagram

Schema type: [dropdown]

Ok Cancel



### 3. DB-Main







### 3. DB-Main

DB-MAIN

File Edit Product **New** Transform Assist Engineering Log View Window Help

Project explorer

- Aula11
  - Esquema\_Empresa

Collection...  
Entity type...  
Rel-type...  
Attribute  
Role/rel-type/link...  
GR Group...  
Constraint...  
PU Processing unit...

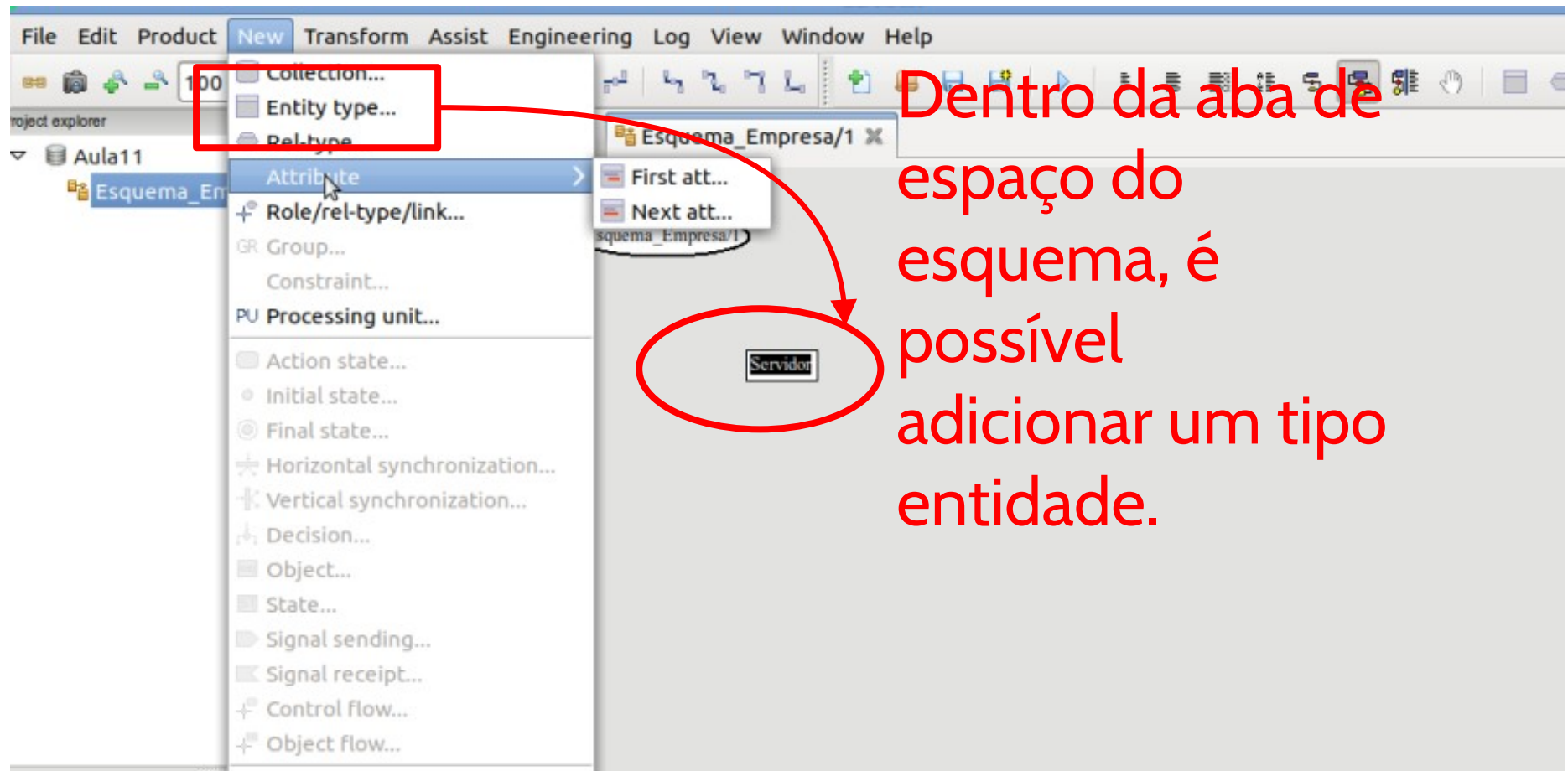
Esquema\_Empresa/1

Na aba de trabalho do respectivo produto/esquema, as possibilidades de objetos válidos são apresentadas.

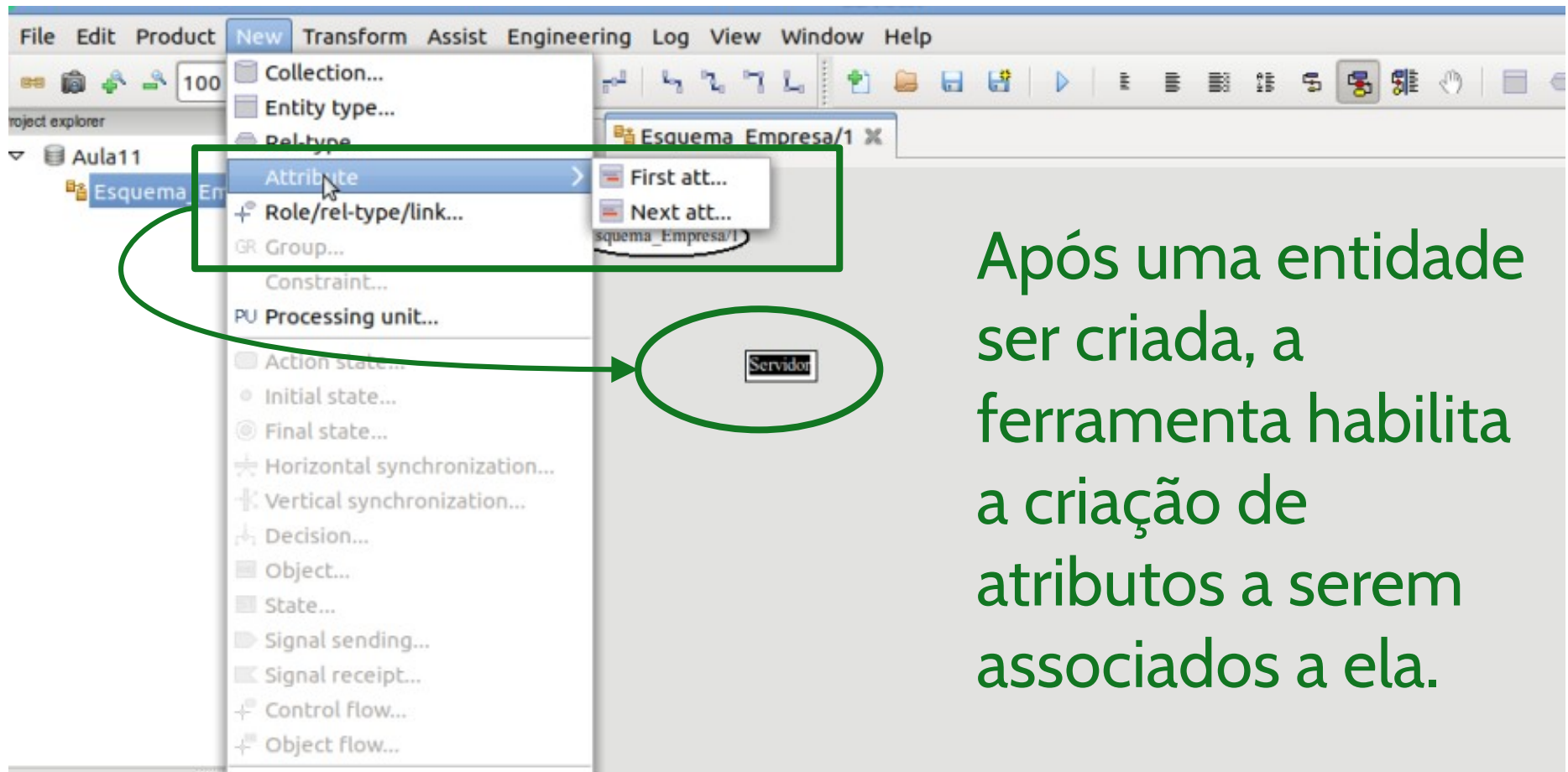
Uma vez que criamos um dado esquema, podemos adicionar elementos (entidades, relacionamentos, etc) nele.



### 3. DB-Main



## 3. DB-Main

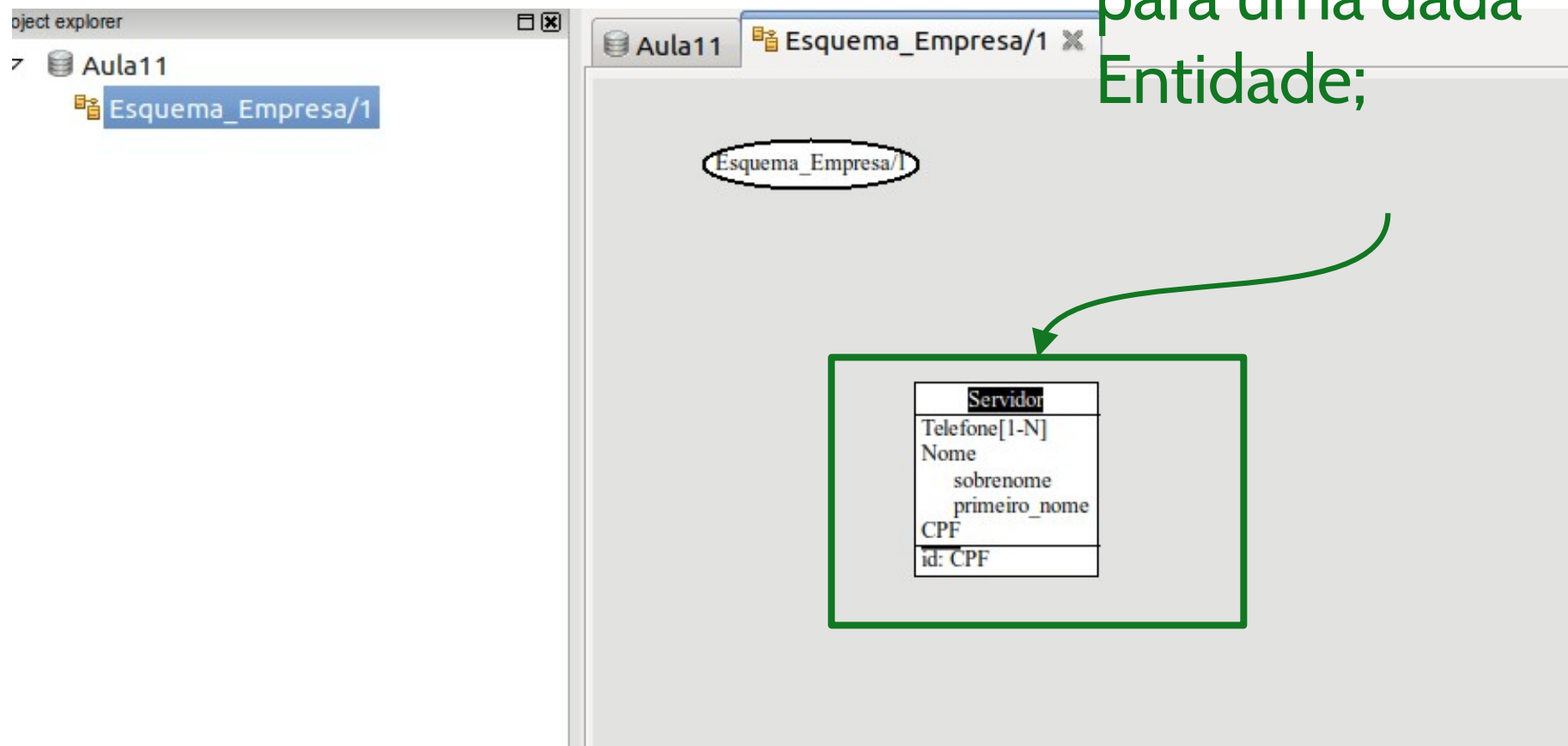


Após uma entidade ser criada, a ferramenta habilita a criação de atributos a serem associados a ela.



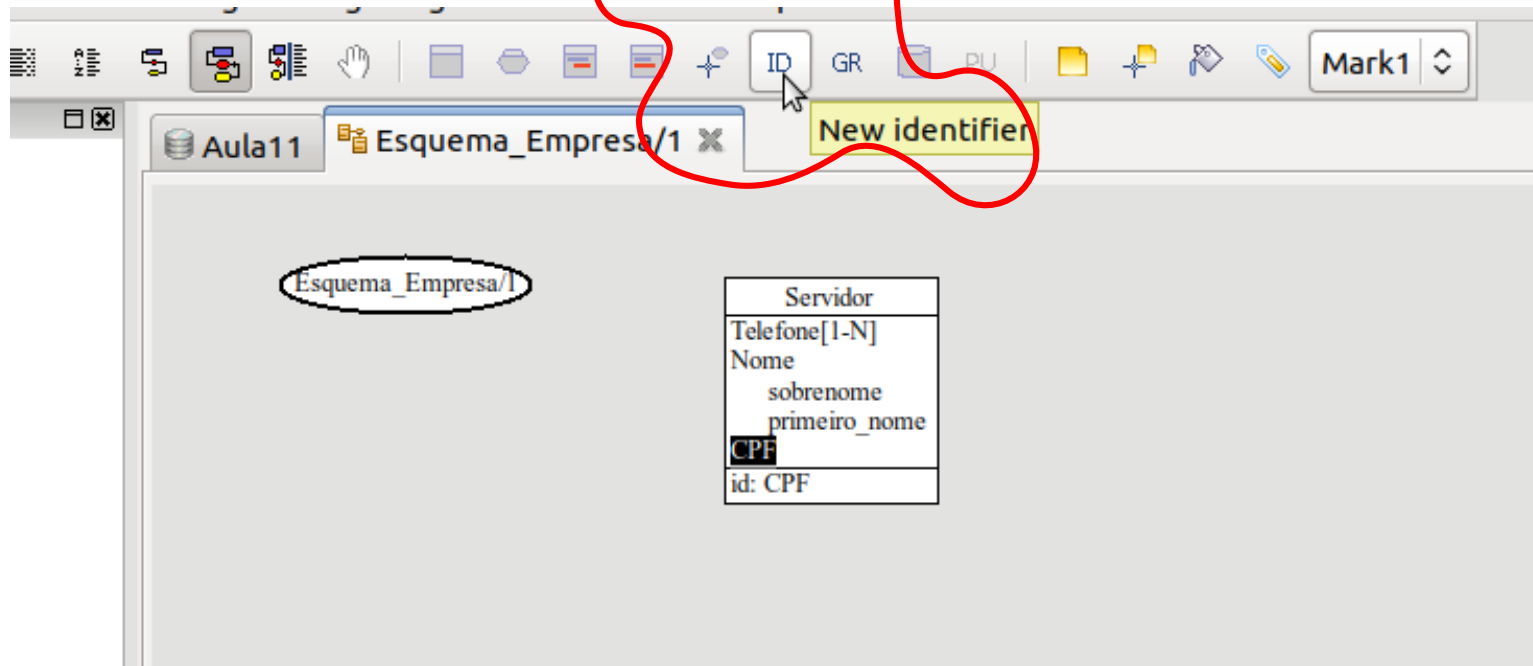
### 3. DB-Main

Vários atributos  
podem ser criados  
para uma dada  
Entidade;



### 3. DB-Main

Usamos essa opção para  
setar um atributo como  
chave.

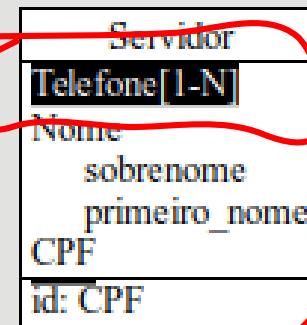


## 3. DB-Main

Property box

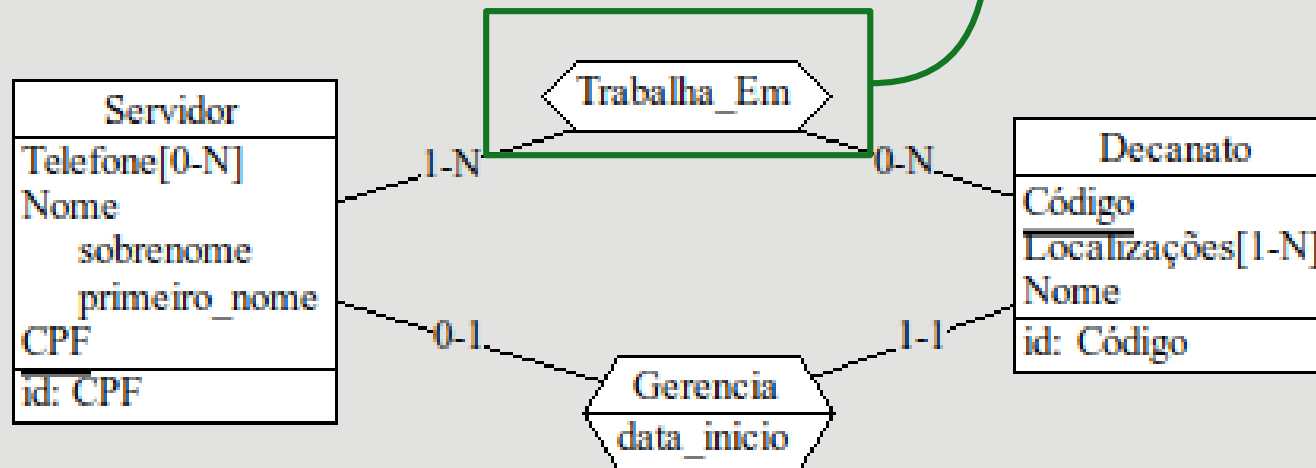
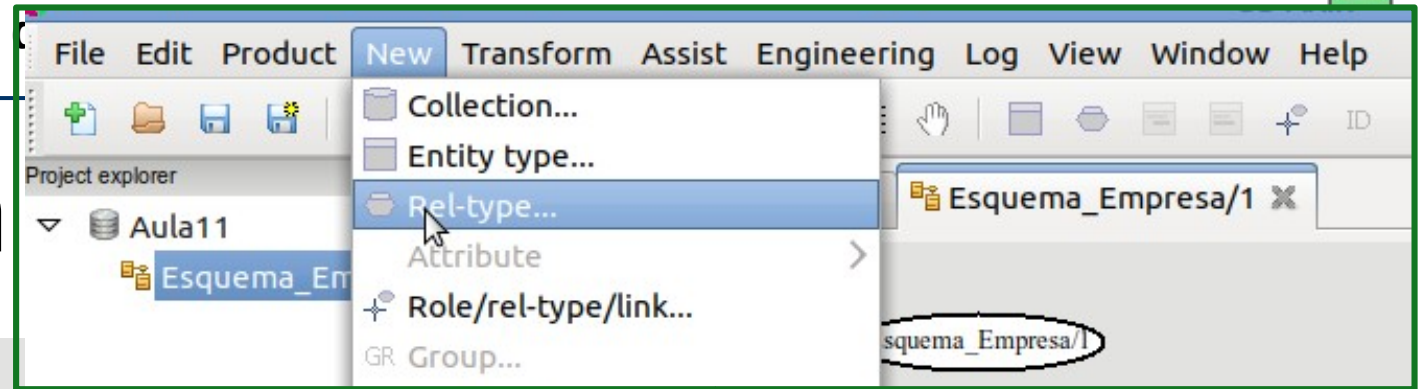
attribute: Servidor.Telefone

Prop	User prop	Sem	Tech
name	Telefone		
short name			
cardinality	1-N		
collection type	set		
type	char		
length	1		
decimal length			
domain			
stable	<input type="checkbox"/>		
non recyclable	<input type="checkbox"/>		



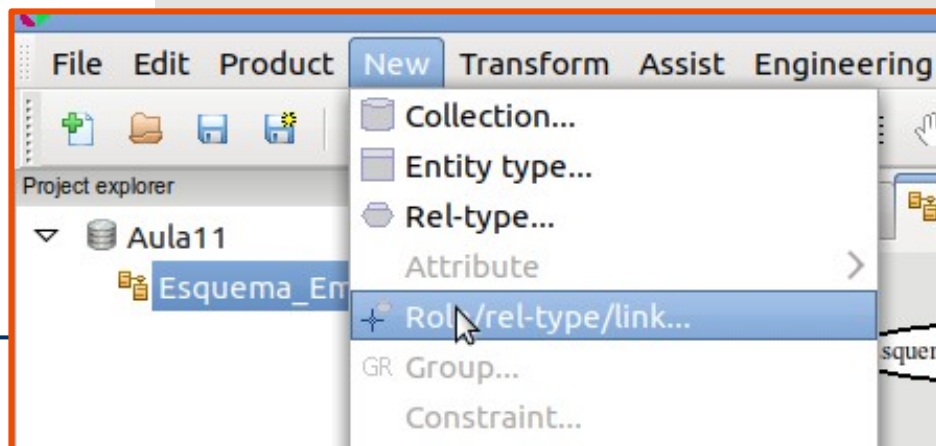
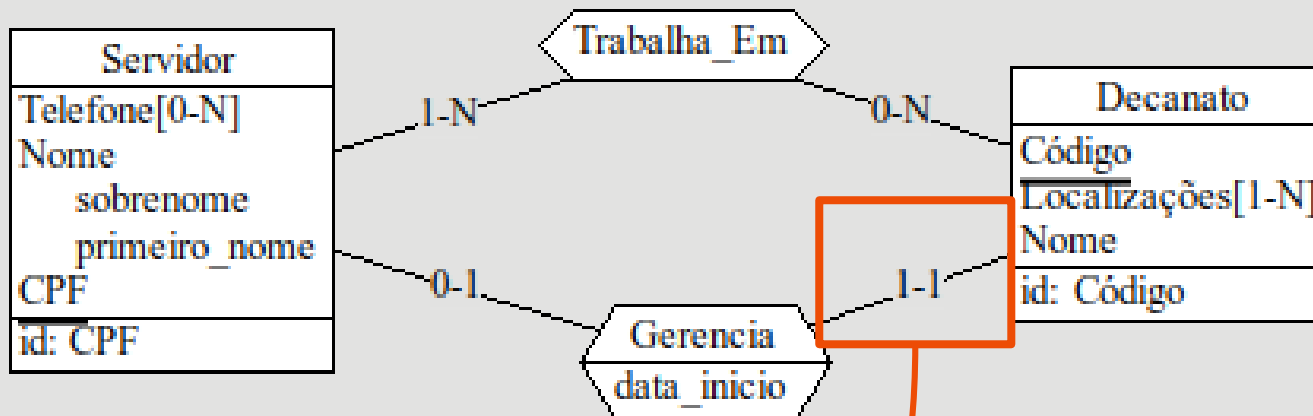
A cardinalidade de um atributo é modificada na caixa de propriedades.

### 3. DB-Main



Um relacionamento é criado via esse menu. Note que na DB-Main o relacionamento não é um losango.

## 3. DB-Main

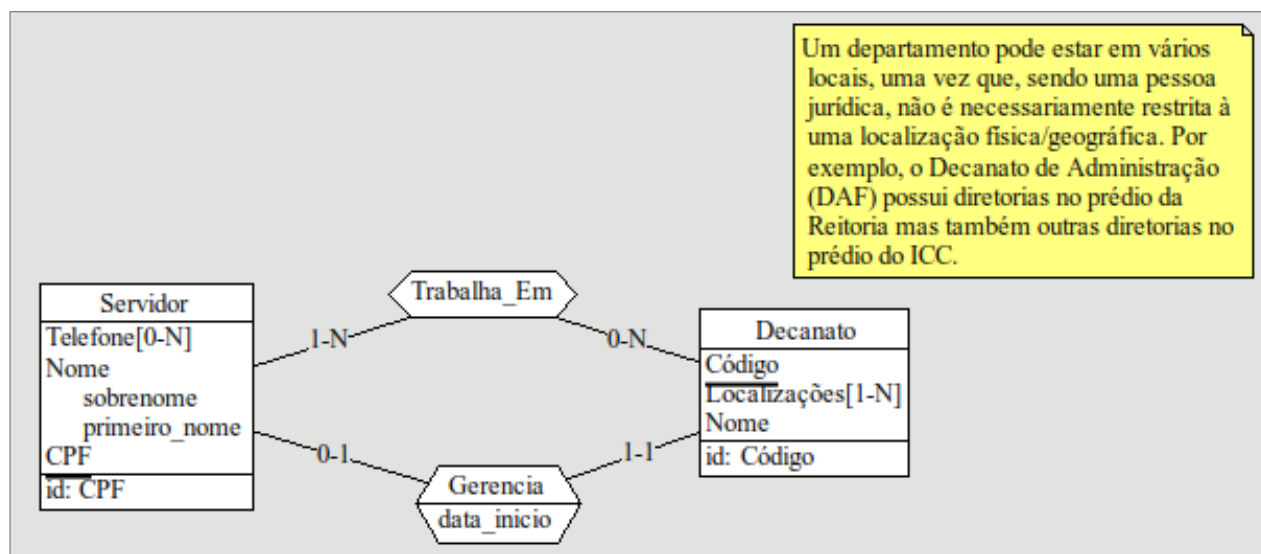
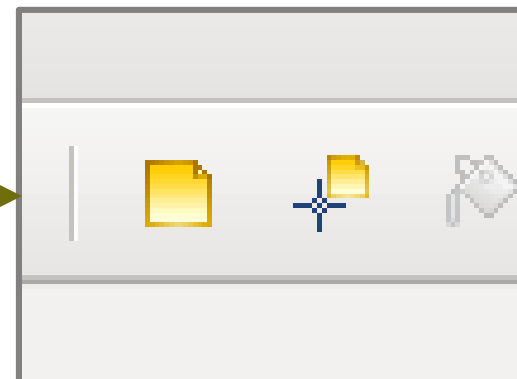


Uma ligação de relacionamento é via este menu.



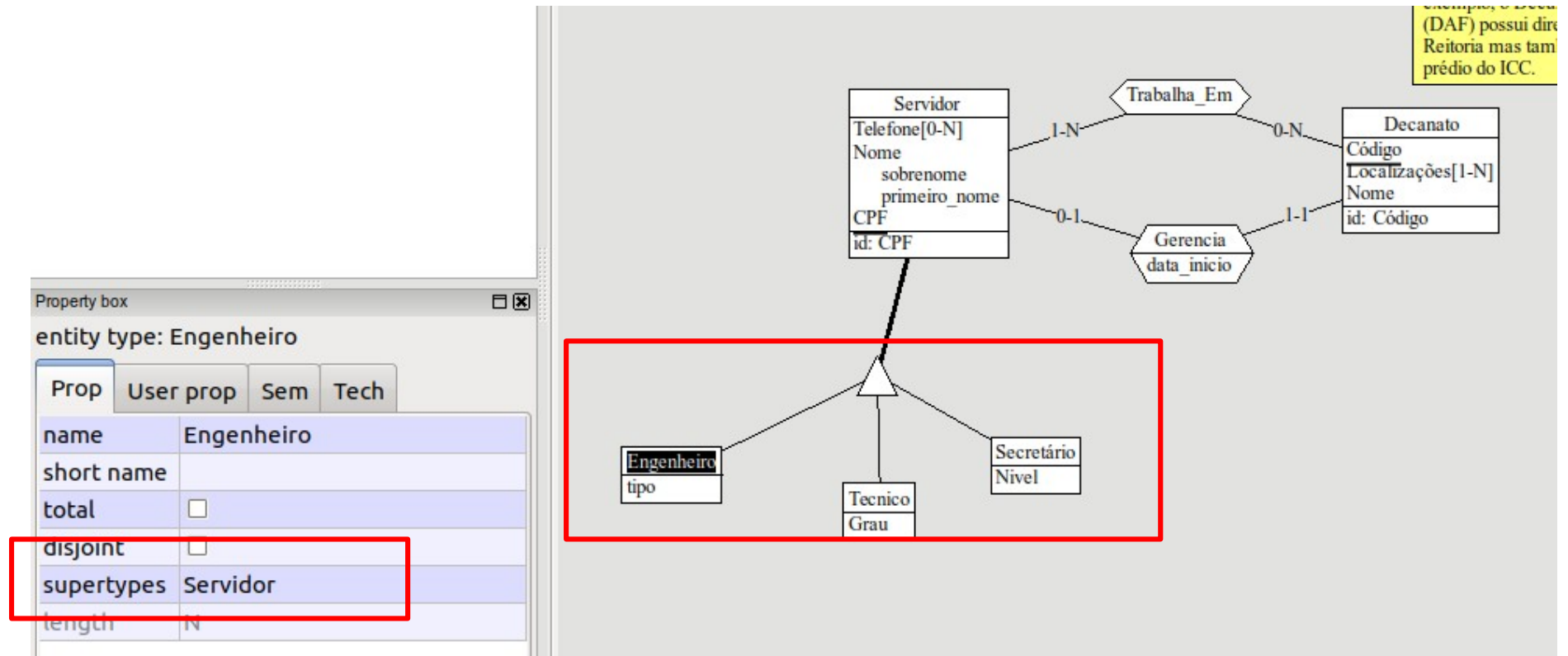
### 3. DB-Main

Notas podem ser inseridas usando essa opção de menu. Elas não entram no modelo.



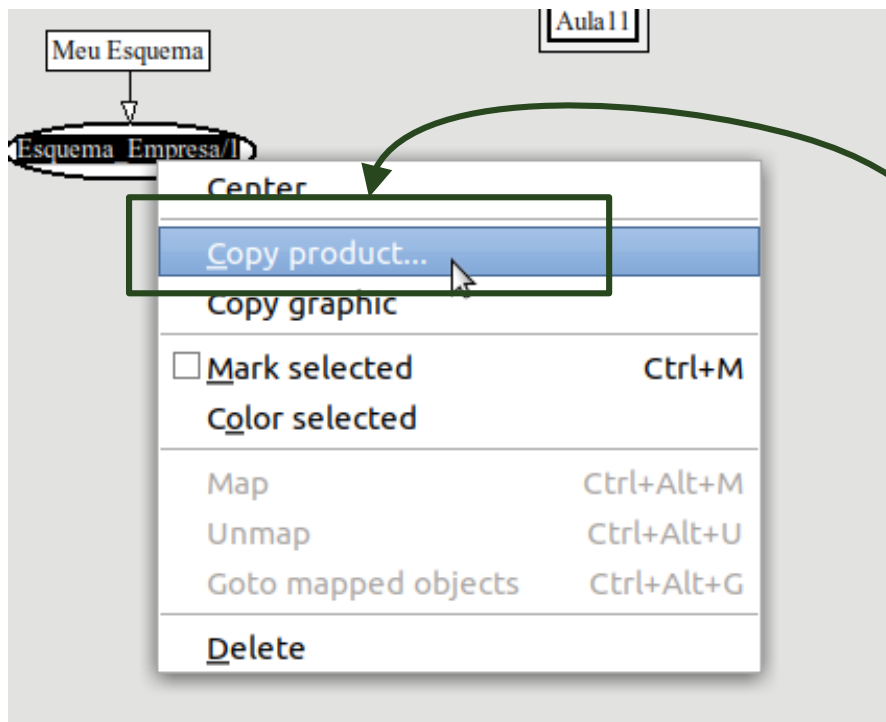
### 3. DB-Main

Definimos as hierarquias de especialização através dessa propriedade da entidade derivada.



### 3. DB-Main

Mapeamento MER → MR



Crie uma cópia do esquema usando essa opção



## 3. DB-Main

Mapeamento MER → MR

Essa janela deve aparecer

Schema Properties

Copy schema

Name: Esquema\_Empresa\_Relacional

Short name: Empresa

Version: 1-1

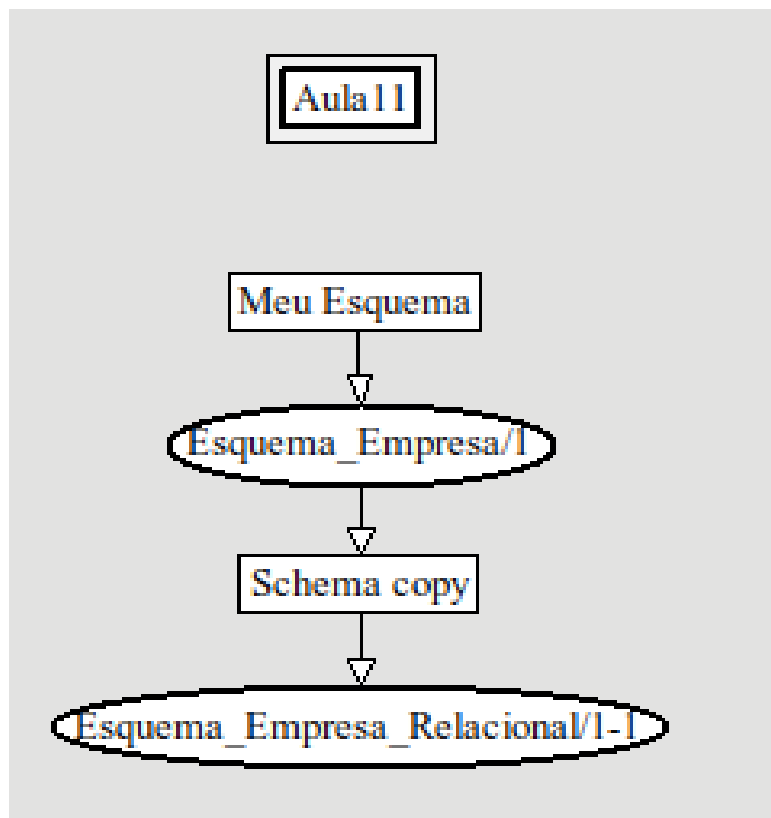
☒ Entity/Relationship schema ☐ UML class diagram  
☐ UML activity diagram ☐ UML use cases diagram

Schema type: [dropdown]

Ok Cancel

## 3. DB-Main

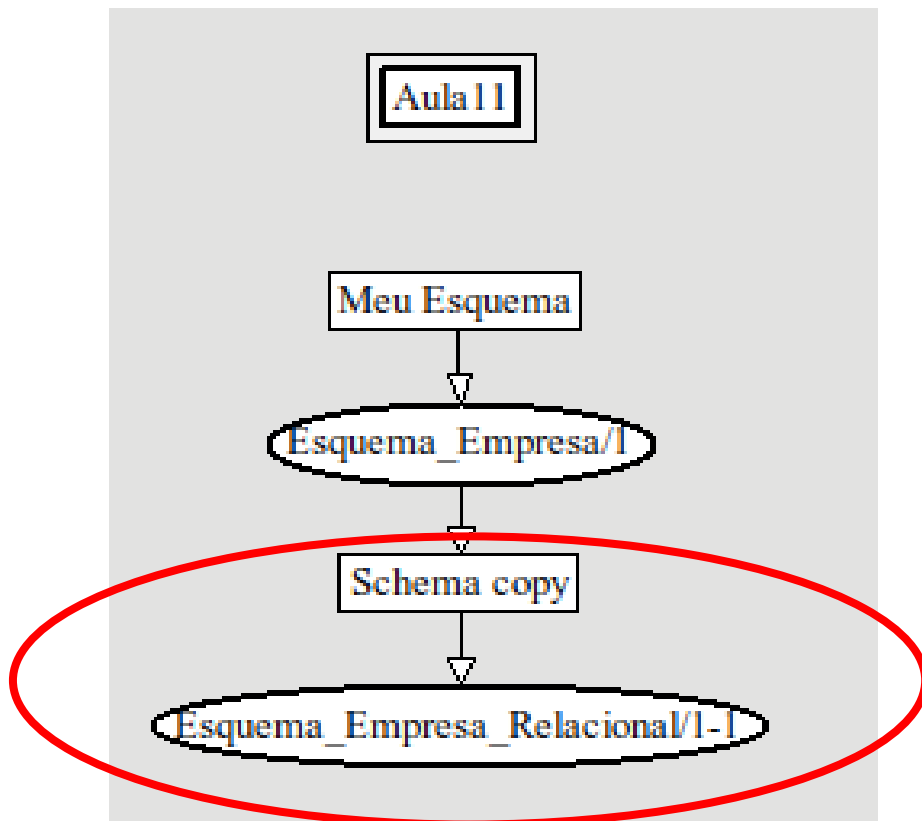
### Mapeamento MER → MR



- Uma cópia do esquema é criado e deve aparecer assim na aba do projeto.

## 3. DB-Main

### Mapeamento MER → MR



- Uma cópia do esquema é criado e deve aparecer assim na aba do projeto.
- É a partir dessa cópia que faremos a transposição do modelo.



### 3. DB-Main

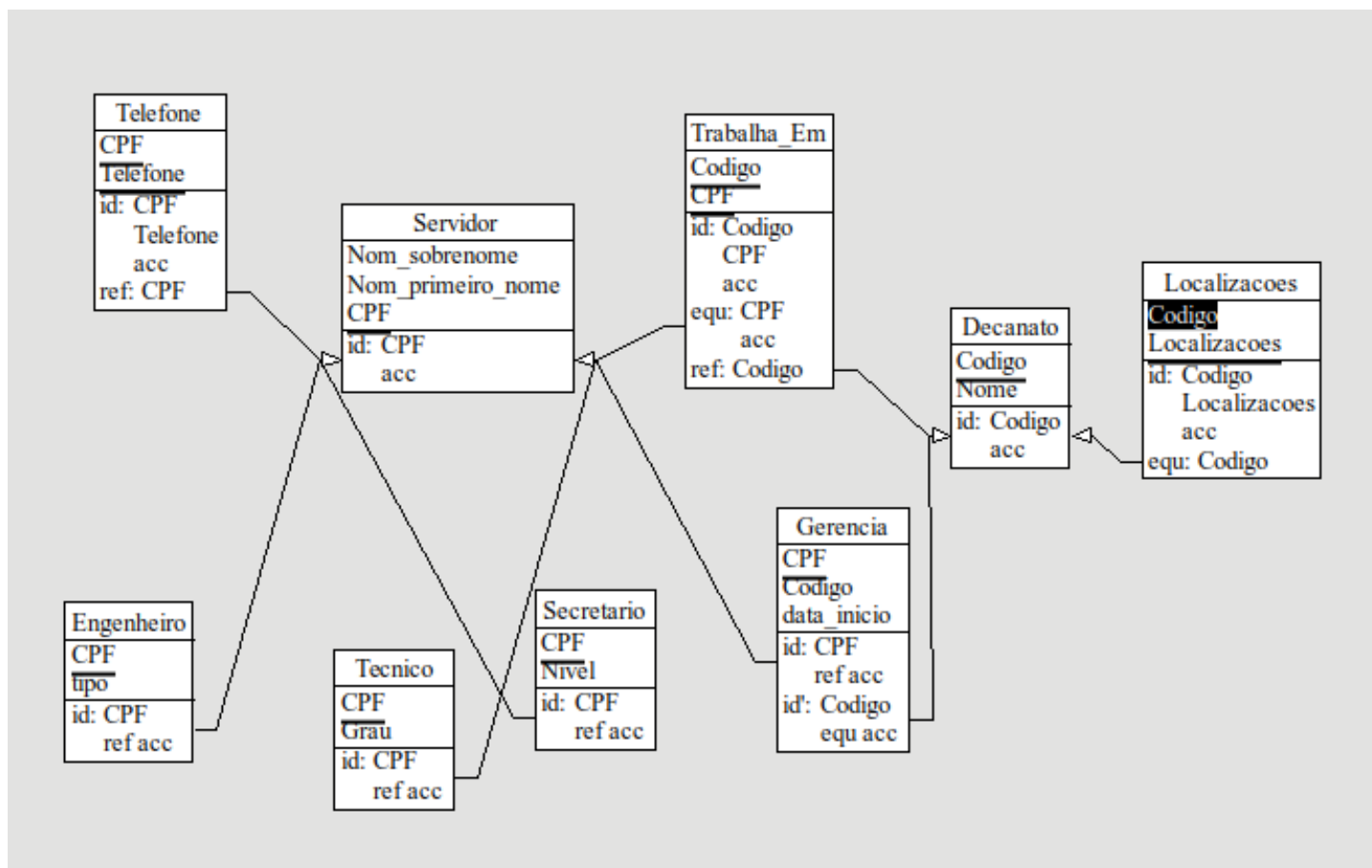
Mapeamento MER → MR

The screenshot shows the DB-Main software interface. The 'Transform' menu is open, highlighting the 'Relational model' option. A green arrow points from this menu item to a text box. Another green arrow points from the 'Relational model' option to a diagram in the background. The diagram shows a transformation from an Entity-Relationship (ER) model to a Relational (MR) model. The ER model includes entities like 'Engenheiro' and 'Tecnico' with attributes like 'nome', 'sobrenome', 'id: CPF', and 'telefone'. The MR model shows a table 'Esquema\_Empresa\_Relacional' with columns like 'nome', 'short name', and 'version'. The text box contains the following text:

A partir da aba do modelo copiado, transformamos o MER no MR via esse menu.

## 3. DB-Main

Mapeamento MER → MR

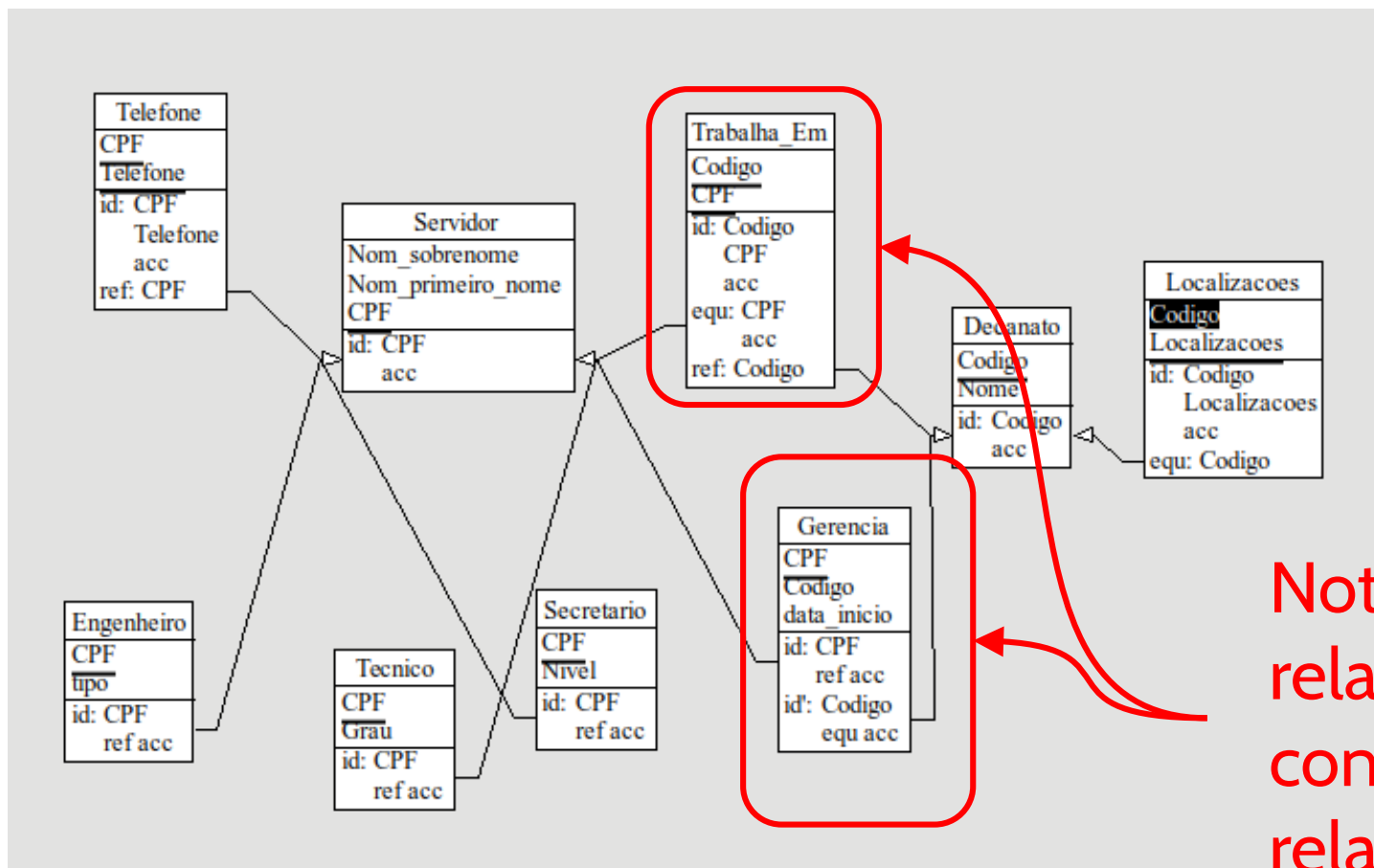


MR gerado



### 3. DB-Main

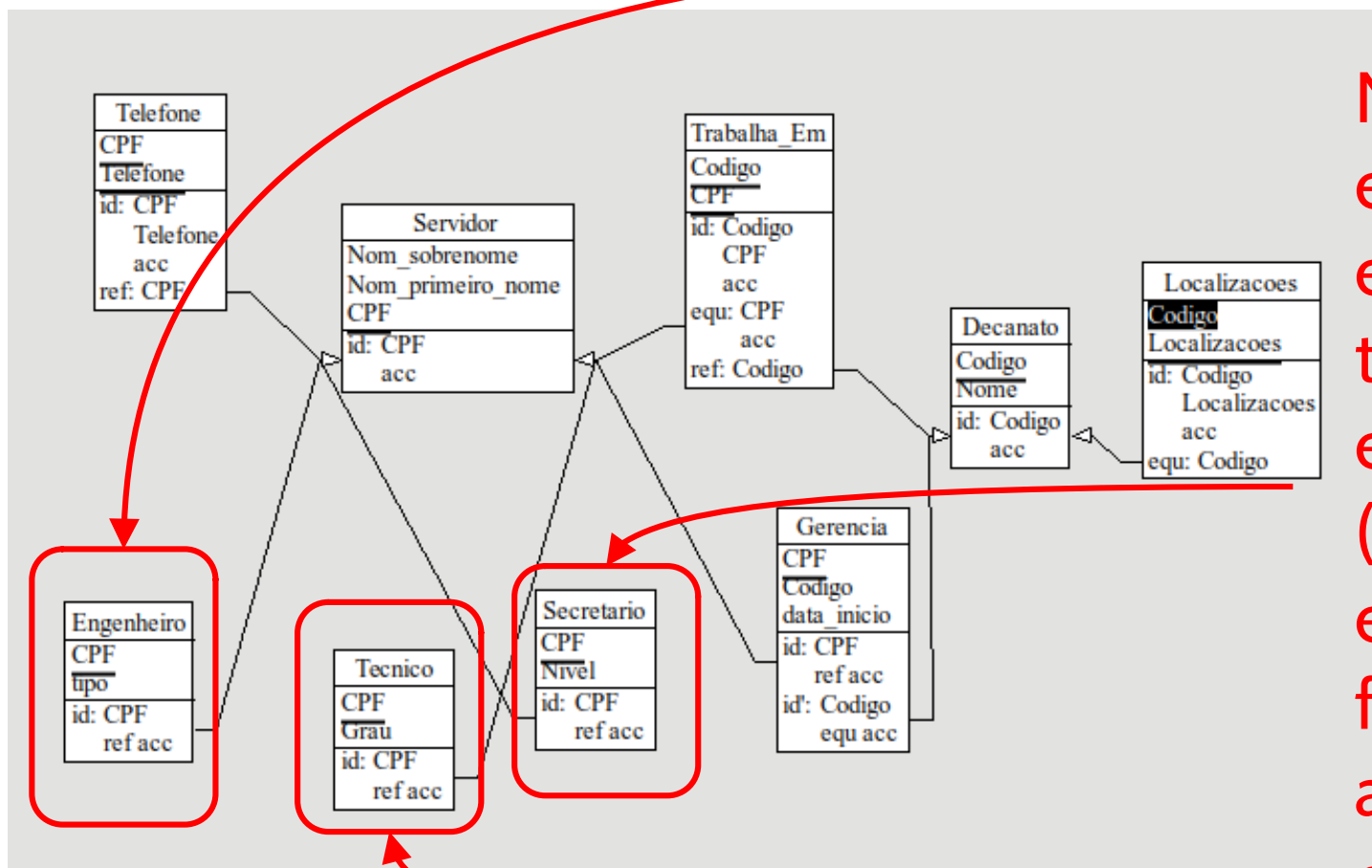
Mapeamento MER → MR



Note que os relacionamentos se converteram em relações!

### 3. DB-Main

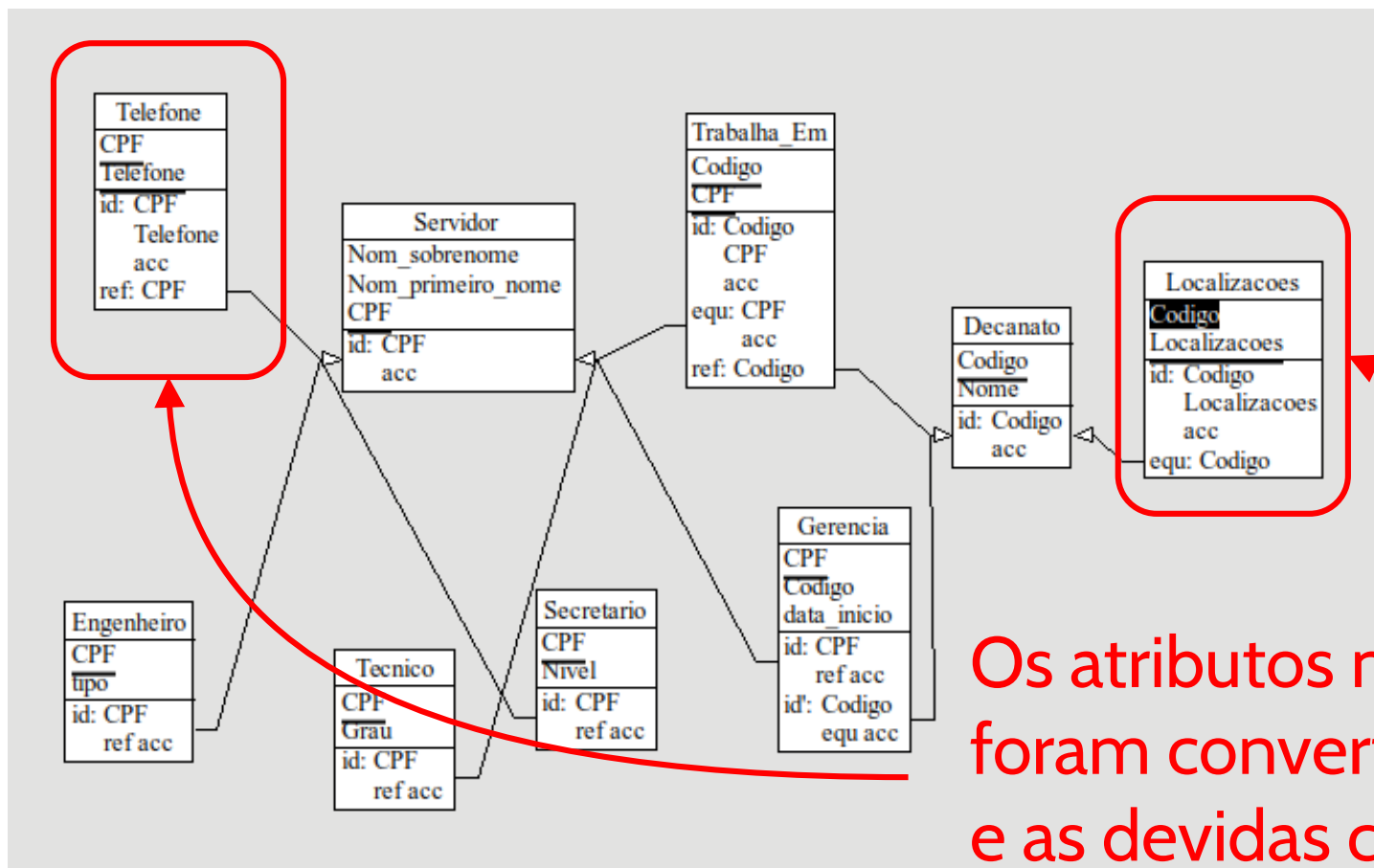
### Mapeamento MER → MR



Note que as entidades especializadas se tornaram relações e as ligações (chaves estrangeiras) foram automaticamente criadas.

### 3. DB-Main

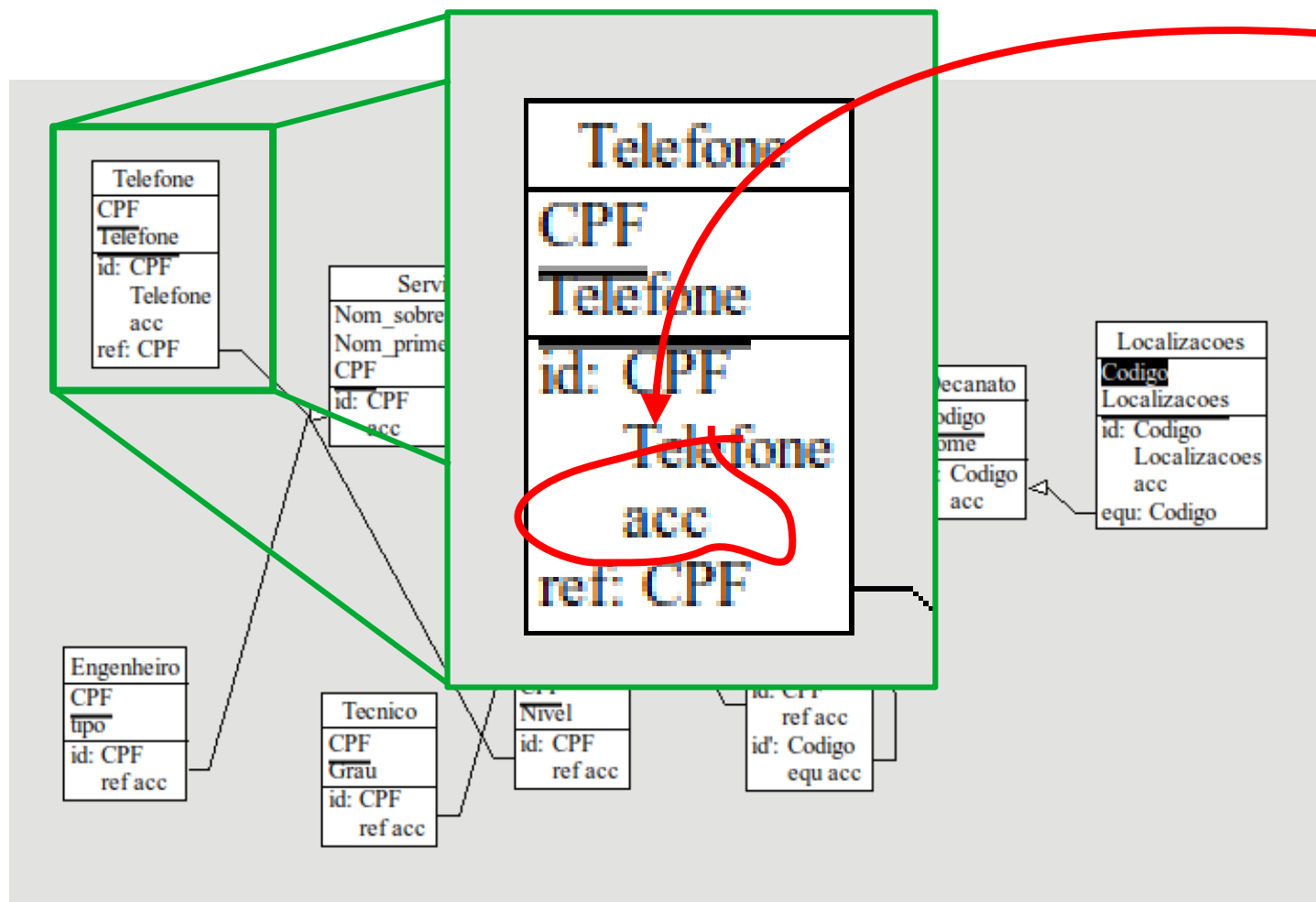
Mapeamento MER → MR



Os atributos multivalorados foram convertidos em relações e as devidas chaves criadas.

### 3. DB-Main

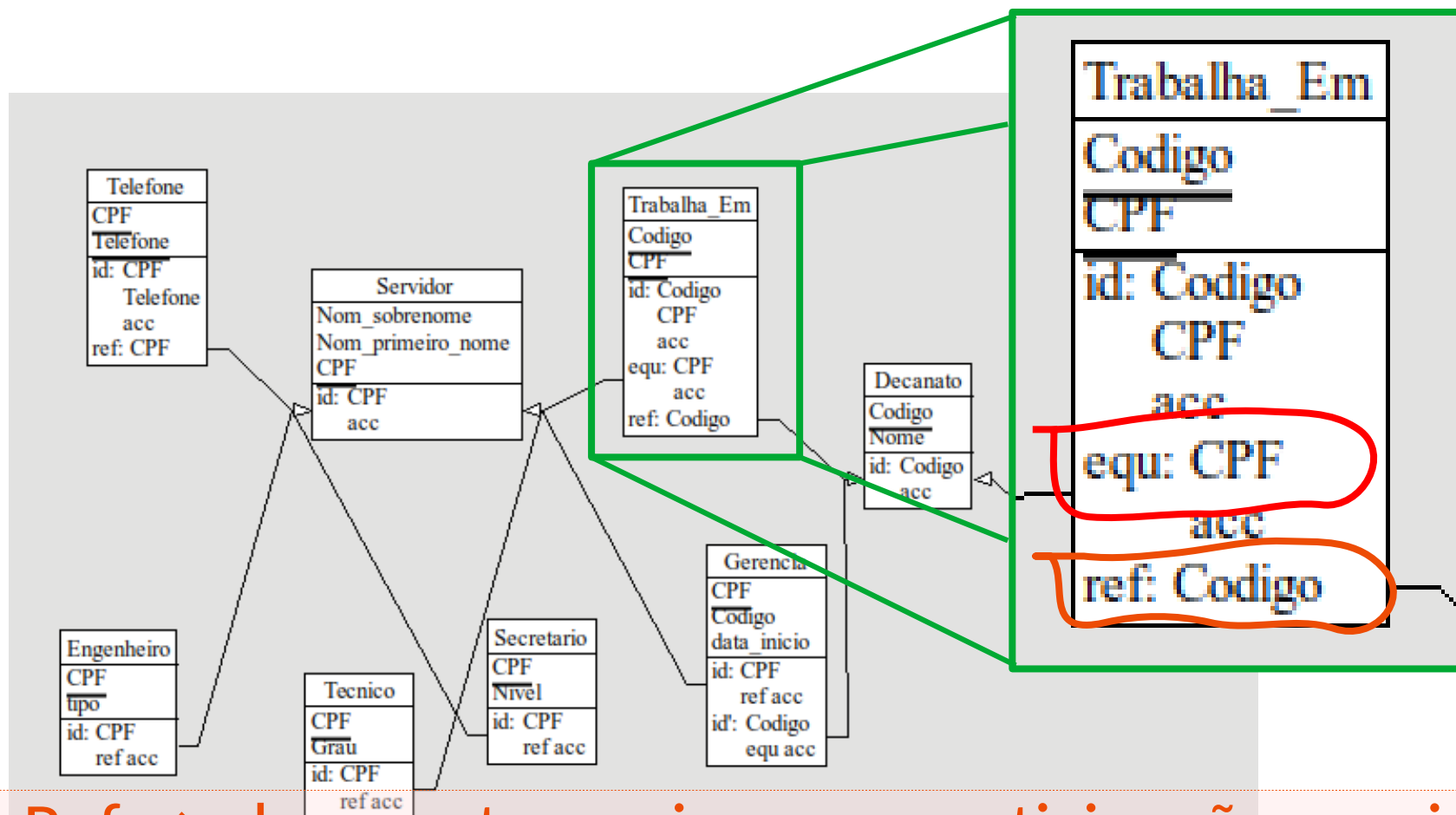
### Mapeamento MER → MR



O DB-Main cria o campo acc nas relações que representam os atributos multivalorados para indicar ao que se trata de um índice incremental.

### 3. DB-Main

Mapeamento MER → MR

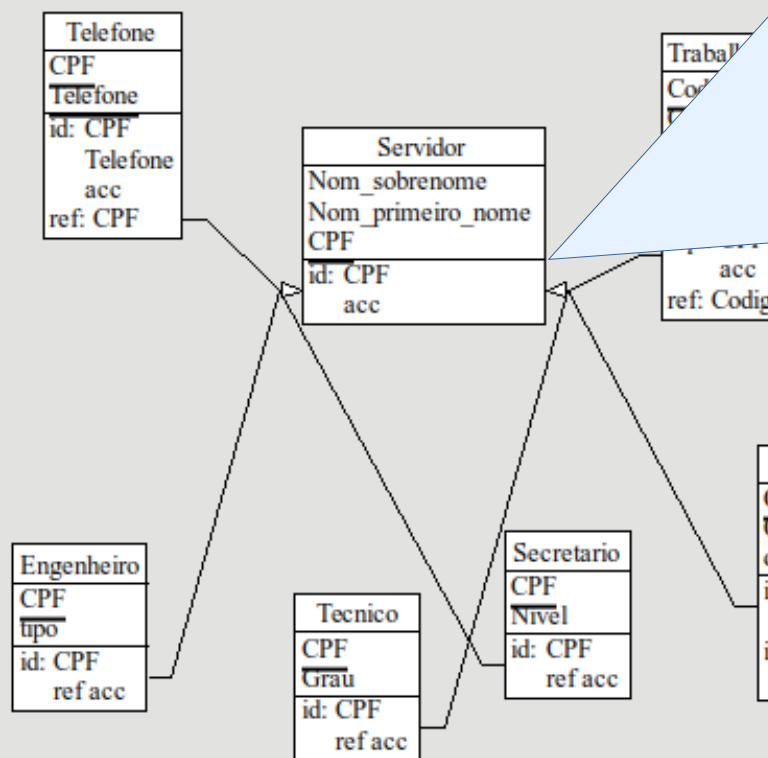


- Ref → chave estrangeira com participação parcial
- Equ → chave estrangeira com participação total

## 3. DB-Main

### Mapeamento MR → SQL

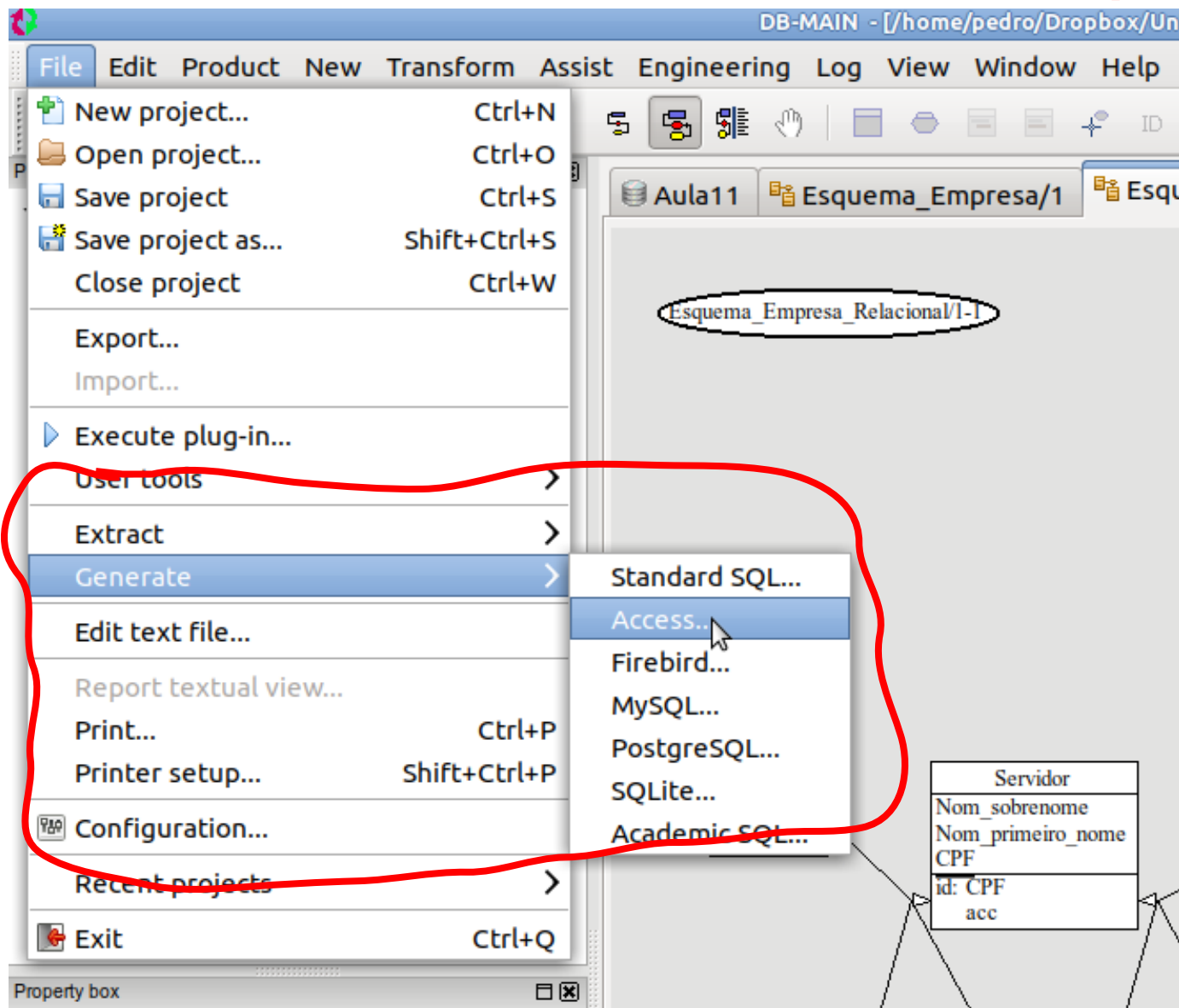
A partir do Modelo Relacional gerado pelo DB-Main, podemos criar um esquema em SQL que pode ser importado apropriadamente num SGBD para criar um banco de dados conforme modelado conceitualmente pela ferramenta.





### 3. DB-Main

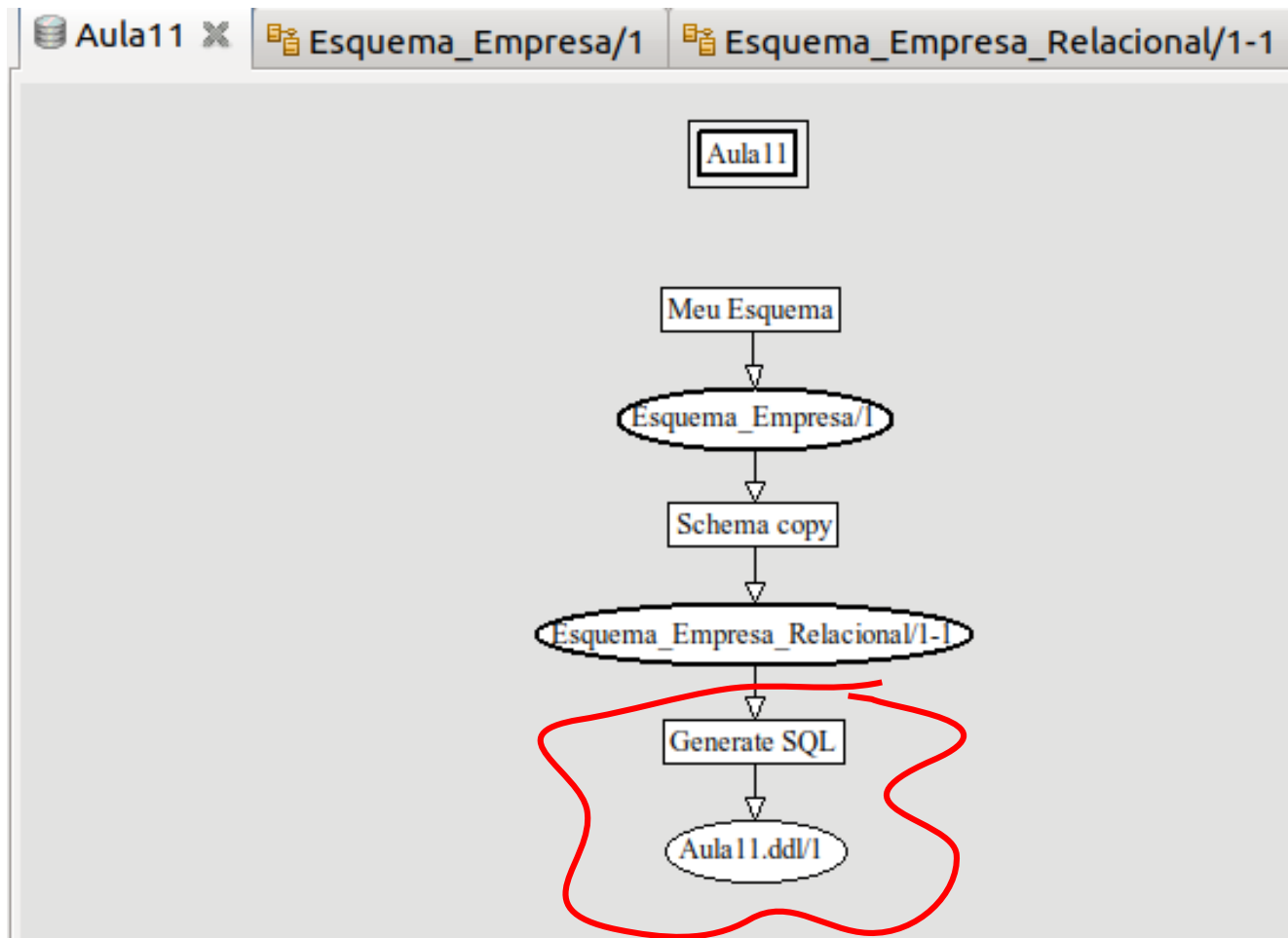
### Mapeamento MR → SQL



- Usamos essa opção para gerar o script SQL equivalente ao modelo relacional.
- O DB-Main irá criar um arquivo **.ddl** (DDL=Data Definition Language)

## 3. DB-Main

### Mapeamento MR → SQL



- Se tudo correr bem, a aba correspondente ao projeto terá esse processo adicionado ao fluxo do projeto.





## Mapeamento MR → SQL

Exemplo de modelo relacional representado em linguagem SQL (*DDL subset*) e gerado automaticamente pela ferramenta CASE DB-Main.

```
pedro@suse:...pbox/UnB/BD/2023.1 70x35
GNU nano 6.3 Aula11.ddl
*****
* Standard SQL generation
*****
* DB-MAIN version: 11.0.2
* Generator date: Sep 20 2021
* Generation date: Tue May  2 15:21:07 2023
* LUN file: /home/pedro/Dropbox/UnB/BD/2023.1/Aula11.lun
* Schema: Esquema_Empresa_Relacional/1-1
*****

-- Database Section
-- -----

create database Esquema_Empresa_Relacional;

-- DBSpace Section
-- -----

-- Tables Section
-- -----

create table Decanato (
  Codigo char(1) not null,
  Nome char(1) not null,
  constraint ID_Decanato_ID primary key (Codigo));

create table Engenheiro (
  [ Read 161 lines ]

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify
[0] 0:nano*  "suse" 15:33 02-May-23
```

A word cloud featuring the word "THANK YOU" in large, bold, black capital letters. Surrounding it are various translations of "thank you" in different languages, also in bold, black capital letters, but in smaller sizes. The translations include: DANKSCHEEN, TASHAKKUR ATU, GRACIAS, ARIGATO, SHUKURIA, GOZAIMASHITA, FCHARISTO, JUSPAXAR, KOMPASUMNDA, MAKE, LAH, GRAZIE, MEHRBANI, PALMES, BOLZİN, MERCI, BİYAN, SHUKRIA, TINGKI, YAQHANYELAY, SUKSAMA, EKHMET, and many others. The word cloud is set against a white background.



# Dúvidas?

