

FIAP GRADUAÇÃO

# ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

## DATABASE MODELING & SQL

Profa. Rita de Cássia Rodrigues



[rita@fiap.com.br](mailto:rita@fiap.com.br)

EXTENSÕES DO MODELO ENTIDADE-RELACIONAMENTO  
AGREGAÇÃO em Banco de Dados (ENTIDADE ASSOCIATIVA)

- ✓ Objetivo.
- ✓ Conceitos referentes a Modelo Entidade-Relacionamento, suas extensões.
- ✓ Revisão dos Conceitos.
- ✓ Exercícios.

- ☐ Explicar os conceitos de modelagem de dados.
- ☐ Caracterizar o modelo entidade-relacionamento, suas extensões.
- ☐ Projetar banco de dados, identificar e abstrair as necessidades.
- ☐ Aplicar os conceitos trabalhados para construir um modelo de dados.

- ❑ Modelo Entidade-Relacionamento – Extensões

  - ✓ Entidade Associativa – Agregação em Banco de Dados

- ❑ Exercícios

Podemos dizer que um **modelo de dados**, **pode e deve** passar por um **processo de refinamento**. Observamos algumas situações particulares, que chamamos de **extensões**, cuja **função** é de **possibilitar a correção** destas **particularidades** para que o modelo de dados possa ser posteriormente implementado de forma física, através de um SGBD.

A **agregação** é uma forma de **ajustar a implementação de um relacionamento do tipo muitos para muitos**, onde um **relacionamento** é **representado por uma entidade**.

Não é possível implementar fisicamente um relacionamento muitos para muitos, considerando que uma chave estrangeira não pode ser multivalorada. Uma chave estrangeira deve fazer referência a um único valor (monovalorada).

Quando ajustamos um relacionamento do tipo muitos para muitos, observamos que em boa parte das situações, conseguimos encontrar **novos atributos que caracterizam essa relação** (atributos do relacionamento).

Como encontramos atributos que caracterizam o relacionamento e ainda há a necessidade de manter a chave estrangeira monovalorada, transformamos o relacionamento e os atributos que caracterizam este relacionamento em um objeto agregado, ou seja, em outras palavras **criamos uma nova entidade** para representar a associação, ou seja, criamos uma entidade **ASSOCIATIVA**.

Uma entidade associativa não existe por si só em um modelo entidade relacionamento, sua existência está condicionada a existência de duas ou mais entidades existentes no modelo.

É chamado de agregação pois a nova entidade agrega atributos (chaves PK) das duas entidades que anteriormente se associavam, mais os atributos que caracterizam o relacionamento.

Normalmente encontramos um nome que caracteriza a nova entidade (associativa) em função das necessidades de negócio.

Há situações onde não encontramos atributos que caracterizam o relacionamento, em um relacionamento muitos para muitos, neste caso utilizamos uma nova entidade com intuito de ligação e solução para as chaves estrangeiras, de modo que sejam monovaloradas. Neste caso teremos uma entidade que receberá apenas as chaves estrangeiras oriundas das entidades associadas.

Quando temos entidades apenas de ligação (associativas) normalmente não conseguimos identificar um nome que represente a entidade, no momento de nomear esta entidade utilizamos os nomes das entidades que se associavam anteriormente.

Trabalharemos mais à frente, alguns exemplos para deixar mais claro estes conceitos.



Suponha a situação onde um médico pode avaliar, examinar, consultar um paciente e que um paciente possa ser avaliado, examinado, consultado por um médico.

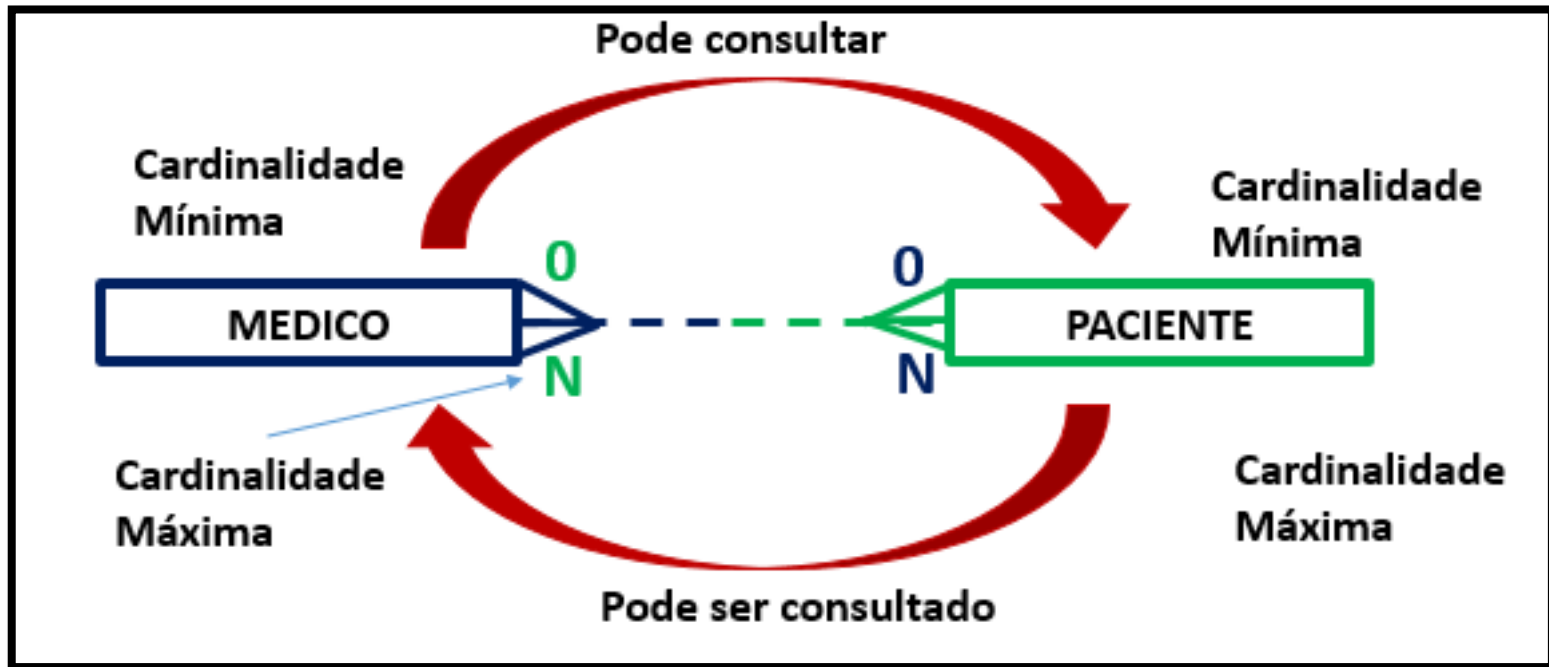
Cada médico pode avaliar, examinar, consultar vários pacientes, inclusive o mesmo paciente várias vezes.

Cada paciente pode ser avaliado, examinado, consultado por vários médicos, inclusive com o mesmo médico várias vezes.

Podemos representar a associação entre médico e paciente da seguinte maneira:



Exemplificando de uma forma um pouco mais detalhada:



Observe que temos um relacionamento com cardinalidade M:N.



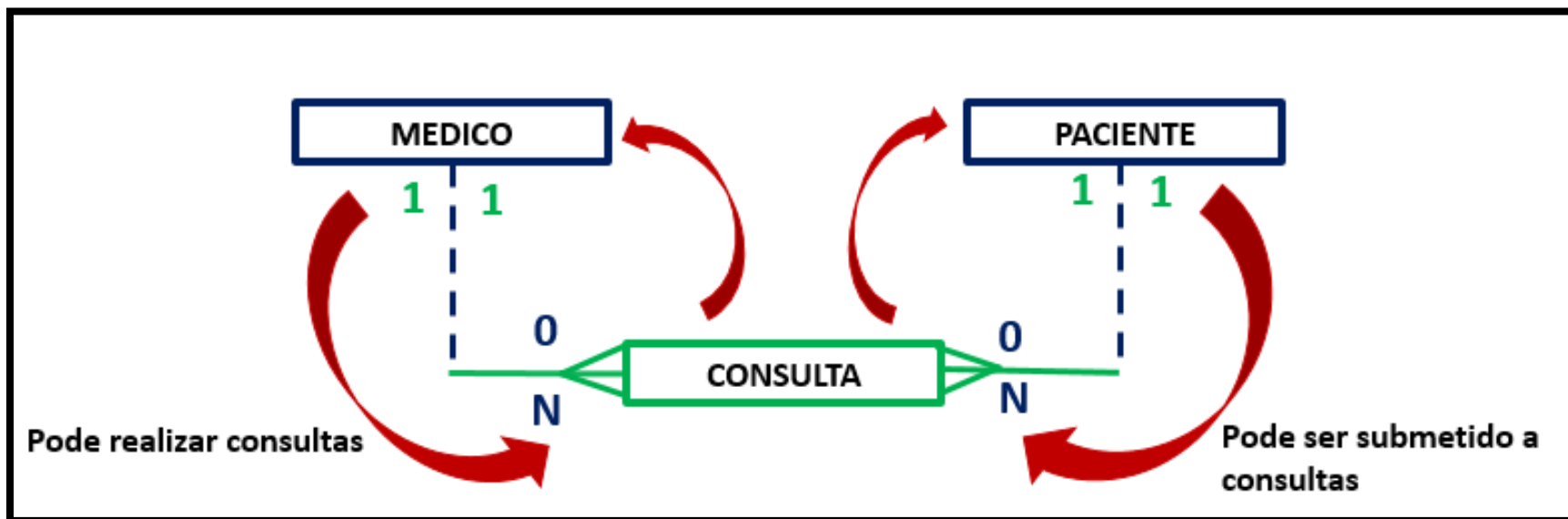
Veja que quando pensamos em uma consulta médica, conseguimos encontrar atributos que caracterizam o relacionamento M:N. Por exemplo, uma consulta ocorre em uma data e hora e em uma determinada sala ou consultório.

Fique atento, data e hora da consulta não é uma característica (atributo) de Médico e nem de Paciente, podemos dizer que a sala ou consultório também não caracteriza Médico ou Paciente, ou seja, estas características só existem quando uma consulta existir, por este motivo são características (**atributos**) **do relacionamento**. Se a consulta não existir, estas características não existem.

A partir do momento que encontramos os atributos que caracterizam o relacionamento, observamos mais nitidamente a necessidade de uma nova entidade.

Neste exemplo, o nome mais adequado para a nova entidade é “CONSULTA”.

Quando transformamos o relacionamento em uma nova entidade, passamos a ter dois relacionamentos 1:N, um entre “MÉDICO” e “CONSULTA” e outro entre “PACIENTE” e “CONSULTA”.



Características da entidade **CONSULTA** →

Data e hora

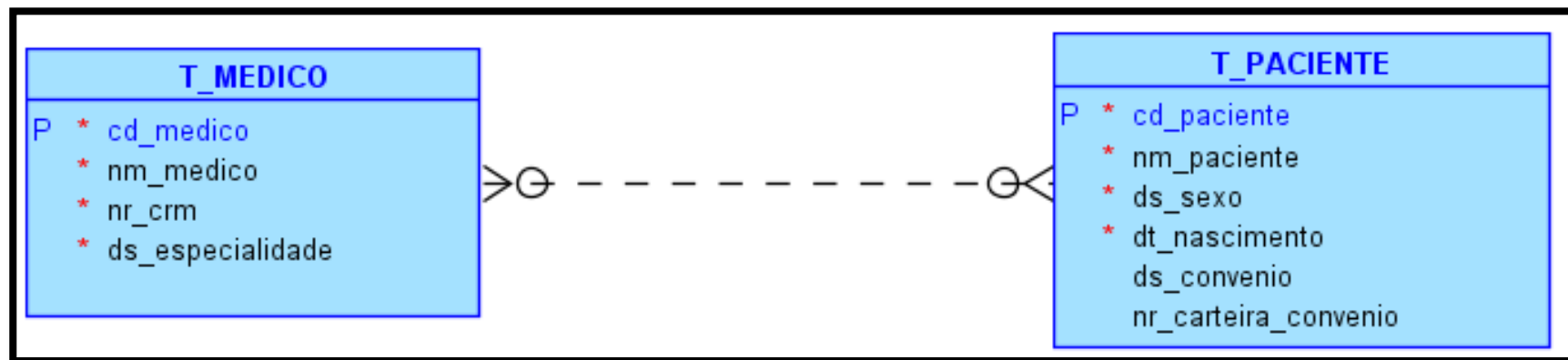
Sala ou consultório



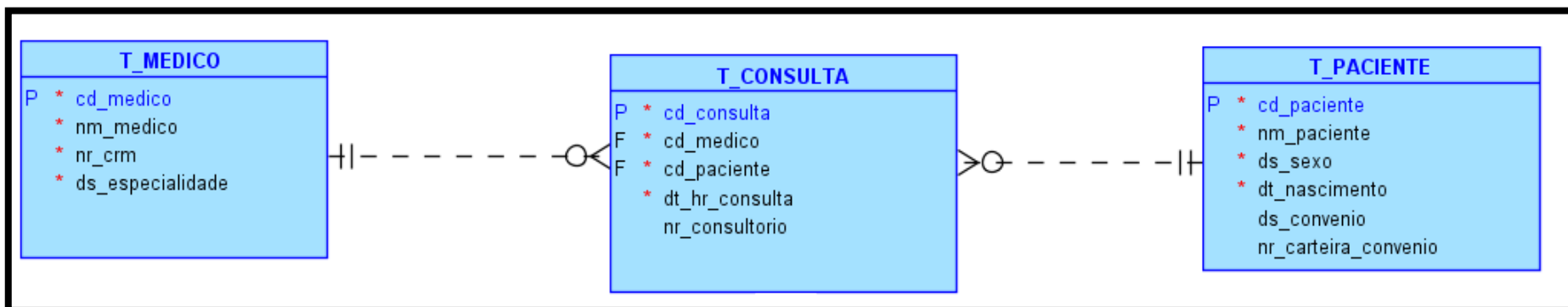
# Exemplificando Entidade Associativa - Agregação

Utilizando a representação gráfica através da ferramenta SQL Developer Data Modeler, a partir da notação da Engenharia da Informação:

Representação do relacionamento M:N, entre “MEDICO” e “PACIENTE”.



Representação do relacionamento através da nova entidade “CONSULTA”.



## Regras ou Soluções possíveis para determinar a chave primária em uma Agregação

**1ª Solução:** Podemos criar uma chave primária composta pelas chaves estrangeiras. Caso garantam unicidade das ocorrências. Obtendo assim um relacionamento identificado.

**2ª Solução:** Podemos criar um novo atributo para ser a chave primária desta entidade. As chaves estrangeiras não são utilizadas como identificadores. O novo atributo será utilizado para garantir a unicidade das ocorrências.

**3ª Solução:** Podemos criar uma chave primária composta por uma das chaves estrangeiras, mais um atributo da própria entidade. Desde que juntos garantam a unicidade das ocorrências. Teríamos um relacionamento identificado de um dos lados. Considerando que a nova entidade, é uma entidade FRACA, pois depende das outras entidades para existir, utilizamos então a mesma regra da chave primária da entidade FRACA.



# Exemplificando Entidade Associativa - Agregação

No exemplo ilustrado anteriormente, observamos que aplicamos a 2ª. Solução para determinar a chave primária da nova entidade “CONSULTA”.

Vamos visualizar em termos de ocorrências para um melhor entendimento da solução proposta.



Entidade: MEDICO			
cd_medico	nm_medico	nr_crm	ds_especialidade
1234	JOÃO	32.004	CLÍNICO GERAL
1235	CARLOS	66.008	OFTALMOLOGIA
1236	MARIA	72.345	PEDIATRIA
1237	ROSA	11.234	ORTOPEDIA

Entidade: PACIENTE					
cd_paciente	nm_paciente	ds_sexo	dt_nascimento	ds_convenio	nr_carteira_convenio
20151001	JOSÉ MARIA	M	03/07/1976	AMIL	100-00019878
20151002	ROSA MARIA	F	12/12/1960	ALLIANZ	110-09876453
20151003	MARIA DE LOURDES	F	17/09/1985	UNIMED	120-98764433
20151004	MANOEL PORTUGUÊS	M	23/11/1970	PORTO SEGURO	130-00876521
20151005	MARIA APARECIDA	F	27/12/1980	SUL AMERICA	140-00076542
20151006	ROBERTO DE SOUZA	M	19/08/1995	AMIL	100-00980765

Entidade: CONSULTA				
cd_consulta	cd_medico	cd_paciente	dt_hr_consulta	nr_consultorio
201500001	1234	20151001	10/02/2015 10:00	1
201500002	1235	20151002	17/03/2015 09:00	2
201500003	1235	20151001	17/03/2015 10:00	3
201500004	1236	20151003	23/04/2015 10:00	5
201500005	1234	20151003	10/02/2015 10:45	
201500006	1236	20151001	23/04/2015 11:00	1
201500007	1235	20151004	17/03/2015 11:00	
201500008	1235	20151003	17/03/2015 13:00	4
201500009	1235	20151002	17/03/2015 18:00	2
201500010	1234	20151004	10/02/2015 11:15	
201500011	1236	20151006	23/04/2015 11:00	5

Fazendo a análise observem que utilizamos a **2ª solução**, criamos um novo atributo “CD\_CONSULTA” para a entidade “CONSULTA” para identificação única (chave primária) das ocorrências.

Se utilizássemos a 1ª solução, teríamos repetição do par médico-paciente. Pois existe a possibilidade de um médico atender um paciente mais de uma vez ao longo do tempo e até no mesmo dia.

Analisando a 3ª. Solução, seria a utilização de uma das chaves estrangeiras mais um atributo da própria entidade, observamos que como o paciente e o médico podem se repetir, e os demais atributos, número da sala ou consultório são opcionais e a data não é um bom atributo para compor a chave, observamos que a 3ª. Solução não é aplicável neste caso.

Observando o exemplo, vale ressaltar que para cada situação deve-se avaliar a melhor solução a ser aplicada de acordo com o contexto de negócio e as necessidades de armazenamento.



Suponha a situação, em que durante uma consulta médica (exemplo demonstrado anteriormente), um Médico tenha prescrito medicamentos para um Paciente.

Em cada consulta o Médico pode prescrever medicamentos diferentes para o Paciente, ou até mesmo não prescrever nenhum medicamento.

Mas para existir uma prescrição (receita que o Médico nos fornece durante uma consulta), o paciente deve obrigatoriamente passar por uma consulta médica.

Portanto, a prescrição seria um relacionamento entre a entidade Medicamento e o relacionamento consulta.

Cada consulta pode prescrever nenhum, um ou vários medicamentos.

Cada medicamento pode ser prescrito em nenhuma, uma ou várias consultas.

## EXERCÍCIO PRÁTICO



Oracle  
SQL Developer  
Data Modeler



FIAP



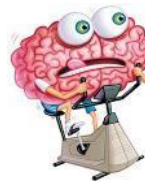
Vamos aproveitar e incrementar o projeto RHSTU. A partir do gabarito disponibilizado pelo professor, inclua as seguintes regras:

**RN01:** Na RHSTU, um paciente pode ser atendido por vários médicos durante o seu ciclo de vida e o médico atende vários pacientes durante seu tempo de trabalho na RHSTU. O atendimento do paciente sempre será feito por meio de uma consulta médica, e informações como data da consulta e número da sala são informações importantes em nosso projeto.

**RN02:** Como a RHSTU é uma empresa privada, a forma de pagamento a ser feita pelo paciente pode ser feita por meio de seu plano de saúde, porém existem situações em que o paciente não tem plano, então o pagamento é feito em dinheiro, pix, cartão de crédito ou cartão de débito. É muito importante identificar a forma de pagamento para cada consulta. Caso o pagamento seja feito com o plano de saúde, é necessário identificar os dados desse plano, bem como o número da carteira do plano de saúde do paciente.

**RN03:** Essa consulta teve origem em uma Unidade Hospitalar da RHSTU. Então é necessário saber qual é essa unidade e seu endereço completo.

# EXERCÍCIO PRÁTICO



Oracle  
SQL Developer  
Data Modeler



FIA/P

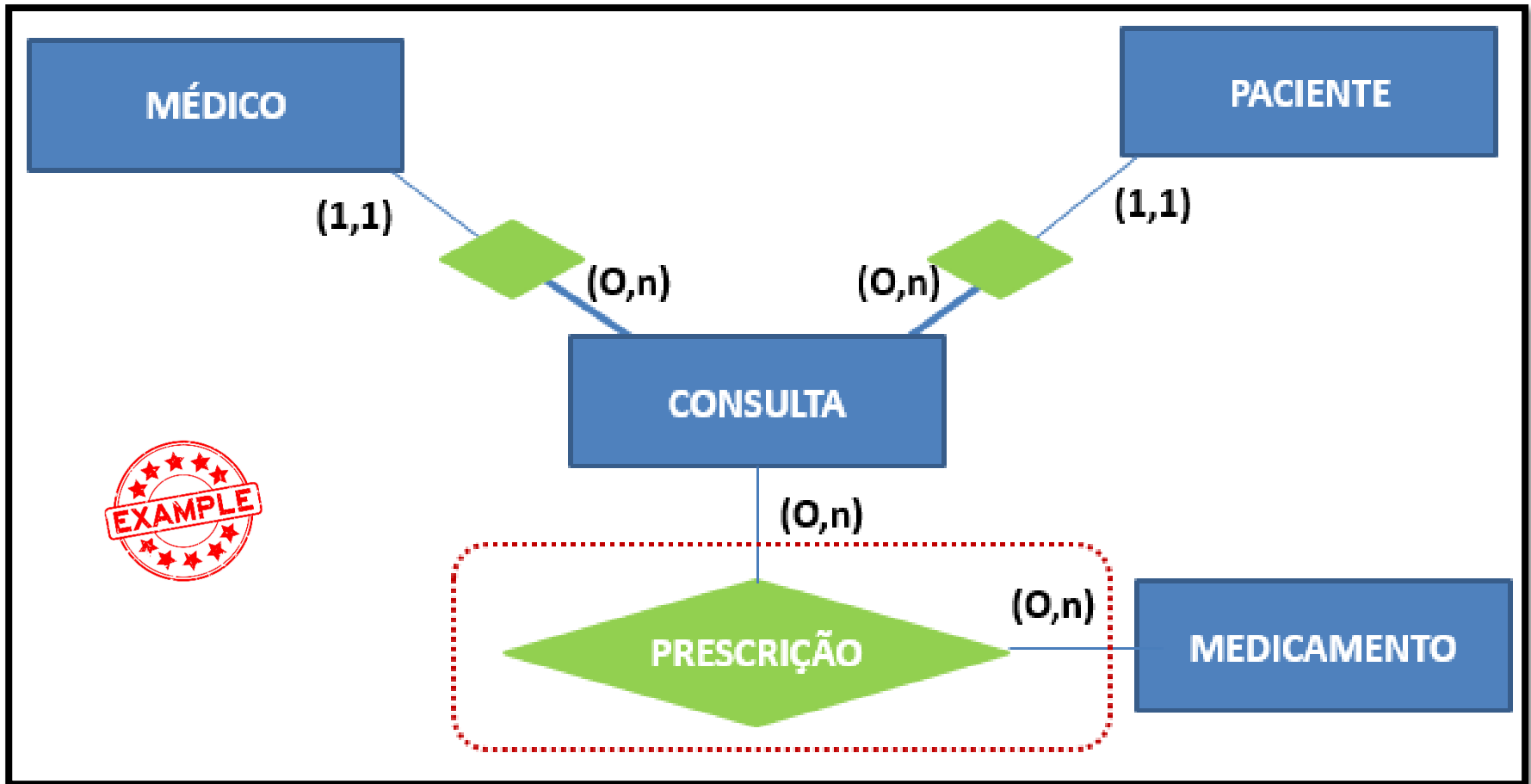


Trabalhe em dupla e a partir do modelo de dados lógico disponibilizado pelo professor, realize esses ajustes e crie uma versão nova (v2) de seu projeto de Banco de Dados Lógico.

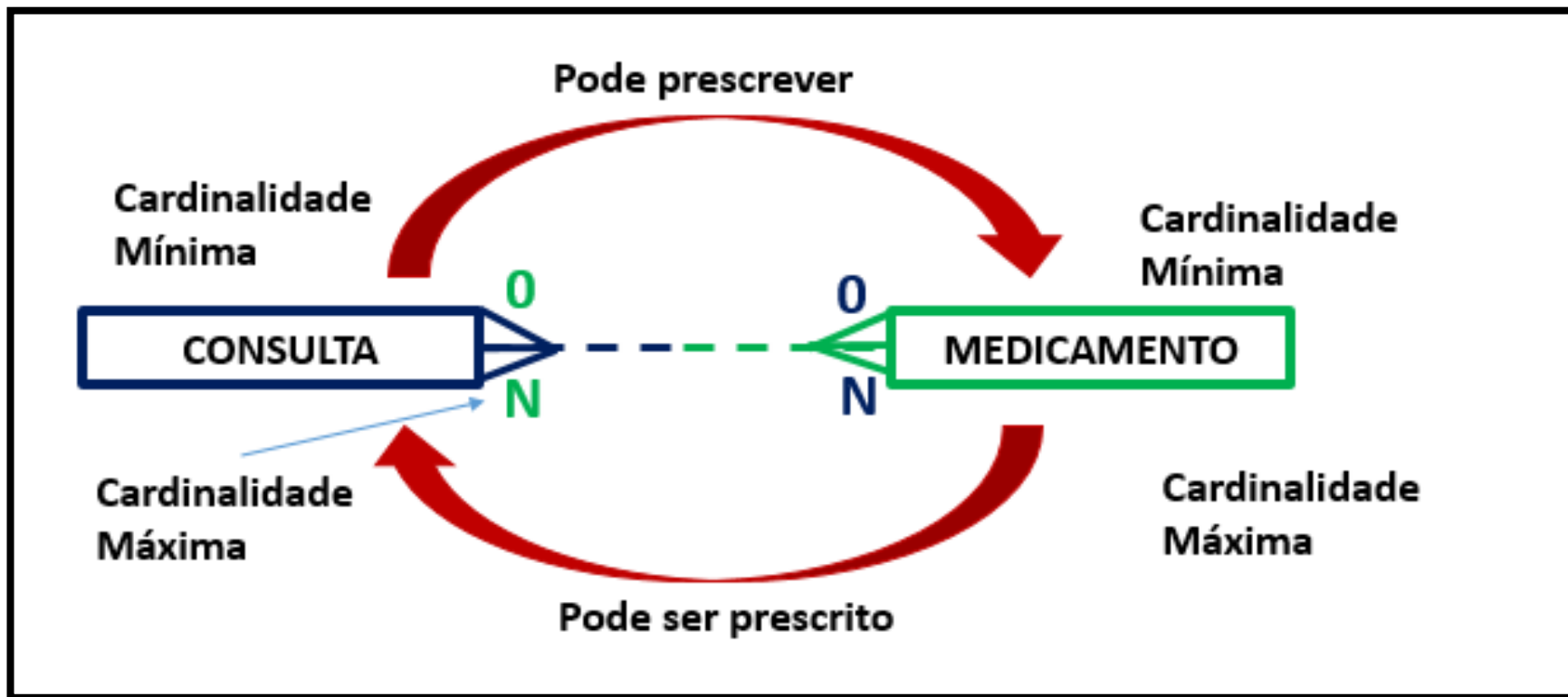
## Apresente para o seu professor em laboratório:

- Arquivo componentes.txt
- Modelo de dados lógico completo no Oracle Data Modeler
- Arquivo \*.pdf do modelo de dados lógico

Podemos representar a associação entre consulta e medicamento da seguinte maneira:



Exemplificando de uma forma um pouco mais detalhada:



Observe que temos um relacionamento com cardinalidade M:N.



Veja que quando pensamos em uma prescrição de medicamentos durante uma consulta, conseguimos encontrar atributos que caracterizam o relacionamento M:N. Por exemplo, uma prescrição contém posologia, via, orientações de uso, além é claro do medicamento, nome médico, crm e data da prescrição (encontrado na descrição da entidade Medicamento, Médico e Consulta, respectivamente, não é necessário repetir esses atributos).

Exemplificando a estrutura mínima da prescrição, temos:

- Posologia (dosagem certa de um medicamento) = 1 comprimido de 50 mg.
- Via = Oral.
- Orientações de uso = a cada 8 horas após as refeições.

Analisando este exemplo observamos que posologia, via e orientações de uso são atributos que só existem, quando uma prescrição existir, portanto não caracteriza Consulta e nem caracteriza Medicamento. Podemos dizer que são atributos do relacionamento. Se Prescrição não existir, os atributos citados não se aplicam as demais relações existentes.

A partir do momento que encontramos os atributos que caracterizam o relacionamento, observamos mais nitidamente a necessidade de uma nova entidade.

Neste exemplo, o nome mais adequado para a nova entidade é “PRESCRIÇÃO”. Mas porque não “RECEITA”, receita representa uma espécie de relatório que contém todos os medicamentos prescritos por um médico durante uma consulta. Pensando em armazenamento, precisamos guardar cada um dos medicamentos prescritos em uma determinada consulta.

# Evoluindo o Exemplo de Agregação

Exemplo de uma receita médica (prescrição de medicamentos) :



The image shows a handwritten medical prescription on a piece of paper. The text is written in cursive. At the top, the patient's name is written and then blacked out. Below this, the word 'Rx' is written, followed by 'Uso oral'. There are two numbered items: 1) Amoxicilina (250mg/ml) with instructions 'Tomar 10 ml de 8 em 8 horas por 7 dias'. 2) Ibuprofeno (100mg/ml) with instructions 'Tomar 30 gotas'. At the bottom, there is a signature and a stamp that reads 'Dr. João Sá-Te', 'C.R.O. 12345', and '05/05/19'. Blue arrows point from labels on the right to specific parts of the prescription: 'Dados do Paciente' points to the blacked-out name; 'VIA' points to 'Uso oral'; 'POSOLOGIA' points to 'Amoxicilina (250mg/ml)'; 'ORIENTAÇÕES DE USO' points to 'Tomar 10 ml de 8 em 8 horas por 7 dias'; and 'Dados do Médico' points to the signature.

Dados do Paciente

VIA

POSOLOGIA

ORIENTAÇÕES DE USO

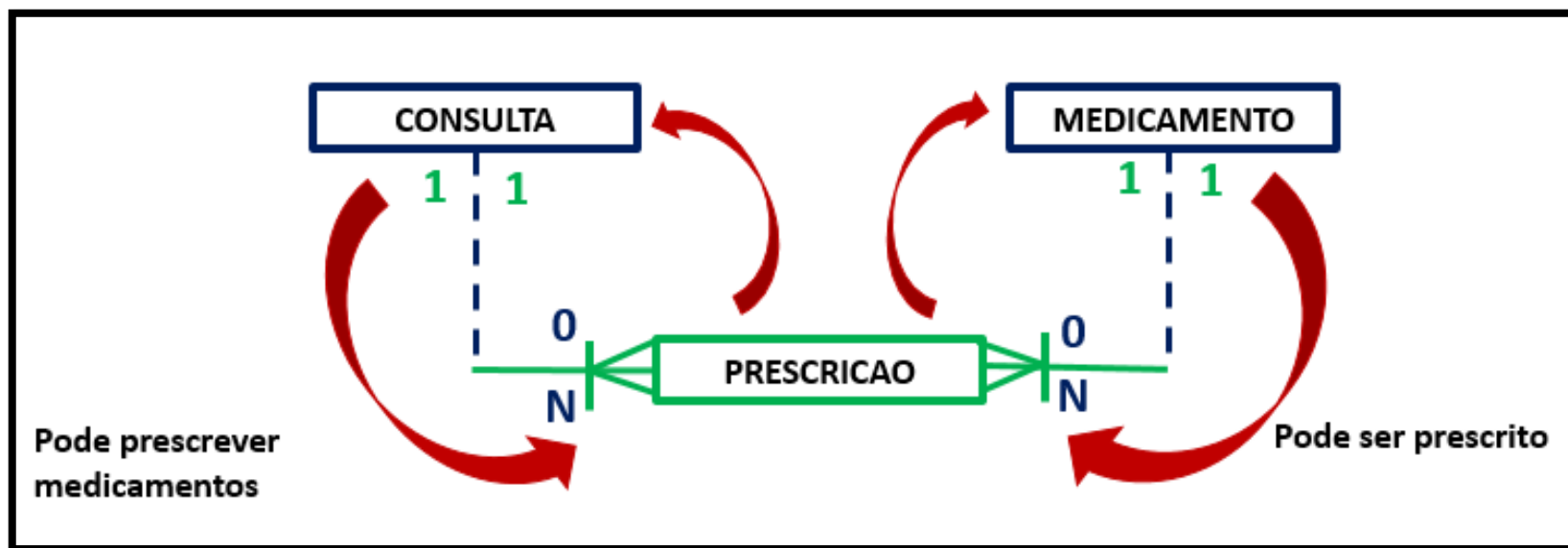
Dados do Médico

Cada medicamento prescrito, contém informações sobre via, posologia e orientações de uso.





Quando transformamos o relacionamento em uma nova entidade, passamos a ter dois relacionamentos 1:N, um entre “CONSULTA” e “PRESCRICAO” e outro entre “MEDICAMENTO” e “PRESCRICAO”.



Características da entidade **PRESCRICAO** →

Posologia

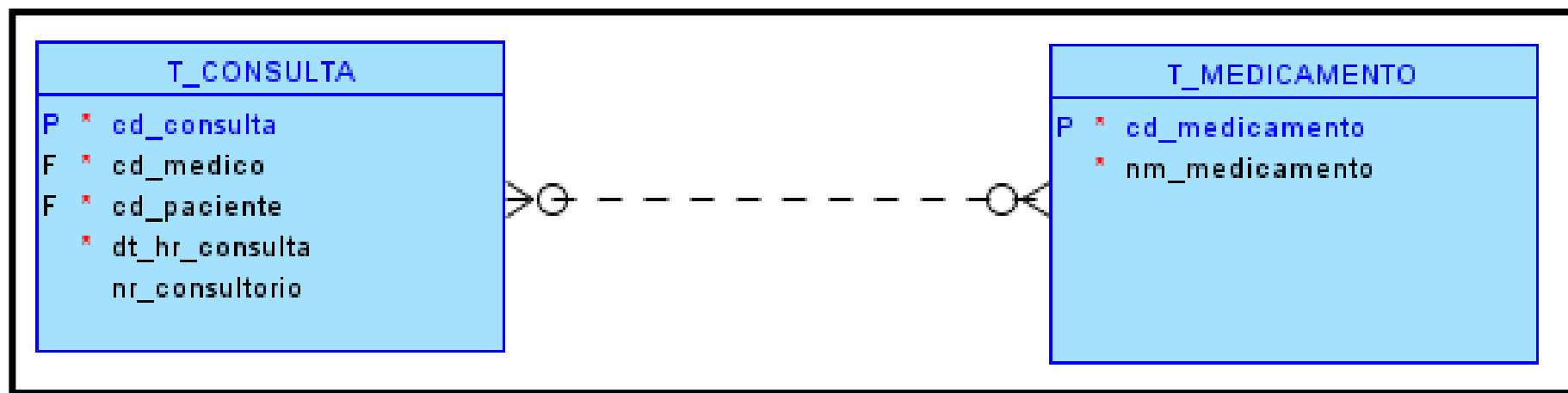
Via

Orientações de uso



Utilizando a representação gráfica através da ferramenta SQL Developer Data Modeler, a partir da notação da Engenharia da Informação:

Representação do relacionamento M:N, entre “CONSULTA” e “MEDICAMENTO”.

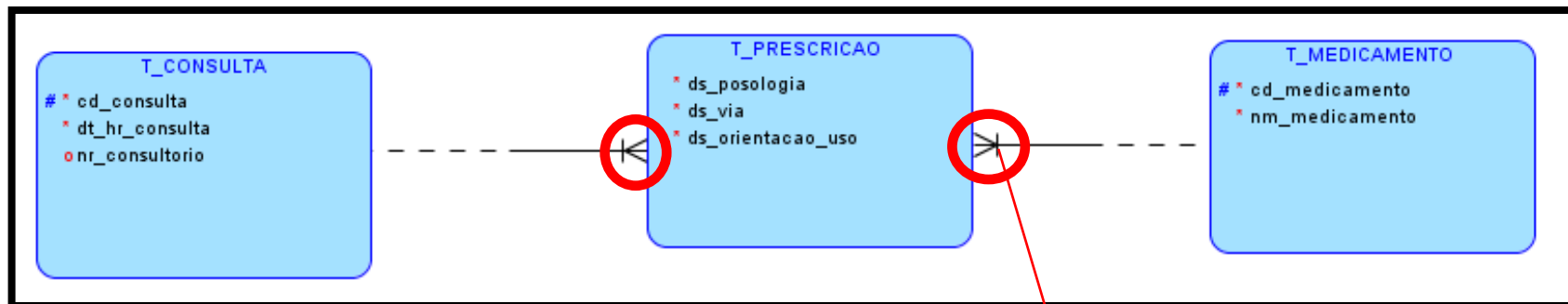




# Evoluindo o Exemplo de Agregação

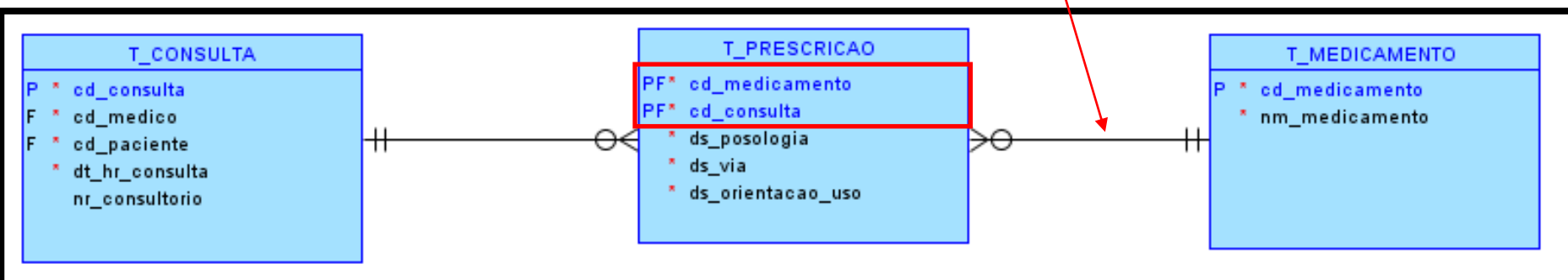
Representação do relacionamento através da nova entidade “PRESCRICAO”.

## SQL Data Modeler – Notação de Barker



O sinal | observado no diagrama acima, representa o relacionamento identificado na notação de Barker, equivalente a linha sólida, representado na notação da Engenharia da Informação.

## SQL Data Modeler – Notação da Engenharia da Informação



# Evoluindo o Exemplo de Agregação

No exemplo ilustrado acima, observamos que aplicamos a 1ª. Solução para determinar a chave primária da nova entidade “PRESCRICAO”.

Vamos visualizar em termos de ocorrências para um melhor entendimento da solução proposta.

Entidade: CONSULTA				
cd_consulta	cd_medico	cd_paciente	dt_hr_consulta	nr_consultorio
201500001	1234	20151001	10/02/2015 10:00	1
201500002	1235	20151002	17/03/2015 09:00	2
201500003	1235	20151001	17/03/2015 10:00	3
201500004	1236	20151003	23/04/2015 10:00	5
201500005	1234	20151003	10/02/2015 10:45	
201500006	1236	20151001	23/04/2015 11:00	1
201500007	1235	20151004	17/03/2015 11:00	
201500008	1235	20151003	17/03/2015 13:00	4
201500009	1235	20151002	17/03/2015 18:00	2
201500010	1234	20151004	10/02/2015 11:15	
201500011	1236	20151006	23/04/2015 11:00	5

Entidade: MEDICAMENTO	
cd_medicamento	nm_medicamento
100100	Novalgina
100101	Rinossoro
100102	Advil
100103	Tilenol
100104	Omeprazol
100105	Dorflex

Entidade: PRESCRICAO				
cd_consulta	cd_medicamento	ds_posologia	ds_via	ds_orientacao_uso
201500001	100100	40 gotas	oral	A cada 8h
201500001	100101	25 gotas	nasal	A cada 8h
201500001	100102	1 comprimido	oral	A cada 6h
201500003	100102	1 comprimido	oral	A cada 8h
201500003	100103	1 comprimido	oral	A cada 6h
201500007	100102	1 comprimido	oral	A cada 8h
201500007	100104	1 comprimido	oral	A cada 8h



Chave primária composta pelos atributos “cd\_consulta” e “cd\_medicamento”. Os dois atributos garantem a identificação única de cada ocorrência da entidade.



Fazendo a análise observem que utilizamos a **1ª solução**, utilizamos as chaves estrangeiras para compor a chave primária da entidade “PRESCRICAO”. Podemos observar que um medicamento não é prescrito mais de uma vez para uma mesma consulta. Identificando de forma única cada uma das ocorrências.

Temos então uma chave primária composta pelas chaves estrangeiras na entidade “PRESCRICAO”. Quando isto ocorre temos um **relacionamento identificado**.

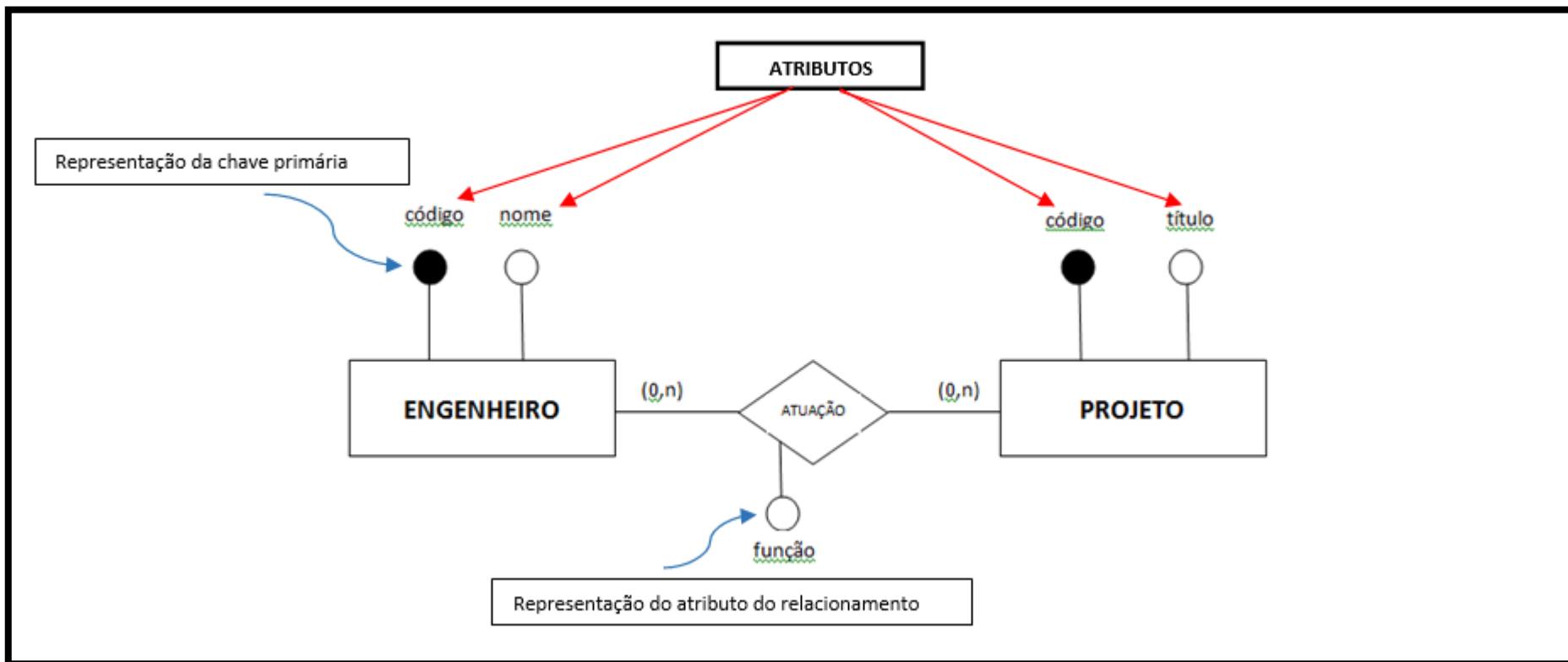
Relacionamento identificado recebe este nome, em função das chaves estrangeiras serem utilizadas como chave primária na nova entidade, ou seja, as chaves estrangeiras são identificadores na entidade “PRESCRICAO”.

Vale ressaltar que quando escolhemos a 1ª. Solução (chave composta), é prudente uma análise da solução proposta como um todo, pois qualquer outra entidade que se relacionar com Prescrição, terá como chave estrangeira, uma chave composta por dois atributos. O cuidado é em relação aos demais relacionamentos que podem existir, tornando as chaves primárias ou estrangeiras longas demais. Nestas situações vale a utilização da 2ª. Solução que é a criação de um novo atributo como chave primária.

Como ponto de atenção, podemos dizer que não é boa prática criar chaves primárias compostas por muitos atributos, a chave fica longa, ocupa espaço em disco e o SGBD tem um custo maior para ordenação e busca dos dados através da chave.

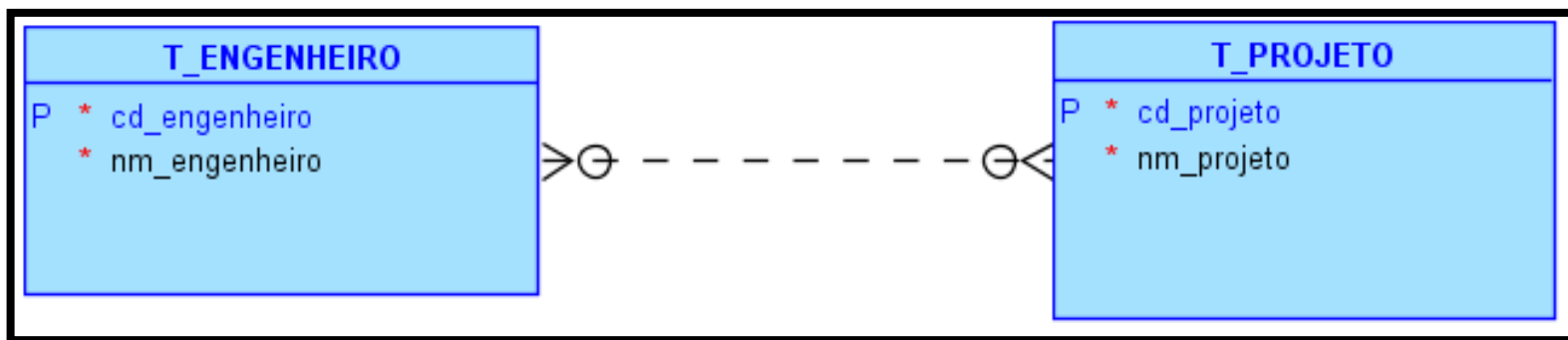
Vamos abordar um outro exemplo, onde teremos atributos do relacionamento. No exemplo proposto, podemos dizer que um engenheiro pode exercer funções diferentes em diferentes projetos em que atuar, portanto na representação do relacionamento “ATUAÇÃO”, podemos utilizar um atributo que determinará a função que o engenheiro exerce dentro de um projeto.

O atributo função não pode ser considerado característica do “ENGENHEIRO”, uma vez que ele pode atuar em diversos projetos. E também não pode ser um atributo do “PROJETO”, uma vez que em um projeto podem atuar vários engenheiros com funções diferentes.





Representação do relacionamento M:N, entre “ENGENHEIRO” e “PROJETO”. Utilizando a notação da Engenharia da Informação.



O atributo função (**atributo do relacionamento**) será característica de uma nova entidade, que podemos chamar “ENGENHEIRO\_PROJETO” ou “ATUACAO” ou outro nome que for pertinente de acordo com a necessidade de negócio. Podemos inclusive, inserir outros atributos que poderão facilitar análises futuras, como início e término da função no projeto. Que também são atributos que caracterizam a associação M:N entre ENGENHEIRO e PROJETO. Poderíamos desta forma ter:

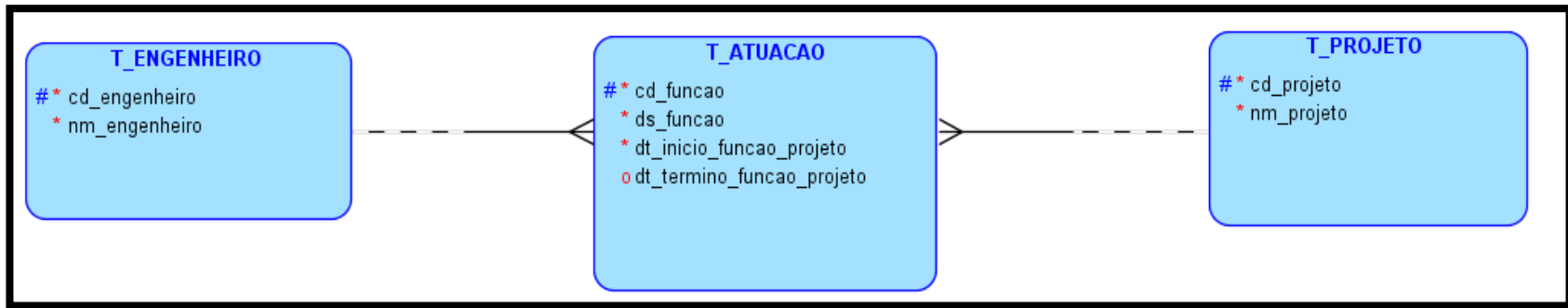
Representação do relacionamento através da nova entidade “ATUACAO”.



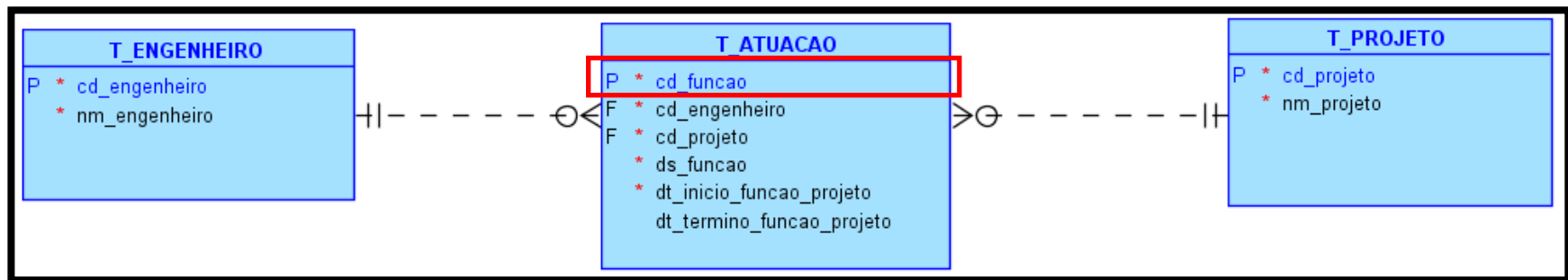
# Um pouco mais de atributos do relacionamento



## SQL Data Modeler – Notação de Barker



## SQL Data Modeler – Notação da Engenharia da Informação



Pensando na chave primária da nova entidade “ATUACAO”, **podemos aplicar a 2ª Solução**, pois um mesmo engenheiro poderia entrar e sair de um projeto várias vezes, de acordo com a necessidade, e até executar diferentes funções dentro de um mesmo projeto.

Foi então criado o novo atributo “CD\_FUNCAO”, conforme destacado na imagem acima.

## EXERCÍCIO PRÁTICO



Oracle  
SQL Developer  
Data Modeler



FIAP



Vamos aproveitar e incrementar o projeto RHSTU. A partir do gabarito disponibilizado pelo professor, inclua as regras abaixo e crie uma nova versão (v3):

**RN01:** Na consulta feita pelo paciente, o médico irá prescrever uma série de medicamentos, se o medicamento deve ser aplicado via oral, injetável ou por outro meio, a posologia com sua orientações de uso.

**RN02:** É muito importante sabermos informações adicionais do médico que prescreveu o medicamento: além da data de contratação na RHSTU, número do CRM e a data de cadastro do médico no CRM são informações obrigatórias e que serão utilizadas posteriormente.

# EXERCÍCIO PRÁTICO



Oracle  
SQL Developer  
Data Modeler



FIA/P



Trabalhe em dupla e a partir do modelo de dados lógico disponibilizado pelo professor, realize esses ajustes e crie uma versão nova (v3) de seu projeto de Banco de Dados Lógico.

## Apresente para o seu professor em laboratório:

- Arquivo componentes.txt
- Modelo de dados lógico completo no Oracle Data Modeler
- Arquivo \*.pdf do modelo de dados lógico



- HEUSER, C.A. Projeto de Banco de Dados. Série Livros Didáticos, V. 4. Bookman, 2009. Capítulo 2 e 3 – p. 34 a 117
- MACHADO, Felipe Nery R. Banco de Dados - Projeto e Implementação. Érica, 2004. Capítulo 4 e 5 – p. 67 a 124
- SILBERSCHATZ, A; KORTH, H. F.; SUDARSHAN, S. Sistema de Banco de Dados. Campus, 2006. Capítulo 6 – p. 133 a 174
- ELMASRI, R.; NAVATHE, S.B. Sistemas de Banco de Dados: Fundamentos e Aplicações. Pearson, 2005. Capítulo 4 – p. 60 a 86

**Copyright © 2022 Profa. Rita de Cássia Rodrigues**

**Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).**