

DIAGRAMA DE CLASSES DE PROJETO

SSC-526 – Análise e Projeto Orientados a Objeto
Profa. Dra. Elisa Yumi Nakagawa
2º semestre de 2017

Diagrama de Classes de Projeto

- Diagrama de Classes de Projeto apresenta especificações para classes de software e interfaces (ex: interfaces Java) de uma aplicação
- Informação típica:
 - classes, associações e atributos
 - interfaces, com operações e constantes
 - métodos
 - tipos dos atributos
 - navegabilidade
 - dependências

Definição (cont.)

- Modelo Conceitual \Rightarrow abstrações de conceitos ou objetos do mundo real
 - conceitos são também chamados de classes conceituais
- Diagrama de Classes de Projeto \Rightarrow definição de classes como elementos de software
 - classes de software

Definição (cont.)

- Diagrama de classes pode ser construído à medida que a fase de projeto avança, a partir dos diagramas de comunicação
- Cada classe que aparece no diagrama de comunicação automaticamente é incluída no diagrama de classes de projeto
- Atributos são inicialmente os que estão no modelo conceitual

Classes que aparecem nos diagramas de comunicação

Leitor
nome
tipo

Emprestimo
data_do_emprestimo
situacao : char

LinhaDoEmprestimo
data_prevista_
devolução

CopiaDoLivro
nro_sequencial
situacao : char
liberadoParaEmprestimo :

Associações e Navegabilidade

- Associações e navegabilidade entre classes são indicadas pelos diagramas de comunicação
 - Navegabilidade indica possibilidade de navegação unidirecional por meio de uma associação entre classes
 - geralmente implica visibilidade por atributos
- Multiplicidade e os nomes das associações podem ser retirados do Modelo Conceitual
- Notação: seta contínua

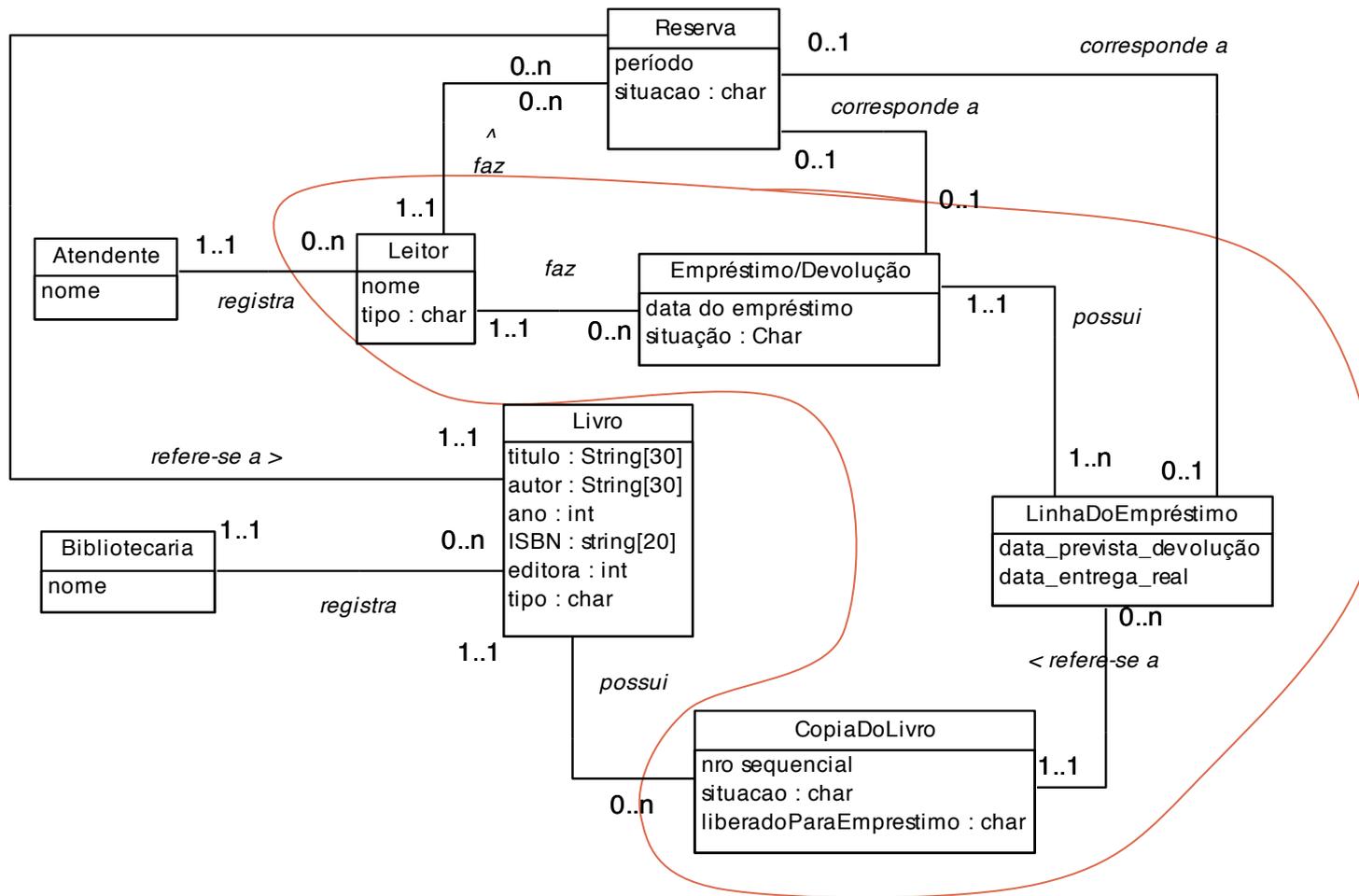
Associações e Navegabilidade

- Indícios de associação e com presença de navegabilidade:
 - A envia mensagem para B
 - A cria B
 - A precisa manter uma conexão com B

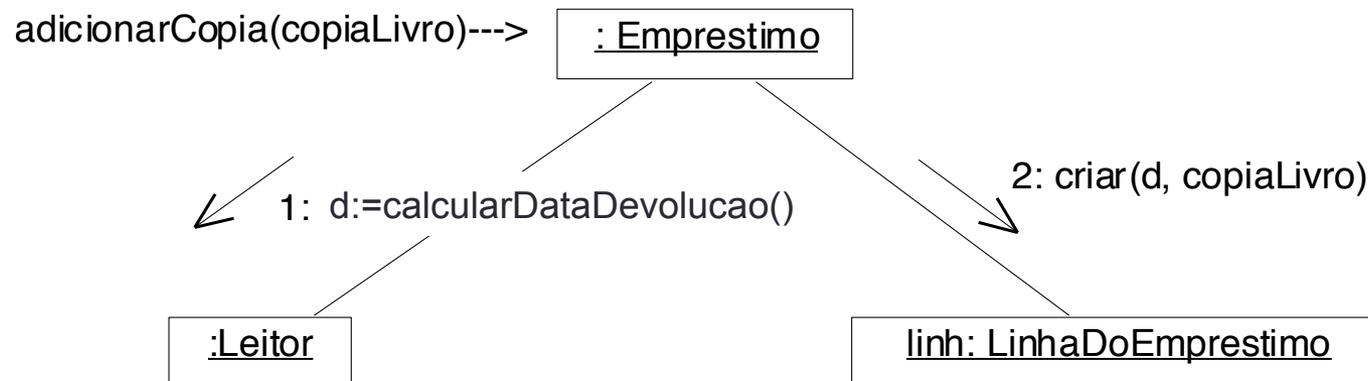
Como determinar a navegabilidade?

- Verificar o envio de mensagens de objetos que possuem visibilidade por atributo
- Desenhar a seta no sentido da classe que envia a mensagem para a classe que recebe a mensagem

Modelo Conceitual



Navegabilidade

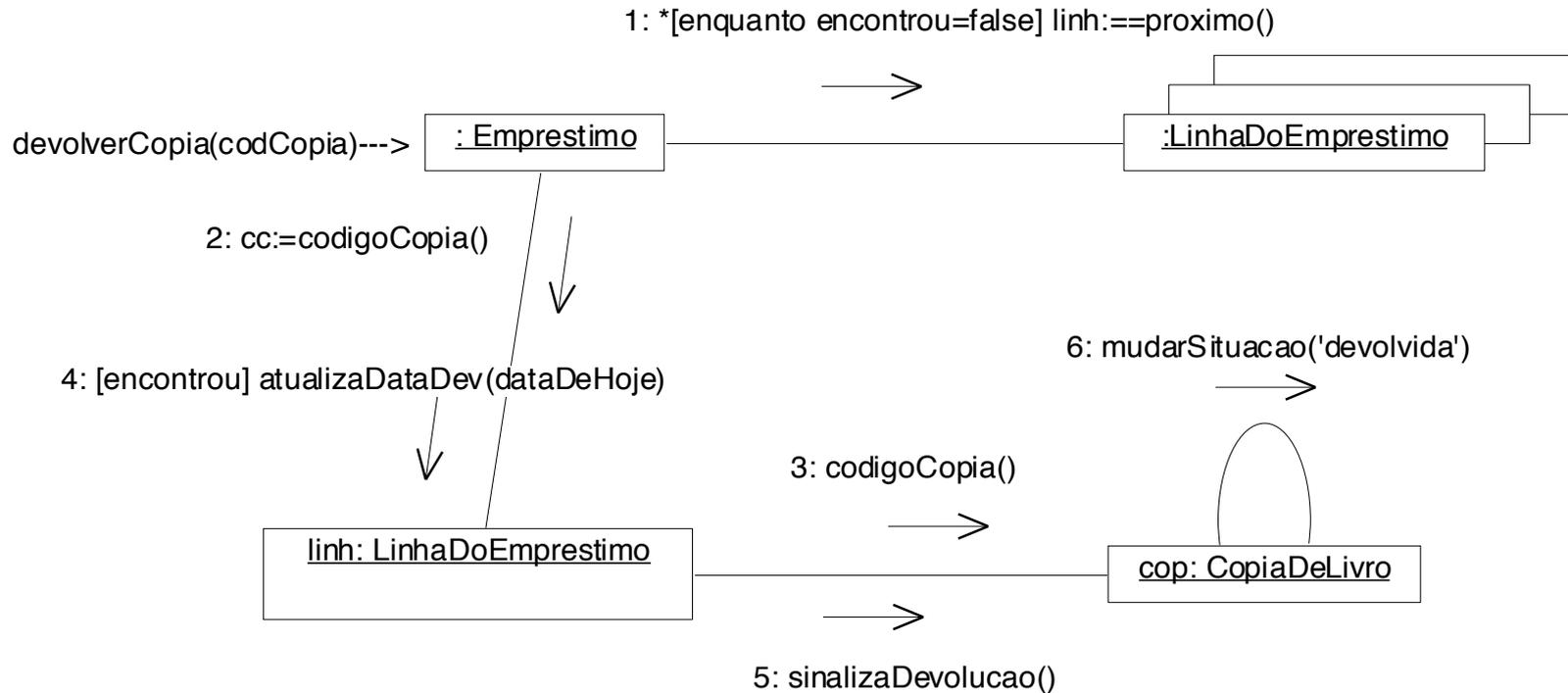


Este diagrama de comunicação implica nas navegabilidades:

Emprestimo → Leitor

Emprestimo → LinhaDoEmprestimo

Navegabilidade

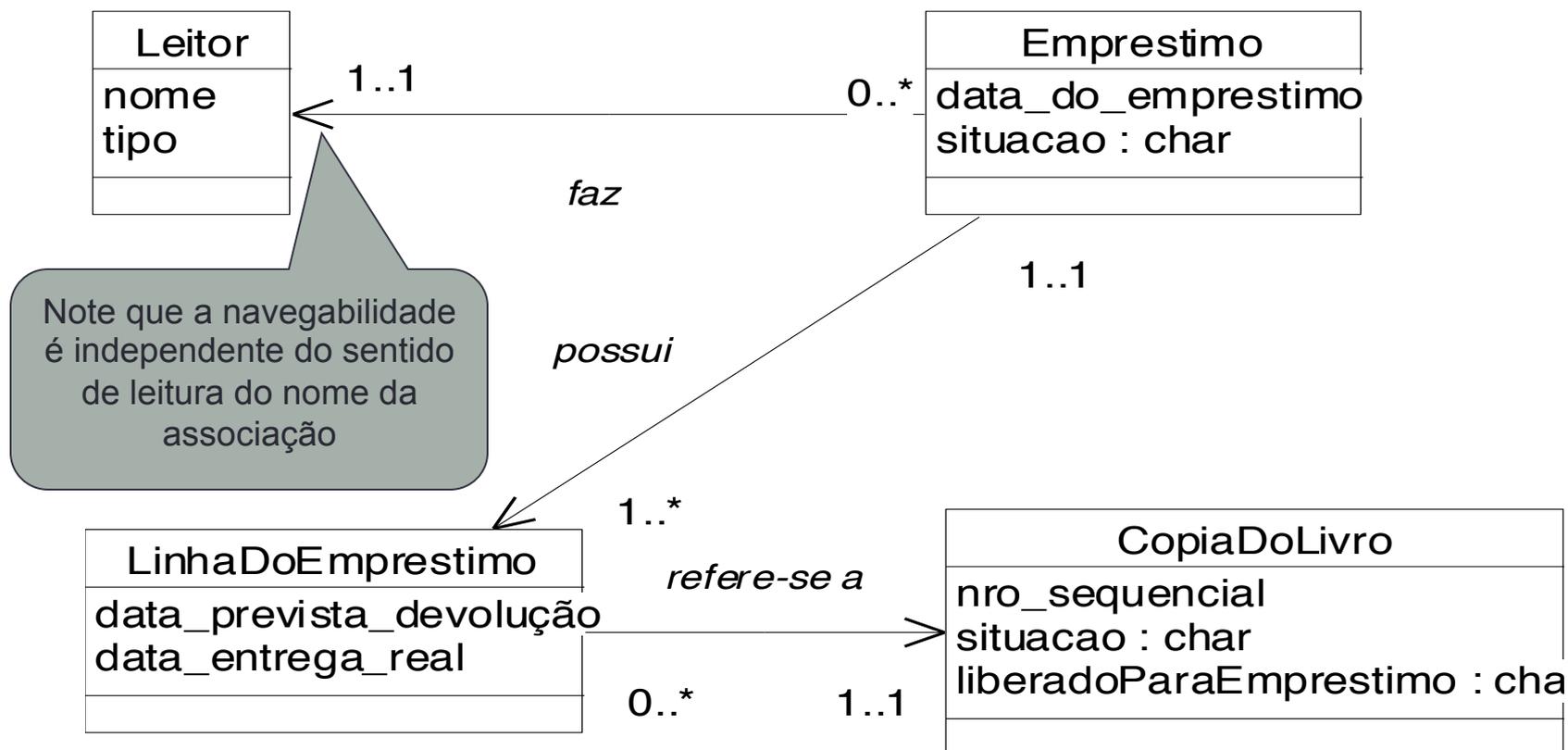


Este diagrama de comunicação implica nas navegabilidades:

Emprestimo → LinhaDoEmprestimo

LinhaDoEmprestimo → CopiaDeLivro

Diagrama de Classes com navegabilidade



(com base apenas nos dois diagramas de comunicação mostrados)

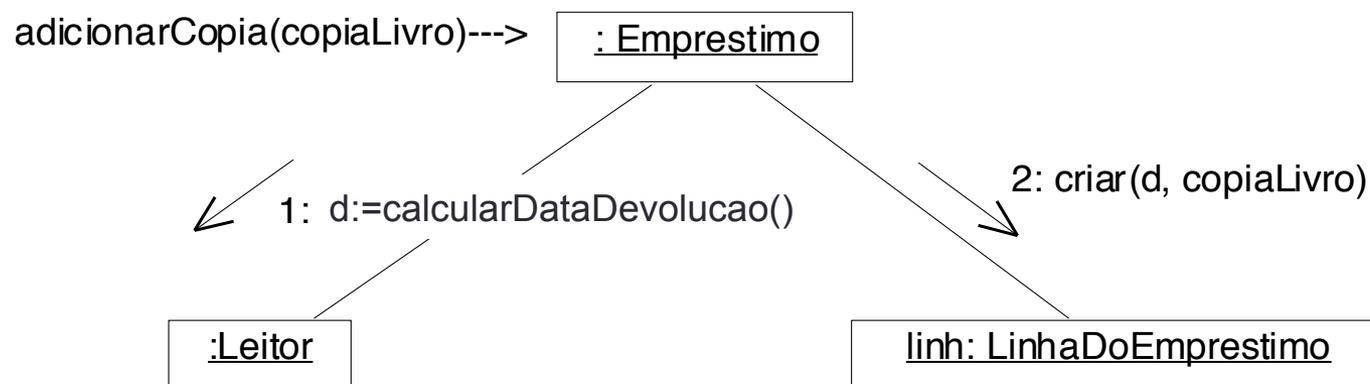
Como incluir os métodos nas classes?

- Operações são incluídas nas classes controladoras
- Métodos são incluídos nas classes que recebem a mensagem
- Linguagens de programação distintas podem ter sintaxes distintas para métodos
 - recomendável: usar sintaxe básica UML
 - Exemplo: *nomeMetodo(Par₁, Par₂, ... Par_n)*

Como incluir os métodos nas classes?

- Não incluir:
 - Métodos enviados às coleções (pois dependem da implementação)
 - Método criar (linguagem OO provê o criador)
 - Métodos de acesso a atributos, pois assume-se que cada atributo tem necessariamente esses métodos
 - Exemplo: `setNome()`, `getNome ()`,

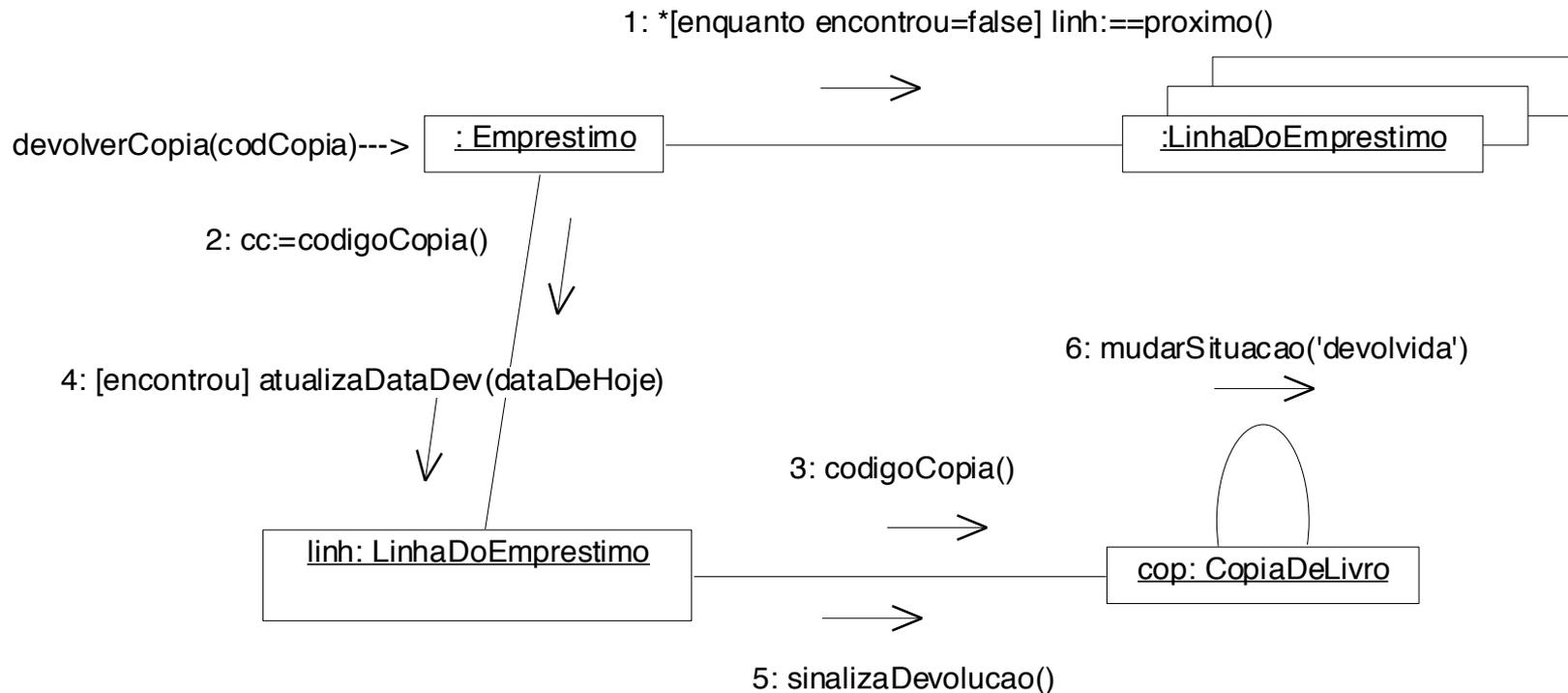
Inclusão de métodos



Este diagrama de comunicação implica nos seguintes métodos:

Emprestimo → adicionarCopia()

Leitor → calcularDataDevolucao()



Este diagrama de comunicação implica nos seguintes métodos:

Emprestimo → devolverCopia()

LinhaEmprestimo → codigoCopia()

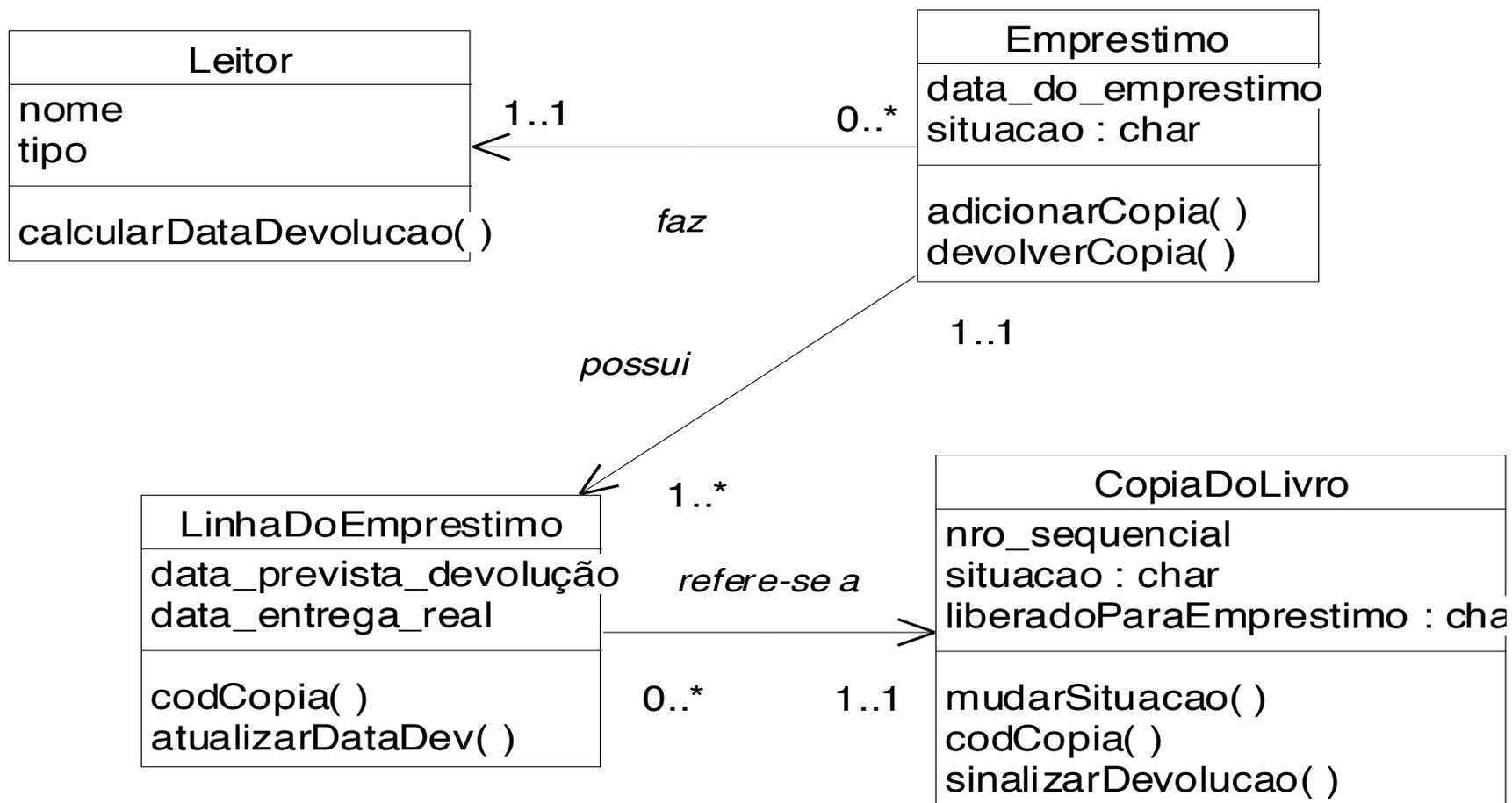
LinhaEmprestimo → atualizaDataDev()

CópiaDeLivro → mudarSituacao()

CópiaDeLivro → codigoCopia()

CópiaDeLivro → sinalizaDevolucao()

Diagrama de Classes de Projeto



(com base apenas nos 2 diagr. de colaboração mostrados)

Atributos

- Pode-se acrescentar tipos de atributos, parâmetros e retornos de métodos, observando os diagrama de comunicação
- Atributos identificados durante o projeto podem ser incluídos
 - se uma ferramenta CASE for utilizada para geração automática de código, os tipos detalhados são necessários
 - se o diagrama for usado exclusivamente por desenvolvedores de software, o excesso de informação pode “poluir” o diagrama e dificultar seu entendimento

Observações

- Embora uma visibilidade por atributo venha a ser implementada posteriormente como um atributo na classe origem, isso não deve ser mostrado no diagrama de classes
- Novas classes podem surgir nos diagramas de comunicação, portanto, deve-se pensar em nomes para elas, bem como nas multiplicidades das associações correspondentes.

Relacionamento de Dependência

- No Diagrama de Classes, o relacionamento de dependência representa a visibilidade entre classes que não é implementada por atributo
 - visibilidade por parâmetro
 - visibilidade local ou global
- Um objeto de uma classe A tem conhecimento (enxerga) um objeto da classe B
- Notação: seta tracejada

Exemplo: Dependência

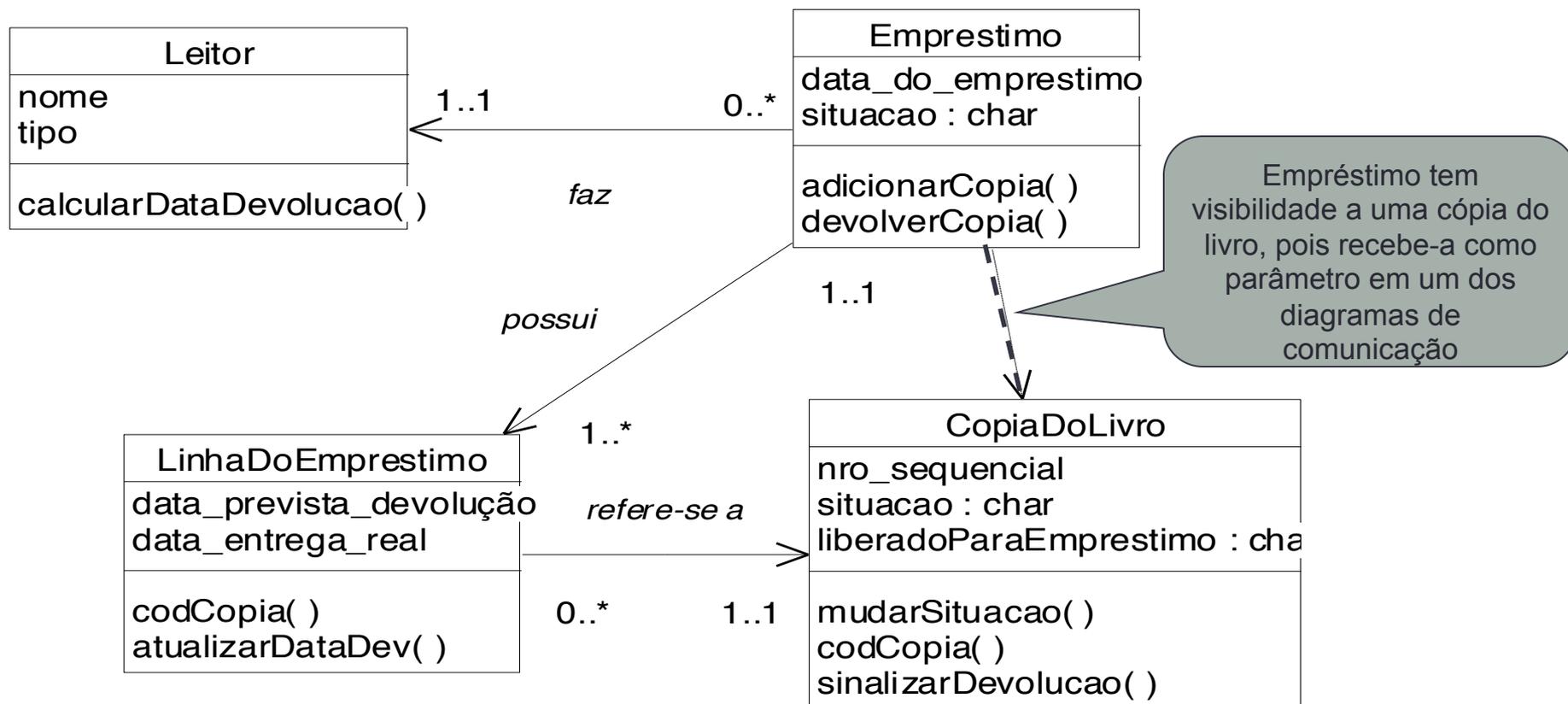


DIAGRAMA DE CLASSES DE PROJETO

SSC-526 – Análise e Projeto Orientados a Objeto
Profa. Dra. Elisa Yumi Nakagawa
2º semestre de 2017