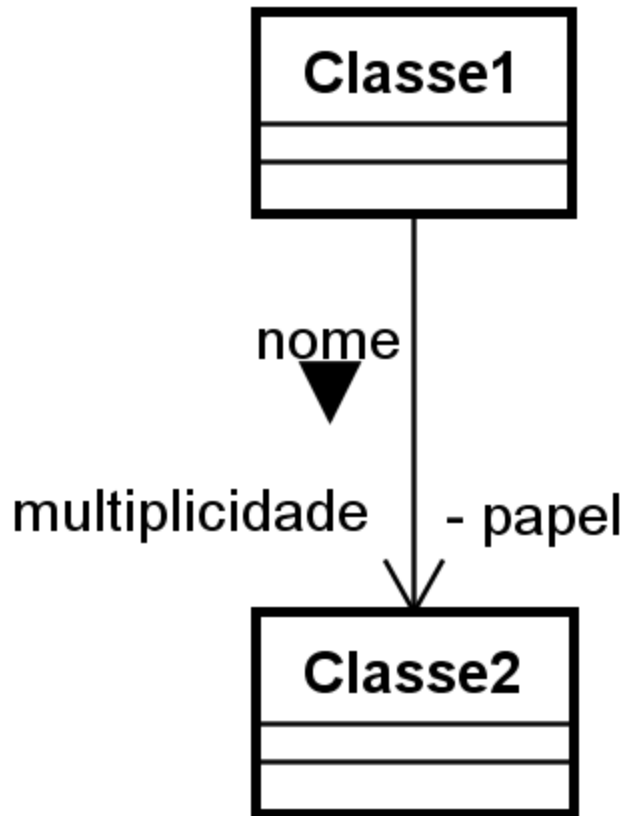


Relacionamento entre objetos

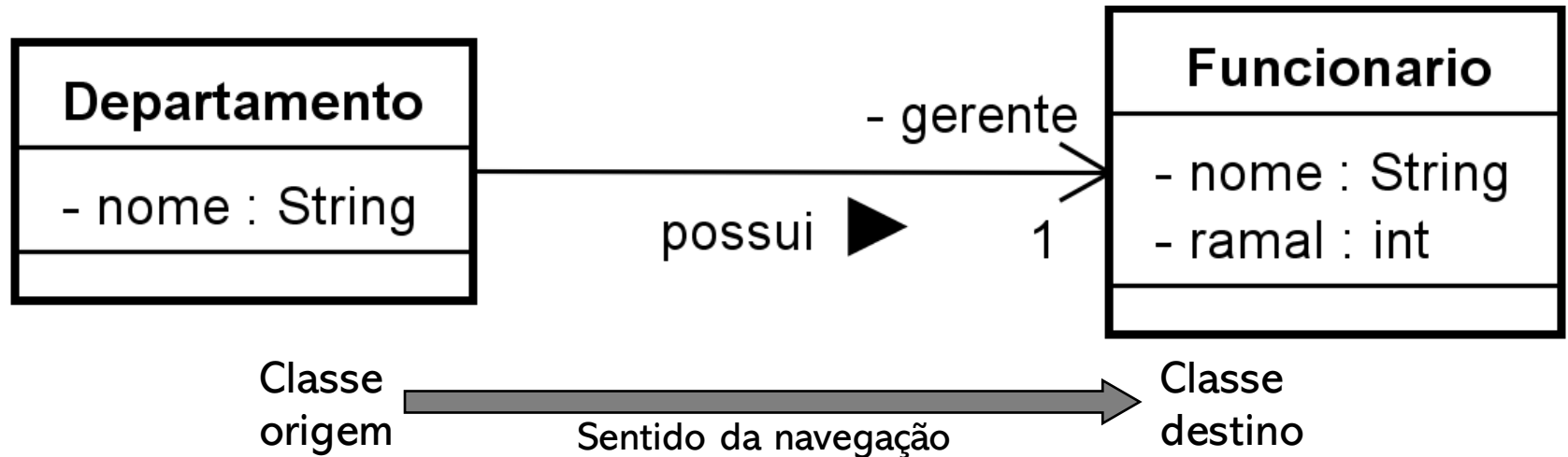
Associações



- Acompanhada de adornos (ou aprimoramentos):
 - Nome
 - Papel
 - Multiplicidade
 - Navegabilidade

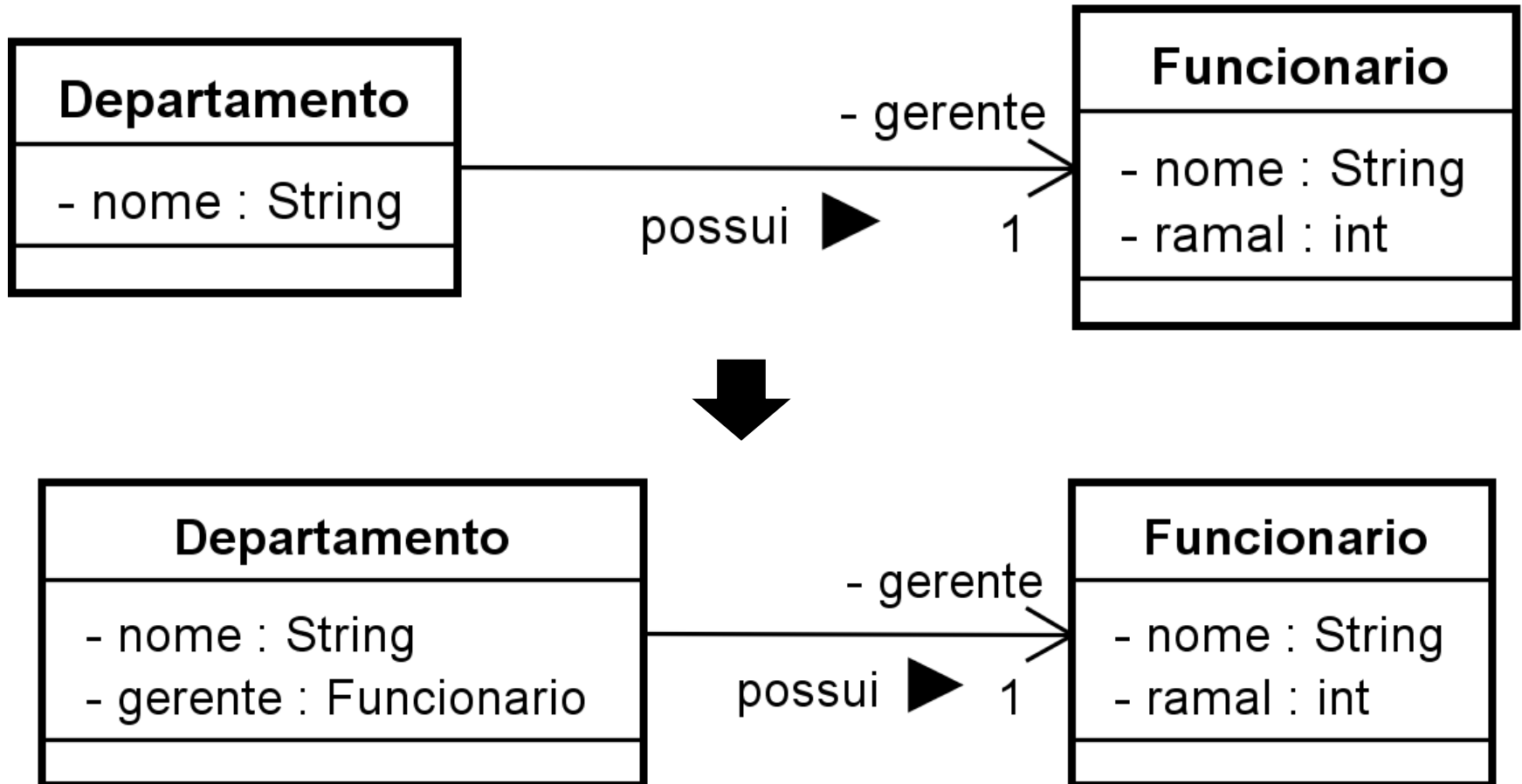
Associações com multiplicidade 0..1 ou 1

Associações com multiplicidade 0..1 ou 1



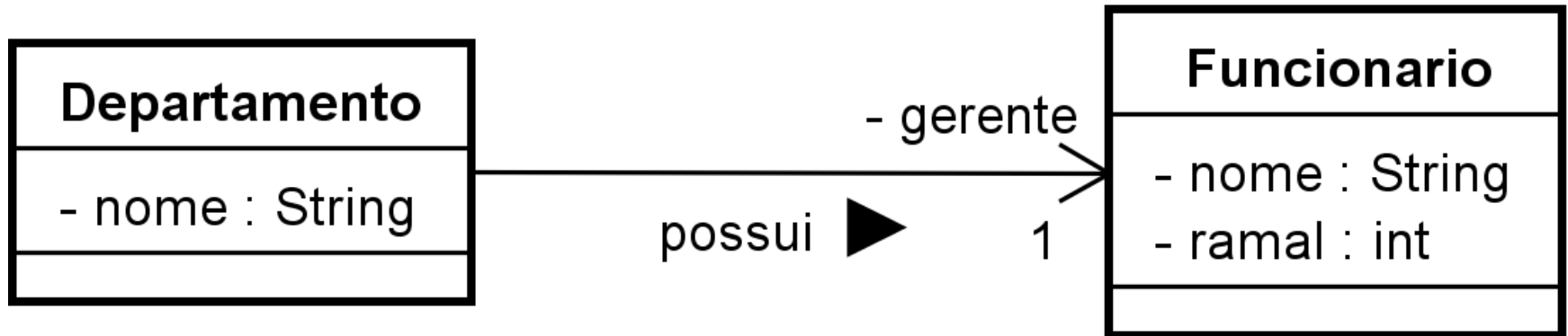
- Tradução para Java requer:
 - Criar uma variável de instância na classe de origem
 - O *identificador desta variável* deve ser igual ao papel que está próximo à classe de destino
 - Na ausência de papel, utilizar o nome da classe como identificador
 - O tipo de dado desta variável deve ser igual à classe de destino

Associações com multiplicidade 0..1 ou 1

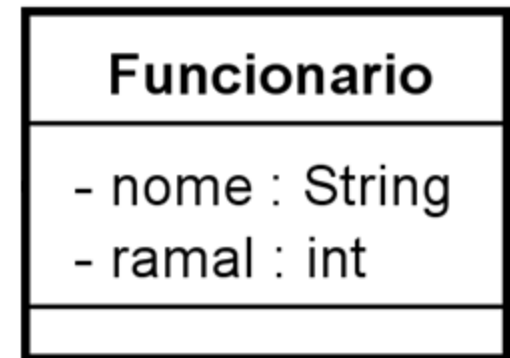
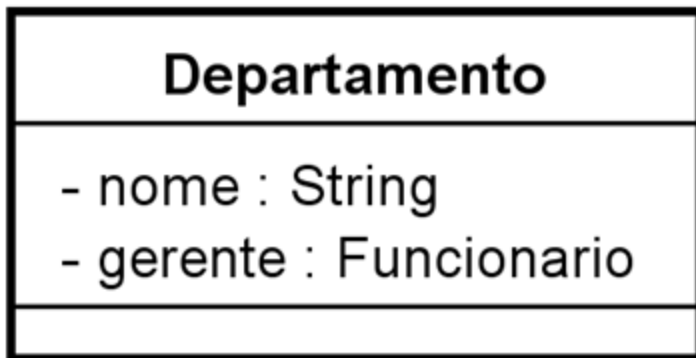


Representação alternativa, que apresenta o relacionamento de forma redundante

Associações com multiplicidade 0..1 ou 1



É equivalente à:



Associações com multiplicidade 0..1 ou 1

- Geralmente utiliza-se métodos de acesso para manipular a nova variável de instância

Departamento
- nome : String - gerente : Funcionario
+ getGerente() : Funcionario + setGerente(func : Funcionario) : void

Exemplo em Java

```
public class Departamento {  
  
    private String nome;  
    private Funcionario gerente;  
  
    public String getNome() {  
        return nome;  
    }  
  
    public void setNome(String nome) {  
        this.nome = nome;  
    }  
  
    public Funcionario getGerente() {  
        return gerente;  
    }  
  
    public void setGerente(Funcionario gerente) {  
        this.gerente = gerente;  
    }  
  
}
```


Exemplo em Java

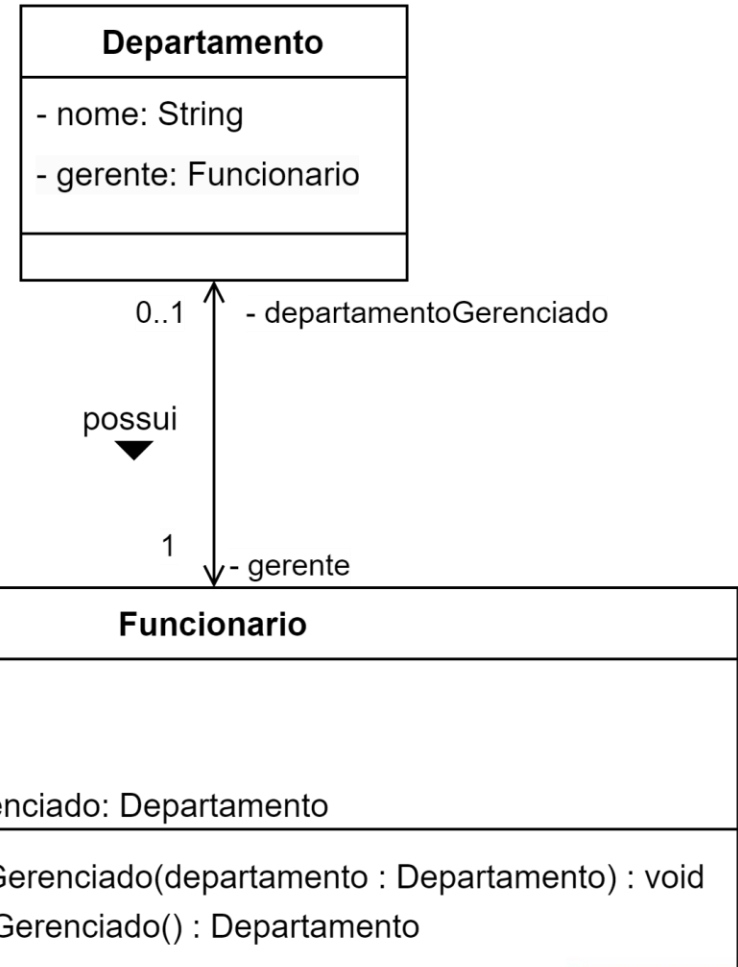
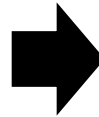
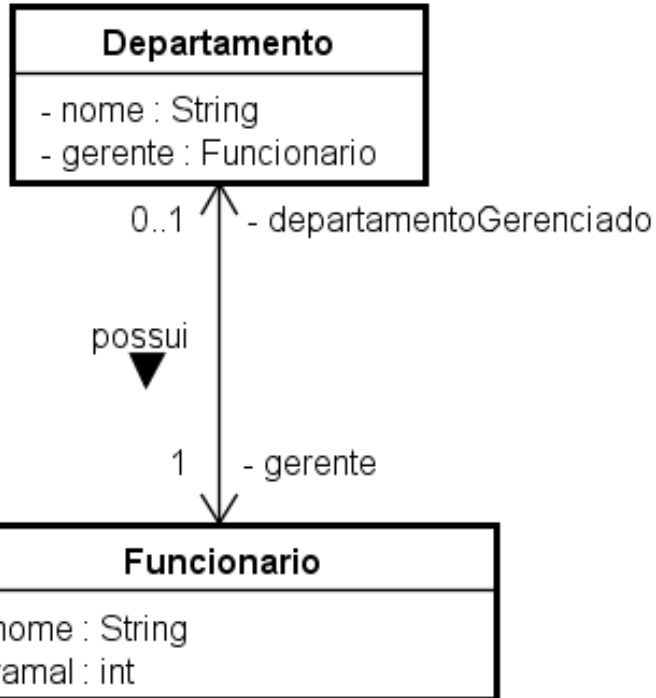
```
public class Funcionario {  
  
    private String nome;  
    private int ramal;  
  
    public String getNome() {  
        return nome;  
    }  
  
    public void setNome(String nome) {  
        this.nome = nome;  
    }  
  
    public int getRamal() {  
        return ramal;  
    }  
  
    public void setRamal(int ramal) {  
        this.ramal = ramal;  
    }  
  
}
```

Exemplo de utilização

```
1 public void exemplo() {  
2     Funcionario f1 = new Funcionario();  
3     f1.setNome("Juliano Korz");  
4     f1.setRamal(4901);  
5  
6     Departamento d1 = new Departamento();  
7     d1.setNome("Contabilidade");  
8  
9     d1.setGerente(f1);  
10 }
```

```
System.out.println( d1.getGerente().getNome() );
```

Relacionamento bidirecional



Exemplo

```
1 public void exemplo() {  
2     Funcionario f1 = new Funcionario();  
3     f1.setNome("Juliano Korz");  
4     f1.setRamal(4901);  
5  
6     Departamento d1 = new Departamento();  
7     d1.setNome("Contabilidade");  
8  
9     d1.setGerente(f1);  
10    f1.setDepartamentoGerenciado(d1);  
11  
12    System.out.println( d1.getGerente().getNome() );  
13    System.out.println( f1.getDepartamentoGerenciado().getNome());  
14 }
```

Tradução diagrama de classes

Tradução diagrama de classes

1. Traduzir 1º compartimento
2. Traduzir 2º compartimento
3. Traduzir relacionamentos
4. Traduzir 3º compartimento