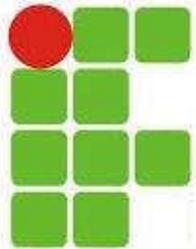


---

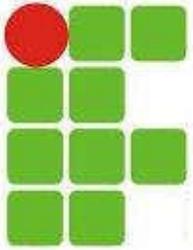
# PROJETO DE DESENVOLVIMENTO DE SOFTWARE

**Professor: Diego Oliveira**



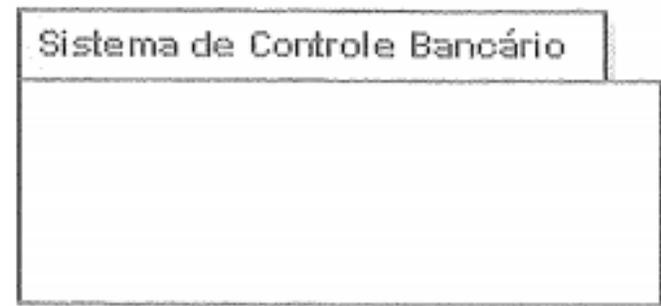
**Aula 14:**  
**Diagrama de Pacotes**





# Diagrama de Pacotes

- Descreve como os elementos estão organizados em pacotes e suas dependências
- É útil para a modelagem de subsistemas e subdivisões da arquitetura
- Exemplo de pacote:

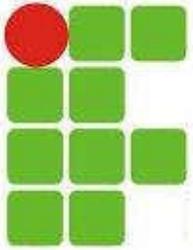




# Diagrama de Pacotes

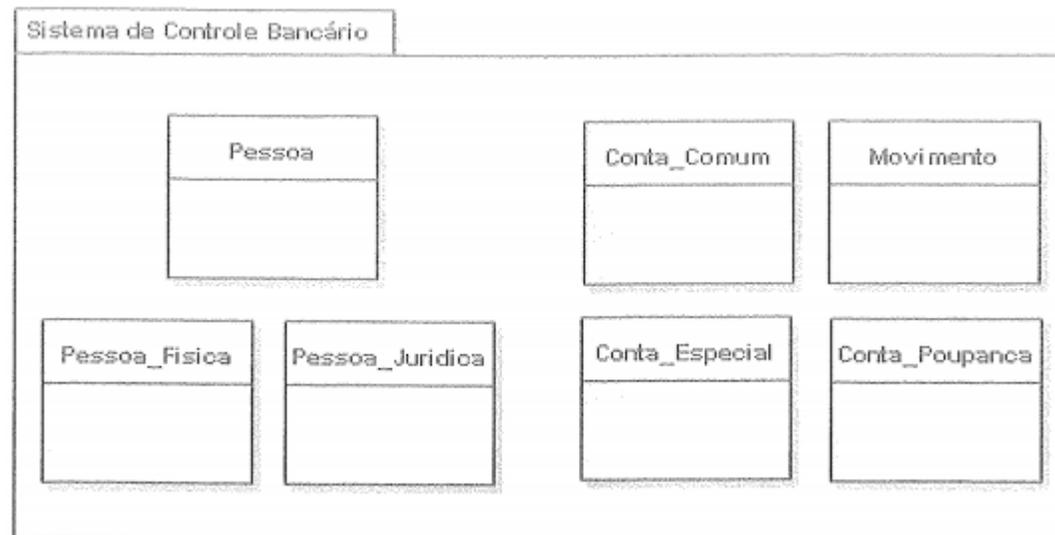
- Os pacotes são utilizados para agrupar elementos e fornecer denominações para os grupos
- Um pacote pode representar um sistema, subsistema, biblioteca, etapa de desenvolvimento ou outras 'partes' do sistema
- Um pacote pode mostrar as classes contidas, por exemplo, se for interessante para o diagrama naquele momento





# Diagrama de Pacotes

- Exemplo de pacote com classes internas:



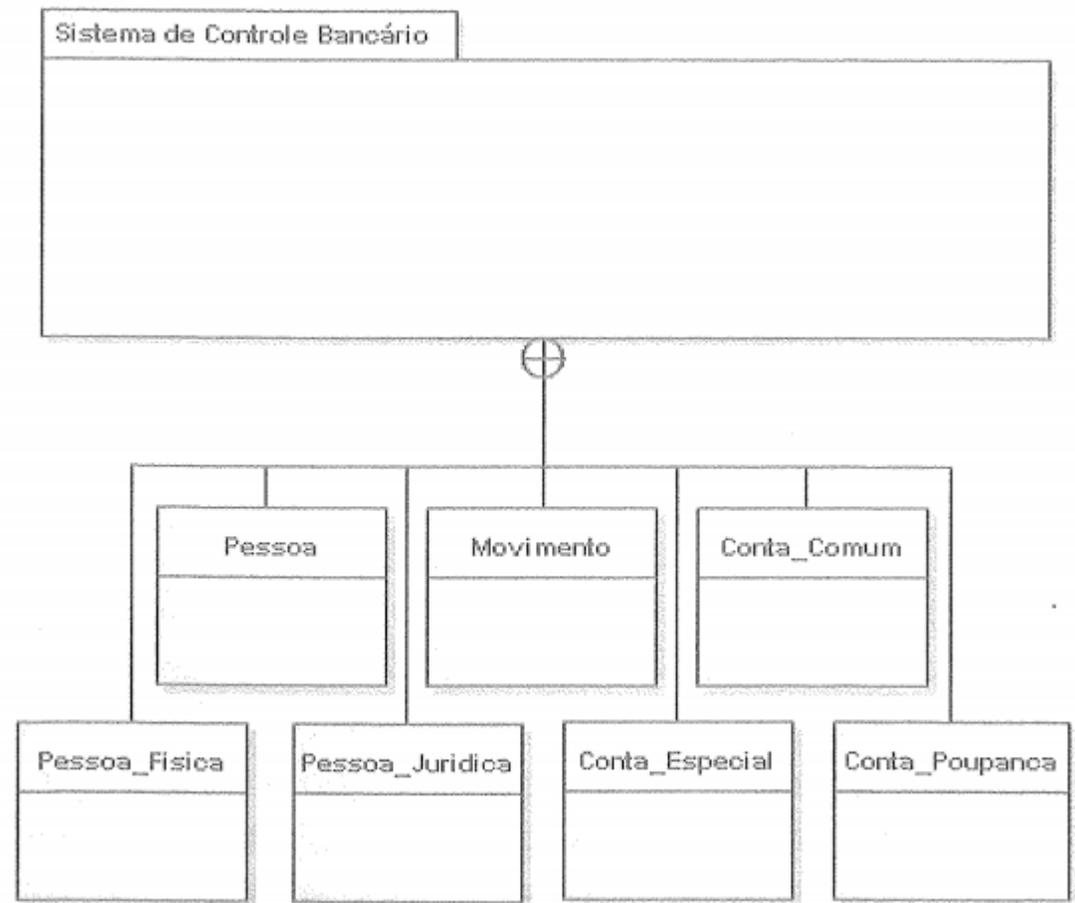
- Note que os métodos e atributos das classes foram omitidos pois para esta representação não são importantes

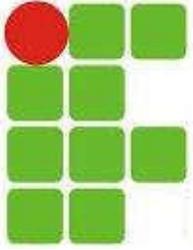




# Diagrama de Pacotes

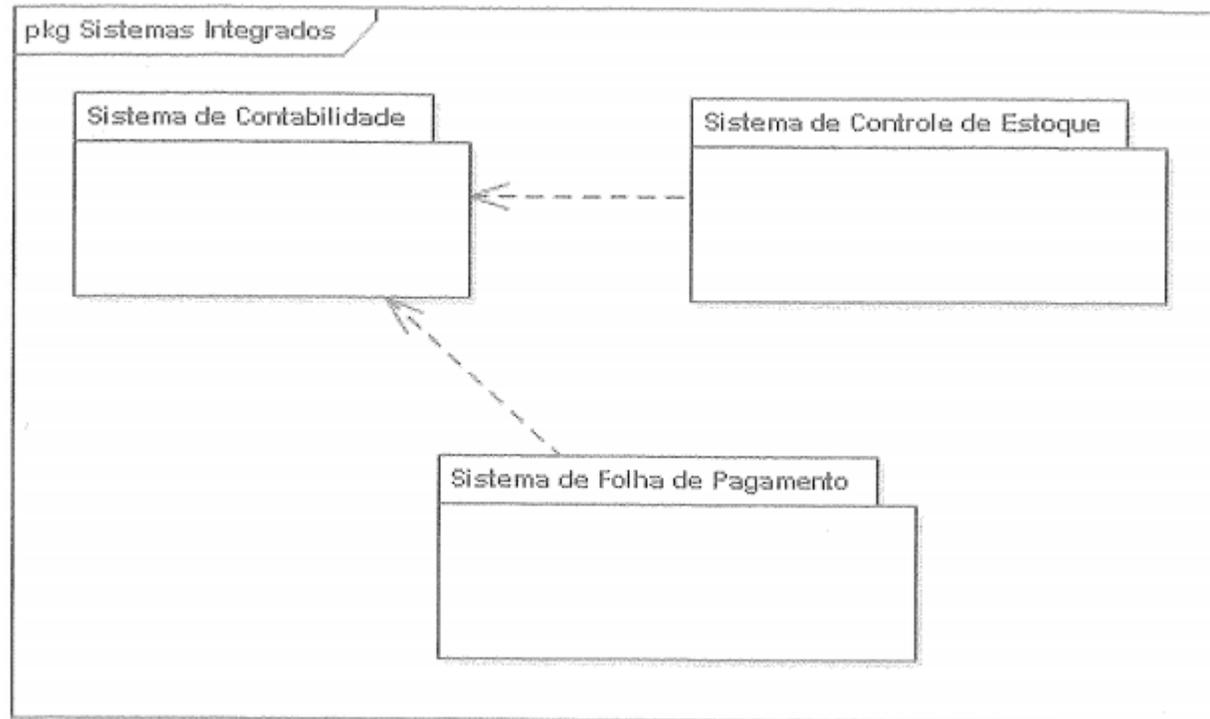
- Outra representação quando houver muitas classes dentro de um pacote:





# Diagrama de Pacotes

- Pacotes também podem ter dependências entre si:





# Diagrama de Pacotes

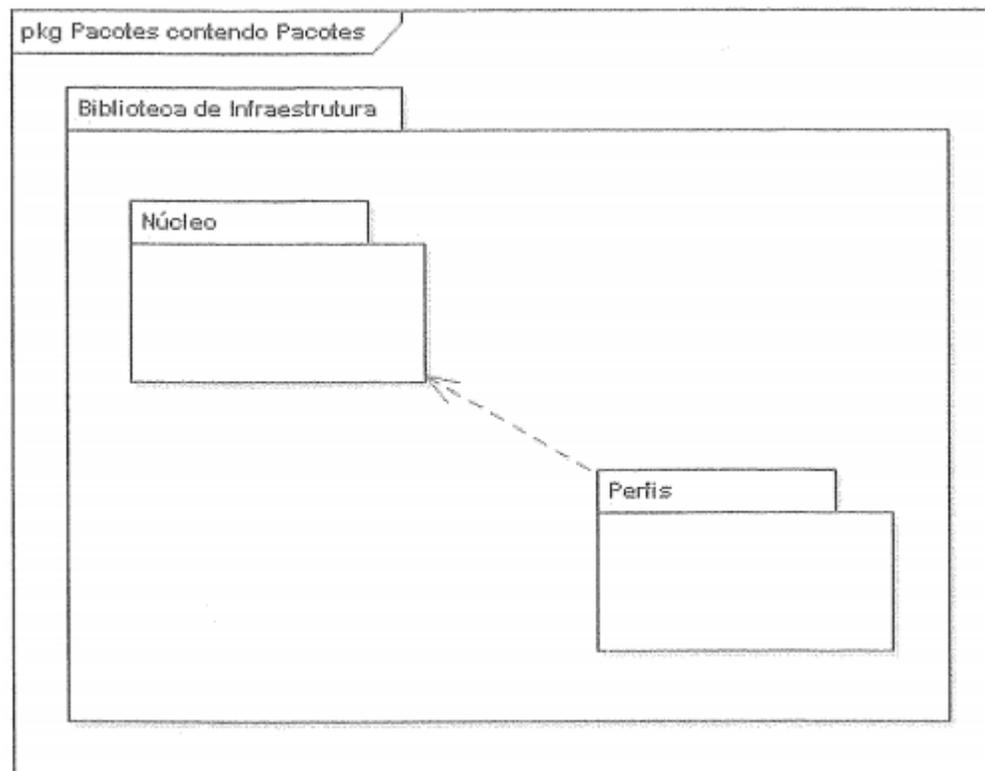
- As dependências podem ter dois tipos de estereótipos:
  - <<merge>> o pacote e sua dependência serão unidos em um só pacote
  - <<import>> o pacote de onde a seta parte importa alguma característica do outro pacote
- O <<import>>, a princípio, pode lembrar o import do Java, porém lembre-se que no Java estamos em uma CLASSE e aqui estamos falando de um PACOTE.

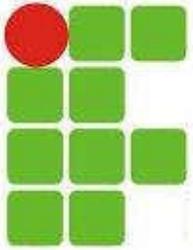




# Diagrama de Pacotes

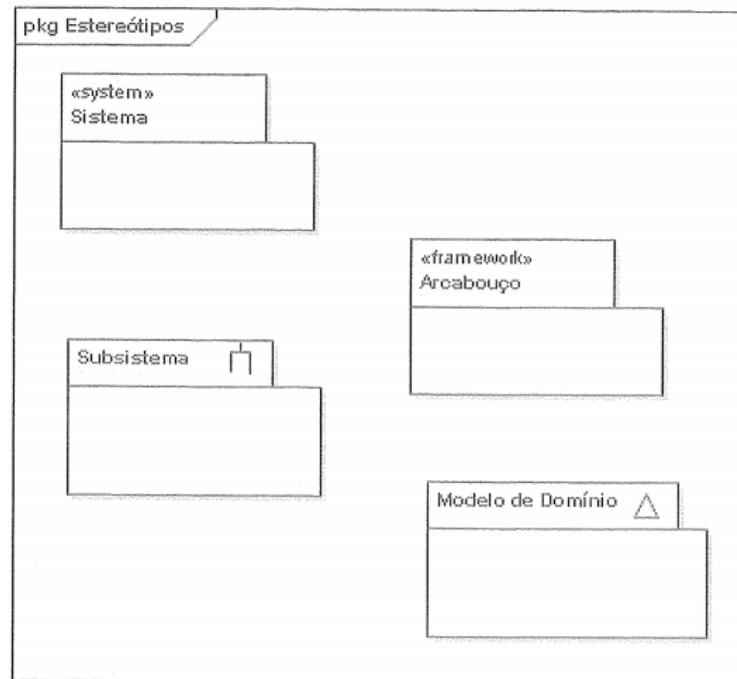
- Pacotes podem conter subpacotes, como acontece no Java:



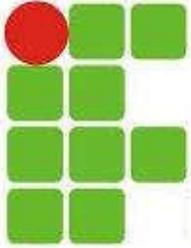


# Diagrama de Pacotes

- É possível aplicar estereótipos aos pacotes:



- Lembre-se de que é possível criar seus próprios estereótipos!



## Exercício

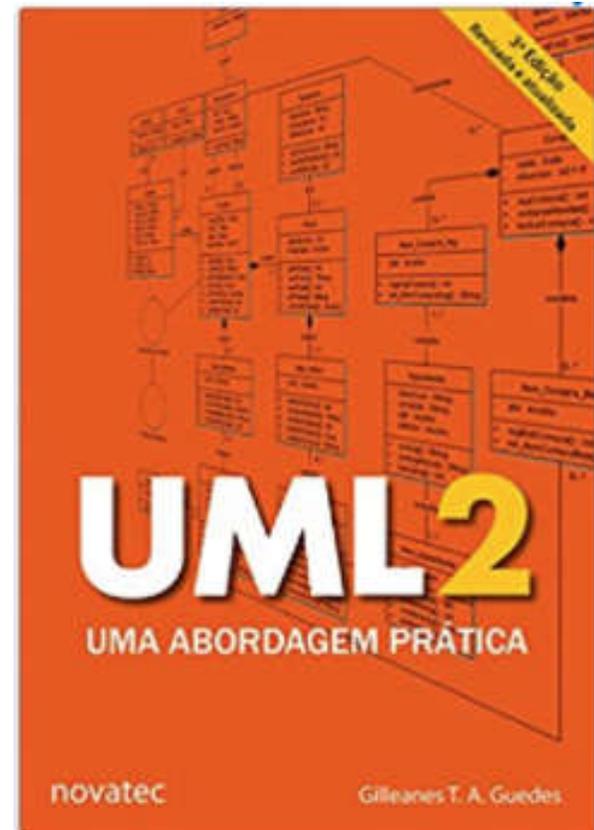
- Crie o diagrama de pacotes do SUAP com base nos diagramas anteriores já feitos
- Lembre-se de representar as dependências, estereótipos e subpacotes
- Os pacotes mais importantes devem representar suas classes internas
- Será preciso entrevistar alguém da T.I. para tomar conhecimento de possíveis **subsistemas** do SUAP e sistemas de terceiros

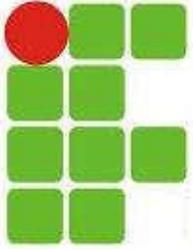




# Referências

- UML2: Uma Abordagem Prática  
3ª Ed. 2018  
Gilleanes T. A. Guedes





# Perguntas?

