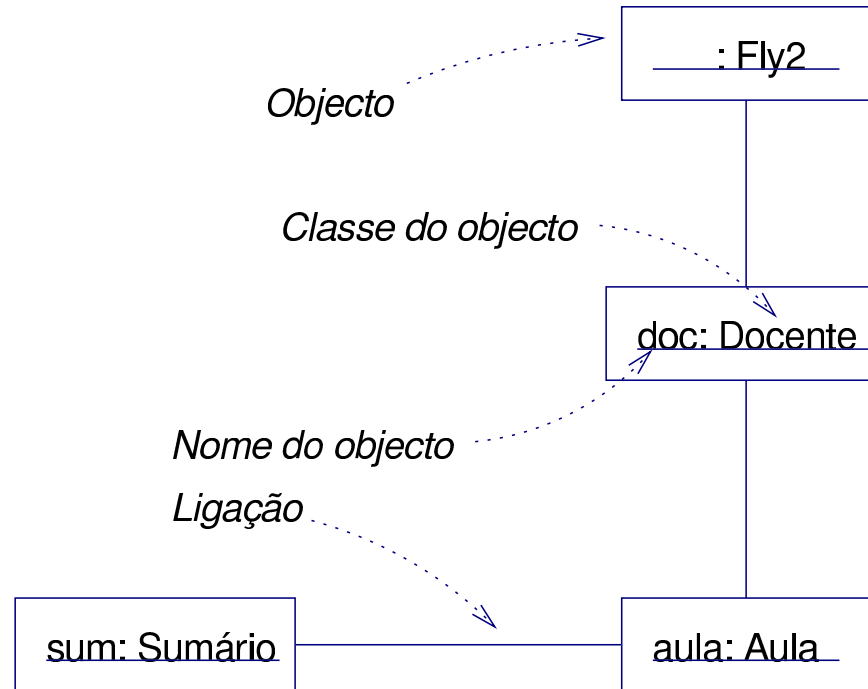


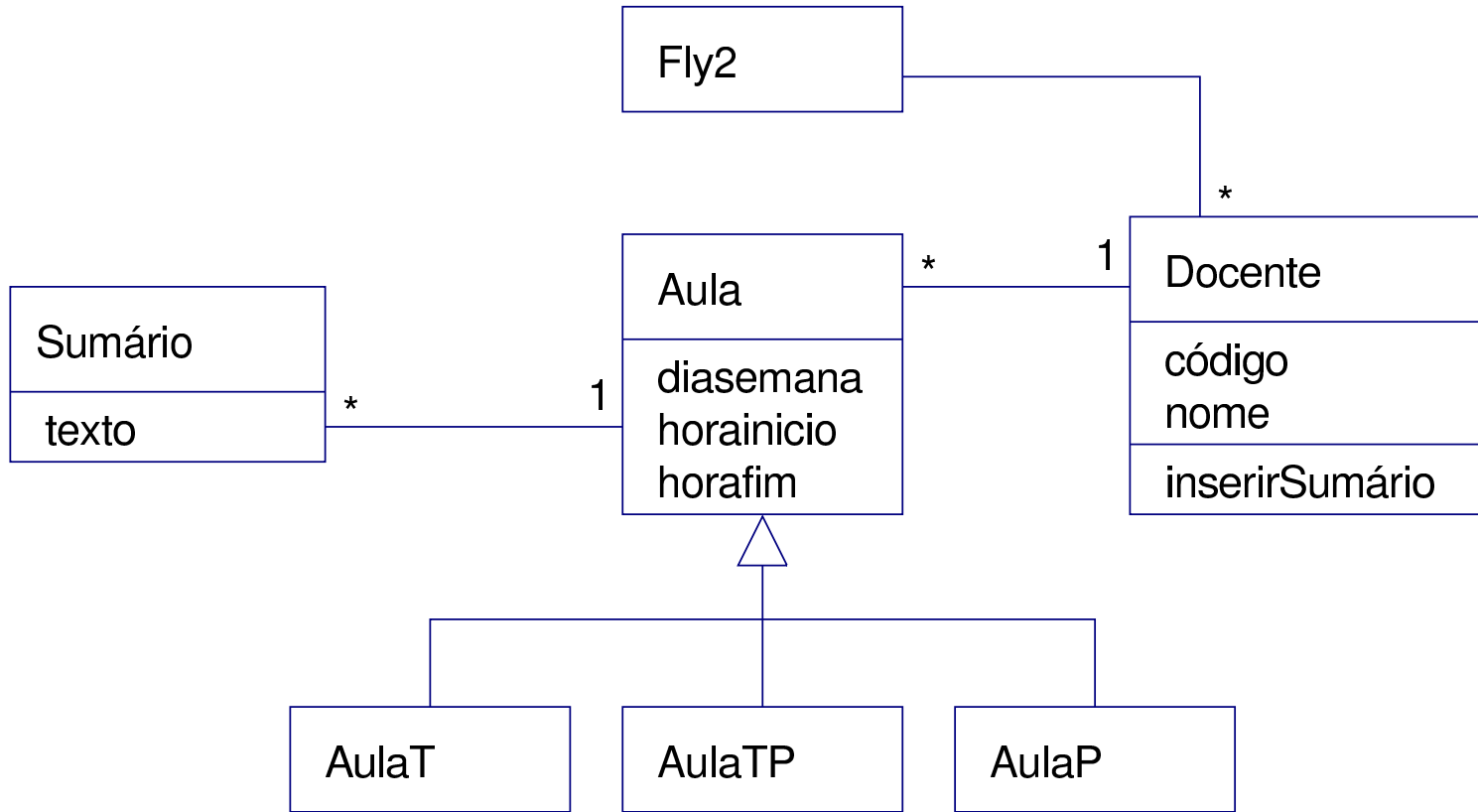


Diagramas de Objectos

131/168



- Apresentam uma configuração particular de objectos no sistema.
- Modelam a visão estática do sistema, do ponto de vistas de instâncias reais:
 - objectos são instâncias das classes do modelo;
 - ligações são instâncias das associações entre as classes.
- Útil, por exemplo, durante *debugging*.





Diagramas de Interação

Sumário

- Tipos de Diagramas de Interação
- Interações
- Diagramas de Colaboração
- Diagramas de Sequência

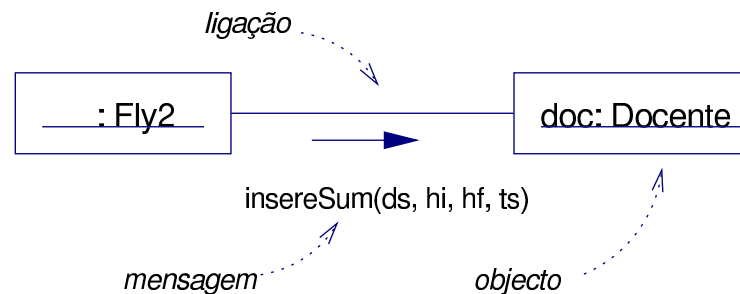


Tipos de Diagramas de Interação

- Os diagramas de classe modelam a arquitectura de objectos do sistema.
- Os Diagramas de Interação modelam o diálogo entre os objectos que compõem o sistema.
- O UML define dois tipos de diagramas de interação:
 - Diagramas de Colaboração;
 - Diagramas de Sequência.

Interações

- Uma interação representa comunicação entre objectos (envio de mensagens).



- Ligação:
 - Instância de uma associação
 - Decorações:

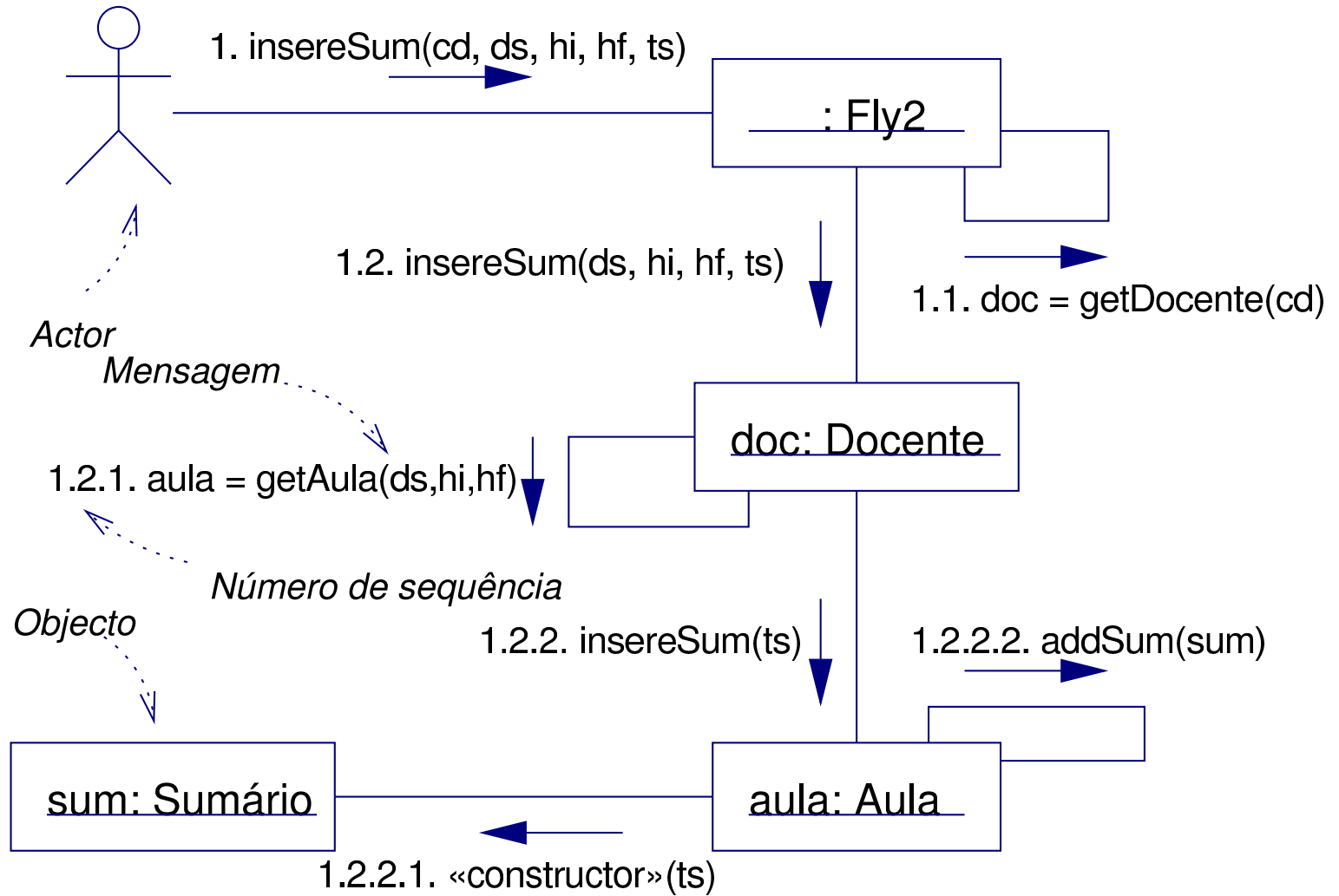


- * association — a instância é conhecida por intermédio de uma associação
- * self — a instância refere-se a ele própria
- * global/local/parameter — *scope* da instância num dado contexto
- Mensagens:
 - * Especificam comunicação entre objectos.
 - * Tipos
 - call — invocação síncrona (\rightarrow)
 - asynchronous — invocação assíncrona (\dashrightarrow)
 - return — (\leftarrow)
 - create — criar objecto
 - destroy — destruir objecto



Diagramas de Colaboração

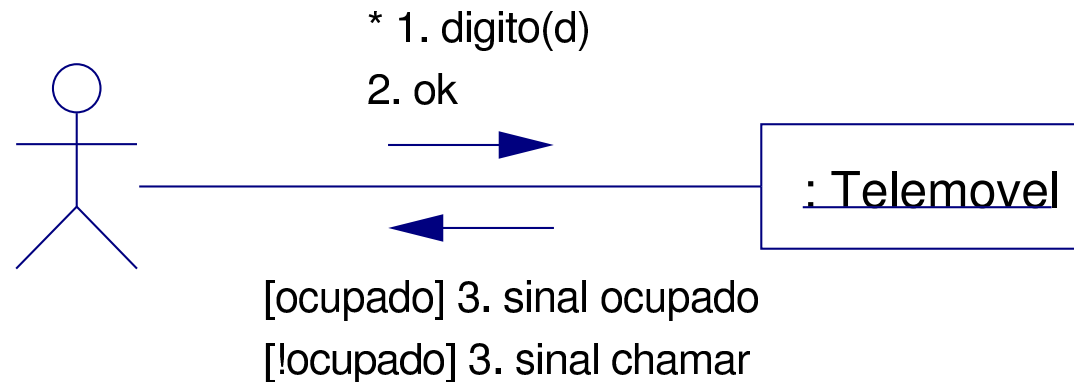
- Apresentam uma configuração específica do sistemas que suporta um dado use case (que objectos deverão existir em *runtime*).
- Para essa configuração, mostra como os objectos comunicam entre si (colaboram) para satisfazer um dado pedido.
- A ênfase está na organização estrutural dos objectos.
 - Permite analisar se a configuração necessária é suportada pela estrutura de classes idealizada nos diagramas de classe.
- Ordenação temporal das mensagens não é claramente representada.





Lógica Condicional

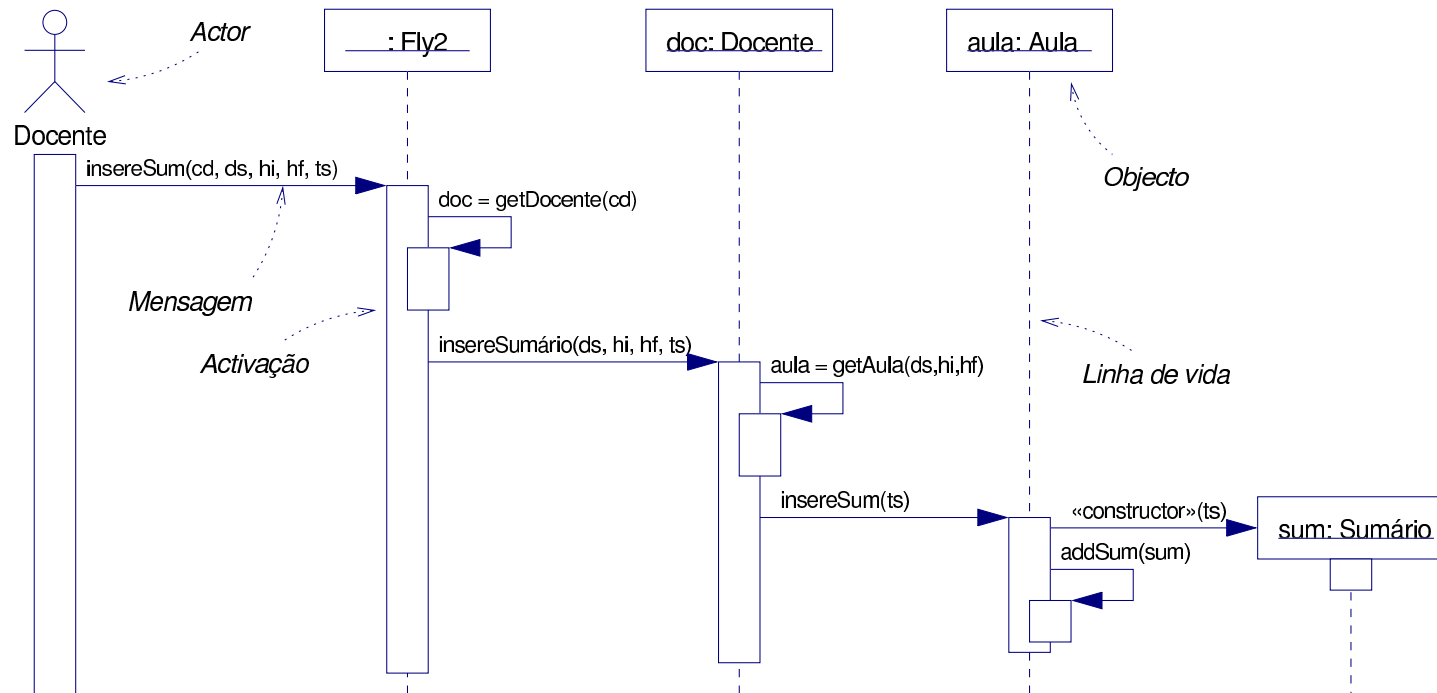
- A notação suporta a modelação de estruturas repetitivas e condicionais:
 - Iteração: *[cond]
 - Condição: [cond]
- Em alternativa, podemos utilizar mais que um diagrama para modelar o comportamento.



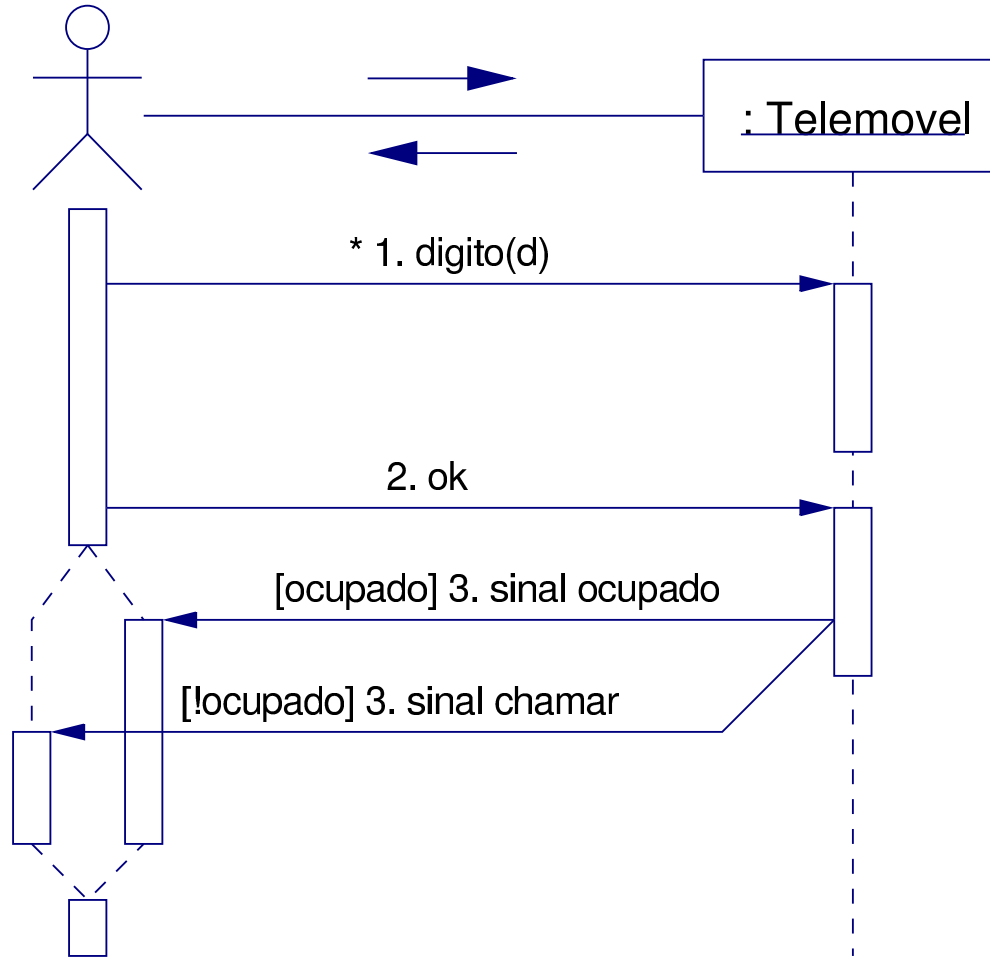


Diagramas de Sequência

- Representam o mesmo que os diagramas de colaboração, mas a ênfase é posta na ordenação temporal das mensagens.
- Permitem analisar a distribuição de “responsabilidades” pelas diferentes classes (analisar onde está a ser efectuado o processamento).



Lógica Condicional





Diagramas de Estado

Sumário

- Aplicação
- Exemplos
- Estados e Transições

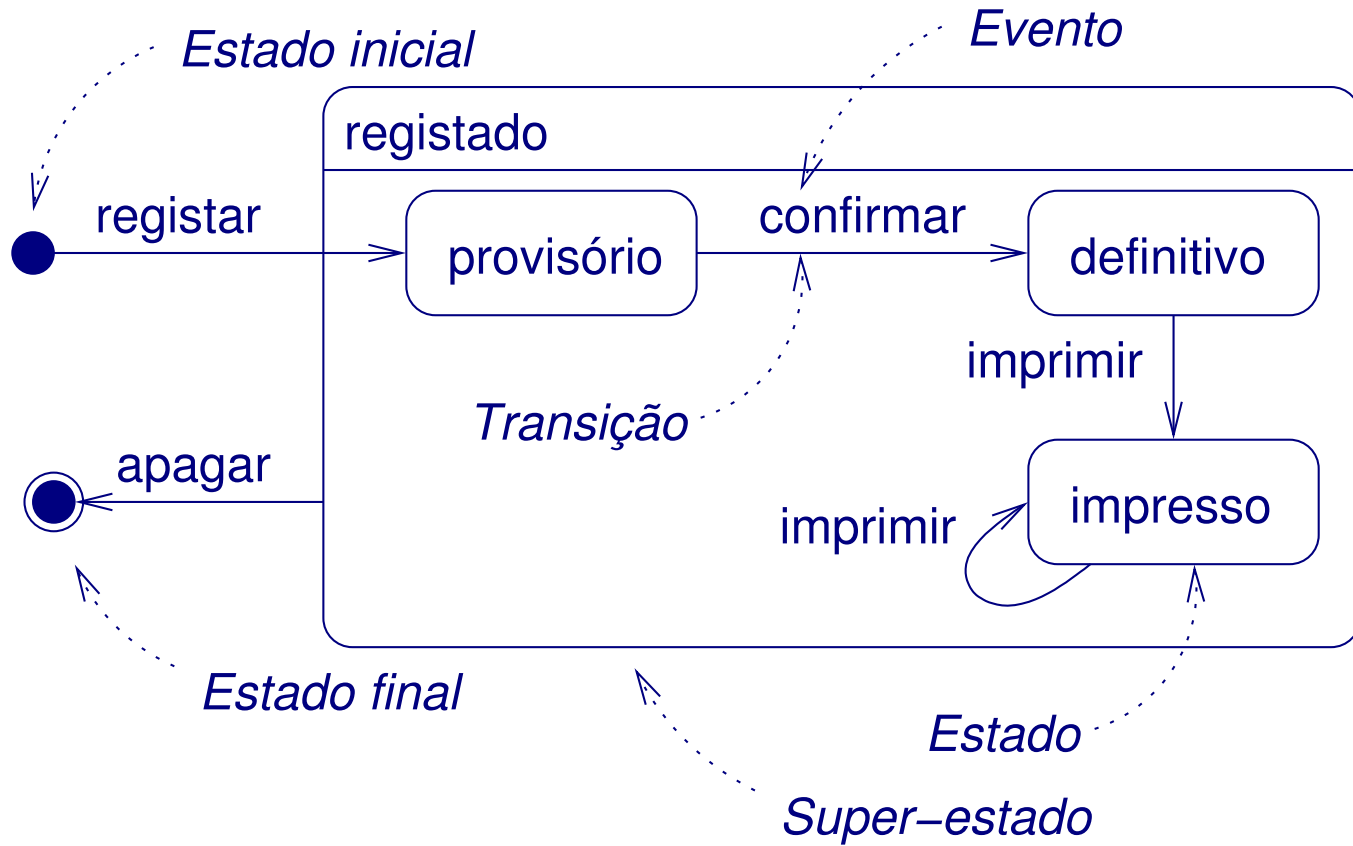


Aplicação

- Diagramas de Interação modelam a colaboração entre objectos tendo em vista atingir um determinado fim — para cada objecto modelam apenas parte do seu comportamento.
- Os Diagramas de Estado permitem modelar o comportamento de um dado objecto/sistema de forma global.
- A ênfase é posta no estado do objecto/sistema — modelam-se todos os estados possíveis que o objecto/sistema atravessa em resposta aos eventos que podem ocorrer.
- Úteis para modelar o comportamento de um objecto de forma transversal aos *use cases* do sistema.
- Devem usar-se para classes em que se torna necessário compreender o comportamento do objecto de um modo global ao sistema.
- Nem todas as classes vão necessitar de diagramas de estado.



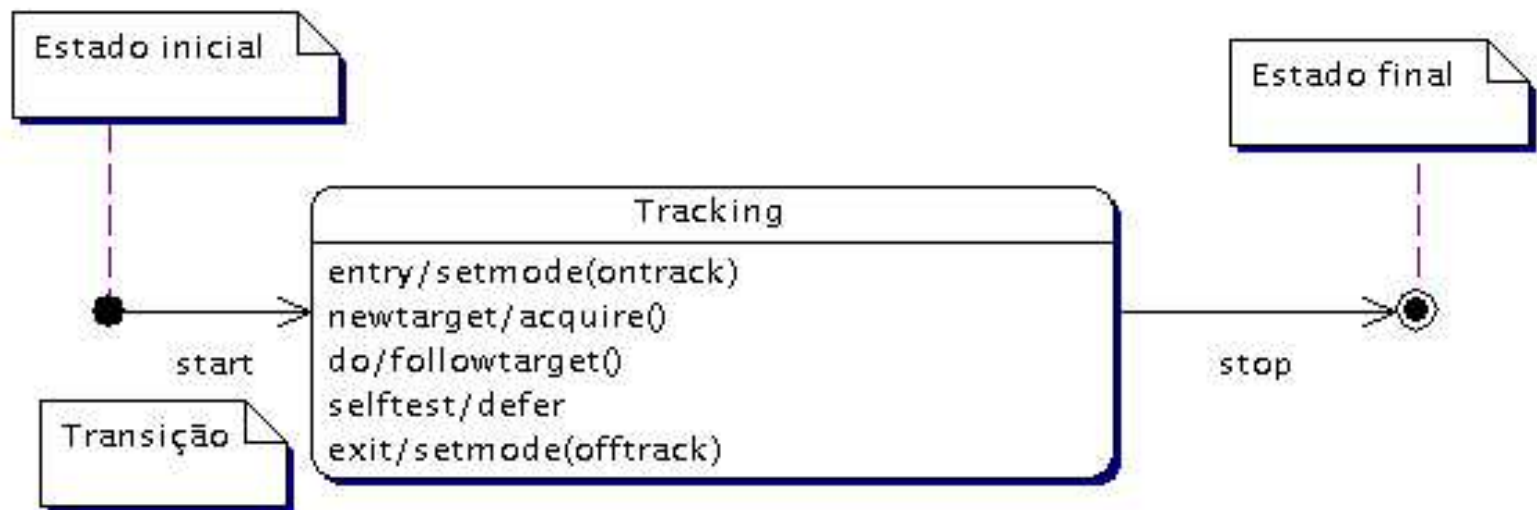
Exemplo

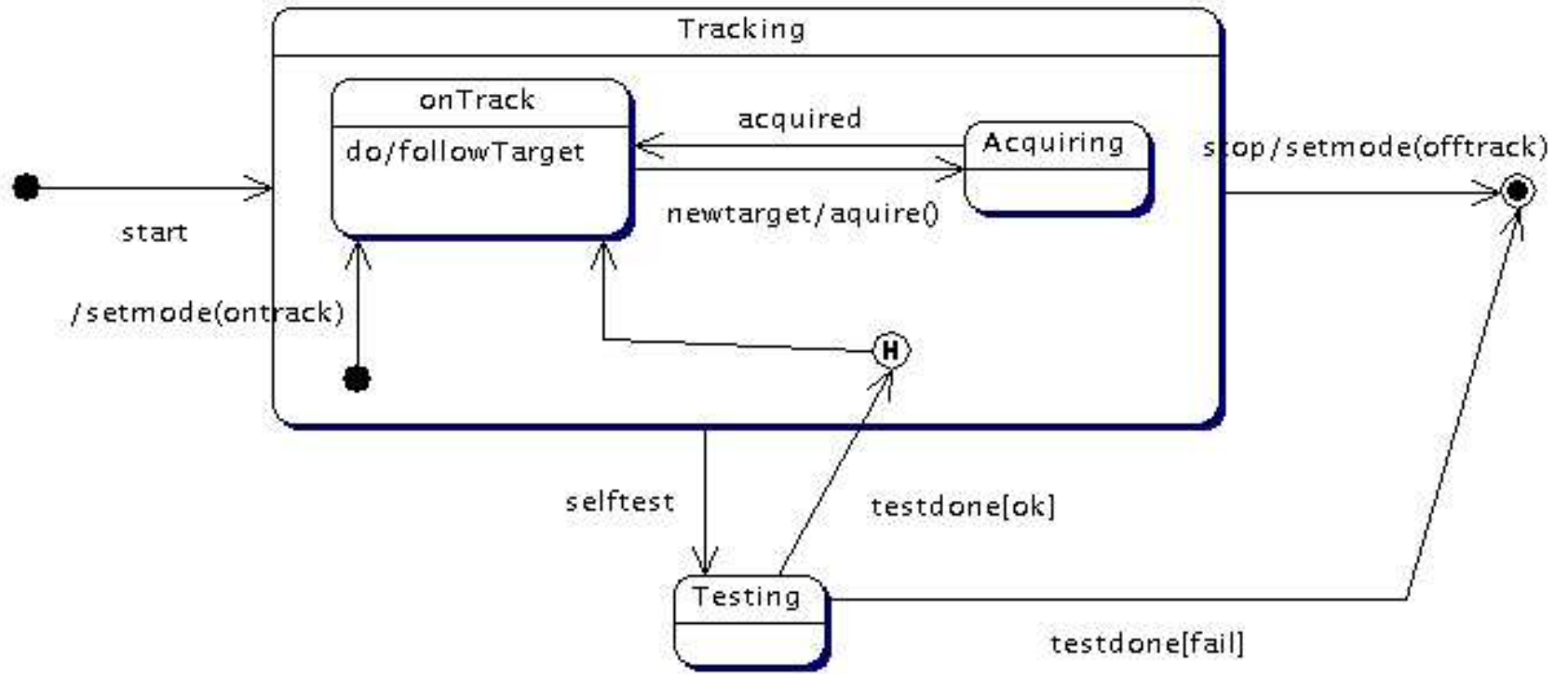




Estados e Transições

- estado inicial — estado do objecto quando é criado
- estado final — destruição do objecto
- estado com história — permite modelar interrupções;
- transições (internas/externas) — evento[condição]/acção







Diagramas de Actividade

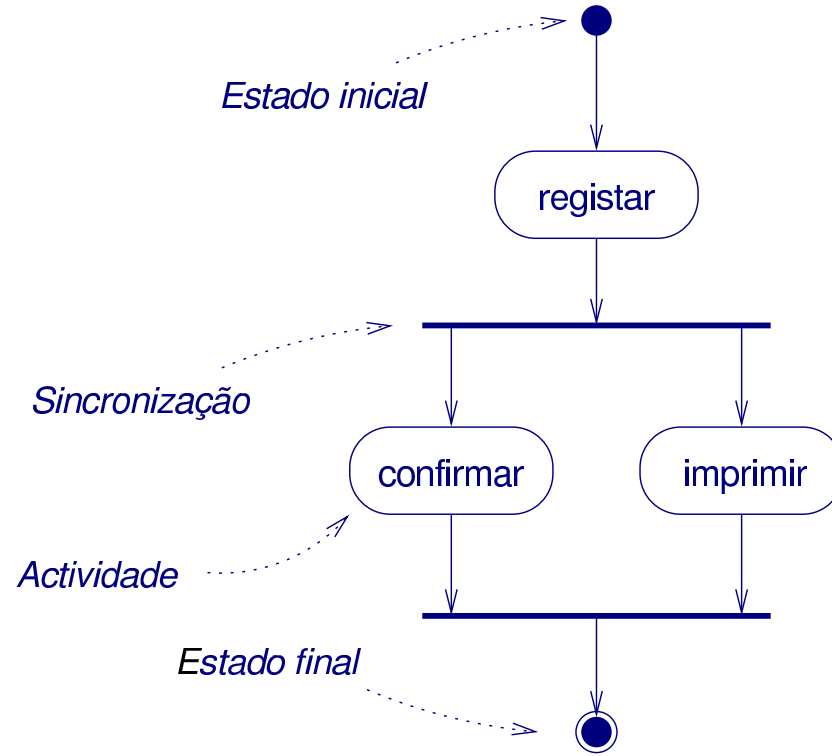
Sumário

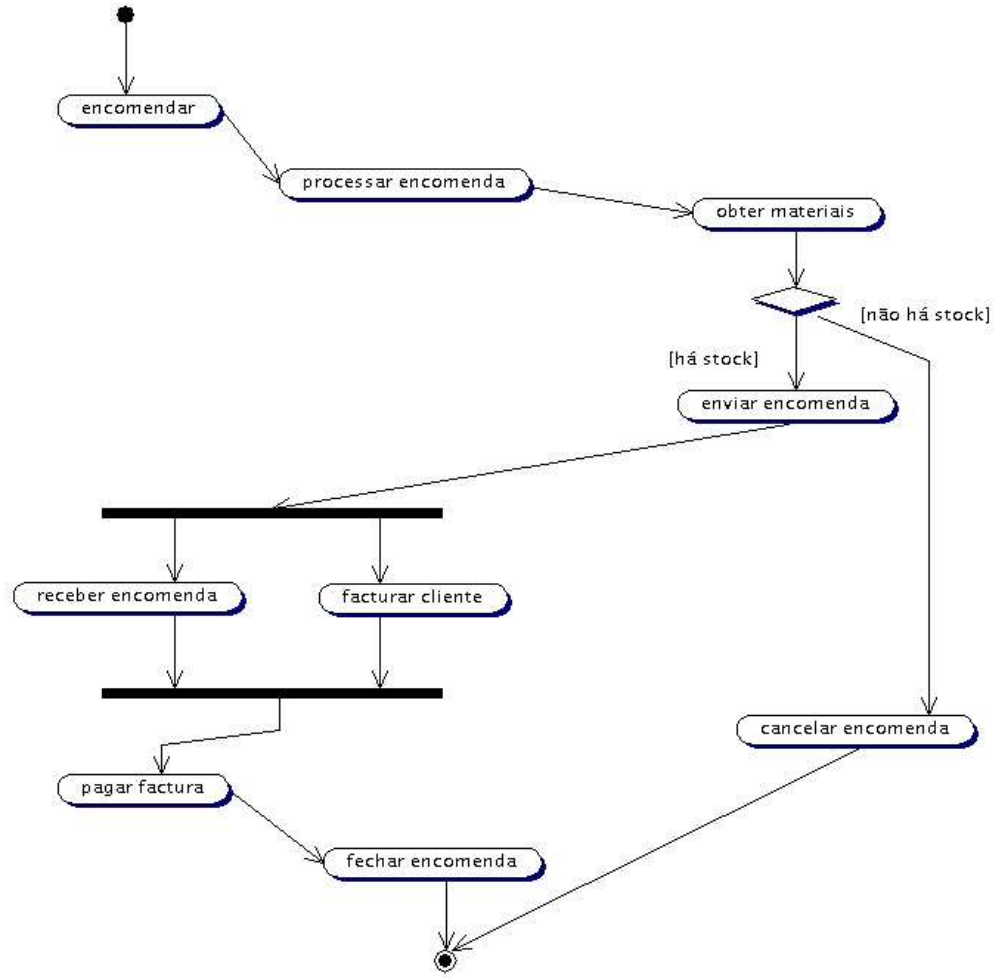
- Aplicação
- Exemplos

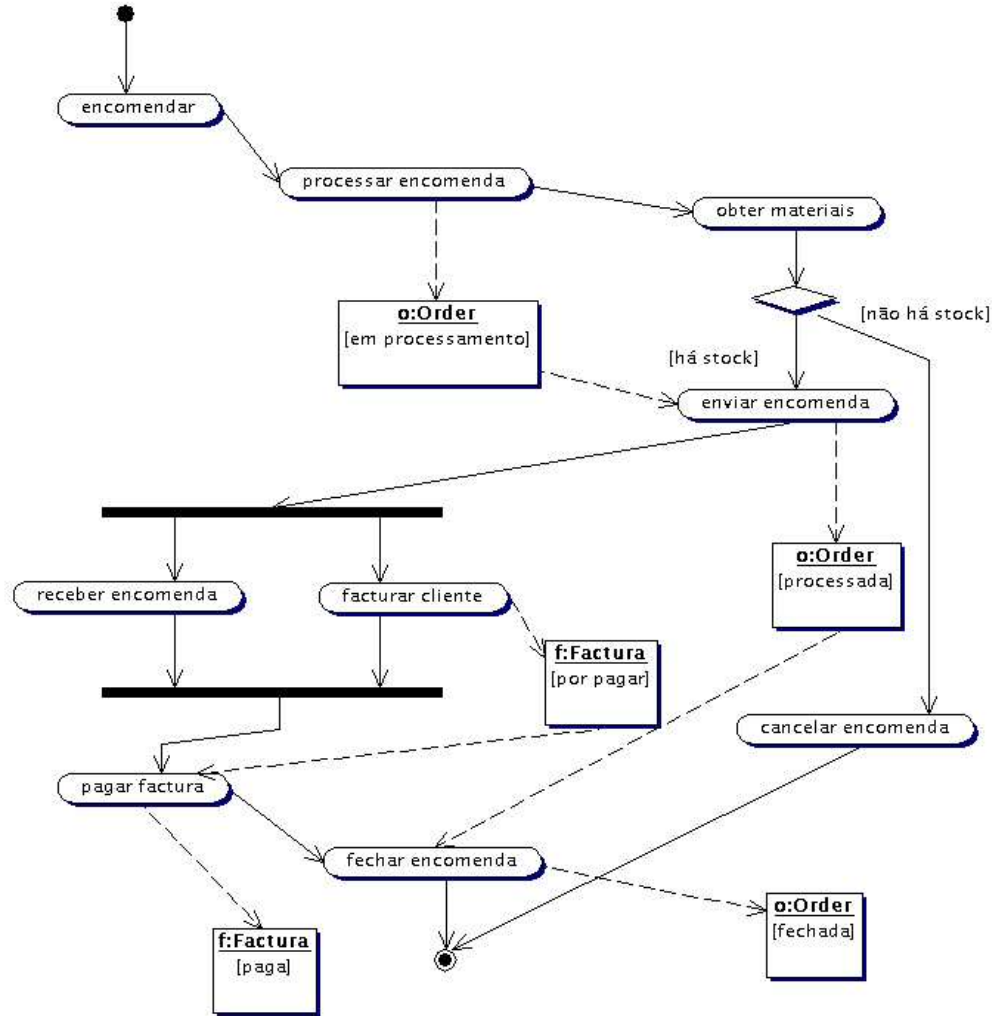


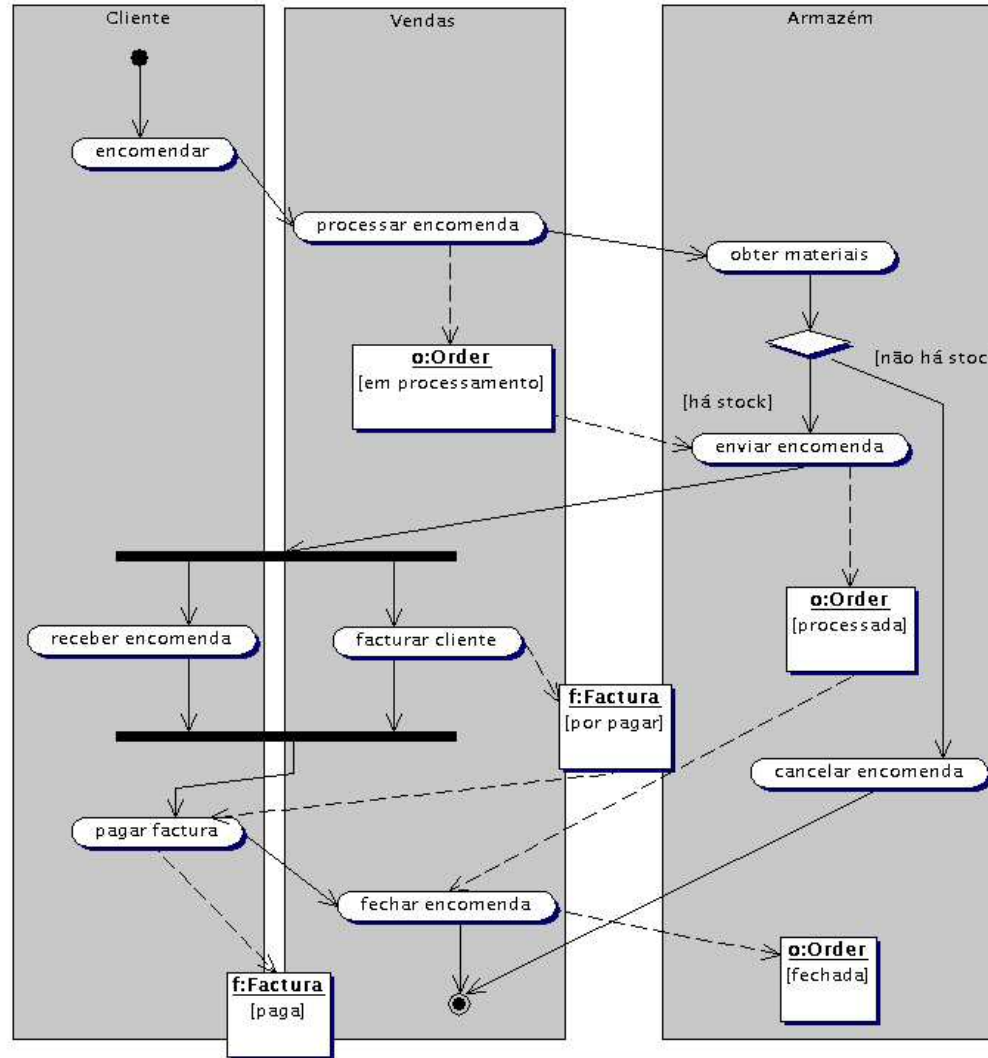
Aplicação

- Diagramas de estado modelam comportamento dando ênfase aos estados que os objectos atravessam.
- Diagramas de actividade são uma variação dos diagramas de estado em que os estados representam actividades.
- Diagramas de actividade modelam comportamento dando ênfase às actividades realizadas.
- Úteis para modelar o comportamento associado a operações de um objecto/sistema.
- Permitem modelar o fluxo de objectos entre as actividades.
- Permitem modelar que é responsável por que actividade.











Diagramas de Componentes

Sumário

- Aplicação
- Exemplo

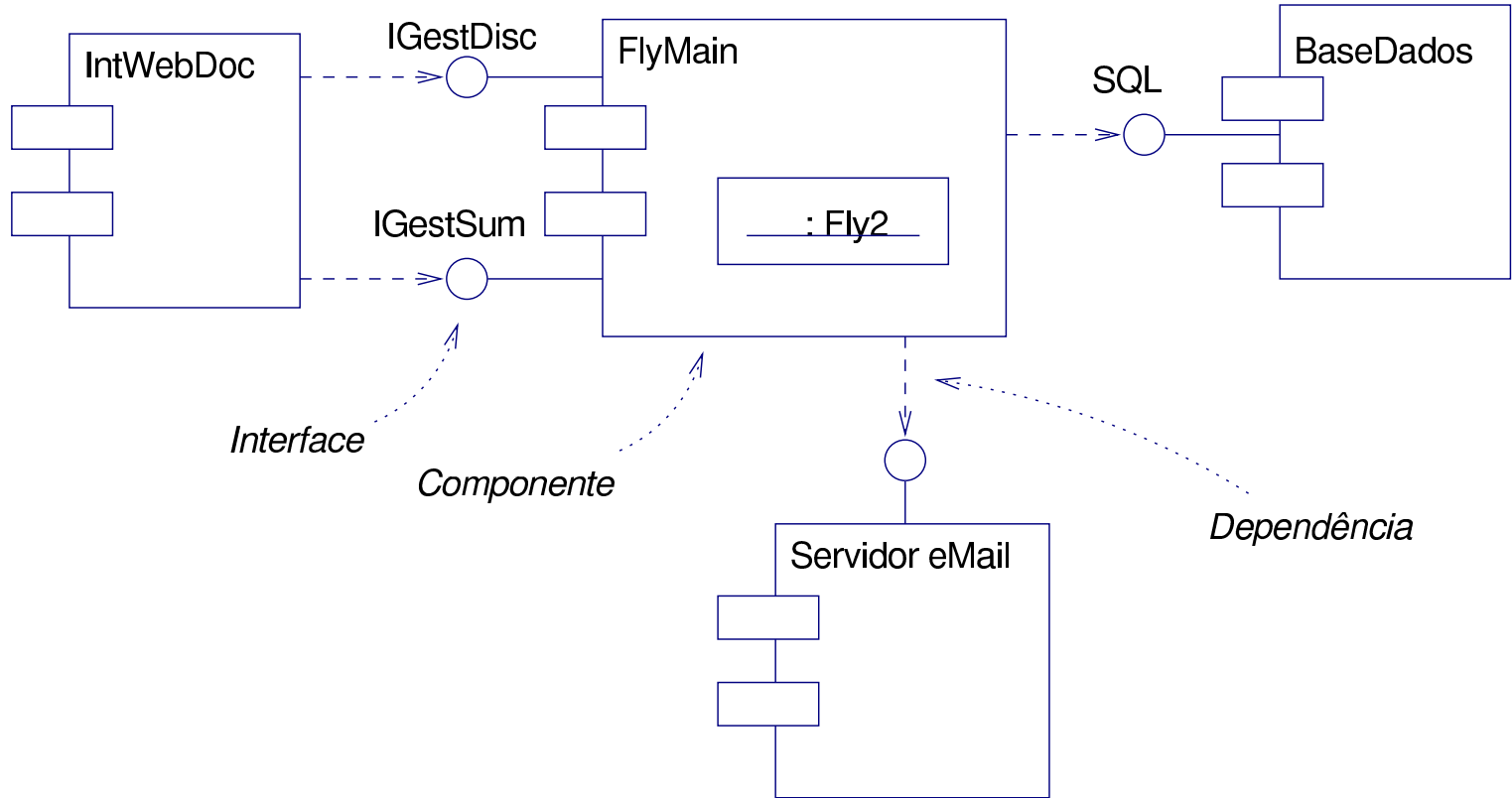


Aplicação

- Um diagrama de componentes mostra as depêndencias entre os componentes software do sistema.
- Pode ser desenhado a diferentes níveis: código fonte, componentes binários (dlls), componentes executáveis.
- Permitem identificar, a cada nível, o que é necessário para construir o sistema.



Exemplo





Diagramas de Instalação

Sumário

- Aplicação
- Exemplo



Aplicação

- Diagramas de instalação modelam a configuração de *run-time* do sistema.
- Mostram apenas os componentes que existem em run-time, e as máquinas (nodos) em que estão instalados.



Exemplo

