

MACIASZEK, L.A. (2007):
Requirements Analysis and System Design, 3rd ed.
Addison Wesley, Harlow England
ISBN 978-0-321-44036-5

Chapter 4
Moving from Analysis to Design

© Pearson Education Limited 2007

Topics

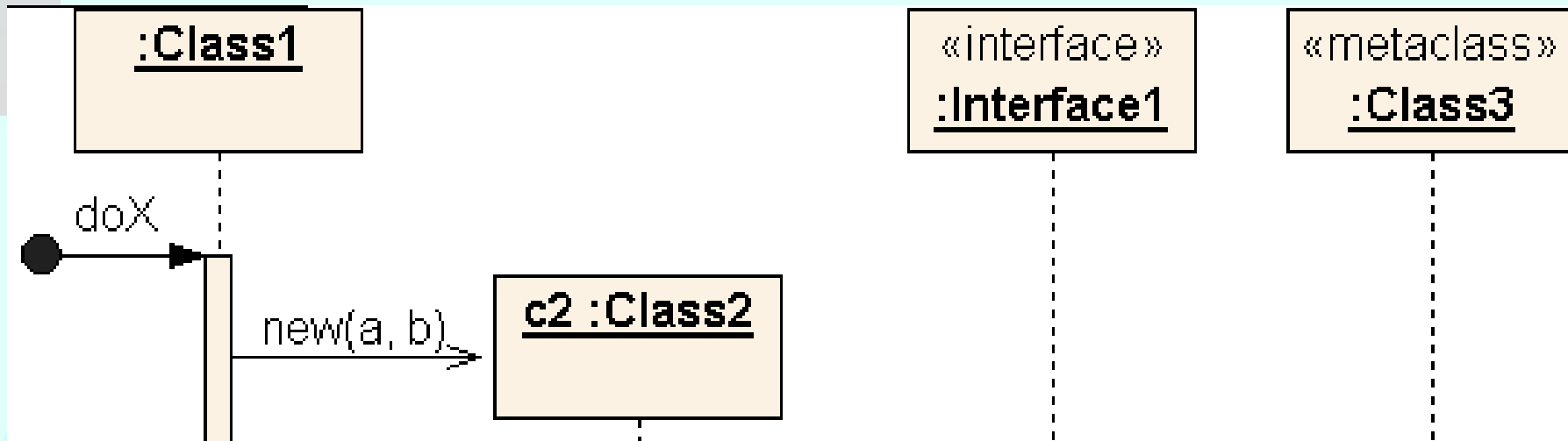
- Advanced class modeling
- Advanced generalization and inheritance modeling
- Advanced aggregation and delegation modeling
- Advanced interaction modeling

4. *Advanced interaction modeling*

- *sequence diagrams* concentrate on the sequence of messages
- *communication diagrams* emphasize object relationships

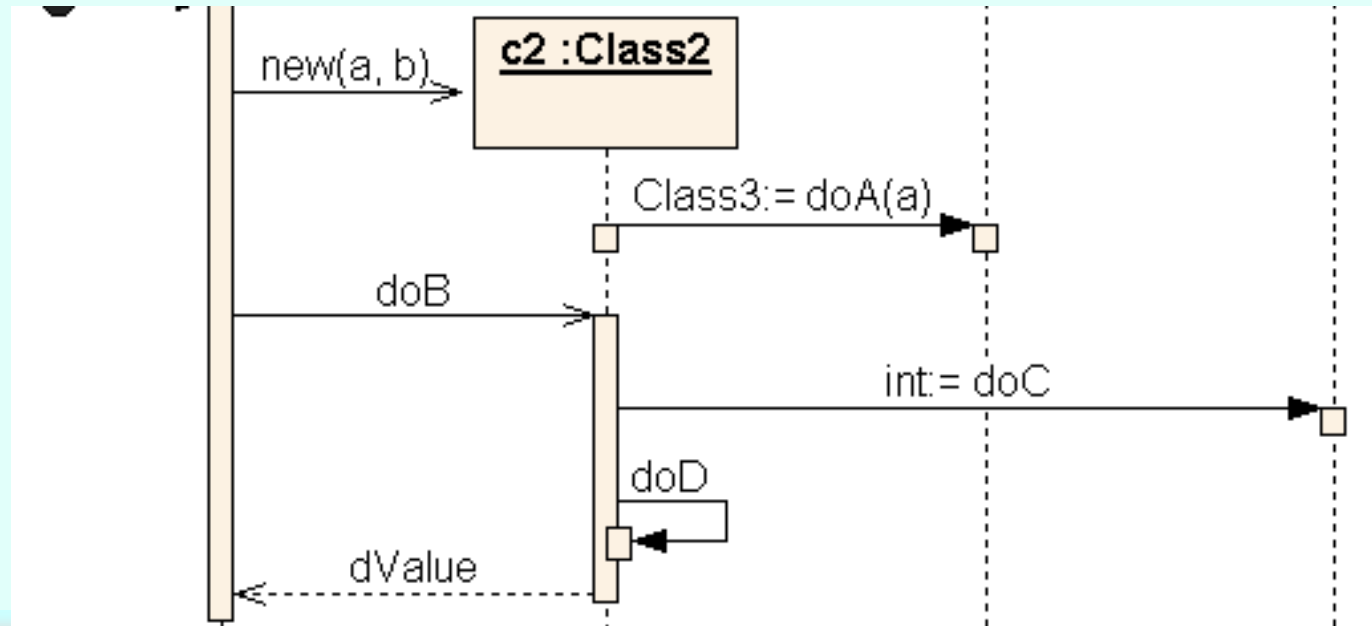
Lifelines

- A lifeline box can be named to represent:
 - an unnamed instance of a class (`:Class1`)
 - a named instance of a class (`c2:Class2`)
 - a class, i.e. an instance of a metaclass (`:Class 3`)
 - to show static method calls to a class itself
 - an interface (`:Interface1`)

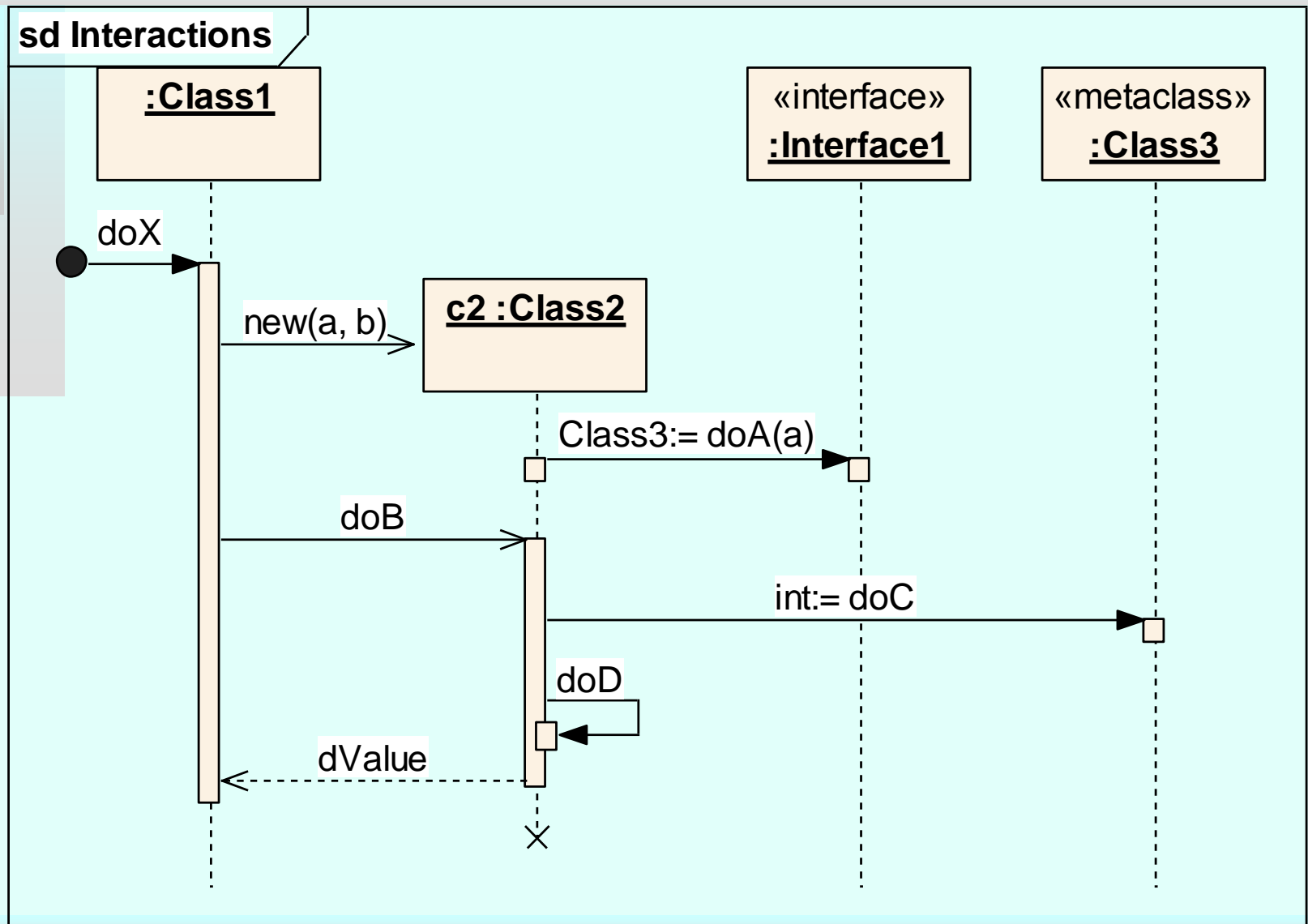


Messages

- **synchronous messages** in which the caller blocks, i.e. it waits for a response – a filled arrow head (doA, doC, doD)
- **asynchronous messages** in which the caller does not block, thus allowing multi-threaded executions – an open arrow (doB)
- **object creation messages** – a line with an open arrow (new(a,b))
- **reply messages** that transmit output values to the caller – a dashed line with an open arrow (dValue)



Interactions – notation by example



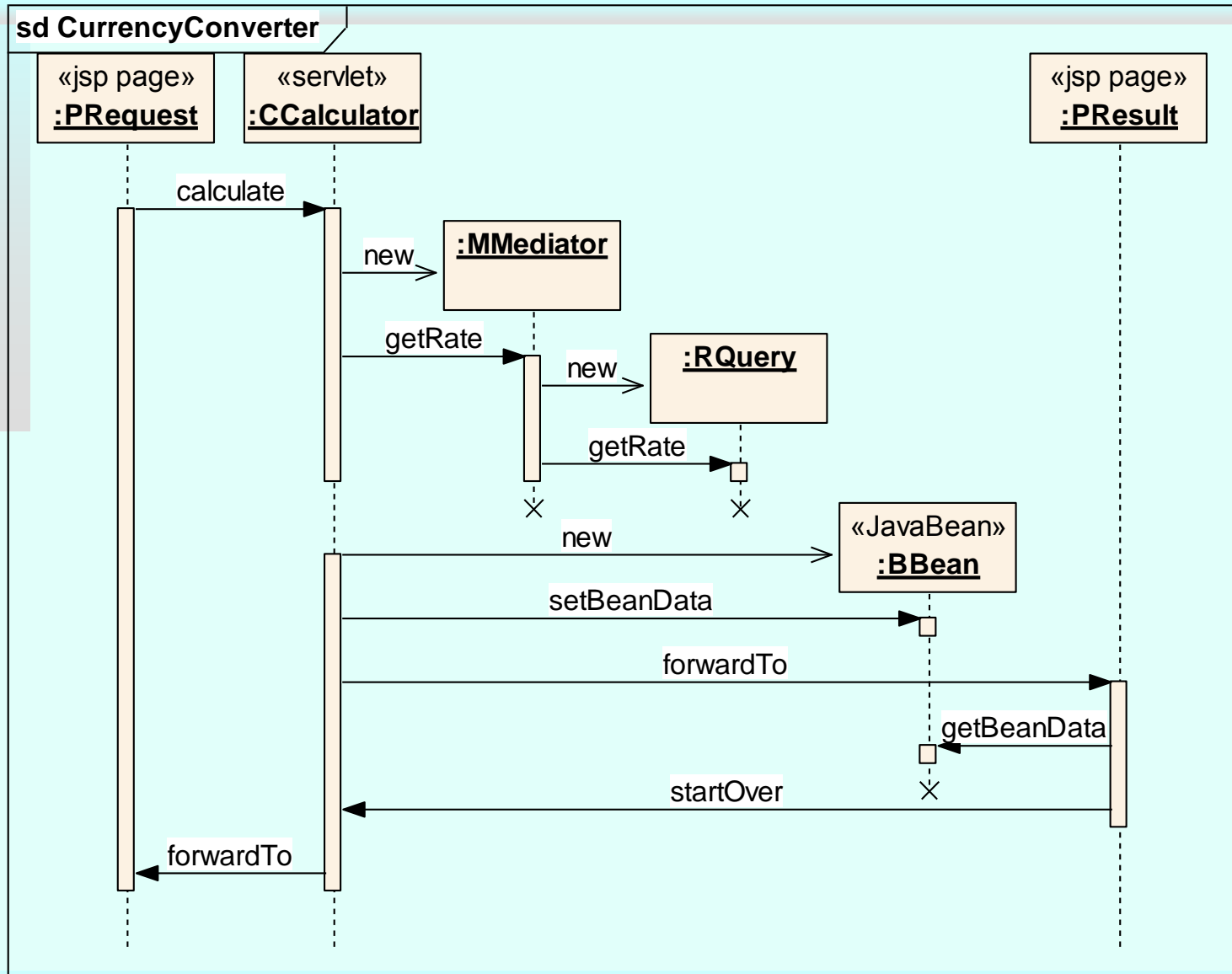
Interactions – more concepts

- the time when the flow of control is focused in an object is called in UML 2.0 an execution specification (previously called an *activation*)
- the return result from a message can be shown by a reply message or the return variable can be shown in the message syntax (e.g. `Class3 = doA(a)` or `int = doC()`)
- found message (doX) – represents a message whose sender is not specified
- object destruction is indicated on the diagram by the large X
- a lifeline represented by an interface (`:Interface1`) or an abstract class implies that the called method executes from a class that implements the interface or a concrete class that inherits from the abstract class

Accounting for basic technology

- A *servlet* is a Java program deployed and run inside a Web server.
 - Typically, a servlet does not have a graphical user interface (GUI) and, therefore, can cleanly belong to the Controller layer.
 - The GUI is supplied by the servlet's clients, such as a server page or an applet.
- *Java Server Pages* (JSP) are HTML pages with Java code pieces embedded in them.
 - JSPs belong to the Presentation layer.
- *JavaBeans* are Java classes that can store data and follow predefined rules allowing to `get()` and `set()` the data.
 - Java provides a mechanism on beans that allows JSP form values to be automatically loaded/unloaded from/into beans.
 - JavaBeans reside in the Bean layer.

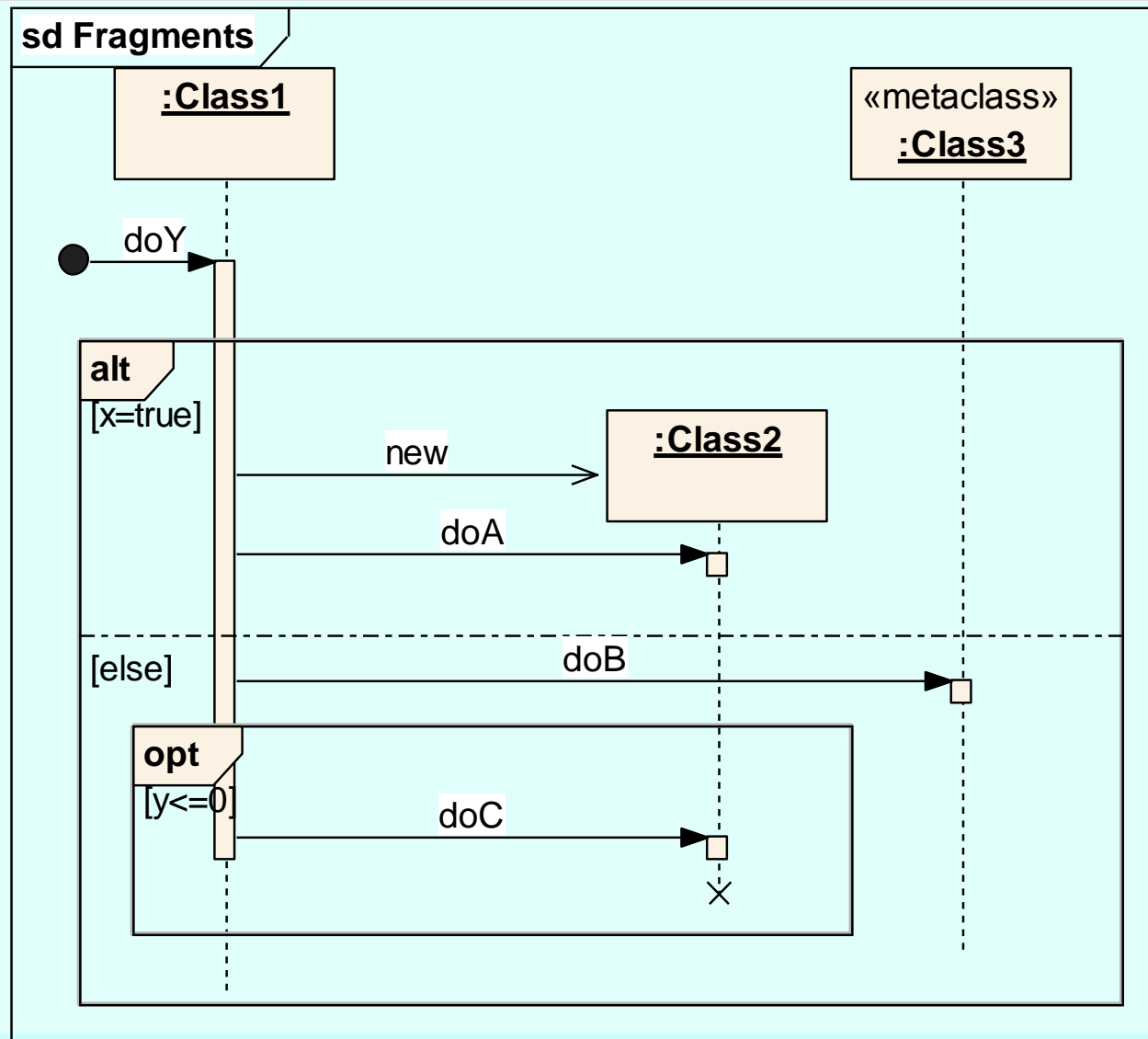
Sequence diagram using JSP/servlet/JavaBean



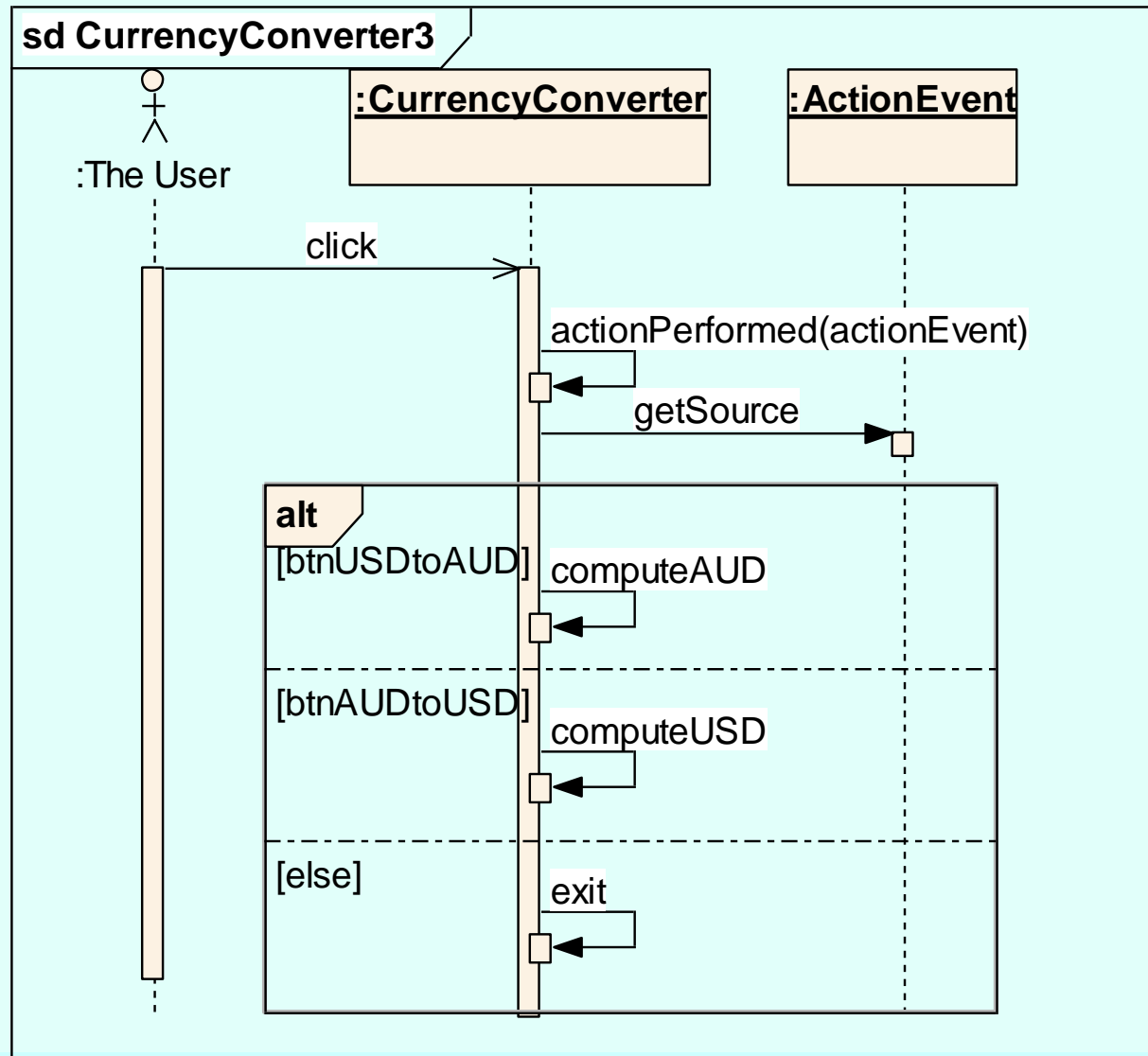
Fragments

- A piece of interaction is called an **interaction fragment**.
- Interactions can contain smaller interaction fragments called **combined fragments**.
- The semantics of a combined fragment is determined by the **interaction operator**
 - `alt` – alternative fragment for if-then-else conditional logic expressed in guard conditions,
 - `opt` – option fragment that executes if guard condition is true,
 - `loop` – loop fragment that is repeated many times subject to the loop condition,
 - `break` – break fragment that executes instead of the rest of the enclosing fragment if the break condition is true,
 - `parallel` – a parallel fragment that allows for interleaved execution of contained behaviors.

Fragments - notation

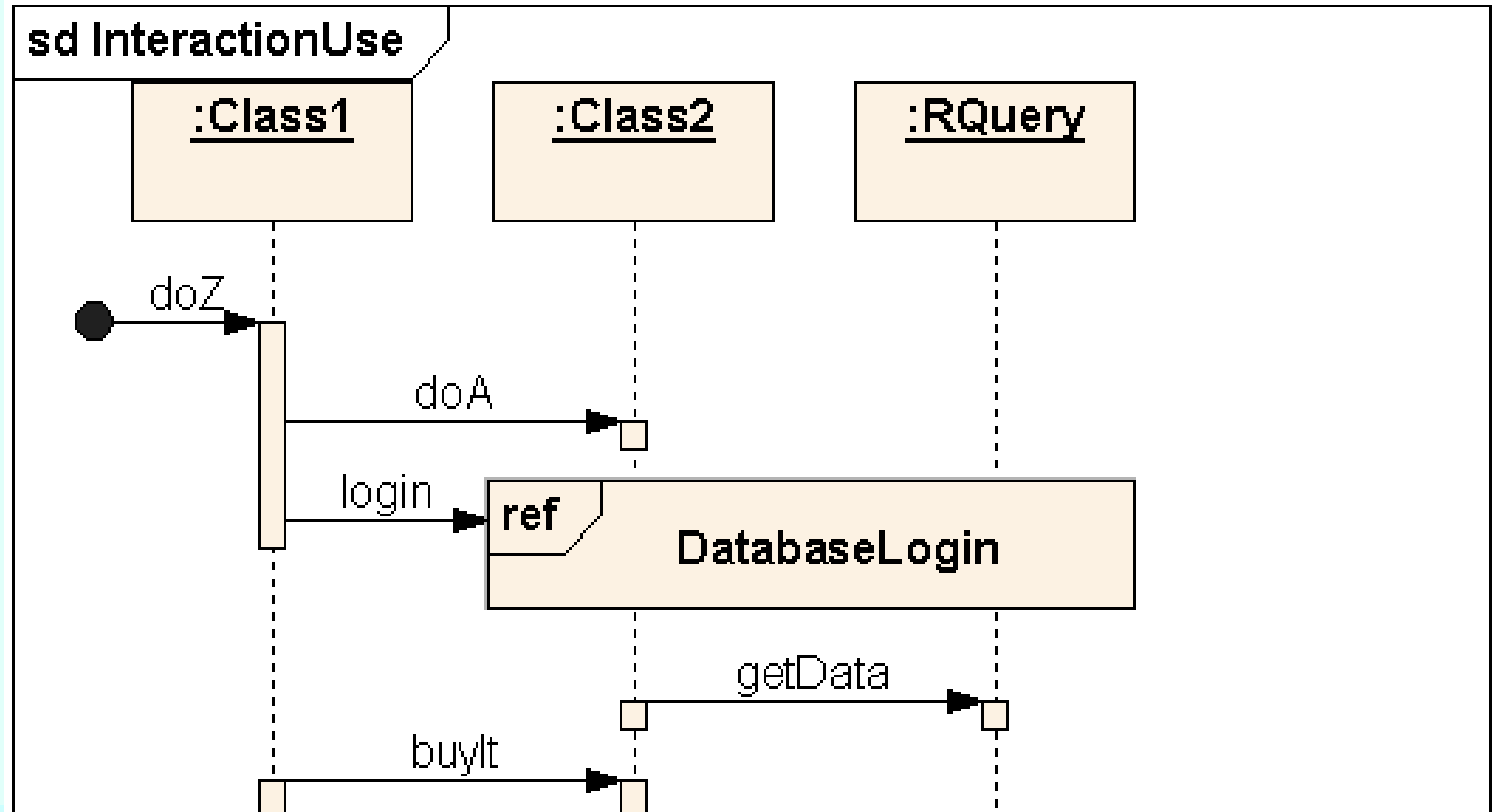


Fragments - example



Interaction uses

- A reference to an interaction from an enclosing interaction is called an ***interaction use***.
 - The enclosing interaction is labeled with the tag `sd` (sequence diagrams).
 - Interaction use is labeled with the tag `ref` (reference) and it refers to another `sd` interaction.



Review Quiz 4.4

1. What interaction modeling concept needs to be used to specify multi-threaded execution?
2. What interaction modeling concept needs to be used to specify interaction from unknown sender?
3. What architectural layer does a servlet belong to?
4. What tag is used to label interaction use?