

Object Oriented Software Development (Recycling Machine)

- Marc Conrad
 - D104 (Park Square Building)
 - Marc.Conrad@beds.ac.uk
 - <http://perisic.com/marc>
- These slides:
 - Sequence Diagrams

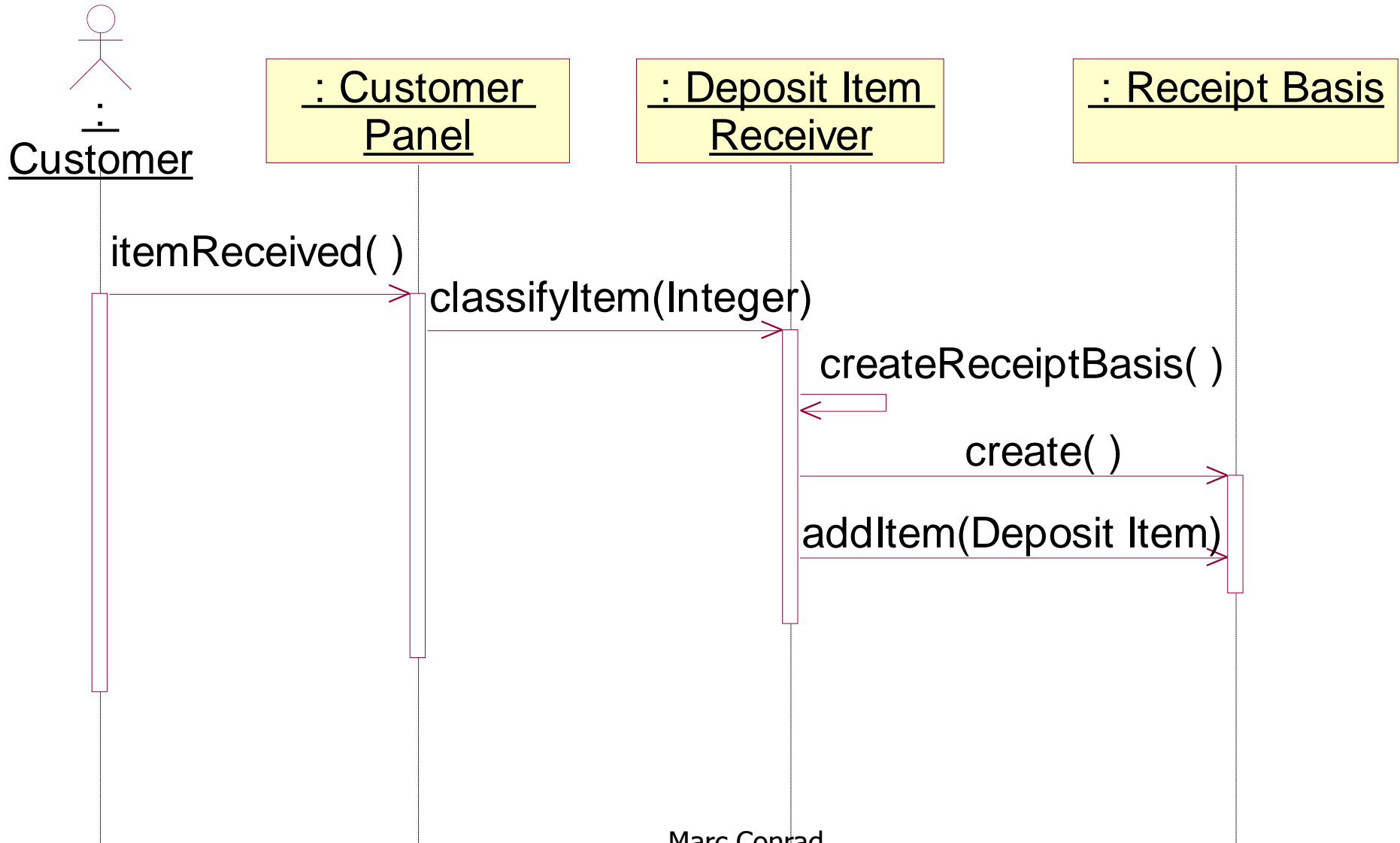
Interaction Diagrams

- The UML includes *interaction diagrams* to illustrate how objects interact via messages to fulfil tasks.
- A *sequence diagram* is a kind of an interaction diagram
- The next slides discuss mostly the *syntax* and *semantics* of sequence diagrams.
- Later topic: Strategies for good design (patterns).

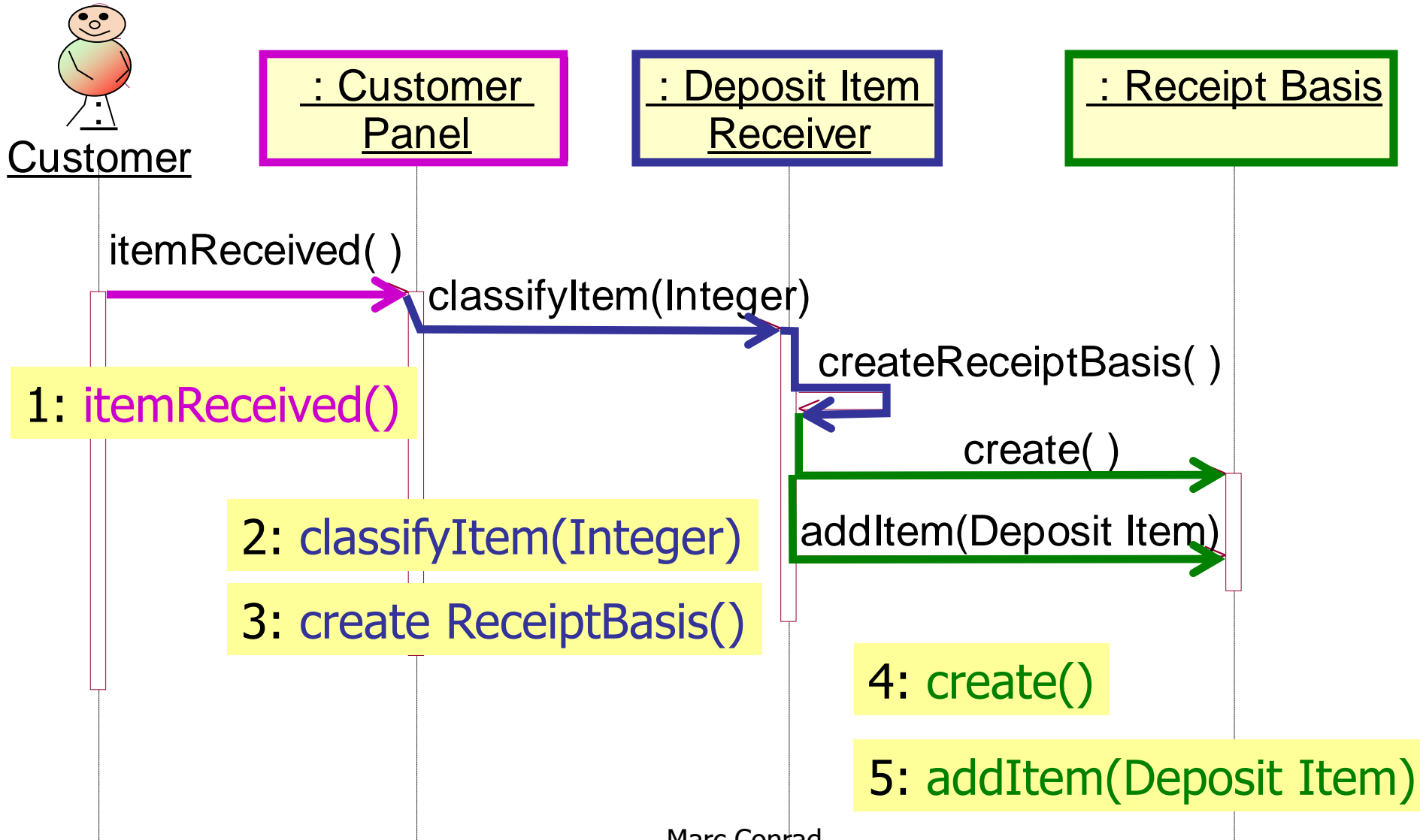
Interaction Diagrams - when?

- Identify the system events that are implied by the use cases.
- Make at least one interaction diagram for each system event.
- Make additional interaction diagrams for alternative courses of events.
- Example (Recycling machine, use case "enter item"):
 - System events: enter item, print receipt.
 - Enter item has two alternative courses:
 - Enter first item
 - Enter next items
 - Therefore **three** interaction diagrams for the "enter item" use case.

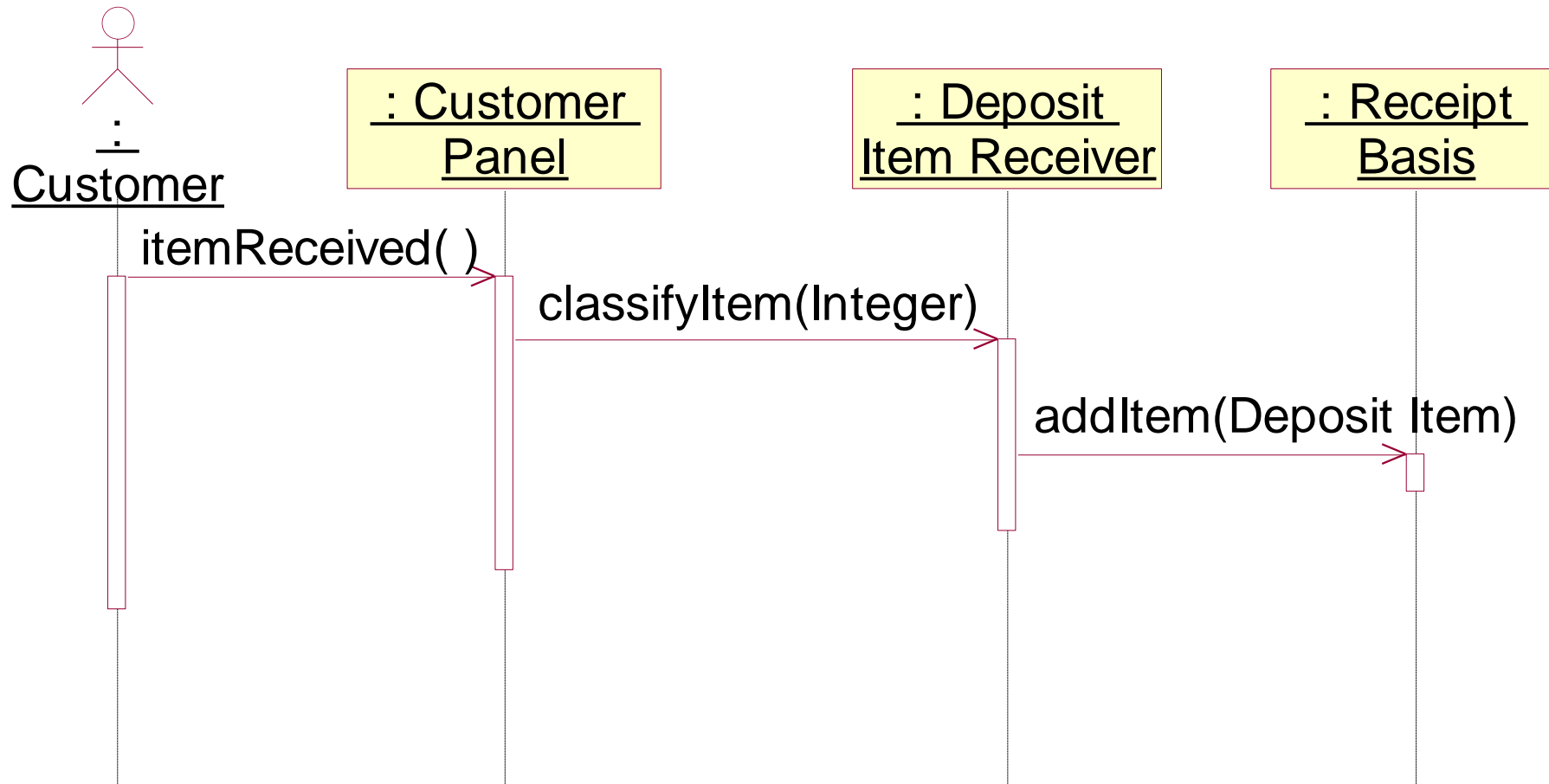
Sequence Diagram: Insert first item.



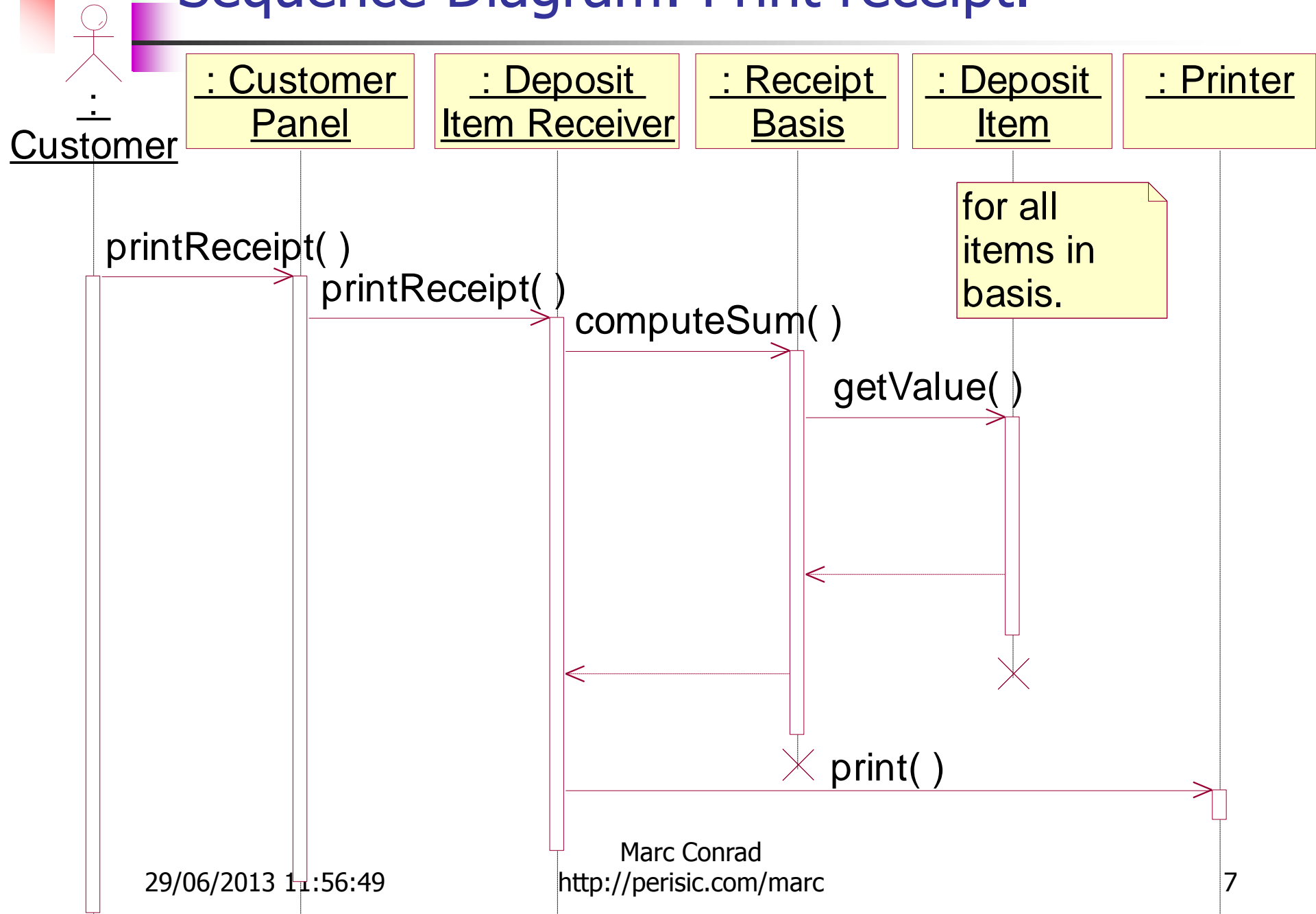
Flow of messages



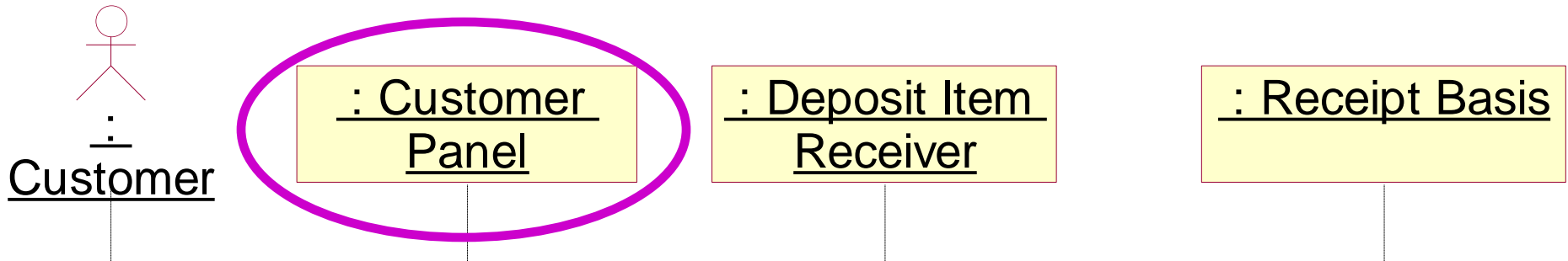
Sequence Diagram: Insert next items.



Sequence Diagram: Print receipt.



Syntax - classes and objects



- The following notation is used in the UML for classes and objects:

- Class:

Person

- Instance of a Class
(object without a name):

:Person

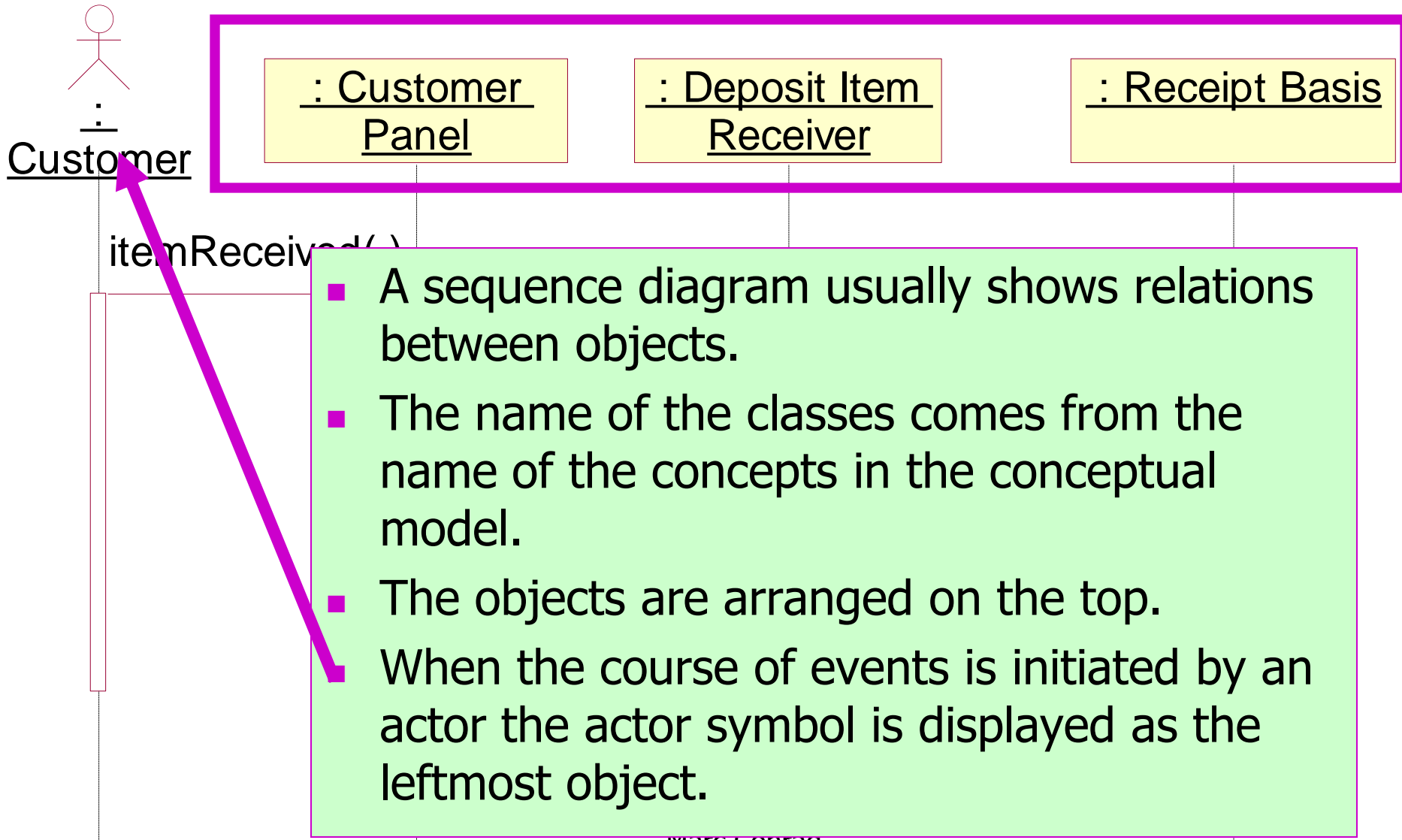
- Named instance of a class
(named object):

fabian:Person

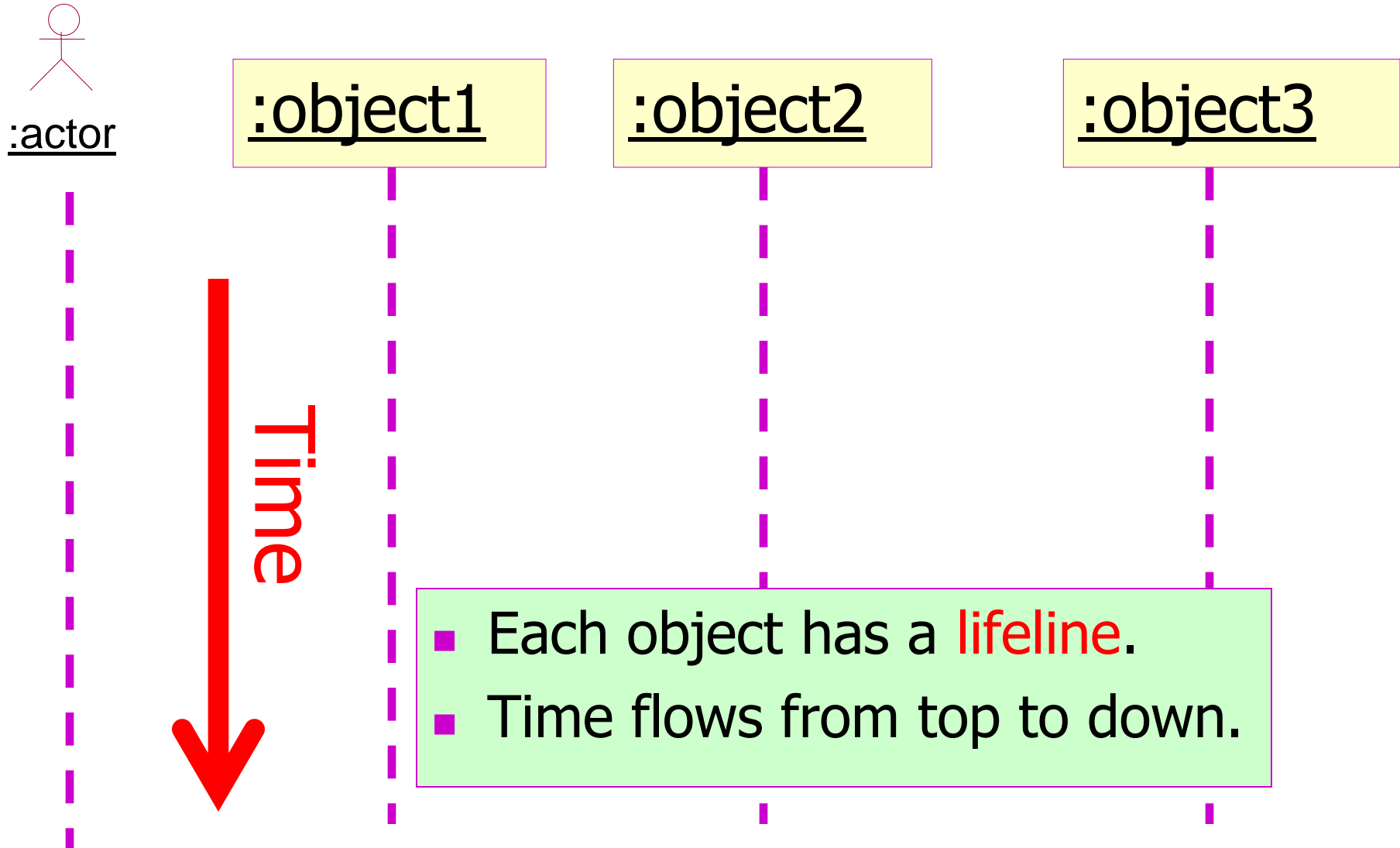
- Named object only
(shown without class):

fabian

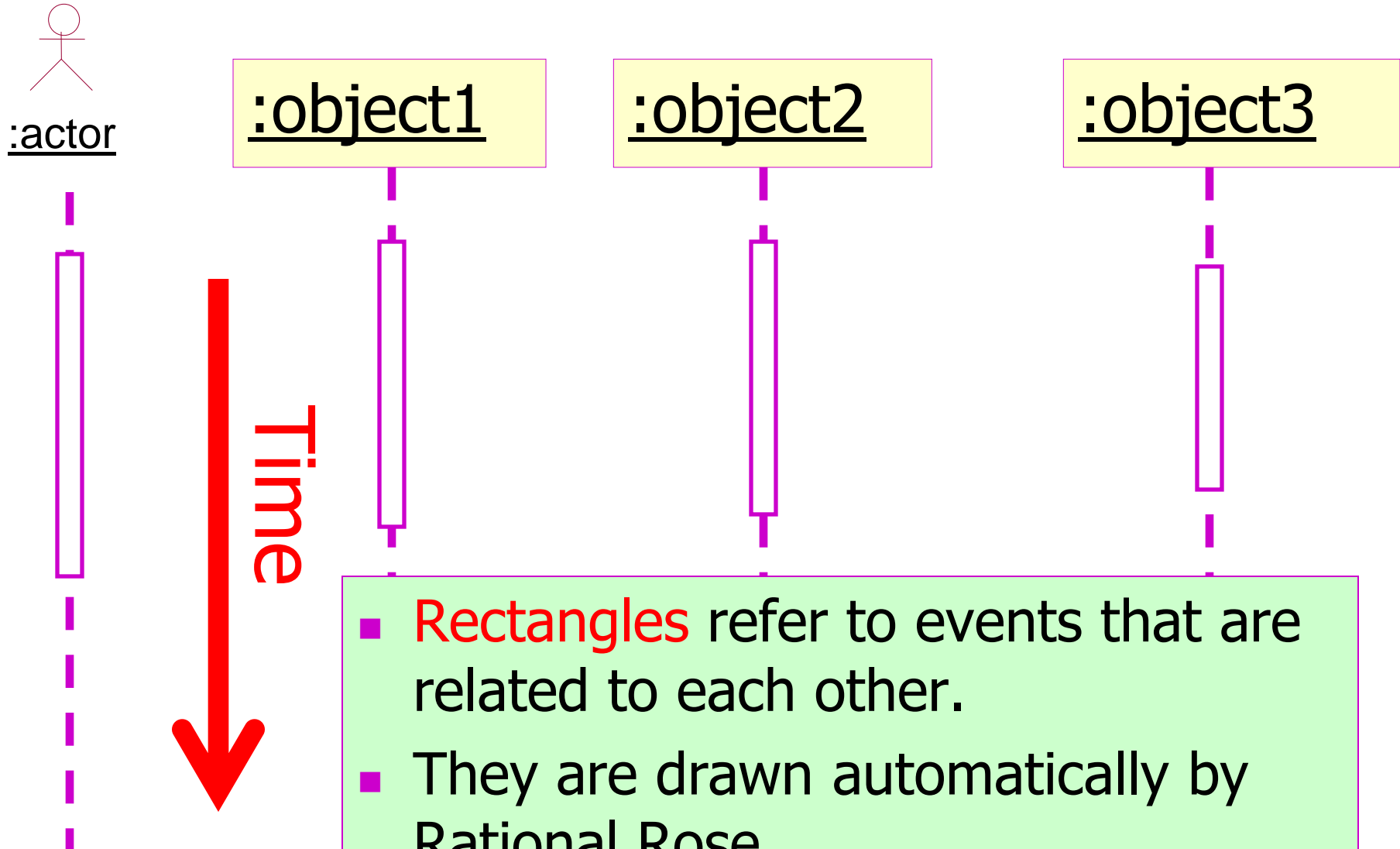
Sequence Diagram: Objects



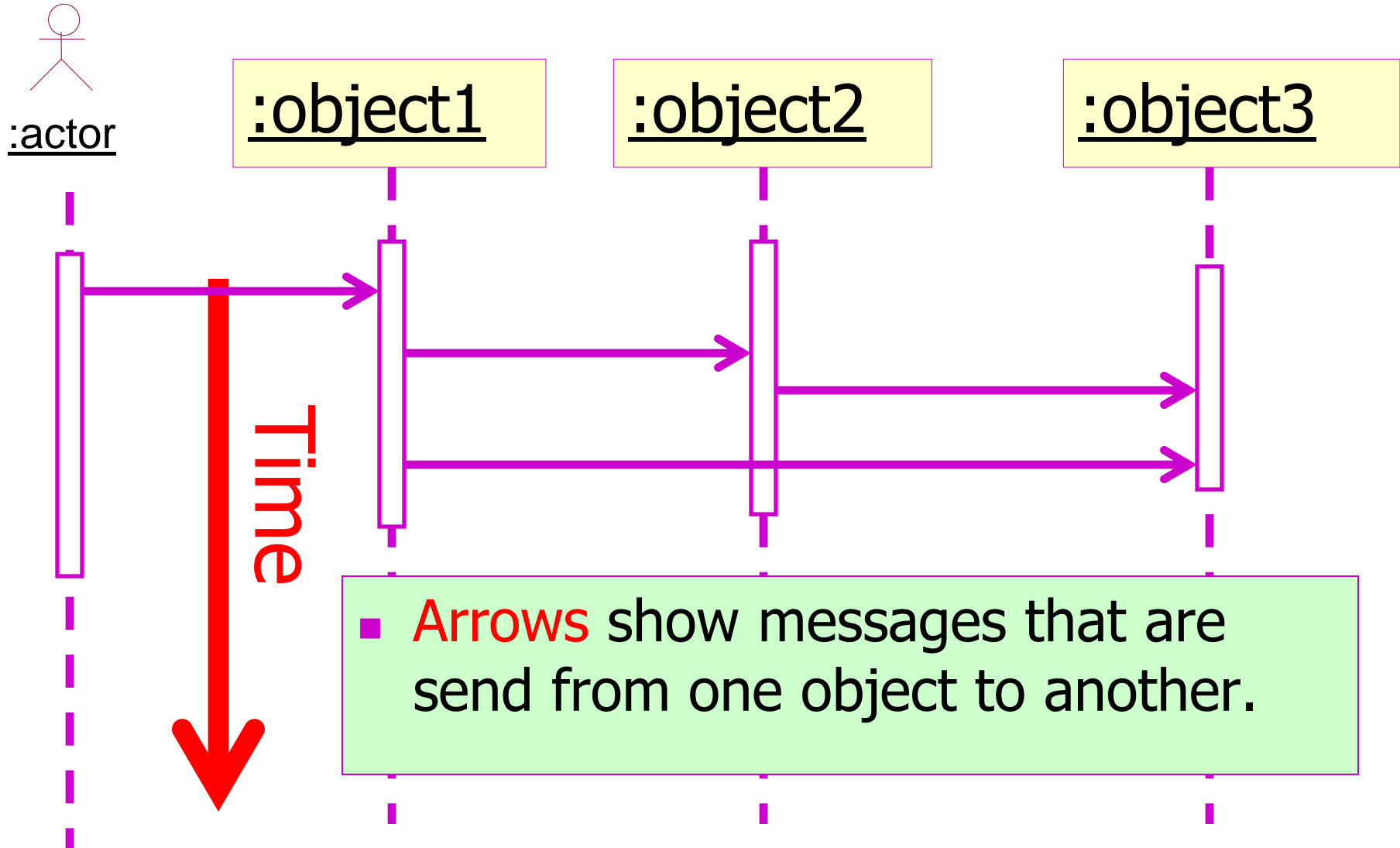
Sequence Diagram: Lifelines



Sequence Diagram: Rectangles.

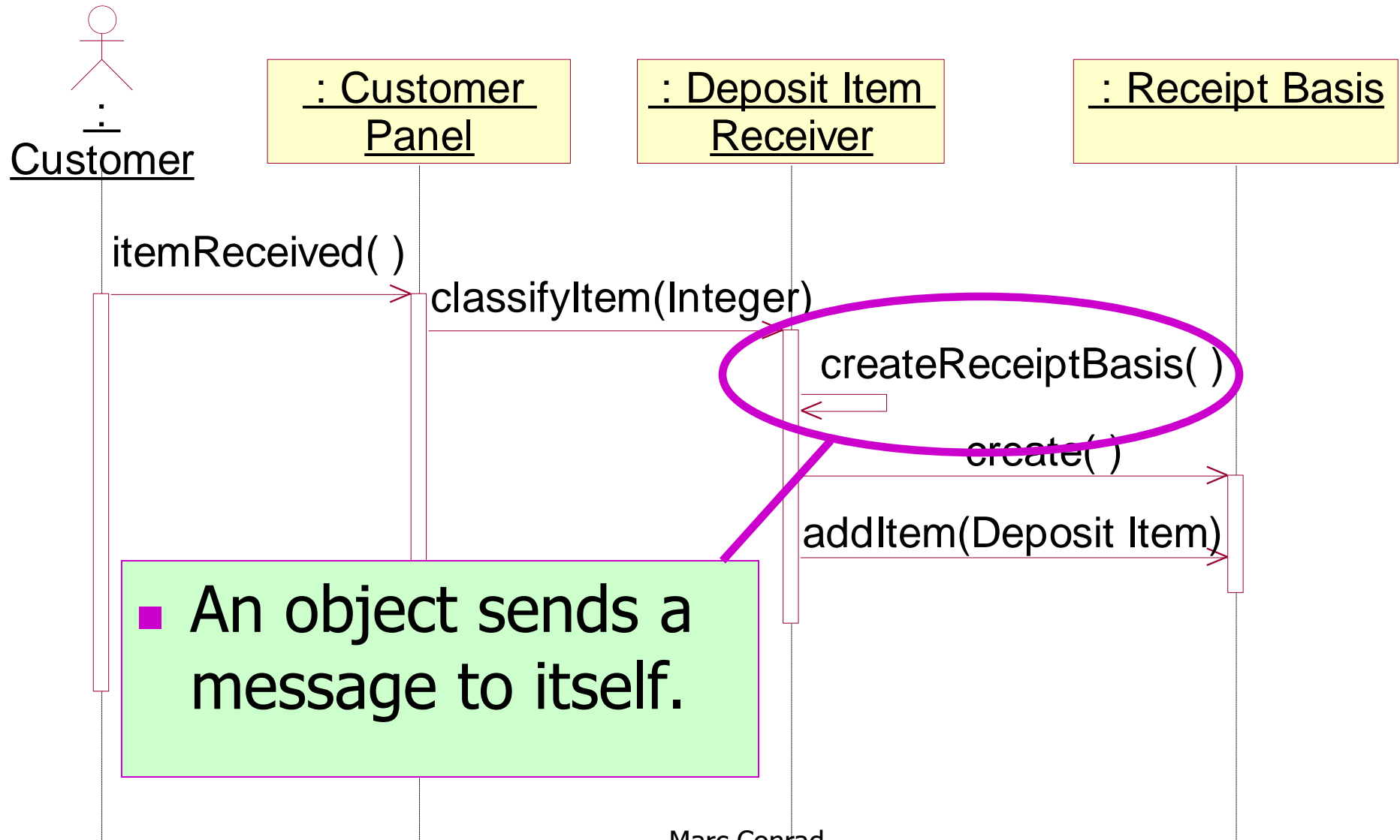


Sequence Diagram: Messages.



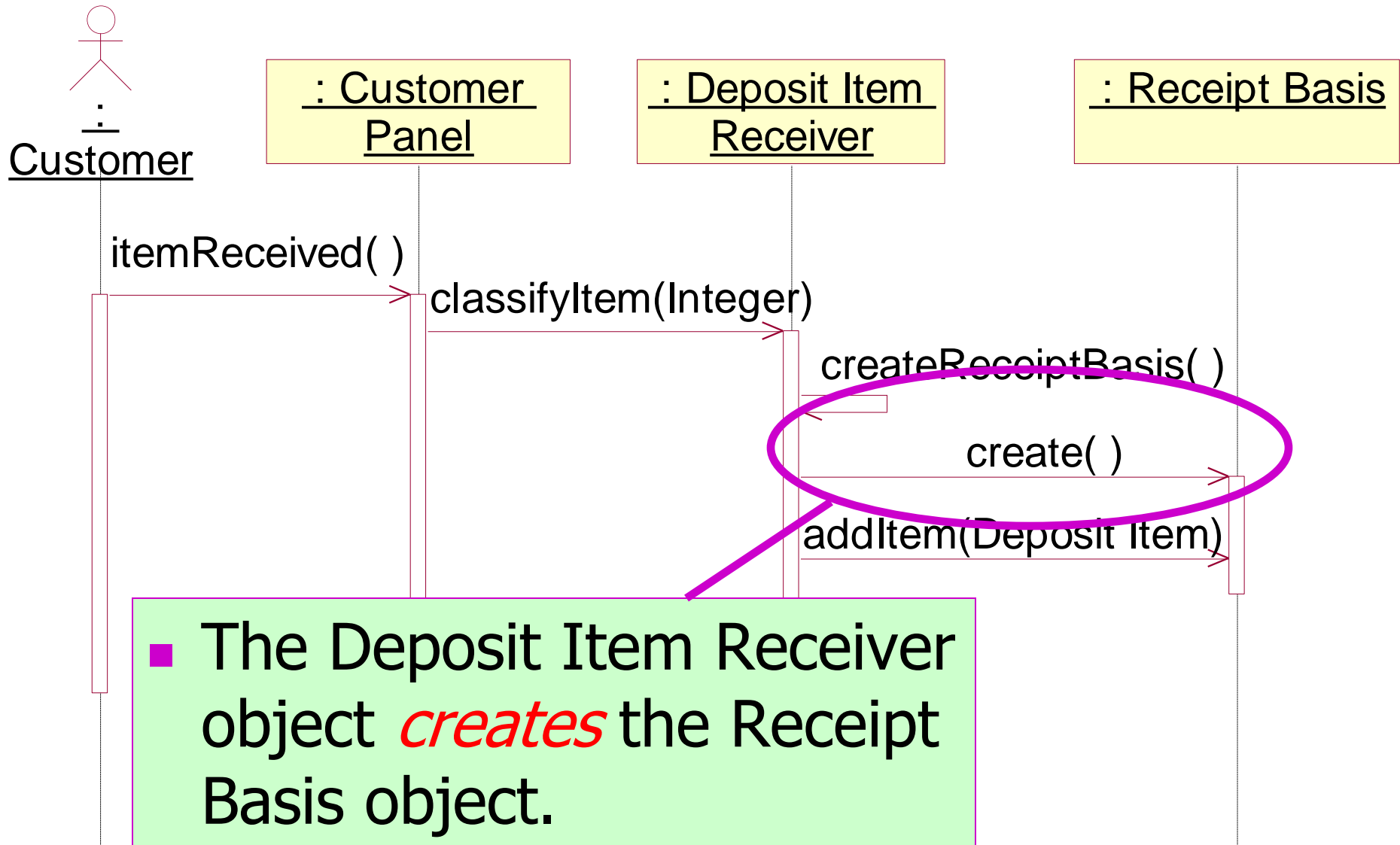
■ **Arrows** show messages that are send from one object to another.

Sequence Diagram: Message to self



- An object sends a message to itself.

Sequence Diagram: Creation of Objects



Other Syntax Elements

