Object Oriented Analysis & Design
面向对象分析与设计

Lecture_03 面向对象分析（一）

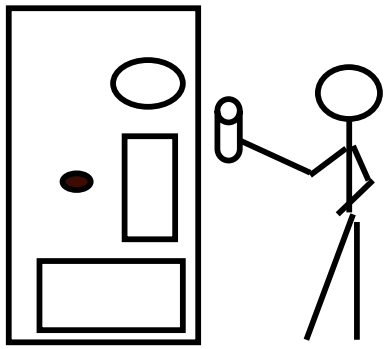
主讲: 姜宁康 博士

3、面向对象分析方法（一）名词法案例

- 通过案例
  - 帮助同学们理解概念模型、概念类、学会定义概念类

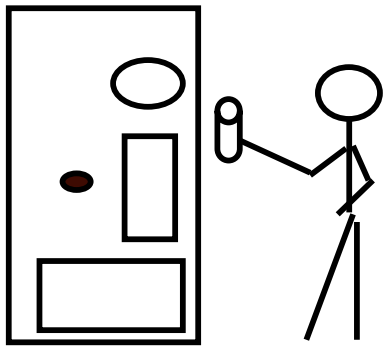
1、名词法定义概念类：废品回收机

- 用例 Use case：The Recycling machine 废品回收机



- The system controls a recycling machine for returnable bottles, cans and crates(板条箱). The machine can be used by several customers at the same time and each customer can return all three types of item on the same occasion. The system has to check, for each item, what type has been returned.
- The system will register how many items each customer returns and when the customer asks for a receipt, the system will print out what was deposited , the value of the returned items and the total return sum that will be paid to the customer.

1、名词法定义概念类：废品回收机



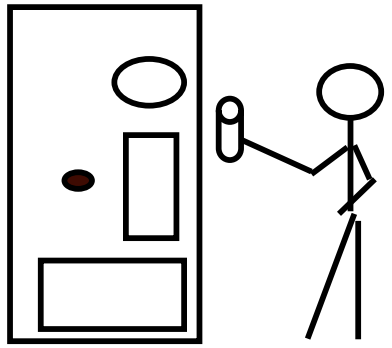
- 用例 Use case：The Recycling machine - The “return item”

- The system controls a recycling machine for returnable bottles, cans and crates. The machine can be used by several customers at the same time and each customer can return all three types of item on the same occasion. The system has to check, for each item, what type has been returned.

- The system will register how many items each customer returns and when the customer asks for a receipt, the system will print out what was deposited , the value of the returned items and the total return sum that will be paid to the customer.

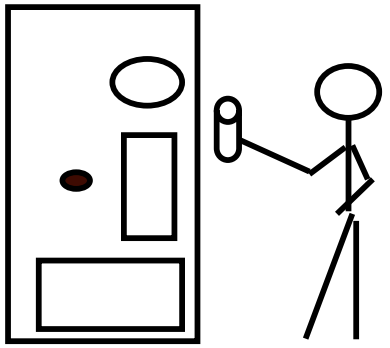
1、名词法定义概念类：废品回收机

- recycling machine
- bottles, cans, and crates
- machine
- customers, customer
- types of item, item, type, returned items
- system
- receipt
- return sum



2.1 讨论 “recycling machine”

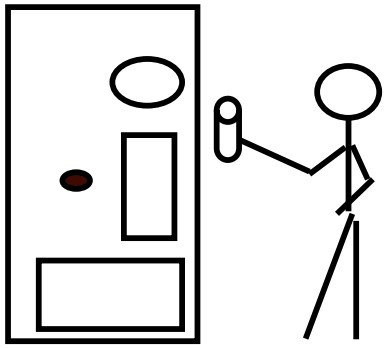
- recycling machine
- bottles, cans and crates
- machine
- customers, customer
- types items
- syster
- receip
- returr



- This concept is the “overall system”. As we consider only one single use case, it is better to name this concept in the context of this use case, e.g.
  - Deposit item receiver

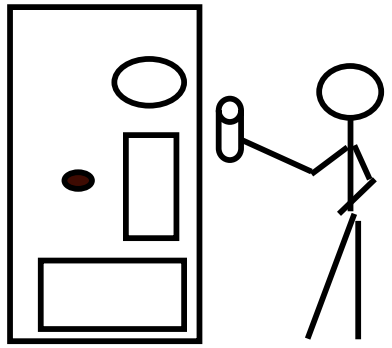
2.2 讨论"bottles, cans, and crates"

- deposit item receiver
- bottles, cans, and crates
- machine
- customers, customer
- types of items
- system
- receipt
- return su



- In a conceptual model it is usually better to use singular and multiplicities instead of plural.
- As bottle, can and crate have much in common (they are processed as items), they could be generalised to an "item".
- We should remember this for later (inheritance).

2.3 讨论 “machine” and “system”

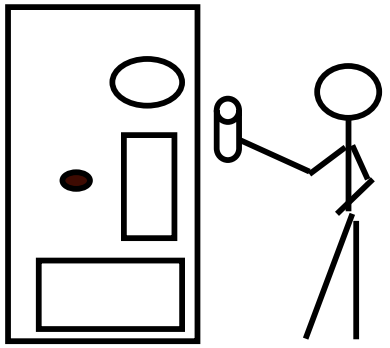


- deposit item receiver
- bottle, can, crate
- machine
- customers, customer
- types of item, ite
- items
- system
- receipt
- return sum

- “Machine” and “System” mean here the same, namely the “Recycling machine”, i.e. the
  - Deposit item receiver

2.4 讨论 “customers” and “customer”

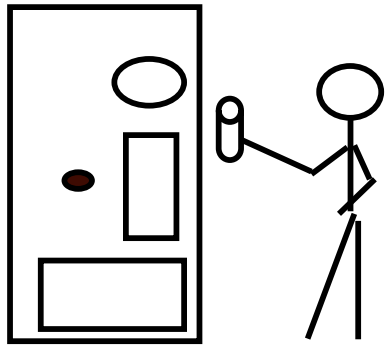
- deposit item receiver
- bottle, can, crate
-
- customers, customer
- types of item, item, items
-
- receipt
- return sum



- The customer has already been identified as an actor.
- They are outside of the system.
- We establish a concept, that interfaces with the customer (and is inside the system):
  - Customer panel
  - (Note: this is a fabled (虚构的) idea)

2.5 讨论 “item” (etc.)

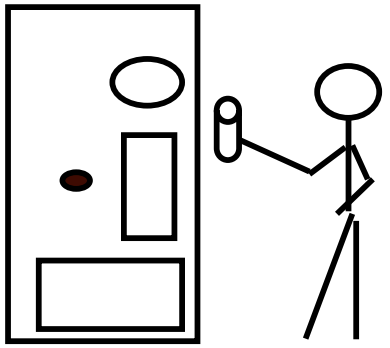
- deposit item receiver
- bottle, can, crate
-
- customer panel
- types of item, item, type, returned items
-
- receipt
- return sum



- The items that are inserted in the machine.
- Good candidate as superclass for bottle, can, crate.
- Let’s call it
  - Deposit item, (instead of ‘item’)

2.6 讨论 “receipt”

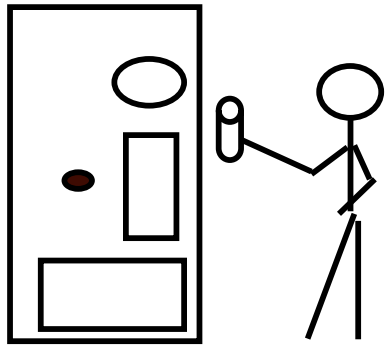
- deposit item receiver
- bottle, can, crate
-
- customer panel
- deposit item
-
- receipt
- return sum



- The concept that “remembers” all items inserted in the machine.
- To distinguish it from the piece of paper returned to the customer, call it
  - Receipt basis

2.7 讨论 “return sum”

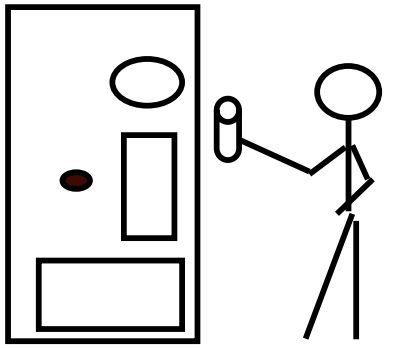
- deposit item receiver
- bottle, can, crate
-
- customer panel
- deposit item
-
- receipt basis
- return sum



- The sum that it is returned to the customer is actually computed by adding up all values of the items stored in the receipt basis.
- The sum itself is only a primitive data value, and may therefore not be considered as a concept.

2.8 讨论 其他概念

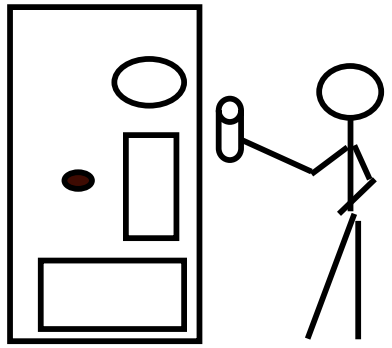
- deposit item receiver
- bottle, can, crate
-
- customer panel
- deposit item
-
- receipt basis
-



- These are the concepts identified by nouns. Did we forget something?
- Check the “Concept Category List” !
- The system “interfaces” with the physical object “printer”, so we add an interface concept
  - Receipt printer

2.9 最终找到的概念类

- deposit item receiver
- bottle, can, crate
-
- customer panel
- deposit item
-
- receipt basis
-
- receipt printer

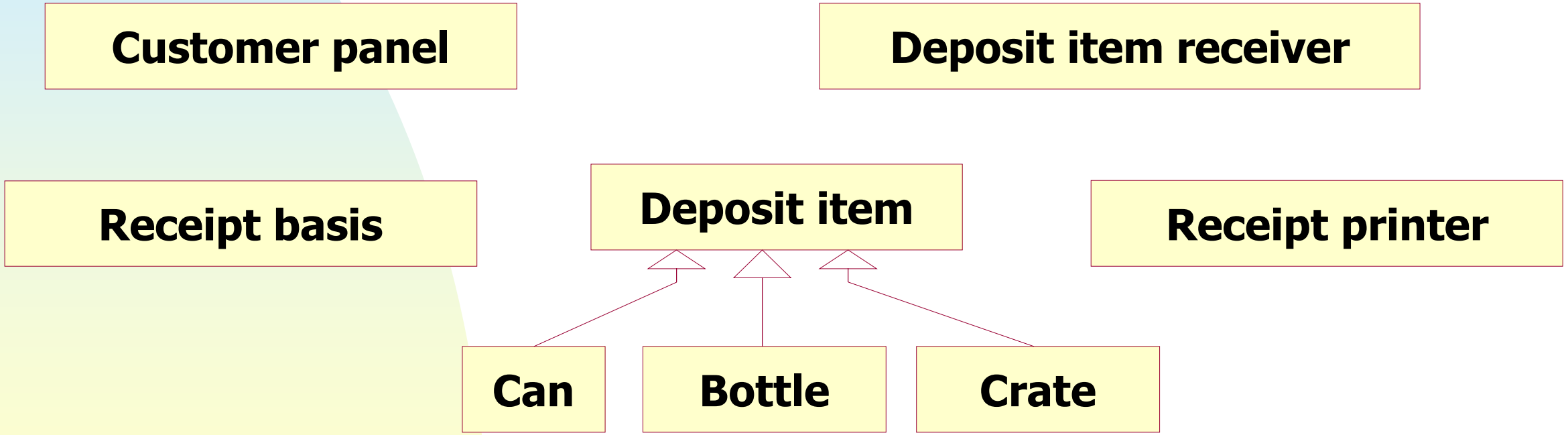


- So far we have identified:
  - Concepts
  - A generalisation relationship.
- Next step: Finding associations.

3 构建概念模型 conceptual model.

- Find the concepts
- Draw them in a conceptual model
- Add associations
- Add attributes

3.1 画出概念 Drawing of Concepts



3.2 添加关系 Adding Associations

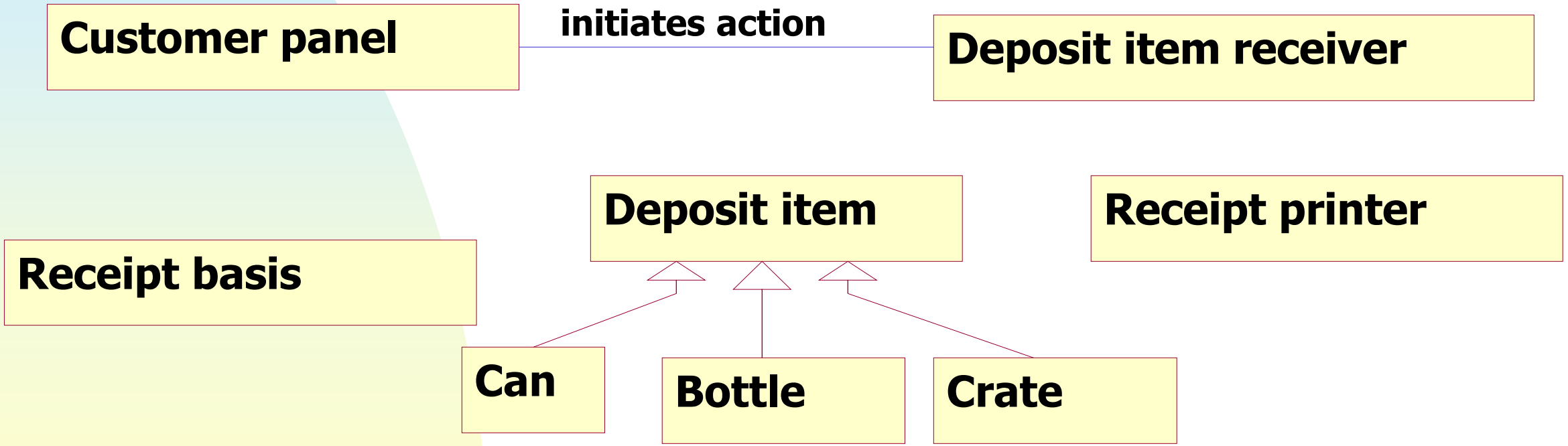
- **If one concept needs to know of another concept for some duration, they should be linked by an association.**
  - Add "Adornment" to association 增加关联关系的一些修饰
- **Also use the following list in order to identify associations.**

- A is a part of B
- A is contained in B
- A is a description for B
- A is a line item of B
- A is known or reported in B
- A is a member of B
- A is a organisational subunit of B

- A uses or manages B
- A communicates with B
- A is related to a transaction B
- A is a transaction related to another transaction B
- A is next to B
- A is owned by B

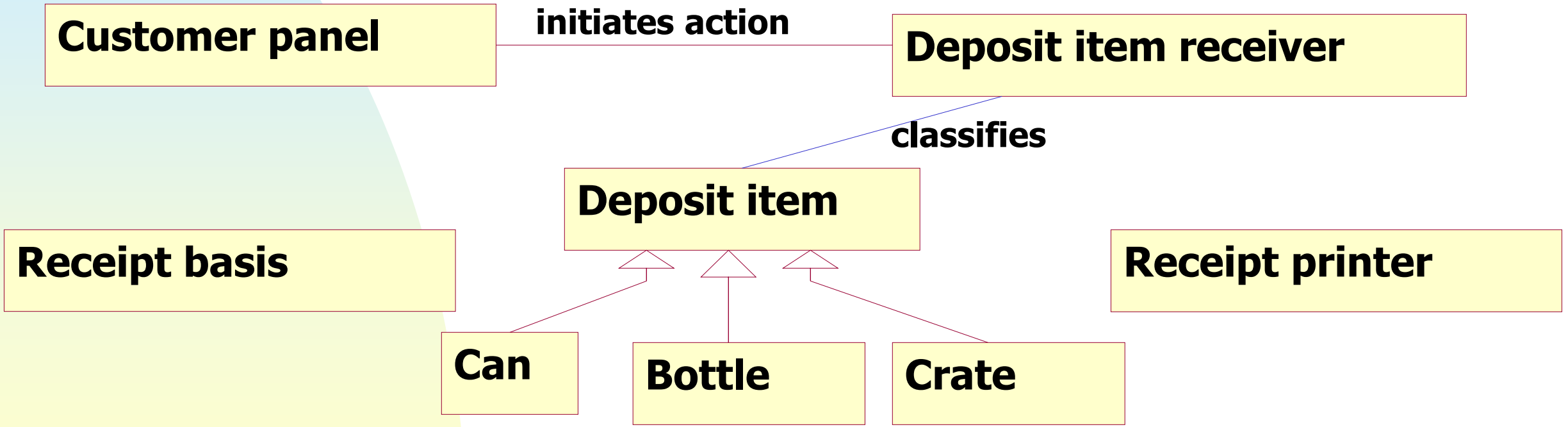
3.2 添加关系 Adding Associations

- The Customer panel communicates to the receiver when an item is inserted.
- Also when the receipt is requested.



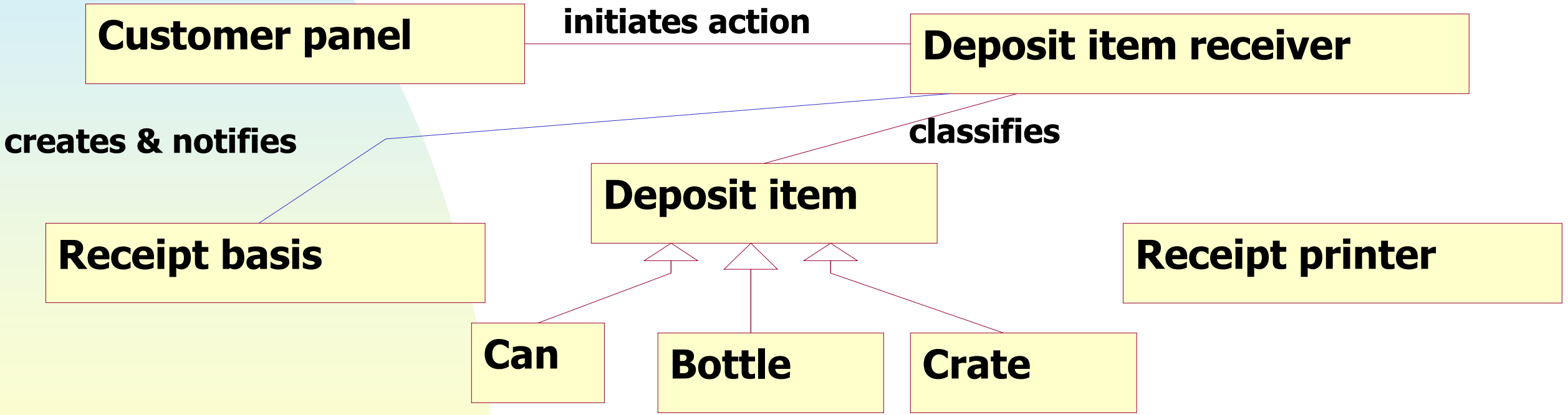
3.2 添加关系 Adding Associations

- **The Deposit item receiver manages Deposit items:**
  - **The items are classified.**



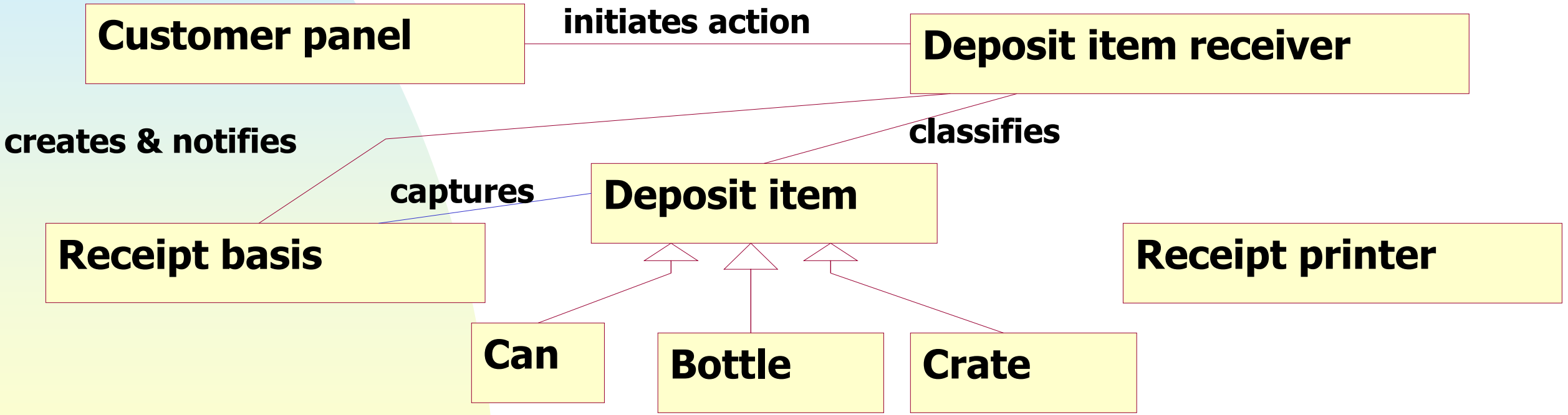
3.2 添加关系 Adding Associations

- The Deposit item receiver communicates to Receipt basis:
- Items received and classified are stored.
- It also creates the receipt basis when it is needed for the first time.



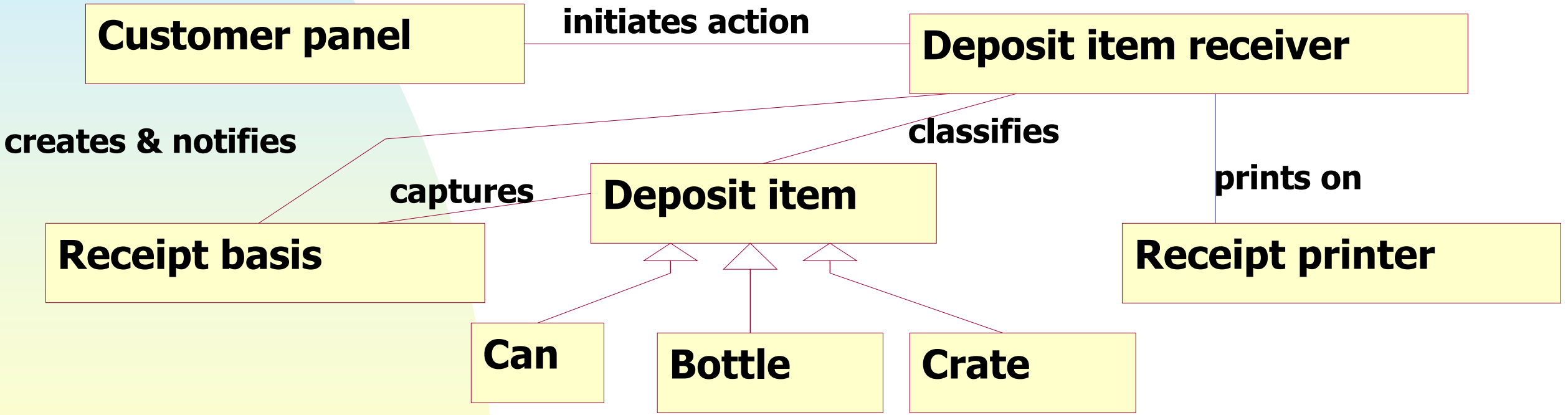
3.2 添加关系 Adding Associations

- The Receipt basis collects Deposit items.



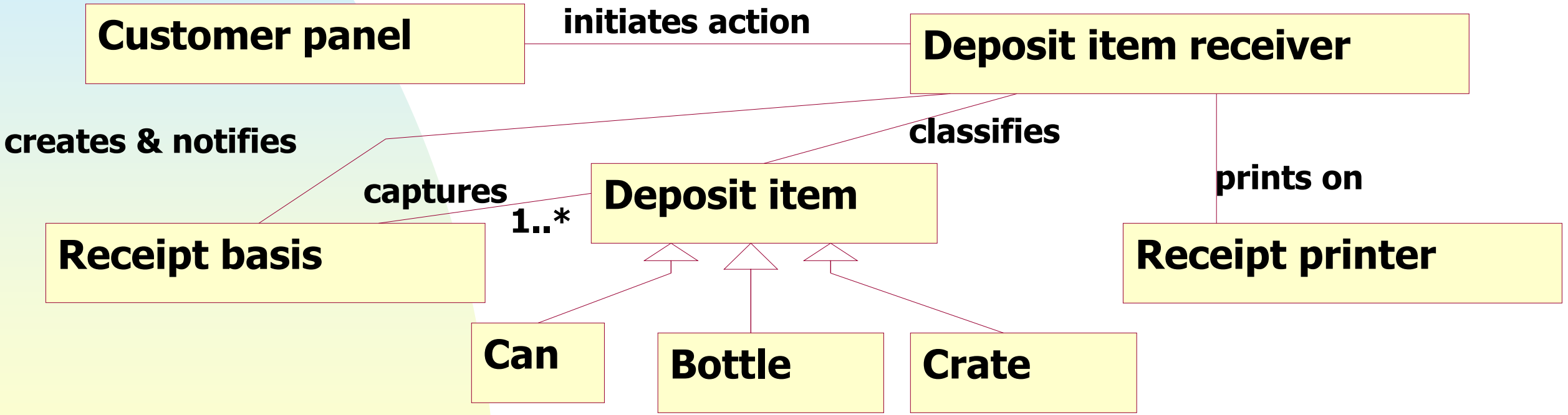
3.2 添加关系 Adding Associations

- On request by the Customer Panel, the Deposit item receiver initiates printing of a receipt on the printer.



3.2 添加关系 Adding Associations

- Adding multiplicities（多重性）
- Only one association here is a 1 to many relationship
- All others are 1 to 1.

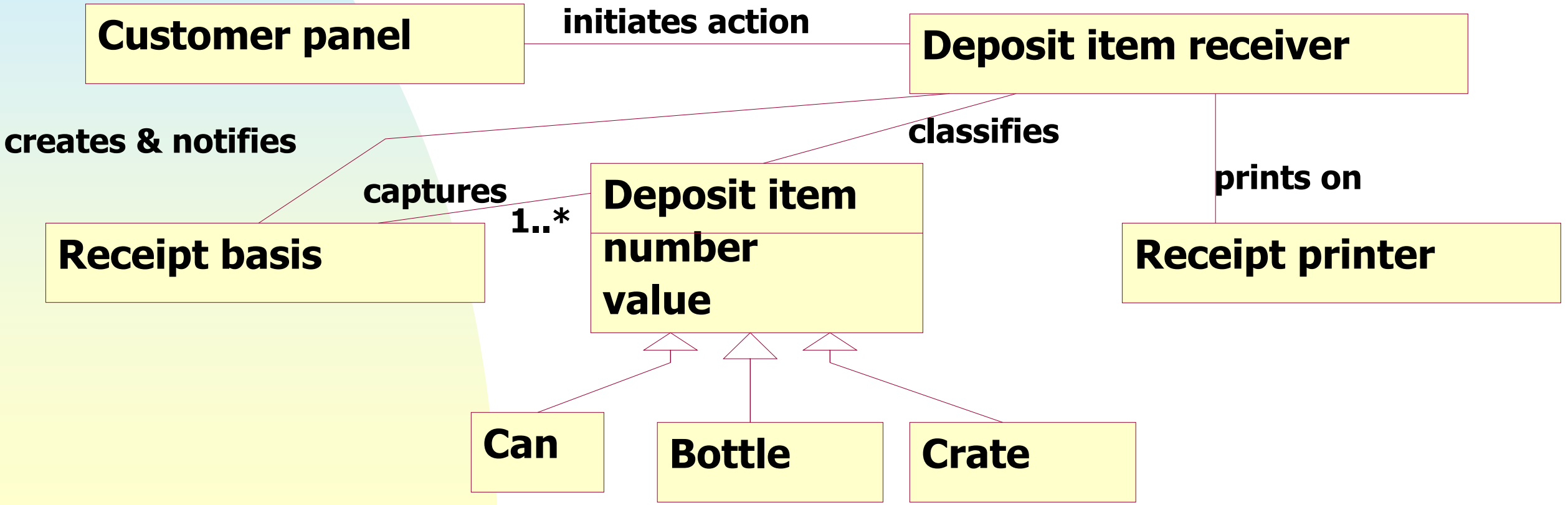


3.3 添加属性 Adding Attributes

- An attribute is a logical data value of an object.
- Attributes in a conceptual model are simple data values as
  - Boolean, Date, Number, String (Text), Time, Address, Colour, Price, Phone Numbers, Product Codes, etc.
- Sometimes it is difficult to distinguish between attributed and concepts 有时比较难以区分一个概念，究竟是作为概念类还是作为其他概念的属性
  - 比如，员工 && 地址?

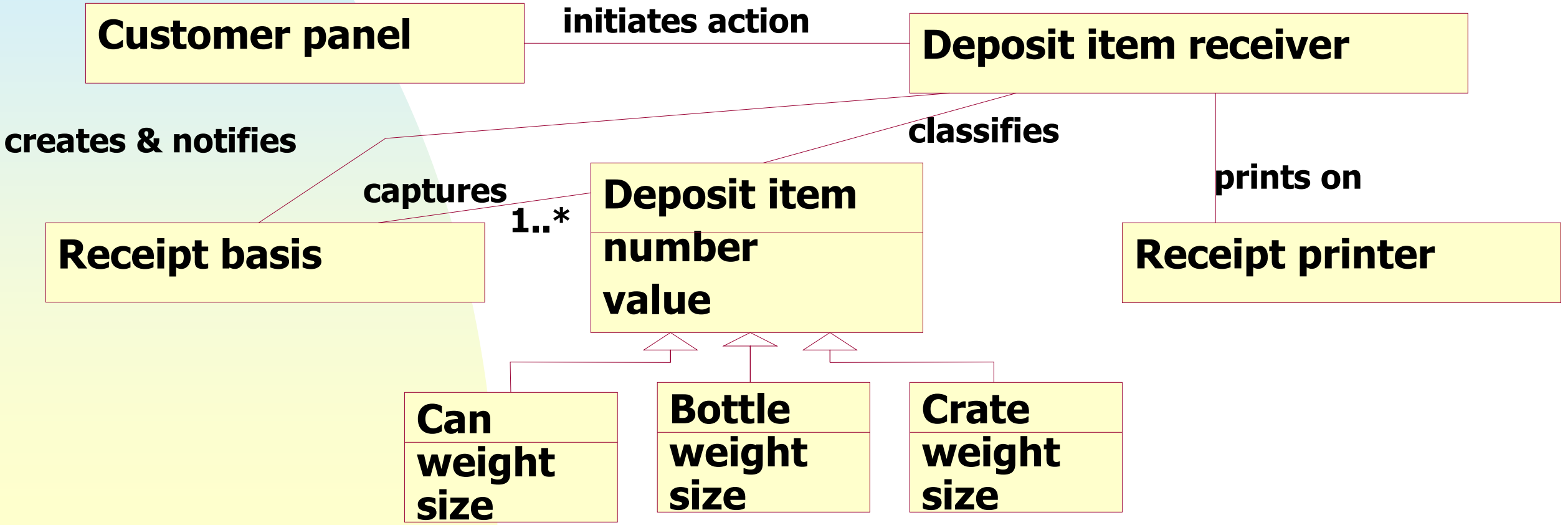
3.3 添加属性 Adding Attributes

- The Deposit item has a value.
- Also it will be assigned a number that shows later on the receipt.



3.3 添加属性 Adding Attributes

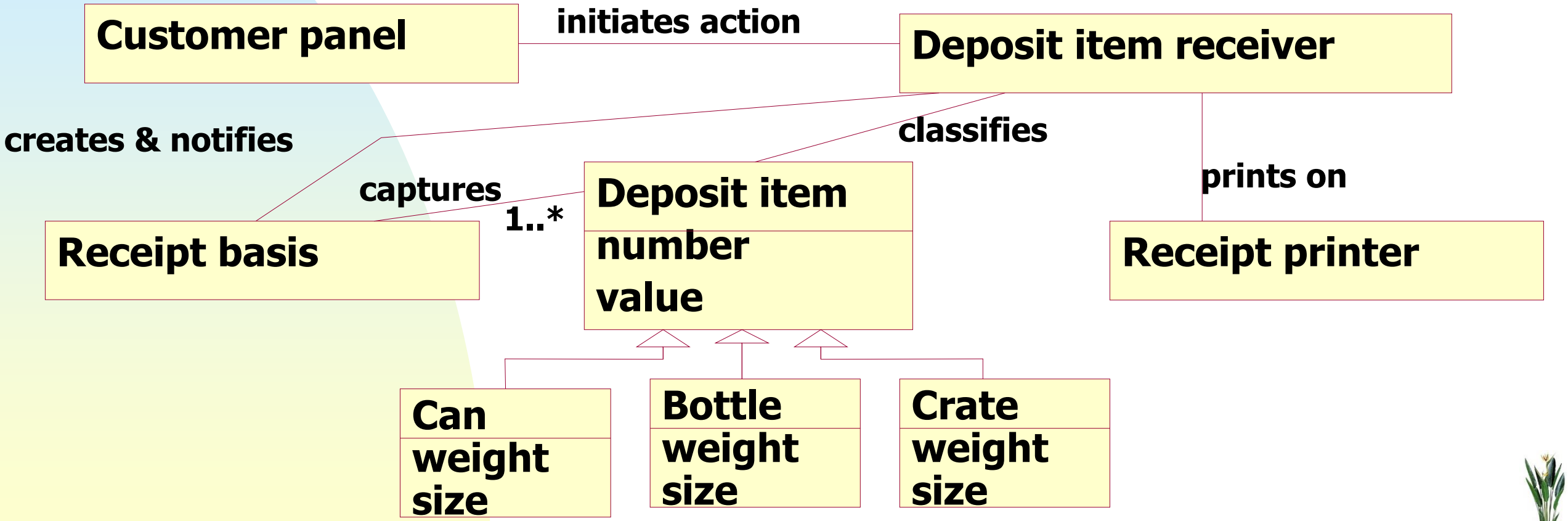
- In order to be classified by the Deposit item receiver each item has also a weight and a size.
- However this is the same for each type of item, but different between the types.



4 废品回收机的概念模型

- 我们得到了最后的结果

- NOT in a conceptual model:
  - methods, operations, functions



- 本讲结束