Behavioral Patterns: State



Gerald BrittonIT Solutions designer

@GeraldBritton www.linkedin.com/in/geraldbritton



Overview



Shopping cart

Supermarket, eStore

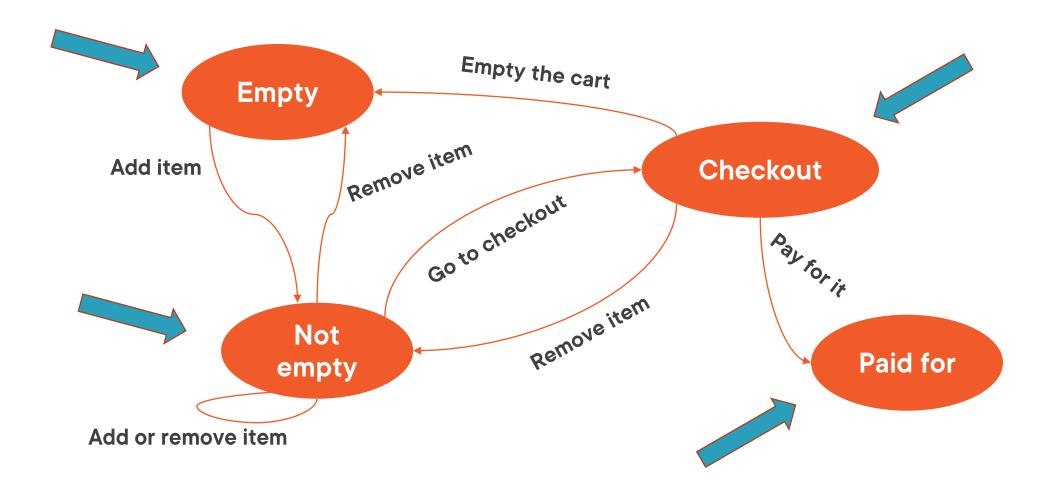
Various states

- Empty
- Containing some items
- At the checkout
- Paid for

Transitions:

- Adding and removing items
- Checking out
- Paying for your purchases

Shopping Cart State Diagram





Demo



Model the shopping cart

Use one variable to track the state

Create methods for state transitions

Run the model

See if you like the result

State

Classification: Behavioral

Operates in a particular context

Uses a class for each state

Transitions delegated to the state objects

Clients interface with the context

State Pattern Structure



Demo



Implement the State Pattern

Create a shopping cart context

Create state classes

Add transition handles

Make sure it still works

Consequences

Encapsulates state-specific behavior

Distributes behavior across state classes

Makes state transitions explicit

State objects can be shared

Flexible transition definitions

Can create states at transition time

Summary



When is the State Pattern applicable?
When object's behavior depends on state
Remove long if/elif/else statements
Similar in some ways to Strategy