Memento Pattern

CSCI-4448 - Boese

University of Colorado **Boulder**

Objectives

- Problem
- Definition
- Examples
- How
- Design Considerations

Problem

Problem

- Need to restore an object back to its previous state (e.g. "undo" or "rollback" operations).

Definition

Definition

“Without violating encapsulation, capture and externalize an object’s internal state so that the object can be restored to this state later.”

-Gang of Four

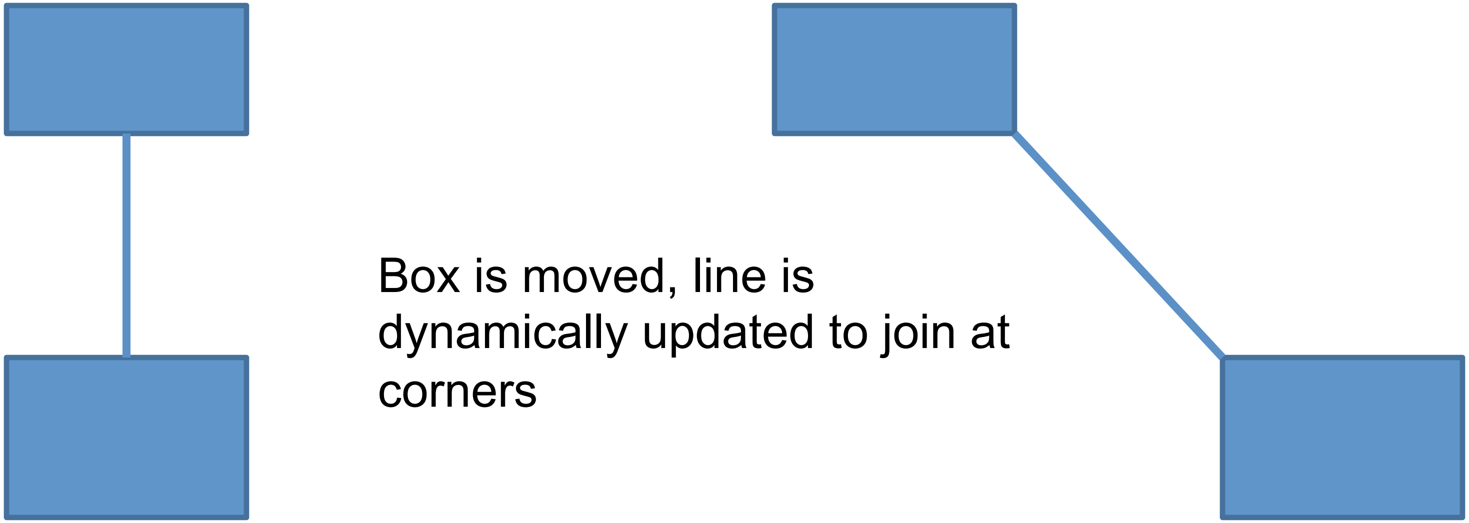
Definition

- **Name** “Memento”
  - A record of some object’s internal state, useful for future restoration
- **Intent**
  - Support undo operations via rollback
    - *Command-type undo may not result in the original state*
  - Support reverting back to a far-back state, without needing to maintain several Command operations

Why

Difficult Undo Operation

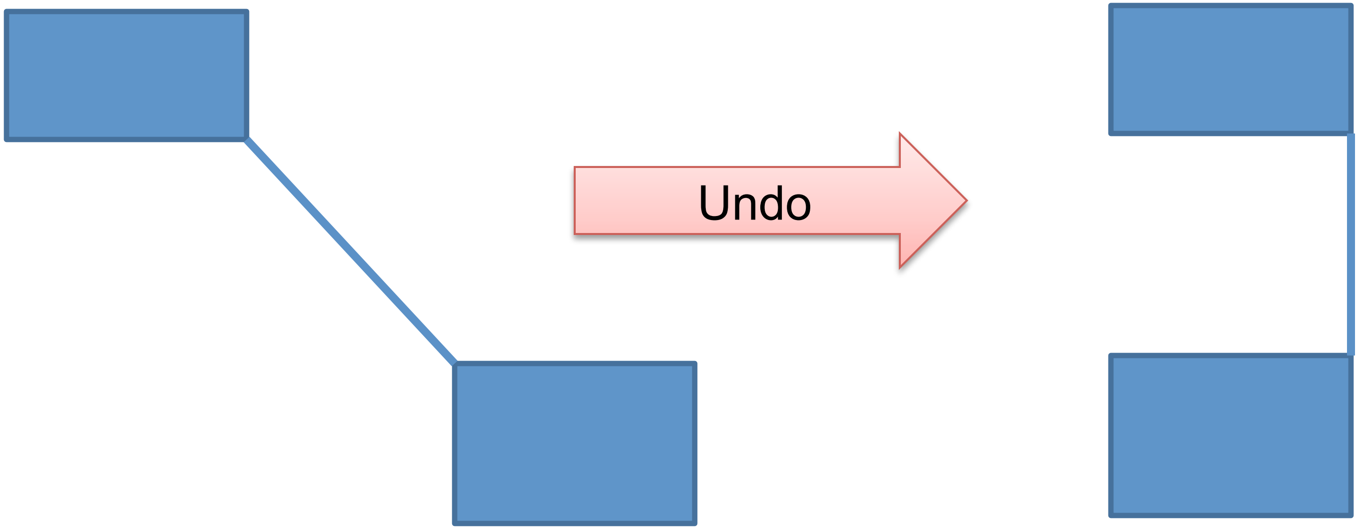
- A graphics layout program may allow us to move objects, and have connected object be recalculated after the move



Box is moved, line is dynamically updated to join at corners

Difficult Undo Operation

- An undo operation (a la Command object) would recalculate the connected lines
  - *Don't know how many lines are connected to box, so cannot simply store line states…*



Line recalculated to be connected at corner

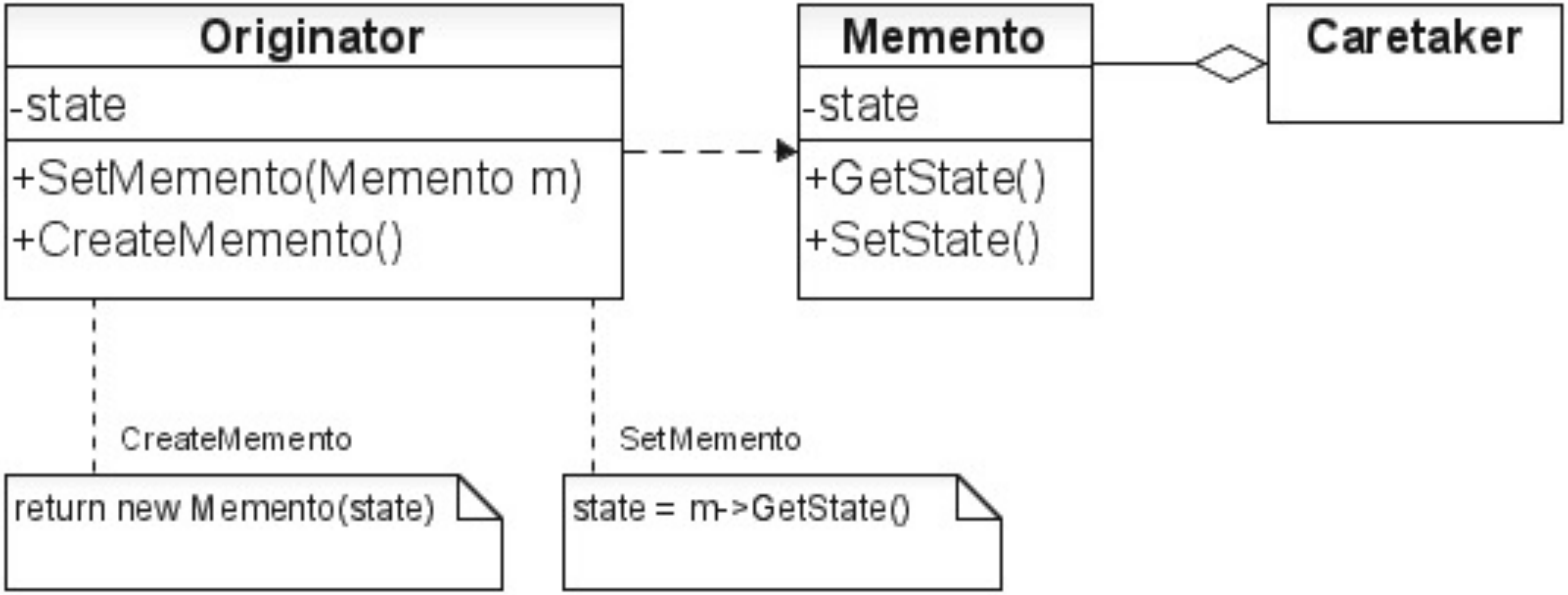
How

Memento Pattern - Participants

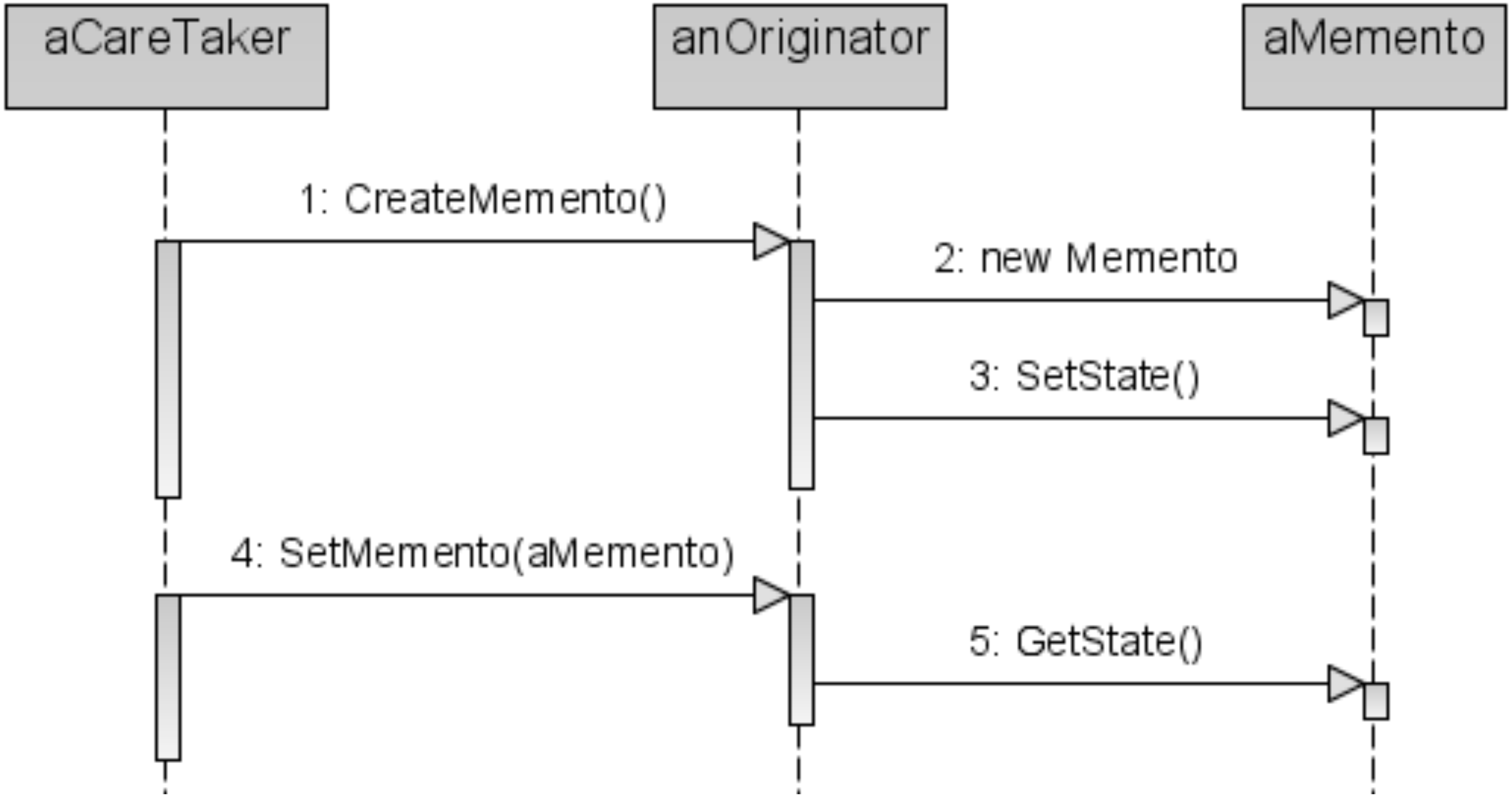
Participants

- **Memento**
  - Stores internal state of the Originator object.
  - Protects against access by objects other than the originator
- **Originator**
  - Creates a memento containing a snapshot of its current internal state
  - Uses memento to restore its internal state
- **Caretaker**
  - Is responsible for the memento’s safekeeping
  - Never operates on or examines the contents of a memento

Memento Pattern - Structure



Memento Pattern - Behavior



Examples

Sudoku Solving

- Automatically solving Sudoku puzzles can use a technique called **Constraint Programming**
  - **Each cell has a set of possible values**
    - *Constraints define which values can be in a cell's set*
  - **If one value is available, assign it to the cell**
    - *Once single values are assigned, "guess" by randomly selecting another value in a random cell*
  - **If a cell has no possible values, solution is invalid!**

Sudoku Solving

- Several forced moves may be performed at a time
  - Don't want to save *all* the Commands and systematically undo – too many commands!

	9	3	1		5	6	4	
7								5
5		1	2		9	3		7
2								3
	3	6	9		7	5	2	
9								1
3		2	4		8	1		9
6								4
	4	7	3		2	8	5	

	9	3	1		5	6	4	
7								5
5		1	2		9	3		7
2								3
	3	6	9		7	5	2	8
9								1
3		2	4		8	1		9
6								4
	4	7	3		2	8	5	

	9	3	1		5	6	4	X
7								5
5		1	2		9	3		7
2								3
	3	6	9		7	5	2	8
9								1
3		2	4		8	1		9
6						2		4
	4	7	3		2	8	5	

Upper right corner
has no valid value –
not a valid solution!

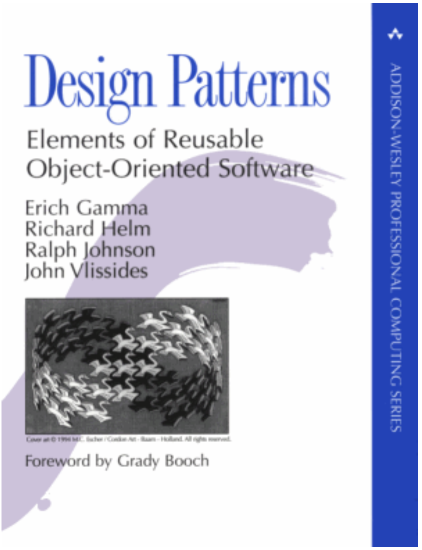
Design Considerations

Design Considerations

- **Command** and Memento act as magic tokens to be passed around and invoked at a later time. In Command, the token represents a request; in Memento, it represents the internal state of an object at a particular time. Polymorphism is important to Command, but not to Memento because its interface is so narrow that a memento can only be passed as a value.
- **Command** can use Memento to maintain the state required for an undo operation.
- Memento is often used in conjunction with **Iterator**. An Iterator can use a Memento to capture the state of an iteration. The Iterator stores the Memento internally.

https://sourcemaking.com/design_patterns/memento

Further Reading



Design Patterns

Elements of Reusable Object-Oriented Software

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides

Foreword by Grady Booch

ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

- **Design Patterns**
  pp. 283-291