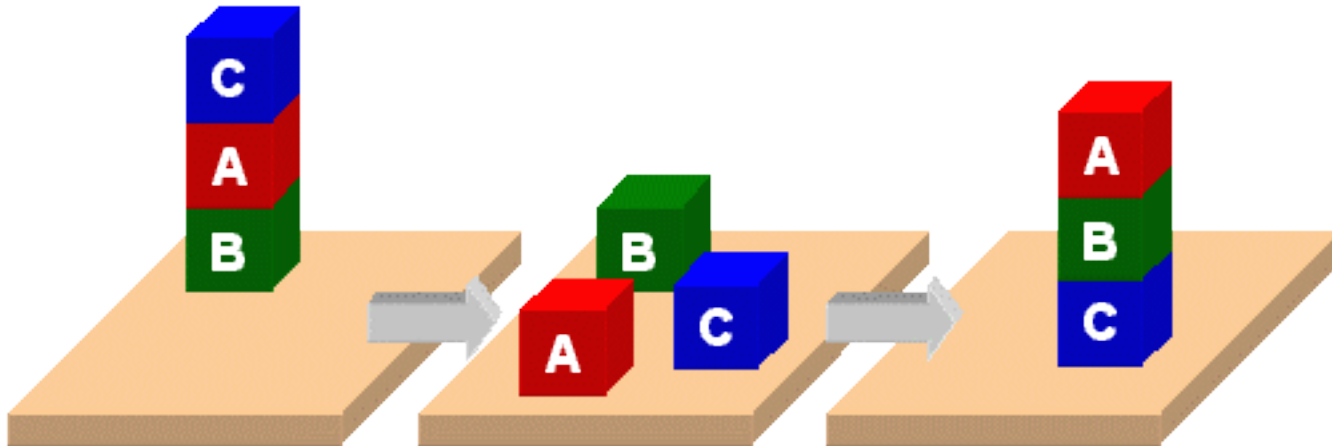
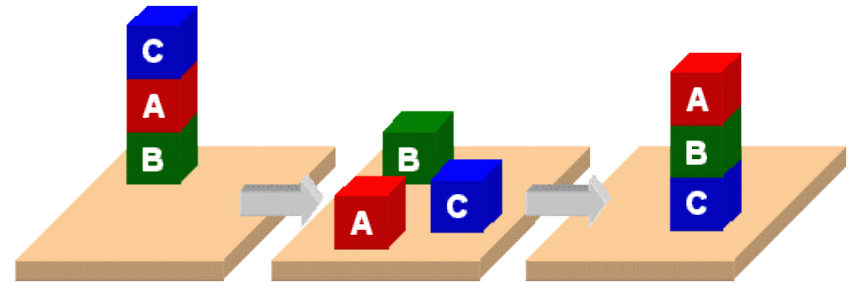


PDDL



PDDL



- Planning Domain Description Language
- Based on STRIPS with various extensions
- Originally defined by Drew McDermott (Yale) and others
- Used in biennial International Planning Competition (IPC) series (1998-2018)
- Many planners use it as a standard input

PDDL Representation

- A task specified via two files: **domain file** and **problem file**
- **Problem file**: gives objects, initial state and goal state
- **Domain file** gives predicates & operators and may be re-used for different problem files
- **Domain file** corresponds to the transition system; **problem files** constitute instances in that system

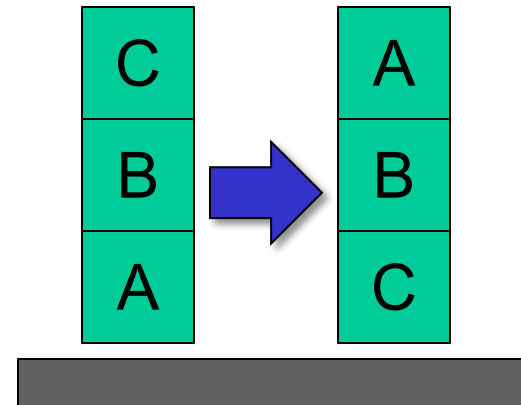
Blocks Word Domain File



```
(define (domain hw5)
  (:requirements :strips)
  (:constants red green blue yellow)
  (:predicates (on ?x ?y) (on-table ?x) (block ?x) ... (clean ?x))
  (:action pick-up
    :parameters (?obj1)
    :precondition (and (clear ?obj1) (on-table ?obj1)
                       (arm-empty))
    :effect (and (not (on-table ?obj1))
                 (not (clear ?obj1))
                 (not (arm-empty))
                 (holding ?obj1)))
  ... more actions ...)
```

```
(define (problem 00)
  (:domain hw5)
  (:objects A B C)
  (:init (arm-empty)
    (block A)
    (color A red)
    (on-table A)
    (block B)
    (on B A)
    (block C)
    (on C B)
    (clear C))
```

Blocks World Problem File



```
(:goal (and (on A B) (on B C))))
```

(define (problem 00)

(:domain hw5)

(:objects A B C)

(:init (arm-empty)

(block A)

(color A red)

(on-table A)

(block B)

(on B A)

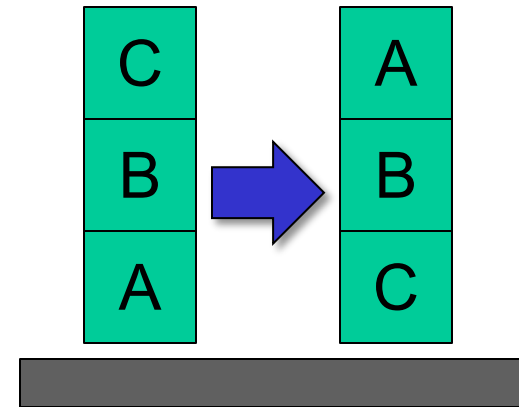
(block C)

(on C B)

(clear C))

(:goal (and (on A B) (on B C))))

Blocks World Problem File



Begin plan

1 (unstack c b)

2 (put-down c)

3 (unstack b a)

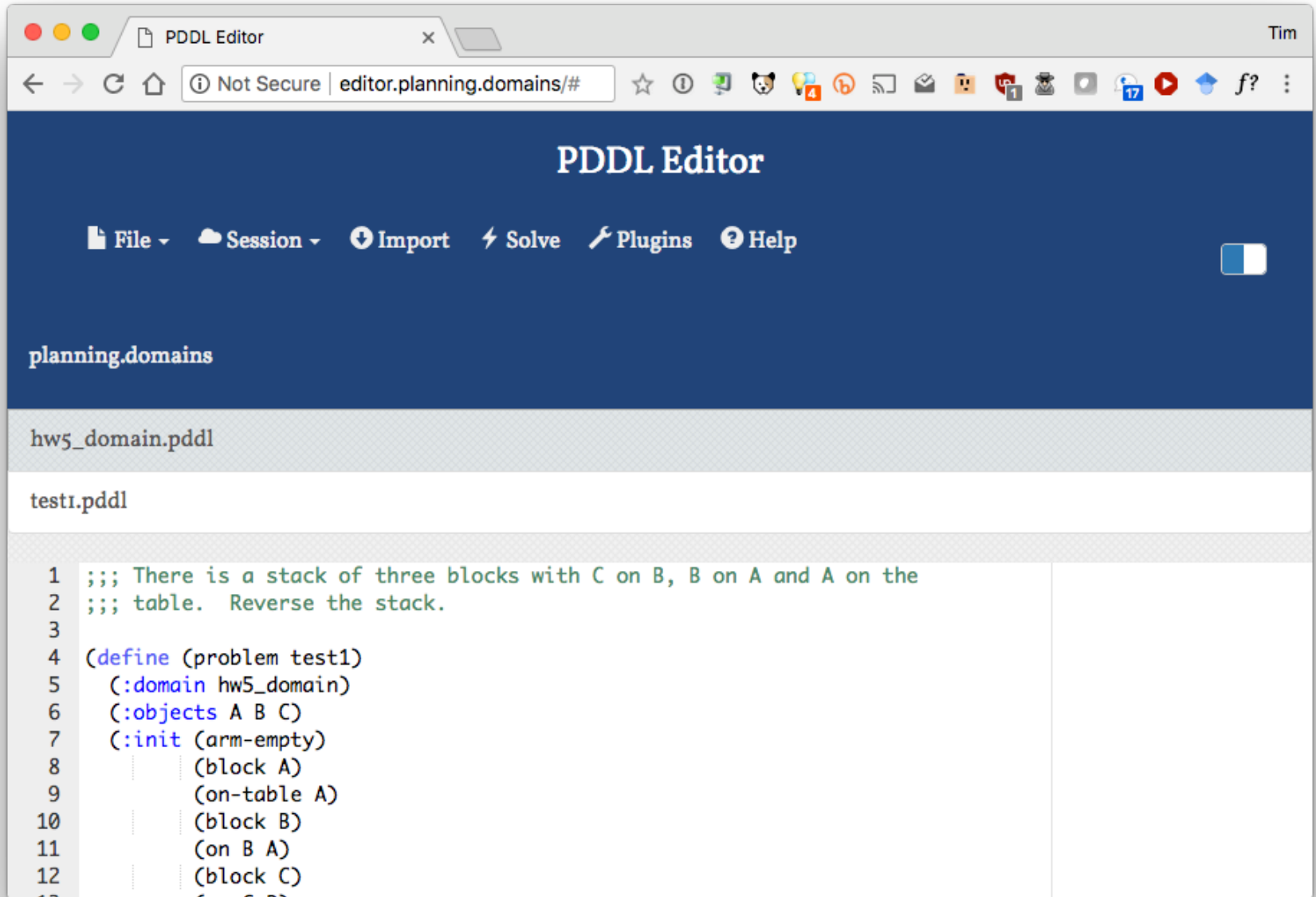
4 (stack b c)

5 (pick-up a)

6 (stack a b)

End plan

http://planning.domains/



The screenshot shows a web browser window titled "PDDL Editor" with the URL "editor.planning.domains/#". The browser's address bar shows "Not Secure" and various extension icons. The application interface has a dark blue header with the title "PDDL Editor" and a menu with "File", "Session", "Import", "Solve", "Plugins", and "Help". Below the header, there are tabs for "planning.domains", "hw5_domain.pddl", and "test1.pddl". The "test1.pddl" tab is active, displaying the following PDDL code:

```
1 ;;; There is a stack of three blocks with C on B, B on A and A on the
2 ;;; table. Reverse the stack.
3
4 (define (problem test1)
5   (:domain hw5_domain)
6   (:objects A B C)
7   (:init (arm-empty)
8           (block A)
9           (on-table A)
10          (block B)
11          (on B A)
12          (block C)
13          (on C B))
```

Planning.domains

- Open source environment for providing planning services using PDDL
- Default planner is [ff](#)
 - very successful forward-chaining heuristic search planner producing sequential plans
 - Can be configured to work with other planners
- Use interactively or call via web-based API
- Use for HW5 to extend blocks world domain