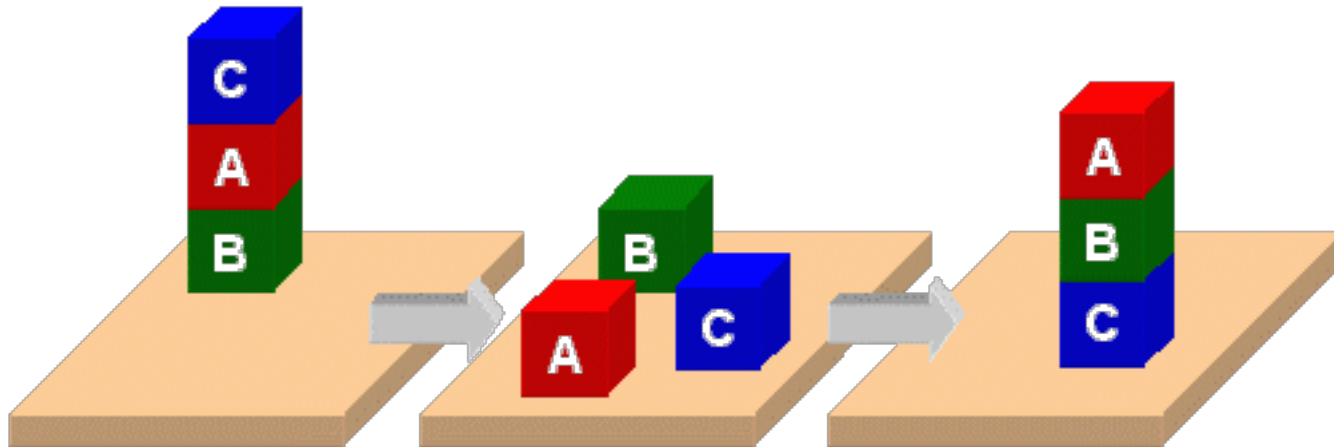


Classic Blocks World



Classic Blocks World

- We'll look at the classic blocks world domain
- Starting with
 - BW: a domain file
 - Several problem files
- We'll use [planning.domains](#) to demonstrate solving the problems
- And then show simple extensions to the domain by adding predicates and constants

bw.pddl 1

(define (domain **bw**)

Allows basic add and delete effects in actions

(:requirements :strips)

(:predicates

List all the predicates with their arguments

(on ?x ?y) ; object ?x is on ?object ?y

(on-table ?x) ; ?x is directly on the table

(clear ?x) ; ?x has nothing on it

(arm-empty) ; robot isn't holding anything

(holding ?x) ; robot is holding ?x

Variables begin with a ?

;; 4 actions to manipulate objects: pickup, putdown, stack, unstack

... actions in next four slides ...

bw.pddl 2

(:action pick-up

:parameters (?ob)

Variable for the argument of a pick-up action

:precondition

(and (clear ?ob)

(on-table ?ob)

(arm-empty))

These three statements must be True before we can do a pick-up action

:effect

(and (not (on-table ?ob))

(not (clear ?ob))

(not (arm-empty))

(holding ?ob)))

After doing a pick-up action, these become True

Adding (not ?X) removes ?X if it's in the KB and vice versa

bw.pddl 3

(:action put-down

:parameters (?ob)

:precondition (holding ?ob)

:effect

(and (not (holding ?ob))

(clear ?ob)

(arm-empty)

(on-table ?ob)))

put-down means put the thing you're holding on the table

(:action stack

:parameters (?ob1 ?ob2)

:precondition (and (holding ?ob) (clear ?ob2))

:effect

(and (not (holding ?ob))

(not (clear ?ob2))

(clear ?ob)

(arm-empty)

(on ?ob ?ob2)))

stack means put the thing you are holding on another object

bw.pddl 5

```
(:action unstack
:parameters (?ob1 ?ob2)
:precondition
  (and (on ?ob1 ?ob2)
        (clear ?ob1)
        (arm-empty))
:effect
  (and (holding ?ob1)
        (clear ?ob2)
        (not (clear ?ob1))
        (not (arm-empty))
        (not (on ?ob1 ?ob2))))
```

unstack means take the first arg off the second arg

First arg can't have anything on it & the robot can't be holding anything

Updates to KB describing new state of the world

) ; this closes the domain definition

;; The arm is empty and there is a stack of three blocks: C is on B which is on A
;; which is on the table. The goal is to reverse the stack, i.e., have A on B and B
;; on C. No need to mention C is on the table, since domain constraints will enforce it.

(define (**problem** p03)

(:**domain** bw)

(:**objects** A B C)

(:**init** (arm-empty)

(on-table A)

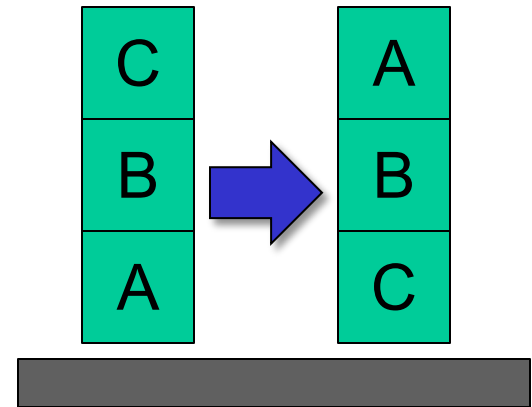
(on B A)

(on C B)

(clear C))

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

(on B C))))



p03.pddl

http://planning.domains/

The screenshot shows a web browser window with the URL `planning.domains/`. The browser's address bar shows the site name and a search box. The website's navigation bar includes links for `API`, `Solver`, `Editor`, `Education`, and `About`. The main content area features the title `Planning.Domains` and the subtitle `A collection of tools for working with planning domains.` Below this, there are four colored buttons: a dark grey button for `planning.domains`, a green button for `1) api.planning.domains`, a purple button for `2) solver.planning.domains`, a blue button for `3) editor.planning.domains`, and a red button for `4) education.planning.domains`. A cyan callout box with a pointer to the blue button contains the text: `Open the PDDL editor, upload our domain and problem files, and run the solver.`

planning.domains/

planning.domains

API Solver Editor Education About

planning.domains

Planning.Domains

A collection of tools for working with planning domains.

planning.domains : 1) api.planning.domains 2) solver.planning.domains

3) editor.planning.domains 4) education.planning.domains

Open the PDDL editor, upload our domain and problem files, and run the solver.

Online Demonstration

Using [planning.domains](#) and files in the [planning](#) directory of our 2020 [671 code repo](#)

- Blocks world
 - bw.pddl
 - p01.pddl
 - p02.pddl ...
- Air Cargo
 - ac_domain.pddl
 - Ac_p0.pddl

Fín