Prodigy

- A classical STRIPS-style planner
  - Domain Representation: objects, operators
  - Problem Representation: initial state, goal state

- Operators have preconditions and effects
Example – Blocksworld

(On A B)  
(On B Table)  
(On C Table)  
(Clear A)  
(Clear C)  
(Clear Table)  
(Arm–empty)

Initial State

Goal State

Operators:

(Pickup x)

preconds:  (Clear x)  
(Arm–empty)

adds:  (Holding x)  
if (On x y), (Clear y)

dels:  (Arm–empty)  
if (On x y), (On x y)

(Putdown x y)

preconds:  (Holding x)  
(Clear y)

adds:  (On x y)  
(Arm–empty)

if (y != Table), (Clear y)
Putdown C A
(Holding C)
(Clear A)

(On C A)
(On B C)
Prodigy/Blocksworld (cont.)

Pickup A
(Clear C)

Pickup B
(Holding B)
Prodigy/Blocksworld (cont.)

Pickup A
(Pickup B (Arm−empty))

Putdown A Table
Prodigy/Blocksworld (cont.)

Putdown A Table
Prodigy/Blocksworld (cont.)
Prodigy/Blocksworld (cont.)
Issues in Planning

- Representations
- Algorithms
- Conditional effects
- Dynamic worlds
- Mixing planning and execution
- Learning
- Large-scale applications

Fairly mature field
# Example – Blocksworld

<table>
<thead>
<tr>
<th>Initial State</th>
<th>Goal State</th>
</tr>
</thead>
<tbody>
<tr>
<td>A B C</td>
<td>B C A</td>
</tr>
<tr>
<td>(On A B)</td>
<td>(On C A)</td>
</tr>
<tr>
<td>(On B Table)</td>
<td>(On B C)</td>
</tr>
<tr>
<td>(On C Table)</td>
<td>(+ whatever)</td>
</tr>
<tr>
<td>(Clear A)</td>
<td></td>
</tr>
<tr>
<td>(Clear C)</td>
<td></td>
</tr>
<tr>
<td>(Clear Table)</td>
<td></td>
</tr>
<tr>
<td>(Arm–empty)</td>
<td></td>
</tr>
</tbody>
</table>

### Operators:

- **(Pickup x)**
  - **preconds:** (Clear x)
    - (Arm–empty)
  - **adds:** (Holding x)
    - if (On x y), (Clear y)
  - **dels:** (Arm–empty)
    - if (On x y), (On x y)

- **(Putdown x y)**
  - **preconds:** (Holding x)
    - (Clear y)
  - **adds:** (On x y)
    - (Arm–empty)
  - **dels:** (Holding x)
    - if (y != Table), (Clear y)