Prodigy

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

- Operators have preconditions and effects
Example – Blocksworld

\[
\begin{array}{c}
\text{State} \\
\text{Initial} \\
\text{Goal} \\
\text{State}
\end{array}
\]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>(On A B)</td>
<td>(On B Table)</td>
<td>(On C Table)</td>
</tr>
<tr>
<td>(Clear A)</td>
<td>(Clear C)</td>
<td>(Clear Table)</td>
</tr>
<tr>
<td>(Arm−empty)</td>
<td></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)
Prodigy/Blocksworld (cont.)

Putdown C A
(Holding C)
(Clear A)

C

A

B

C

A

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

- **A**
- **B**
- **C**

(A) (Clear C) (Holding B)

- **C**
- **A**

(B) Pickup A

- **C**
- **B**

(Pickup B)

- **A**
- **B**

- **C**

(A) Pickup A

- **C**
- **B**

(Pickup B)

- **A**
- **B**

- **C**

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

- Pickup A
- (Arm-empty)
- Pickup B
- Putdown A Table
- B
- A
- C
- C
- B
- C
Prodigy/Blocksworld (cont.)

Putdown A Table
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>(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>(Clear Table)</td>
</tr>
<tr>
<td>(Clear C)</td>
<td>(Arm–empty)</td>
</tr>
<tr>
<td>(Clear Table)</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)
- if (y != Table), (Clear y)