Controlling Execution

Maurice Pagnucco
School of Computer Science and Engineering
University of New South Wales
NSW 2052, AUSTRALIA
morri@cse.unsw.edu.au

The Cut Operator (!)

- Sometimes we need a way of preventing Prolog finding all solutions
- The cut operator is a built-in predicate that prevents backtracking
- It violates the declarative reading of a Prolog program
- Use it VERY sparingly!!
Backtracking

\[
\text{lectures(adnan, Subject), studies(Student, Subject)}? \\
\text{Subject = 4418} \\
\text{Student = jane} \\
\text{Subject = 9518} \\
\text{Student = jack}
\]
Cut Prunes the Search

- Prevents backtracking to goals left of the cut by throwing away remaining choice points

```prolog
lectures(adnan, Subject), !, studies(Student, Subject)?
```

Diagram:

- lectures(adnan, 9414)
- lectures(adnan, 9518)
- studies(jane, 9414)
- studies(jack, 9518)
Example

overdue(Today, Title, CatNo, MemFamily) :-
  loan(CatNo, MemNo, _, DueDate),
  later(Today, DueDate), !,
  book(CatNo, Title, _),
  member(MemNo, name(MemFamily, _), _).
Controlling Execution

- Some methods for controlling execution in Prolog:
  - Ordering of clauses (facts and rules)
  - Ordering of subgoals within a rule
  - Cut (!) operator

- Use each with care