Deductive Databases and Logic Programming (Winter 2007/2008)

## Chapter 9: Constraint Logic Programming

- Introduction, Examples
- Basic Query Evaluation
- Finite Domain Constraint Solver

Stefan Brass: Deductive Databases and Logic Programming



- Constraint logic programming (CLP) extends standard logic programming by constraints, which can in principle be any kind of logical formulae.
- A concrete system permits only a subset of formulae, because it needs a constraint solver that can
  - ◊ check a set of constraints for consistency,

Sometimes incomplete constraint solvers are used, which are not able to detect all inconsistencies.

◊ simplify a set of constraints, so that it can be presented to the user as an answer.

Optimal would be Variable = Value, but this is not always possible.

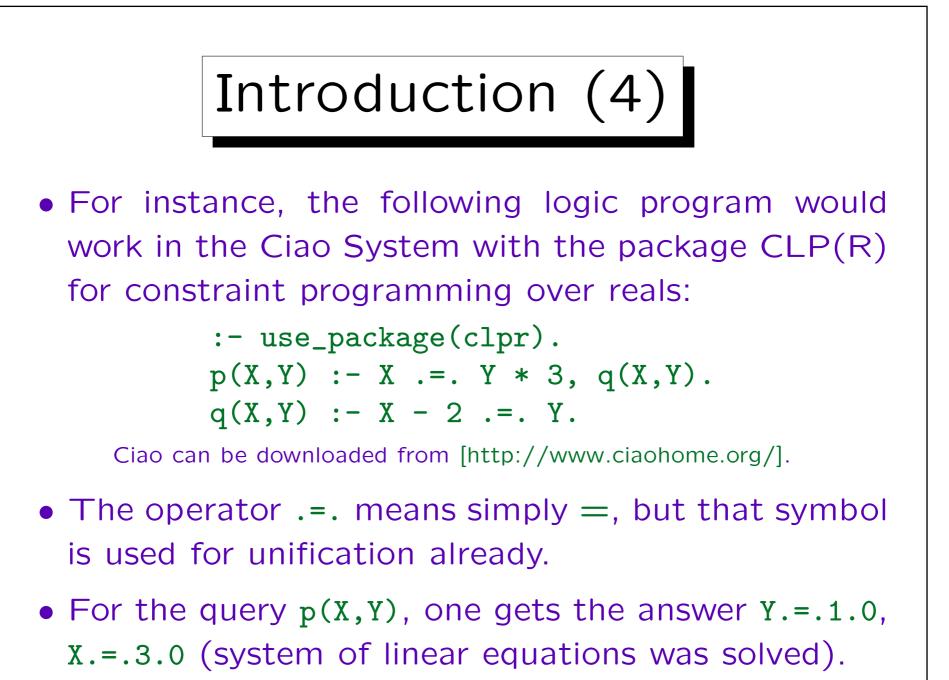
Introduction (2)

- Normally, the constraints are simply literals with special predicates.
- The semantics of these predicates is not defined by rules, but by a logical "constraint theory".
- The semantics is implemented or approximated by the constraint solver in the system.

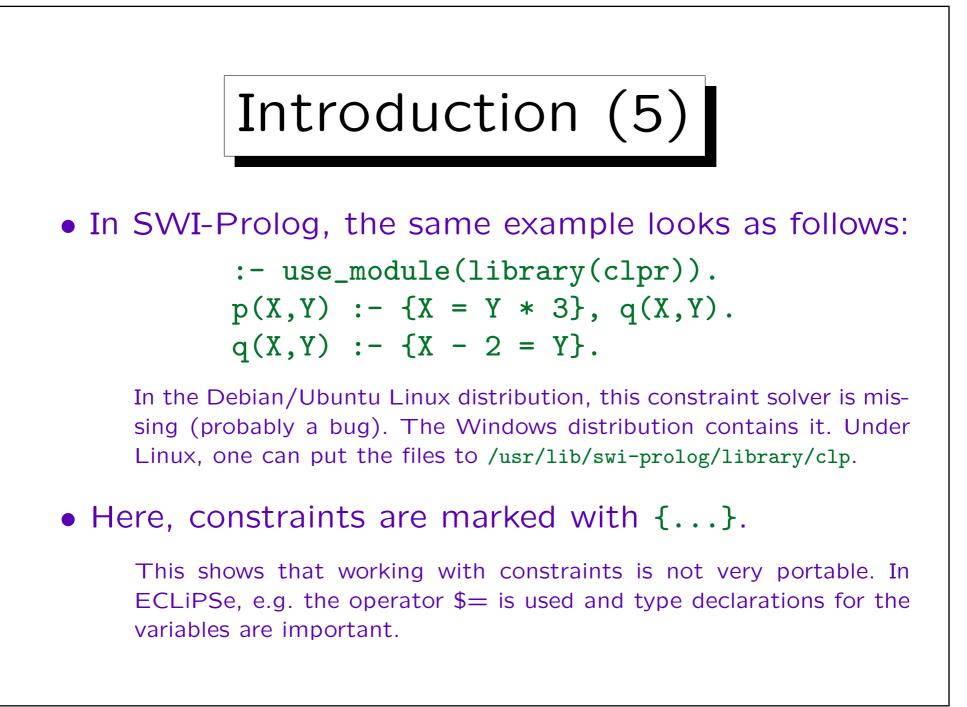
"Constraint Handling Rules" (CHR) is an approach to define constraint solvers by special rules. However, these look quite different from the usual logic programming rules.

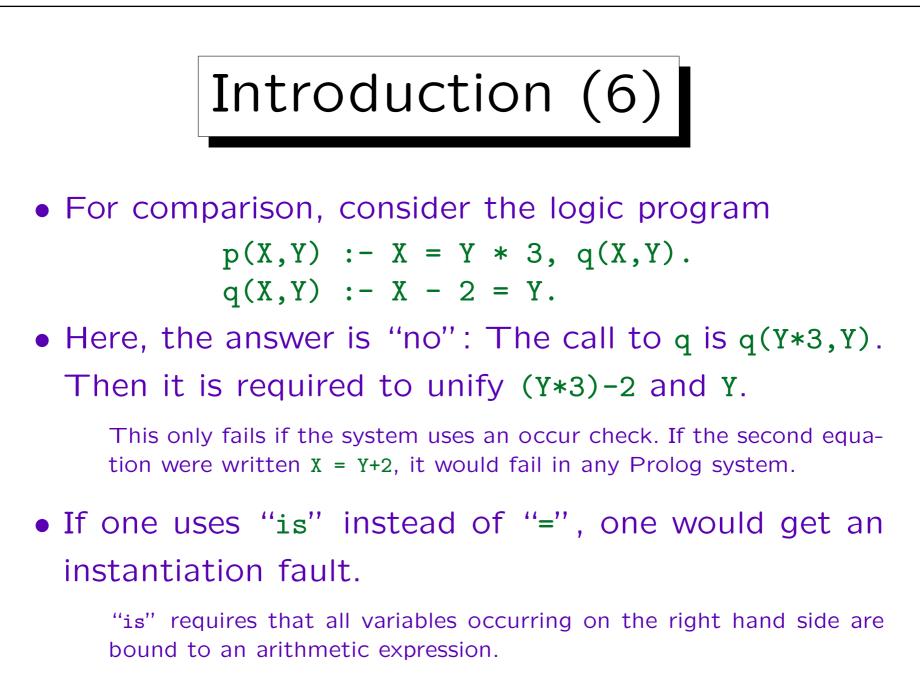


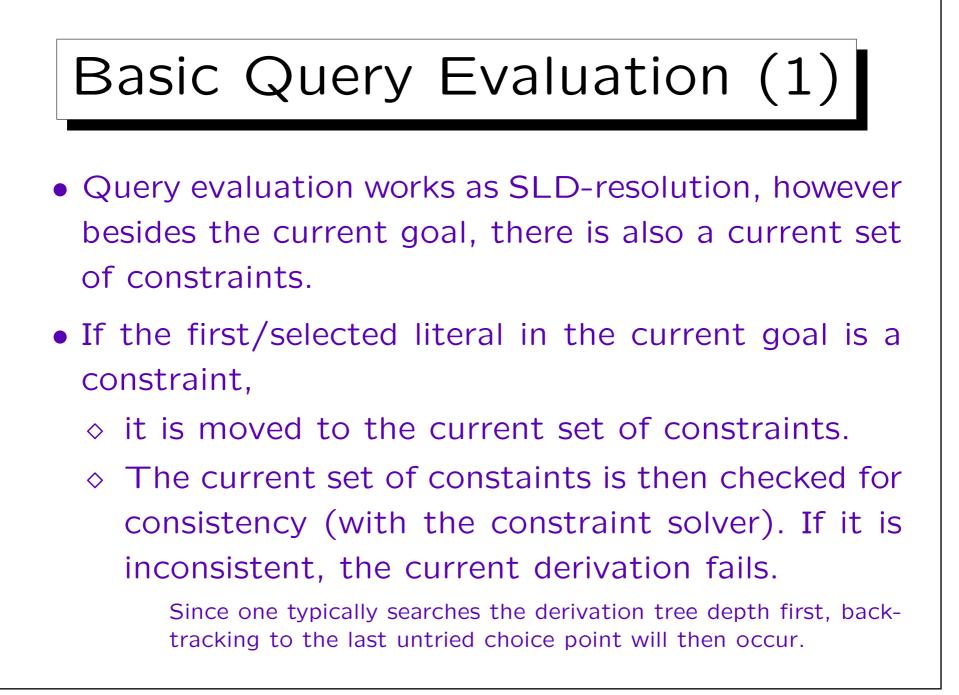
- The function symbols that can be used for term construction in constraint literals is limited: It must be evaluable functions which the constraint solver knows (e.g., +, -, \*, /).
- Variables in constraints have a specific domain (e.g., integers). They cannot be bound to arbitrary terms.
- Note the difference between
  - $\diamond$  The value of variable X is the term Y+3.
  - ♦ The value of variable X is an integer, at the moment it is known that X = Y+3.

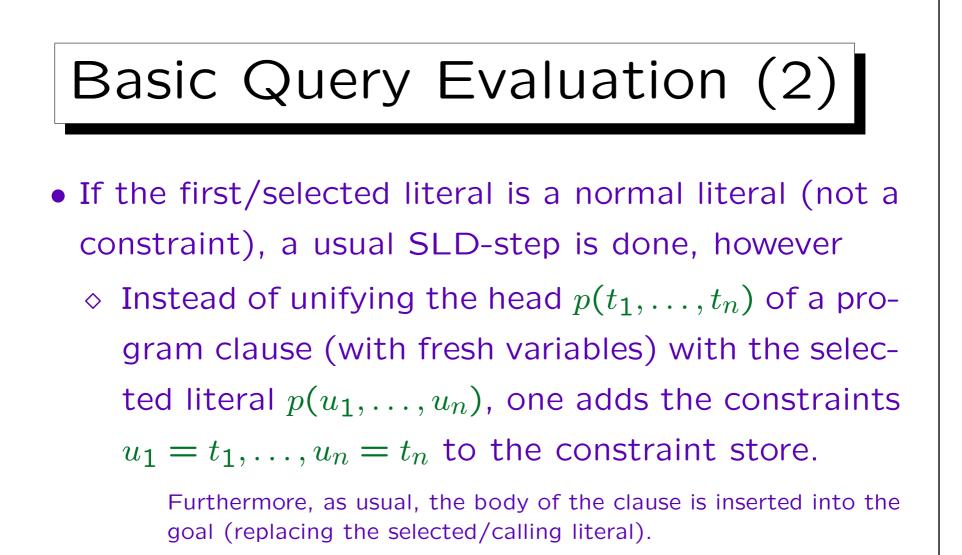


Stefan Brass: Deductive Databases and Logic Programming Univ. Halle, 2011

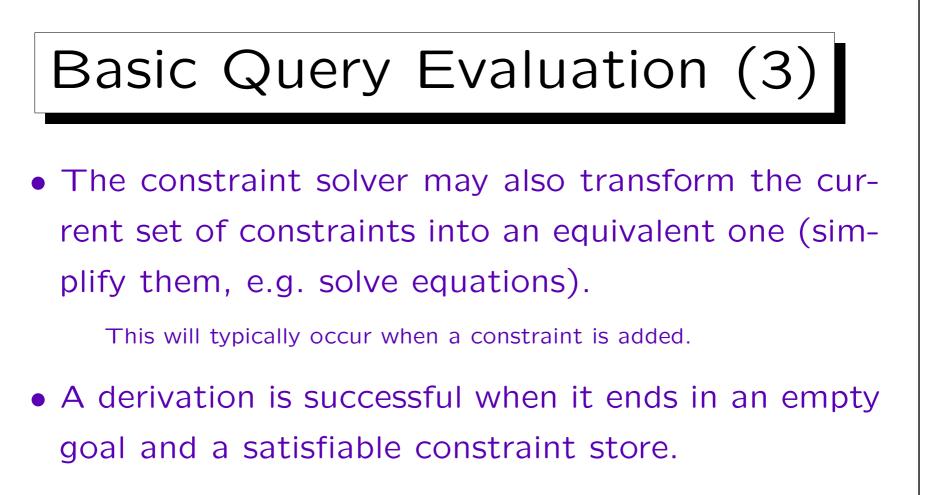






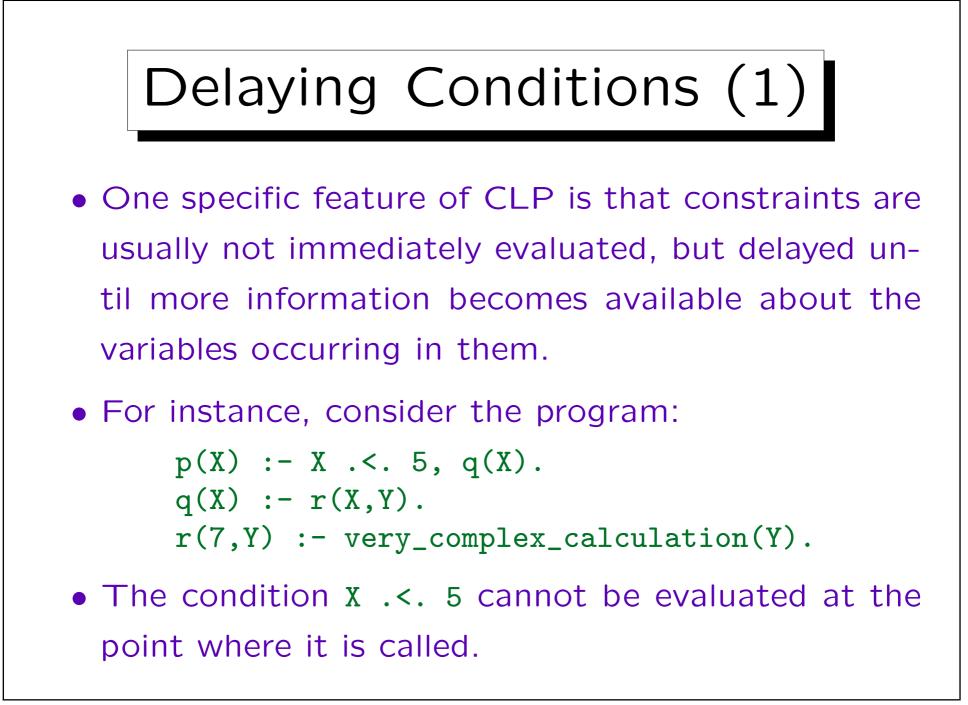


◊ In this way, unification can be seen as a very specific constraint solver.



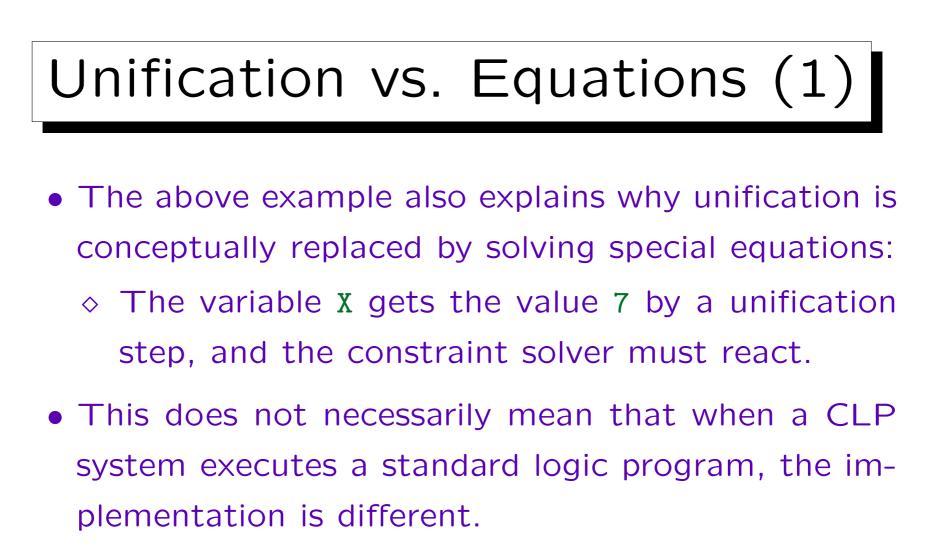
• Instead of an answer substitution, the contents of the constraint store will then be printed.

Possibly restricted to the variables that occurred in the query.

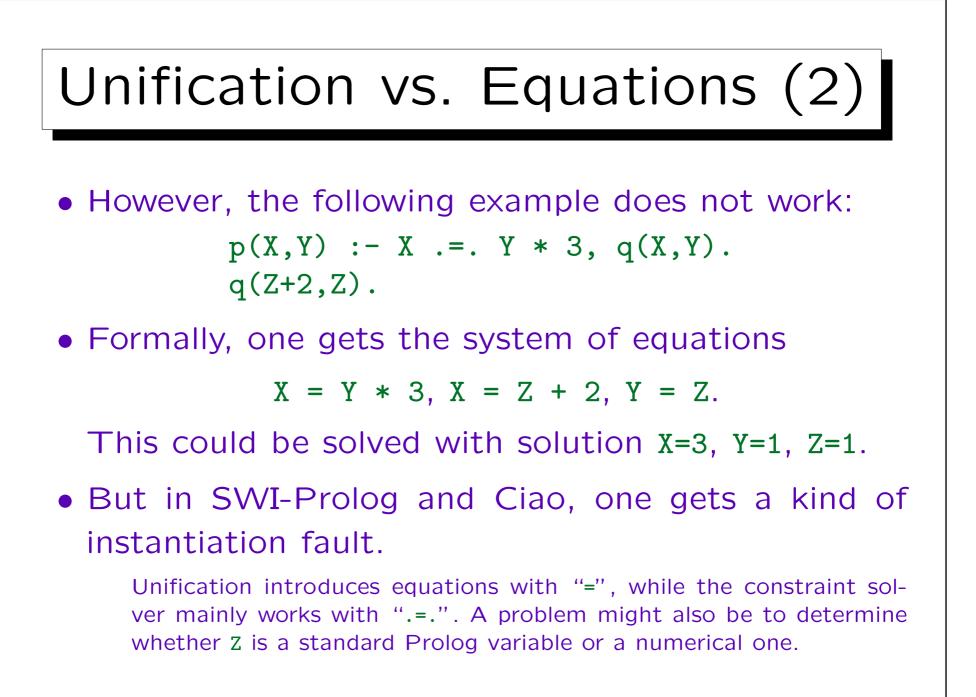




- The constraint X .<. 5 waits in the background and as soon as X is bound to 7, it causes the failure.
- In Prolog (with < instead of .<.) one would get an instantiation fault.
- One would have to move X < 5 after the call q(X), but then the very\_complex\_calculation is done, and the failure is detected only afterwards.
- This gives logic programming a kind of coroutining.

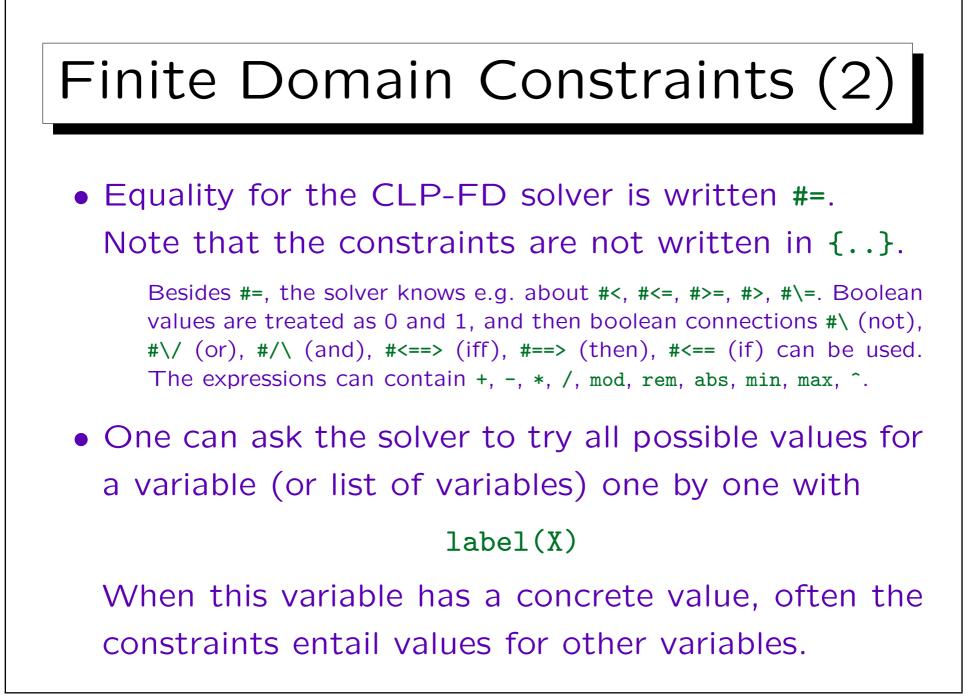


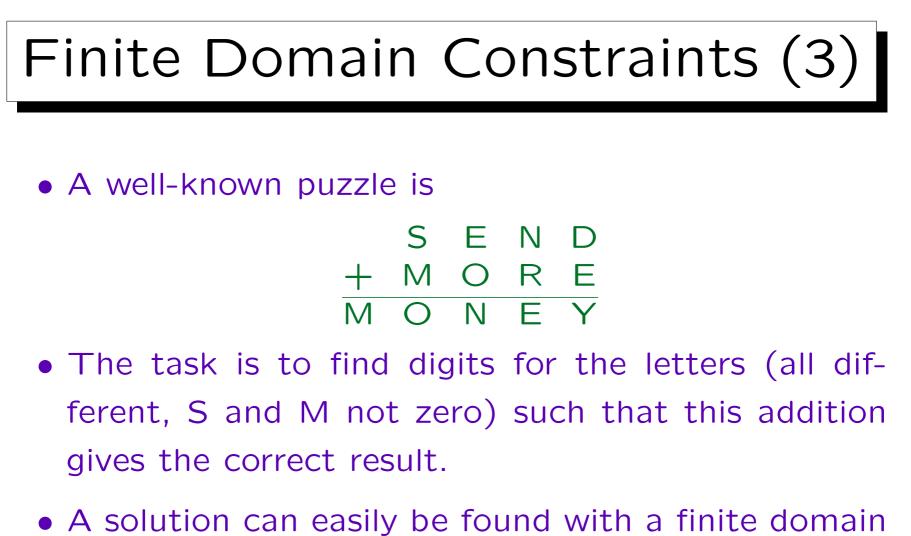
Actually, most modern Prolog systems have CLP libraries.



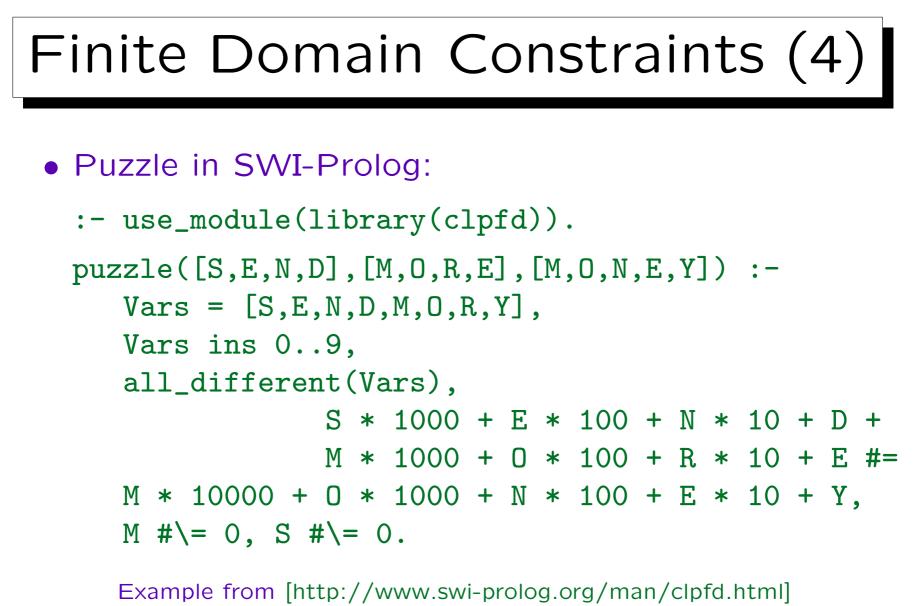
Finite Domain Constraints (1)
<ul> <li>A finite domain solver keeps for each variable a fini- te domain of values, usually an interval of integers (or union of such intervals).</li> </ul>
<ul> <li>For the CLP-FD Solver in SWI-Prolog, one declares a domain for a variable e.g. with</li> </ul>
X in 09
<ul> <li>It is also possible to do this for a list of variables, e.g.</li> </ul>

```
[X, Y, Z] ins 0..100
```





• A solution can easily be found with a finite domain solver. One only needs a formal specification of the problem as shown on the next slide.



(with one small modification in the head).

Stefan Brass: Deductive Databases and Logic Programming

