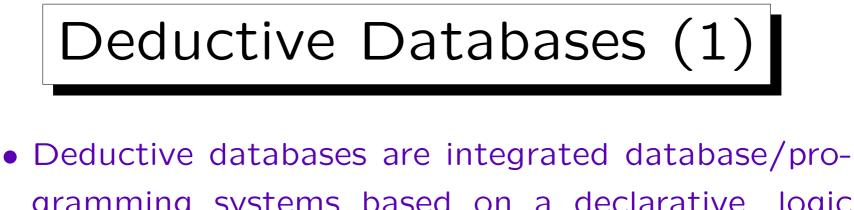
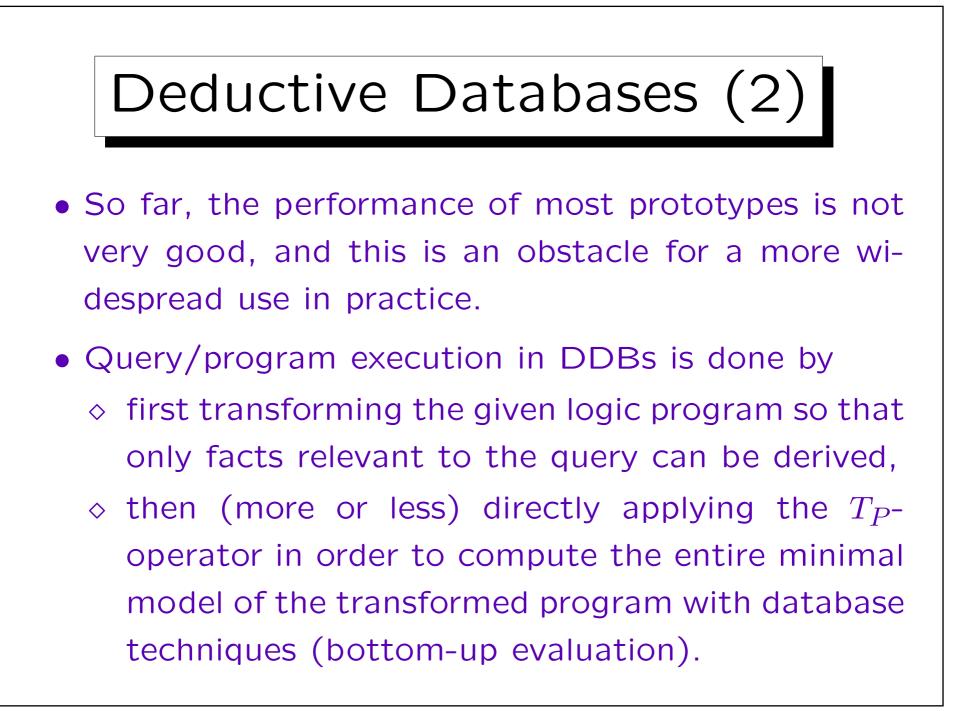
Implementation Alternatives for Bottom-Up Evaluation

Stefan Brass University of Halle, Germany



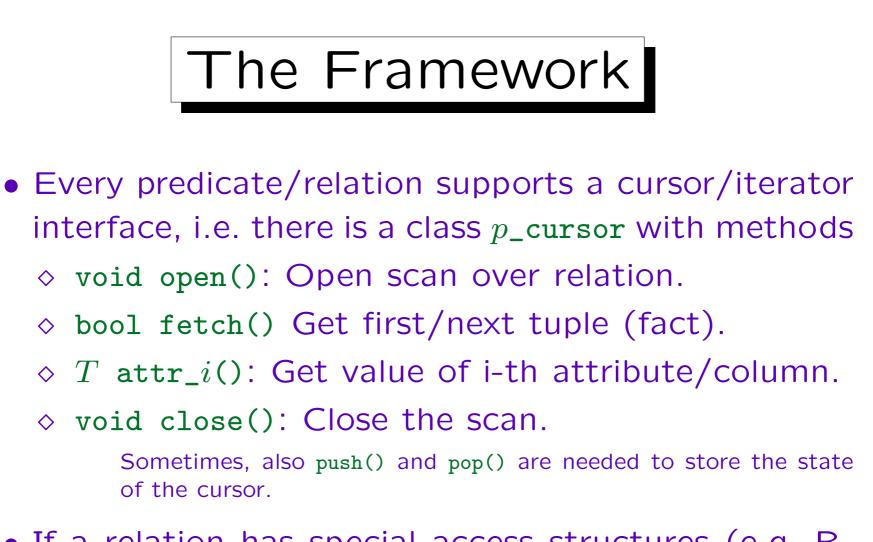
- gramming systems based on a declarative, logic programming style language ("Datalog").
- SQL is a declarative, logic-based language, but supports only database queries (and updates etc.).
- Any serious database application needs also programming. Currently, SQL must be combined with a conventional programming language for this task.

The need for programming has grown because of stored procedures, object-relational extensions and web applications.

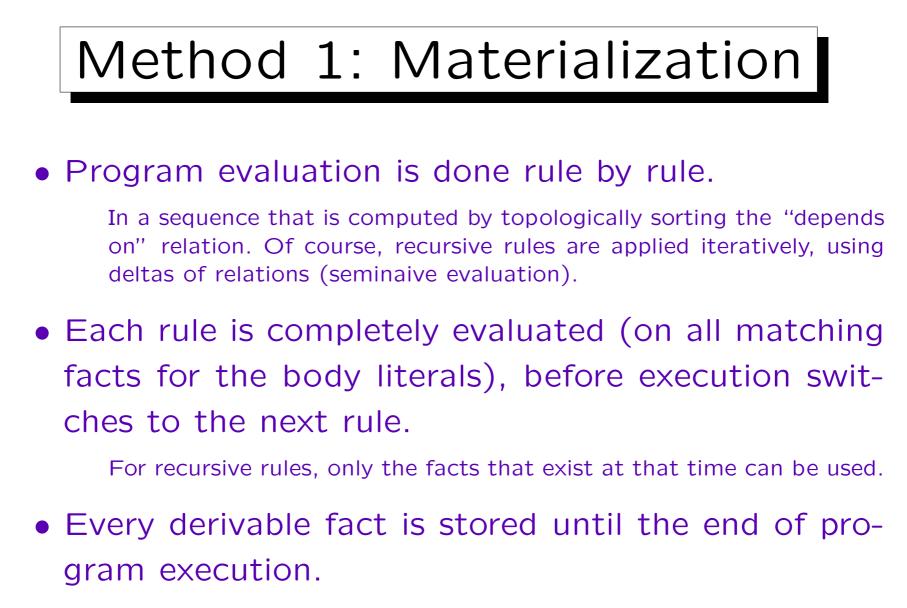




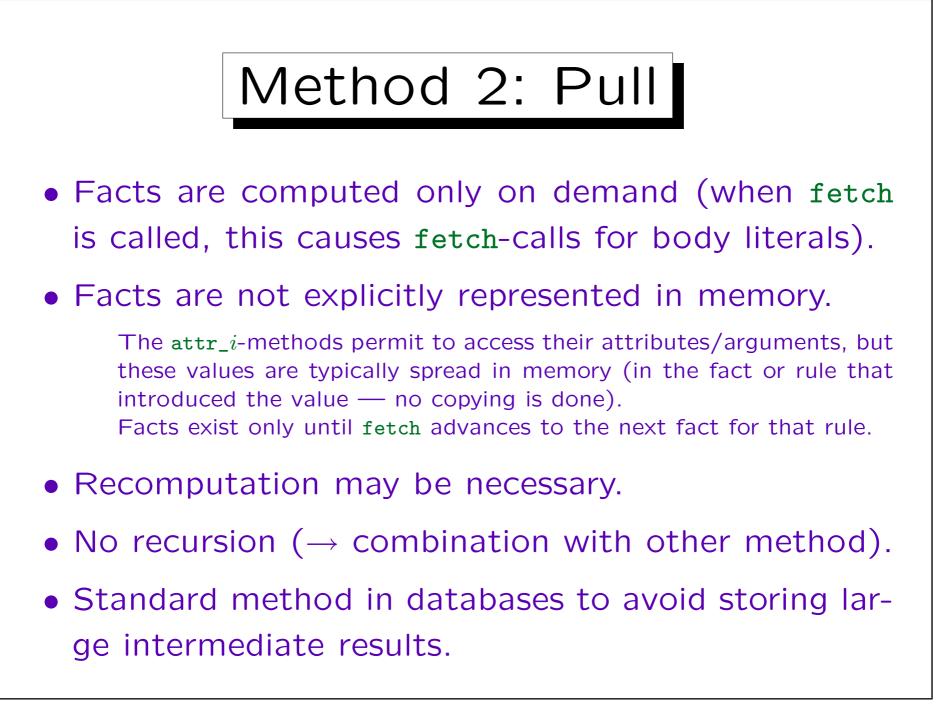
- I previously made a nice proposal for the transformation part (SLDMagic).
- Now I need a good bottom-up evaluation machine.
- My goal is to do this by translation from Datalog to C++.
- A student developed a prototype, but it turned out that there are several implementation alternatives (the prototype contains only the most basic one).
- I did performance measurements for these techniques with manual translations of several examples.



• If a relation has special access structures (e.g. Btree index), there are additional cursor classes for specific binding patterns (then open has args).



Or at least until the last rule is applied that might use the fact.



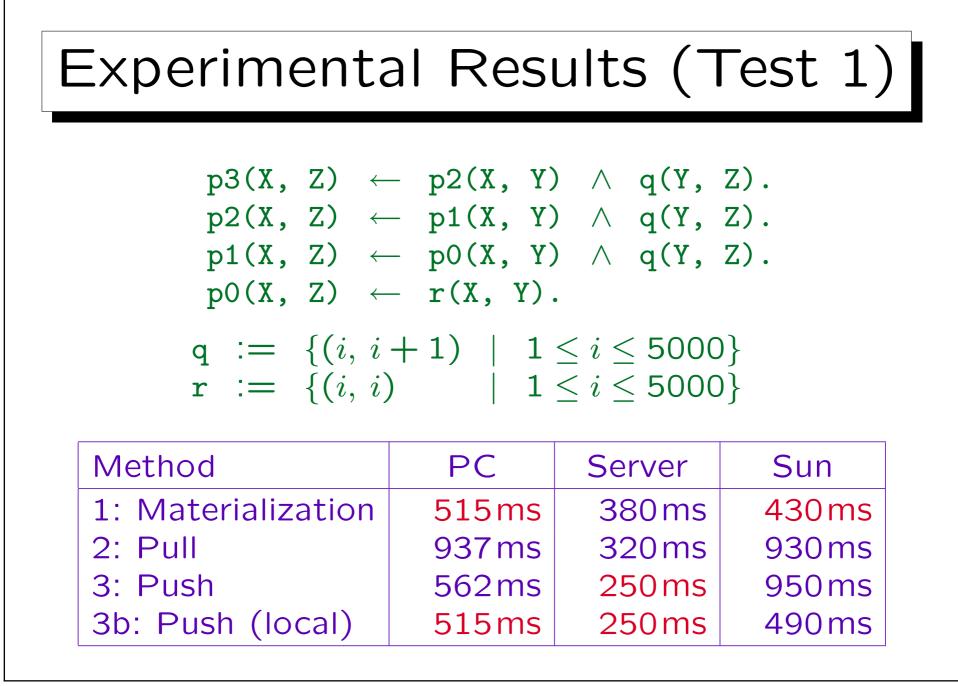
Method 3: Push

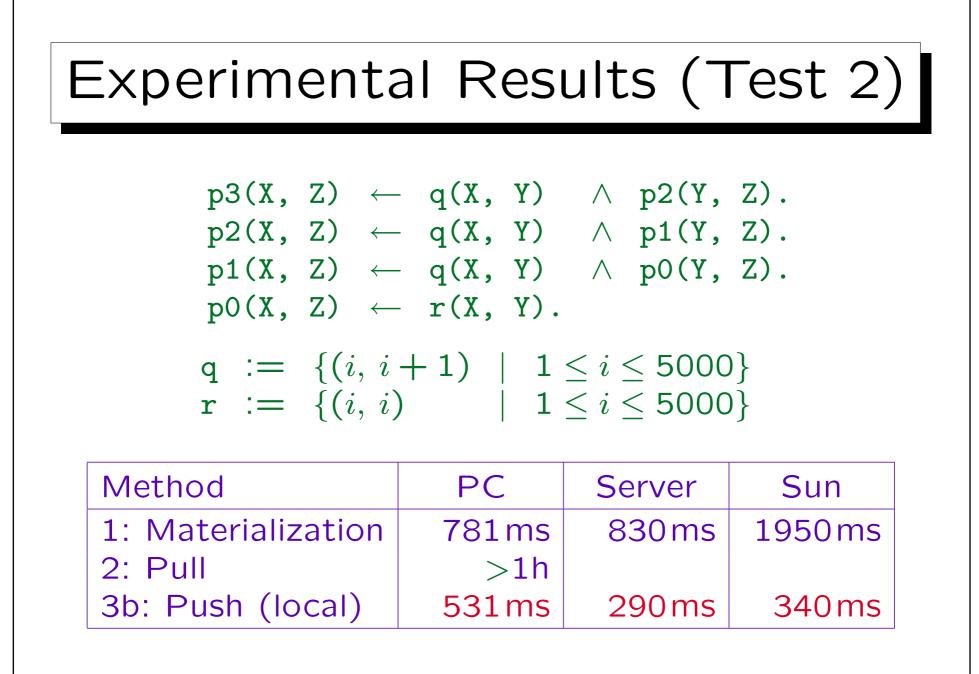
- Derived facts are immediately used to derive more facts with other rules. Backtracking is needed if a fact can possibly be used in different places.
- Only a single fact of each derived predicate evaluated with this method is stored in memory.

Unless duplicate elimination is needed (e.g. for cyclic recursions).

• This method works only with rules containing only one body literal with a predicate defined by rules.

But: (1) SLDMagic produces such rules. (2) A combination with other methods is possible, e.g. materialization can be used for some predicates (which then no longer count for the limitation).







- There are several different ways how to implement bottom-up evaluation (more than the three above).
- The best method depends on the input program, as well as the hardware and the compiler.
- Different methods can be combined.
- Next goals: Implement the transformation for these three and possibly other methods, develop heuristics for choosing a method, experiment with different data structures for storing relations.
- http://www.informatik.uni-halle.de/~brass/botup/