# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

AD-A247 039

...is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, ...ing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this ...ing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson ...to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

2. REPORT DATE
1/31/92

3. REPORT TYPE AND DATES COVERED
final technical 12/1/90-11/30/91

"Research into the Design and Implementation of Knowledge-Base Systems" (U)

5. FUNDING NUMBERS
61102F
2304/A2

6. AUTHOR(S)
Jeffrey D. Ullman

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Computer Science Department
Stanford University
Stanford, CA 94305

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR- 92 0157

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
Building 410
Bolling AFB DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFOSR-91-0066



11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release: distribution unlimited.

12b. DISTRIBUTION CODE
UL

13. ABSTRACT (Maximum 200 words)

A working prototype of the NAIL! system was implemented. This system extends SQL, providing a general purpose computing capability. The two elements of NAIL! consist of GLUE, a logical rule formulation of SQL, and NAIL, a declarative language which generates GLUE code. Applications have been demonstrated that implement a building construction schedule and a VLSI CAD logic simulator. Various query optimization algorithms have been studied and implemented in NAIL.

14. SUBJECT TERMS

15. NUMBER OF PAGES
4

16. PRICE CODE

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | SAR |

Final report on Grant AFOSR-90-0066
Dec., 1990 — Nov., 1991
Jeffrey D. Ullman, PI



## NAIL System Implementation

We developed a working prototype of the NAIL system. We implemented a number of application programs and, and our experience with these programs has enabled us so far to improve system performance by a factor of 6. A description of the earliest version of the system appears in Phipps, Derr, and Ross [1991].

We are just beginning the process of optimizing the system and hope for much better performance next year. The following are the principal components of the system and the authors thereof.

- The GLUE language is by Geoff Phipps. This language can be thought of as SQL statements connected by conventional flow of control, procedures, and modules. However, the statements themselves are written as logical rules, rather than with the algebraic SQL syntax. The GLUE manual is available: Phipps [1991].
- The Iglue intermediate language has been implemented by Marcia Derr. This language consists of single operations of relational algebra, but like GLUE it uses a logical syntax. Derr is beginning to optimize the storage structures for the relations that represent the values of logical predicates. A key optimization that is not analogous to the "standard" code optimization tricks is detection of relations whose values are singletons and the resulting simplification of operations on these relations.
- The NAIL front end has been implemented by Ashish Gupta, Ken Ross, and Kate Morris. This language is fully declarative logic with negation, using the well-founded semantics (an idea developed under a predecessor grant and recently journal-published as Van Gelder, Ross, and Schlipf [1991]). It implements the algorithm described in Ross [1991b], which applies the magic-sets optimization and generates GLUE code.

## Applications

We have also developed a number of applications. The first significant application was a scheduler for building construction by Ashish Gupta and Sanjai Tiwari. This program gave us the measurement for speed up of the system as the implementations of GLUE and Iglue improved. The largest program, by Geoff Phipps, is an equivalent of the Thor logic simulator used by the VLSI CAD group at Stanford; this program is 1500 lines of GLUE.

## Theory of Deductive Databases

There are a number of other ideas, not directly related to the system, that have been developed and/or published by the NAIL group over the past year.

- Gupta and Mumick [1992] shows that the magic-sets optimization, which was developed for recursive logic, can be applied to nonrecursive programs as well. The problem was that there are examples where nonrecursive programs become recursive when magic-sets is applied. These students show that the resulting rules can always be safely modified to eliminate the introduced recursion. In earlier work, Mumick demonstrated that magic-sets applied to nested, nonrecursive queries is at least as good as the best of the previously known methods for dealing with such queries, so there is reason to believe that magic-sets will be the method of choice for all nonrecursive queries now.

- Hakan Jakobsson [1991, 1992] has developed new main-memory algorithms for computing transitive closures of large, sparse relations, and for a family of operations generalizing transitive closures as well. It is difficult to claim that some algorithm is "best" for transitive closure on arbitrary graphs, because one can always invent an algorithm that looks for graphs of a special type, say chains, and spews out the transitive closure for those graphs as fast as possible but does something expensive on all other graphs. However, Jakobsson's algorithms can be compared with a family of algorithms that are "data independent" in a formal sense he defines; this class of algorithms includes all the usual combinations of path doubling and depth-first search. He then shows that he is never worse than the best of these algorithms, and beats all the known ones by a factor that is at least the square root of the number of nodes.

- Ross [1991a] generalizes the techniques called "regular recursions" that beat magic-sets algorithms when they are applicable. He shows that regular recursions are but one example of how to eliminate tail recursion in logic programs.

- Brodsky and Sagiv [1991] develops further the theory of when one can show that top-down search (as in Prolog) for solutions to a logical query will terminate.

- Mumick and Pirahesh [1991] deal with the fact that sometimes queries in which "too much" is specified require more work than if less were specified. The most common example of this phenomenon is in graph algorithms, where it is accepted that to answer a point-to-point reachability problem (can I get from $a$ to $b$), you need to forget about $b$ and ask the more general question "where can I get to from $a$?" These authors study algorithms for optimizing queries that have this property of more bound arguments than can be used productively.

- Chaudhuri [1991] gives algorithms for detecting two opportunities for optimizing cacheing of relations during a query. First is the "emptiness" property, where a tuple not used in one round of a recursive query evaluation will never again be used. The second, "used-at-most-once," is that a tuple which is used on a given round is guaranteed never again to be needed. These properties appear in many common recursive queries.

- Ullman [1991a] demonstrates certain incompatabilities between the deductive (logical) and object-oriented database paradigms. In particular, deductive databases cannot regard object-identity as an inviolate principal and cannot support dynamic types in a nontrivial way.

- Ullman [1992b, c] are surveys of optimization techniques for conjunctive queries and of techniques for parallelizing logical queries, respectively.

### Theses Completed

There were three doctoral theses completed this year with support of the grant or previous AFOSR grants to the NAIL project: Morris [1991], Mumick [1991], and Ross [1991c].

### Bibliography


Brodsky, A. and Y. Sagiv [1991]. "Inference of inequality constraints in logic programs," *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 227–240.

Chaudhuri, S. [1991]. "Detecting redundant tuples during query evaluation," *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 115–126.

Chaudhuri, S. and M. Y. Vardi [1992]. "On the equivalence of datalog programs," submitted to PODS, Dec., 1992.

Gupta, A. and I. S. Mumick [1992]. "Magic sets transformation in nonrecursive queries," submitted to PODS, Dec., 1992.

Jakobsson, H. [1991]. "Mixed approach algorithms for transitive closure," *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 199–205.

Jakobsson, H. [1992]. "On tree-based techniques for query evaluation," submitted to PODS, Dec., 1992.

Morris, K. [1991]. "Subgoal order for query optimization in logic databases," Doctoral thesis, Dept. of CS, Stanford Univ., Aug., 1991.

Mumick, I. S. [1991]. "Query optimization in deductive and relational databases," Doctoral thesis, Dept. of CS, Stanford Univ., Dec., 1991.

Mumick, I. S. and H. Pirahesh [1991]. "Overbound and right linear queries," *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 127–141.

Phipps, G. [1991]. *GLUE Manual*, STAN–CS–91–1379, April, 1991.

Phipps, G., M. A. Derr, and K. A. Ross [1991]. "Glue-Nail: a deductive database system," *ACM SIGMOD International Conf. on Management of Data*, pp. 308–317.

Ross, K. A. [1991a]. "Modular acyclicity and tail recursion in logic programs." *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 92–101.

Ross, K. A. [1991b]. "On negation in HiLog," *Proc. Tenth ACM Symposium on Principles of Database Systems*, pp. 206–215.

Ross, K. A. [1991c]. "The semantics of deductive databases," Doctoral Thesis, Dept. of CS, Stanford Univ., Sept., 1991.

Ullman, J. D. [1991a]. "A comparison between deductive and object-oriented database systems," invited paper, *Proc. Second Intl. Conf. on Declarative and Object-Oriented Databases*, Munich, Germany, Dec., 1991, pp. 263–277.

Ullman, J. D. [1991b]. "Fast algorithms for deductive database queries," to appear in *Proc. NEC Conf. on Algorithms and Architectures*, Tsukuba, Japan, Aug., 1991.

Ullman, J. D. [1991c]. "The interface between database theory and language theory," in *Theoretical Studies in Computer Science*, a volume in honor of Seymour Ginsburg's 64th birthday, pp. 133-152, Academic Press, Cambridge.

Van Gelder, A., K. A. Ross, and J. S. Schlipf [1991]. "The well-founded semantics for general logic programs," *J. ACM* **38**:3, pp. 620–650.