During this research project, we have investigated facets of the next generation DBMS, POSTGRES. Specifically, we have worked on four of its aspects, namely:

1) the data model and query language
2) the rule system
3) a novel no-overwrite storage manager
4) execution parallelism

and we briefly report on our accomplishments in each of these areas.

92-00824
Data Base Support for Complex Objects and Expert Systems

by

Michael Stonebraker

July 31, 1991

U.S. Army Research Office

DAAL03-87-0083

University of California

Approved for Public Release;

Distribution Unlimited.
Final Report

Data Base Support for Complex Objects and Expert Systems

During this research project, we have investigated facets of the next-generation DBMS, POSTGRES. Specifically, we have worked on four of its aspects, namely:

1) the data model and query language
2) the rule system
3) a novel no-overwrite storage manager
4) execution parallelism

and we briefly report on our accomplishments in each of these areas.

Our initial proposal for a data model was presented in [ROWE87], and represented our approach to adding object management capabilities to a relational data model. The features of our data model with regard to extendability were indicated in [STON88]. Furthermore, a refinement of this data model in the area of complex objects was suggested in [JHIN88] and performance considerations of the various options analyzed in [JHIN90].

We have also made significant progress in the integration of a rules system and a DBMS. Our first rules system was reported in [STON87B]. After nearly completing the implementation, we elected to revise the rules system and a sketch of a second rules system was presented in [STON89] and a complete design in [STON90]. We have now completed the implementation of the second rules system and are preparing a report on its capabilities and performance.

The design of our no-overwrite storage manager is suggested in [STON87], and we have constructed it essentially as indicated.

With regard to parallelism, we presented our approach to utilizing the capabilities of a shared-memory multiprocessor in [STON88]. This proposal combined parallel execution of queries with a novel file system which striped files over a collection of disks and ensured that I/O parallelism also occurred. Our query optimization strategy is now implemented and a report on its design has been submitted for publication [HONG90]. A further report on the performance of our prototype will be forthcoming in a few months.

These ideas have all been integrated into a next generation DBMS, POSTGRES. We have released three versions of the code to external users and are about to release the fourth one. POSTGRES is getting relatively stable and performance is improving rapidly. A commentary on the design of the system appears in [STON90B], and a paper on its current capabilities with performance numbers on two popular benchmarks appears in [STON91].


