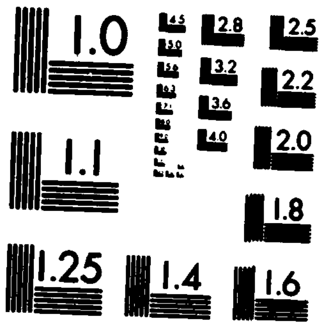
AD-A125 168 DESIGN OF A UNIVERSAL RELATION DATABASE SYSTEM(U) 1/1
STANFORD UNIV CA DEPT OF COMPUTER SCIENCE J D ULLMAN
01 SEP 82 AFOSR-TR-83-0020 AFOSR-80-0212

UNCLASSIFIED F/G 9/2 NL

END
DATE
FILMED
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AFOSR-TR- 83-0020	2. GOVT ACCESSION NO.: AD-A125168
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): "DESIGN OF A UNIVERSAL RELATION DATABASE SYSTEM,"	5. TYPE OF REPORT & PERIOD COVERED: Annual Report 9/1/81 - 8/31/82
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Professor Jeffrey D. Ullman	8. CONTRACT OR GRANT NUMBER(s): AFOSR 80-0212
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Computer Science, Stanford University, Stanford, CA 94305	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 61102F / 2304/A2
11. CONTROLLING OFFICE NAME AND ADDRESS: United States Air Force, Air Force Office of Scientific Research /NM, Building 410, Bolling Air Force Base, DC 20332	12. REPORT DATE: 9/1/82
	13. NUMBER OF PAGES: 4
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

DTIC ELECTE MAR 0 2 1983 D E

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

While the relational model removes some of the task of navigation in the database from the user, it still leaves navigational responsibility with the user when multirelational queries are involved. To remove this responsibility from the user, we have begun implementing a universal relation user interface, that allows the user to see the data as one large relation, i.e., table, over all the attributes. Since there may be more than one connection among the attributes mentioned in a query, we have developed a fairly complex theory to allow the database designer to force certain connections to be taken by the database system

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-LF-014-6601

UNCLASS.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD A125168

DTIC FILE COPY

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block Number 20 continued:

and to aid him by suggesting connections that we believe to be "natural." This theory includes the hypergraph representation of databases, explored last year, and the notion of acyclic hypergraphs, which are those with unique connections among attributes.

Accession For		
NTIS GRA&I	☒	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A		

DTIC COPY INSPECTED 2

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

AFOSR-TR- 83-0020

ANNUAL REPORT ON GRANT AFOSR-80-0212

Design of a Universal Relation Database System

Sept. 1, 1981—Aug. 31, 1982

SUMMARY

While the relational model removes some of the task of navigation in the database from the user, it still leaves navigational responsibility with the user when multirelational queries are involved. To remove this responsibility from the user, ~~we~~ The authors have begun implementing a *universal relation user interface*, that allows the user to see the data as one large relation, i.e., table, over all the attributes. Since there may be more than one connection among the attributes mentioned in a query, ~~we~~ They have developed a fairly complex theory to allow the database designer to force certain connections to be taken by the database system and to aid him by suggesting connections that ~~we~~ The authors believe to be "natural." This theory includes the hypergraph representation of databases, explored last year, and the notion of acyclic hypergraphs, which are those with unique connections among attributes.

Acyclic Structures

In the previous grant year, we began studying the hypergraph model of relational databases, where the nodes are attributes, and the (hyper)edges are the sets of fundamental relationships among certain sets of attributes that define the structure of the universal relation. For example, if we define our universal relation over attributes $ESDM$ (Employee, Salary, Department, Manager) as

$$\{esdm \mid \text{employee } e \text{ makes salary } s,\ e \text{ works in department } d, \text{ and department } d \text{ has manager } m\}$$

then the objects are ES, ED, and DM, corresponding to the stated relationships between employees and their salaries, and so on. In this case, all edges are sets of two nodes, but sets of three or more nodes are occasionally necessary. The hypergraph for this database is the following.

This hypergraph has a very simple structure, but more elaborate examples may have cycles and other complexity.

The basic theory was expounded last year, in [FMU], which has just been published in a journal. We also studied *acyclic hypergraphs* last year. These are tree-like hypergraphs that can be totally consumed by the steps:

1. If a node is in only one edge, delete the node.

83 02 028 150

Approved for public [illegible]ase;
distribution unlimited.

2. If one edge is a subset of another, delete the smaller edge.

Our basic result showing that acyclic hypergraphs in the above sense are exactly those for which unique connections among sets of attributes exist was published this year [MU1]. We also revised, to make clearer the underlying principles, the paper [MU2], which describes the "maximal object" approach to answering queries, in which a universal set of attributes is partitioned into overlapping sets of objects, each of which is acyclic and therefore allows only one connection among any set of attributes. System/U, our universal relation system using this approach, then answers a query by giving the unique connection in each maximal object that includes all relevant attributes.

Another "old" idea that got published during the past year is [BK], where a way to extend the universal relation concept to include "isa" relationships between attributes is given. For example, we could ask for the salary of Jones' manager, and obtain the result from the above database by realizing that although salary is related most directly to employees, not managers, every manager is an employee, so after finding Jones' manager by the connection through departments, we can take that manager name, treat it as an employee name, and find its salary through the ES object.

System/U Implementation

We completed the design and implementation of an algorithm that translates queries over the universal relation into a parse tree of ordinary relational operations. Begun, but not yet completed, are pieces of the system that translate the parse trees into (nearly) optimal sequences of evaluation steps and that translate the evaluation sequences into operations on UNIX files.

The algorithm used to translate queries was described in [U1]. Briefly, the Data Definition Language allows us to set up a sensible set of maximal objects, giving advice to the designer but allowing itself to be overridden. Once the maximal objects are established, each query is translated by the following steps.

1. Depending on the number of tuple variables involved in the query, the system starts by imagining that the query is applied to the Cartesian product of one or more copies of the universal relation. For each copy, there is a set of attributes X_i that the query involves.
2. For each i, the i^{th} copy of the universal relation is replaced by the union of all the maximal objects that include X_i.
3. Each maximal object is replaced by the natural join of all the objects it contains.
4. The natural joins are "pruned" to eliminate objects that don't connect attributes in X_i. The process, logically, is like the reduction test for acyclic hypergraphs described above, but with a modification that forbids the elimination of an attribute in X_i, even if that attribute appears in only one object of

the maximal object. The actual algorithm to reduce the number of join terms is an efficient two-pass operation described in [K].

5. Selections and projections associated with the query are applied to each of the resulting terms.

Defending the Universal Relation Concept

Our belief is that the universal relation concept has promise as a user interface; indeed the primary goal of this research is to realize that promise. We also believe that the concept is essential to describe the meaning of data dependencies among attributes that are no associated with a single relation. It is therefore quite disturbing to find published in (presumably) refereed journals, papers that are nothing but attacks on the concept. Further, these attacks are full of illogic, and we have yet to see a point that we feel is a valid reason to drop the universal relation from consideration as a user interface.

As an example of the issues with which we have had to contend, W. Kent published a criticism in the March issue of *ACM Trans. on Database Systems.* Among other claims, Kent offers an example of an intuitive semantics for a database involving landlords, buildings, and tenants. He then gives a formal semantics in terms of functional and multivalued dependencies. Next, he (correctly) derives a consequence of these dependencies, using the universal relation assumption as a justification for combining them. Then, he (correctly) points out that the derived dependency violates the intuitive semantics, and he points the finger of blame at the universal relation assumption.

The only problem is that Kent's informal and formal semantics do not agree. The error was not in the universal relation assumption, but in his presumption that the dependencies he wrote down mean what he said they mean; they do not.

To try to get both theoreticians and practitioners to understand the issues, [U1] deals not only with the design of System/U, but with an explicit rebuttal to some arguments put forth by the Kent article and others. Further, [U2] is a critique and, we believe, counterargument to every criticism of the universal relation idea raised in Kent's article.

Works Supported by the Grant

[BK] Beeri, C. and H. Korth, "Compatible attributes in a universal relation," *Proc. ACM Symposium on Principles of Database Systems*, pp. 55–62, March, 1982.

[FMU] Fagin, R. L., A. O. Mendelzon, and J. D. Ullman, "A simplified universal relation assumption and its properties," *ACM Transactions on Database Systems* 7:3, pp. 343–360, Sept., 1982.

[K] Kuper, G., "An algorithm for reducing acyclic hypergraphs," STAN-CS-822-892, Dept. of C. S., Stanford Univ., Jan., 1982.

[MU1] Maier, D. and J. D. Ullman, "Connections in acyclic hypergraphs," *Proc. ACM Symposium on Principles of Database Systems*, pp. 34-39, March, 1982.

[MU2] Maier, D. and J. D. Ullman, "Maximal objects and the semantics of universal relation databases," STAN-CS-81-878, Dept. of C. S., Stanford Univ., Oct., 1981.

[U1] Ullman, J. D., "The U. R. strikes back," *Proc. ACM Symposium on Principles of Database Systems*, pp. 10-2, March, 1982.

[U2] Ullman, J. D., "On Kent's 'Consequences of assuming a universal relation'," submitted to *ACM Trans. on Database Systems*, May, 1982.

Personnel Supported by Grant

1. J. D. Ullman (faculty), 25%, Oct., 1981-June, 1982.
2. G. Wiederhold (faculty), 30%, Aug., 1982.
3. G. Kuper (research assistant), 100%, Sept., 1981, and July-Aug., 1982; 50%, Oct., 1981-June, 1982.
4. J. Feigenbaum (research assistant), 50%, Oct., 1981-June, 1982.
5. A. VanGelder (research assistant), 50%, April-June, 1982.
6. D. Maier (consultant), 4 days, May, 1982.

END

DATE
FILMED

3-83

DTIC