

AD-A090 949

BROWN UNIV PROVIDENCE RI DEPT OF COMPUTER SCIENCE  
INTERACTIVE COMPUTER GRAPHICS.(U)  
OCT 80 A VAN DAM

F/6 9/2

N00014-75-C-0427

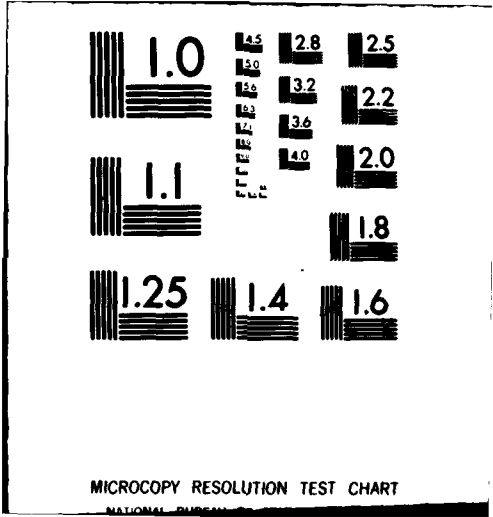
NL

UNCLASSIFIED

1 of 1  
12/80



END  
DATE  
FILMED  
12-80  
DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

LEVEL II

B.6 (12)

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

AD A 090949

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER N/A	2. GOVT ACCESSION NO. AD-A090949	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) (6) Interactive Computer Graphics		5. TYPE OF REPORT & PERIOD COVERED final report; 1/1/75-6/30/80
7. AUTHOR(s) (10) Andries van Dam		6. PERFORMING ORG. REPORT NUMBER 0
9. PERFORMING ORGANIZATION NAME AND ADDRESS Brown University Providence, RI 02912		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0427
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS (15) N00014-75-C-0427		11. CONTROLLING OFFICE NAME AND ADDRESS (9) Final rept. 1 Jan 75 - 30 Jun 80
12. REPORT DATE (11) 24 Oct 80		13. SECURITY CLASS. (of this report)
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) (12) 11		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE DTIC ELECTE OCT 29 1980 E
16. DISTRIBUTION STATEMENT (of this Report) Distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) Report of work done on Interactive Computer Graphics at Brown University during the period 1/1/75-6/30/80 under contract No. N00014-75-C-0427 or		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) computer graphics, distributed processing, host/satellite, interactive computer graphics, division of labor		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The goal of our research in distributed processing is to evolve fundamental concepts as well as implementation techniques for improving system performance by judicious runtime distribution of functions to processors and to levels of a hierarchical operating system. This distribution of computing tasks among the various processors in a network or among the many levels of a hierarchical operating system in a single computer is a critical issue in current and future military, industrial and educational computer		

DDC FILE COPY

80 10 27 058

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

↓  
systems. Our approach is to provide the applications programmer with tools that allow both the estimation of performance improvements of potential migrations, either horizontal (inter-processor) or vertical (intra-processor), and the implementation of the most promising migrations.  
↑

<b>Accession For</b>	
NTIS Global	<input checked="" type="checkbox"/>
BDC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist.	Avail and/or special
A	

**Interactive Computer Graphics Research**

**Final Report**

*Office of Naval Research  
Information Systems Division*

Contract N00014-75-C-0427

*Numerical*

**Principal Investigator:**

**Andries van Dam**  
**Department of Computer Science**  
**Box 1910, Brown University**  
**Providence, Rhode Island 02912**  
**401/863-3300**

**Date Submitted: 24 October 1980**

## Table of Contents

1. Introduction .....	1
2. Summary of Work Performed .....	1
3. Future Directions .....	5
4. Publications, Reports, and Theses .....	6
4.1. Publications .....	6
4.2. Reports .....	6
4.3. Theses .....	6
5. Personnel .....	7

## **1. Introduction**

The research reviewed in this report was supported by the Office of Naval Research as a continuation of Contract N00014-67-A-0191-0023. It was initially funded for the period 1 January 1975 to 31 December 1975. Subsequent renewals were funded for the following periods: 1 January 1976 to 31 March 1977, 1 April 1977 to 31 March 1978 (under the title *Performance Improvement through Function Migration in Distributed Computing systems*), and 1 April 1978 to 31 March 1979 (under the title *A Methodology for Vertical Migration in Layered Hardware/Firmware/Software Systems*). A no-cost extension was granted for the period 1 April 1979 to 30 June 1980 in order to complete a number of experiments and to produce this report.

In addition to the support received for this work from the Office of Naval Research, we were also funded under joint sponsorship from the National Science Foundation, Division of Computer Research, Grant MC576-04002.

## **2. Summary of Work Performed**

The goal of our research in interactive computer graphics and distributed processing has been to evolve fundamental concepts as well as implementation techniques for improving system performance by judicious runtime distribution of functions to processors and to levels of a hierarchical operating system. Our approach to this problem has been to provide the applications programmer with tools that allowed both the estimation of performance improvement of potential migrations, either horizontal (inter-processor) or vertical (intra-processor), and the implementation of the most promising migrations. Our concentration has been on tools (models, mechanisms, and implementation conventions) that have been efficient and as easy to use as possible, and preferably largely automated. It has been our intention to provide maximum design flexibility during the application *implementation* as well as to enable extensive tuning during *production*.

use of the resulting applications system. The methodology developed as a part of this research allows the programmer to write programs for a multiprocessor configuration as if it were a virtual uniprocessor. This view is similar to that of a virtual memory system, which hides the complexities of a memory hierarchy.

Our work under this contract has been directed at two major topics: the development of an inter-connected processing system for the Brown University Graphics System (BUGS), and the development of system performance and structure analysis tools that aid in the migration of functions between and within BUGS processors. The individual results of this work have been described in detail in the various publications, reports, and theses listed in Section 4. In the paragraphs below we will briefly highlight broad research areas addressed in our work, and provide a correlation between these and the particular documents referenced. Interested parties should also consult the various progress and annual reports produced during the course of this work.

Our major emphasis during the first years of this contract was on examination, revision, and measurement of the prototype Inter-Connected Processing system (ICOPS) for our host/satellite configuration. After the prototype system, which ran a geometric layout application program, was successfully demonstrated work proceeded at two levels. First, minor enhancements to expand usability of the basic facilities were considered and implemented. These included revision of the statistics-gathering facilities and extension of the satellite loader to enable dynamic loading of activated modules from disk. Second, a concurrent effort was begun to evaluate the system's performance in detail. This involved use of the revised statistics-gathering facilities to obtain measurements of system parameters, both of ICOPS and the application program, in a variety of host/satellite load environments. This work is described in detail in reference (3) of Section 4.1.



Our work on module migration using ICOPS and our interest in the movement of functions from software to firmware resulted in the initiation of a detailed study concerning the migration of functions among all software, firmware, and hardware levels in a single-processor system. This work resulted in the Structure Analysis System (STRUCT) for gathering and ergonomically presenting performance data associated with this type of migration. Significant progress was made towards both a better understanding of fundamental problems involved in horizontal and vertical migration, and the development of tools needed for analyzing these problems and actually performing migrations.

The ICOPS was used to run a number of scenarios for a real-time (graphics) applications program under varying loading conditions of the IBM 360/87 time-shared host. Runtime execution data gathered was plugged into a "max-flow, min-cut" commodity network graph model developed by our research partner, Professor Harold Stone at the University of Massachusetts, and implemented at Brown. The optimal distribution of modules between the host and the runtime satellite was then calculated for varying load conditions, and experimentally verified by runtime migration and data gathering. We feel that we have essentially solved the two-processor distributed computing problem, using our unique ICOPS tools. These consist of an ALGOL-W compiler for both host and satellite which can make symbol table information available, a runtime monitor for communicating between modules (both inter- and intra-processor), gathering statistics and performing migrations, and a network analysis program for predicting optimal distributions. This work is described in detail in (4) of Section 4.1.

In the area of vertical migrations the STRUCT system was used for analyzing our satellite's operating system to improve its performance in the context of several frequently used applications programs. We developed a methodology for systematically examining candidates for migration and then predicting the performance improvement for the most promising candidates, primarily based on

having the system compute Instruction Fetch and Decode overhead which could be saved. We also investigated the types of interactions between migrations which occur because the CPU-use of levels in a hierarchy changes drastically when functions in lower levels on which they depend are migrated; because of these interactions, vertical migration is intrinsically and substantially more difficult than horizontal migration. Our investigations into vertical migration were documented in (1) and (5) of Section 4.1, and (2) of Section 4.3.

Our work with the vertical migration of functions led to a general investigation into the relationship between good system structure and system performance. Building upon our previous efforts, predictive measures were developed for accessing the consequences of altering the structure of a system from "bad" to "good", both in terms of the complexity of the system and its transformation cost. In addition, a multi-level performance model was developed to extend the utility and understanding of vertical migration, and to prove several results concerning the various performance improvements attainable. This work was described in detail in (7) of Section 4.1, (9) of Section 4.2, and (3) of Section 4.3.

Finally, in the area of high-performance graphics, we produced an extremely powerful hierarchical picture data structure which our homegrown matrix and data structure processor, the SIMALE, can interpret at thirty frames per second to achieve full motion dynamics for well over a thousand 2D, 3D, or 4D vectors, as well as "logical zooming" via "extents." Extents are alternative representations of picture components, with varying levels of detail as a function of the amount of magnification of the total picture. Real-time user-controlled animation of complex scientific subjects is facilitated by using extents. This set of programs and devices were detailed in the various reports listed in Section 4.2.

### **3. Future Directions**

After completing the work supported by the series of grants listed above, we have essentially concluded our investigation into methodologies and tools supporting horizontal and vertical migration. While the tools we have developed to date are not transportable in their current implementation, our experiments have proven that the ideas embodied in them and the methodologies for their use are sound, effective, and "transportable" at the algorithmic level. We are now turning towards other problems which make use of multiprocessor and multilevel computer systems supporting interactive graphics. For example, we are studying the problem of abstracting system properties of digital systems (i.e., hardware and software systems) in a form suitable for simulation in an animated environment. This work will support the definition and verification (through the use of the animations) of possibly asynchronous systems at multiple levels of detail.

#### **4. Publications, Reports, and Theses**

##### **4.1. Publications**

- (1) "STRUCT Programming Analysis System", J. Stockenberg and A. van Dam, *IEEE Transactions on Software Engineering*, SE-1, 4, December 1975.
- (2) Transcripts from the Brown University Distributed Processing Workshop", *Computer Architecture News*, Vol 5, 5, December 1976.
- (3) "Experience with Distributed Processing on a Host/Satellite Graphics System", J. Michel and A. van Dam, *ACM SIGGRAPH 10*, 2, July 1976.
- (4) "Evaluation of Performance Improvement in Distributed Processing", J. Michel and A. van Dam, presented at the Brown University Distributed Processing Workshop, August 1977.
- (5) "Firmware Engineering: A Methodology for Vertical Migration in Layered Hardware/Firmware/Software Systems", J. Stockenberg and A. van Dam, *IEEE Computer* Vol 11, 5, May 1978.
- (6) "Research Directions in (Cooperative) Distributed Processing", J. Stankovic and A. van Dam, in *Research Directions in Software Technology*, P. Wegner, Ed., MIT Press, 1979.
- (7) "Good System Structure: It's Complexity and Transformation Cost", J. Stankovic, submitted to *IEEE Transactions on Software Engineering*, 1979.
- (8) "Horizontal/Vertical Migration", J. Stockenberg and A. van Dam, Invited paper in Firmware Engineering session at 1981 NCCC.

##### **4.2. Reports**

- (1) "IOCPDS Application Development Manual", J. Michel, Project Manual, March 1975.
- (2) "Hal's Useful Graphics System (HUGS) -- User's Guide", J. Michel, Project Manual, March 1975.
- (3) "Language for System Design (LSD II) Reference Manual", I. Carlborn, June 1975. "SIMALE Assembler, User's Guide", R. W. Burns, January 1976.
- (4) "ALGOL-W Reference Manula", C. Sorgie, Project Manual, July 1976.
- (5) "Programmer's Guide to ALGOL-W", H. Koslow, Project Manual, August 1976.
- (6) "Floating Point Processor for BUGS", Pal Verebely, Project Manual, August 1976.
- (7) "The SIMALE Standard Graphics Package", Preliminary Description, R. W. Burns and H. H. Webber, December 1976.
- (8) "Brown University Graphics System (BUGS) Overview", Project Manual, updated December 1976.
- (9) "Structured Systems, and Their Performance Improvement Through Vertical Migration", J. Stankovic, Brown University Technical Report CS-41, August 1978.

##### **4.3. Theses**

- (1) "An ALGOL-W/380 to M4A Cross Compiler", C. Sorgie, Master's Thesis, Brown University, June, 1976.
- (2) "Optimization through Migration of Functions in a Layered Firmware/Software System", J. Stockenberg, Ph. D. Dissertation, Brown University, June 1977.

- (3) "Structured Systems, and Their Performance Improvement Through Vertical Migration", J. Stankovic, Ph. D. Dissertation, Brown University, June 1978.

**5. Personnel**

**Principal Investigator:**  
Andries van Dam

**Faculty Associates:**  
Thomas Doeppner  
Charles Strauss  
Robert Sedgewick

**Professional:**  
Russel Burns  
Harold Webber  
Janet Michel  
Josiah Strandberg

**Graduate Research Assistants:**  
Dick Bulterman  
Ingrid Carlborn  
Paul Hanau  
David Irvine  
Kenneth Magel  
Jack Stankovic  
John Stockenberg  
Charles Sorgie  
David Taffs

**Undergraduate Research Assistants:**  
John Crawford  
Kenneth Field  
Steven Feiner  
Sidney Gudes  
David Holland  
Howard Koslow  
Janet Levitt  
Steven Lewis  
Andrew Malis  
Craig Mathias  
Robert Schiavone  
John Zahorjan