Computers in Chemical Education Newsletter

Fall 1992

Published by the ACS Division of Chemical Education
Committee On Computers in Chemical Education
Don Rosenthal and Al Latta co-chairs

Editor Brian Pankuch, Department of Chemistry, Union County College, Cranford, NJ 07016

Submissions: General articles should be sent to editor Brian Pankuch at the above address. We would appreciate both 1) printed copy (hardcopy) and 2) a readable file on a Macintosh or IBM compatible 3 1/2" diskette. We have fewer problems with 3 1/2 " diskettes.

Submission deadlines: Fall issue - Sept. 25; Spring issue - March 25.

Some Supported Word Processors	File Format
DCA files	.DCA
Microsoft Windows Write	.WRI
Microsoft Word	.DOC
MultiMate	.DOC
WordPerfect	.WP
WordStar 3.3	.WS
XyWrite III	.XYW

For example if you use WordPerfect to make a file named foo send it as foo.WP. FOR OTHER WORD PROCESSORS SEE Aldus PageMaker 4.1. We can read ASCII FILES.

ALL NEW AND RENEWAL SUBSCRIPTIONS : PLEASE SEND REMITTANCE TO M. Lynn James, Department of Chemistry, University of Northern Colorado, Greeley, CO 80639.

RATES: USA 1 year $2.50, two years $4.50: Other countries 1 yr $5, two yr $9. Please make a check or money order payable in US funds to Computers in Chemical Education Newsletter. One to three issues are published per year.

Consulting Editor Donald Rosenthal, Department of Chemistry, Clarkson University, Potsdam, NY 13676. Send meeting notices, etc., to Don., ROSEN@CLVM.BITNET.

Managing Editor Henry R. Derr, Laramie County Community College, Cheyenne, WY 82007 HDERR@Corral.UWYO.EDU.

Contributing Editors: Send your contribution on specific areas to the appropriate editor.

For hardware Queries and Replies. Jim Beatty,IBM and compatibles Chemistry Dept., Ripon College, Ripon, WI 54971, BEATTYJ@ACAD.RIPON.EDU.

For software Queries and Replies. Ken Loach, Department of Chemistry, SUNY College, Platts-burgh, NY 12901. BitNet address: Loachkw@snyplaba.

For Book Reviews. Larry M. Julian , Chemistry Department , Michigan Technological University, Houghton, MI 49931. lmjulien@mtu.edu.

Cover: Playing with a picture, courtesy of T.C. O'Haver, see Windows and Networks.

The newsletter is done using Aldus PageMaker 4.,2 and is printed using a LaserWriterII.

Contents

FROM THE CHAIR

Alfred J. Lata, Co-Chair

In the past three years,there have been great and dramatic changes in computers, hardware, software, and computing activities, and we have been fortunate to have reaped (or look forward to reaping) the benefits of these changes.

There are faster CPUs, larger memories, larger capacity hard drives, as well as video-disk, CD-ROM, etc. The software revolution is led by multimedia and animation programs, the growth of object-oriented programming and the use of objects in our programs. The use of e-mail for personal communication and bulletin boards has grown tremendously both in number of users and volume of traffic. Applications of all of these factors have been incorporated into advances in Chemical education.

There still exists that broadening chasm between the haves and the have-nots in the use of computers in the teaching of Chemistry. Some are limited by lack of funds for the purchase of hardware and software. As prices come down, and the use of computers for teaching and learning is more broadly accepted and applied, hopefully this limitation will be removed.

But there is further limitation: the dissemination of information about what software is available and appropriate for instruction, and how it is used.

At the 12th Biennial ChemEd Conference in Davis in August of this year, the Committee on Computers in Chemical Education (CCCE) held an open meeting. Twenty five Conference attendees joined us to discuss the use of computers in chemistry instruction. A principal concern voiced at this meeting was how people can learn what software is available and how information on available software suitable for use in chemistry can be most effectively disseminated.

In an effort to solve this information problem, I would hope that each of us individually will undertake to broadcast information about those programs we use and like (and dislike), and how we use them. Conferences, regional meetings, electronic bulletin boards, and THIS NEWSLETTER can be effective tools in this regard. Consider it your personal challenge to help your colleagues! How about submitting an article or just a paragraph or two describing software you use and how you or your students use it, the price and where to purchase it? Submissions should be sent to Brian Pankuch (see the inside front cover for details).

I would like to take this opportunity to recognize two of our colleagues Paul Cauchon of The Canterbury School (New Milford CT) has stepped down from CCCE. Paul has served CCCE as Chairman, and has participated as instructor in more of our workshops than anyone else. We appreciate and thank him for his work as one of the pioneers in Computers in Chem Ed.

Don Rosenthal, co-chair, will assume the individual responsibility this coming year. Don has organized and conducted workshops for CCCE, was editor of this Newsletter, and now serves as Consulting Editor. Those of us who have the opportunity to work with him know how effective and persuasive he can be. I thank him for his past efforts and accomplishments for CCCE, and look forward to his leadership.

HOW TO CRASH HARD DISKS AND SYSTEM SOFTWARE WITHOUT REALLY TRYING.

Editor Brian Pankuch

(For people who may have to take care of their own Mac.)

Actually the idea was to minimize problems with hard disks. I've been thinking for some time that I need a more effective way of protecting my work than backing up to a diskette. I'm conscientious about backing up big projects, but I have a lot of small things I do often and don't have a special diskette to back up to. With over 200 M on my system, I'm just not going to back the whole thing up on diskettes, nor am I going to set up a filing system to find a specific diskette (out of 200 or more) for a small piece of work. The easiest solution seemed to be to get a 88 M removable Sysquest cartridge and drive, about $500 for both. This would also give me a backup hard disk. Additional 88 M cartridges are about $95, so you really have backup limited only by your pocketbook.

On the other hand I've been using computers for quite awhile and have never had a hard disk problem. I've heard of plenty, but I've got the best quality hard disks and have been lucky. Also I like a bargain and I got an outstanding mail offer from Symantec for their Norton Utilities and MORE, a presentation manager, for about 85% off the list price. Symantec combined the best of Norton and their own SUM II utilities to do preventive medicine on hard disks and to recover lost files on a crashed disk. Total price was $95. This didn't give me the same protection as a spare hard disk, but it should lessen the chances of problems. If problems do occur I

have a way of recovering files. It is cheaper and probably easier than backing up 88 M often.

So I bought the Symantec software rather than the hardware. I set it up on my Quadra 700. Installation of Norton was easy. To use some of the utilities such as Disk Doctor was straight forward. The utility chugged through the hard disk and found a number of anomalies- some programmers amuse themselves by giving the date of completion of a program 1909 or 2040. Harmless but the Norton disk doctor fixed these and other anomalies.

Next day I decided to use Speed Disk, which optimizes the disk. Files on your disk are usually written to one physical area on the disk at first. As the file changes size the original space set aside may not be large enough. The part that doesn't fit is put into another physical location on the disk. After many repetitions your disk is fragmented. Parts of the same file are all over the disk. Reading the file from the disk will take longer since the reading head has to find each part. This also results in a lot of head movement with additional wear. Speed Disk will put files back into contiguous areas. It will also place files that don't change much in the beginning of the disk and the often modified files at the end where they can grow. This should save a lot of wear on the hard drive mechanism and put off eventual failure. My drive has an estimated mean time between failure of over 22 years, but only a 5 year warranty.

To use Speed Disk you have to run the utility from a disk other than the disk you are checking. Symantec thoughtfully supplied a boot disk for this purpose. I restart my Quadra with this disk, and it comes up in Speed Disk and tells me there is not enough memory. The Quadra has 20 M of RAM and access to another 300 M of virtual memory! I can't get at the disk to check because it won't let me out of Speed Disk. So I try to restart and the entire system crashes. I try everything I can think of and finally replacing the entire system file lets me restart the computer. I run Disk Doctor and it says everything is great, I restart just to see if it will, and again the whole system crashes. After about 6 hours I learn more about disk utilities than I really want to know.

My problems were first, the Speed Disk program didn't have enough memory set aside for itself, my Quadra had plenty, but programs aren't smart enough to go grab some. You change allotted memory by selecting the utility, then selecting Get Info from the file menu (Command I). When the Get Info window opens change the 600 K allotted to at least 1000 K. I changed mine to 14,000 K since I have a lot of memory — the more memory set aside the faster Speed Disk will optimize your hard disk (it can take hours). You can't change the allotted memory on the boot disk while you are using it because you can't get at the file. You also can't change this setting while you're using the program (this is true for all programs). Make the change while this diskette is not the startup disk.

The second problem is more general. Most of us use many utility programs (INITs) — clocks, calculators, etc., on the screen. I get a lot, and I test a lot. These utilities can change the System software that runs the computer. They frequently take over the same parts of memory. If not well designed they wipe out part of the system memory when they collide. After taking out many of these INITs that I no longer needed, and replacing and updating the system, everything works fine.

As an experiment, I ran Speed Disk on a Mac IIX running System 6.07 — it did exactly the same thing. The SAM virus utility (also made by Symantec) was causing the problem.

Speed Disk ran fine on both machines after removing the problem INITS. This all happened on a weekend so Symantec telephone help was not available. I called even though I had Speed Disk working. As a general preventative Symantec suggested prefixing the Norton file name with a 'z' so it loads last and lessens the chances of problems with other utilities.

MORE I've only started to use MORE, but as I mentioned previously the balloon help under the Mac System 7 makes exploring a lot easier and more interesting. Even in a well-designed program exploring is sometimes quite complicated. I find using balloon help much faster than trying to look through the 5 books (2 books and 3 booklets to be more precise) included with MORE. Also if I do need to look up additional detail in the reference manual, balloon help gives me the name of what I'm looking at. This is an enormous time saver in a complicated program.

This has little to do with chemistry, but tax prepa ration packages can be time savers. I used MacInTax two years ago and found it very helpful. All the tax forms come up on the screen with help available, and act like spreadsheets. If you add or subtract something, all amounts affected are updated. If you're 3 or 4 forms deep this can save a lot of drudgery. When it works you can automatically move names of banks, charities,

and other categories to next years form. Last year ChipSoft, a DOS oriented company, took over MacInTax. The new program was so bad I had to return it. Considering I didn't return Norton Utilities that's bad. I hope this year will be better. Less time for taxes may give you more time for students.

Wanted! by Larry M. Julien

Names of books to be reviewed. These can be titles of books that you have read or seen that are of interest. These books will be reviewed and published in the Newsletter.
Please send your suggestions to me at one of the following addresses.

Larry M. Julien
Chemistry Department
Michigan Technological University
Houghton, MI. 49931

or by electronic mail at

lmjulien@mtu.edu

Editor: I assume Larry will also be pleased to receive actual book reviews by readers.

COMPUTER CONFERENCE ON APPLICATIONS OF TECHNOLOGY IN TEACHING CHEMISTRY JUNE 14 THROUGH AUGUST 20, 1993.

by Don Rosenthal Co-Chair

The ACS Division of Chemical Education's Committee on Computers in Chemical Education is sponsoring a computer conference on "Applications of Technology in Teaching Chemistry." We hope that papers will be submitted on the use of computers, video, audio, films and other technologies which are used in teaching chemistry. The "meeting" will consist of several sessions each containing five papers.

This is the first computer conference held by the Division of Chemical Education and is viewed as an experiment. Participants and authors of papers will be asked to evaluate the "meeting" at the end of the conference. Thomas C. O'Haver, of the University of Maryland will manage the meeting and prepare a report summarizing the conference and the evaluations. It is expected that some of the papers, discussion and evaluation will be published in book form. It is hoped that computer conferencing will become a permanent part of the Division's program.

CALL FOR PAPERS

Authors must have access to Bitnet or Internet. The title of the paper and an abstract of not more than 150 words should be sent to Thomas C. O'Haver(TO2@UMAIL.UMD.EDU) via electronic mail before February 1, 1993. The full paper must be submitted by May 1, 1993. Assistance on handling graphics will be available through the conference manager. Authors will be sent instructions either prior to or after submission of an abstract. Submit your request to TO 2@UMAIL.UMD.EDU. Presenting a paper does not preclude publication; in fact we encourage subsequent publication.

REGISTRATION

There will be no registration fee. However, everyone who registers must make a commitment to participate in the conference. All participants must have access to either Bitnet or Internet and become familiar with electronic mail prior to the meeting.

Registrants will have an opportunity to practice prior to the conference. Assistance will be available through Thomas O'Haver. You will register by sending the following message to LISTSERV@UMDD.UMD.EDU : SUBSCRIBE CHEMCONF-L <your name> before June 1, 1993. The titles and abstracts will be available to registrants on March 1. Those wishing not to participate can notify LISTSERV by April 1.

Those having a terminal or work station and the necessary communication software can receive assistance in obtaining a link to Bitnet or Internet by writing Dr. Thomas C. O'Haver, Department of Chemistry, University of Maryland, College Park MD 20742 (Phone: 301-405-1831).

THE COMMITTEE ON COMPUTERS IN CHEMICAL EDUCATION WHAT IT IS AND WHO ITS MEMBERS ARE The PURPOSE of this Committee is to promote and publicize computing usage in chemical education. by Don Rosenthal Co-Chair

The PURPOSE of this Committee is to promote and publicize computing usage in chemical education.

COMMITTEE ACTIVITIES

The Committee:
Sponsors and Helps to Organize Symposia and Other Sessions at National and regional ACS Meetings Organizes National Computer Workshops
Publishes the COMPUTERS IN

CHEMICAL EDUCATION NEWSLETTER

Holds open and closed meetings at CHEM ED Biennial Meetings

Suggests and promotes new initiatives like computer conferencing

Maintains a closed electronic mail network (CCCENET) to facilitat communication among its members.

Committee members participate by:

Providing ideas

Providing suggestions

Speaking and writing

Organizing and participating in seminars, symposia, meetings and workshops

Acting as resource people for others

The Committee is anxious and willing to receive suggestions and comments from other chemical educators. Those wishing to communicate with the Committee can write or send electronic mail to Donald Rosenthal or any other member of the Committee. We welcome participation in any of our activities - articles for the Newsletter, symposium speakers and organizers of workshops and symposia. We also look for new Committee members from time to time. If you are interested in or would like to recommend someone for Committee Membership, please contact Don Rosenthal.

COMMITTEE MEMBERS

The Committee presently consists of twenty-one members. The membership directory follows listing name, electronic mail address, and phone number on the first line of each entry. The mailing address is shown on the following lines. Not all Committee members currently have electronic mail addresses.

James W. Beatty - BEATTYJ@ACAD.RIPON.EDU - 414-748-8123 Chemistry Department, Ripon College, Ripon, WI 54971

Bruce N. Campbell, Jr. CAMPBEBN@SNYPOTVX.BITNET - 315-267-2273 Department of Chemistry, SUNY Colege, Potsdam, NY 13676

Joseph Casanova-CASANOVA @CSULA.EDU - 213-343-2338 Department of Chemistry and Biochemistry, California State University, Los Angeles, CA 90032-8202

Wilmon B. Chipman CHIPMAN @TOPCAT.BSC.MASS.EDU - 508-697-1200 Ext. 2198 Department of Chemical Science, Bridgewater State College, Bridgewater, MA 02325

Henry Derr - HDERR @CORRAL .UWYO.EDU - 307-778-5222 Ext 129 Division of Life Physical Sciences, Laramie County Community College, 1400 East College Drive, Cheyenne, WY 82007-3299

William Halpern - WHALPERN @UWF.BITNET - 904-474-2741 Department of Chemistry, University of Western Florida, Pensacola, FL 32504

M. Lynn James - 303-351-1285 Department of Chemistry and Biochemistry, University of Northern Colorado, Greeley, CO 80639

Larry M. Julien - LMJULIEN @MTU.EDU - 906-487-2055 Department of Chemistry, Michigan Technological University, Houghton, MI 49931

Alfred Lata - LATA@UKANVAX. BITNET - 913-864-4054 Department of Chemistry, 2010 Malott Hall, University of Kansas, Lawrence, KS 66045

Kenneth W. Loach - LOACHKW @SNYPLAVA.BITNET - 518-564-4116 Chemistry Department, SUNY College, Plattsburgh, NY 12901

Diana Malone - 319-588-6334 Department of Chemistry, Clarke College, 1550 Clarke Drive, Dubuque, IA 52001

Thomas C. O'Haver - TO2@UMAIL.UMD.EDU - 301-405-183 Department of Chemistry & Biochemistry, University of Maryland, College Park, MD 20742-2021

Brian Pankuch - 908-709-7177 Department of Chemistry, Union County College, Cranford, NJ 07016

LeRoy J. Pazdernik - 819-376-5052 Departement de Chimie-Biologie, Universite de Quebec a Trois-Revieres, C.P. 500, Trois-Revieres, Quebec , CANADA G9A 5H7

Harry E. Pence - PENCE @SNYONEVA.BITNET - 607-431-3179 Department of Chemistry, SUNY College, Oneonta, NY 13820

Kenneth Ratzlaff - RATZLAFF @UKANVM.BITNET - 913-864-3754 Instrument Design Laboratory, Department of Chemistry, University of Kansas, Lawrence, KS 60045

Donald Rosenthal - ROSEN@CLVM. BITNET - 315-265-9242 Department of Chemistry, Clarkson University, Potsdam, NY 13699

Gwen Sibert - SIBERT@VTVM1. BITNET - 703-981-2116 Roanoke Valley Governor's School, 2104 Grandin Road, Roanoke, VA 24015
Allan L. Smith - ASMITH@DUVM. OCS.DREXEL.EDU - 215-895-1861 Chemistry Department, Drexel University, 32nd and Chestnut Streets, Philadelphia, PA 19104
Stanley Smith - SMITH @UIUCSCS.BITNET - 217-333-3839 Department of Chemistry, University of Illinois, 1209 W. California, Urbana , IL 61801

William J. Sondgerath - BSONDGER@PURCCVM.BITNET - 317-589-3877 Harrison High School, 5701 N. 50 W, West Lafayette, IN 47906

INTEGRATING COMPUTERS INTO THE HIGH SCHOOL CHEMISTRY CLASSROOM

William J. Sondgerath, Chemistry TeacherHarrison High School, West Lafayette, Indiana

Our project began in 1987 with a grant from the Indiana Department of Education to increase the use of technology in the chemistry classroom. Our purpose was to use technology to enhance traditional instruction and to increase learning opportunities for students. The project, "Classroom Without Walls," was comprised of Thomas Adams, then a doctoral candidate from Purdue University, and Dolores Handy and William Sondgerath, chemistry teachers at Harrison High School. There follows an outline of how we use computers and related technology in our classroom today.

We have the following hardware available:
In the classroom:
8-IBM Model/25
4-IBM Proprinter II
4-ESM Corporation printer switch boxes
1-IBM Model/30 (Mostly for teacher use.)
1-Sharp QA-50 Liquid Crystal Display
Outside the room:
1-Local Area Network (in media center)
30-IBM Model/25 (in media center)

Used extensively for tutorials in beginning chemistry is "Introduction to General Chemistry" by Stan Smith from the University of Illinois, published by Compress. Fifty-five lessons in twenty instructional days are used, covering such topics as nomenclature, writing formulas, per cent composition, solutions, the gas laws, and acid-base chemistry. KC? Discoverer by Journal of Chemical Education Software is used extensively during our study of periodicity. We have found in our experiences that it is essential for student learning to have teacher-prepared

LAB SPREADSHEETS PREPARED FOR THE FOLLOWING IN HEATH CHEMISTRY: D. C. Heath, 1987)

Experiment 1C: Analysis of Experimental Results
Experiment 2B: The Law of Definite Composition
Experiment 3A: Thickness of an Aluminum Sheet
Experiment 3Y: The Relationship Between the Mass & Volume of Cu
Experiment 2B: The Law of Definite Composition (Students use data in Ch4 to find Emperical Formula)
Experiment 6A: Mole-Mass Reaction
Experiment 7B: The Molar Volume of a Gas
Experiment 17A: Heat of Reaction
Experiment 17B: Heat of Fusion

Spreadsheets for Other Labs:

Calorimeter Constant
Specific Heat of Unknown Metal
Enthalpy of Neutralization Reaction
Acid-Base Titrations: Standardization of NaOH with KHP
Acid-Base Titrations: Citric Acid in Grapefruit Juice
Acid-Base Titrations: Acetic Acid in Vinegar

worksheets or guidelines to accompany these tutorials; otherwise, students proceed as if playing computer games!

Another important component is the liquid crystal display (LCD), used to project concepts which are being taught to the total class when software is not available or appropiate for small group or individual instruction. The LCD is used for introducing technology or equipment that the students will be using, such as spreadsheeting and graphing. The graphics program which is used to design "concept stories" is IBM's Storyboard Plus. Included is a list of teacher-made concept stories which have been developed with the project. Pictures, diagrams, definitions, and descriptions in a series of frames teach the following concepts.

1\Bonding
2\Density
3\Exo- and Endothermic
4\History of Atom
5\Mass Spectrometer
6\Forming Sodium Ion
7\Phase Diagrams (CO2 and H2O)
8\Colligative Properties
9\Molecular Model
10\Use of Buret
11\Heating Curve
12\Simple Orbitals
13\Part 1 Equilibrium (Qualitative)
Part 2 Equilibrium (Quantitative)

The concept stories can be utilized by the individual student, but usually they are used in large group instruction because they are not interactive.

Harrison High School has Microsoft Works on its network. Students use word processing for lab reports and summary reports on current topics in chemistry, but students are not required to word process throughout the course. Using/making spreadsheets and graphing is the aspect from Microsoft Works that is most used in our chemistry classes. Included is a list of spreadsheet templates which have been developed with the project.

One advantage of spreadsheeting is that students can readily relate a large collection of data to the concept the lab is teaching. Initially students enter individual data for the spreadsheet; they also have access to all other students' data as well. This allows them to understand the experiment more clearly. Eventually they construct the spreadsheet and learn to graph the data from it. The spreadsheet also makes it easier for the instructor to check the students' lab results.

Another feature of our technology is the purchasing of Personal Science Laboratory, which has improved the laboratory interfacing capabilities. The thermistor is used to determine such concepts as heating curves, heat of fusion, and heat of reaction. One major use of the pH probe is to plot the neutralizaion curves involving various combinations of strong/weak acids and strong/weak bases.

For the instructor the Excelsior Grade Book is a most useful and powerful management tool. Keeping students informed of their continual progress, whether for eligibility for extra curricular activities, or for their own curiosity, is most efficient. Questions on attendance can be answered quickly. Statistics on graded material can be determined instantaneously.

Through our project we have gained a better insight of how the tools of technology can enhance traditional instruction, providing still another method of teaching. A second advantage is that "Classroom Without Walls" increases students' opportunities to work with technology that many will face in future education and/or employment.

Sources of Software

Storyboard Plus from IBM
Microsoft WORKS for IBM PC's and Compatibles
PSL(Personal Science Laboratory) from IBM
COMPRESS(tutorials)
P.O. Box 102 Wentworth, NH 03282

(KC? Discoverer)
Journal Chemica Education Software University of Wisconsin-Madison, Dept of Chemistry 1101 University Avenue Madison,WI 53706

Excelsior Software, Inc.(grades)
P. O. Box 3416
Greeley, CO 80633

Can the Organic Laboratory be Computer Assisted?

Bruce N. Campbell, Jr., Department of Chemistry, Potsdam College, Potsdam, N.Y., 13676, campbebn@snypotvx.btnet

This is the question I posed at the 12th Biennial Conference on Chemical Education in August. It is the question I want to pose now to the readers of this newsletter.

In this brief paper, I would like to suggest some ways computers have been or could be used to make the Organic Laboratory experience more effective. My examples are drawn mostly from software available for the Macintosh®. If readers have suggestions or comments, I would like to hear from them and will prepare a future column summarizing such comments in this newsletter.

Some possible uses and software are given in Table I.

Table I
Software Samples

Graphing	Searchable Data Bases
CA-Cricket Graph	STN, HODAC
Graphical Analysis III	HSDB, TOXNET
Kaleidagraph	TOXLINE, TOXLIT

Drawing, Structure Building & Presentation		Nomenclature/Simple Reactions
ChemDraw	CHEMINTOSH	Beaker
LabVision	CHEMDRAFT	Jay Hawk
Molecular Editor	Desktop Molecular	Organic Rxns.
Alchemey III	Modeller	
MacStereo	Moby	
CHEMWINDOWS	Chem 3D Plus	

Spectral Information	Qualitative Organic Analysis
Instrumental Software	MacSQUALOR
SpectraDeck	MacQual
Proton NMR Simulator	
Mock NMR	

Math Expressions/Equations	Spreadsheets
MathType	Excel
Expressionist	Full Impact
MathCAD	Lotus 1, 2, 3
TK Solver Plus	Wingz

Table II
Preparing for Lab

Needs	Types/Names
Structures and Names	Beaker
Reactions Byproducts Mechanisms	Beaker
Stereochemistry	3D Drawing program/Spatial Display/Structure Building
Physical Properties, etc.	Searchable Data Base STN
Toxicity, Hazards, Disposal	Searchable Data Bases TOXNET, HSDB
Preliminary Calculations	Spreadsheet Excel, Lotus 1, 2, 3 Wingz, Full Impact

Table III During the Lab

NEEDS	TYPE/NAME
Collecting & processing data (personal) Collecting & processing data (class)	Spreadsheet Excel, Full Impact Lotus 1,2,3, Wingz
Electronic Notebook	Spreadsheet/Word Processing See above
Graphic data (in progress) Kinetics, Titrations Calibration curves, etc.	Graphics Graphic Analysis III CA Cricket Graph Kaleidagraph
Spectral data Matching Teaching Impurities	Instrument software, Spectra Deck, Mock NMR, Proton NMR Simulator
Dry Lab Teaching Stereochemistry Spectral Analysis	3D Drawing and Manipulating See above
Planning New Experiments Source of Experiments Synthesis	Searchable Data Bases Synthesis Planning Programs
Qualitative Organic Analysis Preparation Simulation	MacQual and MacSQUALOR
Waste Disposal	Searchable Data Base HSDB

Table IV After the Lab

NEEDS	TYPE/NAMES
Report Writing Calculations	Word Processing Spreadsheets Excel, Full Impact Lotus 1, 2, 3, Wingz
Equations Chemical	Drawing Beaker, ChemDraw CHEMINTOSH
Mathematical	MathCAD TK Solver Plus
Expressions	MathType
Plotting and Interpreting Data	Graphics Graphic Analysis III Cricket Graph Kaleidagraph

Tables II, III, and IV suggest some uses before, during, and after the lab.

Graphing results from kinetic experiments, titrations, and apparatus and instrument calibration during the lab will suggest how effective the work has been or whether it should be repeated. Searching for spectra to compare to observed spectra would give feedback as to how successful a synthesis had been. Some students might keep electronic notebooks.

Computer stored or generated spectra and molecular modeling programs could be used for preparatory assignments or dry labs. We have had students use qualitative organic analysis simulation to increase the efficiency and effectiveness of unknowns in the lab.

Students are asked to come to the Organic lab having done prior preparation. This preparation usually includes finding the chemical and physical properties of the substances to be used or prepared. Now we need to add to that, information on toxicity and methods of disposal. Computer searchable data bases could facilitate this process.

Participants at the conference in August were asked to fill out a related questionnaire. A few did. The questions were:

1. What software programs do you use in conjunction with teaching the Organic lab?
2. How would you like to use the computer in this activity?
3. Do or should your students have access to computers "in" the Organic lab as they do to IR and GC?
4. How would you like your Organic lab students to use computers?
5. What stands in the way of more use of computers in the Organic lab?

Responses for question 1 included different spectral programs (mentioned by all but one responder and one of these mentioned the Sadtler Library Search program), Beaker, word processing, and drawing programs. One respondent said they had a qual organic search program for compound identification containing data for 6000 compounds stored on their IBM mainframe.

Suggestions for question 2 included input data and check calculations or receive suggestions (i.e. error, possible compounds), analysis of gas chromatographic data, simulations, modeling, lab notebook, and to identify spectral unknowns before doing qual. The responder with the ID file said their students get a list of possible compounds. That sounds interesting.

Answers in response to question 3 range from no response through yes (several) to one lab which is connected by cable to mainframe with input from the lab.

Question 4 drew responses expanded from #2. They include: Preview lab set-up, compare and average class results (Good for discovery labs!), grade their own preparations with IR and NMR as resources, write lab reports, spectral searches and simulators, and using the library that comes with FT-IR.

In response to number 5, almost all said space and/or money. One threw in the administration. No one mentioned concern for damage to the computers. One honest responder spoke for all of us—"my inertia."

Don't let your inertia get in the way. Write today if these topics are of interest to you. If people are interested, I would be willing to act as a clearinghouse and seek to air your ideas and concerns (quality of software, software needed, ways to use them, successes, etc.) in print.

Windows and Networks: Lowering the Activation Energy for a Chemistry Department Microcomputer Facility.

T. C. O'Haver, Department of Chemistry and Biochemistry, University of Maryland, College Park, MD 20742 (to2@umail.umd.edu)

Abstract

A synergistic combination of a window-oriented user environment and high-speed local and wide area networking is the technological basis for a successful faculty-student computer workstation room in a university chemistry department. This paper describes the equipment, software, organization, operation, and application of the facility.

A. Introduction

How can a university chemistry department make the best use of modern personal computer technology to help support its teaching, research, and service mission? There is no question that computers can perform many useful tasks for the scholar and researcher, but how can we insure that the technology will be utilized in an effective way and will actually contribute to productivity? It is essential that the real benefits to the users be weighed against the increased burden of complexity that unavoidably accompanies any high technology development. Ours is a large research-oriented department of chemistry and biochemistry with 45 full-time faculty members, 200 graduate students, and 400 chemistry and biochemistry undergraduate majors. Students and faculty in our

department typically have no formal computer training, are not really interested in the computer per se, and are often too busy to read manuals. Computers are not part of the mainstream of our discipline, so time spent learning about the computer is often considered time wasted. The ideal system for our purposes would be powerful enough to meet most needs but relatively intuitive to operate, so users can bootstrap themselves into literacy without wasting too much time. We clearly needed to capitalize on the benefits of the personal computer revolution: namely, the wide range of user-oriented application software. It is important that the facility exhibit a low intimidation factor so as to reduce the energy barrier to computer use. Additionally, in our situation there is no possibility of a full-time computer administrator, so the computer facility would have to be relatively easy to maintain with existing personnel. Although many of our faculty had already purchased personal computers for their offices and labs, we felt that a public computer facility could serve most of the needs of graduate and undergraduate students and would provide a common area for sharing expensive peripherals such as high-quality printers, plotters, and scanners.

B. Application goals

One area of computer application that is useful for both students and faculty is scientific communication: writing papers, reports, and proposals, preparing posters and slides for presentations, and developing computer-based lecture demonstrations. Scientific papers often involve mathematical equations that are difficult to type with a typewriter or ordinary word processor and usually include line drawings and plots (graphs) of data or calculations. Chemistry writing tends to be especially complex typographically, with lots of Greek letters, math characters, subscripts and superscripts, italics and boldface, and drawings of chemical structures that are embedded into the text as an integral part of the document. These requirements suggest that a high degree of graphics capability will be desirable. Popular text editors designed for programmers and word processors designed for the business office do not meet all these requirements. The recent trend toward graphics-oriented microcomputer environments and "what you see is what you get" operation is very beneficial in this respect. We felt that this was even more important than raw operational speed and the "keep-your-hands-on-the-keyboard" ethic of products designed for the professional typist. We could meet this software requirement with either a single large technical document preparation application or with a set of separate applications that can work together to produce the finished product.

In addition to scientific document preparation, we expected to be able to use our computers to perform some of the more traditional scientific computer applications, including data reduction and "number crunching"; simulation (e.g. of molecular spectra); molecular modeling (3-D model building, energy minimization); curve fitting; statistics and exploratory data analysis; symbolic math and equation solving; digital signal processing; DNA analysis and plasmid map drawing, and kinetic analysis. For the computer programmers among us, we would need conventional programming languages as well as authoring systems for developing custom applications and courseware. Although conventional computer aided instruction (CAI) was not a high priority, we also wanted to be able to use the facility for instructional purposes where particularly effective software is available. Clearly, with such a wide range of potential applications, consistency of operation between all software packages would be a key factor. This suggests that an environment based on a modern graphic user interface (GUI) might be most appropriate, such as MS-Windows, the Macintosh, or one of the developing Unix-based GUIs (1-3).

C. Equipment

The deployment of computer workstation resources on our campus is based on workstation clusters located in various departments, libraries, and dormitories throughout the campus. Most of these are public, walk-in facilities that serve all students and faculty. Departmental personnel operate those facilities located in academic departments and offer a discipline-oriented mix of software and peripherals. A typical cluster contains 10-50 workstations and one or more laser printers. The current equipment the chemistry departmental workstation room consists of 23 Macintosh II computers (4). Each machine has a 32-bit CPU and data bus, floating-point math chip, dual 800K floppies, an Ethernet card, and 5 MBytes of random access memory (RAM). These machines are networked with a single file server and print spooler with 200 MBytes of disk storage, one LaserWriter IINT Postscript laser printer, a flat-bed scanner, and a CD-ROM drive. One of the

stations is equipped with a 19" monochrome monitor that can display two entire 8" x 10" pages side-by-side, which is useful for those working with large drawings, molecular models, spreadsheets, or complex page layouts. (All the application software works with this monitor without special drivers, a consequence of the windowing environment). Another station is equipped with a Textronix Cache system, a co-processor board designed for high-performance molecular modeling. There are also facilities for reading and writing MS-DOS (IBM-PC) format floppy disks (3.5" 720K and 1.44 Mb, and 5.25" 360 K formats). The particular model of the Mac II we have used is now no longer sold; the closest current model is the Mac IIsi or the Mac II ci, which are both less expensive and about a factor of two faster than our model.

D. Application distribution

One of the important choices to be made when designing a microcomputer cluster for general access is the method of distributing commercial application software to users. There are three distinct possibilities:

1. Set up a floppy disk lending library. Users would check out floppies from a central desk and return then when finished. This requires no special equipment and would meet the licensing requirements of any commercial software, that is, that the number of simultaneous users not exceed the number of copies purchased. However, we considered this to be an unacceptable solution because there is no practical way to guarantee that unethical or ignorant users will not copy the commercial software and because of the likelihood of maintenance problems cause by accidental deletion of needed files and the introduction of junk files and viruses. Moreover, much of the best software is simply too big to run from a floppy disk.

2. Provide each workstation with its own hard disk and install the software according to the number of purchased copies of each application. Each station would have to be labeled with the particular mix of software installed on its hard disk. This would share all the difficulties of the previous approach except the last. Experience in other public microcomputer facilities of this type had showed that the maintenance of multiple hard disks is a substantial problem.

3. Connect the workstations together in a local area network (LAN) and put all the applications on a file server that can distribute them electronically. Each workstation would not require its own hard disk, because they can be booted from a floppy disk or over the network. The main advantage to this approach is that all access to the application software is strictly through the network and the fileserver, so that it is in principle possible to copy-protect the commercial software, to lock and write-protect the server against modification by the user, and to control the number of simultaneous users according to the number of copies purchased, provided that the file server software offers those capabilities. Every software application would then be available at every workstation, limited only by the requirement that the total number of copies in use at a given time not exceed the license or number of copies purchased. (The license requirements of some software, however, explicitly forbid network operation). An additional advantage is that the cost of a single large file server hard disk is less per byte than smaller disks. A disadvantage is that it requires some additional hardware and software for a file server, as well as network connections and client software for each workstation. Moreover, it depends on well-behaved application software that works well in a shared environment. This is the approach used in our facility.

E. Networking

Early tests by our computer science center showed that the Mac's built-in 230 Kbps LocalTalk network port did not provide the required level of performance for a network of our size. As a result, the decision was made to standardize on 10 Mbps Ethernet interfaces for each individual workstation. Ethernet is the departmental networking standard throughout the University and Ethernet wiring (based on thin cable) is already in place in most offices in our department. Each workstation has an add-in Ethernet card and connects directly to the departmental Ethernet. Local network services operate on AppleTalk phase 1 protocols, using the AppleShare 2.0 file sharing software, the LaserShare print server and spooler, and the Internet Router, all of which run concurrently on one dedicated Mac II acting as the "Server". This software implements transparent "remote device mounting", in which the file server's hard disk and the laser printer seems to be connected directly to each workstation, so that disk operations and printing are done in the same way as for a stand-alone computer. A utility called KeyServer provides the software checkout counting and user notification required for proper application distribution and also keeps a

record of how often and when each software program is used. The print spooling software makes it practical for one LaserWriter to be shared by the 23 workstations. The router bridges the Ethernet to the LaserWriter, which connects to the fileserver via the usual LocalTalk connection. Since it sees only printer traffic and not file server traffic, the LocalTalk connection is adequate in this case. No additional hardware bridges or routers are necessary. In addition to the 23 Mac II's in the public workstation room, computers of all types in individual faculty or staff offices can be connected to the Ethernet with the appropriate network adapters in order to have access to TCP-IP services or the laser printer.

Note that we use file server primarily for application distribution rather than document sharing which is the standard model for office networks. We expect walk-in users to save their document on their own disks (which they provide) and to take them with them when they leave. Each workstation has two built-in floppy drives, one for the boot disk and one for user files. Each station also have a SCSI (Small Computer System Interface) port, which can be used to attach an external hard disk drive. Users who are working on very large files sometimes bring their work in on a small portable hard disk which they connect to their station while they are working. We do not provide long-term private user file storage on the file server. However, we have created a public subdirectory (folder) called the Temporary Storage Folder that users do have write privileges to. This is useful for temporary storage while working on large documents, facilitates the use of applications brought in by the users on floppy disks, and can be used as a convenient means of informal document sharing between users throughout the building.

F. Wide area access

In addition to the AppleTalk-based network services described above, we use TCP-IP protocols to provide access to the Chemistry Department VAXs, the campus electronic mail system, the library card catalog, the campus electronic bulletin board, any of the campus host computers, and national networks such as Internet, Bitnet, NSFnet, SURAnet, etc. Each station on the network is assigned a unique IP number for TCP-IP services. Anyone on campus can obtain a personal account on one of several Unix-based host systems that support mail servers. We use the MacTCP drivers and a public-domain terminal program (NCSA Telnet) to access the various departmental and campus computers. An FTP front end program provides a convenient point-and-click interface for downloading files from remote hosts on the Internet.

The connection between the departmental Ethernet and the campus broadband network is performed by a dedicated gateway machine that passes TCP-IP traffic but not AppleTalk traffic. Access to our departmental file server and printer is therefore restricted to workstations that are physically connected to the departmental network; this is desirable considering that it is the department itself that purchases the application software and the paper and toner for the laser printer. For similar reasons we currently do not have the ability to dial in to the file server or print spooler from outside the university via telephone.

G. Application Software

Currently, the departmental file server provides the following application software: Scientific word processing with spelling checking; math equation formatting; organic structure drawing; molecular modeling (with 3-D graphics, MM2 energy minimization, and molecular dynamics trajectories); general-purpose paint and draw programs; graphic scanning; optical character recognition; DNA analysis; plasmid map drawing; data plotting and least-squares fitting; spreadsheet with graphics and macros; statistics and exploratory data analysis; non-procedural equation solver; a vector/matrix processor with graphics; digital signal processing; electronic system simulation; computer languages and authoring systems for developing custom applications and courseware; a student expert system for organic chemistry ; first- and second-order proton NMR simulator; a mass spectroscopy fragment identifier; educational courseware for organic chemistry and organic spectroscopy (proton and carbon NMR, MS, IR); Hückel MO calculator. Each of these packages has been purchased in a quantity dependent on its anticipated level of use: in some cases enough copies for the entire room, and in other cases a single copy (in which case only one user at a time is allowed by the file server). In addition to the commercial software, we have collected several MBytes of utilities and science- related public domain software that users can copy, as well as a collection of chemistry "clip-art" organized in a Hypercard stack, for use in illustrating reports and lab manuals, etc. Some additional

applications were suggested by the availability of the technology rather than a perceived prior need: in particular, graphic and text scanning, using the scanner to capture, reduce, clean up, and annotate large chart recordings, gel electrophoresis plates, etc, for insertion into a report or thesis; optical character recognition for converting printed text into editable word processor files, and conversion of scanned graphs and plots into numeric data tables.

H. Memory and disk space requirements

In additional to the usual random access memory (RAM) requirements of applications running on a stand-alone personal computer, the individual workstations must additionally have enough RAM for the Ethernet driver, the AppleShare client software, the TCP-IP driver, and an automatic virus checker. As delivered, the computers had only 1 MByte of RAM, but we gradually expanded them to 5 MBytes, which allows easier operation with very large data files. One MByte of that RAM is allocated to a virtual disk (RAM-disk) on which the boot disk files are stored. This makes some operations much faster, and it frees up one disk drive, because the boot disk ejects automatically after its files have been transferred to the virtual disk in RAM.

I. Performance issues

So far all the software that we have tested works well from the server. Several simultaneous users can use one copy of a program on the server (subject to the license requirement). Most applications work from a write-protected volume or folder (that is, one for which the user does not have "make changes" privileges). Tests have shown that launching programs from the file server is nearly as fast as having a hard disk on each station. Network slowdown when the lab is very busy is noticeable but not objectionable. Overall reliability is good, but there are occasional temporary lapses of file service. Computationally, the Mac II is roughly as fast as an old MicroVAX, based on standard math benchmarks.

J. Maintenance

All such facilities require a certain amount of routine maintenance. The administrator can install new application software and upgrades and set up access privileges to volumes and folders from any workstation on the network, simply by logging on to the file server with a password. However, the administrator can copy protect the commercial software only on the file server itself. We do not give users write privileges to the file server, except for the Temporary Storage Folder, which the administrator must periodically clean out. The boot disks require periodic checking for viruses and stray files left behind by users. Viruses have been a problem. They can be spread from a user's disk to the boot disk and thence to another user's disk. An automatic virus checking *init* on each boot disk is a necessity. We update these regularly to respond to new virus strains. The boot disks get rather heavy use and tend to become corrupted or wear out eventually; a supply of backups is always on hand. Because of the problem of boot-disk maintenance, we are currently in the process of installing a modification that allows each station to boot directly over the network, without using any local storage disks. This modification will eliminate the boot disk maintenance problem and will make it much easier to install global changes in boot configuration to the entire lab, such as changes in system version or font selection.

Hardware and software upgrades and expansions have caused no problems. The author accomplished the installation of additional memory, an additional server hard disk, a CD-ROM drive, a scanner, and a large screen monitor with no difficulty.

K. Operation

The workstation room is a public facility that is open to all students, faculty and staff on a walk-in basis. No charge or authorization is required to use this facility. Undergraduate students hired as attendants are present during normal hours of operation. We leave the server on 24 hours a day, seven days a week, so that networked users outside the room can have access to the server and print spooler at all times. We keep manuals for all the software nearby. One wall of the room is equipped with a large corkboard which we have covered with screen dumps and sample print-outs of software doing chemistry-related work. We have prepared a substantial number of handouts that explain such topics as basic workstation operation, how to type special characters, transferring graphics from one application into another, operating the scanner, plotting data, typing equations, creating a spreadsheet or equation solver model, file transfer, etc. Copies are available in the room and are archived on the FileServer so users can view and print them out from any workstation. For the novice and first-time user, we have found that the "Guided Tour" disks that come with the computers are very useful. The Chemistry Department

offers no user training classes. The Computer Science Center does offer free peer training classes for students, as well as more formal classes and workshops in a variety of topics. Most users learn by watching other users and by experimenting.

L. Curriculum integration

We base our strategy for integration of computer use into the chemistry curriculum primarily on the use of commercial application software to support instructor-developed exercises. We emphasize the use of the computer as a student-directed tool rather than for lesson presentation, sequencing, and grading (5). For example, in our undergraduate organic chemistry classes, students perform a kinetics experiment in the laboratory, then bring their data up to the workstation room to key it into a general-purpose data plotting program to transform and plot the data, do a least-squares curve fit, and compute the activation energy of the reaction. These students are also exploring organic nomenclature and reactivity using Beaker 2.0, a type of expert system for organic chemistry students. The graduate organic class has been using molecular modeling and MM2 energy minimization to investigate structural concepts (6). Students in the environmental chemistry class use a spreadsheet to compute and plot elemental enrichment factors for atmospheric particulates using instructor-prepared data sets (7) and to investigate the influence of atmospheric chemistry on the pH of rain, using the Charson-Vong equilibrium model implemented in a non-procedural equation solver. An elective "electronics for chemists" class uses an electronic system simulation package to investigate signal-to-noise ratio enhancement systems, such as tuned filters, AC detection systems, and lock-in amplifiers (7). Students in a laboratory computer interfacing course investigate digital signal processing using a locally-developed program called SPECTRUM (8).

Most conventional CAI chemistry education software, such as the materials from SERAPHIM and JCE software, runs on Apple II or IBM-PC platforms. Third-party emulation software is available that allows these programs to run on the Mac II. We evaluated Insignia's SoftPC DOS emulator and found that it did indeed run all the MS-DOS programs we tried; moreover, it allowed convenient copy and paste of text and graphic between MS-DOS and Macintosh programs. Its screen update speed is somewhat poorer than one expects on a modern PS/2-class machine, but this is nevertheless a very inexpensive way to provide basic PC software compatibility.

M. Conclusion

The workstation room described here has been in continuous use since 1989 and has become a very popular facility. It offers a reasonable compromise between human factors (ease of learning and ease of use), computational performance, and the scope of available software applications. There is also ample opportunity for future hardware expansion: each workstation has 4 unused expansion slots for add-in cards and there is room on the motherboards for 8 MBytes of RAM using conventional 1 Mbit memory. As many as five more hard disks could be attached to the server's SCSI port. Future developments in hardware and software will certainly offer greater computational performance, but our hope is that the human factors will receive continued and even increased attention. Even the most "user friendly" computer can still sometimes be very frustrating, especially for the novice user and occasionally even for the more experienced user.

References

1. T. C. O'Haver, "Trends in Computer User Interface Technology", *Anal. Chem.* **4**, 191 (1985).

2. T. C. O'Haver, "User Environments for Personal Computer Workstations", *American Laboratory* **18**, 22 (1986).

3. T. C. O'Haver, "The Object is Productivity", *Intelligent Instruments and Computers*, March-April, 1992, p 67-70.

4. T. C. O'Haver, "Macintosh: A Viable Alternative?", *Science Software* **3**, 286 (1988).

5. T. C. O'Haver, "Software we use", *Computers in Chemical Education Newsletter*, Spring, 1992.

6. P. DeShong and B. B. Jarvis, "Use of the Cache System in Undergraduate Education: An Integrated, Multi-year Approach", Symposium on Integrating Computers into the Undergraduate Chemistry Curriculum, Fall ACS meeting, Washington, DC, 1992.

7. T. C. O'Haver, "Applications of Computers and Computer Software in Teaching Analytical Chemistry", *Anal. Chem.* **68**, 521A (1991).

8. T. C. O'Haver, "Introduction to Signal Processing in Analytical Chemistry", *J. Chem. Educ.* **68** (1991)

Quantum Mechanics on a Microcomputer

by Wilmon B. Chipman
Bridgewater State College
Bridgewater, MA 02325

The Quantum Chemistry Program Exchange (QPCE, Creative Arts Buikding 181, Indiana University, Bloomington, IN 47405) has recently announced the availability of a compiled version of the Fortran-77 MOPAC Version 6.0 quantum chemistry package that will run on IBM/clone 386 and 486 microcomputers with math coprocessors. The price is only $125 for the compressed program on two 750KB 3 1/2-inch diskettes. Additional requirements for running the compiled version of MOPAC include 8MB of RAM, 10MB of disk space and the Phar Lap Dos Extender (Par Lap Software, Inc, 60 Aberdeen Avenue, Cambridge, MA 12138, $495). As delivered, the executable file will handle structures with up to 25 heavy (non hydrogen) atoms and 25 light atoms. (The normal workstation version of MOPAC is set for 40 heavy and 40 light atoms.) If the source files have to be recompiled for use with larger molecules, the NDP-Fortran compiler (MicroWay, Inc., P. O. Box 79, Kingston, MA, 02364, the 386 version is $695.; the 486 version is $995.) is required. Compilation for forty light atoms and forty heavy atoms will require 20MB of RAM.

In order to make the best use of MOPAC 6.0 a front-end program is uselful to prepare MOPAC input files by drawing on the computer screen, and to view and manipulate MOPAC output. PCMODEL Version 4.0 (Serena Software, Box 3076, Bloomington, IN 47402-3076) can be used to prepare MOPAC input files easily. MOBY Version 1.5 (Springer-Verlag NY, Inc., 175 Fifth Ave., New York, NY 10010), which will be available soon, will prepare input files for MOPAC, and also allow stereo viewing of the resulting structure with the ability to rotate the stereo view in real time. With the companion program, QC4MOBY, MOBY will display graphically (in motion) each vibration from the infrared spectra predicted by MOPAC and will also show graphical representations of the calculated molecular orbitals. (MOBY alone will run AM1 and MNDO calculations of molecules up to the size of glucose.)

Don't plan to do quantum chemistry on a microcomputer if you expect very fast calculations. A small molecule which can be run in less than a minute on an IBM RISC/6000 can take upwards of a half hour (using the AM1 PRECISE calculations).

I used a beta test copy of MOBY Version 1.5 this past summer in Organic Chemistry I, II. I found that students really enjoyed being able to construct and rotate stereoviews and to view infrared vibrations. After trhis experience, I am convinced that at least half of the students who currently pass organic chemistry do it without much of a sense of what an organic molecule looks like in three dimensions. This may explain why many students find organic chemistry so dfficult.

BOOK REVIEW COLUMN

by Harry E. Pence, Chemistry Department, SUNY-Oneonta, Oneonta, NY 13820

Many chemistry instructors are discovering that it's possible to do rather complex computer applications in their classes without expensive hardware or complicated software. A popular way of accomplishing this is to adapt commonly available spreadsheet programs for chemical problems. Most of the popular spreadsheets, like Lotus, Quattro, and Excel, include not only excellent graphing capabilities but also the mathematical functions that are needed for most types of chemical calculations. In addition, these programs are often available in inexpensive student editions and are relatively easy to learn.

Those who wish to pursue this approach need a textbook that teaches spreadsheet fundamentals using chemical examples. Such a book can both serve as a source of homework problems and also minimize the lecture time which an instructor must spend discussing how to use

the spreadsheet. The two books reviewed in this column provide exactly this type of support. It should also be noted that those instructors who use a computer during their lectures will find these books to be sources of simple, but effective computer simulations, which make a useful lecture supplement.

SPREADSHEET CHEMISTRY by O. Jerry Parker and Gary L. Breneman Prentice Hall, Englewood Cliffs, New Jersey, 1991 288 pages, softcover, $22.00

The book should be useful with a variety of the commonly encountered spreadsheets and/or general chemistry texts. The spreadsheets in the book are intended to be used with the DOS version of Microsoft Excel, and disks are available with the data in either the IBM or Macintosh versions of Excel. Beyond that, it shouldn't be too difficult to translate the instructions into one of the other commonly used software packages. The order of the topics is designed to correspond to that in Brown, LeMay, and Bursten's Chemistry: The Central Science, but it shouldn't create serious difficulty if the topics are rearranged to suit the sequence found in most standard General Chemistry textbooks.

The authors start with simple problem types that are commonly treated at the beginning of a general chemistry course, such as temperature conversions and density, and use these topics to introduce the student to basic spreadsheet techniques. The fundamentals of using Excel are also discussed in several appendices. As one proceeds through the book, the topics become more involved and the spreadsheet use also becomes more complex. Some support from the lecture and/or lab practice sessions should be enough to allow most students to develop reasonable facility with the software. The descriptions include both the command text to be placed in the cells as well as a copy showing what the resulting worksheet should look like. Each chapter concludes with a good selection of problems that can be best handled with a spreadsheet, and appendices are included that briefly discuss successive approximations and the Newton-Raphson Method.

This book is quite useful, both for learning spreadsheet techniques and also as a reference. The discussions are clear, there is an adequate index, and the many diagrams are a significant complement to the text. The discussion of the chemical background for each topic is brief, but it's more than adequate if a student needs a quick review of a specific point while working on the spreadsheet. It's a pleasure to recommend this book. DYNAMIC MODELS IN CHEMISTRY by Daniel E. Atkinson, et al.N. Simonson & Company, Marina del Rey, California, 1990,320 pages, softcover, $19.95

There are many similarities between this book and that which is reviewed above. Both are intended for students in the introductory chemistry course, can be used with a variety of spreadsheets, and invite the reader to order spreadsheet diskettes in either the MS DOS or Macintosh formats. This book doesn't follow the standard set of topics from freshman chemistry as closely as the book by Parker and Breneman. It omits some topics, such as atomic structure and orbitals, but does add some additional topics, such as fractional distillation and extraction, that may also be helpful.

Atkinson et al make extensive use of sidebars to highlight important ideas. For example, each chapter begins with a sidebar that briefly lists the prerequisite knowledge necessary for the upcoming exercises. Other sidebars discuss specific points of concern regarding the model discussed or show sample graphs of the resulting data. This approach, which lets the reader decide how much help is needed at a given point in the discussion, is welcome. On the other hand, it's regrettable to find that a book that will be used as a reference lacks an index.

It can be argued, and I believe that the authors of both of these books would probably agree, that the most important aspect of spreadsheet use is not just solving problems but rather building a model that can be used to explore a topic in much more detail than would be possible if each individual calculation must be done by hand. If students can be encouraged to interact with chemistry in this way, regardless of whether it's done in lecture or on their own, they will be much more likely to understand chemistry rather than simply memorizing a set of algorithms for commonly encountered problem types. This book includes a number of problems that invite this type of exploration, and this is one of its strengths.

Both of these books provides an excellent basis for eitherstudents or faculty who wish to explore spreadsheet applications in chemistry. This reviewer uses Parker and Breneman somewhat more, but both books are well worth owning.

A FINAL COMMENT

Since this column will be my last as the book review editor for "The Newsletter," I wish to take this opportunity to express my thanks to everyone who has helped with reviews and suggestions during the past few years. Even though I can't mention all of you by name, I do wish to offer a special word of appreciation to Don Rosenthal, who first invited me to undertake this job and has helped me so often. I welcome the new book review editor, Larry Julien from Michigan Technological University, and wish him good luck as he begins his new job.

——End—— BYE Harry.

INTEGRATING COMPUTERS INTO THE UNDERGRADUATE CHEMISTRY CURRICULUM

by Harry E. Pence, Department of Chemistry SUNY Oneonta, Oneonta, NY

The Committee on Computers in Chemical Education sponsored a one-day symposium titled "Integrating Computers into the Undergraduate Chemistry Curriculum" at the ACS National Meeting in Washington, DC this Fall. The symposium was organized by Tom O'Haver (Univ. of Maryland) and Harry Pence (SUNY Oneonta) in order to encourage chemistry instructors at all levels to share information on their progress in integrating the variety of available computer software and hardware into an environment that will be most conducive for learning. Both the number of papers submitted to the symposium as well as the attendance demonstrated the high level of interest in this topic.

As might be expected, the various presentations represented a broad spectrum of approaches and technology. These papers covered all levels from high school through the senior year in college, and even at the introductory levels the computer techniques used were often extremely powerful and sophisticated. It was, however, obvious that even modest financial resources were sufficient to create productive computer integration. Several speakers described how they had used inexpensive software, matching grants, and other avenues to develop innovative programs.

The presentations at this symposium demonstrated the wide variety of ways in which computers are being used in undergraduate chemistry courses. Individualized learning is supported not only as computer-assisted instruction, but also by providing each student in large classes with unique homework and quiz assignments. Computer-controlled, multimedia presentation systems ("hyperbooks") will soon be widely available for lecture and self-study. Strategies for electronic literature searches are becoming a routine part of the instructional program. The increasing availability of resource rooms for chemical computing allows students to share their knowledge and participate in applications that go beyond those discussed in class.

It may be unfair to single out one development that was most exciting, but the frequent mention of programs that allow students to visualize chemical systems was especially impressive. Even though such methods have only recently obtained wide acceptance at the research level, several papers described their use in introductory classes. A broad range of software is capable of supporting molecular modelling and visualization, ranging from share-ware like MacMolecule to very elegant and expensive packages, like the CAChe system. These systems allow students at all levels to interact with molecules with a facility that's impossible with models. Allowing students to manipulate molecular systems in this way not only captures their imaginations, but it also seems likely that those who learn three-dimensional visualization in this way will develop a new understanding of chemistry and chemical reactions.

This symposium gave an exciting look at one of the more active areas of innovation in chemical education. No one seriously expects that computers will solve all problems in chemical education, but if these papers are an accurate indication, computers have an important role to play in whatever form the actual solutions may take.

INFORMAL NOTES ON PROGRAMMING-LANGUAGES

by K. W. Loach

Chemistry Dept., SUNY Plattsburgh.

LOACHKW@SNYPLAVA.BITNET or (518)564-4116

This is a follow-on from Part One (Comp. Chem. Educ. Newsletter, Fall, 1991), which covered the 'classical' languages

Fortran, Cobol, Pascal, etc. See Part One also for an introduction to the topic overall. The emphasis is on languages that can be used on personal computers, especially under DOS.

PART TWO: NEO-CLASSICAL LANGUAGES

This deals with well-defined languages that are not yet widely used by chemists. Each shows some interesting features and some useful capabilities not easily available in the classical languages.

Prolog:

The language was defined by A. Colmerauer (et al.) in the early 70's at the University of Marseilles. It grew at the University of Edinburgh and became widely known in the 'Edinburgh syntax' from Clocksin and Mellish's text. Prolog came to world notice in 1981 when it was selected by the Japanese government for their 'Fifth Generation Computer Systems Project'. It is often used in artificial intelligence work, as an alternative to Lisp. Prolog exists in many dialects and is not yet standardized. (This discussion uses mainly the Edinburgh syntax.)

'Prolog' is short for 'programming in logic'. Prolog programs define data relationships, but specify few procedural details. Instead, Prolog uses a generalized and powerful strategy of recursive backtracking, to discover all allowable combinations of database facts that fit the declared relationships.

.Suppose we have an aliphatic organic chemistry data-base, e.g.:

```
ketone(acetone).
ketone('2-propanone').
.....
primary_alcohol(ethanol).
sec_alcohol('2-propanol').
.....
```

Each line of the above is a typical Prolog program predicate (or database fact), ending in a period and asserting a relationship for one or more arguments. Predicate names and argument values require lower-case letters (this can be over-ruled by use of singly-quoted values, e.g. 'Br' or '2-propanol'). A predicate is not a function and does not have a value, e.g. to provide the boiling point of acetone, we would have to assert:

```
bp(acetone, '56.5Celsius').
```

(These examples are simplistic; in a useful system such predicates could be more informative, and could perhaps be generated automatically from a standard structural and reaction dataset.)

The database could then be queried with predicates containing variables (beginning with upper-case letters), e.g.

```
?- ketone(X).
...inquiry
  X = acetone      ...response
  X = 2-propanone ...another response
  .....            ...and so on
```

The database is scanned exhaustively for all values of X such that the inquiry is true. (If no match could be found, the system would respond: "NO".)

Reactions could be added to the database as assertions of general rules, e.g.

```
reduce(X, 'NaBH4', Y) :-
  ketone(X),
  sec_alcohol(Y).
```

meaning, X can be reduced by NaBH4 to Y if X is a ketone and Y is a secondary alcohol. This rule would then allow queries such as:

```
?- reduce(acetone, 'NaBH4', X)...inquiry
  X = 2-propanol          ..response
```

meaning, to what product would acetone be reduced by NaBH4? (Of course, the above reduce() predicate is a necessary but not sufficient condition; structural rules would have to be added to preclude e.g. the reduction of acetone to 2-butanol, but that is beyond the scope of this brief introduction.) The above rule could be applied in other ways, e.g.

```
?-reduce(X,'NaBH4','2-propanol').
 X = acetone
?-reduce(ketone(_), X, sec_alcohol(_)).
   X = NaBH4
```

and so on. In the last query above, the underline arguments (_) are generalized Prolog place-holders for arguments whose values are currently unimportant.

In general, Prolog rules can be written of the form:

```
p1(a1...) :- p2(...), p3(...).....
```

meaning the relation p1 is asserted for matching values of its arguments if predicates p2.... can be verified for their arguments. Of course, verification of p2.... may require further backtracking through the database. A predicate becomes an inquiry by giving one or more of its arguments as variables, instead of values. The important thing is that recursive backtracking will find every combination of argument values (explicit and implied) for which an inquiry can be verified.

Recursive backtracking can be very powerful, but can also lead to very long run-times and very large numbers of solutions (the 'combinatorial explosion') for a large database. Matching can be made more efficient by use of a 'cut' predicate, which can preclude search down some logical paths, if such further search is not wanted or unlikely to be useful.

A Prolog database is a file of

assertions and rules, in predicate form. The database is dynamic, because assertions and rules can be added, deleted and altered freely during the run of a program. Like Lisp and Snobol, Prolog programs are capable of generating and executing new code during the run, so the programs can be self-modifying and adaptive.

The above gives only a very brief taste of Prolog capabilities. Other operations are possible. Early Prologs could do only 16-bit integer arithmetic (in the range +32768), but newer versions can do rational, complex and double-precision floating-point arithmetic. Standard predicates exist to process data lists, e.g. ['F','Cl','Br','I'], trees (as lists of lists), arrays (as lists), strings (as lists of symbols) and graphs (as lists of nodes, with connections). Molecules could be coded and processed as atomic or functional-group graphs. There are also predicates to handle input, output and file manipulations, including Prolog data-base editing. Values (e.g. '2-propanol', 39.06) can be converted to and created from strings of characters. Prolog libraries are available to supply pre-written predicates for most common operations.

Prolog is particularly suited to the expression and application of logical relationships, (including fuzzy logic). In chemistry, it would be useful for exploring structure/property relationships, deducing structures from spectra, and generating chemical structures. Because it can handle logical inference, it would be usable in computer-assisted learning.

ADA Prolog (Computer Solutions).
PDC Prolog (Prolog Development Center).
Prolog (Arity).
Prolog (Cogent Software).
Prolog-86 Plus (Coders Source).
Prolog++ (Quintus Computer Systems).
tiny-Prolog (Austin Code Works).

Dr. Dobbs Journal, Mar. 1985, p.36.
Dr. Dobbs Journal, Apr. 1986, p.46.
Dr. Dobbs Journal, July 1987, p.30.

G. J. Kleywegt et al., Chemometrics Tutorials, Chapters 6, 7 (Elsevier, 1990).

W. F. Clocksin, C. S. Mellish, Programming in ˇˇˇˇˇProlog, 3rd.ed. (Springer-Verlag, 1987).

B. Filipic, Prolog Users Handbook (Wiley-Halsted, 1988).

N. C. Rowe, Artificial Intelligence Through Prolog (Prentice-Hall, 1988).

AWK:

AWK was never intended as a full language, but became one anyway. It was invented by Aho, Weinberger and Kernhigan (hence the acronymic name) as a utility for the Unix system. They wanted a 'tiny language' that could be used to write one-line programs to be used as text-filters and data-file scanners. All three of them being superb programmers, they created better than they knew. They were surprised when AWK began to be used as a serious programming language. DOS and Mac versions of AWK are now available. AWK has inspired a descendant called Perl (a sort of super-AWK).

AWK is an interpreted language. It resembles C but without the elaborate declarations and data structures. The syntax is simpler. AWK has a number of special features that makes it very versatile and powerful in scanning and manipulating text files.

The simplest AWK program has the structure:

```
{ action }
```

where 'action' is one or more AWK commands in free form layout. An input file is read, line by line, and the action clause is executed repeatedly using each line in turn as input, until end-of-file.

A more general program structure is:

```
pattern1 { action1 }
pattern2   { action2 }
.....
```

This would process the input line by line. The first line would be scanned in turn for each of pattern1, pattern2, and so on. Whenever a pattern was matched, then the corresponding action would be executed, with the current line as input. When all patterns had been applied, the next line would be read and the whole pattern-matching cycle repeated. This would continue, until all the file had been read, and all patterns had been applied to every line. It is possible to match patterns to chosen words or phrases, instead of the whole line. Patterns can contain arithmetic expressions and can test numeric values. If any pattern has no action, then the matched line is printed by default.

AWK patterns can contain 'regular expressions', e.g / hex[aey]ne/ would match 'hexane' or 'hexene' or 'hexyne', / [A-Z][a-z]+/ would match any word with an initial capital letter, and so on. Very complex patterns can be created, if de-

sired.

AWK input and output can be very easy, because of the I/O defaults. There is often no need for loops or formats. Pattern matching allows selective processing of input lines. It is possible to alter the defaults so that the unit of processing is a word, file, paragraph or page, rather than a line.

AWK data structures are very simple and flexible. There are no data declarations. Any variable can contain either numeric or text values. A numeric value is treated either as a number or as a string of numeric characters according to context. All arithmetic is floating point. Variables are automatically initialized to a zero and/or an empty string before first use. Arrays are one-dimensional and dynamic (i.e. of varying length) and can contain mixed numeric and string values. An array can be indexed with either numeric or text values, e.g. bp["acetone"] to obtain a boiling point from a bp data array. It is possible to test for use of a subscript, e.g. if ("acetone" in bp) print bp["acetone"]

AWK should not be considered an 'only language', but is a very valuable 'supplementary language'. It is useful for writing short programs, for transforming files, and for preparing file excerpts and data-base reports. Many experienced programmers now use it for 'program prototyping'. They try out algorithms in AWK, then translate the tested code into a more efficient, strongly typed language for production use. In some colleges, students are taught AWK as their first computer language, because of its simplicity and power.

Coherent AWK (Mark Williams).
Minix BAWK (Prentice-Hall).
MKS AWK (Mortice Kern Systems).
PolyAwk (Sage/Polytron).
A. V. Aho, B. W. Kernighan, P. J. Weinberger, The AWK Programming Language, (Addison-Wesley, 1988).

D. Dougherty, SED and AWK, (O'Reilly, 1991).

R. Kolstad, Unix Review, 8(5)30; 8(6)79; 8(7)44.

L. Wall, R. L. Schwarz, Programming PERL, (O'Reilly, 1991).

Forth:

The Forth language was created in the early 60's by C. H. Moore. It had its first major application in the early 70's as a data-control language for a radio-telescope. It is often used as a real-time instrument control language, but has spread to other applications. The Forth community is small but enthusiastic, with something of the 'true-believer' attitude once common among APL users. There are many dialects and two commonly used standards (Forth-79 and Forth-83). Forth is a readily extensible language, so dialects often differ greatly in the details of their floating-point arithmetic, string handling, file manipulation, numeric I/O and system interfaces.

At its heart, Forth is a postfix (or reverse polish notation) calculator language, based on the manipulation of an abstract stack. Numbers, strings and memory addresses can be pushed onto the stack. The value currently at the stack-head can be inspected and removed. Data can also be moved between the stack and addresses in memory, and this in turn allows the definition of named variables, arrays, matrices and strings. All such data structures are manipulated through the stack. This insistence on a stack-based operation gives Forth its characteristic style, very different from most other languages.

In the following one-line examples, entered numbers are pushed onto the stack top when encountered. Each operator acts on the one or two numbers at the stack top; the operands are popped from the stack and the result is pushed back onto the stack; e.g.

```
.                   ...pop and print.
3 5 + .             ...print (3+5)
3 5 + 2 /     ....print ((3+5)/2).
DUP     ...duplicate stack-top.
3 5 + DUP * .     ...print (3+5)2.
```

Named variables are created by reservation of an addressed memory segment, then values can be assigned to and read from the named address, e.g.

```
CREATE IC 2 ALLOT ...IC = 2-byte integer.
  3 5 + 2 / IC !...assign IC = (3+5)/2.
  IC @ .            ...access and print IC.
```

Arrays can be handled in the same way, but by reserving multibyte memory space. e.g.
CREATE AC 20 ALLOT
...AC = 10-integer array.
Manipulation of array elements requires explicit array-element address arithmetic, but usually there are library commands designed for this.

Forth can very readily be extended to supply capabilities lacking in the primitive language. Each of the operators we have seen above (such as . + DUP and ALLOT) are called 'words'. A word is the name of a pointer (i.e. a memory address) to a brief segment of machine-code. The machine code segments use stack-top and memory-addresses as operands. New words can be

readily defined, e.g.

```
: SQUARE DUP * ;        ...define 'square'.
: CUBE DUP SQUARE * ;   ...define 'cube'.
12 CUBE .               ...print 123
```

Both of these new words operate on the stack-top.

As a result of the 'word'-based structure, a Forth program is a sequence of pointers (or jumps) to brief primitive segments of machine-code . This is called 'threaded code'. Forth programs do not need to be compiled, and yet are much faster in execution than interpreted languages. Because the primitive word-segments are accessed by address, they do not need to be copied each time they are supplied to a program sequence, so Forth programs are usually very much more compact than the equivalent machine-code created by a compiler. The word-based organization lends itself very well to program development by a bottom-up approach (rather than the more widely used top-down approach of functional programming).

Forth resembles assembly language in many of its operating details and its efficiency of execution, and yet it is high-level and transferable. At first sight, Forth source-code seems dense and cryptic, but it becomes intelligible on acquaintance. Its compactness and efficiency make Forth a valuable language for numerical real-time instrument control with small computers. Forth has string and text processing capabilities and has been used as an implementation language for interpreters, compilers and text-editors.

If you consider acquiring a Forth system, pay great attention to the systems-library of defined words, to ensure that it has the range of capabilities that you need. The Kelly and Spies text covers most of the principal Forth dialects and has an good bibliography of Forth texts and sources of software and information.

M. G. Kelly, N. Spies, Forth: A Text and Reference, (Prentice-Hall, 1988).

L. Brodie, Starting Forth, (Prentice-Hall, 1981).

HOW I USE COMPUTERS IN MY LABORATORIES

by Edward Kelly
Chemistry Department
Marian College
3200 Cold Spring Rd.
Indianapolis, IN 46222

Marian College is a small, liberal-arts college on the northwest side of Indianapolis. Despite limited resources there has been progress in developing the use of computers in my courses.

The first microcomputers purchased by the school were TRS-80, models III and IV. Eventually they ere replaced by Apple and IBM computers. The Chemistry Department was offered full-time use of the TRS-80 computers. While not state-of-the-art, the TRS-80 has proved to be reliable and useful for introductory interfacing experiments. Finding a suitable interfacing circuit was my first problem. Fortunately I found one while attending a Chautauqua Short Course (Microcomputers as Laboratory Tools, Rex Berney, University of Dayton, 1986). For about $25 per computer I was able to interface the TRS-80 computers.

There is a strong emphasis in my labs on data collection, the analysis of that data and the interpretation of the results. The role of the computer in the lab to do the collection and assist with the analysis of data has been given high priority.

The lab manual (Inquiries Into Chemistry, Michael Abraham and Michael Pavelich, 2nd ed., Waveland Press, Inc., 1991), used for the first chemistry course for science majors, General Inorganic Chemistry, uses the discovery approach. Discovery experiments have the students collect data, graph the data, obtain an algebraic equation and "discover" the appropriate chemical principle. During the first semester the students do their own graphing to develop their technique; during the second semester the students use Graphical Analysis II (Vernier Software, 2920 W. 89th St., Portland, OR 97225) to learn how to use the computer to do their graphing. They always do a Least-Squares Analysis to find the best straight line through their data points; several experiments require a transformation of the data to obtain the desired straight line.

There are two types of discovery labs: guided inquiry (directions are given for the experiment, but only minimal directions are given for the analysis of the data) and open inquiry (only a statement of the problem to be investigated is given). Several of these labs have been modified to allow the use of a computer to collect and analyze the data. Additional interfacing labs have been developed and incorporated into the

lab schedule. These are described below.

The first interfacing lab covers some of the fundamental concepts of electronics and laboratory interfacing. These fundamental concepts are then used in other labs throughout the remainder of the year.

An open inquiry lab asks the student to investigate (collect the data and find the algebraic equation) the cooling curve for hot water. The data is collected using a thermistor interfaced to a TRS-80 computer. The computer controls the experiment, makes a measurement every 10 minutes, stores the data on a disk file and displays a crude graph on the screen during the experiment. Later, the students read the disk file , graph the data and find the algebraic equation.

Experiment K-1 in the lab manual investigates the kinetics of the reaction of bromocresol green and bleach. The bleach and dye are mixed and the time it takes the changing color to match a standard solution is determined. When the two colors match, the timer is stopped. This proved difficult for some students. A more effective way is to interface a Blocktronic (a photoresistor and green LED encased in two 2x4's bolted together) to a TRS-80. The internal clock of the computer is used to measure the elapsed time. The voltage output of the Blocktronic, while the color is changing, is compared to the constant voltage output of the standard. When the two voltages match, the internal clock in the computer is stopped and the elapsed time is displayed on the screen.

The oxidation of persulfate ion by iodide ion is investigated by measuring the changing absorbance with time, using the Blocktronic (see laboratory module, LM024, Project SERAPHIM, Department of Chemistry, University of Wisconsin-Madison, Madison, WI 53706). The computer collects the data and stores it on a disk file for future analysis. The initial slope of the line on the absorbance-time graph is proportional to the initial rate of the reaction. Seven reaction rates at different concentrations and temperatures are measured, enabling the student to determine the order of the reaction and the activation energy.

The molar mass of an unknown solute is determined by freezing point depression. A thermistor is interfaced to a TRS-80 computer and the cooling curves are obtained by the computer. The computer makes the measurements, stores the data and displays a crude graph on the screen. Later the student reads the disk file and graphs the data. The freezing points of the solvent and appropriate solutions are measured and eventually the molar mass of the unknown solute is determined.

A crude gas chromatograph is interfaced to a TRS-80 computer to separate a pentane-hexane mixture using a column of Tide detergent. This is a modification of procedures used by several authors (Chemtrek: Small-Scale Experiments for General Chemistry, Stephen Thompson, Prentice Hall, 1990; Interfacing the High School Science Laboratory to a Computer, John Fox, Vernier Software, Portland OR, 1988). The carrier gas is natural gas. An ir phototransistor detects changes in the temperature of the flame. The changing temperature is displayed on a line printer connected to a TRS-80. Various parameters, e. g. , retention times, plate thickness and the number of theoretical plates, are measured. The computer will determine the area under the curve by numerical integration (trapezoid method). From this a crude calibration curve can be determined and the percent composition of various pentane-hexane mixtures determined.

The gas chromatograph is the last of a set of three chromatography experiments (TLC separation of washable and permanent inks, PC separation of 5 transition metal ions, and the gas chromatographic separation of a pentane/hexane mixture). The SERAPHIM computer program SEPARATIONS (AP808) is used to introduce the TLC experiment. During the first 45 minutes of a two hour lab period the class views the program projected on a large screen using a LCD in the classroom. The program reviews some basic concepts of electronegativity and polarity in order to introduce some simple ideas of chromatography. A simulation of a TLC separation of washable and permanent inks shows the considerations necessary in choosing a solvent. The students then move into the lab and apply these concepts to separate the components of some washable and permanent inks.

During the last four lab periods, the students are required to do a mini-independent research project. To introduce the students to the research process, they independently work through the SERAPHIM program LAKE STUDY (AP804 or PC3704), a simulated investigation of the cause of a fish kill. They work through an assigned worksheet outlining the details of the lake study investigation.

To assist them in selecting a research project and to acquaint them with computer literature searching and the use of key

words, the students use CHEMLAB (PC2001), a data base of labs from the Journal of Chemical Education.

In the Physical Chemistry Laboratory, students work with computers in a variety of ways. This course is taken in their junior/ senior year. By that time they have had a programming course, either BASIC or Pascal. They are asked to write several computer programs during this course.

The first computer assignment is to write a program to find the root of an equation (volume in van der Waals cubic equation) by four different algorithms: (1) successive approximations, (2) bisection, (3) tangent, and (4) Newton. They are given two modules: ITERATION AND COMPUTER PROBLEM SOLVING, and ALGORITHMS FOR FINDING ZEROS OF FUNCTIONS (#478 and #264, respectively; COMAP, Inc., 60 Lowell St., Arlington, MA 02174), which describe these methods. They are asked to compare the methods for speed of convergence, accuracy, etc.

The second computer assignment requires the student to find the root of an equation (the solubility of an insoluble salt whose anion interacts with the solvent). They must write the six equations involving the various equilibria (solubility product, two acid dissociation equations, dissociation of water, mass balance and charge balance). (See, for example, Fundamentals of Analytical Chemistry, Douglas Skoog, Donald West and James Holler, Saunders, 6th ed., 1992, chapter 8.) The six simultaneous nonlinear equations must be reduced to one very nonlinear equation, whose root is determined by one of the methods mentioned in the previous paragraph.

The next computer assignment involves numerical integration. The students are given Cp-T data and a module, ELEMENTARY TECHNIQUES OF NUMERICAL INTEGRATION AND THEIR COMPUTER IMPLEMENTATION (#379, COMAP), and told to determine the entropy. The module describes three methods of numerical integration (left-rectangle method, trapezoid rule and Simpson's rule); the students write a program using the three methods and compare the results.

The students are asked to find experimentally the Ksp of $Cu(IO_3)_2$ by potentiometric titration. An ADALAB card interfaces the electrodes to an Apple computer. The computer stores the data, displays a graph of emf versus volume of titrant (making it easier to follow the titration and select an appropriate volume of titrant for the next addition); when the titration is over, the computer prints out the emf-volume data, including the average first and second derivatives of emf versus volume. The endpoint can be determined very accurately from the derivatives.

The students use a computer program that demonstrates the Monte Carlo simulation of a variety of chemical reactions. A module, MONTE CARLO: THE USE OF RANDOM NUMBERS TO SIMULATE EXPERIMENTS (#269, COMAP), describes the Monte Carlo method. The Monte Carlo method is applied to five chemical reactions; some simple first and second order reactions so that the Monte Carlo results can be compared to the results from the integration of the appropriate rate equation. Then more complicated reactions, that are not easily solved analytically, are "solved" by the Monte Carlo method to demonstrate its power.

The data from a kinetics experiment (the oxidation of $S_2O_8^{-2}$ by I^-) is analyzed, using two statistical techniques: analysis of variance and factorial design, to investigate the interaction of temperature and catalyst concentration on the reaction rate. Spreadsheets area very convenient way of obtaining the required sums, sum of squares, squares of sums, etc. needed in this statistical analysis.

To illustrate the power of the computer in doing quantum chemical calculations, the students are asked to find the minimum energy of the He atom by a self-consistent-field linear-combination-of-atomic-orbitals molecular-orbital (SCF-LCAO-MO) calculation using the trail wave function $C_1e^{-zeta\text{-}1\ r} + C2e^{-zeta\text{-}2\ r}$. One needs to guess the values of zeta-1 and zeta-2, then find the C_i's and the energy by iteration. The students are given the computer program that optimizes the C_i's and energy for a given set of zetas. The students spend one lab period trying to determine by whatever method they choose the best set of zetas, i. e. , those that give the lowest energy. The following week they are given another computer program which uses a simplex optimization algorithm to find the best set of zetas. They are then asked to compare the methods.

In Analytical Chemistry Laboratory the computer is used for record keeping and calculations. A potentiometric titration of iron is done by interfacing the electrodes to an Apple computer by an ADALAB card. The computer stores the data, displays a graph of emf versus volume of titrant. At the conclusion of the titration, the emf-volume data is printed out, including the average first and second derivatives of emf and volume; these derivatives enable the students to more accurately determine the endpoint of

the titration.

Copies of the laboratories described here are available from the author upon request.

COMPUTER CONFERENCING AND THE COMPUTER CONFERENCE ON APPLICATIONS OF TECHNOLOGY IN TEACHING CHEMISTRY

by Donald Rosenthal, Department of Chemistry, Clarkson University, Potsdam NY 13699 (ROSEN@CLVM.BITNET) **and Thomas O'Haver**, Department of Chemistry and Biochemistry, University of Maryland, College Park, MD 20742 (TOH@UMD2.UMD.EDU)

INTRODUCTION

An announcement of the Computer Conference on Applications of Technology in Teaching Chemistry appears elsewhere in this issue. This is the first such conference sponsored by the Division of Chemical Education's Committee on Computers in Chemical Education.

Computer conferencing has been used in education, business, industry, government and the military. Many of the applications in education are directed towards course work and degree programs. A few references are presented at the end of this article. Some of these references contain extensive bibliographies.

SELECTION OF CONFERENCING SOFTWARE

The authors are interested in conferencing software which is available at little or no cost and is accessible to a wide variety of chemistry educators. It was decided to use LISTSERV since it can serve those having BITNET or INTERNET addresses.

LISTSERV is used by the Chemistry Education Discussion List (CHEMED-L). Many chemists are already familiar with some of the features of LISTSERV. In the configuration we are using anyone can establish a CHEMCONF LISTSERV account and can remove himself from CHEMCONF. When mail is sent to CHEMCONF it is automatically distributed to all those on CHEMCONF. Each piece of mail is indexed and filed in a "notebook." Users can obtain the index (or part of the index). A file can be retrieved from the "notebook" and printed. Also, it is possible to obtain the names and electronic mail addresses of all those who are signed on to the LISTSERV account. Under favorable conditions messages are distributed within a few minutes, but in some cases longer delays may be observed. Thus, it is not possible to carry on a "real-time" (synchronous) conversation in the usual sense. However, this may have the advantage of encouraging more thoughtful replies.

LISTSERV is almost strictly a text oriented facility. Graphics mostly in the form of slides and transparencies are important to most presentations at meetings. The authors have identified the software which will convert a graphic image to an ASCII character file and additional software which will produce the original graphic image from the ASCII file. Much of this software is either in the public domain or available at a small cost. We have developed simple instructions to assist authors and participants in using the software. Graphics, chemical structures, spreadsheets, executable binaries, HyperCard stacks and digitized photographs can be handled. In addition to these instructions, individual help will be available through the conference manager (Tom O'Haver).

FORMAT OF THE MEETING

The meeting will be divided into two sessions. Each session will consist of five papers and will extend over a period of three weeks. On the first day (a Monday) all five papers will be distributed to all participants. Each paper will consist of an ASCII text file which may contain tables. Each graphic will be transmitted as a separate ASCII file which will have to be converted by each participant. Participants will have the remainder of the first week to read the papers. Authors are encouraged to stimulate discussion by including questions for the participants in their papers. Participants will have an opportunity to send short (less than 100 word questions) to the author or other participants on a designated day during the first week.

The second and third weeks will be devoted to asynchronous discussion and questions. Two days will be devoted to discussion of each paper. This format has some distinct advantages over a conventional on-site conference.

a. Each participant will have a week or more to read the paper and prepare any comments, suggestions or detailed questions. The participant will be able to research and carefully word his comments and questions. Relevant references can be located and cited to reinforce a point of view.
b. Authors will have time to consider responses to questions.
c. Since discussion will occur over two days, there will be time for an extended dialog and considerable discussion.
d. All papers and discussion will be saved in files until the end of the conference. Participants and authors will be able to access these files at any time.
e. Authors planning to submit their papers for publication subsequent to the conference may receive de-

tailed comments and suggestions which will improve the quality of a revised version of the manuscript.

This conference is to be regarded as an experiment. Its success depends upon the quality of the papers submitted, the ingenuity of authors to adapt to a format which has different characteristics from that of a traditional meeting, and the quality of discussion generated by participants and authors. Authors are encouraged to submit non-traditional as well as traditional papers. The questions and discussion generated by a paper may be even more important than the paper itself.

A brief description of the scenarios we expect for authors and participants is presented below.

Authors must submit a title and abstract of not more than 150 words by February 1, 1993. The deadline for submission of the paper is May 1, 1993. Instructions concerning the format of the paper are available from Tom O'Haver (TO2@UMAIL.UMD.EDU). Authors are expected to review first-week short questions received from the participants on the day following the receipt of such questions and to prepare responses to appropriate questions. Responses should be sent to CHEMCONF early on the day that discussion begins. For example, short questions for the first paper in a session would be received on Tuesday and examined by the author by early Wednesday of the first week. Responses would be sent early on Monday of the following week when discussion of the first paper begins. (Not all short questions are necessarily directed to the author.) Authors would be expected to examine the discussion in the middle and towards the end of the first day (Monday for paper one) and respond in a timely manner.

Participants are expecteto retrieve the papers on Monday of the first week. We suggest that each paper be printed and that graphics be viewed on the monitor or printed. The first paper will be read on Monday. Short questions will be sent on Tuesday (for paper one). Longer questions and comments will be prepared and sent (on Monday of the second week of the session for paper one). Participants will examine the discussion several times during day one and day two of the discussion (Monday and Tuesday of the second week of the session for paper one). Participants might add additional comments or respond to comments made by others. While it is not essential that every participant read and discuss every paper, it is expected that everyone on CHEMCONF will participate significantly in the conference.

At the end of the conference participants will be asked to evaluate the conference and to make suggestions regarding changes which might be made to improve future conferences.

TRIAL SESSIONS IN FEBRUARY

We will conduct a trial of this conferencing technique during the first week in February. On Monday, February 1 two papers will be distributed to everyone signed on to CHEMCONF. Short questions will be submitted on Tuesday, February 2 for paper one and on February 3 for paper two. Discussion of paper one will occur on Monday and Tuesday, February 8 and 9, and paper two would be discussed on February 10 and 11. February 12 will be used for evaluation of the trial sessions and for obtaining suggestions on how to improve the June sessions. Not everyone signed on to CHEMCONF may elect to participate in these trial sessions. However, authors and participants will have this opportunity to obtain a better understanding of computer conferencing before June. If participants have difficulties, they may wish either to seek local assistance or to contact Tom O'Haver. We may revise some of the instructions and suggestions to participants as a result of these trial sessions.

ADVANTAGES OF COMPUTER CONFERENCING

1. Time and expense of travel are eliminated.
2. A permanent record of the papers and discussion is available.
3. Participants can select the dates and times for reading papers and for participating in discussion.
4. In conventional meetings, discussion is limited and those participating must respond within a very short time period. In this "meeting" discussion is possible between speaker and participant, and between partici-

pant and participant. Participants and speakers need not respond immediately; they can think about a question, even research a topic before responding. Responses may occur within a few minutes or a few weeks. Ample time is provided for discussion. Collaborative problem solving and polling may occur.

5. The sharing of data, computer programs, images, graphics and other digital forms of information are facilitated.
6. Authors can submit more detailed presentations than are possible at an ordinary meeting.
7. Authors of papers can ask questions of participants with the expectation of receiving well thought out answers. Participants can let authors know where their papers are unclear or may make suggestions. Authors can learn from participants about relevant work or publications.
8. Conference expenses and preparations are minimal compared to what is involved in a conventional meeting.
9. Participants and authors from all over the world can in teract.
10. Hearing and speaking disabilities, as well as many other physical handicaps, do not prevent or hinder participation. Those from abroad who can read and write English but have difficulty in speaking or understanding the spoken language are not at a disadvantage.
11. Participants can ignore papers in which they have little interest and can examine the papers in each session in any order they wish.

DISADVANTAGES OF COMPUTER CONFERENCING

1. The pleasure of socializing at a physical meeting are absent, as is the valuable communication "back channel" of body language and inflection.
2. Computer conferencing would be limited to those who have some form of computer connectivity — at least an electronic mail address.
3. We are already familiar with the mechanics of physical meetings (traveling, making slides and transparencies, public speaking, socializing) and of print publication (using libraries, subscribing to journals, preparing manuscripts). Computer conferencing requires skills that are not necessarily more complex but are generally less familiar.
4. A physical conference occurs in real-time, and discussion is synchronous and consecutive. In computer conferencing discussion of a particular aspect of a paper may occur over several days with other discussion being intertwined. Participants and authors will need to sort out the different threads of the discussion.
5. Computer conferencing does not yet have recognition in the scientific community - one of the reasons scientists go to meetings and publish papers is that it enhances their prestige and reputations.

CONCLUSION

This conference is different. We need to discover how best to exploit the technology and format of the conference. Its success depends upon the ingenuity of the authors and participants. Won't you help by participating.

CHEMCONF will be made available for other Chemistry Conferences to be held after September 1, 1993. If you are interested in organizing such a conference, please contact Tom O'Haver at the address given above.

REFERENCES

1. LindaM. Harasim (editor), "ONLINE EDUCATION: Perspectives on a New Environment", Praeger, New York, 1990 (ISBN 0-275-93448-9).
2. Matthew Rapaport, "COMPUTER MEDIATED COMMUNICATIONS: Bulletin Boards, Computer Conferencing, Electronic Mail, and Information Retrieval", John Wiley, New York, 1991 (ISBN 0-471-51642-2).
3. John S. Quarterman, "THE MATRIX: Computer Networks and Conferencing Systems Worldwide", Digital Press, 1990 (ISBN 1-55558-033-5)
4. Michael Strangelove and Diane Kovacs, "Directory of Electronic Journals, Newsletters and Academic Discussion Lists", Association of Research Libraries, Washington, DC, 1991 (ISSN: 1057-1337)
5. Tracy L. LaQuey (editor), "The User's Directory of Computer Networks," Digital Press, 1990 (ISBN 1-55558-047-5)
6. Ed Krol,"THE WHOLE INTERNET: User's Guide & Catalog", O'Reilly & Associates, Sebastopol, CA, 1992 (ISBN: 1-56592-025-2)

COMPUTERS IN CHEMICAL EDUCATION

SUBSCRIPTION RENEWAL FORM

To renew fill out this form and return it with your remittance to:

Dr. M. Lynn James, CCCE Newsletter
Department of Chemistry and Biochemistry
University of Northern Colorado
Greeley, CO 80639

Please check one:	_____ Renewal	_____ New Subscription
USA:	_____ 1 yr. for $ 2.50	_____ 2 yrs. for $ 4.50
All Other Countries:	_____ 1 yr. for $ 5.00	_____ 2 yrs. for $ 9.00

Payment **MUST** accompany this form. Please make your check or money order payable in US funds to **Computers in Chemical Education Newsletter**

Your Name: _____

Address: _____

_____

_____

Telephone: Work _____ Home _____

BITNET, INTERNET, or other electronic mail address _____

Courses which you teach _____

Name of school or professional affiliation
(if not indicated in the above address) _____

Types of articles you would like to see in future issues:
Rate on the following scale: 1 - Very important, 2 - Important,
3 - Average importance 4 - Not important

1. General articles on how teachers are using computers ____
2. Reviews of 'useful' software ____
3. Reviews of hardware ____
4. Brief "Who Done It" ____
5. Queries and Answers ____
6. Book Reviews ____
7. Programming tips ____
8. Calendar of Events of interest to computer users ____
9. Networking and networks ____
10. Other - please describe _____ ____

OVER PLEASE

11. Are you a member of the:

ACS?	____ Yes	____ No
Division of Chemical Education?	____ Yes	____ No
Division of Computers in Chemistry?	____ Yes	____ No

12. Areas of Computer Activity and Interest:
Leave the space provided below blank, if you have no present interest or activity. Insert a number from 1 to 4 depending on the amount of activity.
1 means with a consuming passion, 2 means considerable
3 means moderate, and 4 means a little.

	Activity	Interest	Description of Use
Word Processing	____	____	_____
Spreadsheets	____	____	_____
Data Bases	____	____	_____
Other Languages	____	____	_____
Simulation	____	____	_____
Numerical Methods and Statistics	____	____	_____
Graphics	____	____	_____
Interfacing	____	____	_____
Laboratory Automation	____	____	_____
Drill and Practice	____	____	_____
Other (specify	____	____	_____

13. Provide a brief description of the hardware you use.

_____

14. Other Comments or Suggestions: