

COMPUTER USAGE COMMUNIQUÉ

Vol. 2 No. 3

CONTENTS

	Page
Profile: John W. Sheldon	2
STRETCH	4
NCR 315 Seminar	6
Sequencing Source Deck Program	8
Reflections of a Twelfth Floor Defector	9
Book Review	10
An Analyst's Bookshelf	11
New Employees	12
News from Washington	15
New Burroughs Release	16
Who's Got the Zebra	17

Please address all inquiries to the library

PROFILE: JOHN W. SHELDON

B. P. Lesser

In 1955, in Mr. Sheldon's bachelor apartment, CUC was born. Together with Mr. Kubie, the company was founded to assist people in the use of computers, i.e., in the use of the equipment rather than in giving them advice. Needless to say, advice was offered but this aspect played a relatively minor role in the general scheme. Since 1955 Mr. Sheldon has had the main responsibility for scientific and engineering applications performed at CUC and since then he has also married Ginny, a former writer for Time Magazine, and become the father of a brilliant $2\frac{1}{2}$ year old, Leonie.

One of the first to use large-scale digital computers for the numerical solution of partial differential equations, Mr. Sheldon entered the computing field in 1949 as a mathematical physicist at the IBM Watson Scientific Computing Laboratory. There he developed a numerical solution for a modified Hartree-Foch equation and Patterson-Harker maps for the insulin molecule, as well as obtaining numerical solutions of nonlinear elliptic and parabolic partial differential equations on the IBM 701. He was the first to solve successfully the transient two-phase flow equations for oil and gas in a porous medium. In 1952-53 Mr. Sheldon was Manager of IBM's "Scientific Computing Service" and was engaged in the design of the 701. He also prepared the numerical methods for a two-dimensional reactor multigroup diffusion computer program in 1953. This was the first computer program to solve the two-dimensional reactor diffusion problems relatively economically. At a later date, CUC programmers and analyst, supervised by Mr. Sheldon, developed jointly with Knolls Atomic Power Laboratory a second two-dimensional reactor-diffusion computer program. This program, called CURE, is one of the nation's important reactor codes.

Although a vast number of problems covering a wide variety of subjects have been solved under his supervision, his preference is solving problems in differential equations. When working on a specific problem, he especially enjoys conjuring up numerous solutions --- and then choosing the one which will flow best on the particular computer involved. As for the machine itself, his main interest is with its speed, capacity, etc., rather than with its internal mechanics.

Mr. Sheldon and his wife Ginny have long maintained an interest in the South American countries, culminating in a trip there about four years ago. Together they studied Spanish at the Berlitz School and have been regularly taking part in a "Spanish Evening" together with others who share this interest. The group meets once a week.

When asked about his musical preferences, he smiled and answered "very square," naming Bach and Vivaldi among his favorites, to -gether with baroque oratory. He naturally has a HIFI in his home replete with antenna and motor in his bedroom to ensure the best possible reception. Other favorite activities include camping trips, sailing, long walks, and his red-haired daughter Leonie. "She loves to sing songs, say poems her mother exposes her to and," he willingly admitted, "has a far better memory than I." Then with a note of fatherly pride in his voice, he concluded that "her present ambition is to become a conchologist."

STRETCH

G. R. Trimble

STRETCH (the IBM 7030) is the largest, fastest, operating general purpose computer. It combines fixed word length arithmetic for performing floating point operations with the flexibility of variable word length arithmetic in which the words can be composed of "bytes" with from one to eight bits in a byte. The logic includes "look ahead" of instructions, in which as many as four instructions can be processed simultaneously. The arithmetic unit is extremely fast and the main memory has a two micro-second cycle time with multiple memory modules operating simultaneously. It is extremently difficult, however, to estimate accurate timing for individual instructions because of the complexity of "look ahead" plus the variable length of time required to execute VFL (variable field length) instructions. A reasonable estimate, however, is that approximately 1,000,000 instructions per second can be executed.

Eight machines have been delivered. These are the only STRETCHes which have been and will be produced. They are: British Atomic Energy Commission, Livermore Research Laboratories, Los Alamos Research Laboratories, Naval Weapons Laboratory -- Dahlgren, United States Weather Bureau -- Washington, D. C., Mitre Corporation, International Business Machines Corporation -- Poughkeepsie, United States Navy -- Bureau of Ships.

The cost of a typical system is between \$8,000,000 and \$13,000,000 and the rents vary from \$150,000 per month to \$250,000 per month.

Several programming systems exist for STRETCH. The most basic is STRAP, STRETCH Assembly Program. This is a basic symbolic assembly system. SMAC, STRETCH Macro System, is an extension of STRAP which includes approximately thirty basic macro-instructions. It is logically equivalent to the TOOL System developed by CUC for the H800. The SMAC macros are designed to facilitate the construction of other macros and include "string type" routines which are useful in analyzing macro statements or other high level language statements to construct generators. A FORTRAN II system has been developed by the British Atomic Energy Commission. This system compiles fairly rapidly, but does not produce efficient object programs. IBM provides a FORTRAN IV system which takes apprximately three times as long to compile on STRETCH as the equivalent FORTRAN II program does to compile on the 7090. Object programs produced by FORTRAN IV are executed on STRETCH in approximately 1/3 the time of the same program on the 7090.

In spite of the tremendous number of variations on instructions, STRETCH is a very easy machine to program. The major problem in programming STRETCH is deciding how you want to do something, since there are so many alternative ways in which it can be accomplished.

CUC has undertaken four programming tasks on STRETCH. The first STRETCH task involved writing functional test routines for IBM to verify that machines coming off the production line operated in the manner that they should. These routines become part of the STRETCH maintenance programs and diagnostic package. This task was undertaken before any STRETCHes were actually operating. Our second task involved the simulation of the Control Data Corporation Fire Control Computer on STRETCH for the Naval Weapons Laboratory at Dahlgren. The third project currently underway is ASTRO-STRETCH and is basically a large scientific program to analyze satellite data and derive geodetic information which will fix the postitions of points on the surface of the earth far more accurately than standard surveying techniques can. The fourth project is a follow-up to the third and consists of writing sort routines which are required in conjunction with the ASTRO-STRETCH project.

Hillel Bardin

The National Cash Register Company's large BCD EDP System, the NCR 315, was described and demonstrated at a two-day orientation session which I attended this April. The group, consisting primarily of NCR salesman and EDP consultants, met at the NCR data processing headquarters on Madison Avenue and 61st Street.

Some of the 315's most interesting features are its fixed 12-bit syllables within variable length words, its large assortment of peripheral equipment, including a magnetic "Card Random Access Memory," and a 2000 card per minute photo-electric punched card reader. The 315's main core memory is expandable to 40K addressable syllables (called "slabs"). Instructions can treat slabs as two alphameric or as three numeric characters, and can deal with subsections of the slab individually. The computer uses an accumulator of up to eight slabs for data moves and arithmetic. Instead of using word marks in memory, the number of slabs actually involved are coded in the instruction. Most instructions have a single operand and are contained in two slabs of memory, although some, like COUNT, take four slabs.

COUNT, which is used in looping, increments a counter each time through the loop and compares it to a terminal value, setting a High, Low, or Equal flag which can then be interrogated. Other instructions include Edit, Binary Add, Shift Accumulator, Memory Scan for High, Low or Equal compare, and a demand linkage command for interrupt. Some commands like Add and Multiply can use numeric literals up to 999 in the actual instruction coding. Some of the terminology (e.g. "Jump" rather than "Branch") and instructions (e.g. Skip forward or backward a specified number of slabs) are reminiscent of Control Data Corporation's machines and, the NCR 310 computer, which was shown to us in a film, is in fact the CDC 160 painted brown! CDC has been closely connected with the National Cash Register, NCR providing outlets to the commercial market for CDC's scientifically oriented hardware.

The "Card Random Access Memory" (CRAM) is one of NCR's innovations. A pack of 256 mylar cards, each 14"x3 1/4", is placed in a cartridge and simply loaded into the CRAM unit. Eight rods then support the cards, until an eight-bit binary code is given causing rods to turn and one unique card to drop into a vacuum chamber and wrap around a rotating drum.

The card contains seven magnetic tracks, any one of which may be selected to be read or written on. The card can be kept on the drum as long as needed, after which it is shot back up to the pack. Each track accommodates 1550 slabs and the entire cartridge holds five andone-half million alphanumeric characters. Data can be transferred to memory at the rate of 100K alphanumeric characters per second after the initial card drop time of 235 ms.

At the end of the two-day session, we were given a demonstration of the system. I was most impressed by the 2000 card a minute reader, which sucks the cards serially past photo-electric readers at a tremendous clip, without ever grabbing the cards as does our IBM 1402. The demonstration which most aroused the interest of the class was the check sorter. The instructor took a check, crumpled it, twisted it, tore it right through the magnetic character it was to be sorted on, blacked it in with pencil, and yet was able to sort it with no difficulty.

In our library we now have the programming manual for the NCR 315 (which is concise and clearly written), the Autocoder manual (NEAT), as well as write-ups on all peripheral equipment. NCR has also written a COBOL, which they say is working, and various executive programs for use with their system.

Nerhaten!

"Achtung! Alles Lookenspeepers. Was computenmachine is nicht für gefingerpoken und mittengrabben. Is easy zu schnappen der springenwerk, blowenfusen, und poppencorken mit spitzensparken. Is nicht für gewarken bei das dummkopfen. Das rubbernecken sightseeren keepen hands in das pockets-relazen and watch die blinkenlights."

-7-

The following program has been prepared by <u>Rita Altucher</u> for card sequencing. All source decks, including those for such languages as FORTRAN and COBOL, should be sequenced when prepared for processing.

SEQUENCING SOURCE DECK PROGRAM

If sequencing is required in columns 73-80, indicate on a control card in these positions the job identification code and the sequence number of the first card to be punched. In positions 70-72, indicate the card column of the first numeric character after the identification code. A zero should be placed in column 70. If the sequence number on each card is to be incremented by ten or by one, punch 10 or 01 in columns in columns 68 and 69.

The control card should be inserted behind the object deck of this program and in front of the programmer's source deck.

If page and line number sequencing is wished, indicate on the control card in columns 103 the difference in line number value between each card, i.e., 010 or 001.

Both reader and punch are to be ON at all times.

The same source deck that is read in will be punched out provided that:

- 1. The I/O switch is OFF
- 2. The source deck is placed in the punch hopper with two blank cards behind it
- 3. Columns 73-80 and 1-5 of the source deck are blank if sequencing is wished in these respective fields
- 4. The object deck of this program, the control card, and blank cards equal in number to the source deck are placed in the reader

-8-

REFLECTIONS OF A TWELFTH FLOOR DEFECTOR

E 12 m

-Frank Lee

During my brief tenure on the fourth floor, I have often been asked three general questions. How much different is programming from console operations? How do I enjoy programming? and Whad-dya doin down here?

In reply to the last question, I could say that I get off at the wrong floor every morning, but the truth of the matter is that when I was first hired I had asked for an opportunity to write a program or two. Ye gads, I'm on CUCU.

Do I enjoy programming? Yes, for two reasons. It's always a challenge to write a program and have it go to "End of Job." The fact that after the program is loaded, it branches immediately and automatically to "End of Job" doesn't detract from my enjoyment one bit. Also, after two years I finally have a chair and desk which I can call my own. One does get tired holding up a wall which one is unable to park. Remember 18 East 41 Street !

As to the difference between operating and programming, both programmer and operator are means to an end -- the end being the successful processing of the program in a machine. The programmer provides the means, the operator uses the means. Now and then, however, there are divergent views. Particularly when the machine stops and all the lovely lights come on. It's then that the programmer with his or her immense knowledge of logic, op codes, etc. informs the operator that the machine is "down." In reply, the operator with his enormous knowl edge of systems, machine hardware and functions, attempts (often unsuccessfully) to convince said programmer that the machine is not "sick" but rather the program contains an "oversight" -and so, far into the night, we leave them arguing.

-9-

BOOK REVIEW

A. Opler

Dr. Kenneth E. Iverson of IBM has published a stimulating and worthwhile book entitled <u>A Programming Language</u>. While the title is pertinent to the contents, it may be misleading since it is not a problem-oriented computer programming language that is being described. Iverson makes the bold suggestion that many problems of interest to applied mathematicians and computer scientists can be better formulated in "a programming language" than with mathematical symbols. Thus, the book describes a new descriptive way of writing mathematical formulations. With 5,000 years of development of mathematical symbolism contrasted to a very short period for the development of programming notation, his suggestion does seem striking.

In the long first chapter, he develops the notation for his programming language. In the remaining six chapters, he demonstrates the use of his notation by applying it to six diverse fields. He shows how it may be used for representing mathematical structures like trees, chains, matrices, etc.; for describing searching and sorting procedures; for describing details of IBM 7090 instructions; for describing manipulations used by algebraic compilers and for describing forms used in the logical calculus.

Iverson writes in a clear, concise style and rounds out each chapter with exercises and a bibliography. White the book is scarcely suitable for reading from beginning to end, the reader might do well to familiarize himself with the first chapter and with one or more of the chapters which demonstrate the use of the language.

*Note: A copy of <u>A Programming Language</u> can be found in our library.

AN ANALYST'S BOOKSHELF *

Peter Hagis, Jr., Temple University

- 1. The Jacobians and Their Struggle for Independence
- 2. A Ten Day Diet to Improve Indeterminate Forms
- 3. Cheaper by the Googol
- 4. 1001 Best Loved Double Integrals
- 5. The Torus and I
- 6. A Short Table of Even Primes (Abridged)
- 7. Will Success Spoil Runge-Kutta?
- 8. Dining Out in Hilbert Space
- 9. 100 Tasty Fillings for Empty Sets
- 10. Life Begins at e
- 11. How to Keep Condensation Points from Dripping into Open Sets
- 12. A Child's Garden of Tchebycheff Polynomials
- 13. Tom Swift and Electric Cycloid
- 14. A Treasury of Matrices -- Upright and Inverted
- 15. Improving Lipschitz Conditions in the Slums of New York
- 16. The Decline and Fall of e^{-X}
- 17. How to Prevent Rust on Riemann Surfaces
- 18. The Peano Postulates Transcribed for Violin and Cello
- 19. First Aid for Dedekind Cuts and Bruises **
- 20. A Collection of Happy Endings for Incomplete Beta Functions

* Taken from the American Mathematical Monthly, Volume 69. Number 10, December 1962, Pages 980-981

** An appropriate subtitle, believed to have been originated by P. D. Lax, might be "What to do Until the Mathematician Arrives."

-11 -

WELCOME TO -- NEW EMPLOYEES

Stuart Baden--Programmer Trainee

A physics major and a "big wheel" in college, Stu attended CCNY where he was the president of his fraternity, Phi Epsilon Pi. After graduating in January, 1962, he alternately worked for Equitable Life Assurance as an insurance underwriter and for IT&T programming the ADX 7300. Then followed "five glorious months of total unemployment" during which he travelled to Florida, the Midwest, Canada and began writing the Great American Novel. He is currently attending Brooklyn College two evenings a week where he is taking Education courses. He loves the summer, swimming (he worked as a life-guard in Far Rockaway), sailing, boating, and convertibles.

Hillel Bardin--Programmer

A native of Israel, Hillel reached the United States at the age of four. After graduating from Columbia College as a sociology major, he continued his studies in this field, completing a year of graduate work at Columbia. Prior to joining CUC, he was employed by the NYU Statistical and Computation Laboratory in the Engineering Research Division. There he worked on systems and scientific programs, programming Control Data Corporation machines as well as the IBM 1401.

As a hobby, Hillel has, for many years, conducted classes in international dancing--having taught folk dances of America, Bulgaria, Germany, Greece, Holland, Ireland, Israel, Russia, Syria, etc. He is specializing in and concentrating on Irish folk dancing and is currently studying step-dancing with Kathleen Collins, T.C.R.G., one of the outstanding teachers in this field. Step-dancing, an intricate form of Irish dance originating around the 18th century, is a solo dance in which the only parts of the body in motion are the legs and feet--arms, head, and upper body remaining immobile. Last year he danced with the Bronx Gaelic League in the group dancing competition at

the FEIS held at Iona College. (The FEIS is an annual Irish festival in which many groups from all over the country compete in the performance of various art forms, e.g. dance, music, oratory, drama, etc. --a requisite of the latter being that they be performed in Gaelic). At present Hillel is working on a notation system for step-dancing as well as studying Gaelic Literature at Columbia University.

Martin Connolly

Having graduated from Queens College in 1959 with a B.S. in Chemistry, Marty went into the U.S. Air Force for two years. During that time he accumulated 200 hours of flying time, worked in the ammunitions section and managed to spend time in Japan, Manila, Korea, Georgia, Texas and Colorado. Prior to joining CUC as a programmer trainee, he was employed by Inter-Chemical, IPI (International Printing Inks) in New Jersey for a year. There he was engaged in developing inks used on various commercial wrappers.

At present, he is attending CCNY two evenings a week where he is studying electrical engineering as well as participating weekly in a bowling league. Other favorite sports include judo, karate, skiing, ice skating and hunting.

Robert McLean--Programmer Trainee

After graduating from City College in 1957 with a degree in Electrical Engineering, Bob went to Witchita, Kansas for a year where he worked as a junior engineer for Boeing Airplane Company. Given an assistantship by Oklahoma A & M, he left Kansas for Stillwater, Oklahoma where he taught first year engineering. Returning to N.Y. shortly thereafter, he taught Mathematics at Van Buren and Forest Hills High Schools for 3 1/2 years and has just recently received his M.A. in Mathematics from N.Y.U.

Lora Naigles -- Programmer Trainee

Lora comes to CUC with a strong background in music! She has been studying piano for twelve years, has taught folk singing and guitar at summer camp and in 1958 spent the summer studying music in Aspen, Colorado. Prior to joining

-13-

CUC, Lora worked for Music Book Associates, preparing layouts, setting up pages, etc. She graduated from Chicago University in 1960 with a B.A. in Political Science and obtained an M.A. in History from Columbia University in 1962.

Al Nelson--Programmer Trainee

A wanderer, Al was born in N.Y., alternately growing up in this city, Virginia, New Jersey, and Florida. He graduated from the University of Florida in 1962 having majored in Physics and Mathematics. Prior to entering the University of Florida, he served in the U.S. Navy for two years. There he worked as an electronics technician being stationed in Oklahoma, Maryland, Tennessee and Key West. A member of the Lisa Lane Chess Club (located in Greenwich Village), he usually plays three times a week. Lisa Lane is, by the way, the current woman's chess champion of the U.S.

Gene Whitley--Analyst

An analyst, Gene joined CUC after working 4 1/2 years for RCA. There he was engaged in pre and post sales systems development for customers and prospective customers; programming customer applications for the RCA 501; assistance to customers in planning computer sites, in selecting computer personnel and in planning optimum computer utilization; training company and customer personnel in the use of Service Routines, Automatic Assembly System (pseudo code), COBOL, console operation and systems analysis. Prior to his work at RCA, Gene was employed with the Bureau of Ships, Navy Department, for seven years. There he worked as an analyst and programmer on the Univac I, IBM 705 and 650. Working evenings, days he attended Howard University in Washington, D.C., where he majored in English literature-his particular preference being American writers of the 20th century (Faulkner, Lewis, Wolfe, etc.). Favorite sports are tennis and swimming, excelling enough in the latter to make the varsity team. He is married, has three children and commutes every day from Philadelphia.

NEWS FROM WASHINGTON

5 8D 11

Since the start of 1963, Washington has added several employees to our staff. They are:

Joe Ciervo, Senior Programmer, who comes from Philadelphia and was previously employed by the Defense Industrial Supply Center there.

Ken Fulton, Mary Grover, and Denis Reilly, programmer-trainees. Ken and Mary were both mathematics teachers in Washington before coming with CUC. Denis is also a native Washingtonian and was formerly employed by Raytheon Corporation.

Carol Allan, technical typist, who joined CUC after working for Page Communications Engineers, Inc.

This brings the Washington staff to 42, and several more new employees are already clated to join the staff during March and April.

. . . .

Our rapidly expanding staff necessitated expansion of office facilities. We were most fortunate in obtaining additional space right next door to our present rooms, and we hope to assume occupancy before the end of March.

Besides office space, we are obtaining space in our present building to house the 8-K 1401 with 6 tape drives now on order for Washington and expected in late April.

. . . .

CUC is continuing to work with the Defense Supply Agency now that they have established their offices in Battle Creek, Michigan. Jim Smith, Dale Rheineck, Sam Sutton, Jim Mauchly, and Jim Mulligan are all assigned to DSA at Battle Creek. Diana Hunt and Jack Halley have made several trips to assist the effort.

The DSA contract, which began in April 1962, has involved a complete reanalysis and reprogramming of the DSA cataloging system. Among the areas covered were the complete redesign of the paper tape input system and the conversion from IBM 705¹s to 7080¹s, including many system revisions and modifications, However, the actual programming effort has only been a part of the work CUC is performing for DSA. The move to Battle Creek has meant nearly a complete turnover of DSA personnel. CUC has been filling this gap temporarily with operators as well as programmers and has been training new DSA programmers and operators to take over the workload. Since the move to Battle Creek, the CUC staff working on the project has been considerably reduced, and as soon as DSA has enough trained personnel to handle the system, CUC will phase out completely.

• • •

NEW RELEASE BY BURROUGHS CORPORATION

Shortly Burroughs will release a new revolutionary random access storage system. Here are some of the highlights:

Head-Per-Track

Variable Length Records

Huge Capacity

Economical Expansion

Communications and Interrogation

Reliability

Programming

Every track of information on each disk has an individual permanently located read/write head. Since arm movement and positioning time are eliminated, average random access time to record is 20 milliseconds. Data transfer rate of 100,000 characters per second allows the files to be rapidly dumped into or from core storage.

A flexible addressing scheme provides for variable length records. Thus storage is efficiently used, and total file and cost requirements are minimized.

Up to 960 million characters allows extensive Management Information Systems to be developed with all required data on-line and immediately available.

Storage is available in economical increments of 9,600,000 positions that can be field installed. The system can be easily tailored to meet expansion or contraction requirements.

Extensive Data Communications System provides for centralized data processing. Reaction and response is immediate to remote locations. The Data Communications System can also be field installed.

With the elimination of delicate arm positioning mechanisms, the major cause for equipment failure is removed. Complex mechanical linkages are replaced by relatively simple and reliable electronic circuits. Hence, a dramatic improvement in reliability is achieved.

Comprehensive programming aids include Disk Sort Generator, Report Generator and various Operating Control Programs.

WHO'S GOT THE ZEBRA, ETC?

For a welcome diversion from creating tape files and producing statistical reports -- or as training for just that kind of mental gymnastics -- we can't resist passing on this brain teaser below. The authors call it a logical problem, without tricks or double meanings. They say it's been done in twenty minutes. Have a try.

THE FACTS:

2

- \checkmark 1. There are five houses.
- 2. The Englishman lives in a red house.
- -3. The Spaniard owns a dog.
- -4. Coffee is drunk in the green house.
- ___ 5. The Ukranian drinks tea.
- -6. The green house is immediately to the right of the ivory house.
- -7. The Old Gold smoker owns snails.
- -8. Kools are smoked in the yellow house.
- 10. The Norwegian lives in the first house.
- —11. The man who smokes Chesterfields lives next door to the man with the fox.
- \sim 12. Kools are smoked in the house where the horse is kept.
- -13. The Lucky Strike smoker drinks orange juice.
- 14. The Japanese smokes Parliaments.
- -15. The Norwegian lives next door to the blue house.
- 16. Each man has one house, one pet, one type of smoke, and a different drink.

THE QUESTIONS:

- 1. Who drinks water?
- 2. Who owns a zebra?