Information Communication System*

V. N. SCHRODT,** R. J. BALSKE, G. L. EILRICH, H. L. HYNDMAN, J. T. MORAN, J. C. SCHAEFER, E. R. STEWART, and W. H. WALDO
Agricultural Research and Development Department, Monsanto Co., St. Louis, Mo. 63166

Received April 19, 1971

An existing information storage and retrieval system was extensively revised to provide improved service and to serve as a vehicle for the communication of chemical and biological information among members of the Agricultural Research and Development Department. The resulting system being used by members of this department is described.

The object of this effort by the Agricultural Research Department was to convert an existing information system to a fast response communication system and to expand substantially the whole operation. The existing service was basically a vehicle for data storage and report generation with some search capabilities.[1,2] These latter included a chemical substructure searching ability.[3]

The initial effort was directed toward involving all sections of the department, and a member of each section was assigned to a computer task force. Only one of these people had any prior computer experience except for an instant Fortran course, so the first phase was educational. During this period, one new member was added to the group whose function was to be the computer expert. At a later stage, another person was added who was to become the supervisor of the system. There were seven people involved in the task force.

The training schedule was one used successfully before, and the sequence and timing are listed in Table I. This training sequence is very flexible and can be as elementary or as advanced as required depending on the particular need at the time. The BASIC language is an excellent language for the introduction of computer concepts and to demonstrate how to manipulate various parts of the system. The use of time-sharing overcomes most of the inhibitions of the neophyte computer user. The approach to Fortran was to teach formatting first followed by everything else, which takes care of the biggest problem that new users encounter in going from BASIC to Fortran. Upon completion of this training, most people are equipped to handle their own programming needs.

Table I. Training Schedule

BASIC Language
5 days—2 hours formal instruction/day followed by practice problems
10 days—intensive practice work
Fortran Language
5 days—2 hours formal instruction/day followed by practice problems
10 days—intensive problem solving

A fundamental part of the project was to recode all of the chemical structure data using the Wiswesser Line Notation (WLN) approach. A consultant was hired to do this and to teach one-week courses on a twice monthly basis until all the needs of the department were satisfied. The approach was that everyone would be responsible for doing his own coding and searching so a number of weeks were required beyond the week devoted to members of the computer task force.

Past experience with computer time-sharing indicated that a combination of time-sharing and remote batch terminal access would satisfy present and future needs. The intent was to provide easy access to a range of user needs and to achieve fast turnaround so that needed answers would be obtained in a few minutes. The computer services used are listed in Table II. One of the most important criteria in addition to that of easy access was to provide for error recovery, and only those services that could both handle user errors and allow for their correction were considered. This capability overcomes one of the more basic frustrations of any user-oriented computer operation, which is the loss of hours of work because of a typing error.

The assumption was made that the users would not be of the push button variety and would be knowledgeable with input-output techniques and error recovery procedures.

The WLN was selected primarily because there was a detailed book of rules[4] available which adequately covered the line notation and its use. The coding of the file of compounds was done by an outside contractor over a six-month time interval. This worked out quite well for the intended purposes.

Coding of new compounds is done by the chemist submitting the compound. No difficulty has been experienced with this procedure to date. All the information system supervisor does is check the input sheet for accuracy. The data are keypunched at this point unless errors are present in which case the sheet is returned to the submitter for any necessary correction. After this operation, the notation is checked by a computer program, and it may again be returned for error corrections. The error rate at this stage runs less than 3%.

Table II. Computer Services Provided

Service	Computer
Time-sharing	XDS-940
	GE-235
Remote batch	CDC-6600
Combination	CDC-6400
	XDS-Σ7
	CDC-1700

The search routines were originally written in SNOBOL, which is an interpretive string manipulation language, and these routines were tested on a file of 700 Wiswesser Line Notations. The SNOBOL language has extensive and elegant pattern-matching capabilities, and any substructure that was desired could be found with virtually com-

*Presented at the ACS/CIC Joint Meeting, Toronto, Canada, 1970.
**To whom correspondence should be addressed.

plete accuracy. Unfortunately, SNOBOL is an interpreter and not a compiler and is exceedingly slow. It took 3.5 minutes of time on a CDC 6400 to search the 700 notations. To circumvent the execution speed problem those features of SNOBOL which were needed were recoded in Fortran.

The search algorithm (SRCH) used for conducting searches for specified chemical moieties operates in conjunction with a program named DECODE. DECODE itself includes two subroutines and reads the input card for the search and, from the WLN and the logical operators included in it, creates a list of patterns. The flowchart for the search algorithm is shown in Figure 1.

DECODE will accept up to 100 patterns each of which may be a maximum of 100 characters on a maximum of 10 cards. It scans the input card for right parentheses, noting the left parentheses. Upon finding a right parenthesis it removes the characters back to the closest left parenthesis—i.e., the pattern—and checks the pattern list to see if the pattern is already there. If it is not present in that list (SLIS), it is entered and assigned an ID number. If it is already present on the list, it is disregarded thus preventing the searching for the same pattern twice. The ID number is then inserted into its corresponding position in the input line. (This line then becomes the ANDOR subroutine, which is written and compiled automatically for each search.) This is continued until an ↑ is encountered, signifying the end of input. The program also checks the numbers of left parentheses and right parentheses, both of which it has noted, for agreement indicating proper format for the input card. If the number of left parentheses and right parentheses do not agree or if an ↑ is not encountered after 800 characters, the job aborts.

Having created the list of input patterns for SRCH, DECODE sorts them, longest first, and uses the sorted list as actual input to SRCH. The input line using the ID numbers is used as the logical operation statement of a subroutine, written and then compiled within DECODE, called ANDOR. After a line of notation has been searched for all patterns, SRCH calls ANDOR to verify that the input logic is true. (If so, SRCH then prints the notation line.)

The program SRCH then reads in the patterns from the list created earlier and prints them in the output file. In each case, SRCH makes the following checks: if the pattern is one character long, if the first or last character is a blank, and if the first character is an ≡ symbol. If none of these conditions is true, then the pattern is reversed, character by character, and the new pattern is added to the pattern list and associated with the same ID number as its counterpart for use in the ANDOR logical operations. If a pattern meets any of the above conditions, the pattern is not reversed to avoid complications and conflicts within the WLN and to give the operator an arbitrary choice (by use of ≡) as to which patterns he does not want reversed. The complete notation including contractions is used.

After all the patterns have been read, the logical input pattern is printed on the output file. The first line of the notation file is read, and the first character of each pattern is compared with the first and then successive characters of the notation. When the first character of a pattern is found to match a character in the notation, the second character of the pattern is compared with the next character of the notation. This procedure is continued until a character does not match or the entire pattern is matched. If a pattern does not match, it is marked as "false"; if it does match, it is marked as "true." After all patterns have been tested on a given line of notation, the ANDOR subroutine is called, and the logical expression is evaluated as true or false. If it is true, the line is printed on the output and the next line of notation is read. If it is false, the printing is omitted and the next line of notation is read. The

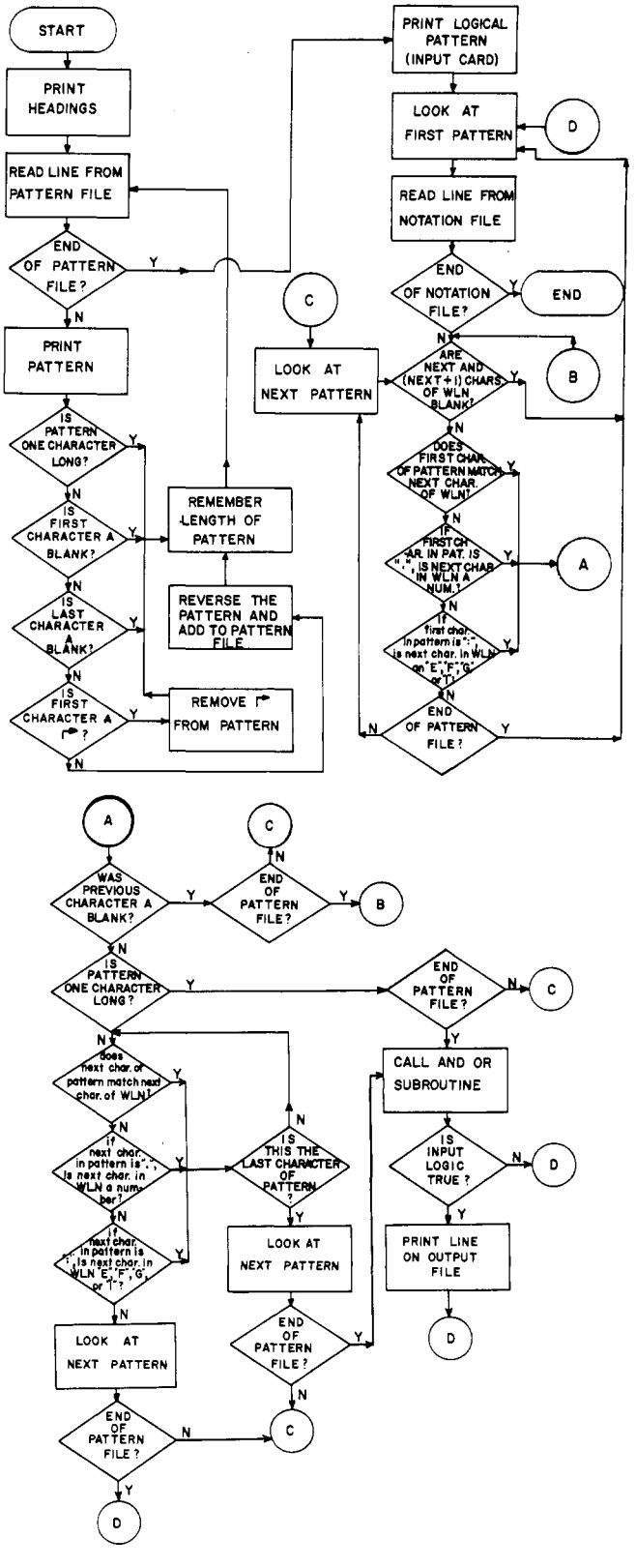

Figure 1. Flowchart of structure search program

patterns are then tested, as before, on the next line. This procedure continues until the end of the notation file when the job is ended. The output file is then printed on the line printer.

The operation of this program has been designed to involve as little card handling and file manipulation as pos-

sible by the operator. The compiled versions of the operating programs are stored on magnetic tape. The notation file is stored on magnetic tape and must be transferred to a common disc file prior to conducting a search.

After doing this, the system is ready to accept the job of conducting a search. After carefully selecting the patterns to be searched for, the user determines the logical operators he wishes to use. The functions .NOT., .AND., and .OR. are available. The order of their evaluation is shown; however, this precedence can be changed by using parentheses to group these operators just as one can use parentheses to group arithmetic operators. The patterns themselves must be enclosed in parentheses and the logical expression must be ended with an up-arrow (↑) as shown.

(A).AND.(B).AND.(C) ↑
(A).AND.(B).OR.(C) ↑
(A).AND.((B).OR.(C)) ↑
(A).AND.(B).AND. .NOT.(C) ↑

Table III. Aldrich Card File Search

Structural Patterns Searched for: ≡T50J
B
Logical Patterns Used: (≡T50J.AND.(B)) ↑

Results

10701-8	T50J B CV02 EVH
11179-1	T50J BYUNQV-BT50J
11630-0	T50J B1 MPO+02+02
11820-6	T50J B1UYVQSH
11821-4	T50J B1YVQUNQ
11822-2	T50J B1CN
12985-2	T50J BV01
13044-3	T50J BVMZ
13731-6	T50J BVN E
13802-9	T50J BV 2
13827-4	T50J B01
14280-8	T50J BV01 E1G
14411-8	T50J BV1 C10V1 D10V1
14542-4	T50J BVH E10V1
A01625-4	T50J BV1
B06740-6	T50J BE EVQ
E02850-1	T50J BV02
F01950-7	T50J BVH
F01957-4	T50J B1U1 CN
F01990-6	T50J B1Q
F02000-9	T50J B1Z
F02010-6	T50J B1M2SH
F02040-8	T50J B1SH
F02050-5	T50J BVQ
F02060-2	T50J B1U1VH
F02080-7	T50J B1U1VQ
H04080-7	T50J BVH E1Q
M04684-5	T50J B
M04685-3	T50J B1M1
B02235-5	T-50J BYM2+1R
F02045-9	T50J B1MVMSWR D
N01723-8	T6N CNJ DQ F BMNU1- BT50J ENW
N01730-0	T5NMVOJ E- BT50J ENW
10124-9	T6NYN EYJ BUS DQ FQ EU1- BT50J

Table IV. Aldrich Card File Search

Structural Patterns Searched for: BZ
FZ
≡ZR
Logical Pattern Used: (≡ZR).AND.((BZ). OR. (FZ)) ↑

Results

10887-1	ZR BZ DG
10889-8	ZR BZ DNW
B00410-4	ZR BZ DZ EZ + GH 4
D01238-4	ZR BZ D- 2 TECH
D01240-6	ZR BZ D- 2 +QHQH +GH 4 TECH
D01260-0	ZR BZ DVQ
D02602-4	ZR BZ D
D02603-2	ZR BZ D +GHGH
D07160-7	ZR BZ DG EG
D17660-5	ZR BZ D E
M02040-4	ZR BZ D01
N02130-8	ZR BZ CNW
P02393-8	ZR BZ

Table V. Aldrich Card File Search

Structural Patterns Searched for: QX1U1
≡XQ1U1
Logical Pattern Used: (QX1U1). OR. (≡XQ1U1) ↑

Results

13681-6	QX1U1
H05960-5	1Y+U3 Y+U3XQ1 U1
L00260-2	1 Y+3XQ1 U1

Table VI. Aldrich Card File Search

Structural Patterns Searched for: CZ
EZ
≡ZR
C
E
R
Logical Pattern Used: ((≡ZR).AND.((C). OR. (E))). OR.
((R).AND.((CZ). OR. (EZ))) ↑

Results

10024-2	ZR CQ
10080-3	ZR C01R
10135-4	ZR BF E
10162-1	ZR CG B
10164-8	ZR CG D
10191-5	ZR CZ D
10202-4	ZR BG EG
10214-8	T C666 BN INJ BR+ EZ F L MZ +G
10216-4	QVYZR CQ DQ
10225-3	ZR B E
10328-4	ZR C F01
10355-1	ZR DR+ CNW
10432-9	ZR BF CF DF EF
10434-5	ZR BF CF EF FF
10554-6	T4MN DNJ CZ ER
10831-6	ZR CSW0 +-NA- TECH
10948-7	ZR BQ ESWQ
11098-1	ZR CZ DNUNR +GH
11383-2	ZR CZ EVQ +GH +GH
11584-3	ZR B CNW
11644-0	ZR DYCN+R C +GH
11729-3	ZR CG D FSWQ
12058-8	ZR CZ DNUN 2 CR +GH +GH
12142-8	T C666 BN INJ AR+ EZ F L MNUN- BL66J CQ +G
12146-0	T C666 BN INJ EZ FMR DN1+1 MN1+1 +GN
12211-4	ZR DQ C EY +GH
12220-3	ZR CZ DNUNR BQ CNW ENW TECH
12221-1	ZR CZ DG FNUNR BQ CNW ENW
12242-4	ZR CV1 TECH.
12286-6	ZR DG CVQ +WSQQ
12299-8	ZR C FVQ
12517-2	ZR EZ BG
12588-1	ZR BE EE

One must be very careful in using the .NOT. function as it will override all other conditions and, if met, will discard a result even though acceptable alternative conditions are also met. The logical expression may include up to 10 80-character cards. A single pattern may consist of up to 100 characters and the search may consist of up to 100 patterns (up to 50 patterns, if all 50 are reversible patterns).

After the logical expression has been determined, it is keypunched by the user and this input card is then inserted into the operating deck as the next-to-last card. Several examples of searches made of the Aldrich file are listed in Tables III through IX. The Aldrich Co. catalog is available coded in WLN on punched cards and has about 10,000 structures.

As many of our biological test data as possible are being entered either directly to a computer for processing and storage or collected in machine readable form and then transmitted for subsequent operations. Mark sense cards, punched paper tape, punched cards, and cathode-ray tubes with keyboards are used for data entry.

In general, the computer programs, which operate as the dynamic part of the system, collect, interpret, and cast the data into a final format. The first 20 characters of this format are fixed and are the key to the system. This record

Table VII. Aldrich Card File Search

Structural Patterns Searched for: VG
Logical Pattern Used: (VG) ↑

Results	
10377-2	GVR-/F 5
10384-5	GV1R-/F 5
10391-8	GVR BG
10427-2	L E5 B666 LUTJ A E FY+1 U1 Y2+Y 00VG
10449-3	GV1G
10663-1	GVR D
10840-5	GYGV01 99PC
10961-4	GV3
11190-2	GVR DG
11193-7	GVR BG DG
11194-5	GVR CG DG
11220-8	WNR DVG
11415-4	L55 CU ATJ FVG GVG
11418-9	GV1
11772-2	L66 B6 1 B A B- CTJ BVG
11993-8	GV01R
11994-6	GVR DF
12084-7	GVR BF
12087-1	GVR DVG
12201-7	GVR B
12225-4	GVR C
12482-6	NCR DVG
12524-5	GV4G
12766-3	WNR CVG
12778-7	GV1U2
13096-6	GYR+VG
13178-4	GV8VG
13244-6	20VY3GV02
13401-5	WSFR CVG
13430-9	L3TJ AVG BF-T
13638-7	WSFR DVG
13736-7	GYGVYGG
13739-1	GXGGV1
13860-6	GYGVXGGG
13912-2	GVY
14029-5	GV9
14132-1	L C666J BVG
14142-9	GVR-/G 4 DVG

Table VIII. Aldrich Card File Search

Structural Patterns Searched for: OVG
VG
Logical Pattern Used: (VG). AND. . NOT. (OVG) ↑

Results	
10377-2	GVR-/F 5
10384-5	GV1R-/F 5
10391-8	GVR BG
10449-3	GV1G
10663-1	GVR D
10961-4	GV3
11190-2	GVR DG
11193-7	GVR BG DG
11194-5	GVR CG DG
11220-8	WNR DVG
11415-4	L55 CU ATJ FVG GVG
11418-9	GV1
11772-2	L66 B6 1 B A B- CTJ BVG
11994-6	GVR DF
12084-7	GVR BF
12087-1	GVR DVG
12201-7	GVR B
12225-4	GVR C
12482-6	NCR DVG
12524-5	GV4G
12766-3	WNR CVG
12778-7	GV1 U2
13096-6	GYR+VG
13178-4	GV8VG
13401-5	WSFR CVG
13430-9	L3 TJ AVG BR -T
13638-7	WSFR DVG
13736-7	GYGVYGG
13739-1	GXGGV1
13860-6	GYGVXGGG
13912-2	GVY
14029-5	GV9
14132-1	L C666J BVG
14142-9	GVR-/G 4 DVG
14251-4	GV2E

Table IX. Aldrich Card File Search

Structural Patterns Searched for: .V:
OVG
Logical Pattern Used: (.V:). AND. .NOT. (OVG) ↑

Results		Results	
10384-5	GV1R-/F 5	C08110-1	GV1U1R
10449-3	GV1G	F01690-2	20VYGV1
10961-4	GV3	E05020-5	FXFFV1V02
11418-9	GV1	G00460-8	GV3VG
12524-5	GV4G	I02930-1	GVYU1+1VG
12778-7	GV1U2	M00160-1	GV1VG
13178-4	GV8VG	M00965-3	GV101
13596-8	EV1	M03535-5	GV3V01
13739-1	GXGGV1	O00473-3	GV7
14029-5	GV9	P00007-8	GV1 5
14087-2	EV3VE	P01675-3	GV1R
14251-4	GV2E	P05155-9	GV2
A02410-9	GV1U1	P07660-8	NCV1
B05641-2	EV1E	10382-9	GV1R-/F 5
B08880-2	GV1X	10395-0	GV2R-/F 5
C01104-9	GV2VO1	10501-5	FYR+YEV1Y
C03060-4	GV3G	S00645-2	GV2VG
C06912-8	GV2G	T06280-4	FXFFV1

contains the compound number, dates tested, type of test, etc. The remaining part of the record is variable and may change from time to time to accommodate changes in our testing procedures.

The data in this final state are kept on magnetic tape and are available for reports, searches, etc. In general, searching is quite easy, being mainly a read operation followed by a set of logical IF statements and a write operation. Reports describing test results are circulated to the laboratory personnel on a regular basis.

The reports may be more or less complex depending on the form and use. As anyone familiar with this business knows, the logical or operating part of a search program is generally rather simple. The complications arise if one tries to accommodate a user who does not know the programming techniques and who insists on a fancy tabular presentation of results.

The data and computer services provided are available for more esoteric uses and these are limited only by the users interest and expertise. Extensive computational ability is provided, and the computer power available ranges from simple library routines on the GE-235 to the number-crunching capability of the CDC-6000 series machines.

LITERATURE CITED

(1) Waldo, W. H., R. S. Gordon, and J. E. Porter, Routine Report Writing by Computer," *Am. Doc.* **9**, 28–31 (1958).
(2) Waldo, W. H., and M. DeBacher, "Printing Chemical Structures Electronically: Encoded Compounds Searched Generically with IBM-702," International Conference on Scientific Information, Washington, D.C., Area 4, pp 49–68, Nov. 1958.
(3) Waldo, W. H., "Searching Two-Dimensional Structures by Computer," *J. Chem. Doc.*, **2**, 1 (1962).
(4) Smith, E. G., "The Wiswesser Line-Formula Chemical Notation," McGraw-Hill, New York, N.Y., 1968.

Document Access*

B. H. WEIL
AID Company and Literature Information Center, Esso Research and Engineering Co., Linden, N. J.

Received April 14, 1971

Long-range, copies of needed documents will be rapidly and inexpensively supplied to users in libraries or at their desks by querying an electronic network linking document sources (central libraries and publishers). In the interim, local libraries will slowly progress through increased dependence on local holdings of microfilm to development of and dependence on regional, national, and discipline networks for access to most of the documents desired. These developments will depend, however, on resolution of the copyright problem by mechanisms that will fairly remunerate the copyright owners. As background, and because radical change is not expected overnight, the paper also reviews conventional and sophisticated storage systems, including microfilm, facsimile, and video; copyright aspects; costs; need for speed; other user considerations; and standards.

Libraries, file rooms, depositories, and archives have long been the sources to which chemists and others turn for desired documents—in person, in writing, or by wire. To meet this demand, local sources of documents have usually tried hard to possess copies of most of the documents normally requested. On-demand purchasing or borrowing have come next, with photocopies employed almost universally as a substitute for the latter.

Lately, however, stress has been placed on this system by the vast number and growing cost of documents available and issuing, the intensification of demands caused by improved methods for identifying relevant documents, increasing costs and charges for present access routines, increasing impatience with the time required for these routines, and increasing establishment of branches which need substantially the same document services. These trends have coincided with a growing awareness of the potentials of modern techniques and equipment for some of these routines—lower cost, better quality facsimile; closed-circuit TV or videotape files, with videotape buffers; large-scale and remote-access systems employing microfilm and microfilm reader-printers; and even computerized files, replete with cathode-ray-tube or microfilm outputs. Accompanying this has been the growth of state and regional library networks using conventional methods but slowly experimenting with the newer ones.

Some of the problems and potentials inherent in this

*Presented before the Division of Chemical Literature, 161st Meeting, ACS, Los Angeles, Calif., March 29, 1971.