

Software AG International Users' Group Newsletter

Published by the International Software AG Users' Group and Software AG of North America, Inc.

Vol. 6, No. 3, February, 1981

Users' Group News

Technical Agenda for Innisbrook

With Education the theme of Innisbrook '81, Software AG is committed to offering a highly technical and informative agenda.

In keeping with this commitment, specially developed tutorial sessions addressing areas of user interest and concern will be offered. The Software AG Education and Consulting Department will develop and present highly specialized 1-1/2

hour sessions. While scheduling details have not been finalized, Software AG expects that each class will be offered more than once during the four-day conference. Topics include:

- Restart/Recovery for ADABAS V4.1
- ADABAS Tuning and Optimization
- ADABAS V4 Operating Environment
- Tuning the COM-LETE System
- On-Line System Design Considerations (applicable to all TP systems)

- NATURAL Screen Design Techniques

In addition to these, Software AG presentations will broach a variety of technical topics. Attendees will have the opportunity to hear "status and futures" for all products, updated information on the Data Base Machine and channel-to-channel software, and DBM performance, ADABAS-M technical overview, and NATURAL V1.2 considerations.

Workshop topics have expanded to reflect current user interests and concerns. Software AG will chair an ADABAS De-

(Continued)

debugging Tools Workshop, describing ADAICK and ADALOG as debugging aids. New user workshops will include Data Administration, Staffing and Operation, Managing The NATURAL Environment, and Techniques for Teaching NATURAL to non-data processing personnel. Users interested in participating in these new workshops should contact Cathy Vivona at Software AG.

Informal one-to-one technical support and consulting services will be available as regularly scheduled conference sessions. Technical Support sessions will provide a convenient way for users to meet with Technical Support Representatives to review technical support structure and procedures and resolve questions regarding product use. The Education and Consulting staff will be available to review your file design and to discuss your project implementation plans and concerns. Additionally, users will have the unique capability of discussing their DDP and Networking requirements with Dr. Dixon Doll, an expert in this field.

User presentations on experiences with all Software AG products will complete this technical agenda. Topics covered in these presentations will be made available in future conference literature.

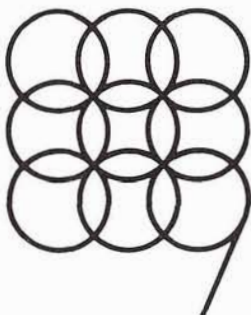


Table of Contents

Users' Group News	1
Technical Agenda for Innisbrook	1
President's Message	3
Three Speakers Augment Technical Agenda	3
Users Participate in Conference Planning	4
Regional Groups Maintain High Activity	4
Conference Registration Form	5
Hotel Registration Form	7
Transportation Reservation Form	9
Regional Reports	11
ADABAS News	21
ADABAS-M Helps Track Artists' Royalties in the U.K.	21
Online Data Dictionary System	22
ADABAS Technical Notes	23
ADABAS/VM—An Overview	23
Frequently Encountered Problems—V4.1.1	24
ADABAS V4.1 Experiences	28
ADABAS Product Representative's Report	30
ADABAS Special Interest Group Chairpersons	30
COM-LETE News	31
COM-LETE Technical Notes	31
Security Under COM-LETE	31
COM-LETE At Franchise Tax Board	33
Experiences with COM-LETE and VSAM	33
COM-LETE Special Interest Group Reports	37
COM-LETE Product Representative's Report	38
COM-LETE Special Interest Group Chairpersons	38
NATURAL News	39
NATURAL V1.2 Enhancements	39
NATURAL V1.1 Update Applications and ADABAS V4.1 Transaction Logic	39
NATURAL Technical Notes	41
Last Call for Languages	41
NATURAL CMPEXIT Revisited and NATURAL With No Data Base	42
NATURAL Special Interest Group Reports	44
NATURAL Special Interest Group Chairpersons	45
NATURAL Product Representative's Report	46
Software AG News	47
Technical Services	47
Note of Thanks	49
Software AG Announces Organizational Changes	50
Publications Update	51
1981 User Education Plans Announced	51
Software AG Class Schedule	52
Software AG Introduces	54
Current Release Levels	54
Regional Representatives	55
International Software AG Users' Group Executive Committee	56
Dates to Remember	56



President's Message

President's Message

Michael Berman
INCO Ltd.

The Executive Committee of the Software AG International Users' Group met in Reston, Virginia on October 20-21 to review the accomplishments since the last Conference and to plan out the required work for the next eight months, leading us into the next Conference. I would like to take this opportunity to thank the individuals who attended this meeting for their hard work and for making this meeting a success. The attendees included myself; Ellen Kasun, Vice President; Jennifer Lee, ADABAS Product Representative; Alan Routhier, COMPLETE Product Representative;

Steve Froneberger, NATURAL Product Representative; Peter MacRoberts, Change/Enhancement Coordinator; David Berg, North American Area Representative; and Cathy Vivona, Users' Group Liaison. In addition to the Executive Committee members, the following individuals were invited to the meeting to give presentations: Bob Taylor, Technical Support Committee Chairman; Dan Nolan, to discuss the User Profile System; Ann Rodriguez, to review future Conference sites; and Edward Forman, to review Software AG Technical Support.

Bob Taylor reviewed the purpose of the Technical Support working committee and discussed the future plans of the Group. Dan Nolan discussed the Users' Profile System and reviewed enhancements which he

will make to the system. These enhancements will allow users to input more data regarding their installation; however, only the basic information will be required to be part of the system and, therefore, receive the User Profile Report. Edward Forman discussed Software AG Technical Support and other questions which arose during the meeting. The final presentation was given by Ann Rodriguez which dealt with an update on Innisbrook and a discussion of future Conference sites.

As you are all aware, the Conference is only a few short months away and the deadline for Change/Enhancement ballots is March 6, 1981. Please make sure that your ballot is received before this date to allow it to be counted in this year's voting.

Three Speakers Augment Technical Agenda

Supplementing a full technical conference agenda, three speakers will participate at Innisbrook. These speakers will add breadth to each attendee by discussing state-of-the-art technical topics, common user concerns, and means for personal and professional growth.

Dr. Dixon Doll, an expert in all phases of advanced computer systems and communications, will present "Future Trends in Data Base/Data Communications". Dr. Doll's credentials in this field are extensive and impressive. The author of *Data Communications: Facilities Network and Systems*, he has con-

tributed to over 1,000 papers and journals. President of the DMW Group, Dr. Doll's expertise has been shared by industry, government and academic interests. Dr. Doll's presentation will provide an overview of recent signi-



Dr. Dixon Doll

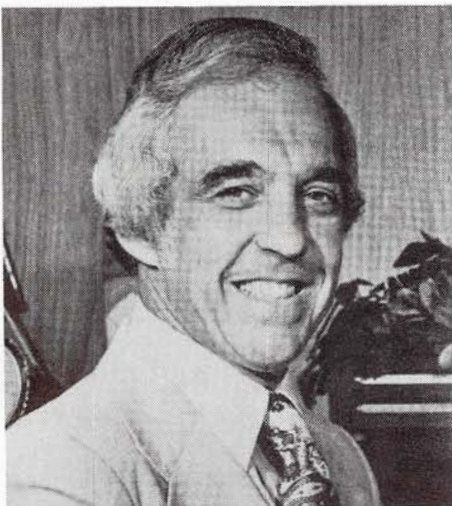
ficant developments in the data communications, distributed processing and data base area, with specific emphasis on new technology and application trends likely to produce significant benefits for the end users of these offerings.

He will present the concepts of enterprise specification, logical data analysis, data modeling, analysis techniques for designing distributed data base configurations, new offerings from the value-added common carriers, satellite communications offerings, and the general directions being taken to integrate the office automation, data processing and telecommunications functions. Additionally, local networking technologies and the role which these offerings play in the automated corporation of the future will be considered.

(Continued)

Harold Uhrbach, President of the DBD Consulting Group, will offer his experiences and expertise in organizing corporate data base administration groups by presenting "Organizational Implications of Data Base". Skilled in all aspects of data base design and implementation, Mr. Uhrbach specializes in organizational analysis and management information systems. His consulting experiences span all industries but have a specific emphasis on the application of the Integrated Data Store in manufacturing, retail, banking, transportation and medical fields. With over 25 years experience in information systems and data base development, Mr. Uhrbach's contribution to the professional growth of Software AG users will be significant.

David Yoho provides an interesting and informative break in data processing concerns. An acknowledged leader in the field of professional and personal motivation, Mr. Yoho will discuss motivational techniques and how they relate to conference attendees. He is currently a sales and marketing consultant and



Mr. David Yoho

has traveled widely addressing audiences throughout the world. Mr. Yoho's topic will provide personal satisfaction to attendees as well as adding a new dimension for them in their business environments.

Addressing the total needs of conference attendees, these three speakers will be a most welcome addition to Innisbrook '81.

Users Participate in Conference Planning

Members of the International Software AG Users' Group will play an active role in conference agenda preparation for Innisbrook '81.

Efforts for this participation will be centered around and directed by "product" coordinators. Charles Rolls (ADABAS), Alan Routhier (COM-LETE) and Jim Reiner (NATURAL) have agreed to serve the Users' Group in this capacity.

One of their major responsibilities is to ensure variety and timeliness in user presentations. By reviewing user abstracts, product coordinators may suggest speakers emphasize a specific idea or theme in their chosen presentation topic. In areas of particular user interest, coordinators may be requesting user presentations. Coordinators will work with Special Interest Group Chairpeople and Software AG to plan for meaningful and productive workshop sessions. They are also responsible for enlisting user participation in the "new" user workshops included on the Conference agenda.

Regional Groups Maintain High Activity

The Software AG users' community claims the most active network of software users in the world. Comprised of 21 regions, the international user community provides a forum for users' ideas, exchange of ideas, information and techniques. Typically, these groups meet quarterly.

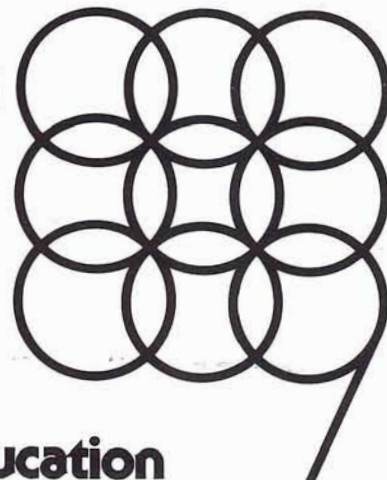
Geographic proximity of users has fostered more frequent meetings for the Northeastern Region (5-6 times yearly) and the Pacific Southwestern Region (monthly). Both groups report an active informal communications network.

One other region, the Southwestern Region, had distinguished itself by intense user interaction. Growing rapidly, this group not only had highly active regular meetings, but also boasts of the Houston Users' Group. This local group meets monthly for breakfast meetings and discussions relating to their use of Software AG products. For more than a year, these meetings have offered attendees an update on product use, techniques, problems, and resources. Houston users, chaired by Doyle Cook of Dow Chemical, find these informal sessions highly valuable in planning and "troubleshooting" to ensure optimal performance of Software AG products.

Conference Registration Form

Innisbrook 81 Ninth International

Software AG
Users' Conference
May 10-14, 1981
Innisbrook, Florida



The Olympics of Education

CONFEREE NAME
COMPANY
COMPANY ADDRESS
CITY, STATE, ZIP
COUNTRY
TELEPHONE
COMPANION NAME
FIRST NAME (OR NICKNAME)
FOR BADGE

CONFEREE _____ COMPANION _____

CONFEREE
COMPANION
NON-USER
LATE FEE AFTER APRIL 10

\$495.00
\$150.00
\$750.00
\$ 50.00

PAYMENT ENCLOSED \$ _____
PLEASE INVOICE \$ _____

PLEASE SEND INVOICE TO:
COMPANY
ADDRESS
CITY, STATE, ZIP
COUNTRY
ATTENTION
PURCHASE ORDER #

PLEASE CIRCLE

Will you be playing golf? YES NO
Will you bring your own clubs? YES NO
Will you be playing tennis? YES NO
Will you bring your own racquet? YES NO
Is this your first Software AG Users' Conference? YES NO

FOLD



No Postage
Necessary
if Mailed
in the
United States

BUSINESS REPLY MAIL

FIRST CLASS

PERMIT NO. 3115

HERNDON, VA.

POSTAGE WILL BE PAID BY ADDRESSEE

SOFTWARE AG
OF NORTH AMERICA, INC

**Reston International Center
11800 Sunrise Valley Drive
Reston, Virginia 22091
Attention: Conference Registrar**

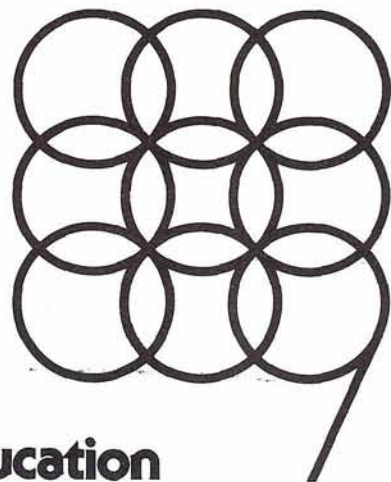


FOLD

Hotel Registration Form

Innisbrook 81 Ninth International

Software AG
Users' Conference
May 10-14, 1981
Innisbrook, Florida



The Olympics of Education

PLEASE ENTER THE FOLLOWING
RESERVATION FOR:

PLEASE RESERVE ACCOMMODATION
AS CIRCLED:

RATES

NO TIPPING POLICY

APRIL 10 CUT OFF DATE

THIS ROOM WILL BE OCCUPIED BY

Arrival Date _____ Departure Date _____

Deluxe Hotel Room	\$62.00
Club Suite	\$72.00
One Bedroom Suite	\$82.00
One Bedroom Deluxe Suite	\$92.00

Rates are based on single or double occupancy per bedroom; additional occupants 13 years of age and over, \$8.00 per person per night.

Innisbrook has a no tipping policy and the above rates are subject to a daily \$4.00 per person service charge (not including children under 18) which covers all guest services except food and beverage, where a 20% service charge (tax and gratuity) is in effect. Room rates are subject to 4% Florida State Sales Tax and a 2% Pinellas County Tourism Tax.

All reservation requests must be received **in writing** by Innisbrook no later than 30 days prior to arrival. Any **adjustment** to the original reservation must be submitted **in writing** until 10 days prior to arrival, at which time you should call reservations directly.

Name(s) _____ Phone _____

Company _____

Address _____

City, State, Zip _____

of Adults _____

Each category of accommodation is available on a limited basis. Should the category requested be sold out to other members of your group, the next higher category will be confirmed to you.

Transportation is available from Tampa International Airport, and a separate form has been provided for your convenience.

FOLD

PLACE
STAMP
HERE

Reservations Department
Innisbrook
P.O. Drawer 1088
Tarpon Springs, Florida 33589

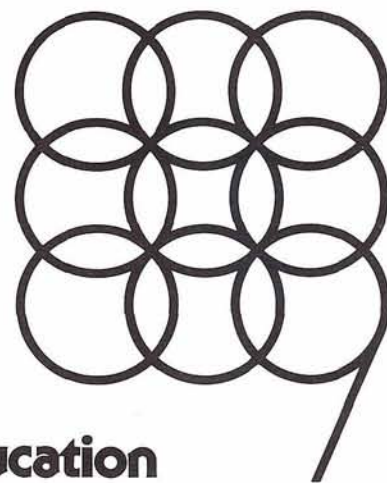


FOLD

Transportation Reservation Form

Innisbrook 81 Ninth International

Software AG
Users' Conference
May 10-14, 1981
Innisbrook, Florida



The Olympics of Education

Innisbrook is a pleasant 25 minute drive from Tampa International Airport. Two forms of ground transportation are available through Innisbrook: Shuttle Service at \$8.00 per person, each way; or Dollar Rent-A-Car. Either (or both) arrangements may be made by completing and returning this form to Innisbrook's Transportation Desk.

AVAILABLE TRANSPORTATION

NAME

COMPANY

ADDRESS

CITY, STATE, ZIP

PLEASE ARRANGE

ARRIVAL INFORMATION

DEPARTURE INFORMATION

CAR RENTAL INFORMATION

SHUTTLE SERVICE INFORMATION

APRIL 27 CUT OFF DATE

Shuttle Service _____ Dollar Rent-A-Car _____
For Dollar Rent-A-Car, the following information is required:
Driver's License No. _____ State Issued _____

Arrival Date _____ # of Persons _____
Airline/Flt. #/Arrival Time _____

Departure Date _____ # of Persons _____
Airline/Flt. #/Departure Time _____

We have arranged to have your car waiting for you at Innisbrook. Should you desire to pick up the car at Tampa International Airport, please check here _____. If so, please contact Dollar Rent-A-Car (upon your arrival at the airport) for transportation to your car.

I will use the car for _____ days and will return it to the Dollar Rent-A-Car desk at: Tampa International Airport _____ Innisbrook _____.

Guests traveling to Innisbrook via Shuttle Service report to the **ORANGE BOOTH** located just outside the baggage claim area.

This form must be returned to Innisbrook's Transportation Desk no later than April 27, 1981.

FOLD

PLACE
STAMP
HERE

Transportation Desk
Innisbrook
P.O. Drawer 1088
Tarpon Springs, Florida 33589



FOLD



Regional Reports

Far East Region

*Yasuo Shimizu
Fuji Photo Film*

Software AG (Far East) Users' Group meeting was held on October 9, 1980 at Hotel Grand Palace in Tokyo. Ninety-six users representing 46 installations were present.

Mr. Shimizu, President of Software AG Users' Group (Far East Region), presided over the meeting. He restated his belief that the Users' Group meeting should be a voluntary exchange of users' experiences so that all might grow in their awareness and use of Software AG products. He invited and encouraged active participation by many users.

Mr. Maguire, President of Software AG of North America, gave a slide presentation showing the growth of independent software manufacturers and the decline of IBM leadership in the software marketplace. He named many prominent Software AG users in the worldwide marketplace.

Mr. Maguire described his role in the recent Trade Mission to the People's Republic of China, relating the present status of computer technology in China and how the Chinese people are endeavoring to automate and computerize their business sectors. In the course of his 2 week trip, Mr. Maguire gave seminars on data base/data communications technology for Chinese technicians and government officials.

Mr. Takenaka, Manager of Sumitomo Metal Industries, reported on the Eighth International Software AG Users' Con-



Software AG of Far East users discuss product features and use.

ference and offered an abbreviated presentation on ADABAS applications at Sumitomo Metal Industries. Mr. Takenaka presented these experiences at the International Conference and was a panelist at an ADABAS Version 4.1 session.

According to Mr. Takenaka, the Eighth International Conference was more than a get-acquainted meeting. It offered the place where the makers, vendors and users got together to discuss the future of Software AG. Furthermore, it provided the opportunity to each user to promote its own technical standard and also to educate qualified engineers.

Our business discussion focused on future meetings. Many users agreed the general meeting to be a social gathering and that technical issues should be left to Special Interest Groups, such as ASSIGN (ADABAS Special Interest Group of Nippon). Users agreed that the talks on the tendencies of the software industry, software itself, Software AG directions, and new information on each product were essential to the meetings. We also discussed forming a study group for young engineers.

Australian Region

*John Palmer
Australian Bureau of Statistics*

The Australian Region met in Sydney on 27 November. Hosted by Overseas Containers Australia Ltd. (OCAL), it was attended by 19 persons representing 17 of the 13 user organizations. This, together with the very active participation by the attendees, indicates that we are strongly endorsing the existence and role of the Group.

For the first item on the agenda Mr. Les Coyne provided an interesting description of OCAL's computing operation. This operation supports the shipping of container cargo in and out of Australia and is based on an IBM 4341 (DOS/VSE) with 4M bytes memory. OCAL is in the process of implementing a "star" configuration by connecting 5 IBM SYSTEM 38s situated in their State offices to the central 4341. They plan to establish a distributed data base utilizing ADABAS, which is already in production

(Continued)

use on the 4341 and SYSTEM 38 Data Base on the SYSTEM 38 machines. Another aspect of OCAL's operation that provoked considerable interest is their singular success in limiting the programmer resource devoted to maintenance. By developing systems that can be maintained by simple adjustments to tables and other parameters, they are able to implement almost all routine modifications via a Systems Control Group of non-programmer staff. Only 4% of their total programmer resource of approximately 40 programmers is needed for maintenance of existing applications.

Mr. Coyne's presentation was followed by general business. The establishment of the Technical Support Committee was noted; however, information regarding the Committee did not reach us till after the meeting (it appears to have spent two months at sea!). The Australian SAGGROUP committee will appoint a coordinator so that we can contribute to this effort without further delay.

We are all eager to see the Early Warning System implemented.

We decided to establish a FACOM users SIG. It is of interest that there are four FACOM M-Series installations of ADABAS in Australia incorporating five users (two separate Government departments share a common installation). This SIG has a set of common concerns relating to the support of interfaces between components of the OSIV/F4 operating system and ADABAS. They are particularly interested in sharing techniques regarding the use of ADABAS in conjunction with AIM, the M-Series TP Monitor.

The General Business session was followed by an informative segment from Frank Zaknic of SPL who outlined his experiences at SAG in Germany and answered a number of questions regarding how the development of ADABAS software is organized. Bob Warburton then presented information on the Software AG NET-WORK proposal and the Data Base Machine.

Our last agenda item was a session at which members presented details of problems that have arisen in their use of ADABAS and their solutions to these problems. This segment provoked considerable discussion and has been requested as a regular feature at meetings of the group.

The next meeting of the group will be at the Australian Bureau of Statistics in Canberra in March 1981.

South American Region

Members of the South American Region met for a business meeting on Thursday, 23 October 1980. The meeting was held in conjunction with the Thirteenth National Congress of Data Processing (SUCESU) held in Rio de Janeiro, Brazil. This international conference is the major gathering for the South American data processing professionals. Representing the Software AG product line, the CONSIST exhibit included online demonstrations of ADABAS, COM-LETE and NATURAL. Natalio Fridman was pleased to announce the availability of the Data Base Machine. An overview of this new Software AG product was included in the

(Continued)



South American Software AG users meet at SUCESU.

"Jornal do Congresso", a newsletter highlighting conference activities and significant data processing news.

The Software AG users discussed plans for their annual Users' Conference scheduled for late March or early April 1981. This will be held in Rio de Janeiro. All South American users are welcome to attend. The group is also planning for various informal regional meetings in Rio de Janeiro, Brasilia, Sao Paulo and Buenos Aires. These meetings are open to all Software AG users. Please contact Natalio Fridman at CONSIST (in Brazil) or Pan American Computer Systems (in Argentina) for details.

Northwestern Region

Doug Hart
Weyerhaeuser Company

The fall conference for the Northwestern Region was held on September 12, 1980 with over 30 individuals from Alaska, British Columbia, Oregon and Washington attending. Steve Carter represented Software AG at the meeting.

The first presentation was given by Ravi Mahur of NOVA (formerly Alberta Gas Trunk Lines) on a table handling system using ADABAS. The system will load two-directional tables contained in an ADABAS file for efficient table look-ups by an application program. Two subroutines are used in this system: ATLOAD, for loading the table in-core, and ATFIND, for finding the description data for a supplied

key from the named in-core table. Utilization of the two subroutines is made via a CALL statement with a parameter string from the application program. Three types of calls can be made. The first initializes the table and loads it from the data base; the second finds a data item in the in-core table by supplying the proper index key; the third adds more items to the table for the duration of the job without affecting the permanent copy in the data base. More than one table may be loaded from the same ADABAS file by using different table names. Additional tables may be loaded from other ADABAS file(s) using different table names. The subroutines have been written in PL/1 code and are called as standard subroutines.

Hans Stoker of Administrator for the Courts gave an update on ADABAS performance tools being used currently at the Courts. Using the transaction log as origin of needed information, the following "tools" are produced on a daily basis:

1. A Command Log exception report based on a set of user-specified selection criteria including CPU time, ECB list, Associator I/O, Data I/O, Work I/O, time queued and response code. The above parameters can be adjusted via a control card for a particular run. The report is useful for analyzing commands which use an excessive amount of CPU or I/O or both.
2. A report summarizing each type of command issued against each ADABAS file. Summary fields for each command on the report are average CPU time, average

time queued, percentage of CPU time to total CPU time for session and Associator, Data and Work I/Os.

3. ADABAS Command Queue depth profiles. Two Command Queue depth profiles are printed as histograms. One is for teleprocessing and the other for batch. Each histogram gives the queue depth position, number of commands executed at that position and its percentage of the total commands issued during that session.
4. Worst-case statistics. For each session, the longest time in seconds a command was queued, worst-case CPU time for a particular command and total accumulated time in seconds that commands waited in the queue for both batch and TP.
5. Interval utilization report. At a user-specified interval (e.g., five minutes), the analyzer program summarizes all accumulated command information during that interval and produces a summary line of the following information: I/O service time, number of I/Os per command, ADABAS response time, CPU time and I/O counts for both batch and TP.

"Such reports give a very clear picture of ADABAS performance," stated Hans. Changes can now be made and their effect on the performance of the system can now be evaluated.

Jon Hudson of King County spoke next about his latest effort—development of an ADABAS interface to DMS. The

(Continued)

generalized interface is viewed as a called user program by DMS programs. The assembler-written program interprets the parameters passed to it and generates low-level calls to ADABAS. Current functions provided are inquiry on one key, update, delete and adding of records to an ADABAS file. Jon stated that the first system to use the interface is just coming up in a production mode.

After lunch, a group discussion on NATURAL impressions was conducted. Most installations in attendance did not currently have NATURAL in-house but most intended to evaluate it sometime in the future. We discussed a variety of topics including NATURAL in a CICS environment, ease of program development and the logistical problem of controlling the type of ADABAS commands which would be issued by a NATURAL program.

The final item of the day was SAGGROUP business. It was formally decided to hold our quarterly regional meetings on the first Friday in September, December, and March with a meeting at the International Users' Conference as the fourth quarterly meeting.

Pacific Southwest Region

*David Berg
Hughes Aircraft Co.*

The September meeting was hosted by Linda Simon and Tom Potts at the UCSB campus. In response to the letter from Bob Taylor, chairperson of the Technical Support Committee,

Jeff Chroman of City National Bank was elected the Pacific Southwestern Region's Technical Problems Coordinator. The users have agreed to fill out the Technical Support Incident Log forms for any significant ADABAS technical problem that is reported to Software AG and submit them to Jeff.

John Weil announced the organizational changes that have occurred recently in the Western sales region and which were reported in the last newsletter.

Walt Hopkins has observed that under MVS, an MPM can be brought up twice as fast as under VS1 and that the only changes required were to the ADASET parameters. It was noted that under Version 4.1 all the utilities will run in one update thread.

The remainder of the meeting was devoted to ADABAS security. David Berg noted that a program can only handle one password and that all secured files accessed concurrently by that program must have the same password.

A NATURAL problem discovered by Hughes Aircraft and confirmed by Nissan Motors occurs when NATURAL is run in batch single user mode and any file on the data base happens to be secured. ADABAS returns a response code 208 or 209 which is translated by NATURAL into a 311 message. This problem is awaiting resolution.

Linda Simon explained that UCSB uses dynamic passwords, where a four-character password is randomly generated for an end user and a production job reads the password into a program from an input card. Linda Simon warned that if a password is added to the security table and

the secured file is then loaded to the data base, the file cannot be accessed by that password until MPM is reinitiated. She also noted, confirmed by David Berg, that field security is maintained in the FDT and must be reapplied if a file is deleted and reloaded.

At the October meeting, David Berg reviewed the minutes of the Executive Committee.

George Kamida of Software AG presented Software AG's On-Line Support, whereby an ADABAS file that contains fields to maintain problem information may be queried for a variety of reports.

John Weil of Software AG announced the availability of the Data Base Machine with an Externally Supported Processor (ESP36) and Channel-to-Channel Communications system.

Walt Hopkins led a discussion on ADABAS V4.1 usage and features. A round table discussion took place as to Change/Enhancement suggestions.

The next meeting was hosted by TICOR and Title Insurance and Trust. Bill Sanders, MISD manager of TICOR, provided users with an organizational overview of TICOR, which has an IBM 370/168 and is converting from VS1 to MVS. Max Stratton reviewed the Realty Tax System design as a relational data base application comprising three major data groupings: contracts and loans, tax information, and legal description.

Paul Keith and Max Stratton then presented the results of benchmark tests on the various approaches to the data base structuring of the Realty Tax logical design. The tests compared the loading and retrieval times for the Realty Tax data

(Continued)

when it was grouped in three different ways: as three separate ADABAS files of contracts, liens, and legal descriptions; as two separate files in which the contract and lien data were combined as one file with the legal description as the second file; and as one file with the contract data, lien data, and legal description information combined into a single record and single file with 1/8 of the file as redundant data.

The load time result showed that it required less time on the whole to load the data as a single file than as three separate files. (Keep in mind that separately the three data groupings required 11 descriptors, while as one file only 9 descriptors were needed.) Retrieval times also revealed a substantial savings using the single file approach. It was noted that once a certain large number of records is reached in data base files, then logical or physical coupling becomes a tremendous overhead. TICOR considers 500,000 ISNs to be a large file. TICOR has found it better to store redundant data in a single large file than to couple the same non-redundant data as separate files. TICOR also realized a 45-50% savings without the batchlog facility.

Myron Pullen discussed the heuristic design approach being employed at TICOR. First of all, he found it better to use more understandable wording with users, such as pragmatic or iterative approach. The advantages of the heuristic design process as explained by Myron Pullen are the early involvement on the part of end users and consequently the submission of better user requirements and specifications with earlier suc-

cessful results in being able to produce usable reports. The only disadvantage in capturing management attention through the use of heuristic development is the sudden increase in the demand for complex management reports. TICOR is training users in ADACOM and is encouraging a hands-on approach among the user community.

Midwestern Region

George White
West Publishing

The Midwestern Region met in St. Louis, Missouri on November 6, 1980 with NAVCO acting as host. A total of fifty people attended. The meeting was chaired by Mr. George White.

Mr. White presented the report of the Executive Committee and a preview of the next International Conference. The preview of the International Conference led to a discussion of education preferences. Data Dictionary and NATURAL courses were suggested as well as a NATURAL course for non-data processing personnel. This course would replace the current method of teaching them by having them read the *NATURAL Reference Manual*.

The Executive Committee had asked each regional users' group to appoint a Technical Problems Coordinator. Mr. Dave Antes of RAY-O-VAC Corporation was appointed.

Software AG's technical support structure and procedures were discussed. Some users voiced their concern that their "first line" of support is on a

special SAGNA project and is not directly available. These users felt that a technical representative should be available at all times so users can know who is accountable for their support. Others indicated that there is currently a line of communication when problems arise and that backup for procedures for support are in place.

The group passed a resolution that Software AG should provide users with a list of all known problems for software they are currently using or installing.

We also heard a brief overview of the Data Base Machine.

Jack Reitz of Evans Products gave a presentation of how their company is using the Data Dictionary. Using the standard file to describe all fields at the company, the DBA can create new files and DDMs with NATURAL programs. Jack has also written NATURAL programs to maintain and report the Data Dictionary.

Bob Becker of Foremost Insurance spoke on the new features of Version 4. He indicated the procedures that he would use in converting to Version 4. He will be able to make the conversion transparent to the application programs by changing the dynamically loaded interface program that the application programs use to access ADABAS and making the zaps to set all the correct defaults so that the ADARUN parameters are not required.

Ron Egger of NAVCO presented his experiences with converting to Version 4. He said that by using program stubs, it was very easy for them to convert programs. He also described his experiences with ADAM files.

Wade Peterson of West Publishing Co. gave a talk on a sys-

(Continued)

tem written in NATURAL used at West for scheduling and reporting their computer hardware equipment. Standards and programming techniques for NATURAL programs were presented.

Bruce Cochran of NAVCO described how their managers have been trained in the use of ADACOM and showed the complexity of some of the programs they have coded. He stated that this is just an example of the changing role of data processing departments to only the storing and custody of the data and not the dissemination of that data.

The next conference will be held in Indianapolis during the week of February 11, 1981.

Rocky Mountain Region

*Ellen Kasun
Hewlett-Packard Company*

A Rocky Mountain SAGGROUP meeting was held on September 24, 1980 in Denver with nineteen attendees representing twelve user sites.

During the business meeting, Mr. Jerry Schade of the General Government Computer Center was appointed to the position of Change/Enhancement Coordinator for our region. Jerry will compile all Change/Enhancement requests from our region and submit them to the SAGGROUP Committee by the December deadline.

Our Education Coordinator, Larry Watson of Hewlett-Packard, will send out questionnaires to determine both the interest of having Software AG classes taught to Rocky Moun-

tain users and the user priorities for these classes.

After discussion, we implemented the following procedure: If a user registers people for a class, thereby filling possibly limited spaces, and an invoice is sent to that user charging him for the number of people he indicated, that user is responsible to pay whether his people attend or not. If that user does not pay the amount charged, his organization will not be allowed to attend our locally organized classes in the future. The intent of this policy is to facilitate scheduling and billing, not to discourage class enrollment.

Dave Johnson of the State of Utah presented "Data Base Organization at the State of Utah". The State currently has a network consisting of one Amdahl and two Itels (soon to be three) linked together via two Com-ten Communication Controllers and a multitude of peripherals. Dave discussed his philosophy of system development using ADABAS

going into detail in numerous areas. He shared much useful information with us and did an excellent job of presenting it.

After lunch, Ron Tada of Software AG provided an update on Software AG news, informing us of some changes that have taken place at Software AG. Ron answered questions about the DOS release of Version 4.1.1 and availability of NATURAL fixes. While local users generally expressed satisfaction with technical support, we suggest that Software AG periodically distribute to each user a report containing that user's reported technical problems and the status of each.

Larry Watson of Hewlett-Packard concluded the meeting with a discussion of his experiences with Version 4.1.1 to date. In summary they are:

- Coding parameters properly is critical;
- Some of the documentation is unclear and confusing;

(Continued)



Rocky Mountain users pose for a photograph.

- Security and mass-update modules were not available in the initial release.

At our December 3rd meeting, we discussed reporting schemes for technical support problems that have been distributed by the Technical Support Committee. We debated whether we should coordinate our efforts and present problems to the Committee as a group or report on an individual basis. We decided to submit our concerns on an individual basis.

Cathy Vivona of Software AG outlined the technical agenda for the 1981 Users' Conference to be held at Innisbrook, Florida. She reviewed the status of the Users' Profile project and the Product Design Committee. She mentioned the formation of two new Special Interest Groups—DBA Staffing and NATURAL Features.

Our Education Coordinator, Larry Watson of Hewlett-Packard, distributed a turnaround form intended to determine the interest in holding local classes in the January-March 1981 time frame. Highlighted on the form were the following three (3) classes: NATURAL, Structured Data Base Design, COMPLETE Internals.

Jerry Schade of the General Government Computer Center then headed up a discussion of Change/Enhancement procedures. We have decided to act as a group with our Change/Enhancement requests. At the March 6 meeting we will vote as a group.

Paul Peterson of Software AG gave a presentation on the use of ADALOG as a means of troubleshooting ADABAS applications. He pointed out things to look for on the ADALOG Report

(excessive excps or CPU time) and explained that Version 4.1 gives us a user exit so that we can create ADALOG on an exception basis (that is, only print transactions in error). Paul strongly recommended the use of READONE = YES on FIND-SETs whenever possible.

After lunch Jim Engel and Sherman Bollinger of Jefferson County Schools conducted a demonstration of the Student Enrollment System and the Mark (grade reporting, student history, class scheduling) System, both up under CICS and ADABAS. The programs are written in PL/1 and Security by Value is used to control access to the data base. The Systems have been up for about a year and enable the schools in the Jefferson County School District to operate in an online environment.

Dan Filkins then brought us up to date on the latest happenings at Software AG, describing the new marketing structure and introducing Mike Ater as the new Vice President of Sales for the South Central region. Paul Peterson will be increasing the size of his Technical Support group in Denver by January 1, 1981.

Our next meeting is scheduled for March 6, 1981 in Salt Lake City, Utah.

Southwestern Region

*Bob Holub
Superior Oil*

The Southwestern Region of SAGGROUP met October 23, 1980 in Houston, Texas.

Peter MacRoberts, the current Change/Enhancement Coordinator, gave us a report on the Software AG Executive Commit-

tee Meeting held on October 20-21. Discussed at this meeting were future Conference sites, the technical agenda for the 1981 Conference and documentation of and notification of fixes to ADABAS V4.1.

Bob Holub reviewed the newly formed Technical Support Committee. The Committee will try to quantify the degree of user technical support problems and to observe how and where the problems happen. Cleburne Fritz of Pioneer Corporation will collect this information for Committee use.

The Change/Enhancement procedure and deadlines were described by Peter MacRoberts.

Each of the Software AG users present discussed the use of ADABAS, NATURAL, and COMPLETE at his installation. Uses of the products, problems encountered and their solutions, and tools and techniques which have been developed were discussed.

Bob Holub agreed to work with Cleburne Fritz on a regional system whereby users in our area will document all fixes and problem solutions and make them available to all other regional users who participate.

As this system is developed, more information will be sent to each user.

Randy Ebeling and Greg Neunaber of the University of Texas at Austin gave an excellent presentation of their experiences with ADABAS Version 4.1 and their experiences in converting from TOTAL to ADABAS. This was featured in the last Newsletter.

The next meeting of the Southwestern Region Users' Group will be held in Dallas on February 26, 1981.

Southeastern Region

*Steve Froneberger
Duke Power*

On October 1, Southeastern SAGGROUP met in Atlanta. Twenty-five persons representing eight installations and Software AG were present. The major topics at the meeting were experiences with NATURAL and Version 4.1 of ADABAS. After brief introductions, Cathy Vivona provided information on the status and corrections of SM7 for NATURAL and planned for release 1.2 of NATURAL.

The next portion of our meeting was devoted to experiences with NATURAL from application programmers' viewpoints. George Painter highlighted NATURAL implementation at Bigelow Sanford on an IBM 370/168 system operating under OS/VS2. Together with his associate James Cheeks, George described their Total Operations Planning System (TOPS). George cited both advantages and the problems he had experienced during project development. Advantages included ease of learning, strong screen mapping facilities, elimination of the need for ADAMINT, overall interfacing capabilities with ADABAS, and faster coding (due to NATURAL's editing capabilities). Disadvantages or problems he encountered were difficulty in table handling, program size limitation, and confusion caused by misleading error messages. He also cited incompatibilities of NATURAL with structured programming design methodology. The overall opinion of NATURAL

as expressed by George is that NATURAL is very powerful and with some refinement has very good potential.

The next presentation was by James Cheeks, also of Bigelow Sanford. James had worked on the development of an order entry system using NATURAL and COM-LETE. The project had been developed in such a manner as to take advantage of the paging capabilities of COM-LETE. Data was stored in page format and any user had the ability to recall any page of data created by the NATURAL program. James was very optimistic concerning project development using NATURAL.

The third presentation was given by Steve Deese of Duke Power Company. Steve had developed a batch-update application using NATURAL. Steve expressed that he had experienced periods of frustration and optimism during the development of the project but that his general feeling was one of optimism. The problems he experienced were similar to the ones expressed by George earlier. He mentioned misleading error messages, program size limitation, and inconsistencies in documentation as his problems. Advantages he saw were ease of learning, ease of development, and ease of future maintenance. His project was developed using less time than was estimated originally despite changes and additions to the system.

Following these presentations, there was an informal discussion of the problems which had been experienced and the features needed by the users of NATURAL. These items have been forwarded to Software AG.

Al Wolfson of R. J. Reynolds

then presented his experiences in the testing of Version 4.1 of ADABAS. His overall experiences were favorable, but he had encountered problems regarding some of the new features of 4.1. Al gave suggestions for establishing parameter values if running a test and production data base, but noted that the establishment of the actual values to maximize efficiency was still mainly guesswork because the variables used in calculating the values of the parameters are often hard to obtain. Al also discussed some considerations to be taken when converting Version 3.2.1 applications to Version 4.1, ET data considerations, ADAMINT considerations, and restarting application programs.

Our next meeting will be held on February 4th in Atlanta.

Eastern Region

*Daniel Nolan
Civil Aeronautics Board*

The September, 1980 meeting was held at the National Education Association in Washington, D. C. Over 80 individuals attended, representing over 38 of the 57 clients in the area. The program began with the circulation of a master copy of the user's profiles for verification and correction. The data collected on the master is being entered into the enhanced profile system.

Regional business included a call for reformation into a more stable structure, with offices of Vice-President and Past President in addition to the Regional Representative, and selection of five Programme Directors to coordinate activities. The VP,

(Continued)

selected at the meeting following the International Conference, will be elected Regional Representative at the next International Conference, while the Past-President provides additional support for the year following. Programme Directors were selected to coordinate presentations for ADABAS, COM-LETE, ADAMINT, NATURAL and Data Administration.

Presentations by users began with the topic of Security. Ron Shirey, Guif Science & Technology, began by describing (without giving away secrets) the ways they have neatly solved the problems of having 1000 users globally, blending the security features of IBM, SAG and some home-grown protocols.

Washington, D. C. SHARE's Jim Miller followed with a presentation of their methods for insuring that security changes were rapidly implemented. The consensus of both Ron's and Jim's presentations was that Security is both technical and administrative, and that the technical cost, in performance, is minimal.

Dick Jones (PRC/NASA) presented an overview of phonetics and how they gave some therapy to the soundex routine to remove its "German accent" and replaced the mystery of it all with some predictable results.

Dick followed this with a presentation of some applications using phonetics including the State of Utah's system for validating names of new corporations.

Ann Karpick presented the Consumer's Union experiences in 4.1 conversion. With a massive data base and behemoth hardware, any problems are guaranteed to be significant.

Ann's major points (and caveats) are:

- Schedule and staff carefully;
- Establish a Disposable Test Data Base which can be recovered swiftly;
- Test all utilities and standardize operation;
- Test all applications (or variations thereof) and set standards;
- Test all new features expected in 4.1;
- Take a long vacation and sleep through it;
- Cut over cold, to the new from the old.

Ann also presented her experiences with some of the features above and invited Software AG to give their present status.

This was followed by an open session, largely dominated by ADABAS Version 4.1 and NATURAL questions. (Follow-up calls to the participants showed many dicey little problems had been cleared up and that probably as many more were prevented.)

On behalf of the other Software AG users and myself, it was a pleasure to have such enthusiastic participation and I am extending our appreciation to the National Education Association for their hospitality, their facilities and their refreshments.

At our December meeting sixty companies were represented by almost 90 users.

Regional business included the status of the Users Profile and the position of Regional VP which is yet unfilled. Change/Enhancement coordination was attempted as an open forum, but cannot be considered a success since each of the proposals required too much time for discussion. Nevertheless, many

changes were presented and submitted.

Duane Reaugh of NASA introduced us to his installation's Job Controller, which generates JCL dynamically to marry the standards to the user requirements. His system uses menu selection of options to provide symbolic overrides and presents the resultant jobstream for review before submission. Duane offered that he would provide it to the audience on request with casual support but hopefully having a "Help" command to assuage trepidations.

Steve Robinson gave us a look at possibilities for the future decade. Steve introduced such concepts as true conversational languages, the development of computer games into full-fledged processors, and the view of software as a "breadboard" or development medium which would have production systems in hard-coded (chip) form.

Bob Taylor of Evans Products (Bluebell, Pa.) introduced us to ADAMINT operations from a remote processing location - their mainframe is located in the Pacific Northwest - and to the organizational structure employed. The DBA functions in line with other operations to provide reliability and control flexibility to applications areas. Problems of storage, of support and of the "ripple effect" as one change causes others are dealt with in a manner devised to encourage dialogue between DBA and developers. Bob has found that having one ADAMINT module per file, dynamically linked, melds the question of standardization and flexibility into one easily managed set of procedures.

Harry Mackay of Sun Information (Philadelphia) related their

(Continued)

experiences as a new ADABAS installation and gave some insight into the items with which a new data base administration is concerned. He notes that the patience of his users and the assistance of the SAGGROUP community contributed significantly to this success. Sun, now using NATURAL as the sole production language, is beginning to confront some familiar DBMS problems (the users each want their own data bases) with innovative solutions (as, a Corporate data base). Sun is now setting up a "high-level" DBA with quality assurance, file delegation and standards enforcement. It will be interesting to have another presentation by Sun to see if the transition to DBMS is destined to be always as traumatic as it was with the older installations or if

the experiences of early users have benefitted the newcomers.

Ed Davidson of Uni-Coll gave the members a view of a large scale system development using NATURAL as the main programming language. Ed's claim that the system could not have been done without NATURAL was reinforced with a quick review of the project lifecycle. Resourceful programming and imaginative design stood them in good stead, as they spliced a COBOL CICS mapping product into NATURAL giving it extended capabilities. Ed acquainted us with some of the additional features incorporated, such as the ability for the user to set passwords for upcoming sessions, the master/slave relationship of the user's terminals and the use of program function (PFn) keys to select, scroll, re-

view, cancel, balance, accept and process information. The system was up and running in three weeks.

The last presentation of the day was from Jerry Hoerger of Software AG on the Data Base Machine. Following the "premier" showing of the slides, Jerry fielded such questions as price, performance and field support.

The balance of the day was devoted to a combination problem session and continuation of the Change/Enhancements of the morning with Marilou Bocobo and Duane Reaugh fielding the 4.1, internals and teleprocessing problems.

Our next meeting will be held at the Crystal City Holiday Inn on March 4, 1981.

ADABAS News

ADABAS-M Helps Track Artists' Royalties In The U.K.

*David Lane
Mechanical-Copyright Protection Society*

Mechanical-Copyright Protection Society (MCPS) is using ADABAS-M to both increase its revenues and decrease costs. MCPS is the British equivalent of BMI and ASCAP in America. It is responsible for collecting royalty monies for recordings on mechanical media, e.g., gramophone records and cassettes, on behalf of its members from recording companies and radio stations who pay commissions for such services.

MCPS intends to use ADABAS-M to bring down its commission costs (and thus its rates) so as to bring in more clients and revenue.

The decision to acquire DBMS software at MCPS was directed by several requirements: the need for interrelated information, reduction of programming costs via DBMS-provided utilities and interactive query, and integrated recovery. Several commercially available DBMSs were

evaluated for their Digital Equipment Corporation (DEC) PDP-11/70 minicomputer. In addition to 320 KW memory, RP06 disks and 20+ VT05 and VT52-compatible terminals, the site runs DEC's IAS operating system. DEC's own product (DBMS-11) was eliminated on the basis of the poor performance track record of DEC's multi-key file management system (RMS-11K). Also it was felt that DEC was no longer committed to the DBMS-11 product line for the PDP-11 series. Cincom Systems, Inc.'s TOTAL was eliminated primarily because of its limited data structuring facilities. ADABAS-M was further preferred over TOTAL because MCPS felt a strong Software AG commitment to the DEC product line as very important versus Cincom's un-directed commitment to the many minis TOTAL runs on. ADABAS-M would also facilitate any future migration to DEC's VAX super-mini.

The projected MCPS data base will eventually consist of around 20 ADABAS files and approximately 50 COBOL programs. MCPS previously used DEC's RMS-11K software for relative and keyed access to their data and are currently converting their program inventory to ADABAS-M. One of the ADABAS files numbers over 800,000 records with an average record

length of 100 bytes. This file also has seven descriptor or keyed fields and contains data on composers, arrangers and publishers.

One of the descriptors will make use of the ADABAS-M phonetic key facility. Because of ADABAS-M data compression techniques, a saving of 53% in disk storage will be achieved on the above file alone.

ADABAS-M was installed as a test site for the IAS operating system version in January of 1980 and purchased in September that year. Our organization, which has a staff of 150, collected revenues of around 8 million pounds last year and are looking forward to ADABAS-M enabling them to reduce costs and generate more revenues through lower commission rates. Future plans call for the use of the ADASCRIP-T-M query and report facility to provide ad hoc information for management.

Online Data Dictionary System

Software AG

The online Data Dictionary combines most of the functions of the batch Data Dictionary with the flexibility and immediacy of an online system.

The online Data Dictionary System extends the ADABAS Dictionary System by providing facilities not found in the batch Data Dictionary System. Some of these facilities are:

- Support for multiple online users;
- Ease of use; the online Data Dictionary System is a menu driven system;
- Establishes and maintains, automatically, the logical relationship between system, program modules and report record types;

- The online Data Dictionary System provides the same edits the batch Data Dictionary System does, but also provides meaningful error messages and the opportunity for the user to correct the error immediately;
- Ability to display attributes online; the online Data Dictionary System includes functions which allow the user to:
 - Display all data bases on the Dictionary;
 - Display of all files, systems, verifications, relationships, and ownerships for a data base;
 - Display of all programs for a system; all modules and reports for a program;
 - Display of all fields for a file;
- Special DBA functions; the online Data Dictionary System provides special functions designed for use by the Data Base Administrator

(DBA). These functions allow the DBA to control and interrogate user access to the dictionary file via the online Data Dictionary programs.

The functions provided by the batch Data Dictionary System that are not currently supported by the online Data Dictionary System are:

- Response Code data maintenance;
- Pull forward of group elements;
- Rippling of group element deletions.

The online Data Dictionary programs are all written in NATURAL. The system was designed so that each NATURAL module performs one discrete function and then passes control and appropriate parameters to the next module and function.

The expected general availability for the online Data Dictionary System (initially to users of NATURAL only) is March 1981.



ADABAS

Technical Notes

ADABAS/VM — An Overview

The ADABAS/VM interface, the most recent addition to the ADABAS Product Group, enables ADABAS execution in a virtual machine running under the Conversational Monitor System (CMS) or the VS1, MVS, DOS/VS or DOS/VSE operating systems. It allows programs residing in one or more Virtual Machine to access a data base residing in the same or other Virtual Machine.

ADABAS/VM can be installed with little or no modification to existing ADABAS software. In addition, this interface supports existing application programs and all ADABAS-supported TP systems without modification or re-linking. Initialization and termination of ADABAS/VM is independent of ADABAS or the TP system. Furthermore, initialization is dynamic and does not require an initial program load (IPL) of the operating system.

ADABAS/VM supports up to 255 ADABAS data bases within each virtual machine. These data bases can be moved from one operating environment to another with guaranteed compatibility.

OPERATIONAL OVERVIEW

Each Virtual Machine in the ADABAS/VM environment has its own view of the data base network. ADABAS/VM manages all traffic (messages, commands, and responses) to and from any data base within its Virtual

Machine.

The VM 370 Virtual Machine Communication Facility (VMCF) is used to establish the link between the ADABAS/VM software executing in different Virtual Machines. ADABAS/VM monitors and collects performance statistics for tuning purposes on all traffic.

STRUCTURE

ADABAS/VM structure depends upon whether ADABAS/VM is running under one of the supported operating systems or under CMS.

When running under a supported operating system, ADABAS/VM executes as a separate operating system task in its own region/partition/address space. The task will consist of the following components:

- The Dispatcher which controls the flow of all messages, ADABAS commands and responses, and operating system dependent services;
- An Interrupt Handler which examines all external interrupts received by the Virtual Machine and enqueues them for processing by the Interrupt Processor;
- An Interrupt Processor which examines and processes interrupts built by the Interrupt Handler. A message header supplied by the VMCF identifies the interrupt as requiring a response or not, as well as the identity of the sending Virtual Machine and the length of the message;
- Optional Pseudo-users

which are created by the Interrupt Processor and appear as an ADABAS user to the target data base. This feature provides great flexibility to ADABAS/VM; because of it, no modification to the ADABAS SVC or user code is required;

- Optional Pseudo-MPMs which create an interface between an ADABAS user and the ADABAS/VM software. Because the Pseudo-MPM emulates the ADABAS MPM, no modifications need to be made to the existing MPM;
- Optional Internal Command Elements which are created by the Pseudo-MPM to control VMCF Processing associated with an ADABAS command.

In a CMS Virtual Machine, ADABAS/VM can be used as either a Source Machine or a Sink Machine.

An ADABAS user executes in a Source Machine. The ADABAS/VM interface consists of a single interface module AVMLCMSX (AVMLCMS3 or AVMLCMS4 depending upon the ADABAS version being used). AVMLCMSX is entered each time a user calls ADABAS. AVMLCMSX is similar in function to the existing ADABAS interface modules provided for the various supported environments.

The ADABAS data base and its MPM execute in a Sink Machine which receives commands via the VM/370 VMCF interface.

The Sink Machine consists of an ADABAS nucleus for process-

(Continued)

ing commands, a set of MPM-like routines to interface between the ADABAS nucleus and VMCP and a set of data base maintenance utilities.

ADABAS/VM capabilities and ease of installation and operating make it a powerful addition to ADABAS for VM sites.

Frequently Encountered Problems—V4.1.1

Software AG

To date, we know of over 100 sites using ADABAS Version 4.1 in test or production mode. Many users are experiencing similar difficulties in implementing Version 4.1; some of these frequently encountered problems are described below. Please note that the majority of the information discussed below is also covered in greater detail in the Version 4 manuals. We strongly recommend that any installations planning to implement ADABAS Version 4 carefully review the Version 4 manuals. Also, the Version 4 Release Notes and Early Warnings distributed with the release tape should be reviewed thoroughly to avoid unnecessary problems. Finally, many users have contributed valuable information regarding their experiences with Version 4 and these articles have been published in past Newsletters and in the Proceedings for the Users' Conference held in Houston in 1980.

SVC Installation

The ADABAS SVC must be installed prior to the running of the

installation verification procedures (i.e., phase 1-9).

If a type 3 or type 4 SVC is being used, it must be made page fixed and resident.

The correct SVC number must either be specified on the ADARUN SVC parameter or ADARUN must be zapped with the correct SVC number. The link routines also require the correct SVC number (i.e., ADALNC, ADALNI, ADALTM and ADALIMS). The SVC number in all TP link routines must be set by updating the source before assembly (e.g., SVCNR EQU 249). After a link routine is assembled and linked, you may change the SVC number with a zap (i.e., rep 0084 0AXX—where XX is the desired SVC number in HEX).

If V3.2.1 programs linked with ADAHOOK are going to be run against a Version 4 data base in multi-user mode, then the batch link routine 'ADALNK' should have the SVC number zapped (i.e., rep 0084 0AXX—where XX is the desired SVC number in HEX).

Some of the ADABAS utilities require the SVC regardless of the mode parameter. These utilities issue an SVC call to determine whether the nucleus is active or not in order to identify the method of creating a checkpoint record.

If the SVC is not installed, is installed improperly, or the wrong SVC number is used, the results are unpredictable. If the SVC number specified is not an ADABAS SVC, the operating system may be brought down by the application.

ADABAS System Files

It is recommended that at least the Security, Checkpoint

and Data Dictionary files be loaded at installation time. It is mandatory that the Checkpoint file be loaded. If this file is not loaded, the results are unpredictable.

The Security and Checkpoint files are not to be modified by the user. ADABAS accesses these files through special internal routines and not through the standard file access routines. If these files are modified, the results are unpredictable.

Periodic maintenance to purge old checkpoint records must be performed on the Checkpoint file by a user written program; otherwise, the Checkpoint file will continue to grow larger and larger until it runs out of extents. We also recommend the use of the REUSEISN option for the Checkpoint file to prevent the Address Converter from running out of extents. Note that the Checkpoint file also contains user ET data records. When you write your purge program, please take this fact into consideration (i.e., don't delete the user ET data records unless you desire them to be deleted).

Machine Addressing

Version 4 utilizes BDAM machine addressing as the default. This means that when an existing V3.2.1 data base is converted to V4, the 'OPTCD' in the DSCB for all ADABAS BDAM files will be set to X'08' instead of X'00' (i.e., for relative track addressing). In order to convert from a V4 data base to a V3.2.1 data base, it is required to apply the optional V3.2.1 machine addressing zaps or to zap the DSCB OPTCD for all ADABAS BDAM files back to X'00'. If this is not done, the data base will be unusable.

(Continued)

User Queue Elements

A User Queue Element (UQE) is created when an application issues its first command and each UQE is identified by a 4-byte internal userid. For TP transactions, the internal userid is the terminal ID. For batch applications the starting 'store clock' value is used to create a unique internal userid. (For 360 users the 'time of day' value is used.)

If the first command issued to ADABAS is an OPEN command and contains an 8-byte external userid, ADABAS will store this external userid in the UQE (in addition to the internal userid).

The UQE is deleted when an application issues a CLOSE command. The UQE is also deleted if the user is timed out (see below for more detail on timeout conditions) as follows:

- Access only users are deleted at timeout;
- ET logic users at ET status are deleted at timeout.

The UQE is not deleted for the following user timeouts:

- EXU users;
- Cluster users;
- ET logic users not at ET status;
- Users that issue an OPEN command with an external userid.

The 'STOPU' operator command can be used to delete UQEs.

The 'NU' ADARUN nucleus startup parameter specifies the size of the UQE area. If the UQE area becomes filled up, all new users will receive a response code 72 until some old user issues a CLOSE command or some UQEs are deleted with the STOPU command.

OPEN commands are required with Version 4 only if ET data is

to be written or the user is to be an access only, EXU or cluster user. If an OPEN is not done, the user will become an ET logic user. (However, ADAMINT still requires the SIGNON to be done when running under CICS because the SIGNON performs some initial housekeeping. A loop results if the SIGNON is omitted.)

The following is a summary of what happens when a timeout condition is detected by ADABAS (i.e., transaction timeout, non-activity timeout) or AUTOBACKOUT/AUTO-RESTART occurs.

Timeout Conditions

Transaction Time Limit—

Applies only to ET Logic users.

The time 'window' begins when the first command which results in a record being placed in hold status is issued, and ends when an ET, BT, or CL command is issued. (The CLOSE command issues an internal ET command.)

Non-Activity Time Limit—

Applies to all users.

Different non-activity time limits may be defined for each user type. The time 'window' begins when the last command was issued to ADABAS and ends when the next command is issued by the application.

AUTOBACKOUT/AUTO-RESTART— system failure.

Timeout Action By User Type

Access only— The UQE is deleted.

EXU — The file list is purged from the UQE and the user type is changed to an access only user (i.e., exclusive use of the file is lost). The

user will receive a response code 22 on any hold-type, add, delete, or update commands and all other commands will receive a response code of 0. The UQE is not deleted.

Cluster

- The file list is purged from the UQE. ADABAS suspends all updating of the cluster and initiates an AUTO-CLUSTER-BACKOUT for the cluster. All users of this cluster will receive a response code 9 on the next command issued. ADABAS will advise the operator of the names of the jobs which are to be cancelled. All held records are released. The UQE is not deleted.

ET Logic

- The file list is purged, the incomplete transaction is backed out, and the user returned a response code 9 on the next command issued. All held records are released. If the user is at ET status, the UQE will be deleted.

If the timeout occurred due to an abend or system failure, TP users will be the only users that will receive the response code 9. All batch users that are restarted

(Continued)

are considered new users and a new UQE will be created (see UQE for batch programs above).

If a TP user is an ET logic user and is restarted after an abend and before the timeout limit is exceeded, ADABAS is unable to detect this situation and will treat all updates since the last ET as belonging to one logical transaction (i.e., updates before and after the abend issued from that terminal will all be considered part of the same logical transaction). Therefore, we recommend that all TP programs start by issuing either a BT or OPEN command. This will ensure the logical integrity of the data base. The BT command will backout any incomplete transaction. If there is nothing to backout, the BT will do nothing. The OPEN command will cause an internal BT command to be issued.

In general, it is a good idea to always issue an OPEN command, specifying which files are going to be updated. Attempts to run EXU, cluster user or any utilities against these files will be caught immediately, rather than when well into the program processing.

If a batch user program had issued an OPEN with an external userid and is restarted after an abend, this user will receive a response code 48. Since batch programs are considered new users, ADABAS will detect that the external userid is not unique since this external userid still exists in the previous UQE that was created for this same batch user. In this situation you must delete the UQE with the 'STOPU' operator command, then restart the job.

The use of an external userid in TP will cause no problems as long as the user is restarting the

transaction from the same terminal (i.e., same internal userid).

Utility Error 10

Version 4 utilities can abnormally terminate with an error 10 if the utility is run in parallel with the Nucleus and conflicting file usage is detected. This can result if an application program has the file open for update or access (see the *ADABAS V4 Operations Manual*, page 115-116) which is indicated by the presence of a UQE containing the file in the file list. As noted previously, a UQE could exist for an application which has finished running but has not successfully issued a CLOSE command and has also not exceeded the Non-Activity Time Limit.

A file-usage conflict can also result if the 'DIB' block has not been reset. The DIB block is set by most Version 4 utilities and is reset upon successful completion of the utility. If the second step of a two-step utility (i.e., ADALD2, HD2, CP2) run doesn't complete successfully, the DIB remains set and any utility that tries to access the file will get an error 10. The DIB block may be reset via the ADAVUS utility function 'RESET RABN = DIB, JOBNAME = XXXXXXXX'

NOTE: When running the ADAVUS utility function 'RESET RABN = DIB', it is always recommended that the JOBNAME parameter be used for the following reasons: If other utilities are running in parallel, 'RESET RABN = DIB' clears all entries (the entire DIB block is set to binary zeros). This can be disastrous if conflicting utilities happen to be starting up at the same time.

Error 10 can also result if MODE = MULTI is not specified

for the ADAVUS 'NEWWORK' function or if MODE = MULTI is specified and a Nucleus is active against the same data base (see Early Warning 34). If this situation should arise, the Work data set should be formatted again, the ADAVUS utility function 'RESET RABN = DIB, JOBNAME = XXXXXXXX' run, and the ADAVUS 'NEWWORK' function rerun as per Early Warning 34.

Error 10 may also result for certain utilities not running in parallel and a pending AUTORESTART exists (i.e., ADAVUS, ADAFIX). In this situation run a Single User Mode ADASCRIP+ that does an OPEN and a CLOSE command, then restart the utility.

Version 4 Interregion Communication

It is recommended that Version 4 application programs be linked with ADAUSER. At execution time, ADAUSER will dynamically load ADARUN. If MODE = SINGLE is specified, ADARUN will load ADAEMO, ADADAK, etc. (i.e., the ADABAS Nucleus modules). If MODE = MULTI is specified, ADARUN will load ADALNK which will in turn issue ADABAS SVC calls to the ADABAS Nucleus.

Version 3.2.1 programs which have been linked with ADAHOOK to run in Multi-User Mode can also be run against a V4 data base. Update programs must be modified to use ET/BT, EXU, or CLU logic. The JOBLIB/STEPLIB for the program must also be changed to refer to the Version 4 load library. ADAHOOK will load a program 'ADALINK' (which is the alias for the Version 4 ADALNK module). ADALNK

(Continued)

should have been zapped with the Version 4 SVC number at installation time (i.e., REP 0084 0AXX).

The Version 4 data base ID is set dynamically by ADAHOOK via the '//AENQx' DDcard. However, for Version 4, only the first character of the ENQ-ID is used. For example, if '//AENQPROD' is entered, only the 'P' is used and is translated from EBCDIC to its decimal equivalent (e.g., P=215). In this example, 'P' would represent a Version 4 data base ID = 215.

When converting from a V3 data base to V4, it is recommended that V4 data base IDs be selected which can be readily translated to displayable EBCDIC characters. This will facilitate using the '//AENQx' DDcard to set V4 data base ID into ADALNK and will make conversion and testing simpler.

V3.2.1 programs which have been linked with ADAHOOK that are to be run in single user mode can also be run against a V4 data base. In this case, the JOBLIB/STEPLIB for the program should be changed to point to the Version 4 library. ADAHOOK will load a program 'ADANUC' which will load the program 'ADARUN' which will load the ADABAS nucleus modules. Two additional DDstatements must also be specified:

```
//DDPRINT DD SYSOUT = A
//DDCARD DD *
```

ADARUN control cards (see the ADABAS Version 4 Operations Manual)

User ABEND Codes

ABEND U020	Error in ADABAS Nucleus initialization. This will be accompanied by a
---------------	---

more descriptive error code which can be found at the end of the SYSOUT list of the ADABAS startup messages. (See the *ADABAS Messages and Codes Manual*.)

ABEND
U021

A recursive AUTORESTART condition exists. The corrective measure is to reformat the WORK file, restore the data base from the last fix tape, then run REGENERATE. This condition should be reported to Software AG.

ABEND
U034

Error in a utility. This will be accompanied by a more descriptive error code which can be found in the output listing of the utility control cards. (See the *ADABAS Messages and Codes Manual*.)

ABEND
U999

MPM STAE exit (trap for system ABENDs). This ABEND is accompanied with a display of all the nucleus modules 'entry-points', PSW and registers at the time of the ABEND condition. This condition should

be reported to Software AG.

ADARUN User ABEND Codes (ABEND U001/U002/U003/U005/U006/U007)

When ADARUN detects an error, it does not display an error message. Instead, ADARUN returns a user ABEND code indicating what the error was. Refer to Early Warning 31 and the ADABAS Messages and Codes Manual for a detailed explanation of these user ABEND codes.

Device-Type Specification

When running utilities, ADARUN/utility parameters are available for specifying the device types for various ADABAS data sets (i.e., ASSODEV, DATADEV, WORKDEV, SORTDEV, TEMPDEV, etc.). These parameters should be explicitly coded for all utility runs, regardless of device type.

Version 4 Sequential Command Table

The Sequential Command Table is an in-core table used by Version 4 to prevent the 'record migration' problem which existed when sequential processing and updating were performed concurrently by one program under V3. The size of this table is specified via the 'LQ' parameter of ADARUN. A 60-byte entry is created in this table for each Command ID (CID) associated with a sequential processing command and this entry is deleted when end-of-file is reached. The CID is deleted via the 'RC' command, or a CLOSE is issued by the application. This table is shared by all applications running concurrently against the

(Continued)

ADABAS nucleus.

This table must be large enough to hold all sequential CID entries, and applications doing sequential processing must be examined to ensure that they are issuing 'RC' commands and not leaving unnecessary entries in the Sequential Command Table. If this table fills up, a response code 70 will be returned to the next command requiring the creation of a new entry.

For applications using hierarchical ADAMINT LOKATE-SEQREAD logic, the following approach is recommended.

LOKATE-SEQREAD to the first level file and save a KEY/ISN of the record being processed. Continue with the LOKATE-SEQREADs to the files at levels two and beyond. When the hierarchy has been traversed, call the RELCID entry point. Return to the first level file and reissue a LOKATE to reposition for the next SEQREAD specifying the KEY/ISN saved from the previous LOKATE-SEQREAD sequence. This approach will ensure that position is maintained for the sequential sweep and will use a minimal number of entries in the Sequential Command Table.

Version 4 Logging

There are several ADARUN logging parameters. 'LOGGING=' specifies whether logging will be permitted during a particular nucleus session and the remaining parameters specify what should be logged. (Please refer to the *ADABAS V4 Operations Manual* for a detailed explanation of these parameters.) Note that if 'LOGGING=NO' is specified at startup, this cannot be dynamically overridden later

during the session as can be done for the other logging parameters. If logging is to be initially turned off and then turned on at some later point in the session, specify 'LOGGING=YES' on the ADARUN startup parameter and issue the 'NOLOG' operator command when ADABAS is active. When logging is desired, enter the operator command 'LOG'.

PRILog and User Exits

A sample 'PRILog' and user exit 'ADAEX4' are distributed with Version 4 on 'ADABAS.V411.ADASRCE'. These are samples only and are meant to serve as starting points for user-developed routines. NOTE: These sample routines are *not* supported by Software AG.

Version 4 Data Dictionary

Changes have been made to some of the Data Dictionary modules for Version 4. These are outlined in the Version 4 Release Notes and should be carefully reviewed prior to installing Version 4. The new Data Dictionary libraries are distributed on the Version 4 installation tape as data sets 8 and 9.

NATURAL and ADABAS Version 4

Existing NATURAL programs which are to perform updating against a Version 4 data base must be modified. They must have a 'SET GLOBALS TR=ON' statement and an 'END OF TRANSACTION' statement. If the statement 'SET GLOBALS TR=ON' is not specified, you will receive response code 144 (record not held) for all UPDATES, DELETES and ADDs

will not receive a response code 144 because ADABAS will put these records in hold automatically. If the statement 'END OF TRANSACTION' is not specified, all UPDATES, DELETES, and ADDs will be backed out if the user is timed out before a CLOSE command is issued by NATURAL.

Any release of NATURAL before Zap Level 7 will ignore response code 9 from ADABAS. With Zap Level 7 and any subsequent releases of NATURAL, the user will be notified of all backout conditions via the NATURAL error message 'NAT3009'.

This information will be distributed with each Version 4 system.

ADABAS V4.1 Experiences

Larry Watson
Hewlett-Packard

Hewlett-Packard is currently running ADABAS Version 4.1 "in production". Eager to use the improved facilities of Version 4.1, we set the following plan for conversion:

1. Place all test files under Version 4.1;
2. Once the test environment becomes stable, place all test files in production;
3. Convert "batch" applications by changing JCL as appropriate and reassembling all update ADAMINTs with "EXU=";
4. Convert NATURAL programs by incorporating ET syntax into online update programs.

(Continued)

During our testing and initial production "phases", we encountered various difficulties. These included product "bugs" and incompatibilities as well as user "misunderstandings". Since many of the "product problems" have been resolved by later versions (SM1), I'd like to focus on those areas where your prior awareness can minimize difficulties in implementing Version 4.1:

- MPM parameters are complex; we found them difficult to adjust; however, they are critical. When they were misadjusted, we received everything from spurious response codes to abends of the MPM. Typically these indicated insufficient buffer sizes.

The trial and error aspect of determining suitable MPM parameters, albeit necessary, is time-consuming and somewhat frustrating. I feel that more thorough documentation is needed in this area;

- We had difficulties with Sequential Command Queue overflows, User Queue Element build-ups, and batch update programs that did not issue closes under Version 3.2.1. A description of these and other Version 4 problems is now distributed with the system. (This information is also found in "Frequently Encountered Problems — V4.1.1," an article in this Newsletter.);
- The online Data Dictionary (Beta Version using NATURAL), designed for Version 3.2.1, does not operate under Version 4.1. The Version 4.1 compatible system is nearing completion;
- We received a response code 21 when trying to load a DDM which is more than 3 reads large. Software AG indicates that the size of the read is not causing the response code; however, the problem is still unresolved.

Like other ADABAS users who are operating under V4.1, your best approach to conversion is to plan carefully and test thoroughly. By reviewing all documentation carefully, you can eliminate some problems; by slow, thorough testing you can more easily isolate and define problems as they arise.

In spite of the problems mentioned above, we consider our move to V4.1 a wise decision. We feel it places us in a future-oriented position and we no longer need to worry about how a new or changed application will work under V4.1.

We also feel that technical support for Version 4.1 is better than for Version 3.2.1, probably because SAG is dealing with an intrinsically cleaner piece of code. Version 4 is the direction of the future.



ADABAS Product Rep's Report

Product Representative's Report

*Jennifer Lee
Aerospace Corporation*

Approximately 350 Change/Enhancement requests have been received for all Software AG products. Peter MacRoberts deserves a great deal of credit and appreciation for his effort in compiling and classifying the many requests. The next step was for the Product Reps to consolidate the previous years' unimplemented requests with the current requests and to consolidate requests whenever appropriate.

The C/E requests are presented to Software AG without any prior year comments to solicit Software AG's most current response. All requests that were indicated last year as "will implement" items have been reviewed for their inclusion in the latest release. Only those requests that remain unimplemented appear in the C/E list. Software AG is being encouraged to respond with the projected release level on "will implement" items.

All users should be careful to cast their ballots for Change/Enhancements so that they accurately reflect user concerns. Splintering your votes too finely may not prove effective since any request that receives 10 votes or fewer will be dropped

from the next year's C/E requests. Group voting by region or international area has de facto acceptance and votes will be carefully counted to ensure that the sum total balances with the number of participating organizations.

Voting is a measure of user concern and it is important that every user exercise their influence in the voting process to demonstrate the need for positive action in the pertinent product areas. The reporting capability now available with the C/E request system will permit the users for the first time to determine the effectiveness of the Change/Enhancement process. I strongly urge you to participate in this primary user activity.

ADABAS Special Interest Group Chairpersons

DATA DICTIONARY

Ms. Virginia Sullivan
INCO, Ltd.
1 New York Plaza
New York, New York 10004
(212) 742-4165

DBA/TOOLS

Ms. Julie Cottrell
National Education Association
1201 16th Street, N.W.
Washington, D.C. 20036
(202) 833-4371

DOS

Mr. Bernie Conner
Ray-O-Vac
101 East Washington
Madison, Wisconsin 53703
(608) 252-7557

PUBLIC SECTOR

Mr. William Belisle
State of California
Department of Youth Authority
4241 Williamsborough Drive
Sacramento, California 95823
(916) 322-2535

EDUCATION/ DOCUMENTATION

Mr. John Kaye
Foremost Insurance Company
5800 Foremost Drive, S.E.
Grand Rapids, Michigan 49501
(616) 942-3364

PERFORMANCE

Mr. Bob Becker
Foremost Insurance Company
5800 Foremost Drive, S.E.
Grand Rapids, Michigan 49501
(616) 942-3364

RESTART/RECOVERY

Mr. Bob Taylor
Evans Products Company,
Homes Group
1777 Walton Road
Dublin Hall, Suite 200
Blue Bell, Pennsylvania 19422
(215) 628-4800

RETRIEVALS/INTERFACES

Mr. Jack Reitz
Evans Products Company
Railcar Division
2550 Golf Road
Rolling Meadows, Illinois 60008
(312) 640-7000

COM-LETE News



COM-LETE Technical Notes

Security Under COM-LETE

S. M. Isaacs

Massachusetts General Hospital

In planning for security at Massachusetts General Hospital we implemented an open-ended security design suitable for changing organizational needs, which would be responsive to the varying needs for security integrity or privacy of user departments. The system permits the computer center to assure users that they may enter data in the computer while retaining some control over the access of that data by others. In terms of privacy, data may be protected from being read for display, used in computations, or accumulated for statistical purposes. In terms of data integrity, others may be permitted to read the data for any reason but prevented from changing, updating or erasing the data.

BACKGROUND AND SCOPE

Currently, data at Massachusetts General Hospital computer center may be accessed in five ways:

1. Traditional batch program with input and output in the computer room;
2. Remote batch initiated and with output possible through terminals;
3. Present MG On-Line system as developed here in the 60's;
4. COM-LETE transaction system which is replacing method 3 above;
5. COM-LETE interactive development system for programming.

The data security system will not affect or add security to method 1, the traditional batch environment. Administrative controls provide some protection as few strangers have access to the computer room.

The data security system does not apply to the present MG On-Line. As quickly as possible, applications are being moved from

this environment as it is to be phased out.

The data security proposed here applies to the remaining methods, remote batch, transaction processing, and the development facilities.

Data which is classified as 'public' (described under access and restriction levels subsequently) is available to anyone who logs on to the COM-LETE system. This is probably a member of the MGH community with a valid userid and password. We recognize that there are, however, some non-MGH users with valid passwords and the possibility that a user may have compromised his password; so we clearly advise our users that absolute security should not be expected by anyone. The system permits various bars and levels to maintain flexibility and maximize the probability that only authorized users may get to data.

SECURITY ELEMENTS

The security mechanism uses the following elements:

(Continued)

- Userid to identify the user of a terminal;
- Password by which the user validates his right to access the system;
- Access class categorizing the data classes a user may get to;
- Access level defining the read-write rights within a user class;
- Restriction class defining what user classes may get to the data;
- Restriction level defining the read-write minimums by class.

USERID AND PASSWORD

Each MGH user is assigned a unique personal userid which consists of the first six or fewer characters of last name. Departments with many users requiring intermittent access to the data base and for whom individual userids are not practical are assigned generic userids for department use. (Generic userids have limited data access.)

Passwords are required at log on time to access any of the system's facilities. They may and should be changed by the user as a precaution against improper data access. Periodically, the security officer checks for improper passwords such as first or last names.

Journals are maintained showing log on, log off, and resource usage. A journal of security violations of both the system log on and the data security mechanisms is maintained for administrative use.

ACCESS AND RESTRICTION CLASSES

The access and restriction classes are assigned in such a way as to permit or deny access

based upon the present application identification scheme as well as organizational units. Superset classes may be also defined permitting multi-application access and institution-wide financial or medical data classifications.

The assignment of access classes to users and of restriction classes to data is an administrative function which is performed by the Data Base Administrator. The structure is itself relatively inflexible because of the high overhead of trapping all computer input-output to pass through a security mechanism.

The flexibility of the mechanism is obtained through assignment of class codes; each user may be assigned up to four access classes.

ACCESS AND RESTRICTION LEVELS

Once the user and data "class codes" match, levels are checked. Both user and data levels are defined by a four value code (0-to-3) which identifies data as public; "first degree," "second degree" or "third degree."

IMPLEMENTATION

On the user side, there are four access class codes, each with a related access level code. Blanks indicate unused class codes. The class codes are three alphanumeric characters.

These fields are protected from display through normal user-available commands.

A table of userids is maintained, indexed by userid and linked to fields containing the MGH ID-number, department and division or budget coding, etcetera. A first step would be from userid to MGH ID-number with the reasonable expectation

that organization data, full name, telephone and address will all be available from the application personnel file or from the MGH ID file.

On the data side, there are six restriction class codes, each with its own related restriction level. Data security restrictions require at least two tables, one for use in RJE and handling other non-COM-LETE files, and one table for COM-LETE files, including ADABAS files. The extension to handle fields within ADABAS files is deferred until it seems manageable to us.

The security screening logic traps all file openings, explicit or implicit, or related equivalent first I/O to members of files through use of the security exits. COM-LETE has provided for both production and utility programs.

At the exit, a common in-core security routine is invoked. This routine gets the user access class and limits and also references the MGH defined data tables with restriction class and levels. The routine is resident in COM-LETE. It contains tables for userids, ADABAS file numbers, DSnames, program names, and DDnames.

Each entry in each table has space for appropriate sets of access codes and level numbers. These tables are now filled with every known userid, file number, DSname, program name, and DDname. They may or may not yet have access codes and level numbers assigned to their entries.

Each access code has a pair of level numbers.

The verification procedure is as follows:

1. The ADABAS file number (or DSname, program

(Continued)

name, or DDname) is checked for the presence of one or more access codes;

2. If there are none present, allow the requested function;
3. If there are any present, check them against the access code(s) associated with the userid;
4. If a match is not found, disallow the function;
5. If a match is found, check the read or write level numbers as appropriate for the function requested. The level number of the userid must be higher than or equal to that on the protected file (or . . .) or the function will be disallowed. (Level checking does not occur for program names.)

The implementation of the data security mechanism is such that the security programmer cannot violate it without some key knowledge of codes which hopefully is known only to the security officer.

To support application level extensions to the security or privacy facility, the application should be able to call a security routine, provide a list of required user classes and levels, and be prepared for standard systems action for a security violation. The routine would check the actual USER = ID Block but this data is not to be available to standard programs.

Restriction on individual records or restrictions based on terminal location must be implemented within application level programming; it is not provided by this system facility which acts only at the file level.

This security system has met our needs and so far has been manageable.

COM-LETE At Franchise Tax Board

*Bill Bruening
Franchise Tax Board*

Franchise Tax Board runs COM-LETE on a 4 megabyte Amdahl V5 using the MVS operating system. Like most installations we are concerned with performance of our teleprocessing system and have devoted a considerable amount of time and effort to evaluating its performance with an eye to increasing throughput. Here are some of the results of our findings:

We have monitored our applications using a typical transaction which reads 20 ISAM records, writes to a capture tape, and executes four excps excluding terminal I/O. When executing an application on one terminal, the elapse time is quite adequate. However, upon repeatedly entering applications on 25 terminals, the response time degrades noticeably.

A trace of the applications was taken to obtain elapse time data. Eight terminals were run in six threads with repeat keys held down so that work was always being entered into COM-LETE. An analysis of the elapsed time in the applications indicated that, excluding file and terminal I/O, only 25 milliseconds per transaction were spent in the applications whereas an average of 200 milliseconds per transaction were spent "outside" any of the applications. We also noted that COM-LETE switched execution of applications in as little as eight milliseconds at file or terminal I/O.

We have considered some possible approaches to improving applications processing, including adding more threads to COM-LETE, executing multiple COM-LETEs and executing COM-LETE "stand-alone". None of these methods seems to answer our applications processing needs in the context of our daily operating requirements.

I have concluded that throughput would be increased if COM-LETE utilized more tasks for terminal I/O. This is a proven technique to improve throughput of teleprocessing systems and would allow for growth of additional terminals or threads. By servicing more terminals each time it is dispatched, COM-LETE would utilize elapsed time not spent in threads. I think that allowing for one terminal I/O task per thread or one task per local I/O controller and remote drop line would be an outstanding enhancement to the COM-LETE system.

I suggest that Software AG establish a performance group whose task is to improve throughput of COM-LETE; I believe this should be given the highest priority as COM-LETE performance can affect the performance of other Software AG products.

Experiences with COM-LETE and VSAM

*Ken Torri
R.J. Hansen Associates*

Recently, I developed an on-line Work Scheduling System for the Sacramento County Public Works Department. The system

(Continued)

was implemented on an IBM 4341 OS/VS1 system, utilizing Software AG's COM-LETE telecommunications monitor and VSAM as the data access method.

The use of VSAM with COM-LETE has been an interesting experience, and one I thought I would share with you.

First, a few notes about VSAM for non-VSAM users. VSAM is an IBM access method that provides three file types, secondary indexing capabilities, and a useful utility referred to as Access Method Services (better known as IDCAMS).

The three file types, or clusters in VSAM terms, are:

- KSDs — Key Sequential Data sets
- ESDs — Entry Sequenced Data sets
- RRDs — Relative Record Data sets

The Key Sequential Data sets are indexed data sets whose key is some variable or combination of variables in a file. The file is composed of two parts (actually two subfiles):

- Index Component
- Data Component

The index component contains the keys and a pointer to its associated record. This pointer is often referred to as an RBA or Relative Byte Address. The data component contains the data record associated with the key. The primary key must be unique within the file; however, any secondary keys, or alternate indices in VSAM terms, can have duplicates. KSDs can be addressed sequentially, randomly, or skip sequentially.

The Entry Sequenced Data sets are sequential files that can be addressed randomly through a Relative Byte Address (RBA)

from the beginning of the file.

The key for a Relative Record Data set is the Relative Record Number—much like an ISN for you ADABAS people. In other words, if you want the third record in the file, the key would be 3.

All VSAM files are divided up into logical units referred to as control intervals and control areas. A control interval consists of what could be considered a blocked record. That is, there are multiple records per control interval. A control area is nothing more than a block of control intervals. Records can be variable or fixed lengths and may span control intervals.

VSAM has a much more efficient record storage concept for its KSDs than other indexed access methods. It automatically builds multiple levels for its index components, and also handles new insertions, which cause other records to be shifted in an efficient manner. The buffering services and sharing of control blocks are of great use in a multi-region/task environment.

One of the key capabilities of VSAM is its record sharing capabilities. These capabilities must be taken into consideration when using "VSAM" in an online environment. These capabilities and restrictions are discussed below.

From the application standpoint, the use of VSAM with COM-LETE is a very easy task. The application programmer writes standard OS/VS COBOL VSAM I/O statements to access any VSAM files. That's right—it's just like writing a batch program using COBOL. The only special processing the application programmer must concern himself with is the checking of VSAM

status codes and writing the COM-LETE screen I/O calls. It is **very** important that the file status code be checked after every VSAM I/O. For instance, if you do not check the VSAM status code after an open statement and a bad open was encountered, you will get a very nice 'OC4' completion code when you try to reference the I/O area. You get that 'OC4' because addressability to the I/O area is not established until a valid open has been accomplished.

VSAM has the capabilities of providing data integrity at the control interval level. In other words, VSAM will reserve a control interval for one user and not let any other users access this control interval until the first user releases this control interval. The types of integrity that VSAM provides are based on the share options specified at the time the clusters are defined. If a record is currently held by a user, a '93' status code is returned and the I/O call should be reissued. VSAM does not queue I/O calls that cannot be serviced. In other words, if a record is held by another task, then VSAM will tell you about it, but it will not wait for that record to be freed before returning to you.

Normally, the procedure would be to issue a COM-LETE roll-out function (i.e., wait) and then re-issue the call. If after several tries (I usually try 10 times) the record is still not free, return a message to the screen indicating that the record is currently held and to try later.

The status codes in the table below are those that are most often returned. The column to the right indicates what may be causing the problem.

Status Code**Reason**

- 90 Bad Open: File not closed properly by a previous user. Either run an IDCAMS VERIFY or do a COM-DELETE *ULIB BTCH and ONLN command.
- 93 Record or resource not available. Either another user has got the record or the file is being used by another task in another region. On an Open there is not enough virtual storage in COM-DELETE's region.
- 95 Logic error in I/O call. Usually your FD in COBOL does not match that defined in the VSAM cluster definition.
- 96 DDstatement in COM-DELETE initialization procedure is missing. Contact your COM-DELETE systems programmer.

The *COBOL Programmer's Guide* provides a more detailed description of VSAM status codes and should be consulted. The user must be aware that COBOL Status Codes have different meanings based on the request type, i.e., Open, Read, Write, etc. The table in the *COBOL Programmer's Guide* also indicates the associated VSAM error codes. These codes are defined in the *VSAM Programmer's Guide* and should be consulted.

There are several considerations in cataloging a VSAM 'DDN' to COM-DELETE. These considerations include:

- Locally shared vs. non-shared resources;
- COBOL use of VSAM string processing;

- Defining buffer areas of VSAM files;
- Defining share options in the VSAM cluster definition.

The following explains the record sharing and integrity options available under VSAM. These sharing options are defined in the SHARE OPTIONS parameter when defining a cluster. Although VSAM allows sharing options for both across-region and across-system, only across-region options are discussed. COM-DELETE users in different threads fall into the across-region category.

The share option 1,3 provides COM-DELETE read/write data integrity. In other words, multiple users can access the data set for read processing while another user has it for update processing. Under this option a "read" user cannot process a control interval that is currently being updated, i.e., read integrity. No two users can access the same control interval for update, thus providing write integrity; however, they may simultaneously update data in different control intervals. The share option 2,3 is very similar to the 1,3 option, but it does not provide read integrity. In other words, it is possible for a read user to access the same control interval that another user is updating, thus losing read integrity. There is no problem in updating with this option because VSAM ensures that only one user has a control interval for update.

It should be noted that if you specify options 2,3 but access the data in a shared resources environment (i.e., COM-DELETE DDN cataloged with LSR instead of NSR), VSAM forces share option 1,3. So if you want a 2,3 option, you must catalog the DDN

with NSR, non-shared resources.

I recommend VSAM option 2,3 if you wish to read a file in a batch program while the file is still online.

There are two other options: 3,3 and 4,3. Neither one of these provide read or write integrity. It is up to the user to provide this integrity which requires assembler level coding. I would not recommend using these options unless it is absolutely necessary.

It should also be noted that a control interval is not released from an update held status until the update user accesses another control interval, issues a ENDREQ assembler macro, or closes the file. This can lead to problems where a record is read for update and sent to the screen. The control interval will be tied up until the user finally responds, unless the program closes the file before returning to the screen. The user has two choices. One is opening and closing a file on every scheduling of an update program (never do this on a read-only program). The second is to allow the records in a control interval to be tied up until the user responds. I chose to use the second approach and inform the user that the records were held and he should try later. This worked well in our environment, but may not work well in others. The choice is yours.

Another important factor in the performance of VSAM processing is the use of buffers. VSAM will automatically allocate one index buffer and two data component buffers for each string of a KSD. A string is a unique access path to a portion of a file. Each string requires VSAM to maintain the position in the file. The amount of space required for a buffer is determined by the con-

(Continued)

control interval size. So the larger the control interval size, the larger the buffer space. This also means that you have more records available. If the user is using sequential processing, then increased performance can be obtained by using more data component buffers. If a user is using direct processing, then increased performance can be obtained by using more index buffers. You need more index buffers when using direct processing because VSAM accesses the index component for every request and will refresh the data component on every request. Thus, the more index control intervals you can keep in memory results in fewer I/Os to access records. Under sequential processing VSAM will not access the whole index on every request but will use the horizontal pointers in the sequence set (the lowest level of an index) to access the next record. VSAM can use additional data buffers to do read ahead functions and thus have data available for future requests. Sequential processing does not refresh its data component buffers on every request. A complete discussion on optimizing performance and VSAM buffering techniques is contained in the *VSAM Programmer's Guide* and should be consulted. This information is extremely important when you define the size and number of buffers to allocate when using the LSR (Locally Shared Resources) option.

One of the problems I encountered in using COBOL, COMPLETE, and VSAM is the consistency and meanings of terminologies used in the different manuals involved. The following information should aid you in

resolving this problem.

- COBOL does not support skip sequential processing, so there is no need to specify the SKP option when cataloging a DDN to COMPLETE;
- The way COBOL decides if you are going to update, or just read a file, is based on how a file is opened, i.e., I/O is update, INPUT is read-only, OUTPUT is update;
- COBOL does not support addressed processing but does support key processing;
- The table shown relates some COBOL terms to VSAM terms:

- COBOL does not support multiple string processing. However, there essentially can be multiple strings to a data set because of the concurrent running of programs accessing the same file from different threads;
- There is one RPL (Request Parameter List) for sequential processing and a second RPL for direct processing;
- The definitions of the option codes you specify when cataloging a VSAM DDN to COMPLETE are explained in the *VSAM Programmer's Guide*;

(Continued)

COBOL Term	Associated COBOL Verbs	VSAM Term
Random Processing	Access is Random or Dynamic, READ, WRITE, REWRITE, DELETE	Direct Processing
Sequential Processing	Access is Sequential or Dynamic, READ NEXT, START, WRITE, REWRITE, DELETE	Sequential Processing
File Status Code	File status is	From/Return Code. COBOL interprets the VSAM codes and returns its own codes.
Current Record Pointer	N/A	Position within file
READ	READ	Get Macro
WRITE	WRITE	Put Macro
REWRITE	REWRITE	Put Macro with update
DELETE	DELETE	ERASE Macro

- The following manuals are a must when using VSAM, COBOL, and COM-LETE:
 - a. *Access Method Services*
 - b. *VSAM Programmer's Guide*
 - c. *VSAM Programmer's Guide for Advanced Applications*

- d. *OS/VS COBOL Compiler and Library Programmer's Guide*
- e. *OS/VS COBOL Manual*
- f. *COM-LETE COBOL Programmer's Reference Manual*
- g. *COM-LETE System Programmer's Manual*

- h. *COM-LETE User Utilities Manual (*ULIB)*

I hope this information has been of some use to you and would be glad to answer any questions you may have on this article. I may be reached at (916) 441-7232.



COM-LETE Special Interest Group Reports

Integration

Steve Clark
County of Sacramento

Change/Enhancements are already upon us. I am not sure we have all had time to recover from last year's many requests. So far this year we have discussed two new areas where better integration would be beneficial to the user community:

1. A single full screen editor syntax shared by both UEDIT and NATURAL.
2. Allow a NATURAL program to make a decision if an abnormal ADABAS response code is received.

I have submitted Change/Enhancement requests for each of these. To this list I have added a third item. The title of the request is 'Configuration Definition Facility; Category - Integration; Product - COMPLETE/ADABAS'.

During the last session of the Sierra Pacific Users' Group meeting, an extensive discussion of the various data dictionary products was held. The summation of this discussion is that none of the dictionary products (ADABAS, IDMS, DATA-MANAGER, DATA CATALOG 2, etc.) were able to meet the total dynamic needs of today's increasingly complex user. The user community has already requested major assistance from Software AG in this area. The number one vote-getting Change/Enhancement for COM-LETE in 1980 was the user definition facility. The number four vote-getting Change/Enhancement was the revised Data Dictionary. In order to fully make automated documentation functional, the configuration definition facility is mandatory.

In this age of distributed processing and/or distributed data bases, an organization must keep track of the description of data, programs, processes, and users, and also be able to

answer the questions where and how. Current correction documentation must be maintained concerning which CPU, channel, controller, lines, modems, I/O devices, etc. are used to meet each user's information requirements. COM-LETE and ADABAS already have a start in this area with the Terminal Table (TIBTAB) and Field Definition Table (FDT), respectively. Taking these definitions further and incorporating them with the description of the computer system on which they are running, the user definition facility and the integrated Data Dictionary would allow for immediate and/or dynamic response to many areas that are currently problems in today's environment. Through this type of facility immediate notification to users of data unavailability could be made. Immediate or dynamic recovery processes could be initiated. Volumes could be cleared or consolidated to accommodate emergency processing of important data minimizing the affect of a failing device. Alternate routing

(Continued on page 38 bottom)



COM-PLETE Product Rep's Report

Product Representative's Report

*Alan Routhier
County of Sacramento*

As the COM-PLETE Product Representative and long-time (three and one-half years) COM-PLETE user, I frequently receive calls from potential purchasers of COM-PLETE and from new users of COM-PLETE. The former usually ask general questions about the amount of hardware resources which are needed to support COM-PLETE and about the features available with COM-PLETE. The latter usually ask questions about design techniques, programming tips, and formal standards which we have established for online systems, programming in general, and COM-PLETE in particular. About the only thing I can do for these callers is to scratch the surface on each topic and

point them in the right direction for more information.

The direction to which I point prospective and new COM-PLETE users is to visit an established COM-PLETE shop in their area. These arrangements can be made through Software AG sales representatives for prospective users, or by direct call to another user by a new COM-PLETE user. There are several benefits to such a visit. First of all, you can meet and get to know someone nearby who shares your concerns and problems. New users will feel more comfortable going to regional meetings or the annual Conference if they already know someone there. Also, having met someone in person, it is easier to pick up the telephone and call them to exchange ideas and information. Secondly, you can sit at a terminal with your host for hours and exchange information that you could never get over the telephone. An old proverb states that a display on a terminal is worth a thousand words. It has

been my experience that this is true.

Anyone who wants to know about any design techniques, programming tips, and formal standards developed at Sacramento County can look at the Users' Group Newsletters beginning with the July, 1978 issue. This is when I became responsible for the COM-PLETE portion of the Newsletter and, needless to say, I have relied heavily on my organization for articles. The article on "VSAM Use Under COM-PLETE" by Ken Torri in this issue is an example of this. Many of the papers presented at the annual conferences are also complete enough to be helpful. I do not know how articles and papers could be made available to users, but if enough interest is shown by the Users' Group, I am sure that something could be arranged.

Please feel free to call me to discuss this and other things that may be on your mind. In order for me to represent you, I need to find out what you are thinking.

capabilities could be greatly enhanced. Batch processes in the input queues could be held dynamically until all the resources they require are again available. These are but a few of the benefits that could be realized from this type of facility. This facility would round out the requirements for an automated, integrated "data encyclopedia". According to the users with

whom I have talked, the need is here today.

If you as a user are impacted at all by any of these three Change/Enhancements, I urge you to cast some votes for them. Software AG does listen to its users. It is only in this manner that we, as users, may gain additional benefit from the software to which we have already made major commitments.

COM-PLETE Special Interest Group Chairpersons

ADABAS/COM-PLETE/NATURAL
INTEGRATION
Mr. Steve Clark
County of Sacramento Data
Processing
700 H Street, Room 0740
Sacramento, California 95814
(916) 440-6301

PERFORMANCE AND TUNING
Vacant

NATURAL News

NATURAL V1.2 Enhancements

With the Change/Enhancement voting process in the near future, some users have asked for definition of the features which will be included in the NATURAL 1.2 system. Here is a list of the features:

- A generalized 3270 logical driver which will provide support for video devices other than 24 x 80;
- Dynamically specified parm options which will allow users to configure multiple NATURAL environments while retaining a single copy of NATURAL in virtual memory;
- TP driver enhancements including full-screen TSO capabilities, the ability to run NATURAL in pseudoconversational mode in a CICS environment, extended hard-copy device support for all TP monitors, and improved support for various "break" keys;
- Internal enhancements to the product include a generalized security mechanism, display of NATURAL session parameters, terminal type interrogation, PF and PA key support, removal of the requirement for the

"obtain" verb, generalized debug capability, generalized user security exit at terminal input time, a transaction driver mechanism, clarification of some of the NAT3XXX messages, a facility to provide a file PASSW parameter at execution time, and other changes to allow for operational efficiencies;

- Improved integrity for "SAVE" and "CAT" operations. Support for multiple printers for TSO (batch) and COM-LETE users;
- An execution-only configuration;
- NATURAL 1.2 will officially support TASKMASTER and SHADOW II.

NATURAL V1.1 Update Applications and ADABAS V4.1 Transaction Logic

When designing any online update application in NATURAL, it is important to know when ADABAS ET (END OF TRANSACTION), BT (BACKOUT TRANSACTION), OP (OPEN), and CL (CLOSE) commands are issued. These commands are discussed as they relate to the NATURAL

1.1 OS Zap Level 7 (current maintenance level) user who is operating in the multi-programming mode under ADABAS Version 4.1.

The ADABAS OPEN command is issued automatically by NATURAL:

- At the beginning of each NATURAL session;
- At the beginning of each NATURAL program which has made reference to an ADABAS file. NOTE: If the OPRB parameter is supplied to the executable NATURAL module, no OPEN command is issued here.

The ADABAS CLOSE command is issued automatically by NATURAL:

- At the normal termination of each NATURAL session;
- At the termination of each NATURAL program which has made reference to an ADABAS file. The program may end through normal end of job, a FETCH operation, or through an interrupt indication such as the %% command or hitting the terminal clear or break key. NOTE: If the OPRB parameter is supplied to the executable NATURAL module, no CLOSE command is issued here.

The ADABAS BACKOUT TRANSACTION command is issued:

(Continued)

- Automatically by NATURAL, beginning with OS Zap Level Z, when any interrupt indication is given by the user. This may be %, or hitting the clear or break key;
- Under user control, when a BACKOUT TRANSACTION statement is executed within a NATURAL program;
- Automatically by ADABAS, in the event that ADABAS or the TP monitor under which the NATURAL program is running abnormally terminates due to hardware or software failure;
- Automatically by ADABAS, when a user transaction exceeds any of the limits specified by the ADABAS ADARUN parameters CT, TNAE, and TT. These parameters are discussed in the *ADABAS Version 4.1 Operations Manual*.

The ADABAS END OF TRANSACTION command is issued:

- Automatically by ADABAS immediately before any CLOSE command (see CLOSE command above);
- Under user control, when an END OF TRANSACTION statement is executed within a NATURAL program.

In reviewing the general command behavior described above, the user should pay particular attention to the OPRB parameter (also discussed in the *NATURAL Installation and Operations Manual*). With the OPRB parameter, NATURAL issues only one OPEN command and one CLOSE command at the beginning and end of a NATURAL session, respectively. Without it, NATURAL issues multiple OPEN and CLOSE commands throughout the session. Thus, the OPRB parameter affects not only

NATURAL performance, but also may affect a user's ability to take full advantage of ADABAS transaction logic.

Two general application types are used in creating a NATURAL update application. The first type we will call Type I, the simple case, where one update transaction is to be performed within the confines of one NATURAL program. Because only one program is involved, the user need not be concerned with the presence or absence of the OPRB parameter. Type II is the more complex case where an update transaction spans multiple NATURAL programs. The Type II application may be the most common for large online update applications and here the OPRB parameter is important.

When developing a Type II application, it is important to consider the consequences of not using the NATURAL OPRB parameter. Without coding a file list in the OPRB parameter, NATURAL automatically issues a CL command each time a program is FETCHed. Immediately

before the CL command, ADABAS automatically issues an ET command. This means that if a user desired to interrupt (or backout) the transaction in the second or subsequent program of a multi-program update, the backout would only be partial because an ET command was issued in the prior program. If, however, the ADABAS files to be accessed were coded in the NATURAL OPRB parameter and included in the NATURAL executable module, then only one OPEN and one CLOSE command would be issued for the entire NATURAL session. Thus, no ET/CL command combination would be issued as a result of the FETCH. This would allow a user to backout the entire transaction at any point in the application.

Below is sample NATURAL coding for a Type II application which may be used in combination with the OPRB parameter and would allow the NATURAL user to take advantage of ADABAS Version 4.1 transaction logic.

```
0010* UPTPGM1
0020 SET GLOBALS TR=ON
0030 INPUT 'ENTER A NAME:' A(A20)
0040 STORE PERSONNEL NAME = A
0050 GET PERSONNEL *ISN (0040)
0060* MOVE REQUIRED BECAUSE PACKED DATA CANNOT
0070* BE PASSED IN A FETCH OPERATION
0080 MOVE *ISN (0050) TO B(N8)
0090 FETCH 'UPTPGM2' B
0100 END
```

Example 1

```
0010* UPTPGM2
0020 SET GLOBALS TR=ON
0030 INPUT #ISN(N8)
0040 GET PERSONNEL #ISN
0050 INPUT 'ENTER A FIRST NAME:' B(A15)
0060 UPDATE (0040) FIRST-NAME = B
0070 INPUT 'ENTER THE APPROPRIATE OPTION:' //
0080 '1 END THE TRANSACTION'//
0090 '2 BACKOUT THE TRANSACTION' //
0100 'ENTER OPTION NUMBER:' N(A1)
0110 IF N = '1' OR = '2' MOVE 1 TO X ELSE
0120 REINPUT WITH 'INVALID OPTION' MARK 1
0130 IF N = '1' END TRANSACTION
0140 IF N = '2' BACKOUT TRANSACTION
0150 FETCH 'UPTPGM1'
0160 END
```

Example 2



NATURAL Technical Notes

Last Call for Languages

Dan Nolan
Civil Aeronautics Board

"Nature uses as little as possible of anything." — Johannes Kepler (1571-1630).

"Too good to be true" is an aphorism generally reserved for the ephemeral phenomenon, and when I hear it applied to the language NATURAL, it rings like a death knell. It is almost too good to be true, and therein lies its foible, the Achilles' Heel which could spell its demise. NATURAL doesn't behave like a programming language. It arrives in a slick package, ready to go with a minimum of effort. It resembles too much one of those glossy brochures your director obtained at the latest trade fair, extolling the ease of producing multi-color graphics with your present 140, with his comment pencilled in the corner, "Why can't we do something like this. . ."

The danger to NATURAL is that it appears to be too easy, that it dispenses with too much work. It is difficult for some to view claims of 75% savings in time and 90% savings in code without associating a corresponding reduction in power.

Another problem with NATURAL lies in its ease of use. By failing to provide a technical challenge to the programmer to gain recognition, it may be forever relegated to the status of a program generator, of a limited-purpose access method, of a toy.

The facts are, NATURAL doesn't eliminate work - it simply uses the effort more wisely. The actual functions still exist, placed in a more convenient aspect. Hence, effort is reduced. Examination of the tasks common to all programming languages will pinpoint the individual tasks and their associated efforts:

Data. Each programming language is based on a data definition, reflecting a file design, and needed for access. Conventional languages require constant repetition of this for each program, enforceable through standards and procedures which are, in turn, at the mercy of political and personal winds of change. The file carries within its structure the purpose and intent of the designers, access by which will unfold the wonders of a perfect design. Unfortunately, the conventional programs cannot tap into this fund of knowledge and must, instead, rely on some form of documentation for their expertise.

NATURAL programmers, on the other hand, are furnished with the original design of the initial developers through the Data Dictionary, automatically available at the drop of a file name.

(This set of benefits presupposes the reader's agreement that DBMS introduced a new concept of "data driven" structure to programming. Conventional files reflected their state of repose, while DBMS files are constructed for their purpose or activity with only secondary regards to the storage medium employed. Hence, coding which "flows" with the data rather than wresting fields from files will be closer to the design, to the purpose, more efficient.)

I/O. Files must be identified, logic must be written to avoid until open; once open, read; once read, sort (if needed), to select and discard, to process until processed and read until read; and when read, to close and never to return.

Those familiar with NATURAL recognize that all the above is incorporated in a single statement which specifies the file, the nature of those records desired, and which creates a processing loop which will stop when done.

Logic. This is the area causing most of the problems encountered by programmers. This is the birthplace of the "Alter", the multiplexing "If" statements, of the flowcharts, decision tables, and branches into the vast unknown. In establishing the processing logic, the programmer forms the stuff which will make or break a program - later.

NATURAL processing takes place before, within or after a file accessing loop. In simple terms, processes are placed where they belong. To be sure, overrides exist such as the "If no record found. . ." statement, but their relevance diminishes. (If no record was found, what the blazes are you doing here?) To take refuge in earlier statements, the logic of a program follows the design as embodied in the data description.

Work. Our reason for being. As conventional programmers, we have spent over three quarters of our time on prerequisites. If our project has experienced any form of overrun or shortfall, it will become manifest here. This will be the area that receives short shrift or, worse, at this point errors in design become apparent which could

(Continued)

negate all previous efforts.

NATURAL programmers will have achieved this in the previous tasks. They may use this time in documentation (manuals), proofing, improving the system or themselves.

"We are confronted by a condition, not a theory." - Grover Cleveland.

We have seen how the four tasks (Data, I/O, Logic and Work) are accomplished in different ways both with NATURAL and with other languages. With NATURAL, the design of the data carries much of the burden, saving most of the effort and removing nearly all risk. Even this represents only one of the areas benefitted by a NATURAL language. How about debugging?

The latest universally recognized Act of God was the Creation. All other problems can be attributed to error or mishap. Data exceptions are guarded by NATURAL languages, and the language will recognize an inappropriate command at the time it is attempted and will not continue until corrected. There are no branching instructions, so your code won't wander off on its own. File control is implicit and removed as a source of trouble. With earlier languages, as with small children, we became accustomed to indigestion at the slightest offense, unable to determine whether the source of the problem was wet diapers or meningitis without a quick (and costly) trip to the doctor or a dip into the dump. Can it be that our infant industry is growing up, and can now tell us what the trouble is - whether input, syntax or logic?

The Final Point: NATURAL can do anything COBOL can, better. In preparation for the arguments,

remember that everything is controlled by the design. Conventional languages are accustomed to ripping meaningful information out of indifferent forms, of activating a sedentary design and whipping it into shape. With NATURAL, if you can't do it, look at the design. Correct the design (being clever not to disturb existing programs) and whip out the program. I trod carefully for the first few months, avoiding that statement lest it incur the wrath of Those Who Know Better. By realizing that the design is not sacrosanct, that change to design is not synonymous with rewriting programs, others are making that statement. By using the language on non-DBMS files, they are making it come true.

"You can't sit on the lid of progress. If you do, you will be blown to pieces." - Henry J. Kaiser.

P.S. I own, and drive regularly, a Kaiser.

NATURAL CMPEXIT Revisited and NATURAL With No Data Base

*Walt Kapica
Johnson & Johnson*

Since my article on "NATURAL User Exit to Save Programs on an OS PDS" was published (November issue of the Users' Group Newsletter), I have been receiving calls from people all over the country requesting additional information. I would like to share these questions and my replies with all

NATURAL users.

Q. Does your copy of CMPEXIT work in BATCH as well as TSO?

A. Yes.

Q. Does your copy of CMPEXIT work under COM-LETE?

A. Unknown. I'm told that it will work with some small modifications. Several people who have COM-LETE have requested the exit. Since J&J does not have COM-LETE, I am not sure of the protocol used. Perhaps, someone who has COM-LETE and CMPEXIT will respond.

Q. The article mentioned that you can run NATURAL with no data base. How is this done? What are the advantages of doing this?

A. Actually, you can create a NATURAL module to run with no data base even if you don't use CMPEXIT. The problem is that you would not have a way to retrieve programs that have already been written since these programs would normally reside on an ADABAS file. This would be a very limited NATURAL for "online" processing, but you could use it with BATCH. However, with CMPEXIT EXIT installed, all your programs (source and object) are still available. The NATURAL load module to run with no data base would be created as is shown in Example 1:

(Continued)


```

//J081800N JOB (13,2993,10,10),'W KAPICA SEC S',CLASS = K,
// MSGLEVEL = 1,MSGCLASS = L,NOTIFY = KAPICA
//* THIS NATURAL MODULE DESIGNED TO RUN WITHOUT THE DATA BASE.
//STEP1 EXEC ASMFCL,MAC1 = 'ADABAS.NATV11.SOURCE'
//* FNR = 8 ONLY TO ALLOW SAVING OF SOURCE AND OBJECT PROGRAMS
//* THRU CMPEXIT. THE DATA BASE IS NOT USED.
//* TPSYS = CICS ONLY BECAUSE NATTSO LOOKS LIKE CICS AT THIS POINT
NTPRM FNR = 8,USIZE = 32,OPSYS = OS/370,TPSYS = CICS
END
//SYSPRINT DD SYSOUT = *,DCB = BLKSIZE = 1089
//LKED.SYSLMOD DD DSN = DBA.ADABAS(NATPNODB),DISP = SHR
//LKED EXEC PGM = IEWL,PARM = 'XREF,LIST,LET,NCAL,SIZE = (114688,45056)'
//X DD DSN = SYS1.LINKLIB,DISP = SHR
//MYLIB DD DSN = DBA.ADABAS,DISP = SHR
//SYSLMOD DD DSN = ADABAS.NATV11.LOAD,DISP = SHR
//SYSPRINT DD SYSOUT = *
//SYSUT1 DD DISP = (,PASS),SPACE = (TRK,(70,35),,CONTIG),UNIT = SYSDA
//SYSLIN DD *
INCLUDE SYSLMOD(NATTSO) NATURAL FULL SCREEN DRIVER
CHANGE IEFBR14(ADABAS) FAKE A DUMMY ADABAS
INCLUDE X(IEFBR14) DUMMY 'ADABAS' MODULE
INCLUDE SYSLMOD(NATURAL) NATURAL MODULES
INCLUDE SYSLMOD(NATDISPT) NATURAL TP DISPATCHER
INCLUDE MYLIB(NATPNODB) PARM MODULE (TP NO ADABAS)
INCLUDE SYSLMOD(NATTEXT) SYNTAX TEXT MODULE
INCLUDE SYSLMOD(NATTXT2) MISCELLANEOUS TEXT MODULE
INCLUDE MYLIB(CMPEXIT) J&J EXIT TO READ SAVE PDS
ENTRY CMSTART
NAME JJNNODB(R)
/*

```

Example 1

The key to NATURAL with no data base is with the linkedit instructions:

```
CHANGE IEFBR14(ADABAS)
INCLUDE X(IEFBR14)
```

(This information was supplied thanks to Mr. Mick Wokich of Software AG.)

IEFBR14 is an IBM utility that sets a condition code of zero (Register 15 = zero) and does a branch back on Register 14. This is enough to satisfy NATURAL into thinking that it has connected to the data base.

Some of the advantages of NATURAL with no data base are as follows:

- You can use NATURAL to do full screen mapping (NATURAL INPUT statement) and pass responses to COBOL application pro-

grams (NATURAL CALL statement). Used in this way, NATURAL becomes a great front end to existing programs. Many of these programs have nothing to do with data base, so that it would be foolish if these programs failed because the data base happened to be down. Also, you could write a totally new application just using NATURAL alone;

- NATURAL can be turned into a nice little utility program that you could use to change the contents of sequential files or do some record selection and/or reformatting;
- You can also use the NATURAL editor to change

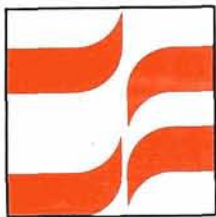
```
TSO: ALLOC F(NSSTUFF)
DA('SYS2.PROCLIB') SHR
```

```
BATCH: //NSSTUFF DD
DSN = SYS2.PROCLIB,DIS P = SHR
```

```
LOGON to NATURAL as follows:
LOGON STUFF
EDIT (member name of the PDS)
```

Since CMPEXIT makes the connection via DDNAME (NSSTUFF), it is possible to use NATURAL as a utility editor.

This is only a partial list of things that could be done. Use your imagination. One final note: We would like to see Software AG release CMPEXIT as a part of their normal release. Any encouragement offered by other users of the exit would be welcome.



NATURAL

Special Interest Group Reports

NATURAL Features

*Jim Reiner
County of Sacramento*

During the Sierra Pacific Regional meeting held on November 7, 1980, several users got into an active discussion concerning future NATURAL features. After the dust had settled, there were clearly two needs that required attention: the end user and the NATURAL programmer. The end user needs attracted the most interest by far; almost everyone at the meeting had something to say about it.

First, the end user and his needed features will be discussed. We determined that the end user works best with specific elements of data according to a predetermined procedure, uses a limited number of commands, and is basically unconcerned with data structure and flow. In addition to this, end users are transaction oriented, require current or timely data, and have diverse views of data. Frequently, end users have a library of tailor-made programs for their use.

We believe the following features would respond to these generalized needs:

- Freedom to construct a query in whatever manner seems most natural, that is, in any order the user likes. This requires a highly non-procedural language that is capable of supporting several different ways of thinking about a problem, a language

that would not mold the user's thought process. Related to this is the clarity and ease with which the user may formulate his request. The naturalness and human engineering of the language for user interrogation is significant. It may be difficult for a nonprocedural language to make certain that what the user wanted was what he received. However, the easier the specification of selection criteria, the greater the chance the user will correctly specify the proper data selection criteria;

- The communication interface between the end user and his data in a file must be understandable, flexible, and simple. The INPUT (or MAP) statement has been a quantum leap in the area of display terminals. However, the INPUT statement is procedural. Some means for the end user to understand the nature of terminal input data and how to "get at it" is needed;
- Finally, we agreed that many of the end user needs tend to be discussed on an abstract level. Perhaps the development of a profile of end users would clarify needs so NATURAL enhancement can provide the capabilities to serve these needs. We heartily encourage this approach.

The second need topic of discussion was the NATURAL programmer. First, his environment and his actions must be defined. The programmer is a user who acts on the data base in unanticipated ways. He is familiar

with the concepts of programming and knows the structure of the data base; thus, he is able to develop systematic approaches to using NATURAL. The programmer requires easy to understand diagnostic messages and error messages in order to make correct use of the data structure, access paths, and data editing. He must also have confidence in the performance of the system and the quality of the data. A common characteristic of all new programmers is a relatively high error rate.

With this in mind, the programming features can be analyzed:

- The NATURAL programmer needs a comprehensive, stand alone language to perform the function of inquiry and update. He needs to have access to the data and its structure in the Dictionary. The LIST FILE command makes this possible, but would be a more responsive tool if it supported paging functions like COMPLETE's *UEDIT. (The %P command provides some browsing techniques, but its use is not documented clearly.);
- For NATURAL to be an effective tool for the programmer, the language must support the following functions: security checks, error recovery and interrogation, subschemas, executable load modules, and selection criteria;
- A comprehensive security scheme is definitely needed. The problem of access control is intensified because of the programmer's knowledge of how to get around security and his impatience

(Continued)

with restrictions that impede his use of data. It is desirable to isolate the files that he updates from the remainder of the data base. It would be desirable to have an exit that forces a user to log on, allows no other log ons, and does not allow any RUN, EXECUTE, SAVE, EDIT, or CATALOG commands to retrieve or place data in another log on ID. An execute only mode would be desirable also. Ideally the final security scheme should secure user IDs, application environments, individual programs, and the NATURAL functions against unauthorized access. We understand that some security means have been addressed in Version 1.2 and that future releases will continue to address it;

- Error condition interrogation and diagnostic messages must be a reality for both the end user and the programmer. It is not appropriate for an application program to terminate because the programmer cannot interrogate response codes. This, too, is being addressed in Version 1.2.

The list of programming features needed in addition to those now existing and those first mentioned include:

- Full screen editor;
- Array processing;
- PF Keys definition, ability to call a NATURAL program;
- Listing cross-references in a program;
- Positive feedback to system commands.

Many of these are included in the Version 1.2 system.

Someone in the group sug-

gested that since the end user requirements are different from the programmer, there should be two separate languages. After discussion we decided that this would be unrealistic. While the requirements appear to be different, they are essentially the same, using the data base for inquiry and update.

It would be well to mention here that all users were concerned that NATURAL may approach a COBOL-like language. The general reaction to this was negative. Programmers want to be able to do what they have to as simply as possible.

Dear Mr. Reiner:

I was delighted to hear that a Special Interest Group for NATURAL Features has been formed. I am particularly interested in contacting anyone who has developed a NATURAL security technique where the end user is responsible for his own password. If you are aware of anyone who has worked in this area, please ask them to contact:

Cleburne Fritz
Pioncer Corporation
P. O. Box 511
Amarillo, Texas 79163
(806) 378-3330

Sincerely,
Cleburne Fritz

NATURAL Special Interest Group Chairperson

Features

Mr. Jim Reiner
County of Sacramento Data Processing
700 H Street, Room 0740
Sacramento, California 95814
(916) 440-6301



NATURAL

Product Rep's Report

Product Representative's Report

*Steve Froneberger
Duke Power Company*

The use and acceptance of NATURAL at Duke Power Company is growing rapidly. Our users (mostly programming personnel) are progressing along the learning curve and the amount of support we have to provide is decreasing. One area which causes us some concern

is the fact that NATURAL runs in a conversational mode under CICS. Our standards are that CICS programs should be pseudoconversational and the rapid growth of NATURAL in the CICS environment has attracted attention. We also hope to see the 32K size limitation lifted. Although we acknowledge the fact that small programs are easier to write, debug, and maintain, we have some situations in which breaking a program down into smaller programs is a difficult task. Program format and comments usually suffer in these cases. If you have the same opinions or other problems that

you face, please let Jim Reiner (Chairperson of the NATURAL Features Special Interest Group) or me know of these.

I was pleased to see the number of articles related to NATURAL in the November Newsletter. Please keep these coming because they provide great benefits to all other users.

The Innisbrook Conference promises to be the best Users' Conference to date. Software AG is going to have more technical presentations and, as usual, the users' presentations should be excellent. I am looking forward to seeing everyone in May.

Software AG

OF NORTH AMERICA, INC.

News

Technical Services

Gordon Perrins
Software AG

We have received many questions regarding technical support procedures and would like to take the opportunity to review these now.

Technical Support—Problem Reporting Requirements

The Software AG user should assign one person per product as the primary contact for reporting problems to the Software AG Technical Support Group. Whenever feasible all problems relating to a Software AG product should be reported by that person. In this context it is currently reasonable to assign responsibilities for the following product categories:

- ADABAS (including ADAMINT, Data Dictionary, ADASCRIP+)
- NATURAL/ADACOM
- COM-LETE

When a problem is first encountered, the user should effect the following procedures:

- Collect together all appropriate documentation for the problem. The type of documentation required is described in detail later on in this article;

- Ensure that the problem can be demonstrated or reproduced using an unmodified product, i.e., the product as it existed on the release tape with only those changes applied that have been distributed by Software AG in the form of system modifications or early warnings.

Phone Support

Once the user has ascertained that there is a valid problem, a trouble call should be placed to Software AG Technical Support. The call should be placed as follows:

- First level of support—Regional Technical Representative.
- Second level of support—Field Support Group.

Where feasible, whenever a Regional Technical Representative is not available, their calls will automatically be routed to the Field Support Group;

Currently, phone support (Field Support Group) is available at two of our North American locations, Reston and Denver. After February 1, 1981 all calls will be handled from the Denver location under the management of Paul Peterson. The support phone numbers are

shown in the box at the end of this article.

The user should provide the Software AG contact with:

- A description of the problem;
- The severity of the problem. Please use the following categories when identifying the severity of your problem:
 - Production system not functioning, i.e., data base down (severity level 15);
 - Production program not working or loss of production data (severity level 12);
 - Important delays (testing) experienced (severity level 8);
 - Points of information and non-critical problems (severity level 4);
- A phone number where the user contact can be reached;
- Your internal Software AG customer number, if known.

If the user is in contact with a Regional Technical Representative, an immediate resolution of the problem will be attempted.

If the user is in contact with Field Support Group, the incoming call will be initially handled by a Technical Support Coordinator who will log the call. If possible, the call will then be routed to the appropriate Technical Sup-

(Continued)

port member for immediate action.

Product Support

There are occasions when Technical Support will request that dumps and/or other related listings be sent to Software AG in order to effect a problem resolution, i.e., it was not possible to resolve the problem over the phone.

The Product Support people responsible for troubleshooting user problems which cannot be resolved over the phone constitute the third level of support. This area is known as the Product Support Group. This group is based in Reston under the management of Gordon Perrins and is also responsible for:

- Quality assurance
- System modifications
- Software distributions
- Technical distributions (e.g., Early Warning, Technical Memoranda, etc.)
- Backup support for the Field Support Group and the Regional Technical Representatives
- International affiliate support

It is important that we receive all relevant documentation from a user regarding a problem. The following list details some of the items that are necessary for the troubleshooting process.

The package should include a letter or a note which provides:

- The organization name;
- User contact name;
- Description of the problem;
- When available, the internal problem log number generated by Software AG. The latter item should normally be available at this time for reference and tracking purposes. If the user has already been in contact with

Technical Support, then the problem will have been assigned an internal log number and a problem log will exist on the Software AG trouble log system. If the user has not been quoted a trouble log number, they should ask the Software AG representative for the number. (Please note that this number may not be immediately available if you are in communication with a Regional Technical Representative as they have limited access to the online trouble log system.) If this number is referenced with the documentation, we will use the same number for problem tracking purposes.

When a dump is received, a "Dump Receipt" letter is generated and mailed to the user. There are exceptions to this when a user is contacted verbally, because perhaps an immediate solution is apparent. However, if the user does not receive any written or verbal communication within a reasonable time frame, he should contact the relevant support center.

There are certain items that should normally be included in any problem documentation package:

- Data Base report(s) (ADAREP);
- ADAICK of file(s) in question;
- Linkage editor output (where applicable);
- Core dump (where applicable);
- All listings (Assembly/Compilation/Execution) of problem program including JCL and input streams (where applicable);
- MPM log (LOG = ALL if possible)

(where applicable);

- Listing of all special/optional zaps applied to the software.

There are items that should be included dependent upon which product has the problem:

Data Dictionary

- Include DD07 report for the file(s) concerned;
- Include the appropriate report that shows the area of the dictionary in which the problem is occurring, i.e.,
 - Data Base—DD05
 - File/Fields—DD07
 - System/Program/Module—DD21, 22, 23, 24 or DD25
 - Verification—DD16
 - Owner—DD40 through DD44
 - Relationships—DD06

ADAMINT

ADAMINT assemblies

- UTIL02 output listing or listing of the assembly SYSIN stream;
- All modules should specify PRINT = DEBUG and FIXDCS = YES on the AMPARMS macro.

User program

- Compilation listing;
- Linkage editor output with LIST, XREF, and LET options specified.

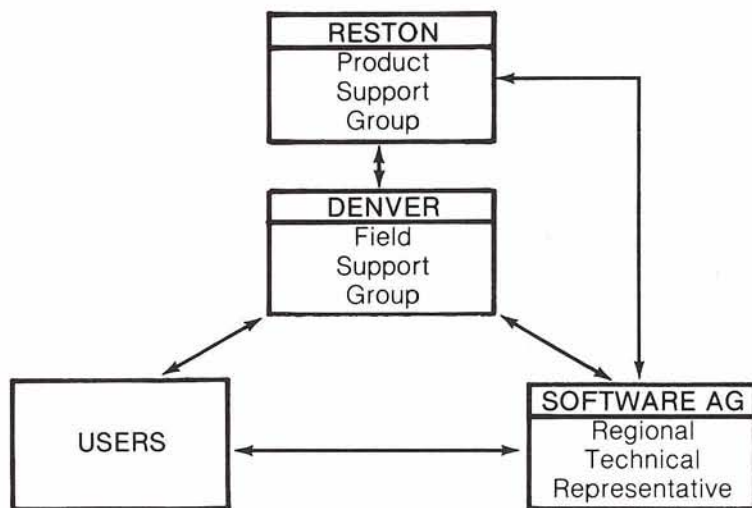
Run time

- MPM log;
- SNAPINT and TESTDUMP outputs (where applicable).

ADASCRIP +

- HNAMEs assembly/UTIL09 outputs when generating the DDM;
- Assembler listing of ASRDSC (only required when the ADASCRIP+ defaults have been modified).

Field Support Group (After February 1st, 1981)



Technical Support Phone Numbers

Field Support Group

Denver 303—233-6500

The above number may also be used for **emergency** technical support.

Product Support Group

Reston 703—860-8100

NATURAL/ADACOM

- HNAMEs assembly/UTIL09 outputs from generation of DDM;
- Assembler listing of NTPRM/CMPRM modules.

COM-PLETE

If COM-PLETE itself abends:

- Linkedit map of COM-PLETE
- SYSABEND dump

If a user program abends:

- Program listing, where applicable;
- Thread dump of the program;
- User exit listings, where applicable;
- (MVS only) "SLIP" dumps of the COM-PLETE address space at the time of the abend.

NOTE OF THANKS

I would like to express my appreciation to those users who responded to this year's Datapro Software Survey. I know it takes time to fill out the survey form, but it is a great help in maintaining the "public credibility" of our products.

Datapro sent 5,000 questionnaires to a sample of COM-PUTERWORLD subscribers. 301 software products received 3 or more ratings. ADABAS received 14. COM-PLETE received 5. NATURAL did not receive the minimum 3 ratings, so was not included. We hope that NATURAL will receive many more ratings next year.

Thanks again for your cooperation.

Sincerely,
John Norris Maguire

Software AG Announces Organizational Changes

In keeping with the growth of Software AG product use throughout the United States and the increasing requirements of the user community, Software AG has redefined its United States sales regions. Kenneth D. Rardin, Executive Vice President, named Jeff Pollack and Michael Ater as Marketing Vice Presidents for two of these regions. Furthermore, Mr. Rardin announced the opening of new sales offices in Denver, Colorado; Atlanta, Georgia; Seattle, Washington; and Laguna Beach, California.

Effective January 1, 1981, U.S. users will report to one of five sales regions. The Western Region is under the direction of John Weil. It includes Washington, Idaho, Montana, Oregon, Wyoming, California, Nevada, Utah, Arizona, and New Mexico. Software AG users may contact the Western Region marketing representatives at either the Los Angeles or Laguna Beach office. The Northeastern Region, headquartered in New York City, continues to be headed by Michael Jakes. This area includes Maine, New Hampshire, Vermont, Massachusetts, Connecticut, Rhode Island, New York, Pennsylvania, and New Jersey. The North Central Region, under Ted Charlebois, includes Wisconsin, Illinois, Indiana, Ohio and Michigan. The newly formed Southeastern Region is headed by Jeff Pollack. Mr. Pollack's sales

responsibility covers Maryland, Virginia, West Virginia, Kentucky, Tennessee, North Carolina, South Carolina, Georgia, Alabama, Mississippi, Washington, D.C., Delaware, and Florida. The Central Sales Region will be directed by Michael Ater. North Dakota, South Dakota, Minnesota, Nebraska, Iowa, Kansas, Missouri, Colorado, Oklahoma, Arkansas, Texas and Louisiana are in this region. Sales representatives in this territory may be contacted via the Houston or Denver offices.

Mr. Rardin was pleased to announce the promotion of Sunday Lewis to Director of Marketing. In addition to maintaining a full range of marketing services and public relations activities, Ms. Lewis will coordinate and direct marketing efforts for all Software AG products.

Aaron Zornes has recently joined the Marketing Staff as Product Manager for ADABAS-M.

In addition to these significant changes in the Marketing Department, Mr. Rardin has appointed John Okulski as Vice President of Finance and Chief Financial Officer. Mr. Okulski's expertise spans all accounting and financial responsibilities. He will continue to streamline accounting procedures and direct all financial and accounting needs within Software AG.

Gary Bowers, Vice President of Consulting and Education, has established an Education Department and three regional field offices for Consulting and Education.

The regional Consulting and Education field offices are responsible for primary coverage of education and consulting assignments. The foremost priority is to ensure adequate staffing

and experience levels in order to present all introductory COMPLETE/ADABAS/NATURAL courses at new customer sites. Field office managers will coordinate all current consulting projects in the respective region and will respond to requests for consulting services. Steve Carter will head the Western region from the Seattle office; Joe Gilley assumes management responsibilities for the Eastern region; Gary Bowers is Acting Manager of the Central region, headquartered in Kansas City.

Larry Jayne serves as Manager of the Education Department. His department will provide central development and support services for all Software AG education activities. This includes registration and scheduling for all classes, as well as coordination of new course development. Requests for education registration and class scheduling will continue to be processed by the Education Registrar in the Education Department.

Mr. Bowers has organized the division in this way to provide for improvement and expansion of our education curriculum, while continuing to provide adequate coverage of our education commitments. He views this as the basis for expanding both education and consulting services throughout the 1980s.

Publications Update

Rick Josephs
Software AG

1980 Summary of Activities

1980 was a most productive year as concerns Software AG product documentation. Three long-term documentation efforts were concluded:

- NATURAL Version 1.1 (3 manuals plus a reference card);
- ADABAS Version 4.1 (10 manuals plus 2 reference cards);
- ADABAS-M Version 1.1 (4 manuals plus a reference card);
- NATURAL Version 1.1 Update Series 01.

All of the above documentation has been well received by the user community. We at Software AG feel the above documentation represents, in terms of quality, the high standards we have established for all of our software products and supportive material.

1981 Documentation Plans

Documentation efforts planned for 1981 include the following major projects:

1st quarter:

COM-LETE V4.1
ADABAS/VM
Data Base Machine
ADABAS V4.1 Update
Series 01
ADABAS-M V1.1 Update
Series 01

2nd quarter:

NATURAL V1.2
ADABOMP V2.0

3rd and 4th quarters:
ADABAS-M V2.0
COM-LETE DOS
ADABAS V4.1 Update
Series 02

A Documentation Evaluation Survey is also planned during the second quarter of 1981. The objective of this survey will be to determine specific areas in which the user community feels better and/or additional documentation is needed.

The Publications Staff wishes each of our users a most successful new year. We are looking forward to working with you during the coming year in order that we may continue to achieve our ongoing goal of providing the best in software documentation.

1981 User Education Plans Announced

Larry Jayne
Software AG

With an eye toward responding to the requirements of Software AG product users, the Consulting and Education Division has developed new classes and has initiated a new approach to scheduling class offerings.

The success of the DBA Skills Workshop and the opening of the Seattle Education Center has contributed to our growing emphasis on the workshop approach to product understanding. A five-day ADABAS Applications Workshop has been added to the 1981 course offerings. This intensive seminar provides the programmer or analyst an active, working knowledge of ADABAS

programming techniques. Offered for the first time in Seattle, March 9-13, attendance is limited to ensure an optimal, workshop environment.

The second new class included in the 1981 curriculum is ADABAS Service Analysis Methodology. Offered in conjunction with Performance Development Corporation, this three-day course will offer attendees a methodology for ADABAS Data Base requirements definition and the means to translate this into the logical and physical data base architecture.

An expanded three-day NATURAL course will be available for classes scheduled at Reston and Seattle. Workshop oriented, this class will offer users the opportunity to write and test NATURAL programs. The two-day NATURAL course will continue to be scheduled for public classes offered throughout the United States.

We have also made some significant changes in the "order" in which classes are presented in our public training schedule. 'Using ADABAS', the 'ADABAS Design Workshop', and 'ADACOM Usage' will comprise the first week of training during each month while Direct Call Programming, ADABAS Internals and Data Base Administration complete the second week. Public classes are scheduled so that some time (around two months) elapses between the introductory classes and the more advanced classes. This will allow users the opportunity to reflect on how ADABAS should be implemented in their environment.

SOFTWARE AG CLASS SCHEDULE — January - June 1981

Month	Class Using ADABAS	ADABAS direct calls	ADACOM	ADABAS internals	DATA BASE adminis- tration	ADABAS design workshop	ADABAS design methodology
January	RESTON Software AG January 12-14	RESTON Software AG January 15	RESTON Software AG January 16	RESTON Software AG January 19	RESTON Software AG January 20-21	RESTON Software AG January 22-23	RESTON Software AG January 26-28
February	SEATTLE Software AG February 2-4	SEATTLE Software AG February 5	SEATTLE Software AG February 6	SEATTLE Software AG February 9	SEATTLE Software AG February 10-11	SEATTLE Software AG February 12-13	SEATTLE Software AG February 18-20
March	RESTON Software AG March 9-11	RESTON Software AG March 12	RESTON Software AG March 13	RESTON Software AG March 16	RESTON Software AG March 17-18	RESTON Software AG March 19-20	RESTON Software AG March 23-25
	CHICAGO Holiday Inn Mart Plaza March 23-25	CHICAGO Holiday Inn Mart Plaza March 26	CHICAGO Holiday Inn Mart Plaza March 27				
April	NEW YORK Marriott Essex House April 6-8	NEW YORK Marriott Essex House April 9	NEW YORK Marriott Essex House April 10	SEATTLE Software AG April 27	SEATTLE Software AG April 28-29	SEATTLE Software AG April 30-May 1	SEATTLE Software AG April 13-15
	SEATTLE Software AG April 6-8	SEATTLE Software AG April 9	SEATTLE Software AG April 10				
May	RESTON Software AG May 20-22			RESTON Software AG May 4	RESTON Software AG May 5-6	RESTON Software AG May 7-8	
June	SEATTLE Software AG June 8-10	SEATTLE Software AG June 11	SEATTLE Software AG June 12	NEW YORK Marriott Essex House June 8	NEW YORK Marriott Essex House June 9-10	NEW YORK Marriott Essex House June 11-12	RESTON Software AG June 1-3
				SEATTLE Software AG June 15	SEATTLE Software AG June 16-17	SEATTLE Software AG June 18-19	SEATTLE Software AG June 22-24
				CHICAGO Holiday Inn Mart Plaza June 22	CHICAGO Holiday Inn Mart Plaza June 23-24	CHICAGO Holiday Inn Mart Plaza June 25-26	

NATURAL	DBA skills workshop	ADABAS data admin. & control	ADABAS V.4 facilities & conversion	COM-LETE applications workshop	COM-LETE Internals	COM-LETE systems control
	RESTON Software AG January 8-9		SEATTLE Software AG January 12-16			
RESTON Software AG February 9-12			RESTON Software AG February 23-27	RESTON Software AG February 18-20		
	SEATTLE Software AG March 16-17	RESTON Software AG March 2-4	SEATTLE Software AG March 23-27	SEATTLE Software AG March 9-13		RESTON Software AG March 30-April 1
				RESTON Software AG April 13-17	LOS ANGELES Sheraton Town House April 27-29	
SEATTLE Software AG May 18-21	RESTON Software AG May 27-28		SEATTLE Software AG May 4-8			
		SEATTLE Software AG June 1-3	RESTON Software AG June 15-19	SEATTLE Software AG June 29-July 3	NEW YORK Marriott Essex House June 22-24	

Software AG Introduces . . .

Dawn Myhre is a voice which you have all heard. As PBX operator for Software AG, Dawn is in many instances your first contact with Software AG. Her day starts promptly at 8:00 AM and she is kept busy responding courteously and efficiently to a high volume of phone calls. Dawn is able to find some time for light administrative responsibilities, also.

She is assisted by Renie Fulton and Cathy Ames.



Dawn Myhre

ADABAS V4.1

Status: OS—Available
DOS—Beta Testing

General Availability: DOS January 1981

NATURAL V1.2

Status: Beta testing

General Availability: May 1981

ADABAS-M V1.1

Status: Available

COM-LETE DOS

General Availability: 12-18 months

COM-LETE VTAM SUPPORT

Status: Beta Testing

General Availability: Release 4.1

COM-LETE V4.1

Status: Beta Testing

General Availability: April 1981

DATA BASE MACHINE

Status: Available with 60 day delivery

Online Data Dictionary

Status: Beta Testing

General Availability: For NATURAL Users March 1981

Current Release Levels

The following is a list of current System Modifications that are available. Please contact Gordon Perrins, Manager of Technical Services, if you need current SMs.

ADABAS:

ADABAS Nucleus & Utilities—
Version 4.1.1

— OS SM Level 11

ADAMINT Version 1.3

— OS SM Level 10

— DOS SM Level 4

ADACOM Version 1.2

— OS Zap Level 6

— DOS Zap Level 5

ADASCRIP+ Version 1.1

— OS SM Level 10

— DOS SM Level 4

Data Dictionary Version 1.1 OS
and DOS

NATURAL:

NATURAL Version 1.1

— OS Zap Level 7

— DOS Zap Level 6

COM-LETE:

MVS Version 3.1

— SM Level 3

VS1 Version 3.0

— SM Level 6

MVT and SVS Version 3.0

— SM Level 6

MVS & VS1, both with VTAM—
Beta Test Version 3.2

Regional Representatives

Northwestern Region

Mr. Douglas Hart
Weyerhaeuser Company
Tacoma, Washington 98447
(206) 924-6875

Northeastern Region

Mr. Jack Meth
City University of New York
5555 West 57th Street
New York, New York 10019
(212) 977-8000

Eastern Region

Mr. Daniel Nolan
Civil Aeronautics Board
4103 College Heights Drive
University Park, Maryland 20782
(301) 927-5755

Hawaiian Region

Mr. Patrick Gilbert
College of Business Administration
University of Hawaii
Honolulu, Hawaii 96822
(808) 948-8064

Eastern Canada Region

Mr. John Bridgman
North American Life
105 Adelaide W. Toronto
(416) 362-6011, ext. 520

Midwestern Region

Mr. George White
West Publishing Company
P.O. Box 3526
St. Paul, Minnesota 55165
(612) 228-2770

Southwestern Region

Mr. Robert Holub
The Superior Oil Company
P.O. Box 1521
914 Main
Chamber of Commerce Building
Houston, Texas 77001
(713) 751-4046

Pacific Southwestern Region

Mr. David Berg
Hughes Aircraft
Radar Systems Group Div. 21
Box 92426
Building R1/C335
Los Angeles, California 90009
(213) 615-7356

Sierra Pacific

Mr. William Belisle
State of California
Department of Youth Authority
4241 Williamsborough Drive
Sacramento, California 95823
(916) 322-2535

Southeastern Region

Mr. Steven Froneberger
Duke Power Company
422 S. Church Street
P.O. Box 33189
Charlotte, North Carolina 28242
(704) 373-8471

Rocky Mountain Region

Ms. Ellen Kasun
Hewlett-Packard Company
P.O. Box 301
Loveland, Colorado 80537
(303) 667-5000

United Kingdom

Mr. Peter Spooner
ICI Organics Division
P.O. Box 42
Hexagon House
Blackley, Manchester, England M93DA

Germany

Mr. Lutz Heymer
Hessische Landesbank-Girozentrale
Postfach 3163, Junghofstrasse 18-26
6000 Frankfurt am Main 1, Germany
(0611) 132-3403

Switzerland

Mr. Dominic Renaud
Centre de Calcul Electronique de
l'Administration Federale
Hallwylstrasse 15
3003 Bern, Switzerland
(031) 618783

Italy

Mr. Elio Caramagna
S.A.I.
Via Marengo, 15
10126 - Torino, Italy
(011) 65621

South Africa

Mr. Peter Feasey
Shell South Africa Pty Ltd.
P.O. Box 2231
Cape Town, 8000 South Africa

South America

Antonio Amaral Filho
Centrais Eletricas Brasileiras S.A.
Av. Presidente Vargas 64213
2000 Rio de Janeiro, Brasil

Australia

Mr. John Palmer
Data Base Services Section
Australian Bureau of Statistics
P.O. Box 10
Belconnen A.C.T.
527911

Far East

Mr. Yasuo Shimizu
Fuji Photo Film Co., Ltd.
2-26-30 Nishi-Azabu, Minato-ku
Tokyo 106 Japan
(03) 406-2777

Israel

Mr. Yigal Ben Shalom
National Insurance Institute
13 Weizman Institute Avenue
Jerusalem, Israel
02-528082

Saudi Arabia

Mr. Roger Martin
International Airports Project
Ministry of Defense and Aviation
P.O. Box 6326
Jeddah
Saudi Arabia

Software AG Users' Group Regions in the United States



This map represents changes made in the United States regional Users' Group boundaries at the International Conference.

Dates to Remember

Change/Enhancement Ballots
March 6, 1981

Conference Papers
March 9, 1981

April Newsletter Copy
March 20, 1981

Editors:

Jennifer Lee, ADABAS Product Representative and Contributing Editor
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009
(213) 615-4521

Alan Routhier, COM-LETE Product Representative and Contributing Editor
Sacramento County
Systems and Data Processing
700 H Street, Room 0740
Sacramento, California 95814
(916) 440-6301

Steven Froneberger, NATURAL Product Representative and Contributing Editor
Duke Power Company
422 S. Church Street
P.O. Box 33189
Charlotte, North Carolina 28242
(704) 373-8471

Cathy Vivona, Users' Group Liaison
Software AG of North America, Inc.
11800 Sunrise Valley Drive
Reston, Virginia 22091
(703) 860-5050

Sheri Gotay, Production Editor
Software AG of North America, Inc.
11800 Sunrise Valley Drive
Reston, Virginia 22091
(703) 860-5050

International Software AG Users' Group Executive Committee

President
Mr. Michael Berman
INCO, Ltd.
1 New York Plaza
New York, New York 10004
(212) 742-4180

Vice President
Ms. Ellen Kasun
Hewlett-Packard Company
Loveland Instrument Division
P.O. Box 301
Loveland, Colorado 80537
(303) 667-5000

Secretary
Mr. Roger Martin
International Airports Project
Ministry of Defense and Aviation
P.O. Box 6326
Jeddah
Saudi Arabia

ADABAS Product Representative
Ms. Jennifer Lee
Aerospace Corporation
P.O. Box 92957
Los Angeles, CA 90009
(213) 615-4521

COM-LETE Product Representative
Mr. Alan Routhier
County of Sacramento Data Processing
700 H Street, Room 0740
Sacramento, California 95814
(916) 440-6301

NATURAL Product Representative
Mr. Steven Froneberger
Duke Power Company
422 S. Church Street
P.O. Box 33189
Charlotte, North Carolina 28242
(704) 373-8471

Change/Enhancement Coordinator
Mr. Peter MacRoberts
Pennzoil Company
Pennzoil Place
P.O. Box 2967
Houston, Texas 77001
(713) 236-7878

Area Representatives

North American Area
Mr. David Berg
Hughes Aircraft
Radar Systems Group Div. 21
Box 92426
Building R1/C335
Los Angeles, California 90009
(213) 615-7356

European and Middle Eastern Area
Mr. Yigal Ben Shalom
National Insurance Institute
13 Weizman Institute Avenue
Jerusalem, Israel
02-528082

South African Area
Mr. Peter Feasey
Shell South Africa PTY Ltd.
P.O. Box 2231
Cape Town, 8000 South Africa

South American Area
Mr. Benjamin Correa
Telecomunicacoes Brasileiras S.A.
SCS Ed. Anapolis 4 Andar Sala 404
Brasilia DF 70000 Brasil

Oceania Area
Mr. Patrick Gilbert
College of Business Administration
University of Hawaii
Honolulu, Hawaii 96882
(808) 948-8064

Far Eastern Area
Mr. Yasuo Shimizu
Fuji Photo Film Co., Ltd.
2-26-30 Nishi-Azabu, Minato-ku
Tokyo 106 Japan
(03) 406-2777

 **SOFTWARE AG**
OF NORTH AMERICA, INC

Reston International Center
11800 Sunrise Valley Drive
Reston, Virginia 22091