# LAS NEWS

## Land Analysis System

Issue No. 6 | NASA Goddard Space Flight Center, Greenbelt, MD | August 1991

## Contents



## An LAS Valedictory

*by Marilyn Mack (GSFC) - Editor*

It is with great regret that I must announce the termination of not only this newsletter, but of the LAS project at Goddard Space Flight Center. The task will no longer be funded by NASA headquarters since most efforts are now being redirected toward the newer related project, EOS. We are sorry for any inconvenience this may cause you.

You may have been puzzled and/or worried by the lack of newsletter issues for July, 1990 and for January, 1991. Since we had planned and budgeted for only one final issue and since writing articles for the newsletter would have infringed on the valuable time needed to complete the project, we decided to wait until the description of the new LAS 5.0 system would be more complete. That time has arrived, and this issue will attempt to familiarize you with the new features of this latest and final form of the system. When we began this newsletter, we had about 50 user sites and a newsletter distribution list of less than 300. Now we service more than 84 user sites, and more than 640 persons are receiving this publication! Such growth has had its problems, but the rewards have far outweighed them. Thus, this, our farewell, is a nostalgic one for us all.

As a valedictory speech leads its listeners away from the thoughts of school and the past toward a brighter day in the future, this issue will attempt to connect those two elements and point the way for you all to continue your research and communication in the future. There will be some articles on the use of LAS 4.1, but most will concentrate on the new 5.0 release. For those who are not so closely associated with LAS, the technical detail in this issue may be somewhat overwhelming. We apologize for this, but it was a deliberate effort to provide users with as much information as we could in our last official contact with them.

In this, our final issue, I wished to highlight once again those who have helped to make this newsletter a reality. Fred Irani (ST Systems Corp. - Goddard Space Flight Center) has not only written articles (see his by-lines in this issue), but has also suggested sources for other articles. Though the name of Jim Tilton of Goddard has seldom been explicitly seen as an author since our early issues until this one (see his article in this issue), he has continued to provide most of the information used in our *Calendar of Activities*. Phil Pease (see his article in this issue), the LAS Configuration Manager at Goddard, supplies most of the information used in the *LAS "Contrib" Contents* and *LAS Problems and Solutions* features. Yun-Chi Lu, GSFC LAS Project Manager (see his article below), has given both support and potential article sources. These persons have given assistance without any compensation or expected acknowledgment.

The project has been assisted by various forms of support from both Goddard Space Flight Center Management and both Management and Staff personnel at NASA Headquarters. All of the contractor staff at Goddard have contributed significantly to making Version 5.0 of LAS a reality in 1991 (see article in this issue). Needless to say, the entire project would not have been possible without the continued support and cooperation of the many EROS Data Center employees who have worked on the conversion tasks. Thanks also to you, our readers, who enjoyed the newsletter and passed it to other interested individuals. Since we did not advertise its existence, the phenomenal increase in readership is a direct result of your efforts.

Once again, thank you, best wishes to you all, and farewell!

## A Project Manager's Farewell

*by Yun-Chi Lu (GSFC)*

Many of the readers who responded to our survey already know that making LAS run on multiple operating platforms was our primary goal during the last three years. As project manager, I am happy to report that with limited resources the milestone has been reached through a joint effort between NASA and the EROS Data Center. The LAS Version 5.0 is ready in COSMIC for public distribution, and detailed information can be found in related articles in this newsletter.

Since our first public release in 1985, LAS user sites have been increased from 5 to 83 throughout the world. After all these years, there are still over 50 active user sites which are loyal to and dependent on the system for research, education, and production work.

The success of this project can be attributed to many contributors, both individual and institutional, including users such as yourself who have provided us with many valuable comments and suggestions throughout those years. We at the project office did maintain and will always maintain that the system must be run by users for the users to be successful.

As a research organization, operating on and maintaining a mature system such as LAS would not be NASA's primary function. However, termination of the project after Version 5.0 release will not diminish the usefulness of the system, and LAS will continue to serve its purpose in the Earth Science community. Specifically, the system has been routinely used in several sites among the seven Distributed Active Archive Centers (DAACs) under the Earth Observing System (EOS) Data Information System (EOSDIS) Program, a pivotal part of the U.S. Global Change Research Program.

It has been my pleasure to serve as Goddard LAS Project Manager for the last several years, so I will miss working with you in this capacity. Good luck in all your future research!

## LAS 5.0 Sent to Cosmic

*by Marilyn Mack and Philip Pease (GSFC)*

Though it may have seemed to all of you that this day would never arrive, LAS 5.0 has finally been delivered to COSMIC for distribution. Members of the LAS staff both at the Goddard Space Flight Center (GSFC) and at the EROS Data Center (EDC) continued to convert the application software in LAS to the more portable form (primarily through a recoding in the C language) until about one month ago. The final product is one that encompasses many changes from those with which most of you are familiar. Although the primary and most prominent of these is the ability of LAS 5.0 to run on UNIX systems as well as on VAX/VMS machines, others are worthy of consideration.

The catalog manager has been completely redesigned. It no longer uses a catalog file, but, instead, is based around the host operating system file management, (i.e., directory structure) plus a knowledge of the file types that LAS employs. (Note: See the article on new file structures in this issue.) This change has resulted in better responsiveness and a simpler design. It also provides an easier mechanism for getting files into the LAS catalog.

Geometric manipulation software has been greatly enhanced with several new functions and enhancements to other functions. Interfaces to external software have been standardized with a facility named "labeled table" (LT). Another way to provide external interfaces that has been added is the ability of each function that produces print output to have that output go to an ASCII file. ASCII files can also be converted to or from an LT file. Image display devices supported have changed. On the positive side, IIS IVAS and X-Window display devices can now be used for displaying LAS images. On the negative side, the dd's for the IIS Model 75, Rastertek one/25, and DeAnza IP8500 display devices are not provided in this release.

Other changes made involved modifications to individual modules either to correct discovered problems or to enhance their operation. A complete list of these differences can be found with the software delivery documentation and notes. Some highlights of these changes are:

1. ANNOTATE now includes the functionality of RASTERIZE and uses the same character definitions as that used by CLASSMAP and FILM.

2. New functions have replaced 4.1 function(s) -

| 5.0 name(s): | 4.1 name(s): |
|---|---|
| ANNOTATE | RASTERIZE |
| CMDEL | LASDEL |
| CMLIST | LISTCAT |
| CMRENAME | RENAME |
| COPY | CONVERT |
| CORRELATE | EDGECORR and GREYCOR |
| DIFF | COMPARE |
| DELDLTF | DELLUT |
| DITTO | DMOUNT |
| DSPDDR | DSPLBL |
| EDITDDR | DDREDT |
| EDITGRID | COEFFEDT |
| EDITTIE | COOREDT and STVEDT |
| GOF2STAT and GOF2TIE | TRANSGOF |
| HISS | RGB2HIS and HIS2RGB |
| MAPPER | WHATISIT |
| MATH | ADDPIC, ANDPIC, DIVPIC, MAX, MIN, MULTPIC, ORPIC, SUBPIC, and XORPIC |
| NOT | UNARY |
| POLYFIT | TIEFIT |
| STATS2 | STATCUT |
| TAPEIN and SETTAPIN | IENTER |
| TAPEOUT and SETTAPOUT | ITRANSFER |
| TRIG | ARCCOS, ARCSIN, and ARCTAN |
| WINDOP | AVERAGE |
| ZOOMCUR | ZOOPAN |

3. TOPO includes BANGLE.

4. DITTO not only replaces DMOUNT but also includes other functions dealing with tape and disk file operations.

5. The performance speed of BAYES has been improved.

6. EDITSTAT now has four subcommands: INPLACE which performs the same function as version 3.2 EDITSTAT by creating a new file name if name is not found; COPY, which replaces the old interactive copy; SORT, which repositions classes in a file; and LIST, which sends statistics file to the terminal, line printer, and/or an ASCII file (via fileutil). Command HISTPLT was dropped since its functionality is available in the HISTPLT function. Note that changes were also made to interactive command names. In addition, new commands were implemented including: DROPHIST, EDITCOR, and QUIT. Other changes are listed in the documentation.

7. FLIP now rotates an image 90, 180, or 270 degrees, or flips it about its horizontal axis, or mirrors it about its vertical axis.

8. UNARY now includes 4.1 function NOT.

9. The following functions were deleted: ARCHIVE, CCTTIPSA, CONTROL, DEFDDR, DPFMAP, DSPCONTBL, DSPGROUP, DSPVERSA, FILMTAPE, GROUP, NULLCORR, TPLOT.

10. These functions are no longer provided in the baseline, but may be available in CONTRIB: ARITH, BLDPW, BLWTHR, CLEANUP, CONVOL, CURPOS, CURSOR, DEBLUR, FFT1, FFT1FL, FFT1PIX, FFT2, FFT2FL, FFT2PIX, FITLIN, FLAG, GRMDYN, IFFT2, LINEOFF, MAGPHASE, MAGPOWER, PLANE, REGLABEL, RETRIEVE, SMOOTH, SURFACE, TDPLOT, TIC, UNIVENTER, WTGEN.

11. New functions include:

| | |
|---|---|
| ASC2TAB | - Converts an ASCII data file to a Labeled Table (LT) |
| CMMKDIR | - Creates a new directory in the user's directory tree |
| COMPOSIT | - Generates a min or max image from given images |
| CORRECT | - Applies systematic corrections to EDIPS images |
| DESWATH | - Performs one-dimensional filtering (primarily to eliminate swathing/banding found in Landsat TM data) |
| DSPRWT | - Displays a resampling weight table |
| DSPTIE | - Displays tie point selection, merged tie point, and tie point location files |
| EDIPSIN | - Ingests EDIPS version 0.0 formatted image tapes |
| EDITDLTF | - Creates or edits a look-up table |
| FSTFMTIN | - Ingests EOSAT fast format tapes |
| GETELEV | - Gets the elevation of a tie point from a DEM image file |
| GRIDFORM | - Converts geometric mapping grids in other formats to LAS standard format |
| MAPFILE | - Matches two histograms created by PIXCOUNT according to the cumulative percentages |
| MAPPTS | - Creates a tie point selection file consisting of latitude/longitude or user-defined coordinates (x,y) from a map; (it is a ground point collection function) |
| MATH2 | - Creates average and standard deviation bands |
| MATRIX | - Performs matrix algebra on images |
| NUT | - Acts as a tutorial on basic LAS concepts and functions for new LAS users |
| PROCHECK | - Evaluates a procedure pdf and reports errors in the parameters |
| PROJPRM | - Creates a projection definition file |
| PROJWIND | - Creates a valid image window specification in lines and samples from a window specification containing projection coordinates |
| RELIEF | - Corrects ground control points for relief displacement |
| RGB2CMYCK | - Generates a cyan, magenta, yellow, and black image from a red, green, and blue image |
| RTABLE | - Generates resampling weight table files for use in the rectification process from either a user-entered inverse point spread function or various sizes of sinc functions |
| SCAN | - Displays successive windows of an image |
| SETAPIN | - Processes up to 50 images and utilizes dynamic parameters to retrieve information about each input image |
| SETAPOUT | - Processes up to 50 images and uses dynamic parameters to retrieve information about each input image |
| TAB2ASC | - Converts a Labeled Table (LT) file to an ASCII data file |
| TAB2TIE | - Converts a Labeled Table (LT) file to a tie point selection file |
| TAPEIN | - Includes all 4.1 IENTER features and adds blocking of image lines per physical tape record, ability to convert from input data format to the host system data format for VAX, SUN, GOULD, IBM, and IEE systems for all types and HP for all except real, ability to read and |

update a DDR file from tape, and processing of as many as 50 images

TAPEOUT - Includes all 4.1 ITRANSFER features and adds blocking of image lines per physical tape record, ability to output standard IEEE data format, option to update and write a DDR file to tape, and processing of as many as 50 images

TASSELCP - Generates brightness, wetness, and greenness bands from either MSS or TM images

TIE2GOF - Converts a tie point selection file to a Graphic Overlay File (GOF)

TIE2TAB - Converts a tie point selection file to a Labeled Table (LT) file

TRANSLAS - Converts UNIX and/or VMS images to LAS 5.0 images

TRIG - Incorporates ARCSIN, ARCCOS, and ARCTAN and allows input of images with windows having differing start pixel and start line values so long as the size of the window is the same

VERIFY - Computes statistics errors in verification points and includes an option to calculate the amount of error in the verification points due to relief displacement

12. Results from some 4.1 functions may differ from the same functions in 5.0 because of the switch from Fortran to C and their enhancement in 5.0. Many have incorporated improved algorithms, and/or have eliminated dependency on IMSL package routines. Among those so affected are: ALBEDO, DESTRIPE, DISCRIM, GETSAMP, LOWCAL, POWER, RANDSAMP, SEARCH, STEREO, TESTGEN, TEXTDIF, TEXTURE, UNARY

The final baseline package consists of 259 functions. These functions are separated into sections as follows: (1) LAS applications functions, (2) LAS/DMS display functions, (3) WORLD/DITTO tape and disk file manipulation functions, and (4) startup and shutdown functions.

The 187 LAS application functions include:

| | | | |
|---|---|---|---|
| ABS | ADD2STAT | ALBEDO | ANNOTATE |
| ANNOTE | ASC2TAB | AVHRRB0 | BANDRATIO |
| BAYES | BDIST | CANAL | CCTTIPSP |
| CLASSMAP | CMALIAS | CMCOPY | CMDEL |
| CMLIST | CMMKDIR | CMRENAME | CMSET |
| CMSHOW | COMPLEX | COMPOL | COMPOSIT |
| COMSEP | CONCAT | CONJ | CONTABLE |
| CONVOLVE | COPY | CORRECT | CORRELATE |
| COVAR | CROSSCOR | CWTGEN | DELTAB |
| DEMENTER | DESTRIPE | DESWATH | DIFF |
| DISCRIM | DIVERGE | DROPHIST | DSPDDR |
| DSPGLOB | DSPGRID | DSPHISTRY | DSPLOG |
| DSPRLT | DSPRWT | DSPTIE | EDIPSIN |
| EDITDDR | EDITDLTF | EDITGRID | EDITSTAT |
| EDITTIE | FACTOR | FILM | FILTER |
| FIXLIN | FLIP | GAUSTRAN | GEOM |
| GETBLOB | GETELEV | GETSAMP | GOF2STAT |
| GOF2TIE | GRAD | GRID | GRIDFORM |
| GRIDGEN | GRNDSPOT | HINDU | HIS |
| HISTEQ | HISTFAST | HISTPLT | HRPTCAL |
| INFO | INSERT | INTERSECT | ISOCLASS |
| KARLOV | KMEANS | LACIN | LASGBLS |
| LINEOFF | LINEPLOT | LINEREPR | LIST |
| LOGICAL | LOWCAL | LPMAP | LT |
| MAGNIFY | MAKEIMG | MAP | MAPFILE |
| MAPPER | MAPPTS | MASK | MASKSTAT |
| MATH | MATH2 | MATRIX | MINDIST |
| MINMAX | MULTIPLY | NEIGHBOR | NOISE |
| NORMD | NUT | OVERLAY | PERSPEC |
| PICSHADE | PIXCOUNT | PIXSERT | PLENTER |
| POLYFIT | POWER | PROCCHECK | PROJPRM |
| PROJWIND | RADIOM2 | RADIUS | RANDSAMP |
| RECLASS | REDIST | REGISTER | RELIEF |
| REMAP | RENUMBER | RGB2CMYK | ROTATE |
| ROTRNSCL | RTABLE | SAMPLET | SCALE |
| SCATTER | SEARCH | SEGMOFF | SEGMREPR |
| SETAPIN | SETAPOUT | SPECCOMB | SPECSTRT |
| SPREAD | STAT2GOF | STAT2TAB | STATPLOT |
| STATS | STATS2 | STEREO | STRETCH |
| STRUCT | SYSSAMP | TAB2ASC | TAB2STAT |
| TAB2TIE | TAETUTOR | TAPEIN | TAPEOUT |
| TASSELCP | TESTGEN | TEXTDIF | TEXTURE |
| TICMARK | TIE2GOF | TIE2TAB | TIEMERGE |
| TOPO | TRANCOORD | TRANSLAS | TRANSPOS |
| TRIG | TSCLASS | UHIST | UMAP |
| UNARY | UNKNOWN | VERIFY | WINDOP |
| WORGBLS | ZIP | ZOOM | |

The list of 45 LAS/DMS functions includes:

| | | | |
|---|---|---|---|
| ADJUST | ALLOC | CPYGOF | DALLOC |
| DEFATT | DEFCUR | DELATT | DELDLTF |
| DELGOF | DM | DMSGBL | DPTLUT |
| DSTAT | FILL | FLICKR | FRMDSP |
| GOF2TAB | HISTO | INIT | LDM |
| LODDLTF | LODGOF | LOGIC | LSTDLTF |
| LSTGOF | LSTIMG | MAPP | MEASURE |
| MKMASK | MODGOF | PIVOT | PROTEC |
| PSD | PUT | SAVDLTF | SAVIMG |
| SCAN | SHOGOF | SHOIMG | SHOMAP |
| TAB2GOF | TIEPTS | TODSP | WINDOW |
| ZOOMCUR | | | |

The 17 WORLD/DITTO Tape and Disk File Manipulation functions include:

| | | | |
|---|---|---|---|
| ALLOC | BSF | BSR | COPY |
| DEALLOC | DITTO | DITTOX | DUMP |
| FSF | FSR | GBLS | MAPPER |
| REWIND | SCAN | STATUS | VERIFY |
| WEOF | | | |

Finally, the 10 Startup and Shutdown functions include:

| | | | |
|---|---|---|---|
| SLOGON | LASLOGON | WORLOGON | DMSLOGON |
| ULOGON | SLOGOFF | ULOGOFF | LASLOGOFF |
| WORLOGOFF | DMSLOGOFF | | |

To insure the portability of the transformed code, each delivered function was tested on a SUN II and SUN IV under UNIX, a VAX 6000 under VMS, and a Gould Power

Node (PN) 9050 under UNIX at the two developing installations, GSFC and EDC, and at previously selected cooperating, qualifying Beta sites. [Ed. Note: See article in this issue.]

Clearly the work involved in producing this new release was monumental, but the product should prove useful to you, the LAS community!

The names of old user sites and of new ones who had responded positively to the survey sent to sites of record and to sites expressing interest in the new release previously, have been forwarded via the NASA/Goddard Technology Utilization Office (TUO) to COSMIC. They were divided into six categories: NASA sites, (some of whom already have received their copies of the new code), sites under NASA contract, sites formerly under NASA contract, other government agencies, universities related or affiliated, and a general "other" category. The Goddard TUO and COSMIC jointly determine which sites will receive the software without charge and which may have to pay partial fees. Any inquiries about the distribution, or any queries about obtaining copies of the software for new sites should be directed to COSMIC, The University of Georgia, 382 East Broad Street, Athens, Georgia 30602, telephone (404) 542-3265, or Bitnet:COSMIC@UGA.

## File Handling Changes Incorporated into LAS 5.0

*by Lalit Wanchoo (ST Systems Corp. - GSFC)*

The Land Analysis System (LAS) application programs create various files depending upon the purpose of the program. These created files may include:

- An image (IMG) file that contains the resultant image data after processing of the input image with the selected LAS application program
- A history (HIS) file that contains a record of all application programs that were used in creating the image file
- A data descriptor record DDR) file that lists image file data information
- A statistics STAT) file
- A display and lookup table DLTF) file
- A print (PRT) file that has data and data information that can be printed on a line printer
- A weight table (WGT) file that contains resampling weight tables
- A grid (GRD) file that has mapping grid data
- A graphic overlay (GOF) file

Additional files created by Geometric functions are tie point location and selection files, merged tie point files, and projection files.

For each output file of LAS, Version 4.1 had a host file name (DCL) and an LAS file name. These names were different from each other. Users only had to specify the LAS name and the system automatically assigned a host name. In addition, LAS 4.1 permitted multiple versions of a file. But LAS 5.0 has the same file name and does not permit multiple versions of a file. Some of the file naming structure comparisons are as follows:

| File | LAS 4.1 | LAS 5.0 |
|---|---|---|
| IMAGE | | |
| FILE NAME | | |
| Host | 91025Q10.100 | MAY10_A1.IMG |
| LAS | MAY10.GM1 | MAY10_A1;IMG |
| HISTORY | | |
| FILE NAME | | |
| Host | LAB.LAB | MAY10_A1.HIS |
| LAS | LAB.LAB | MAY10_A1;HIS |
| DATA DESCRIPTOR | | |
| FILE NAME | | |
| Host | LAB.LAB | MAY10_A1.DDR |
| LAS | LAB.LAB | MAY10_A1;DDR |
| GROUP | | |
| FILE NAME | | |
| Host | 91025Q10.100 | ----- |
| LAS | MAY10 | ----- |
| | Each band is a file. The Image file has no header record. | All bands are in one file. Image file has a header. |
| STATISTICS | | |
| FILE NAME | | |
| Host | 91025Q10.100 | MAY10.STAT |
| LAS | MAY10.STAT | MAY10;STAT |
| PRINT | | |
| FILE NAME | | |
| Host | MAY10.DAT | MAY10.PRT |
| LAS | MAY10.DAT | MAY10;PRT |
| LOOK-UP TABLE | | |
| FILE NAME | | |
| Host | 91025Q10.100 | MAY10.DLTF |
| LAS | MAY10.LUT | MAY10;DLTF |

When specifying the input file name, one does not have to specify the file name extension such as .IMG, .STAT, .PRT, etc. If the file extension is to be specified, one must specify the extension with a semicolon and not a period. Otherwise, the "file not found" error will terminate the run.

Every time a LAS application program is executed and an output image file is required, the application program will create three output files:

a) file name.IMG as image file;
b) filename.HIS as history; and
c) filename.DDR as data description.

For multiband output, only one image file is created in LAS 5.0 containing all bands of data, whereas LAS 4.1 creates one file for each band. In addition, if the output file already exists, LAS 5.0 programs will not overwrite the file but instead will terminate the program execution with an error message of "file already exists."

Users who desire to use LAS 4.1 files for LAS 5.0 version software will have to convert 4.1 files to 5.0 version format. The LAS 5.0 CONTRIB directory has LAS4TO5 and LAS5TO4 program source code. These two programs enable conversion between the two systems. Both of them have to be executed under the LAS 4.1 environment. Further information on these programs may be obtained by consulting their documentation.

## Implementing X-Windows at GSFC for EDC's DMS Version 3.0

*by Fred Irani, (ST Systems Corp. - GSFC)*

The implementation of X-Windows under DMS has produced a usable but limited display capability under the X-Windows X11R3 protocol for 8-bit graphics work stations. The ability to display a single-band byte image at a graphics terminal, pseudo-color the image, and select polygons and tie points has been tested at GSFC on a SUN 4 /286 workstation under the SUN/UNIX operating system.

Although the X-Window protocol is reputed to be transparent to different operating systems, some problems were encountered under the VMS operating system using an X-Terminal monitor. Users who only wish to display 512 X 512 pixel single-band images for relatively inexpensive "quick-look" purposes will benefit the most from this implementation of display capabilities under DMS. Users who attempt to take the system to the limit of its capabilities will encounter inconsistencies in the operability of the system that are annoying at best and may prevent certain objectives from being obtained. EDC streamlined and improved the vanilla form of the DMS system and released it as Version 3.0. Though this enhanced version was a blessing for most generic implementations, it presented new problems for the X-Windows implementation. For example, its new error handling design has weakened the viability of the DM umbrella program concept. An alternative approach to the current implementation of X-Windows under DMS needs to be taken in the future.

The first implementation of X-Windows by Mike Mazzella at GSFC was performed to meet requirements set by the United Nations Environmental Programme (UNEP). (Note: See LAS Newsletter issue no. 2, July, 1988 and issue no. 4, July, 1989.) The LAS Display Modules (LDMs) were originally designed at GSFC and EDC to be run individually to completion. This presented the problem in the X-Windows environment of maintaining the image window between runs of different display applications.

For example, some mechanism was needed to prevent the image displayed by TODSP from disappearing, along with the X-Window itself, upon the termination of the TODSP program. To allow the minimum modification of existing software all LDMs were implemented as subroutines of a main driving program referred to as the DMS "Umbrella" or simply as the Display Manager (DM). Existing programs were then accessed via the dynamic tutoring capabilities of the TAE. The DMS umbrella was also implemented to drive conventional display devices such as the International Imaging System (IIS) and Comtal display device. The effect here was to increase the speed of image activation and completion. But the umbrella also introduced new drawbacks. Not only did one lose the convenience of interspersing nondisplay LAS commands with LDM operations, he or she also lost the "up arrow" command modification/repeat capability to which many users had grown accustomed under TAE.

Certain weaknesses in TAE that were normally regarded as minor inconveniences became the cause for much extra coding as well. For example, the inability to retrieve TAE Global Variable values from the dynamic tutor required all TAE global values to be retrieved by the main program at start-up time and held for any display command that the user might run. The introduction of the FILEUTIL report printing standard to the LDMs further complicated the use of dynamic tutoring as well as listing to the screen or X-window. Users may still lose their image windows if they quit out of a report listing under the DM umbrella. Also the use of the XABORT flag during retrieval of all TAE parameter values had to be changed to XCONT to prevent program termination if a user exited a dynamic tutor session without running the associated executable image.

Changes made at EDC at the XD level also complicated plans at GSFC to improve on the existing X-Window display capabilities. An example of these changes is the introduction of a generic error handling routine by EDC which was implemented at the XD rather than DD level of DMS. This routine was designed to receive a status code from other DMS routines, report the error to the user and return to the calling routine or terminate depending on the severity of the error. This capability had to be defeated under the DM umbrella to prevent users from losing their images upon receiving a fatal error from any routine.

Most calls to the error handler were made by placing a call to other XD routines as a parameter of the error handler as in:

```
cxderrchk(cxddrop(a, b, c));
```

In this example the status returned by cxddrop is intercepted by cxderrchk, which prints a message and terminates processing in the case of a fatal error. An attempt was made at GSFC to redefine the cxderrchk call as a macro which would print the error message, never terminate, and require a carriage return from the user before continuing processing to allow the user to read the message before the DM command menu is repainted on the screen. This solution was therefore implemented to avoid a redesign and rewrite of every call to any CXD or DD routine in the entire body of LDMs and CXDs. It is not regarded as a reliable solution, however, due to the fact that certain fatal errors may require that the entire application terminate.

If, for example, the client or server is unable to allocate enough memory to contain image buffer space and the program is not allowed to terminate, internal memory is likely to be overwritten. Further processing is likely to result in unexpected results. Users will therefore have to use a certain amount of caution and judgment during long display sessions. In fact, it is recommended that users terminate and restart the DM program themselves if performance degrades after receiving fatal error messages.

The macro substitute for the error handler routine also causes LDM routines that can be run outside of the DM umbrella, such as ALLOC, LSTGOF, and others, to require carriage returns from the user after receiving error messages.

Other changes in EDC's new version of the DMS that may have aided the generic form of DMS unfortunately slowed the X-Windows implementation. The cxddrop routine was originally designed to load as many image lines at a time as allowed by the particular display device being used. The new version of cxddrop depended upon global symbols in the C language to determine how many lines to drop at a time. The X-Windows protocol demands that the entire image to be mapped to a window exist in memory at the time of the mapping. The X-Windows implementation at GSFC intends to allow users to define the size of their X-Window by changing the values of the device characteristics in a run of the ADDDEV DMS Manager Function. The combination of these restrictions required recoding of declaration statements that used C global symbols to set array bounds. Such arrays had to by dynamically allocated using the C symbols as parameters at run time rather than preallocated by the C compiler.

As many traditional LDMs as possible were implemented under X-Window environment. Certain applications were not implemented, however, due to limitations either of X-Windows or due to conceptual generic X-windows work-station capabilities. Most work-stations, for example, do not have a hardware Zoom and Pan capability, and therefore the Zoom/Pan LDM was not implemented. The FLICKR LDM was implemented, but it was found that the X-Windows protocol caused images to be slowly "wiped" over one another rendering the traditional use of FLICKR-IMAGE to be useless. Problems were also encountered in attempting to emulate the "turning off" of a graphics plane once anything was written to it due to the way in which bits are shared between graphics and image planes under X-Windows. Also, the Window Manager on the SUN, could not be persuaded to iconify and/or repaint an image at a window, or allow the window to be moved or resized without losing the image. This may be a quirk of the version of Window Manager being run under X11R3 on our SUN but we have not received information from any beta test sites regarding the operability of the DMS X-Windows with their Window Managers.

The umbrella concept was implemented as the least intrusive method of porting DMS to the X-Windows environment. With 20-20 hindsight, however, a better solution may have been to manipulate X-Windows as shareable windows. It may have been possible to modify the ALLOC program to create an X-Window and continue to run as a background process for the duration of the display session. If the ALLOC routine could pass its window ID number to TODSP, to PSD, and to any other LDM when each is run, then most of the problems encountered in the DM port would have disappeared. An X-Window could have acted similar to a conventional display device, never disappearing until DALLOC is run. Programs could have terminated without dire consequence, and memory management problems would have been isolated within each LDM run.

The design of DMS and the LDMs is most appropriate for the conventional display device. X-Window users who are unfamiliar with LAS and/or DMS may be frustrated initially by the round-about operations they have to make to manipulate their image windows. The design of the LDMs and CXDs would be vastly different if they had originated from an X-Window environment point of view. Much of the information being held in the device tables would no longer be relevant due to the ephemeral quality of the X-Window environment. A heavier reliance on the TAE Facelift and or PLUS capabilities would have increased this system's popularity ten-fold. If, for example, a user could open a window with his or her mouse, click on a pop-up window to see a list of his or her existing image file names, and then cut and paste a file name into the window, title bar, etc., to display the image as opposed to "Tutoring" on some animal called "TODSP", that user might be more inclined to demand support for and dream up new requirements for the system. But this is only speculation on my part. Perhaps future projects might have the luxury of anticipating all possible uses and destinations for their software packages, and then designing their systems to accommodate most easily every one of them!

## *Call for a Volunteer*

*by Philip Pease (GSFC)*

Our plans at Goddard were to have sites that make ports of the LAS to a new UNIX system provide us with changes or information so as to allow LAS to be usable on a wider range of UNIX systems. With the termination of the LAS project here at the Goddard Space Flight Center, not only will we not be providing further revisions to the LAS software, but there will not even be a User Support Office to be a clearing house for identifying sites that may have made ports.

This clearing house feature was only one aspect of the tremendous loss the User Community may experience as a result of the termination of the task and the resulting elimination of the LAS Support office. This realization triggers a need to solicit a volunteer site to fill the void. If one of you has the interest and resources, you might consider serving voluntarily, "free-of-charge," as the new center for LAS support activities. Here are the functions that should be provided, along with my thoughts on setting this up with minimal resources via an anonymous FTP (ANON-FTP) account and an electronic mail (EMAIL) address. This unit should:

1. Serve as a clearing house for problems discovere and possible solutions, such as revised programs and or workarounds. EMAIL should be used to receive problems detected by other sites, and ANON-FTP to access the INFO-Problems file to identify problems that have been previously reported, suggested work-arounds, and/or pointers to updated routines that fix the problem.

2 Serve as a clearing house for contributed functions (new functions that operate within the LAS system environment). Use EMAIL for other sites to send new functions to your support site, and ANON-FTP for other sites to identify and get these contributed functions.

3 Serve as a clearing house for ports to new operating systems (a new service with the LAS Version 5.0 release). Use EMAIL for other sites to let you know about their new port, and ANON-FTP for

other sites to find out what ports are available and where to get them.

4. Provide technical assistance on the installation and use of LAS. Of course, this would take many more resources, so unless you are willing to really commit to being a "full-service" Support Office, you probably would not want to commit to this.

5. Send out newsletters to keep sites updated about new contributed functions, new software fixes, problems and solutions, etc. With the suggested ANON_FTP type of operation, this does not need to be done.

6. Send out new releases of LAS. With the suggested ANON-FTP type of operation, this might also be unnecessary.

The work involved in such an effort might dovetail nicely with, for example, a University environment or a research facility. The shared information that such an effort provides enhances the progress not only for your own site, but for the entire scientific community! If you would be interested in becoming the focal-point of the LAS User Community, contact COSMIC Customer Support, The University of Georgia, 382 East Broad Street, Athens, Georgia 30602, or phone (404) 542-3265, Bitnet: COSMIC@UGA.

## Lessons Learned During Conversion

*by Katherine Donaldson (ST Systems Corp. - GSFC)*

As LAS comes to an end, and one looks back at the effort expended during the conversion process, several thoughts come to mind. Converting a system from one language to another in an attempt to make it portable across different machines is a far more complex process than it may seem at first. This is the case even if the two languages exhibit a similar programming philosophy such as that of being a procedural language. Project scheduling should take into account the effort needed to resolve potential software and/or hardware problems, as well as the effort required to make decisions affecting the new system's design. It should also take into account the time needed to coordinate such decisions with other parties if the conversion is a cooperative effort.

Although this article deals with the conversion of an image processing system from Fortran to C, the issues involved here are general enough to illustrate problems that may arise in converting any other set of programs from one language to another and porting it to a new machine.

The first lesson learned during the conversion of LAS is that regardless of how standard and portable was the Fortran code in the application to be converted, the differences in features inherent in the two languages often created a need for redesign. In this case, the image processing algorithm remained the same, but the implementation of it had to be changed to compensate for Fortran features missing in C or to take advantage of C features not supported by Fortran. Early in the project, a development tool, FOR2C, was designed at GSFC to automate and speed up the conversion process. It translated a Fortran code to a pseudo-C code by converting all text to lower case, changing Fortran comment symbol C to /* and */, flagging new or modified LAS support routines so that the programmer could verify the routine's calling sequence, etc. The differences in the two programming languages, however, made it difficult to completely automate the conversion process; several modifications had to be done by hand.

One place where Fortran and C differ is the implementation of arrays. Fortran arrays are stored in column major form. This means that in multidimensional arrays, the left index varies the fastest. Their indexes can range from any negative to any positive integer supported by a given machine. C arrays are stored in row major form which means that the right index varies the fastest. The array indexes range from 0 to any positive integer supported by the given machine.

Image processing applications often involve several array-based computations. A conversion from the column major array implementation in Fortran to the row major arrangement in C created a need to switch the order of indexes in multidimensional arrays. This type of change can only be made manually and can be very time consuming.

Processing of histograms and look-up tables needed to be recoded because C does not support negative indices. Access of arrays needed to be changed so that it was zero-relative. This included changing formulas that compute the current array index. Having an array index off by 1 can cause problems that take a long time to find. If the index computation is data dependent, the problem may show up with some test images but not others, or affect only a few output image samples. A program like DIFF that compares two images sample by sample is indispensable to assure the same results. There were programs that due to the off-by-one error generated output images that differed from the 4.1 version by only a couple of pixels. In a 256 by 256 four band image, it is nearly impossible to spot this kind of error by visual inspection.

During the conversion, it was also found that the difference in computational precision between Fortran and C (in C all computations are performed in double precision) can have a significant effect on the output of computation intensive programs. This effect was most noticeable when real*4 output images from version 4.1 were compared to real*4 output images from version 5.0. It was the programmers responsibility to verify that the differences in the output were indeed due to the difference in the computational precision and not to some other bug.

More implementation decisions had to be made to handle Fortran common blocks, equivalence relations, or parameter statements. In general, the LAS version 4.1 Fortran 77 programs often abounded in implicitly declared variables that have nondescriptive names, let alone comments about their purpose. Though it is true that good programming style dictates these requirements, the nature of Fortran in accommodating nonexplicitly declared variables tended to encourage these violations. The programming standards for the 5.0 version required among other things that all variables have instructive comments. This forced the programmers to gain a very intimate knowledge of the algorithm in order to be able to give meaningful names and descriptions to the implicitly declared variables.

Converting to C allowed for use of programming features not supported by Fortran, such as dynamic memory allocation, pointers, address arithmetic, etc. The use of these features resulted in a more efficient converted program but often at the cost of additional code and more debugging time. For example, errors in the original program were sometimes discovered only after memory for an array was allocated dynamically. Once the memory is allocated on an "as needed" as opposed to a "maximum needed" basis, one has to be much more careful to avoid writing out of bounds of the allocated space. Since C does not perform run-time bounds checking, the overwriting can go unnoticed and have very serious consequences.

Differences between various machines can also be a source of potential problems. Machine architecture dictated some of the project requirements. Reversal of bytes on the VAX made it necessary to write special programs that ensure correct transfer of data when an image is ftp-ed from a VAX machine to a non-VAX machine. Having a set of standards intended to maximize portability did not guarantee that the same program would work identically on different machines. Compilers, memory management, and run-time performance vary on different systems. Some bugs can cause abnormal process termination on one machine while being virtually undetected on another. The VAX, for example, is very tolerant of overwriting array bounds. The SUN usually aborts processing with an access violation error. The IBM RT, on the other hand, "hangs" for no apparent reason and without any error message. To make things even more complicated, the place in the code at which the IBM RT "hangs" would not even be close to that at which the overwriting took place!

All the conversion caveats mentioned above create challenges and add variety to an otherwise monotonous task. They also make it very difficult to evaluate the project time requirements. Conversion is an ideal time to fix existing bugs and add enhancements, both of which increase the time needed to complete the program. The programmers' experience in the use of the system to be converted as well as their proficiency in the two programming languages also play an important role. New programmers who have image processing experience can be left alone to explore the system on their own. New programmers without image processing experience benefit from a short hands-on training supervised by an experienced user. This way they know not only what to expect from the converted version, but also the reasons for the particular output. This enables them to identify reasons for any abnormal behavior more easily thereby decreasing the time spent on problem fixes.

If any of you ever engage in such a conversion endeavor, please remember these difficulties in your planning. In this way, you may benefit from the lessons learned from this project.

## *Testing of Land Analysis System*

*by Lalit Wanchoo (ST Systems Corp. - GSFC)*

LAS 5.0 consists of a large number of support programs and over 200 application programs. Application programs are the ones that users execute for performing a desired function such as geometric correction, radiometric correction, classification, image statistics, etc. Support programs are called within the application programs for input/output files, creating/updating of history and data descriptor record files, input parameters, etc. Successful execution of an application program with correct results will not only be dependent upon proper implementation of the application program but will also depend on successful operation of all of the support programs that are called within those application programs.

Thus, to insure the integrity of the system, testing is one of the key elements in the development of such a successful, operational software system. The testing procedure adopted should be such that the system to be tested is subject to actual designed usage. This can be achieved only by developing testing procedures based on an actual system application that is implemented by a potential user distinct from the development staff. The testing approach that was developed and implemented for LAS 5.0 incorporated to a great extent actual operational use. A systematic procedure of testing involved the following levels of testing:

**Level A:** Programmer's Testing—A programmer's check list was developed that accounted for essential items that would have insured successful implementation of an application program. This testing was used by the programmer who developed the application program. After successful programmer testing, an application program was delivered for independent functional validation testing.

**Level B:** Functional Testing—A check list as a guide was developed. This checklist included:

- checked function with a single-band image and a multi-band image
- implemented subband and window option (separately and together)
- checked with input images of all data types
- checked with image of 256 bands
- checked with a large image (at least 8096 pixels)
- checked with an image of all zero values
- checked with an image containing negative values
- exercised all parameter options at least once with non-default values
- tested with a combination of input images specified using plus sign
- checked output history file, data descriptor record file, minmax values, and processing messages
- compared output with LAS 4.1 function output
- checked first and second level online help
- checked user guide for accuracy and completeness

The purpose of this check list was to ensure LAS 5.0 version equivalency to the LAS 4.1 version. LAS 4.1 functional problems that were discovered in LAS 4.1 version and reported to the LAS support group had been fixed and/or were fixed during the conversion of LAS 4.1 version to new LAS 5.0 version. Therefore comparing LAS 5.0 program output to that of equivalent LAS 4.1 code assured accurate results and operation. In addition to the checklist items, programs were executed for other possible selections of parameters and their combinations as might be appropriate for the application program. This testing was performed in the following two steps by persons who were actively using LAS 4.1 and who were specifically assigned for this testing.

**Step 1:** Local Testing—Perform detailed functional testing at the application program's developing site. This procedure involved detailed testing that:

a) used all default values
b) changed values as seemed appropriate to the application
c) invoked all subcommands

The National Aeronautics and Space Administration (NASA) at Goddard Space Flight Center (GSFC) and EROS Data Center (EDC) had jointly converted the LAS 4.1 version to a LAS 5.0 version both for the VAX/VMS and for the work-station environment under the UNIX operating system. Conversion of application programs was shared between GSFC and EDC. The facility that performed the initial conversion was referenced as the developing site and the other facility was referenced as the other testing site. After successful, thorough, local testing was performed by the developing site, application programs were delivered to the other testing site for their validation testing.

**Step 2:** Validation testing—Application programs that were developed and fully tested at the developing facility were subjected to scaled-down testing at other site. The reason for doing so was that initial, thorough, local application program testing performed at the development facility had been designed to insure successful functional operation. It was to answer the question, "Were the results accurate and correct as the analyst user would expect?" On the other hand, the purpose of validation testing was to insure that the programs would successfully execute not only on the developing site's systems and machines but also on different systems and machines at the other testing facility. If, in the course of such validation testing, some functional errors were observed and/or some changes were suggested, the developing facility would be notified and the code returned for modification. After the developing facility incorporated the appropriate changes, the application program was once again subjected to both local and validation testing. This testing was implemented by the other site after the developing site had performed local testing. Validation testing involved:

a) using all default values and
b) invoking all subcommands

A special data set was created for the testing of LAS 5.0. A Washington, DC, Landsat-5 Thematic Mapper image of 256 by 256 pixels was the standard test data set. All associated files and statistics that were required were created using that test data set. Other data were obtained or created as required by application programs. These included images with zero and negative values, AVHRR data, MSS data, etc.

Hardware setups used in testing LAS 5.0 at both GSFC and EDC were:

a) VAX 6000-440 (VMS 5.4) at GSFC
b) VAX 11/780 (VMS 4.7) at EDC
c) SUN 4 work-station (UNIX) at GSFC
d) SUN 2 work-station (UNIX) at EDC
e) A Gould Power node 9080 at EDC

In addition, GSFC tested LAS 5.0 with X-WINDOW with an IVAS display device. Once functional testing of all the application programs was successful, the LAS 5.0 version was ready for release.

## *PIXTINT: An Interface to ChapTerrains*

*by Katherine Donaldson (ST Systems Corp. - GSFC)*

Programmers at GSFC successfully integrated the ChapTerrains package of the PIXAR Image Computer with the Transportable Applications Environment (TAE). ChapTerrains is a set of routines allowing users to per form terrain rendering operations on images. The images are expected to have red, green, and blue bands plus a fourth band consisting of elevation data. Through a special PIXAR-supplied interface called TINT, users can generate perspective views, create shadow and visibility maps, and perform a variety of other imaging operations. TINT is a relatively low-level command-based interface that gives users a great deal of flexibility in managing image processing sessions, but at a cost of an increase in session complexity. For the users new to ChapTerrains, the only recourse is hard-copy documentation listing brief descriptions of the available commands and their parameters. A prototype interface PIXTINT was developed at GSFC to facilitate users' interaction with ChapTerrains without requiring them to leave the familiar environment of LAS.

The PIXTINT program used the PIXAR monitor for image display and the SUN 4 workstation for the user interface through TAE. The terrain rendering session was controlled by a menu-driven program written in C that displayed a list of selected utility and ChapTerrains operations such as LOAD, RENDER, SHADOWMAP, TRANSPOSE, ZOOM, EXIT, etc. PIXLOAD, a program developed at EROS Data Center (EDC) and described in the January 1990 Newsletter, was used to load an LAS image into the PIXAR frame buffer memory in order to display it on the screen. Once an image was loaded into the frame buffer, any of the listed ChapTerrains operations could be applied to it. The user was expected to supply parameters needed by these operations through the TAE tutor screens. Then, the parameters specified in TAE were passed to an appropriate command in a shell script file. The commands in that script file corresponded to commands in the ChapTerrains interface TINT. The shell script file was run by PIXTINT and the program then executed the selected operation by calling the appropriate ChapTerrains routine with the user-specified parameters. The result of each operation was immediately displayed on the PIXAR monitor screen.

Since this was a prototype interface, PIXTINT consisted of only an illustrative sample of functions provided by ChapTerrains Images created by ChapTerrains routines could not be saved for future LAS use because, at the time of PIXTINT development, programmers at EDC were working to improve the performance of PIXSAVE, a program designed to save PIXAR images in the LAS format. A decision was made to wait for the improved version of PIXSAVE. Since work on LAS at GSFC will terminate once Version 5.0 is delivered, work on inclusion of PIXSAVE will not be performed here.

In order to demonstrate the program's capabilities, the same operation was sometimes implemented twice to show that PIXTINT could provide different degrees of user control. For example, the location of input and output images could be determined by the program or it could depend on the location coordinates supplied by the user. The program provided the means for encapsulation of several related TINT operations under one PIXTINT command. This illustrated that the degree of flexibility and complexity involved in managing an interactive session could be tailored to the needs and wishes of the users. Through TAE, users were provided with a familiar LAS-like, easy-to-use interface with online help describing the available commands and parameters as well as examples and algorithm descriptions.

In summary, PIXTINT demonstrated the feasibility of providing users with the capability of running the PIXAR display device without leaving the LAS environment and without any prior knowledge of ChapTerrains conventions or commands. It also showed that a similar method could be used to make other PIXAR capabilities, such as creation of fly-bys, real-time, fast Fourier transforms, etc., available to users. Though a description of the PIXAR capabilities was beyond the scope of this article, this should demonstrate the possibility of having a system in which the users would be able to take advantage of the capabilities of both LAS and the software available on the PIXAR while working on one computer with the same images, using one user-friendly interface, and without even being aware of which capabilities belonged originally to LAS and which ones belonged to the PIXAR!

## Notes On the Image I/O Error Messages in LAS Application Software

*by Wei Xia (ST Systems Corp. - GSFC)*

Although LAS programs are well documented, users may still be confused by informational input/output error messages when running LAS application programs.

To illustrate this problem, consider the program PERSPEC. When a user runs this application program to generate a three-dimensional image from elevation and reflectance images, the user may get the error message:

"[imageio-pixel] Pixel values were larger than maximum allowed by output data".

This error message may actually be caused either by the output data type the user specified or by the input elevation data he or she used.

Consider the first possibility. In PERSPEC, the input reflectance image is opened as the same data type of the image file and processed internally as the original file data type. Most other LAS application programs always read and process data as real data, regardless of the input image data type. The output data type for PERSPEC is chosen by the user through the parameter ODTYPE. If ODTYPE=SAME, then the data type of the output image will be the same as the data type of the input image. If ODTYPE=BYTE, I*2, I*4 or R*4, the output image will be opened and processed as either BYTE, I*2, I*4 or R*4 internally. If the pixel values of the output images are larger than the maximum allowed by the specified output data type, the user will get the above image I/O error message. In that case, the program will continue to execute, but the output pixel value will be clipped to the maximum data value allowed by the output data type. For example, if the user specifies ODTYPE=BYTE, then the allowed range for output image pixel values are within 0:255. If the calculated output pixels value is beyond this allowed range, then the program will execute, but in the output file, any calculated output pixel values larger than 255 are clipped to 255 and any calculated output pixel values less then 0 are converted to 0. Meanwhile, the user will get the above error message on the display screen.

A second source of the error message is the elevation data (IN1). According to the design of the program, IN1 is allowed to be any one of the four standard data types. However, if it is not INTEGER*2 before a processing starts, it will be converted to INTEGER*2 without aborting the program. This conversion is done in the image I/O program c_eread.c. The range of INTEGER*2 data varies with different computers. Taking the VAX 6000-4 as an example, the data range for INTEGER*2 is between -32768 to +32767. Therefore, any IN1 data (regardless of the data type) with pixel values beyond this range will cause a data clip and introduce the above error message. For example, if the IN1 is REAL*4 data with pixel values greater than 32768 and/or smaller than -32767, the user can either truncate or round off the pixel values using COPY before running PERSPEC, or rescale/remap the data using SCALE, MAP, or STRETCH as appropriate. Otherwise, PERSPEC will truncate the IN1 data automatically, and report the above error message to the user.

The above discussion is true for many other LAS application programs as well. Knowing the possible causes may help users to have a better understanding of such image I/O messages.

## Technology 2000 Presentation on LAS

*by Yun-Chi Lu and Marilyn Mack (GSFC)*

LAS was invited to speak at the Technology 2000 conference held at The Washington Hilton Hotel in Washington D.C. on November 27-28, 1990. This conference was the first national conference and exhibition of NASA's technology for transfer ever held.

The conference allowed attendees to access technology from NASA and its contractors, meet government/industry tech transfer experts, find new partners for R&D ventures, discover the latest innovations in a variety of high-tech fields, and explore future applications of space based research. There were exhibits by more than 150 companies and research institutions. More than 100 presentations were made by top NASA researchers and industry leaders in areas as diverse as Artificial Intelligence, Computer Technology and Software Engineering, Environmental Science, Human Factors Engineering and Life Sciences, Information and Data Management, Manufacturing and Fabrication Technology, Materials Science, Optics and Communications, Power,Energy, and Control Systems, Robotics, Sensors and Measurement Technology, and Superconductivity!

In the Environmental Technology session, Dr. Lu described LAS as an interactive system for the analysis, display, and management of multispectral and other digital data. He noted that it was available in the public domain and then detailed the comprehensive set of more than 240 applications functions and utilities, the flexible user interface, complete on-line and hard copy documentation, extensive image data-file management, reformatting, and conversion utilities.

The Conference was well attended and will undoubtedly be repeated in the future. Further information on any future conference plans may be obtained from the Technology Utilization Foundation, 41 East 42nd Street, Suite 921, New York, New York 10017.

## Experiences in Porting LAS Version 5.0 to a Silicon Graphics Iris Workstation

*by Fred Irani, (ST Systems Corp. - GSFC)*

In April, 1991 a portion of the LAS Version 5.0 software was ported to a Silicon Graphics Iris (SGI) workstation at the visualization lab at GSFC. The experience is one that may be of benefit to others approaching the same or similar tasks.

The TAE V4.1 was first built on the system using .cf files for the CRAY computer as provided by the TAE V4.1 delivery tape. Only the TAE Classic portion of the package was built and no unusual Problems were encountered either in building or testing the TAE on the SGI. This phase took the equivalent of one morning .

The LAS Version 5.0 build procedures required some modifications to point to new directory locations, but in the majority of cases worked as they were designed to do. None of the display or magnetic tape oriented support or application software was built. The major problem encountered was that so many environmental variables were defined on the system by the time the build procedures were invoked that the IRIS kernel ran out of space to contain them! As a result, environmental variables would not be defined as new makefiles were initiated by the automated build procedures provided in LAS. The effect at the screen was that only a portion of the makefiles ran to completion before error messages began to appear complaining of syntax errors, etc. The temporary solution was to intersperse "unset" commands to "undefine" LAS environmental variables between runs of makefiles. This solution allowed us to build LAS applications, but a need for better management of the creation, deletion, and use of environmental variables during the LAS build was indicated by this experiment.

Other problems surfaced during the building and testing of LAS on the SGI which were caused by the use of the C language #ifdef convention used in the support and some applications software. Certain C include files and modules define C global symbols depending on the operating system being used. In several instances, no general case is defined. As a result, some symbols are never defined at the compile stage and storage for some data arrays may not be allocated. A search was made for '#ifdef" statements in the code and an appendix was added to the LAS V5.0 installation guide to assist users who may be porting the LAS to a new operating system. These files will have to be edited to add new #ifdef statements to define symbols appropriate for the environment under which they will be run.

A minimum of testing was performed on the SGI computer but basic functions such as TESTGEN, COPY, and other utilities were run to establish that the system appeared to be running generally well. A full implementation of LAS V5.0 on an SGI may be performed by the end of September at GSFC.

Though much time has been spent in updating manuals to reflect difficulties when they are discovered, this experience shows that care must always be taken when porting to a new environment.

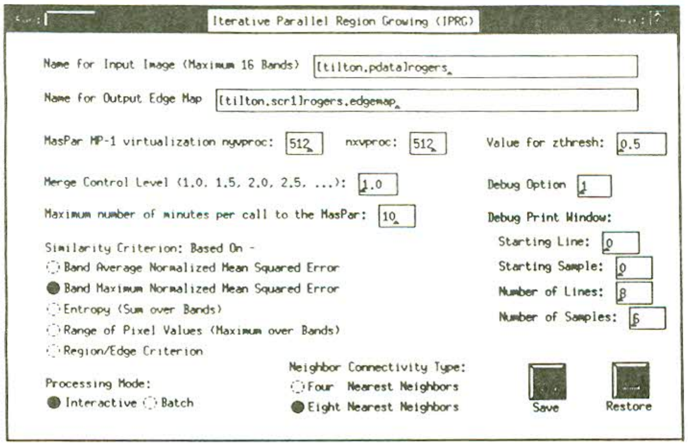
## Image Segmentation and Data Compression Algorithms Developed with the TAE/LAS Interface

*by James C. Tilton (GSFC)*

Image segmentation and data compression algorithms are currently under development in the Information Systems Development Facility (Code 936) at the NASA Goddard Space Flight Center. Many of these algorithms are implemented under TAE Classic and TAE Plus using LAS data formats. This allows the exploitation of the extensive LAS algorithm library for the more conventional image processing and analysis steps that are sometimes required when developing new algorithms.

Both TAE Classic and TAE Plus interfaces have been developed for some of the algorithms. The TAE Classic interface gives compatibility with LAS, and allows the algorithms to be run from a non-graphical terminal. The TAE Plus interface gives a convenient graphical interface to the program and calls the TAE Classic Procedure when actually executing the program.

The Iterative Parallel Region Growing (IPRG) program is an example of one of these algorithms being developed under TAE Classic and TAE Plus using LAS data formats. Under TAE Plus, the user enters parameters through the following interface panel:




Other panels give status reports on the program as it executes, and when the program finishes, program statistics are summarized in another display panel. The interactive features of TAE Plus allow the user to tell the program to terminate after the current iteration saving the results to that point.

The TAE Classic interface is similar to most LAS functions. Once the program is started under TAE Classic, the user can abort the program, but the program will not store the results to that point.

The IPRG program produces a hierarchical segmentation of multispectral image data. This means that it produces segmentations at several levels of refinement, with the segmentation from iteration x + 1 being coarser than the segmentation produced at iteration x. Since the segmentation produced at a particular iteration is generally not appropriate for all areas of the image, one must be able to select different iterations from the segmentation hierarchy at different locations in the image. An interactive tool, "Hierarchical Segmentation Exploration" (HSEGEXP), is currently under development to facilitate making this variable selection from the segmentation hierarchy. Together, IPRG and HSEGEXP are a powerful means of producing computer-aided image segmentations.

Due to its computationally intensive nature, the IPRG program is implemented on the MasPar MP-1. The MasPar MP-1 is a massively parallel computer with 8192 4-bit processors connected in a SIMD (Single Instruction, Multiple Data stream) mesh architecture. The MasPar MP-1 is currently accessed through a VAXStation running ULTRIX. Note that in the near future the MasPar "host" machine will be upgraded to a DECStation. The interactive TAE part of the program is run from the host machine, and special MasPar supplied functions are used to transfer the parameters and data to the MasPar, and to retrieve results from the MasPar.

For more information about IPRG and HSEGEXP, the methods used to implement these algorithms under TAE Classic and TAE Plus, and the MasPar MP-1, please contact Dr. James C. Tilton at (301) 286-9510, "tilton@chrpisis.gsfc.nasa.gov" on Internet, or "AMARNA::TILTON" on SPAN.

## *LAS Support of Scientific Activities at PGD/SOEST, University of Hawaii*

*by Harold Garbeil and Peter Mouginis-Mark (University of Hawaii)*

*[Ed. Note: This article was written almost a year ago, but the delay in publishing this newsletter has inadvertently dated the material in it. The University has undoubtedly continued their research far beyond what is reflected in this article.]*

LAS is being utilized to support a variety of geological research activities within the Planetary Geosciences Division (PSD) of the School of Ocean and Earth Science and Technology (SOEST), University of Hawaii. PGD is strongly involved in an interdisciplinary volcanology investigation as part of NASA's Earth Observing System (EOS) program, with the Team Leader and several of the co-investigators being based at the facility. Other mission-related activities include participation in the shuttle Imaging Radar (SIR-C) experiment, processing of the Near-Infrared Mapping Spectrometer (NIMS) data to be obtained by the Galileo spacecraft, and basic geological research utilizing the Viking and Voyager data sets of Mars and the moons of the outer planets. PGD consists of 14 faculty, 16 graduate students, and about 20 support personnel. The School is a Sea Grant and Space Grant College, which serves to expose undergraduate and graduate students to various aspects of satellite remote sensing of the Earth and planets.

LAS 5.0 is invoked on a SUN 4 with an IVAS display. LAS 4.0/4.1 is currently operational on a VAX 11/750 with an IVAS display. In addition to this equipment, the facility has several other CPU's, peripherals, and software packages such as GRASS, OV~WAV E, and PICS to assist in the support of the research.

LAS is expected to play a major role in the PGD EOS volcanology investigation. It is planned that at least 200 volcanoes around the world will be imaged on a regular basis in order to investigate eruption processes and to detect potential new eruptions. Data sets that will be collected from EOS include high resolution imaging spectrometer and radar over selected sites, and almost global low resolution visible and thermal infrared data. The LAS package will be used for general image processing tasks as well as providing an environment for algorithm development.

Also, SOEST has recently acquired a receiving station for the collection of Advanced Very High Resolution Radiometer (AVHRR) data. These data will be used for the development of algorithms to detect volcanic eruption clouds, and to determine the temperature and distribution of hot spots associated with Kilauea Volcano on the island of Hawaii. Algorithms will also be developed to discriminate between water vapor clouds of volcanic origin. The AVHRR algorithms developed with LAS will then be carried over to the interpretation of the Moderate Resolution Imaging Spectrometer on EOS, which will have a similar spatial resolution but superior radiometric information.

For additional information on programs at PGD/SOEST, please contact Harold Gabreil or Peter Mouginis-Mark, at the University of Hawaii at Manoa, Planetary

Geosciences Division, Hawaii Institute of Geophysics, 2525 Correa Road, Honolulu, Hawaii 96822 or call Harold Garbeil at (808) 948-6326.

## Utilization of LAS 4.1 at The University of Texas at Austin

*by Nahid Khazenie and Melba Crawford (The University of Texas at Austin)*

The University of Texas has continued to use the VMS version of LAS for reading MSS and TM tapes and for performing standard image processing operations on data currently being used for geological projects. Typically this involves destriping, band ratioing, histogram modification, application of directional filters, and sometimes use of edge detection routines. The software contains all the subroutines required for standard analysis and is relatively robust in handling even old MSS data tapes. The difficulties in dealing with the catalog manager still persist, so we are anticipating the release of the UNIX version that we are told does not suffer from this problem. We also do not have a display device that is supported, so we are using other software packages to display the images and the processed data. This is so frustrating that we have actually taken the code for several subroutines and incorporated them into other software systems that we are using. We hope that the UNIX implementation in X-Windows also solves this portability of display problem.

In addition, we tried to use LAS in a teaching environment this spring. Students acquired TM data measured over Chernobyl, and attempted to use LAS to read the tapes and classify the images. The students were experienced in the use of the computer system and all had Fortran programming experience. They had difficulty in learning to use the LAS system, although they were eventually successful in using the software. Currently, we find LAS to be a software package with a great deal of potential, but one that is lacking the final link to make it a useful tool in operational environments. Hopefully, LAS 5.0 may provide that tool.

## LAS Implementation Table

This is the latest list of machines, operating systems, and display devices that are known to be compatible with LAS and on which LAS was successfully running either partially or completely.

| LAS Version | Computer (Operating Systems) | Display Devices (Software Version) |
|---|---|---|
| 5.0 | VAX/VMS | IIS IVAS |
| | Gould PN 9050 | X-Windows |
| | SUN II | |
| | SUN IV | |
| | ** IBM RT (AIX) | |
| | !! COMPAQ DESKPRO 386 (XENIX) | |
| | !! Cray II (UNICOS) | |
| 4.1 | VAX/VMS | Rastertek One/25 |
| | | IIS Model 75 (575) |
| | | DeAnza IP8500 |
| | | Comtal Vision/120 |

** means previously in development for 5.0 (only a subset was implemented)

!! means that only one or two routines were converted to illustrate system compatibility, but no further plans were ever made to port the system to the system and machine.

## LAS "Contrib" Contents

The LAS project has provided a mechanism for user sites to contribute functions, which they have implemented for use with the LAS, and which may then be made available to other sites.

These contributed functions are provided on an "as is" or "caveat emptor" basis. No guarantees were or are made. The difference between previous CONTRIB libraries and this is that in the past, this software was only available from GSFC. Since the project is being terminated, all CONTRIB libraries will be distributed with the system.

The library structure has also changed since the last newsletter. Its final form, found in CONTRIB.SOURCE, will contain six directories:

**APP** - Contributed functions that are designed to be used under the LAS Version 5.0 environment.

The functions in this directory include:

- EDIPSOUT - Creates an output tape in EDIPS 0.0 format from an input LAS image
- FSTFMTIN - Reads EOSAT fast format tape(s)
- GRIDMODEL - Performs GIS Modelling using the raster-based GRMDL software (VMS only).
- KERNL - Generates a resampling weight table file
- LTWGIN - Reads EOSAT LTWG full-scene formatted tapes
- PDFUSR - Creates a User's Guide from the pdf file help information
- SARTOPO - Calculates radar viewing angles and range vectors
- TDPLOT - Creates Versatec plots of training site statistics in 3-D space

**V4_APP** - New contributed functions that must be used under the LAS Version 4.1 environment (i.e., a user must be under the LAS 4.1 environment when running these).

The functions in this directory include:

- LAS4TO5 - Converts the LAS 4.1 image into an LAS 5.0 image
- LAS5TO4 - Converts one LAS 5.0 image into an LAS 4.1 image

**V4_CONTRIB** - The functions currently available from this directory are from the old LAS Contributed Library and can only bE used with LAS version 4.1:

- CCTEDCN - Obsolete tape format ingest
- CCTEDCS - Obsolete tape format ingest (LGSOWG MSS tape)
- CCTF - Output a tape in the format used by Optronics or Dicomed film recorder
- CCTL4 - Obsolete tape format ingest
- CCTTIPSP(w) - Window selection from a TIPSP tape
- COMPRSLAB - Compression of a label file in batch mode; the interactive capability is in the program CATMAN
- DALIN - DAL format tape ingest that was originally in the 4.0 baseline release
- DALOUT - DAL format tape output that was originally in the 4.0 baseline release
- DSPLAB - Display of the label file in batch mode; the interactive capability is in the program CATMAN
- DSPLOG - Version of DSPLOG that allows output to terminal, file, or line printer (LP)
- INFONEWUSR - A brief introduction to LAS for new users
- LUTS - Use of DMS look-up tables in LAS functions that include support routines, EDITLUT to edit/create a LUT, and revised versions of CLASSMAP, COPY, FILM, and STRETCH programs
- MAX - Creates an output image composed of the maximum values from an overlay of input images
- MIN - Creates an output image composed of the minimum values from an overlay of input images
- TAPEFEET - Calculation of the number of feet of tape that will be required to store an image onto tape using archive
- WINDOW - A DMS function to interactive select and display of a window of a subsampled scene on another display memory in full resolution

**MGR** - All of these are system manager functions. Some were available for the LAS site manager in LAS Version 4.1 but have not been validated to work under LAS Version 5.0.

The functions in this directory are:

- LASFUNC - Summarizes LAS usage by function
- LASUSER - Summarizes LAS usage by user
- LOGNAME - Checks for a LOGICAL NAME definition

**SUPPORT** - This is a support subroutine package used by some of the tape input/output functions in the TO_CONV applications. They are provided in case the LAS 5.0 tapeio support does not include the needed subroutine.

**TO_CONV** - These are applications that have NOT been converted to work under the LAS 5.0 environment. They are provided for those persons who need them under the LAS 5.0 environment and who wish to make the effort to convert them. Note that some of these functions were in the LAS 4.1 baseline, but were not selected as crucial enough to be converted.

This directory is further divided into seven subdirectories. The first subdirectory, **CORE**, includes:

- BLDPW - Area filtering routine that builds a pairwise weight table
- BLDTHR - Area Filtering Routine that builds a threshold table
- CLEANUP - Cleans up (deletes) user's TAE files created today
- CONTROL - Generates four point pairs and grid file to correct MSS image distortion
- DEBLUR - Corrects linear motion blur of an image
- DSPLOG - Prints or displays the Session Log
- FFT1PIX - Creates a disk-image containing amp and phase information for one-dimensional Fourier transform
- FFT2PIX - Creates disk-image containing amp and phase information for two-dimensional Fourier transform
- FILMTAPE - Transfers image data to tape for use by Dicomed or Optronix film recorders
- FLAG - Area Filtering Routine—Multi threshold area filter
- HRPTIN - Input an AVHRR tape in HRPT format as a LAS image
- MAGPHASE - Separates the magnitude and phase components of a Fourier image
- REGLABEL - Raster to Vector - Region labeling
- UNIVENTR - Read BYTE data in Universal tape format

The second subdirectory, **ANC_IMSL** , includes:

- FFT1 - Performs a one-dimensional Fourier transform
- FFT1FL - Performs a one-dimensional Fourier transform filter
- FFT2 - Performs a two-dimensional Fourier transform
- FFT2FL - Performs a two-dimensional Fourier transform filter
- IFFT2 - Perform an inverse two-dimensional Fourier transform
- MAGPOWER - Raises the magnitude of each pixel in a frequency-domain image to a given power
- SMOOTH - An Area Filtering Routine that smooths subthreshold regions
- WTGEN - Generates a matrix of filter weights

The third subdirectory, **ANC_PRINTRONIX**, contains code that uses a Printronix printer. It includes:

- TPLOT - Generates a line printer plot of an image and calculates the percentage of samples within defined classes

The fourth subdirectory, **ANC_VERSATEC**, contains code that uses a Versatec printer/plotter. It includes:

- DSPVERSA - Display an image on the Versatec printer

The fifth subdirectory, **CONTRIB**, contains codes that were contrib functions in the LAS 4.1 release, plus some additional newer ones. It includes:

- BOXCAR - Moving average filter (FLT8B)
- CALC - Calculator program
- CCTSPOT - Processes SPOT data on tape
- CCTTIPSA_WINDOW - Reads Landsat 4 and 5 TM data contained on TIPS-A format tapes and allows selecting subarea of the data
- IMGCALC - Performs CALCULATIONS on the input images
- LNFILL - Replace lines of data by averaging neighboring lines
- LOWCAL_FPS - See LAS 5.0 version for description, 5.0 version not for FPS device
- MINC - Perform grid interpolation from random data using the biharmonic equation
- SCHEDULE - Schedule tracking and reporting package
- TMSENTER - Ingests NS-001 Thematic Mapper Simulator imagery from tape

The programs in the following two directories use a Floating Point Systems, Inc. array processor device. In 4.1 there existed two version of these codes: one which simulated FPS The sixth subdirectory, **ANC_FPS**, contains the following code:

- BAYES - See LAS 5.0 version for description, 5.0 version not for FPS device
- BDIST - See LAS 5.0 version for description, 5.0 version not for FPS device
- CANAL - See LAS 5.0 version for description, 5.0 version not for FPS device
- CONVOLVE - See LAS 5.0 version for description, 5.0 version not for FPS device
- COVAR - See LAS 5.0 version for description, 5.0 version not for FPS device
- DISCRIM - See LAS 5.0 version for description, 5.0 version not for FPS device
- DIVERGE - See LAS 5.0 version for description, 5.0 version not for FPS device
- FFT1 - Performs a one-dimensional Fourier transform (FPS version)
- FFT1FL - Performs a one-dimensional Fourier transform filter (FPS version)
- FFT1PIX - Creates disk-image containing amp & phase information for one-dimensional Fourier transform (FPS version)
- FFT2 - Performs a two-dimensional Fourier transform (FPS version)
- FFT2FL - Performs a two-dimensional Fourier transform filter (FPS version)
- FFT2PIX - Creates disk-image containing amp & phase information for two-dimensional Fourier transform (FPS version)
- GRAD - See LAS 5.0 version for description, 5.0 version not for FPS device
- IFFT2 - Perform an inverse two-dimensional Fourier transform (FPS version)
- ISOCLASS - See LAS 5.0 version for description, 5.0 version not for FPS device
- KARLOV - See LAS 5.0 version for description, 5.0 version not for FPS device
- KMEANS - See LAS 5.0 version for description, 5.0 version not for FPS device
- MAGPOWER - Raises the magnitude of each pixel in a frequency-domain image to a given power (FPS version)
- MINDIST - See LAS 5.0 version for description, 5.0 version not for FPS device
- UHIST - See LAS 5.0 version for description, 5.0 version not for FPS device

The functions in the seventh subdirectory, **ANC_FPS_IMSL**, not only used the FPS array processor, but they also used routines from the IMSL subroutine library.

| | | |
|---|---|---|
| EDGECORR | - | Function CORRELATE replaces it, but CORRELATE does not use FPS device |
| GETSAMP | - | See LAS 5.0 version for description, 5.0 version not for FPS device |
| GREYCORR | - | Function CORRELATE replaces it, but CORRELATE does not use FPS device |

It is hoped that making these functions available to the user community may facilitate development and research at active sites.

## LAS Problems and Solutions

Since the last issue of the newsletter, several problems were reported. In addition, some problems in Version 5.0 of a few of the functions were discovered, but could not be fixed prior to our final delivery deadline. Since no one at Goddard will be working on this project in the future, and since no time nor funds existed to examine them in detail, they will continue to be unresolved problems until and unless one of you solves them. So that you will be prepared for them, these are the problems that were reported:

**General System Problems:**

While not actually a problem, you should be aware that when specifying a file type for the output that is smaller than the input data (using the ODTYPE parameter), the output will be truncated and not scaled. If you want to produce a scaled output, leave the ODTYPE as same and then run the output file through scaling programs such as MAP or SCALE.

Installers of LAS Version 5.0 under UNIX **MUST** edit the file lassys/las_start to change RUNTM to runtm. This change is needed to start up the TAE when the alias las or las 5.0 is used.

Many programs do not check for sufficient disk space to create the outputs and report this condition to the user. They may give the error message "fatal error opening file" without explanation of what the error is. Check quota or free disk space, also check the file name (Note: Some UNIX systems may be so restricted that default file names may be too large. This is especially likely for print files.)

If a DLTF contains 256 LUTs, the 256th LUT can not be read (limit the number of LUTs to 255).

| | | |
|---|---|---|
| FILEUTIL | - | On a system that has no printer the program does not give any error even when a printer output is selected. |

**Applications Problems:**

| | | |
|---|---|---|
| CCTTIPSP | - | The DDR has incomplete time of acquisition in individual band information. In the history file the program name is tipsp, not ccttipsp-nowin with the -nowin subcommand; shows parameter window—but there is no window parameter for the -nowin subcommand; it also lists prtfile=tipsp_timecode.pr. |
| CMCOPY | - | When in=name;ext and out=newname (with no ;ext) program gives abnormal process termination. |
| CMDEL | - | Tape delete sub-function is not a valid operation since the related cmstor, cmrestor functions, that handle cm tapes are not available in this release. It is therefore suggested that you always set the parameter for ONLINE). CMDEL-DIR does not work on the SUN. Instead, use the rmdir UNIX shell command. |
| CMLIST | - | Setting NLEVELS to more than one gives abnormal termination. |
| CMSET | - | Does not work on UNIX systems; gives an error that it cannot find proc chdir. cd to desired directory before entering LAS/TAE). |
| DIFF | - | If the user specifies a file name that already exists for the print parameter, then the program continues to process and gives a successful completion message with output file name as blank. Fileutil then aborts saying file already exists! |
| DM | - | After doing a help * the menu screen is not redisplayed when exiting from the help. You can give a nonsense command to recover the menu screen. When using the PSD-MAN function under DM, the menu option includes ZOOM & PAN, but this option requires function buttons that are not implemented. Thus the function will hang. The only recovery is to use <CTRL> C and abort. Users should not run this option. |
| DSPGLOB | - | Display is not paged , so be ready to stop scrolling when selecting the TAE or ALL globals. |
| EDIPSIN | - | This does not put correct information into the "capture time" field of the DDR. |

| | | |
|---|---|---|
| FILTER | - | For subcommand -MED the processing message gives sample number instead of line number for the band. |
| LIST | - | If a user requested bands to be listed that do not exist, it lists zero values. The program should give an error "no such band". |
| LT | - | With PRINT=LP an error message "[fileutil-length] Text line is too long. Line is truncated to 80 characters." is given. The output is all right however. |
| MASKSTAT | - | May get access violation when compiled with optimize (Note: Although this problem was reported, we were unable to reproduce this problem. If the problem occurs, try rebuilding without optimize.) |
| MATH | - | In LAS 4.1, MAX and MIN provided a mode parameter which could be set to either single or multi. The multi mode is not available in the 5.0 MATH program. |
| PROCHECK | - | This gives an abnormal termination for misspelling the input pdf file name (i.e., users are told that the file doesn't exist). |
| SEARCH | - | A report of getting an access violation was received; but we can't reproduce this error. |
| TAPEIN | - | On VMS, this tries to dismount a tape on all nodes in a VAX cluster. |
| TAPEOUT | - | On VMS, this tries to dismount tape on all nodes in a VAX cluster. |

**Support Subroutine Package Problems:**

| | | |
|---|---|---|
| ANSI | - | This package is not covered in the LAS Programmer's Documentation. |
| PIXMAN | - | Routine C_PXL10.C does not return correct results. |
| TM | - | This package is not covered in the LAS Programmer's Documentation. |

If you discover a new problem, or if you want to determine the status of any problem fixes, you should contact your LAS Site Manager (identified with INFO-WHO).

## LAS "Bits"

Goddard Space Flight Center (GSFC) obtained a PIXAR machine and will be working to implement LAS software on it. Development will be expedited by the related work done at EROS Data Center.

EDC has begun work on DMS dd's for the Rastertek and DeAnza IP 8500 for their own internal use. They were not available for our inclusion in the 5.0 release, but should be completed sometime soon.

If you are reading "LAS News" but are not yet an LAS user and would like to be, you can purchase the package through the Computer Software Management and Information Center (COSMIC), by writing to COSMIC, University of Georgia, 382 East Broad St., Athens, Georgia 30602 U.S.A., or by calling (404) 542-3265 for further information.

## Calendar of Activities

| | | |
|---|---|---|
| Early Fall | '91 | Paper submission deadline for ISY Conference on Earth and Space Science Information Systems (ESSIS) [Sponsor: NASA Office of Space Science and Applications] at Pasadena Convention Center, Pasadena, CA. Contact Dr. George H. Ludwig, NASA, Code EEU, FOB 10 Room 225B, Washington, D.C. 20546 or phone (202) 479-0360 or FAX (202) 755-2552, or EMAIL GLUDWIG (NASAMAIL). |
| Sep 16 | '91 | Paper submission deadline for Sixth International Parallel Processing Symposium (IPPS '92) [Sponsor: IEEE Computer Society in Cooperation with ACM Sigarch] at Beverly Hilton, Beverly Hills, CA. Contact Dr. Viktor Prasanna, IPPS '92 Program Chair, Dept. of Electrical Engineering-Systems, EEB 244, University of Southern California, Los Angeles, CA 90089-2562 or EMAIL at ipps92@ashoka.usc.edu. |
| Sep 30 | '91 | Abstract submission deadline for Pacific Ocean Remote Sensing Conference (First-PORSEC in International Space Year Conference) at Okinawa, Japan. Contact PORSEC Secretariat, P.O. Box 10, Shimizu, Shizuoka, 424, Japan. Phone 0543-34-0411 (ext. 3422), or or Telemail Y.Sugimori/omnet or FAX 81-543-35-4155. |
| Oct 16-18 | '91 | 20th AIPR Workshop on Applied Imagery Pattern Recognition [Sponsor: AIPR Committee, the Air Force Rome Laboratory, and SPIE - The International Society for Optical Engineering] at McLean, Virginia. Contact: Joan Lurie, AIPR Program Chair, 02/1338, TRW, One Space Park, Redondo Beach, CA 90278, phone (213) 814-8690. |
| Oct 22-25 | '91 | Visualization 91 [Sponsor: IEEE Computer Soc. Technical Committee on Computer Graphics] at SanDiego, CA. Contact: Bruce Brown, Oracle, 500 Oracle Pkwy., MD 40P12, Redwood Shores, CA 94065, phone (415) 726-0983, fax (415) 506-7200. |

| | | | |
|---|---|---|---|
| Oct | 31 | '91 | Extended abstract submission deadline for 11th IAPR International Conference of Pattern Recognition (11th ICPR) [Sponsor: International Association for Pattern Recognition] at The Hague, The Netherlands. Contact: 11th ICPR Secretariat, Delft University of Technology, Dept. Electrical Engineering 11:03, P.O. Box 5031, NL - 2600 GA Delft, The Netherlands. |
| Nov | 5-7 | '91 | Ninth TAE User's Conference [Sponsor: TAE Support Office, NASA Goddard Space Flight Center] at Sheraton Greenbelt Hotel, New Carrollton, MD. Contact TAE User Support Office, Goddard Space Flight Center, Mail Code 522, Greenbelt, MD 20771 or phone (301) 286-6034, or FAX (301) 286-4627. |
| Nov | 15 | '91 | Summary and Abstract submission deadline for International Geoscience and Remote Sensing Society Symposium (IGARSS '92) [Sponsor: IEEE Geoscience and Remote Sensing Society and URSI] at South Shore resort and Conference Center, Clear Lake (Houston), TX Contact: A.J. Blanchard, Space Technology and Research Center, Houston Advanced Research Center, 4802 Research Forest Drive, The Woodlands, TX 77381 or phone (713) 363-7922 or FAX (713) 363-7923. |
| Nov | 18-21 | '91 | IJCNN'91 Singapore, International Joint Conference on Neural Networks [Sponsor: IEEE Neural Networks Council and Int'l Neural Networks Society] at Singapore, China. Contact: Ms. Nomi Feldman, IJCNN'91 Singapore, Meeting Management, 5665 Oberlin Dr., Suite 110, San Diego, CA 92121, fax (619) 535-3880. |
| Nov | 30 | '91 | Abstract of paper due for XVII Congress ISPRS [Sponsor: ISPRS (International Society for Photogrammetry and Remote Sensing] at Washington, DC. Submit abstracts to: Lawrence W. Fritz, ISPRS Congress Director, GE Aerospace, P.O. Box 8048-10A26, Philadelphia, PA 19101, fax (215) 899-3296. Congress Contact: XVII ISPRS Congress Secretariat, P.O. Box 7147, Reston, VA. 22091, phone (703) 648-5585. |
| Dec | 1-5 | '91 | Third International Symposium on Parallel and Distributed Processing [Sponsor: IEEE Computer Society] at Dallas, TX. Contact: Behrooz Shirazi, Univ. of Texas at Arlington, Computer Science Eng. Dept., Box 19015, Arlington, TX 76019-0015, phone (817) 273-3605, fax (817) 273-2548, email shirazi@evax.utarl.edu. |
| Dec | 13 | '91 | Paper acceptance notification date for Sixth International Parallel Processing Symposium (IPPS '92) [Sponsor: IEEE Computer Society in Cooperation with ACM Sigarch] at Beverly Hilton, Beverly Hills, CA. Contact Dr. Viktor Prasanna, IPPS '92 Program Chair, Dept. of Electrical Engineering-Systems, EEB 244, University of Southern California, Los Angeles, CA 90089-2562 or EMAIL at ipps92@ashoka.usc.edu. |
| Jan | 13 | '92 | SPIE/IS&T Symposium on Electronic Imaging: Science & Technology [Sponsors: The Society for Imaging Science and Technology (IS & T), and The International Society of Optical Engineers (SPIE)] at San Jose Convention Center, San Jose, CA. Contact: SPIE/IS&T Electronic Imaging: Science and Technology '92, SPIE, P.O. Box 10, Bellingham, WA 98227-0010, or phone (206) 676-3290, or FAX (206) 647-1445. |
| Jan | 17 | '92 | Camera ready paper deadline for Sixth International Parallel Processing Symposium (IPPS '92) [Sponsor: IEEE Computer Society in Cooperation with ACM Sigarch] at Beverly Hilton, Beverly Hills, CA. Contact Dr. Viktor Prasanna, IPPS '92 Program Chair, Dept. of Electrical Engineering-Systems, EEB 244, University of Southern California, Los Angeles, CA 90089-2562 or EMAIL at ipps92@ashoka.usc.edu. |
| Feb | 10-13 | '92 | ISY Conference on Earth and Space Science Information Systems (ESSIS) [Sponsor: NASA Office of Space Science and Applications] at Pasadena Convention Center, Pasadena, CA. Contact Dr. George H. Ludwig, NASA, Code EEU, FOB 10 Room 225B, Washington, DC 20546 or phone (202) 479-0360 or FAX (202) 755-2552, or EMAIL GLUDWIG (NASAMAIL). |
| Feb | 28 | '92 | Final Paper due date for International Geoscience and Remote Sensing Society Symposium (IGARSS '92) [Sponsor: IEEE Geoscience and Remote Sensing Society and URSI] at South Shore resort and Conference Center, Clear Lake (Houston), TX Contact: A.J. Blanchard, HARC, 4802 Research Forest Drive, The Woodlands, TX 77381 |
| Mar | | '92 | Notification of acceptance of paper date for 11th IAPR International Conference of Pattern Recognition (11th ICPR) [Sponsor: International Association for Pattern Recognition] at The Hague, The Netherlands. Contact: 11th ICPR Secretariat, Delft University of Technology, Dept. Electrical Engineering 11:03, P.O. Box 5031, NL - 2600 GA Delft, The Netherlands. |
| Mar | 4 | '92 | Final paper due date for International Geoscience and Remote Sensing Society Symposium (IGARSS '92) [Sponsor: IEEE Geoscience and Remote Sensing Society and URSI] at South Shore resort and Conference Center, Clear Lake (Houston), TX Contact: A.J. Blanchard, Space Technology and Research Center, Houston Advanced Research Center, 4802 Research Forest Drive, The Woodlands, TX 77381 or phone (713) 363-7922 or FAX (713) 363-7923. |
| Mar | 23-26 | '92 | Sixth International Parallel Processing Symposium (IPPS '92) [Sponsor: IEEE Computer Society in Cooperation with ACM Sigarch] at Beverly Hilton, Beverly Hills, CA. Contact Dr. Viktor Prasanna, IPPS '92 Program Chair, Dept. of Electrical Engineering-Systems, EEB 244, University of Southern California, Los Angeles, CA 90089-2562 or EMAIL at ipps92@ashoka.usc.edu. |
| Mar | 24-27 | '92 | 1992 Data Compression Conference (DCC '92) [Sponsor: IEEE Computer Society] at Snowbird, Utah. Contact Dr. James Storer, Computer Science |

Dept., Brandeis University, Waltham, MA 02254 or EMAIL at Storer@chaos.cs.brandeis.edu.

| | | | |
|---|---|---|---|
| Apr | | '92 | Full paper camera ready deadline for 11th IAPR International Conference of Pattern Recognition (11th ICPR) [Sponsor: International Association for Pattern Recognition] at The Hague, The Netherlands. Contact: 11th ICPR Secretariat, Delft University of Technology, Dept. Electrical Engineering 11:03, P.O. Box 5031, NL - 2600 GA Delft, The Netherlands. |
| May | 15 | '92 | Final deadline for manuscripts due for XVII Congress ISPRS [Sponsor: ISPRS (International Society for Photogrammetry and Remote Sensing] at Washington, D.C. Submit abstracts to: Lawrence W. Fritz, ISPRS Congress Director, GE Aerospace, P.O. Box 8048-10A26, Philadelphia, PA 19101, fax (215) 899-3296. Congress Contact: XVII ISPRS Congress Secretariat, P.O. Box 7147, Reston, VA. |
| May | 26-29 | '92 | International Geoscience and Remote Sensing Society Symposium (IGARSS '92) [Sponsor: IEEE Geoscience and Remote Sensing Society and URSI] at South Shore resort and Conference Center, Clear Lake (Houston), TX Contact: A.J. Blanchard,Space Technology and Research Center, Houston Advanced Research Center, 4802 Research Forest Drive, The Woodlands, TX 77381 or phone (713) 363-7922 or FAX (713) 363-7923. |
| Aug | 2-14 | '92 | XVII Congress ISPRS [Sponsor: ISPRS (International Society for Photogrammetry and Remote Sensing)] at Washington, DC. Contact: XVII ISPRS Congress Secretariat, P.O. Box 7147, Reston, VA. |
| Aug | 25-30 | '92 | Pacific Ocean Remote Sensing Conference (First-PORSEC in International Space Year Conference) at Okinawa, Japan. Contact PORSEC Secretariat, P.O. Box 10, Shimizu, Shizuoka, 424, Japan. Phone 0543-34-0411 (ext. 3422), or or Telemail Y. Sugimori/omnet or FAX 81-543-35-4155. |
| Sep | 3 | '92 | 11th IAPR International Conference of Pattern Recognition (11th ICPR) [Sponsor: International Association for Pattern Recognition] at The Hague, The Netherlands. Contact: 11th ICPR Secretariat, Delft University of Technology, Dept. Electrical Engineering 11:03, P.O. Box 5031, NL - 2600 GA Delft, The Netherlands. |
| Jan | 15 | '93 | Abstract Submission Deadline for International Geoscience and Remote Sensing Society Symposium (IGARSS'93) [Sponsor: IEEE Geoscience and Remote Sensing Society] at Kogakiun University, Tokyo, Japan Contact: Prof. Mikios Takagi, Institute of Industrial Science, University of Tokyo, 15-17, Rappongi 7-Chome, Minatoku, Tokyo 106, Japan, phone - 03-3402-6231, FAX 03-3402-6226. |
| Mar | 31 | '93 | Acceptance notice date for International Geoscience and Remote Sensing Society Symposium (IGARSS'93) [Sponsor: IEEE Geoscience and Remote Sensing Society] at Kogakiun University, Tokyo, Japan Contact: Prof. Mikios Takagi, Institute of Industrial Science, University of Tokyo, 15-17, Rappongi 7-Chome, Minatoku, Tokyo 106, Japan, phone - 03-3402-6231, FAX 03-3402-6226. |
| May | 31 | '93 | Final paper due date for International Geoscience and Remote Sensing Society Symposium (IGARSS'93) [Sponsor: IEEE Geoscience and Remote Sensing Society] at Kogakiun University, Tokyo, Japan Contact: Prof. Mikios Takagi, Institute of Industrial Science, University of Tokyo, 15-17, Rappongi 7-Chome, Minatoku, Tokyo 106, Japan, phone - 03-3402-6231, FAX 03-3402-6226. |
| Aug | 18-21 | '93 | International Geoscience and Remote Sensing Society Symposium (IGARSS'93) [Sponsor: IEEE Geoscience and Remote Sensing Society] at Kogakiun University, Tokyo, Japan Contact: Prof. Mikios Takagi, Institute of Industrial Science, University of Tokyo, 15-17, Rappongi 7-Chome, Minatoku, Tokyo 106, Japan, phone - 03-3402-6231, FAX 03-3402-6226. |