HISTORY OF THE LAB FOR HIGH ENERGY (LHEA) COSMIC RAY SOFTWARE: VOYAGER, PIONEER, IMP, ISEE AND HELIOS

- Software for cosmic ray projects was developed in the 1960's and 1970's, and has been maintained throughout the 1980's.
- IBM Center mainframes were the primary project computer resource. Computers were too expensive for each Laboratory to buy.
- Technology dictated optimization of computer time, I/O, memory. The result is more complex (i.e. less maintainable) code.
  - Assembler code
  - "Old" FORTRAN 66 / G / H code
  - Tape-based data bases
- Instrument changes (new modes, bit rate changes, the extended life times) and IBM system changes have occurred over the years, causing resources to be required for this software maintenance.
- Resources have been constrained and recent funding has been modest.
- Data is still being received and analyzed, and is scientifically important.
- Current data holding is 100 GB (plus backups).

CODE 660 GOALS FOR COSMIC RAY SOFTWARE SYSTEMS

Science Goals

- Dependable throughput for production processing, i.e. keep the data flowing.
- Satisfactory end-to-end turnaround for investigators requesting analysis products.
- Ensure continuing verifiability and validity.
- Flexible and powerful analysis software, with many requirements.

Software Goals

- Use available resources cost effectively.
- Strengthen overall design as time and resources permit.
- Ensure platform-independence/transportability.
- Maximize maintainability via documentation and simplification.

COMPLEXITY OF A TYPICAL PROJECT: VOYAGER CRS

(Total lines of FORTRAN, Assembly Language, CLISTs, JCL, other.
Subroutine counts.
Top level diagrams of production software, analysis programs, data bases.
Samples of assembly language, CLISTs, JCL.
List of known complications.)

MARCH 24, 1992

COSMIC RAY SYSTEM LINECOUNTS:

SATELLITE	FORTRAN	ASSEMBLER	JCL	CLIST	MODULES
IMP(6,7,8)	80,000	8800	16000(LARGE?) 6300(CNTL)	1200	500
ISEE (CR+DP)	30,067	29,212	6365	2890	300/200 (new/old)
HELIOS $	26,047	2782	6311	2688	250/120
PIONEER (UNDER-ESTIMATE)	23,094	13,387	8271	2298	350
VOYAGER (ESTIM-AFTER SB#IC)%	18,146	11,773	7200	4241	600
SB#HP (INCL SP.ANL.)	20690	6182	1498	5423	200
**********	**********	**********	**********	**********	**********
BASIC SUMS (AS OF 12/91)	198,044	72,136	51,945+IMP	18,740	2200/2520

ASSUMPTIONS:
- FORMER STANDARD ANALYSIS WAS PUT UNDER SB#HP
- FOURIER SEGMENTS ARE PUT UNDER EACH SATELLITE, PIONEER CONTAINS SATELLITE INDEPENDENT SEGMENT ALSO.
- ALL STANDARD IMP PRODUCTION AND ANALYSIS PROGRAMS ARE INCLUDED
- SAME FOR ISEE; DATA POOL IS ABOUT 9000 LINES FORT. AND 1000 ASM.
- HELIOS IS ALL STANDARD PRODUCTION PLUS VARIOUS FIXUP SOURCES
- PIONEER WAS NOT COMPLETE, BUT LINES GIVEN INCLUDE STD PRODUCTION
- VOYAGER WAS NOT COMPLETE, BUT CAN BE RELIABLY ESTIMATED BY COMPARISON WITH ISEE
- SB#HP INCLUDES THINGS SUCH AS LIBMAN, VSCOPY OTHER MISC.

% VOYAGER/ISEE FLUXPLOT AND MATRIX ARE UNDER ISEE

$ PIONEER/HELIOS FLUXPLOT AND MATRIX ARE UNDER PIONEER

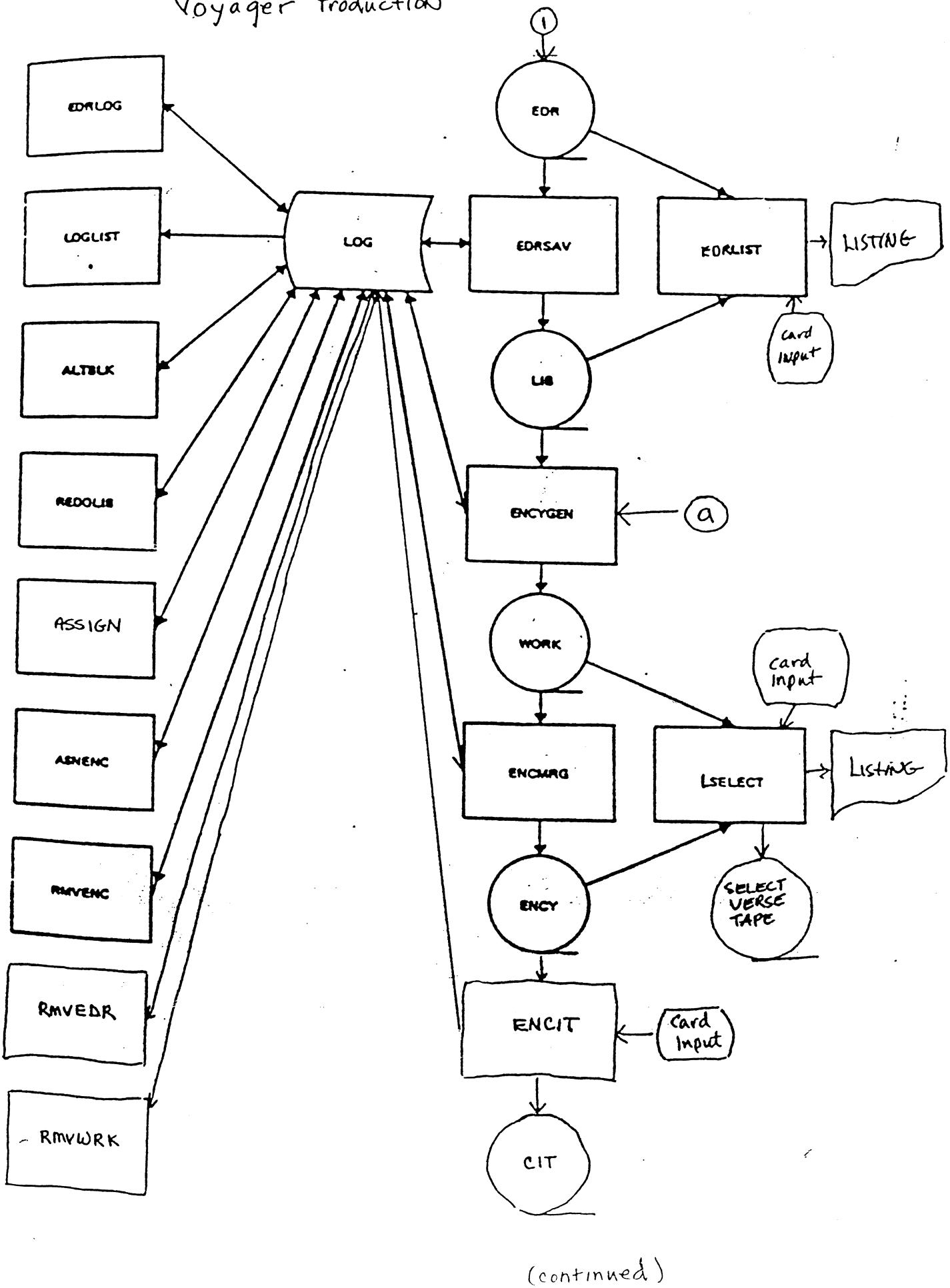
LISTHS2* FLUX CATALOG MANAGEMENT SOURCES IS UNDER SB#IC

current special analysis programs are under sb#hp

PAMS FMCD ANALYSIS CLISTS AND JCL -
NOT READY YET, BUT PROBABLY AT LEAST EQUAL TO CLIST SUMS ABOVE
21 MAJOR DATASETS FULL OF CLIST(S)/SYSTEMS AND JCLS

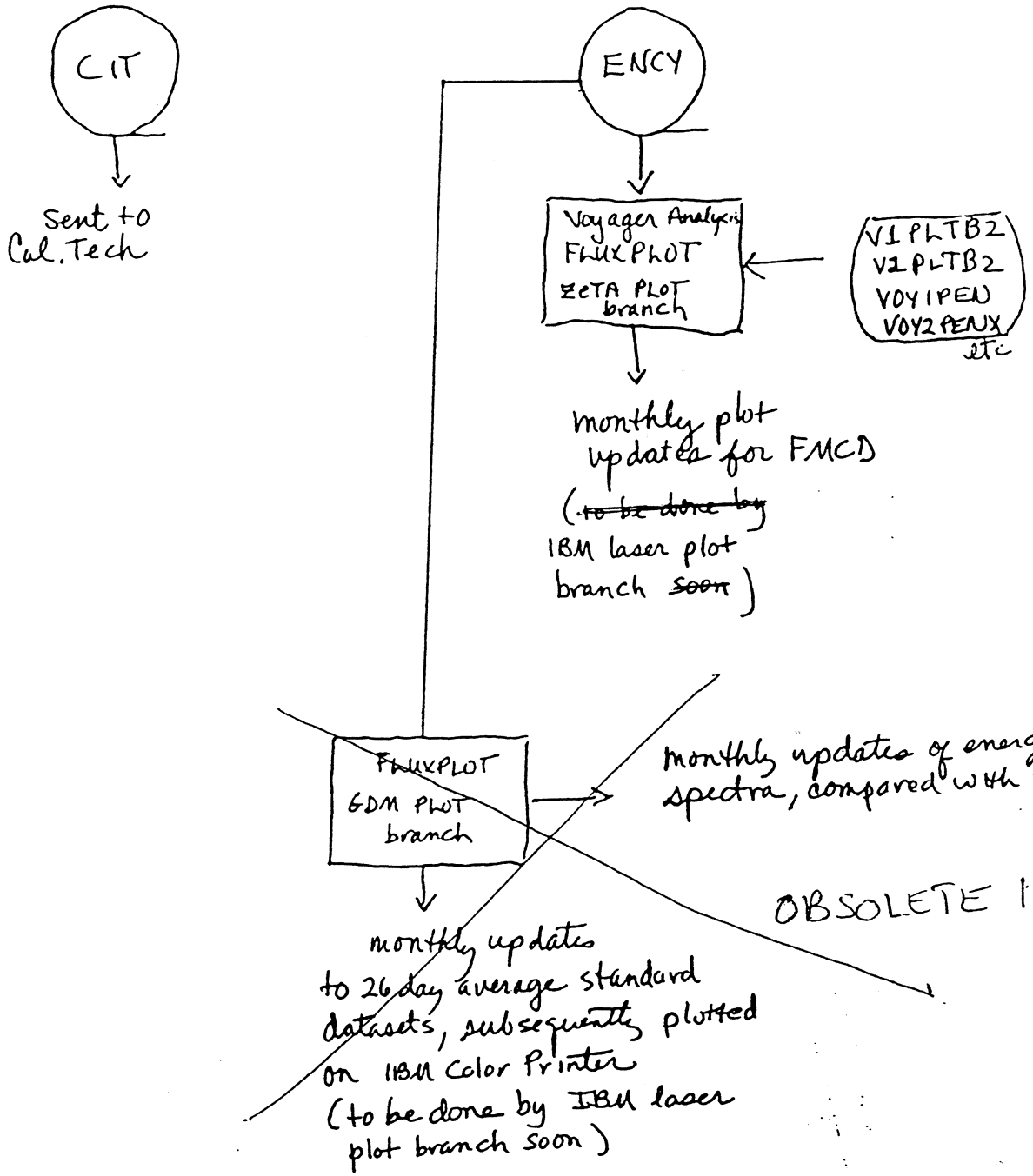
JOHNS FMCD ANALYSIS CLISTS AND JCL - ?
SOME INCLUDED IN VOYAGER & PIONEER JCL SUMS

Voyager Production



(continued)

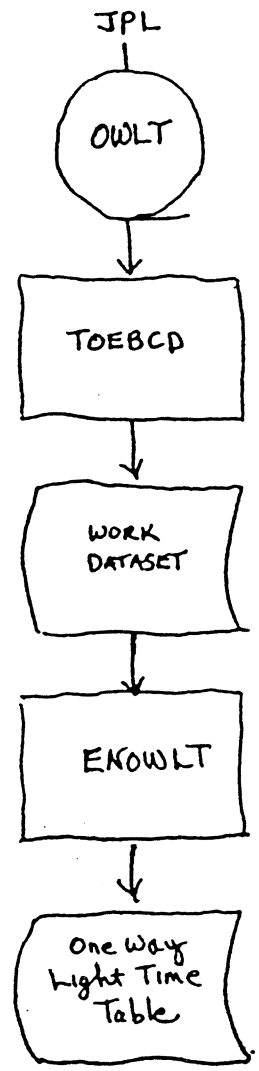
continued:



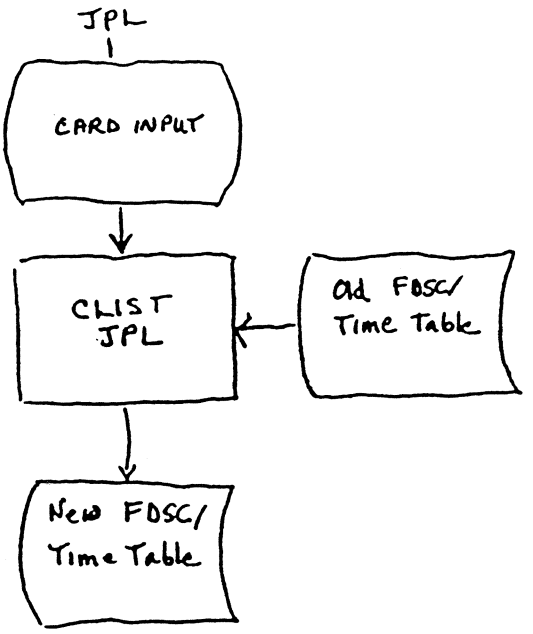
Backup tape copies are made routinely for LIB and ENCY tapes.

(a) Preparation of tables required by ENCYGEN – Voyager

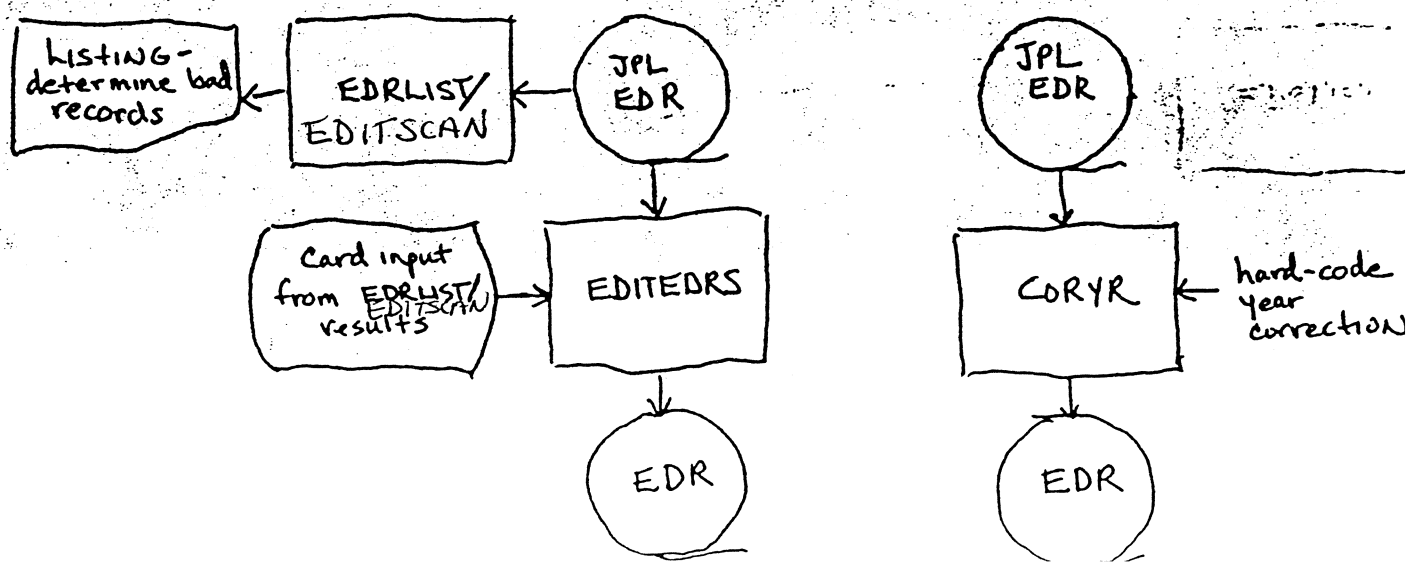
One Way Light Time Tables



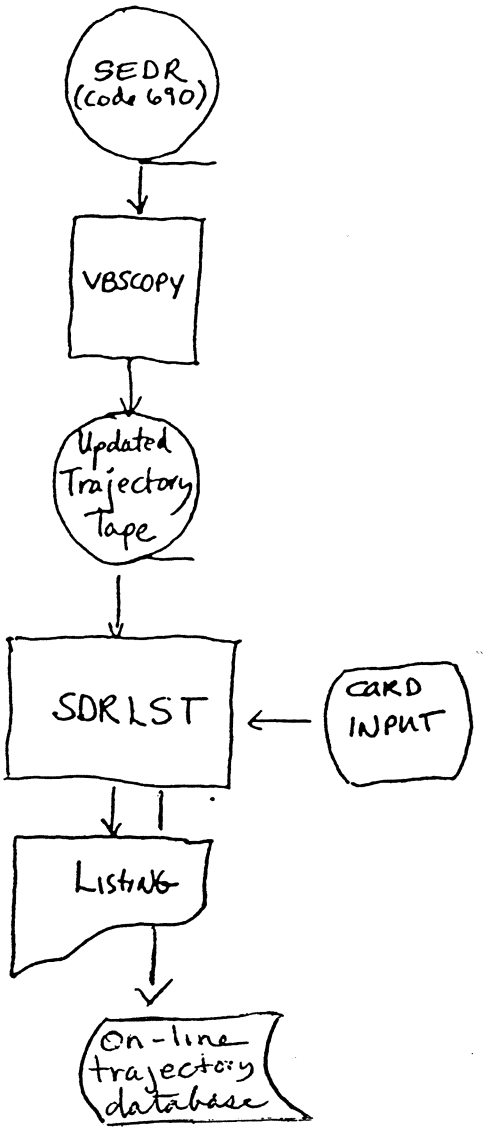
MODGO FDSC/time Tables



(1) EDRs received from JPL may have bad data records, OR YEARS. As needed, two utility programs may be used to edit the original EDR, producing a new EDR which is input to the production system:

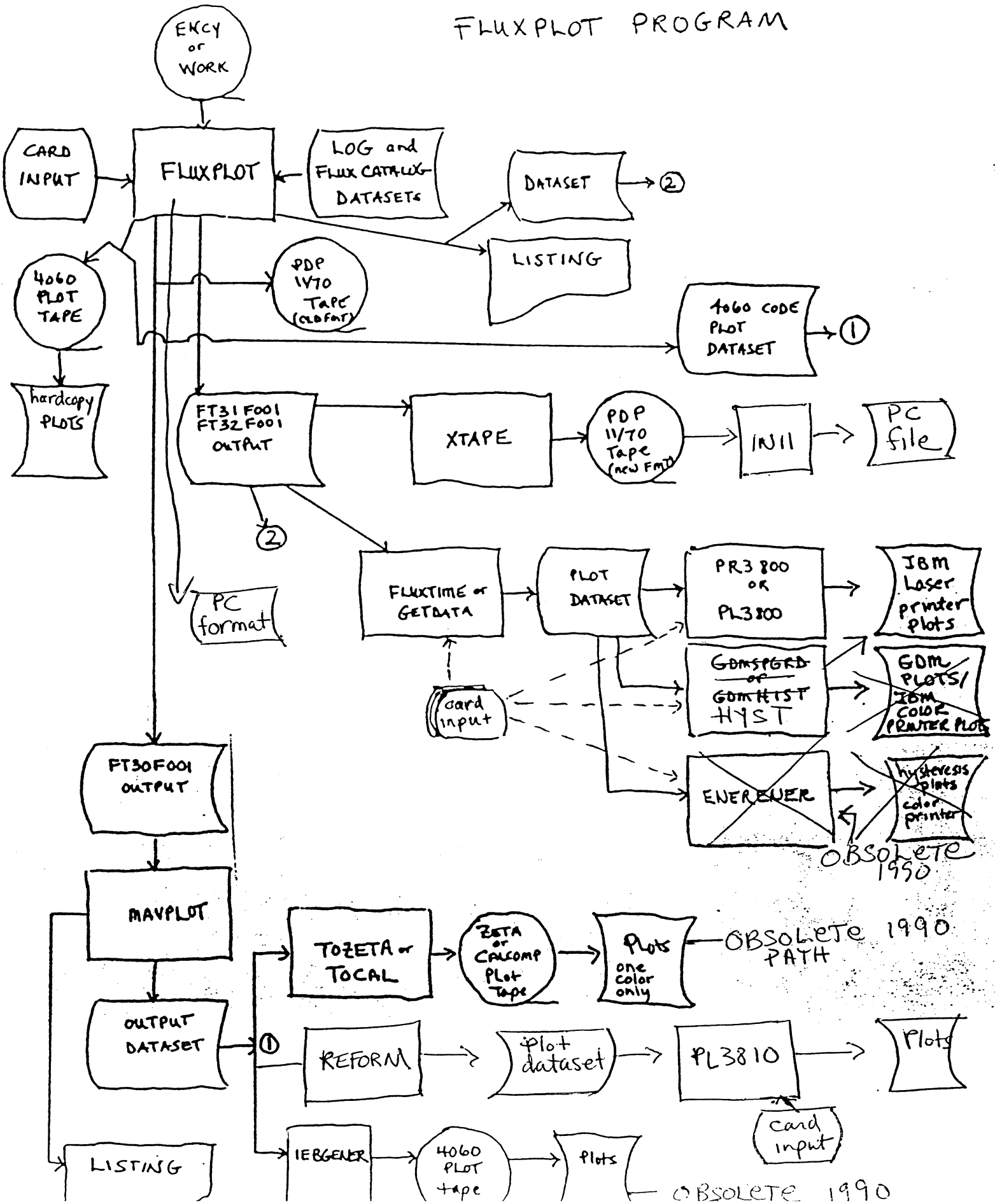


Voyager Trajectory Tapes

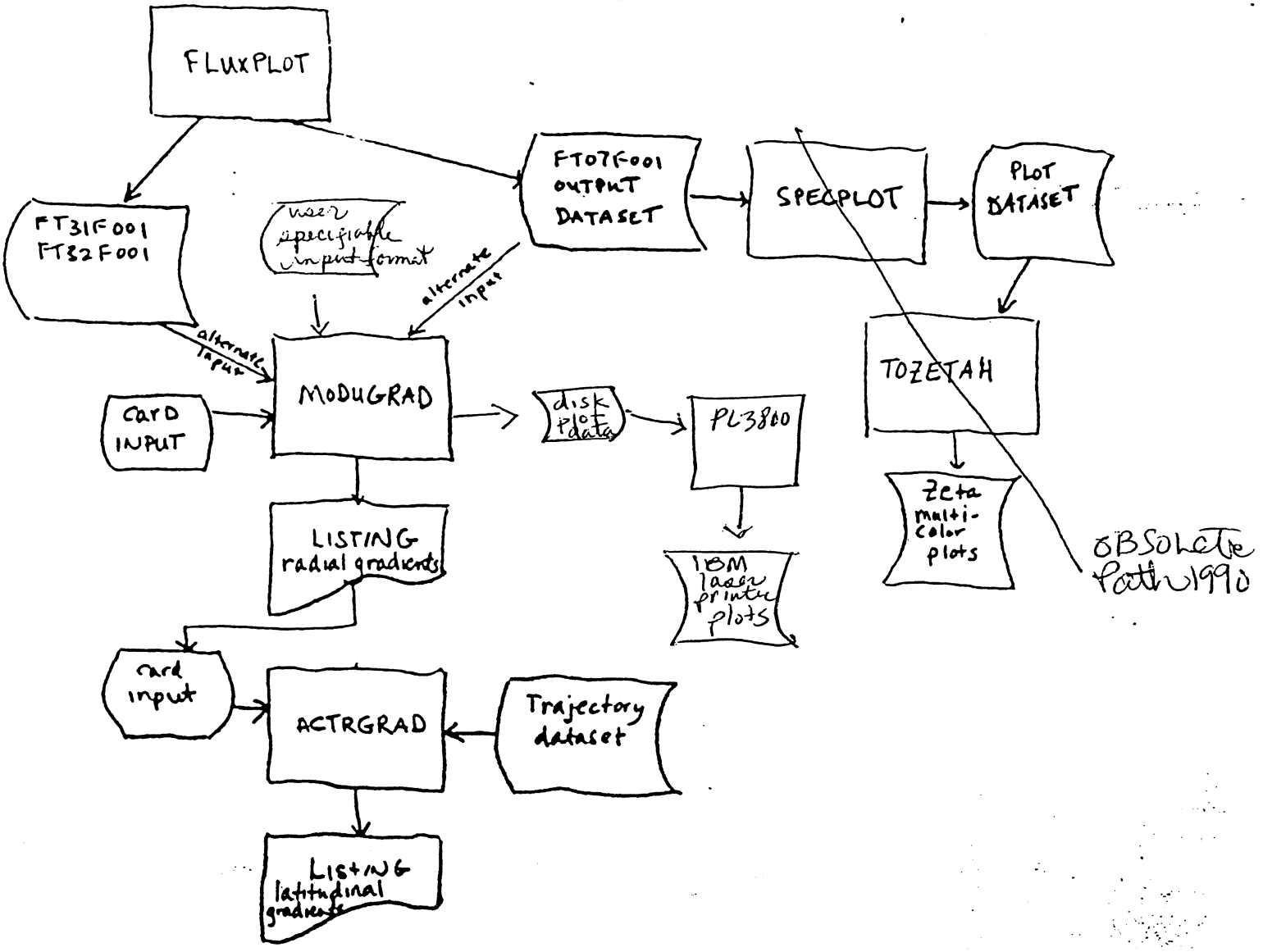


Voyager Analysis

FLUXPLOT PROGRAM

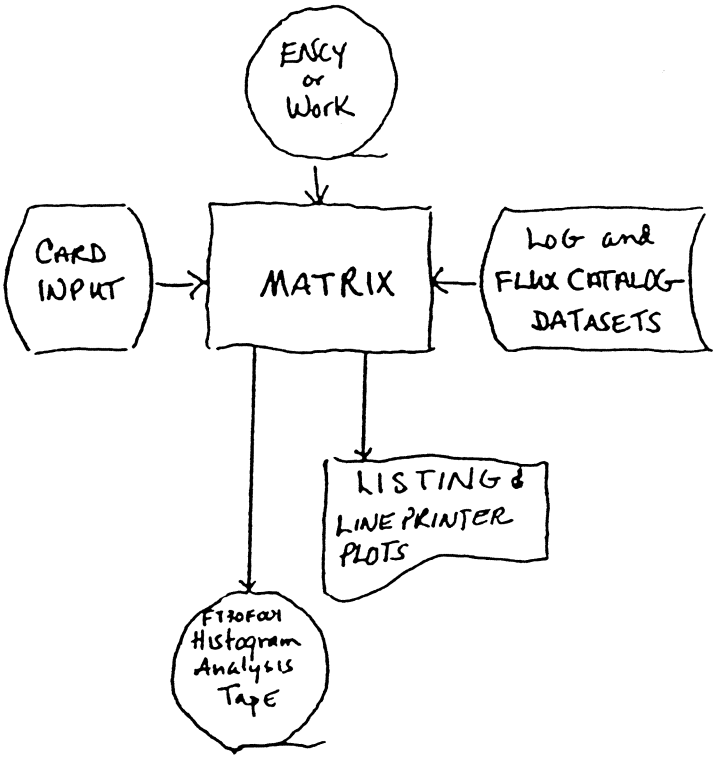


(2) Voyager/ISEE FLUXPLOT



Voyager Analysis

MATRIX PROGRAM



Miscellaneous - Voyager

INTLOG initialize a LOG type dataset

LOGEXPND expand a LOG type dataset when full (same as ISEE)

Statmon.
Cha system for status (ISPF Panels + tables used)

FLUX CATALOG maintenance:

INSTALV install a response table into FLUX catalog

OVRLAY overlay an existing FLUX catalog response table with revised energies and geometry factors

INVENTRY rewrite a FLUX CATALOG, eliminating unused and old response tables.

RESPONSI list FLUX Catalog information and tables; plot tables, if desired, on line printer

RANGES get a formatted listing of the FLUX CATALOG Particle Catalog ONLY.

CATMAN utility subroutines which affect the Particle Catalog of the FLUX Catalog.

ISEE/Voyager

RESPONSE TABLE GENERATION:

TESTM generate particle Track data for the range/energy conditions of a HET

BXGNEW generate input for INSTALV or OVRLAY, using TESTM output as INput to BXGNEW

Table 3-1. Voyager ENCGEN Routine Chart (1 of 3)

VOYAGER ENCYCLOPEDIA GENERATOR	
ISEE-C MODIFICATIONS	
THE FOLLOWING IS A SUBROUTINE CHART OF THE VOYAGER ENCYCLOPEDIA GENERATOR. THE ROUTINES ARE MARKED FOR MODIFIED (•) AND THE NUMBER OF LINES OF CODE MODIFIED FOR ISEE. A BRIEF EXPLANATION IS PLACED AT THE END. (X) INDICATES THE ROUTINE IS DELETED FROM ISEE.	
ENCMAIN	ENCGEN INITIALIZATION PROGRAM
•(2)•ENCGEN	MAIN CONTROL SECTION OF ENCYCLOPEDIA GEN
KTIME	
YDHD	
ANPARM	ANALYZE PARAMETERS
LOL	OPEN LCG LIBRARY BLOCK
MOUNT	
•(20)••• ENINIT	INITIALIZE FLAGS AND ALLOCATE AREAS
XXXXXXXXXXXXXRATTAB	SET UP RATES COMPRESSION TABLE
XXXXXXXXXXXXXXXXXXDECLOG	DECIMAL TO LOG CONVERSION
XXXXXXXXXXXXXXXXXXLOGDEC	LCG TO DECIMAL CONVERSION
•(77)•••••••• PMGEN	SET UP PHA TAG ID MAP
SEECCUNT	DETERMINE NUMBER OF SCIENCE ENG BLOCKS
•(50)••• READER	MAIN DRIVER ROUTINE FOR READ FUNCTIONS
FREAD	READ INPUT TAPE RECORD
•(50)•••••••• RDSEGPRO	DETERMINE NATURE OF SEGMENT
XXXXXXXXXXXXXXXXXXRDSEGSCI	HANDLE SCIENTIFIC DATA RECORDS
XXXXXXXXXXXXXXXXXXRDIDMODE	LOAD DATA MODE ATTRIBUTE TABLE
•(58)•••••••••••••••••• MARKUP	MARKS UP DATA
XXXXXXXXXXXXXXXXXXXXXXXXX DPI	DATA PRESENCE INDICATORS
XXXXXXXXXXXXXXXXXXXXXXXXX DQSW	DATA QUALITY SWITCHES
XXXXXXXXXXXXXXXXXXXXXXXXX BASIC	MARKS UP STATUSES
•(45)•••••••••••••••••• RDIDSEG	IDENTIFIES DATA SEGMENT
•(26)••••••••••••••••• RDESTREL	SENSE A DATAGAP
XXXXXXXXXXXXXXXXXXXXXXXXXRDSAVTIM	SAVE TIMES INTO TIME ARRAY
•(80)••••••••••••••••• DISPATCH	DISPATCH DATA INTO SUBCOM BLOCK
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXDISBTABL	SUBROUTINE LIST
•(30)•••••••••••••••••••••• DISOPEN	ALLOC AND INIT NEXT SUBCOM
XXXXXXXXXXXXXXXXXXXXXXXXXXXXDISSETI	SET NOMINAL FDSC TIMES
XXXXXXXXXXXXXXXXXXXXXXXXXXXXDISSETS	DISPATCH STATUS
XXXXXXXXXXXXXXXXXXXXXXXXXXXXDISSETC	DISPATCH COMMAND DATA
•(80)•••••••••••••••••••••• DISPBR	DISPATCH DATA
•(10)•••••••••••••••••••••• DISPBI	DISPATCH RATES DATA
•(50)•••••••••••••••••••••• DISPP	DISPATCH PHA DATA
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXDISFILL	PAD PHA AND RATES
•(30)•••••••••••••••••••••• DISFILLP	PAD PHA DATA
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXDISCLOSE	SAVE CURRENT POINTERS
•(6)•••••••••••••••••••••• DISUPDSG	UPDATE SEGMENT SOUGHT
•(14)••••••••••••••••••••• DISUPDSB	UPDATE SUPERBLOCK TIME
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOPCODES	(FOR DISPATCHING)
XXXXXXXXXXXXXXXXXXENGPRO	ENGINEERING RECORD PROCESSING
XXXXXXXXXXXXXXXXXXXXXXXXXXVALUES	
XXXXXXXXXXXXXXXXXXXXXXXXXXSETUP	
XXXXXXXXXXXXXXXXXXXXXXXXXXIDMXDP	IDENTIFY ENGINEERING DATA
XXXXXXXXXXXXXXXXXXXXXXXXXXPLGENG	PUT ENGINEERING INTO SUBCOM
XXXXXXXXXXXXXXXXXXDECOM	SEARCH DECOM RECORD FOR POINTERS

Table 3-1. Voyager ENCGEN Routine Chart (2 of 3)

FNEXT		OPEN NEXT EDR DATASET
FNCLOSE		CLOSE CURRENT EDR DATASET
	LGW	CLOSE LOG WORK BLOCK
	LCL	CLOSE LOG LIBRARY BLOCK
(1)	FNOPNEXT	OPEN NEXT DATASET
	LOL	OPEN LOG LIBRARY BLOCK
	ETIO	
(64) MCNITOR		MAIN DRIVER FOR MONITOR FUNCTIONS
	MONQUAL	LOCATE DATA CYCLE TO BE MONITORED
	MOQEVAL	MONITOR STATUS QUALITY
	MCNSTAT	MONITOR STATUS
	MCSMARK	
	MONCAL	MONITOR CALIBRATION
	MCCACHK	
(5)	MCNTREND	PERFORM TREND CHECK
	MCTNCNE	INDEP OF GAIN & SUBCOM
	MCTGAIN	DEP ONLY ON GAIN
	MOTMARK	MARK TREND BITS
(10)	MCTSUBST	DEP ONLY ON SUBCOM
(10)	MCTBOTH	DEP ON BOTH GAIN & SUBC
	TRGEN	TREND CHECK MACRO
(100)	TRTAB	TREND CHECK TABLE
ANLYZE		MAIN DRIVER FOR ANALYZER FUNCTION
(30) ANVOLD		ESTABLISH VOLUME BOUNDARIES
	GETTIM	TIME TABLE INTERPOLATION
	VONUCA	DETERMINE VOLUME NUMBER
	ANCHBD	ESTABLISH CHAPTER BOUNDARIES
	ANCHD1	PUT INFO INTO CHAPTER HEADER
(20)	ANVOLH	PUT DATA INTO VOLUME HEADER
(70)	ANCHHD	PUT DATA INTO CHAPTER HEADER
	ANCHOK	GET QUALITY OF COMMAND DATA
	ANBUMP	ADVANCE SUBCOM LIST POINTER
(30) OUTPUT		MAIN DRIVER FOR OUTPUT FUNCTIONS
	OUTOPEN	OPEN OUTPUT DATASETS
	LOW	OPEN LOG WORK BLOCK
	ROJECB	
	OUVOLCUT	WRITE VOL HEADER; PRINT SUMMARY
	UPDFEET	UPDATE TAPE FOOTAGE COUNT
(3)	OUCHPOUT	WRITE CHAPTER HEADER
	OURROUT	LOCATE RAW RATE VERSE IN OUTPUT
	VERSHEAD	FILL IN PREVERSE FIELDS
	RAWRAT	CONSTRUCT RAW RATES VERSE
	OUCCMCUT	LOCATE AREA FOR COINC CONDITION
	CONMPG	CONSTRUCT COIN COINDITION MAP
	DISHET	DISABLE HET HANDLER
	LETDEL	DELETE LET TERM HANDLER
	BHETDEL	BUILD HET DELETE CODE
	HEIDEL	HET DELETE TERM HANDLER
(1)	OURSMCUT	LOCATE AREA FOR RATES SUMMARY
	RATESM	CONSTRUCT RATES SUMMARY
	RSDISP	DISPLACEMENT TABLE
(5)	OUINITP	TRANSFER EVENT RATES INTO PHA BLOCKS
(30)	OUPHICUT	CONSTRUCT PHA HISTORY MAP
(5)	OUPHAOUT	MOVE PHA VERSES INTO OUTPUT
(8)	CUPHASRT	SORT PHA DATA
	CUPHALOC	ALLOCATE NEW PHA BLOCK
	OUPHASET	FILL PREVERSE OF PHA VERSE
	OUPHAPUT	MOVE PHA DATA INTO OUTPUT

Table 3-1. Voyager ENCGEN Routine Chart (3 of 3)

```
                                OUPHALIS  IDENTIFY VERSE PROCESSED
•(20)•••••••• ENTERM                      TERMINATE PROCESSING
              LTERM                       CLOSE LOG BLOCK

*************************************************************************
DSECTS----
        CCM       COINCIDENCE CONDITION MAP
•(40)••• CHINT    CHAPTER INTRODUCTION
•(10)••• ENCMCB   MASTER CONTROL BLOCK
•(15)••• GLOBAL   LIST OF GLOBAL EQUATES AND MASKS
        INREC     EDR HEADER ENTRIES (DFNHDR MACRO)
        LOGDAT    LOG BLOCKS
XXXXXXXXMATTRIB   DATA MODE ATTRIBUTES
XXXXXXXX SEGELMNT SEGMENT ELEMENT
•(84)••• SEGHDR   EXPERIMENT DATA RECORD HEADER (SEGMENT)
        STAT      STATUS INPUT
•(50)•••• SUBCOM  SUBCOM BLOCK
XXXXXXXX TABLEHDR DATA MODE TABLE PREFACES
•(70)••• VERSE    OUTPUT VERSES
•(6)•••• VOLINT   VOLUME INTRODUCTION

UTILITY ROUTINES-----
XXXXXXXX CNVTME---PACK LINE COUNTS INTO R*8
XXXXXXXX DECLOG   DECIMAL TO LOG CONVERSION
XXXXXXXX DLCUFK   UNPACK R*8 LINE COUNT
XXXXXXXX DTMJS    UNPACK MSEC OF YEAR
XXXXXXXX DTUPK    UNPACK MSEC OF YEAR
XXXXXXXX FDSCOT   CONVERT FDSC TO FLOATING LINE COUNT
        FLOAT     TURNS FIXED INTO FLOATING POINT
        GETDATE   FETCH DATE
XXXXXXXX GETFDSC  EXTRACT FDCS FROM DATA RECORD
XXXXXXXX GETTIM   TIME INTERPOLATION ROUTINE
        INVOKE    CALL PROCEDURE
        I4TODP    CONVERT I*4 INTO R*8
XXXXXXXX LOGDEC   LOG TO DECIMAL CONVERSION
        MOVE      GENERAL DATA MOVE ROUTINE
        MSG       GENERAL MESSAGE ROUTINE
        TIMCHK
•(10)••• TIMECT   CONVERT TIME INTO R*8 MSEC

ATTRIBUTE BLOCKS------
XXXXXXXX CR1ATTR
XXXXXXXX CR2ATTR
XXXXXXXX CR3ATTR
XXXXXXXX CR6ATTR
XXXXXXXX GS3ATTR
XXXXXXXX MAITRGEN MACRO TO GENERATE ATTRIBUTE BLOCKS

MISC----
XXXXXXXX DDCBGEN          GENERATE DATA DISPOSITION CONTROL BLOCK
XXXXXXXXXXXXX DDCBCR6     DATA DISPOSITION CONTROL BLOCK FOR CR6
XXXXXXXXXXXXX DDCB        DATA DISPOSITION CONTROL BLOCK FOR OTHERS
XXXXXXXXXXXXXXXXXX DATAGAP ALLOCATE SUBCOM STATE AFTER DATA GAP
        LOPRCLOG          CREATE PROLOG FOR LOG BLOCK
XXXXXXXX PHAMSK           EVENT DETECTOR CONDITIONS
XXXXXXXX RECCNT
XXXXXXXXXXXXX SKPHD
XXXXXXXX SCEDAT           READ CARD
XXXXXXXXXXXXX SKPHD       READ CARD
XXXXXXXX STATUS           MACRO
```

TECHNICAL ISSUES / UNKNOWNS

Technical Issues

- Much assembler code, using macros.
- The entire CLIST and JCL front end will have to be redone.
- Data base systems use complex cataloguing schemes and sequential tape data bases. (Potential data base redesign.)
- Other MVS dependencies, such as FTIO, DAIO, and TLS.
- We do not presently have the resources ($ or scientific manpower) to convert all of our software to be machine-independent, nor to rewrite our systems.
- Scientific manpower is tight.
- Verification/validation of all converted software is critical.

Unknowns

- AIX Assembler (including macros) compatibility
- AIX VS FORTRAN/FORTRAN 77 compatibility
- AIX internal data representation (i.e. little endian, big endian, IEEE)
- Tape Library System (TLS) replacement or other tape/cartridge use on data analysis machine? and how much storage will we retain?
- MDSDS capabilities, policies, costs
- NCCS commitments, i.e. how long will MVS be around, how long will AIX be around, and for how long can we compute on the IBM? Perhaps we need an MOU?

TESTING / VERIFICATION / VALIDATION

- Maintaining valid scientific software requires adequate overlap between MVS and new platform for adequate post-conversion testing. Must have before and after cases. Must test programs individually and as a system.
- Must compare before and after output data "bit by bit". Complications include precision incompatibilities and internal data representation problems.
- Must compare before and after output products over many time periods for the full range of output products.
- Must confirm no errors have been introduced due to the conversion, e.g. lost data, bad numbers.
- Must have traceability to source code.
- Must be directed and overseen by experienced personnel.

APPROACH

- Obtain information on AIX specifics
- Confer with the science team, identify capabilities which may not need to be converted (prioritize, and consider projected resources)
- Assemble documentation, and perform an audit of code to be converted
- Begin conversion of sample of one or two programs, to assess costs and technical difficulty
- Revisit current requirements and consider simplifications in design:
  - common data formats
  - eliminate unneeded data paths
  - unifying systems where possible
- Decide on priorities of conversion
- Allocate software to be converted to AIX or directly to another platform.

POTENTIAL SOLUTIONS

- Run an MVS partition for the life of the IBM mainframe. We can then invest resources to **move applications** to other platforms (PC & UNIX) with adequate study and redesign.
- If conversion to AIX is "easy", i.e. not a big investment, then **convert to AIX now**, and then **move applications** to other platforms (PC & UNIX) with adequate study and redesign.
- If conversion to AIX is not "easy" and MVS has a limited lifetime here at GSFC, selectively **move applications** off of the IBM mainframe quickly or **convert to AIX**.

CONCLUSIONS

- January 1994 conversion to AIX will not allow us to have all satellites continue processing data.
- We need resources to do the conversion.
- We need to address the Technical Issues and answer the Unknowns above.

March 24, 1992
Nancy Laubenthal Code 664 NASA/GSFC (301) 286-5778