



Environment
Canada

Environnement
Canada

Environmental
Conservation

Conservation de
l'environnement

COM81-2
C 2

INTERNAL WORKINGS OF COMPUTERIZED LABORATORY DATA MANAGEMENT

BY WATER QUALITY BRANCH, PACIFIC & YUKON REGION

Robert J. Waid, Paul H. Whitfield and Fred T.S. Mah

TD
227
B74
COM81-
3 c 2

**Inland Waters Directorate
Pacific and Yukon Region
Vancouver, B.C.**

BVAE North Van. Env. Can. Lib./Bib.



36 002 276

TD
227
B74
COM91-3
C.2

INTERNAL WORKINGS OF
COMPUTERIZED LABORATORY DATA MANAGEMENT
by WATER QUALITY BRANCH, PACIFIC AND YUKON REGION

R.J. WAID, P.H. WHITFIELD AND F.T.S. MAH

WATER QUALITY BRANCH
INLAND WATERS DIRECTORATE
PACIFIC AND YUKON REGION
VANCOUVER, B.C.

FEBRUARY 1981

10 334

INTRODUCTION

A computerized data and laboratory management system has been developed by Water Quality Branch, Pacific and Yukon Region. The system which has been developed is designed to operate independent of data capture activities, relieving the system of real time obligations. The system is self explanatory, hence user friendly. The current implementation is accessed by dialup terminals located in the laboratory to an IBM 4341 located at Simon Fraser University.

I. FILE STRUCTURE

Some permanent and all temporary data files used by the system are created by the system when needed. Each fiscal year when the first user signs on to the system two data files are created for that year, these being TABLE.YR and ANALY.YR. For instance in 1982 the files created would be ANALY.82 and TABLE.82. The file TABLE.YR holds data on the samples to be analysed by the lab while ANALY.YR contains various information such as the last sample number used in TABLE.YR, and various sample, analysis and laboratory statistical data.

Eight permanent files are used by the system which must exist prior to their use. These are 'PARAMETER', 'PROJECT', 'STATIONS', 'PASSWORD', 'WORKSHTDAT', 'COMMANDS', 'COMMANDS1' and 'COMMANDS2'. Each file used by the system has associated with it a unique logical unit (see Table I) which is assigned when needed. User specified flow, monitoring and Naquadat files are also used, the user being prompted for the names of these files when using the command 'EXTRACT.SAMPLES'.

The following section gives an indepth description of each file used and the data contained within them.

Logical Unit	File
2	*MSINK*
3	*MSOURCE*
4	TABLE.YR
5	SCHEMA
6	COMMANDS, COMMANDS1, COMMANDS2
7	-HOLD
8	PARAMETER
9	PROJECT
10	STATIONS
11	ANALY.YR
12	-HOLD1
13	WORKSHTDAT
14	PASSWORD
15	ANALY.YR
16	-DD
17	D.FILE
18	DD.FILE
19	NAQ.FILE

Table I Logical Units and File Associations Used by the System

A. ANALY.YR

This file holds statistics on samples and analysis, for both entered and completed samples. The overhead time, calendar days worked, man-days available, time other duties, and last sample number used by TABLE.YR are also contained in this file.

Specific lines in this MTS file are reserved to hold the various information. For example monthly analysis and sample counts are stored on the line number corresponding to the month, therefore information on January is on line number 1 and information for December is on line 12. The yearly counts are stored on line year-1900 so that data for 1982 is on line 1982-1900 or line 82 in the file. Individual counts and TMU's for the three lab sections are stored in a similar manner only 100 is added to the line number. Table II shows a complete breakdown of where the various data is stored, what data is stored and the format it is in.

TYPE OF DATA	LINE # MTS	DATA CONTAINED	FORMAT
MONTHLY	MONTH NUMBER 1-12	NUMBER ANALYSIS ENTERED, # SAMPLES ENTERED, # ANALYSIS COMPLETED, # SAMPLES COMPLETED	(I5,1X,I3,1X,I5,1X,I3)
YEARLY	YEAR-1900 (3)	# ANALYSIS ENTERED, # SAMPLES ENTERED, SAMPLES COMPLETED # ANALYSIS COMPLETED	(I7,1X,I5,1X,I5,1X,I7)
Monthly	100+MONTH (2)	# INORGANIC (SAMPLES COMPLETED, STANDARDS, RECHECKS), # ORGANIC (SAMPLES STANDARDS, RECHECKS), # BIOLOGY (SAMPLES STANDARDS, RECHECKS), TOTAL TMU's: IN- ORGANIC (SAM,STA,REC), ORGANIC (SAM,STA,REC), BIOLOGY (SAM, STA,REC)	(9(I4,1X),9(F6.2,1X))
YEARLY	100+(YEAR-1900) (1)	SAMPLE TMU, STANDARD TMU, RECHECK TMU, # STANDARDS COM- PLETED, # RECHECK COMPLETED # SAMPLES COMPLETED.	(3(F7.2,1X),3(I7,1X))
	200+(YEAR-1900)	LAST SAMPLE # ENTERED IN TABLE YR	(I5)
YEARLY	300	OVERHEAD TIME FISCAL YEAR TO DATE, # CALENDAR DAYS WORKED, MAN-DAYS AVAILABLE, TIME-OTHER DUTIES.	(F7.2,1X,F6.2,1X, F7.2,1X,F8.2)

2

NOTE: (1) # Samples Completed = (2) Inorganic (Sam) + Organic (Sam) + Inorganic (Sam) and is not the same as (3) # samples completed.

TABLE II Breakdown of file structure and data for file ANALY.YR

B. PARAMETER

This file holds, for each parameter, the parameter code, description, type, and number of analysis done for the last two months and two years. The file is sorted according to the parameter code so that searching for a given parameter need not be done sequentially but instead a binary search can be used. The file structure is shown in table III.

At present a max of 800 parameters are allowed in this file. For this amount to be increased one change must be made in subroutine INSPAR.

C. TABLE.YR

This file holds the information on all samples still in the system. Each sample entered is given a unique sample number which is one more than the previous. The previous number being held on line 200+(year-1900) of file ANALY.YR. A group of lines in this file is devoted to each sample. The line numbers for a specific sample are those from SAMPLE# X 10 through (SAMPLE# X 10)+9.999. A header card for each sample is on line SAMPLE# X 10 and the analysis to be done are on the following lines starting at line (SAMPLE# X 10)+.001. Recheck data starts on lines (SAMPLE# X 10)+.05. A complete format of this file is shown in table IV.

The stage of a sample is a two digit code indicating the stage the sample is in. When a sample is first entered into the system its stage is ' '. After all the analysis have been completed the system changes this stage to 'CO' for completed. The lab head then checks the results and when satisfied with them replaces the stage with '**', which indicates the sample is ready to be written to Naquadat, monitoring and replicate files after which its stage is changed to 'ND' by the system. This tags the sample for later deletion from the system.

The parameter stage is a one digit code which is ' ' for parameters which have not yet been analysed, '0' for those which have been analysed, and '1' for those whose bottles have been broken.

The recheck code is a one digit code indicating whether rechecks were done on that analysis. A '0' indicates none and a '1' indicates that rechecks were done.

DATA CONTAINED	FORMAT
TWO DIGIT CODE, PARAMETER CODE, DESCRIPTION, COST TO ANALYZE, TMU, # ANALYSIS (SAMP, STAND, RECHECKS) MONTH, # ANALYSIS (SAMP, STAND, REC) PREVIOUS MONTH, # ANALYSIS (SAMP, STAND, REC) YEAR, # ANALYSIS (SAMP, STAND, REC) PREVIOUS YEAR.	(2A1,1X,6A1,1X,50A1,1X, F6.2,1X,F7.4,1X,3I3,I2, 1X,3I3,IX,3I9,I2,1X,3I9, I2)

TABLE III BREAKDOWN OF FILE STRUCTURE AND DATA FOR FILE PARAMETER.

SAMPLE NUMBER	MTS LINE NUMBER	DATA CONTAINED	FORMAT
I	I X 10	NUMBER BOTTLES, # PARAMETERS, # PARAMETER COMPLETED, STAGE, DATE ENTERED, TIME SAMPLE TAKEN, LEFT BANK, DEPTH, STATION #, TITLE, COST OF ANALYSIS, PROJECT#, PROJECT DESCRIPTION, TIME ZONE	(3(I2,1X),2A1,1X,1X, 8A11X,5A1,1X,I3,I2,1X, 12A1,1X, 50A1, 1X, F7.2, 1X, I3, 1X, 60A1, 1X, 3A1)
	(IX10)+.001 through (IX10)+(NPARX.001)	PARAMETER CODE, BOTTLE #, STAGE, VALUE, DESCRIPTION, DATE DONE, RECHECK CODE INTERNAL MTS LINE #	(6A1, 1X, 10A1, 1X, 1A1, 1X, 8A1, 1X, 35 A1, 1X, 6A1, 1X, I1, 1X, I9)
	(IX10)+.05 + (PARAMETER ORDER X .001)	# RECHECKS, RECHECK VALUES	(I2, 1X, 10(7A1,1X))

Table IV FILE FORMAT FOR TABLE.YR

D. PROJECT

This file contains the projects presently in use and the various data associated with each. This data includes a description, total costs associated with the project for the present and previous year as well as totals of the number of analysis done and number of samples entered for a twelve month period. Table V shows the data stored and its format.

E. WORKSHTDAT

This file holds the information on the worksheets produced for the various parameters. The data and format description is given in table VI. Once a worksheet has been completed and the number of analysis equals the number of analysis completed then those lines are deleted from the file and the worksheet number may be reused.

F. PASSWORD

This file contains the names of all personnel allowed access to the system along with their password and a list of the parameters for which they have access to. It also contains their security code which at present one of the following:

- 1 - TECHNICIANS
- 2 - SHIPPER/RECEIVER
- 5 - LAB HEAD
- 6 - PAUL WHITFIELD, ROBERT WAID

The file format is given in table VII

PROJECT NUMBER	MTS LINE NUMBER	DATA CONTAINED	FORMAT
I	I	DESCRIPTION, COST YEAR1, YEAR1, COST YEAR2, YEAR2, (COST(J), # ANALYSIS,(J), #SAMPLES (J), MONTH(J), J=1...12)	(4X, 60A1, 1X, F8.2 I2, 1X 12 (F7.2, I4, I3, I2, 1X))

TABLE V FORMAT OF FILE PROJECT

WORKSHEET NUMBER	MTS LINE NUMBER	DATA CONTAINED	FORMAT
I	I	NUMBER ANALYSIS, NUMBER ANALYSIS COMPLETED	(I3, 1X, I3)
	I+(.001XNUM) NUM=1,...,# ANALYSIS	YEAR, SAMPLE NUMBER, BOTTLE #, INTERNAL MTS LINE# INTO TABLE.YR	(2A1, 6A1, 1X, 10A1, 1X, I9)

TABLE VI BREAKDOWN OF FILE STRUCTURE AND DATA FOR FILE WORKSHTDAT.

MTS LINE NUMBER	DATA CONTAINED	FORMAT
I	NAME, PASSWORD, CODE	(10A1, 1X, 10A1, 1X, I1)
I+.001	PARAMETERS	(30(6A1, 1X))

Table VII BREAKDOWN OF FILE STRUCTURE FOR FILE PASSWORD

G. COMMANDS, COMMANDS1, COMMANDS2

These three files contain a list of the commands available to each of the three types of users. The file 'COMMANDS' holds those commands for use by 'LAB HEADS', 'COMMANDS2' for 'SHIPPER', and 'COMMANDS1' for 'TECH'. No file format is given for these files since it is straight forward. Each line in the file contains the name of a command available to that user.

II. SYSTEM SUBROUTINES

A fair number of system subroutines are used within the system to perform various functions, most of which involve character handling. Table X contains a list of the subroutines along with a brief description of what each does. If a more detailed description is needed see MTS volume 3.

SUBROUTINE	PURPOSE
ATRAP	To override the break key.
ATTNTRP	To override the break key.
BTD	To convert FORTRAN INTEGER numbers into numeric character strings.
CMD	To execute an MTS command from a program and return to the program after the command has been executed. Command is echoed on *MSINK*.
CMDNOE	To execute an MTS command from a program and return to the program after the command has been executed. Command is never echoed on *SINK*.
DTB	To convert a string of numeric characters into a FORTRAN INTEGER number.
EDIT	To call the MTS file editor from a user program.
EMPTYF	To empty a file without destroying it.
EQUC	To compare two characters for equality.
EQCMP	To compare two character strings for equality.
FINDC	To search for any one of a set of characters.
FREAD	To provide a free format input facility.
FTNCMD	To allow a program to issue commands to the FORTRAN I/O library.
LCOMC	To determine whether one character string is less than, equal to, or greater than another string.
MOVEC	To move character strings from one place to another.
RENUMB	To renumber the line in an MTS file.
SORT	To sort files.
TIME	To allow the program access to the date in convenient units.
UNLK	Unlocks a file attached to a specified logical unit.

Table X SYSTEM SUBROUTINES USED BY THE SYSTEM.

III. SYSTEM SETUP

The system is set up under the signon I.D. WQB6 in such a way that when a user signs on to this I.D. a short program is run. This program prompts the user for one of 'TECH', 'SHIPPER' or 'LAB HEAD' which in turn starts one of three main programs running. These programs then prompt the user for various information and commands. The system is set up this way so that subroutines corresponding to commands which are for the use of the lab heads only are not loaded along with the other commands when the user is a technician or a shipper. In this way the cost of a run is reduced somewhat.

All the subroutines making up the commands are contained in a library, LAB.LIB, which was created for the system with the use of the public file '*OBJUTIL', which is described fully in MTS Volume 2.

Since the UBC sort subroutines are used in some of the system's subroutines these were also included in the library using the following set of instructions:

```
#R *OBJUTIL
EDIT LAB.LIB
INCLUDE *LIBRARY ONLY SORT
STOP
```

You may wish to refer to Figure 1 at this point for a clear understanding of the system setup.

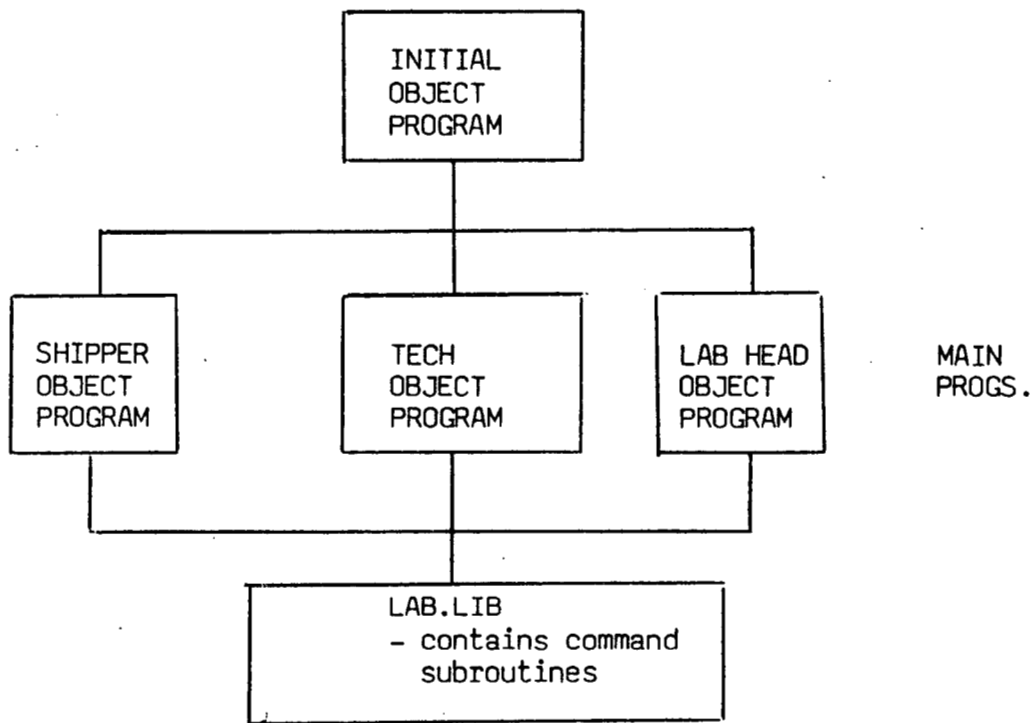


Figure 1 System Setup

Since the system uses various subroutines from the LAB.LIB library, SFU system library and UBC library it is important that they are searched in the correct order. To accomplish this the following command is added to the end of each of the object files of the main programs:

```
$CONTINUE WITH WQB4:LAB.LIB+UNSP:LIBRARY+UNSP:UBCLIB
```

If a change is made to an existing command or a new command is to be entered into the library the following instructions are used:

```
$R *FTN PAR=S=SOURCE.FILE L=-L
```

```
$R *OBJUTIL SCARDS=-L O=LAB.LIB
```

If the source file being entered is for a new command make sure it is entered in the appropriate main programs and their corresponding command files (see Fig.1).

IV. PASSWORDS AND SYSTEM PROTECTION

The system is protected from misuse by having each user have access to only a limited set of the total commands available.

When a command is run the users code is first checked to see if access is allowed to that command, if not a message is printed and the command is aborted.

Also, within commands involving parameters, the parameter entered by the user is checked against a string of parameters to which he has access to. Anyone with a code of 5 or more has access to all commands and all parameters. The parameters allowed, code and password for each user is kept in the file 'PASSWORD' under the signon I.D. WQB4

V. MULTIUSERS

With the use of the system subroutine 'UNLK' the system is made multiuser. After most reads and writes to a file it is simply unlocked to allow other users to access that file. This should pose little or no problems since it is unlikely that two users will be altering the same part of a file at the same. For instance the probability of two users entering data for the same sample at the same instance is very low and most accesses to other files are for reads only.

The only files which are not unlocked right away are the two temporary files '-HOLD' and '-HOLD1' which are not unlocked until just before the command is left. This may cause a slight delay in response time if another user's command wishes to use one of these files while it is locked.

VI. COMMANDS AND THE FILES THEY USE

Every command available uses at least one of the data files, and different commands use different parts of these files. Table XI gives a list of the commands along with the subroutine they correspond to, the files they access and in what way they access them.

This will be helpful if in the future additional fields are added to the existing records of any of the data files or if new records are added to these files.

There is also a command called 'MTS' available to persons with a security code of '6'. This command is used to enter MTS mode after which the program may be restarted from where it left off with the use of 'RESTART'.

COMMAND	SUBROUTINE	FILES ACCESSED	WAY ACCESSED
A.S.PAR	ASPAR	PARAMETER PROJECT ANALY.YR -HOLD	READ,WRITE READ,WRITE READ READ,WRITE
WORKSHEET	WKSHET	WORKSHTDAT PARAMETER ANALY.YR	READ,WRITE READ READ,WRITE
ENT.DAT	ENTDAT	WORKSHTDAT PARAMETER ANALY.YR -HOLD TABLE.YR	READ,WRITE READ,WRITE READ,WRITE READ,WRITE READ,WRITE
ENT.DIRECT.DATA	ENTDIR	PARAMETER ANALY.YR TABLE.YR -HOLD	READ,WRITE READ,WRITE READ,WRITE READ,WRITE
PAR.CHG.VAL.	CHGPRV	TABLE.YR	READ,WRITE
LAB.STATISTICS	LABST	PARAMETER ANALY.YR	READ READ
BALANCE	BALAN	ANALY.YR TABLE.YR	READ,WRITE READ,WRITE
ALL.PARAMETER	ALLPAR	PARAMETER TABLE.YR	READ READ
CHN	CHN	TABLE.YR	READ,WRITE
CHANG.COST	CHGCOS	PARAMETER	READ,WRITE
BILLING	BILL	PROJECT	READ

COMMAND	SUBROUTINE	FILES ACCESSED	WAY ACCESSED
INIT.SAMPLE	INTSAM	PARAMETER SCHEMA TABLE.YR ANALY.YR PROJECT STATIONS -HOLD	READ READ READ,WRITE READ,WRITE READ,WRITE READ READ,WRITE
CR.SCHEMA	CRSCH	PARAMETER SCHEMA -HOLD	READ READ,WRITE READ,WRITE
PROJECT.LIST	PROLST	PROJECT	READ
STATION.LIST	STALST	STATIONS	READ
LIST.PARAMETER	PARLST	PARAMETER -HOLD	READ READ,WRITE
ADD.PROJECT	PROADD	PROJECT	READ,WRITE
INSERT.PARAMETER	INSPAR	PARAMETER	READ,WRITE
ADD.STATION	STADD	STATIONS	READ,WRITE
CHG.PASSWORD	CHGPSW	PASSWORD	READ,WRITE
BROKEN.BOTTLE	BOTBRK	TABLE.YR	READ,WRITE
LIST.SAMPLE	SAMLST	TABLE.YR	READ
SCHEMA	SCHEMA	SCHEMA	READ
STAT.CHANGE	STATCG	STATIONS TABLE YR	READ READ,WRITE

COMMAND	SUBROUTINE	FILES ACCESSED	WAY ACCESSED
STAGE.CHANGE	CHGSTG	TABLE.YR	READ,WRITE
UNFINISHED.SAMPLES	UNFIN	TABLE.YR	READ
DELETE.SAMPLE	DELSAM	TABLE.YR	READ,WRITE
STATUS	STATUS	ANALY.YR	READ
SAMPLE.IN.PROJECT	SAMPRO	TABLE.YR ANALY.YR PROJECT	READ READ READ
TIC-TOC	TICTOC	TABLY.YR	READ,WRITE
TEST.FOR.BALANCE	BALTES	TABLE.YR	READ

TABLE XI