UEA VMS Demand Control

System Description

K.Worvill
May 1990

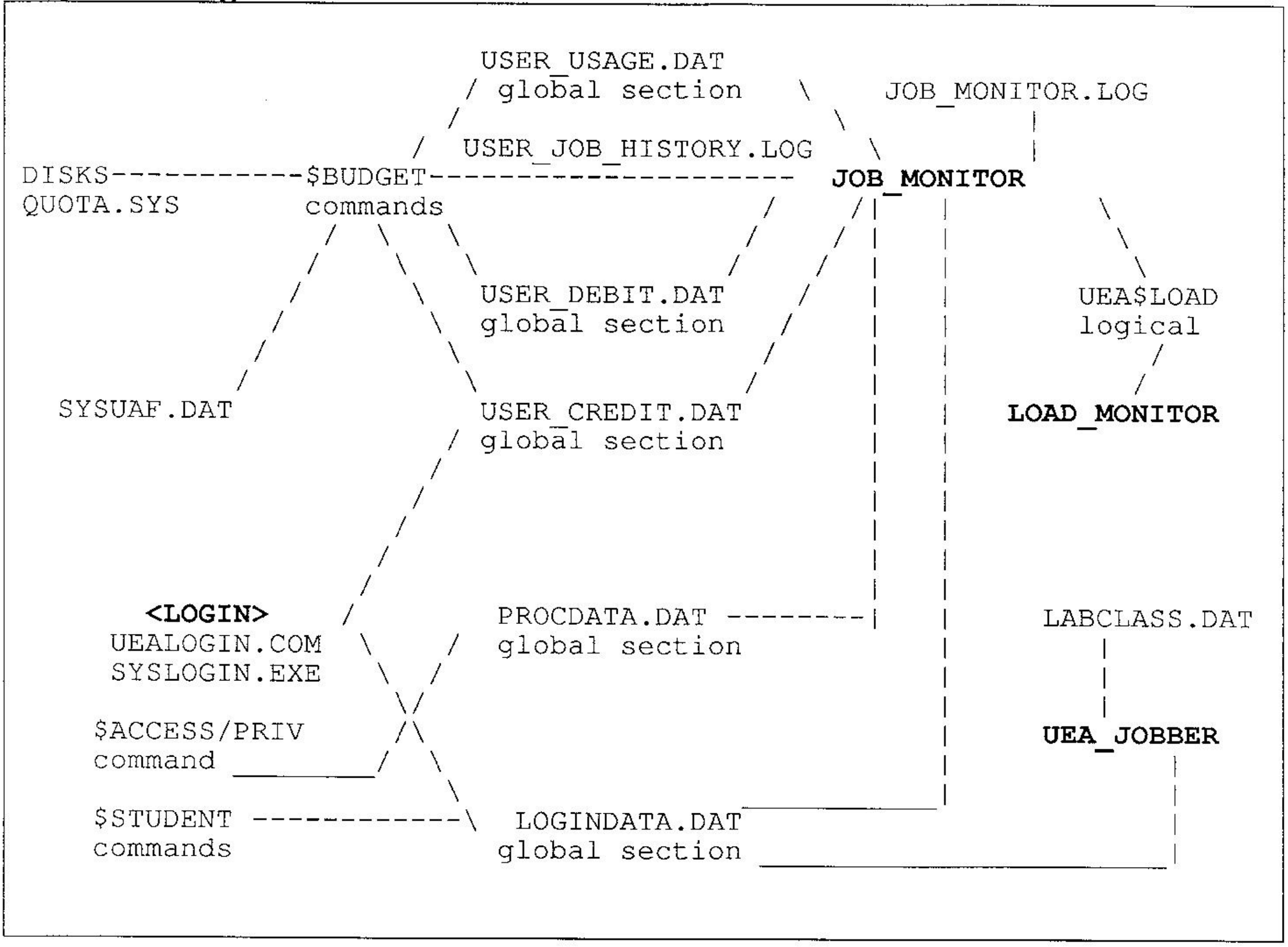
Technical description of the system used to control interactive and batch use on CPC VMS systems including the scheduled class mechanism.

Contents

Version 2.0

Cost £0.xx

0 System Diagram



1 Main functions of each process

o JOB_MONITOR

Collects system wide process data
Updates user credit according to system load
Maintains student user counts,casual & class
Implements forced logouts for
. overdrawn users
. idle terminals
. excess casual students

(CPU overhead approx. 5 secs/hour on 8650 running 50 processes)
(" 20 secs/hour 780 " 20 ")
(" 25 secs/hour microVax 10 ")

o LOAD_MONITOR

Collects system wide process state counts
Calculates system demand factor

(CPU overhead approx. 0.5 secs/hour on 8650 as above)

o UEA_JOBBER

Updates labclass flags in LOGINDATA.DAT from timetable LABCLASS.DAT at 1 Hr. & 1/2 Hr.
Updates user credit from quotas in SYSUAF using $BUDGET commands at daily & weekly intervals

o <LOGIN>

Checks user credit & student login counts
Collects batch jobs queue priority

1.1 Files accessed

SYSUAF.DAT

extended to hold user interactive & batch quotas
(and credit/debit/usage for cluster wide records).

DISK0:[UEASYSTEM.JOB_MONITOR]USER_CREDIT.DAT

user interactive & batch credit by UIC.
Memory mapped file section USER_CREDIT .

DISK0:[UEASYSTEM.JOB_MONITOR]USER_DEBIT.DAT

user interactive & batch debits since start of week.
Memory mapped file section USER_DEBIT .

DISK0:[UEASYSTEM.JOB_MONITOR]USER_USAGE.DAT

user interactive & batch usage since start of week.
Memory mapped file section USER_USAGE .

DISK0:[UEASYSTEM.JOB_MONITOR]:LOGINDATA.DAT

student user login counts & labclass flags
also batch job queue priority & UEALOGIN accepted flag per process.
Memory mapped file section UEA_LOGINDATA .

DISK0:[UEASYSTEM.JOB_MONITOR]:PROCDATA.DAT

process data & various logical flags carried forward to new JOB_MONITOR on switching.
Memory mapped file section UEA_PROCDATA .

SYS$MANAGER:JOB_MONITOR.LOG;

Log file for tracing job logins/logouts & actions taken.
New file produced on switching.

DISK0:[UEASYSTEM.JOB_MONITOR]USER_JOB_HISTORY.LOG

Log file for user job records ... resources used & cost
New file produced on switching.

DISK0:[JOBBER]LABCLASS.DAT

student labclass timetable

2 General description

The main aim of the system on the research nodes is to limit the load placed on a busy machine to maintain interactive response but otherwise not to interfere with users rate of working . At the heart of the system is a permanently resident process JOB_MONITOR which collects system wide data & handles job charging in real time.

Additionally on the teaching nodes the same process is used to control student access according to 'class' or 'casual' status .(Student in this case means any user holding the rights identifier STUDENT & is independent of group UIC or username format.)

A separate process LOAD_MONITOR calculates a system 'demand factor' based on process counts in the various wait states. The demand factor is used to modify job costs in real time & hence users work rate.

For batch jobs the $SUBMIT queue priority replaces the demand factor as the job cost modifier to allow users some scheduling control.

2.1 User Quotas (Budgets) & Job costs

Each user is given two quotas additional to the VMS set. An interactive demand quota & a batch quota. These quotas are held in an extension to the standard VMS UAF record for the user. Interactive demand quota is added to the users current interactive credit each day with a credit limit of three times quota. Similarly the batch quota is added to the users batch credit but at weekly intervals.

During the life of a job the users credit (batch or interactive) is debited at one minute intervals. If an interactive user becomes overdrawn then the user is logged off after a minutes warning & is not allowed to log on again until the next day (when credit will again be positive) .

An overdrawn batch job is allowed to complete but further jobs are rejected until credit is positive either as a result of the weekly update or Budget Manager intervention .

Subprocesses are charged according to the mode of the parent job. Hence a subprocess of an interactive job is subject to the demand factor while a subprocess of a batch job is subject to the queue priority of the parent. All other processes are treated like batch jobs with a fixed priority . In general these other processes are created with UIC's outside the chargeable range (40-377) anyway. Users are currently prevented from creating detached processes by protection on SYS$SYSTEM:RUNDET.EXE. The UAF MAXDETACH control in VMS 5 is still not useful because MAXDETACH=NONE also prevents tape mounts !

.

Interactive & batch credits are held in a memory mapped file USER_CREDIT.DAT & accessed using the users UIC as an index. Users sharing a UIC will therefore share budgets. For historical reporting & manager information debits made in any week are also recorded in a similar section USER_DEBIT.DAT & basic resource units used in USER_USAGE.DAT; .

The basic cost of a job depends on processor time used,input/output counts & page faults. For interactive jobs an elapsed time element is added.

2.1.1 Budget control in a VAX CLUSTER

In a cluster the UAF records hold cluster wide CREDIT/DEBIT/USAGE in addition to QUOTA for both interactive & batch control.

If the JOB_MONITOR CLUSTER switch is ON then at process logout the UAF record for the Username corresponding to the current UIC is updated in the CREDIT/DEBIT/USAGE fields. The system wide login image SYSLOGIN will also check cluster wide credit from the UAF as well as current node credit in the global section.

The cluster wide interactive credit is updated each day using the command $BUDGET/GLOBAL_UPDATE/INTERACTIVE/CLUSTER on one of the nodes as well as the node specific credit. Similarly cluster wide batch credit is updated weekly. The commands are issued from UEA_JOBBER files.

The CLUSTER switch (in the PROCDATA.DAT section) is controlled using the $JSWITCH command (eg. $JSWITCH/CLUSTER/ON). If the switch is on and only a single node is in use then the UAF budget record will track the current node values (but will only match at process logout).

Budget control & reporting commands for the cluster wide values are the same as the node specific commands but with the added /CLUSTER qualifier.

2.2 Login checks

Credit checks at login are performed by SYSLOGIN.EXE from UEALOGIN.COM (the system wide login command procedure) . This program shares global sections LOGINDATA.DAT & PROCDATA.DAT with the JOB_MONITOR image (MONITOR.EXE) & also maps in read mode to the USER_CREDIT.DAT file section.
If a user login is successful at this stage a flag is set in LOGINDATA.DAT which JOB_MONITOR uses as a signal to start charging for the job. This overcomes problems associated with transient process data changes (eg. UIC) during VMS login & maintaining succcessful login counts.

3 JOB_MONITOR process

This is created as a detached process from the system start up file. The data collection routine runs in KERNAL mode and is locked into the working set.
It will reboot itself on request or crash (see JOB_MONITOR.COM for details).

JOB_MONITOR collects system wide data by direct reference to the VMS process control blocks in a similar fashion to the VMS $MONITOR facility.
The system service $GETJPI is not used because of the overheads associated with its mode of operation (delivering an AST interrupt to each process).
The macro routine UEAJPI.MAR replaces GETJPI . A FORTRAN equivalent using the $GETJPI service is available for comparison & testing.

System wide data is collected at 10 sec. intervals (see COLLECT.FOR) & actions based on that data are dealt with at one minute intervals in the main routine (see MONITOR.FOR).The data at 10 sec. intervals is interpreted in terms of logins & logouts & accumulated data over a minute for each resident process is passed back to the main routine.

Two phases of login are recognised, VMS & UEA . The JOB_MONITOR may see VMS login data for a process at any time from the Username? prompt, when the only thing known about the process is its internal identifier & terminal identifier . This login may then be rejected by VMS (eg. invalid password) or complete but be rejected by the UEALOGIN procedure (eg. credit overdrawn). For this reason SYSLOGIN.EXE & MONITOR.EXE communicate through the global section LOGINDATA.DAT to flag a successful login before any action is taken by the JOB_MONITOR.

Process data accumulated at the minute intervals is held in a global section PROCDATA.DAT & is therefore available to a new JOB_MONITOR process when switched. The JOB_MONITOR is normally switched once a day by command from UEA_JOBBER ($ASSIGN/SYSTEM SWITCH MON_SWITCH). In the unlikely event of a JOB_MONITOR crash (well what did you expect me to say !) or VMS crash close the next JOB_MONITOR process will clear the PROCDATA.DAT section. A crash close may also be requested ($ASSIGN/SYSTEM CRASH MON_SWITCH) or a normal shutdown ($ASSIGN/SYSTEM STOP MON_SWITCH) . Regular log files are purged to keep about one weeks data but crash events cause a rename to JOB_MONITOR.LOG_CRASH so these don't get automatically purged.

```
$!      JOB_MONITOR.COM VMS V5 VERSION OCT. 1989
$       SET VERIFY
$       ASSIGN/PRO SYS$DISK SYS$SCRATCH
$       SET DEF SYS$SPECIFIC:[SYSMGR]
$       SHOW TIME
$       SET PROCESS/PRIORITY=6
$       SET COMMAND UEA$SYSROOT:[COMMANDS]JSWITCH.CLD
$       JSWITCH ! Display current JOB_MONITOR switches.
$       ON WARNING THEN $ CONTINUE
$       PURGE/KEEP=3 JOB_MONITOR.LOG
$       ON WARNING THEN $ CONTINUE
$       SET PROT=(W) JOB_MONITOR.LOG
$       SET DEF UEA$SYSROOT:[JOB_MONITOR]
$       ON WARNING THEN $ CONTINUE
$       PURGE/KEEP=10 USER_JOB_HISTORY.LOG
$       ON WARNING THEN $ CONTINUE
$       SET PROT=(W)  USER_JOB_HISTORY.LOG
$       ON WARNING THEN $ CONTINUE
$       RUN MONITOR
$       EXIT_STATUS = $STATUS
$       SHOW SYMBOL EXIT_STATUS
$       SET DEF SYS$SPECIFIC:[SYSMGR]
$       IF EXIT_STATUS.EQ.9 THEN $ GOTO SWITCH_JOB
$       IF EXIT_STATUS.EQ.8 THEN $ GOTO STOP_JOB
$       GOTO CRASH_JOB
$SWITCH_JOB:
$       ON WARNING THEN $ CONTINUE
$       DEASSIGN/SYSTEM MON_SWITCH
$       PRCNAM := 'F$GETJPI("","PRCNAM")'
$       DOT = 'F$LOC(".",PRCNAM)'
$       LENGTH = 'F$LEN(PRCNAM)'
$       IF 'DOT'.EQ.'LENGTH' THEN $ GEN = 0
$       IF 'DOT'.NE.'LENGTH' THEN $ GEN = 'F$EXT(DOT+1,LENGTH,PRCNAM)'
$       GEN = GEN + 1
$       IF 'GEN'.GE.100 THEN $ GOTO STOP_JOB   ! limit to catch switching loop
$       ! Assume VMS will be rebooted before limit is reached legally .
$       PRCNAM := 'F$EXT(0,DOT,PRCNAM)'.'GEN'
$       RUN/UIC=[1,4]/PROCESS_NAME='PRCNAM'/DELAY="0 00:00:10.00"-
        /BUFFER_LIMIT=20480-
        /PAGE_FILE=20480-
        /INPUT=SYS$COMMON:[SYSMGR]JOB_MONITOR.COM-
        /OUTPUT=SYS$SPECIFIC:[SYSMGR]JOB_MONITOR.LOG SYS$SYSTEM:LOGINOUT
$       EXIT
$STOP_JOB:
$       ON WARNING THEN $ CONTINUE
$       DEASSIGN/SYSTEM  MON_SWITCH
$       JSWITCH/SYSANNOUNCE/OFF    ! stops UEALOGIN student checks
$       ASSIGN/SYSTEM ""F$LOG("MON$ANNOUNCE")'" SYS$ANNOUNCE    ! revert
display
$       EXIT
$CRASH_JOB:
$       RENAME JOB_MONITOR.LOG JOB_MONITOR.LOG_CRASH
$       ON WARNING THEN $ CONTINUE
$       MAIL/SUBJECT="JOB_MONITOR CRASH !" SYS$INPUT SYSTEM
 JOB_MONITOR crash  see   SYS$SPECIFIC:[SYSMGR]JOB_MONITOR.LOG_CRASH;*
$
$       GOTO SWITCH_JOB   ! try again SWITCH_ENTRY in PROCDATA will be false
$         ! causing PROCDATA to be cleared for fresh start.
```

4 $BUDGET commands

Command qualifiers allow Budget display by user or group & Budget changes by Budget Managers or system users. Group reports of interactive, batch or disk quotas & usage ,also job history reports are available in a scrollable VT100 window. These reports can also be sorted on any field.

Various images are involved with qualifier dependent syntax in the standard VMS command interface (see BUDGET.CLD the command definition file).
In all cases the relevant Username is determined by a UIC lookup in the RIGHTS database so that the UAF record consulted is unique in the case of Usernames sharing a UIC .
Budget Managers (UIC's [n,0]) can specify a /USER='username' qualifier to obtain information on users in their group.
System users (UIC [n,*] n =< 10) can specify any registered user in the same way.
The /GROUP qualifier for a system user means the world & for a Budget Manager means the group.

Access to the UAF is normally read only & shared , to obtain a users interactive demand quota or batch quota. In the case of new user set up then the users UAF record is locked while it is extended to hold the new budget fields. All UAF access (read serial,read indexed by username,read indexed by UIC,write with locking) is through macro routines in UEA_ACCESS.MAR

.

The command images map to the file section USER_CREDIT.DAT to read or write as appropriate a users current node credit.

Additional to accessing the new budgets the $BUDGET command provides a job history from the USER_JOB_HISTORY.LOG files and /SECURE or /INSECURE disk quota information in a similar fashion to the standard $SHOW QUOTA (through ACP/XQP QIO system services).

$BUDGET will also allow a disk quota update by Budget Managers (or System users) with group restrictions as above. This is done directly via ACP/XQP QIO without reference to the VMS DISKQUOTA utility.

Group reports of disk quota are produced by a serial read of the actual QUOTA.SYS file (which can be slightly out of date with respect to the live system).

Images for $BUDGET are $INSTALLed at system start up with appropriate privileges to allow access to the UAF,credit & disk quota data. Access rights are then determined by a UIC check in each image (eg. to forbid /USER= to ordinary users).

BUDGET.CLD

```
DEFINE SYNTAX BUDGET_QUERY
   IMAGE UEA$SYSROOT:[BUDGET]BUDGET_QUERY.EXE
DEFINE SYNTAX QUOTA_TRANSFER
   IMAGE UEA$SYSROOT:[BUDGET]QUOTA_TRANSFER.EXE
DEFINE SYNTAX CREDIT_TRANSFER
   IMAGE UEA$SYSROOT:[BUDGET]CREDIT_TRANSFER.EXE
DEFINE SYNTAX CREDIT_UPDATE
   IMAGE UEA$SYSROOT:[BUDGET]CREDIT_UPDATE.EXE
DEFINE SYNTAX CREDIT_REPORT
   IMAGE UEA$SYSROOT:[BUDGET]CREDIT_REPORT.EXE
DEFINE SYNTAX QUOTA_SET
   IMAGE UEA$SYSROOT:[BUDGET]SET_QUOTA.EXE
DEFINE SYNTAX CREDIT_SET
   IMAGE UEA$SYSROOT:[BUDGET]SET_CREDIT.EXE
DEFINE SYNTAX CLEAR_USAGE
   IMAGE UEA$SYSROOT:[JOB_MONITOR]CLEARSECT_USAGE.EXE
DEFINE SYNTAX CLEAR_DEBIT
   IMAGE UEA$SYSROOT:[JOB_MONITOR]CLEARSECT_DEBIT.EXE
DEFINE SYNTAX CLEAR_CREDIT
   IMAGE UEA$SYSROOT:[JOB_MONITOR]CLEARSECT_CREDIT.EXE
DEFINE VERB BUDGET
   IMAGE UEA$SYSROOT:[BUDGET]BUDGET_QUERY.EXE
   QUALIFIER GROUP_QUERY,VALUE,SYNTAX=CREDIT_REPORT
   QUALIFIER QUOTA_TRANSFER,SYNTAX=QUOTA_TRANSFER
   QUALIFIER CREDIT_TRANSFER,SYNTAX=CREDIT_TRANSFER
   QUALIFIER GLOBAL_UPDATE,SYNTAX=CREDIT_UPDATE
   QUALIFIER Q_SET,SYNTAX=QUOTA_SET
   QUALIFIER C_SET,SYNTAX=CREDIT_SET
   QUALIFIER U_CLEAR,SYNTAX=CLEAR_USAGE
   QUALIFIER D_CLEAR,SYNTAX=CLEAR_DEBIT
   QUALIFIER C_CLEAR,SYNTAX=CLEAR_CREDIT
   QUALIFIER INTERACTIVE
   QUALIFIER BATCH
   QUALIFIER DISK,VALUE
   QUALIFIER INSECURE
   QUALIFIER SECURE
   QUALIFIER USERNAME,VALUE(REQUIRED)
   QUALIFIER FROM_USER,VALUE(REQUIRED)
   QUALIFIER TO_USER,VALUE(REQUIRED)
   QUALIFIER TRANSFER,VALUE(REQUIRED)
   QUALIFIER HISTORY,VALUE
   QUALIFIER DEMAND
   QUALIFIER OUTPUT,VALUE(REQUIRED)
   QUALIFIER FULL
   QUALIFIER SORT,VALUE
   QUALIFIER SCREEN
   QUALIFIER CLUSTER
   QUALIFIER ASCENDING
   QUALIFIER DESCENDING
```

5 UEA_JOBBER/JOB_MONITOR/UEALOGIN & Student Classes

The UEA_JOBBER process is permanently resident and at each hour of each day executes a command file HOURLY.COM .This file obeys LABCLASS.COM which runs LABCLASS.EXE. The program reads the timetable file LABCLASS.DAT & sets/unsets the requested group flags in LOGINDATA.DAT according to each groups class status.

Groups may be defined by UIC or by rights identifiers in the special range %X800A0001 to %X800A0064 (MAXRIGHTS). The high word 800A is used as a key to distinguish 'student class identifiers' from other identifiers.Each user can hold up to 10 (MAX_USER_RIGHTS) of these identifiers.

The information is used by JOB_MONITOR to maintain a count of students in and out of lab. classes these counts being saved in the same LOGINDATA.DAT section. The system wide login procedure UEALOGIN.COM runs SYSLOGIN.EXE which accesses LOGINDATA.DAT to enforce student logged in limits. These student limit checks may be switched off using the JOB_MONITOR switch SYSANNOUNCE with the $JSWITCH command (which also disables the update of SYS$ANNOUNCE information).

A command $STUDENT (in UEATABLES.EXE along with $BUDGET) allows read access to the labclass flags & logged in counts & limits. System users can change the casual student limit & reset any groups labclass flag. Because this data is held in a global section it is preserved over a VMS reload. The $STUDENT command runs various images depending on qualifiers present (see STUDENT.CLD for the command definition).

JOB_MONITOR gives priority to students in a scheduled class by increasing process base priority from 4 to 5. The priority is reset when the class ends.

A student login is always accepted (up to VMS SET/LOGINS=) if the group has a class in progress. Casual logins are allowed if the total population is below the $STUDENT/CASUAL= limit . A class login will displace a casual user if this limit is exceeded and any casual user logged out in this way is given a one minute warning .

6 SYS$ANNOUNCE display

This is maintained by JOB_MONITOR (see UEA_ANNOUNCE.FOR). For all nodes the login count & demand factor is appended to the preset SYS$ANNOUNCE from the system startup file. For teaching nodes the student login counts are also displayed. (IF System logical TEACHING_NODE = TRUE).

The update of SYS$ANNOUNCE is also controlled by the JOB_MONITOR switch SYSANNOUNCE which can be toggled using the $JSWITCH command. If SYSANNOUNCE is false then the display is not updated & also the SYSLOGIN.EXE image run in UEALOGIN does check student class/counts at login.

7 LOAD_MONITOR

This process is run detached at system start up & is restarted once a day by commands from a UEA_JOBBER file. It runs LOADING.EXE to collect process wait state counts using code run in KERNAL mode to directly access the VMS process control blocks . Data is collected at 5 sec. intervals to produce a demand factor averaged over five minutes at one minute intervals.

The demand factor is written to a system logical UEA$LOAD & is therfore easily available to any other process.

8 Brief description of source files

Directory DISKB:[UEA_SOFTWARE.MONITOR]

ACCOUNT.FOR;12 3-APR-1986 15:52:10.67

Subroutine to send budget record to VMS accounting file
... not used currently.

ALLJOBS.FOR;2 28-JUN-1989 11:54:05.51

Program to list JOB_MONITOR process data held in PROCDATA.DAT

CLASS_CHECK.FOR;2 17-AUG-1987 12:23:51.72

Determines class/student status of process from UIC group flags & rights identifiers held.

CLEARSECT_CREDIT.FOR;1
7-OCT-1986 17:26:23.65

Program to clear global section USER_CREDIT.DAT, ie zero all elements. Emergency use only ! Update section according to quotas with $BUDGET commands afterwards.

CLEARSECT_DEBIT.FOR;1
1-SEP-1987 17:35:02.20

Program to clear global section USER_DEBIT.DAT, ie zero all elements & date stamp . Normally run weekly by UEA_JOBBER process .

CLEARSECT_USAGE.FOR;1
1-SEP-1987 17:33:42.79

Program to clear global section USER_USAGE.DAT, ie zero all elements & date stamp. Normally run weekly by UEA_JOBBER process .

COLLECT.FOR;1 1-NOV-1989 11:01:34.60

Subroutine to collect system wide process data (see UEAJPI) Handles logins,logouts & maintains login counts etc. at 10s. intervals. Returns to main routine at one minute intervals with accumulated process data for charging etc.

DEBIT.FOR;1 14-APR-1989 10:08:18.65

Subroutine to update users credit according to resources used & demand factor(interactive) or queue priority(batch) Called by main routine at one minute intervals & at process logout/endjob. Broadcasts message to users who become overdrawn.

ELAPSED.FOR;1 17-MAR-1989 13:59:12.48

Subroutine to calculate an elapsed time in minutes between two absolute system times.

EXCEPTION_HANDLER.FOR;2
1-NOV-1989 10:21:19.32

Subroutine to unwind back to MAIN in the event of a fatal error.

GET_CREPROC_COUNT.FOR;1
2-JUN-1988 12:33:50.79

Subroutine to collect count of subprocesses created by nominated process.

GET_ID_USER.FOR;1 18-JUN-1986 13:06:03.37

Subroutine to do UIC to Username translation.

GET_PROC_COUNT.FOR;1
29-SEP-1986 14:56:04.51

Collects the no. of processes owned by specified job

GET_RIGHTS.FOR;3 1-NOV-1989 11:08:49.24

Uses $FIND_HELD to set up a list of rights identifiers held by the specified UIC of type CLASS_ID (high order word 800A) from the rights database RIGHTSLIST.DAT

Used independently by SYSLOGIN.FOR & UEALOGIN_CHECK in JOB_MONITOR.

IDLE_TIMEOUT.FOR;1 9-DEC-1988 10:11:23.62

Checks for idle process timeout based on buffered i/o & cpu usage. Subprocess trees are recognised . Two levels of inactive periods are available depending on rights ident. ALT_TIMEOUT .

INITSECT_CREDIT.FOR;1
7-OCT-1986 17:11:49.19

Program to create/initialise global section USER_CREDIT.DAT
Only needed for a fresh start ! Otherwise use CLEARSECT_CREDIT followed by $BUDGET/CREDIT_UPDATE commands

INITSECT_DEBIT.FOR;2
7-OCT-1986 17:18:49.73

Program to create/initialise global section USER_DEBIT.DAT
Only needed for a fresh start ! Otherwise use CLEARSECT_DEBIT

INITSECT_USAGE.FOR;2
3-MAR-1987 15:33:02.10

Program to create/initialise global section USER_USAGE.DAT
Only needed for a fresh start ! Otherwise use CLEARSECT_USAGE

JOB_MONITOR_SWITCHES.FOR;1
15-AUG-1989 17:06:45.15

Program to list or change state of JOB_MONITOR
logical switches for SYSANNOUNCE,BUDGET_LOGOUT,
TERMINAL_TIMEOUT,LAB_PRIORITY,LOGOUT,CLUSTER

Implements $JSWITCH command

MAIN.FOR;2 1-NOV-1989 10:15:10.72

Master routine of JOB_MONITOR calls MONITOR.

MONITOR.FOR;1 1-NOV-1989 15:38:03.11

Main routine of JOB_MONITOR .
Maps global sections USER_CREDIT.DAT etc.
Collects demand factor UEA$LOAD from system logical table
Calls COLLECT for process data over 1 min. intervals
Handles forced logouts for
excess casual students
overdrawn users
idle terminals
Controls switching & reboot action etc.

QNAME.FOR;2 6-MAR-1987 17:31:07.19

Subroutine to get batch queue name from current job.
Called by system wide login program SYSLOGIN.EXE
from user process.

QPRIORITY.FOR;4 6-MAR-1987 17:30:22.57

Uses SYS$GETQUI to determine current batch jobs
SUBMIT/PRIORITY. Called by system wide login program
SYSLOGIN.EXE.

RMSOPEN_ERROR.FOR;2
2-NOV-1989 11:06:07.94

Subroutine to display RMS error reply .

SETSECT_USAGE.FOR;2
9-MAR-1987 12:51:50.66

Program to set a specified UIC usage in USER_USAGE.DAT

SETUP_RIGHTS.FOR;3 10-NOV-1987 14:30:40.18

Uses information from GET_RIGHTS to set up a process rights list in PROCDATA.DAT

SHELLSORT.FOR;1 25-FEB-1983 15:18:46.41

Subroutine to do a shell sort on specified data

STUDENTINFO.FOR;11 11-OCT-1989 12:09:05.17

Program to list JOB_MONITOR data on logged in student users ... class membership etc.

Implements $STUDENT/LOGINS command

STUDENT_CHECK.FOR;1
9-DEC-1988 10:16:30.48

Subroutine to check student login counts for excess casual students. Uses global section LOGINDATA.DAT for login counts & class flags. Broadcasts logout warning .

SYSJOBS.FOR;4 7-APR-1989 16:10:20.80

Program to list 'system' jobs data held by JOB_MONITOR

Implements $GET/SJ command

UEAJPI.FOR;29 28-JUN-1989 10:55:12.45

JOB_MONITOR data collection routine using $GETJPI ... alternative DEC approved method but slower.

UEALOGIN_CHECK.FOR;1
9-DEC-1988 10:17:45.58

Subroutine to determine process accepted by UEALOGIN from data in global section LOGINDATA.DAT set up by SYSLOGIN.EXE. Queue priority for batch jobs also .

UEA_ANNOUNCE.FOR;5 7-AUG-1987 11:43:36.84

Subroutine to reset logical SYS$ANNOUNCE according to login counts & Demand factor. A test for the system logical name TEACHING_NODE determines whether the student class/casual count is appended.

USERINFO.FOR;1 7-APR-1989 17:06:12.54

Program to list JOB_MONITOR process data held in
PROCDATA.DAT ... excludes 'system' jobs.

Implements $GET/UD command.

USERJOBS.FOR;2 7-APR-1989 16:59:08.08

Program to list JOB_MONITOR entries for 'user'
jobs.

Implements $GET/UJ command.

UEAJPI.MAR;2 1-NOV-1989 14:59:47.06

Macro routine to collect system wide process data.
Direct access to VMS PCB's. Avoids the process
interrupt method of system service $GETJPI.

USEROPEN.MAR;1 2-NOV-1989 11:51:02.91

Macro routine for open/map to global sections USER_CREDIT.DAT
, USER_DEBIT.DAT & USER_USAGE.
Also collects RAB for USER_JOB_HISTORY.LOG $FLUSH

Directory DISKB:[UEA_SOFTWARE.LOAD_MONITOR]

LOADING.FOR;1 17-APR-1989 17:14:02.72

Program to calculate demand factor based
on process wait states & counts.

LOADPLOT.FOR;1 13-JUN-1986 12:50:24.34

Simple plotting routine for demand factor.

STATES.FOR;1 15-MAY-1989 15:28:32.78

Process states data collection subroutine
used by LOADING.FOR & based on $GETJPI
... alternative to STATES.MAR but slow.

STATES.MAR;1 27-JUN-1989 12:07:46.98

Process states data collection subroutine
used by LOADING.FOR . Collects data by direct
reference to PCB's.

Directory DISKB:[UEA_SOFTWARE.BUDGET_UTILITIES]

BUDGET_HISTORY.FOR;2
17-AUG-1989 17:54:41.57

Subroutine to list user job records from USER_JOB_HISTORY.LOG files.

BUDGET_QUERY.FOR;1 26-AUG-1988 15:00:27.97

Program for User & Manager Budget query

BUDGET_SET.FOR;1 26-MAY-1988 17:13:25.34

Main routine for Quota/Credit setting operations

BUDGET_TRANSFER.FOR;1
26-AUG-1988 15:11:30.89

Routine to transfer UAF Quota or Credit .
(Cluster wide values).

CLEARUAF_USAGE_DEBIT.FOR;1
18-MAR-1988 17:12:06.06

Program to clear usage & debit fields in UAF budget record.

CLEARUAF_XQUOTA.FOR;1
18-MAR-1988 17:08:41.05

Program to clear all fields in UAF budget record bar quota.
(ie. credit,debit,usage)

CREDIT_REPORT.FOR;1
1-SEP-1988 10:46:57.52

Program to produce interactive or batch or
disk (secure or insecure) user credit report.
Global for System Manager (UIC grp. < 10)
Group for Budget Manager (UIC mem. = 0)

CREDIT_TRANSFER.FOR;1
26-AUG-1988 15:08:28.89

Program to transfer interactive or batch credit
between users .
Global for System Manager.
Group for Budget Manager.

CREDIT_UPDATE.FOR;1
26-MAY-1988 17:27:51.85

Program to update all users credit by quota from UAF
System Manager function only.
Currently run by UEA_JOBBER once per day for interactive
& once per week for batch. Access is by $BUDGET command.

DISK_QUERY.FOR;2   20-SEP-1989 15:10:19.72

Subroutine to report user disk credit (secure/insecure)
Called by BUDGET_QUERY if $BUDGET/DISK

DISK_SET.FOR;9   12-NOV-1987 12:28:39.60

Subroutine to reset disk quota file entry.

DISK_TRANSFER.FOR;1
29-JUN-1988 14:59:09.76

Subroutine to transfer disk quota between users.
Global for System Manager
Group for Budget Manager
Called by QUOTA_TRANSFER if $BUDGET/DISK

DQ_READ.FOR;3   24-OCT-1986 15:09:54.39

Subroutine to read a disk quota entry by QIO to ACP/XQP

DQ_WRITE.FOR;3   24-OCT-1986 16:42:43.61

Subroutine to write a disk quota entry by QIO to ACP/XQP

GROUP_DISK_REPORT.FOR;1
13-OCT-1989 13:53:37.07

Subroutine to produce group disk report(secure/insecure)
for the calling process UIC group.
Called by CREDIT_REPORT
System Manager with $SET UIC or $BUDGET/GROUP=
Budget Manager for own group.

SYSTEM $BUDGET/GROUP/DISK= gives global report

GROUP_SORT.FOR;1   18-AUG-1989 10:58:14.01

Subroutines to set up & sort records from group reports
Interactive,Batch,Disk or Job history. Also includes set up
for screen (VT100+) output to allow scrolling of user records
in a preset window.

QUOTA_TRANSFER.FOR;1
18-MAR-1988 11:03:45.64

Program to transfer interactive,batch or disk quota between
users.
Global for System Manager
Group for Budget Manager

READ_CREDIT.FOR;1   2-MAY-1986 17:17:05.84

Subroutine to read users interactive or batch credit from
global section USER_CREDIT.DAT

READ_DEBIT.FOR;3 19-MAR-1987 17:26:45.43

Subroutine to read users interactive or batch debit from
global section USER_DEBIT.DAT

READ_USAGE.FOR;3 19-MAR-1987 17:24:20.08

Subroutine to read users interactive or batch usage from
global section USER_USAGE.DAT

SET_CREDIT.FOR;1 17-MAR-1988 15:43:31.14

Program to set credit of specified user
System Manager function only in special circumstances .

SET_QUOTA.FOR;1 17-MAR-1988 15:22:42.53

Program to set quota of specified user.
System Manager function only in special circumstances .
eg creation of new Budget Manager allocation.

SYSTEM_DISK_REPORT.FOR;1
13-OCT-1989 13:55:58.59

Procedure to produce a system level report on group
disk usage.

UAFPARAMS.FOR;5 4-FEB-1988 15:56:41.18

INCLUDE file for UAF record field offset definitions

UAF_BUDGET_CHECK.FOR;1
15-AUG-1989 12:44:06.86

For use by SYSLOGIN.EXE/UEALOGIN to check cluster wide credit
in UAF entry. UIC look up in rights database gives the correct
Username to check in the case of shared UIC accounts.

UAF_BUDGET_READ.FOR;1
14-AUG-1989 17:40:31.53

For use by JOB_MONITOR/UEALOGIN in Cluster to read UAF
credit debit & usage fields.

UAF_BUDGET_RESET.FOR;1
18-MAR-1988 10:47:46.54

Routine to reset selected budget fields in all UAF records .

UAF_BUDGET_UPDATE.FOR;1
18-MAR-1988 17:02:30.62

For use by JOB_MONITOR in Cluster to update UAF credit & usage
fields at job logout.

UAF_REC_CHECK.FOR;1
16-MAR-1988 12:07:10.44

Fortran routine used with UAF_ACCESS to expand UAF record
if budget field to be added.

WRITE_CREDIT.FOR;1 2-MAY-1986 17:23:04.84

Subroutine to reset user interactive or batch credit
in USER_CREDIT.DAT global section.

QUOTAOPEN.MAR;1 25-OCT-1989 16:55:37.77

Macro useropen routine for QUOTA.SYS files

UAF_ACCESS.MAR;1 16-AUG-1989 10:12:10.14

Macro subroutines to access UAF records in either of
sequential or indexed by username or UIC.

Directory DISKB:[UEA_SOFTWARE.LABCLASS]

CLASS_RIGHTS.FOR;1 9-JUN-1989 17:21:52.45

Program to display class identifiers held
$STUDENT/GROUP command.

LABCLASS.FOR;1 30-MAR-1989 14:36:21.06

Program to read LABCLASS.DAT timetable file & set
corresponding UIC group or Rights identifier flags in
LOGINDATA.DAT

SET_CASUAL.FOR;1 30-MAR-1989 14:37:53.69

Program to set casual student limit
Activated by $STUDENT/CASUAL command

SET_CLASS.FOR;1 30-MAR-1989 14:39:14.58

Program to set/unset UIC group or Rights ident. labclass
flags. Activated by $STUDENT/CLASS
Used to override UEA_JOBBER/LABCLASS.EXE action if required.

WHATCLASS.FOR;1 1-NOV-1988 10:15:26.13

Program to report casual student limit & class & casual
logins. Also lists labclass flags set .
Available through $STUDENT command to all users

DELESECT_LOGINDATA.FOR;1
4-MAY-1990 11:40:57.27
Program to delete permanent section UEA_LOGINDATA.

INITSECT_LOGINDATA.FOR;4
8-MAY-1990 16:56:01.32
Program to create & clear section file LOGINDATA.DAT
& install section.

INSTALL_LOGINDATA.FOR;2
9-MAY-1990 10:56:33.29
Program to install existing section file at system boot.

MAPSECT_LOGINDATA.FOR;4
4-MAY-1990 14:41:29.23
Subroutine to map to UEA_LOGINDATA section file.

Directory DISKB:[UEA_SOFTWARE.MONITOR]

BRKDEF.CMN;5 14-APR-1989 09:39:39.56

INCLUDE file for broadcast system service definitions

MAXPROCS.CMN;2 21-NOV-1986 10:43:13.75

INCLUDE file for max. UIC values etc. to determine array
bounds.

PROCDATA.CMN;22 15-AUG-1989 16:57:14.59

COMMON data for global section PROCDATA
which holds JOB_MONITOR process data collected.
INCLUDE file for PROCDATA common blocks.

UEAPARAMS.CMN;24 12-SEP-1989 17:18:03.84

INCLUDE file to define 'UEA' parameters such as
timeout intervals etc.

DELESECT_PROCDATA.FOR;2
4-MAY-1990 14:44:30.42
Program to delete permanent section UEA_PROCDATA.

INITSECT_PROCDATA.FOR;4
8-MAY-1990 17:08:54.44
Program to create & clear section file PROCDATA.DAT
& install section.

INSTALL_PROCDATA.FOR;2
9-MAY-1990 10:46:00.22
Program to install existing section at system boot.

MAPSECT_PROCDATA.FOR;1
4-MAY-1990 10:49:16.60
Subroutine to map to UEA_PROCDATA section.

Directory DISKB:[UEA_SOFTWARE.LABCLASS]

LOGINDATA.CMN;5 17-AUG-1987 12:06:40.49

COMMON data for global section LOGINDATA
which holds login information & student counts/class flags
INCLUDE file for LOGINDATA common blocks

Directory DISKB:[UEA_SOFTWARE.SYSLOGIN]

CHECK_BUDGET.FOR;1 16-AUG-1989 11:48:27.78

Subroutine to check node & cluster credit of user.

CLASS_LOGIN_CHECK.FOR;1
17-DEC-1987 14:56:22.93

Subroutine to check class status of student users.

SYSLOGIN.FOR;4 13-OCT-1989 10:53:24.23

System wide login program run by UEALOGIN.COM
(SYS$SYLOGIN)

9 Parameters controlling JOB_MONITOR actions

```
C   Idle process timeout/logout parameters (mins.)

     PARAMETER(TWARNING=10,TLOGOUT=15)
     PARAMETER(ALT_TWARNING=10,ALT_TLOGOUT=15)

C   UIC group control parameters ... JOB_MONITOR & $BUDGET utilities

     PARAMETER(MINACT='30'O)            ! minimum UIC group for JOB_MONITOR actions
     PARAMETER(MIN_CHARGED_GROUP='41'O) ! for JOB_MONITOR charging
     PARAMETER(SYS_REPORT_GROUP='15'O)  ! max. UIC groups allowed 'system' reports from
$BUDGET/GROUP

     PARAMETER(SYS_UIC='10'O)             ! system UIC limit

C   Limits on JOB_MONITOR/BUDGET field values

     PARAMETER(QUOTA_LIMIT=30000)            ! UAF     16 bit
     PARAMETER(CREDIT_LIMIT=-30000)        ! Credit section    16 bit
     PARAMETER (DEBIT_LIMIT=30000)         ! Debit section     16 bit
     PARAMETER (USAGE_LIMIT=9999999)        ! Usage section     32 bit
     PARAMETER(USER_MAX_CREDIT=10000)        ! $BUDGET/CREDIT_TRANSFER


C   Job cost coefficients

     PARAMETER(C_CPU=100,C_BUFIO=100,C_DIRIO=100,C_PFLTS=200)
         i.e. cost = 1 per 100 cpu ticks , 1 per 100 buffered i/o etc.

C   Rights identifiers

     PARAMETER( MAXRIGHTS = 100 )         ! total of type CLASS_ID
     PARAMETER( MAX_USER_RIGHTS = 10 )      ! max held per user proc.

C   masks on high word of rights id's associated with JOB_MONITOR

     PARAMETER(CLASS_ID='800A'X)           ! mask to select class id's.
     PARAMETER(MONITOR_ID='800B'X)           ! mask to select other id's.

     PARAMETER(STUDENT_ID='800A0001'X)          !  id for 'STUDENT' status
     PARAMETER(ALT_TIMEOUT_ID='800B0032'X)  !  id for alternative idle timeout ( also clears screen at
logout )
     PARAMETER(OVERDRAFT_ID='800B0031'X)    !  id for no overdrawn budget action
     PARAMETER(MULTIPLE_LOGIN_ID='800B0030'X)!  id to allow multiple logins
     PARAMETER(NO_TIMEOUT_ID='800B0029'X)   !  id for no idle timeout
     PARAMETER(NO_SUB_TIMEOUT_ID='800B0028'X)!  id for no subprocess timeout
C   Main process & subprocess tree may be deleted if all idle even if user holds NO_SUB_TIMEOUT .
```