UPDATE INFORMATION FILE -- REV 18

THIS UFD CONTAINS ALL SOFTWARE UPDATES GENERATED AFTER THE INITIAL REV 18 RELEASE TO THE FIELD. INFORMATION ABOUT ALL PREVIOUS UPDATE RELEASES SINCE THE INITIAL RELEASE IS ALSO PRESENTED IN THIS FILE. THE INITIAL REV 18 RELEASE WAS 18.1.

DUE TO THE REORGANIZATION OF MUCH OF PRIME'S SOFTWARE, THE FOLLOWING PROCEDURE SHOULD BE FOLLOWED WHEN UPDATING A REV. 13.1 MASTER DISK.

IN UPDATE RELEASES OF REV 18, SOME SOURCE FILENAMES MAY HAVE SUFFICES WHICH WERE NOT THERE AT REV 18.1. SINCE THE FILENAMES ARE DIFFERENT IN THESE CASES, THE USER SHOULD DELETE THE OLD DIRECTORIES TO AVOID HAVING UNNECESSARY FILES ON THEIR DISK. IN ADDITION, MANY CHARGEABLE PRODUCTS NOW HAVE THEIR SOURCE AND RUN FILES IN SEPARATE DIRECTORIES. FOR EXAMPLE, AT REV 18.1, THE DIRECTORY BASIC CONTAINED BOTH THE RUN FILES AND THE SOURCE FILES FOR THE PRODUCT. AT REV. 18.2, THERE IS A DIRECTORY BASIC WHICH CONTAINS THE RUN FILES, AND A DIRECTORY BASICSRC WHICH CONTAINS THE SOURCE. IF A USER UPDATES FROM 18.1 TO 18.2 WITHOUT DELETING THE OLD DIRECTORIES, HE ENDS UP WITH THE SOURCE OF BOTH RELEASES.

BEGIN WITH A COMPLETE REV 18.1 DISK. RESTORE THE FILE UPINFO18.X FROM YOUR REV. 18.X UPDATE TAPE. THIS FILE SHOULD BE THE FIRST LOGICAL TAPE. USING THE UPINFO18.X FILE AS A GUIDE, DELETE FROM YOUR 18.1 DISK ANY DIRECTORIES WHICH ARE BEING REPLACED AT 18.X. ENTRIES IN THE UPINFO18.X FILE WHICH ARE DIRECTORIES ARE LABELLED (DIRECTORY). THESE ARE THE ONES WHICH THE USER SHOULD DELETE PRIOR TO RESTORING HIS UPDATE TAPE. THERE IS NO NEED TO DELETE THE FILES BEING REPLACED IN SYSTEM DIRECTORIES (CMDNC0,SYSTEM,LIB,ETC.) ON THE <M181A1>MFD SINCE THEY WILL BE OVERWRITTEN WHEN THE TAPE IS RESTORED.

FOLLOWING UPINFO 18.X ON THE UPDATE TAPE ARE LOGICAL TAPES A1,B1, AND B2. TO UPDATE YOUR MASTER DISK, RESTORE A1 ONTO YOUR MASTER DISK PARTITION MXXXA1, RESTORE B1 ONTO PARTITION MXXXB1, AND RESTORE B2 ONTO PARTITION MXXXB2. THE TAPE WILL OVERWRITE THE EXISTING FILES THAT ARE ON YOUR DISK. IF YOU DO NOT HAVE THREE SEPARATE DISK PARTITIONS ON YOUR MASTER DISK, RESTORE THE TAPE ONTO THE PARTITION WHERE THE FILES YOU ARE UPDATING CURRENTLY EXIST.

MASTER PARTS LIST FOR THE MASTER DISK FOR REV. 18.3

PART	MASTER DISK UFD
8525	BASIC, BASICSRC
8520	BASICV, BASICVSRC
8540	COBOL, COBOLLIBSRC
8538	DBG

```
8550      DBMS (OBSOLETE PART NUMBER INCLUDES:)
          DBMSDEF, DBMSDEFBIN, DBMSFTN, DBMSFTNBIN,
          DBMSCOB, DBMSCOBBIN, DBMSLGCL, DBMSLGCLBIN,
          DBMSEX, DBMSEXBIN
8640      DBMSDEF, DBMSDEFBIN, DBMSEX, DBMSEXBIN
8641      DBMSFTN, DBMSFTNBIN
8642      DBMSCOB, DBMSCOBBIN
8643      DBMSLGCL, DBMSLGCLBIN
8644      DBMSEX, DBMSEXBIN
8649      VISTA
8650      DBMSDEF, DBMSDEFBIN, DBMSFTN, DBMSFTNBIN,
          DBMSCOB, DBMSCOBBIN, DBMSLGCL, DBMSLGCLBIN,
          DBMSEX, DBMSEXBIN, VISTA
8410      DPTX-DSC
8430      DPTX-TCF
8420      DPTX-TSF
8510      F77
8548      FORMS
8162      FORMS, FED
8163      FORMS, FED (UPGRADE)
8164      FORMS, FED, AND PT45 TERMINAL
8515      FTN, FTNSRC
8555      MIDAS, MIDASSRC
8522      PASCAL
8523      PASCAL AND PASLIBSRC
8532      PL1 - NOT CURRENTLY AVAILABLE
8530      PL1G
8562      POWERPLUS
8440      PRINET
8180      RJE1004
8460      RJE7020
8020      RJEX80
8470      RJEGRTS
8120      RJEHASP
8060      RJE200UT
8546      RPG, RPGSRC, RPGLIBSRC
8450      X.25
8610      WPS
8620      OAS
8630      ATM
8631      ENGLDICT
8632      NORWDICT
8633      FRCHDICT
8634      GERMDICT
8635      SPANDICT

8100 INCLUDES ALL DIRECTORIES LISTED BELOW

ACCEPT          APPLIB          AVAIL           BASINP
BATCH           BATCHQ          BOTSRC          CMDNCO
CMPF            CONCAT          CPLDEMO         CPMPC
CRMPC           CX              CX***           DIRECV
DOS             ED              EDB             FILMEM
FILVER          FIXRAT          FUTIL           INDEX
```

```
       INFO18           LABEL            LATE             LIB
       LIB7             LOAD             LOGPRT           MAGNET
       MAGSR            MATHLB           MDL
       MSORTS           PHYSR            PMA              PRIMOS
       PRIMOS2          PRIRUN           PRMPC            PRSER
       PRVER            PSD              PTCPY            RFTNLIB
       RJECOM           RUNOFF           SEG              SEGSRC
       SIZE             SLIST            SPOOL            SPOOLQ
       SRTLIB           SYSCOM           SYSOVL           SYSTEM
       TERM             TOOLS            TRAMLC           UII
       UPCASE           VFTNLIB          VMSORT           VPSD
       VSRTLI                            HELP*

       ACCEPT, T&M, T&MSRC, T&MSR1, TMS400


*
*
****  REV 18.2  MARCH 23, 1981  ****
*
NAME                DIRECTORY                   SCN NO.          DATE

FIXRAT              SUPERCEDED BY 18.3  (SCN NO.  652)
COPY_DISK           <MXXXA1>CMDNC0 (RUN)          565            032381
COPY                <MXXXA1>CMDNC0 (RUN)          565            032381
DEREMER             SUPERCEDED BY 18.3   (SCN NO. 647)
ED                  <MXXXA1>ED (DIRECTORY)        567            032381
ED                  <MXXXA1>CMDNC0 (RUN)          567            032381
NSED                <MXXXA1>CMDNC0 (RUN)          567            032381
ED2000              <MXXXA1>SYSTEM (RUN)          567            032381
EDB                 <MXXXA1>EDB (DIRECTORY)       568            032381
EDB                 <MXXXA1>CMDNC0 (RUN)          568            032381
*LIBEDB             <MXXXA1>LIB (RUN)             568            032381
LOAD                <MXXXA1>LOAD (DIRECTORY)      569            032381
LOAD                <MXXXA1>CMDNC0 (RUN)          569            032381
LOGPRT              <MXXXA1>LOGPRT (DIRECTORY)    570            032381
LOGPRT              <MXXXA1>TOOLS (RUN)           570            032381
LOGPRT.COMI         <MXXXA1>LOGPRT (COMMAND)      570            032381
MAGSR               SUPERCEDED BY 18.3    (SCN NO. 651)
MAGSAV              SUPERCEDED BY 18.3    (SCN NO. 651)
MAGRST              SUPERCEDED BY 18.3    (SCN NO. 651)
PLP                 SUPERCEDED BY 18.3    (SCN NO. 654)
PRIMOS              SUPERCEDED BY 18.3    (SCN NO. 656)
PRIRUN              SUPERCEDED BY 18.3    (SCN NO. 656)
RFTNLIB             <MXXXA1>RFTNLIB (DIRECTORY)   574            032381
FTNLIB              <MXXXA1>LIB (BINARY)          574            032381
SVCLIB              <MXXXA1>LIB (BINARY)          574            032381
RUNOFF              <MXXXA1>RUNOFF (DIRECTORY)    575            032381
RUNOFF              <MXXXA1>CMDNC0 (RUN)          575            032381
SEG                 SUPERCEDED BY 18.3    (SCN NO. 659)
SEGSRC              SUPERCEDED BY 18.3    (SCN NO. 659)
SEG                 SUPERCEDED BY 18.3    (SCN NO. 659)
SHARE4              SUPERCEDED BY 18.3    (SCN NO. 659)
```

```
SPL                 SUPERCEDED BY 18.3    (SCN NO. 662)
SPOOL               SUPERCEDED BY 18.3    (SCN NO. 663)
SPOOLQ              SUPERCEDED BY 18.3    (SCN NO. 663)
SPOOL               SUPERCEDED BY 18.3    (SCN NO. 663)
PROP                SUPERCEDED BY 18.3    (SCN NO. 663)
SPOOL$              SUPERCEDED BY 18.3    (SCN NO. 663)
VSPOO$              SUPERCEDED BY 18.3    (SCN NO. 663)
SPS                 <MXXXA1>INDEX>SPS (DIRECTORY) 579            032381
VFTNLIB             SUPERCEDED BY 18.3    (SCN NO. 664)
IFTNLB              SUPERCEDED BY 18.3    (SCN NO. 664)
PFTNLB              SUPERCEDED BY 18.3    (SCN NO. 664)
NPFTNLB             SUPERCEDED BY 18.3    (SCN NO. 664)
S4000               SUPERCEDED BY 18.3    (SCN NO. 664)
S2050               SUPERCEDED BY 18.3    (SCN NO. 664)
ONCODES.P           SUPERCEDED BY 18.3    (SCN NO. 664)
ONCODES.PL1         SUPERCEDED BY 18.3    (SCN NO. 664)
ONCODES.INS.PMA     SUPERCEDED BY 18.3    (SCN NO. 664)
ONCODES.INS.PL1     SUPERCEDED BY 18.3    (SCN NO. 664)
BASIC               <MXXXB1>BASIC    (DIRECTORY)  581            032381
BASICSRC            <MXXXB1>BASICSRC (DIRECTORY)  581            032381
BASICV              <MXXXB1>BASICV (DIRECTORY)    582            032381
BASICVSRC           <MXXXB1>BASICVSRC (DIRECTORY) 582            032381
COBOL               SUPERCEDED BY 18.3    (SCN NO. 665)
COBOLLIBSRC         SUPERCEDED BY 18.3    (SCN NO. 665)
DBG                 <MXXXB1>DBG (DIRECTORY)       584            032381
DPTX-DSC            SUPERCEDED BY 18.3    (SCN NO. 671)
DPTX-TCF            SUPERCEDED BY 18.3    (SCN NO. 672)
DPTX-TSF            SUPERCEDED BY 18.3    (SCN NO. 673)
FED                 SUPERCEDED BY 18.3    (SCN NO. 674)
FORMS               SUPERCEDED BY 18.3    (SCN NO. 674)
FORMS               SUPERCEDED BY 18.3    (SCN NO. 675)
FTN                 SUPERCEDED BY 18.3    (SCN NO. 676)
FTNSRC              SUPERCEDED BY 18.3    (SCN NO. 676)
MIDAS               SUPERCEDED BY 18.3    (SCN NO. 678)
MIDASSRC            SUPERCEDED BY 18.3    (SCN NO. 678)
PRINET              SUPERCEDED BY 18.3    (SCN NO. 679)
RPG                 SUPERCEDED BY 18.3    (SCN NO. 686)
RPGSRC              SUPERCEDED BY 18.3    (SCN NO. 686)
RPGLIBSRC           SUPERCEDED BY 18.3    (SCN NO. 686)
X.25                SUPERCEDED BY 18.3    (SCN NO. 687)
DBMSDEF             SUPERCEDED BY 18.3    (SCN NO. 667)
DBMSDEFBIN          SUPERCEDED BY 18.3    (SCN NO. 667)
DBMSEX              SUPERCEDED BY 18.3    (SCN NO. 667)
DBMSEXBIN           SUPERCEDED BY 18.3    (SCN NO. 667)
DBMSFTN             SUPERCEDED BY 18.3    (SCN NO. 669)
DBMSFTNBIN          SUPERCEDED BY 18.3    (SCN NO. 669)
DBMSCOB             SUPERCEDED BY 18.3    (SCN NO. 666)
DBMSCOBBIN          SUPERCEDED BY 18.3    (SCN NO. 666)
DBMSLGCL            SUPERCEDED BY 18.3    (SCN NO. 670)
DBMSLGCLBIN         SUPERCEDED BY 18.3    (SCN NO. 670)
DBMSEX              SUPERCEDED BY 18.3    (SCN NO. 668)
DBMSEXBIN           SUPERCEDED BY 18.3    (SCN NO. 668)
F77                 SUPERCEDED BY 18.3    (SCN NO. 677)
PL1G                SUPERCEDED BY 18.3    (SCN NO. 689)
```

```
POWERPLUS           SUPERCEDED BY 18.3   (SCN NO. 690)
VISTA               SUPERCEDED BY 18.3   (SCN NO. 691)
RJEHASP             SUPERCEDED BY 18.3   (SCN NO. 681)
RJEGRTS             SUPERCEDED BY 18.3   (SCN NO. 680)
RJE7020             SUPERCEDED BY 18.3   (SCN NO. 685)
RJEX80              SUPERCEDED BY 18.3   (SCN NO. 682)
RJE200UT            SUPERCEDED BY 18.3   (SCN NO. 684)
RJE1004             SUPERCEDED BY 18.3   (SCN NO. 683)
RJECOM              SUPERCEDED BY 18.3   (SCN NO. 658)
*
*
*
****  REV 18.3   OCTOBER 1, 1981  ****
*
NAME                DIRECTORY                               SCN NO.        DATE

BACKEND             <MXXXA1>INDEX>BACKEND (DIR.)              644          100181
BATCH               <MXXXA1>BATCH (DIRECTORY)                 643          100181
BATCHQ              <MXXXA1>BATCHQ (DIRECTORY)                643          100181
BATCH               <MXXXA1>CMDNC0 (RUN)                      643          100181
BATGEN              <MXXXA1>CMDNC0 (RUN)                      643          100181
JOB                 <MXXXA1>CMDNC0 (RUN                       643          100181
$$                  <MXXXA1>CMDNC0 (RUN)                      643          100181
CPL_ERR_TABLE       <MXXXA1>SYSOVL (FILE)                     645          100181
CRMPC               <MXXXA1>CRMPC (DIRECTORY)                 646          100181
CRMPC               <MXXXA1>CMDNC0 (RUN)                      646          100181
C_INSTALLSTD        <MXXXA1>SYSTEM (FILE)                     693          100181
C_SHLB              <MXXXA1>SYSTEM (FILE)                     693          100181
C_CREATEALL         <MXXXA1>SYSTEM (FILE)                     693          100181
DEREMER             <MXXXA1>INDEX>DEREMER (DIR.)              647          100181
FIXRAT              <MXXXA1>FIXRAT (DIRECTORY)                652          100181
MAKE                <MXXXA1>CMDNC0 (RUN)                      652          100181
FUTIL               <MXXXA1>FUTIL (DIRECTORY)                 648          100181
FUTIL               <MXXXA1>CMDNC0 (RUN)                      648          100181
HELP*               <MXXXA1>HELP* (DIRECTORY)                 649          100181
MAGSR               <MXXXA1>MAGSR (DIRECTORY)                 651          100181
MAGRST              <MXXXA1>CMDNC0 (RUN)                      651          100181
MAGSAV              <MXXXA1>CMDNC0 (RUN)                      651          100181
PHYSR               <MXXXA1>PHYSR (DIRECTORY)                 653          100181
PHYRST              <MXXXA1>CMDNC0 (RUN)                      653          100181
PHYSAV              <MXXXA1>CMDNC0 (RUN)                      653          100181
PLP                 <MXXXA1>INDEX>PLP (DIRECTORY)             654          100181
PMA                 <MXXXA1>PMA (DIRECTORY)                   655          100181
PMA                 <MXXXA1>CMDNC0 (RUN)                      655          100181
PRIMOS              <MXXXA1>PRIMOS (DIRECTORY)                656          100181
PRIRUN              <MXXXA1>PRIRUN (DIRECTORY)                656          100181
USAGE               <MXXXA1>TOOLS  (RUN)                      656          100181
RJECOM              <MXXXA1>RJECOM (DIRECTORY)                658          100181
R3P0FH.PMA          <MXXXA1>DIRECV (FILE)                     657          100181
SEG                 <MXXXA1>SEG (DIRECTORY)                   659          100181
SEG                 <MXXXA1>CMDNC0 (RUN)                      659          100181
SHARE4              <MXXXA1>LIB (RUN)                         659          100181
SEGSRC              <MXXXA1>SEGSRC (DIRECTORY)                659          100181
SLIST               <MXXXA1>SLIST (DIRECTORY)                 661          100181
```

SLIST	<MXXXA1>CMDNC0 (RUN)	661	100181
SPL	<MXXXA1>INDEX>SPL (DIRECTORY)	662	100181
SP2121	<MXXXA1>SYSTEM (BINARY)	662	100181
SP4000	<MXXXA1>SYSTEM (BINARY)	662	100181
SPLLIB	<MXXXA1>LIB (BINARY)	662	100181
SPOOL	<MXXXA1>SPOOL (DIRECTORY)	663	100181
SPOOLQ	<MXXXA1>SPOOLQ (DIRECTORY)	663	100181
SPOOL	<MXXXA1>CMDNC0 (RUN)	663	100181
PROP	<MXXXA1>CMDNC0 (RUN)	663	100181
SPOOL$	<MXXXA1>LIB (BINARY)	663	100181
VSPOO$	<MXXXA1>LIB (BINARY)	663	100181
VFTNLIB	<MXXXA1>VFTNLIB (DIRECTORY)	664	100181
IFTNLB	<MXXXA1>LIB (BINARY)	664	100181
PFTNLB	<MXXXA1>LIB (BINARY)	664	100181
NPFTNLB	<MXXXA1>LIB (BINARY)	664	100181
S4000	<MXXXA1>SYSTEM (BINARY)	664	100181
S2050	<MXXXA1>SYSTEM (BINARY)	664	100181
ONCODES.P	<MXXXA1>SYSCOM (INSERT)	664	100181
ONCODES.INS.PMA	<MXXXA1>SYSCOM (INSERT)	664	100181
ONCODES.PL1	<MXXXA1>SYSCOM (INSERT)	664	100181
ONCODES.INS.PL1	<MXXXA1>SYSCOM (INSERT)	664	100181
VPSD	<MXXXA1>VPSD (DIRECTORY)	692	100181
VPSD	<MXXXA1>CMDNC0 (RUN)	692	100181
VPSD16	<MXXXA1>CMDNC0 (RUN)	692	100181
COBOL	<MXXXB1>COBOL (DIRECTORY)	665	100181
COBOLLIBSRC	<MXXXB1>COBOLLIBSRC (DIR.)	665	100181
DBMSCOB	<MXXXB2>DBMSCOB (DIR.)	666	100181
DBMSCOBBIN	<MXXXB2>DBMSCOBBIN (DIR.)	666	100181
DBMSDEF	<MXXXB2>DBMSDEF (DIR.)	667	100181
DBMSDEFBIN	<MXXXB2>DBMSDEFBIN (DIR.)	667	100181
DBMSEX	<MXXXB2>DBMSEX (DIR.)	668	100181
DBMSEXBIN	<MXXXB2>DBMSEXBIN (DIR.)	668	100181
DBMSFTN	<MXXXB2>DBMSFTN (DIR.)	669	100181
DBMSFTNBIN	<MXXXB2>DBMSFTNBIN (DIR.)	669	100181
DBMSLGCL	<MXXXB2>DBMSLGCL (DIR.)	670	100181
DBMSLGCLBIN	<MXXXB2>DBMSLGCLBIN (DIR.)	670	100181
DPTX-DSC	<MXXXB1>DPTX-DSC (DIRECTORY)	671	100181
DPTX-TCF	<MXXXB1>DPTX-TCF (DIRECTORY)	672	100181
DPTX-TSF	<MXXXB1>DPTX-TSF (DIRECTORY)	673	100181
FED	<MXXXB1>FED (DIRECTORY)	674	100181
FORMS	<MXXXB1>FORMS (DIRECTORY)	674	100181
FORMS	<MXXXB1>FORMS (DIRECTORY)	675	100181
FTN	<MXXXB1>FTN (DIRECTORY)	676	100181
FTNSRC	<MXXXB1>FTNSRC (DIRECTORY)	676	100181
F77	<MXXXB2>F77 (DIRECTORY)	677	100181
MIDAS	<MXXXB1>MIDAS (DIRECTORY)	678	100181
MIDASSRC	<MXXXB1>MIDASSRC (DIRECTORY)	678	100181
PASCAL	<MXXXB2>PASCAL (DIRECTORY)	688	100181
PASLIBSRC	<MXXXB2>PASCALLIBSRC (DIR.)	688	100181
PL1G	<MXXXB2>PL1G (DIRECTORY)	689	100181
POWERPLUS	<MXXXB2>POWERPLUS (DIRECTORY)	690	100181
PRINET	<MXXXB1>PRINET (DIRECTORY)	679	100181
RJEGRTS	<MXXXB1>RJEGRTS (DIRECTORY)	680	100181
RJEHASP	<MXXXB1>RJEHASP (DIRECTORY)	681	100181

```
RJEX80          <MXXXB1>RJEX80 (DIRECTORY)          682      100181
RJE1004         <MXXXB1>RJE1004 (DIRECTORY)         683      100181
RJE200UT        <MXXXB1>RJE200UT (DIRECTORY)        684      100181
RJE7020         <MXXXB1>RJE7020 (DIRECTORY)         685      100181
RPG             <MXXXB1>RPG (DIRECTORY)             686      100181
RPGSRC          <MXXXB1>RPGSRC (DIRECTORY)          686      100181
RPGLIBSRC       <MXXXB1>RPGLIBSRC (DIRECTORY)       686      100181
VISTA           <MXXXB2>VISTA (DIRECTORY)           691      100181
X.25            <MXXXB1>X.25 (DIRECTORY)            687      100181
*
*
***************************************************************
*********                                         *************
*********                                         *************
*********      REASON FOR CHANGE                  *************
*********                                         *************
*********                                         *************
***************************************************************


*
565                        (COPY_DISK)
```

COPY_DISK REV. 18.0
====================

1. THE NAME OF THE COPY UTILITY HAS BEEN CHANGED TO COPY_DISK.

2. A NEW FIELD HAS BEEN ADDED TO THE COPY_DISK COMMAND. THIS ALLOWS THE USER TO SPECIFY THAT THE COPY VERIFICATION IS TO BE OMMITTED.

   IF 'NOVERIFY' OPTION IS SELECTED THE RESULTING OPERATION IS APPROX. 60% FASTER THAN THE NORMAL COPY_DISK.

```
   EXAMPLE :              <COMMAND> <FIELD 1> <FIELD 2>

                   --->    COPY_DISK -NOVERIFY -NOCHECKSUM
```

   THE EXISTING FIELD OF 'NOCHECKSUM' CAN NOW BE ENTERED AS BEFORE OR WITH THE STANDARD CONVENTION I.E AS '-NOCHECKSUM'.

   DURING THE COPY OPERATION THE FOLLOWING MESSAGES WILL BE CONVEYED TO THE USER :

```
                    COPY STARTED
                    COPY COMPLETED
                    VERIFY STARTED     <----
                    VERIFY COMPLETED   <----   DISPLAYED ONLY IF 'NOVERIFY'
                                               OPTION IS OMMITTED FROM THE
                                               COMMAND LINE.
```

3. THE UTILITY NO LONGER ALWAYS PRINTS THE MESSAGE 40MB DISK YES/NO

FOR A STORAGE MODULE TYPE DEVICE; THIS MESSAGE IS OUTPUT ONLY FOR A PARTITION USING A SURFACE GREATER THAN HEAD OFFSET 4.

REV 18.2
==========

1. COPY_DISK NOW NO LONGER CALLS ITSELF COPY WHEN IT PRINTS OUT THE CORRECT COMMAND LINE FORMAT TO USE.
2. A PERFORMANCE IMPROVEMENT HAS BEEN MADE FOR ALL PROCESSORS BELOW A P750. COPY_DISK ASSUMES IT IS RUNNING ON A P750 WITH BURST MODE DISK CONTROLLER. IF THIS IS NOT SO, THEN TO ACHIEVE THE PERFORMANCE IMPROVEMENT (250% APPROX.) A NEW 'LOWEND' OPTION SHOULD BE SPECIFIED.

EXAMPLE: COPY_DISK -NOVERIFY -LOWEND

*
567 (DEREMER)

THIS IS AN INTERNAL TOOL -- IT IS NOT FOR CUSTOMER USE

*
567 (ED/NSED)

THIS UPDATE DESCRIBES TWO BUG FIXES AND ONE ENHANCEMENT MADE TO THE REV. 18.2 EDITOR.

ACCORDING TO DOCUMENTATION, 'X' IS THE ABBREVIATION FOR 'XEQ', THEREFORE 'X', 'XE' OR 'XEQ' TYPED WITHOUT A STRING BUFFER SHOULD EXECUTE THE LAST COMMAND, ED NOW FUNCTIONS AS PER THE DOCUMENTATION.

TWO NEW MODES HAVE BEEN ADDED, MODE NOSEMI AND MODE SEMI. MODE NOSEMI WILL IGNORE SEMICOLONS WHEN IN INPUT MODE - IDEAL FOR TYPING IN PL1 PROGRAMS, MODE SEMI, WILL SET IT BACK TO NORMAL.

PREVIOUSLY, IF YOU DUNLOADED MORE LINES THAN WERE LEFT IN THE FILE, AND THEN EXITED FROM THE EDITOR, THE MESSAGE "FILE MODIFIED, OK TO QUIT?" WOULD FAIL TO APPEAR AND THE FILE WOULD BE LEFT AS IT WAS BEFORE THE LINES WERE DUNLOADED. THIS BUG HAS BEEN FIXED.

*
568 (EDB/LIBEDB)

EDB/LIBEDB FOR REV 18.2

SOURCE FILES NAMES WERE CHANGED TO USE SUFFIX CONVENTION.

BUILD FILES WERE CHANGED TO EDB.BUILD.CPL & LIBEDB.BUILD.CPL

THERE WERE NO CODE CHANGES

*

569 (LOAD)

LOAD FOR REV 18.2

LOAD NOW CONFORMS TO REV 19 SUFFIX STANDARDS.
THE BUILD FILE IS LOAD.BUILD.CPL

THERE ARE NO CODE CHANGES.

*

570 (LOGPRT)

LOGPRT REV 18.2 CONTAINS MINOR BUG FIXES.

*

571 (MAGSAV/RST)

MAGSAV/RST FOR REV18.2

```
WHEN MAGSAV RUNS OUT OF UNITS TO OPEN UFDS ON THEN INSTEAD OF GIVING THE ERROR -
  BAD UNIT NUMBER
 IT NOW GIVES
    RUN OUT OF UNITS
    ALTER NUMBER OF UNITS
    TREENAME
   TYPE 'S' TO CONTINUE
```

THE FILE CONCERNED WILL BE LOST .

FAM PROBLEM FIXED WHEN SAVING REMOTE MFD.

IF MAGSAV CANNOT FIND A UFD IN A PATHNAME THEN MAGSAV WILL PRINT THE PRIMOS ERROR MESSAGE PLUS:-

'MAGSAV UNABLE TO CONTINUE'

AND EXITS.
MAGSAV WILL NOW SAVE UP TO 18 LEVELS ON PRIMOS4, PRIMOS 2 WILL STILL HANDLE

13. IF MORE THAN THESE LEVELS ARE ATTEMPTED THEN,
IT WILL GIVE THE MESSAGE - 'TOO MANY LEVELS' TREENAME,
IGNORE THAT FILE AND CONTINUE BACK UP THE TREE.

*
572 (PLP)

THIS IS AN INTERNAL TOOL -- IT IS NOT FOR CUSTOMER USE

*
573 (PRIMOS)

SUBJECT: PRIMOS

RELEASE: 18.2

DATE: MAY 29, 1981

TABLE OF CONTENTS

1 NEW FEATURES

1.1 PRIMOS CHANGES NEEDED TO SUPPORT P850

MANY OF THE CHANGES FROM REV. 17 TO BASE LEVEL REV. 18 PRIMOS ARE FOR P850 SUPPORT. THE P850 CONFIGURATION CONSISTS OF TWO INSTRUCTION STREAM UNITS (ISUS) WHICH CAN COMMUNICATE WITH THE ASSISTANCE OF A STREAM SYNCHRONIZATION UNIT (SSU) AND A PAGE OF COMMON MEMORY. THE SSU SERVES PRIMARILY TO KEEP BOTH ISUS' CACHE CELLS FROM BECOMING STALE. ONLY ONE OF THE ISUS IN AN P850 CONFIGURATION IS CAPABLE OF PERFORMING I/O. IT IS REFERRED TO AS THE MASTER ISU. THE OTHER ISU IS REFERRED TO AS THE SLAVE. FOR MORE INFORMATION ABOUT THE P850, SEE THE P850 HARDWARE SPECIFICATION.

THERE ARE NO VISIBLE CHANGES TO USERS OF AN P850 CONFIGURATION EXCEPT THAT THE SYSTEM IS FASTER THAN A P750. THERE ARE CHANGES TO THE OPERATING SYSTEM, THOUGH, TO SUPPORT P850. THESE CHANGES INCLUDE THOSE TO SOLVE DUAL-INSTRUCTION STREAM CONTENTION PROBLEMS, TO START THE SLAVE ISU, AND TO PROVIDE MASTER/SLAVE ISU INTERCOMMUNICATION.

1.1.1 THE P850 WINDOW A COMMUNICATIONS AREA FOR THE TWO P850 ISUS IS PLACED AT PHYSICAL PAGE NUMBER '477. FURTHER, IT

IS A P850 REQUIREMENT THAT THE PHYSICAL AND VIRTUAL ADDRESS OF THE COMMUNICATIONS AREA BE EQUAL (I.E., THAT THE VIRTUAL ADDRESS BE IN SEGMENT 4, VIRTUAL PAGE NUMBER '77). THIS ALLOWS THE P850 TO ACCESS ITS COMMUNICATIONS AREA IN BOTH ABSOLUTE AND SEGMENTED MODES. THE P850 SSU INTERCEPTS ALL MEMORY REFERENCES TO THE COMMUNICATIONS AREA.

1.1.1.1 P850 WINDOW INITIALIZATION COLD START CODE FIXES PAGE MAPS ON AN P850 CONFIGURATION SUCH THAT THE P850 WILL RESPOND TO THE PROPER VIRTUAL ADDRESS. PRIMOS MAPS THE LAST PAGE OF SEGMENT 4 TO THE 1K OF ADDRESSABLE CELLS IN THE P850 COMMUNICATIONS AREA. THE PAGE MAP ENTRY FOR THIS PAGE IS SET TO PAGE NUMBER '477. THE P850 PAGE IS MARKED AS WIRED AND NON-ENCACHEABLE.

1.1.1.2 P850-RELATED INSERT FILE AN INSERT FILE CALLED APCOM.INS.PMA HAS BEEN CREATED TO ALLOW MNEMONIC REFERENCE TO CELLS IN THE P850 COMMUNICATIONS AREA. THIS INSERT FILE IS EFFECTIVELY A MAP OF THE P850 COMMUNICATIONS AREA. APPENDIX C CONTAINS A DESCRIPTION OF THE CELLS DEFINED IN THIS INSERT FILE.

1.1.1.3 DATA TRANSFERS TO/FROM THE P850 COMMUNICATIONS AREA DATA TRANSFERS TO AND FROM THE P850 COMMUNICATIONS AREA MUST BE SINGLE-WORD TRANSFERS (I.E., STA/LDA, NOT STL/LDL/DMX/ETC.). MULTIPLE-WORD REFERENCES WILL RESULT IN BAD DATA BEING TRANSFERRED.

THE P850 COMMUNICATIONS AREA CONTAINS CELLS WHICH, WHEN REFERENCED, ACTIVATE MESSAGES FROM ONE INSTRUCTION STREAM UNIT (ISU) TO THE OTHER. SUCH CELLS ARE UTILIZED BY U-CODE SO THAT ONE ISU CAN NOTIFY THE OTHER ABOUT A SPECIFIC EVENT.

1.1.1.4 PRIMOS USE OF THE COMMUNICATIONS AREA PRIMOS CODE IN GENERAL SHOULD NOT REFERENCE THE COMMUNICATIONS AREA. MOST INTER-ISU MESSAGES CAN AND SHOULD BE GENERATED BY NECESSARY MESSAGES IN U-CODE. CURRENTLY, PRIMOS REFERENCES THE P850 COMMUNICATIONS AREA ONLY DURING COLD START AND WARM START.

1.1.2 COLD AND WARM START THE COLD AND WARM START MECHANISMS IN PRIMOS HAVE BEEN MODIFIED IN ORDER TO SUPPORT THE P850 CONFIGURATION. MOST OF THE CHANGES ARE DESCRIBED BELOW.

1.1.2.1 APCNFG FLAG THE OPERATING SYSTEM WILL SET A FLAG CALLED APCNFG DURING COLD START IF RUNNING ON A P850 CONFIGURATION, AS DETERMINED BY THE PROCESSOR I.D. THE OPERATING SYSTEM CONTAINS CODE WHICH IS EXECUTED DEPENDENT UPON WHETHER APCNFG IS TRUE OR FALSE.

1.1.2.2 ENTERING PROCESS EXCHANGE MODE BEFORE REV. 18, COLD AND WARM START CODE COMPLETED BY ENTERING PROCESS EXCHANGE MODE AND BECOMING A SYSTEM PROCESS. COLD START CODE COMPLETED BY BECOMING THE SUPERVISOR PROCESS AND WARM START CODE COMPLETED BY BECOMING THE CLOCK PROCESS. THE OPERATING SYSTEM HAS BEEN MODIFIED TO ENTER THE DISPATCHER TO COMPLETE COLD AND WARM STARTS. THIS IS DONE THROUGH THE USE OF THE LPSW INSTRUCTION WITH THE IN-DISPATCHER BIT SET. PROCESS EXCHANGE U-CODE (I.E., THE DISPATCHER) SCANS THE READY LIST FOR A READY PROCESS TO RUN. THIS MECHANISM IS ALSO USED BY THE SLAVE ISU WHEN IT IS STARTING ITSELF UP.

1.1.2.3 STARTING THE SLAVE INSTRUCTION STREAM UNIT AT THE COMPLETION OF COLD AND WARM START, THE SLAVE ISU MUST BE STARTED. THIS IS DONE BY SENDING THE SLAVE AN ADDRESS TO START EXECUTION AT. THIS LOCATION CONTAINS CODE TO ALLOW THE SLAVE TO INITIALIZE ITSELF AND ENTER THE PROCESS EXCHANGE DISPATCHER.

A FLAG CALLED APRUN HAS BEEN ADDED TO THE SYSTEM. THIS FLAG IS SET TRUE WHEN THE SLAVE ISU HAS BEEN SUCCESSFULLY STARTED. FAILURE TO START THE SLAVE ISU AT EITHER WARM OR COLD START TIME WILL GENERATE AN ERROR MESSAGE AT THE OPERATOR TERMINAL, AND THE SYSTEM WILL HALT.

1.1.2.4 P850 AND COLD START PAGE THE SECOND PAGE OF SEGMENT 14 IN PRIMOS HAS BEEN RESERVED FOR CODE AND DATA USED ONLY DURING COLD START, AND FOR CODE AND DATA USED ONLY BY PRIMOS WHEN RUNNING ON A P850 CONFIGURATION. IF THE SYSTEM IS COLD STARTED ON A NON-P850 CONFIGURATION, THIS PAGE IS UNWIRED AT THE END OF COLD START.

1.1.3 PROCESS EXCHANGE PROCESS EXCHANGE FOR THE P850 HAS LITTLE VISIBLE CHANGE ABOVE THE U-CODE LEVEL. ON P850 CONFIGURATIONS, PROCESS EXCHANGE DATA IS KEPT BOTH IN EACH ISU'S REGISTER FILE AND IN CELLS IN THE P850 COMMUNICATIONS AREA. NOTE THAT ANY PROCESS EXCHANGE ACTIVITY MAY AFFECT BOTH ISUS IN AN P850 CONFIGURATION.

1.1.3.1 THE PX LOCK A LOCK IN THE P850 COMMUNICATIONS AREA CALLED THE PX (PROCESS EXCHANGE) LOCK IS USED BY BOTH U-CODE AND SOFTWARE WHEN REFERENCING ANY PROCESS EXCHANGE DATA. THIS PREVENTS TWO PROCESSES RUNNING ON DIFFERENT ISUS FROM REFERENCING ANY CRITICAL PROCESS EXCHANGE DATABASES AT THE SAME TIME. ALL PROCESS EXCHANGE INSTRUCTIONS AND U-CODE WHICH REFERENCE PCBS, SEMAPHORES, AND THE READY LIST OBTAIN THE PX LOCK. ALL SOFTWARE WHICH REFERENCES THESE DATABASES MUST ALSO RESPECT THIS LOCK.

1.1.3.2 OBTAINING THE PX LOCK RUN TIME EXAMINATION OR MODIFICATION OF ANY PROCESS EXCHANGE DATA MUST ALWAYS BE

DONE WITH THE PX LOCK OWNED. THE PX LOCK IS OBTAINED BY ISSUING AN INHP INSTRUCTION AND RELEASED WITH THE ENBP INSTRUCTION. THESE TWO NEW INSTRUCTIONS HAVE THE EFFECT OF BOTH INHIBITING AND ENABLING INTERRUPTS AS WELL AS OBTAINING AND RELEASING THE PX LOCK. ON A NON-P850 CONFIGURATION, THE OP-CODES EXECUTE THE SAME AS INH AND ENB INSTRUCTIONS. NOTE THAT ENBP CLEARS THE PX LOCK DURING ITS EXECUTION. THERE IS NO ONE-INSTRUCTION GRACE PERIOD FOLLOWING ENBP.

1.1.3.3 PROCESS EXCHANGE INSTRUCTIONS ALL PROCESS EXCHANGE INSTRUCTIONS CAUSE THE PX LOCK TO BE OBTAINED. THE LOCK IS RELEASED AT THE END OF EXECUTION OF SUCH AN INSTRUCTION. NOTE THAT ANY PROCESS EXCHANGE INSTRUCTION MAY BE ISSUED AFTER AN INHP HAS BEEN EXECUTED. THE PX LOCK WILL BE RELEASED AT THE END OF THE INSTRUCTION, BUT A REDUNDANT ENBP INSTRUCTION SHOULD BE ISSUED ANYWAY.

1.1.3.4 CONDITIONAL WAITS AND NOTIFIES CASES OF WAITS OR NOTIFYS WHICH ARE CONDITIONAL ON THE VALUES FOUND IN SEMAPHORES WERE FORMERLY COVERED BY AN INH OR ENB INSTRUCTION. IN ADDITION, THE ENB WAS GENERALLY ISSUED PRIOR TO THE WAIT OR NOTIFY INSTRUCTION AND AFTER THE EXAMINATION OF THE SEMAPHORES. (THE ONE INSTRUCTION GRACE PERIOD OF THE ENB INSTRUCTION COVERED THE WAIT OR NOTIFY). ON AN P850 CONFIGURATION, THE PX LOCK MUST BE HELD DURING THE TEST OF SEMAPHORES AND MUST NOT BE RELEASED BEFORE THE WAIT OR NOTIFY IS EXECUTED. HENCE, AN INHP MUST BE EXECUTED BEFORE TESTING IS STARTED AND THE ENBP ISSUED AFTER THE WAIT OR NOTIFY IS EXECUTED (E.G., SEE N1LOCK OR SCHED).

1.1.3.5 PROCESS EXCHANGE REGISTERS THE FOLLOWING LISTS ALL PROCESS EXCHANGE REGISTERS IN THE ISU REGISTER FILES WHICH DEVIATE FROM NON-P850 CPUS:

REGISTER	USAGE
PPA	DUPLICATE COPIES OF PPA ARE MAINTAINED BOTH IN EACH ISU'S REGISTER FILE AND IN THE P850 COMMUNICATIONS AREA.
PPNEXT, APADR	PPNEXT REPLACES PPB AND IS KEPT ONLY IN THE P850 COMMUNICATIONS AREA. THE REGISTER PREVIOUSLY USED FOR PPB IS USED IN A P850 CONFIGURATION TO STORE THE ADDRESS OF THE P850 COMMUNICATIONS AREA AND IS CALLED APADR.
CPUNUMH	REGISTER 33H OF THE CURRENT REGISTER SET IS ALLOCATED TO CONTAIN A BIT FIELD INDICATING THIS ISU'S ATTRIBUTES (E.G., ISU NUMBER, SLAVE

OR MASTER ISU).

1.1.3.6 PROCESS CONTROL BLOCK WORD 5

PCB WORD 5 (SIXTH WORD IN THE PCB, PCB+5) IS USED TO STORE THE FOLLOWING PER-PROCESS INFORMATION: ISU DESIRED TO RUN ON, REGISTER SET LAST USED, REGISTER SET SAVED IN PCB, ISU LAST RUN ON, TEMPORARY ISU PREFERENCE. ONLY THE FIELD INDICATING ISU DESIRED TO RUN ON SHOULD BE MODIFIED BY SOFTWARE. THIS FIELD IS TO BE UPDATED ONLY WHEN THE PX LOCK IS OWNED.

THE FOLLOWING FIELDS ARE DEFINED IN PCB+5:

BIT	VALUE	MEANING
1-4	170000	TEMPORARILY DO NOT RUN ON: 040000 MASTER ISU 100000 SLAVE ISU
5	004000	RESERVED
6-7	003000	PROCESS WAS LAST RUN ON: 001000 MASTER ISU 002000 SLAVE ISU
8	000400	REGISTER SET HAS BEEN SAVED IN PCB
9-11	000340	REGISTER SET LAST USED
12	000020	RESERVED
13-16	000017	RUN PROCESS ONLY ON: 000004 MASTER ISU 000010 SLAVE ISU

1.1.3.7 LOCKING A PROCESS TO AN INSTRUCTION STREAM UNIT

ISU PREFERENCE (MASTER OR SLAVE) IS DETERMINED BY WORD 5 IN A PROCESS' PCB AS DETAILED IN THE SECTION ABOVE. OPERATING SYSTEM PROCESSES WHICH ARE TO BE CONSTRAINED TO RUN ON THE MASTER ISU INITIALLY HAVE THEIR PCB+5 WORD SET TO '000404. (THE MODULE SEG4.PMA PREDEFINES THIS WORD IN SYSTEM PROCESS' PCBS.) THIS WORD MAY BE MODIFIED DURING SYSTEM OPERATION BY FIRST OBTAINING THE PROCESS EXCHANGE LOCK.

1.1.4 NEW SYSTEM PROCESSES

TWO ADDITIONAL SYSTEM PROCESSES HAVE BEEN ADDED TO PRIMOS IN ORDER TO FACILITATE RUNNING IN A DUAL-INSTRUCTION STREAM P850 CONFIGURATION. THESE ARE DESCRIBED BELOW:

1.1.4.1 SLAVE BACKSTOP PROCESS

EACH ISU IN AN P850 CONFIGURATION MUST RUN ITS OWN BACKSTOP PROCESS SINCE BOTH ISUS MAY NEED TO RUN A BACKSTOP CONCURRENTLY. SEPARATE INFORMATION FOR EACH BACKSTOP PROCESS IS KEPT FOR METERING PURPOSES.

1.1.4.2 SLAVE FRONTSTOP PROCESS

THE FRONTSTOP PROCESS IS THE HIGHEST PRIORITY PROCESS THAT CAN RUN ON THE SLAVE. THE FRONTSTOP PROCESS PERFORMS FUNCTIONS DONE BY THE CLOCK PROCESS ON THE MASTER. IT RUNS AT A HIGH ENOUGH

FREQUENCY TO ALLOW TIMELY PROCESS ABORTS TO OCCUR ON THE SLAVE. IT ALSO METERS DATA FOR THE PB HISTOGRAM.

1.1.5 ERROR HANDLING

1.1.5.1 CHECKS PRIMOS CHECK HANDLERS ARE FUNDAMENTALLY UNAFFECTED BY THE P850. CHECK HANDLING U-CODE WILL AUTOMATICALLY OBTAIN A CHECK HANDLING LOCK IN THE P850 COMMUNICATIONS AREA BEFORE INVOKING A SOFTWARE CHECK HANDLER. THIS LOCK PREVENTS BOTH ISUS FROM ATTEMPTING TO RESPOND CONCURRENTLY TO THE SAME HARDWARE PROBLEM. IT WILL AUTOMATICALLY BE CLEARED BY THE LPSW INSTRUCTION WHEN RETURNING FROM A CHECK HANDLER.

AN ERROR IN THE P850 HARDWARE WILL CAUSE BIT 16 OF DSWPARITYH TO BE SET AND A MACHINE CHECK TO BE GENERATED.

1.1.5.2 HALTS A HALT EXECUTED ON EITHER ISU WILL CAUSE THE OTHER ISU TO HALT ALSO. THIS IS ACCOMPLISHED IN U-CODE BY SENDING A MESSAGE TO THE OTHER ISU VIA THE P850 COMMUNICATIONS AREA. THE HLT MESSAGE IS THE ONLY MESSAGE THAT CAN BE RECEIVED BY AN ISU WHILE INHIBITED.

1.1.5.3 CRASH SAVE MECHANISM THE P850 CRASH SAVE MECHANISM PROVIDES FOR A CRASH SAVE FOR EACH ISU. BOTH ISUS HAVE THEIR REGISTER SAVE AREA POINTERS (RSAVPTR) INITIALIZED DURING COLD AND WARM STARTS. P850 U-CODE WILL NOT CAUSE THESE REGISTERS TO BE CLEARED AFTER A REGISTER DUMP TO THE CRASH SAVE AREA.

NON-P850 CONFIGURATIONS WILL CLEAR RSAVPTR AFTER REGISTERS HAVE BEEN DUMPED TO MEMORY AS ALWAYS, THUS THE CLOCK PROCESS WILL CONTINUE TO LOAD RSAVPTR ON BOTH P850 AND NON-P850 CONFIGURATIONS.

THE CRASH REGISTER SAVE AREA FOR THE P850 MASTER ISU IS '600 WORDS LONG. (THIS IS '200 WORDS GREATER THAN CRASH SAVE AREA FOR A NON-P850 CONFIGURATION.) THE EXTRA '200 WORDS IN THE MASTER'S CRASH SAVE AREA ARE USED TO STORE '200 WORDS FROM THE P850 COMMUNICATIONS AREA. THE CRASH REGISTER SAVE AREA FOR THE P850 SLAVE ISU IS '400 WORDS LONG.

1.1.5.4 CRASH SAVE AREA THE CRASH SAVE AREA FOR THE MASTER ISU (AND FOR NON-P850 CONFIGURATIONS) HAS BEEN MOVED TO LOCATION '1400 IN SEGMENT 14. IN NON-P850 CONFIGURATIONS, THE CRASH SAVE AREA ENDS AT LOCATION '1777, OR AT THE END OF PAGE 0 OF SEGMENT 14. IN P850 CONFIGURATIONS, IT EXTENDS TO LOCATION '2177. THE CRASH SAVE AREA FOR THE SLAVE ISU IS FOUND AT LOCATION '2200

IN SEGMENT 14. NOTE THAT THE PAGE CONTAINING THE EXTENSION TO THE MASTER'S CRASH SAVE AREA AND THE SLAVE'S CRASH SAVE AREA IS UNWIRED AFTER COLD START WHEN PRIMOS IS RUN ON A NON-P850 CONFIGURATION.

1.1.6 I/O CONSIDERATIONS ONLY THE MASTER ISU IN AN P850 CONFIGURATION IS CAPABLE OF PERFORMING I/O. THEREFORE, THOSE PROCESSES WHICH ARE TO PERFORM I/O MUST BE CONSTRAINED TO RUN ON THE MASTER ISU.

1.1.6.1 PIO INSTRUCTIONS WHENEVER ANY P850 ISU TRIES TO EXECUTE AN I/O INSTRUCTION (PIO), A CHECK OF ISU NUMBER IS CALCULATED IN U-CODE. IF IT IS DETERMINED THAT THE I/O INSTRUCTION IS BEING ATTEMPTED ON THE SLAVE ISU, THEN THE PROCESS IS TAGGED IN ITS PCB AS REQUIRING THE MASTER ISU. THE PROCESS IS SUSPENDED FROM RUNNING ON THE SLAVE AND IS PICKED UP LATER BY THE MASTER. NOTE THAT A PIO INSTRUCTION EXECUTED ON THE MASTER ISU DOES NOT LOCK A PROCESS TO THE MASTER.

1.1.6.2 INTERRUPT PROCESSES OPERATING SYSTEM INTERRUPT PROCESSES SUCH AS THE CLOCK PROCESS, THE MPC, THE AMLC, AND DISK PROCESSES, ETC. MUST BE CONSTRAINED TO RUN ON THE MASTER ISU SINCE THEY PERFORM I/O. SUCH PROCESSES HAVE THEIR PCB+5 WORDS INITIALLY SET TO '000404 IN THE MODULE SEG4.PMA.

1.1.6.3 USER PPOCESSES USER PROCESSES WHICH DO I/O (E.G., MAGTAPE USERS) MUST BE CONSTRAINED TO RUN ON THE MASTER ISU BEFORE DOING ANY I/O OR LOADING ANY DMA CHANNELS. THIS CAN BE ACHIEVED BY MODIFYING PCB+5 (UNDER PROTECTION OF THE PX LOCK) AND ISSUING ANY PIO INSTRUCTION (E.G., INA '1620). WHEN A PROCESS HAS FINISHED PERFORMING I/O, THE ISU REQUIRED FIELD OF PCB+5 CAN BE CLEARED (UNDER PROTECTION OF THE PX LOCK). PROCESSES SHOULD BE LOCKED TO AN ISU NO LONGER THAN IS ABSOLUTELY NECESSARY IN ORDER TO EVENLY SPREAD THE PROCESSING LOAD BETWEEN BOTH ISUS.

1.1.6.4 SEGMENT 0 WINDOWS SOFTWARE RUNNING ON THE SLAVE ISU SHOULD AVOID MAKING ANY REFERENCES TO SEGMENT 0, ESPECIALLY SEGMENT 0 I/O WINDOWS. THE IOTLB ON THE SLAVE ISU IS NEVER LOADED BY THE LIOT INSTRUCTION. WHEN LIOT IS EXECUTED ON A SLAVE ISU, A MESSAGE IS SENT TO THE MASTER ISU TO CAUSE THE MASTER TO LOAD ITS IOTLB. REFERENCES TO SEGMENT 0 ON THE SLAVE ISU WILL CAUSE THE SLAVE'S IOTLB TO BE LOADED, BUT THE CACHE LEAF SELECTION BITS WILL BE SET AT RANDOM. HENCE, ANY PAGES OF SEGMENT 0 WHICH ARE TO EVER BE REFERENCED BY THE SLAVE ISU IN A P850 CONFIGURATION SHOULD BE MARKED AS NON-ENCACHEABLE IN THE PAGE MAP ENTRIES FOR THOSE PAGES.

1.1.7 **DUAL INSTRUCTION STREAM CONSIDERATIONS** MANY MECHANISMS IN PRIMOS (AS WELL AS OTHER SOFTWARE) MAY BE AFFECTED BY THE FACT THAT TWO PROCESSES MAY BE RUNNING SIMULTANEOUSLY ON TWO INSTRUCTION STREAM UNITS. SOME OF THE PROBLEMS WHICH CAN BE ENCOUNTERED AND THE MECHANISMS IN PRIMOS AND THE P850 HARDWARE TO SOLVE THEM ARE DISCUSSED BELOW.

1.1.7.1 **IMPLICIT PRIORITY ASSUMPTIONS** OPERATING SYSTEM CODE MAY NO LONGER MAKE IMPLICIT PRIORITY ASSUMPTIONS ABOUT TWO PROCESSES RUNNING AT DIFFERENT PRIORITIES. THAT IS, A PROCESS MUST NEVER ASSUME THAT ANOTHER PROCESS OF LOWER PRIORITY THAN IT CANNOT BE EXECUTING SIMULTANEOUSLY. ANY TWO PROCESSES MAY RUN CONCURRENTLY ON AN P850 CONFIGURATION (EXCEPT THOSE BELONGING TO A SET OF PROCESSES CONSTRAINED TO RUN ON A SPECIFIC ISU).

1.1.7.2 **PREVENTION OF INTERRUPTS** PRIMOS CODE WHICH MUST NOT BE INTERRUPTED BY ANOTHER PROCESS IS OFTEN SURROUNDED BY INH AND ENB INSTRUCTIONS. ON A P850 CONFIGURATION, THIS MECHANISM WILL PREVENT INTERRUPTS FROM OCCURRING ONLY ON THE PROCESSOR ON WHICH THE INHIBITED PROCESS IS RUNNING. ANOTHER PROCESS CAN, HOWEVER, BE RUNNING ON THE OTHER ISU. THUS, INH AND ENB ARE STILL USEFUL FOR PREVENTING INTERRUPTS, BUT SHOULD NEVER BE USED BY A PROCESS TO ENSURE THAT A CODE SEQUENCE IS INDIVISIBLE OR ATOMIC. IT SHOULD BE NOTED THAT INTERRUPTS CAN OCCUR ONLY ON THE MASTER ISU; THE SLAVE ISU NEVER TAKES INTERRUPTS.

1.1.7.3 **INHL AND ENBL INSTRUCTIONS** TWO NEW OPERATION MNEMONICS HAVE BEEN INTRODUCED IN REV. 18 PRIMOS, INHL AND ENBL. THESE MNEMONICS GENERATE THE SAME OP-CODES AS THE INH AND ENB INSTRUCTIONS, BUT THE "L" SERVES AS A REMINDER THAT THE INHIBIT OPERATION IS EFFECTIVE ONLY ON THE LOCAL ISU. THE MNEMONICS FOR THE INH AND ENB INSTRUCTIONS HAVE BEEN DEFINED TO GENERATE ASSEMBLY ERRORS. THIS HAS BEEN DONE IN ORDER TO FORCE ALL SOFTWARE TO BE EXAMINED FOR THE PROPER USE OF INHIBITS AND ENABLES IN THE P850 CONFIGURATION.

1.1.7.4 **INHIBITED CODE** IN MANY PLACES IN THE OPERATING SYSTEM, INH AND ENB HAVE BEEN USED TO GUARANTEE THAT NO OTHER PROCESSES MAY EXECUTE SIMULTANEOUSLY. THIS HAS OFTEN BEEN DONE IN LIEU OF SETTING A SOFTWARE LOCK TO PROTECT A DATABASE. INHIBITED CODE WILL HAVE NO SUCH GUARANTEE ON A P850 CONFIGURATION. THUS, USE OF THE P850 INTRODUCES THE REQUIREMENT THAT DIFFERENT METHODS BE USED TO GUARANTEE THAT CERTAIN OPERATIONS CAN BE PERFORMED IN A NON-INTERRUPTIBLE AND MUTUALLY EXCLUSIVE MANNER. SOME OF THE AVAILABLE METHODS ARE DISCUSSED IN

THE FOLLOWING SECTIONS.

INHIBITED CODE SHOULD NEVER BE PLACED IN A PAGE WHICH IS NOT WIRED. THIS SUPERSEDES THE RULE STATING THAT INHIBITED CODE MUST NOT CROSS A PAGE BOUNDARY BECAUSE A PAGE FAULT MAY NOT OCCUR WHILE INHIBITED. ON AN P850 CONFIGURATION, INHIBITED CODE MUST BE WIRED BECAUSE THE OTHER ISU MAY PAGE OUT ANY UNWIRED PAGES WHILE HANDLING A PAGE FAULT.

1.1.7.5 THE MX LOCK IN ORDER TO AID THE OPERATING SYSTEM IN PERFORMING INDIVISIBLE CODE SEQUENCES, A LOCK CELL HAS BEEN PROVIDED IN THE P850 COMMUNICATIONS AREA. THIS LOCK IS REFERRED TO AS THE MX (MUTUAL EXCLUSION) LOCK. IT CAN BE SET AND TESTED BY SOFTWARE TO PREVENT TWO PROCESSES ON DIFFERENT ISUS FROM RUNNING SIMULTANEOUSLY IN "PROTECTED" CODE.

TWO NEW INSTRUCTIONS, INHM AND ENBM, HAVE BEEN PROVIDED TO OBTAIN AND RELEASE THE MX LOCK AT THE SAME TIME THAT INTERRUPTS ARE INHIBITED AND ENABLED. ON NON-P850 CONFIGURATIONS THEY EXECUTE IDENTICALLY TO THE INH AND ENB INSTRUCTIONS. THE INHM INSTRUCTION WILL WAIT FOR THE MX LOCK TO BE RELEASED IF IT IS OWNED BY THE OTHER ISU. OVERUSE OF THE MX LOCK WILL DRASTICALLY DECREASE INSTRUCTION STREAM OVERLAP, AND IS THEREFORE TO BE AVOIDED. NOTE THAT THE ENBM INSTRUCTION HAS NO SINGLE-INSTRUCTION GRACE PERIOD FOLLOWING ITS EXECUTION. IT, THEREFORE, MUST BE ISSUED FOLLOWING THE END OF A "PROTECTED" CODE SEQUENCE.

1.1.7.6 INDIVISIBLE INSTRUCTIONS MOST MACHINE INSTRUCTIONS CONSIST OF A SERIES OF U-CODE STEPS WHICH MAY MAKE MEMORY REFERENCES. A MEMORY BUS LOCK CAN BE OBTAINED IN U-CODE FOR U-CODE SEQUENCES WHICH MAKE MULTIPLE MEMORY REFERENCES. THIS LOCK CAN BE HELD THROUGHOUT AN ENTIRE INSTRUCTION. (USE OF THE MEMORY BUS IS ALWAYS ARBITRATED BY HARDWARE REGARDLESS OF LOCK USAGE.) MOST INSTRUCTIONS DO NOT OBTAIN THIS LOCK, MEANING THAT MEMORY AS SEEN BY AN ISU CAN BE CHANGING DURING THE U-CODE STEPS IT IS EXECUTING. SEVERAL INSTRUCTIONS WHICH EITHER REFERENCE A SINGLE MEMORY CELL MORE THAN ONCE, OR WHICH REFERENCE MORE THAN ONE CELL, HAVE BEEN MODIFIED IN THE P850 CONFIGURATION TO OPERATE IN AN ATOMIC OR INDIVISIBLE MANNER. THESE ARE LISTED BELOW:

1.1.7.6.1 QUEUEING INSTRUCTIONS THE INSTRUCTIONS ATQ, ABQ, RTQ, RBQ, AND TSTQ ARE INDIVISIBLE.

1.1.7.6.2 WAITS AND NOTIFIES ALL PROCESS EXCHANGE INSTRUCTION ARE INDIVISIBLE UNDER THE PROTECTION OF THE PX LOCK. THESE INCLUDE WAIT, NFYB, NFYE, INEC, INEN, INBC, INBN.

1.1.7.6.3 CONDITIONAL STORE THE STAC INSTRUCTION IS ALWAYS INDIVISIBLE. STLC IS INDIVISIBLE ONLY IF THE TARGET ADDRESS IS EVEN. THE OTHER ISU IS PREVENTED FROM REFERENCING MEMORY BETWEEN THE TEST AND THE STORE PORTIONS OF THESE TWO INSTRUCTIONS.

1.1.7.6.4 32-BIT INSTRUCTIONS 32-BIT MEMORY REFERENCE INSTRUCTIONS WILL REFERENCE A WORD-PAIR INDIVISIBLY IF, AND ONLY IF, THE FIRST WORD OF THE PAIR IS IN AN EVEN LOCATION. SUCH INSTRUCTIONS INCLUDE LDL, STL, ADL, SBL, MPL, DVL, ANL, ERL, AND CLS.

1.1.7.7 IMA AND IRS IMA AND IRS INSTRUCTIONS ARE NOT ATOMIC IN AN P850 CONFIGURATION. THEY SHOULD NEVER BE USED TO REFERENCE CELLS IN SHARED MEMORY WHICH COULD BE SIMULTANEOUSLY REFERENCED BY THE OTHER ISU. FAILURE TO AVOID SUCH USAGE COULD RESULT IN MEMORY CELLS BEING MODIFIED BY THE OTHER ISU IN THE MIDDLE OF EXECUTION OF THESE INSTRUCTIONS. IMA AND IRS INSTRUCTIONS IN THE OPERATING SYSTEM SHOULD BE CAREFULLY CHECKED. THEY SHOULD BE PROTECTED BY A LOCK OR BY INHM, OR REPLACED WITH A STAC SEQUENCE.

1.1.7.8 USE OF CONDITIONAL STORE INSTRUCTIONS THE STAC AND STLC INSTRUCTIONS CAN BE USED TO RECOVER FROM DUAL-INSTRUCTION STREAM INTERFERENCE WHEN READING AND MODIFYING MEMORY CELLS SUBJECT TO CONCURRENT REFERENCE. THESE INSTRUCTIONS WILL NOT PERFORM A STORE IF THE TARGET MEMORY CELL HAS CHANGED (DUE TO A CONCURRENTLY RUNNING PROCESS UPDATING THE SAME CELL) BETWEEN THE TIME THE VARIABLE HAS BEEN PICKED UP AND THE TIME OF THE CONDITIONAL STORE. THE SUCCESS OR FAILURE OF A CONDITIONAL STORE OPERATION CAN BE TESTED IN THE CONDITION CODES.

CONSIDER THE FOLLOWING CODE SEQUENCES BEING EXECUTED CONCURRENTLY BY PROCESSES A AND B:

```
PROCESS A              PROCESS B
---------              ---------
LDA   FLAGS            LDA   FLAGS
ORA   ='002000         ORA   ='000001
STA   FLAGS            STA   FLAGS
```

THE INTENDED OVERALL RESULT IS TO SET BIT 6 AND BIT 16 IN FLAGS. WHICH BIT WAS SET IN THE ABOVE EXAMPLE? CERTAINLY NOT BOTH. THE ANSWER IS INDETERMINATE. THE FOLLOWING CODE SEQUENCE SOLVES THIS PROBLEM THROUGH THE USE OF STAC:

```
LOOP  LDA   FLAGS
      TAB
```

```
          ORA     =(BIT_TO_SET)
          STAC    FLAGS
          BCNE    LOOP
```

THE STAC INSTRUCTION WILL NOT STORE THE A REGISTER TO THE TARGET WORD IF THE TARGET WORD HAS BEEN MODIFIED (AS IN THE EXAMPLE ABOVE, AFTER THE LDA AND BEFORE THE STAC). THE TESTING AND SUBSEQUENT POSSIBLE STORE DONE BY THE STAC INSTRUCTION ARE GUARANTEED TO BE INDIVISIBLE. NOTE THAT THE CODE SEQUENCE WILL ITERATE UNTIL THE VARIABLE HAS BEEN SUCCESSFULLY MODIFIED.

1.1.7.9 PAGE WIRING IN ORDER TO SOLVE THE CONTENTION PROBLEM OF BOTH P850 ISUS POSSIBLY TRYING TO PAGE IN OR PAGE OUT THE SAME PAGE, AN ALGORITHM FOR WIRING AND UNWIRING PAGES WAS DESIGNED AND IMPLEMENTED. THE ALGORITHM SHOWN BELOW WORKS PROPERLY ON A DUAL-INSTRUCTION STREAM CONFIGURATION, EVEN IF THE PROCESS EXECUTING THE ALGORITHM IS INTERRUPTED AND RESUMES OPERATION ON THE OTHER ISU.

1.1.7.9.1 PAGE WIRING ALGORITHM THE FOLLOWING IS THE ALGORITHM USED TO WIRE PAGES:

1. ISSUE INHM.
2. SET WIRED BITS IN LOGICAL PAGE MAP (LMAP) ENTRY.
3. ISSUE ENBM.
4. IS PAGE IN MEMORY? IF SO, WE ARE DONE.
5. INVALIDATE STLB ENTRY.
6. REFERENCE THE PAGE.
7. GO TO STEP 4.

1.1.7.9.2 NEW WIRING/UNWIRING SUBROUTINES TWO SUBROUTINES, WIRE AND WIREA, ARE PROVIDED TO WIRE MEMORY ACCORDING TO THIS PROTOCOL. WIRE AND WIREA ARE SHORT-CALLABLE PROCEDURES THAT ARE PASSED A STRUCTURE CONTAINING THE 32-BIT VIRTUAL ADDRESS OF THE PAGE TO BE WIRED FOLLOWED BY THE 16-BIT PAGE MAP INDEX OF THAT PAGE. WIRE SETS THE WIRED BITS TO '01'B, AND WIREA ADDS '01'B TO THE WIRED FIELD. TWO SHORT-CALLABLE PROCEDURES, UWIRE AND UWIRES HAVE ALSO BEEN PROVIDED. THEY ACCEPT THE PAGE MAP INDEX AS A SINGLE ARGUMENT AND WORK BY CLEARING THE WIRED FIELD OR BY SUBTRACTING '01'B FROM IT.

ALL OPERATING SYSTEM CODE WHICH EXPLICITLY PERFORMS PAGE WIRING AND UNWIRING SHOULD CALL THESE PRIMITIVES. MODULES WHICH CALL SUCH PRIMITIVES AS MAPIO OR LOCKPG NEED NOT BE MODIFIED. THOSE

PRIMITIVES HAVE, THEMSELVES, BEEN APPROPRIATELY MODIFIED.

1.1.7.10 INSTRUCTIONS REQUIRING INTERPROCESSOR COMMUNICATION THE U-CODE FOR SOME INSTRUCTIONS HAS BEEN CHANGED FOR P850 CONFIGURATIONS IN ORDER TO CAUSE A MESSAGE TO BE SENT TO THE OTHER ISU. THESE INSTRUCTIONS MUST NEVER BE EXECUTED WHILE INHIBITED. FAILURE TO COMPLY WITH THIS RESTRICTION CAN RESULT IN A DEADLOCK SITUATION.

1.1.7.10.1 LIOT WHEN EXECUTED ON THE MASTER ISU, NO INTERPROCESSOR MESSAGE IS SENT. THE MASTER ISU'S IOTLB IS LOADED. WHEN EXECUTED ON THE SLAVE ISU, A MESSAGE IS SENT TO THE MASTER TO CAUSE IT TO LOAD ITS IOTLB. THE SLAVE ISU'S IOTLB IS UNAFFECTED.

1.1.7.10.2 PTLB THIS INSTRUCTION ALWAYS SENDS A MESSAGE TO THE OTHER ISU TO ENSURE THAT BOTH ISUS CONTAIN VALID INFORMATION IN THEIR STLBS. NOTE THAT THE ITLB INSTRUCTION DOES NOT SEND AN INTERPROCESSOR MESSAGE.

ITLB IS USED ONLY IN SITUATIONS WHERE THE LOCAL ISU'S STLB MUST BE CLEARED.

1.2 128 AMLC LINES A NUMBER OF CHANGES WERE MADE IN THE AMLC PROCESSING FOR PRIMOS RELEASE 18.2 AND BEYOND. THESE CHANGES ARE SPECIFICALLY:

1. THE MAXIMUM NUMBER OF AMLC TERMINAL LINES HAS BEEN INCREASED FROM 64 TO 128.

2. THE DMC INPUT TUMBLE TABLES ARE NOW CONFIGURED AT COLD START INSTEAD OF BEING A STATIC SIZE. THIS PERMITS LARGER INPUT BUFFERS FOR SYSTEMS NEEDING MORE INPUT DATA THROUGHPUT.

3. THE INTERRUPT AND AMLC INPUT PROCESSING HAVE BEEN IMPROVED IN PERFORMANCE.

4. THE AMLC INPUT PROCESSING DETECTS WHEN A DMC INPUT TUMBLE TABLE OVERFLOW HAS OCCURRED AND RECORDS SUCH EVENTS.

THIS DOCUMENT DESCRIBES A NUMBER OF CHANGES IN THE AMLC PROCESSING OF PRIMOS RELEASES 18.2 AND BEYOND. THESE CHANGES WERE MADE IN THE SYSTEM TO INCREASE THE NUMBER OF USABLE TERMINAL LINES TO 128 AND TO ALLOW HIGHER INPUT DATA THROUGHPUT NEEDED BY THE TERMINAL TRAFFIC EXPECTED WITH FUTURE SYSTEMS.

1.2.1 INCREASE IN AMLC TERMINAL LINES

THE NUMBER OF AMLC CONTROLLERS THAT A SINGLE CONFIGURATION

CAN SUPPORT HAS BEEN CHANGED FROM FOUR TO EIGHT CONTROLLERS. THIS MEANS THAT A SYSTEM CAN NOW SUPPORT UP TO 128 AMLC TERMINAL LINES. THE NTUSR AND NAMLC CONFIGURATION DIRECTIVES ARE USED TO CONFIGURE THE SYSTEM FOR THE NUMBER OF LINE BUFFERS REQUIRED. THE TOTAL NUMBER OF USER PROCESSES SUPPORTED HAS NOT BEEN CHANGED.

1.2.2 CONFIGURABLE TUMBLE TABLES

A NEW CAPABILITY HAS BEEN ADDED WHICH WILL CONFIGURE THE SIZE OF THE DMC INPUT TUMBLE TABLES AT COLD START. THE SIZE OF THE TUMBLE TABLES IS SET USING THE AMLIBL CONFIGURATION DIRECTIVE (SEE OTHER DOCUMENTATION DESCRIBING CONFIGURATION DIRECTIVES). THE DIRECTIVE CAN BE USED TO SET THE SIZE EXPLICITLY OF THE INPUT BUFFERS OR TO AUTOMATICALLY ALLOCATE THE MAXIMUM SIZE ALLOWED BY THE AVAILABLE BUFFER SPACE.

THE SYNTAX OF THE AMLIBL DIRECTIVE IS:

```
AMLIBL <BUFFER_SIZE>
```

THE PARAMETER <BUFFER_SIZE> IS AN OCTAL NUMBER WHICH REPRESENTS THE NUMBER OF WORDS ALLOCATED TO EACH INPUT BUFFER. THERE ARE TWO BUFFERS FOR EACH AMLC CONTROLLER AND ALL BUFFERS ARE MADE THE SAME SIZE. EXCEPT FOR THE SPECIAL VALUE OF ZERO DESCRIBED BELOW, THE NUMBER MUST BE GREATER THAN '20 (OCTAL). THE UPPER BOUND IS VARIABLE DEPENDING ON THE NUMBER OF CONTROLLERS CONFIGURED AND THE AMOUNT OF SPACE AVAILABLE IN THE SYSTEM FOR BUFFERS. IF <BUFFER_SIZE> IS ZERO OR OMITTED, THE SIZE OF THE BUFFERS IS AUTOMATICALLY CALCULATED AS THE MAXIMUM POSSIBLE. IF THE AMLIBL DIRECTIVE IS NOT SPECIFIED, THE DEFAULT BUFFER SIZE IS 60 (OCTAL).

IF THE <BUFFER_SIZE> ARGUMENT IS TOO SMALL, THE ERROR MESSAGE:

```
BAD AMLIBL PARAMETER (CINIT)
```

WILL BE GENERATED DURING COLD START INITIALIZATION. IF THE <BUFFER_SIZE> ARGUMENT IS TOO LARGE, THE ERROR MESSAGE:

```
INPUT BUFFERS TOO LARGE (AMINIT)
```

IS GENERATED AT COLD START INITIALIZATION. THE USER SHOULD MODIFY THE PARAMETER TO BE A VALUE WITHIN THE PERMISSIBLE RANGE AS DESCRIBED ABOVE.

1.2.3 INTERRUPT HANDLING IMPROVEMENTS

SOME CODE CHANGES WERE MADE IN THE INTERRUPT PROCESSING WHICH REDUCE THE NUMBER OF PIO'S NEEDED AND RESET THE DMC INPUT TUMBLE TABLE POINTERS AFTER EACH INPUT PROCESSING CYCLE. THIS LATER CHANGE

REDUCES THE NUMBER OF INPUT END-OF-RANGE INTERRUPTS GENERATED.

1.2.4 DMC TUMBLE TABLE OVERFLOW DETECTION SOME FUNCTIONALITY HAS BEEN ADDED WHICH PERMITS THE DETECTION OF DATA OVERFLOW IN THE DMC INPUT BUFFERS. THIS PROBLEM RESULTED IN DATA BEING INCORRECT OR LOST FOR RANDOM TERMINAL LINES. THE SIXTEEN BIT VALUE WHICH REPRESENTS THE NUMBER OF OVERFLOWS IS MAINTAINED AT THE SYMBOL AMLOVR WHICH CAN BE FOUND IN THE RING 0 SEG MAP FOR PRIMOS. CHECKING THIS VALUE WILL INDICATE WHETHER DATA IS BEING LOST FROM YOUR TERMINAL SYSTEM.

THE PROBLEM ABOVE CAN BE CORRECTED IN TWO WAYS. THE FIRST OF THESE IS TO ENLARGE THE DMC INPUT BUFFER SPACE VIA THE AMLIBL CONFIGURATION DIRECTIVE DISCUSSED ABOVE. THE OTHER IS TO MOVE SOME HIGH SPEED OR HIGH THROUGHPUT LINES FROM ONE AMLC CONTROLLER BOARD TO ANOTHER BOARD DISTRIBUTING THE INPUT LOAD.

1.3 ENHANCED FORCEW PRIMITIVE

THE FORCEW PRIMITIVE HAS BEEN ENHANCED GIVING THE USER THE OPTION OF OBTAINING THE STATUS OF DISK WRITE OPERATIONS TO A FILE.

WHEN A DISK WRITE ERROR OCCURS, ALL UNIT TABLES ASSOCIATED WITH THE FILE HAVE BIT 4 OF VSTAT SET. WHEN FORCEW IS CALLED WITH THE ERROR CODE PARAMETER GIVEN, IF AN ERROR CONDITION EXISTS E$DISK IS RETURNED AND THE ERROR BIT IS RESET. IF CODE IS NOT SUPPLIED NO ACTION IS TAKEN, THE ERROR BIT IS NOT RESET AND MAY BE SENSED AT A LATER TIME.

NOTE: THE ERROR CONDITION IS SET IN ALL UNIT TABLES ASSOCIATED WITH THE FILE REGARDLESS OF WHO CAUSED THE ACTUAL ERROR.

1.3.1 FORCEW CALL

FORCES AN IMMEDIATE DISK UPDATE UNDER PRIMOS. A USER NOW HAS THE ABILITY TO DETECT A DISK ERROR BY USING THE OPTIONAL CODE ARGUMENT ADDED TO THE CALLING SEGUENCE.

USAGE:

```
CALL FORCEW (0, FUNIT, CODE)
```

FUNIT

IS THE UNIT ASSOCIATED WITH THE FILE TO BE IMMEDIATELY UPDATED TO THE DISK.

CODE

THIS IS AN OPTIONAL ARGUMENT, IF NOT OMITTED RETURNS A STANDARD ERROR CODE. POSSIBLE ERROR CONDITIONS ARE:

E$DISK	DISK ERROR DETECTED
E$FIFC	REMOTE FAM DOES NOT SUPPORT THIS FUNCTIONALITY

1.3.2 DISK DRIVER IMPROVEMENTS

AS OF REV. 18.2, TWO MAJOR IMPROVEMENTS HAVE BEEN MADE TO THE DISK DRIVER. A TIMEOUT MECHANISM HAS BEEN IMPLEMENTED TO ALLOW AUTOMATIC RECOVERY FROM DISK CONTROLLER HANGS. DISK OPERATIONS WHICH DO NOT COMPLETE IN A TIMELY MANNER WILL BE ABORTED, AND THE DISK CONTROLLER WILL BE REINITIALIZED. AS WITH OTHER DISK ERRORS, 10 RETRIES WILL BE ATTEMPTED. THE STATUS REPORTED ON A DISK TIMEOUT WILL BE 000001.

NEW METERING INFORMATION HAS BEEN ADDED TO THE DISK DRIVER. I/O TIMES AND COUNTS ARE MAINTAINED ON A PER-DRIVE BASIS FOR ALL DISK DRIVES CONNECTED TO 4004/4005 CONTROLLRS AND 4002 CONTROLLERS. NO SUCH INFORMATION IS KEPT FOR FLOPPY DISKS. THIS INFORMATION CAN BE FOUND AT THE EXTERNAL SYMBOL DKMETR. ITS FORMAT IS DESCRIBED BY THE FOLLOWING:

```
DCL  1  DKMETR EXT,

          2  G-WAITS FIXED BIN (31),

          2  DMA_OVERRUNS FIXED BIN (31),

          2  HANGS FIXED BIN (31),

          2  CO_TIME (1:4,0:3) FIXED BIN (31),

          2  CO_COUNT (1:4,0:3) FIXED BIN (31);
```

G-WAITS IS THE NUMBER OF TIMES A CALLER TO THE DISK DRIVER HAD TO WAIT FOR A FREE REQUEST QUEUE ENTRY.

DMA_OVERRUNS IS THE NUMBER OF TIMES DISK CONTROLLERS TOOK DMA OVERRUN ERRORS.

HANGS IS COUNT OF THE NUMBER OF RECOVERIES FROM CONTROLLER HANGS, AS DETECTED BY THE TIMOUT MECHANISM.

CO_TIME AND CO_COUNT ARE THE TOTAL I/O TIMES AND OPERATION COUNTS FOR EACH OF FOUR POSSIBLE DRIVES ON EACH OF FOUR CONTROLLERS. THE POSSIBLE CONTROLLERS ARE, RESPECTIVELY, 2

4004/4005 CONTROLLERS (I/O ADDRESSES :26 + :27) AND 2 4002 CONTROLLERS (ADDRESSES :21 + :23).

THE FORMAT OF THIS METERING INFORMATION MAY BE SUBJECT TO CHANGE IN FUTURE REVISIONS OF PRIMOS.

1.3.3 TIMERS

TWO NEW TIMER FUNCTIONS HAVE BEEN ADDED TO THE OPERATING SYSTEM. ONE IS A CPU WATCH DOG TIMER AND THE OTHER IS A REAL TIME WATCHDOG TIMER. BOTH ARE AVAILABLE BY CALLING LIMIT$. BOTH TIMERS CAN BE SET AND READ. THE CPU TIMER IS MEASURED IN SECONDS AND THE REAL TIME TIMER IS IN MINUTES. IF EITHER TIMER EXPIRES A SIGNAL IS GENERATED. THE CPU TIMER SIGNAL IS 'CPU_TIMER$' AND THE REAL TIME TIMER IS 'ALARM$'.

1.3.4 PAGING DEVICE COMPRESSION AS OF REV. 18.2, PAGING DEVICE RECORDS ARE GIVEN TO SEGMENTS ON AN AS-NEEDED BASIS. THUS, A SEGMENT WITH ONLY HALF OF ITS PAGES USED WOULD NEED PAGING DEVICE SPACE FOR ONLY 32 PAGES. THIS FEATURE ALLOWS ABOUT TWICE AS MANY SEGMENTS TO BE USED. NOTE THAT THE TOTAL VIRTUAL ADDRESS SPACE IS STILL THE SAME. NOW, HOWEVER, THIS SPACE CAN BE APPORTIONED AMONGST A GREATER NUMBER OF SEGMENTS.

1.3.5 MAGLIB SUPPORT A NEW GATE, ASNMT$, HAS BEEN ADDED TO PRIMOS. IT ALLOWS A USER PROGRAM TO PERFORM MAG TAPE ASSIGNMENT, UNASSIGNMENT, AND TAPE MOUNT REQUESTS. IT IS NOT BEING GENERALLY RELEASED; I.E. ITS USE IS RESTRICTED TO PRIME SUPPLIED SOFTWARE, SUCH AS MAGLIB. FOR THIS REASON, A DYNT (DIRECT CALL DEFINITION) WILL NOT BE ADDED TO THE FORTRAN LIBRARY FOR IT.

THE NEW GATE HAS THE FOLLOWING CALLING SEQUENCE:

```
 DCL ASNMT$ ENTRY (FIXED BIN, CHAR(*) VAR, FIXED BIN, FIXED B
IN);

    CALL ASNMT$ (KEY, COMMAND, DEVICE, CODE);
```

KEY

IS 1 IF THE OPERATION IS "ASSIGN" OR "MOUNT", AND 0 IF "UNASSIGN". (INPUT)

COMMAND

IS THE ASSIGN OR UNASSIGN COMMAND, EXACTLY AS IT WOULD HAVE BEEN ENTERED AT COMMAND LEVEL BY THE USER. EXAMPLE: "ASSIGN MTX -ALIAS MT1". (INPUT)

DEVICE

IS THE TAPE DEVICE (UNIT) NUMBER THAT WAS ASSIGNED, UNASSIGNED, OR CAUSED AN ERROR. THIS QUANTITY IS A LOGICAL OR PHYSICAL DEVICE NUMBER, AS APPROPRIATE. FOR EXAMPLE, IF COMMAND IS "AS MTX -ALIAS MT1" AND

PHYSCIAL DEVICE MTO IS ASSIGNED, DEVICE HAS THE VALUE 0. IF THE USER HAS ALREADY ASSIGNED LOGICAL DEVICE MTO, A DEVICE IN USE ERROR CODE IS RETURNED, AND DEVICE HAS THE VALUE 1 (THE LOGICAL DEVICE THAT WAS IN USE). (OUTPUT)

CODE

IS A STANDARD ERROR CODE IF POSITIVE, AND INDICATES THAT THE REQUESTED OPERATION COULD NOT BE PERFORMED. IF ZERO OR NEGATIVE, INDICATES SUCCESSFUL COMPLETION AS FOLLOWS: 0 MEANS ASSIGN OK, -1 MEANS MOUNT COMPLETE, AND -2 MEANS UNASSIGN OK. (OUTPUT)

THE COMMANDS ACCEPTED IN THE COMMAND ARGUMENT ARE THOSE THAT EXISTED AT REVISION 17, WITH THE FOLLOWING EXTENSION.
THE CONTROL ARGUMENT -MOUNT MAY BE USED IN AN ASSIGN COMMAND TO REQUEST THE OPERATOR TO MOUNT A NEW TAPE ON A DRIVE THAT IS ALREADY ASSIGNED. IF OPERATOR TAPE HANDLING IS DISABLED, THE DEVICE REMAINS ASSIGNED AND NOTHING ELSE HAPPENS. THE SYNTAX IS:

```
    ASSIGN MTI -MOUNT <OTHER ARGS>
OR
    ASSIGN -ALIAS MTI -MOUNT <OTHER ARGS>
```

THE FIRST FORMAT IS USED WHEN THE PHYSCIAL DEVICE NUMBER IS KNOWN. THE LATTER IS USED WHEN THE LOGICAL DEVICE NUMBER IS KNOWN.

WARNING: THIS INTERFACE MAY BE CHANGED WITHOUT PRIOR NOTICE.

1.3.6 FAM II FAM II IS FASTER AND MORE RELIABLE THAN FAM I AND HENCE IT IS TO THE USERS BENEFIT TO CONVERT TO IT AS SOON AS POSSIBLE. REV 18.2 AND 19 WILL SUPPORT FAM COMPATIBILITY MODE WHICH WILL ALLOW SYSTEMS RUNNING FAM II TO COMMUNICATE WITH SYSTEMS RUNNING FAM I. THIS ALLOWS NETWORK A GRADUAL PHASE OVER FROM FAM I TO FAM II. FAM COMPATIBILITY MODE WILL END AT REV 20.
UNDER FAM II EACH USER ACCESSING REMOTE FILES GETS EXCLUSIVE ACCESS TO A SLAVE PROCESS ON THE SYSTEM WHERE THE FILES RESIDE. THE SLAVES START UP FROM AINIT BY USING A BATCH$ CALL. HENCE, THEY NEED A PHANTOM FILE AND A FIXED PLACE TO FIND IT. THE TOP LEVEL UFD PRIMENET* MUST EXIST ON THE SYSTEM DISK PARTITION AT SYSTEM START UP, AND MUST CONTAIN THE FILE SLAVE.COMI.
FAM II MUST BE CONFIGURED VIA NETCFG AND THE CONFIG DIRECTIVE NSLUSR MUST BE USED. FAM II IS BROUGHT UP AUTOMATICLY AT SYSTEM COLD START TIME. NO SPECIAL RING 3 PROGRAMS OR SOFTWARE NEED TO BE INSTALLED SINCE IT IS INTEGRATED WITH THE OPERATING SYSTEM.
IN ORDER TO RUN FAM COMPATABILITY MODE, FAM I MUST BE INSTALLED. THE SOURCE, OBJECT, RUN, AND COMMAND FILES FOR THE FILE ACCESS MANAGER FAM VERSION I ARE CONTAINED IN THE

DIRECTORY CHAIN PRINET>FAM OR X.25>FAM. THE FILES IN THE UFD FAM THAT ARE OF SPECIAL IMPORTANCE TO THE FAM INSTALLER ARE AS FOLLOWS:

PH_FAM	PHANTOM COMMAND FILE
#FAM	SEGMENTED RUN FILE

TO INSTALL THE FAM, THE FOLLOWING MUST BE DONE:

1) CREATE A TOP LEVEL UFD CALLED FAM (WHICH MAY BE LOGGED INTO). THIS UFD MUST NOT HAVE A PASSWORD.

2) FUTIL THE FILES PH_FAM AND #FAM TO THE NEWLY CREATED UFD.

TO ENABLE FAM, SIMPLY DO ONE OF THE FOLLOWING:

1) LOGIN UNDER THE USERNAME OF FAM:

```
OK, LOGIN FAM

FAM (XX) LOGGED IN AT ...
OK, SEG #FAM
```

FAM WILL NOW RUN, AND NO FURTHER COMMANDS WILL BE READ FROM THE TERMINAL.

2) RUN THE FAM AS A PHANTOM:

```
OK, PH FAM>PH_FAM

PHANTOM IS USER ...
OK,
```

TO ENABLE FAM TO COMMUNICATE WITH A PARTICULAR REMOTE NODE, SEE SEPARATE DOCUMENT DESCRIBING NETCFG.

1.3.6.1 NEW FAM II ERROR MESSAGES

1.3.6.1.1 CAN'T ATTACH TO PRIMENET* (AINIT) THE PRIMINET* UFD WAS UNFOUND OR UNUSABLE.

1.3.6.1.2 CAN'T START SLAVE (AINIT) THE NPX SLAVE WAS NOT STARTED (PROBABLY SLAVE.COMI WAS NOT FOUND).

1.4 T$AMLC

PRIMOS OPERATING SYSTEM HAS LONG SUPPORTED THE CONCEPT OF

"ASSIGNABLE AMLC LINES". THIS FUNCTIONALITY ALLOWS A USER TO OWN AND CONTROL THE I/O FOR A PARTICULAR AMLC LINE. IN MANY INSTALLATIONS, PERIPHERAL DEVICES SUCH AS PRINTERS, PLOTTER ETC. ARE USED VIA ASSIGNABLE AMLC LINES.

THIS DOCUMENT DESCRIBES IMPROVEMENTS TO THE INTERNAL MECHANISM WITHIN PRIMOS FOR ASSIGNED LINES. THESE CHANGES ARE EFFECTIVE AT REV 18.2. THIS DOCUMENT IS WRITTEN FOR READERS ALREADY FAMILIAR WITH THE ASSIGNED AMLC LINE FUNCTIONALITY OF PRIMOS. PLEASE REFER TO PRIMOS COMMAND MANUAL AND THE SYSTEM ADMINISTRATOR'S GUIDE FOR ADDITIONAL INFORMATION.

```
----------
+ T$AMLC +
----------
```

USAGE:

CALL T$AMLC (LINE, USER_BUF_ADDR, CHAR_COUNT, KEY, STAT_VEC, CHAR_POS_ARG, ERRCODE);

DCL T$AMLC ENTRY (FIXED BIN, PTR, FIXED BIN, FIXED BIN, (2) FIXED BIN, FIXED BIN, FIXED BIN);

DESCRIPTION

THIS ROUTINE IS AN EXISTING PRIMOS DIRECT ENTRANCE CALL. IT PERFORMS RAW DATA MOVEMENT AND PROVIDES STATUS INFORMATION ABOUT ASSIGNED AMLC LINES. THIS ROUTINE TRANSFERS CHARACTERS TO AND FROM THE CALLER'S BUFFER TO A DESIRED ASSIGNED LINE'S BUFFER. THE CALLER MUST OWN THE DESIRED LINE, I.E., THE CORRESPONDING LBT ENTRY MUST CONTAIN THE CALLER'S USER NUMBER.

ENHANCEMENTS

----

1) THE 80 CHARACTER LIMITATION HAS BEEN REMOVED. THE CALLER SPECIFIES THE MAXIMUM BY THE CHAR_COUNT ARGUMENT.

2) FUNCTION CALLS WITHIN T$AMLC HAVE BEEN CHANGED TO BE ALL SHORT CALLS AS OPPOSED TO LONG PCL CALLS, THUS INCREASING PERFORMANCE.

3) BEFORE THIS RE-WRITE, SCHAR AND GCHAR ROUTINES (BOTH LONG PCL CALLS) WERE USED TO PERFORM CHARACTER MOVEMENTS BETWEEN THE AMLC LINE AND CALLER BUFFERS. T$AMLC IS RE-WRITTEN IN PL/P AND THE SUBSTR FUNCTION (IN-LINE CODE) IS NOW USED.

(USER VISIBLE IMPROVEMENTS ARE NOTED BY REVISION BARS IN THE

FOLLOWING TEXT)

ARGUMENTS AND KEYS

LINE
DESIRED AMLC LINE NUMBER.

USER_BUF_ADDR
ADDRESS (POINTER) TO THE CALLER'S BUFFER.

CHAR_COUNT
DESIRED NUMBER OF CHARACTERS TO MOVE. NO MAXIMUM LIMIT IS ENFORCED.

```
KEY
        1 => INPUT CHAR_COUNT CHARACTERS.

        2 => INPUT CHAR_COUNT CHARACTERS OR UNTIL .NL.
             STAT_VEC(1) = ACTUAL NUMBER OF CHARACTERS READ.

        3 => OUTPUT CHAR_COUNT CHARACTERS.
             MAXIMUM = CHAR_COUNT.  THIS KEY ASSURES THE  CALLER  THAT
             CHAR_COUNT CHARACTERS  WILL  BE  OUTPUT.  FOR EXAMPLE, AN
             ERROR IS NOT RETURNED  IF  THE  LINE'S  INPUT  OR  OUTPUT
             BUFFER IS  SMALLER  THAN  CHAR_COUNT.  T$AMLC WILL OUTPUT
             BLOCKS OF  DATA  FROM  THE  CALLER'S  BUFFER  INTO  THE
             AVAILABLE  ROOM  IN  THE  LINE'S  OUTPUT  BUFFER  UNTIL
             CHAR_COUNT IS EXHAUSTED.  A ONE  SECOND  WAIT  IS  ISSUED
             BETWEEN OUTPUT CHUNKS TO ALLOW TIME FOR THE LINE'S OUTPUT
             BUFFER TO  CLEAR.  IN  MOST CASES, THE ENTIRE CHAR_COUNT
             SHOULD BE OUTPUT AT ONCE.

        4 => STAT_VEC(1) = NUMBER OF CHARACTERS IN INPUT BUFFER.
             STAT_VEC(2) = STATE OF CARRIER.

        5 => RETURN STATUS OF OUTPUT BUFFER.
             STAT_VEC(1) = 1 IF ROOM FOR CHAR_COUNT IN OUTPUT  BUFFER.
             STAT_VEC(1) = 0 IF NOT ENOUGH ROOM FOR CHAR_COUNT.
             STAT_VEC(2) = STATE OF CARRIER.

        6 => INPUT ALL AVAILABLE CHARACTERS IN THE INPUT BUFFER.
             MAXIMUM =  CHAR_COUNT.  THIS  KEY  WILL  PLACE  ALL  THE
             AVAILABLE CHARACTERS IN THE LINE'S INPUT BUFFER INTO  THE
             CALLER'S BUFFER.
             STAT_VEC(1) = NUMBER OF CHARACTERS ACTUALLY INPUT.

        7 =>  RETURN ADDITIONAL OUTPUT BUFFER STATUS (REFER TO KEY 5).
             STAT_VEC(1) = AMOUNT OF CHARACTER SPACE REMAINING IN  THE
                                 OUTPUT BUFFER.

        8 => FLUSH INPUT BUFFER.
        9 => FLUSH OUTPUT BUFFER.
```

10 => FLUSH BOTH OUTPUT AND INPUT BUFFERS.

11 => OUTPUT CHARACTERS TO AVAILABLE ROOM IN OUTPUT.
THIS KEY WILL OUTPUT AS MANY CHARACTERS AS POSSIBLE INTO THE LINE'S OUTPUT BUFFER. A WAIT WILL NOT BE DONE TO EXHAUST CHAR_COUNT.
STAT_VEC(1) = NUMBER OF CHARS THAT WERE NOT SUCCESSFULLY OUTPUT, I.E. STAT_VEC(1) = 0: MEANS ALL CHARACTERS WERE OUTPUT.

STAT_VEC:
TWO WORD STATUS VECTOR USED BY CERTAIN KEYS.

CHAR_POS_ARG:
THE CALLER MAY WISH TO INDICATE A STARTING POSITION WITHIN THE BUFFER ADDRESSED BY USER_BUF_ADDR. CHAR_POS_ARG APPLIES FOR BOTH INPUT AND OUTPUT KEYS. THIS IS AN OPTIONAL ARGUMENT. IF OMITTED, THE DEFAULT IS TO START WITH THE FIRST CHARACTER. NOTE: IF CHAR_POS_ARG IS USED, THE FIRST CHARACTER POSITION SHOULD BE INDICATED BY ONE (THERE IS NO CHARACTER AT POSITION ZERO). ALSO, CHAR_POS_ARG IS NOT UPDATED WITHIN T$AMLC.

ERRCODE:
OPTIONAL ARGUMENT TO RETURN ERROR STATUS. IF ERRCODE IS PRESENT, ERROR MESSAGES WILL NOT BE PRINTED AT THE CALLER'S PHYSICAL TERMINAL.

2 CORRECTED REVISION 18.1 PROBLEMS

2.1 COMO CORRECT INCORRECT COMMENT IN COMO$$.FTN REGARDING :200 KEY. TAR #34341.

2.2 CPL TO ALLOW A FULL 1024 CHARACTER VARIABLE NAME TO BE USED. ALSO, TO ALLOCATE MORE MEMORY ONLY WHEN IT IS NEEDED IN ORDER TO REDUCE THE BASIC WORKING SET.

2.3 CPL MAKE CPL PHANTOMS COMPATIBLE WITH EXTERNAL LOGIN CONVENTIONS.

2.4 CPL PROVIDE ANOTHER GATE FOR CPL AND REGULAR PHANTOMS. FIX BUG IN GLOBAL VARIABLE ROUTINES.

2.5 REST$$ WHEN REST$$ IS CALLED WITH A KEY OF K$WRIT USING AN OPEN FILE UNIT TO RESTORE FROM, THE OPTIONAL SEGMENT NUMBER ARGUMENT IS IGNORED (ALWAYS RESTORES THE IMAGE INTO SEG 4000).

*
574 (RFTNLIB)

LIBRARIES NOW CONFORM TO THE REV 19.0 STANDARD.

*
575 (RUNOFF)

THIS DOCUMENT DESCRIBES CHANGES TO RUNOFF FOR REV 18.2.

THE .TAB PROCESSING HAS BEEN CHANGED SO THAT A TAB CAN BE SET IN COLUMN 1. (TAR 82213)

THE USER'S ERASE AND KILL CHARACTERS WILL NOW BE RECOGNIZED IN ALL SITUATIONS. (TARS 34759 AND 23936)

A WARNING WILL BE ISSUED WHEN AN ALPHA IS USED AS A COMMAND PARAMETER WHERE A NUMERIC IS EXPECTED. (TAR 23928)

*
576 (SEG)

SUBJECT: SEG
RELEASE: REV18.2
DATE: MARCH 13, 1981

1 NEW FUNCTIONALITY

- A NEW GROUP TYPE HAS BEEN ADDED WHICH WILL ENHANCE THE F77 COMPILER PERFORMANCE IN SPEED AND CODE SIZE WHEN INITIALIZING LARGE COMMON BLOCKS.

2 PROBLEMS FIXED

- IF A USER TRIES TO USE SEGMENT 4035, WHICH IS USED BY SEG INTERNALLY FOR ITS OWN SYMBOL TABLE, HE WILL NOW BE TOLD DURING THE LINKING SESSION THAT THIS IS ILLEGAL AND SEG WILL ABORT TO PRIMOS, RETURNING AN ERROR SEVERITY CODE.

3 OUTSTANDING PROBLEMS

POLAR NO.	DESCRIPTION
24250	NOT ALL LIB ROUTINES ARE LOADED WITH A SINGLE LIB COMMAND.
34484	DELETE SUBCOMMAND ABORTS COMMAND FILES IF FILE DOES NOT EXIST.
36524	SEG DOES NOT REPORT ERROR IF USER LOADS A SEG FILE IN THE VLOAD SUBPROCESSOR.
36890	MULTIPLE INDIRECT FROM FTN.

4 ENVIRONMENT

. THIS VERSION OF SEG REQUIRES REV18 PRIMOS AND REV18 PFTNLIB

5 INSTALLATION AND BUILD PROCEDURES

. STANDARD INSTALLATION AND BUILD PROCEDURE.

*
577 (SPL)

THIS IS AN INTERNAL TOOL -- IT IS NOT FOR CUSTOMER USE

*
578 (SPOOL)

AT REV 18.2 SPOOL SUPPORTS THE NEW "BAND" PRINTER. THIS PRINTER USES AN ELECTRONIC VERTICAL FORMAT UNIT (EVFU) TO DETERMINE THE FORM LENGTH, SO ITS ENVIRONMENTS MUST ALWAYS HAVE EVFU TURNED ON. THE DEVICE PARAMETER MAY BE PR0, PR1, PR2 OR PR3. SINCE THE FORMAT OF THE EVFU IS DIFFERENT FROM THE ONE IN THE 300 LPM PRINTER-PLOTTER, THESE TWO DEVICES MUST BE DISTINGUISHED. A NEW ENVIRONMENT PARAMETER CALLED "TYPE" HAS BEEN DEFINED FOR THIS PURPOSE. THE BAND PRINTER IS TYPE 1; THE PRINTER-PLOTTER IS TYPE 0.

*
579 (SPS)

THIS IS AN INTERNAL TOOL -- IT IS NOT FOR CUSTOMER USE

*
580 (VFTNLIB)

AN ENTRY WAS ADDED TO THE LIBRARY WHICH DETERMINES THE REV NUMBER AND DOT REV NUMBER OF THE CURRENT LIBRARY. THIS ROUTINE WILL SHOW UP IN ALL SUBSEQUENT SEG LOADS (AS WELL AS WHEN EDBING THE LIBRARIES) AS THE ROUTINE:

"F###NNNN"

WHERE '###' IS THE CURRENT REV AND DOT REV (IE - '182') AND 'NNNN' IS A SUB IDENTIFIER (SUCH AS 'BKSP') WHICH FURTHER IDENTIFIES THE LIBRARY. THIS MODULE TAKES UP NO SPACE AND GENERATES NO LINKS - IT APPEARS ON YOUR MAPS ONLY.

THIS VERSION TRACKS ALL PREVIOUS VERSIONS, AND THE FOLLOWING BUGS HAVE BEEN FIXED:

BACKSPACE ON A BINARY RECORD NOW WORKS.

PACKAGE # IS NOW PRINTED OUT IN 2 DIGITS (TAR 27108).
THE FOLLOWING ARE KNOWN TAR'S:

BACKSPACE AFTER ENDFILE WHILE USING TAPE STILL DOESN'T WORK (THIS IS A MAJOR CHANGE & WILL COME OUT AT A MAJOR REV RELEASE).

DON'T TRY TO RAISE A COMPLEX NUMBER TO A POWER UNLESS YOU CAN FIND THE ROUTINE F$SCIPWR. FLOATING THE POWER WON'T WORK. MAKING BOTH NUMBERS COMPLEX*32, HOWEVER, SHOULD WORK. WATCH THIS SPACE FOR FUTURE DEVELOPMENTS.

*
581 (BASIC)

ALL FILES FOR BASIC/DBASIC AND NUMBER ARE MODIFIED AND/OR REORGANIZED IN THE FOLLOWING MANNER:

THE SOURCE FILES ARE ORGANIZED IN SUB-UFDS BASICSRC,DBASICSRC AND NUMBERSRC UNDER UFD BASICSRC IN PARTITION D1. ALL SOURCE FILE NAMES ARE APPENDED WITH APPROPRIATE SUFFICES.

BOTH BASIC/DBASIC AND NUMBER ARE BUILT BY SINGLE FILE BASIC.BUILD.CPL WHICH IS ALSO IN UFD BASICSRC.

THE COMMAND FILE C_INSTALLBASIC IN UFD BASIC ON THE C1 PARTITION WILL INSTALL ALL 3 PRODUCTS.

ALL FILES NOW INCLUDE HEADERS CARRYING COPYRIGHT INFORMATION.

*
582 (BASICV)

BASICV FOR 18.2 CONTAINS THE FOLLOWING BUG FIXES

20240 - 'ENTER' CAUSING TOO MUCH OVERHEAD. FIXED.

20757 - MIDAS LOCK PROBLEM ON OPENING FILE. FIXED.

21838 - 'ENTER' CAUSING TOO MUCH OVERHEAD. FIXED.

21594 - 'ENTER' CAUSING TOO MUCH OVERHEAD. FIXED.

27502 - 'RFSEQUENCE' UNABLE TO DETECT REM STATEMENT. FIXED.

30069 - 'ENTER' CAUSING TOO MUCH OVERHEAD. FIXED.

33521 - 'RESEQUENCE' UNABLE TO DETECT QUOTED STRINGS. FIXED.

35152 - BAD ERROR MESSAGE ON 'RENAME' COMMAND. FIXED.

35334 - DIFICULTIES WITH LARGE PROGRAMS. FIXED.

35551 - BASICV FAILS TO ACCEPT NEW FILENAMES. FIXED.

*
583 (COBOL)

REVISION 18.2 COBOL

THE REVISION 18.2 COBOL COMPILER AND RUN-TIME LIBRARIES (VCOBLB AND NVCOBLB) HAVE BEEN UPDATED TO REFLECT THE MASTER DISK SOFTWARE STANDARDS.

IN ADDITION, TAR #29442 HAS BEEN FIXED AT REVISION 18.2.

THIS PROBLEM OCCURRED WHEN THE FOLLWING ACTIONS WERE PERFORMED ON AN INDEXED MIDAS FILE OPENED FOR I-O:

READ (FOUND)

READ (NOT FOUND)

WRITE (MIDAS 33 ERROR)

THE MIDAS RUN-TIME SYSTEM WOULD DISPLAY THE "MIDAS 33 ERROR" INSTEAD OF ALLOWING THE COBOL PROGRAM TO PROCESS THE INVALID KEY CLAUSE AND RETURN THE APPROPRIATE FILE-STATUS CODE.

*
584 (DBG)

SUBJECT: DBG
RELEASE: 18.2
DATE: MARCH 13, 1981

1 NEW FUNCTIONLITY

1) EVALUATOR SUPPORT FOR FULL PL/1, INCLUDING:

   - NEW DATATYPES: AREA, OFFSET, COMPLEX, BIT VARYING, FIXED BIN WITH SCALE FACTOR.
   - FULL PL/1 BUILTIN FUNCTIONS.
   - ARITHMETIC DATATYPES ACCEPTED BY LOGICAL AND CONCATENATE OPERATORS.
   - ARGUMENT CONVERSION OF VARIABLE EXTENT ARRAYS AND STRINGS.

2) SUPPORT FOR PASCAL.

3) SUPPORT FOR EPF'S (EXECUTABLE PROGRAM FORMAT).

4) NEW OPTIONS ON THE MACRO COMMAND TO CHANGE THE NAME OF A MACRO, AND TO DISABLE AND REENABLE MACRO EXPANSION.

5) NEW ENTRY / EXIT VALUE TRACING FACILITY WHICH CHECKS FOR CHANGES IN THE VALUES OF WATCHED VARIABLES ONLY AT ENTRY TO AND EXIT FROM EACH PROCEDURE.

6) NEW -ALL OPTION ON THE UNWATCH COMMAND TO REMOVE ALL VARIABLES FROM THE WATCHLIST.

7) NEW OPTIONS ON THE SAVESTATE COMMAND TO SAVE MACROS, BREAKPOINTS, AND/OR TRACEPOINTS INDEPENDENTLY.

8) NEW -SYM_LIST OPTION ON THE HELP COMMAND TO LIST ALL SYNTAX SYMBOLS DEFINED BY THE HELP COMMAND.

9) REENTER$ CONDITION IS NOW HANDLED BY DBG.

2 PROBLEMS FIXED

2.1 TARS AND POLERS FIXED

- TAR#12503
  IT IS NOW POSSIBLE TO WATCH AN ARRAY WITH A STAR EXTENT SUBSCRIPT.

- TAR#24279
  IN FORTRAN, EVALUATION OF A REAL VARIABLE RAISED TO AN INTEGER POWER NO LONGER FAILS WHEN THE VALUE OF THE REAL IS NEGATIVE.

- PSF#32357
  PL/1 MOD BUILTIN IS NOW AVAILABLE.

- PSF#33889
  F77 LOGICAL*1 ARRAYS ARE NOW EVALUATED CORRECTLY.

- PSF#34802
  A SPACE IS NO LONGER REQUIRED AROUND THE FORTRAN DIVIDE AND MULTIPLY OPERATORS.

- PSF#35293
  AN ARGUMENT MISMATCH WARNING IS NO LONGER GIVEN FOR VARIABLE OR STAR EXTENT PARAMETERS.

- PSF#35294
  THE PL/1 CONCATENATE OPERATOR NOW ACCEPTS ARITHMETIC DATATYPES AS OPERANDS.

2.2 OTHER BUGS FIXED

1) THE PH_LOGO$ CONDITION IS NOW IGNORED BY DBG.

2) USER QUITS ARE NOW PROCESSED DURING EXECUTION OF THE LET COMMAND.

3) THE "EXTERNAL" ATTRIBUTE OF A MEMBER OF AN EXTERNAL PL/1 STRUCTURE IS NOW PRINTED BY THE TYPE COMMAND.

4) THE LOADSTATE COMMAND NOW CLOSES THE LOADSTATE FILE WHEN EXECUTED FROM A COMINPUT FILE.

5) WHEN RESUBMIT IS THE FIRST COMMAND ENTERED IN DBG, THE "PREVIOUS" COMMAND LINE IS NOW BLANKS.

6) INFINITE RECURSION ON MACRO EXPANSION IS NOW DETECTED.

7) EVALUATION OF PL/1 SYMBOLS IMPLICITLY BASED ON A LINK BASE RELATIVE POINTER NOW WORKS CORRECTLY WHEN ENTRY BREAKPOINTS ARE SET.

8) THE SCOPING OF AN EXTERNAL VARIABLE WITH THE SAME NAME AS A BUILTIN FUNCTION IS NOW HANDLED CORRECTLY.

9) EVALUATION OF PICTURED VARABLES IN A DIFFERENT PRINT MODE NOW WORKS.

2.3 INTERNAL MODIFICATIONS

1) TWO NEW "MAIN" ROUTINES HAVE BEEN ADDED.

2) THE COMMAND LINE IS NOW PARSED USING CL$PIX.

3) ALL ERROR HANDLING HAS BEEN CENTRALIZED.

4) THE EVALUATOR HAS BEEN REORGANIZED AND SOME OBSOLETE CODE REMOVED.

3 OUTSTANDING PROBLEMS

THE FOLLOWING DBG TARS AND POLERS ARE OUTSTANDING:

- PSF#28711
  CANNOT EVALUATE OR ASSIGN LARGE NEGATIVE CONSTANTS.

- PSF#30330
  THE RESULT OF A FTN COMPLEX FUNCTION IS INCORRECT.

- PSF#32661
  CANNOT EVALUATE AN ARRAY OF LABEL CONSTANTS.

- PSF#34352
  FILE UNITS NOT HANDLED CORRECTLY BY DBG IF MACHINE IS CONFIGURED FOR LESS THAN 62 FILE UNITS.

4 ENVIRONMENT

REVISION 18.2 OF DBG REQUIRES A REVISION 18.0 OR LATER VERSION OF PRIMOS.

5 INSTALLATION AND BUILD PROCEDURES

STANDARD: BUILD USING DBG.BUILD.CPL, INSTALL USING DBG.INSTALL.COMI, AND SHARE USING DBG.SHARE.CPL.

*
585-587 (DPTX-DSC, DPTX-TCF, DPTX-TSF)

FOR DETAILED INFORMATION ON CHANGES
TO REV 18.2 DPTX, SEE

```
DPTX-DSC>INFO
DPTX-TCF>INFO
DPTX-TSF>INFO
```

*
588 (FED)

SUBJECT: FED

RELEASE: 1.0

DATE: 3RD MARCH 1981

FUNCTIONALITY

AS THIS IS A NEW PRODUCT, PLEASE REFER TO THE FED USER GUIDE.

ENVIRONMENT

FED 1.0 REQUIRES FORMS 18.2, AS THE LATTER CONTAINS THE ENHANCED RUN-TIME PLUS CALLABLE FDL AND FAP.

BUILD AND INSTALLATION PROCEDURES

BUILD:

THE BUILD PROCEDURE (IN FEDSRC>FED.BUILD.CPL) USES THE FOLLOWING PRODUCTS:

```
          COMPILERS:  PLP
                      FTN
                      PMA
                      FDL  (MUST BE 18.2, OR LATER)

          LIBRARIES:  VFORMS  (MUST BE 18.2, OR LATER)
                      VAPPLB
                      VSPOO$

          UTILITIES:  EDB
                      SEG
                      FUTIL
```

NOTE THAT THIS IMPLIES THAT AT 18.2, FORMS MUST ALREADY HAVE BEEN BUILT AND INSTALLED BEFORE FED CAN BE BUILT.

INSTALLATION

INSTALLATION OF FED REQUIRES THAT THE UFD FED* ALREADY EXISTS, AND THAT A WORKING FORMS SYSTEM EXISTS.

FED.INITINSTALL.COMI ADDS FED'S FORMS TO THE FORMS LIBRARY, COPIES THE PROGRAM AND SHARE FILE TO SYSTEM AND CMDNC0, AND INSTALLS THE FIRST VERSION OF THE HELP TREE ON FED*.

FED.INSTALL.COMI REPLACES FED'S FORMS IN THE FORMS LIBRARY, COPIES THE PROGRAM AND SHARE FILE AS BEFORE, AND INSTALLS THE

LATEST VERSION OF THE HELP TREE ON FED*.

*
589 (FORMS)

SUBJECT: FORMS

RELEASE: 18.2

THIS DOCUMENT CONTAINS INFORMATION ON RELEASE 18.2 OF THE FORMS SYSTEM. THIS RELEASE INVOLVES ENHANCEMENTS TO FDL, FAP AND RUN-TIME REQUIRED BY FED (THE FORMS EDITOR).

NEW FUNCTIONALITY

FDL COMPILER

THE FDL SOURCE LINE LIMIT HAS BEEN EXTENDED TO 90 CHARACTERS (FROM 72). THIS IMPLIES THAT ANY TEXT THAT USED TO FOLLOW COLUMN 72 MUST NOW BE COMMENTED, IE THE TEXT MUST BE PRECEEDED BY '/*'.

FDL NOW SUPPORTS THE CORRECT STANDARD SUFFICES, .FORM FOR SOURCE, .FBIN FOR BINARY AND .LIST FOR LISTING.

THE COMPILER HAS BEEN CONVERTED FROM R-MODE INTO A CALLABLE V-MODE PROGRAM. THIS HAS NO EFFECT ON THE USER.

FAP

FAP NOW HANDLES THE STANDARD BINARY SUFFIX OF .FBIN (SEE THE FDL COMPILER).

TCB HANDLING HAS BEEN EXTENDED TO ALLOW (IF REQUIRED) MACHINE UNIQUE TCB LISTS, INSTEAD OF THE STANDARD SYSTEM UNIQUE TCB. THIS

CAN BE ACHIEVED BY COPYING TCB.BN FROM FORMS* AND PLACING IT IN MACHINE UNIQUE TCB* UFDS. THE ORIGINAL UNDER FORMS* WOULD THEN NO LONGER BE NEEDED.

THE MAXIMUM USER NUMBER ALLOWED IN THE TCB LIST HAS BEEN EXTENDED FROM 64 TO 128.

AS WITH THE FDL COMPILER, FAP IS NOW V-MODE AND CALLABLE.

RUN TIME

THE PT45 DRIVER HAS BEEN EXTENDED TO HANDLE THE EXTRA FUNCTIONS REQUIRED BY FED. THIS HAS NO VISIBLE EFFECT TO THE USER.

PROBLEMS FIXED

FDL COMPILER

REPEAT BLOCKS NOW WORK WITH $INSERT FILES.

THE COMPILER PROCESSES SINGLE QUOTES IN LITERALS IN ACCORDANCE WITH THE STANDARD USED IN OTHER COMPILERS, THAT IS, A PAIR OF SINGLE QUOTES IN THE STRING REPRESENTS ONE QUOTE ON THE SCREEN.

THE BUG THAT CAUSED THE FIRST TWO CHARACTERS OF THE LISTING PAGE HEADER TO DISAPPEAR HAS BEEN SOLVED.

FAP

THE BUGS THAT CAUSED LIBRARY CORRUPTION AT ENTRIES 500 AND 512 HAVE BEEN FOUND AND SOLVED.

RUN TIME

FORMS NOW RESTORES THE ORIGINAL TERMINAL DUPLEX SETTINGS, INSTEAD OF DISABLING X-ON/X-OFF.

THE BUG IN THE PT45 DRIVER THAT DISALLOWED ENABLED-NODISPLAY FIELDS HAS BEEN FIXED.

THE DRIVER NOW ALLOWS THE TERMINAL ENOUGH TIME TO PERFORM CERTAIN ACTIONS THAT USED TO LEAVE NULLS ON THE TOP LINES OF THE SCREENS, SUCH AS ABSOLUTE HOME, CLEAR SCREEN.

*
590 (FTN)

SUBJECT: FTN
RELEASE: 18.2
DATE: 1981 FEBRUARY 11

1 NEW FUNCTIONALITY

NONE

2 PROBLEMS FIXED

2.1 DBG INTERFACE -- SOURCE FILE NAME

THE COMPILER REPORTS THE WRONG SOURCE FILE NAME TO DBG.

2.2 POLERS 32037

THE COMPILER USED TO LOSE TRACK OF CERTAIN ARRAY INDEX TEMPORIES.

3 OUTSTANDING PROBLEMS

3.1 POLERS 21197, 82611, AND 824502

THE COMPILER-GENERATED CROSS-REFERENCE LIST OMITS ALL VARIABLES WITH $ AS THE SECOND CHARACTER.

3.2 POLERS 30130

$INS IN LIEU OF $INSERT NEITHER GENERATES A COMPILE ERROR, NOR DOES IT INSERT A FILE INTO A PROGRAM.

3.3 POLERS 81994

THE STATEMENT " UX=U(1,1)+(M-1,1)" COMPILED WITHOUT ERRORS PRODUCING INCORRECT PROGRAM RESULTS.

3.4 POLERS 36980

A PROGRAM'S OBJECT OUTPUT HAS MULTIPLE INDIRECT ERRORS WHICH ARE DETECTED BY SEG.

3.5 POLERS 33631

BAD CODE IS PRODUCED IN A CASE OF FLOATING POINT COMPARISON (64R MODE).

3.6 COMPILE TIME FAULT

A SIMPLE PROGRAM EXISTS WHICH CAUSES THE COMPILER TO HALT IN THE MIDDLE OF ITS PROCESSING.

3.7 COMPILE TIME FAULT

A SHORT PROGRAM EXISTS WHICH CAUSES THE COMPILER TO LOOP ENDLESSLY.

3.8 POLERS 12484

THE "PARAMETER IS BETTER" WARNING MESSAGE IS OCCASIONALLY NOT ACCURATE.

4 ENVIRONMENT

REQUIRES PRIMOS 18.0.

5 INSTALLATION AND BUILD PROCEDURES

STANDARD

*
591 (MIDAS)

DATE: JUNE 11, 1981
SUBJECT: REV 18.2 MIDAS

1 NEW FUNCTIONALITY

MIDAS IS PARTIALLY COMPLIANT WITH THE NEW PRIME SOFTWARE STANDARDS. THE MAJOR CHANGE IS TO THE WAY IT IS ORGANIZED ON THE MASTER DISK, BUT USER PROGRAMS WILL RUN AS BEFORE.

O THE UFD MIDAS HAS BEEN REPLACED BY THE TWO UFD'S MIDAS AND MIDASSRC. MIDAS NOW CONTAINS ONLY INSERT FILES AND THE VARIOUS RUN FILES IN THE SUB-UFD'S CMDNC0, LIB, SYSCOM, AND SYSTEM. MIDASSRC CONTAINS ALL THE SOURCE MODULES IN THE SUB-UFD SOURCE.

O INSTALL AND SHARE COMMAND FILES REMAIN MIDAS>C_INSTALLMIDAS AND MIDAS>SYSTEM>C_SHAREMIDAS, BUT BUILD FILES HAVE BEEN REWRITTEN IN CPL AND MOVED TO MIDASSRC. MIDAS.BUILD.CPL IS THE MASTER FILE TO BUILD ALL OF MIDAS AND CALLS AN INDIVIDUAL CPL FILE PER SUB-PRODUCT SUCH AS CREATK.BUILD.CPL, VKDALB.BUILD.CPL, ETC.

O SOURCES IN MIDASSRC>SOURCE OBEY THE SUFFIX STANDARD. ALL FORTRAN SOURCE FILES HAVE THE SUFFIX .FTN, ALL FORTRAN INSERT FILES HAVE THE SUFFIX OF .INS.FTN, ETC.

O INSERT FILES IN MIDAS AND MIDAS>SYSCOM EXIST WITH AND WITHOUT THE PROPER SUFFIX FOR COMPATIBILITY. E.G. PARM.K AND PARM.K.INS.FTN. USERS SHOULD USE THE NEW SUFFIX'ED FORM IN FUTURE APPLICATIONS.

O THE MIDAS UTILITES CREATK, IMIDAS, KBUILD, KIDDEL, AND MPACK DO NOT ACCEPT COMMAND LINE ARGUMENTS OR OPTIONS. THEY HAVE BEEN MODIFIED TO REPORT 'NO COMMAND LINE ARGUMENTS POSSIBLE (XXXXXX)' AND ABORT WHEN A USER TRIES TO DO SO. MCLUP HAS BEEN MODIFIED TO DETECT A USER ATTEMPTING TO USE ITS '-USER ##' OPTION MORE THAN ONCE PER INVOCATION.

```
   O  SEVERAL MODULES IN MIDASSRC>SOURCE HAVE BEEN BROKEN UP AS  FOLLOWS
      (A SUFFIX OF .FTN IS IMPLIED IN ALL CASES):

      CNCRTN => KX$CCE, KX$CKV, KX$CPV, KX$GIE, KX$REP, KX$SAV

      KIDAFL => ADD1$,  CLOSM$, DELET$,  FIND$, FIND$$, GDATA$,  KX$LDR,
                KX$UDR,  LOCK$,  MSGCTL,  NEXT$, NEXT$$, NTFYM$, OPENM$,
                PARAMS, UMODE$, UPDAT$

      OFF2RT => KX$DUP, KX$ELP, KX$GNE, KX$GPE, KX$NX1, KX$NX2,  KX$NX4,
                KX$SBK, KX$ULV

      OFF3RT => KX$DCD, KX$LVL, KX$NBK, KX$RDR

      OFF4RT => KX$BNS, KX$GIB, KX$LNT, KX$PCP, KX$UPT
      OFFRTN => KX$ADD, KX$ECD, KX$ESH, KX$SIB

      ON2RTN => KX$DAD

      ON3RTN => KX$EDA

      ON4RTN => KX$IDE, KX$IIE

      ON5RTN => KX$ASP, KX$DLT, KX$IDC, KX$MDN, KX$MIP, KX$REC,  KX$WBK,
                KX$WFG

      ONRTN  => KX$CLS, KX$OIT, KX$RIT, KX$SUP, KX$WIT

      RE4SUB => KX$MYB, KX$NVR, KX$NWP, KX$PRR, KX$RAD, KX$RPR,  KX$VLD,
                KX$WPR, SYSINI
```

2 BUG FIXES

TAR 20047
KX$RCK GAVE A MISLEADING ERROR MESSAGE. THIS ROUTINE HAS BEEN REMOVED - SEE SECTION THREE.

TAR 20050
ADD1$ DID NOT ALWAYS TAKE THE ALTERNATE RETURN ON ERROR OR REPORT THE ERROR CODE.

TAR 21656
GDATA$ WOULD LEAVE MIDAS SEGMENT DIRECTORY SUBFILES OPEN, DIDN'T WORK WITH VARIABLE LENGTH DATA RECORDS, DIDN'T WORK IF THE USER SUPPLIED A '0' AS THE BUFFER SIZE TO INDICATE THE DEFAULT RECORD SIZE, AND SOMETIMES READ TOO MUCH CAUSING OTHER VARIABLES OR CODE TO BE MODIFIED.

TAR 23826, TAR 24229
COBOL PROGRAMS USING MIDAS RETURNED MIDAS ERROR 33 BECAUSE FIND$$ AND NEXT$$ WEREN'T ALWAYS CLEARING THE 'RECORD LOCKED' BIT IN THE 14-WORD USER ARRAY. THIS CAUSED THE COBOL RUNTIME LIBRARY VCOBLB TO SOMETIMES

ATTEMPT TO UNLOCK ALREADY UNLOCKED RECORDS WITH THE ABOVE RESULTING MESSAGE.

PSF 27016
KX$SUP, THE ROUTINE TO RETURN THE 14-WORD MIDAS ARRAY AND THE ERROR CODE, WAS BEING CALLED AS A FUNCTION WHEN IT WAS CODED AS SUBROUTINE. BECAUSE OF THE WAY IT COMPILED, THIS JUST HAPPENED TO WORK! THIS PSF WAS FIXED IN TANDEM WITH TAR 20050.

PSF 27259
SEE TAR 21656.

PSF 27967
ATTEMPTING TO 'FIND' A SECONDARY KEY THAT POINTS TO A DATA RECORD THAT HAS BEEN DELETED RETURNED THE NEXT KEY IN THE INDEX INSTEAD OF RETURNING A MIDAS ERROR 7 (KEY NOT FOUND) THE FIRST TIME THIS IS ATTEMPTED AFTER THE RECORD POINTED TO WAS DELETED.

PSF 29435
BUGS IN PRIBLD AND SECBLD COULD CAUSE FILES BUILT WITH KBUILD OR RUN THROUGH MPACK TO GENERATE 27, 32, OR 44 MIDAS ERRORS WHEN SUBSEQUENTLY ACCESSED, ESPECIALLY AFTER ADDITIONAL ENTRIES HAD BEEN ADDED THROUGH ADD1$. SOMETIMES IT APPEARED THAT NEW RECORDS WERE OVERWRITING OLD RECORDS. OTHER TIMES KBUILD WOULD BLOW UP WITH AN ACCESS VIOLATION OR A SORT LIBRARY ERROR. PATCHES WERE MADE TO KBUILD, PRIBLD, AND SECBLD.

PSF 29968
THE S[IZE] COMMAND IN CREATK WOULD OMIT TOTALS WHEN TOTAL WAS REQUESTED. ALSO, SOME NUMBERS WERE JAMMED AGAINST TEXT.

PSF 30131
SEE TAR 21656.

PSF 31247
ATTEMPTING TO CREATE A DIRECT ACCESS FILE WHICH WILL ALLOCATE MORE THAN ONE DATA SEGMENT SUBFILE CAUSED AN EOF ERROR. OTHER UTILITIES MAY BE SEEN TO HAVE THE SAME PROBLEMS. KBUILD AND PRIBLD/BILD$R HAD PROBLEMS COMPUTING THE RECORD ADDRESS WHEN BUILDING DIRECT ACCESS FILES.

PSF 32294
MIDAS FILES THAT HAD BEEN BUILT WITH KBUILD, PRIBLD, OR HAD BEEN RUN THROUGH MPACK WOULD GENERATE 27, 32, OR 44 MIDAS ERRORS WHEN ACCESSED THROUGH FIND$, NEXT$, ADD1$, ETC. SEE PSF 29435 ALSO.

PSF 32680
SEE PSF 32294

PSF 32682
WHEN KBUILD'ING A FILE THAT HAS DUPLICATE PRIMARY KEY ENTRIES, THE DUPLICATE PRIMARY KEYS WERE NOT ADDED (CORRECT), BUT THE SECONDARY KEYS WERE ADDED (INCORRECT).

PSF 33235
PREVIOUSLY WHEN AN ATTEMPT TO READ AN INDEX BLOCK IN KX$RDR FAILED ON

AN EOF ERROR BECAUSE PART OF THE INDEX BLOCK WAS BEYOND THE END OF FILE, KX$RDR HALVED THE SIZE OF THE INDEX BLOCK IT WAS TRYING TO READ AND TRIED AGAIN. IF THE BLOCK RESIDED ENTIRELY BEYOND THE END OF FILE, KX$RDR WOULD LOOP FOREVER. A CHECK HAS BEEN ADDED THAT IF THE SIZE OF THE BLOCK TRYING TO BE READ REACHES ZERO, A MIDAS 21 ERROR WILL BE GENERATED.

PSF 33263
SEE PSF 33294

PSF 34244
ADDITIONS AND DELETIONS TO SECONDARY KEYS WITH LONG STRINGS OF DUPLICATE KEYS SEEMINGLY CAUSED MIDAS INDEXES TO BE CORRUPTED. MIDAS THEN MAY HAVE PRODUCED ERROR 20, 86, OR 87. WHAT WAS ACTUALLY HAPPENING WAS WHEN BOTH A DELETE AND AN INDEX BLOCK SPLIT OCCURED ON A CALL TO ADD1$, THE DELETE WOULD DESTROY THE ACCESS PATH STACK CAUSING THE INDEX BLOCK SPLIT ROUTINE TO THINK IT WAS SPLITTING THE ROOT WHEN IT WAS SPLITTING A LAST LEVEL INDEX BLOCK.

PSF 34247
MIDAS FILES THAT HAD BEEN MPACK'ED WOULD GENERATE MIDAS ERROR 26'S WHEN A USER ATTEMPTED TO ADD NEW ENTRIES AFTERWARDS. THIS USUALLY ONLY HAPPENED ON THE PRIMARY INDEX, THOUGH THERE MAY HAVE BEEN OTHER SYMPTOMS.

PSF 34250
PROGRAMS THAT RAN OK WITH THE V-MODE MIDAS LIBRARIES (VKDALB, NVKDALB) DIDN'T WORK UNDER THE R-MODE MIDAS LIBRARY (KIDALB). AN INCORRECT STORE CAUSED THE LIBRARY TO BE RE-INITIALIZED WITH EACH CALL.

PSF 34251 (PARTIAL)
THE PARAMETER STSIZ IN KPARAM HAS BEEN CHANGED FROM 20 TO 70. THIS INCREASES THE NUMBER OF SUBFILES IN A MIDAS FILE SEGMENT DIRECTORY THAT

MIDAS WILL KEEP OPEN FROM CALL TO CALL FROM 20 TO 70. SOME USER APPLICATIONS THAT DEAL WITH SEVERAL MIDAS FILES IN THE SAME PROGRAM MAY EXPERIENCE A PERFORMANCE IMPROVEMENT. SOME COBOL PROGRAMS THAT ARE RUN BY MORE THAN ONE USER TO ACCESS THE SAME MIDAS FILE MAY SEE SOME OF THEIR CONCURRENT MIDAS USAGE PROBLEMS ALLEVIATED BECAUSE MORE MIDAS CALLS PER USER TIME-SLICE MAY BE COMPLETED. THE PROPER SOLUTION IS THE USE OF A MULTIPLE CALL OR 'GLOBAL' MIDAS LOCK. SEE PSF 35428 AND SECTIONS THREE AND FOUR.

PSF 34671
SEE PSF 27967.

PSF 34720
WHEN A USER CALLS NEXT$ WITH FLAGS = FL$USE + FL$RET, THEN CALLS DELET$ WITH FLAGS = FL$USE, AND THEN NEXT$ AGAIN WITH FLAGS = FL$USE + FL$RET, A MIDAS ERROR 13 IS RETURNED ON THE LAST CALL TO NEXT$. THIS IS NOT A BUG. BECAUSE THE 14-WORD MIDAS ARRAY THAT IS PASSED TO THE SECOND CALL TO NEXT$ POINTS TO A DELETED ENTRY, THE USER RECEIVES A LEGITIMATE CONCURRENCY ERROR (ERROR 13). SPECIFYING FLAGS = FL$USE + FL$RET TO

DELET$ WILL CAUSE DELET$ TO RETURN AN ARRAY CORRESPONDING TO THE ENTRY PRIOR TO THE ONE JUST DELETED SO THE FOLLOWINGG CALL TO NEXT$ EXECUTES CORRECTLY. THE ONE EXCEPTION TO THIS RULE IS WHEN DELETING THE FIRST ENTRY IN AN INDEX, DO NOT SPECIFY FL$RET TO DELET$, BUT SPECIFY FL$FST TO THE SECOND NEXT$ CALL TO AVOID ERRORS.

PSF 35428
THE REV 17.4 AND 17.6 COBOL RUNTIME LIBRARIES (VCOBLB AND NVCOBLB) EXHIBITED SEVERE MULTI-USER PROBLEMS. EXAMPLE: IF USER A HAD A RECORD LOCKED AND USER B ATTEMPTED TO READ IT, USER B WOULD GET A 'RECORD LOCKED' ERROR (91) WHICH IS CORRECT, BUT HE WOULD HAVE ALSO UNLOCK THE RECORD FOR USER A, WHICH IS INCORRECT. THIS WAS FIXED AT REVS 17.8/18.0. SEE SECTION FOUR FOR OTHER OUTSTANDING COBOL/MIDAS PROBLEMS.

PSF 35429
COBOL PROGRAMS WITH MIDAS FILES OPEN FOR INPUT CAN READ RECORDS LOCKED BY OTHER USERS WITHOUT RECEIVING ANY ERROR OR WARNING. COBOL PROGRAMS THAT OPEN MIDAS FILES FOR I-O CANNOT READ RECORDS LOCKED BY OTHER USERS AND WILL RECEIVE AT COBOL STATUS CODE OF 91. THERE ARE NO PLANS TO CHANGE THIS FUNCTIONALITY. WHAT HAS BEEN FIXED IS THE EASING OF GETTING AROUND THE LOCKED RECORD BY USER A IN THE SECOND CASE. PREVIOUSLY, IF A COBOL PROGRAM WAS LOOPING ON DOING READ NEXT RECORD'S AND ENCOUNTERED A RECORD LOCKED BY ANOTHER USER (STATUS CODE 91), THE USER DOING THE READ NEXT HAD TO DO A START OPERATION TO GET PAST THE LOCKED RECORD. WITH REV 18.2 MIDAS, ANOTHER READ NEXT WILL RETURN THE RECORD AFTER THE LOCKED RECORD (ASSUMING THE USER WHO HAS LOCKED THE RECORD HASN'T MOVED). THIS CHANGE WAS AFFECTED BY CHANGING THE RETURN ARRAY LOGIC IN MIDAS. NORMALLY WHEN A USER EXECUTES A MIDAS CALL WITH FLAGS = FL$RET SET AND THE CALL RESULTS IN AN ERROR, MIDAS WILL RETURN ONLY THE FIRST WORD OF THE 14-WORD USER ARRAY WHICH CONTAINS THE ERROR CODE AND NOT THE REST. NOW MIDAS WILL RETURN THE ARRAY ON A MIDAS ERROR 10 (RECORD LOCKED BY ANOTHER USER ENCOUNTERED) IF THE USER REQUESTS IT, AS WELL WHEN THERE IS NO ERROR.

PSF 35742
FIXED WITH R-MODE INTERLUDE INTRODUCED AT REV 17.6 PLUS BUG FIXED IN PSF 34250.

PSF 35748
A CALL TO DELET$ ON A DIRECT ACCESS MIDAS FILE (INDEX = -1) WOULD HANG THE USER UNTIL FORCED LOGOUT.

PSF 36467
THE FLAG FL$PRE THAT WAS MISSING FROM PARM.K.PL1 HAS BEEN ADDED.

PSF 36841
ALL REFERENCES BY MIDAS TO THE PARAMETER FILENO IN THE CALLING SEQUENCE OF MIDAS ROUTINES SUCH AS ADD1$, FIND$, NEXT$, ETC. HAVE BEEN DELETED OUT OF KIDALB. THE USER MAY NOW SET IT TO WHATEVER IS DESIRED (0 IS SUGGESTED). ADDITIONAL REFERENCES WERE DELETED OUT OF ISMCOM, LONGPL, AND OUT OF KX$RDR TOGETHER WITH REFERENCES TO THE VARIABLE PRIOR AND OBSOLETE CODE RELATING TO THE NEVER RELEASED 'RETAINED INDEX BLOCK' FEATURE.

PSF 37793
SEE PSF 35428, PSF 35429.

NO TAR OR PSF
AN MPACK 'ALL' OR MPACK OF INDEX 0, BUT NOT AN MPACK 'DATA' WOULD MESS UP THE DATA GROWTH POINTER CAUSING NEW RECORDS ADDED AFTERWARDS TO BE PLACED ON TOP OF ALREADY EXISTING RECORDS.

NO TAR OR PSF
IF A USER DID (1) FIND$ ON A SECONDARY KEY WITH FLAGS = FL$RET, (2) DELETED THAT SECONDARY KEY, (3) ADD1$ WITH FLAGS = FL$USE ON THE 14-WORD ARRAY RETURNED IN STEP (1), THE ENTRY WOULD BE PLACED IN THE INDEX OUT OF ORDER. THE CONCURRENCY LOGIC WAS CHANGED TO NOT CHECK FOR THE EXISTANCE OF THE KEY, BUT ONLY FOR THE EXISTANCE OF THE DATA RECORD. USERS MAY EXPERIENCE A PERFORMANCE IMPROVEMENT.

+

NO TAR OR PSF
IF A USER DID (1) LOCK$ USING A SECONDARY KEY WITH FLAGS = FL$RET, (2) DELETE THAT SECONDARY KEY, (3) UPDAT$ WITH FLAGS = FL$RET ON THE 14-WORD ARRAY RETURNED IN STEP (1), A MIDAS ERROR 13 OR 33 GETS RETURNED ON STEP (3) AND THE UPDATE DOES NOT TAKE PLACE NOR THE RECORD GET UNLOCKED. THE CONCURRENCY LOGIC WAS CHANGED TO NOT CHECK FOR THE EXISTANCE OF THE KEY, BUT ONLY FOR THE EXISTANCE OF THE DATA RECORD. USERS MAY EXPERIENCE A PERFORMANCE IMPROVEMENT.

+

---

3 NON-VISIBLE INTERNAL FIXES AND ENHANCEMENTS.

O TWO NEW ENTRY POINTS HAVE BEEN ADDED TO MIDAS: KX$GLK AND KX$GUK.

KX$GLK - GLOBAL LOCK
THIS SETS A 'GLOBAL LOCK' FLAG IN MIDAS. WHEN THE GLOBAL LOCK FLAG IS SET, ONCE A MIDAS USER GETS THE MIDAS LOCK DURING THE COURSE OF A MIDAS CALL, THE MIDAS LOCK WILL NOT BE RELEASED EVEN AFTER RETURNING TO THE USER PROGRAM. THERE ARE NO ARGUMENTS IN THE CALL TO KX$GLK.

KX$GUK - GLOBAL UNLOCK
THIS RESETS THE 'GLOBAL LOCK' FLAG SET BY KX$GLK. IF THE USER CURRENTLY HOLDS THE MIDAS LOCK, IT IS RELEASED AT THIS TIME AND THE NEXT USER WAITING ON THE MIDAS SEMAPHORE IS NOTIFIED OFF IT TO NOW RUN. THERE ARE NO ARGUMENTS IN THE CALL TO KX$GUK.

THE PURPOSE OF THESE ROUTINES IS TO MAKE A SERIES OF MIDAS CALLS NON-INTERRUPTABLE, PARTICULARLY WHEN TWO OR MORE USERS OF MIDAS ARE ACTING UPON THE SAME MIDAS FILE. THESE CALLS MAY BE INCORPORATED IN OTHER PRIME SOFTWARE AT A FUTURE DATE. THESE ROUTINES ARE INTENDED ARE INTENDED FOR USE ONLY BY PRIME SOFTWARE. CUSTOMER USE, OTHER THAN THROUGH PRIME SUPPLIED SOFTWARE, IS NOT

SUPPORTED.

O KX$RCK, KX$GNI, KX$IDX HAVE BEEN REMOVED. PREVIOUSLY THEY CHECKED TO SEE WHETHER OVERFLOW CHAIN POINTERS FROM REV 15 OR EARLIER MIDAS FILES EXISTED. IF THERE WEREN'T ANY, BUT THE REV STAMP WAS 15 OR LESS, KX$RCK UPDATED IT. SINCE MOST USERS ARE NOW ON REV 16 OR LATER, THIS METHOD WAS COSTING A NEEDLESS EXTRA CALL TO PRWF$$ WITH EVERY MIDAS CALL. INSTEAD, A SIMPLE CHECK FOR REV 15 OR EARLIER HAS BEEN SUBSTITUTED IN KX$VLD WHICH PRINTS THE MESSAGE 'STOP! REMAKE THIS FILE' IF THE REV STAMP ON THE MIDAS FILE IS 15 OR EARLIER, REGARDLESS OF WHETHER OVERFLOW CHAINS EXIST OR NOT. USERS MAY SEE A PERFORMANCE IMPROVEMENT.

O CALLS TO THE SUBROUTINES I$AD07 AND O$AD07 HAVE BEEN REPLACED WITH CALLS TO RDLIN$ AND WTLIN$ FOR MORE EFFICIENT COMPRESSED ASCII FILE I/O. THIS AFFECTS ONLY THE OFFLINE MIDAS ROUTINES AND UTILITIES. USERS MAY SEE A PERFORMANCE IMPROVEMENT.

O THE MIDAS OFFLINE ROUTINES AND UTILITIES NOW ATTEMPT TO READ INDEX BLOCKS IN ONE CALL TO PRWF$$. USERS MAY SEE A PERFORMANCE IMPROVEMENT.

O CODE DEALING WITH THE OBSOLETE ARGUMENT IN MIDAS RUNTIME LIBRARY CALLS DOCUMENTED AS 'FILE-NO' HAS BEEN DELETED. USERS MAY NOW SET THIS ANY VALUE THEY WISH; ZERO IS SUGGESTED.

O CODE DEALING WITH RETAINED INDEX BLOCKS, A FEATURE NEVER OFFICIALLY DOCUMENTED OR RELEASED TO USERS, HAS BEEN REMOVED, PRINCIPALLY FROM KX$RDR.

O THE COMMON AREA /XFILES/ HAS BEEN REMOVED FROM LONGPL AS IT WAS UNUSED AND THE LINKAGE SPACE WAS NEEDED.

O SEVERAL SMALL PIECES OF UNUSED CODE WERE REMOVED FROM THE MIDAS RUNTIME LIBRARY ROUTINES TO GET RID OF THE 'NO PATH TO STATEMENT' COMPILATION ERROR MESSAGES.

4 OUTSTANDING BUGS

O BECAUSE KBUILD USES FILE UNIT 1 FOR A TEMPORARY FILE WHEN BUILDING SECONDARY INDEX 1, FILE UNIT 2 FOR INDEX 2, ETC., USERS RUNNING KBUILD FROM A COMMAND FILE MAY EXPERIENCE PROBLEMS WITH KBUILD INTERFERING WITH THEIR RUNNING OF THEIR COMMAND FILE OR VICE-VERSA. A PATCH EXISTS AND THIS BUG WILL BE FIXED IN REV 18.3/19.0.

O WHEN DOING A KBUILD THAT ADDS ADDITIONAL PRIMARY KEY/DATA RECORDS TO A MIDAS FILE THAT ALREADY CONTAINS DATA, THE FIRST RECORD ADDED WILL SEEMINGLY BE GARBAGED, BUT MPACKING THE FILE WILL FIX IT. THIS BUG WILL BE FIXED AT REV 18.3/19.1.

O THE ANSI COBOL STANDARD DOES NOT ALLOW COBOL PROGRAMS TO CHANGE THE PRIMARY KEY ON A REWRITE. UNFORTUNATELY, ATTEMPTS TO DO SO ARE NOT DETECTED, RESULTING IN THE PRIMARY KEY ACCORDING TO THE DATA RECORD NOT AGREEING WITH THE KEY IN THE PRIMARY INDEX BECAUSE ONLY THE DATA RECORD AND NOT THE PRIMARY INDEX IS UPDATED. THIS MANIFESTS ITSELF WITH SEEMINGLY OUT OF ORDER RECORDS WHEN THE

PRIMARY INDEX IS TRAVERSED SEQUENTIALLY OR CONCURRENCY ERRORS ON DELETES. THIS PROBLEM WILL BE FIXED IN THE COBOL RUNTIME LIBRARY AT 18.3/19.1.

0 IF A COBOL PROGRAM ATTEMPTS TO CHANGE THE VALUE OF AN ALTERNATE KEY (MIDAS SECONDARY KEY) ON A REWRITE TO A VALUE THAT IS ALREADY PRESENT IN THE INDEX AND THAT INDEX DOES NOT ALLOW DUPLICATE KEYS, THE USER WILL CORRECTLY RECEIVE AN ERROR, BUT THE COBOL RUNTIME LIBRARY WILL NOT ROLL BACK THE MIDAS FILE TO ITS PRIOR STATE. THE DATA RECORD WILL REFLECT THE NEW KEY VALUES, BUT SOME INDEXES MAY HAVE THE NEW KEY VALUES, SOME INDEXES MAY HAVE THE OLD KEY VALUES, AND THE ONE INDEX WITH WHICH THE ERROR OCCURED WILL HAVE NO KEY VALUE POINTING TO THE DATA RECORD. THIS PROBLEM WILL BE FIXED IN THE COBOL RUNTIME LIBRARY AT 18.3/19.1.

0 BECAUSE THE COBOL VERBS WRITE AND REWRITE POTENTIALLY TRANSLATE INTO MORE THAN ONE MIDAS CALL, IT IS POSSIBLE FOR USERS TO INTERFERE WITH EACH OTHER WHEN ALL ARE SIMULTANEOUSLY USING THE SAME MIDAS FILE. THE KX$GLK AND KX$GUK ROUTINES INTRODUCED AT 18.2 (DESCRIBED IN SECTION THREE) REPRESENT THE MIDAS MODIFICATIONS NECESSARY TO TAKE ADDRESS THIS PROBLEM. THE NECESSARY COBOL RUNTIME LIBRARY MODIFICATIONS WILL BE RELEASED AT REV 18.3/19.1.

*
592 (PRINET)

USING FAM II

THIS SHORT DOCUMENT DESCRIBES THE INITIALIZATION AND OPERATION OF FAM II, WHICH WILL BE AVAILABLE IN PRIMOS REV 18.2.

1 OVERVIEW

FAM II IS FASTER AND MORE RELIABLE THAN FAM I AND HENCE IT IS TO THE USERS BENEFIT TO CONVERT TO IT AS SOON AS POSSIBLE. REV 18.2 AND 19 WILL SUPPORT FAM COMPATIBILITY MODE WHICH WILL ALLOW SYSTEMS RUNNING FAM II TO COMMUNICATE WITH SYSTEMS RUNNING FAM I. THIS ALLOWS NETWORK A GRADUAL PHASE OVER FROM FAM I TO FAM II. FAM COMPATIBILITY MODE WILL END AT REV 20.

2 THE CONFIG DIRECTIVE NSLUSR

EACH USER ACCESSING FILES ON YOUR SYSTEM FROM A REMOTE SYSTEM WILL REQUIRE A SLAVE PROCESS FOR THE DURATION OF THE ACCESS. THESE SLAVE PROCESSES COME OUT OF THE PRIMOS 128 PROCESS POOL. THE NUMBER OF SLAVES CONFIGURED FOR A SYSTEM IS SET BY THE NSLUSR DIRECTIVE IN CONFIG (ANALOGOUS TO THE NTUSR, NPUSR, NRUSR DIRECTIVE).

THE NSLUSR DIRECTIVE TAKES AN OCTAL PARAMETER EQUAL TO THE NUMBER OF SIMULTANEOUS REMOTE FILE ACCESSES YOUR SYSTEM WISHES TO SUPPORT. IF THIS POOL IS EXHAUSTED WHEN A REMOTE USER MAKES AN ATTACH REQUEST THE E$NSLA (NO NPX SLAVES AVAILABLE) ERROR CODE IS RETURNED.

NTUSR+NPUSR+NRUSR+NSLUSR MUST BE LESS THAN 129.

3 NETCFG CHANGES

NETCFG, THE NETWORK CONFIGURATION PROGRAM, ASKS THREE NEW QUESTIONS IN ORDER TO SUPPORT FAM II.

1. "ENABLE FAM II?" - THIS IS A YES/NO QUESTION. IF YOU ANSWER "NO" THEN NETCFG WILL ASK ABOUT FAM I AND REMOTE DISK PERMIT.

   IF YOU ANSWER "YES" THEN NETCFG ASKS THE OTHER 2 NEW QUESTIONS:

2. "IS THIS NODE IN YOUR NAMING SPHERE?" - THIS, TOO, IS A YES/NO QUESTION. A NAMING SPHERE IS A SET OF PRIMENET NODES WHERE LOGIN NAMES (UFD) ARE HANDED OUT FROM A SINGLE ADMINISTRATION AND THIS ADMINISTRATION IS RESPONSIBLE FOR ELIMINATING CONFLICTING OR DUPLICATE LOGIN NAMES.

   FOR EXAMPLE, YOU MIGHT HAVE TWO SUBNETWORKS, EACH COMPRISING A DIFFERENT NAMING SPHERE. THERE MAY BE A LOGIN NAME "JONES" IN EACH SUBNETWORK. THERE MAY BE SEVERAL LOGIN NAMES OF "JONES" IN A NAMING SPHERE, BUT THAT NAMING ADMINISTRATION HAS DELIBERATELY DONE SO, AND IS AWARE OF THE EFFECTS.

   NAMING SPHERES WILL CARRY ADDITIONAL MEANING UNDER ACCESS CONTROL LISTS AT REV 19.

3. "RINGO-RINGO PASSWORD?" - THIS IS A 32 CHARACTER PASSWORD USED TO INSURE THE INTEGRITY OF THE NETWORK. THIS BECOMES MORE IMPORTANT FOR SYSTEMS WORKING IN A POTENTIALLY HOSTILE NETWORK (E.G., A SYSTEM CONNECTED TO A PDN CAN EASILY BECOME THE TARGET OF A COMPUTERIZED SECURITY ATTACK, OR A COLLEGE RING SHARED BY ADMINISTRATIVE NODES AND NODES WHERE STUDENTS MAKE AND BOOT THEIR OWN SYSTEMS.) THE PROBABILITY OF CRACKING THE PASSWORDS IN 10 YEARS OF CONSTANT TRYING IS 8.6X10-54. PASSWORDS MAY BE EASILY CHANGED WITH THE -PASSWORD COMMAND LINE OPTION TO NETCFG. NETCFG WILL ASK WHAT NODE AND WHAT NEW PASSWORD IS DESIRED.

NOTE: PASSWORDS ARE BETWEEN PAIRS OF NODES. IN THE EXAMPLE IN

APPENDIX A. SYSB MUST HAVE SPECIFIED THAT THE RINGO-RINGO PASSWORD TO SYSA IS "SMALL-BROWN-FOX."

NETCFG NOW PROTECTS THE NETCON FILE WITH NO ACCESS RIGHTS FOR NON OWNERS.

4 THE PRIMENET* UFD

THE SLAVES START UP FROM AINIT BY USING A BATCH$ CALL. HENCE, THEY NEED A PHANTOM FILE AND A FIXED PLACE TO FIND IT. THE TOP LEVEL UFD PRIMENET* MUST EXIST ON THE SYSTEM DISK PARTITION AT SYSTEM START UP, AND MUST CONTAIN THE FILE SLAVE.COMI.

5 ADDISK

ADDISK FOR FAM II SYSTEMS IS DIFFERENT THAN FOR FAM I SYSTEMS. IT TAKES THE FORM:

```
     ADDISK -ON <NODENAME> <DISKNAME 1>...<DISKNAME 9>
```

FOR EXAMPLE FROM ENA:

```
     AD -ON ENB SOFTWR SPOOLB
```

THE EXISTENCE OF THE DISKS OR THE UP/DOWN STATUS OF THE REMOTE SYSTEM IS NOT VERIFIED BY ADDISK. THE DISKNAMES ARE MERELY ADDED TO THE DISK SEARCH LIST SEEN IN STAT DISK.

THE SEARCH RULES UNDER FAM II ARE THE SAME AS FAM I - THAT IS: ALL

LOCAL DISKS ARE SEARCHED FIRST, THEN REMOTE DISKS ARE SEARCHED. IF A REMOTE LINE DOWN ERROR OCCURS DURING THE REMOTE SEARCH, THE SEARCH CONTINUES WITH THE NEXT REMOTE DISK TILL THE UFD IS FOUND OR ALL REMOTE DISKS ARE SEARCHED.

6 STATUS COMMAND

STAT DISK DOES NOT DISPLAY THE PHYSICAL DEVICE NUMBER FOR REMOTE DISKS THAT ARE ACCESSED VIA FAM II.

STAT USERS DISPLAY ALL THE SLAVES THAT ARE WORKING, IN BEHALF OF REMOTE USERS AS USER SLAVE$.

APPENDIX A

```
OK, NETCFG
REVIEW OLD NETWORK CONFIGURATION? YES

REV    19.0 NETWORK CONFIGURATION FILE

RING NET
           NAME              ADDR          RING ID   FAM INFO   RLOG
        --------    ---------------      -------  ---------- ----
*ME*   SYSA                                    1
       SYSB                                    2     I/RDP      YES

CREATE NEW NETWORK CONFIGURATION? YES

PLEASE DESCRIBE YOUR NODE

NODE NAME? SYSA
PDN ADDRESS (CR IF NONE)?

DO YOU HAVE A RING? YES
NUMBER OF RING NODES (INCLUDING YOURSELF)? 2

YOUR RING NODE ID #? 1

PLEASE DESCRIBE THE OTHER NODES

NODE NAME? SYSB
PDN ADDRESS (CR IF NONE)?
RING NODE ID #? 2
ENABLE FAM II? YES
IS THIS NODE IN YOUR NAMING SPHERE? YES
RINGO-RINGO PASSWORD? SMALL-BROWN-FOX
ENABLE REMOTE LOGIN? YES

DO YOU HAVE FULL DUPLEX SYNCHRONOUS LINES? NO

DO YOU HAVE HALF DUPLEX SYNCHRONOUS LINES? NO

REVIEW NEW NETWORK CONFIGURATION? YES

REV    19.0 NETWORK CONFIGURATION FILE

RING NET
           NAME              ADDR          RING ID   FAM INFO   RLOG
        --------    ---------------      -------  ---------- ----
*ME*   SYSA                                    1
       SYSB                                    2  II/SAME-NS  YES
                        RINGO-RINGO PASSWORD: SMALL-BROWN-FOX
```

*
593 (RPG)

DATE: MARCH 13, 1981
TO: RPG USERS
SUBJECT: INFORMATION PERTAINING TO REV18.2 RPG

BUG FIXES

1) A BUG IN THE DIV OPERATION CODE WAS FIXED.

*
594 (X.25)

USING FAM II

THIS SHORT DOCUMENT DESCRIBES THE INITIALIZATION AND OPERATION OF FAM II, WHICH WILL BE AVAILABLE IN PRIMOS REV 18.2.

1 OVERVIEW

FAM II IS FASTER AND MORE RELIABLE THAN FAM I AND HENCE IT IS TO THE USERS BENEFIT TO CONVERT TO IT AS SOON AS POSSIBLE. REV 18.2 AND 19 WILL SUPPORT FAM COMPATIBILITY MODE WHICH WILL ALLOW SYSTEMS RUNNING FAM II TO COMMUNICATE WITH SYSTEMS RUNNING FAM I. THIS ALLOWS NETWORK A GRADUAL PHASE OVFR FROM FAM I TO FAM II. FAM COMPATIBILITY MODE WILL END AT REV 20.

2 THE CONFIG DIRECTIVE NSLUSR

EACH USER ACCESSING FILES ON YOUR SYSTEM FROM A REMOTE SYSTEM WILL REQUIRE A SLAVE PROCESS FOR THE DURATION OF THE ACCESS. THESE SLAVE PROCESSES COME OUT OF THE PRIMOS 128 PROCESS POOL. THE NUMBER OF SLAVES CONFIGURED FOR A SYSTEM IS SET BY THE NSLUSR DIRECTIVE IN CONFIG (ANALOGOUS TO THE NTUSR, NPUSR, NRUSR DIRECTIVE).

THE NSLUSR DIRECTIVE TAKES AN OCTAL PARAMETER EQUAL TO THE NUMBER OF SIMULTANEOUS REMOTE FILE ACCESSES YOUR SYSTEM WISHES TO SUPPORT. IF

THIS POOL IS EXHAUSTED WHEN A REMOTE USER MAKES AN ATTACH REQUEST THE E$NSLA (NO NPX SLAVES AVAILABLE) ERROR CODE IS RETURNED.

NTUSR+NPUSR+NRUSR+NSLUSR MUST BE LESS THAN 129.

3 NETCFG CHANGES

NETCFG, THE NETWORK CONFIGURATION PROGRAM, ASKS THREE NEW QUESTIONS IN ORDER TO SUPPORT FAM II.

1. "ENABLE FAM II?" - THIS IS A YES/NO QUESTION. IF YOU ANSWER "NO" THEN NETCFG WILL ASK ABOUT FAM I AND REMOTE DISK PERMIT.

   IF YOU ANSWER "YES" THEN NETCFG ASKS THE OTHER 2 NEW QUESTIONS:

2: "IS THIS NODE IN YOUR NAMING SPHERE?" - THIS, TOO, IS A YES/NO QUESTION. A NAMING SPHERE IS A SET OF PRIMENET NODES WHERE LOGIN NAMES (UFD) ARE HANDED OUT FROM A SINGLE ADMINISTRATION AND THIS ADMINISTRATION IS RESPONSIBLE FOR ELIMINATING CONFLICTING OR DUPLICATE LOGIN NAMES.

   FOR EXAMPLE, YOU MIGHT HAVE TWO SUBNETWORKS, EACH COMPRISING A DIFFERENT NAMING SPHERE. THERE MAY BE A LOGIN NAME "JONES" IN EACH SUBNETWORK. THERE MAY BE SEVERAL LOGIN NAMES OF "JONES" IN A NAMING SPHERE, BUT THAT NAMING ADMINISTRATION HAS DELIBERATELY DONE SO, AND IS AWARE OF THE EFFECTS.

   NAMING SPHERES WILL CARRY ADDITIONAL MEANING UNDER ACCESS CONTROL LISTS AT REV 19.

3. "RINGO-RINGO PASSWORD?" - THIS IS A 32 CHARACTER PASSWORD USED TO INSURE THE INTEGRITY OF THE NETWORK. THIS BECOMES MORE IMPORTANT FOR SYSTEMS WORKING IN A POTENTIALLY HOSTILE NETWORK (E.G., A SYSTEM CONNECTED TO A PDN CAN EASILY BECOME THE TARGET OF A COMPUTERIZED SECURITY ATTACK, OR A COLLEGE RING SHARED BY ADMINISTRATIVE NODES AND NODES WHERE STUDENTS MAKE AND BOOT THEIR OWN SYSTEMS.) THE PROBABILITY OF CRACKING THE PASSWORDS IN 10 YEARS OF CONSTANT TRYING IS 8.6X10-54. PASSWORDS MAY BE EASILY CHANGED WITH THE -PASSWORD COMMAND LINE OPTION TO NETCFG. NETCFG WILL ASK WHAT NODE AND WHAT NEW PASSWORD IS DESIRED.

NOTE: PASSWORDS ARE BETWEEN PAIRS OF NODES. IN THE EXAMPLE IN APPENDIX A. SYSB MUST HAVE SPECIFIED THAT THE RINGO-RINGO PASSWORD TO SYSA IS "SMALL-BROWN-FOX."

NETCFG NOW PROTECTS THE NETCON FILE WITH NO ACCESS RIGHTS FOR NON OWNERS.

4 THE PRIMENET* UFD

THE SLAVES START UP FROM AINIT BY USING A BATCH$ CALL. HENCE, THEY NEED A PHANTOM FILE AND A FIXED PLACE TO FIND IT. THE TOP LEVEL UFD PRIMENET* MUST EXIST ON THE SYSTEM DISK PARTITION AT SYSTEM START UP, AND MUST CONTAIN THE FILE SLAVE.COMI.

5 ADDISK

ADDISK FOR FAM II SYSTEMS IS DIFFERENT THAN FOR FAM I SYSTEMS. IT TAKES THE FORM:

```
     ADDISK -ON <NODENAME> <DISKNAME 1>...<DISKNAME 9>
```

FOR EXAMPLE FROM ENA:

```
     AD -ON ENB SOFTWR SPOOLB
```

THE EXISTENCE OF THE DISKS OR THE UP/DOWN STATUS OF THE REMOTE SYSTEM IS NOT VERIFIED BY ADDISK. THE DISKNAMES ARE MERELY ADDED TO THE DISK SEARCH LIST SEEN IN STAT DISK.

THE SEARCH RULES UNDER FAM II ARE THE SAME AS FAM I - THAT IS: ALL

LOCAL DISKS ARE SEARCHED FIRST, THEN REMOTE DISKS ARE SEARCHED. IF A REMOTE LINE DOWN ERROR OCCURS DURING THE REMOTE SEARCH, THE SEARCH CONTINUES WITH THE NEXT REMOTE DISK TILL THE UFD IS FOUND OR ALL REMOTE DISKS ARE SEARCHED.

6 STATUS COMMAND

STAT DISK DOES NOT DISPLAY THE PHYSICAL DEVICE NUMBER FOR REMOTE DISKS THAT ARE ACCESSED VIA FAM II.

STAT USERS DISPLAY ALL THE SLAVES THAT ARE WORKING, IN BEHALF OF REMOTE USERS AS USER SLAVE$.

APPENDIX A

```
OK, NETCFG
REVIEW OLD NETWORK CONFIGURATION? YES

REV    19.0 NETWORK CONFIGURATION FILE

RING NET
        NAME            ADDR           RING ID   FAM INFO  RLOG
      --------    ---------------      -------  ---------- ----
      *ME*  SYSA                                      1
            SYSB                                      2    I/RDP     YES

      CREATE NEW NETWORK CONFIGURATION? YES

      PLEASE DESCRIBE YOUR NODE

      NODE NAME? SYSA
      PDN ADDRESS (CR IF NONE)?

      DO YOU HAVE A RING? YES
      NUMBER OF RING NODES (INCLUDING YOURSELF)? 2

      YOUR RING NODE ID #? 1

      PLEASE DESCRIBE THE OTHER NODES

      NODE NAME? SYSB
      PDN ADDRESS (CR IF NONE)?
      RING NODE ID #? 2
      ENABLE FAM II? YES
      IS THIS NODE IN YOUR NAMING SPHERE? YES
      RINGO-RINGO PASSWORD? SMALL-BROWN-FOX
      ENABLE REMOTE LOGIN? YES

      DO YOU HAVE FULL DUPLEX SYNCHRONOUS LINES? NO

      DO YOU HAVE HALF DUPLEX SYNCHRONOUS LINES? NO

      REVIEW NEW NETWORK CONFIGURATION? YES

      REV    19.0 NETWORK CONFIGURATION FILE

      RING NET
              NAME            ADDR             RING ID   FAM INFO   RLOG
            --------    ---------------    -------  ---------- ----
      *ME*  SYSA                                 1
            SYSB                                 2  II/SAME-NS   YES
                              RINGO-RINGO PASSWORD: SMALL-BROWN-FOX
```

```
*
595-599                   (DBMS)
```

```
      SUBJECT:     DBMS

      RELEASE:     18.2
```

DATE: JUNE 30, 1981

I. NEW FUNCTIONALITY

DBUTL

TWO NEW COMMANDS HAVE BEEN ADDED:

VERIFY [DECIMAL #]

FOR EACH B-TREE IN THE CURRENT SET, VERIFIES THAT EVERY LEAF NODE DBK IS IN THE DATABASE. IF THE OPTIONAL DECIMAL NUMBER IS SPECIFIED, A CHECKPOINT MESSAGE WILL BE DISPLAYED AFTER THE SPECIFIED NUMBER OF OWNER DIRECTORIES (SET OCCURRENCES) HAVE BEEN PROCESSED. (THIS COMMAND SERVES AS A COMPLIMENT TO THE NEW DMLCP VERIFY OPTION).

VERIFY BUG TYPES:

1 = COULD NOT POSITION TO ROOT OF B-TREE

2 = COULD NOT POSITION TO LEFT-MOST LEAF NODE

3 = INFINITE LOOP IN LEAF NODE 'RIGHT' POINTERS

4 = COULD NOT POSITION TO THE NEXT (RIGHT) LEAF NODE

6 = DBK NOT IN DATABASE

7 = DBK MARKED AS DELETED IN DATABASE

DBK <3 NUMBERS SEPARATED BY SPACES> ‖ DBK

IF A DBK IS SPECIFIED IN ITS LOGICAL (AREAID RECID OCCNO BUCKNO) FORMAT, THEN THE INTERNAL 48 BIT REPRESENTATION OF THE DBK IS DISPLAYED (AS 3 DECIMAL NUMBERS). IF THE INTERNAL REPRESENTATION OF THE DBK IS SPECIFIED (THE 3 DECIMAL NUMBER OPTION), THEN THE DBK IS DISPLAYED IN ITS LOGICAL (UNPACKED) FORMAT. THIS COMMAND IS USED TO DETERMINE THE LOGICAL (UNPACKED) FORMAT OF A DBK GIVEN THE PACKED FORMAT DISPLAYED IN DMLCP TRACES (E.G. FTRACE).

DMLCP

THE COMMAND LINE OPTION -VERIFY HAS BEEN ADDED TO DMLCP.

THE -VERIFY OPTION ALLOWS AN APPLICATION PROGRAM TO VERIFY THE INTEGRITY AND CONSISTENCY OF RECORDS IN

PARTICULAR AREAS OF THE DATABASE AS WELL AS CHECK THE CONSISTENCY OF CALC AND SET FILES. THE APPLICATION PROGRAM SHOULD CONTAIN A SERIES OF FIND (OR FETCH) NEXT RECORD OF AREA AREA-NAME TO CHECK EACH AREA OF INTEREST. DMLCP WILL LOCATE THE RECORD USING ALL KEYS (CALC, SORT, SEARCH) DEFINED FOR THE RECORD TYPE, AND WILL VERIFY THAT THE RECORD IS CONTAINED IN THE NON-SORTED SET OCCURRENCES OF WHICH IT IS CURRENTLY A MEMBER. IN ADDITION, FOR EACH SET OCCURRENCE WHICH THE RECORD OWNS, DMLCP WILL CHECK THAT THE RECORD'S DBK IS CONTAINED IN THE SET FILE.

THE ONLY INCONSISTENCY THE DMLCP -VERIFY WILL NOT DETECT IS THE CASE IN WHICH A DBK IS CONTAINED IN A SET LIST, BUT IT IS EITHER NOT CONTAINED IN THE AREA FILE SPECIFIED BY THE DBK OR IS MARKED AS DELETED IN THE AREA FILE. THE DBUTL VERIFY COMMAND CAN BE USED TO DETECT INCONSISTENCIES OF THIS TYPE. (THE DMLCP AND DBUTL VERIFIES ARE COMPLIMENTARY AND DO NOT OVERLAP IN FUNCTIONALITY).

WHEN AN INCONSISTENCY IS DETECTED IN -VERIFY MODE, DMLCP WILL WRITE A (BINARY) DESCRIPTION OF THE ERROR TO A FILE OPENED ON UNIT 45. A FILE MUST BE OPENED ON THIS UNIT NUMBER BEFORE THE APPLICATION PROGRAM IS INVOKED.

EXAMPLE:

```
OPEN BUG.FILE 45 3

SEG #PROGRAM -VERIFY

CLOSE 45
```

THE PROGRAM DBMSLB>VFYPRT.SAVE IS USED TO DISPLAY THE CONTENTS OF BUG.FILE IN A FORMATTED FASHION. VFYPRT WILL PRINT A MENU OF OPTIONS THAT CAN BE USED TO DISPLAY OR ANALYZE THE BUG FILE. FOR EXAMPLE, VFYPRT CAN BE USED TO PRINT THE DEFINITION OF A PARTICULAR BUG NUMBER DISCOVERED VIA THE -VERIFY.

A RUN-TIME TRAP WILL NOW CHECK FOR PRE-18 (ONLY) CREATED SCHEMAS WHICH HAVE NOT BEEN CONVERTED (VIA DBUTL 'REV18' COMMAND) TO REV 18 (AND HIGHER) REQUIRED FORMAT. IF THE CONVERSION HAS NOT BEEN DONE, ERROR 1428F IS GENERATED WITH 'SCHEMA REQUIRES REV18 CONVERSION' ERROR MESSAGE.

NOTE: THERE IS CURRENTLY NO MECHANISM TO PREVENT INAPPROPRIATE CONVERSION (I.E. CONVERSION OF SCHEMAS EITHER CREATED WITH AN 18 OR HIGHER VERSION OF DBMS OR SCHEMAS WHICH HAVE ALREADY BEEN CONVERTED). THE DBA MUST EXERCISE CAUTION IN THIS AREA!

A BINARY SEARCH (RATHER THAN A LINEAR SEARCH) IS NOW USED TO SEARCH FOR A KEY IN A SET B-TREE NODE. THIS CHANGE MAY RESULT

IN A 15 PERCENT ELAPSED TIME REDUCTION FOR KEYED FIND OPERATIONS PERFORMED ON A SET THAT HAS A LARGE NUMBER OF KEYS PER NODE (E.G. A SYSTEM OWNED SET).

ITEMS IN A DML GET ITEM-LIST OPERATION MAY NOW BE CHUNKED. PREVIOUSLY ITEMS WERE ONLY CHUNKED WHEN AN ENTIRE RECORD WAS RETURNED TO THE USER WORK AREA.

ULIB

SEVERAL EXISTING ULIB ROUTINES HAVE BEEN REPLACED BY PMA VERSIONS OF THE SAME.

ASG, ASI

RUNTIME SUPPORT FOR DBMS-QUERY PRODUCT (VISTA).

II. PROBLEMS FIXED

DBACP

THE ALGORITHM FOR DETERMINING THE NUMBER OF ADDITIONAL NODES REQUIRED AS A RESULT OF THE EXPAND SET COMMAND HAS BEEN CHANGED TO REFLECT ORIGINAL ALLOCATION OWNER-TO-MEMBER RATIOS. [TAR #29132]

WHEN RESTORING A SAVED SCHEMA OR AFTER IMAGE FILE(S) FROM TAPE, DBACP NOW REPORTS AN INCORRECT TAPE REEL WITHOUT READING TO THE END OF THE TAPE. [TAR #15939]

BEFORE IMAGE RECOVERY ALGORITHM MODIFIED TO CONSISTENTLY APPLY THE CORRECT BEFORE IMAGE.

'HANG' SOMETIMES EXPERIENCED DURING MULTIPLE VERIFICATION OR RECOVERY REQUESTS HAS BEEN CORRECTED.

ROLL FORWARD RECOVERY HAS BEEN CORRECTED SO THAT AN INITIAL ABORT TRANSACTION IN THE SORTED AFTER-IMAGE FILE WILL NOT CAUSE THE READ OF THE FILE TO TERMINATE PREMATURELY (AND RESULT IN NO AFTER IMAGES BEING ROLLED FORWARD). [REV 18.0, 18.1 BUG]

THE ALGORITHM FOR GENERATING THE LIST OF INCOMPLETE TRANSACTIONS (TO THE TERMINAL) DURING ROLL FORWARD RECOVERY HAS BEEN CHANGED TO ACCOMODATE RETRIEVAL TRANSACTIONS NON-UNIQUE TR NUMBERS. [REV 18.0, 18.1 BUG]

ERROR MESSAGE REVISED FROM 'FOUND INCOMPLETE TRANSACTIONS...BUT THERE MAY BE PHANTOM INCOMPLETE TRANSACTIONS IF ANY RUN-UNITS ARE CURRENTLY ACCESSING SCHEMA' TO THE FOLLOWING MORE

DESCRIPTIVE MESSAGE: 'FOUND INCOMPLETE TRANSACTIONS...BUT IF THIS DBACP COMMAND DOES NOT LOCK THE SCHEMA, THEN SOME USERS MAY BE ACCESSING THE SCHEMA WHICH WOULD ACCOUNT FOR SOME INCOMPLETE TRANSACTIONS'.

THE SAVE AND RESTORE SCHEMA COMMANDS HAVE BEEN ENHANCED TO SUPPORT A LOGICAL END-OF-TAPE IN ADDITION TO THE PHYSICAL ONE TO OVERCOME THE PREVIOUS PROBLEMS OF LOST DATA BLOCKS AT THE END OF A REEL. THE NEW FORMAT IS COMPATIBLE WITH THE OLD FORMAT BUT THE ADDED PROTECTION IS ONLY AVAILABLE WHEN BOTH SAVE AND RESTORE ARE DONE WITH REV 18.2 OR SUBSEQUENT SOFTWARE.

DMLCP

THE STORE OF A RECORD TYPE WHICH HAS NO DATA ITEMS AND WHICH SPANS BUCKET BOUNDARIES HAS BEEN MODIFIED SO THAT SET INFORMATION IS INSERTED CORRECTLY. [TAR #31541]

A SCHEMA WHICH DECLARES PRIVACY LOCKS FOR AREAS (ONLY), NO LONGER RECEIVES A PRIVACY BREACH DURING A FIND OPERATION ON A SET. [TAR #36360]

WHEN A DUPLICATES VIOLATION OCCURS IN A STORE (1205), THE FIRST ITEM IN THE KEY IS NOW RETURNED IN ERITEM. (NOTE: THE KEY MAY CONSIST OF MULTIPLE ITEMS; ERITEM IS ASSIGNED THE VALUE OF ONLY THE FIRST KEY ITEM). [TAR #81976]

THE UNLOCKING MECHANISM HAS BEEN MODIFIED TO ELIMINATE INAPPROPRIATE LOCK WAITS (AND THUS HUNG USERS).

WHEN DUPLICATE KEYS SPAN TWO NODES, DELETION OF ALL OF THE ITEMS IN THE SECOND NODE WILL NO LONGER AFFECT DBMS' ABILITY TO FIND THE OTHER DUPLICATE. [TAR #35718]

RECORDS THAT CONTAIN VARIABLE REPEATING GROUPS NESTED TO TWO OR MORE LEVELS ARE NOW STORED AND RETRIEVED CORRECTLY.

AN AUTOMATIC MEMBER RECORD STORED AFTER A FIND USING DBK IS CORRECTLY REFLECTED IN ITS SET AND OWNER DIRECTORIES. [TARS #36367 AND #34482]

FSUBS

FSUBS NOW OPTIMIZES A SINGLE LEVEL ARRAY. [TAR #36004]

RLIB

AN ERROR WITH CONTYP 15 WILL NOW RETURN 'CLOSE OR EXIT DBMS IS INVALID WITHIN ACTIVE TRANSACTION' ERROR MESSAGE. [TAR #36993]

DISK ERROR HANDLING SUPPORT FOR DMLCP--A CHECK IS NOW MADE FOR SUCCESSFUL FORCE-WRITING OF DATA TO THE DISK.

THE LOCKING ALGORITHM NOW CHECKS FOR AN IDENTITY MATCH BETWEEN A LOCK REQUESTOR AND THE LOCK POSSESSOR, THEREBY ELIMINATING LOCK WAITS DURING AN ABORT TRANSACTION.

DURING THE EXECUTION OF AN EXIT DBMS, ONLY THOSE FILE UNITS WHICH THE RUN-UNIT HAS OPENED WILL BE CLOSED.

SCHED

WHEN NEW ITEMS ARE ADDED TO AN EXISTING RECORD TYPE AND A NEW SET (WHICH INCLUDES THE EDITED RECORD AS OWNER OR MEMBER) IS ALSO ADDED, THE RECORD OCCURRENCES IN THE DATABASE NOW PROPERLY REFLECT THE ENTIRE EDITING SESSION. [TAR #32582]

IF A NEW SET IS ADDED TO THE SCHEMA, AND ONE OR MORE OF THE KEYS IN THE SET (EITHER SORT OR SEARCH KEYS) IS SPECIFIED AS DESCENDING, THE SIZE OF THE KEY IS NOW CALCULATED CORRECTLY. [TAR #34471]

WHEN ADDING NEW FILES (E.G. AREAS, SETS) OR MODIFYING EXISTING FILES, SCHED WILL NOW CREATE DAM RATHER THAN SAM FILES. [TARS #24258, #34497]

> NOTE: SEE TECH PUBS DBMS UPDATE RELEASE FOR PROCEDURE TO REPLACE SCHED-CREATED SAM FILES (I.E. PRE-FIX FILES) WITH DAM FILES.

SCHEMA

IF THE SET OCCURRENCE SELECTION CLAUSE FOR A MEMBER RECORD OF A SET INDICATES THRU LOCATION MODE OF OWNER, AND THE ORDER OF THE SET IS NEXT OR PRIOR, THE SCHEMA COMPILER WILL NOW FLAG THIS CONDITION AS AN ERROR. THE SET OCCURRENCE SELECTION CLAUSE SHOULD SPECIFY CURRENT OF SET. [TAR #34467]

DBUTL

THE MONITOR COMMAND NO LONGER GIVES AN ACCESS VIOLATION DUE TO A REV 18.0 INTRODUCED BUG.

III. OUTSTANDING PROBLEMS

CDML

DOES NOT HANDLE ON ERROR CLAUSE PARAGRAPH NAMES WHICH BEGIN WITH NUMBERS. [TAR #12613]

DBACP

DOES NOT CONSISTENTLY CLOSE ALL OPENED FILES. [TARS #34472, #36602, #37900]

DOES NOT ACCEPT LOWER CASE INPUT. [TAR #33543]

DOES NOT ALWAYS RESTORE/EXPAND MULTI-VOLUME SAVES/FILES CORRECTLY. [TARS #36992, #32706]

DOES NOT HANDLE TAPE ERRORS GRACEFULLY. [TARS #34481, 36464]

DBUTL

SWITCH AREA (AREA X) DESTROYS DATA FOR ANY SUBSEQUENT SET COMMAND. [TAR #27966]

DMLCP

DOES NOT ALWAYS HANDLE BIT STRINGS CORRECTLY. [TARS #34469, #34470]

FIND USING DBK CAUSES STORE TO SET OD POINTER IN MEMBER RECORD TO ZERO. [TARS #34483, #36367]

EXECUTION OF A SECOND 'PROTECTED UPDATE' PROGRAM CAUSES THE FIRST PROGRAM TO ABORT AND DBMS TO STOP. [TAR #27283]

SET SORT ORDER WITH MEMBERS CONTAINING A FOUR SEGMENT SORT KEY AND AUTOMATIC INSERTION IS NOT MAINTAINED IN SORTED ORDER. [TAR #20941]

UNABLE TO ACCESS NEXT/PRIOR IN A SET FOLLOWING A DELETE OF A RECORD OCCURRENCE. [TARS #82630, #37971, #82606, #34678]

FSUBS

DOES NOT INDICATE THE LINE NUMBER OF A DUPLICATE ELEMENT NAME. [TAR #36876]

RLIB

BIT MAP OVERFLOW. [TARS #29298, #36006]

SCHDEC

DOES NOT ACCEPT SINGLE QUOTES AROUND A UFD AND PASSWORD TREENAME. [TAR #33119]

TRUNCATES OUTPUT SOURCE FILE TREENAME TO 35 CHARACTERS. [TAR

#36005]

TRUNCATES V99X PICTURE TO X. [TAR #33849]

SCHEMA

DATA TYPE OF 'DECIMAL' OR 'PICTURE' DOES NOT PERMIT CHECK RANGE CLAUSE USAGE. [TAR #23841]

SIGN CHARACTER IN A PICTURE CLAUSE IN A SCHEMA HAS NO EFFECT. DBMS RELIES ON SUBSCHEMA IN COBOL TO ENFORCE THE PRESENCE OR ABSENCE OF A SIGN. [TAR #34766]

IV. ENVIRONMENT

REV 18.2 DBMS REQUIRES PRIMOS 18.2, SEG 18.2 AND PL1LIB.

V. INSTALLATION AND BUILD PROCEDURES

THE BUILD IS STANDARD. WHAT FOLLOWS IS INSTALLATION AND GENERAL INFORMATION.

```
                              FILES ON SYSTEM TAPE
+                             --------------------


DBMSEX (UFD)
+ ------------

  C_INITINSTALLDBMSEX
  C_INSTALLDBMSEX
  C_SHAREDBMS

    CMDNCO (SUB-UFD)

      DBACP   \
      DBUTL    >-  PRODUCT INTERLUDES
      CLUP    /


    DBMSLB (SUB-UFD)
  DBACP   \
  DBUTL    >-  PRODUCT SEGMENT DIRECTORIES
  CLUP    /
  IDBMS.SEG    USED BY C_SHAREDBMS TO INITIALIZE SHARED SEGMENTS
  DUMP.SEG     DBMS FILE DUMP UTILITY
  SUMMARY.SAVE UTILITY TO SUMMARIZE DML COMMAND TIMINGS
  VFYPRT.SAVE  UTILITY TO FORMAT OUTPUT OF -VERIFY OPTION
  DB2001  \
  DB2003   \_  SHARED SEGMENTS FOR EXECUTABLE DBMS PRODUCTS
  DB2012   /
  DB4000  /
  C_LOAD.LIB   COMMAND STREAM TO CREATE SHARED DYNAMIC LIBRARY
  C_USER.LIB   COMMAND STREAM TO INSTALL USERS DYNAMIC LIBRARY
  HTAB.INS.PMA SOURCE NEEDED BY C_LOAD.LIB
  DYNT         SOURCE NEEDED BY C_USER.LIB


INFO (SUB-UFD)

  CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)


JOBS (SUB-UFD)

  CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.




     COPY (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


     COMPILE (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


     LOAD (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
       FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
       FILES OF THE FORM @.L.CPL LOAD SEGMENTS.


PDBMS (SUB-UFD)

  SCHDIR       SCHEMA DIRECTORY
  DALIST       DATA ADMINISTRATORS LIST
  FDMLER  \
  CDMLER   \_  ERROR MESSAGE FILES
```

```
            DAERRS     /
            DBMSE     /


          INSERT (SUB-UFD)

            CONTAINS FILES OF THE FORM @@.INS.FTN USED TO BUILD DBMS
            (SPECIFICALLY UFD-DATA.INS.FTN [SEE SECTION "CHANGING THE
            DATABASE FILE UFD NAME AND PASSWORD"] AND FUNIT-DATA.INS.FTN
            [SEE SECTION "INTRODUCTORY MESSAGE CONTROL"]).


     DBMSEXBIN (UFD)
+    ---------------

       CLUP.B     (SUB-UFD) --
       DBACP.B    (SUB-UFD)    \
       DBUTL.B    (SUB-UFD)     \
       DMLCP.B    (SUB-UFD)      !
       ILIB.B     (SUB-UFD)      !
       RLIB.B     (SUB-UFD)      !
       ULIB.B     (SUB-UFD)      !   THESE SUB-UFDS CONTAIN THE @@.BIN FILES
       CLIB.B     (SUB-UFD)      >-  FOR RUNTIME SHARED LIBRARIES AND
       IDBMS.B    (SUB-UFD)      !   EXECUTABLE PRODUCTS.
       ASI.B      (SUB-UFD)      !
       ASG.B      (SUB-UFD)     /
       DUMP.B     (SUB-UFD)    /
       SUMMARY.B  (SUB-UFD) --


     DBMSDEF (UFD)
+    -------------

       C_INITINSTALLDBMSDEF
       C_INSTALLDBMSDEF


          CMDNCO (SUB-UFD)

            SCHEMA     INTERLUDE


          DBMSLB (SUB-UFD)

            SCHEMA     SEGMENT DIRECTORY


     DBMSDEFBIN (UFD)
+    ----------------

       SCHEMA.B (SUB-UFD)  CONTAINS @@.BIN FILES FOR SCHEMA COMPILER.


     DBMSFTN (UFD)
```

```
+     -------------

          C_INITINSTALLDBMSFTN
          C_INSTALLDBMSFTN


             CMDNCO (SUB-UFD)

               FDML       \_   PRODUCT INTERLUDES
               FSUBS      /
               EXEC            ACTUAL SEGMENT DIRECTORY


             DBMSLB (SUB-UFD)

               FDML       \_   PRODUCT SEGMENT DIRECTORIES
               FSUBS      /
               C_FDML          EXEC PROCEDURE TO PRECOMPILE FTN DBMS APPLICATIONS
               C_FLOAD           "      "        " LOAD        "   "       "


             INFO (SUB-UFD)

               CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)



             JOBS (SUB-UFD)

               CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


                  COPY (SUB-UFD)

                    CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


                  COMPILE (SUB-UFD)

                    CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


                  LOAD (SUB-UFD)

                    CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
                    FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
                    FILES OF THE FORM @.L.CPL LOAD SEGMENTS.


             PDBMS (SUB-UFD)
          SCHDIR        SCHEMA DIRECTORY
          DALIST        DATA ADMINISTRATORS LIST
          FDMLER   \
          CDMLER    \_  ERROR MESSAGE FILES
          DAERRS    /
          DBMSE    /


       LIB (SUB-UFD)

          DMLLIB        DBMS RUN-TIME SHARED LIBRARY


   DBMSFTNBIN (UFD)
+  ----------------

     FDML.B  (SUB-UFD) \_   CONTAINS @@.BIN FILES FOR FTN PREPROCESSOR
     FSUBS.B (SUB-UFD) /    AND SUBSCHEMA COMPILER.


   DBMSCOB (UFD)
+  -------------

     C_INITINSTALLDBMSCOB
     C_INSTALLDBMSCOB


       CMDNCO (SUB-UFD)

          CDML    \_    PRODUCT INTERLUDES
          CSUBS   /
          EXEC          ACTUAL SEGMENT DIRECTORY


       DBMSLB (SUB-UFD)

          CDML    \_   PRODUCT SEGMENT DIRECTORIES
          CSUBS   /
          C_CDML       EXEC PROCEDURE TO PRECOMPILE COBOL DBMS APPLICATIONS
          C_CLOAD      "      "      " LOAD       "    "      "


       INFO (SUB-UFD)

          CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)


       JOBS (SUB-UFD)

          CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


            COPY (SUB-UFD)
```

```
                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

               COMPILE (SUB-UFD)

                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

               LOAD (SUB-UFD)

                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
                 FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
                 FILES OF THE FORM @.L.CPL LOAD SEGMENTS.

          PDBMS (SUB-UFD)

            SCHDIR      SCHEMA DIRECTORY
            DALIST      DATA ADMINISTRATORS LIST
            FDMLER \
            CDMLER  \_  ERROR MESSAGE FILES
            DAERRS  /
            DBMSE  /

          LIB (SUB-UFD)

            DMLLIB      DBMS RUN-TIME SHARED LIBRARY


     DBMSCOBBIN (UFD)
+    ----------------

       CDML.B  (SUB-UFD) \_   CONTAINS @@.BIN FILES FOR COBOL PREPROCESSOR
       CSUBS.B (SUB-UFD) /    AND SUBSCHEMA COMPILER.

     DBMSLGCL (UFD)
+    --------------

       C_INITINSTALLDBMSLGCL
       C_INSTALLDBMSLGCL


          CMDNC0 (SUB-UFD)

            SCHED  \_   PRODUCT INTERLUDES
            SCHDEC /


          DBMSLB (SUB-UFD)

            SCHED  \_   PRODUCT SEGMENT DIRECTORIES
            SCHDEC /
          INFO (SUB-UFD)

            CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)

          JOBS (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

               COPY (SUB-UFD)

                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

               COMPILE (SUB-UFD)

                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.



               LOAD (SUB-UFD)

                 CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
                 FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
                 FILES OF THE FORM @.L.CPL LOAD SEGMENTS.

          PDBMS (SUB-UFD)

            SCHDIR      SCHEMA DIRECTORY
            DALIST      DATA ADMINISTRATORS LIST
            FDMLER \
            CDMLER  \_  ERROR MESSAGE FILES
            DAERRS  /
            DBMSE  /


     DBMSLGCLBIN (UFD)
+    -----------------

       SCHED.B  (SUB-UFD) \    CONTAINS @@.BIN FILES FOR SCHEMA EDITOR
       SCHDEC.B (SUB-UFD)  >-  AND SCHEMA DECOMPILER.
       TEXTED.B (SUB-UFD) /



       INSTRUCTIONS FOR INITIAL INSTALLATION OF DBMS
+      ---------------------------------------------
```

1. IF YOU ALREADY HAVE A VERSION OF DBMS ON YOUR SYSTEM, SEE THE SECTION UPGRADING AN EXISTING DBMS INSTALLATION.

2. RESTORE THE UFDS SUPPLIED ON TAPE. THESE MAY BE ONE OR MORE OF THE FOLLOWING:

   DBMSEX   DBMSDEF   DBMSFTN   DBMSCOB   DBMSLGCL

3. ON EACH PARTITION WHERE DATABASE FILES ARE TO BE STORED, CREATE A UFD PDBMS WITH OWNER PASSWORD ISIS. THEN USE FUTIL PROTECT 7 1 TO GIVE NON-OWNER READ RIGHTS TO NEW UFDS. THE DIRECTORY OF ALL SCHEMAS (SCHDIR), THE LIST OF VALID DATA ADMINISTRATORS (DALIST), AND VARIOUS DBMS ERROR MESSAGE FILES ARE ASSUMED BY THE SYSTEM TO BE IN THE UFD PDBMS ON THE PARTITION WITH THE LOWEST LOGICAL DISK NUMBER.

   TO CREATE THESE FILES, DO A FUTIL UFDCPY FROM ANY ONE OF THE DBMSXXXX>PDBMS UFDS TO THE PDBMS WITH THE LOWEST LOGICAL DISK NUMBER.

4. ONCE THE VARIOUS UFDS DESCRIBED ABOVE HAVE BEEN CREATED, ATTACH TO THE MFD WHERE YOU WANT DBMS TO RESIDE AND RUN THE COMMAND INPUT STREAMS C_INITINSTALLDBMSXXX FROM EACH OF THE NON-BINARY UFDS RESTORED FROM TAPE. FOR EXAMPLE:

   ```
   CO DBMSEX>C_INITINSTALLDBMSEX
   ```

5. FINALLY USE FUTIL TO DELETE THE UFD(S) RESTORED FROM TAPE.

UPGRADING AN EXISTING DBMS INSTALLATION

TO UPGRADE AN EXISTING DBMS SYSTEM THE FOLLOWING STEPS SHOULD BE FOLLOWED:

1. SAVE ALL SCHEMAS TO TAPE. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18, THIS STEP WILL BE REPEATED AGAIN AFTER CONVERTING ALL SAVED SCHEMAS (SEE BELOW) SO THAT YOU WILL NEVER HAVE TO RESTORE AN UNCONVERTED SCHEMA FOR RECOVERY PURPOSES.

2. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18.2 USE FUTIL TO DO A UFDDEL OF THE CURRENT TOP LEVEL UFDS DBMS AND DBMSLB. THIS WILL CLEAN OUT THE OUTDATED SOFTWARE AND AVOID CONFUSION OVER WHICH VERSION IS CURRENT.

3. ATTACH TO EACH OF THE NON-BINARY UFDS RESTORED FROM

TAPE (DBMSXXXX) AND:

```
CO C_INSTALLDBMSXXXX
```

4. COPY THE ERROR MESSAGE FILES DAERRS, DBMSE, FDMLER, CDMLER FROM DBMSXXXX>PDBMS TO THE UFD PDBMS. (THIS ONLY NEEDS TO BE DONE ONCE. THEY ARE THE SAME IN ALL THE DBMSXXXX>PDBMS UFDS.)

5. SHARE DBMS FROM THE SYSTEM CONSOLE THUS:

   ```
   CO SYSTEM>C_SHAREDBMS (SEE SECTION DMLCP INSTALLATION)
   ```

6. WITH DBACP SET THE CONCURRENCY/RECOVERY ATTRIBUTES DESIRED WITH THE NEW DBACP COMMANDS.

7. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18, USE THE REV18 COMMAND OF DBUTL ON EVERY SCHEMA IN THE SYSTEM AND REPEAT STEP 1, SAVING ALL SCHEMAS TO TAPE.

   WARNING: THE REV18 CONVERSION IS TO BE DONE **ONLY** ON SCHEMAS CREATED WITH A REV 17 OR LOWER DBMS.

DATA ADMINISTRATOR AUTHORIZATION

THE FILE PDBMS>DALIST CONTAINS THE LOGIN NAMES OF ALL PERSONS AUTHORIZED AS VALID DATA ADMINISTRATORS. WITHOUT SUCH AUTHORIZATION, A USER MAY NOT USE ANY OF THE DBACP COMMANDS WHICH ALTER A DATABASE OR DISPLAY SENSITIVE INFORMATION (SUCH AS PRIVACY KEYS). DALIST IS ORGANIZED WITH ONE LOGIN NAME PER LINE. NAMES MAY BE ADDED OR DELETED USING ED. IF A LINE IS LEFT BLANK IT IS IGNORED. INITIALLY, THE FILE IS EMPTY. THE FIRST THREE LINES OF DALIST CONTAIN THE LOGIN NAMES OF THE PRIVILEGED DATA ADMINISTRATORS. THESE ARE DATA ADMINISTRATORS WHO MAY BYPASS THE VARIOUS SCHEMA PRIVACY LOCKS WHEN USING DBACP. A PRIVILEGED DATA ADMINISTRATOR WOULD BE RESPONSIBLE FOR THE MANAGEMENT AND INTEGRITY OF THE DBMS AS A WHOLE, INCLUDING THE MODIFICATION OF DALIST. SEE NEXT SECTION FOR PASSWORD PROTECTION OF PDBMS.

CHANGING THE DATABASE FILE UFD NAME AND PASSWORD

THE DATABASE ADMINISTRATOR CAN NOW CHANGE THE DEFAULT NAME AND PASSWORD FOR THE DATABASE FILE UFDS. THE CURRENT DEFAULTS ARE UFD PDBMS, WITH THE OWNER PASSWORD ISIS. TO DO THIS, EDIT THE FILE DBMSEX>INSERT>UFD-DATA.INS.FTN. CHANGE THE DATA STATEMENT FOR THE UFD NAME (VARIABLE ISIS) AND PASSWORD (VARIABLE ISPASS). THE UFD NAME AND PASSWORD ARE STILL LIMITED TO NO MORE THAN SIX CHARACTERS EACH. SEE THE FOLLOWING SECTIONS ON RELOADING PRODUCTS AND INTRODUCTORY MESSAGE CONTROL. THEN RELOAD ILIB, SCHEMA, DBUTL, DBACP, FSUBS, CSUBS, FDML, CDML, SCHED, SCHDEC, IDBMS, AND CLUP (OR WHATEVER SUBSET OF THIS LIST WAS DELIVERED).

THE DATABASE ADMINISTRATOR MUST THEN RENAME THE EXISTING DATA BASE FILE UFDS AND CHANGE THEIR PASSWORDS.

INTRODUCTORY MESSAGE CONTROL

THE USER HAS THE ABILITY TO INHIBIT THE PRINTING OF AN INTRODUCTORY MESSAGE AT RUN-TIME. IN THE INSERT FILE DBMSEX>INSERT>FUNIT-DATA.INS.FTN, IF THE VARIABLE "INTROM" IS SET TO .TRUE., AN INTRODUCTORY MESSAGE WILL BE PRINTED WHEN DBMS IS INVOKED. IF ITS VALUE IS .FALSE., NO MESSAGE WILL BE PRINTED.

TO SUPPRESS THE PRINTING OF THE INTRODUCTORY MESSAGE, EDIT THE FILE DBMSEX>INSERT>FUNIT-DATA.INS.FTN MAKING THE DESIRED CHANGES. THEN REFERING TO THE SECTION RELOADING PRODUCTS, RELOAD DMLCP.

RELOADING PRODUCTS

THERE ARE TIMES WHEN A SPECIFIC SUB-PRODUCT OF DBMS NEEDS TO BE RELOADED, THAT IS THE SEGMENT DIRECTORY NEEDS TO BE CREATED ANEW. TO DO THIS IT IS POSSIBLE TO USE THE SAME JOB STREAMS WHICH WERE USED IN THE ORIGINAL BUILDING OF THE COMPONENTS OF DBMS. HOWEVER, SINCE THE ORIGINAL BUILD WAS RUN UNDER A DIFFERENT UFD STRUCTURE, THE CPL COMMAND FILES WILL NOT WORK AS THEY ARE. THE SIMPLEST WAY TO ALLEVIATE THIS PROBLEM IS TO CREATE THE FOLLOWING UFD STRUCTURE AND MOVE THE NECCESSARY FILES INTO IT BEFORE RUNNING THE LOAD. (THE CPL PROCEDURE MERGE.CPL IN DBMSEX>JOBS DOES THIS.)

```
DBMS            (NO FILES)

        DBMSLB  (COPY THIS FROM TOP LEVEL UFD  DBMSLB)

        INSERT  (COPY THIS FROM DBMSEX>INSERT)

        JOBS    (COPY THIS FROM DBMSEX>JOBS)
```

BINARY (TRECOPY INTO HERE ALL NECCESSARY UFDS OF THE FORM XXXX.B FROM THE TOP LEVEL UFDS OF THE FORM DBMSXXXXBIN. AS A MINIMUM GET THE ENTIRE DBMSEXBIN BECAUSE THE LIBRARIES WILL BE NEEDED TO CREATE THE SHARED SEGMENTS AGAIN.)

```
DMLCP.B    (FROM DBMSEXBIN>DMLCP.B)

ILIB.B     (FROM DBMSEXBIN>ILIB.B)

SCHEMA.B   (FROM DBMSDEFBIN>SCHEMA.B)

 ETC.
```

NOW ATTACH TO DBMS>JOBS>LOAD AND FOR EACH SUB-PRODUCT WHICH NEEDS TO BE RELOADED RUN THE CPL PROCEDURE BY THE SAME NAME. NOTE THAT IF YOU ARE DOING ANY OF THE LIBRARIES (ULIB, CLIB, RLIB, ILIB, TEXTED, DMLCP, ASI, OR ASG), YOU SHOULD DO IT BEFORE ANY OTHERS SINCE THEY ARE INCLUDED IN THE OTHERS.

NOW ATTACH TO DBMS>DBMSLB. THE LOAD PROCEDURES FOR SUB-PRODUCTS PRODUCE SEGMENT DIRECTORIES OF THE FORM DB.XXXXX IN THIS UFD, SO YOU WILL NEED TO USE FUTIL'S TREDEL TO GET RID OF THE OLD VERSION OF THE SUB-PRODUCT(S) AND THEN CN TO PROMOTE THE NEW ONE.

BEFORE YOU CAN CREATE THE SHARED SEGMENTS BE SURE THAT THE PL1LIB IS IN TOP LEVEL UFD LIB. IF NOT, COPY IT FROM INDEX>SPL>LIB. NOW CREATE THE SHARED SEGMENTS THUS: CO DBMS>DBMSLB>C_LOAD.LIB. ALL THE NEW COMPONENTS ARE IN DBMS>DBMSLB AND YOU CAN USE FUTIL'S UFDCPY TO PROMOTE THEM TO THE TOP LEVEL UFD DBMSLB. WHEN THIS IS COMPLETE, THE ENTIRE UFD DBMS CAN BE DELETED TO RECOVER SPACE.

DMLCP INSTALLATION

DMLCP REQUIRES THE EXCLUSIVE USE OF SHARED SEGMENTS 2001, 2002, 2003 AND 2012 AND PRIVATE SEGMENTS 4030, 4031, 4032. TO INSTALL THE SHARED LIBRARY VERSION OF THE DML COMMAND PROCESSOR, THE FOLLOWING COMMAND MUST BE EXECUTED FROM THE SYSTEM CONSOLE AFTER EVERY COLD START:

```
CO SYSTEM>C_SHAREDBMS
```

THIS COMMAND STREAM INSTALLS THE DBMS SHARED LIBRARY, SHARES AND INITIALIZES THE DBMS SEGMENTS, AND INITIALIZES THE RING 3 SEMAPHORES. THIS COMMAND SHOULD BE INCORPORATED INTO C_PRMO, THE

COMMAND FILE WHICH IS ALWAYS RUN AFTER A COLD START.

CREATION OF A DML APPLICATION PROGRAM

+

ONCE A SCHEMA HAS BEEN WRITTEN AND COMPILED AND A SUBSCHEMA HAS BEEN WRITTEN AND COMPILED, AND THE DATA BASE FILES HAVE BEEN ALLOCATED WITH DBACP, THE USER CAN WRITE APPLICATION PROGRAMS FOR THE DATA BASE IN EITHER COBOL OR FORTRAN. THE SEQUENCE USED TO TRANSFORM THE SOURCE CODE INTO EXECUTABLE CODE IS AS FOLLOWS:

(1) PREPROCESS THE SOURCE CODE WITH THE HOST LANGUAGE PREPROCESSOR (CDML OR FDML).

(2) COMPILE THE OUTPUT OF THE PREPROCESSOR (D_XXXXX) WITH THE HOST LANGUAGE COMPILER (COBOL OR FTN).

(3) LINK THE BINARY OUTPUT OF THE COMPILER TO THE DML COMMAND PROCESSOR WITH THE SEGMENTED LOADER SEG.

SAMPLE JOB STREAMS TO DO THESE OPERATIONS WITH EITHER A COBOL OR FTN PROGRAM MAY BE FOUND IN UFD DBMSLB CALLED C_CDML, C_CLOAD, C_FDML, AND C_FLOAD. THESE JOB STREAMS ARE DESIGNED TO BE USED WITH THE EXEC UTILITY. FOR EXAMPLE, TO COMPILE AND LOAD A COBOL PROGRAM CALLED "PROG", EXECUTE THE FOLLOWING COMMAND:

```
EXEC DBMSLB>C_CDML PROG
```

THIS IS EQUIVALENT TO EDITING C_CDML AND C_CLOAD REPLACING EACH OCCURRENCE OF "&1" WITH "PROG", AND DOING A CO OF THE FILE.

TO COMPILE A FORTRAN PROGRAM, USE THE JOB STREAM C_FDML INSTEAD OF C_CDML AND C_FLOAD INSTEAD OF C_CLOAD.

THE OUTPUT FILES CREATED WHEN USING C_CDML OR C_FDML ON THE SOURCE FILE "PROG" ARE:

L_PROG - THE PREPROCESSOR AND COMPILER LISTINGS.

B_PROG - THE BINARY FILE OUTPUT BY THE COMPILER.

THE OUTPUT FILES FROM USING C_CLOAD OR C_FLOAD WITH PROGRAM "PROG" ARE:

M_PROG - SEG PROGRAM MAP.

#PROG - THE SEGMENTED RUN FILE.

THE RESULTING USER PROGRAM IS EXECUTED WITH THE COMMAND:

```
SEG #PROG
```

*
600 (F77)

PRIME'S FORTRAN 77 AT MASTER DISK REVISION 18.2 CONTAINS MAJOR PERFORMANCE IMPROVEMENTS IN COMPILATION SPEED, SELECTABLE OPTIMIZATION LEVELS, AND BUG FIXES. THIS COMPILER CONTAINS ALL FIXES PREVIOUSLY FOUND IN BOTH THE 18.1 AND 17.8 REVISIONS.

PERFORMANCE IMPROVEMENTS:

OVERALL COMPILATION RATE FOR REASONABLY-COMMENTED FORTRAN SOURCE TEXT SHOULD EXCEED 1500 LINES PER MINUTE. WE HAVE OBSERVED SPEED IN EXCESS OF 2500 LPM ON SOME REAL BENCHMARKS. AS OF THIS REVISION, PROGRAMS INITIALIZING LARGE ARRAYS IN DATA STATEMENTS WILL GENERATE MORE EFFICIENT BINARY OUTPUT DUE TO A ADDITION TO THE RANGE OF SUPPORTED OBJECT TEXT FORMATS. THE DETAILS OF THESE CHANGES NEED NOT BE UNDERSTOOD BY F77 USERS. THE TIME IN THE FTNDECLARE PHAS

AS SHOWN USING THE -STAT COMMAND LINE OPTION WILL DECREASE SIGNIFICANTLY.

PREVIOUSLY IMPLEMENTED CHANGES TO BOTH THE F77 CODE GENERATOR AND THAT OF THE PL1G COMPILER USED TO COMPILE F77 MAKE ADDRESSING ARRAY OFFSETS MUCH MORE EFFICIENT. THUS, SHORTER CODE IS GENERATED FOR REFERENCES LIKE X(I+5,J-4).

REWRITTEN MATH ROUTINES (SQRT, SIN, ETC.) ARE SLIGHTLY FASTER IN MOST CASES, ALTHOUGH THE MOTIVATION FOR THIS REWRITE WAS TO IMPROVE ACCURACY. ALL SINGLE PRECISION ROUTINES HAVE BEEN UPDATED AS OF REVISION 18.2.

NEW COMMAND OPTIONS:

IN ADDITION TO THE OPTIONS LISTED IN TABLE 7-2 OF THE FORTRAN 77 REFERENCE GUIDE (IDR 4027), THE USER MAY SELECT AMONG THREE LEVELS OF OPTIMIZATION: -OPT1, -OPT2, AND -OPT3; WHERE THE DEFAULT IS NORMALLY -OPT2. THE OLD OPTION -OPTIMIZE IS RETAINED AND IS SYNONYMOUS WITH -OPT2. THE CHOSEN LEVEL IS NOTED IN THE OPTION HEADER LINE OF THE COMPILER'S LISTING OUTPUT FILE. OPTIMIZATION IS TURNED OFF, AS BEFORE, BY SPECIFYING -NOOPT.

THE EFFECT OF SPECIFYING -OPT2 IS ELIMINATION OF THE OPTIMIZER LOGIC THAT MOVES INVARIENT CODE OUT OF LOOPS. THIS IS A COSLTY PROCESS THAT WAS FOUND TO CONSUME UP TO 15 PER CENT OF TOTAL COMPILE TIME ON PROGRAMS WITH MANY NEST DO LOOPS. IT IS STILL AVAILABLE USING -OPT3 AND IS USEFUL IN COMPILING FULLY DEBUGGED PROGRAMS TO BE USED IN FREQUENT PRODUCTION SITUATIONS. THE DEFAULT OPTIMIZATION(-OPT2) PERFORMS BOTH CODE PATTERN REPLACEMENT AND REDUNDANCY ELIMINATION. THE LOWEST LEVEL(-OPT1) DOES ONLY PATTERN REPLACEMENT.

AS OF 18.2, ANY LEVEL OF OPTIMIZATION MAY BE SET TO BE THE SITE DEFAULT BY USING THE DISTRIBUTED PROGRAM F77DF IN THE UFD F77>TOOLS. THESE VALUES ARE STORED IN THE DRIVER FILE CALLED F77DATA IN THE SYSTEM UFD SYSOVL. THIS FILE ALSO STORES THE ERROR MESSAGES RETURNED WHEN COMPILATION ERRORS OCCUR.

BUG FIXES:

SOURCE STATEMENT ORDERING PER ANSI X3.9-1978, SECTION 3.5, IS ENFORCED. SPECIFICATION STATEMENTS AND THE LIKE MUST PRECEDE EXECUTABLE CODE.

THE FREQUENTLY ENCOUNTERED DIFFICULTY WITH COMMON BLOCK REFERENCES WHICH MOST OFTEN RESULTED IN ERROR MESSAGES FROM SEG AT LOAD-TIME HAS BEEN FIXED.

IMPLIED DO-LOOPS IN DATA STATEMENTS(TAR 81506) NOW WORK CORRECTLY.

THE SOURCE LEVEL DEBUGGER DBG CAN ACCESS VERY LARGE COMMON BLOCKS.

USE OF $INSERT TO INCLUDE PROGRAM TEXT AND/OR PROGRAM UNITS HAS BEEN FIXED.

THE DCMPLX INTRINSIC FUNCTION IS INCLUDED. (AS OF 17.8)

WRITE AND PRINT* WITH NO I/O LIST PRODUCE BLANK LINES PER ANSI-78 STANDARD.

*
601-602 (NOT USED)

*
603 (PL1G)

DATE:- MARCH 2, 1981
SUBJECT: PL1G COMPILER FOR REV. 18.2

THIS MEMO DESCRIBES THE OPERATING PROCEDURES OF THE PL1G COMPILER. IN PARTICULAR, THE COMMAND LINE OPTIONS OF THE COMPILER ARE LISTED AND NOTES ON THE USE OF THE COMPILER ARE FURNISHED. THIS INFORMATION WILL ALLOW USERS ALREADY FAMILIAR WITH PL/I SUBSET G TO COMPILE, LOAD, AND EXECUTE THEIR PROGRAMS.

PL1G COMPILER FOR REV. 18.2

**** THERE ARE NO CHANGES FROM REV 17.8 ****

1 BUGS FIXED

GET LIST/EDIT NOW WORK CORRECTLY ON NULL ENTRIES.

GET EDIT ON F DESCRIPTOR NOW WORK CORRECTLY.

DECLARE STATEMENT NOW HANDLE UP TO 1024 ITEMS.

TITLE OPTION IN OPEN STATEMENT NOW ACCEPT THE '-' WITHIN THE FILE NAME.

IF STATEMENT NOW PROVIDE SAME RESULT INDEPEND ON SYNTAXES.

RELOAD X REGISTER AFTER CALL COSD.

2 PL1G COMMAND LINE OPTIONS

PL1G IS THE PL/I SUBSET G COMPILER. IT IS INVOKED BY THE COMMAND:

```
PL1G NAME [OPTIONS]
```

OPTIONS ARE PRECEDED BY A '-'. THE NAME MAY BE A PATH NAME, BUT NEITHER IT NOR ANY FILE NAME MADE FROM IT MAY EXCEED 128 CHARACTERS. THE COMMAND LINE SYNTAX IS THE SAME AS OTHER PRIME COMPILERS: THE -S, -B, AND -L OPTIONS ARE ALL SUPPORTED.

THE FOLLOWING OPTIONS ARE SUPPORTED:

```
-XREF        -- PRODUCE A CROSS-REFERENCE LISTING.  (IMPLIES L)
-OFFSET      -- PRODUCE AN OFFSET MAP IN L_NAME (IMPLIES L)
-EXPLIST     -- PRODUCE A PSEUDO-ASSEMBLY  LISTING  OF  THE  GENERATED
                CODE IN L_NAME (IMPLIES L)
-OPTIMIZE    -- EXECUTE THE OPTIMIZER PHASE
-NOOPTIMIZE  -- DON'T USE THE OPTIMIZER
-STATISTICS  -- PRINT OUT STATISTICS ABOUT THE COMPILATION
-RANGE       -- COMPILE CODE TO CHECK SUBSCRIPT AND SUBSTR RANGES
-UPCASE      -- MAP LOWER CASE TO UPPER CASE IN IDENTIFIERS
-LCASE       -- UPPER AND LOWER CASE ARE DISTINCT IN IDENTIFIERS
-NESTING     -- PUT A  NESTING LEVEL NUMBER IN THE LISTING (IMPLIES L)
-SILENT      -- SUPPRESS LEVEL 1 (WARNING) ERROR MESSAGES
-DEBUG       -- PRODUCE A FULL DEBUGGER (DBG) SYMBOL TABLE
-64V         -- PRODUCE V-MODE CODE
-32I         -- PRODUCE I-MODE CODE
-BIG         -- DENOTES THAT ARRAYS MAY BE LARGER THAN 1 SEGMENT
-PRODUCTION  -- PRODUCE "PRODUCTION" DEBUGGER SYMBOL TABLE
-ERRTTY      -- LIST ERRORS ON THE TERMINAL
-NOERRTTY    -- DO NOT LIST ERRORS ON THE TERMINAL
-ERRLIST     -- PRODUCE AN ERRORS-ONLY LISTING FILE
-FRN         -- ROUND THE FLOATING ACCUMULATOR BEFORE STORING A  FLOAT
BIN(23)
```

THE DEFAULT OPTIONS AS DISTRIBUTED ARE '-B YES -L NO -64V -OPTIMIZE -UPCASE -ERRTTY'. THE DEFAULT OPTIONS MAY BE CHANGED BY USE OF THE PROGRAM PL1GDF, WHICH IS FOUND IN THE TOOLS UFD.

EXAMPLE:

```
PL1G FOO -L YES -NESTING
```

WILL COMPILE FOO TO PRODUCE AN OBJECT FILE NAMED B_FOO AND A LISTING FILE NAMED L_FOO. THE LISTING WILL CONTAIN A NESTING LEVEL NUMBER AND THE CODE WILL BE OPTIMIZED.

EACH COMPILATION PRODUCES TEMPORARY FILES (NAMED "T$XXXX") IN THE CURRENT WORKING DIRECTORY. THESE FILES ARE NORMALLY DELETED AT THE END OF THE COMPILATION.

3 ERROR MESSAGES

FOUR LEVELS OF ERRORS ARE REPORTED: LEVEL 1 IS A WARNING, LEVEL 2 IS AN ERROR THAT HAS BEEN FIXED, LEVEL 3 IS AN ERROR THAT HAS NOT BEEN FIXED, AND LEVEL 4 IS AN ERROR THAT PREVENTS CONTINUED COMPILATION. ANY ERROR OF LEVEL 3 PREVENTS OPTIMIZATION AND CODE GENERATION IF DETECTED PRIOR TO THOSE PHASES.

4 PROGRAM LOADING

AT REV. 17.8 THE PL1G COMPILER OUTPUTS V-MODE AND I-MODE CODE. THUS, THE SEGMENTED LOADER (SEG) MUST BE USED TO LOAD THE OBJECT MODULES PRODUCED BY THE COMPILER. ALSO, THE PL1G LIBRARY MUST BE LOADED PRIOR TO LOADING THE STANDARD LIBRARY. THIS LIBRARY IS NAMED PL1GLB AND IS LOCATED IN UFD LIB. THUS, THE FOLLOWING COMMANDS ISSUED TO SEG'S VIRTUAL LOADER SHOULD BE USED FOR PL1G PROGRAMS (AFTER ALL USER MODULES HAVE BEEN LOADED):

```
$LI PL1GLB
$LI
```

5 MISCELLANEOUS NOTES

5.1 CROSS-REFERENCE OPTION

THE CROSS-REFERENCE OPTION MAY CAUSE THE COMPILER'S VIRTUAL SYMBOL SPACE TO OVERFLOW FOR VERY LARGE SOURCE PROGRAMS.

5.2 SEGMENT USAGE

COMPILATION OF PL1G PROGRAMS USES SEGMENTS 4004-4007 AND 4027. IN USER PROGRAMS SEGMENTS 4027 THROUGH 4010 ARE USED - IN DESCENDING ORDER - AS THE SYSTEM FREE STORAGE POOL (IN WHICH ALLOCATE AND FREE REQUESTS OPERATE AND IN WHICH SOME COMPILER-GENERATED TEMPORARIES ARE ALLOCATED).

5.3 ONCODE BUILTIN FUNCTION

VALUES RETURNED BY THE ONCODE BUILTIN FUNCTION ARE DIVIDED INTO TWO CLASSES ACCORDING TO WHETHER OR NOT THEY REPRESENT AN INPUT-OUTPUT ERROR. VALUES WHICH ARE LESS THAN THE VALUE OF THE SYMBOL "ONCODE_BASE" ARE INPUT-OUTPUT ERRORS AND VALUES GREATER THAN OR EQUAL TO "ONCODE_BASE" REPRESENT ALL OTHER RUNTIME ERRORS. THIS SYMBOL IS DEFINED IN THE FILE SYSCOM>ONCODES.PL1. SINCE THE VALUES RETURNED BY THIS FUNCTION ARE SUBJECT TO CHANGE, IT IS RECOMMENDED THAT THIS FILE BE INCLUDED IN THE SOURCE FILE (%INCLUDE 'SYSCOM>ONCODES.PL1') AND THE SYMBOLIC KEYS IN THE FILE REFERENCED

INSTEAD OF THE NUMERIC VALUES THEMSELVES.

THE SYMBOLS DEFINED IN THIS FILE REPRESENT ALL ERRORS WHICH ARE NOT RELATED TO INPUT-OUTPUT. THUS, THESE SYMBOLS SHOULD HAVE THE VALUE OF THE SYMBOL "ONCODE_BASE" ADDED TO THEM BEFORE THEY ARE USED IN CALLING "SIGNL$", SO THAT THEY ARE NOT CONFUSED WITH THE INPUT-OUTPUT RELATED ERRORS.

TWO ONE-DIMENSIONAL ARRAYS OF CHARACTER STRINGS ARE ALSO DEFINED IN THIS FILE. THEY CONTAIN THE TEXT OF THE ERROR MESSAGES OUTPUT BY THE DEFAULT ONUNIT HANDLER. THE ARRAY "IO_ONCODE_MESSAGE" - WHICH CONTAINS STRINGS DECLARED AS "CHAR(68) VARYING" - CONTAINS THE TEXT OF THE INPUT-OUTPUT RELATED ERROR MESSAGES, AND THE ARRAY "ONCODE_MESSAGE" - WHICH CONTAINS STRINGS DECLARED AS "CHAR(56) VARYING" - CONTAINS THE TEXT OF ALL THE OTHER POSSIBLE ERROR MESSAGES. TO ACCESS THE MESSAGE CORRESPONDING TO A GIVEN ONCODE VALUE, THE FOLLOWING CONSTRUCTS SHOULD BE USED:

```
ONCODE_VAL = ONCODE();
IF ONCODE_VAL > 0 & ONCODE_VAL <= MAX_IO_ONCODE
   THEN MSG = IO_ONCODE_MESSAGE(ONCODE_VAL);
ELSE IF ONCODE_VAL >= ONCODE_BASE &
               ONCODE_VAL < ONCODE_BASE + NEXT_AVAILABLE_CODE
   THEN MSG = ONCODE_MESSAGE(ONCODE_VALUE - ONCODE_BASE + 1);
ELSE /* NO MESSAGE AVAILABLE */;
```

*
604   (POWERPLUS)

POWERPLUS INFO -- REV 18.2

SPECIAL INSTALLATION PROCEDURES FOR REV 18.2

AT REV 18.2, POWERPLUS CODE RELATING TO SCREENS HAS BECOME EVEN MORE DEPENDENT ON THE SPECIFIC INDEX NUMBER (NOT NAME) OF A USER'S TERMINAL. HENCE, WHEN REV 18.2 IS INSTALLED, BOTH TERM** AND TERM## SHOULD BE COPIED FROM POWERPLUS>POWER*. IF THE CUSTOMER USES OTHER THAN THE 8 PR1ME SUPPORTED TERMINALS, THE APPROPRIATE TERMINAL COMINPUT FILE SHOULD THEN BE RUN.

CHANGES SINCE REV 18.1

THE FOLLOWING BUGS HAVE BEEN FIXED IN REV 18.2:

TAR 35382 'WARNING - LINE TRUNCATED' MESSAGE IS DISPLAYED WHEN COMBINING SETS.

TAR 35383 INCORRECT NUMBER OF ENTRIES GENERATED ON A COMBINE.

TAR 11983 FORTRAN I/O ERROR OCCURS ON A COMPUTE WITH NO SETS AVAILABLE.

TAR 11982 FIND ON A LONG INTEGER GIVES INCORECT RESULTS (ALSO TAR 20786.)

TAR 82294 CHANGE DESCRIPTOR NAME FOLLOWED BY CHANGE DESCRIPTION CORRUPTS FILENAME.

TAR 36763 WHEN A CHARACTER DESCRIPTOR IS MISSING FROM A BATCH ADD FILE, POWER INSERTS THE VALUE FROM THE PREVIOUSLY ADDED RECORD.

TAR 82285 OCCURRANCE OF ERROR IN CHANGE DESCRIPTOR CAUSES PREVIOUS CHANGES NOT TO BE MADE.

TAR 25397 NO PROMPT IS ISSUED AFTER POWER PROCESSES A COMMENT LINE IN A PROCEDURE.

TAR 32173 INCORRECT OUTPUT OF VARIABLES IN REPORTS SUPPRESSING DUPLICATE NUMERIC VARIABLES.

TAR 32818 SPECIFYING ZERO TITLE LINES RUINS THE FIRST LINE OF THE HEADING.

TAR 82023 MEMBER LINKED FIELDS ARE OFFSET IN A LIST DISPLAY.

TAR 36521 VALIATE DOES NOT WORK PROPERLY WHEN USING MORE THAN 51 NUMERIC FIELDS.

TAR 37513 A REPORT WITH ZERO TITLE LINES AND/OR NUMERIC DESCRIPTORS (N1 - N99) CANNOT BE MODIFIED.

TAR 82202 UNEQUAL FIND ON A CHARACTER FIELD RETURNS EQUAL AS WELL AS CORRECT DATA.

TAR 81600 INCORRECT FIELD IS ADDED WHEN $FILL IS ENTERED AFTER ERRONEOUS ENTRY IN VALIDATION MODE.

TAR 36242 BLANK DATE INDICATOR OF '9999' CONNOT BE SUPPRESSED IN A REPORT. (THE '9999' INDICATOR WILL STILL APPEAR IN A DISPLAY.

TAR 34346 COMPUTE VERB INSERTS MONTH IN A DATE FIELD CONTAINING ONLY A YEAR.

TAR 82128 POWER CREATES NULL OR INCORRECT SPOOL NAMES FOR SPOOLED

FILES.

DOCUMENTATION ADDITIONS

TAR XXXXX UNLESS THE USER HAS CREATED A HEADING FOR A FILE, ALL DESCRIPTORS FROM THE FILE WILL BE DISPLAYED USING POWER'S DEFAULT FORMATS AS LISTED BELOW:

DATA TYPE	DEFAULT DISPLAY
NUM1 (R*8)	-ZZZZZZZ.##
NUM2 (R*4)	-ZZZZZZZ.##
NUM3 (I*2)	-ZZZZZ
NUM4 (I*4)	-ZZZZZZZ.##
NUM5 (DECIMAL)	-ZZZZZZZ.##
NUM6 (COMP-3)	-ZZZZZZZ.##

IF THESE DEFAULT DISPLAYS ARE NOT DESIRED, THE USER SHOULD CREATE A HEADING (USING HEADING CREATE).

TAR 33638 A MIDAS SEARCH DESCRIPTOR MAY NOT BE ADDED OR HAVE ITS DATA TYPE CHANGED WITH THE ADD AND CHANGE OPTIONS OF THE CREATE COMMAND. IF A USER DESIRES TO ADD A NEW SEARCH DESCRIPTOR (OR CHANGE A DISPLAY DESCRIPTOR TO A SEARCH DESCRIPTOR) OR CHANGE THE DATA TYPE OR LENGTH, THEY SHOULD PERFORM THE FOLLOWING STEPS:

1) DUMP ALL DATA TO A FILE.

2) DESTROY THE FILE IN POWER.

3) EXIT POWER AND TREDEL THE DATA FILE.

4) ENTER POWER AND RECREATE THE FILE AS DESIRED.

5) BATCH ADD THE DATA.

IF ANY DESCRIPTOR NAMES ARE BEING CHANGED FOR THE NEW

FILE, THESE NAME CHANGES SHOULD BE MADE ON THE OLD FILE (USING CHANGE DESCRIPTOR OPTION) , ELSE DATA IN THOSE DESCRIPTORS WILL NOT BE ADDED.

IF A USER HAS DATA IN A FILE, THE METHOD DESCRIBED ABOVE SHOULD BE USED TO MODIFY ALL TYPES OF FILES.

PT65 POWER NOW FUNCTIONS WITH THE NEW PR1ME PT65 TERMINAL. USERS MUST INITIATE THE ONTEL DOWNLINE-LOADER (THE USER TYPES OA->TERM) BEFORE ENTERING POWER. AS THE PT65 OPERATES IN LOW INTENSITY MODE, THE LOW INTENSITY VISUAL ATRIBUTE IN A POWER SCREEN WILL NOT FUNCTION. ENTERING THIS CHARACTERISTIC WILL HAVE NO EFFECT ON THE SCREEN.

PLEASE NOTE THAT FIELDS MUST NOT BEGIN BEFORE COLUMN 3 IN SCREENS USED BY A PT65.

*
605 (VISTA)

SUBJECT: VISTA

RELEASE: REV 18.2

DATE: APRIL 9, 1981

THIS DOCUMENT OUTLINES THE OUTSTANDING PROBLEMS, ENVIRONMENT, AND INSTALLATION PROCEDURES FOR DBMS/QUERY (VISTA), THE DBMS QUERY REPORT WRITER. INCLUDED ARE STEP-BY-STEP INSTRUCTIONS FOR INSTALLATION AND DESCRIPTIONS OF THE CONFIGURATION FILE AND UFD STRUCTURE NECESSARY FOR INSTALLATION.

OUTSTANDING PROBLEMS

GENERAL

* A FATAL ERROR IN DBMS INVARIABLY CAUSES A DBMS "INFINITE ERROR LOOP". VISTA DOES NOT ACTUALLY GO INTO AN INFINITE LOOP; THE LOOP IS CAUGHT AND TERMINATED.

* WHEN A USER ABORTS A COMMAND LEVEL SORT BY HITTING <BREAK> OR <CNTRL-P>, THE TABLE BEING SORTED IS DESTROYED.

REPORT GENERATOR

* "LIST DET P1 = 5/0;" (IN A FORMAT) CAUSES:
  1) NO DIVIDE-BY-ZERO CONDITION TO BE RAISED.
  2) P1 TO BE PRINTED AS 0 IN REPORT.
  3) INCONSISTANT RECOVERY: IN SOME RUNS, VISTA RETURNS TO VISTA COMMAND LEVEL NORMALLY; IN OTHERS, A FATAL VISTA ERROR IS SIGNALLED AND THE USER IS THROWN OUT TO PRIMOS.

  THIS BUG SEEMS TO BE CAUSED BY A COMPILER LIMITATION.

* WHEN THE LIST DETAIL USED IN THE FIRST DETAIL ITEM HAS AN OCCURS CLAUSE, AN EXTRA LINE IS INSERTED BETWEEN TWO SUCCESSIVE TABLE ROWS.

* THE RG SKIPS MORE LINES THAN NECESSARY BETWEEN DIFFERENT SECTIONS OF THE REPORT.

* THE RG ACCEPTS THE SPECIFICATION THAT AN ITEM SHOULD BE PRINTED ON LINE 00, BUT PLACES IT AT LINE 1. THIS SHOULD NOT BE ACCEPTED AT ALL; IT SHOULD BE FLAGGED AS AN ERROR.

* IN LIST AND BLOCK DETAIL, IF EJECT AND A FOOTER ARE SPECIFIED IN A FORMAT, THE FOOTER IS NOT PRINTED AT THE BOTTOM OF THE PAGE WITHIN THE BOTTOM MARGIN (EXCEPT FOR THE LAST PAGE). IT IS PRINTED ON THE LINE IMMEDIATELY BELOW THE INFORMATION REQUESTED.

ENVIRONMENT

TO USE REV 18.2 DBMS/QUERY, IT IS NECESSARY TO INSTALL A REV 18.2 DBMS WHICH INCLUDES NEW VERSIONS OF FSUBS, CSUBS, AND DMLCP. IT SHOULD BE NOTED THAT THE UFD VISTA>DBMSLB CONTAINS A NEW DBMS SEGMENT, VI2070. THIS SEGMENT, WHICH IS REQUIRED FOR DBMS/QUERY TO OPERATE, IS SHARED BY C_SHAREVISTA. ALSO REQUIRED IS A REV 18.2 OPERATING SYSTEM (PRIMOS).

INSTALLATION INSTRUCTIONS

TO INSTALL DBMS/QUERY, TAKE THE FOLLOWING ACTIONS:

1) ATTACH TO THE MFD WITH THE OWNER PASSWORD AND RESTORE THE VISTA TAPE (THIS WILL CREATE THE VISTA UFD AND ITS SUB-UFD'S).

2) CREATE A TOP LEVEL UFD NAMED VISTA*, IF IT DOESN'T ALREADY EXIST. THIS UFD WILL CONTAIN ALL OF THE FILES AND UFDS NECESSARY TO SUPPORT THE EXECUTION OF DBMS/QUERY.

3) MODIFY VISTA>C_INSTALLVISTA TO CONTAIN THE OWNER PASSWORDS NECESSARY TO MODIFY THE SYSTEM UFDS CMDNC0 AND SYSTEM.

4) INSTALL REV 18.2 DBMS ON THE SYSTEM AS PER THE INSTRUCTIONS INCLUDED WITH THAT PRODUCT.

5) MODIFY (IF NECESSARY) THE CONFIGURATION FILE, VISTA>SYSTEM> VISTA.CONFIG (SEE ADDENDUM 2 OF THIS DOCUMENT FOR A FULL DESCRIPTION), TO FIT THE SYSTEM DBMS/QUERY IS BEING INSTALLED ON. THIS FILE WILL BE COPIED TO THE SYSTEM UFD; IT IS THE FILE IN SYSTEM WHICH IS ACTUALLY USED BY DBMS/QUERY. NOTE: IF A MODIFIED CONFIGURATION FILE ALREADY EXISTS IN YOUR SYSTEM UFD, WHEN C_INSTALLVISTA IS RUN, IT WILL BE OVERWRITTEN. THE EASIEST WAY TO AVOID RE-MODIFYING THE CONFIGURATION FILE IS TO COPY THE CONFIGURATION FILE FROM SYSTEM TO VISTA>SYSTEM>VISTA.CONFIG BEFORE STARTING C_INSTALLVISTA.

6) RUN C_INSTALLVISTA (TYPE "CO C_INSTALLVISTA"). THIS WILL COPY THE NECESSARY UFDS AND FILES FROM THE VISTA UFD TO THE APPROPRIATE SYSTEM UFDS (SEE ADDENDUM 1 OF THIS DOCUMENT FOR A FULL DESCRIPTION OF THE VISTA UFD).

7) FROM THE SYSTEM CONSOLE, SHARE DBMS USING SYSTEM>C_SHAREDBMS.

8) FROM THE SYSTEM CONSOLE, SHARE DBMS/QUERY USING THE COMMAND FILE SYSTEM>C_SHAREVISTA. THIS WILL SHARE THE VISTA SEGMENTS AND CONFIGURE THE SYSTEM BASED UPON THE CONTENTS OF THE FILE SYSTEM>VISTA.CONFIG.

9) COBOL AND FORTRAN SUBSCHEMAS WHICH ARE GOING TO BE ACCESSED BY DBMS/QUERY MUST BE DELETED AND RECOMPILED USING THE REV 18.2 COBOL AND FORTRAN SUBSCHEMA COMPILERS.

10) DBMS/QUERY IS NOW READY FOR USE AND MAY BE INVOKED BY TYPING THE COMMAND "VISTA" AT THE TERMINAL. INFORMATION ABOUT THE USE OF DBMS/QUERY MAY BE OBTAINED FROM THE REV 18.2 DBMS/QUERY MANUALS OR BY TYPING THE COMMAND "HELP" AFTER INVOKING THE SUBSYSTEM.

ADDENDUM 1: DBMS/QUERY UFD STRUCTURE

THE STRUCTURE OF THE VISTA UFD (WHICH RESIDES IN THE MFD AND IS CREATED WHEN THE VISTA TAPE IS RESTORED) IS AS FOLLOWS:

```
VISTA
          CMDNC0
                     VISTA          THE COMMAND WHICH INVOKES THE QUERY
                                    REPORT WRITER SUBSYSTEM.

          SYSTEM
                     VI2073         THE SHARED CODE AND DATA OF DBMS/QUERY;
                     VI2074         THE ACTUAL RUNFILE TO BE SHARED.
                     VI2075
                     VI2076

          DBMSLB
                     VI2070         THE DBMS SHARED SEGMENT REQUIRED FOR
                                    DBMS/QUERY TO OPERATE.

          BINARY                    DBMS/QUERY BINARIES.

          VISTA_CONFIG.SEG          THE PROGRAM WHICH IS RUN TO SET
                                    DBMS/QUERY'S CONFIGURABLE PARAMETERS
                                    WHEN THE SYSTEM IS SHARED.

          VISTA.CONFIG              THE FILE ACCESSED BY VISTA_CONFIG.SEG;
                                    CONTAINS THE CONFIGURATION PARAMETER
                                    VALUES.

          VISTA*
                     ERROR_FILE     THE FILE WHICH CONTAINS THE DBMS/QUERY
                                    ERROR MESSAGES.

                     ERROR.LOG      THIS FILE WILL BE CREATED BY DBMS/QUERY
                                    IF A SYSTEM ERROR OCCURS; CONTAINS
                                    INFORMATION USEFUL IN FIXING THE
                                    PROBLEM.

                     HELP           A UFD WHICH CONTAINS DBMS/QUERY'S HELP
                                    DATABASE.

                     CATALOG        A UFD STRUCTURE FOR THE DBMS/QUERY
                                    CATALOGS.
```

INFO	A UFD CONTAINING RUNOFF FILES TO HELP WITH THE INSTALLATION OF DBMS/QUERY.

NOTE THAT C_INSTALLVISTA (WHICH COPIES EACH PORTION OF DBMS/QUERY TO THE APPROPRIATE UFD) DOES NOT DEAL WITH THE PASSWORDS OR PROTECTION OF ANY OF THE FILES OR UFDS IT COPIES. THEREFORE, IT IS THE RESPONSIBILITY OF THE INSTALLER TO MAKE SURE THE NEW FILES/UFDS ARE PROPERLY PROTECTED AND THAT C_INSTALLVISTA IS CHANGED TO CONFORM TO THE PASSWORDS PRESENT ON THE SYSTEM DBMS/QUERY IS BEING INSTALLED ON. (THE SUGGESTED PROTECTION VALUES FOR THE VISTA* UFD ARE ALL ACCESS FOR OWNERS, NONE FOR NON-OWNERS; THE REQUIRED PROTECTION VALUES FOR ALL FILES COPIED TO THE SYSTEM AND CMDNC0 UFDS ARE ALL ACCESS FOR OWNERS, READ-ONLY FOR NON-OWNERS.)

ADDENDUM 2: CONFIGURATION FILE FORMAT

THE DBMS/QUERY CONFIGURATION FILE, SYSTEM>CONFIG.VISTA, CONSISTS OF 16 LINES. EACH LINE MUST BE EXACTLY AS DESCRIBED IN THESE INSTRUCTIONS OR DBMS/QUERY CANNOT BE EXPECTED TO WORK PROPERLY. THE INFORMATION ON EACH LINE IS AS FOLLOWS:

LINE 1: THE NUMBER OF CHARACTERS PER LINE ON THE TTY DBMS/QUERY IS RUN WITH. THIS NUMBER SHOULD BE 1 CHARACTER LESS THAN THE ACTUAL SCREEN WIDTH TO AVOID UNWANTED AUTOMATIC LINEFEEDS. (DEFAULT = 79) NOTE: THIS NUMBER SHOULD BE GREATER OR EQUAL TO 71 FOR OPTIMUM PERFORMANCE OF DBMS/QUERY.

LINE 2: THE NUMBER OF LINES PER SCREEN ON THE TTY DBMS/QUERY IS RUN WITH. THIS NUMBER SHOULD BE 1 LESS THAN THE ACTUAL SCREEN LENGTH TO ALLOW FOR THE SCROLLING PROMPT. (DEFAULT = 23)

LINE 3: THE NUMBER OF CHARACTERS PER LOGICAL LINE ON THE PRINTER DBMS/QUERY IS RUN WITH; THE NUMBER OF CHARACTERS ON THE LINE AFTER THE PRINTER HAS INSERTED ITS SIDE MARGINS. (DEFAULT = 108)

LINE 4: THE NUMBER OF LINES PER LOGICAL PAGE ON THE PRINTER DBMS/QUERY IS RUN WITH; THE NUMBER OF LINES ON THE PAGE AFTER THE PRINTER HAS INSERTED ITS TOP AND BOTTOM MARGINS. (DEFAULT = 47)

LINE 5: THE NAME OF THE MASTER DBMS UFD (WHERE THE SCHEMAS ARE STORED). (DEFAULT = 'PDBMS')

LINE 6: THE OWNER PASSWORD OF THE MASTER DBMS UFD. (DEFAULT = 'ISIS')

LINE 7: THE NAME OF THE MASTER DBMS/QUERY UFD. (DEFAULT = 'VISTA*')

LINE 8: THE OWNER PASSWORD OF THE MASTER DBMS/QUERY UFD. (DEFAULT = '')

LINE 9: THE OWNER PASSWORD OF THE DBMS/QUERY CATALOG UFD (WHERE THE PROCEDURES, FORMATS AND ABBREVS ARE STORED). (DEFAULT = '')

LINE 10: THE MASTER UFD OF THE DBMS/QUERY HELP SUBSYSTEM FILES. (DEFAULT = 'VISTA*)

LINE 11: THE OWNER PASSWORD OF THE MASTER HELP UFD. (DEFAULT = '') (NOTE: IF THE DEFAULT NAME, 'VISTA>VISTA*' IS USED AND THE MASTER DBMS/QUERY UFD NAME IS LEFT AS THE DEFAULT, THEN THE PASSWORDS OF THE MASTER HELP AND MASTER DBMS/QUERY UFD'S MUST BE THE SAME, SINCE THE UFD'S ARE THE SAME.)

LINE 12: THE DBMS/QUERY HELP UFD (THE ACTUAL DATA FILES OF THE HELP SUBSYSTEM RESIDE HERE. (DEFAULT = 'HELP')

LINE 13: THE DBMS/QUERY HELP SUBSYSTEM UFD OWNER PASSWORD. (DEFAULT = '')

LINE 14: THE DBMS/QUERY HELP SUBSYSTEM TOPMOST LEVEL PREFIX. SINCE THE HELP SUBSYSTEM PRINTS THE ACTUAL UFD NAME WHERE IT IS CURRENTLY LOCATED, IT DELETES THE TOPMOST (PROTECTED) UFD NAMES AND THEIR PASSWORDS FROM THE HELP SUBSYSTEM HEADER. THIS PREFIX REPLACES THE DELETED PORTION. (DEFAULT = 'HELP DBMS/QUERY')

LINE 15: THE SCROLLING DEFAULT: IF 'SCROLL ENABLED' IS TO BE THE DEFAULT, SET TO '1'B, IF 'SCROLL DISABLED' IS TO BE THE DEFAULT, SET TO '0'B. (DEFAULT = '1'B)

LINE 16: THE NUMBER OF VIRTUAL RECORDS RETRIEVED BETWEEN PRINTING THE VIRTUAL RECORD COUNT. (DEFAULT = 1) NOTE: IF DBMS/QUERY WILL BE USED ON A HARD-COPY TERMINAL, THIS CONSTANT SHOULD BE SET TO A LARGE NUMBER (UP TO 32767).

*
606-612 (RJE PRODUCTS)

SEE <M182A1>RJECOM (HISTORY FILES) FOR INFO

CHANGES AT REV. 18.3

IT IS IMPORTANT THAT REV 18.3 PRIMOS AND REV 18.3 SHARED LIBRARIES ALL BE INSTALLED AT THE SAME TIME ON A SYSTEM. REV 18.3 SHARED LIBRARIES WILL NOT WORK WITH REV 18.2.PRIMOS. THIS IS ALSO TRUE FOR MOST OF THE LANGUAGE PRODUCTS WHICH DO NOT HAVE SHARED LIBRARIES.

AT REV 18.3 THE SPL LIBRARY MUST BE SHARED FOR MANY PRODUCTS TO WORK. THIS CAN BE DONE BY RUNNING THE FILE C_SHLB IN UFD SYSTEM. THE FOLLOWING COMMANDS SHOULD BE ADDED TO THE C_PRMO FILE IN CMDNCO.

```
SHARE SYSTEM>SP2121 2121
R SYSTEM>SP4000 1/10
```

*
643 (BATCH)

ABSTRACT

BATCH HAS NO FUNCTIONAL IMPROVEMENTS AT REVISION 18.3 VS. REVISION

18.1 (ITS LAST RELEASE).

HOWEVER, IT HAS MANY BUG FIXES AND IMPROVEMENTS. THESE ARE ALL DESCRIBED HEREIN.

THE MAJOR CHANGE TO BATCH AT REV 18.3 IS THE INSTALLATION OF A NEW QUEUE CONTROL FILE MANAGEMENT SYSTEM. IT IS SIMILAR IN MANY RESPECTS TO THE OLD ONE, BUT IS FASTER, EASIER TO UNDERSTAND AND MAINTAIN.

BECAUSE OF THE NEW QUEUE MECHANISM, ALL INSTALLATIONS WHICH INSTALL REV 18.3 BATCH MUST INVOKE THE COMMAND FILE C_RSET WHILE ATTACHED TO BATCHQ AS AN OWNER. C_BDIF MAY ALSO BE RUN, BUT IT CREATES A NEW (NULL) BATDEF FILE, WHICH IS NOT NECESSARY; OLD BATDEF FILES (FROM REV 18.1) WILL WORK UNDER REV 18.3 BATCH.

ASIDE FROM THAT, REV 18.3 BATCH IS COMPLETELY COMPATIBLE WITH REV 18.1 BATCH. THE FOLLOWING IMPROVEMENTS WERE MADE TO BATCH AT REV 18.3:

O THE SUBROUTINE TIME$B NOW HAS AN ARGUMENT. IT IS AN INTEGER*2 VALUE WHICH ACTS AS A KEY TO THE SUBROUTINE. WHEN THAT KEY IS 0, THE SUBROUTINE FUNCTIONS AS IT DID BEFORE.

  TIME$B WILL NOW PRINT THE DATE IF IT IS A NEW DAY, OR WHEN IT IS FIRST CALLED. IT WILL THEN PRINT THE TIME ON THE NEXT LINE AS USUAL. ALSO, IT WILL ONLY PRINT THE TIME IF IT IS DIFFERENT FROM THE LAST TIME TIME$B WAS CALLED; ELSE, IT WILL PRINT SUFFICIENT SPACES TO MAKE EVERYTHING LINE UP.

  CALLING TIME$B WITH A KEY OF 0 CAUSES IT TO CHECK FOR BATMON SET AND NEWLIN SET (INDICATING THAT WE ARE THE BATCH MONITOR AND AT A NEW LINE). IF EITHER ARE NOT SET, THE ROUTINE RETURNS. OTHERWISE, IT PRINTS THE TIME, DATE/TIME, OR SPACES AS APPROPRIATE AND THEN RETURNS.

  CALLING TIME$B WITH A KEY OF 1 CAUSES IT TO FORCE A REPRINT OF THE DATE AND TIME THE NEXT TIME TIME$B IS CALLED.

  CALLING TIME$B WITH A KEY OF 2 CAUSES IT TO ONLY FORCE A REPRINT OF THE TIME THE NEXT TIME TIME$B IS CALLED.

O THE METHOD FOR SENDING MESSAGES TO THE CONSOLE IS TO USE THE SMSG$ SUBROUTINE. AT REV 18.3, THE SUBROUTINE IS CALLED WITH ALL 0 ARGUMENTS WHEN DEFINING THE RECEIVER - THIS SPECIFIES THE SYSTEM CONSOLE, AND AN "IMMEDIATE" MESSAGE. PREVIOUSLY, IT SPECIFIED USER 1 AND AN "IMMEDIATE" MESSAGE VIA NON-ZERO ARGUMENTS, WHICH REQUIRED THAT THE SENDING USER BE IN "-ACCEPT" MODE.

  BY USING THE NEW CALLING SEQUENCE, THE SENDER MAY BE IN "-DEFER" MODE OR "-ACCEPT" MODE. THEREFORE, THE SUBROUTINE ACPT$B NOW WILL

  ONLY RESET THE USER'S MESSAGE STATE TO "-DEFER" MODE IF IT WAS IN "-REJECT" MODE. THE MESSAGE OUTPUT BY ACPT$B HAS CHANGED CORRESPONDINGLY.

O THE INTERNAL SUBROUTINE SRCH$B HAS BEEN MODIFIED TO AUTOMATICALLY SET THE READ/WRITE LOCK OF A FILE THAT IT CREATES, BUT ONLY IF THE FILE IS NOT A TREENAME. SINCE BATCH NEVER REFERENCES ANY DATABASE FILES VIA TREENAME, THIS PREVENTS BATCH FROM SETTING THE READ/WRITE LOCK OF A USER-DEFINED FILE (SUCH AS A COPY OF BATDEF).

IT WILL ONLY SET THE READ/WRITE LOCK OF THE FILE WHEN IT CREATES IT. THIS MEANS THAT A CALL TO SRCH$B WITH A KEY OF K$RDWR (OR K$RDWR+K$NDAM TO CREATE A DAM FILE) WILL IN FACT CAUSE A CALL TO OPEN THE FILE FOR READING, FOLLOWED BY A CALL TO CHANGE ACCESS TO READING & WRITING VIA THE K$CACC KEY IF THE OPEN SUCCEEDS.

IF THE OPEN FAILS DUE TO NOT FOUND, THE FILE WILL BE OPENED AGAIN WITH THE ORIGINAL KEY. SINCE THE KEY ASKS FOR WRITE ACCESS, THE FILE WILL BE CREATED. SRCH$B WILL THEN CALL SATR$$ TO SET THE READ/WRITE LOCK OF THE FILE TO 1 (N READERS OR 1 WRITER).

AFTER DOING THAT SRCH$B WILL CLOSE THE FILE AND THEN REOPEN IT.

THIS GUARANTEES THAT NO OTHER PROCESS WAS SIMULTANEOUSLY OPENING THE FILE FOR READING & WRITING BEFORE THE READ/WRITE LOCK WAS SET.

O WHILE BATCH NEVER HAS CAUSE TO HAVE MULTIPLE PROCESSES OPENING A TEMPORARY FILE SIMULTANEOUSLY (SINCE TEMPORARY FILES ARE ONLY CREATED BY THE BATCH MONITOR AND THERE IS ONLY ONE SUCH MONITOR), TEMP$B, THE SUBROUTINE WHICH CREATES TEMPORARY FILES, HAS BEEN MODIFIED TO SET THE READ/WRITE LOCK OF A CREATED FILE IN A MANNER SIMILAR TO SRCH$B.

O THE SUBROUTINE COUT$B, WHICH CREATES AN INTERNAL OUTPUT FILE WHICH BATCH USES TO STORE A COPY OF THE COMMAND/CPL FILE, HAS BEEN MODIFIED TO SET THE READ/WRITE LOCK OF THE FILE TO 1 INITIALLY, IN A SIMILAR FASHION AS SRCH$B AND TEMP$B.

O THE C_RSET AND C_BDIF FILES IN BATCHQ HAVE BEEN CHANGED TO DELETE FILES IN BATCHQ WHICH ARE USED AS "FLAGS" OR "LOCKS". THESE FILES ARE LOCK.B (USED TO LOCK THE DATABASE), VALID. (VALIDATES THE DATABASE WHEN MONITOR IS RUNNING, LOCKS IT WHEN *FIXBAT IS RUNNING), MON.ST (VALIDATES MONITOR LOGOUT WHEN MONITOR IS NOT RUNNING, IS ALSO THE FLAG TO TELL RUNNING MONITOR TO SHUT DOWN), MON.GO (FLAG TO TELL MONITOR TO START UP), AND ERROR. (WHERE ERROR INFORMATION IS STORED). FOR LOCK.B, DELETING IT WILL FORCE ITS RE-CREATION BY REV 18.3 BATCH, WHICH WILL CAUSE ITS READ/WRITE LOCK TO BE SET TO 1 RATHER THAN LEFT AT 0. ERROR. IS RECREATED BY C_RSET AND C_BDIF AS A NULL FILE, SO THAT A FAILURE EVEN WHEN THE DISK IS FULL WILL BE LOGGED. VALID. IS RECREATED, BUT *FIXBAT WILL SET ITS READ/WRITE LOCK EVERYTIME IT RUNS. MON.ST IS RECREATED, AS IT VALIDATES VALID. WHEN IN.USE IS NOT IN USE. MON.GO IS NOT RECREATED.

ALSO, C_RSET AND C_BDIF WILL NOW EXPLICITLY SET THE READ/WRITE LOCK OF THE IN.USE FILE TO 1. THIS FILE IS OPEN BY THE MONITOR WHILE IT IS RUNNING, AND OTHER PARTS OF BATCH ATTEMPT TO DELETE IT TO DETERMINE IF THE MONITOR IS STILL RUNNING.

O THE *FIXBAT PROGRAM NOW SETS THE READ/WRITE LOCK OF BATCHQ>VALID. TO 1 WHEN IT RUNS (AS IT DOES NOT USE SRCH$B TO CREATE IT, IT MUST DO THE SETTING ITSELF), AND WILL ALSO SET THE BATCHQ>O_LOG READ/WRITE LOCK TO 2 WHEN IT RUNS (AND IF LOGGING IS TURNED ON). THAT WAY, USERS MAY READ O_LOG WHILE IT IS IN USE BY THE MONITOR. NOTE: THE LAST LINE OR TWO OUTPUT BY THE MONITOR ARE GENERALLY NOT READABLE IN O_LOG UNTIL THE FILE IS CLOSED BY THE MONITOR (BY LOGGING OUT).

O THE BATCH MONITOR WILL NOW RESET THE READ/WRITE LOCK OF THE BATDEF FILE TO 1 WHENEVER IT NOTICES THAT IT HAS STRAYED FROM THAT VALUE. WHENEVER IT DOES THIS, IT WILL SEND THE MESSAGE "CHANGING BATDEF RWLOCK FROM X TO 1." TO THE LOG FILE, WHERE <X> IS THE OLD READ/WRITE LOCK VALUE. WHEN IT FIRST RUNS, THE MONITOR WILL INITIALLY SET THE READ/WRITE LOCK OF BATDEF TO 1, BUT IT WILL OUTPUT NO MESSAGE CONCERNING THE OLD VALUE (IT DOES NOT BOTHER TO DETERMINE WHAT THE OLD VALUE IS).

THIS IS DONE BECAUSE WHEREAS BATCH WILL ALWAYS CREATE BATDEF WITH A READ/WRITE LOCK OF 1, A SYSTEM ADMINISTRATOR MAY USE FUTIL OR SOME SUCH UTILITY TO COPY A NEW VERSION OF BATDEF IN, PERHAPS RESETTING THE READ/WRITE LOCK VALUE.

O THE BATCH MONITOR NOW SETS THE READ/WRITE LOCK OF IN.USE WHENEVER IT STARTS UP, AS IT DELETES IT FIRST TO DETERMINE WHETHER A MONITOR IS ALREADY RUNNING.

THE FOLLOWING BUGS WERE FIXED IN REV 18.3 BATCH:

O THE "MULTIPLE MONITORS ILLEGAL" MESSAGE IS NOW RELIABLY SENT WHENEVER AN ATTEMPT IS MADE TO SPAWN ONE BATCH MONITOR WHILE ANOTHER IS RUNNING, NO MATTER WHAT THAT MONITOR IS DOING (RUNNING *FIXBAT OR *MONITR, OR EVEN IN BETWEEN).

O THE CHECKS IN INIT$B AND WAIT$B TO MAKE SURE THAT THE SYSTEM DATE/TIME ARE CORRECTLY SET HAVE BEEN UPDATED TO RECOGNIZE "00" (ASCII) AS AN ILLEGAL DATE. THIS WILL PREVENT FIXBAT FROM RUNNING BEFORE THE SYSTEM DATE AND TIME ARE SET. THIS WILL ALSO THEREFORE PREVENT THE MESSAGE "WAITING FOR BATCH SYSTEM -START" FROM BEING OUTPUT UNTIL SOMETIME AFTER THE DATE AND TIME ARE SET.

O JOB -CHANGE HAS BEEN FIXED AT REV 18.3. A LINE OF SOURCE CODE WAS SOMEHOW DELETED, PROBABLY AT REV 18.1, AND HAS BEEN REINSTALLED. UNTIL THIS FIX, JOB -CHANGE ON AN EXISTING (CHANGEABLE) JOB WOULD FAIL WITH A "UNIT NOT OPEN" OR "END OF FILE" ERROR.

O THROUGHOUT BATCH, PARTICULARLY IN THE JOB SUBMISSION, SPAWNING, AND ABORTION CODE, BATCH NOW USES THE PROTECTION KEYS <4 0> WHEN PROTECTING AN INTERNAL COMMAND FILE SO THAT IT MAY DELETE IT. PREVIOUSLY, IT USED <7 0>, WHICH LEFT A SMALL WINDOW DURING WHICH ANOTHER PROCESS MIGHT SUCCEED AT OPENING THAT FILE FOR READING AND/OR WRITING.

BY PROTECTING TO <4 0> INSTEAD, NO OTHER PROCESS WILL BE ABLE TO OPEN THE FILE, YET THE PROTECTING PROCESS WILL BE ABLE TO DELETE IT (THE 4 SIGNIFIES "DELETE" RIGHTS).

O THE BATCH SYSTEM -START/-STOP COMMANDS ARE FIXED TO OUTPUT REASONABLE ERROR MESSAGES IF THE MONITOR IS NOT STARTED. PREVIOUSLY, CONFUSING ERROR MESSAGES SUCH AS "ILLEGAL NAME" OR "NOT FOUND" WOULD RESULT IN THIS CASE.

O THE "HELD JOBS IN A QUEUE" BUG HAS BEEN FIXED. PREVIOUSLY, IF QUEUE A WAS DEFINED BEFORE QUEUE B, AND ALL JOBS IN QUEUE A WERE IN THE "HELD" STATE, NO JOBS IN QUEUE B WOULD EVER BE EXECUTED.

O A BUG EXISTED WHEREBY AN ATTEMPT TO RUN *MONITR FROM THE SYSTEM CONSOLE WOULD CAUSE GARBAGE TO BE OUTPUT. THIS HAS BEEN FIXED, SO THAT NOW THE CORRECT MESSAGE IS OUTPUT. THIS MESSAGE IS "CAN'T PROCESS BATCH JOBS FROM SYSTEM CONSOLE.".

O A BUG IN THE ACPT$B ROUTINE CAUSED THE ATTEMPT TO OUTPUT THE "I HAVE RESET YOUR MESSAGE STATE TO -ACCEPT" MESSAGE TO FAIL. THE MESSAGE WAS NOT IN THE CORRECT FORMAT FOR INTERNAL BATCH STRING MANIPULATION ROUTINES. THIS IS NOW FIXED. PLEASE NOTE THAT THE BEHAVIOUR OF ACPT$B, AND THEREFORE THE MESSAGE I AM REFERRING TO, WAS CHANGED AS A RESULT OF AN IMPROVEMENT LISTED ABOVE.

THE FOLLOWING MISCELLANEOUS CHANGES WERE MADE TO REV 18.3 BATCH:

O A NEW QUEUEING MECHANISM HAS BEEN IMPLEMENTED IN BATCH. IT ALLOWS JOB SUBMISSION AND ABORT HANDLING TO BE SIGNIFICANTLY FASTER ON SYSTEMS WHERE MANY BATCH JOBS MAY BE WAITING AT A TIME.

TO IMPLEMENT THIS MECHANISM, THE B$LIBF MODULES SNAP$B, RNXT$B, WNXT$B, WRIT$B, AND POSN$B WERE DELETED. THEY WERE REPLACED BY THE MODULES RDEN$B, ADEN$B, MDEN$B, DLEN$B, AND PHYS$B.

THESE MODULES ARE DESCRIBED IN GREATER DETAIL FOLLOWING THE BUG LIST IN THIS DOCUMENT.

THE $INSERT FILE B$QCOM, AND ITS COMMON AREA, WERE DELETED. THE FUNCTION OF THAT COMMON AREA IS NOW PROVIDED BY THE B$JOBS COMMON AREA, DECLARED IN THE $INSERT FILE B$JOBS.

O A NEW MODULE NAMED MOVE NOW EXISTS. IT IS IN PMA, AND IT SIMPLY

MOVES DATA FROM ONE BUFFER TO ANOTHER WORD-BY-WORD.

O A NEW COMMON AREA, TEMPJB, HAS BEEN CREATED. IT IS A MIRROR IMAGE OF B$JOBS, AND IT IS DECLARED IN THE $INSERT FILE B$JOBS. IT IS USED BY THE NEW QUEUEING MECHANISM FOR OPTIMIZATION PURPOSES.

WHEREAS THE LAST THREE WORDS OF B$JOBS ARE JCURR, JNEXT, AND JPREV, THE TEMPJB AREA DECLARES THEM AS TCURR, TNEXT, AND TPREV. NO OTHER STORAGE IN TEMPJB IS DECLARED SPECIFICALLY, AS IT IS ONLY USED FOR TEMPORARY COPIES OF B$JOBS.

O IN SOME PLACES, BATCH NOW USES THE PRIMOS SUBROUTINE RTIME$ TO TIME EVENTS, RATHER THAN USING TIMDAT AND ATTEMPTING TO RESOLVE ALL MIDNIGHT/NEW-YEAR WRAP-AROUND PROBLEMS.

O THE USER NAME IN THE "JOB XX ABORTED/COMPLETED" MESSAGE SENT TO THE SYSTEM CONSOLE NO LONGER HAS TRAILING BLANKS.

O WHEN THE BATCH MONITOR SPAWNS A JOB, IT WILL INTERNALLY LOOP 15 SECONDS TO MAKE SURE THE PHANTOM HAS IN FACT GRABBED THE PHANTOM FILE. AT REV 18.3, IT DOES THIS BY WAITING 1 SECOND 15 TIMES, INSTEAD OF WAITING 2.5 SECONDS 6 TIMES AS IT DID AT REV 18.1.

O CERTAIN TEXT SENT TO THE BATCH MONITOR LOG FILE HAS CHANGED. "++ABORTED:" IS NOW "++FINISHED:". THE "++EXECUTING" ETC. MESSAGE NOW HAS " USER #NN" APPENDED, TO IDENTIFY THE USER NUMBER OF THE SPAWNED PHANTOM. AND THE "++SPAWN JOBS." MESSAGE NOW READS "++SPAWN JOB.".

O THE UPDT$B MODULE, WHICH HAS NEVER BEEN USED BY BATCH, HAS BEEN REMOVED.

O ATCH$B, THE SUBROUTINE WHICH MANAGES ATTACH POINTS FOR BATCH, HAS A BUG FIX WHICH PREVENTS IT FROM DISABLING QUITS FOR BATCH JOBS WHICH ARE CPL FILE (VS. COMMAND FILE) SUBMISSIONS. AT REVISION 18, THIS BUG CAUSED NO PROBLEMS TO EITHER BATCH OR THE JOB WHICH WAS RUN; IT DID, HOWEVER, AT REV. 19, AND SO IT WAS ALSO FIXED AT REV. 18.3.

O THE SUBROUTINE QURD$B HAS BEEN MODIFIED TO ALLOW OLD BATDEF FILES TO BE READ IN. THE FORMAT OF BATDEF FILES IS UNCHANGED AT REV 18.3, ALTHOUGH THE REST OF THE DATABASE IS CHANGED AND MUST BE REINITIALIZED.

O ALL CALLS TO THE SUBROUTINE BILD$B HAVE BEEN FIXED SO THAT THEY ARE UNAMBIGUOUS. THERE WERE PREVIOUSLY MULTIPLE OCCURRENCES OF SIMILAR TEXTS SENT TO BILD$B, WHICH COULD MAKE DEBUGGING MORE DIFFICULT.

O IN THE BUILD FILE C_BATCH, AND ALSO THE LISTING BUILD FILE C_LIST, THE MODULES FIXBAT, JOB, AND MONITR ARE NOW COMPILED WITH THE -DEBASE OPTION, AND THE "AUTO" COMMANDS IN THE LOAD SEQUENCE FOR THE JOB COMMAND WERE REMOVED.

0 THE SRCH$B MODULE NOW WAITS UP TO 60 SECONDS FOR A FILE TO BECOME NOT "IN USE", NOT 30 AS IT DID PREVIOUSLY. THIS CAUSES THE MAXIMUM WAIT TIME FOR THE DATABASE TO UNLOCK TO ALSO EXTEND, SINCE SRCH$B IS USED BY THE LOCKING MECHANISM.

0 THE QCTR$B SUBROUTINE NO LONGER SUPPORTS A KEY OF K$EXST. THIS FUNCTION WAS USED BY JREQ$B, WHICH NO LONGER USES IT. QCTR$B WILL NOT CREATE A QUEUE CONTROL FILE IF IT DOES NOT EXIST, EVEN IF THE SUPPLIED KEY IS K$RDWR. TO CREATE A QUEUE CONTROL FILE, ONE MUST SET THE APPROPRIATE BIT IN THE B$QUEU COMMON AREA (SEE THE JOB SUBMISSION CODE IN JOB FOR AN EXAMPLE).

THE NEW QUEUEING MECHANISM INVOLVES FIVE NEW SUBROUTINES AND INCREASES THE MODULARITY OF THE QUEUE HANDLING THROUGHOUT BATCH.

PREVIOUSLY, ACTIVE ENTRIES IN A GIVEN QUEUE CONTROL FILE WERE FORWARD-THREADED, SO THAT INACTIVE ENTRIES (COMPLETED, ABORTED OR CANCELLED JOBS) WOULD NOT NEED TO BE READ IF NOT DESIRED.

THE FIRST ENTRY IN A QUEUE CONTROL FILE WAS CONSIDERED "ALWAYS ACTIVE" WHEN IT CAME TO BEING ON THE ACTIVE LIST, SINCE IT WAS NECESSARY TO KNOW WHERE THE LIST BEGAN; THE FIRST ENTRY IS A LOGICAL PLACE.

THE LAST TWO WORDS IN AN ACTIVE ENTRY HELD THE INTEGER*4 LOCATION OF THE NEXT ACTIVE ENTRY IN THE FILE. THIS LOCATION WAS THE VERY SAME THAT WOULD BE PASSED TO PRWF$$ IN A CALL USING THE K$PREA SUBKEY. IF AN ENTRY WAS THE LAST ACTIVE ENTRY IN THE FILE, IT WOULD POINT TO THE END OF THE FILE.

BECAUSE THE LIST WAS ONLY FORWARD-THREADED, MANY OPERATIONS WHICH LOGICALLY NEEDED SIMPLY TO POSITION TO THE DESIRED ENTRY WOULD HAVE TO TRAIPSE THE LIST. THIS WOULD BE DONE SO THAT WHEN THE DESIRED ENTRY WAS "SNAPPED OUT" OF THE LIST (I.E. MADE INACTIVE), THE ADDRESS OF THE PREVIOUS ACTIVE ENTRY WOULD BE KNOWN SO THAT ITS "NEXT ACTIVE ENTRY" POINTER COULD BE CHANGED.

EVEN IN PLACES WHERE THE LIST DID NOT NEED TO BE WALKED, BATCH WOULD WALK IT! IT MADE JOB SUBMISSION, ABORT HANDLING, ETC. RATHER INEFFICIENT. THOSE OPERATIONS WOULD BECOME GROSSLY INEFFICIENT WHEN HIGH NUMBERS OF ACTIVE JOBS (MORE THAN 500) WERE INVOLVED IN A SINGLE QUEUE CONTROL FILE. AT AROUND 2500 JOBS OR SO, THE BATCH SUBSYSTEM WOULD SIMPLY DIE BECAUSE THE AMOUNT OF TIME IT TOOK TO ADD OR DELETE AN ENTRY WOULD EXCEED THE OLD 30-SECOND LIMIT PLACED ON IT BY OTHER PROCESSES ATTEMPTING TO LOCK THE DATABASE.

NOW, THE LIST IS LINKED FORWARDS AND BACKWARDS. INSTEAD OF USING THE ONE INTEGER*4 POINTER (NAMED NXTJOB OR NEXT DEPENDING ON WHAT YOU HAD TO DO), THREE INTEGER*2 POINTERS EXIST IN EACH ENTRY.

THESE POINTERS ARE NAMED JCURR, JNEXT AND JPREV. THESE NEW POINTERS ARE ONLY INTEGER*2 BECAUSE THEY REFER TO ENTRY NUMBER, NOT ENTRY

POSITION. ENTRIES START WITH ENTRY NUMBER 0, THE ROOT ENTRY; THIS ENTRY IS STILL ALWAYS IN THE LINKED LIST.

JCURR IS THE ENTRY NUMBER OF THE CURRENT ENTRY. WHILE THIS MAY SEEM SUPERFLUOUS, IT IS NECESSARY FOR LOWER LEVELS OF BATCH TO KNOW WHICH ENTRY THEY ARE DEALING WITH, AND PUTTING THAT INFORMATION IN THE ENTRY ITSELF MADE LIFE VERY EASY.

JNEXT IS THE ENTRY NUMBER OF THE NEXT ACTIVE ENTRY IN THE LIST, OR IS 0 IF THERE ARE NO MORE ACTIVE ENTRIES; IN OTHER WORDS, THE FORWARD LIST POINTS BACK TO THE ROOT ENTRY.

JPREV IS THE ENTRY NUMBER OF THE PREVIOUS ACTIVE ENTRY IN THE LIST. THE ROOT ENTRY, ENTRY NUMBER 0, HAS A JPREV VALUE OF THE LAST ACTIVE ENTRY IN THE LIST. AGAIN, THIS MAKES THE BACKWARD POINTERS CIRCULAR.

A NEW SUBROUTINE, PHYS$B, CONVERTS AN ENTRY NUMBER TO A PHYSICAL ADDRESS IN THE FILE (AS USED IN PRWF$$). IT IS AN INTEGER*4 FUNCTION FOR EASE OF USE. IT IS CALLED WITH ONE ARGUMENT, THE INTEGER*2 ENTRY NUMBER. IT WILL SIMPLY MULTIPLY THE ENTRY NUMBER TIMES THE LENGTH OF AN ENTRY (DEFINED IN $INSERT FILE B$KEYS, PARAMETER B$JLEN), AND ADD 1 FOR THE VALIDATION WORD. IT WILL ALSO VALIDATE THE ENTRY NUMBER, MAKING SURE IT IS IN THE RANGE 0 THROUGH 32767. IT WILL RETURN THE RESULTING PHYSICAL ADDRESS AS ITS FUNCTION VALUE.

ALL ENTRY PROCESSING IS DONE IN THE COMMON AREA B$JOBS, WHICH IS WHERE JCURR, JNEXT AND JPREV RESIDE (REPLACING NXTJOB). AS OF REVISION 18.3, ANOTHER COMMON AREA NAMED TEMPJB IS DEFINED, WHICH IS THE SAME LENGTH AS B$JOBS. IT IS SIMPLY AN ARRAY WHICH IS (B$JLEN-3) WORDS LONG, AND ITS LAST 3 WORDS ARE NAMED TCURR, TNEXT AND TPREV; THEY ARE MIRROR IMAGES OF JCURR, JNEXT AND JPREV.

THE IDEA IS THAT TEMPJB MAY BE USED TO HOLD A TEMPORARY COPY OF B$JOBS WHILE THE B$JOBS ARRAY IS USED FOR READING AND WRITING ENTRIES IN THE QUEUE CONTROL FILE. TEMPJB SHOULD BE CAREFULLY USED; IF A SUBROUTINE MOVES DATA INTO IT, THEN CALLS ANOTHER SUBROUTINE WHICH USES IT, PROBLEMS WILL RESULT.

THE FOLLOWING SUBROUTINES HAVE BEEN ADDED TO BATCH TO IMPLEMENT THIS NEW QUEUEING MECHANISM:

RDEN$B(ENTRY,CODE) - READS ENTRY NUMBER <ENTRY> INTO B$JOBS FROM THE CURRENT QUEUE CONTROL FILE (OPEN ON UNIT COMQUN), RETURNS ERROR CODE IN <CODE>.

ADEN$B(CODE) - WRITES ENTRY IN <ENTRY> OUT TO END OF CURRENT QUEUE CONTROL FILE. WILL SET UP JCURR, JNEXT AND JPREV AS APPROPRIATE, BUT WILL RETURN WITH THE REST OF B$JOBS AS IT WAS WHEN THE SUBROUTINE WAS CALLED. USES TEMPJB TO KEEP THE ORIGINAL B$JOBS IN, WHILE IT USES RDEN$B AND MDEN$B TO READ AND WRITE THE FIRST AND LAST ACTIVE ENTRIES IN THE FILE TO REFLECT THE ADDITION OF THE NEW ENTRY.

MDEN$B(CODE) - WRITES THE CURRENT ENTRY IN B$JOBS BACK OUT TO THE CURRENT QUEUE CONTROL FILE.

DLEN$B(CODE) - DELETES THE CURRENT ENTRY IN B$JOBS FROM THE QUEUE CONTROL FILE BY SNAPPING IT OUT OF THE LINKED LIST. WILL USE TEMPJB TO KEEP THE ORIGINAL COPY OF B$JOBS IN WHILE IT USES B$JOBS TO UPDATE THE PREVIOUS AND NEXT ACTIVE ENTRIES TO REFLECT THE DELETION. IT WILL MOVE TEMPJB BACK INTO B$JOBS BEFORE RETURNING.

NOTE: THIS SUBROUTINE DOES NOT REFERENCE THE ENTRY ITSELF IN THE DISK FILE, SO IT CAN BE USED TO RECOVER FROM ERRORS CAUSED BY CREATING A NEW ENTRY TO CHANGE THE POINTERS BACK. ADEN$B USES DLEN$B FOR THIS VERY PURPOSE.

THE SUBROUTINE MOVE IS ADDED TO BE A FAST DATA-MOVER FOR BATCH. ITS CALLING SEQUENCE IS:

MOVE(SOURCE,DEST,WORDS) - WILL MOVE <WORDS> WORDS FROM <SOURCE> TO <DEST>.

THE USE OF MOVE IN BATCH IS PRIMARILY FOR MOVING COPIES OF B$JOBS AND TEMPJB BACK AND FORTH, AS FOLLOWS:

```
   CALL MOVE(USER,TEMPAR,B$JLEN)
   CALL MOVE(TEMPAR,USER,B$JLEN)
```

THE SUBROUTINE FIND$B HAS BEEN MODIFIED TO MAKE EFFICIENT USE OF TEMPJB. AFTER SEARCHING THE ENTIRE DATABASE, IF IT FOUND THE REQUESTED ENTRY, IT SIMPLY MOVES A COPY OF THAT ENTRY INTO B$JOBS AND THEN RETURNS. PREVIOUSLY, IT HAD TO OPEN THE QUEUE CONTROL FILE AGAIN, POSITION TO THE ENTRY (AND IT DID THIS THE SLOW WAY, WHICH WAS UNNECESSARY), AND READ IT IN.

THE COMMON AREA B$EXEC HAS BEEN CHANGED SO THAT BATCH MAY MAKE EFFICIENT USE OF THE DOUBLY-THREADED LINKED LIST NOW PRESENT IN THE QUEUE MANAGEMENT SYSTEM.

THE VARIABLES EXQNUM, EXPRI AND EXADDR HAVE BEEN ADDED. FOR A SPECIFIC EXECUTING JOB, THESE IDENTIFY THE JOB'S ENTRY IN TERMS OF QUEUE NUMBER, QUEUE PRIORITY, AND ITS ADDRESS IN THE FILE. BY HAVING THIS INFORMATION, THE ABORT PROCESSOR ABRT$B CAN FIND THE ENTRY IMMEDIATELY WHEN A JOB FINISHES. PREVIOUSLY, IT USED FIND$B TO ACCOMPLISH THIS TASK.

*
644 (BACKEND)

THIS IS AN INTERNAL TOOL -- NOT INTENDED FOR CUSTOMER USE.

*
645 (SYSOVL)

A NEW ERROR MESSAGE WAS ADDED TO CPL_ERR_TABLE.

*
646 (CRMPC)

SUBJECT: CRMPC

RELEASE: MASTER DISK RELEASE 18.3
PRIMOS 18.3
CRMPC 18.3

1. NEW FUNCTIONALITY
NONE

2. PROBLEMS FIXED

TAR 20729

THIS TAR DESCRIBES ERROR RECOVERY FOR CARD READERS AS BEING POOR TO NON-EXISTANT UNDER THE PROGRAM CRMPC. THE MAJOR PROBLEM HERE IS THAT WHILE ERROR MESSAGES ARE DISPLAYED ON A USER'S CONSOLE INDICATING THE NATURE OF ANY PROBLEMS, THE PROGRAM CONTINUES TO READ CARDS. THIS PREVENTS ANY ERROR RECOVERY FORCING ONE TO READ IN THE ENTIRE CARD DECK AGAIN.

THE SOLUTION TO THIS PROBLEM WAS TO MODIFY CRMPC AND THE CORRESPONDING PRIMOS DRIVER TO STOP READING CARDS AFTER ENCOUNTERING ANY UNUSUAL CONDITIONS. AS BEFORE THE PROGRAM WILL DISPLAY THE APPROPRIATE ERROR MESSAGE. UNLIKE BEFORE HOWEVER, A USER MAY CORRECT THE ERROR AND START THE PROGRAM FROM PRIMOS COMMAND LEVEL TO CONTINUE READING THE REMAINING CARDS.

3. OUTSTANDING PROBLEMS
   NONE

4. ENVIRONMENT
   CRMPC REQUIRES PRIMOS 18.3 TO PROPERLY FIX THE ABOVE PROBLEM.

5. INSTALLATION AND BUILD PROCEDURES
   FOLLOW THE STANDARD PRIME BUILD PROCEDURE

*
647 (DEREMER)

DEREMER HAS BEEN MODIFIED FOR REV 18.3.
DEREMER IS AN INTERNAL TOOL, IT IS NOT INTENDED FOR CUSTOMER USE.

*
648 (FUTIL)

AT REV 18.3, THE FROM, TO AND ATTACH COMMANDS IN FUTIL ARE FIXED TO SCAN FOR DISK VOLUMES BY NAME CORRECTLY. THEY WILL NOW DETECT BOTH STORAGE MODULE AND OTHER DISKS, AND YET WILL NOT ERRONEOUSLY USE THE FIRST DISK WHICH HAS A UFD NAMED THE SAME AS THE VOLUMENAME SPECIFIED.

*
649 (HELP)

```
HELP                                        PRINTS INFORMATION AT TERMINAL

    HELP [COMMAND-NAME]
```

IF HELP IS INVOKED WITH A COMMAND NAME, INFORMATION ABOUT THAT COMMAND IS PRINTED AT THE TERMINAL. IF HELP IS INVOKED WITHOUT A COMMAND NAME, IT PRINTS A LIST OF COMMANDS FOR WHICH INFORMATION IS AVAILABLE.

THE FORMAT OF COMMAND DESCRIPTION IS:

```
COMMAND_NAME                                          BRIEF DESCRIPTION

[ABBREVIATION - IF ANY]

    COMMAND LINE SYNTAX
```

TEXT DESCRIBING COMMAND, OPTIONS, ETC. AT THE END OF THE LISTING ARE

ANY REFERENCES TO FURTHER INFORMATION AND THE DATE THE HELP INFORMATION WAS CREATED OR UPDATED.

INFORMATION IS ALSO AVAILABLE FOR CERTAIN CATEGORIES OR GROUPINGS OF COMMANDS, SUCH AS COMPILERS, LOADERS, ETC.

JULY 1981

*
651 (MAGSAV/RST)

1 REV 18.0

THERE ARE NO CHANGES VISIBLE TO THE USER. THE COMMAND FILES HAVE BEEN CHANGED TO LOAD ROUTINES FROM SVCLIB AND NOT FTNLIB.

2 MAGSAV/MAGRST REV 18.1

TAR #21592/#28767

MAGRST HAS BEEN MODIFIED TO OVERCOME THE OFFLINE OR NOT READY MESSAGE AFTER BEING UNABLE TO READ A TAPE LABEL DUE TO TAPE ERRORS.

TAR #20333

MAGRST HAS ALSO BEEN MODIFIED TO GIVE A MORE MEANINGFUL MESSAGE WHEN IT CANNOT ATTACH TO PART OF A TREENAME BEING RESTORED (TAR #20333.)

THE NEW MESSAGES APPEAR AS THIS.

FOR SEG. DIRECTORIES :-

```
FILE ERROR IN SEGDIR.: <FILE-NAME>

OMITTING TREE-PATH:  <TREE-NAME>
```

FOR FILES AND UFDS WHICH CANNOT BE RESTORED BECAUSE THEIR PARENT UFD DID NOT ALLOW MAGRST TO ATTACH TO IT:-

```
BAD ATTACH ON:  <FILE-NAME>
```

OMITTING TREE-PATH: <TREE-NAME>

3 MAGSAV/RST FOR REV18.2

REMOTE DISK ATTACH PROBLEM.

ATTCH$ DID NOT OPERATE THE SAME OVER FAM AS LOCALLY WITH PARAMETER K$ALLD THIS REFERENCE HAS BEEN REMOVED.

IF MAGSAV CANNOT FIND A UFD IN A PATHNAME THEN MAGSAV WILL PRINT THE PRIMOS ERROR MESSAGE PLUS:-

'MAGSAV UNABLE TO CONTINUE' AND EXITS.

MAGSAV WILL NOW SAVE UP TO 18 LEVELS ON PRIMOS4, PRIMOS 2 WILL STILL HANDLE 13. IF MORE THAN THESE LEVELS ARE ATTEMPTED THEN, IT WILL GIVE THE MESSAGE - 'TOO MANY LEVELS' TREENAME, IGNORE THAT FILE AND CONTINUE BACK UP THE TREE.

4 MAGSAV/RST FOR REV 18.3

THERE ARE NO MAJOR DESIGN OR FUNCTIONALITY CHANGES IN THIS REV OF MAGSAV/RST. THERE ARE THOUGH SOME FIXES TO THE TARS AND POLERS RECIEVED.

4.1 POLER #28883

THIS PROBLEM AFFECTED BOTH MAGSAV AND MAGRST AND IT WAS FOUND THAT DUE TO TRUNCATION OR THE LACK OF IT IN SOME CASES FILES WERE HAVING THEIR DATA CORRUPTED. THIS WAS FIXED BY CHANGING THE PARAMETERS TO THE PRWF$$ CALLS. THE USER INTERFACE IS UNAFFECTED.

4.2 POLER #31022

THIS PROBLEM ONLY AFFECTED MAGSAV AND IT WAS FOUND THAT WHEN SAVING A NAMED ELEMENT, THE SAVE WOULD NOT STOP WHEN IT SHOULD HAVE IF A FILE OR UFD WHERE ADDED AT THE SAME LEVEL. THE USER INTERFACE IS UNAFFECTED.

4.3 RWLOCK PROBLEM

THIS PROBLEM ONLY AFFECTED MAGRST AND IT WAS FOUND THAT WHEN RESTORING A SEGMENT DIRECTORY, WITH NO SUB-FILES, AND NON DEFAULT RWLOCK, THE RWLOCK WOULD NOT BE SET PROPERLY. THE USER INTERFACE IS UNAFFECTED.

4.4 NEXT TAPE OUT OF SEQUENCE PROBLEM - POLER 37450

THIS PROBLEM ONLY AFFECTED MAGRST WHEN THE WRONG TAPE WAS LOADED IN A MULTI REEL RESTORE. IT NOW RECOVERS AND CONTINUES NORMALLY.

4.5 DISK FULL WHEN WRITING INDEX FILE TO DISK TAR 16026.

THIS PROBLEM AFFECTED MAGSAV AND MAGRST. IT NOW ALLOWS THE USER TO DELETE FILES AND CONTINUE WITH THE SAVE OR RESTORE.

4.6 NO LEVELS MESSAGE WAS INCORRECT .

HAVE FIXED THIS PROBLEM FOR MAGSAV IT DOES NOT APPLY TO MAGRST.

4.7 INCORRECT PASSWORD WITH $A COMMAND

IF ATTCH$ IS GIVEN AN INCORRECT PASSWORD THEN IT QUITS TO COMMAND LEVEL. AT THIS POINT NO END OF LOGICAL TAPE HAS BEEN WRITTEN AND THE USER WILL HAVE HAD DIFFICULTY IN GETTING HIS INFO BACK. MAGSAV NOW WRITES AN EOLT BEFORE PERFORMING A $A COMMAND AND UNDOES THIS IF THE ATTACH IS SUCCESFUL. THIS MEANS THAT IF A USER USES A BAD PASSWORD HE WILL BE ABLE TO RESTORE THE FILES HE HAS ALREADY SAVED ON TO THE TAPE. THE USER CAN THEN CONTINUE HIS SAVE FROM THE ABORTED ATTACH POINT ON A NEW LOGICAL TAPE BY INVOKING MAGSAV AGAIN EITHER ALREADY ATTACHED TO THE REQUIRED UFD OR BY USING THE $A COMMAND CORRECTLY.

*
652 (MAKE)

MAKE AY REV 18.3 HAS BEEN MODIFIED TO SUPPORT THE NEW 160 AND 600MB FIXED MEDIA DRIVES. BECAUSE THESE DRIVES HAVE MAXIMUM CYLINDER VALUES OTHER THAN

823 (THE 160MB PHYSICALLY HAS 823 CYLINDERS. HOWEVER, THE LAST 2 CYLINDERS ARE RESERVED FOR DIAGNOSTIC PURPOSIS.) CERTAIN RESTRICTIONS HAVE BEEN IMPOSED ON THE USE OF THESE DRIVES.

1) THESE DISKS MUST ONLY BE COPIED TO A DISK OF SIMILAR TYPE (160 -> 160, 600 -> 600). THIS IS TO ELIMINATE THE POSSIBILITY OF LOST DATA AND TO PRESERVE THE DIAGNOSTIC TRACKS.

2) AT REV. 19 BOTH THE 160MB AND THE 600MB DISKS MUST BE REMADE. THIS WILL ALLOW ACCESS TO THE REMAINING 18 CYLINDERS ON A 600MB DRIVE AND ALSO ENABLE THE USE OF A NEW BADSPOT HANDLING MECHANISM.

3) THE LAST PARTITION ON A 600 PARTITION MUST BE 10 HEADS, THIS IS IMPOSED DUE TO A DEFICENCY IN THE PDEV.

REFERENCE POLERS 29712

MAKE ALLOWS LOWERCASE PACKNAMES MAKING THE DSKRAT NAME UNREADABLE BY THE FILE SYSTEM. MAKE HAS BEEN FIXED TO UPCASE THE PACKNAME.

*

653 (PHYSAV/RST)

PHYSAV/RST REV 18.3
----

0 THE -TTY PARAMETER HAS BEEN ADDED TO THE COMMAND LINES FOR PHYSAV AND PHYRST.
IF THIS PARAMETER IS SPECIFIED, THE MAGNETIC TAPE UNIT NUMBER IS TAKEN FROM THE USER TERMINAL (EVEN IF THE CURRENT INPUT STREAM IS A COMMAND INPUT FILE).

0 LOGICAL TAPE NUMBER 0 (ZERO) IS NOW LEGAL. IT IS TAKEN TO MEAN THE CURRENT (OR NEXT) LOGICAL TAPE NUMBER.

0 POLER 32189 FIXED. PHYSAV NOW CORRECTLY HANDLES REN AFTER USER HAD BROKEN-IN WHEN ASKED 'WRITE NEXT LOG. TAPE (YES/NO)?'

0 PHYSAV CAN NOW WRITE MORE THAN ONE LOGICAL TAPE WHEN USED WITH AN INTEGRATED FORMATTER.

*

654 (PLP)

PLP HAS BEEN MODIFIED FOR 18.3. PLP IS AN INTERNAL TOOL, IT IT IS NOT INTENDED FOR CUSTOMER USE.

*
655 (PMA)

THIS DOCUMENT DESCRIBES THE CHANGES MADE TO PMA FOR PRIMOS RELEASE 18.3

1 PMA

THERE IS ONE BUG FIX TO PMA FOR REV 18.3. PMA WILL NOW CORRECTLY HANDLE SOURCE FILE NAMES WITH PASSWORDS IN THE DIRECTORY PORTION OF THE TREENAME.

2 P850 SUPPORT

THE FOLLOWING MNEMONIC OPCODES ARE NOW IMPLEMENTED:

```
          ENBL, ENBM, AND ENBP,
          INHL, INHM, AND INHP.
```

3 P300 SUPPORT

THE FRAC INSTRUCTION HAS BEEN REMOVED FROM PMA AT REV. 18.3. THE INSTRUCTION ONLY EXISTED ON THE P300 WITH SPECIAL HARDWARE AND WAS NEVER IMPLEMENTED ON THE P400, P500 OR THE 50 SERIRS. SUPPORT FOR THE P300 WAS DROPPED AT REV. 15.0.

*
656 (PRIMOS)

FOLLOWING IS A LIST OF ALL BUGS & TARS FIXED IN REV 18.3

POLER NUMBER: 32445,42687
FIXES A BUG WHERE XOFF IS NOT PROCESSED IN TIME TO AVOID BUFFER OVERRUN ON NEC SPINWRITER PRINTERS. ALSO INSURES THAT OUTPUT IS RESUMED IF XON/XOFF IS DISABLED WHILE OUTPUT WAS SUSPENDED.

FIX CODING ERROR IN FLOPPY DISK DRIVER THAT CRASHES SYSTEM IF A TIMEOUT OCCURS.

```
FIX A BUG IN MPC4 ASSIGN CODE WHICH COULD CAUSE THE SYSTEM TO HALT
WITH A LOCK PRIORITY VIOLATION.

CLEAN UP CODE AND ALLOW RINGO STACKS TO BE UNWIRED WHEN USERS ARE
LOGGED OUT.

MAKE &TTY WORK AS DOCUMENTED.  ADD &TTY_CONTINUE.
FIXED CPL SO A NOT TO "EXECUTE" COMMENTS SO SEVERITY$ NOT TRASHED.

IF USER ONE HAS COMO ON AND DOES AN ADDISK -RENAME THAT FAILS THE SYSTEM HALTS.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.14

      DESC:     FIX     BUG     IN     PRELOADER     BADSPOT     HANDLING.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.13

      DESC:         ADD            SEM$OU           AS          A          GATE.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.12

      DESC:  TO ALLOW INIT THE STATIC MODE LIB.  BITS IN THE RING 3

      DESC:          STACK,            FOR          EXTERNAL          COMMANDS.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.12

      DESC:     FIX       BUG       IN       &DO       ERROR       HANDLING.
      ------------------------------------------------------------------

      ORIGINAL TYPE FUNC REV NO:  18.3.12

      DESC:  TO ALLOW THE SYSTEM CONSOLE TO NOT BE ATTACHED TO CMDNCO

      DESC:  FOR LOGIN AND DBG.  I.E.  MAKE THEM BEHAMVE LIKE ALL THE

      DESC:        REST       OF       THE       EXTERNAL       COMMANDS.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.11

      DESC:                  PREVENT            SPURIOUS            NOTIFIES.
      ------------------------------------------------------------------

      ORIGINAL TYPE BUG REV NO:  18.3.11

      DESC:  INSURE PROCESS ABORTS WILL BE ENABLED BEFORE RETURNING TO

      DESC:                                             RING           3.
      ------------------------------------------------------------------
```

ORIGINAL TYPE BUG REV NO: 18.3.11

DESC: A SYSTEM RUNNING WITH TWO OR MORE OF THE OLD AMLC BOARDS,
DESC: IE.E, THE 5054 DMT BOARDS, WILL NOT COLD START. THE
DESC: SYSTEM HALTS AT LOCATION 6/121007, IN AMINIT.PMA

ORIGINAL TYPE BUG REV NO: 18.3.11

DESC: INSURE THAT PENDING PROCESS ABORTS ARE TAKEN.

ORIGINAL TYPE BUG REV NO: 18.3.10

DESC: TO MOVE CLDATA AWAY FROM STATIC MODE LIBRARY INIT BITS.

ORIGINAL TYPE BUG REV NO: 18.3.10

DESC: CORRECT BAD DCL IN GT$PAR STRUCTURE. FIX CPL BUG IN WHICH
DESC: &TTY DID NOT POP STATE OF CMD INPUT. CORRECT CPL BUG IN
DESC: WHICH A NON LOCAL GOTO OUT OF A CPL CONDITION HANDLER
DESC: INVOKED WHILE IN A CPL ROUTINE, COULD SOMETIMES NOT FIND
DESC: THE TARGET LABEL OF THE GOTO. FIX CPL EXPR EVALUATION
DESC: TO HANDLE CASE: WHERE V1 ENDS IN ' AND V2
DESC: IS TRUE-NULL. FIX BUG IN WHICH SUBSYS_ERR$ HANDLER
DESC: DOES'T FORCE SEVERITY CODE POSITIVE FOR STATIC MODE
DESC: PROGRAMS. FIX BUG IN WHICH IF THE KEEP_QUOTES BIT IN
DESC: KEYS OF CL$PIX IS SET, OPTION ARG FLAGS GET INITIALIZED
DESC: TO '1'B INSTEAD OF '0'B.

ORIGINAL TYPE TAR #20729 REV NO: 18.3.10

DESC: CORRECTED ERROR HANDLING LOGIC FOR CRMPC.

ORIGINAL TYPE FUNC REV NO: 18.3.10

DESC: MAKE AREA-MANAGEMENT PACKAGES ATOMIC.

```
-----------------------------------------------------------------------

ORIGINAL TYPE TAR #41507 REV NO:  18.3.10

DESC:  FIX COMMAND ENV.  BUG IN WHICH A NONLOCAL GOTO FROM A CPL

DESC:  ON-UNIT TO A "START <ADDRESS>" COMMAND MIS-THREADED THE

DESC:  STACK, CAUSING A FATAL$ ERROR.
-----------------------------------------------------------------------

ORIGINAL TYPE TAR # 44397 REV NO:  18.3.10

DESC:  FIX BUG IN P850 SLAVE STARTUP.
-----------------------------------------------------------------------

ORIGINAL TYPE TAR #36821 REV NO:  18.3.10

DESC:  FIX PROBLEM WITH FLOPPY DRIVES BEING SLOW.
-----------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.9

DESC:  TO ALLOW THE STATUS SEMAPHORE COMMAND TO WORK CORRECTLY.

DESC:  TO INSTALL CORRECT VERSION OF LIOCOM.INS.PLP;  MODIFIED

DESC:  WHEN SEMSEM REMOVED FROM LIOCOM IN TFLIO$.PMA
-----------------------------------------------------------------------

ORIGINAL TYPE FUNC REV NO:  18.3.9

DESC:  TO FIX SOFTWARE BUG WHICH CAUSED PROCESSES TO HANG ON NAMED

DESC:  SEMAPHORE MUTUAL EXCLUSION LOCK SEMSEM.

DESC:  TO STABALIZE EXISTING FUNCTIONALITY:  INCLUDING A CODE AUDIT

DESC:  AND EXTENSIVE REWRITIG;  HANDLE ERROR RETURN CLEANER.

DESC:  INTRODUCE NEW FUNCTIONALITY:  OPEN SEM.  WITH FILE OPEN ON

DESC:  FILE UNIT;  SEM.INITIALIZATION CAPABILITY.
-----------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.8

DESC:  1.  PRWF$$.FTN:  TO TAKE CARE OF POSKEYS (EACH POSKEY)

DESC:  CORRECTLY FOR THE REMOTE CASE.

DESC:  2.  SRCH$$.FTN:  ADD A CODE TO INCLUDE THE VMFA KEYS.
-----------------------------------------------------------------------
```

```
ORIGINAL TYPE FUNC REV NO:  18.3.8
DESC:  TO BRING THE FACILITIES NEGOTIATION AND PARSING FOR
DESC:  PRIMINET LEVEL III X.25 CLOSER TO THE 1980 SPECIFICATION.
-------------------------------------------------------------------
ORIGINAL TYPE BUG REV NO:  18.3.7
DESC:  PASS BACK SEVERITY ERROR CODE WHEN ERROR OCCURS WHILE
DESC:  ATTEMPTING TO ASSIGN MAGTAPE DRIVE.
-------------------------------------------------------------------
ORIGINAL TYPE TAR #29261 REV NO:  18.3.7
DESC:  MODIFY MAGTAPE ASSIGN LOGIC SO THAT AN ATTEMPT TO ASSIGN
DESC:  A MAGTAPE WITH THE CMPL 'ASSIGN M" WILL NOT BE SUCCESSFUL
DESC:  IF 'SETMOD -NOASSG' HAS BEEN ISSUED AT THE SYSTEM CONSOLE/
-------------------------------------------------------------------
ORIGINAL TYPE BUG REV NO:  18.3.7
DESC:  PEVENT AMLBUF FROM CHANGING THE SIZES OF THE REMOTE LOGIN
DESC:  TERMINAL I/O BUFFERS.
-------------------------------------------------------------------
ORIGINAL TYPE TAR #33062 REV NO:  18.3.7
DESC:  DO NOT ALLOW AMLC COMMAND TO OVERLAY REMOTE LOGIN BUFFERS.
-------------------------------------------------------------------
ORIGINAL TYPE TAR # 44399 REV NO:  18.3.7
DESC:  MODIFY THE SVC HANDLER FOR T$AMLC TO SUPPORT THE
DESC:  TWO EXTRA ARGUMENTS.
-------------------------------------------------------------------
ORIGINAL TYPE BUG REV NO:  18.3.7
DESC:  FIX BUG IN BSCMAN REGARDING ENQS.
-------------------------------------------------------------------
ORIGINAL TYPE TAR #'S 29686 + 29688 REV NO:  18.3.7
DESC:  A) MAKE A "MESSAGE" MESSAGE LESS AMBIGUOUS,
DESC:  B) BLOCK DEFERRED MSGS FOR -REJECT USER.
-------------------------------------------------------------------
```

```
ORIGINAL TYPE BUG REV NO:  18.3.7

DESC:  EASE   OF   USE.   FIX   ADDISK   BUGS   FOR   REMOTE   ADDISK.
----------------------------------------------------------------------

ORIGINAL TYPE:  BUG REV NO:  18.3.7

DESC:  CREATE  COMMAND  WITH  EXCESS  ARGS  DOESN'T  GET  AN  ERROR.
----------------------------------------------------------------------

ORIGINAL TYPE:  BUG REV NO:  18.3.7

DESC:  DON'T   ABORT   REMOTE   ATTACH   SCAN   ON   A   BAD   MFD.
----------------------------------------------------------------------

ORIGINAL TYPE TAR #37480 REV NO:  18.3.6

DESC:  RESET QUIT EVENT FLAG (QUITYP) BEFORE CREATING A USER'S

DESC:  R3 ENV.  THIS PEVENTS THE PROPAGATION OF PAST USER'S QUIT

DESC:                                                         EVENTS.
----------------------------------------------------------------------

ORIGINAL TYPE:  TAR #'S 45172 + 45376 REV NO:  18.3.6

DESC:  &SEVERITY &WARNING DID NOT RECOGNIZE VALUES OTHER THAN -1.

DESC:  DEFINE_GVAR CREATED FILE ANYWAY WHEN FULL PATHAME WAS NOT

DESC:  GIVEN FOR PASSWORDED UFD.

DESC:  ERRORS IN &DO HAD WRONG LINE # AND TEXT.

DESC:  &DATA DIDN'T WORK WITH NULL &THAN AND &ELSE CLAUSES.

DESC:  &ARGS  DIR  DID  NOT  COMPLAIN  ABOUT  GLOBAL  VAR  NAMES.
----------------------------------------------------------------------

ORIGINAL TYPE TAR #41441 REV NO:  18.3.6

DESC:  CALL TO  PHANT$  WITH  LARGE  LENGTH  ARGUMENT  CRASHES  SYSTEM.
----------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.6

DESC:  TO  STOP  FINDPROC  FROM  TRYING  TO  ACCESS  RING  0  STACK.
----------------------------------------------------------------------

ORIGINAL TYPE FUNC REV NO:  18.3.6

DESC:          ENHANCE          THE          UTILITY          USAGE.
----------------------------------------------------------------------
```

```
ORIGINAL TYPE BUG REV NO:  18.3.6

DESC:  TWO BUGS IN THE NPX SIGNAL MECHANISM WERE FIXED.  (1)

DESC:  CONDITION NAME DOESNOT GET PASSED TO THE MASTER (2) THE

DESC:  SLAVE GOES INTO A LOOP AND TAKES UP CPU TIME BECAUSE OF

DESC:  THE RECURSIVE SIGNAL CONDITIONS.

DESC:  WE NOW SUPPORT X.29 PARAMETER 14.

DESC:  THE VALUE OF X.29 PARAMETER 3 WAS CHANGED FROM 86 TO 126.

DESC:  THE PROBLEM OF THE NETWORK BOUNCING WHEN LOTS OF VC'S WERE

DESC:  CLEARED HAS BEEN FIXED.

DESC:  QUEUES ARE NOW CHECKED TO BE SURE THEY ARE LONG ENOUGH TO

DESC:    HOLD    THE    MAX    #    OF    ENTRIES    PUT    ON    THEN.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.6

DESC:  FIX PLPLIB TO SIGNAL STRINGRANGE CONDITION IN A COMPATIBLE

DESC:    WAY    TO    THE    PL/X    LIBRARY    SUPPORT    ROUTINE.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.5

DESC:  TO ADD F CONVERSIONS;  FIX PRINTING OF -2147483648;

DESC:  CAUSE ) TO BE LEGAL AND IGNORED;  ALLOW FOR

DESC:          (UNSIGNED          FIXED          BIN          (32)).
--------------------------------------------------------------------

ORIGINAL TYPE FUNC REV NO:  18.3.5

DESC:      ADD      DISK      METERING      INFO      INTO      USAGE.
--------------------------------------------------------------------

ORIGINAL TYPE TAR #33803 + 32160 REV NO:  18.3.5

DESC:  USRCM$.INS.PLP   IS   NOT   USED   SO   IT   HAS   BEEN   DELETED.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  FIXED BUG WHICH CAUSED MACHINE CHECKS UNDER CERTAIN
```

```
DESC:        CIRCUMSTANCES        WHEN        READING        CARDS.
-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  CORRECT  BUG  IN  CONTROLLER  VERIFICATION  LOGIC  FOR  MPC4.
-------------------------------------------------------------------

ORIGINAL TYPE TAR #45374 REV NO:  18.3.4

DESC:  ALLOW OUTPUT BUFFER TO EMPTY UPON LOGOUT IF DRPDTR CONFIG

DESC:                          DIRECTIVE           IS            SET.
-------------------------------------------------------------------

-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  IF USER ATTEMPTS TO UNASSIGN AN AMLC LINE WHICH HE DOES

DESC:  NOT HAVE ASSIGNED, RETURN ERROR CODE E$NASS, "NOT ASS

DESC:    RATHER     THAN      E$DVIU,      "DEVICE      IN      USE".
-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  FIX BUG TO ALLOW FOR ATTACHING TO A SUBUFD WITHOUT SETTING

DESC:                        THE           HOME          DIRECTORY.
-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:   TIMEOUT   WHILE   LOGGING   OUT   CAN   CRASH   SYSTEM.
-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  FIX A MICROCODE BUG ON P850 WHICH THE SLAVE MAY BE LEFT

DESC:  WITH  BAD  PARITY  ON  LOCATION  1000/1001  DURING  START  UP.
-------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4

DESC:  ALLOW STATUS COMMAND TO USE 3 CHAR.  FIELD WIDTH WHEN

DESC:    PRINTING     ASSIGNED      AMLC      LINE      NUMBERS.
-------------------------------------------------------------------

ORIGINAL TYPE TAR #32163 REV NO:  18.3.4
```

```
DESC:  MAKE SURE MACHINE STATE IS CORRECTLY SAVED AND RESTORED
DESC:  WHEN CHECKING DATASET STATUS.
DESC:  ALLOW XON TO BE RECOGNIZED WHEN ERROR CHECKING IS ENABLED
DESC:  (BUG #520, TAR #32163).

DESC:      PEVENT     BUFSEM     FROM      BEING      OVERNOTIFIED.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.4
DESC:  PROHIBIT PHANTOMS FROM ALTERING ASYNC LINES DATABASE VIA
DESC:                                              DUPLX$          CALL.
--------------------------------------------------------------------

ORIGINAL TYPE TAR #'S 32161 + 32162 REV NO:  18.3.3
DESC:  TO  FIX  A  COUPLE   OF   MINOR   BUGS   IN   MESSAGE   COMMAND.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.3
DESC:  FIX EMBEDDED SYNC BUG.  FIX PA1-PA3 UPDATE OF VBE.  CHANGE
DESC:  REV NUMBERS FOR DPTX.  FIX DE HANDLING FOR TSF.  ADDED
DESC:  LOGOUT  CLEAN  UP  FOR  BSCMAN.   FIX  BAD  DIM  DATA   PROBLEM/
--------------------------------------------------------------------

ORIGINAL TYPE FUNC REV NO:  18.3.2
DESC:  ADD THE GENERAL PURPOSE PARALLEL INTERFACE (T$GPPI) SUPPORT
DESC:  TO PRIMOS AS DEFINED IN PE-T-739.  THIS INTERFACE IS
DESC:    CURRENTLY    USED    TO    DRIVE    THE    MPC4    CONTROLLER.
--------------------------------------------------------------------

ORIGINAL TYPE BUG REV NO:  18.3.2
DESC:  CORRECT REV 18 CODES TO CHECK THE SIZE OF THE BUFFER AND
DESC:  IF  IT  EXCEEDS  4K  WORDS,  SEND  TO  THE  CALLER  THE   ERROR.
--------------------------------------------------------------------

ORIGINAL TYPE FUNC REV NO:  18.3.1
DESC:      ADD      HELP      COMMAND      TO      REV.        18.
--------------------------------------------------------------------
```

```
ORIGINAL TYPE: TAR #29087 REV NO: 18.3.1

DESC: CHECK USER 1'S MSG BUFFER INSTEAD OF OUTPUT BUFFER.

DESC: PROVIDES A MORE CONSISTENT CHECK FOR BUFFER OVERFLOW.
-----------------------------------------------------------------

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX BUG IN T$MG WHICH CAUSED SYSTEM CRASHES IN CASE OF

DESC: ERROR ASSIGNING A MEGATEK DEVICE: LOCK PRIORITY VIOLATIONS

DESC: WERE CAUSED BECAUSE DEVLCK WAS NOT RELEASED.
-----------------------------------------------------------------

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX CNAME COMMAND TO PRINT THE NEW NAME IN THE ERROR

DESC: MESSAGE IF THE NEW NAME IS ILLEGAL.
-----------------------------------------------------------------

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX GARBAGE-CHARACTER-IN-LISTF-HEADER-BUG.
-----------------------------------------------------------------

ORIGINAL TYPE TAR #30221 REV NO: 18.3.0

DESC: FIX BUG IN SETTIME COMMAND, WHICH WILL ACCEPT ILLEGAL TIMES.
-----------------------------------------------------------------

ORIGINAL TYPE TAR #27063 REV NO: 18.3.0

DESC: PREVENT SYSTEM CRASH WHEN TYPING 'DISK NOT' COMMAND AT THE

DESC: SYSTEM CONSOLE WHILE THE COMOUTPUT IS ON.
-----------------------------------------------------------------

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: MAKE SURE NOT TO OVER NOTIFY BUFSEM UPON RECEIPT OF A QUIT.
-----------------------------------------------------------------

ORIGINAL TYPE TAR #10520 REV NO: 18.3.0

DESC: MAKE SURE A FORCE WRITE TO AN NR/NW FILE HAPPENS PROPERLY.
-----------------------------------------------------------------

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX REMOTE TOTALS BUG;
```

DESC: FIX TO ZERO-FILL (NOT SPACE-FILL)

DESC: OVERLONG DIR-ENTRIES.

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX PATHNAME CF TO HANDLE QUOTED ARGS CORRECTLY.

DESC: FIX PATHNAME CF TO HANDLE MFD-LVEL OBJECTS CORRECTLY.

DESC: FIX BUG IN DIR CF RELATED TO #2 ABOVE.

ORIGINAL TYPE BUG REV NO: 18.3.0

DESC: FIX BUG IN COMPUTING CHAR(X) WHERE X=-2**31: LENGTH (RESULT)

DESC: NOT SET.

1 MPC4 SUPPORT - A GENERAL PURPOSE PARALLEL INTERFACE

THE MPC4'S MICROCODE MAY EXIST IN EITHER RAM (RANDOM-ACCESS MEMORY) OR ROM (READ-ONLY MEMORY) DEPENDING ON THE MPC4 MODEL IN USE. IF RAM IS USED, MICROCODE IS LOADED FROM A PRIMOS DISK FILE AT SYSTEM COLD-START.

1.1 OVERVIEW AND BACKGROUND

SUPPORTING A NEW I/O DEVICE IN PRIMOS TYPICALLY INVOLVES SPECIFYING (AT LEAST) THREE INTERFACES: DEVICE TO CONTROLLER, CONTROLLER TO PRIMOS DRIVER, AND PRIMOS TO USER (APPLICATIONS PROGRAMMER). ULTIMATELY, THIS LEADS TO BUILDING AND SUPPORTING A NEW OR SPECIALIZED VERSION OF PRIMOS. THERE IS, HOWEVER, A LARGE CLASS OF DEVICES THAT HAVE RELATIVELY STRAIGHTFORWARD SUPPORT REQUIREMENTS AND FOR WHICH IT MAKES GREAT SENSE TO PROVIDE A STANDARD INTERFACING MECHANISM FOR.

THE INTENT AND SPECIFICATION OF THE GENERAL PURPOSE PARALLEL INTERFACE (GPPI), WHICH IS INITIALLY REALIZED AS AN MPC4 BOARD, IS TO REMOVE PRIMOS FROM THE DEVICE-DEPENDENT CONSIDERATIONS OF NEW CONTROLLERS BY ' DEFINING A PIO/DMX INTERFACE OF BROAD APPLICABILITY. THIS ALLOWS A SINGLE PRIMOS DRIVER -- T$GPPI -- TO SUPPORT THIS INTERFACE. ANY NEW CONTROLLER THAT CONFORMS TO THIS SPECIFICATION WILL AUTOMATICALLY HAVE PRIMOS SUPPORT. (IN THIS CHAPTER, THE TERMS GPPI AND MPC4 ARE USED INTERCHANGEABLY SINCE THE MPC4 IS THE HARDWARE IMPLEMENTATION OF THE GPPI CONCEPT.)

OBVIOUSLY, SUPPORT FOR A NEW DEVICE WILL STILL REQUIRE THE IMPLEMENTATION OF DEVICE-DEPENDENT CODE AT SOME LEVEL. THIS CODE MAY NOW BE IN THE FORM OF USER (RING 3) ROUTINES THAT CALL T$GPPI AND WILL BE INDEPENDENT OF PRIMOS RELEASES. THE DEVICE SUPPORT WILL APPEAR TO AN APPLICATIONS PROGRAM JUST AS IF IT WERE A DRIVER IMPLEMENTED ENTIRELY IN PRIMOS.

THROUGHOUT THIS CHAPTER, THE READER MAY BE REFERRED TO THE DOCUMENTATION ON A "PARTICULAR GPPI IMPLEMENTATION". (THE QUOTED PHRASE REFERS TO USING STANDARD OFF-THE-SHELF MPC4 HARDWARE WITH CUSTOM MICROCODE FOR INTERFACING SOME DEVICE.) SINCE THIS CHAPTER DEFINES THE T$GPPI SOFTWARE INTERFACE FOR THE GENERAL CASE, CERTAIN BEHAVIOR CAN'T BE DESCRIBED. BEAR IN MIND THAT SOME FEATURES OF THE GPPI MAY OR MAY NOT BE USED WHEN INTERFACING A PARTICULAR DEVICE. ALSO, SOME USER-VISIBLE DECOR MAY BE STRICTLY DUE TO THE CHARACTERISTICS OF THE DEVICE OR DUE TO THE DESIGN OF THE CONTROLLING MICROCODE. THEREFORE, THE DESIGN SPECIFICATION FOR THE "PARTICULAR IMPLEMENTATION" MUST BE CONSULTED.

PRIMOS SUPPORT IS BUILT TO HANDLE A MAXIMUM OF TWO MPC4 CONTROLLERS AT DEVICE CODES '75 AND '76.

1.2 ASSIGN/UNASSIGN LOGIC FOR GPPI DEVICES

THIS SECTION DESCRIBES HOW GPPI DEVICES ARE LOGICALLY ALLOCATED TO A USER.

1.2.1 NORMAL OPERATION

A GPPI-CLASS DEVICE MUST FIRST BE ASSIGNED TO THE USER VIA AN ASSIGN COMMAND. AT COMMAND LEVEL, THE ASSIGN COMMAND IS USED:

```
AS GPN
```

WHERE N=0, 1, ..., 7. SINCE THE GPPI MAY LOGICALLY HANDLE UP TO FOUR DEVICES, N=0,1,2, OR 3 WILL ASSIGN THE APPROPRIATE DEVICE ON THE FIRST GPPI INTERFACE (WHICH IS AT DEVICE CODE '75) AND N=4,5,6,OR 7 WILL ASSIGN THE APPROPRIATE DEVICE ON THE SECOND GPPI INTERFACE (WHICH IS AT DEVICE CODE '76).

TO RELEASE AN ASSIGNED DEVICE AT COMMAND LEVEL, THE UNASSIGN COMMAND IS USED:

```
UN GPN
```

1.2.2 ABNORMAL CONDITIONS

IT'S POSSIBLE THAT AN ASSIGN COMMAND WILL NOT BE SUCCESSFUL. THIS CAN HAPPEN WHEN A USER TRIES TO ASSIGN A GPPI DEVICE AND THE SYSTEM HAS DISCOVERED SOME UNRECOVERABLE ERROR CONDITION. IN THIS CASE, THE USER WILL RECEIVE ONE OF THE FOLLOWING MESSAGES:

MPC4 CONTROLLER DID NOT RESPOND.

DURING SYSTEM COLD START, THE CONTROLLER WAS NOT FOUND AT THE EXPECTED DEVICE ADDRESS. IT IS THEREFORE CONSIDERED NON-EXISTENT (OR BROKEN) BY THE SYSTEM.

MPC4 MICROCODE NOT LOADED.

DURING SYSTEM COLD START, THE CONTROLLER COULD NOT BE LOADED WITH ITS CORRESPONDING MICROCODE EITHER DUE TO THE NON-EXISTENCE OF THE MICROCODE FILE IN THE UFD "SYSTEM", OR DUE TO AN ERROR IN ATTEMPTING TO LOAD THE FILE INTO THE CONTROLLER.

INSUFFICIENT SEG 0 WINDOWS FOR MPC4.

DURING SYSTEM COLD START, THE OPERATING SYSTEM COULD NOT ALLOCATE ENOUGH SEGMENT ZERO WINDOW SLOTS FOR THIS CONTROLLER. THIS ERROR IS AN OPERATING SYSTEM PROBLEM AND SHOULD BE REPORTED AS SUCH.

MPC4 CONTROLLER DID NOT VERIFY.

AFTER THE MICROCODE WAS LOADED AND THE CONTROLLER WAS STARTED, THE MICROCODE VERIFICATION ROUTINE DID NOT COMPLETE SATISFACTORILY. THIS COULD BE DUE TO A DAMAGED MICROCODE FILE OR MALFUNCTIONING HARDWARE.

1.3 T$GPPI -- DEVICE DRIVER FOR GPPI CONTROLLERS

T$GPPI IS THE DEVICE DRIVER FOR ALL GPPI-CLASS CONTROLLERS. IT PROVIDES A COMMON INTERFACE FOR DEVICE-DEPENDENT LOGIC RESIDING IN NON-RING 0 DEVICE DRIVERS. THIS SECTION DESCRIBES HOW TO CALL T$GPPI.

1.3.1 GENERAL CALLING SEQUENCE

CALLING SEQUENCE: CALL T$GPPI(UNIT,KEY,DATA1,DATA2,ARRAY,CODE)

ALL ARGUMENTS ARE ONE WORD INTEGERS WITH THE EXCEPTION OF ARRAY WHICH IS AN ARRAY OF ONE WORD INTEGERS.

UNIT THE LOGICAL UNIT NUMBER OF THE DISPLAY. UNIT MUST BE IN

THE RANGE 0 TO 7, CORRESPONDING TO THE ASSIGNABLE DEVICES GP0 THROUGH GP7.

KEY A VARIABLE INDICATING THE FUNCTION TO BE PERFORMED AS DESCRIBED IN THE FOLLOWING SECTIONS:

1 - READ BLOCK

2 - WRITE BLOCK

3 - READ WORD

4 - WRITE WORD

5 - WAIT/POLL FOR INTERRUPT

6 - LOAD INTERRUPT MASK REGISTER

7 - LOAD COMMUNICATION REGION ADDRESS REGISTER

8 - EXECUTE DEVICE-DEPENDENT OTA

9 - RESET DEVICE

10 - LOAD DEVICE TIMEOUT REGISTER

11 - RELEASE COMMUNICATION REGION

:100001 - EXECUTE OCP. (RESTRICTED)

:100002 - EXECUTE SKS. (RESTRICTED)

:100003 - EXECUTE INA. (RESTRICTED)

:100004 - EXECUTE OTA. (RESTRICTED)

DATA1 A VARIABLE WHOSE VALUE DEPENDS UPON KEY. (SEE NEXT SECTION)

DATA2 A VARIABLE WHOSE VALUE DEPENDS UPON KEY. (SEE NEXT SECTION)

ARRAY FOR READ BLOCK, WRITE BLOCK, AND WAIT/POLL KEYS, ARRAY IS AN ARRAY CONTAINING DATA TO BE WRITTEN OR INTO WHICH DATA IS TO BE READ. FOR OTHER KEYS, ARRAY IS IGNORED. (THE ARGUMENT MUST BE PRESENT IN THE CALL, HOWEVER.)

CODE A STANDARD RETURN CODE AS FOLLOWS:

0 FUNCTION PERFORMED (OR STARTED) WITH NO ERRORS.

E$BKEY AN INVALID KEY WAS SPECIFIED.

E$BPAR AN INVALID PARAMETER WAS SPECIFIED.

E$BUNT AN INVALID UNIT NUMBER WAS SPECIFIED.

E$DNSK A PIO OPERATION (SKS, INA, OTA) DID NOT SKIP.

E$EXST THE COMMUNICATION REGION ALREADY EXISTS.

E$IEDI AN I/O ERROR OR DEVICE INTERRUPT HAS OCCURRED.

E$NASS THE UNIT HAS NOT BEEN ASSIGNED BY THE USER.

E$NRIT A PIO OPERATION WAS ATTEMPTED BY A USER NOT AT THE SYSTEM CONSOLE.

E$ROOM NO SEGMENT ZERO WINDOW SLOTS ARE AVAILABLE FOR USE.

E$WMST A SYSTEM WARM-START HAS OCCURRED.

1.3.2 DESCRIPTION OF FUNCTIONS

IN THE FOLLOWING CALLING SEQUENCES, 'XX' INDICATES A PARAMETER THAT MUST BE PRESENT IN THE CALL BUT WHOSE VALUE IS IGNORED. (A ZERO MAY BE SPECIFIED FOR CONVENIENCE.) ALSO, DATA1 AND DATA2

ARGUMENTS AS DEFINED IN THE PREVIOUS SECTION ARE REPLACED WITH NAMES THAT MORE CLOSELY DEPICT THEIR FUNCTION WITH RESPECT TO KEY.

1.3.2.1 READ BLOCK

CALLING SEQUENCE: CALL T$GPPI(UNIT,1,NWORDS,XX,ARRAY,CODE)

NWORDS 16-BIT WORDS ARE READ FROM THE DEVICE SPECIFIED BY UNIT AND PLACED IN ARRAY.

NOTES:

THE MAXIMUM NUMBER OF WORDS THAT CAN BE READ IS 4096.

1.3.2.2 WRITE BLOCK

CALLING SEQUENCE: CALL T$GPPI(UNIT,2,NWORDS,XX,ARRAY,CODE)

NWORDS 16-BIT WORDS ARE READ FROM ARRAY AND OUTPUT TO THE DEVICE SPECIFIED BY UNIT.

NOTES:

THE MAXIMUM NUMBER OF WORDS THAT CAN BE WRITTEN IS 4096.

1.3.2.3 READ WORD

CALLING SEQUENCE: CALL T$GPPI(UNIT,3,DATA,SOURCE,XX,CODE)

A SINGLE 16-BIT WORD OF DATA IS READ FROM THE CONTROLLER OR DEVICE SPECIFIED BY UNIT AND PLACED IN DATA. SOURCE IS AN INTEGER THAT IDENTIFIES THE SOURCE FROM WHICH THE DATA IS TO BE READ AS FOLLOWS:

SOURCE	SOURCE OF DATA
0	WORD OF DATA READ FROM DEVICE INTERFACE
1	WORD OF STATUS READ FROM DEVICE INTERFACE
2	INTERRUPT FLAG REGISTER (REGISTER IS THEN ZEROED)

3	MICROCODE IDENTIFICATION REGISTER
4	PIO ERROR REGISTER
5	RESERVED
6	RESERVED
7	RESERVED
8	RESERVED
9	RESERVED
10	RESERVED
11	TIMEOUT REGISTER
12	DMX CHANNEL ADDRESS REGISTER
13	INTERRUPT MASK REGISTER
14	INTERRUPT VECTOR REGISTER
15	DEVICE SELECTION REGISTER
16-31	DEVICE-DEPENDENT INFORMATION

NOTES:

THE READ WORD FUNCTION CAUSES T$GPPI TO EXECUTE AN OTA 00/INA 00 SEQUENCE.

1.3.2.4 WRITE WORD

CALLING SEQUENCE: CALL T$GPPI(UNIT,4,DATA,XX,XX,CODE)

A SINGLE WORD OF DATA IS OUTPUT FROM DATA TO THE DEVICE SPECIFIED BY UNIT.

NOTES:

THE WRITE WORD FUNCTION CAUSES T$GPPI TO EXECUTE AN OTA 01.

1.3.2.5 WAIT/POLL FOR INTERRUPT

CALLING SEQUENCE: CALL T$GPPI(UNIT,5,TIMLIM,XX,STATUS,CODE)

T$GPPI CHECKS TO SEE IF ONE OR MORE INTERRUPTS HAVE BEEN RECEIVED FROM THE DEVICE SPECIFIED BY UNIT SINCE THE LAST TIME THIS FUNCTION WAS CALLED. IF THERE ARE OUTSTANDING INTERRUPTS, UP TO TWO WORDS OF STATUS INFORMATION ARE RETURNED IN STATUS. (THEREFORE, STATUS NEED ONLY BE A TWO-ELEMENT ARRAY IN THIS CALL.) IF THERE ARE NO OUTSTANDING INTERRUPTS WHEN THE WAIT/POLL REQUEST IS ISSUED, THE ACTION TAKEN BY T$GPPI IS DETERMINED BY THE VALUE OF TIMLIM AS FOLLOWS:

TIMLIM < 0 T$GPPI WILL NOT RETURN TO THE USER PROGRAM UNTIL AN INTERRUPT HAS BEEN RECEIVED. WHEN AN INTERRUPT

OCCURS, INFORMATION WILL BE RETURNED IN STATUS AS DESCRIBED BELOW. (DURING THIS WAIT, QUITS, FORCED LOGOUTS, AND OTHER PROCESS FAULTS MAY OCCUR.)

TIMLIM = 0 T$GPPI WILL RETURN IMMEDIATELY. IF THERE IS AN OUTSTANDING INTERRUPT, INFORMATION WILL BE RETURNED AS DESCRIBED BELOW. IF THERE IS NO INTERRUPT, STATUS(1) AND STATUS(2) WILL BE SET TO ZEROES.

TIMLIM > 0 T$GPPI WILL RETURN IMMEDIATELY IF THERE IS AN INTERRUPT OUTSTANDING WITH STATUS SET AS DESCRIBED BELOW. IF THERE IS NO OUTSTANDING INTERRUPT, T$GPPI WILL WAIT FOR AN INTERRUPT TO OCCUR. AS SOON AS IT OCCURS, T$GPPI WILL RETURN WITH STATUS SET AS DESCRIBED BELOW. HOWEVER, IF AN INTERRUPT DOES NOT OCCUR WITHIN TIMLIM TENTHS OF A SECOND, T$GPPI RETURNS WITH STATUS SET TO ZEROES.

1.3.2.5.1 CONTENTS OF STATUS

THE FIRST WORD OF STATUS INFORMATION, STATUS(1), CONTAINS THE ACCUMULATED CONTENTS OF THE INTERRUPT FLAG REGISTER READ FROM THE GPPI. EACH TIME AN INTERRUPT OCCURS, THE CONTENTS OF THIS REGISTER IS LOGICALLY OR'ED INTO THE COPY KEPT BY T$GPPI. THIS ACCUMULATED COPY IS RETURNED TO THE USER AS STATUS(1). THE COPY IS THEN ZEROED AS A RESULT OF THIS CALL SO THAT THE USER DOESN'T RECEIVE DUPLICATE INFORMATION. THE BIT DEFINITIONS IN THIS WORD ARE EXACTLY AS DEFINED IN THE GPPI SPECIFICATION. ITS DESCRIPTION IS REPEATED HERE FOR CONVENIENCE.

BIT MEANING WHEN SET

1 END OF RANGE. END OF RANGE HAS OCCURRED ON THE LAST CHAIN OF A DMA/DMC I/O OPERATION OR A DMA/DMC TRANSFER HAS TERMINATED PREMATURELY DUE TO AN ERROR CONDITION. IN THE LATTER CASE, BIT 2 WILL ALSO BE SET.

2 DMX ERROR. A DMX TRANSFER WAS PREMATURELY TERMINATED DUE TO AN ERROR NOTICED BY THE GPPI (BAD PARITY, INCORRECT DMX SETUP, ETC.).

3 DEVICE INTERFACE ERROR. A DEVICE INTERFACE SEQUENCE WAS ABORTED DUE TO AN INCORRECT RESPONSE FROM THE EXTERNAL DEVICE.

4 TIMEOUT. THE EXTERNAL DEVICE FAILED TO RESPOND TO AN INTERFACE SEQUENCE WITHIN THE TIME ALOTTED TO IT BY THE VALUE IN THE DEVICE TIMEOUT REGISTER.

5 INTERNAL FAILURE. THE GPPI'S INTERNAL CHECKING LOGIC HAS ENCOUNTERED AN INCONSISTENCY. FURTHER OPERATIONS SHOULD NOT BE ATTEMPTED. HARDWARE DIAGNOSIS IS PROBABLY WARRANTED.

6 PIO ERROR. AN INVALID PIO SEQUENCE HAS BEEN ISSUED (E.G., ATTEMPTING TO START DMX ACTIVITY BEFORE LOADING THE DMX CHANNEL ADDRESS REGISTER). PIO ERRORS ARE SIGNALED BY THE GPPI, NOT THE EXTERNAL DEVICE(S) AND SHOULD BE TREATED AS DRIVER (AS OPPOSED TO APPLICATIONS PROGRAM) RELATED LOGIC ERRORS.

7-8 RESERVED.

9 DEVICE INTERRUPT-1. THE FIRST EXTERNAL DEVICE HAS ASSERTED AN INTERFACE LINE DEFINED TO GENERATE AN INTERRUPT OF THE USER PROGRAM.

10 DEVICE INTERRUPT-2. THIS BIT IS AVAILABLE FOR DEVICES THAT MAY DEFINE TWO INTERRUPTS. ITS OPERATION IS THE SAME AS DEVICE INTERRUPT-1.

11-12 TWO INTERRUPT FLAGS FOR THE SECOND DEVICE (IF ANY).

13-14 TWO INTERRUPT FLAGS FOR THE THIRD DEVICE (IF ANY).

15-16 TWO INTERRUPT FLAGS FOR THE FOURTH DEVICE (IF ANY).

THE SECOND WORD OF INTERRUPT INFORMATION, STATUS(2), IS RETURNED ONLY IF A DEVICE INTERRUPT HAS BEEN RECEIVED FROM THE DEVICE SPECIFIED BY UNIT. IN THIS CASE, T$GPPI PERFORMS A READ STATUS FUNCTION TO THE SPECIFIED DEVICE AND PLACES THE RESULTING STATUS WORD IN STATUS(2). (OTHER DEVICE INTERRUPT FLAGS (FOR OTHER UNIT'S) MAY BE SET, BUT THEY WILL NOT CAUSE STATUS TO BE READ.) THE CONTENTS OF THE DEVICE STATUS WORD ARE DEFINED IN THE SPECIFICATIONS OF THE PARTICULAR GPPI IMPLEMENTATION.

1.3.2.6 LOAD INTERRUPT MASK REGISTER

CALLING SEQUENCE: CALL T$GPPI(UNIT,6,MASK,XX,XX,CODE)

THE INTERRUPT MASK REGISTER OF THE CONTROLLER FOR THE DEVICE SPECIFIED BY UNIT IS MODIFIED ACCORDING TO THE CONTENTS OF MASK. THE CONTENTS OF MASK ARE AS FOLLOWS:

```
0 000 000 000 0AB
```

BITS A AND B ARE USED TO MASK DEVICE INTERRUPTS 1 AND 2 RESPECTIVELY (SEE PE-T-738); A .ONE BIT ENABLES THE CORRESPONDING INTERRUPT.

NOTES:

THE CONTENTS OF MASK ARE SHIFTED TO BE IN THE APPROPRIATE POSITION IN THE GPPI'S INTERRUPT MASK REGISTER, ORED WITH THE CURRENT CONTENTS OF THE MASK REGISTER, AND OUTPUT TO THE GPPI WITH AN OTA 15. THERE IS NO WAY A USER PROGRAM CAN MASK CONTROLLER-DEFINED INTERRUPT CONDITIONS OR INTERRUPTS FROM OTHER DEVICES CONNECTED TO A SHARED GPPI.

1.3.2.7 LOAD COMMUNICATION REGION ADDRESS REGISTER

CALLING SEQUENCE: CALL T$GPPI(UNIT,7,LENGTH,XX,REGION,CODE)

THE ADDRESS REGION OF THE COMMUNICATION REGION ARRAY FOR DEVICE UNIT IS LOADED INTO THE GPPI CONTROLLER. ITS LENGTH IN WORDS IS SPECIFIED BY LENGTH. THE PAGE(S) OF MEMORY THAT CONTAIN REGION ARE "LOCKED" (OR "WIRED") TO THEIR CURRENT PHYSICAL PAGES UNTIL THE DEVICE IS UNASSIGNED. THIS INSURES THAT REGION WON'T BE PAGED OUT OR MOVED AROUND WITH RESPECT TO PHYSICAL MEMORY. THUS, THE GPPI IS GUARANTEED THAT IT IS ALWAYS REFERENCING REGION AS DEFINED BY THE USER.

NOTES:

THE USE OR NON-USE OF THIS FUNCTION IS STRICTLY DEPENDENT ON THE PARTICULAR GPPI IMPLEMENTATION. HENCE, IF USED IN A CERTAIN GPPI DESIGN, THE USER IS REFERRED TO THAT DESIGN SPECIFICATION FOR THE LENGTH, CONTENTS, FORMAT, AND USAGE OF THE COMMUNICATION REGION. IN ANY IMPLEMENTATION, THE MAXIMUM LENGTH LENGTH OF THE REGION REGION IS LIMITED TO ONE PAGE (1024 WORDS).

1.3.2.8 EXECUTE DEVICE-DEPENDENT OTA

CALLING SEQUENCE: CALL T$GPPI(UNIT,8,DATA,FCN,XX,CODE)

A DEVICE-DEPENDENT OTA IS EXECUTED. FCN CONTAINS THE OTA FUNCTION NUMBER AND DATA CONTAINS A WORD OF DATA LOADED INTO

THE A REGISTER PRIOR TO THE OTA.

NOTES:

THE GPPI RESERVES OTA'S '02-'10 FOR THE IMPLEMENTATION OF DEVICE DEPENDENT FUNCTIONS. OTHER VALUES OF FCN WILL CAUSE AN INVALID PARAMETER ERROR. NO VALIDITY CHECKING IS PERFORMED ON DATA.

1.3.2.9 RESET DEVICE

CALLING SEQUENCE: CALL T$GPPI(UNIT,9,XX,XX,XX,CODE)

A RESET SEQUENCE IS PERFORMED ON THE INTERFACE TO THE DEVICE SPECIFIED BY UNIT.

NOTES:

FOR A DESCRIPTION OF THE STATE OF A DEVICE FOLLOWING A RESET, REFER TO THE SPECIFICATIONS OF THE PARTICULAR GPPI CONTROLLING THE DEVICE AND/OR DOCUMENTATION OF THE DEVICE.

IF A DEVICE IS BEING USED BY MULTIPLE USERS (E.G., A MEGATEK WITH TWO MONITORS) THE RESET WILL AFFECT ALL USERS ON THE DEVICE. T$GPPI MAKES NO ATTEMPT TO SELECTIVELY RESET A SUBUNIT OF A SHARED DEVICE.

1.3.2.10 LOAD DEVICE TIMEOUT REGISTER

CALLING SEQUENCE: CALL T$GPPI(UNIT,10,TIME,XX,XX,CODE)

THE DEVICE TIMEOUT REGISTER FOR DEVICE UNIT IS LOADED WITH THE VALUE TIME. TIME IS A SIGNED POSITIVE INTEGER REPRESENTING THE AMOUNT OF TIME IN MILLISECONDS THAT A GPPI CONTROLLER SHOULD WAIT FOR A DEVICE RESPONSE DURING AN INTERFACE SEQUENCE. IF NO RESPONSE IS RECEIVED IN THE INDICATED TIME PERIOD, A TIMEOUT ERROR WILL OCCUR. THIS REGISTER MAY ALSO BE LOADED WITH THE VALUE -1 WHICH WILL DISABLE THE TIMEOUT FACILITY ENTIRELY. THIS REGISTER WILL BE INITIALIZED TO A DEFAULT VALUE OF -1 WHEN A DEVICE IS ASSIGNED.

1.3.2.11 RELEASE COMMUNICATION REGION

CALLING SEQUENCE: CALL T$GPPI(UNIT,11,XX,XX,XX,CODE)

THE PAGE(S) OF MEMORY THAT CONTAIN THE COMMUNICATION REGION DEFINED IN A PREVIOUS CALL (T$GPPI FUNCTION 7) ARE "UNLOCKED" (OR "UNWIRED") FROM THEIR ASSIGNED PHYSICAL PAGES. ALSO, THE CORRESPONDING ADDRESS THAT HAS BEEN CARRIED IN THE GPPI CONTROLLER IS ZEROED.

NOTES:

WHENEVER A COMMUNICATION REGION IS BEING USED, THIS T$GPPI FUNCTION SHOULD ALWAYS BE CALLED BEFORE THE PROGRAM EXITS OR AFTER A WARM START OCCURS TO INSURE PROPER CLEAN-UP OF THE MAPPING OF THIS REGION BETWEEN THE CONTROLLER AND THE USER'S MEMORY. UNDESIRABLE CONSEQUENCES MAY RESULT IF THIS IS NOT DONE.

1.3.2.12 PROGRAMMED I/O (OCP, SKS, INA, OTA) (RESTRICTED CALL)

CALLING SEQUENCE: CALL T$GPPI(UNIT,:1000N,DATA,FCN,XX,CODE)

N SELECTS A PIO INSTRUCTION TO BE EXECUTED -- OCP, SKS, INA, OTA FOR N = 1, 2, 3, 4. FCN IS THE FUNCTION CODE FOR THE INSTRUCTION; DATA IS THE DATA RETURNED BY AN INA OR LOADED INTO THE A-REGISTER PRIOR TO AN OTA.

NOTES:

THE PROGRAMMED I/O FUNCTIONS ARE AVAILABLE ONLY TO A USER LOGGED IN AT THE SYSTEM CONSOLE.

THESE FUNCTIONS ARE PROVIDED ONLY FOR THE PURPOSE OF EXPEDITING THE IMPLEMENTATION OF CERTAIN TEST AND MAINTAINENCE FUNCTIONS BY PRIME PERSONNEL. USE OF THESE FUNCTIONS IN USER PROGRAMS IS STRONGLY DISCOURAGED, AND FUTURE SUPPORT OF THESE FUNCTIONS IS NOT GUARANTEED BY PRIME COMPUTER, INC.

1.3.3 DESCRIPTION OF T$GPPI OPERATIONS

1.3.3.1 SYSTEM COLD START AND MICROCODE LOADING

THE GPPI DIM PROCESS, GPIDIM, TAKES CARE OF ALL SYSTEM COLD START PROCEDURES FOR THE MPC4 CONTROLLERS. WHEN FIRST ACTIVATED, THE GPIDIM PROCESS WILL DETERMINE WHETHER ANY MPC4'S EXIST AT DEVICE CODES 75 AND 76. FOR EACH CONTROLLER THAT IT FINDS, IT WILL PERFORM THE FOLLOWING THINGS:

FIRST, GPIDIM EXAMINES THE ID READ FROM THE CONTROLLER TO SEE IF A MICROCODE LOAD OPERATION IS REQUIRED. THE ID OF A GPPI IS OF THE FORM:

```
R RRS SSS SEW III III
```

WHERE RRR IS THE CONTROLLER REVISION LEVEL, SSSSS IS THE BACKPLANE SLOT NUMBER, E IS AN EXTENSION TO THE ID FIELD, W IS 1 IF MICROCODE LOADING IS REQUIRED, AND IIIIII IS THE CONTROLLER ID. IF W=1, GPIDIM CONSTRUCTS A FILE NAME OF THE FORM:

```
GPPIII.UCODE
```

WHERE 'II' IS THE OCTAL REPRESENTATION OF THE CONTROLLER'S DEVICE ADDRESS. THIS FILE IS EXPECTED TO BE THE MICROCODE FILE FOR THE GPPI AND TO BE IN THE UFD "SYSTEM". THE FILE WILL BE READ AND LOADED INTO THE GPPI. IF ANY KIND OF A FILE SYSTEM ERROR IS ENCOUNTERED DURING THIS OPERATION, AN ERROR MESSAGE OF THE FOLLOWING FORM WOULD APPEAR ON THE SYSTEM CONSOLE:

```
ERROR ATTEMPTING MICROCODE  LOAD  ON  DEVICE  ADDRESS  II.
(STANDARD ERRPR$  MESSAGE  GOES  HERE) SYSTEM>GPPIII.UCODE
(GPIDIM)
```

A TYPICAL ERRPR$ MESSAGE THAT MIGHT APPEAR WOULD BE "NOT FOUND" INDICATING THAT THE FILE WAS NOT FOUND IN THE "SYSTEM" UFD. IF ANY ERROR DOES OCCUR, THIS CONTROLLER IS NOW CONSIDERED UNUSABLE BY THE OPERATING SYSTEM. OF COURSE, IF W=0 (IN THE ID), THIS WHOLE STEP IS UNNECESSARY.

NEXT, GPIDIM WILL START THE CONTROLLER (I.E. THE MICROCODE) AND CHECK TO SEE THAT THE MICROCODE VERIFICATION ROUTINE COMPLETES SATISFACTORILY. IF IT DOES NOT, THE FOLLOWING MESSAGE WILL APPEAR ON THE SYSTEM CONSOLE:

```
MPC4 CONTROLLER DID NOT VERIFY.
```

AND THIS CONTROLLER IS NOW CONSIDERED UNUSABLE.

LASTLY, GPIDIM ALLOCATES THE NECESSARY SEGMENT ZERO "WINDOWS" FROM THE POOL FOR THIS CONTROLLER. IF THIS SHOULD FAIL, THE FOLLOWING MESSAGE WILL APPEAR ON THE SYSTEM CONSOLE:

INSUFFICIENT SEG 0 WINDOWS FOR MPC4.

AND THIS CONTROLLER IS NOW CONSIDERED UNUSABLE.

1.3.3.2 GENERAL OPERATION OF I/O

MOST OF THE T$GPPI CALLS OPERATE BY 1) VALIDATING THE ARGUMENTS, THEN 2) QUEUEING A REQUEST FOR I/O TO THE GPPI DIM (THE DEVICE INTERFACE MODULE, OR INTERRUPT PROCESS), AND 3) RETURNING TO THE CALLER. IF THE USER MAKES A SUBSEQUENT T$GPPI CALL AND THE PRECEDING REQUEST HAS NOT YET BEEN SATISFIED, HIS PROCESS WILL WAIT IN T$GPPI UNTIL THAT PRECEDING REQUEST HAS COMPLETED. THE OVERALL EFFECT IS THAT THE USER MAY HAVE ONE I/O REQUEST PENDING (OR IN PROGRESS)

WHILE PERFORMING OTHER PROCESSING.

IF AN I/O ERROR (BITS 2 THROUGH 6 IN STATUS(1)) OR DEVICE INTERRUPT FOR THE USER'S UNIT SHOULD OCCUR, THE ERROR CODE E$IEDI WILL BE RETURNED BY T$GPPI. THESE CONDITIONS ARE CHECKED FOR IN T$GPPI IMMEDIATELY BEFORE QUEUEING THE NEXT I/O OPERATION TO THE DIM. IF THEY ARE FOUND, THE CURRENT I/O REQUEST IS NOT QUEUED AND E$IEDI IS RETURNED IMMEDIATELY. THIS IS TO ALLOW THE USER TO CUT DOWN ON THE NUMBER OF WAIT/POLL CALLS THAT HE WOULD OTHERWISE HAVE TO MAKE TO T$GPPI. IN EFFECT, T$GPPI NOTIFIES THE USER WHEN ANYTHING OF INTEREST HAPPENS. NOTE THAT T$GPPI MAKES NO CLAIMS ABOUT THE CORRESPONDENCE OF PARTICULAR I/O OPERATIONS WITH THE OCCURRENCE OF ERRORS AND/OR DEVICE INTERRUPTS. IT IS STRICTLY THE CALLER'S RESPONSIBILITY TO INTERPRET THIS KIND OF INFORMATION.

1.3.3.3 SYSTEM WARM START AND T$GPPI

IF A SYSTEM WARM START OCCURS, THE GPPI IS RESET TO THE INITIALIZED STATE. THIS PROBABLY MEANS THAT ANY DEVICE(S) CONNECTED TO THE GPPI HAS RECEIVED A "RESET" SIGNAL OF SOME SORT. (AGAIN, THIS IS REALLY DEPENDENT UPON THE PARTICULAR GPPI IMPLEMENTATION.) THEREFORE, TO NOTIFY THE USER PROGRAM OF THIS CONDITION, T$GPPI WILL RETURN THE E$WMST CODE WHEN THIS HAPPENS. NOTE THAT THIS CODE WILL BE RETURNED TO THE USER ONLY ONCE AFTER A WARM START OCCURS. THE USER PROGRAM SHOULD RESPOND IN A MANNER APPROPRIATE TO THE CONNECTED DEVICE.

T$GPPI CHECKS FOR THE WARM START CONDITION JUST BEFORE RETURNING TO THE USER FROM A T$GPPI CALL. DEPENDING UPON "WHERE" IN T$GPPI THE WARM START HAPPENED, THE CURRENT I/O

REQUEST MAY OR MAY NOT HAVE ACTUALLY BEEN PERFORMED. THE USER SHOULD TAKE THIS INTO ACCOUNT WHEN HANDLING A WARM START CONDITION.

*

657 (DIRECV)

DATE: OCTOBER 7, 1981
SUBJECT: R3POFH AND RELATED SUBJECTS

THERE IS A PRESENT PROBLEM WHENEVER AN SPL (OR ANY OTHER LANGUAGE THAT SUPPORTS ON-UNITS) PROGRAM IS INTERRUPTED VIA A HARDWARE FAULT (THE CONDITIONS ARE DETAILED IN THE SUBROUTINES GUIDE). IF THE ON-UNIT HANDLER CAUSES A DYNAMIC LINK TO A SHARED LIBRARY ROUTINE, THE LIBRARIES WILL BE RE-INITIALIZED. THE MOST SERIOUS IMPACT OF THIS IS THAT THE STORAGE ALLOCATED BY THE PROGRAM IN WHICH THE ORIGINAL FAULT OCCURRED, WILL GO AWAY, RESULTING IN APPARENTLY MYSTERIOUS BEHAVIOR FOLLOWING THE COMPLETION OF THE ON-UNIT HANDLER.

THE SOLUTION INVOLVES A CHANGE IN THE ROUTINE THAT PERFORMS THE DYNAMIC LINKS (R3POFH) AND A COMPANION CHANGE IN THE COMMAND PROCESSOR. NO CHANGE TO PR1ME PRODUCTS OR USER PROGRAMS ARE REQUIRED. HOWEVER, ANY SHARED LIBRARIES MUST BE REBUILT WITH THIS REV, IN ORDER TO PICK UP THE
+
R3POFH CHANGE. THIS WILL AUTOMATICALLY BE DONE FOR ALL SHARED LIBRARIES ON THE MASTER DISK.

*

658,680-685 (RJE PRODUCTS)

SEE <M183A1>RJECOM (HISTORY FILES) FOR INFO.

*

SUBJECT: SEG

RELEASE: REV18.3

DATE: SEPTEMBER 29, 1981

1 NEW FUNCTIONALITY

- SEG NOW AUTOMATICALLY LOADS SPLLIB WHENEVER THE PURE FORTRAN LIBRARY IS LOADED. THE SUBCOMMANDS LI AND PL DO THIS. THE RESULT IS THAT REV18.3 SEG CANNOT BE RUN ON ANY SYSTEM WHICH DOES NOT HAVE THE SPL LIBRARY.

2 PROBLEMS FIXED

POLAR NO.	DESCRIPTION
36524	SEG NOW REPORTS AN ERROR IF USER LOADS A SEG FILE IN THE VLOAD SUBPROCESSOR.

- SEG NOW DELETES SEG FILES WHICH WERE PREVIOUSLY TRASHED FOR VARIOUS REASONS (E.G. USER HIT CONTROL-P IN THE MIDDLE LINKING SESSION).
- IF A USER TRIES TO USE SEGMENT 4035, WHICH IS USED BY SEG INTERNALLY FOR ITS OWN SYMBOL TABLE, A WARNING MESSAGE WILL BE GIVEN AND FLOW OF CONTROL WILL ALWAYS RETURN TO SEG SUBCOMMAND LEVEL. A WARNING ERROR CODE WILL BE RETURNED AT THE END OF THE SESSION. THIS IS AN IMPROVEMENT OVER REVISION 18.2 SEG WHICH WOULD ALWAYS ABORT TO PRIMOS UNDER THIS CIRCUMSTANCE, EVEN IF UNINITIALIZED DATA WAS THE ONLY CODE LOADED INTO 4035.

3 ENVIRONMENT

- THIS VERSION OF SEG REQUIRES REV18 PRIMOS, REV18 PFTNLIB, AND REV18.3 SPLLIB.

4 INSTALLATION AND BUILD PROCEDURES

- STANDARD INSTALLATION AND BUILD PROCEDURE.

- THIS VERSION OF SEG.BUILD.CPL WILL NOT OPERATE CORRECTLY WITH REV18.2 SEG, BUT IS COMPATIBLE WITH ALL EARLIER VERSIONS OF REV18 SEG.

*
660 (NOT USED)

*
661 (SLIST)

THE MAXIMUM LINE LENGTH HAS BEEN INCREASED FROM 140 TO 1024 CHARACTERS.

*
662 (SPL)

SPL HAS BEEN MODIFIED FOR REV 18.3.
SPL IS AN INTERNAL TOOL, IT IS NOT INTENDED FOR CUSTOMER USE.

*
663 (SPOOL)

THE FOLLOWING ALTERATIONS HAVE BEEN MADE TO 18.3 SPOOL TO CORRECT THE PROBLEMS THAT WERE INTRODUCED WITH REV 18;

1. FILES WHICH CONTAIN NULL LINES CAN BE SPOOLED TO ENVIRONMENTS THAT HAVE UPCASE ON.

2. FILES THAT HAVE BEEN SUBMITTED WITH THE -OPEN KEY WILL NOT HAVE TO WAIT SO LONG TO BE PRINTED. THIS IS BECAUSE THE SPOOLER WILL RESCAN THE QUEUE IMMEDIATELY AFTER SIZING THE FILE.

3. THE CENTRONICS IS NOW ASSIGNED PROPERLY.

4. THE CONVENTION FOR OVERPRINTING USED BY RJE IS NOW OBSERVED IN NO FORMAT MODE. THAT IS, A LINE WHOSE LAST TWO CHARACTERS ARE CARRIAGE RETURN AND LINE FEED WILL BE OVERPRINTED ON THE PREVIOUS LINE.

5. THE SPOOLER PHANTOM NOW SEARCHES THE QUEUES EVERY MINUTE

INSTEAD OF EVERY TWO MINUTES.

6. THE PROP -BACKUP COMMAND LEAVES A FILE IN THE CORRECT PRINT MODE INSTEAD OF PUTTING IT INTO NO FORMAT MODE.

7. THE EXTRA DOTS IN PLOT FILES (FOR THE 300 LPM PRINTER/PLOTTER) HAVE BEEN REMOVED.

8. A LINE THAT CONTAINS ONLY AN OCTAL ONE WILL NO LONGER CAUSE THE SPOOLER PHANTOM TO LOG OUT. (THAT CONTROL CODE INSTRUCTS THE SPOOLER TO ENTER PAGINATE MODE, CREATE A BLANK HEADER LINE, AND DO A PAGE EJECT.)

9. PRINTERS MAY NOW BE RUN ON AMLC LINES WHOSE LINE NUMBERS ARE GREATER THAN TWO DIGITS.

10. THE SPOOL$ INTERFACE HAS BEEN EXPANDED TO INCLUDE A LOGICAL DESTINATION NAME FOR THE FILE. (THIS IS ANALOGOUS TO THE -AT NAME OF THE SPOOL COMMAND LINE.) TO SPECIFY A LOGICAL DESTINATION THE CALLER MUST SET BIT(10) OF INFO(3) AND PUT THE SIXTEEN CHARACTER, BLANK PADDED LOGICAL DESTINATION NAME IN INFO(13) - INFO(20).

*
664 (VFTNLIB)

REV 18.3 ---- VFTNLIB

BINARY F77 I/O NOW WORKS 20% - 40% FASTER.

NAMELIST SUPPORT FOR F77 HAS BEEN UNSHARED AS WE NEEDED THE ROOM IN THE SHARED LIBRARY (NAMELIST WAS THOUGHT TO BE LITTLE USED).

TWO MORE LOGICAL UNITS WERE ADDED TO THE IOCS SYSTEM FOR USE BY PL1.

THE UNITS ARE 139 AND 140 FOR PRINTER UNITS 0 AND 1 RESPECTIVELY. FTN/F77

PROGRAMS MAY REFERENCE THESE UNITS AS WELL...

THIS VERSION TRACKS ALL PREVIOUS VERSIONS, AND THE FOLLOWING BUGS HAVE BEEN FIXED:

NAMEQ$ NOW HAS A TEST FOR LOWER CASE 'A'.

CABS NO LONGER OVERFLOWS IF ARGUMENTS ARE WITHIN LEGAL BOUNDS.

F$IO77 NOW ACCEPTS B-FORMAT STATEMENTS WITH A TRAILING (AND PRECEEDING) BLANK.

F$IOFTN OPERATES PROPERLY ON MULTIPLE, INTERNAL SEQUENTIAL COMMAS.

*
665 (COBOL)

COBOL AT REVISION 18.3

NOTE: THE ALLOCATION OF SEGMENT '2014 TO THE SHARED COBOL LIBRARY AND THE SHARED MIDAS LIBRARY HAS BEEN CHANGED DUE TO THE INCREASE IN SIZE OF THE COBOL LIBRARY AT REVISION 18.3. THE ALLOCATION OF SEGMENT '2014 TO THESE LIBRARIES WILL NOW BE AS FOLLOWS:

```
C2014A    '100   TO    '277    /* COBOL
K2014A    '300   TO    '777    /* MIDAS
C2014B   '1000   TO  '37777    /* COBOL
K2014B  '40000   TO '177777    /* MIDAS
```

BECAUSE OF THIS NEW CONFIGURATION IT IS VERY IMPORTANT NOT TO USE AN 18.3 SHARED COBOL LIBRARY WITH A MIDAS LIBRARY THAT IS BELOW 18.3. REVISION 18.3 COBOL SHOULD ONLY BE RUN WITH 18.3 MIDAS AND UP.

THE FOLLOWING TARS (TECHNICAL ACTION REQUESTS) AND PSFS (POLERS SUBMITTAL FORMS) HAVE BEEN ANSWERED AT REVISION 18.3 COBOL:

PSF# 33223-

A CALLED COBOL PROGRAM WITH MORE THAN SEVENTEEN ARGUMENTS IN THE USING CLAUSE OF THE PROCEDURE DIVISION GAVE INTERNAL ERROR = TBL GROUP.

PSF# 14164 23993 27712 31610 31857 32001 824559 824569-

COBOL ALLOWED THE REWRITE OF A PRIMARY KEY IN A MIDAS FILE, WHICH EVENTUALLY WOULD CAUSE INDEXES TO BE CORRUPTED. COBOL NOW RETURNS A FILE-STATUS OF 22.

PSF# 29442 33612 34268-

WHEN DOING A READ (KEY FOUND), READ (KEY NOT FOUND), AND A WRITE TO AN INDEXED MIDAS FILE, A MIDAS 33 ERROR WOULD OCCUR.

PSF# 20014 21675 27321 31699 34328-

COBOL WOULD NOT NAME A FILE WITH THE FIRST FOUR CHARACTERS OF THE PROGRAM-ID IF IT HAD BEEN RE-OPENED FOR OUTPUT AND WRITTEN TO WITH THE ADVANCING OPTION.

PSF# 21651 22550 32278-

THE COMPILER WOULD ALLOW A DELETE STATEMENT ON A SEQUENTIAL FILE AND GENERATE A CALL TO C$DS (WHICH DOES NOT EXIST). THIS IS ILLEGAL COBOL SEMANTICS.

PSF# 30267-

THE COMPILER WOULD NOT RECOGNIZE THE PLURAL FORM OF THE VALUES CLAUSE ("VALUES"). IT WAS BEING FLAGGED AS A SYNTAX ERROR.

PSF# 25387 29232 30101 33611 82495-

THE SHARED COBOL COMPILER WOULD HALT WITH AN "ACCESS_VIOLATION$" WHEN SELECTING THE -XREF OPTION. THE NON-SHARED COBOL COMPILER (NCOBOL) WOULD PROCESS THE -XREF OPTION, BUT WOULD HALT WITH A "POINTER_FAULT$" IF THE PROGRAM WAS OVER ABOUT 1500 LINES.

PSF# 28616 37750-

AN INTERNAL ERROR 116 WAS OCCURRING IF AN ACCEPT <DATA-NAME> FROM DATE STATEMENT WAS THE FIRST STATEMENT AFTER A REFERENCE TO A SUBSCRIPTED DATA-NAME.

PSF# 34308-

ADD STATEMENTS FOLLOWING A SUBTRACT CORRESPONDING STATEMENT WERE BEING FLAGGED AS ERRORS.

PSF# 24161 35416 82275-

A GROUP ITEM WITH SIZE GREATER THAN 32K BYTES AND LESS THAN 64K BYTES WOULD CAUSE THE COMPILER TO ABORT WITH AN ERRONEOUS ERROR MESSAGE.

PSF# 29484-

AN ATTEMPT TO PERFORM A REWRITE TO AN INDEXED FILE, WITHOUT FIRST PERFORMING THE REQUIRED READ, WOULD CAUSE THE PROGRAM TO ABORT INSTEAD OF ACTIVATING THE INVALID KEY CLAUSE AND RETURNING A FILE-STATUS OF 91.

PSF# 35232 35428 35429-

COBOL WOULD NOT RETURN THE CORRECT RECORD WHEN THE "READ NEXT" STATEMENT ENCOUNTERED A LOCKED RECORD IN A MIDAS FILE.

PSF# 29964-

THE COMPILER WOULD TERMINATE WITH "INTERNAL ERROR 106" IN A "CALL" STATEMENT WITH A SUBSCRIPTED VARIABLE. A "D" LEVEL ERROR WILL NOW BE PRODUCED FOR CALLS WITH SUBSCRIPTED VARIABLES.

*

666-670 (DBMS)

SUBJECT: DBMS

RELEASE: 18.3

DATE: OCTOBER 12, 1981

I. NEW FUNCTIONALITY

DBUTL

TWO EXISTING COMMANDS HAVE BEEN EXTENDED TO ACCESS THE CALC FILES. THE HELP COMMAND HAS BEEN MODIFIED TO DESCRIBE THE NEW EXTENSIONS.

DUMP CALC [REC-ID]

THIS ALLOWS THE USER TO SEE THE CONTENTS OF THE CALC FILE ASSOCIATED WITH A PARTICULAR RECORD TYPE. IF THE REC-ID IS NOT PROVIDED, IT WILL DEFAULT TO THE LAST RECORD TYPE REFERRED TO. UNUSED OR DELETED ENTRIES WILL NOT BE DISPLAYED.

ED R [REC-ID]

THIS COMMAND WILL ALLOW EDITING OF THE CALC FILE IN THE SAME MANNER AS THE EXISTING ED COMMANDS ALLOW IT FOR AREAS AND SETS.

DBACP

WHENEVER THE BEFORE IMAGE HEADER IS READ (EG: VERIFY, RESTORE, ETC.), IF THE CONVERSION TO REV 18 FORMAT HAS NOT YET BEEN DONE, THEN DBACP WILL DO IT. THE USER IS NOTIFIED OF THE START AND COMPLFTION OF THE CONVERSION. THIS REPLACES THE REV18 COMMAND FORMERLY IN DBUTL WHICH LEFT TOO MUCH CHANCE FOR ERROR AND THE REV17 DATABASE CHECK IN DBACP. FORMERLY DBACP WOULD GIVE A MISLEADING ERROR MESSSAGE (ERROR 1) WHEN THE USER ATTEMPTED TO ACCESS AN UNCONVERTED SCHEMA WITH VERIFY, RESTORE, ETC.

II. PROBLEMS FIXED

DMLCP

THE EMERGENCE OF THE RAM MONITOR IN REVS 17.6/18.0 REPLACED REGISTER SETTINGS FOR DML TRACES WITH KEY WORDS (EG: -TRACE LONG). THE OLD REGISTER 3 SETTING, WHICH WAS USED TO PRODUCE TIMING INFORMATION WHICH WOULD LATER BE FORMATTED BY *SUMMARY (NOW SUMMARY.SAVE) WAS SOMEHOW NOT INCLUDED IN THE NEW TRACE OPTIONS. AS OF REV 18.3, AN OPTION OF -TRACE TI WILL RESTORE THE FORMER CAPABILITY.

THE ACCESS STRATEGY GENERATOR (ASG) HAS BEEN CORRECTED TO USE ONLY MANDATORY AUTOMATIC MEMBERS IN DEVISING ITS ACCESS STRATEGY SO THAT VISTA WILL FIND ALL THE VIRTUAL RECORDS WHEN THEY ARE COMPOSED OF SEVERAL RECORDS. (DISCOVER POLER #37495)

THE -VERIFY COMMAND LINE OPTION INTRODUCED IN REV 18.2 IS SWALLOWED BY THE COMMAND LINE INTERPRETER AT REV 19 SINCE IT IS A RESERVED KEYWORD. TO RESOLVE THIS WE HAVE CHANGED THIS KEYWORD TO -VALIDATE. (FOR A COMPLETE DESCRIPTION OF THIS FEATURE SEE THE REV 18.2 DBMS SOFTWARE RELEASE DOCUMENT.)

THE SUPPRESS VERB HAS BEEN CORRECTED TO PERFORM AS SPECIFIED IN THE PDR. SUPPRESS ALL HAD NO EFFECT AND SUPPRESSION OF MORE THAN ONE CLASS (RECORD, AREA, SETS) WOULD ONLY WORK FOR THE FIRST CLASS. (POLERS #32899, #11101 AND #27351)

IF A SUBSCHEMA DOES NOT INCLUDE ALL OF THE MEMBER RECORD TYPES OF A MULTI-MEMBER SET, FIND/FETCH FIRST REC-NAME OF SET SET-NAME WOULD SOMETIMES FAIL TO FIND THE FIRST MEMBER OF THE SPECIFIED TYPE GIVING AN EXCEPTION CODE OF 26. (POLER #34242)

FIND USING DBK CORRECTED SO THAT OWNER DIRECTORY POINTER IN MEMBER RECORD NO LONGER SET TO ZERO. [TAR #34482, #36367]

CONCURRENT UPDATING OF A SET FILE COULD RESULT IN LOST UPDATES THEREIN IF A PROGRAM HAPPENED TO RETURN TO THE SAME SET NODE IN THE NEXT TRANSACTION BEFORE ACCESSING ANY OTHER NODES. (POLER #40402, #36531)

SET SORT ORDER WITH MEMBERS CONTAINING A FOUR SEGMENT SORT KEY AND AUTOMATIC INSERTION IS NOW MAINTAINED IN SORTED ORDER. [TAR #20941]

IF A SUBSCHEMA FAILS TO INCLUDE ALL ITEMS, AND AN EXCLUDED FIELD IS THE LAST ITEM AND IS LESS THAN ONE WORD IN LENGTH (E.G. PIC(X)), ALL CALC AND RELATED POINTERS ARE NOW UPDATED CORRECTLY. [TAR #34496]

RLIB

IF THE DISK PARTITION FILLED UP DURING A SCHED SESSION, THE ERROR MESSAGE RETURNED WAS "EOF IN A RAM FILE." IT HAS BEEN CHANGED TO "DISK FULL."

THE AFTER IMAGE LOG FILE WAS GETTING RETRIEVAL TRANSACTIONS WRITTEN TO IT WHICH ADDED UNNECCESSARY OVERHEAD TO SUCH TRANSACTIONS AND TO ROLL FORWARD RECOVERY. (POLERS #34498 AND #32209)

DBUTL

THE REV18 COMMAND TO CONVERT A SCHEMA'S BEFORE IMAGE FILE TO THE NEW FORMAT HAS BEEN DEACTIVATED IN DEFERENCE TO THE ABOVE MENTIONED ENHANCEMENT TO DBACP. THIS PROTECTS THE USER FROM INADVERTANTLY RUNNING THE CONVERSION MORE THAN ONCE WHICH MAKES THE BEFORE IMAGE FILE UNUSABLE.

THE DUMP AFTER COMMAND WOULD SOMETIMES GIVE GARBLED RESULTS DUE TO AN UNINITIALIZED ARRAY. THIS HAS BEEN CORRECTED.

SWITCH AREA (AREA X) NO LONGER DESTROYS DATA FOR ANY SUBSEQUENT SET COMMAND. [TAR #27966]

III. OUTSTANDING PROBLEMS

CDML & FDML

DOES NOT HANDLE ON ERROR CLAUSE PARAGRAPH NAMES WHICH BEGIN WITH NUMBERS. [TAR #12613]

GENERATES A FATAL ERROR FOR A NONRESERVED WORD FIELD NAME (OTHER). [TAR #37464]

NONDESCRIPTIVE ERROR MESSAGE RETURNED WHEN ATTEMPTING TO OPEN A SPECIFIC AREA (WITH ONLY ONE RECORD TYPE) WHICH HAS NOT BEEN INCLUDED IN THE SUBSCHEMA. [TAR #41434]

TABLE OVERFLOW ERROR RECEIVED WHEN ATTEMPT A FETCH VIA CURRENT OF SET USING NINE (9) DATA ITEMS. [TAR #32202]

ALL-PURPOSE 'I/O ERROR ON UNIT 6' ERROR MESSAGE DOES NOT INDICATE THE FILE IN ERROR. [TAR #82916]

PREPROCESSOR DOES NOT CLOSE LISTING FILE AFTER AN ERROR. [TAR #20712]

ON ERROR, THE MANIPULATED OUTPUT FILE (D_) IS NOT BEING TRUNCATED. [TAR #32346]

CLUP

CLUP CLOSES VIRTUALLY ALL FILE UNITS AND SO CANNOT BE RUN FROM A CPL PROGRAM. [TAR #40184, #35558]

DBACP

'END OF FILE IN A DATABASE FILE' RECEIVED DURING EXPAND SET TO A VOLUME WITH INSUFFICIENT DISK SPACE; ENTRY REMAINS IN SD FILE. [TAR #29777]

DOES NOT CONSISTENTLY CLOSE ALL OPENED FILES. [TARS #34472, #36602, #37900]

DOES NOT ACCEPT LOWER CASE INPUT. [TAR #33543]

DOES NOT ALWAYS RESTORE/EXPAND/DELETE MULTIVOLUME SAVES/FILES CORRECTLY. [TARS #36992, #32706, #28882, #40119, #32190, #34245, #34493]

DOES NOT HANDLE TAPE ERRORS GRACEFULLY. [TARS #34481, #36464, #43922]

IF AFTER IMAGE RESTORE FILE TREENAME INCLUDES PASSWORDS AND IS ENTERED INCORRECTLY, A POINTER FAULT IS RETURNED TO THE USER. [TAR #32211]

DMLCP

DOES NOT ALWAYS HANDLE BIT STRINGS CORRECTLY. [TARS #34469, #34470]

FETCH RECORD NAME USING EIGHT (8) ITEMS CAUSES ACCESS VIOLATION. [TAR #41442]

UNABLE TO STORE CALCED RECORD (WITH ODD NUMBER BYTE CALC KEY FIELD) USING FORTRAN. [TAR #22772]

IF A MEMBER RECORD IS DELETED THEN A NEW MEMBER STORED, ERROR 2217F IS RECEIVED WHEN TRYING TO FIND THE NEW MEMBER IN A SET WHERE ORDER IS 'NEXT'. [TAR #41492]

STORE SHOULD CREATE NEW OWNER DIRECTORY ENTRIES IN SETS WHICH ARE OUTSIDE THE SUBSCHEMA. [TAR #40529, #40537]

ALLOW CHUNKING OF DATE FIELDS WHERE APPROPRIATE. [TAR #34764, #33510]

EXCEPTION CONDITION RATHER THAN FATAL ERROR GENERATED FOR FIND NEXT RECORD REC-NAME IN MULTIMEMBER SET WHEN REQUIRED LIST NUMBER IS LOWER THAN CURRENT LIST NUMBER. [TAR #37463]

LARGE SUBSCHEMA DESTROYS 'AREA-NAME' TABLE. [TAR #37380]

UNABLE TO ACCESS NEXT/PRIOR IN A SET FOLLOWING A DELETE OF A RECORD OCCURRENCE. [TARS #82630, #37971, #82606, #34678, #41436]

FSUBS

DOES NOT INDICATE THE LINE NUMBER OF A DUPLICATE ELEMENT NAME. [TAR #36876]

RLIB

BIT MAP OVERFLOW. [TARS #29298, #36006]

SCHED

DESCRIPTION OF DATA ITEM BY SCHED DIFFERS FROM THAT OUTPUT BY SCHDEC. [TAR #37614]

SCHDEC

DOES NOT ACCEPT SINGLE QUOTES AROUND A UFD AND PASSWORD TREENAME. [TAR #33119]

TRUNCATES OUTPUT SOURCE FILE TREENAME TO 35 CHARACTERS. [TAR #36005]

TRUNCATES V99X PICTURE TO X. [TAR #33849]

SCHEMA

DATA TYPE OF 'DECIMAL' OR 'PICTURE' DOES NOT PERMIT CHECK RANGE CLAUSE USAGE. [TAR #23841]

SCHEMA SHOULD TRAP ERRORS WHERE MORE THAN ONE SET HAS MEMBER CLAUSES DEFINING THE SAME RECORD. [TAR #34475]

SIGN CHARACTER IN A PICTURE CLAUSE IN A SCHEMA HAS NO EFFECT. DBMS RELIES ON SUBSCHEMA IN COBOL TO ENFORCE THE PRESENCE OR ABSENCE OF A SIGN. [TAR #34766]

IV. ENVIRONMENT

REV 18.3 DBMS REQUIRES SPLLIB AS WELL AS AN 18.2 (OR HIGHER) RELEASE OF PRIMOS AND SEG.

V. INSTALLATION AND BUILD PROCEDURES

THE BUILD IS STANDARD. WHAT FOLLOWS IS INSTALLATION AND GENERAL INFORMATION.

FILES ON SYSTEM TAPE

DBMSEX (UFD)

```
C_INITINSTALLDBMSEX
C_INSTALLDBMSEX
C_SHAREDBMS
```

```
CMDNCO (SUB-UFD)

  DBACP    \
  DBUTL     >-  PRODUCT INTERLUDES
  CLUP     /

DBMSLB (SUB-UFD)

  DBACP    \
  DBUTL     >-  PRODUCT SEGMENT DIRECTORIES
  CLUP     /
  IDBMS.SEG     USED BY C_SHAREDBMS TO INITIALIZE SHARED SEGMENTS
  DUMP.SEG      DBMS FILE DUMP UTILITY
  SUMMARY.SAVE  UTILITY TO SUMMARIZE DML COMMAND TIMINGS
  VFYPRT.SAVE   UTILITY TO FORMAT OUTPUT OF -VERIFY OPTION
  DB2001   \
  DB2003    \_  SHARED SEGMENTS FOR EXECUTABLE DBMS PRODUCTS
  DB2012    /
  DB4000   /
  C_LOAD.LIB    COMMAND STREAM TO CREATE SHARED DYNAMIC LIBRARY
  C_USER.LIB    COMMAND STREAM TO INSTALL USERS DYNAMIC LIBRARY
  HTAB.INS.PMA  SOURCE NEEDED BY C_LOAD.LIB
  DYNT          SOURCE NEEDED BY C_USER.LIB

INFO (SUB-UFD)

  CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)

JOBS (SUB-UFD)

  CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

     COPY (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

     COMPILE (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

     LOAD (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
       FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
       FILES OF THE FORM @.L.CPL LOAD SEGMENTS.

PDBMS (SUB-UFD)
```

```
          SCHDIR        SCHEMA DIRECTORY
          DALIST        DATA ADMINISTRATORS LIST
          FDMLER  \
          CDMLER   \_   ERROR MESSAGE FILES
          DAERRS   /
          DBMSE   /


        INSERT (SUB-UFD)

          CONTAINS FILES OF THE FORM @@.INS.FTN USED TO BUILD DBMS
          (SPECIFICALLY UFD-DATA.INS.FTN [SEE SECTION "CHANGING THE
          DATABASE FILE UFD NAME AND PASSWORD"] AND FUNIT-DATA.INS.FTN
          [SEE SECTION "INTRODUCTORY MESSAGE CONTROL"]).


    DBMSEXBIN (UFD)
    ---------------

      CLUP.B     (SUB-UFD) --
      DBACP.B    (SUB-UFD)    \
      DBUTL.B    (SUB-UFD)     \
      DMLCP.B    (SUB-UFD)      !
      ILIB.B     (SUB-UFD)      !
      RLIB.B     (SUB-UFD)      !
      ULIB.B     (SUB-UFD)      !   THESE SUB-UFDS CONTAIN THE @@.BIN FILES
      CLIB.B     (SUB-UFD)      >-  FOR RUNTIME SHARED LIBRARIES AND
      IDBMS.B    (SUB-UFD)      !   EXECUTABLE PRODUCTS.
      ASI.B      (SUB-UFD)      !
      ASG.B      (SUB-UFD)     /
      DUMP.B     (SUB-UFD)    /
      SUMMARY.B  (SUB-UFD) --


    DBMSDEF (UFD)
    -------------

      C_INITINSTALLDBMSDEF
      C_INSTALLDBMSDEF


        CMDNCO (SUB-UFD)

          SCHEMA    INTERLUDE


        DBMSLB (SUB-UFD)

          SCHEMA    SEGMENT DIRECTORY
```

```
DBMSDEFBIN (UFD)
----------------

  SCHEMA.B (SUB-UFD)  CONTAINS @@.BIN FILES FOR SCHEMA COMPILER.


DBMSFTN (UFD)
-------------

  C_INITINSTALLDBMSFTN
  C_INSTALLDBMSFTN


     CMDNCO (SUB-UFD)

       FDML     \_   PRODUCT INTERLUDES
       FSUBS    /
       EXEC          ACTUAL SEGMENT DIRECTORY


     DBMSLB (SUB-UFD)

       FDML     \_   PRODUCT SEGMENT DIRECTORIES
       FSUBS    /
       C_FDML        EXEC PROCEDURE TO PRECOMPILE FTN DBMS APPLICATIONS
       C_FLOAD        "     "      " LOAD        "    "          "


     INFO (SUB-UFD)

       CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)


     JOBS (SUB-UFD)

       CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

          COPY (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

          COMPILE (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

          LOAD (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
            FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
```

```
              FILES OF THE FORM @.L.CPL LOAD SEGMENTS.


        PDBMS (SUB-UFD)

          SCHDIR        SCHEMA DIRECTORY
          DALIST        DATA ADMINISTRATORS LIST
          FDMLER   \
          CDMLER    \_  ERROR MESSAGE FILES
          DAERRS    /
          DBMSE    /


        LIB (SUB-UFD)

          DMLLIB        DBMS RUN-TIME SHARED LIBRARY


    DBMSFTNBIN (UFD)
    ----------------

      FDML.B  (SUB-UFD) \_   CONTAINS @@.BIN FILES FOR FTN PREPROCESSOR
      FSUBS.B (SUB-UFD) /    AND SUBSCHEMA COMPILER.


    DBMSCOB (UFD)
    -------------

      C_INITINSTALLDBMSCOB
      C_INSTALLDBMSCOB



        CMDNCO (SUB-UFD)

          CDML    \_    PRODUCT INTERLUDES
          CSUBS   /
          EXEC          ACTUAL SEGMENT DIRECTORY


        DBMSLB (SUB-UFD)

          CDML    \_   PRODUCT SEGMENT DIRECTORIES
          CSUBS   /
          C_CDML       EXEC PROCEDURE TO PRECOMPILE COBOL DBMS APPLICATIONS
          C_CLOAD      "       "       " LOAD       "     "       "


        INFO (SUB-UFD)

          CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)
      JOBS (SUB-UFD)

        CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


          COPY (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


          COMPILE (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.


          LOAD (SUB-UFD)

            CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
            FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
            FILES OF THE FORM @.L.CPL LOAD SEGMENTS.


      PDBMS (SUB-UFD)

        SCHDIR      SCHEMA DIRECTORY
        DALIST      DATA ADMINISTRATORS LIST
        FDMLER \
        CDMLER  \_  ERROR MESSAGE FILES
        DAERRS  /
        DBMSE  /



      LIB (SUB-UFD)

        DMLLIB      DBMS RUN-TIME SHARED LIBRARY


   DBMSCOBBIN (UFD)
   ----------------

     CDML.B  (SUB-UFD) \_   CONTAINS @@.BIN FILES FOR COBOL PREPROCESSOR
     CSUBS.B (SUB-UFD) /    AND SUBSCHEMA COMPILER.


   DBMSLGCL (UFD)
   --------------

     C_INITINSTALLDBMSLGCL
     C_INSTALLDBMSLGCL


      CMDNC0 (SUB-UFD)
      SCHED  \_   PRODUCT INTERLUDES
      SCHDEC /

    DBMSLB (SUB-UFD)

      SCHED  \_   PRODUCT SEGMENT DIRECTORIES
      SCHDEC /

    INFO (SUB-UFD)

      CONTAINS DOCUMENTATION FOR CURRENT REV (INCLUDING THIS MEMO)

    JOBS (SUB-UFD)

      CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

         COPY (SUB-UFD)

           CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

         COMPILE (SUB-UFD)

           CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.

         LOAD (SUB-UFD)

           CONTAINS CPL PROCEDURES USED TO BUILD ALL DBMS PRODUCTS.
           FILES OF THE FORM @.E.CPL LOAD LIBRARIES.
           FILES OF THE FORM @.L.CPL LOAD SEGMENTS.

    PDBMS (SUB-UFD)

      SCHDIR      SCHEMA DIRECTORY
      DALIST      DATA ADMINISTRATORS LIST
      FDMLER \
      CDMLER  \_  ERROR MESSAGE FILES
      DAERRS  /
      DBMSE  /

 DBMSLGCLBIN (UFD)
 -----------------

   SCHED.B  (SUB-UFD) \    CONTAINS @@.BIN FILES FOR SCHEMA EDITOR
   SCHDEC.B (SUB-UFD)  >-  AND SCHEMA DECOMPILER.
```

TEXTED.B (SUB-UFD) /

INSTRUCTIONS FOR INITIAL INSTALLATION OF DBMS

1. IF YOU ALREADY HAVE A VERSION OF DBMS ON YOUR SYSTEM, SEE THE SECTION UPGRADING AN EXISTING DBMS INSTALLATION.

2. RESTORE THE UFDS SUPPLIED ON TAPE. THESE MAY BE ONE OR MORE OF THE FOLLOWING:

   DBMSEX   DBMSDEF   DBMSFTN   DBMSCOB   DBMSLGCL

3. ON EACH PARTITION WHERE DATABASE FILES ARE TO BE STORED, CREATE A UFD PDBMS WITH OWNER PASSWORD ISIS. THEN USE FUTIL PROTECT 7 1 TO GIVE NON-OWNER READ RIGHTS TO NEW UFDS. THE DIRECTORY OF ALL SCHEMAS (SCHDIR), THE LIST OF VALID DATA ADMINISTRATORS (DALIST), AND VARIOUS DBMS ERROR MESSAGE FILES ARE ASSUMED BY THE SYSTEM TO BE IN THE UFD PDBMS ON THE PARTITION WITH THE LOWEST LOGICAL DISK NUMBER.

   TO CREATE THESE FILES, DO A FUTIL UFDCPY FROM ANY ONE OF THE DBMSXXXX>PDBMS UFDS TO THE PDBMS WITH THE LOWEST LOGICAL DISK NUMBER.

4. ONCE THE VARIOUS UFDS DESCRIBED ABOVE HAVE BEEN CREATED, ATTACH TO THE MFD WHERE YOU WANT DBMS TO RESIDE AND RUN THE COMMAND INPUT STREAMS C_INITINSTALLDBMSXXX FROM EACH OF THE NON-BINARY UFDS RESTORED FROM TAPE. FOR EXAMPLE:

   ```
   CO DBMSEX>C_INITINSTALLDBMSEX
   ```

5. FINALLY USE FUTIL TO DELETE THE UFD(S) RESTORED FROM TAPE.

UPGRADING AN EXISTING DBMS INSTALLATION

TO UPGRADE AN EXISTING DBMS SYSTEM THE FOLLOWING STEPS SHOULD BE FOLLOWED:

1. SAVE ALL SCHEMAS TO TAPE. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18, THIS STEP WILL BE REPEATED AGAIN AFTER INVOKING DBACP'S CONVERSION FOR EACH SCHEMA SO THAT YOU WILL NEVER HAVE TO RESTORE AN UNCONVERTED SCHEMA FOR RECOVERY PURPOSES.

2. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18.2 USE FUTIL TO DO A UFDDEL OF THE CURRENT TOP LEVEL UFDS DBMS AND DBMSLB. THIS WILL CLEAN OUT THE OUTDATED SOFTWARE AND AVOID CONFUSION OVER WHICH VERSION IS CURRENT.

3. ATTACH TO EACH OF THE NON-BINARY UFDS RESTORED FROM TAPE (DBMSXXXX) AND:

   ```
   CO C_INSTALLDBMSXXXX
   ```

4. COPY THE ERROR MESSAGE FILES DAERRS, DBMSE, FDMLER, CDMLER FROM DBMSXXXX>PDBMS TO THE UFD PDBMS. (THIS ONLY NEEDS TO BE DONE ONCE. THEY ARE THE SAME IN ALL THE DBMSXXXX>PDBMS UFDS.)

5. SHARE DBMS FROM THE SYSTEM CONSOLE THUS:

   ```
   CO SYSTEM>C_SHAREDBMS (SEE SECTION DMLCP INSTALLATION)
   ```

6. WITH DBACP SET THE CONCURRENCY/RECOVERY ATTRIBUTES DESIRED WITH THE NEW DBACP COMMANDS.

7. IF THIS IS THE FIRST TIME YOU HAVE UPGRADED TO REV 18, FOR EACH SCHEMA, INVOKE DBACP AND 'VERIFY SCHEMANAME'.

   DBACP WILL AUTOMATICALLY CONVERT THE SCHEMA TO REV 18 COMPATIBLE FORMAT (AND SO NOTIFY USER). REPEAT STEP 1, SAVING ALL CONVERTED SCHEMAS TO TAPE.

DATA ADMINISTRATOR AUTHORIZATION

THE FILE PDBMS>DALIST CONTAINS THE LOGIN NAMES OF ALL PERSONS AUTHORIZED AS VALID DATA ADMINISTRATORS. WITHOUT SUCH AUTHORIZATION, A USER MAY NOT USE ANY OF THE DBACP COMMANDS WHICH ALTER A DATABASE OR DISPLAY SENSITIVE INFORMATION (SUCH AS PRIVACY KEYS). DALIST IS ORGANIZED WITH ONE LOGIN NAME PER LINE. NAMES MAY BE ADDED OR DELETED USING ED. IF A LINE IS LEFT BLANK IT IS IGNORED. INITIALLY, THE FILE IS EMPTY. THE FIRST THREE LINES OF DALIST CONTAIN THE LOGIN NAMES OF THE PRIVILEGED DATA ADMINISTRATORS. THESE ARE DATA ADMINISTRATORS WHO MAY BYPASS THE VARIOUS SCHEMA PRIVACY LOCKS WHEN USING DBACP. A PRIVILEGED DATA ADMINISTRATOR WOULD BE RESPONSIBLE FOR THE MANAGEMENT AND INTEGRITY OF THE DBMS AS A WHOLE, INCLUDING THE MODIFICATION OF DALIST. SEE NEXT SECTION FOR PASSWORD PROTECTION OF PDBMS.

CHANGING THE DATABASE FILE UFD NAME AND PASSWORD

THE DATABASE ADMINISTRATOR CAN NOW CHANGE THE DEFAULT NAME AND PASSWORD FOR THE DATABASE FILE UFDS. THE CURRENT DEFAULTS ARE UFD PDBMS, WITH THE OWNER PASSWORD ISIS. TO DO THIS, EDIT THE FILE DBMSEX>INSERT>UFD-DATA.INS.FTN. CHANGE THE DATA STATEMENT FOR THE UFD NAME (VARIABLE ISIS) AND PASSWORD (VARIABLE ISPASS). THE UFD NAME AND PASSWORD ARE STILL LIMITED TO NO MORE THAN SIX CHARACTERS EACH. SEE THE FOLLOWING SECTIONS ON RELOADING PRODUCTS AND INTRODUCTORY MESSAGE CONTROL. THEN RELOAD ILIB, SCHEMA, DBUTL, DBACP, FSUBS, CSUBS, FDML, CDML, SCHED, SCHDEC, IDBMS, AND CLUP (OR WHATEVER SUBSET OF THIS LIST WAS DELIVERED).

THE DATABASE ADMINISTRATOR MUST THEN RENAME THE EXISTING DATA BASE FILE UFDS AND CHANGE THEIR PASSWORDS.

INTRODUCTORY MESSAGE CONTROL

THE USER HAS THE ABILITY TO INHIBIT THE PRINTING OF AN INTRODUCTORY MESSAGE AT RUN-TIME. IN THE INSERT FILE DBMSEX>INSERT>FUNIT-DATA.INS.FTN, IF THE VARIABLE "INTROM" IS SET TO .TRUE., AN INTRODUCTORY MESSAGE WILL BE PRINTED WHEN DBMS IS INVOKED. IF ITS VALUE IS .FALSE., NO MESSAGE WILL BE PRINTED.

TO SUPPRESS THE PRINTING OF THE INTRODUCTORY MESSAGE, EDIT THE FILE DBMSEX>INSERT>FUNIT-DATA.INS.FTN MAKING THE DESIRED CHANGES. THEN REFERING TO THE SECTION RELOADING PRODUCTS, RELOAD DMLCP.

RELOADING PRODUCTS

THERE ARE TIMES WHEN A SPECIFIC SUB-PRODUCT OF DBMS NEEDS TO BE RELOADED, THAT IS THE SEGMENT DIRECTORY NEEDS TO BE CREATED ANEW. TO DO THIS IT IS POSSIBLE TO USE THE SAME JOB STREAMS WHICH WERE USED IN THE ORIGINAL BUILDING OF THE COMPONENTS OF DBMS. HOWEVER, SINCE THE ORIGINAL BUILD WAS RUN UNDER A DIFFERENT UFD STRUCTURE, THE CPL COMMAND FILES WILL NOT WORK AS THEY ARE. THE SIMPLEST WAY TO ALLEVIATE THIS PROBLEM IS TO CREATE THE FOLLOWING UFD STRUCTURE AND MOVE THE NECCESSARY FILES INTO IT BEFORE RUNNING THE LOAD. (THE CPL PROCEDURE MERGE.CPL IN DBMSEX>JOBS DOES THIS.)

```
DBMS             (NO FILES)

      DBMSLB     (COPY THIS FROM TOP LEVEL UFD  DBMSLB)

      INSERT     (COPY THIS FROM DBMSEX>INSERT)

      JOBS       (COPY THIS FROM DBMSEX>JOBS)

      BINARY     (TRECOPY INTO HERE ALL NECCESSARY UFDS
                  OF THE FORM XXXX.B FROM THE TOP LEVEL
```

UFDS OF THE FORM DBMSXXXXBIN. AS A MINIMUM GET THE ENTIRE DBMSEXBIN BECAUSE THE LIBRARIES WILL BE NEEDED TO CREATE THE SHARED SEGMENTS AGAIN.)

```
DMLCP.B    (FROM DBMSEXBIN>DMLCP.B)

ILIB.B     (FROM DBMSEXBIN>ILIB.B)

SCHEMA.B   (FROM DBMSDEFBIN>SCHEMA.B)

 ETC.
```

NOW ATTACH TO DBMS>JOBS>LOAD AND FOR EACH SUB-PRODUCT WHICH NEEDS TO BE RELOADED RUN THE CPL PROCEDURE BY THE SAME NAME. NOTE THAT IF YOU ARE DOING ANY OF THE LIBRARIES (ULIB, CLIB, RLIB, ILIB, TEXTED, DMLCP, ASI, OR ASG), YOU SHOULD DO IT BEFORE ANY OTHERS SINCE THEY ARE INCLUDED IN THE OTHERS.

NOW ATTACH TO DBMS>DBMSLB. THE LOAD PROCEDURES FOR SUB-PRODUCTS PRODUCE SEGMENT DIRECTORIES OF THE FORM DB.XXXXX IN THIS UFD, SO YOU WILL NEED TO USE FUTIL'S TREDEL TO GET RID OF THE OLD VERSION OF THE SUB-PRODUCT(S) AND THEN CN TO PROMOTE THE NEW ONE.

BEFORE YOU CAN CREATE THE SHARED SEGMENTS BE SURE THAT THE PL1LIB IS IN TOP LEVEL UFD LIB. IF NOT, COPY IT FROM INDEX>SPL>LIB. NOW CREATE THE SHARED SEGMENTS THUS: CO DBMS>DBMSLB>C_LOAD.LIB. ALL THE NEW COMPONENTS ARE IN DBMS>DBMSLB AND YOU CAN USE FUTIL'S UFDCPY TO PROMOTE THEM TO THE TOP LEVEL UFD DBMSLB. WHEN THIS IS COMPLETE, THE ENTIRE UFD DBMS CAN BE DELETED TO RECOVER SPACE.

DMLCP INSTALLATION

DMLCP REQUIRES THE EXCLUSIVE USE OF SHARED SEGMENTS 2001, 2002, 2003 AND 2012 AND PRIVATE SEGMENTS 4030, 4031, 4032. TO INSTALL THE SHARED LIBRARY VERSION OF THE DML COMMAND PROCESSOR, THE FOLLOWING COMMAND MUST BE EXECUTED FROM THE SYSTEM CONSOLE AFTER EVERY COLD START:

```
CO SYSTEM>C_SHAREDBMS
```

THIS COMMAND STREAM INSTALLS THE DBMS SHARED LIBRARY, SHARES AND INITIALIZES THE DBMS SEGMENTS, AND INITIALIZES THE RING 3 SEMAPHORES. THIS COMMAND SHOULD BE INCORPORATED INTO C_PRMO, THE COMMAND FILE WHICH IS ALWAYS RUN AFTER A COLD START.

CREATION OF A DML APPLICATION PROGRAM

ONCE A SCHEMA HAS BEEN WRITTEN AND COMPILED AND A SUBSCHEMA HAS BEEN WRITTEN AND COMPILED, AND THE DATA BASE FILES HAVE BEEN ALLOCATED WITH DBACP, THE USER CAN WRITE APPLICATION PROGRAMS FOR THE DATA BASE IN EITHER COBOL OR FORTRAN. THE SEQUENCE USED TO TRANSFORM THE SOURCE CODE INTO EXECUTABLE CODE IS AS FOLLOWS:

(1) PREPROCESS THE SOURCE CODE WITH THE HOST LANGUAGE PREPROCESSOR (CDML OR FDML).

(2) COMPILE THE OUTPUT OF THE PREPROCESSOR (D_XXXXX) WITH THE HOST LANGUAGE COMPILER (COBOL OR FTN).

(3) LINK THE BINARY OUTPUT OF THE COMPILER TO THE DML COMMAND PROCESSOR WITH THE SEGMENTED LOADER SEG.

SAMPLE JOB STREAMS TO DO THESE OPERATIONS WITH EITHER A COBOL OR FTN PROGRAM MAY BE FOUND IN UFD DBMSLB CALLED C_CDML, C_CLOAD, C_FDML, AND C_FLOAD. THESE JOB STREAMS ARE DESIGNED TO BE USED WITH THE EXEC UTILITY. FOR EXAMPLE, TO COMPILE AND LOAD A COBOL PROGRAM CALLED "PROG", EXECUTE THE FOLLOWING COMMAND:

```
EXEC DBMSLB>C_CDML PROG
```

THIS IS EQUIVALENT TO EDITING C_CDML AND C_CLOAD REPLACING EACH OCCURRENCE OF "&1" WITH "PROG", AND DOING A CO OF THE FILE.

TO COMPILE A FORTRAN PROGRAM, USE THE JOB STREAM C_FDML INSTEAD OF C_CDML AND C_FLOAD INSTEAD OF C_CLOAD.

THE OUTPUT FILES CREATED WHEN USING C_CDML OR C_FDML ON THE SOURCE FILE "PROG" ARE:

L_PROG - THE PREPROCESSOR AND COMPILER LISTINGS.

B_PROG - THE BINARY FILE OUTPUT BY THE COMPILER.

THE OUTPUT FILES FROM USING C_CLOAD OR C_FLOAD WITH PROGRAM "PROG" ARE:

M_PROG - SEG PROGRAM MAP.

#PROG - THE SEGMENTED RUN FILE.

THE RESULTING USER PROGRAM IS EXECUTED WITH THE COMMAND:

```
SEG #PROG
```

SUBJECT: DPTX PRODUCTS

RELEASE: 18.3

DATE: AUGUST 15, 1981

1 NEW FUNCTIONALITY

NONE.

2 BUGS FIXED

2.1 USER VISIBLE BUG FIXES

1. A DYNT INSTRUCTION WAS ADDED TO ALLOW REFERENCES TO BD$LST FROM RING 3.

   MODULES AFFECTED: BDVLIB.PMA

2. A BUG WAS FIXED WHERE A WRITE DATA STREAM WITH NULLS EMBEDDED WOULD CAUSE PT45DSC TO REDISPLAY THE LAST SCREEN, LOOSING THE DATA FROM THE WRITE COMMAND. (POLERS 32109)

   MODULES AFFECTED: PT45DSC.FTN

3. A BUG WAS FIXED WHERE PT45DSC WOULD REJECT LARGE DATA STREAMS WHEN LOGGED IN THROUGH TELENET. (POLERS 35025)

MODULES AFFECTED: PT45DSC.FTN

4. A BUG WAS FIXED WHERE PT45DSC WOULD CHANGE TERMINAL CHARACTERISTICS (I.E. XOFF, DUPLEX) AFTER EXIT FROM THE PROGRAM.

   MODULES AFFECTED: PT45DSC.FTN

5. A BUG WAS FIXED WHERE PT45DSC WOULD EXIT WITH A PROHIBITED FIELD ERROR MESSAGE ON THE FIRST TRANSACTION AFTER THE CLEAR KEY HAD BEEN DEPRESSED. (POLERS 45425)

   MODULES AFFECTED: PT45DSC.FTN

6. PT45DSC NOW ALLOWS 32 CHARACTER STATION NAMES AS ADVERTISED. PREVIOUSLY, ONLY 6 CHARACTERS WERE ACCEPTED.

   MODULES AFFECTED: PT45DSC.FTN

7. PT45DSC WILL NOW EXIT WITH THE MESSAGE 'TERMINAL ATTRIBUTE OVERFLOW' WHEN MORE THAN 256 ATTRIBUTE CHARACTERS ARE TRANSMITTED TO THE TERMINAL.

   MODULES AFFECTED: PT45DSC.FTN

2.2 INTERNAL BUG FIXES

1. A FIX WAS MADE TO REENABLE THE RECEIVER ON A 'BAD DIM DATA' ERROR.

   MODULES AFFECTED: BSCMAN.FTN

2. TM3270 WILL STOP SELECTING A DEVICE IF THE DEVICE IS POWERED OFF. TM3270 WILL NOW WAIT FOR A DE STATUS TO BE RECEIVED BEFORE SELECTING A DEVICE.

   MODULES AFFECTED: TM3270.FTN

3. THE PROGRAM ATTENTION KEYS (PA1, PA2 AND PA3) WILL NOW UPDATE THE VIRTUAL BUFFER.

   MODULES AFFECTED: VBTMPL.FTN, OWLDSC.FTN, PT45DSC.FTN

4. BSCMAN.FTN CAN NOW BE COMPILED WITH THE -XREF OPTION OF THE FTN COMPILER. TO ALLOW THIS THE INSERT FILES PUDCOM.INS.FTN AND PCB.INS.FTN HAD TO BE REMOVED FROM BSCMAN AND THE SETTING OF THE PRIORITY LEVEL AND THE PROCESS NAME ARE DONE IN

SUBROUTINES.

MODULES AFFECTED: BSCMAN.FTN, CHAP.FTN(NEW), DPTNAM.FTN(NEW)

5. BSCMAN WILL NOW CLEAN UP THE LINE AND DROP RTS AND DTR WHEN A FORCE LOGOUT IS DETECTED.

   MODULES AFFECTED: BSCMAN.FTN, DPTXON.INS.FTN

6. THE EMULATOR WILL NOW HANDLE THE CASE OF A SYNC CHARACTER IMBEDDED IN A TEXT FIELD (POLERS 43856,40122). PREVIOUSLY, THE EMULATOR WOULD SEE THIS AS AN INCOMPLETE TRANSMISSION.

   MODULES AFFECTED: EM3270.FTN

7. A BUG WAS FIXED WHERE THE ARRAY ATTLOC WAS BEING OVER INDEXED AND ZEROING OUT CONSTANTS WHICH AFFECTED ERROR RECOVERY (POLERS 36610).

   MODULES AFFECTED: PT45DSC.FTN

3 OUTSTANDING PROBLEMS

THIS SECTION INDICATES, AS OF AUGUST 15, 1981, ALL OUTSTANDING PROBLEMS WHICH ENGINEERING IS AWARE OF IN THE REV 18.3 DPTX SOFTWARE. PROBLEMS WHICH HAVE BEEN REPORTED VIA THE POLERS SYSTEM HAVE ASSOCIATED POLERS NUMBERS WITH THEM.

1. IN CASES OF THE HOST TIMING OUT (NOT RESPONDING WITH AN ACK OR NAK) TO TEXT MESSAGES SENT BY THE EMULATOR (DPTX/DSC), BSCMAN (INCORRECTLY) DOES NOT SEND ENQ'S EVERY THREE SECONDS. INSTEAD, AFTER 10 SECONDS, AN EOT IS SENT. THIS PROBLEM WILL ONLY OCCUR IF A HOST DOES NOT RESPOND IN A TIMELY MANNER TO TEXT MESSAGES FROM DPTX/DSC.

2. AS PART OF THE DPTX/DSC PRODUCT, PT45DSC TRANSMITS ONLY DATA THAT HAS BEEN MODIFIED. IN CERTAIN CASES THIS IS INCONSISTENT WITH THE 3277 IT EMULATES.

3. TM3270, COMMUNICATING WITH COMMAND DEVICES, AS PART OF THE DPTX/TSF AND DPTX/TCF PRODUCTS, USES CHAINED WRITES TO ACKNOWLEDGE RECEIPT OF READ MODIFIED DATA. THIS IS ACCEPTABLE TO THE 3271 MOD 2. HOWEVER, THE SECOND GENERATION BSC CONTROL UNITS (3274 AND 3276) DO NOT ALLOW THIS. BECAUSE OF THIS, IT IS

NOT POSSIBLE TO RUN DPTX/TSF WITH SECOND GENERATION CONTROL UNITS.

4. PT45DSC, RUNNING AS PART OF THE DPTX/DSC PRODUCT, WILL NOT RUN CORRECTLY AT 9600 BPS. BECAUSE THE PT45 IS A SLOW DEVICE COMPARED TO THE PRIME, THE PT45'S BUFFERS CAN BE OVERRUN, RESULTING IN 'BROKEN' SCREEN FORMATS AND SCRAMBLED MESSAGES (POLERS 29480). THE TEMPORARY SOLUTION TO THIS IS TO RUN THE PT45 AT 4800 BPS OR TO INSURE THAT THE DMQ SIZE FOR THAT LINE IS SET AT ITS DEFAULT VALUE.

5. IN THE DPTX/TCF PRODUCT, IF A HOST WRITES ALPHABETIC DATA INTO A FIELD DEFINED AS NUMERIC ONLY (WHICH IS LEGAL IN 3270 PROTOCOL), AND THIS DATA IS TRANSFERRED TO AN IBM CONTROL UNIT ATTACHED TO THE PRIME VIA DPTX/TSF, SUBSEQUENT UPDATES OF THE VIRTUAL BUFFER BY TCF WILL BE REJECTED. TCF WILL REPORT THIS PROBLEM AS HOST DOWN. AS A TEMPORARY WORK AROUND, THE APPLICATION CAN BE CHANGED SO THAT THE FIELD BEING WRITTEN INTO IS NOT NUMERIC ONLY. THIS CAN BE DONE BY CHANGING THE ATTRIBUTE SENT OUT TO DEFINE THE FIELD.

6. THE ERASE INPUT KEY USED ON A TERMINAL CONNECTED TO AN IBM CONTROL UNIT ATTACHED TO THE PRIME VIA DPTX/TSF WILL NOT CAUSE THE VIRTUAL BUFFER TO BE UPDATED. THIS IS ONLY A PROBLEM FOR THOSE USING TCF. (POLERS 35025)

7. USERS OF DPTX/TSF USING DISLOG YES AS A CONFIG OPTION SHOULD BE AWARE THAT TTYNOPING DOES NOT PREVENT THE AMLDIM FROM PERFORMING LOGOUT ABORT CHECKS WHEN DISLOG YES IS SPECIFIED. THIS IS A PRIMOS BUG, AND A POLERS HAS BEEN SUBMITTED. THE TEMPORARY WORK AROUND IS TO NOT SPECIFY DISLOG YES AS A CONFIG OPTION.

8. IF A DATA STREAM SENT TO DPTX/DSC INCLUDES A WRITE OR ERASE/WRITE COMMAND WITHOUT ANY WCC OR DATA FOLLOWING THE COMMAND, OWLDSC WILL TAKE AN ACCESS VIOLATION AND THE EMULATOR WILL CAUSE CERTAIN OF THE VIRTUAL BUFFERS TO BE OVERWRITTEN. IN ADDITION, A RESPONSE TO THIS DATA STREAM CAN TAKE UP TO 10 SECONDS. AS A TEMPORARY MEASURE, TO CORRECT THIS, INSURE THT THE DATA STREAM ALWAYS HAS A WCC FOLLOWING THE WRITE OR ERASE/WRITE COMMAND.

9. IN A WARM START CONDITION, IF A RING 3 USER IS ATTACHED TO DPTX AND IS CURRENTLY SELECTED, THE 'HARD LOCK' WILL NOT BE RESET. (AN 'UNLOCK' MESSAGE IS NOT SENT TO THE SELECTED USER.) THE WORK AROUND FOR THIS IS TO POWER THE TERMINAL OFF AND BACK ON AGAIN AND DEPRESS THE F15 KEY (RECOVER KEY). THE VIRTUAL BUFFER REMAINS UNCHANGED FROM THE LAST SUCCESSFULLY COMPLETED HOST TRANSMISION.

4 ENVIRONMENT

DPTX AT REV 18.3 REQUIRES A REV 18.3 PRIMOS AND SPOOLER.

5 INSTALLAZATION AND BUILD PROCEDURES

STANDARD.

*
674 (FED)

PRODUCT: FED

RELEASE: 1.1

DATE: 7TH MAY 1981

NEW FUNCTIONALITY

NONE.

PROBLEMS FIXED

BUG FIX (NO POLER) TO CONSTANT FIELD TERMINATOR HANDLING WHICH USED TO CAUSE DELETION OF UNUSED TERMINATORS.

FED'S FORMS HAVE BEEN RE-DESIGNED TO TAKE ADVANTAGE OF THE NEW PT45 DRIVER - YOU WON'T GET SEA-SICK WATCHING THE SCREEN ROLL ANY MORE !

OUTSTANDING PROBLEMS

NONE KNOWN.

ENVIRONMENT

FED 1.1 USES FORMS 19.0 (TO USE THE LATEST PT45 DRIVER), BUT CAN RUN WITH 18.2 FORMS.

INSTALLATION AND BUILD PROCEDURES

STANDARD.

*
675 (FORMS)

SUBJECT: FORMS

RELEASE: 18.3

DATE: 15TH OCTOBER 1981

ABSTRACT

THIS DOCUMENT CONTAINS INFORMATION ON RELEASE 18.3 OF THE FORMS SYSTEM. THIS RELEASE INVOLVES ONLY ONE BUG-FIX TO THE RUN-TIME PORTION OF THE PRODUCT.

NEW FUNCTIONALITY

NONE.

PROBLEMS FIXED

THE PROBLEM IN THE PT45 DRIVER THAT CAUSED THE TERMINAL TO ROLL EVERY TIME SOMETHING WAS WRITTEN TO LINES 25-48 HAS BEEN SOLVED AS FAR AS POSSIBLE. THE PROBLEM IS THAT WHENEVER AN ATTRIBUTE HAS TO BE WRITTEN TO THE TERMINAL, THEN THE LINE HAS TO BE ON SCREEN. THIS HAS BEEN MINIMISED BY ONLY WRITING ATTRIBUTES WHEN THEY ARE NON-DEFAULT.

*
676 (FTN)

SUBJECT: FTN

RELEASE: 18.3

DATE: 1981 AUGUST 17

1 NEW FUNCTIONALITY

NONE

2 PROBLEMS FIXED

2.1 COMMAND LINE SOURCE FILE NAME

THE COMMAND LINE PARSER DID NOT HANDLE BLANKS IN THE SOURCE FILE SPECIFIER.

2.2 COMPILE TIME FAULT

A SIMPLE SUBROUTINE WHICH USES THE -DEBUG OPTION AND WHICH USES THE SUBROUTINE NAME INCORRECTLY HAD CAUSED THE COMPILER TO HALT IN THE MIDDLE OF ITS PROCESSING.

2.3 COMPILE TIME FAULT

A SHORT PROGRAM IN WHICH A DIMENSION STATEMENT IS OMMITTED HAD CAUSED THE COMPILER TO LOOP ENDLESSLY.

3 OUTSTANDING PROBLEMS

3.1 POLERS 37636 AND 824502

THE COMPILER-GENERATED CROSS-REFERENCE LIST OMITS ALL VARIABLES WITH $ AS THE SECOND CHARACTER.

3.2 POLERS 30130

$INS IN LIEU OF $INSERT NEITHER GENERATES A COMPILE ERROR, NOR DOES IT INSERT A FILE INTO A PROGRAM.

3.3 POLERS 81994

THE STATEMENT " UX=U(1,1)+(M-1,1)" COMPILED WITHOUT ERRORS PRODUCING INCORRECT PROGRAM RESULTS.

3.4 POLERS 36980

A PROGRAM'S OBJECT OUTPUT HAS MULTIPLE INDIRECT ERRORS WHICH ARE DETECTED BY SEG.

3.5 POLERS 33631

BAD CODE IS PRODUCED IN A CASE OF FLOATING POINT COMPARISON (64R MODE).

3.6 POLERS 12484

THE "PARAMETER IS BETTER" WARNING MESSAGE IS OCCASIONALLY NOT ACCURATE.

3.7 POLERS 34908

IN 64V-MODE COMPILATIONS, THE DMIN1 INTRINSIC FUNCTION WILL NOT ACCEPT MORE THAN 4 ARGUMENTS.

3.8 POLERS 82611

THE CROSS REFERENCE LISTING FOR CERTAIN INTRINSIC FUNCTIONS IS GARBLED.

3.9 POLERS 35339, 82303 AND 82614

RELATIONAL OPERATORS COMPARING INTEGERS MAY PRODUCE CODE WHICH USES THE RESULT OF SUBTRACTING THE TWO INTEGERS. THE RELATIONAL VALUE IS WRONG IF THE SUBTRACTION CAUSES OVERFLOW.

3.10 POLERS 12490

THE MODIFICATION OF AN ARRAY BY AN ENCODE STATEMENT IS NOT REFLECTED IN THE CROSS-REFERENCE LISTING OR IN THE "NEVER GIVEN A VALUE" WARNING OF -DEBUG MODE.

3.11 POLERS 21112

IN 64V MODE, THE COMPILER DOES NOT ACCEPT A VARIABLE AS THE FORMAT SPECIFIER IN AN I/O STATEMENT. AN INTEGER VARIABLE, WHICH HAS BEEN ASSIGNED THE VALUE OF A FORMAT STATEMENT LABEL IN THE CURRENT PROGRAM UNIT, SHOULD BE ALLOWED.

3.12 POLERS 21197

THE CROSS-REFERENCE LISTING IS SOMEWHAT CONFUSING FOR EQUIVALENCED VARIABLES, BECAUSE IT INDICATES THAT THEY WERE SPECIFIED ON THE LINE NUMBER OF THE LAST SPECIFICATION STATEMENT, RATHER THAN ON THE ACTUAL LINE NUMBER.

3.13 POLERS 27520

CONTINUATION LINES ARE SOMETIMES A PROBLEM BECAUSE THE COMPILER MUST BE ABLE TO RECOGNIZE THE TYPE OF EACH STATEMENT BASED ON A SINGLE SOURCE LINE.

4 ENVIRONMENT

REQUIRES PRIMOS 18.0.

5 INSTALLATION AND BUILD PROCEDURES

STANDARD

*
677 (F77)

F77 REVISION 18.3

THIS FORTRAN COMPILER IS BUILT ON A REVISION 19.0 BASE, SO IT INCLUDES THE LARGE INCREASES IN COMPILE SPEED WHICH BEGAN WITH THE 17.8 RELEASE. THESE WERE ACHEIVED BY BOTH INCREASING THE INTERNAL EFFICIENCY OF F77 AND THAT OF THE SPL COMPILER USED TO BUILD IT. THIS RELEASE TAKES FULL ADVANTAGE OF THE SPL FEATURES -QUICK AND -NOCOPY. THE MASTER DISK UFD FOR F77 SOURCE NO LONGER CONTAINS THE CODE GENERATOR AND UTILITY SOURCE FILES. THESE ARE NOW CONTAINED IN THE BACKENDSRC PRODUCT DIRECTORY AND THUS THIS PRODUCT MUST BE BUILT BEFORE F77.

RUNTIME PERFORMANCE:

TWO AREAS OF USER-VISIBLE SPEED INCREASES IN PROCESSING OF DATA STATEMENTS THAT INITIALIZE ITEMS WITH IMPLIED DO-LOOPS AND ARRAY OPERATIONS IN I/O LISTS WHERE THE IMPLIED DO REPRESENTS A CONTIGUOUS AREA IN STORAGE.

IMPROVED CODE GENERATION:

SEGMENT-SPANNING CODE IS NOW GENERATED WHEN A SMALL ARRAY IS EQUIVALENCED TO A LARGE ARRAY IN A COMMON BLOCK WHICH SPANS A SEGMENT BOUNDARY.

IMPROVED DIAGNOSTICS AND ERROR HANDLING:

THE USER WILL GET AN APPROPRIATE ERROR MESSAGE IN THE FOLOWING CASES:

O CONCATINATING ANYTHING ELSE BUT CHARACTER SCALERS.

O COMPARING AN ARRAY TO A CHARACTER SCALER.

RESTRICTION ON COMMON BLOCKS:

THE FOLLOWING RESTRICTIONS ON COMMON BLOCK USAGE ARE DESIGNED TO PREVENT AN ITEM FROM BEING SPLIT OVER A SEGMENT BOUNDARY:

EVERY VARIABLE MUST BE OFFSET BY A MULTIPLE OF ITS ELEMENT LENGTH FROM THE START OF THE COMMON BLOCK. FUTHERMORE, IN THE CASE OF CHARACTER DATA, EACH VARIABLE OR ARRAY ELEMENT MUST HAVE A LENGTH THAT WILL DIVIDE EVENLY INTO THE LENGTH OF A SEGMENT(64K WORDS).

RESTRICTION ON USE OF OCTAL CONSTANTS:

SINCE BOTH THE ANSI FORTRAN CHARACTER SUBSTRING OPERATION AND PR1ME'S CONVENTION FOR OCTAL CONSTANTS USE THE COLON CHARACTER THE MEANING OF SOME USAGE IS AMBIGUOUS. THEREFORE, IF A PROGRAM UNIT INCLUDES ANY CHARACTER OR IMPLICIT CHARACTER STATEMENT, THEN IN ORDER TO USE OCTAL CONSTANTS ANYWHERE IN AN ARGUMENT LIST TO A FUNCTION, THAT FUNCTION MUST BE SPECIFIED IN AN INTRINSIC OR EXTERNAL STATEMENT.

BUGFIXES SINCE REVISION 18.2(LABELED 18.2.1)

A PREVIOUS RELEASE CALLED 18.2.1 CONTAINED A FIX TO THE FORTRAN PARSER TO ELIMINATE ERROR 230 WHEN USING SUBSTRING OPERATIONS ON CHARACTER DATA. THIS CHANGE IS ALSO INCLUDED IN THIS RELEASE.

COMMENTS OF THE FORM "/* TEXT" NEAR COLUMN 72 NOW WORK(POLERS 31490).

SQRT INTRINSIC NOW HANDLES COMPLEX*16 DATATYPE(POLERS 29613).

THE COMMAND LINE ABBREVIATION "-P" NOW SELECTS -PRODUCTION(POLERS 34449).

ERRORS IN $INSERT'ED FILES NOW ARE DISPLAYED ALONG WITH THAT FILENAME.

EQUIVALENCED VARIABLES NOW ARE SHOWN WITH THE SAME LOCATION IN THE LISTING.

SUBJECT: MIDAS RELEASE DOCUMENTATION
RELEASE: REV 18.3
DATE: NOVEMBER 19, 1981

1 NEW FUNCTIONALITY COMPARED TO REV 18.2

- THE R-MODE INTERLUDE FOR UMODE$ HAS BEEN REMOVED FROM KIDALB.

- THE UNUSED DYNTS CLOSE$, CLOS$$, AND OPEN$ HAVE BEEN REMOVED FROM THE LIST OF DYNTS WHICH MIDAS HANDLES.

- THE UPDAT$ ROUTINE NOW IGNORES THE FL$RET FLAG.

- IF USER B HAS A RECORD LOCKED AND USER A ATTEMPTS TO READ THAT RECORD WITH THE FILE OPEN IN I-O MODE, THE COMMUNICATIONS ARRAY IS UPDATED TO POINT TO THE LOCKED RECORD, WHICH IS NOT RETURNED. THIS CAUSES SERIOUS USAGE PROBLEMS SINCE MOST COBOL USERS ARE EXPECTING TO RETRY THE READ UNTIL THE RECORD IS UNLOCKED. AT REV 18.2, THE READ NEXT RECORD WILL RETURN THE RECORD FOLLOWING THE LOCKED RECORD. THIS UNEXPECTED CHANGE IN FUNCTIONALITY IS REVERTED TO ITS ORIGINAL WORKINGS BY THE PATCH TO PSF 35232 AND PSF 31189.

2 PROBLEMS FIXED

PSF 29965

SETTING THE ARGUMENT 'FLAGS' TO M$NR1W ON A CALL TO KX$CRE SETS THE FILE SYSTEM READ/WRITE LOCK TO 'SYS' RATHER THAN NR1W.

PSF 32492, PSF 41464, TAR 82853

CREATK AND MPACK REV 17.6 OR LATER DO NOT RUN AGAINST FILES WHICH HAVE JUST BEEN RUN THROUGH THE REMAKE UTILITY. THE MESSAGE STOP - REMAKE THIS FILE OCCURS WHILE RUNNING THE ABOVE UTILITIES.

PLEASE NOTE

REMAKE IS A UTILITY WHOSE SOLE USE IS TO CONVERT PRE-REV 16.0 FILES TO THE CURRENT FILE FORMAT. REMAKE IS NO LONGER SUPPORTED AND IS INCLUDED ON THE MASTER DISK ONLY AS A CONVENIENCE FOR CUSTOMERS CONVERTING FROM MIDAS REVISIONS NO LONGER SUPPORTED BY PR1ME COMPUTER, INC.

** BEWARE: REMAKE IS PRESENT ON THE MASTER DISK ONLY AS SOURCE CODE.

IF YOU CHOOSE TO MAKE THIS UTILITY AVAILABLE ON YOUR SYSTEM, IT MUST NOT BE USED AS A SUBSTITUTE FOR MPACK.

PSF 33126

WHEN DOING A KBUILD THAT ADDS ADDITIONAL PRIMARY KEY/DATA RECORDS TO A MIDAS FILE THAT ALREADY CONTAINS DATA, THE FIRST RECORD ADDED WILL SEEMINGLY BE GARBAGED, BUT MPACKING THE FILE WILL FIX IT.

PSF 35232, PSF 31189

IN COBOL, IF USER B HAS A RECORD LOCKED AND USER A ATTEMPTS TO READ THAT RECORD WITH THE FILE OPEN IN I-O MODE, USER A WILL RECEIVED THE LOCKED RECORD ERROR CODE. WHEN THE NEXT RECORD IS READ AND RETURNED, THE LOCKED RECORD IS UNLOCKED BY USER A, EVEN THOUGH IT WAS LOCKED BY USER B.

PSF 35865

PERFORMING A NEXT$ ON A PARTIAL KEY WITH FL$FST, FL$NXT, FL$PLW, AND FL$PRE SET OFF, PERFORMANCE DEGRADES SIGNIFICANTLY ON THE CALL TO MIDAS WHICH RETURNS ERROR 7, RECORD NOT FOUND. MIDAS IS READING EVERY RECORD UP TO THE END OF FILE, CAUSING A PERFORMANCE DEGRADATION PROPORTIONAL TO THE FILE SIZE.

PSF 40419

IN PARM.K AND PARM.K.INS.FTN, THE CONSTANTS M$DPFP AND PF$PRE ARE UNDECLARED.

PSF 40428

KBUILD ACCESSES FTN BINARY INPUT FILES INCORRECTLY, NOT SKIPPING OVER THE 16 BIT WORD COUNT IN THE FIRST WORD OF EACH RECORD.

PSF 40981

WHEN A RECORD IN BEING LOCK$ED AND THEN UPDAT$ED, AND MIDAS CALLS ARE MADE TO PROCESS A DIFFERENT FILE BETWEEN THESE TWO CALLS, A CONCURRENCY ERROR OCCURS ON A NEXT$ CALL FOLLOWING THE UPDAT$ CALL WHEN THE ARRAY IS RETURNED FROM THE UPDAT$ CALL USING THE FL$RET FLAG.

PSF 45069

BECAUSE KBUILD HAS USED FILE UNIT 1 FOR A TEMPORARY FILE WHEN BUILDING SECONDARY INDEX 1, FILE UNIT 2 FOR INDEX 2, ETC., USERS RUNNING KBUILD FROM A COMMAND FILE MAY HAVE EXPERIENCED PROBLEMS WITH KBUILD INTERFERING WITH THEIR RUNNING OF THEIR COMMAND FILE OR VICE-VERSA. KBUILD NOW ASKS THE SYSTEM TO ASSIGN ITS FILE UNITS.

3 NON-VISIBLE INTERNAL FIXES AND ENHANCEMENTS.

O ENHANCEMENTS TO THE SHARED COBOL LIBRARY REQUIRED THAT THE STARTING POINT OF THE MIDAS SHARED LIBRARY BE MOVED. IT NOW BEGINS AT LOCATION '4C000 OF SEGMENT '2014.

PLEASE NOTE

THIS ENHANCEMENT IS REQUIRED BY ENHANCEMENTS TO THE REV 18.3 SHARED COBOL LIBRARY. REV 18.3 COBOL REQUIRES THE USE OF REV 18.3 MIDAS AND CANNOT RUN WITH ANY EARLIER RELEASES OF MIDAS. THE REVERSE IS NOT TRUE.

O CHANGES TO CODE IN MPACK TO ENABLE PROPER ERROR HANDLING TO BE ACCOMPLISHED IN CERTAIN CIRCUMSTANCES.

4 OUTSTANDING BUGS

PSF 32195

THE MODIFY OPTION IN CREATK CORRUPTS THE SECONDARY KEY SIZES IF DEFINED AS ASCII AND WILL NOT PERMIT USERS TO CHANGE SUPPORT OF SYNONYMS.

*
679 (PRINET)

REV 18.3 CHANGES TO NETCFG

THREE UPDATES HAVE BEEN INCLUDED IN THE REV 18.3 NETCFG. THE FIRST TWO WERE PATCHED INTO REV 18.2 .

THE PASSWORDS USED BY FAM-II WERE INITIALLY CALLED RINGO-RINGO PASSWORDS. THEY ARE NOW CALLED NODE-NODE PASSWORDS, AND THE USER INTERFACE HAS BEEN UPGRADED TO REFLECT THIS.

ALL REFERENCES TO "NAMING SPHERES" HAVE BEEN REMOVED FROM THE USER INTERFACE SINCE THAT CONCEPT HAS NO MEANING UNDER REV 18 (BUT WILL AT REV 19).

WHEN THE -PASSWORD COMMAND LINE OPTION WAS USED, CHANGES TO NODE-NODE PASSWORDS COULD BE SPECIFIED ONLY IF THE USER FIRST RESPONDED AFFIRMATIVELY TO THE "REVIEW OLD NETWORK CONFIGURATION" QUESTION.

SUBJECT: NETLINK

RELEASE: REVISION 18.3

DATE: SEPTEMBER 25, 1981

1 NEW FUNCTIONALITY

1.1 IMPROVED INTERNATIONAL SUPPORT

NETLINK HAS BEEN UPGRADED TO SUPPORT THE FULL 1980 STANDARD FOR X.3 AND X.29. THIS SOLVES MANY PROBLEMS PREVIOUSLY ENCOUNTERED WHEN TRYING TO CONNECT TO HOSTS WHICH REQUIRED FULL SUPPORT OF THE INTERNATIONAL PARAMETER SET. SOME OF THESE PARAMETERS REQUIRED NEW COMMANDS TO SET UP THEIR VALUE. SEE THE SECTION BELOW ON NEW PROFILE COMMANDS.

IN ADDITION TO SUPPORTING THE COMPLETE INTERNATIONAL PARAMETER SET, NETLINK NOW SUPPORTS A COMPLETE SET OF CONNECT PACKET SPECIFICATIONS. THIS IS USED WHEN COMMUNICATING WITH HOSTS THAT HAVE CONNECT PACKET FORMAT REQUIREMENTS.

1.2 PROFILE COMMANDS

NETLINK NOW SUPPORTS A STANDARD SET OF PROFILE MODIFIERS WHICH CAN BE USED IN SEVERAL DIFFERENT SITUATIONS WITHIN NETLINK. A CIRCUIT'S PROFILE CONSISTS OF TWO DIFFERENT PARTS. THE OPERATIONAL PARAMETERS ARE THOSE WHICH AFFECT THE BASIC OPERATIONAL ENVIRONMENT. EXAMPLES OF OPERATIONAL PARAMETERS ARE THE POLLING RATE (CF POLL COMMAND), OR DEBUG OPTIONS. THE SECOND PART OF A CIRCUIT'S PROFILE IS THE SPECIFIC CONNECT FIELDS WHICH ARE USED TO ESTABLISH THE CONNECTION. EXAMPLES OF THESE ARE FACILITIES (CF FCTY COMMAND) AND USER DATA (CF DATA COMMAND). THE CONNECTION PORTION OF THE PROFILE IS ONLY USED

WHEN ESTABLISHING A CONNECTION TO A REMOTE HOST. THE OPERATIONAL PORTION OF THE PROFILE IS USED THROUGHOUT THE LIFE OF THE CIRCUIT.

PROFILE COMMANDS MAY BE USED IN THE FOLLOWING SITUATIONS:

AS COMMAND LINE OPTIONS TO THE NETLINK COMMAND:

```
    EX:  NETLINK -FCTY 1 1 -POLL 1
```

AS COMMANDS TO NETLINK

```
    EX:  NETLINK
         [NETLINK REV. 18.3]

         @ FCTY 1 1
           ------
         @ POLL 1
           ------
```

AS OPTIONS TO THE 'C' COMMAND TO NETLINK

```
    EX:  NETLINK
         [NETLINK REV. 18.3]

         @ C 999 99 -FCTY 1 1 -POLL 1
           --------------------------
```

WHEN THESE COMMANDS ARE USED AS OPTIONS ON THE NETLINK COMMAND LINE, THERE EFFECT IS TO MODIFY THE 'DEFAULT' PROFILE THAT EVERY CALL IS MADE WITH.

WHEN THERE COMMANDS ARE USED AS SUBSYSTEM COMMANDS TO NETLINK, THEY WILL MODIFY THE 'DEFAULT' PROFILE, AND ALSO THE OPERATIONAL PROFILE OF THE CURRENT CIRCUIT.

WHEN APPLIED AS OPTIONS TO THE 'C' COMMAND, THESE COMMANDS WILL ONLY MODIFY THE CONNECT AND OPERATIONAL PROFILE OF THE CIRCUIT BEING ESTABLISHED.

1.3 CURRENT 'PROFILE' COMMANDS

THE FOLLOWING OLD COMMANDS ARE NOW TREATED AS OPERATIONAL PROFILE COMMANDS AS DESCRIBED ABOVE:

```
    DEBUG <ON | OFF | DUMP> - CONTROL DEBUG PRINTOUT
    POLL <TENTHS_OF_SECONDS> - SET TERMINAL INPUT POLLING RATE
```

THE FOLLOWING OLD COMMANDS ARE NOW TREATED AS CONNECT PROFILE COMMANDS:

```
     DNIC <DATA_NETWORK_ID_CODE> - SET DNIC
     PORT <PORT_NUMBER>          - REMOTE PORT TO CONNECT TO
     FCTY <FACILITIES_BYTES>     - SET A FACILITIES FIELD
     PRID <PROTOCOL_ID_BYTES>    - SET THE PROTOCOL ID FIELD
     DATA <TEXT>                 - SET USER DATA FIELD, NO PARITY
     MDATA <TEXT>                - SET USER DATA FIELD,MARKED PARITY
```

THE FCTY AND PRID COMMANDS USED TO TAKE OCTAL INPUT. THIS HAS BEEN CHANGED TO BE DECIMAL INPUT. THIS IS TO STANDARDIZE THE USAGE OF NUMBERS IN NETLINK. ALL COMMANDS WHICH REQUIRE NUMBERS NOW REQUIRE DECIMAL NUMBERS.

IN ADDITION, THE FCTY COMMAND MAY HAVE ASCII MNEUMONICS FOR SPECIFIC FACILITY PARAMETER/VALUE BYTE PAIRS. CURRENTLY ON THE FOLLOWING MNEUMONICS ARE AVAILABLE:

CHARGE - SET REVERSE CHARGING

NO_CHARGE - SET NO REVERSE CHARGING

A 'NC' COMMAND TO NETLINK IS THE SAME AS SPECIFYING 'C <ADDRESS> -FCTY NO_CHARGE'.

1.4 NEW PROFILE COMMANDS

THE FOLLOWING NEW OPERATIONAL PROFILE COMMANDS ARE NOW PROVIDED:

ESCAPE <ESCAPE_CHARACTER>

THIS COMMAND ALLOWS A USER TO CHANGE THE ESCAPE SEQUENCE FROM 'CR @ CR' TO 'CR <CHAR> CR'. THIS IS USEFUL WHEN RUNNING NETLINK FROM A TELENET PAD WHICH WILL TRAP THE 'CR @ CR' BEFORE NETLINK HAS AN OPPORTUNITY TO INTERPRET IT.

SPEED <BITS_PER_SECOND>

BPS <BITS_PER_SECOND>

THIS COMMAND TELLS NETLINK HOW TO RESPOND FOR REQUESTS FROM THE REMOTE HOST FOR TERMINAL SPEED. SOME HOSTS REQUIRE THAT THEY HAVE A VALID TERMINAL SPEED OR ELSE THEY WILL HANG. NOTE THAT MOST HOSTS DO NOT EXPECT ANY VALUE GREATER THAN 1200 BPS. THIS DIRECTIVE SETS UP X.3 PARAMETER 11.

TTP <ID_NUMBER>

TTP <NAME>

THIS COMMAND TELLS NETLINK WHAT TERMINAL TYPE THE USER IS USING. THIS DIRECTIVE ONLY HAS AN EFFECT IN TELENET, AND ONLY TO A FEW TYPES OF HOSTS. THE KIND OF HOSTS WHICH READS TERMINAL TYPE ARE THOSE WHICH PERFORM THEIR OWN ECHOING AND CARRIAGE CONTROL. ALLOWED TERMINAL NAMES ARE:

```
     OWL           FOX
     BEEHIVE       VT50
     PRINT
```

NAMES ARE TRANSLATED INTO THE APPROPRIATE VALUE FOR TELENET PARAMETER 23. IF A NUMBER IS SPECIFIED, THEN THIS IS USED AS THE VALUE FOR TELENET PARAMETER 23.

MODE [REMOTE_ECHO | NO_REMOTE_ECHO]

THIS COMMAND IS EXTREMELY USEFUL FOR PRIME TO PRIME CONNECTIONS WHENEVER THE USER EXPECTS TO BE USING SERVICES WHICH PERFORM REMOTE ECHOING. EXAMPLES OF THIS ARE OAS AND SCREEN EDITORS. REMOTE_ECHO TRUNS ON THIS MODE, AND NO_REMOTE_ECHO TURNS OFF THIS MODE. THE ACTION OF REMOTE ECHO MODE, IS TO MODIFY NETLINK'S OPERATIONAL CHARACTERISTICS SO THAT IT OPERATES IN A CHARACTER AT A TIME MODE, WHENEVER THE TERMINAL IS PUT INTO HALF DUPLEX MODE. WHEN THE TERMINAL IS IN FULL DUPLEX MODE, NORMAL FORWARDING CHARACTERISTICS ARE OBSERVED.
NOTE THAT THIS MODE MAY DRASTICALLY INCREASE COSTS OVER PUBLIC DATA NETWORKS.

THE FOLLOWING NEW CONNECT PROFILE COMMANDS ARE PROVIDED:

LDATA <TEXT>

SAME AS DATA COMMAND, ONLY START THE USER DATA OVERLAYING THE PROTOCOL ID FIELD.

LMDATA <TEXT>

SAME AS MDATA COMMAND, ONLY START USER DATA OVERLAYING THE PROTOCOL ID FIELD.

TO <ADDRESS>

THIS COMMAND ESTABLISHES A DESTINATION ADDRESS. AS AN EXAMPLE 'C <ADDRESS>' AND 'C -TO <ADDRESS>' ARE THE SAME COMMAND. IF ENTERED AS A COMMAND TO NETLINK, THEN A COMMAND OF 'C' WITH NO ADDRESS WILL

CONNECT TO THE DEFAULT ADDRESS. WHEN THIS OPTION IS USED ON THE NETLINK COMMAND LINE, THEN AN AUTOMATIC CONNECT IS ISSUED TO THE REMOTE SYSTEM. IN ADDITION, WHEN THE CIRCUIT IS DISCONNECTED, NETLINK WILL PERFORM AN AUTOMATIC 'QUIT' COMMAND. THIS ALLOWS THE USER TO CONNECT TO REMOTE SYSTEMS, WITHOUT EVER ENTERING THE NETLINK SUBSYSTEM.

EXAMPLE:

```
NETLINK -TO REMSYS
[NETLINK REV. 18.3]

REMSYS CONNECTED
<USER SESSION WITH REMOTE SYSTEM>
<USER TYPES 'LOGOUT'>

REMSYS DISCONNECTED

OK, <USER NOW RETURNED TO THE LOCAL SYSTEM>
```

1.5 OTHER NEW COMMANDS

IN ORDER TO KEEP TRACK OF A USER'S PROFILE, A COMMAND HAS BEEN ADDED TO PRINT OUT THE PROFILE FOR A CIRCUIT OR TO PRINT OUT THE DEFAULT PROFILE.

PROFILE DEFAULTS

PROFILE

A NULL PROFILE COMMAND PRINTS OUT THE PROFILE OF THE CURRENT CIRCUIT. PROFILE DEFAULTS PRINTS OUT THE DEFAULT PROFILE USED FOR ALL NEW CONNECTIONS.

2 PROBLEMS FIXED

THE ENHANCEMENT TO COMPLETELY SUPPORT THE INTERNATIONAL X.3 PARAMETERS FIXES SEVERAL PROBLEMS WITH USING NETLINK TO NON-PRIME HOSTS. THESE INCLUDE MOST CASES OF CALL REJECTION OR CALLS HANGING DUE TO PROTOCOL ERRORS. THESE ENHANCEMENTS CORRECT PROBLEMS REPORTED IN TAR 32437. IN ADDITION, NETLINK CAN NOW HANDLE FILES LONGER THAN 32K BYTES. THIS PROBLEM HAS NO TAR NUMBER.

3 OUTSTANDING PROBLEMS

NONE KNOWN.

4 ENVIRONMENT

THIS PRODUCT REQUIRES PRIMENET REV. 16.2.

5 INSTALLATION AND BUILD PROCEDURES

STANDARD

*
686 (RPG)

INFORMATION PERTAINING TO REV18.3 RPG

DATE: MAY 4, 1981

TO: RPG USERS

SUBJECT: INFORMATION PERTAINING TO REV18.3 RPG

ABSTRACT

BUG FIXES

THE FOLLOWING TAR'S WERE FIXED AT 18.2 BUT NOT REPORTED AT THAT TIME:

28993, 30636, 36839, 37620.

THE FOLLOWING TAR'S WERE FIXED AT 18.3:

25770, 30148, 33259, 33608, 34309, 37619.

DESCRIPTIONS OF THESE TAR'S FOLLOW AFTER THIS ABSTRACT.

TAR'S FIXED

25770 - RPG NOW RECOGNIZES POSITIVE OVERPUNCH (A THROUGH I, <) IN INPUT FILES.

28993 - DIV WITH HALF-ADJUST NOW YIELDS THE CORRECT RESULT PRECISION.

30148 - PACKED CONTROL FIELDS NOW WORK.

30636 - DUPLICATE OF 28993.

33259 - OUTPUT RECORDS WITH NO OUTPUT FIELDS NOW DOES NOT CAUSE A RESTRICTED INST$ CONDITION.

33608 - DUPLICATE OF 25770.

34309 - AN UNSUCCESSFUL CHAIN NO LONGER FILLS A RECORD WITH 9'S.

36839 - DUPLICATE OF 28993.

37619 - UPDATE FILES READ SEQUENTIALLY NOW ALWAYS REFER TO THE CORRECT RECORD.

37620 - AN UNSUCCESSFUL SETLL NO LONGER FILLS A RECORD WITH 9'S.

REV 18.3 CHANGES TO NETCFG

THREE UPDATES HAVE BEEN INCLUDED IN THE REV 18.3 NETCFG. THE FIRST TWO WERE PATCHED INTO REV 18.2 .

THE PASSWORDS USED BY FAM-II WERE INITIALLY CALLED RINGO-RINGO PASSWORDS. THEY ARE NOW CALLED NODE-NODE PASSWORDS, AND THE USER INTERFACE HAS BEEN UPGRADED TO REFLECT THIS.

ALL REFERENCES TO "NAMING SPHERES" HAVE BEEN REMOVED FROM THE USER INTERFACE SINCE THAT CONCEPT HAS NO MEANING UNDER REV 18 (BUT WILL AT REV 19).

WHEN THE -PASSWORD COMMAND LINE OPTION WAS USED, CHANGES TO NODE-NODE PASSWORDS COULD BE SPECIFIED ONLY IF THE USER FIRST RESPONDED AFFIRMATIVELY TO THE "REVIEW OLD NETWORK CONFIGURATION" QUESTION.

SUBJECT: NETLINK

RELEASE: REVISION 18.3

DATE: SEPTEMBER 25, 1981

1 NEW FUNCTIONALITY

1.1 IMPROVED INTERNATIONAL SUPPORT

NETLINK HAS BEEN UPGRADED TO SUPPORT THE FULL 1980 STANDARD FOR X.3 AND X.29. THIS SOLVES MANY PROBLEMS PREVIOUSLY ENCOUNTERED WHEN TRYING TO CONNECT TO HOSTS WHICH REQUIRED FULL SUPPORT OF THE INTERNATIONAL PARAMETER SET. SOME OF THESE PARAMETERS REQUIRED NEW COMMANDS TO SET UP THEIR VALUE. SEE THE SECTION BELOW ON NEW PROFILE COMMANDS.

IN ADDITION TO SUPPORTING THE COMPLETE INTERNATIONAL PARAMETER SET, NETLINK NOW SUPPORTS A COMPLETE SET OF CONNECT PACKET SPECIFICATIONS. THIS IS USED WHEN COMMUNICATING WITH HOSTS THAT HAVE CONNECT PACKET FORMAT REQUIREMENTS.

1.2 PROFILE COMMANDS

NETLINK NOW SUPPORTS A STANDARD SET OF PROFILE MODIFIERS WHICH CAN BE USED IN SEVERAL DIFFERENT SITUATIONS WITHIN NETLINK. A CIRCUIT'S PROFILE CONSISTS OF TWO DIFFERENT PARTS. THE OPERATIONAL PARAMETERS ARE THOSE WHICH AFFECT THE BASIC OPERATIONAL ENVIRONMENT. EXAMPLES OF OPERATIONAL PARAMETERS ARE THE POLLING RATE (CF POLL COMMAND), OR DEBUG OPTIONS. THE SECOND PART OF A CIRCUIT'S PROFILE IS THE SPECIFIC CONNECT FIELDS WHICH ARE USED TO ESTABLISH THE CONNECTION. EXAMPLES OF THESE ARE FACILITIES (CF FCTY COMMAND) AND USER DATA (CF DATA COMMAND). THE CONNECTION PORTION OF THE PROFILE IS ONLY USED WHEN ESTABLISHING A CONNECTION TO A REMOTE HOST. THE OPERATIONAL PORTION OF THE PROFILE IS USED THROUGHOUT THE LIFE OF THE CIRCUIT.

PROFILE COMMANDS MAY BE USED IN THE FOLLOWING SITUATIONS:

AS COMMAND LINE OPTIONS TO THE NETLINK COMMAND:

```
     EX:  NETLINK -FCTY 1 1 -POLL 1
```

AS COMMANDS TO NETLINK

```
     EX:  NETLINK
          [NETLINK REV. 18.3]

          @ FCTY 1 1
            --------
          @ POLL 1
            ------
```

AS OPTIONS TO THE 'C' COMMAND TO NETLINK

```
     EX:  NETLINK
          [NETLINK REV. 18.3]

          @ C 999 99 -FCTY 1 1 -POLL 1
            --------------------------
```

WHEN THESE COMMANDS ARE USED AS OPTIONS ON THE NETLINK COMMAND LINE, THERE EFFECT IS TO MODIFY THE 'DEFAULT' PROFILE THAT EVERY CALL IS MADE WITH.

WHEN THERE COMMANDS ARE USED AS SUBSYSTEM COMMANDS TO NETLINK, THEY WILL MODIFY THE 'DEFAULT' PROFILE, AND ALSO THE OPERATIONAL PROFILE OF THE CURRENT CIRCUIT.

WHEN APPLIED AS OPTIONS TO THE 'C' COMMAND, THESE COMMANDS WILL ONLY MODIFY THE CONNECT AND OPERATIONAL PROFILE OF THE CIRCUIT BEING ESTABLISHED.

1.3 CURRENT 'PROFILE' COMMANDS

THE FOLLOWING OLD COMMANDS ARE NOW TREATED AS OPERATIONAL PROFILE COMMANDS AS DESCRIBED ABOVE:

```
    DEBUG <ON ▯ OFF ▯ DUMP> - CONTROL DEBUG PRINTOUT
    POLL <TENTHS_OF_SECONDS> - SET TERMINAL INPUT POLLING RATE
```

THE FOLLOWING OLD COMMANDS ARE NOW TREATED AS CONNECT PROFILE COMMANDS:

```
    DNIC <DATA_NETWORK_ID_CODE> - SET DNIC
    PORT <PORT_NUMBER>          - REMOTE PORT TO CONNECT TO
    FCTY <FACILITIES_BYTES>     - SET A FACILITIES FIELD
    PRID <PROTOCOL_ID_BYTES>    - SET THE PROTOCOL ID FIELD
    DATA <TEXT>                 - SET USER DATA FIELD, NO PARITY
    MDATA <TEXT>                - SET USER DATA FIELD,MARKED PARITY
```

THE FCTY AND PRID COMMANDS USED TO TAKE OCTAL INPUT. THIS HAS BEEN CHANGED TO BE DECIMAL INPUT. THIS IS TO STANDARDIZE THE USAGE OF NUMBERS IN NETLINK. ALL COMMANDS WHICH REQUIRE NUMBERS NOW REQUIRE DECIMAL NUMBERS.

IN ADDITION, THE FCTY COMMAND MAY HAVE ASCII MNEUMONICS FOR SPECIFIC FACILITY PARAMETER/VALUE BYTE PAIRS. CURRENTLY ON THE FOLLOWING MNEUMONICS ARE AVAILABLE:

```
    CHARGE - SET REVERSE CHARGING

    NO_CHARGE - SET NO REVERSE CHARGING
```

A 'NC' COMMAND TO NETLINK IS THE SAME AS SPECIFYING 'C <ADDRESS> -FCTY NO_CHARGE'.

1.4 NEW PROFILE COMMANDS

THE FOLLOWING NEW OPERATIONAL PROFILE COMMANDS ARE NOW PROVIDED:

ESCAPE <ESCAPE_CHARACTER>

THIS COMMAND ALLOWS A USER TO CHANGE THE ESCAPE SEQUENCE FROM 'CR @ CR' TO 'CR <CHAR> CR'. THIS IS USEFUL WHEN RUNNING NETLINK FROM A TELENET PAD WHICH WILL TRAP THE 'CR @ CR' BEFORE NETLINK HAS AN

OPPORTUNITY TO INTERPRET IT.

SPEED <BITS_PER_SECOND>

BPS <BITS_PER_SECOND>

THIS COMMAND TELLS NETLINK HOW TO RESPOND FOR REQUESTS FROM THE REMOTE HOST FOR TERMINAL SPEED. SOME HOSTS REQUIRE THAT THEY HAVE A VALID TERMINAL SPEED OR ELSE THEY WILL HANG. NOTE THAT MOST HOSTS DO NOT EXPECT ANY VALUE GREATER THAN 1200 BPS. THIS DIRECTIVE SETS UP X.3 PARAMETER 11.

TTP <ID_NUMBER>

TTP <NAME>

THIS COMMAND TELLS NETLINK WHAT TERMINAL TYPE THE USER IS USING. THIS DIRECTIVE ONLY HAS AN EFFECT IN TELENET, AND ONLY TO A FEW TYPES OF HOSTS. THE KIND OF HOSTS WHICH READS TERMINAL TYPE ARE THOSE WHICH PERFORM THEIR OWN ECHOING AND CARRIAGE CONTROL. ALLOWED TERMINAL NAMES ARE:

```
     OWL            FOX
     BEEHIVE        VT50
     PRINT
```

NAMES ARE TRANSLATED INTO THE APPROPRIATE VALUE FOR TELENET PARAMETER 23. IF A NUMBER IS SPECIFIED, THEN THIS IS USED AS THE VALUE FOR TELENET PARAMETER 23.

MODE [REMOTE_ECHO | NO_REMOTE_ECHO]

THIS COMMAND IS EXTREMELY USEFUL FOR PRIME TO PRIME CONNECTIONS WHENEVER THE USER EXPECTS TO BE USING SERVICES WHICH PERFORM REMOTE ECHOING. EXAMPLES OF THIS ARE OAS AND SCREEN EDITORS. REMOTE_ECHO TRUNS ON THIS MODE, AND NO_REMOTE_ECHO TURNS OFF THIS MODE. THE ACTION OF REMOTE ECHO MODE, IS TO MODIFY NETLINK'S OPERATIONAL CHARACTERISTICS SO THAT IT OPERATES IN A CHARACTER AT A TIME MODE, WHENEVER THE TERMINAL IS PUT INTO HALF DUPLEX MODE. WHEN THE TERMINAL IS IN FULL DUPLEX MODE, NORMAL FORWARDING CHARACTERISTICS ARE OBSERVED.

NOTE THAT THIS MODE MAY DRASTICALLY INCREASE COSTS OVER PUBLIC DATA NETWORKS.

THE FOLLOWING NEW CONNECT PROFILE COMMANDS ARE PROVIDED:

LDATA <TEXT>

SAME AS DATA COMMAND, ONLY START THE USER DATA OVERLAYING THE PROTOCOL ID FIELD.

LMDATA <TEXT>

SAME AS MDATA COMMAND, ONLY START USER DATA OVERLAYING THE PROTOCOL ID FIELD.

TO <ADDRESS>

THIS COMMAND ESTABLISHES A DESTINATION ADDRESS. AS AN EXAMPLE 'C <ADDRESS>' AND 'C -TO <ADDRESS>' ARE THE SAME COMMAND. IF ENTERED AS A COMMAND TO NETLINK, THEN A COMMAND OF 'C' WITH NO ADDRESS WILL CONNECT TO THE DEFAULT ADDRESS. WHEN THIS OPTION IS USED ON THE NETLINK COMMAND LINE, THEN AN AUTOMATIC CONNECT IS ISSUED TO THE REMOTE SYSTEM. IN ADDITION, WHEN THE CIRCUIT IS DISCONNECTED, NETLINK WILL PERFORM AN AUTOMATIC 'QUIT' COMMAND. THIS ALLOWS THE USER TO CONNECT TO REMOTE SYSTEMS, WITHOUT EVER ENTERING THE NETLINK SUBSYSTEM.

EXAMPLE:

```
NETLINK -TO REMSYS
[NETLINK REV. 18.3]

REMSYS CONNECTED
<USER SESSION WITH REMOTE SYSTEM>
<USER TYPES 'LOGOUT'>

REMSYS DISCONNECTED

OK,  <USER NOW RETURNED TO THE LOCAL SYSTEM>
```

1.5 OTHER NEW COMMANDS

IN ORDER TO KEEP TRACK OF A USER'S PROFILE, A COMMAND HAS BEEN ADDED TO PRINT OUT THE PROFILE FOR A CIRCUIT OR TO PRINT OUT THE DEFAULT PROFILE.

PROFILE DEFAULTS

PROFILE

A NULL PROFILE COMMAND PRINTS OUT THE PROFILE OF THE CURRENT CIRCUIT. PROFILE DEFAULTS PRINTS OUT THE DEFAULT PROFILE USED FOR ALL NEW CONNECTIONS.

2 PROBLEMS FIXED

THE ENHANCEMENT TO COMPLETELY SUPPORT THE INTERNATIONAL X.3 PARAMETERS FIXES SEVERAL PROBLEMS WITH USING NETLINK TO NON-PRIME HOSTS. THESE INCLUDE MOST CASES OF CALL REJECTION OR CALLS HANGING DUE TO PROTOCOL ERRORS. THESE ENHANCEMENTS CORRECT PROBLEMS REPORTED IN TAR 32437. IN ADDITION, NETLINK CAN NOW HANDLE FILES LONGER THAN 32K BYTES. THIS PROBLEM HAS NO TAR NUMBER.

3 OUTSTANDING PROBLEMS

NONE KNOWN.

4 ENVIRONMENT

THIS PRODUCT REQUIRES PRIMENET REV. 16.2.

5 INSTALLATION AND BUILD PROCEDURES

STANDARD

*
688 (PASCAL)

SUBJECT : PASCAL

RELEASE : REV. 18.3

DATE : SEPTEMBER 25, 1981

1) NEW FUNCTIONALITY

A) THE COMPILER HAS A NEW OPTION -MAP/-NO_MAP. THE -MAP OPTION IS DEFAULT AND WILL NOT MAKE ANY USER VISIBLE CHANGE. THE -NO_MAP OPTION WILL GENERATE A LISTING FILE THAT HAS ONLY THE PROGRAM IN IT. THE "MAP" THAT LISTS WHERE ALL THE VARIABLES ARE IN MEMORY WILL NOT BE PRINTED.

B) INTEGER IS NOW ALLOWED AS AN ARRAY SUBSCRIPT. THE ARRAY MUST BE AN EXTERNAL ARRAY. EXAMPLE: VAR A : ARRAY[INTEGER] OF INTEGER;

C) THE SYMBOL TABLE SIZE HAS BEEN DOUBLED.

D) /* */ HAVE BEEN ADDED AS COMMENT DELIMITERS.

E) A <$P> HAS BEEN ADDED TO GENERATE PAGE EJECTS IN COMPILE TIME LISTINGS.

F) A CTRL C WAS ADDED TO BE AN EOF CHARACTER FROM THE TERMINAL.

G) PASCAL NOW GIVES SEPARATE LINE NUMBERS TO INCLUDE FILES.

H) THE ALLOWABLE RECORD LENGTH FOR FILES OF RECORDS HAS BEEN DOUBLED TO BE 32K WORDS.

I) A "-TTY" FILE WAS ADDED TO ALLOW RESETTING OF A DATA FILE TO THE TERMINAL.

2) PROBLEMS FIXED

A) THE FOLLOWING POLERS HAVE BEEN FIXED:

28817 B READLN DOES NOT WORK WITH EOF FROM THE TERMINAL. AN EOF CHARACTER (A CTRL C) WAS ADDED TO SIGNIFY EOF FROM THE TERMINAL.

29264 B THE -EXTERNAL OPTION DID NOT WORK. THIS HAS BEEN CORRECTED.

29268 B A PAIR OF RESETS OPERATING ON THE SAME FILE, OPEN ON THE SAME FILE UNIT, USED TO CAUSE ERRORS. THIS HAS BEEN FIXED.

29278 B PASCAL NOW PUTS OUT SEPARATE LINE NUMBERS FOR

EACH INCLUDE FILE IN THE OUTPUT LISTING.

29419 C THE -RANGE OPTION DID NOT WORK. THIS HAS BEEN CORRECTED.

29462 B A FEATURE WAS ADDED THAT ALLOWS ONE TO USE "INTEGER" AS AN ARRAY SUBSCRIPT AS IN VAR A : ARRAY[INTEGER] OF INTEGER; ALL ARRAYS USING THIS FEATURE MUST BE EXTERNAL.

29481 B MULTIPLE WRITES TO THE TERMINAL OF NON-CHARACTER DATA CAUSES RUN-TIME ERRORS. THIS HAS BEEN CORRECTED.

29482 B THE CLOSE OF A TEMP FILE DOES NOT CLOSE THE TEMP FILE. THIS HAS BEEN CORRECTED.

29486 B A PROGRAM WRITING TO A FILE OF RECORD, WHERE THE RECORD WAS GREATER THAN 16K WORDS WOULD CAUSE AN ERROR. THIS HAS BEEN CORRECTED. THE MAXIMUM RECORD SIZE IS NOW 32K.

31065 C A WARNING WAS NOT FLAGGED WHEN A DOUBLE INTEGER TO INTEGER CONVERSION WAS DONE IN A FOR LOOP. THIS HAS BEEN CORRECTED.

31498 B A SPARSE CASE STATEMENT USED TO CAUSE SYMBOL TABLE OVERFLOW. THE SYMBOL TABLE HAS BEEN ENLARGED IN SIZE BY ABOUT TWO TIMES, SO THIS PROBLEM SHOULD BE ELIMINATED. WARNING : THE USE OF SPARSE CASE STATEMENTS SHOULD BE AVOIDED.

32825 B THE COMPILER ABORTS ON VERY LARGE PROGRAMS. THIS IS STILL THE CASE. THE USER SHOULD BE WARNED THAT IF HIS PROGRAM ABORTS ON A LARGE PROGRAM, THE USER SHOULD MODULARIZE IT, AND MAKE SOME OF THE ROUTINES EXTERNAL. THIS SHOULD SOLVE HIS PROBLEM.

32834 B CHR(X) WHERE X IS A LONG INTEGER CAUSES INCORRECT RESULTS. THIS HAS BEEN FIXED.

33809 B THE UPPER BOUND ON A FOR LOOP MUST BE EVALUATED ONCE, NOT EVERY TIME THE LOOP IS EXECUTED. THIS HAS BEEN CORRECTED.

33880 C RESET(INPUT,'DATA') CAUSES THE DEFAULT INPUT FILE TO BE TRANSFERRED TO A DATA FILE. NOW RESET(INPUT,'-TTY') WILL CHANGE THE DEFAULT INPUT FILE BACK TO THE TERMINAL. THIS ALSO WORKS FOR REWRITE.

33949 B RESETTING A BINARY FILE USED TO GENERATE P$AFCK ERRORS. THIS HAS BEEN FIXED.

35318 C A <$P> OPTION HAS BEEN ADDED WHICH GENERATES A PAGE EJECT IN THE COMPILED LISTING OF A PASCAL PROGRAM.

35323 A AN INTERNAL ERROR IS GENERATED WHEN TRYING TO USE A FILE OF RECORDS. THIS HAS BEEN CORRECTED.

35825 B A RESET OF A TEMPORARY BINARY FILE USED TO CAUSE ERRORS. THIS HAS BEEN CORRECTED.

35830 A AN ASSIGNMENT STATEMENT FAILS TO MODIFY THE TARGET VARIABLE WHEN USING COMPLEX IF STATEMENTS. THIS HAS BEEN CORRECTED.

35831 A INTERNAL DBG ERRORS WERE GENERATED WHEN REFERENCING A USER DEFINED TYPE. THIS HAS BEEN CORRECTED.

35840 C PASCAL DOES NOT INFORM THE USER WHEN AN ELEMENTARY DATA ITEM CROSSES A SEGMENT BOUNDRY. THIS IS STILL THE CASE. THE USER IS WARNED TO BE AWARE OF THIS PROBLEM WHEN WRITING PROGRAMS WITH LARGE DATA AREAS AND SHOULD MAKE SURE THAT HIS ELEMENTARY DATA ITEMS WILL NOT CROSS A SEGMENT BOUNDRY.

35841 A PASCAL PROCEDURES, LOADED FROM A LIBRARY WHERE THE FORCE LOAD FLAG IS NOT SET, WERE NOT LOADED PROPERLY. THIS HAS BEEN CORRECTED.

35845 B THE COMPILER ALLOCATES REDUNDANT SYMBOL NODES FOR USER DEFINED TYPES. THIS IS STILL THE CASE, BUT WITH THE INCREASED SYMBOL TABLE SPACE, THIS SHOULD NOT POSE A PROBLEM TO THE USER.

35848 A CHR(X) IS ONLY LEGAL WHERE 128<=X<=255. THIS HAS BEEN CORRECTED SO THAT 0<=X<=255 IS ACCEPTABLE.

36705 A SYMBOL TABLE OVERFLOW CAN BE ENCOUNTERED ON LARGE PROGRAMS. THE SYMBOL TABLE SIZE HAS BEEN ALMOST DOUBLED, SO THIS PROBLEM SHOULD BE LESS OF A LIMITATION.

36713 C THE LENGTH PARAMETER FOR WRITELN'S DID NOT WORK. THIS HAS BEEN CORRECTED.

36744 A S:=[0..255]-[160] GENERATED INCORRECT CODE FOR

SETS. THIS HAS BEEN CORRECTED.

36746 B THE PASCAL IN COMMAND DID NOT WORK CORRECTLY FOR LONG INTEGERS. THIS HAS BEEN CORRECTED.

36747 B PASCAL DID NOT GENERATE AN ERROR MESSAGE WHEN THE IN OPERATOR WAS USED ON A NON-SET TYPE. THIS HAS BEEN CORRECTED.

37325 B LONG INTEGER ARRAY ELEMENTS DID NOT WORK. THIS HAS BEEN FIXED.

37566 B THIS WAS A DUPLICATE POLER RELATING TO EOF. IT WAS FIXED.

37922 B PASCAL ROUNDS INCORRECTLY WHEN PRINTING THE RESULTS OF SOME REAL ARITHMETIC. THE -FRN OPTION CAN BE USED TO CORRECT THIS PROBLEM.

99999 - RESETTING TWO FILES ON THE SAME FILE UNIT USED TO CAUSE ERRORS. THIS HAS BEEN CORRECTED.

99999 - READING AN INTEGER FOLLOWED BY A CHARACTER USED TO MISS THE CHARACTER. THIS HAS BEEN CORRECTED.

99999 - ERROR MESSAGES HAVE BEEN MODIFIED TO FIT ON A TERMINAL SCREEN.

99999 - /* */ WERE ADDED AS COMMENT DELIMITERS.

99999 - THE "NON-STANDARD" ERROR MESSAGE WAS CHANGED TO A WARNING.

99999 - REWRITE (F,'DATA'); REWRITE (G,'DATA'); NOW CAUSES A RUNTIME ERROR INSTEAD OF ACCESS VIOLATIONS.

3) OUTSTANDING PROBLEMS

A) THERE ARE SOME OUTSTANDING POLERS LISTED IN THE ON-LINE POLERS DATA BASE.

4) ENVIRONMENT

A) T4IS COMPILER REQUIRES A REV 18 PRIMOS.

5) INSTALLATION AND BUILD PROCEDURES

A) THE BUILD REQUIRES SPL, PLP, FTN, AND PMA.

B) THE INSTALL AND SHARE ARE STANDARD.

*
689 (PL1G)

SUBJECT : PL1G
RELEASE : REV. 18.3

1. NEW FUNCTIONALITY

A. THE COMPILER HAS A NEW OPTION -MAP/-NO_MAP. -MAP OPTIONS IS DEFAULT AND WILL NOT MAKE ANY USER VISIBLE CHANGES. THE -NO_MAP OPTION WILL TURN OFF THE VARIBLE REFERANCE MAP AT THE END OF THE LISTING.

2. PROBLEMS FIXED

A. THE FOLLOWING POLERS HAVE BEEN FIXED SINCE REV 18.2:

20763 STRING BUILTIN FUNCTION NOW ACCEPTS CHARACTER STRING ARGUMENT.

32369 FLOAT BUILTIN FUNCTION NOW WORKS CORRECTLY.

29103 -RANGE OPTION WORK CORRECTLY IN 32I MODE.

28804 CONSTANT -(2**15) WORKS CORRECTLY, BUT -(2**31) WILL NOT BE SUPPORTED.

34194 PL1G NOW SUPPORTS LARGE LINESIZE I/O UP TO 2056 CHARACTERS.

33429 'COL' INPUT FORMAT HAS BEEN FIXED.

36271 GET LIST STATEMENT WILL SIGNAL THE EOF CONDITION WHEN READING AN EMPTY FILE.

27397 NO CODE WILL BE GENERATED FOR 'DO WHILE('1'B) STATEMENT.

31518 COMPARING BIT STRINGS IS NOT DEPENDENT ON POSITION ANYMORE.

32361 TRANSLATE BUILTIN FUNCTION WORKS FOR A SHORTER SECOND ARGUMENT.

32362 PUT LIST STATEMENT NOW HANDLES TWO SOURCE LINES CORRECTLY.

29102 FLOAT BUILTIN FUNCTION NOW ACCEPTS LITERAL ARGUMENT.

32675 PL1G WILL CONVERT FIXED DEC(7) DATA TO CHARACTER STRING CORRECTLY.

29981 NON-BUG.

37704 VARIABLES DEFINED ON EXT VAR OR INDEX NOW WORK CORRECTLY.

29320 FREE REQUEST IN EMPTY AREA.

33382 PL1G ATTACH TO ANOTHER UFD AND COULD NOT ATTACH BACK AFTER COMPILE.

32369 DCL WITH BUILT IN FUNCTION NOW WROKS CORRECTLY.

43826 A VARIABLE DEFINED ON ITSELF NOW FLAGS CORRECT ERROR.

29979 -NO_MAP OPTION ADDED TO CPMPILER.

30109 IS NOT A BUG BUT A USER ERROR. A PICTURE OF ALL 9'S CANNOT CONTAIN A BLANK CHARACTER. IT CAN ONLY CONTAIN A NUMERIC DIGIT.

36981 SHOWED THAT SEC2 WAS NOT INITIALIZED, HENCE UNPREDICTABLE RESULTS ORCCUR.

43838 READING OF A MIDAS FILE CREATED USING PL1G NOW WORKS CORRECTLY.

32366 FIRST DIGIT OF A VALUE GREATER THAN DEC(14) WAS DROPPED. THIS RESULTED BECAUSE THE LARGEST DECIMAL DIGIT THAT CAN BE PRINTED IS F(14,*). FROM NOW ON A SIZE ERROR WILL OCCUR AND THE VALUE WILL NOT BE PRINTED.

INTERNAL PRINTING OF PICTURE CLAUSE OF 9'S, WAS FIXED.

INTERNAL A SHORT CALL PROC WITH A RETURN IN IT DID NOT GENRTATI THE CORRECT CODE.

INTERNAL THE STMT POINTER (INIT(NULL( )) RUNS CORRECTLY.

INTERNAL THE STMT DCL BIGBITS(1000) BIT(64) NO LONGER

CAUSES AN ACCESS VIOLATION.

3. OUTSTANDING PROBLEMS

A. THERE ARE SOME OUTSTANDING POLERS LISTED IN THE ON-LINE POLERS DATA BASE.

4. ENVIRONMENT

A. THIS COMPILER REQUIRES A REV 18 PRIMOS.

5. INSTALLATION AND BUILD PROCEDURES

A. THE BUILD REQUIRES SPL.

B. THE INSTALL AND SHARE ARE STANDARD.

*
690 (POWERPLUS)

POWERPLUS INFO -- REV 18.3

CHANGES SINCE REV 18.2

THE FOLLOWING BUGS HAVE BEEN FIXED IN REV 18.3:

TAR 29304 CANNOT USE THE LAST BYTE OF A NON-MINIMUM OPTIONS FIXED LENGTH, NON-TEXT FILE DURING FILE CREATE. (ALSO 45078)

TAR 45077 PERFORMING A CREATE CHANGE ON A DESCRIPTOR WHICH WAS BIT STRING AND ODD NUMBER OF CHARACTERS OR ASCII AND AN EVEN NUMBER OF CHARACTERS CAUSED SUBSEQUENT FINDS TO BE INCORRECT. (ALSO, 29046)

TAR 34649 DURING CREATE CHANGE DIALOGUE, NUM6 VARIABLES WERE LISTED AS NUM5.

TAR 35888 IN A PROCEDURE, VARIABLES NAMES OF LENGTH 3 WERE NOT RECOGNIZED. (ALSO, 37512)

TAR 45084 FIND A 'XXX' AND B 'YY' WHERE A IS INDEXED (PARTIAL SPECIFICATION) AND B IS NON-INDEXED (FULL SPECIFICATION) DID NOT FIND SPECIFIED RECORDS.

TAR 45079 PROCEDURE FILE CREATED DURING A CREATE PUBLIC IS SAVED AS A CREATE PRIVATE.

TAR 35804 CHARACTER VARIABLES IN REPORT HEADING PRODUCE SPURIOUS OUTPUT WHEN ASSIGNED NUMERIC VALUES (IE, YOU MUST KNOW EXACTLY HOW MANY CHARACTERS SHOULD BE ALLOCATED WHEN VARIABLES ARE DEFINED.

TAR 27519 POWER EDITOR CORRUPTS A LONG PROCEDURE LINE. (ALSO, 27518)

TAR 37611 MAX TIME CALCULATION IN ERROR IF TIME PERIOD SPANS MIDNIGHT. (ALSO, 29358)

TAR 35697 TEXT IN REPORTS LIMITED TO 7 LINES.

TAR 27485 DECIMAL VARIABLE IN REPORT TITLE IS NOT RECOGMIZED.

TAR 30231 ALIAS IN PRINT COMMAND DOES NOT ACCEPT PERIODS.

TAR 35384 PRINT OF TEXT CAUSES TRUNCATION OF LAST TWO CHARACTERS.

TAR 31135 FILE OPTION OF PRINT COMMAND DOES NOT ACCEPT PASSWORDS. (ALSO, 32047)

TAR 35790 SELECT HANGS WHILE ANOTHER USER PERFORMS A CREATE.

TAR 82431 PROCEDURE WITH CREATE RETURNS COMMAND TO THE TERMINAL AT THE END OF CREATE, NOT THE END OF PROCEDURE FILE.

TAR 33057 WHEN PASSWORD OPTION IS USED, BLANK LINES APPEAR ON THE DESC COMMAND FOR ELEMENTS REQUIRING HIGHER SECURITY LEVELS.

TAR 82727 CREATE CHANGE DOES NOT ALLOW CHANGES TO THE LAST FIELD.

TAR 29579 CANNOT USE PARAMETER INSERTION INTO A FIND COMMAND UNLESS THERE IS A PERIOD ON THE FIND COMMAND LINE.

TAR 29580 CANNOT USE PARAMETERS ON A MULTI-LINE FIND STATEMENT.

TAR 34246 POWER CREATES UNWANTED SECONDARY KEYS 15,16,17 ON MINIMUM OPTIONS-NO TEXT CREATE.

TAR 37795 A 7 CHARACTER PROCEDURE NAME WILL OVERWRITE AN EXISTING 6 CHARACTER PROCEDURE NAME.

INTERNALLY REPORTED BUGS WHICH ARE FIXED IN 18.3:

- CHANGE USING SCREEN WILL NOT CHANGE MORE THAN XX RECORDS IN A SET.

- VALIDATION DOES NOT FUNCTION CORRECTLY DURING CHANGE USING SCREEN.

- DURING A CHANGE USING SCREEN THE LAST MIDAS INDEX WILL HAVE THE DATA CHANGED, BUT NOT THE KEY.

- IN REV 18.2, COMMENTS IN A PROCEDURE ARE ACCEPTED AS VALID INPUT IN ALL POWER SUBSYSTEMS. THIS IS CORRECTED IN REV 18.3. HOWEVER, THE COMMENT LINE AND ENSUING PROMPT WILL APPEAR ON THE TERMINAL (BUT NOT BE READ BY THE SUBSYSTEM.)

- PROTECT FOR USERNAME CAUSES NULLIFICATION OF THE INDIVIDUAL COMMAND FILE, IF IT IS THE FIRST PROTECT FOR COMMAND ISSUED.

DOCUMENTATION ADDITIONS

TAR XXXXX UNLESS THE USER HAS CREATED A HEADING FOR A FILE, ALL DESCRIPTORS FROM THE FILE WILL BE DISPLAYED USING POWER'S DEFAULT FORMATS AS LISTED BELOW:

DATA TYPE	DEFAULT DISPLAY
NUM1 (R*8)	-ZZZZZZZ.##
NUM2 (R*4)	-ZZZZZZZ.##
NUM3 (I*2)	-ZZZZZ
NUM4 (I*4)	-ZZZZZZZ.##
NUM5 (DECIMAL)	-ZZZZZZZ.##
NUM6 (COMP-3)	-ZZZZZZZ.##

IF THESE DEFAULT DISPLAYS ARE NOT DESIRED, THE USER SHOULD CREATE A HEADING (USING HEADING CREATE).

TAR 33638 A MIDAS SEARCH DESCRIPTOR MAY NOT BE ADDED OR HAVE ITS DATA TYPE CHANGED WITH THE ADD AND CHANGE OPTIONS OF THE

CREATE COMMAND. IF A USER DESIRES TO ADD A NEW SEARCH DESCRIPTOR (OR CHANGE A DISPLAY DESCRIPTOR TO A SEARCH DESCRIPTOR) OR CHANGE THE DATA TYPE OR LENGTH, THEY SHOULD PERFORM THE FOLLOWING STEPS:

1) DUMP ALL DATA TO A FILE.

2) DESTROY THE FILE IN POWER.

3) EXIT POWER AND TREDEL THE DATA FILE.

4) ENTER POWER AND RECREATE THE FILE AS DESIRED.

5) BATCH ADD THE DATA.

IF ANY DESCRIPTOR NAMES ARE BEING CHANGED FOR THE NEW FILE, THESE NAME CHANGES SHOULD BE MADE ON THE OLD FILE (USING CHANGE DESCRIPTOR OPTION) , ELSE DATA IN THOSE DESCRIPTORS WILL NOT BE ADDED.

IF A USER HAS DATA IN A FILE, THE METHOD DESCRIBED ABOVE SHOULD BE USED TO MODIFY ALL TYPES OF FILES.

*
691 (VISTA)

SUBJECT: DBMS/QUERY (VISTA)

RELEASE: REV 18.3

DATE: SEPTEMBER 29, 1981

THIS DOCUMENT OUTLINES THE OUTSTANDING PROBLEMS, ENVIRONMENT, AND INSTALLATION PROCEDURES FOR DBMS/QUERY (VISTA), THE DBMS QUERY REPORT WRITER. INCLUDED ARE STEP-BY-STEP INSTRUCTIONS FOR INSTALLATION AND DESCRIPTIONS OF THE CONFIGURATION FILE AND UFD STRUCTURE NECESSARY FOR INSTALLATION.

PROBLEMS FIXED

GENERAL

* WHEN A CPL &DATA CLAUSE RUNNING DBMS/QUERY ABORTS ABNORMALLY (I.E.

WHEN DBMS/QUERY REQUESTS FURTHER INPUT AND THERE IS NO &TTY STATEMENT), DBMS FILES ARE LEFT OPEN & THE TRANSACTION IS NOT TERMINATED (CLUP IS REQUIRED).

- * INCONSISTENT SELECTIONS ARE MADE WHEN A RECORD-TYPE HAS 2 OWNERS (SAME WHERE CLAUSE, SAME FROM CLAUSE IN SELECT). IN ONE CASE, ALL MEMBERS OF THE OWNER ARE SELECTED, AND IN THE OTHER CASE, ONLY OCCURRENCES ACTUALLY OWNED ARE SELECTED. (POLER # 37495)

- * A FATAL ERROR IN DBMS INVARIABLY CAUSES A DBMS "INFINITE ERROR LOOP" IN DBMS/QUERY. THIS LOOP IS CAUGHT AND TERMINATED, BUT CAUSES VERY ALARMING ERROR MESSAGES.

- * GIVEN THE COMMAND:
  "SE DI ALL FROM <REC-NAME> USING <FORMAT-NAME> <CR>",
  WHERE THE RESULTING TABLE IS EMPTY, DBMS/QUERY GETS AN INTERNAL ERROR (NULL POINTER REFERENCE) AND BOMBS OUT. (POLER # 36382)

REPORT GENERATOR

- * AN ITEM ON THE COVER IS SPECIFIED TO BE ON LINE 00. THE RG SHOULD NOT ACCEPT THIS, BUT IS ACCEPTING IT AND PLACING THE ITEM AT LINE 1.

OUTSTANDING PROBLEMS

GENERAL

- * WHEN A USER ABORTS A COMMAND LEVEL SORT BY HITTING <BREAK>, THE TABLE BEING SORTED IS DESTROYED. IF THE USER ABORTS AN RG SORT BY HITTING <BREAK>, SORT FILES ARE LEFT OPEN (EVEN AFTER THE END OF THE DBMS/QUERY SESSION).

REPORT GENERATOR

- * "LIST DET P1 = 5/0 <CR>" (IN A FORMAT) CAUSES:
  1) NO DIVIDE-BY-ZERO CONDITION TO BE RAISED.
  2) P1 TO BE PRINTED AS 0 IN REPORT.
  3) INCONSISTANT RECOVERY: IN SOME RUNS, DBMS/QUERY RETURNS TO DBMS/QUERY COMMAND LEVEL NORMALLY; IN OTHERS, A FATAL

DBMS/QUERY ERROR IS SIGNALLED AND THE USER IS THROWN OUT TO PRIMOS. THIS BUG IS CAUSED BY A PRIMOS LIMITATION.

- A ZERO QUOTED STRING (I.E. "(ZERO 'NULL')") SHOULD BE RIGHT JUSTIFIED SINCE IT IS REPLACING A NUMERIC STRING. IT IS CURRENTLY LEFT JUSTIFIED.

- IN LIST AND BLOCK DETAIL, IF EJECT AND A FOOTER ARE SPECIFIED IN A FORMAT, THE FOOTER IS NOT PRINTED AT THE BOTTOM OF THE PAGE WITHIN THE BOTTOM MARGIN (EXCEPT FOR THE LAST PAGE). IT IS PRINTED ON THE LINE IMMEDIATELY BELOW THE INFORMATION REQUESTED.

- A REPORT CONTAINS A PAGE HEADING WHICH HAS A LABEL FOLLOWED BY AN ITEM OF TYPE PIC 9999. THE ITEM IS PRECEDED BY EXTRA SPACE. (POLER # 36377)

ENVIRONMENT

TO USE REV 18.3 DBMS/QUERY, IT IS NECESSARY TO INSTALL A REV 18.3 DBMS. ALSO REQUIRED IS A REV 18.3 OPERATING SYSTEM (PRIMOS).

INSTALLATION INSTRUCTIONS

TO INSTALL DBMS/QUERY, TAKE THE FOLLOWING ACTIONS:

1) ATTACH TO THE MFD WITH THE OWNER PASSWORD AND RESTORE THE DBMS/QUERY TAPE (THIS WILL CREATE THE VISTA UFD AND ITS SUB-UFD'S).

2) MODIFY VISTA>C_INSTALLVISTA TO CONTAIN THE OWNER PASSWORDS NECESSARY TO MODIFY THE SYSTEM UFDS CMDNC0 AND SYSTEM. (IF THIS IS YOUR INITIAL INSTALLATION, MAKE THESE MODIFICATIONS TO VISTA> C_INITINSTALLVISTA.)

3) INSTALL REV 18.3 DBMS ON THE SYSTEM AS PER THE INSTRUCTIONS INCLUDED WITH THAT PRODUCT.

4) MODIFY (IF NECESSARY) THE CONFIGURATION FILE, VISTA>SYSTEM> VISTA.CONFIG (SEE ADDENDUM 2 OF THIS DOCUMENT FOR A FULL DESCRIPTION), TO FIT THE SYSTEM DBMS/QUERY IS BEING INSTALLED ON. THIS FILE WILL BE COPIED TO THE SYSTEM UFD; IT IS THE FILE IN SYSTEM WHICH IS ACTUALLY USED BY DBMS/QUERY. NOTE: IF A MODIFIED

CONFIGURATION FILE ALREADY EXISTS IN YOUR SYSTEM UFD, WHEN C_INSTALLVISTA IS RUN, IT WILL BE OVERWRITTEN. THE EASIEST WAY TO AVOID RE-MODIFYING THE CONFIGURATION FILE IS TO COPY THE CONFIGURATION FILE FROM SYSTEM TO VISTA>SYSTEM>VISTA.CONFIG *BEFORE* STARTING C_INSTALLVISTA.

5) RUN C_INSTALLVISTA (TYPE "CO C_INSTALLVISTA"). THIS WILL COPY THE NECESSARY UFDS AND FILES FROM THE VISTA UFD TO THE APPROPRIATE SYSTEM UFDS (SEE ADDENDUM 1 OF THIS DOCUMENT FOR A FULL DESCRIPTION OF THE VISTA UFD). IF THIS IS YOUR INITIAL INSTALLATION, RUN C_INITINSTALLVISTA INSTEAD. THIS WILL CREATE THE MASTER DBMS/QUERY UFD (VISTA*) IN THE MFD, CREATE THE DBMS/QUERY CATALOG IN THE VISTA* UFD, AND COPY THE NECESSARY UFDS AND FILES FROM THE VISTA UFD TO THE APPROPRIATE PLACES.

6) FROM THE SYSTEM CONSOLE, SHARE DBMS USING SYSTEM>C_SHAREDBMS.

7) FROM THE SYSTEM CONSOLE, SHARE DBMS/QUERY USING THE COMMAND FILE SYSTEM>C_SHAREVISTA. THIS WILL SHARE THE DBMS/QUERY SEGMENTS AND CONFIGURE THE SYSTEM BASED UPON THE CONTENTS OF THE FILE SYSTEM>VISTA.CONFIG.

8) COBOL AND FORTRAN SUBSCHEMAS WHICH ARE GOING TO BE ACCESSED BY DBMS/QUERY *MUST* BE DELETED AND RECOMPILED USING THE REV 18.3 COBOL AND FORTRAN SUBSCHEMA COMPILERS IF THE SUBSCHEMAS HAVE NOT BEEN COMPILED WITH POST-REV 18.1 SUBSCHEMA COMPILERS.

9) DBMS/QUERY IS NOW READY FOR USE AND MAY BE INVOKED BY TYPING THE COMMAND "VISTA" AT THE TERMINAL. INFORMATION ABOUT THE USE OF DBMS/QUERY MAY BE OBTAINED FROM THE REV 18.3 DBMS/QUERY MANUALS OR BY TYPING THE COMMAND "HELP" AFTER INVOKING THE SUBSYSTEM.

ADDENDUM 1: DBMS/QUERY UFD STRUCTURE

THE STRUCTURE OF THE VISTA UFD (WHICH RESIDES IN THE MFD AND IS CREATED WHEN THE VISTA TAPE IS RESTORED) IS AS FOLLOWS:

```
VISTA
        CMDNC0
                 VISTA     THE COMMAND WHICH INVOKES THE QUERY
                           REPORT WRITER SUBSYSTEM.

        SYSTEM
                 VI2070    THE SHARED CODE AND DATA OF DBMS/QUERY;
                 VI2073    THE ACTUAL RUNFILE TO BE SHARED.
                 VI2074
                 VI2075
```

VISTA_CONFIG — THE PROGRAM WHICH IS RUN TO SET DBMS/QUERY'S CONFIGURABLE PARAMETERS WHEN THE SYSTEM IS SHARED.

VISTA.CONFIG — THE FILE ACCESSED BY VISTA_CONFIG; CONTAINS THE CONFIGURATION PARAMETER VALUES.

VISTA*

- VISTA.ERR — THE FILE WHICH CONTAINS THE DBMS/QUERY ERROR MESSAGES.
- VISTA.LOG — THIS FILE WILL BE CREATED BY DBMS/QUERY IF A SYSTEM ERROR OCCURS; CONTAINS INFORMATION USEFUL IN FIXING THE PROBLEM.
- HELP — A UFD WHICH CONTAINS DBMS/QUERY'S HELP DATABASE.
- CATALOG — A UFD STRUCTURE FOR THE DBMS/QUERY CATALOGS. (CREATED BY THE COMMAND FILE C_INITINSTALLVISTA)

INFO — A UFD CONTAINING RUNOFF FILES TO HELP WITH THE INSTALLATION OF DBMS/QUERY.

NOTE THAT NEITHER C_INSTALLVISTA OR C_INITINSTALLVISTA (WHICH COPY EACH PORTION OF DBMS/QUERY TO THE APPROPRIATE UFD) DO NOT DEAL WITH THE PASSWORDS OR PROTECTION OF ANY OF THE FILES OR UFDS THEY COPY. THEREFORE, IT IS THE RESPONSIBILITY OF THE INSTALLER TO MAKE SURE THE NEW FILES/UFDS ARE PROPERLY PROTECTED AND THAT THE COMI FILES ARE CHANGED TO CONFORM TO THE PASSWORDS PRESENT ON THE SYSTEM DBMS/QUERY IS BEING INSTALLED ON. (THE SUGGESTED PROTECTION VALUES FOR THE VISTA* UFD ARE ALL ACCESS FOR OWNERS, NONE FOR NON-OWNERS; THE REQUIRED PROTECTION VALUES FOR ALL FILES COPIED TO THE SYSTEM AND CMDNCO UFDS ARE ALL ACCESS FOR OWNERS, READ-ONLY FOR NON-OWNERS.)

ADDENDUM 2: CONFIGURATION FILE FORMAT

THE DBMS/QUERY CONFIGURATION FILE, SYSTEM>CONFIG.VISTA, CONSISTS OF 16 LINES. EACH LINE MUST BE EXACTLY AS DESCRIBED IN THESE INSTRUCTIONS OR DBMS/QUERY CANNOT BE EXPECTED TO WORK PROPERLY. THE INFORMATION ON EACH LINE IS AS FOLLOWS:

LINE 1: THE NUMBER OF CHARACTERS PER LINE ON THE TTY DBMS/QUERY IS

RUN WITH. THIS NUMBER SHOULD BE 1 CHARACTER LESS THAN THE ACTUAL SCREEN WIDTH TO AVOID UNWANTED AUTOMATIC LINEFEEDS. (DEFAULT = 79) NOTE: THIS NUMBER SHOULD BE GREATER OR EQUAL TO 71 FOR OPTIMUM PERFORMANCE OF DBMS/QUERY.

LINE 2: THE NUMBER OF LINES PER SCREEN ON THE TTY DBMS/QUERY IS RUN WITH. THIS NUMBER SHOULD BE 1 LESS THAN THE ACTUAL SCREEN LENGTH TO ALLOW FOR THE SCROLLING PROMPT. (DEFAULT = 23)

LINE 3: THE NUMBER OF CHARACTERS PER LOGICAL LINE ON THE PRINTER DBMS/QUERY IS RUN WITH; THE NUMBER OF CHARACTERS ON THE LINE AFTER THE PRINTER HAS INSERTED ITS SIDE MARGINS. (DEFAULT = 108)

LINE 4: THE NUMBER OF LINES PER LOGICAL PAGE ON THE PRINTER DBMS/QUERY IS RUN WITH; THE NUMBER OF LINES ON THE PAGE AFTER THE PRINTER HAS INSERTED ITS TOP AND BOTTOM MARGINS. (DEFAULT = 47)

LINE 5: THE NAME OF THE MASTER DBMS UFD (WHERE THE SCHEMAS ARE STORED). (DEFAULT = 'PDBMS')

LINE 6: THE OWNER PASSWORD OF THE MASTER DBMS UFD. (DEFAULT = 'ISIS')

LINE 7: THE NAME OF THE MASTER DBMS/QUERY UFD. (DEFAULT = 'VISTA*')

LINE 8: THE OWNER PASSWORD OF THE MASTER DBMS/QUERY UFD. (DEFAULT = '')

LINE 9: THE OWNER PASSWORD OF THE DBMS/QUERY CATALOG UFD (WHERE THE PROCEDURES, FORMATS AND ABBREVS ARE STORED). (DEFAULT = '')

LINE 10: THE MASTER UFD OF THE DBMS/QUERY HELP SUBSYSTEM FILES. (DEFAULT = 'VISTA*')

LINE 11: THE OWNER PASSWORD OF THE MASTER HELP UFD. (DEFAULT = '') (NOTE: IF THE DEFAULT NAME, 'VISTA>VISTA*' IS USED AND THE MASTER DBMS/QUERY UFD NAME IS LEFT AS THE DEFAULT, THEN THE PASSWORDS OF THE MASTER HELP AND MASTER DBMS/QUERY UFD'S MUST BE THE SAME, SINCE THE UFD'S ARE THE SAME.)

LINE 12: THE DBMS/QUERY HELP UFD (THE ACTUAL DATA FILES OF THE HELP SUBSYSTEM RESIDE HERE. (DEFAULT = 'HELP')

LINE 13: THE DBMS/QUERY HELP SUBSYSTEM UFD OWNER PASSWORD. (DEFAULT = '')

LINE 14: THE DBMS/QUERY HELP SUBSYSTEM TOPMOST LEVEL PREFIX. SINCE THE HELP SUBSYSTEM PRINTS THE ACTUAL UFD NAME WHERE IT IS CURRENTLY LOCATED, IT DELETES THE TOPMOST (PROTECTED) UFD NAMES AND THEIR PASSWORDS FROM THE HELP SUBSYSTEM HEADER.

THIS PREFIX REPLACES THE DELETED PORTION. (DEFAULT = 'HELP DBMS/QUERY')

LINE 15: THE SCROLLING DEFAULT: IF 'SCROLL ENABLED' IS TO BE THE DEFAULT, SET TO '1'B, IF 'SCROLL DISABLED' IS TO BE THE DEFAULT, SET TO '0'B. (DEFAULT = '1'B)

LINE 16: THE NUMBER OF VIRTUAL RECORDS RETRIEVED BETWEEN PRINTING THE VIRTUAL RECORD COUNT. (DEFAULT = 1) NOTE: IF DBMS/QUERY WILL BE USED WITH HARD-COPY TERMINALS, THIS NUMBER SHOULD BE SET TO A LARGE NUMBER (UPTO 32767). THIS WILL AVOID THE CONSTANT OVERWRITING OF THE VIRTUAL RECORD COUNT.

*
692 (VPSD)

SUBJECT: VPSD

RELEASE: 18.3

DATE: OCTOBER 24, 1981

THE FRAC INSTRUCTION, WHICH IS OBSOLETE AT REV 18.3, HAS BEEN REMOVED FROM VPSD'S TABLES.

THE STEX INSTRUCTION IS NOW RECOGNIZED PROPERLY.

*
693 (C_INSTALLSTD,C_SHLB)

C_INSTALLSTD - A COMMENT WAS REMOVED FROM A FUTIL COMMAND LINE.
THIS WAS IN RESPONSE TO POLER NO. 37082. ALSO ADDED THE

COPY OF HELP*, A NEW SYSTEM DIRECTORY.

C_SHLB - FILE WAS MODIFIED TO INCLUDE THE SHARE OF SPL.

```
C_CREATEALL -- CHANGED CREATE POWERCM TO
                       CREATE POWRCM
```