Edinburgh Regional Computing Centre

User Note 60

(February 1986)

Title:

EMAS-3: Changes from EMAS 2900 for All Users

Author: Roderick McLeod, John Murison, Keith Yarwood

Contact: Advisory service

Software Support Category: n/a

Synopsis

This Note describes the main differences between EMAS 2900 and EMAS-3.

Keyword

Conversion, command, command history, EMAS-3, filename

Edinburgh Regional Computing Centre

James Clerk Maxwell Building, The King's Buildings, Mayfield Road, Edinburgh, EH9 3JZ. Telephone 031-667 1081

Contents

Introduction

Very many features of EMAS-3 are identical or closely similar to those of EMAS 2900. These notes describe the essential differences and enhancements.

1. Naming of parts

Service	Operating System	Computer
EMAS	EMAS 2900	ICL 2976
BUSH	EMAS 2900	ICL 2988
EMAS-A	EMAS-3	Amdahl 470 V7

EMAS-A is the name of the service using the EMAS-3 operating system on the Amdahl 470 V7 computer.

2. Communications

Connect to the Amdahl V7 by calling service EMAS-A. From a PAD, log on in the usual way. From a TCP, respond to the *Host:* prompt with XCALL, then to the further prompt respond with EMAS-A.

On a PAD the handling of interrupts differs from that on a TCP. To interrupt your process when logged on via a PAD, press Ctrl-P followed by B. Then wait for the prompt *Int:*, which may be embedded in current output, if any. Then type the required single or multiple-character interrupt message, followed by RETURN. When using a TCP, press ESCAPE as on EMAS 2900. The *Int:* prompt appears on the next line: type RETURN in response to this prompt, and then await a further *Int:* prompt from the Amdahl front end. When this second prompt appears, then type the required interrupt message followed by RETURN.

3. File names

In a full filename, the user number is separated from the filename by a colon, not a dot. Thus: ERCC99:FILE not ERCC99.FILE

A new concept called a group is introduced. A group is an index of on-line files, whose names are prefixed by the group name. For example: AB.INFO1, AB.DATA are files which are members of group AB. Each on-line file can, but need not, be a member of a group. A group can also be a member of a group; thus the following are valid full filenames:

ABCD99:ALPHA.BETA.FILEA
EFGH22:JOSEPHINE27
IJKL75:GROUPA.RESULTS

Groups are created by using the command NEWGROUP and destroyed by using the command DESTROYGROUP. The command USEGROUP causes a particular group to be selected; thereafter all filenames referenced, other than "T#" files and files containing the ":" username separator, are considered to be prefixed with the groupname of the currently selected group. When a groupname is selected (by USEGROUP), files outside the current group are referenced by giving the full filename, except that ":" is a permitted abbreviation for "username:" for the process owner's username. User Note 61 contains a complete description of groups and their use.

4. Calling commands

There is no "bracket" option available when giving commands. Commands must be typed without spaces; thus, NEWPDFILE not NEW PD FILE. Parameters are separated from the command name by one or more spaces. All Subsystem commands use the new Parameter Acquisition Mechanism (PAM) which extends and unifies the way in which parameters can be given to commands. Commands may in general be typed on one line, as on EMAS 2900. However, if you type the command name followed by '?', e.g.

Command: LIST ?

then you will be prompted for each parameter in turn. If you type the command name followed by '??', e.g.

Command: LIST ??

you will first be given HELP information for the command and then prompted for each parameter. The parameter prompt consists of the parameter name followed by a colon, e.g.

File:

If there is a default value for the parameter, it is shown in brackets, e.g.

To(.OUT):

and if you simply hit the RETURN key, that default value will be passed to the program. To input a *null* value in response to a parameter prompt, type two double-quote symbols.

If you reply ? to a parameter prompt, you may receive further HELP information for the particular parameter. If you reply Ctrl-Y to a parameter prompt, control returns to command level.

Commands which on EMAS 2900 accept a ? as a parameter, for example

Command: DELIVER ?

will instead use = as the enquiry character, e.g.

Command: DELIVER =

No pre-processing of the command-line takes place before the parameters are extracted, except that if a comma is to be included as part of a parameter, then that parameter should be enclosed in double quotes. See the effect on DELIVER, below.

If an input command line finishes with a comma, the command interpreter prompts with a colon for further parameter input.

4.1 Wild-cards and multiple calls

The use of masks in the FILES commands has been generalized on EMAS-3 so that for example:

Command: CHERISH *S

can be used to CHERISH all your files with names ending in "S".

A related facility involves the use of "&" to cause multiple calls on a command. For example if you wish to OFFER three files to another user you could save typing by using this feature:

Command: OFFER S23 & DATA86 & LIST , ERCC87

These are two examples of features provided by the parameter acquisition mechanism (PAM). This mechanism can be used by your own commands so that they can too can be called in these ways if that is appropriate (see User Note 62 for details).

4.2 Command history mechanism

The EMAS 2900 User's Guide (pp 4-4 ff) explains the mechanism for EMAS 2900. The implementation details differ slightly on EMAS-3 as explained below. The following table summarizes the scheme for EMAS-3. Further details and examples are given in file SUBSYS:EXAMPLES.HISTORY on EMAS-A.

Display

!?	display store command-lines with their associated key numbers.

Repeat (without alteration)

!	repeat most recent command-line.
!3	repeat command-line having key-number 3.
!-1	repeat command-line one before the most recent (etc.)
!ABC	repeat most recent command-line commencing ABC.
!ABC!	repeat most recent command-line containing ABC.

Repeat (with alteration)

!ABC!DEF!	repeat most recent command-line containing ABC, replacing each occurrence of ABC with DEF. The replacement string (DEF in this example) may be null. The rightmost "!" may be omitted provided that the replacement string does not end in comma and is not intended to be null.
NEWCOMMAND!	perform NEWCOMMAND with same parameters as the previous command.
! NEWPARAMS	repeat most recent stored command with new parameters.

The above forms are the most commonly useful forms of repetition with alteration. The remainder are given for completeness.

NEWCOMMAND!3	perform NEWCOMMAND with same parameters as on command-line having key-number 3.

!3 NEWPARAMS	repeat command of command-line number 3 with new parameters.
NEWCOMMAND!ABD	perform NEWCOMMAND with the same parameters as on most recently-stored command-line commencing ABD.
!ABD NEWPARAMS	repeat most recent command-line commencing ABD with new parameters.
NEWCOMMAND!ABD!	perform NEWCOMMAND with the same parameters as on the most recently-stored command-line containing ABD.
!ABD! NEWPARAMS	repeat most recent command-line containing ABD with new parameters.
NEWCOMMAND!ABD!DE!	take most recent command-line containing ABD, replace each occurrence of ABD with DE, and perform NEWCOMMAND with the parameters from the resulting command-line.
!ABD!DE! NEWPARAMS	take most recent command-line containing ABD, replace each occurrence of ABD with DE, and perform the command of the resulting line with new parameters.

The principal differences from the EMAS 2900 implementation are:

* Textual searches for substitution are case-independent. Case is preserved in the substituted text.

* Spaces enclosed within exclamation marks have no special significance.

* Each command-line retains its unique identifying number while stored: the number is not dependent upon the position within the storage buffer.

4.3 The Command Macro interpreter

The Macro interpreter provides a convenient shorthand way of typing frequently used commands. It also allows you to create your own composite commands, without having to use a programming language. See User Note 81 for details.

5. Commands

Most of the commands described on the EMAS 2900 Information Card are available on EMAS-3. In general they work in the same way as before although several have been extended to accommodate file groups. Where filenames are used in commands, those names may be full filenames, or may be localnames which are prefixed according to the file group currently selected by command USEGROUP.
User Note 61, EMAS-3: Naming Files, gives definitions of these terms and ways of using filenames. The following sections indicate the main changes:

ALIAS - This command can only be used to provide an alias for a command (or to remove the alias(es) to a command). There is a separate command ALIASPROC which should be used to put an alias for a procedure entry into a directory.

ANALYSE - If you specify a filename for output then the command will fail if a file of that name already exists and is not a CHARACTER file. This is to reduce the chance of your over-writing a file in error.

ARCHIVE - This command works as on EMAS 2900, but note that it is now the only way to cause a file to be moved to the ARCHIVE store. The EMAS 2900 mechanism whereby CHERISHed files which remained unused eventually moved to the ARCHIVE store will not operate on EMAS-3.

COPY - This command has been extended to copy to and from nested members of partitioned files.

Command: COPY SUBSYS:VIEW_EMAS_COMMANDS , MYCOMMANDS

Command: COPY MYFILE , PD_HOLD_FILE23

It will also accept concatenated input files.

Command: COPY HEAD+BODY+TAIL , WHOLE

COPY now checks that if the output file already exists, that it is of the same type as the file you are trying to copy into it. This should reduce the chance of your over-writing the wrong file.

CPULIMIT - Note the use of

Command: CPULIMIT =

to display the current setting.

DEFINE - When output is directed, via DEFINE, to an existing file then a check is made that the type of the file is consistent. For example you cannot use SELECTOUTPUT on a channel which is DEFINEd to an existing OBJECT file. Note the use of

Command: DEFINE =

to get a list of current definitions.

DELIVER - The command interpreter no longer removes spaces and converts lower case to upper case by default. This means there is no need to enclose your text in double quotes. The only exception is where you wish to include a comma. Hence:

Command: DELIVER House of Commons

but

Command: DELIVER "10, Downing Street"

Note the use of DELIVER = to print out your current delivery information.

DETACH - You can now type all the parameters for DETACH on the same logical line. Remember that you can break a line after a comma if you wish to continue on a second line. Note that the keyword LOG replaces the two keywords OUT and OUTNAME.

DETACH is a command which has many optional parameters so it is convenient to use the keyword facility. For example:

Command: DETACH jobfile , 500 , log=jobout

Command: DETACH com+data17 , after=01/12/85

DETACHJOB - EMAS-3 does not support the Job Control Language of EMAS 2900. Use DETACH instead. See User Note 81 which describes the command Macro scheme.

DOCUMENTS - Now provides information about output, batch and transfer documents. For further information about any type of document or any individual document use one of these commands:

OUTPUT - for output documents (e.g. Printer files)

BATCH - for BATCH jobs

TRANSFERS - for files sent using TRANSFER

FILES - Extended to accommodate file groups. In output a file group is indicated by following the name with a dot. To find names of files (and groups) in a group then type a mask which is the name of the group:

Command: FILES accounts

Command: FILES conlib:doc

A new second parameter R should be used to obtain details of files which can be RECOVERed from the EMAS 2900 archive store (see below).

If the output from the FILES command is directed to an existing file then it must be a CHARACTER file.

FORT77 - This is replaced by the command FORTRAN. Note that the fourth parameter (which defines the destination of error messages) defaults to .OUT

IMP80 - This is replaced by the command IMP. Note that the fourth parameter (which defines the destination for error messages) defaults to .OUT

INSERT - This now takes two parameters: the name of an object file and the name of the directory into which you want to put the information. The latter defaults to the current active directory. So, if you use INSERT with only one object file it works as on EMAS 2900. On EMAS-3, however, the INSERT will be performed even if the directory is currently in use by other processes (connected in other virtual memories). To insert several object files, use the ampersand facility:

Command: INSERT OBJ1 & OBJ2 & OBJ3

NEWDIRECTORY - Rarely needed. Now takes two parameters: name and size in pages.

NEWGEN - Checks that both files are of the same type. Under exceptional circumstances it will not be possible to make this check.

OBEYJOB - EMAS-3 does not support the Job Control Language of EMAS 2900. Use OBEY instead. See also User Note 81 which describes the Macro scheme.

OPTION - ACTIVEDIR, SEARCHDIR and REMOVEDIR are now separate commands. So, for example:

Command: OPTION SEARCHDIR=PUBLICDIR

of EMAS 2900 becomes

Command: SEARCHDIR PUBLICDIR

SEARCHDIR takes an optional second parameter to specify the position you want to put the directory in the list. As on EMAS 2900 the default position is at the top of the list.

ACTIVEDIR and SEARCHDIR both accept a parameter of "=" to display the current settings. The parameter to REMOVEDIR may be the directory number as displayed by

Command: SEARCHDIR=

instead of the full directory name.

If you have several OPTIONs to change it is probably simpler to type:

Command: OPTION ?

and reply to the prompts. If the current setting of a particular option is correct just press RETURN. If you don't understand the meaning of a particular option reply "?"

REMOVE - Like INSERT, this takes two parameters: the name of the object file and the name of the directory from which you wish to REMOVE information. The latter defaults to the current active directory. If you just REMOVE one object file then the command works as on EMAS 2900, except that the REMOVE takes effect even if the directory is in use by other processes.

RESTORE - This command now accepts an optional third parameter to specify a new name for the RESTOREd file. This allows you:

to RESTORE a file with the same name as one in your on-line index

to RESTORE a file belonging to someone else into your index

SETMODE - This is designed primarily for use with terminals connected to PADs. It replaces XMODE on EMAS 2900.

XMODE - Now called SETMODE.

6. Transfer of EMAS and BUSH Files to EMAS-A

The TRANSFER command (see User Note 34) should be used to copy on-line files from either of the EMAS 2900 services to EMAS-A. Note that only CHARACTER, PARTITIONED and DATA files can be moved. PARTITIONED files may not be transferred out of EMAS-A since the file-format is not backwards compatible to EMAS 2900.

There are two ways to obtain access to files in the ARCHIVE stores of either BUSH or EMAS. The first method involves RESTORing them on the host machine and then using the TRANSFER command. A more convenient mechanism has been provided to enable you to access these files from EMAS-A. At the time you are allocated a user number on EMAS-A all the archive entries corresponding to the same number on EMAS or BUSH will be placed in the EMAS-A archive index, in a special category known as 'recoverable'. To find their names use.

Command: FILES ,R

There is a new command RECOVER which operates in a similar way to RESTORE in respect of these files. There is, however, one important difference. You can only RECOVER a file once. As soon as it has been RECOVERed, details of it are removed from the archive index. Note too that the files in this category include only the ones which were in the ARCHIVE store on BUSH or EMAS at the time you were accredited to use EMAS-A. Any files which go to the BUSH or EMAS archive after this time can only be moved to EMAS-A by the RESTORE and TRANSFER method.

7. Moving programs

All programs must be recompiled on EMAS-3.

7.1 Fortran77

Fortran77 programs should execute unchanged. The compiler is called FORTRAN, not FORT77. Note that the special routines EMASFC and FPRMPT are not required on EMAS-3 as the Subsystem interface is now language-independent (see section 7.3). An example is given in section 7.5.

The mechanism used by some 2900 Fortran programs to pass either a character string or an integer through the same parameter will not work on EMAS-3. It is not legal Fortran77.

7.2 IMP

The compiler is called IMP, not IMP80; the old IMP9 compiler is not available. IMP80 programs should execute unchanged unless they use any of the IMP language constructions which have been modified (see User Note 58), or use machine-dependent features, or call the Subsystem interface procedures. Most users will be unaffected by these changes.

The IMP Subsystem interface procedures are available and have the prefix S# (S-hashsymbol), e.g. S#DISCONNECT. Parameters are the same as for corresponding EMAS 2900 procedures, except that string value parameters corresponding to filenames are extended to length 255. Note that, on EMAS-3, %system %routine %spec should not be used. The way to give a reference or entry a different external identifier is to use the %alias construction of IMP. For example,

```
%external %routine %spec DISCONNECT %alias "S#DISCONNECT" %c
        (%string (255) S, %integer %name FLAG)
```

The IMP program text formatter SOAP80 (renamed SOAP on EMAS-3) has had a parameter, TOEMAS3, added; if TOEMAS3=Y is specified, any %external or %system %specs in the program are replaced by appropriate EMAS-3 specs; a warning is printed if the procedure name is not recognized.

User Note 84 explains how to write IMP procedures which are readily callable from other languages.

7.3 Subsystem interface with FORTRAN, PASCAL, C, IMP

The S# procedures are accessible from IMP only. The published Subsystem interface, however, comprises a set of procedures with name-type (reference) parameters which are accessible from FORTRAN and IMP, and from PASCAL and C when these languages are available. All procedures in this interface have entry-points prefixed EMAS3, e.g. EMAS3DISCONNECT.
A list of Subsystem procedure entry-points is given in file SUBSYS:EMAS3SPECS.

Users of the EMAS3... procedures from languages which do not allow external references to have long names will need to use MAKECOMMAND (described in User Note 62: Writing Commands) to modify entries and references in the object program as necessary, after compilation.

7.4 Length of filenames

Full filenames can be up to 255 characters in length on EMAS-3, because of nesting of files within groups within groups. See User Note 61, EMAS-3: Naming Files.
The maximum length of a file or group localname remains eleven characters.
Programs handling filenames should take account of this maximum length, since the Parameter Acquisition Mechanism delivers full filenames to calling programs.

7.5 Calling commands from programs

Programs which call commands in EMAS 2900, as for example in:

```
%externalroutinespec DEFINE(%string(255) s)
   :
DEFINE("1,.LP")
```

must be altered, for EMAS-3, to use a new procedure, EMAS3:

```
%externalroutinespec EMAS3(%string %name command, params,
   %integer %name flag)
   :
EMAS3("DEFINE", "1,.LP", FLAG)
```

(This is the IMP specification, but EMAS3, like all the EMAS3-prefixed entry-points, is accessible from other languages.) The variable FLAG is set to zero if the command was found and correctly executed. On EMAS-3, as explained below, the entry point for a command must begin with C#.

An example of a FORTRAN program calling a command follows:

```
      program perman
      character*128 c
      character*130 d
100   format(a128)
      call emas3prompt('Filename:')
      read(5,100) c
      d = c//',p'
      call emas3('analyse',d,iflag)
      end
```

8. Writing commands

Command entry-points are prefixed C# , e.g. C#LIST. The command interpreter prefixes the C# to the command name typed by the user before trying to locate the command..

On EMAS 2900, commands are written with a %string(255) parameter; each command-program extracts the parameters it requires from that string as necessary. Using the EMAS-3 Parameter Acquisition Mechanism (PAM), the parameters are in general acquired more easily and with the benefit of considerable pre-checking. Programs are smaller and a consistent appearance is provided to the terminal user. Command-writers are therefore strongly urged to study User Note 62: Writing Commands and to move to using PAM as soon as possible.

9. New facilities

9.1 Groups

GROUPs have been mentioned above under "File names" (section 3).

9.2 I/O channel allocation

A new Subsystem routine (accessible from all languages):

```
%external %routine %spec EMAS3CLAIMCHANNEL(%integer %name CHANNEL)
```

allows programs to avoid using "built-in" I/O channel numbers, with attendant difficulties where co-operating programs happen to choose the same channel number for different purposes. The channel number supplied may be used in subsequent calls of the DEFINE command, together with the integer-to-string conversion routine

```
%external %routine %spec EMAS3ITOS(%integer %name I, %string %name S)
```

as in the following example:

```
emas3claimchannel(newchannel)
emas3itos(newchannel, chstring)
emas3("define", chstring.",outputfile", flag)
```

The channel-number supplied is in fact negative, though this fact should not normally be of interest to the programmer. A large number of such channel numbers is available. The restriction of channel numbers to the range 1-80 of EMAS 2900 is lifted. Any positive integer is acceptable in EMAS-3 (in addition to the allocation of negative numbers as described above).

9.3 Virtual Storage Simulator (VSS)

This interface software, imported from the Michigan Terminal System (MTS) community, runs under the standard Subsystem and allows direct execution of imported load modules from IBM OS/MVS systems. If you think this facility may be useful to you, contact Advisory in the first instance.

10. Applications Software

Much of the library material on EMAS and BUSH has been transferred to EMAS-A. Most has been placed in the equivalent library, for example ERCLIB.GENERAL has become ERCLIB:GENERAL. Some material which was in CONLIB on the 2900 services has moved to ERCLIB. To find your favourite package or library try using HELP or ask the Advisory service. Major releases of software will be advertised in the ALERT information and the Computing Newsletter.

11. Graphics Packages

The following packages and utilities are available via directory ERCLIB:GRAPHICS.

GRAPHPACK - ERCC's graphical subroutines, callable from IMP or Fortran.

GRANALYSE - Analyses an RCO plotter file.

GRCONVERT - Converts an RCO plotter file for another plotter.

GPLAYOUT - RCO plotter file editor.

GPLIST - Output/convert RCO plotter file to plotter.

Additionally CALCOMPACK is available via the directory ERCLIB:CALCOMPICS.

Other Graphics software is being transferred. See HELP GRAPHICS for details.

12. Further information

The following User Notes contain details of particular aspects of EMAS-3.

58 Moving IMP programs from EMAS 2900 to EMAS-3

61 EMAS-3: Naming Files

62 EMAS-3: Writing Commands

65 Language-independent Event Trapping Mechanism on EMAS-3

80 EMAS-3: Subsystem Language Independent Programming Interface

81 EMAS-3: Command macro scheme

84 EMAS-3: Multi-language calls to IMP procedures

85 EMAS-3: Program Loader

Index