

## EHAS 2900 JOBBER NOTE

No: 1  
Date: 15/1/79

### The Scientific Jobber

The Scientific Jobber is a batch process which provides a fast, low-overhead environment for the development of ALGOL, FORTRAN and IMP programs. It incorporates a control language processor, a source program editor, a batch program loader and a set of standard mathematical procedures.

The Jobber was first developed for ICL's VME/B and VME/K operating systems, and its availability on EHAS 2900 provides a common environment for user programs across the three systems.

The speed and low overheads of the Jobber are achieved by making available a subset of the facilities provided by the full operating system, and by supporting this subset within a single program.

Input to the Jobber can be as a batch of cards assembled by the operators, or as a file of jobs sent from a user process.

### Operator's view

To the operator, a Jobber process is controlled as if it were a spooled device: there is a stream called JOBBER, for which "documents" are queued until requested by a Jobber process.

The appropriate subset of Spooler commands is as follows:

S/Q JOBBER	Display the state of all Jobber processes and list the names and estimated execution times of any files of Jobber jobs waiting to be run.
S/JOBRnn?	(JOBRnn identifies a particular Jobber process). Display the state of this Jobber process.
S/ABORT JOBRnn	Stop this process; if a batch of jobs is being executed, it will be re-appended to the Jobber queue.
S/DELETEnnnnnn	Delete a file from the Jobber queue, as identified by its unique six-digit identifier.

The single Director command required is:

D/START JOBRnn	Starts a Jobber process.
----------------	--------------------------

### User\_view

This is fully expounded in the ERCC 2980 User's Guide (Section F), and in the ICL 2900 series publication "Using the Scientific Jobber". In the latter, ignore references to system-dependent features such as libraries, and note that the EMAS 2900 Jobber runs in ISO mode.

There is an additional input route on EMAS 2900: the user can make up a file of jobs in a normal process and LIST or SEND it to the device ".JOBBER".

### Installation\_manager's\_view

Several Jobber processes can be set up, to a maximum of one per file system. For example, the ERCC 2970 currently has three Jobbers (JOB00, JOB01, JOB02), corresponding to the disc packs EMAS00, EMAS01, and EMAS02.

A Jobber process is a standard process with the following changes:

- a) The basefile is set to a fixed up Jobber subsystem. This is done by the command:

```
D/SET BASEFILE JOB0nn
D/user.filename'
```

- b) The ACR of the process must be less privileged than MANAGR and more privileged than its users, if installation options and level 2 facilities respectively are required (e.g. ACR 5 on the ERCC 2970).
- c) The maximum file size for the process must be raised, currently to 2 Megabytes; this figure may be reduced later.

When required, the process can be run with the Edinburgh Subsystem, by setting the basefile to a null string.

There exists a file in each Jobber process called JOBJOURNAL. This has READ permission granted. It contains a record of work. Specifically:

- a) For each batch of jobs accessed: the name of the file and the date and time of access.
- b) For each user job: the time it was begun, the CPU time used, the number of lines printed to primary output, and the number of sequential or direct access data record transfers.

When this file is full, it is listed and a new one created. It can be used as a basis for accounting outwith the standard EMAS 2900 accounting system. In the event of a major fault, this file can also be used as ultimate diagnostic output.

## Installation\_options

A number of installation management options are available with the Jobber. There are two levels of control:

- a) Options applicable to a particular batch of jobs (set by including a control card with the batch).
- b) Options applicable to all jobs run by a Jobber process (set by command from the MANAGR process).

Note: Installation defaults may be set for type (a) options from MANAGR, but are overridden by any batch control card.

## Options\_for\_a\_particular\_batch

The default values for the options applicable to a single batch of jobs can be modified by the insertion of a card (or line) beginning //JOBBER.

The options which may be modified are as follows:

### 1) Facility level

There are two facility levels for file access. These are specified by the FACLEVEL parameter, which can take the value 1 or 2 (the default). At Level 1 users are not allowed to access files and all input and output for such jobs must be via the Jobber control streams. At facility Level 2, in addition to the Level 1 facilities, creation of and access to source program and data files belonging to any process with a less privileged ACR is allowed (i.e. all normal users).

### 2) CPU limits

The default and maximum CPU limits for each user job can be modified, by the MDEFAULT and HMAX parameters respectively.

### 3) Printed output limits

The default and maximum printed output limits for each job can be modified, by the LDEFAULT and LMAX parameters, respectively.

### 4) File sizes

The maximum size permitted for a file created by a user can be modified (the FMAX parameter).

e.g.     JOBBER(FACLEVEL=2, HMAX=20)

### Options applicable to all jobs

- 1) The default compilation options may be modified by setting the parameter `OPTIONS`, which takes a value of one or more compilation options, separated by ampersands (see the ICL document "Using the Scientific Jobber").
- 2) The amount of printed output written to a Jobber output file can be controlled by the parameter `BREAK`. This parameter takes an integer value, which is a line count. Each output file will be closed and printed after the completion of the Jobber job during which this count was exceeded. A new file will be created and used for the output from subsequent jobs.

### Defaults

If no action is taken to set options then Jobber will be controlled by a set of defaults, fixed in the Jobber program as follows:

```
FACLEVEL = 2
MDEFAULT = 30
MMAX      = 300
LDEFAULT = 1000
LMAX      = 5000
FMAX      = X'200000' (bytes)
OPTIONS   = ISO&QUOTES
```

### Setting options from the MANAGR process

Three commands are available to override the default option values with which the Jobber program was compiled. (The file `ERSC04.JOPTY` must first be INSERTed in MANAGR's directory.)

- 1) To create a file in one or more Jobber processes which will contain default values to be used in preference to those used within the Jobber program (taken from file `JDEFAULTS`), do the following:

```
CREATE J OPTIONS (jobbers)
```

jobbers = the names of one or more Jobber processes, separated by  
commas

- 2) To change particular options, do the following:

```
SET J OPTIONS (Jobber)
```

The program will prompt for data, which should be given in the form:

DATA: KEYWORD = VALUE

The reply STOP causes a return to command level.

Keywords and permissible values are as follows:

<u>keyword</u>	<u>meaning</u>	<u>conditions on value</u>
FACLEVEL	Facility level	1 or 2
MDEFAULT	DEFAULT CPU time per job	10 <=MDEFAULT<=3000 sec
MMAX	Maximum CPU time per job	10 <=MMAX<= 3000 sec
LDEFAULT	Default lines output per job	50 <=LDEFAULT<=100,000
LMAX	Maximum lines output per job	50 <=LMAX<=100,000
FMAX	Maximum file size	10 <=FMAX<= 1024 Kbyte
BREAK	Output break limit	BREAK >= 100
OPTIONS	Compilation options	(See "Using the Scientific Jobber" or the ERCC 2980 User's Guide)

- 3) To list the current option defaults for a particular Jobber process, do the following:

PRINT JOPTIONS(jobber)

#### Future developments

- a) Dynamic loading of external library routines as an installation option.
- b) Direct-access data files.
- c) Shared compilers. Currently the Jobber has its own copy of the compilers, as does the Edinburgh Subsystem. It is possible for all compilations to share a single copy of the compilers, whether executing in a Jobber process or in a standard subsystem, and in background or foreground mode.
- d) Accounting. It may be possible to produce standard EMAS 2900 accounting records.

A. Anderson