SOFTWARE MODIFICATIONS DEVELOPED AT CERN IN THE IBM JOB ENTRY
SUBSYSTEM JES2 RUNNING UNDER MVS

(presented at the 1984 SEAS (Share European Association) Spring Meeting,
Knokke, Belgium, 9-13 April, 1984)
Contents

Acknowledgments .................................................. 2

Part 1: PRESENTATION OF THE CERN COMPUTER ENVIRONMENT ............... 3
  1. MAIN HARDWARE AND IBM SYSTEM SOFTWARE COMPONENTS .......... 3
     1.1 IBM HARDWARE CONFIGURATION ................................ 3
     1.2 IBM SOFTWARE CONFIGURATION ................................ 3

Part 2: Presentation of MVS/JES2 .................................. 7
  1. JES2 - MVS - 370 ARCHITECTURE ................................ 7
  2. EVOLUTION OF JOB MANAGEMENT AND JES2 ...................... 7
  3. THE VARIOUS FUNCTIONS PERFORMED BY JES2 ................. 7

Part 3: JES2 MODIFICATIONS AT CERN ........................... 9
  1. HIGH TAPE UTILIZATION ......................................... 9
  2. JOB ENQUIRY AND CONTROL FOR WYLBUR USERS ................ 10
  3. CERNET AND CDC INTERFACE ................................ 10
  4. CONTROL AND ACCOUNTING .................................. 11
  5. IMPROVED JOB SCHEDULING .................................. 12
     5.1 CERN Job Class hierarchy ................................. 12
     5.2 AUTOMATIC PRIORITY SCHEME ............................. 13
  6. CERN 'FETCH' QUEUE ......................................... 14
  7. CPU TIME NORMALIZATION .................................... 14
  8. DYNAMIC PROCLIB SUPPORT .................................. 14
  9. DISPLAY JOB COMMAND ENHANCEMENTS (SDJ) .................... 15
 10. NOTIFY FUNCTION ENHANCEMENTS ................................ 15
 11. OFFLINE FUNCTION ........................................... 16
 12. XEROX DEVICES SUPPORT ..................................... 16
     12.1 Central XEROX 8700 laser printer support ............... 16
     12.2 Remote XEROX 2700 printer support ..................... 16
 13. MICROFICHE DEVICE SUPPORT ................................ 16
 14. ADDITIONAL FACILITIES FOR OUTPUT JOBS ..................... 17
 15. DISPLAY CONSOLE ALLOWED (SDC) ................................ 17

Part 4: JES2 MAINTENANCE AT CERN .............................. 19
  1. PROPORTION OF 'CERN MODIFIED' JES2 CODE .................... 19
  2. JES2 PROBLEM DETERMINATION .................................. 19
  3. JES2 UPGRADE .................................................. 19
  4. TESTING ......................................................... 19
  5. REASONS FOR UPGRADE ......................................... 20

Appendix A: JES2 JOB PROCESSING ................................. 21
  A.1 INPUT ....................................................... 21
  A.2 CONVERSION .................................................. 21
A.3 EXECUTION .................................................. 21
A.4 OUTPUT .................................................. 21
A.5 PRINT/PUNCH ........................................... 22
A.6 PURGE ................................................... 22
A.7 SHARED-SPOOL ......................................... 22

Appendix B: TECHNICAL DESCRIPTIONS .................. 23
B.1 TAPE UNIT SCHEDULING .................................. 23
B.1.1 USER INTERFACE ...................................... 23
B.1.2 SYSTEM INTERFACE ................................... 23
B.1.3 OPERATIONAL INTERFACE ............................. 24
B.1.4 TAPE UNIT PROCESSING WITHIN THE LIFE OF A JOB 24
B.1.5 CHANGED CONTROL BLOCKS ........................... 25
B.1.6 CHANGED JES2 MODULES ............................... 25
B.2 CERN JOB CLASS HIERARCHY ............................ 25
B.3 CERN JOB PRIORITY SCHEME ........................... 26
B.4 HASPSSSM MODULE LOADED IN CSA (Common Storage Area) 26

Appendix C: JES2 PROGRAM ORGANIZATION ............... 27
Appendix D: BIBLIOGRAPHY ................................... 30

Tables
1. REAL AND LOGICAL TAPE UNITS NAMES ..................... 23
2. TAPE UNITS CONTROL RELATED COMMANDS .................. 24
3. CERN JOB CLASS STRUCTURE ................................ 25
4. PRIORITY LEVELS AND RELATIVE COST FACTORS ............. 26

Figures
1. CERN COMPUTER CENTRE EQUIPMENT ....................... 5
2. JES2 JOB PROCESSING FLOW CHART ........................ 29
SOFTWARE MODIFICATIONS DEVELOPED AT CERN IN THE IBM
JOB ENTRY SUBSYSTEM JES2 RUNNING UNDER MVS

This report reviews the various software modifications that have been developed at CERN in the IBM Package called 'JES2', responsible for job handling under the MVS System, operating on the CERN IBM mainframe. It should be stressed that these modifications proved necessary, and even mandatory given the CERN-specific users' community and computer environment.

The purpose of this paper is to provide a list of these modifications, as well as a justification and a brief description of how they are implemented. The aim is to reveal as non IBM standard some services of the CERN Job Entry Subsystem, and one might even point out that most users are not aware of the fact that various facilities of the job processing and job enquiry are 'CERN-specific'.

Author:
C.Magnin, Data Handling Division, Software Group.
CERN, Geneva CH 1211 Switzerland.

An earlier version of this report was presented at the 1984 SEAS (Share European Association) Spring Meeting, April 09-13 1984 Knokke Belgium.
I acknowledge the essential help of many colleagues in the Data Handling Division of CERN, who made substantial contributions to the work on JES2.

I am particularly indebted to B.Antoine, S.Jarp, O.Martin, A.Maver, L.Robertson and F.Toueg who gave me a useful assistance either during my learning period on JES or during the preparation of this paper.
PART 1

PRESENTATION OF THE CERN COMPUTER ENVIRONMENT

The CERN Computer Center provides the physics community with the computer services and output equipment necessary for their offline work. This includes numerical application packages as well as text and graphics data processing facilities.

1. MAIN HARDWARE AND IBM SYSTEM SOFTWARE COMPONENTS

1.1 IBM HARDWARE CONFIGURATION

The IBM service is based on two processors, namely:

- An IBM 3090 Model 200 with 64 megabytes of real memory, 64 megabytes of extended memory, and 2 CPUs.

- A SIEMENS 7890 S with 64 megabytes of real memory and 2 CPUs.
  These two processors have many features that enable them to be considered as the first IBM (or IBM compatible) machines to be specifically aimed at scientific and engineering needs for the next decade.

There are several Remote Job Entry stations and several other links to external laboratories, notably but not exclusively, Rutherford and Saclay. Another IBM processor, namely a 4341, is installed at CERN to function as the central node of the Swiss part of the European Academic and Research Network (EARN).

1.2 IBM SOFTWARE CONFIGURATION

1. MVS/JES2 OPERATING SYSTEM

The first operating system used on the IBM mainframe has been MVS (Multiple Virtual Storage), run in conjunction with JES2/NJE (Job Entry or Network Job Entry). The available compilers are FORTRAN (77,H,H Extended), PL/I, BCPL,C, ANSI Cobol, H and F assemblers and PASCAL.

2. ACCESS TO THE CENTRAL COMPUTERS

Access to the system is achieved through terminals, using the WYLBUR/MILTEN terminal support time-sharing system developed from a version coming from SLAC, the Stanford Linear Accelerator Center.
3. PERIPHERALS

Peripherals available under the IBM system include two high-quality laser printers, nine-track tape units handling densities up to 6250 bpi, cartridge tapes with a 38,000 bpi density, a KODAK Microfiche Device and Versatec electrostatic plotters.

User data sets are stored on disks of type 3350 having a capacity of 317 MBytes and also on the new fast, high quality model 3380 storing 600 MBytes per volume. The Mass Store (MSS) keeps data on magnetic strips stored in cartridges which are automatically found and loaded by the system and has a capacity of over 100 gigabytes.

Some of this space is used by the Hierarchical Storage Manager (HSM) for storing backup copies of inactive data sets while the rest of the volumes are allocated to individual groups for their private use.

4. VM/CMS OPERATING SYSTEM

The major consequence of the hardware upgrade with the SIEMENS 7890 in January 85 has been the introduction of a VM/CMS service at CERN, (which offers interactive facilities), run on the IBM 3090 together with MVS.

The sharing of the CPUs between the two services can be changed flexibly to suit the load.

5. ON-SITE COMMUNICATIONS - CERNET LOCAL NETWORK

In addition to the IBM facilities described above, the CERN Computer Centre includes CDC equipment based on a CDC 875 dual processor dedicated to batch work, a CDC 835 for interactive services and nine VAX systems.

On the CERN site computers can communicate with the central IBM and CDC systems through the network CERNET, used almost exclusively for file and job transfer.

The diagram on the next page summarises the computer center configuration.
Figure 1: CERN COMPUTER CENTRE EQUIPMENT
PART 2

*Presentation of MVS/JES2*

In order to get started, let us give a general framework for the information that will follow.

1. JES2 - MVS - 370 ARCHITECTURE

The Job Entry Subsystem 2 (JES2) serves as the entry point for jobs in MVS, which is one of the main operating systems that the IBM Company provides under the SYSTEM/370 Architecture; the other ones are DOS (DOS/VS-VSE), OS (OS/VS1,MVS) and VM (VM/370,VM/SP).

2. EVOLUTION OF JOB MANAGEMENT AND JES2

The evolution of Job Management went on with the introduction of Job Control Language (JCL), and OS Reader/Interpreter/Writer to read and interpret the JCL, then print or punch the output. The next step incorporated the Scheduling of Jobs via Class and priority, as jobs were queued before execution. Multiprogramming was then possible with the introduction of OS Multi Programming with a fixed number of Tasks (OS/MFT), and with a variable number of tasks (OS/MVT). In order to improve the speed of Input OS readers, as well as to enhance Job Scheduling, two appendages were built, namely:

- The Houston Automatic Spooling Priority (HASP),
- and the Attached Support Processor (ASP).

With the introduction of MVS, it was agreed that they would be integrated into OS. HASP became the basis for JES2, and ASP for JES3. They do much the same things, but in very different ways, (e.g. Spooling), because each is designed to support different environments.

3. THE VARIOUS FUNCTIONS PERFORMED BY JES2

As we stated earlier, JES2 serves as the entry point for jobs in MVS. JES2 is responsible for reading jobs into the system, scheduling the jobs, executing them, and handling their output from the system. At CERN, JES2 performs its functions within a Multi Access Spool (MAS) configuration, also called shared Spool, consisting of two MVS/JES2 systems supporting jobs that are located on the shared spool. This MAS complex itself is operating within a Network Job Entry (NJE) configuration consisting of more than 800 nodes joined together via communication lines. (All MVS/JES2 nodes transmit among themselves jobs, messages and SYSOUT data sets.)
PART 3

JES2 MODIFICATIONS AT CERN

The first natural question that arises when considering the subject of this report is to wonder why JES2 needed to be modified at CERN. One might sum up the various reasons that drove CERN to adopt a modified version of the Job Entry Subsystem by stating that given the very specific CERN physics community and environment, it seemed attractive to develop a tailored Job Entry Subsystem which could meet the users' requirements and which could also be in harmony with the surrounding world.

FIVE MAIN GROUPS OF MODIFICATIONS CAN BE DISCERNED:

1. One group is concerned about tape handling,
2. Another allows a suitable interface with WYLBUR and CERNET,
3. The third is related to Control and Accounting for the various groups of physicists,
4. The fourth is an improvement to the Job Scheduling,
5. And the last group is a mixture of various enhancements trying to make the CERN Job Entry more 'user-friendly'.

Each local modification to JES2 at CERN will be described in a more technical-oriented way in the appendices of this document, but let us provide here a summary of all modifications relevant to the needs of CERN users. This summary intends to give an overview why and how modifications are implemented.

1. High tape utilization

A typical CERN user's Job is a FORTRAN application program analyzing data on tapes, this raw data being generated by physics experiments and locally recorded on minicomputers. This high tape usage provided a good argument to close the gap between what was standard facilities in MVS/JES2 and what was considered necessary in a tape oriented shop like CERN.

The standard JES2 does not have the concept of 'Tape Jobs', and thus does not make an optimum usage of the physical tape resources. This means that without modifications in the tape area, the MVS Operating System can be easily overcommitted. The typical scenario being that the system selected several jobs for execution without checking that the necessary Tape Units were available. The jobs are then unable to execute and unnecessarily monopolize a system job initiator that could otherwise be used for a non-tape job.

In order to avoid this waste of system resource, JES2 has been modified so that it identifies Tape Jobs and delays their scheduling to MVS initiators until the requested number of tape units becomes available.

---

1 The MVS initiator is a system component responsible for selecting Jobs and Job Steps to be executed, allocating Input/Output devices and placing them under task control.
2. JOB Enquiry and Control for WYLBUR users

When CERN's first 'IBM' mainframe, the 370/168, was installed in 1976, IBM offered no interactive system that was both adapted to CERN's needs and compatible with the MVS Batch Operating system. As a result CERN decided to use the WYLBUR semi-interactive system, which had been developed at the Stanford Linear Accelerator Centre (SLAC).

WYLBUR has proved very popular through its simplicity of use and its ability to help users over the worst obstacles of a purely card based IBM service. Today WYLBUR, which has been considerably developed at CERN, runs on the SIEMENS-FUJITSU 7890 mainframe and Batch jobs submitted by WYLBUR users are run on the IBM 3090. During the daytime these computers are essentially devoted to running WYLBUR and the flood of short Batch jobs trying to simulate an interactive service.

The present WYLBUR service supports 180 simultaneous users, and 1500 people logon at some stage during each week, with 1000 people submitting at least one Batch job. JES2 extensions were developed so that WYLBUR may provide facilities for submitting Jobs for Batch execution, Job enquiry commands to show the status of jobs, as well as job inspection.

As WYLBUR is not an IBM software package, there is no standard facility that would allow it to communicate with JES2. That is why a specific appendage has been made to JES2, namely an additional processor called the 'FETCH' processor. There are in fact several of such processors handling in parallel the Job Services for WYLBUR users.

3. CERNET and CDC interface

It was fundamental to connect CERNET to the IBM machines in order to provide program to program communication, i.e. to enable a program running in an IBM computer to exchange information with one or many other programs running in any of the CERNET connected computers. We earlier raised the point that JES2 needed a CERN interface to be able to communicate with the WYLBUR Time Sharing System. The communication with CERNET is accomplished in a similar way, i.e. via a 'FETCH Processor' dedicated to this function.

The exchange of information, between JES2 and either WYLBUR or CERNET, is based on the so called 'WINDOWS' mechanism. This is another software package that originated at the RAND CORPORATION and that permits the communication between different tasks (or more accurately Address Spaces in the MVS terminology). It thus provides the necessary means for communication between CERNET and JES2 as regards Job submission, Job enquiry and control.

An important early objective has been to provide a symmetrical Job and Output File exchange between CDC and IBM. Each HOST has a File Manager capable of accessing both permanent files and also Job Input/Output file queues stored on the JES2 Spool Volumes.

Both File managers are used by a program running on the IBM mainframe, called CHIMP (CERNET High-Speed Inter Mainframe package), responsible for exchanging via CERNET Job Input Files and Print Files. Modifications were needed in JES2 given the fact that the external CDC node could not be accessed as a standard IBM installation. It was particularly interesting to provide WYLBUR users with the possibility of sending Job Related commands to the CDC. This was implemented with an appropriate extension in CERN JES2 Command Processor.
4. CONTROL and ACCOUNTING

Some further developments were needed inside JES2 in order to establish the basis for controlling the CERN users' access to the IBM system. Given that CERN is quite an open-shop, it was important to set up an automatic scheme that exercises some control on the users' right to access the IBM system as well on their behaviour towards the various computer resources. More precisely, various validity and authorization controls have been made available for checking

- The USER'S ACCOUNT\(^2\) identifying him to the system.
- The private KEYWORD\(^3\) requested when initializing a session.
- The compatibility with the Group Budget concerning the Priority, the Permanent File Space, as well as the Privileged Class access.

These various values are stored in a specific central file, called the KEYWORD FILE\(^4\). These rules have been set up at the beginning of the IBM service at CERN, following a study of a special committee responsible for the accounting and control. On one hand, the chosen set of rules was thoroughly local to CERN, and on the other hand the relevant time to perform the control was at JOB submission (and WYLBUR logon time).

Both arguments justified a modification in JES2, the aim of which is to read the keyword entry for any user submitting a job to the system, and to check that the 'computer' parameters assigned to the users match the CERN's rules. This checking would take place in the early phase when the job is read into the system. In more detail, a batch job would be typically identified by the user's account and his private keyword. Any invalidity reported when checking these key-values will cause the job to be rejected on the spot. The same will happen if the user's group CPU budget, priority budget or permanent space is exceeded.

In order to provide a user-friendly system, some appropriate warning messages are written in the job log as soon as a budget exceedance is near to happen. This allows the concerned user to contact his group administrator before any job is rejected.

As mentioned before, these controls require the keyword file to be accessed and read so that the current values of the group budget or user's space allocation are examined. This needed a non-trivial modification in JES2 since concurrent accesses to the file must be handled coherently. A special sub-task has been set up to read the keyword records. JES2 issues an asynchronous request to this subtask, and is later informed that the read operation completed.

---

\(^2\) Each computer user gets registered at CERN with an account of the form UUUGG; UUU is a 3 character name chosen by the user, and GG are 2 characters identifying the Physics or Users group he belongs to.

\(^3\) A 3 character confidential code.

\(^4\) File under the control of every Group Administrator containing an entry per registered user, with key- information on his computer attributes, such as the user's initials, the group code, the confidential keyword, the disk space allowed, the group budget and various levels of authorization towards the system.
5. Improved JOB Scheduling

The main reason for improving the Job Scheduling part of JES2 is again that CERN is an ‘open shop’ where the different groups of physicists should work on equal terms. We previously addressed the different controls that aim at a fair usage of computer resources among the various groups of physicists.

Another consequence of the ‘open shop’ CERN characteristic is that the computer load is highly variable, and very much depends on the intervening physicists’ experiments. The situation has therefore little in common with other installations where the load is under the control of a restricted number of people responsible for scheduling the work to the machines. In such cases, it is far easier to exercise control over the work load so that the CPU resources are used in an optimum way, balancing the tasks during DAY-NIGHT periods, as well as avoiding long queues of jobs waiting for specific resources to become available.

At CERN, users are free to submit any job (or string of jobs) at any time, provided the controls described above prove successful. Given that 200 WYLBUR users may concurrently use the interactive system and use the Job Entry Service, one had to make sure that JES2 has the capability to efficiently handle the resulting and ever changing work load.

5.1 CERN Job Class hierarchy

It is worth mentioning at this point that the Job Scheduling is part of a more general context, called ‘Scheduling Process’, divided into three main phases:

- **JOB SCHEDULING**: Process of deciding which job will go into execution next.
- **RESOURCE SCHEDULING**: Process of allocating CPU, memory and tape units.
- **OUTPUT FILE SCHEDULING**: Process of deciding which output to print first.

As regards the resources, they are mainly under the control of a specific part of the operating system, namely the SRM (System Resource Manager); the SRM still does not cover the tape requirements, that is why changes had to be made in JES2 for tape unit scheduling.

**Standard JES2 Job Class support:**

As far as the Job Scheduling is concerned, most of what is required to schedule jobs is supported by the standard JES2. As a matter of fact, one may define in the normal JES2 a certain number of job categories, known as 'JOB CLASSES'. (There are 38 available Job Classes under JES2. Two are used by the system, i.e. STC for started task control, TSU for time sharing logon. The other 36 classes, A-Z and 0-9, are for normal jobs and can be used to help control the job mix. This can be achieved with the assignment of initiator/terminators to initiate specific classes of jobs).

This standard control is however very limited since the IBM documentation itself states that 'there is no absolute rules for assigning job classes and some experimentation is needed'. The recommendations are that jobs of similar characteristics or which specify identical processing requirements⁶ should be assigned the same class.

---

⁵ The process of selecting the next job to go into execution among the various ones already read into the system.

⁶ 'Processing requirements' mainly cover: rate of processor-to-I/O processing, usage of special devices, number of used devices and real storage consumption.
For example, if several jobs are time-dependent and execute in non pageable dynamic storage, it may not be desirable to tie up all of non pageable dynamic storage by having these jobs run concurrently. These jobs may all be assigned to the same class. Then if only one initiator is started that can handle this class, there will never be more than one of these jobs executing at once.

CERN automatic Job class assignment:

It still remains that such an empirical system is not suitable for the CERN Computer Community, and one may provide several reasons for this:

1. The standard JES2 expects the user to provide the CLASS value for his batch jobs. As we saw before, the class entity is quite arbitrary and is only meaningful when placed under the control of a limited number of users well aware of the rules to apply. This is not applicable at CERN where the users' community is very big, ever evolving and where users are far more familiar with tangible job requirements such as the CPU time or the number of tapes rather than system resources management.

2. It was therefore important to implement in JES2 a tailored CLASS STRUCTURE which would meet the need of the CERN users, i.e. at least including express non-tape jobs, short tape jobs and production jobs.

One should emphasise at this point that a particular policy of the CERN computer center is to provide the best possible turn-round for express and short jobs in order to save human time. Another substantial feature was to make the class assignment transparent to the user, and make him describe his job requirements in terms of estimated TIME, number of TAPE DRIVES, this was in order to allow the Job Entry Subsystem to handle the resource management. The first modification to the standard Job Scheduling was to create the provision of the following class structure:

- **Class X**: Express, no tape, up to 15 secs.
- **Class S**: Short, no tape, up to 2 mins.
- **Class T**: Short, tape, up to 2 mins.
- **Class M**: Medium, up to 15 mins.
- **Class L**: Long, up to 2 hours.

Some recent statistics showed that over half the jobs are in the express category, using less than 2% of the CPU, although they use much more in other system resources. At the other end of the spectrum, the long jobs consume 67% of the CPU whilst contributing less than 5% to the total number of jobs.

5.2 AUTOMATIC PRIORITY SCHEME

In addition to the CERN specific control over Job class assignment described above, it also appeared necessary to implement an automatic PRIORITY SCHEME which would provide the CERN users with some control over the trade off between the job cost and the job turn-round. This was the second phase of the implementation of a 'CERN tailored' Job scheduling.
This would typically allow the CERN users to manage with urgent work that needs a fast turn-around during the daytime, or less urgent work that may run overnight, or even work that may wait until the machine has no other work to do, and all this under the fair condition that the more you pay, and the quicker you get serviced.

6. CERN 'FETCH' QUEUE

An interesting local feature at CERN is to allow the users to fetch output back to a terminal for inspection before printing all (or part) of it. For this purpose, a pseudo output queue was created at CERN, namely the FETCH queue, located on the SPOOL disk. Output files may remain two days maximum in the FETCH queue. If the owner of a FETCH file does not decide to print it within this period, the file gets automatically purged. The user's interface is very simple since it consists of one control statement inserted in the JCL, i.e. /FETCH.

The typical scenario of a job inspection would be the following:

- Retrieving the printable output at the terminal: FETCH job
- Printing all or part of it if needed: PRINT job
- Getting rid of the printable output: PURGE job

7. CPU TIME NORMALIZATION

In an ever evolving environment it is essential to be able to measure comparative performances of machines. Since the CPU speeds of the two machines making up our IBM mainframe are different, one needed to refer to a common time scale; as a matter of fact, normalizing is essential for meaningful reporting, as well as for allowing machines of differing speeds to work concurrently in a user friendly way.

From our history, we have an excellent yard stick in what was our first IBM 168 machine; In order to avoid the criticisms of the 168 unit as "archaic", we recently proposed to formally refer to it as the CERN Time Unit.

8. DYNAMIC PROCLIB SUPPORT

It is very interesting from the users' point of view to maintain libraries of their own private procedures, provided these can be accessible from BATCH. This facility of concatenating a user procedure library to the public one is not offered by the standard JES2. A local modification has been made to create a new keyword in the standard /*JOBPARM statement, namely:

PROCDSN=GG.UUU.PROCLIB

allowing the user to specify the name of the private procedure library he wants to access. In addition to this, the user receives an error message if the data set is not cataloged, or if it has been migrated by HSM.
9. DISPLAY JOB COMMAND ENHANCEMENTS ($DJ)

The standard JES2 product is able to display the following job information for a specified job:

- **NUMBER**
- **NAME**
- **STATUS** (Input queue, running, in output ...)
- **CLASS OF EXECUTION**
- **PRIORITY**

In order to improve Job monitoring both by users and operators, it was suggested to display additional information on the jobs. The resulting CERN display tries to better reflect the evolving parameters of a job within the various steps of its life within the system. More precisely, the following information has been included in the standard display:

- **JOBS IN INPUT QUEUE:**
  Relative position in the queue, estimate CPU time and number of lines.

- **JOBS EXECUTING:**
  Number of produced lines, time limit, and machine on which the job is running.

- **JOB AWAITING OUTPUT:**
  CPU time, total number of lines, and per output class, number of lines and position in the queue.

- **JOBS PRINTING:**
  Output class, number of lines already printed over the total number.

- **JOBS IN THE FETCH QUEUE:**
  CPU time and number of lines produced.

10. NOTIFY FUNCTION ENHANCEMENTS

The NOTIFY statement requests that a message should be sent to the user specified when the job has completed. This facility exists in the standard JES2 package, but users are notified only on the machine where the job had been read. A modification was made in JES2 to tackle this restriction, and thus to insure that users are always notified of the completion of their job, i.e. regardless of the input machine of the job.

In addition to this, the notification message has been made more informative since the machine name is provided, as well as the system diagnostic such as ‘JCL ERROR’, ‘Job canceled’, ‘Job abended’ as well as the System Return Codes by step.
11. OFFLINE FUNCTION

It is very convenient for WYLBUR users to get a listing of what they have in the so-called 'ACTIVE FILE', which either resulted from a 'USE' command to display a data-set or a member of a library, or from a 'FETCH' command to get at the terminal the output of an executed Job.

This is achieved via the WYLBUR 'LIST OFFLINE' function, which can direct the output either on local or remote printers. WYLBUR then invokes the JES2 services to handle the request, and it is in fact a CERN JES2 control statement that does the job. It is the /*OFFLINE NJE control card which permits printing of a stream of data directly by the network function, and without executing any BATCH job.

12. XEROX DEVICES SUPPORT

12.1 Central XEROX 8700 laser printer support

The XEROX 8700 printer offers useful and flexible facilities such as the page orientation (width-wise landscape or length-wise portrait), the printing mode (SIMPLEX single sided or DUPLEX double sided) and the logical page layout (output presented in n columns).

In order to activate these extended features within a standard MVS/JES2 environment, the users would have to modify their application program(s) to insert the appropriate specific XEROX commands. Given that a total transparency would be far more desirable, it was decided to create and maintain JES2 modifications accomplishing the required interface. In other words, the purpose is to provide the users with the possibility of expressing their XEROX print requirements in terms of familiar standard JCL parameters already used for the 3800 IBM Laser Printer, and let JES2 automatically issue the ad hoc XEROX command.

12.2 Remote XEROX 2700 printer support

Some modifications were required to produce a separator page for the output generated from Document Programs.

13. MICROFICHE DEVICE SUPPORT

Microfiches provide a very convenient way of storing lengthy outputs. The CERN Computer Center is equipped with a KODAK KOMSTAR 200 microfiche device which is connected online to the IBM. Thanks to a modification in JES2, it can be used to put line-printer type output onto fiche, 207 pages fitting onto a small piece of plastic measuring 15*10 cm. This can be extremely convenient (as well as ecologically and financially desirable) for things such as long outputs that are only referred to occasionally, reference listings of big programs or libraries, or copies of job output to send to collaborators.

The JES modification includes the possibility of producing a title of up to 20 large characters along the top of the fiche using the 'USERNAME' parameter on the JOB card. The user interface is very simple since it is based on a particular output class, namely 'M'.

JES2 needed to be modified since the microfiche device is not supported by the standard JES2
Print Processor, particularly in the way it handles Forms Control Buffers.7

14. ADDITIONAL FACILITIES FOR OUTPUT JOBS

Double job separator page:

Except for the cut-sheet XEROX local and remote printers, as well as for the Microfiche device, the JES2 Print Portion had to be modified to send two (instead of the standard one) Job Separator pages, the purpose of this being to make life easier for the person who separates the listings.

Delivery codes:

Some CERN locations with a small number of users cannot afford to buy a remote printer. In addition to this, users may request that centrally printed output is delivered to a specific delivery point. That is why another modification was made in the Print Portion of JES2 to permit the printing of a big 'Room Number' in the format 'Xn' on the front page of the listing, so that the output is forwarded to the remote site. The user interface is again the standard /*JOBPARM statement where the local parameter 'ROOM=Xn' has been made available.

Remote printing notification:

Following some complaints of remote users having difficulties in retrieving their outputs among the pile, a modification has been made in JES2 so that it prints on a teletype adjacent to the remote printer a message for each job being printed, specifying the job name, the time and the size of the output.

15. DISPLAY CONSOLE ALLOWED ($DC)

It is very convenient, especially for system programmers, to be able to visualize the screen of the operator console and this for either of the two IBM mainframes.

A CERN JES command was created for this purpose, namely $DC.

---

7 The FCB is the buffer that is used to store vertical formatting information for printing, each position corresponding to a line on the form.
PART 4

JES2 MAINTENANCE AT CERN

1. PROPORTION OF 'CERN MODIFIED' JES2 CODE

Out of the 160,000 lines of IBM Assembler code that constitute the IBM JES2 Package, 15,000 additional lines have been modified or merged, (which represents almost ten per cent of the total program), and this makes the local 'CERN modified' Job Entry Subsystem.

One may point out that most IBM customers run a non-modified version of JES2; on one hand, this implies they have to strictly comply with the standard rules of the package. But on the other hand, they benefit from 'full' IBM software support.

2. JES2 PROBLEM DETERMINATION

Each time a problem affects the MVS/JES2 system, a CERN System Programmer has to determine whether or not a local software modification is involved. If it is the case, he will have to find out what is wrong with this local change; more precisely, he will have to identify which abnormal condition had not been correctly handled by the local code, and modify it accordingly. If the system programmer identifies that the problem resides in a non modified part of JES2, the IBM software team accepts full responsibility for it.

3. JES2 UPGRADE

Like other system software packages that IBM offers, JES2 is regularly upgraded, which means that new versions are available to the customers. The IBM development teams regularly create new versions of the package including:

- Additional features and users services
- Higher performance and reliability
- Better software design.

The upgrade scenario cannot be used at CERN as in other installations running the standard IBM product, where the IBM new package simply substitutes the old obsolete one. The extra work of implementing the local CERN features within the new version of JES is designated as 'CONVERTING JES2 LOCAL MODIFICATIONS'.

4. TESTING

Once the conversion step comes to an end, the test phase has to begin, the purpose of which is to reveal 'BUGS', i.e. errors in the program. The tests should be as exhaustive as possible, not only covering the various functional parts of JES2, but also the interfaces with the external environment such as WYLBUR, CERNET and TSO. Testing a version of a modified JES can be performed in parallel with the production work thanks to the concept of Secondary JES2.

---

* A secondary JES2 deals with specific test jobs and does not interfere with any of the users jobs.
5. REASONS FOR UPGRADING

One may provide several reasons that persuade us to upgrade our level of JES2 product in addition to the ones given before, i.e. new users services, enhanced performances and reliability.

1. IBM always push their customers for a minimized difference between the basic Operating System (i.e. MVS) Level, and the one of the Job Entry Subsystem running under it, this being based on long-term compatibility reasons. IBM might even threaten some customers running very old releases of the product with giving up the software assistance.

2. Moreover, a new version often relieves some limitations of the old one, which could be more or less constraining. One may provide a good example with the number of external nodes an installation may be connected to. When we migrated from the Version 3.0 of JES2/NJE 1978, to the JES2 component of MVS/SP Release 3, i.e. Version 1 Release 3.3 of JES/SP, (System Product) 1983, the number of other installations CERN could communicate with increased from 99 to 1000; This was in fact a sine qua non to allow the extension of our network, especially towards the USA via EARN and BITNET.

3. Some newly announced IBM Peripheral devices allowing Jobs to read or write data are only supported under recent versions of the Job Entry Subsystem. It is the case for the new IBM CARTRIDGE TAPE Model 3480 that CERN acquired in December 85.
APPENDIX A

JES2 JOB PROCESSING

JES2 processes a job stream made up of the following stages explained below:

A.1 INPUT

JES2 reads a job into the system (See Figure 2) from the incoming job stream that consists of jobs from a variety of sources. The internal reader facility is a special Job Entry service which allows MVS to submit system jobs, started tasks (STC), Time Sharing logons (TSO), and also allows users’ programs to submit jobs. Tape and disk input is also supported through the internal reader facilities via utilities.

At CERN, the terminal based time sharing system WYLBUR uses the internal reader facility to interface with JES2 for submitting jobs from the users’ terminals. Jobs may also be submitted from another JES2 system through an NJE line. Card readers are no longer used at CERN.

A.2 CONVERSION

JES2 invokes the MVS converter to scan each job’s JCL for syntax errors. The MVS converter converts the JCL to internal text, which JES2 writes to the spool. JES2 queues a job with correct JCL to the execution queue by its priority. For a job with JCL errors, JES2 bypasses the execution stage and queues the JCL together with appropriate diagnostic messages to the output queue for direct use by the output stage.

A.3 EXECUTION

An eligible initiator uses the SSI (Subsystem Interface) to request a job for execution from the execution queue by priority within its class. JES2 supplies input data to the processing program by reading from spool and writing output records to spool. JES2 reads input from spool and writes output to it during the execution of the processing program. When the job completes execution, the job termination portion of MVS informs JES2 (via the SSI); JES2 then places the job in an output queue to await processing of its output.

A.4 OUTPUT

Once a job completed its execution, JES2 processes all of its SYSOUT data sets by creating work elements and placing them in the Job Output Table (JOT); JES2 builds these work elements based on the output characteristics of the job.
A.5 PRINT/PUNCH

JES2 selects output work elements for processing from the JOT by priority. The selected output can be in a different states. Output from the job is to be processed locally or at a remote location, or the job output is just passing through this JES2 system and must be transmitted to another JES2 system in a NJE network.

A.6 PURGE

When all processing for a job has been completed, JES2 releases the direct-access space acquired for the job, all control information associated with the job, and any other resources owned by the job.

A.7 SHARED-SPOOL

At CERN, there is one JES2 processor running on the IBM 3090, and another one running on the SIEMENS 7890, both sharing the same work queues located on four 3380 Spool Devices, representing a total capacity of about five thousands jobs. (In an normal situation, the percentage of Spool utilization would average 70%).

Both JES2 processors (one on the IBM 3090, the other on the SIEMENS 7890), share the same work queues. Each JES2 has access to the spool devices and independently selects jobs for processing from the spools. This implies that the various phases constituting the life of a job might be carried out on the two machines, i.e. that one JES2 processor can be processing a job's input while the JES2 of the other machine schedules and executes the same job.
APPENDIX B

TECHNICAL DESCRIPTIONS

B.1 TAPE UNIT SCHEDULING

As we stated before, this modification is aimed at trying to close the gap between standard facilities in MVS/JES2 and what is considered necessary in a tape oriented shop like CERN. The aim is that a job needing a given number of tapes will go into execution only when the requested number of tape units is available. This avoids having an overcommitted system where initiators are tied up with jobs unable to execute because the necessary units are not available.

B.1.1 USER INTERFACE

The user interface consists of an additional control card in the JCL (Job Control Language), namely one of the following type:

/*UNIT unitname1 = n,unitname2 = m,...

This allows the user to specify a meaningful unit name that corresponds to the desired density for all tape requests, and also allows the system a higher degree of flexibility for scheduling 1600 BPI tape mounts. The valid unit names are described by the table below. In case the /*UNIT control statement is invalid, (either due to an illegal unit name or an incorrect syntax), the given job is aborted and immediately scheduled for output.

Table 1: REAL AND LOGICAL TAPE UNITS NAMES

<table>
<thead>
<tr>
<th>REAL UNIT</th>
<th>LOGICAL UNIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>3420-7</td>
<td>T6250</td>
</tr>
<tr>
<td>(800-1600 bpi)</td>
<td></td>
</tr>
<tr>
<td>3420-8</td>
<td>T1600</td>
</tr>
<tr>
<td>(1600-6250 bpi)</td>
<td></td>
</tr>
<tr>
<td>3480</td>
<td>TAPE</td>
</tr>
<tr>
<td>(38000 bpi)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>HARDWARE UNIT TYPES</th>
<th>LOGICAL UNIT NAMES</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>CTI</td>
</tr>
</tbody>
</table>

B.1.2 SYSTEM INTERFACE

The modification has been implemented in such a way that it allows up to 8 different hardware unit types (We currently have three, 3420 Model 7 , 3420 Model 8 and 3480 Cartridge Tape),and any number of logical device names, synonymous with the basic hardware types, provided the ad hoc changes are known by the MVS Operating System.
B.1.3 OPERATIONAL INTERFACE

The following commands have been made available so that the operators may monitor the tape units allocation depending on the needs.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$UA,unitname,ADDR, ..., ADDR</td>
<td>To allocate the tape unit on physical address ADDR under the control of JES2 with the logical function implied by unitname</td>
</tr>
<tr>
<td>$UD,unitname,ADDR, ...,ADDR</td>
<td>To deallocate units from the JES control</td>
</tr>
<tr>
<td>$UD,ALL</td>
<td>To deactivate unit scheduling completely</td>
</tr>
<tr>
<td>$UL,unitname</td>
<td>To list units currently controlled by JES2</td>
</tr>
<tr>
<td>$UF</td>
<td>To give the number of free units of any type</td>
</tr>
<tr>
<td>$UJnnn</td>
<td>To display unit requirements for a job</td>
</tr>
</tbody>
</table>

B.1.4 TAPE UNIT PROCESSING WITHIN THE LIFE OF A JOB

Job input:
As we explained before, the scheduling is based on the /*UNIT card specifying the requested number of tape units. This number is stored in the JQE (Job Queue Element), which is a JES2 resident table containing basic job information, and in the JCT (Job Control Table), which is resident in the Address Space of the Job during execution and otherwise kept on the spool pack.

Job selection:
A job is not selected for execution before the requested number of tape units is available. Global counters (one per hardware unit type), are maintained in the SSVT (Subsystem Vector Table) in the common storage area in order to reflect the current unit allocation.

Job allocation:
There is a standard MVS/JES interface defined to allow JES3 to control the allocation of tape units. (JES3 is another IBM Job Entry Subsystem; JES2 and JES3 perform the same basic functions on a single processor installation, that is, they read jobs into the system, convert them to internal form, select them for execution, process their output and purge them from the system. On an installation with more than one processor in the configuration, there are noticeable differences between the two JESs; the particularity of JES2 is that it exercises independent control over its job processing functions). Each JES2 processor controls its own Job Input, Job Scheduling, and Job Output processing.

Job/Step termination:
The units reserved by a job are kept until end-of-job. At that time, the job termination routine in HASPSSSM will return all units reserved by a job to the global counters thus freeing them for other jobs. The units have also to be freed in case the job is canceled.
B.1.5 CHANGED CONTROL BLOCKS

Job Queue Element (JQE), Job Control Table (JCT), and Subsystem Vector Table (SSVT).

B.1.6 CHANGED JES2 MODULES

Job Input (HASPRDR), Job Selection (HASPNUC), Job Allocation and Job/Step Termination (HASPSSSM) and Command Processing (HASPCOMM).

B.2 CERN JOB CLASS HIERARCHY

We briefly justified before the necessity of implementing at CERN a particular JOB CLASS STRUCTURE which meets the users' needs. Given the previous experiences users had with other operating systems, (especially SCOPE2 under CDC), the policy had been to force jobs into the right JOB class according to

- The amount of time
- The number of tape units
- The amount of work space allocated to the Job

Providing the mechanisms to control Job Class assignment was the first step in the project of improving Job Scheduling. The CERN Job Class hierarchy is summed-up by the table presented on the next page.

<table>
<thead>
<tr>
<th>CLASS</th>
<th>TYPE</th>
<th>MAX TIME</th>
<th>MAX TAPE</th>
<th>MAX REGION</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>Express</td>
<td>15 secs</td>
<td>0</td>
<td>800 KB</td>
</tr>
<tr>
<td>S</td>
<td>Short</td>
<td>2 mins</td>
<td>0</td>
<td>2 MB</td>
</tr>
<tr>
<td>T</td>
<td>Short</td>
<td>2 mins</td>
<td>2</td>
<td>2 MB</td>
</tr>
<tr>
<td>M</td>
<td>Medium</td>
<td>15 mins</td>
<td>3</td>
<td>4 MB</td>
</tr>
<tr>
<td>L</td>
<td>Long</td>
<td>4 hrs</td>
<td>5</td>
<td>4 MB</td>
</tr>
<tr>
<td>I</td>
<td>Cernet</td>
<td>1 mn</td>
<td>0</td>
<td>800 KB</td>
</tr>
<tr>
<td>O</td>
<td>Privi.</td>
<td>24 hrs</td>
<td>10</td>
<td>8 MB</td>
</tr>
</tbody>
</table>
B.3 CERN JOB PRIORITY SCHEME

The second improvement of the Job Scheduling was to set up an automatic Priority Scheme which would allow users to get a turnaround proportional in speed to the Job cost. Four levels of priority have been set up, and are described in Table 4.

<table>
<thead>
<tr>
<th>PRIORITY</th>
<th>COST FACTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>0.5</td>
</tr>
<tr>
<td>P2</td>
<td>1.0</td>
</tr>
<tr>
<td>P3</td>
<td>1.5</td>
</tr>
<tr>
<td>P4</td>
<td>2.0</td>
</tr>
</tbody>
</table>

Table 4: PRIORITY LEVELS AND RELATIVE COST FACTORS

P1 and P2 are foreseen to be the normal production priority levels whilst P3 and P4 are to be used for more urgent 'one off' jobs that users want to run ahead of long production chains. The users' interface is based on the Job card parameter 'PRTY'.

B.4 HASPSSSM MODULE LOADED IN CSA (Common Storage Area)

This is more system-oriented CERN modification than a user-oriented one.

It is far easier to maintain the JES module HASPSSSM in a SYSn.APFLIB library which is shared than in SYS1.APFLIB which is not. Obviously, one has to make HASPSSSM accessible from any address-space given the fact that this module is a set of routines used to handle communication between JES2 and other system components. That is why the modification includes the loading of HASPSSSM in the Common Storage Area (CSA).

Another advantage of this system modification is that it allows a secondary Job Entry Subsystem to run with an HASPSSSM in test without wasting time waiting for an IPL. It also frees some place in the Link Pack Area (LPA) in a non negligible way since HASPSSSM is 32KB in size.
APPENDIX C

JES2 PROGRAM ORGANIZATION

The purpose of this section is to present a brief description of each JES2 module. The module functions, basic methods of operation, and their relationships to MVS and to other elements of JES2 are described in the following order:

1. HASPDOC
   Describes values and control blocks used throughout the executable JES2 modules.

2. HASPTABS
   Describes various tables used by JES2.

3. HASPIRA, HASPIRDA, HASPIRMA, HASPIRPL, HASPIRRE
   Initializes the subsystem and its interfaces and gives control to the JES2 $ASYNC processor.

4. HASPSCAN
   Provides support for $SCAN facility.

5. HASPNUC
   Provides central service subroutines to the JES2 system.

6. HASPIIOS
   Provides job output services.

7. HASPNET
   Establishes direct connections and reacts to disconnections among the various members of a network job entry (NJE) network.

8. HASPSSSM
   Provides the interface between JES2 and MVS.

9. HASPAM
   Provides the access method for processing subsystem data sets.

10. HASPCON
    Provides the console support services for JES2.

11. HASPCOMM
    Receives and processes all JES2 commands from local or remote JES2 input sources.

12. HASPSERV
    Provides command processing scanning services for use by JES2 HASPCOMM.

13. HASPRDR
    Provides the functions of the input service processor.

14. HASPCNVNT
    Provides JCL conversion processing.

15. HASPXEQ
    Provides the conversion and execution services for JES2.
16. HASPRPDU
   Determines the characteristics of a job's output data set and groups, performs the output operations, and loads the FCB and UCS images from the specific library SYS1.IMAGELIB.

17. HASPRTAM
   Provides major functions through which JES2 communicates with remote terminals and other JES2 nodes in a JES2 network.

18. HASPBSC
   Provides remote terminal and network functions for BSC devices.

19. HASPSNA
   Provides remote terminal and network functions for SNA devices.

20. HASPSTAM
   Provides the Spool Offload functions.

21. HASPSPOL
   Provides the JES2 spool volume support.

22. HASPWARM
   Provides warm start processing of the JES2 queues.

23. HASPCKPT
   Provides checkpoint processing services.

24. HASPTERM
   Provides recovery and termination services.

25. HASPMISC
   Provides for the miscellaneous services of JES2; The event trace log processor, the purge processor, the priority ageing processor, SMF subtasks, and the network account routines.

26. HASPPFET
   This 'CERN' module performs the interface between JES2 and the other system components WYLBUR, CERNET and TSO.

27. HASPEXIT
   This local module contains the JES2 EXIT1 (Print/Punch Job Separator). It was written at CERN for allowing a device-dependant Job Separator Page, especially for the XEROX 2700 RIOS Print and Punch devices.
Figure 2: JES2 JOB PROCESSING FLOW CHART
APPENDIX D

BIBLIOGRAPHY

The following materials can be consulted to get further information on the IBM package JES2:

- Introduction to IBM Data Processing Systems ... GC20-1684
- IBM System/370 System Summary-Processors ... Go22-7001
- IBM System/370 and 4300 Processors Bibliography ... GC20-0001
- IBM System /370 Principles of Operations ... HA22-7000
- SPL JES2 Installation, Initialization and Tuning ... SC23-0046
- JES2 LOGIC ... MY24-6006
- Operator’s Library JES2 Commands ... SC23-0048