CP2000 Data Extraction Program (CP2KEXTR)

June 12, 2006

Alex Patterson

Preface

This program can be used to read SMF data to produce a file which is directly useable by CP2000. It is intended for use by IBM Employees and Business Partners.

The program may be obtained by IBM Employees from -

- http://w3-1.ibm.com/support/americas/wsc/cpsproducts.html
- ftp://cpstools.washington.ibm.com directory: CP2KEXTR

Business Partners may obtain the program from the PartnerInfo System.

1. http://www-1.ibm.com/partnerworld/sales/systems/
2. Sign on to PartnerWorld with id and password.
3. Select your GEO.
4. At the top of the left hand NAV bar type CP2000 into the search field and press “GO.”
5. First item should be “CP2000 Tool and CP2000 Data Extract Program.”

Select this item and download “cp2kextr.lg”

Contacts:

- Alex Patterson on
  - NOTES - Alex Patterson/Cedar Rapids/IBM @IBMUS
  - INTERNET - pattrsn@us.ibm.com

Comments, requirements, and questions can be directed to us at any of the electronic addresses.

Notices

Changes for this version:
Removal of the WHPTSID and WTYPE= fields.
Some new fields.

CP2KEXTR Flow

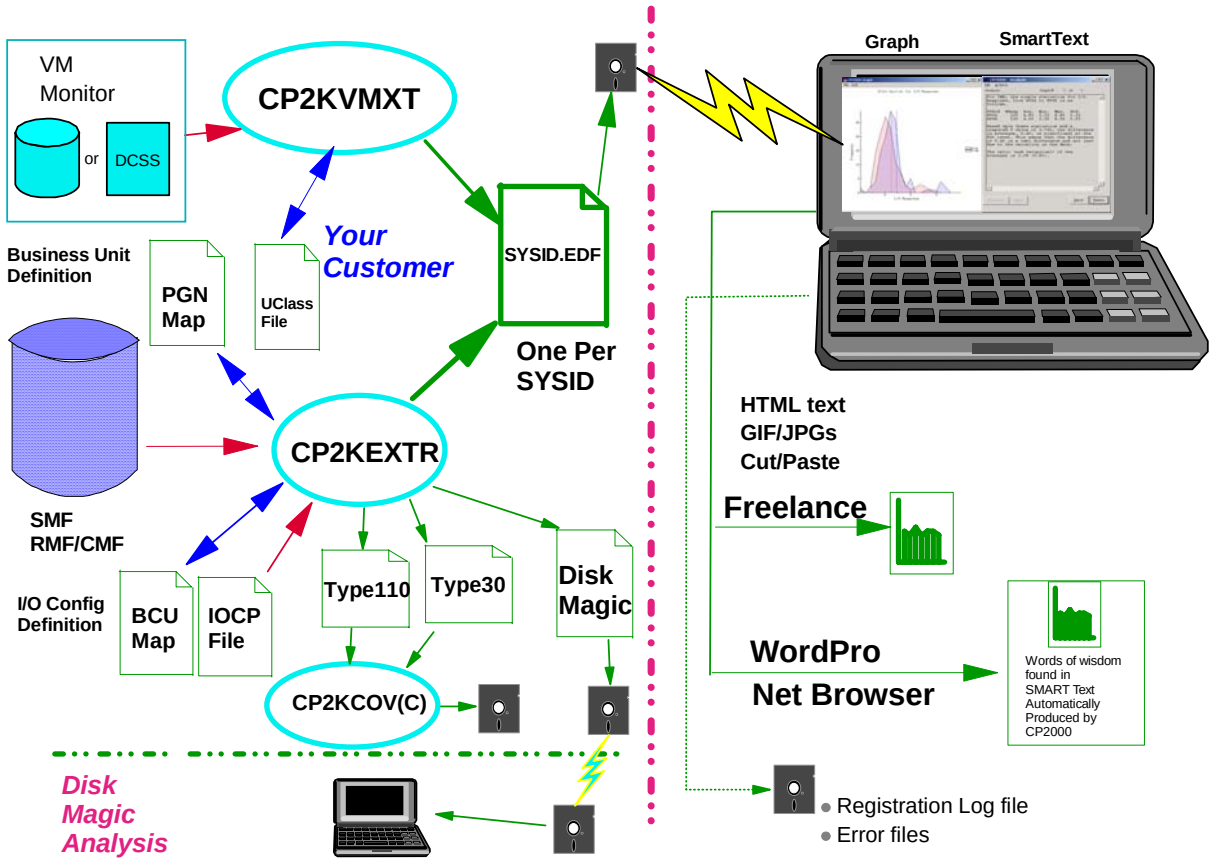

Figure: CP2000 Data Flow

"CP2000 Data Flow" on page 3 shows the data flow from the SMF data set to CP2000. The vehicle is called an EDFI (Enterprise Data File for Input) file. This file is created by CP2KEXTR. Input is the SMF data and supplemental mappings which help in processing the data.

The PGNMAP (Performance group mapping) provides a the method of combining PGNs in MVS into business units. The BCUMAP provides a configuration mapping for the DASD.

Here's the steps in executing CP2KEXTR.

1. If DASD data is to be used (highly recommended), produce a BCUMAP using the BCU=AUTO and PROCESS=NO parameters. If Workload data is to be used (highly recommended), produce a PGNMAP using the PGN=IPS or PGN=GOAL and PROCESS=NO. (Normally done in the same step as 1 above.)
2. EDIT the mapping produced. The BCUMAP should be reviewed for accuracy since the RMF data is often incomplete. Especially verify that BCUs across different partitions or systems are defined with the same BCU ID so CP2000 will recognize them as the same device. The PGNMAP should be edited to combine PGNs into more meaningful business units.
3. Execute CP2KEXTR to produce the EDFIs. One for each SYSID.

In addition an interface file for the Disk Magic DASD marketing tool can be produced.

Disk Magic is now supported for device level input, although old summary style output can still be produced.

Quick Start Guide

This section is intended to help you to get the most data with the least amount of effort. It is written as a cookbook. If it seems too simplistic for you, then just scan it. Even experienced users should find something helpful in it.

If you are using the CP2000 Extract Program only to produce a Disk Magic file, then ignore anything to do with Workloads, PGN=, IEAIPS or PGNMAP, and delete these from your input.

What you need:

1. A diskette with the program downloaded to it.
2. A TSO userid and password.
3. A dataset with an LRECL of exactly 80 and RECFM of fixed blocked (or fixed) to upload to.
4. To know what a JES jobcard needs to look like for this installation.
5. To know the SMF ID of the system in question.
6. To know when and for how long you wish to sample. For performance analysis, 15 minute RMF intervals may be reasonable. But for us doing capacity planning, one hour samples are a minimum. Depending upon the amount of data available, different durations may be suggested.
   - For a weeks worth of prime shift data (9 hours a day x 5 or 45 hours), 45 one hour samples is not a problem for CP2KEXTR but may cause virtual storage problems in CP2000 especially if combined with a number of other system images. One might select two 2 hour durations for each day thus yielding 10 samples.
   - For a days worth of data, hour samples are fine.
7. To know whether you are running in Goal (Workload Manager) mode or not. If the system is not in Goal mode, know the dataset name and member name of the IPS active at the time of the samples.
8. Know the dataset names of the SMF files for the samples you have chosen. Normally this will be on tape, frequently in generation datasets.
9. If possible, sit down with the system programmer and
   1. review the SMFPRMxx member of 'SYS1.PARMLIB'. Verify that you are collecting the SMF records that you will need. These are type 70 through 75 and 78, also type 30 subtype 2 and 3 and type 42 subtype 6.
   2. Also verify that you are doing INTERVAL processing (check also the subsystems in this member.) Originally IBM recommended turning this option off unless the data was required, because of the quantity produced. That recommendation was many years ago, and many installations still have this option turned off. Today with CICS and DB2 using SMF, these records will be a drop in the bucket. Also today SMF parameters can be changed via command. So you are no longer betting an IPL that everything will be OK. One can revert to the former parameters with an MVS command. In any event much more satisfactory I/O data will be produced by CP2000, if INTERVAL is specified.
   3. While you have the system programmer's attention, ask about the job that dumps off the SMF datasets. Sometimes that job will exclude certain types, and they will not make it to the tape. Verify that the ones you will need, do.
   4. Also ask about the RMF data gatherer. In most cases this will have already been setup correctly. However if you interested in TAPE activity, the RMF defaults for Device Activity and I/O Queuing is only DASD. So if you want to study tape make sure that the RMF data gatherer has the parameters DEVICE(DASD,TAPE) and IOQ(DASD,TAPE).
10. Ask if the system programmer knows whether the SMF records are in time sequence or not.

11. If possible, obtain an I/O configuration diagram for the installation.
12. If you will be using the IOCP to define your BCUMAP, learn the name of the IOCP dataset, and if LPAR, the name the LPAR partition you will be studying.

Creating the diskette

CP2KEXTR is an assembler language program which is distributed in LOADER format. The file CP2KEXTR.EXE downloaded from the website contains the program and JCL. Execute the CP2KEXTR.EXE program to create the CP2KEXTR.BIN file and copy that file to a diskette.

Sign onto TSO and allocate the dataset

This will be the dataset to which you upload the binary file. This is most easily done in ISPF/PDF 3.2. Give it a name of CP2KEXTR.UPLOAD. It must have an LRECL of 80 and a RECFM of FB (or F). BLKSIZE should be at least 6400, or if you have system determined blocksize implemented, specify zero. (If system determined blocksize is not implemented, zero will give you an error.) Suggest a primary extent of three cylinders with a secondary extent of one, and no directory blocks.

Upload the program

A dataset must be allocated as specified above. You now have a PC file of binary characters. It must be uploaded in binary. There are many host emulators in use, and each appears to have its own way of specifying its parameters. So as you hunt through the panels and specifications for file transfer in your emulator, please keep in mind: the dataset must be FIXED, the LRECL must be 80 and you must use BINARY mode. Afterwards you will see recognizable JCL in the dataset, where you did not when you TYPEd the file on the PC.

Most successful file transfer seems to be FTP. To upload using FTP start in the directory that you have the cp2kextr.bin file, and enter:
FTP wscmvs.washington.ibm.com
Or substitute the correct node name.
Enter your userid and password.
enter:
binary
enter:
put cp2kextr.bin cp2kextr.upload
enter:
quit

Receiving

The uploaded file must now be reformatted by the RECEIVE command. In ISPF/PDF 6 enter
RECEIVE INDSN(CP2KEXTR.UPLOAD)
The command will now prompt you for the dataset name, type
DA(CPSTOOLS.JCL) SPACE(2,2) CYLINDERS
This will create a dataset with your high level qualifier.CPSTOOLS.JCL.

The First Run - Definitions

This step is going to create two files: the PGNMAP and the BCUMAP. The PGNMAP will tell the Extract Program which performance groups to map to which business elements. The BCUMAP defines the I/O configuration for the Extract Program to use during the data run.

Customize the JCL

Go into the member hlq.CPSTOOLS.JCL(EXTRACT), and change the jobcard to meet the standards of your installation.

Next change the SMFIN DD statement to point to your input SMF dataset.

If you are not running in Goal mode, uncomment the IPS001 DD statement and specify the dataset name and member name of the IPS that is active at the time you are studying. (Uncomment by removing the asterisk in column 3.)

Specify the correct prefix in the dataset name for the EDF001, BCU001 and PGN001 datasets. And also the IOCP001 dataset if you are using the IOCP option of BCU=AUTO option. Note if you are using another dataset, they must have a record format of F or FB and an LRECL of 80.

CP2000 Extract Parameters

- ENT= is used for the name of the enterprise. The value is passed into CP2000. It is required.
- SYSID= is used to specify the SMF ID of the system to be studied. It is required.
- DURATION= is used to specify the sample size in hours. So for one hour sample specify 1, for 2 hour samples specify 2 and so forth. It is required.
- TIME=(xx-yy) is used to specify the times of interest. Be sure that the duration fits evenly into these times. You may have multiple TIME= statements. For example if you are using 2 hour samples you may have TIME=(08-12) and another TIME=(13-17). It is required.
- DATE= is used to specify the days of interest. If it is not specified, then all dates in the input file will be used. If you use it, you will get what you expect. Presuming those dates are on your tape. The following are example DATE= Statements.
  - DATE=(09/25/95)
  - DATE=(09/25/95-09/29/95)
  - DATE=(09/25/95,09/26/95,09/27/95,09/28/95,09/29/95)
- BCU=AUTO is specified to generate a BCU Mapping file. If you are interested in FICON Mapping then you also want to specify an IOCP001 DD name pointing to the current IOCDS. This will generate a more complete BCUMAP suitable for FICON analysis. For the FICON analysis you should also use BCU=FICONAG and PART= to specify the LPAR partition name. PGN= is used to generate the Performance Group Mapping. Specify PGN=IPS to generate the file from an IPS, or PGN=GOAL to generate the file from the Goal Mode data in SMF.
- PROCESS=NO is used to stop processing. This will cause the BCU map and the PGN map to be generated, but no further processing will take place.

See topic "JCL requirements" on page 15 for an example of the JCL.

Submit the job

This job will generate the BCU map and the PGN map.

Please review the messages produced by the Extract program

Messages go into the PRINT001 DD statement and so will be with your JCL. Assuming nothing bad has happened, continue on.

Review the BCU map

See topic "Sample BCU Map" on page 27. The BCU map should be quite complete. In cases where multiple systems share DASD, it sometimes happens that cache data is only collected on one of the connected systems. If that is the case, then that system will have the most accurate BCUMAP. (controller type models and cache sizes and LCUs under a controller.) Build that system first then use this as reference to update the other system's BCUMAPs as you process them.

You may want to change the names of the BCUs using some installation nomenclature. If you are doing multiple systems with shared DASD, be sure you use the same name for shared devices from each system.

If you are going to use Disk Magic, have a look at the names that Disk Magic uses for the types of control units, and types of disks. If you put these into the BCUMAP, you will not need to change them in Disk Magic. Disk Magic uses names like 'RVA 1' rather than the type model of the controller which the device reports. And it uses names like 'RVA 1 (3390-3)' for the disks. (Value in parenthesis is what the device is emulating.) Enter these names into the BCUMAP. Note if the name has blanks in it as in this example, then surround the name with single quotes. i.e. 'RVA 1 (3390-3)'

Note that the Extract program is going to attempt to combine LCUs into BCUs. It does this by device type. So even though a 2105 may have 6 or more LCUs. The extract program will see that all of these are for the same type model serial device, and all of the devices will be accumulated under one BCU.

Review the PGN map

The PGN map will list all of the performance groups that the system sees. Normally this is much more granular than is useful for capacity planning. Eight really should be viewed as the maximum number of performance groups for Capacity Planning. The DESC field is the field used to group these. So decide on what your groupings should be (usually fewer than 8) and then for each system performance group in the corresponding DESC= field enter your grouping name. Normally these would be names like: System, Production Batch, Test Batch, TSO, Production CICS and Test CICS. The DESC field will contain the comments from the IPS (if there are any) or the workload dot the service class if in Goal Mode. Change these to your grouping names. Note that an equals sign '=' is not permitted in this field. If it was in the comments of your IPS, be sure to remove it. If you have 100 system performance groups you will need to enter one of these grouping names in the DESC field for each of the 100 system performance groups.

Note that there is now the facility in CP2000 to combine performance groups. So if you are unsure whether to combine two groups or not you can leave them separate. Then when you are in CP2000 and can see all of the data, you can make a more intelligent decision whether to combine or not.

CP2000 also supports the concept of a workload TYPE. For sysplex workloads CP2000 also does special processing for these. The special types are CICS/IMS, CICS/DB2, IMS/IMS etc. You may add, for example TYPE=CICS/IMS, to any of these workloads and it will be carried into the EDF file and across to CP2000. An asterisk in front of the workload type is the key that it is a sysplex workload. i.e. TYPE=*CICS/IMS.

The Second Run - Data

This job will build the EDFI and Disk Magic files.

Customize the JCL

- If you want Disk Magic output, uncomment the MAGIC001 DD statement.

- If you uncommented the IPS001 DD, comment it again now.

CP2000 Extract Parameters

- Remove the PGN=IPS or PGN=GOAL.
- Remove the BCU=AUTO statement.
- Remove the PROCESS=NO statement.
- SORT=NO/YES. Sometimes operational procedures will leave the SMF records out of time sequence. If they are only mildly out of sequence or in reverse sequence, the Extract program will handle them. If they are significantly out of sequence, the samples will not line up properly and other unfortunate things may happen. If you specify SORT=YES then the extract program will dynamically invoke an internal sort. This will take additional resources and time. Only use this option if you expect that SMF records are out of sequence, or an earlier run indicated they are out of sequence.
- CECID= is used to specify the name of the CEC. This value will default to the SYSID for basic mode machines. For LPAR machines the default is CECnnnn where nnnn are the last 4 digits of the machine serial number. This will be useful for combining LPARs in CP2000.
- For older Sysplex system it used to be that the type of Coupling Facility was not reported in the RMF record. RMF now reports this data. If you are processing very early coupling facility data, you may need to specify CF=(nnnnnnnn,ttttttt). This is used to specify the Coupling Facility model. In the nnnnn field specify the coupling facility name and in the ttttttt field enter the type model such as 9674-R12.

Submit the job

Please review the messages produced by the Extract program

Downloads

The EDFI file will be in the member referenced by the EDF001 DD statement. If you are also producing MAGIC output, that will be in the member referenced by the MAGIC001 DD statement.

These files should be downloaded using normal ASCII CRLF protocol. The suggested extension is EDF, so CP2000 can find it. To have Disk Magic process the MAGIC file, simply do a FILE, OPEN on this file from within Disk Magic. If you have multiple systems, please read the Disk Magic considerations "Disk Magic Output" on page 22.

Stability Analysis (COV Processing)

A Stability Analysis is a study of data from before and after a processor upgrade. Individual units of work are analyzed to determine if they are “stable”. That is: they have the same characteristics both before and after the upgrade. Then of those units of work which are stable an average is used to determine the performance improvement delivered by the upgrade.

To do a Stability Analysis there are 3 steps. The first step must be done once for both the before and after data.

1. Use the EXTRACT program to create a TYP30001 extract file of type 30 subtype 4 data.
2. Use the COV program to sort and compare and compute whether the data is “stable.”
3. Use zMCAT to determine the performance improvment seen and optionally create a deliverable document.

Sample JCL is distributed in CPSTOOLS.JCL to do steps one (EXTRACT) and two (COV30). Comments are in the JCL to guide you. Then, you will need to download in ASCII the file from step two to use as input to zMCAT. (You may also wish to look at the units of work which were not selected. There is another dataset that can be created and loaded into zMCAT to see why a certain job was not found to be “stable.” See the JCL for the COV program to create this additional dataset, which can also be loaded into zMCAT.)

The JCL assumes that you would like to examine the systems both before and after in CP2000. If this is not the case and you wish only to create the COV file (and optionally REJECT file), sample JCL (STAN) can be used to do steps one and two in a single operation.

Required Parameters.

The parameters accepted by CP2KEXTR are below. Parameters **MUST** begin in column 1 and extend to column 72. One parameter per line. A sample of the input is found in "JCL requirements" on page 15.

You may have comments statements in any of CP2KEXTR's input streams: SYSIN, PGNMAP, BCUMAP or T30MAP. Comments are identified by an asterisk, '*' in the first column.

ENT=’cc..cc’ The Enterprise name is specified as a character string in single quotes. Imbedded blanks are permitted.

SYSID=cccc The 4 character SYSID as it appears in the SMF records. This is used to select only that SYSID from the SMF input. SYSID=(cccc) is also accepted.

Sample Parameters

DATE=(MM/DD/YY-MM/DD/YY) The inclusive interval specified will be used to select RMF records. Multiple DATE parameters can appear separately. If no DATE parameter is specified, all dates in the input file will be used.
DATE=(MM/DD/YY) is also accepted, as well as
DATE=(MM/DD/YY,MM/DD/YY,MM/DD/YY).

DURATION=hh The duration value asks to have the RMF interval summarized into hh hour samples for the time interval specified. The time intervals must be a multiple of the duration. An alternative form of hh:mm is also supported. You must be careful to make sure that this value is a multiple of the RMF interval. Thus if your RMF has a 30 minute

interval be sure your duration is 30 minutes or some multiple of 30 minutes. If the DURATION is not specified the program uses a default of one hour. (DURATION=1)

TIME=(hh-hh) The inclusive interval in hours only for the record selection. This means that if the following is specified TIME=(09-11), an RMF record for an interval beginning at 10:59 with a duration of 15 minutes will not be included but a record beginning at 8:59, for the same duration, will be included. Multiple TIME parameters can be specified. The duration value must be specified first.

An alternative of TIME=(hh:mm-hh:mm) is also supported. Using this form be sure that the DURATION fits evenly in this time window. This parameter may be omitted if the SELECT statement is used to select times to be studied.

If neither the TIME= nor the SELECT= is specified, the program defaults to TIME=(00-24).

SELECT DATE=MM/DD/YY,TIME=HH-HH

EXCLUDE DATE=MM/DD/YY,TIME=HH-HH These statements may be used to include or exclude certain times from analysis. For example, you are studying a customer's prime shift during a week, but they are open late on Thursdays. You may add additional hours to the study for Thursday. In addition on Tuesday morning there was a hardware failure which made the data from 9-10 invalid. You can EXCLUDE that data.

Also it is possible to simply look at discrete times. For example if the study of a week's data should reveal that there are three peak hours, it is possible to SELECT only those three hours for study.

Note that TIME=HH:MM-HH:MM is also supported.

Miscellaneous Parameters

CECID=cccccccc Up to 8 characters for the CECID. If omitted, the SYSID will be used for machines in basic mode. For machines in LPAR Mode, the default is CECnnnn where nnnn are the last 4 digits of the machine serial number. No imbedded blanks.

CICS=ONEINTERVAL Used with SMF record 110 processing. This parameter causes only one interval including all data to be output, rather than data for each DURATION to be output individually.

CICS=TWOPASS Used with SMF record 110 processing. This parameter causes the 110 records processing routine to make an initial pass of the input data from start to finish to collect all dictionary data, before processing begins.

CF=(nnnnnnnnn,tttttttt) Early coupling facility SMF data does not contain the type of coupling facility which is needed by CP2000. So specify the nnnnnnnn for the coupling facility name, and tttttttt as the coupling facility type, e.g. 9674-R12.

Specify as many CF= statements as there are coupling facilities.

CPUMOD=ccccccccccc A 10 character CPU models. If this is not specified, a table lookup will be attempted using the CPU model and version code supplied in the RMF type 70 record. This can fail for a few reasons.

- The model is just not in the table.
- Running under VM causes the version code to be x'FF' and therefore can't be deciphered.

•Running in physical partition and the supplied model does not reflect the partition. E.g. a PP 3090-600J will still show a 3090-600J even though it is a partition with 3 CPs and should be shown as a 3090-300J.

PROCESS=YES/NO If PROCESS=NO is encountered, only parameter checking is done and a BCU mapping may be built if BCU=AUTO is specified, and a performance group mapping will be built if PGN=IPS or PGN=GOAL is specified.

RMFINT=nn Number of RMF intervals to be used when calculating the peak to average ratio. Any consecutive nn intervals will be considered. Default is one.

SORT=YES/NO Default is NO. Coding YES will cause CP2KEXTR to do an internal sort on only the SMF data required for this study. In most cases the SMF data is already in time sequence and this sort is not required. If you should see interval starts or durations which are strange, or if CP2KEXTR should give you a message to sort the data, then a sort is in order.

When using the internal sort, the REGION= parameter should be adjusted as high as reasonable. Also some thought should be given to the size of the sort work datasets. Studying A CICS system of about 150 MIPS with a week of prime shift RMF data requires 36 cylinders of sort work space. If you have a larger I/O configuration, add a bit. If you are studying 42s, add about 50 percent. If you are studying 30s multiply by 3. A TSO or batch system will generally write more SMF records than a CICS or IMS system, so adjust your sort work dataset sizes accordingly.

SAVE=DICT SAVE=DICT is used in conjunction with CICS 110 record processing. It will cause all 110 dictionary records to be written to the SMFSAVE DD statement.

SAVE=SORT=ALL/sysid SAVE=SORT= is used to take a tape dataset and copy it to disk while sorting the dataset and eliminating duplicate records. If ALL is specified then all records are written to disk, and if a sysid is specified then only records for that one sysid are written to disk. A report of the sysids encountered, record types and subtypes encountered and earliest and latest date and times will be written to the SYSPRINT DD statement. Note that no other processing is permitted with this paramter. If this parameter is specified no other parameters are allowed except TYPES.

SUB30=n,n,n... Used with SMF records 30 processing. This parameter is used to determine which type 30 subtypes are included in the TYP30001 file. Default is SUB30=4. This parameter is not used for the T30MAP processing, which always uses type 2 and 3.

TYPES=30(2,3,4,5),42(6),70(1),71(1),72(1,3),73(1),74(1,4,5),75(1),78(1,3) This parameter can only be specified with SAVE=SORT=. It defines the records which are to be written to the disk dataset. If this statement is omitted, then the default which is specified here is taken.

Workload Data

PGN=GOAL This option will have the program scan the SMF data and read the SMF type 72 subtype 3 records and build a PGNMAP in the PGN001 file. See "The PGN Map File (GOAL Mode)" on page 19.

PGN=IPS This option will build a workload mapping of performance groups from the IPS001 file. You should consider using this in conjunction with the PROCESS=NO to have the workload mapping built from the IPS and stop. You would then EDIT the output (in PGNMAP file), and then rerun the extract program **without** the PGN=IPS parameter. There are some assumptions for the IPS.

•The dataset is described by the //IPS001 DD statement.

- The statement APGRNG= is before any PGN= statement.

Note that the APGRNG is no longer supported with MVS V5. Even if you specify it, the system will behave as though APGRNG=(0-15) had been specified. Therefore if you are running MVS V5, be sure to remove this parameter, or set it to 0-15. At the time this processing takes place the Extract program does not know what release of MVS is involved and will assume parameters are correct.

- The DP= parameter is on the same line as the PGN=. If not, a default of priority 0 will be used.
- If there is a /* */ combination on the PGN= line, the enclosed text will be used as a description for the workload. Otherwise a default of NONE will be used.
- A performance group will not be split by period when using the IPS. This can be done manually.

TYPE42=YES/NO Default is YES. This option will build records for each of the workload describing the dataset usage by business unit. This data will be built from type 42, subtype 6 records.

There are some assumptions for the type 42 records processing

- A PGNMAP was supplied.
- BCU=DASD or ALL was specified.
- SHOWACT=NONE was not specified.

NOTE In order to record the SMF 42 subtype 6 records, the client must be at DFSMS 1.1 or 1.2, have the DFSMS address space active, and allow recording of SMF 42s in the SMFPRMxx of SYS1.PARMLIB. In addition, the address space must be subject to SMF type 30 interval recording, (otherwise you won't get statistics for started tasks like CICS, DB2, etc).

DASD/Tape Data Parameters

How the Map is built.

When you request that a BCU Map be built through the AUTO parameters below, the processing is as follows: If IOCP001 DD has the IOCDS, then the IOCP data first is read into control blocks. Then in all cases the SMF data is read into control blocks. Next combining of Disk LCUs into BCUs takes place. The first combining takes place using NED data. NED data is control unit/device data which a device sends to the host in response to a particular command. Control unit NED data is in 74.5 records and device NED data is in 74.1 records. Using the NED data the program can determine what is a physical box, and will combine all LCUs and channels for a given NED id into one BCU. If BCU=FICONAG is specified, then only LCUs with identical channel configurations will be combined based on NED data. (This normally results in two BCUs for a shark control unit.)

At this point we should have essentially one BCU per box (unless we have used BCU=FICONAG) If BCU=COM has been specified, additional combining will take place. (Note: BCU=COM should not be used for a Disk Magic Study.) All control units with identical channel configurations are combined. This means that control units on copper channels where the control units are daisy chained are combined. Also if BCU=COM is specified then ESCON control units which share the same directors will be combined. (This does not happen with the IOCDS option, because the channel information now also includes the ESCON switch and port. Naturally these cannot be identical, and so will not be combined.)

BCU=NO/DASD/TAPE/OTHER/ALL BCU data is requested. This maps the volume data to BCUs by SYSID and address. An input file BCU001 must be provided. DASD means to only process the BCUD macros in the BCU map. TAPE means to only process the BCUT macros in the BCU map. Likewise OTHER means the BCUO macros. You may also specify BCU=DASD,TAPE to get only the DASD and TAPE processed. A brief description of a BCU is provided in "BCU Mapping File" on page 17. Default is ALL.

BCU=AUTO/AUTOADD The I/O configuration can generally be automatically built with the use of this parameter. The map is more complete if 74 subtype 5 records are available. If AUTO is specified, the BCUID will be "V" concatenated with the first VOLSER on the LCU. If no VOLSER is found, the name will default to "L" concatenated with the LCU number. Note that this means, for shared DASD, the BCUID may not be unique. This should be changed to be the same value on all systems, preferably a meaningful value to the customer.

If the IOCPnnn DD statement is found, then this will also be read to create a more complete BCUMAP. If AUTOADD is specified, the BCUID will be "A" concatenated with the 4 digit hex address of the lowest numbered device on this BCU. If you will be using CP2000 to analyze shared DASD, CP2000 will expect that a given BCU will have the same BCUID from each system. This will more likely happen automatically with BCU=AUTO than with BCU=AUTOADD.

BCU=LCU/COM BCU=COM will cause the Extract program to combine BCUs which have the same channels (and switches and links if IOCP is used.) This option should not be used with a Disk Magic study. Thus if two BCUs share channels 41, 42, 91, 92 then BCU=COM will cause that to appear as one BCU in the BCUMAP. This only has effect when the BCUMAP is being built in response to BCU=AUTO or BCU=IOCP. Note that the BCU=AUTO function will attempt to combine different LCUs that the program can determine are part of the same control unit. This happens by default. BCU=COM will combine different control units which share the same set of paths. The default is BCU=LCU, which today is a bit of a misnomer, since today one control unit is generally not one LCU. But BCU=LCU should be interpreted for the purposes of this program to mean each control unit should be listed separately.

BCU=FICONAG There seem to be cases in ESCON studies where it is desired that parts of physical boxes not be combined. These cases are when different groups of LCUs connect to different groups of chpids. If you specify BCU=FICONAG, then these LCU groups will be kept separate in the BCUMAP.

MAGIC=DETAIL/SUMMARY Disk Magic can process either device level input or control unit input. If you specify DETAIL, the extract program will produce device level input. SUMMARY will produce control unit input. The default is DETAIL.

PART=nnnnnnnn This parameter specifies the partition name which is used to process the IOCP file in an LPAR environment. It is only used with BCU=AUTO, when an IOCPnnn DD statement is provided. The default is the SYSID specification.

SHOWACT=ALL/NONE Specifying NONE will suppress all actuator data. CP2000 does not yet support files produced by using a value here, but SHOWACT=.001 will cause only those actuators which have I/O rates of .001 or greater to be output. NOT YET TO BE USED WITH CP2000.

Return Codes

Program can return 4 different return codes:

- 16 Something is drastically wrong and the program failed.
- 12 Something is critically wrong with the parameters, or no data fit within the SYSID and timeframe requested.
- 8 Either important data (RMF types) were missing from the file, or there is questionable data in some of the intervals which you may want to exclude.
- 4 Messages of note have been issued.

JCL requirements

The JCL is dependent upon the options selected. The following sample is the one shipped in the package. It builds the PGNMAP and BCUMAP files. Follow the directions in topic "Quick Start Guide" on page 5 to create the EDFI and Disk Magic files given below.

```
////#PATTRSN JOB   (????,????),MSGLEVEL=1,MSGCLASS=O,NOTIFY=????????
//*
//* THIS IS JCL TO RUN THE CP2000 EXTRACT PROGRAM
//* SEE THE USER'S GUIDE FOR COMPLETE DESCRIPTION OF
//* PARAMETERS AND DD STATEMENTS
//*
//* FIND THE USER'S GUIDE AT
//* W3.IBM.COM/SUPPORT/AMERICAS/WSC/CPSPRODUCTS.HTML
//* (NOTE THE URL ABOVE MUST BE IN LOWER CASE.)
//*
//EXTR     EXEC  PGM=LOADER
//* PRINT DATA SET FOR MESSAGES FROM YOUR RUN
//PRINT001  DD   SYSOUT=*
//* INPUT SMF DATASET
//SMFIN    DD    DISP=SHR,DSN=PATTRSN.RMF.CTS51T
//* OUTPUT DATASET FOR EDFI FILE
//EDF001    DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(EDFI)
//* OUTPUT DATASET FOR WORKLOAD GROUPING DATASET
//PGN001    DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(PGNMAP)
//* OUTPUT DATASET FOR AUTO BCU MAP
//BCU001    DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(BCUMAP)
//* COMPAT MODE PROCESSING OF IEAIPS TO GENERATE PGNMAP
//*IPS001   DD   DSN=SYS1.PARMLIB(IEAIPS00),DISP=SHR
//* INPUT DATASET FOR IOCP PROCESSING (FICON ANALYSIS)
//*IOCP001  DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(IOCP)
//* OUTPUT DATASET FOR DISK MAGIC OUTPUT
//*MAGIC001 DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(MAGIC)
//* DDS FOR OPTIONAL INTERNAL SORT OF SMF RECORDS
//*SORTMSGS DD   SYSOUT=*
//*SYSOUT   DD   SYSOUT=*
//*SORTWK01 DD   UNIT=SYSDA,SPACE=(CYL,(200))
//*SORTWK02 DD   UNIT=SYSDA,SPACE=(CYL,(200))
//*SORTWK03 DD   UNIT=SYSDA,SPACE=(CYL,(200))
//* # OUTPUT DATASET FOR TYPE30 RECORDS
//*   NOTE:  DCB INFORMATION FOR THESE DATASETS IS SUBJECT TO CHANGE.
//*         SO IT IS BEST TO LET THE PROGRAM ALLOCATE THEM.
//*TYP30001 DD   SPACE=(CYL,(20,10),RLSE),DISP=(,CATLG),UNIT=SYSDA,
//*           DSN=PATTRSN.TYPE30.DATA
//* # OUTPUT DATASET FOR SAVE= PARAMETER (SAVE CP2K'S SMF RECORDS)
//*SMFSAVE  DD   SPACE=(CYL,(20,10),RLSE),DISP=(,CATLG),UNIT=SYSDA,
//*           BLKSIZE=8192,DSN=PATTRSN.SMF.SAVE
//* # OPTIONAL USER DUMP DATASET
```

```
//*SYSUDUMP DD   SYSOUT=*
//SYSIN001  DD   *
ENT='CP2000 INC.'
SYSID=P84M
DURATION=1
DATE=(09/13/94)
TIME=(08-12)
PGN=GOAL          OR PGN=IPS IF YOU ARE IN COMP MODE
BCU=AUTO          READ SMF AND GENERATE BCUMAP
PROCESS=NO         JUST GENERATE PGN & BCU AND STOP.
/*
//* MESSAGES FROM THE MVS LOADER
//SYSLOUT  DD    SYSOUT=*
//* THE OBJECT MODULE OF THE PROGRAM
//SYSLIN    DD   DISP=SHR,DSN=PATTRSN.CPSTOOLS.JCL(ZOBJEXTR)
```

PRINT001	CP2KEXTR messages.
SYSLOUT	LOADER messages.
EDF001	Is the EDFI output. Logical record length is 80. Mandatory.
MAGIC001	Disk Magic interface output file. Logical record length is 80. Optional.
BCU001	The BCU mapping. Logical record length is 80. Optional.
IPS001	The IEAIPS member that was used when the system was running. Logical record length is 80. Required for PGN=IPS.
IOCP001	The IOCP dataset. Logical record length must be 80. Mandatory when BCU=IOCP is coded.
PGN001	The performance group mapping. Logical record length is 80. Optional.
SORTMSGS	Sort Messages. Required for SORT=YES.
SORTWK0x	Sort Work Space. May be required for SORT=YES.
SMFIN	The input SMF dataset. Mandatory.
SMFSAVE	SMF Records to be saved for analysis in future runs. Required if SAVE= is specified.
SYSIN	Input parameters to CP2KEXTR. Mandatory.
TYP30001	Dataset for extracts of TYPE30 records. Optional.
TYP42001	Dataset for extracts of TYPE42 records. Optional.
TYPE001	Dataset for extracts of TYPE110 records. Optional.
T30M001	The workload decomposition group mapping to be used with type 30 record processing. Logical record length is 80. Optional.

SMF input data requirements

SMF records are expected to be sorted by DATE and Time. If they are not, use SORT=YES control statement. Or use the JCL in "SORT JCL" on page 31 to rectify that situation. While at it, the file can be reduced in size by selecting only the required records. Note - if you don't want any records eliminated, remove the INCLUDE statement.

In addition the Extract program will eliminate duplicate records in the input stream. Occasionally, customer procedures will be such that records are duplicated in the SMF input stream. If records are discarded a message will indicate how many have been discarded.

Type 30	Common Address Space Work. Required for T30MAP processing or creating the TYPE30 dataset. Subtypes 2, 3 and 4.

Type 42 DFP Statistics. Required for TYPE42=YES. Subtype 6.
Type 70 CPU activity record. Required.
Type 71 Paging Activity. Subtype 1 Required.
Type 72 Workload Activity. Required if PGNMAP DD is specified.
Type 73 Channel Path Activity. Required if PGNMAP DD is specified.
Type 74 Device Activity. Required if PGNMAP DD is specified.
Type 75 Page/Swap Data Set Activity. Required.
Type 78 I/O Queuing Activity. Subtype I or III required only if BCU= is specified.
Type 110 CICS Activity. Required only if TYPE110 data is desired.

Record Types 76, 77, and 79 are currently ignored.

Note. If you have CMF data (Boole and Babbage's equivalent of monitor I), their data is being successfully processed by CP2000 Extract Program.

BCU Mapping File

The BCU map provides an address map for MVS data which identifies the BCU characteristics. Three types of statements can be in the BCU map: TIMESTMP has environmental information. There is only one of these and it must be first. BCUx where x is either D, T or O defines the BCU characteristics. ADDRS defines address ranges.

It is important that the label on the BCUx statements match across the different systems so that CP2000 can correlate them.

An example is contained in the JCL sample.

The progam goes through the following steps to build the BCU map:

- If IOCP data is read, control blocks are built from these for control units and devices.
- Then SMF records are read:
  - Type 74 subtype 1 for device information.
  - Type 78 subtype 1 for LCU to path information
  - Type 74 subtype 5for control unit caches and names.
- The information is reduced and attempts are made to determine where LCUs should be combined, so that the many LCUs of a single current technology control unit is reported as only one control unit.
- If there are 74 subtype 5 records, then the NED id is used to determine type/model, manufacturer and serial number. Here accurate combining is assured. Without 74 subtype 5 data there is device level NED data in the 74 subtype 1 records, but it is at the device level. If they are all reporting the same for all devices on a control unit (which seems to be the case with late model control units like Shark.), then it is possible to correctly combine the devices.

Table BCUMAP Definitions

Label	Opcode	Operands
	TIMESTMP Defines Environmental information	**SYSID=** Generated for both BCU=IOCP and BCU=AUTO. Required.
		PART= Generated for BCU=IOCP. The partition name in the IOCP definition.
		IOCPDATE= Generated for BCU=IOCP. The date from the IOCP definition if present.

		IOCPTIME= Generated for BCU=IOCP. The time from the IOCP definition if present.
		IODFDATE= Generated for BCU=IOCP. Last date of the IODF file from type 78 records.
		IODFTIME= Generated for BCU=IOCP. The time of the IODF file from type 78 records.
A character string to uniquely identify this BCU across the enterprise. For shared DASD, this string should be the same for each SYSID mapping.	**BCUD, BCUT or BCUO** The BCU identifier. D is for DASD. T is for Tape and O is for other.	**CTYPE=** Required for RMF before 4.2.0. Device type, e.g. 3990-3.
		CACHE= Cache size if controller is cached. Defaults to 0.
		NVS= Non-volatile storage size if controller is cached. Defaults to 0.
		CHPID= The chpids used by this BCU.
		SWCHS= The switches corresponding to the CHPIDs.
		LINKS= The switch links corresponding to the CHPIDs.
		CHPTYP= The CHPID types corresponding to the CHPIDs.
	ADDRS The address identifier.	**AD=** The range of addresses defined by this line.
		LCU= The LCU number for these addresses.
		RMF= Either Y or N whether RMF data was found for this range of devices or not.
		DEVT= The device type (e.g. 3380K).

Note that the BCU=AUTO/IOCP parameter is useful in conjunction with PROCESS=NO. In combination with the PROCESS=NO parameter, a BCUMAP file could be built and used as a start for editing. This avoids a lot of typing.

The PGN Map File (Not GOAL Mode)

The requirement that performance groups be mapped into business units for capacity planning leads to the data reduction requirement that a file be available which maps the performance groups (in MVS) to business units.

The mapping is from Performance Group (PGN) to Description (DESC). In the examples below, ("Sample PGN" on page 28) both performance groups 3 and 5 are combined into a business unit called "WORK".

A mapping for each SYSID can be provided or where a performance group maps to a business unit for all SYSIDs to be processed, an "*" is permitted indicating any SYSID.

One workload name can now appear across multiple SYSIDs in CP2000.

The TYPE field is used to designate the application type (TSO, CICS, etc.) for each business unit. This parameter is optional.

The PRTY, priority, field is a relative priority of the workload. This may be derived from the IPS actually used using the PGN=IPS parameter already described. It is used in CP2000 for processor response time computations and is optional.

The PGN mapping input contains the following parameters on each line.

SYSID= A character string to identify the SMF ID of the system image corresponding to this data. If this is valid for all images, an '*' may be used.

PGN= The performance group number or the character string '*'. If input is from PMMVS, this can be 'GEN'. The mapping will be processed in order so that any occurrence of PGN=* will negate any subsequent statements.

PER= An optional specification of performance group period. A specification of PER=* (or PER=GEN for PMMVS), which is the default, will group all periods. This parameter is optional. In the sample "Sample PGN" on page 28, PGN=2 is divided into two groups. TSO Trivial from period 1 and TSO Complex for all other periods.

PRTY= Relative priority of the performance group. Parameter is optional and defaults to 0.

DESC= A unique character string describing the business unit. Field length for CP2KEXTR is 48, but CP2000 will only display the first 26 characters. Note that an equals sign '=' is not allowed in this field. If there was one in your IPS comments be sure to remove it. (This is the TYPE= statement in PMMVS style of input.)

TYPE= Optional parameter to specify the workloads LSPR type. (E.g., TSO, CICS, BATCH,...) Field length is 9 characters. If the first character of this field is an asterisk '*', then This workload will be assumed to be a sysplex workload and will be so indicated to CP2000. (The asterisk will not be passed as part of the type name.) This is used in CP2000 for Parallel Sysplex (PTS) Capacity Planning. Types of interest for PTS are:

- CICS/IMS
- IMS/IMS
- CICS/DB2
- IMS/DB2
- CICS/VSAM
- IMS/FP

The PGN Map File (GOAL Mode)

Although there are no PGNs in GOAL Mode, it is still called the PGN Map file.

In compatibility mode, the data was gathered into buckets by PGN. In GOAL Mode it is defined by the WORKLOAD name (R723MWNM) and the Service Class name (R723MCMN). The Workload and Service Class data is mapped to a unique description. (PGN=GOAL) builds a description of

WORKLOAD.SCLASS as a default.) Each description defines a unique business unit. In goal mode, to combine data from different workload and service class, make their descriptions the same.

In the example below, ("Sample PGN Map (Goal Mode)" on page 29) both "SSYSTEM.SYSTEM" and "SYSTEM.SYSSTC" are combined into one unit called "STUFF"..

A mapping for each SYSID can be provided or where a performance group maps to a business unit for all SYSIDs to be processed, an "*" is permitted indicating any SYSID.

One workload name can now appear across multiple SYSIDs in CP2000.

The TYPE field is used to designate the application type (TSO, CICS, etc.) for each business unit. This parameter is optional.

The PRTY, priority, field is a relative priority of the workload. This is not so easy to establish in GOAL mode. CP2KEXTR will artificially construct a relative priority value from the R723CIMP field. It will be 11-R723CIMP. It is an optional field.

The PGN mapping file contains the following parameters. Two lines are allowed.

A "*" for a workload name means put anything not yet processed into this bucket. (See "The Rest" in the example.)

SYSID= A character string to identify the SMF ID of the system image corresponding to this data. If this is valid for all images, an '*' may be used.

PGN= This is ignored. PGN=0 is provided by PGN=GOAL as a default.

PER= An optional specification of performance group period. A specification of PER=* will group all periods. This parameter is optional.

PRTY= Relative priority of the performance group. Parameter is optional and defaults to 0.

DESC= A unique character string describing the business unit. Field length for CP2KEXTR is 48, but CP2000 will only display the first 26 characters.

WORKLOAD= A unique character string describing the Workload name as it would have appeared in an RMF workload report. Field length is 8 characters. If this is WORKLOAD=* then all data not already processed will be collected here. This should be the last entry in the PGNMAP file. See the example for 'The Rest'.

SCLASS= A unique character string describing the Service class name as it would have appeared in an RMF workload report. Field length is 8 characters.

TYPE= Optional parameter to specify the workloads LSPR type. (E.g., TSO, CICS, BATCH,...) Field length is 8 characters. If the first character of this field is an asterisk '*', then This workload will be assumed to be a sysplex workload and will be so indicated to CP2000. (The asterisk will not be passed as part of the type name.) This is used in CP2000 for Parallel Sysplex (PTS) Capacity Planning. Types of interest for PTS are:

- CICS/IMS
- IMS/IMS
- CICS/DB2
- IMS/DB2
- CICS/VSAM

The Type 30 Map File (T30M001)

So we are here because the type 72 workload does not do sufficient workload breakdowns for our purposes. Here we are going to do workload breakdown by the data that is in the type 30 interval records. (Be sure client is collecting type 30 interval records: subtype 2 and 3.)

There are some differences between the workload data generated by using SMF type 72 via a PGNMAP and using SMF TYPE 30 via a T30MAP.

- The T30 data will not include a lot of the SYSTEM information since SMF 30 is only recording address spaces started after SMF. That's not too much of a problem for the business unit capture ratio but can be for the overall system CR. If you use T72 data, the SYSTEM work is often deleted anyway.
- The T30 data offers you the opportunity to break down the workload in CP2000 to a much finer granularity. Often all of CICS is put into one service class which means that it will appear as one workload in CP2000. You can split the CICS workload by using T30 data and jobname as the classification of work.
- You should remember to make the DUR= in CP2KEXTR larger than the type 30 interval.

This option will generate an additional set of workloads like those resulting from the PGNMAP with a different header in the EDFI file. Once in CP2000 you can only process one of these at a time. By default it will be the PGN001 data, but you can set the option in CP2000, so that it will read the T30M001 instead.

This file will is very similar to the PGN map, except there is no way to automatically generate one. Therefore you will need to code this mapping file yourself. Reference "Sample T30M001 Data Set" on page 28. The program looks through this map and the first match it finds will be the entry where the data is accumulated.

Each entry begins with SYSID= in column one and can be at most two lines long. It can have at most one field for job name or racf userid etc. (Note WLM= and SCN= must both be specified together.) They will have a mask associated with them. For example JOBN=PROD* or JOBN=TEST%% will collect either all jobs which begin with PROD or all jobs that begin with TEST and two characters. To combine these entries in the output the DESC fields for both of these should be the same.

A mapping for each SYSID can be provided or where a performance group maps to a business unit for all SYSIDs to be processed, an "*" is permitted indicating any SYSID.

One workload name can now appear across multiple SYSIDs in CP2000.

The TYPE field is used to designate the application type (TSO, CICS, etc) for each business unit. This parameter is optional.

The PRTY, priority, field is a relative priority of the workload. You will need to somehow arrive at values for the different business units. It is an optional field.

If a type 30 record is found which does not satisfy the criteria of one of the entries, the program will create a new entry for this data. This entry will have a description of either "workload name.service class name" or "PGNnnn" where nnn is the performance group number.

The PGN mapping file contains the following parameters. Two lines are allowed. SYSID in column 1 begins the entry.

SYSID= A character string to identify the SMF ID of the system image corresponding to this data. If this is valid for all images, an '*' may be used.

PRTY= Relative priority of the performance group. Parameter is optional and defaults to 0.

DESC= A unique character string describing the business unit. Field length for CP2KEXTR is 48, but CP2000 will only display the first 26 characters.
Workload descriptions must be unique for each SYSID.

TYPE= Optional parameter to specify the workloads LSPR type. (E.g., TSO, CICS, BATCH,...) Field length is 8 characters. If the first character of this field is an asterisk '*', then This workload will be assumed to be a sysplex workload and will be so indicated to CP2000. (The asterisk will not be passed as part of the type name.) This is used in CP2000 for Parallel Sysplex (PTS) Capacity Planning. Types of interest for PTS are:

- CICS/IMS
- IMS/IMS
- CICS/DB2
- IMS/DB2
- CICS/VSAM

JOBN= The job name indicated in the type 30 record.

CLASS= The job class indicated in the type 30 record.

USER= The RACF userid in the type 30 record.

RACF= The RACF group in the type 30 record.

PROG= The program name in the type 30 record.

PGN= The performance group in the type 30 record.

WLM=/SCN= The workload manager name and the service class name. Both must be specified.
See the example below "Sample T30MAP Data Set" on page 28.

Disk Magic Output

In order to produce Disk Magic output the CP2000 Extract program requires SMF record types 70, 73, 74 and 78. If any of these are missing, then the output file will not be created. Also it is highly recommended that RMF cache data be used as well. (Type 74.5)

If you have multiple systems you will get a Disk Magic output file for each one. It is important that the control unit names in the Disk Magic output files match for all of the systems. This control unit name comes from the label on the BCUD statement in the BCUMAP. If you use the BCU=AUTO facility of the extract program, then the BCUIDs will be 'V' and the volser of the first volume on the control unit.

You will need to combine the output files before you read them into Disk Magic. The order in which you combine the files is important: The last occurrence of the "CU" statement is the one which is used by Disk Magic. So if you have a test system with a few devices and/or channels and a production system with more, be sure to concatenate the production system last. To do this pick your favorite editor on the first file. Then "Import text file" the next one, after the first. You will need to manually remove the "new" and "model" statements from the beginning of the new file. Disk Magic expects only one of these in the file. Then import the next file and remove the "new" and "model" statements for these. And so on until all of your systems are in one file. Now you can load this file into Disk Magic.

If you are modeling RVA 9393 DASD the cache uses compression and you should manually double the size of the cache in Disk Magic file.

Type 30, 42 and 110 Output

The CP2000 Extract program is now capable of creating extract files for these SMF record types. A subset of the fields are output to flat files for analysis. The 30 and 42 work somewhat differently from the 110 processing and will be treated separately. Note that these files may be created from files which do not contain RMF (70-79) records, however the program is engineered to create an EDFI file, and a 12 return code will result. The 30, 42 and 110 extract files will be correctly produced in this case.

In the cases of type 30 and 42: when the records are encountered in the input stream, the program will try to open DD statements called TYP30001 (for type 30 records) and TYP42001 (for type 42 records). If they open successfully the data will be written to these files.

CP2000 currently uses the type 30 file for its "Stability Analysis" function. Also CP2000 now has an analysis function for the type 30 file. This function will allow the user to better understand which programs or job classes or jobs or users are using what proportion of the processor resources. To use this function set up the TYPE30 DD to point to a dataset. Then after the run, FTP in ASCII the contents of the resulting file to your PC with a DAT suffix. You will then be able to open this file in the “General Data Comparison” “SMF Type 30 Analysis” function. There are no current users of the type 42 file.

The created files are self describing in that the first several records at the beginning have records to describe the content of the subsequent file.

Self defining records appear as follows:

```
*NAME=SMF30PGN,N,COLS=30,COLE=32
*NAME=SMF30TRS,N,COLS=155,COLE=164
*NAME=SMF30DTE,N,COLS=1,COLE=6
*NAME=SMF30TME,N,COLS=7,COLE=12
*NAME=SMF30STP,C,COLS=13,COLE=13
```

These fields are:

- NAME= the name of the field.
- Whether the output is numeric or character. (Note the output in the file will always be in character format. Thus the SMF30PGN above will have the format of 0xf0f0f2 or a printable '002')
- COLS= is the column in the output in which this field will start.
- COLE= is the column in the output in which this field will end. By default only subtype 4 type 30 records are processed. However by using an additional parameter 'SUB30=2,3,4,5', you may specify which subtypes are to be included. Note that SUB30=2,3 will not include subtype 4. The subtype of the record is included in the output, so the post processing program can know the source.

Type42 processing only includes the subtype 6. No provision is made to process any other subtype.

Since there tends to be much larger numbers of 110 records, the program collects them for a DURATION, then puts out the collected data by applid and transaction id. Thus the output file will contain the date, time, duration, applid and transaction name. Then the number of such transactions, average CPU time etc.

Since it is possible that the records may not be sorted, and there may be too many to sort without enormous consumption of resources, it is possible to have all the data collected for the entire run and output at the end. This is accomplished by using the CICS=ONEINTERVAL parameter, avoiding a large output file as the active interval bounces around.

To process 110 records the program needs 110 dictionary records. Normally these records are produced whenever a CICS region is started. They can also be produced with the CICS utility DFHMNDUP (see CICS Operations and Utilities Guide.). It may be that your CICS system programmer already has a dataset of these dictionary records which you can concatenate in front of the SMFIN dataset. This is the best approach.

Alternatively, you can see which dictionary records are in your SMF data. Since the program cannot process data for which it has not yet found a dictionary record, there is an option to cause the program to make an initial pass through all of the data only looking for dictionary records, before normal processing. This parameter is CICS=TWOPASS.

At the end of processing the 110 processing routines will produce a list of the number of 110 records it was able to process and not able to process by applid.

30 Data File

SMF30DTE	SMF30DTE	Date of record creation	
SMF30TME	SMF30TME	Time of record creation	
SMF30STP	SMF30STP	Record subtype	
SMF30JBN	SMF30JBN	Job Name	
SMF30PGM	SMF30PGM	Program name	
SMF30WLM	SMF30WLM	Workload Name	
SMF30SCN	SMF30SCN	Service Class	
SMF30RUD	SMF30RUD	Racf Userid	
SMF30TCN	SMF30TCN	Connect Time	Units unchanged
SMF30ABD	SMF30ABD	Completion	A = Abend N = Normal
SMF30PTY	SMF30PTY	Dispatching priority	Not valid in goal mode
CPUTIME	SMF30ICU, SMF30ISB, SMF30IIP, SMF30RCT, SMF30HPT, SMF30CPT, SMF30CPS	CPU Time	Hundredths of a second
SMF30TEP	SMF30TEP	Blocks Transferred	
SMF30PGI	SMF30PGI	Pages paged in	
SMF30NSW	SMF30NSW	Swap Sequences	
SMF30HPI	SMF30HPI	Hiperspace page ins	
CS	SMF30ARB, SMF30EAR, SMF30URB, SMF30EUR	Central Storage	Kilobytes
SMF30TAT	SMF30TAT	Transaction Active Time	Hundredths of a second
SMF30RES	SMF30RES	Transaction Residency Time	Hundredths of a second

SMF30PGN	SMF30PGN	Compatibility Mode Performance Group	
SMF30TRS	SMF30TRS	Number of transactions	
SMF30CLS	SMF30CLS	Job Class	
SMF30GRP	SMF30GRP	Racf Group	
SMF30SQT	SMF30SQT	Waiting for Initiator	Hundredths of a second
SMF30WID	SMF30WID	Subsytem identifier	
SMF30SRV	SMF30SRV	Total Service Units	Units
SMF30IO	SMF30IO	I/O Service Units	Units
SMF30CSU	SMF30CSU	CPU Service Units	Units
SMF30SRB	SMF30SRB	SRB Service Units	Units

Tape to Disk: The SAVE=SORT= Parameter

The Extract Program has a facility to copy an input dataset (usually from tape), sort the records removing duplicates, and write them to an output dataset (usually on disk.) As part of this processing a report will be written to the PRINT001 DD statement which lists all SYSIDs encountered, and for each SYSID the record types and subtypes encountered, and for each record type and subtype the earliest and latest time and date. No other processing can take place in such a run.

The input (tape) dataset must be allocated to the SMFIN DD statement. The output (disk) dataset will be written to the SMFSAVE DD statement. Give the job as much region and SORTWKnn's as you can.

The control statement to invoke this type of processing is SAVE=SORT=ALL, and no other statements except for TYPES described below are permitted.

There may be situations where you have received data from multiple SYSIDs on a single tape and you would prefer to keep the data for distinct SYSIDs in distinct datasets. By specifying SAVE=SORT=sysa, only the data for sysa will be written to the SMFSAVE DD statement. So if this is the case, and you have received a tape with data for SYSA, SYSB and SYSC, you would need to make one pass with SAVE=SORT=SYSA allocating a SYSA datasets, then another pass with SAVE=SORT=SYSB allocating a SYSB dataset, and again for SYSC.

By default the program will collect and sort the following record types:

TYPES=30(2,3,4,5),42(6),70(1),71(1),72(1,3),73(1),74(1,4,5),75(1),78(1,3)

The user may change this list adding or subtracting record types. This is the list of all records (except for 110s) that the Extract Program can process. If you are pressed for space and know that you will not be processing the type 30 records, you could place this statement in the input stream and remove the "30(2,3,4,5)" part. Then the output dataset will not contain type 30 records. This is the only other statement that is permitted with the SAVE=SORT= parameter.

Multiple Outputs with a Single Pass

There are cases where it would be nice to be able to create multiple output EDF Files from a single pass of the SMF data. Such cases might be to create an EDF for prime shift, second shift and all shifts with a single pass of the data. Thus creating 3 EDF files with different time parameters in a single run. To do this it is necessary to understand two different classes of DD statements. The first is a DD statement

which is used for all “subruns,” and the second is specific to this particular subrun. The DD statements that are specific to this particular subrun all end in a 3 digit number which the program uses to correlate them.

DDs which are used for all “subruns” and which are unchanged

SORTMSGS	Sort Messages. Required for SORT=YES.
SORTWK0x	Sort Work Space. May be required for SORT=YES.
SMFIN	The input SMF dataset. Mandatory.
SMFSAVE	SMF Records to be saved for analysis in future runs. Required if SAVE= is specified.
SYSLOUT	LOADER messages.
TYPE110	Dataset for extracts of TYPE110 records

New DDs which have new names

Old Name	New Name	Description
SYSIN	SYSINnnn	Input Parameters to CP2KEXTR.
SYSPRINT	PRINTnnn	CP2KEXTR messages.
EDFFILE	EDFnnn	The EDF output.
MAGIC	MAGICnnn	Disk Magic interface output file.
BCUMAP	BCUnnn	The BCU mapping.
IEAIPS	IPSnnn	The IEAIPS member that was used when the system was running.
IOCP	IOCPnnn	The IOCP dataset.
PGNMAP	PGNnnn	The performance group mapping.
T30MAP	T30Mnnn	The workload decomposition group mapping to be used with type 30 record processing
TYPE30	TYP30nnn	Dataset for extracts of type 30 records.
TYPE42	TYP42nnn	Dataset for extracts of type 42 records.
TYPE110	TYPEnnn	Dataset for extracts of type 110 records.

So to use this feature, you will need to setup your first “subrun” to use nnn of 001. So set up each of these new DDs with the correct files. Then your next “subrun” will use nnn of 002, and so on. The program will load the parameters for all of your subruns. If errors are found in any of the inputs then the run will stop. If all parameters are successfully processed, then PGNMAPs and BCUMAPs willl be generated and processed. Assuming no errors are found in this stage the program will continue.

The input parameters will apply only to the “subrun” in which they are specified with one exception. “SORT=YES” will cause the input to be sorted for all “subrun”s.

Sample EDFI output

```
HEAD  ENT=PRONTO SOURCE=CP2KEXTR11/02/01
CEC S CECID=P84M CPUMOD=9121-732 SUPVR=MVS/ESA VC=FF PR=3 SR=20005
SYS S SYSID=P84M SCP=MVS/ESA VERSION=SP5.1.0 HPTSID= NSAMPS=4 GL=0 WC=0 PAR=2.6
      RMFINT=1 RMFINTL=15 SRM=1413.18 GMTOF=-4:00 DASDIO=122.3 PAGE=8.2 CS=112
      ES=336 CSAVAIL=2.1 ESAVAIL=104.9 SCPCS=30.8 SCPES=9.8 SWTOT=0.069 1.153
      0.343 0 0 0.319 0 0 0.004 0.003 0 0.002 0.004 0 0 0 0 CPUWAIT=6958.9
      4329.4 7962.8 1520.7 4 7832.5 7710.6 7962.8 4329.4 CECUTILV=35.6 26.3
      59.9 14.1 4 27.5 28.6 26.3 59.9 PAGEV=8.2 0.5 26.5 10.7 4 26.5 0.5 0.5
      5.5 PAGEDS=6 6 6 0 4 CSV=112 112 112 0 4 ESV=336 336 336 0 4 CSAVAILV=2.1
      1.5 2.9 0.5 4 1.8 1.5 2.9 2.3 ESAVAILV=104.9 61.6 140.4 31.3 4 128.4
      140.4 89.3 61.6 SCPCSV=30.8 28.5 32.4 1.6 4 28.5 32.4 32.3 30.1
      SCPESV=9.8 8.8 10.5 0.7 4 10.4 8.8 10.5 9.4 PGTOES=338 78.2 839.5 308.4 4
      839.5 90.2 78.2 344.3 PGFROMES=320.7 68.5 834.4 310.8 4 834.4 78.7 68.5
      301.3 ESAUX=4.1 0 14.7 6.2 4 1.6 0 0 14.7 AUIC=134.7 67.2 171.6 41 4
      137.4 162.7 171.6 67.2 MUIC=4 0 9 4.1 4 0 9 7 0 MXUIC=254.5 254 255 0.5 4
      255 255 254 254 AMIGR=37872.2 21705.9 43261 9333.6 4 43261 43261 43261
      21705.9 MMIGR=37512.1 20265.3 43261 9957.5 4 43261 43261 43261 20265.3
      AINR=1.8 1.5 2.6 0.5 4 1.5 1.5 1.5 2.6 AIN=94.4 91.8 96.5 1.7 4 91.8 94.5
      94.6 96.5 AOUTR=0.5 0 1.6 0.6 4 1.6 0 0 0.4 LOGSWAP=98.3 96.5 99.9 1.6 4
      96.5 99.8 99.9 97 LOGEFF=79 70.4 86.6 6.3 4 75.9 83.1 86.6 70.4
      BATCHV=11.4 10.1 13.6 1.3 4 10.1 10.8 11.2 13.6 STCV=173.6 171.4 174.9
      1.3 4 174.9 174.3 173.8 171.4 TSOV=40.6 21.4 53.4 12.1 4 21.4 40 47.6
      53.4 ASCHV=0 0 0 0 4 OMVSV=0 0 0 0 4 OCPU=6.7 1.3 22.3 9 4 1.3 1.5 1.8
      22.3 OCPU2=3.6 0.4 13.1 5.5 4 0.4 0.4 0.7 13.1 LREV=0 0 0.1 0 4 0 0 0 0.1
      DASDIOV=122.3 61.1 174.8 40.5 4 61.1 123.2 130.2 174.8 DASDRSPV=23.3 20.5
      26.7 2.7 4 26.7 20.5 21 25.2 DASDSERV=19.1 17.3 22.8 2.2 4 22.8 17.3 17.6
      18.8 TAPEIOV=0 0 0 0 4 LPARNO=1 LPAR=P84M LPWGT=0 -1 0 LPCAP=0 LPPRC1=3 3
      3 0 4 LPPCTM1=10799.9 10799.7 10800.1 0.1 4 10800.1 10799.7 10800.1
      10799.9
SAMPS DATE=09/13/94 TIME=08:00 DUR=01:00 SIO=61.08 SPAGE=26.48 SSTOR=318
SAMPS DATE=09/13/94 TIME=09:00 DUR=01:00 SIO=123.24 SPAGE=0.47 SSTOR=306
SAMPS DATE=09/13/94 TIME=10:00 DUR=01:00 SIO=130.25 SPAGE=0.47 SSTOR=356
SAMPS DATE=09/13/94 TIME=11:00 DUR=01:00 SIO=174.78 SPAGE=5.55 SSTOR=384
WORKS WDESC=SYSTEM WTYPE= WIO=0.3 WCS=4.29 WES=2.29 WPAGE=0 WTRANSRS=0
      WPRTY=255 WMPL=8.03 WRESP=454628 MPLV=8 8 8 0 4 TRANV=0 0 0 0 4
      RESPV=113657.02 0 454628 196859.68 4 0 0 0 454628 WIOV=0 0 0 0 4
      WIORESPV=0 0 0 0 4 WCPUTM=83.2 34.9 180 58.4 4 180 39.6 34.9 78.4
      WPAGEV=0 0 0 0 4 WCSV=4.3 4 4.5 0.2 4 4 4.5 4.5 4.1 WESV=2.3 2.1 2.6 0.2
      4 2.6 2.1 2.1 2.3 WEXCPV=0.3 0.2 0.3 0 4 0.3 0.2 0.3 0.3
WORKS WDESC=BATCH P1 WTYPE= WIO=92.9 WCS=11.84 WES=12.38 WPAGE=0 WTRANSRS=0.01
      WPRTY=064 WMPL=8.65 WRESP=139.28 MPLV=8.6 6.5 10.4 1.4 4 6.5 8.6 9.2 10.4
      TRANV=0.01 0.01 0.01 0 4 RESPV=135.61 90.45 186.72 36.18 4 90.45 115.79
      186.72 149.49 WIOV=0 0 0 0 4 WIORESPV=0 0 0 0 4 WCPUTM=1195.5 279.3
      2450.7 791.2 4 279.3 900.7 1151.2 2450.7 WPAGEV=0 0 0.1 0 4 0 0.1 0 0.1
      WCSV=11.8 5.9 19.4 4.9 4 5.9 10.6 11.5 19.4 WESV=12.4 6.8 22.4 6 4 6.8
      9.3 11.1 22.4 WEXCPV=92.9 8.7 175.8 59.8 4 8.7 80.4 106.7 175.8
WDSNS WDSNV=SMSPU2 WDSNN=*SMSPU2 WDR=7.7 WDHR=33.3 WDHRT=16.1 WDC=3.6 WDD=12.8
```

```
      WDP=0.1 WDRT=24.1
WDSNS WDSNV=SMS005 WDSNN=*SMS005 WDR=0.5 WDHR=30.3 WDHRT=12 WDC=16.8 WDD=11
      WDP=0.1 WDRT=29.3
WDSNS WDSNV=PRN001 WDSNN=*PRN001 WDR=0.1 WDHR=27.4 WDHRT=34.9 WDC=13.4 WDD=13.7
      WDP=0.1 WDRT=30.2
WDSNS WDSNV=PRNLG1 WDSNN=*PRNLG1 WDR=0.1 WDHR=4.1 WDHRT=40.1 WDC=9.9 WDD=12
      WDP=0.1 WDRT=23.4
WDSNS WDSNV=PMOS08 WDSNN=CHURCHW.GECAP.CICS.PRMF WDR=0.3 WDHR=3.9 WDHRT=46.3
      WDC=40.6 WDD=11.8 WDP=0.1 WDRT=53.2
WDSNS WDSNV=SMS009 WDSNN=*SMS009 WDR=0.1 WDHR=1.2 WDHRT=22 WDC=5.8 WDD=10.2
      WDP=0.1 WDRT=16.8
WDSNS WDSNV=SMS012 WDSNN=*SMS012 WDR=0.7 WDHR=4.7 WDHRT=17.5 WDC=9 WDD=12.1
      WDP=0.1 WDRT=25.1
WDSNS WDSNV=PMOS29 WDSNN=CHURCHW.GECAP.CICS.PWF.INDEX WDR=0.2 WDHR=2.6
      WDHRT=21.6 WDC=8.1 WDD=12.8 WDP=0.1 WDRT=21.6
WDSNS WDSNV=PMOS29 WDSNN=CHURCHW.GECAP.CICS.PWF.DATA WDR=0.6 WDHR=9.1
      WDHRT=19.7 WDC=9.3 WDD=9.6 WDP=0.1 WDRT=19.7
WDSNS WDSNV=SMS013 WDSNN=CHURCHW.GECAP.CICS.PMF.INDEX WDR=0.2 WDHR=2.5
      WDHRT=18.7 WDC=1.7 WDD=16.3 WDP=0.1 WDRT=18.7
      ......................
BCU S BCUID=VHBO001 CTYPE=9393-2 CACHE=512 NVS=8 NOAD=221
PATHS PID=P87 PTYPE=E PBUSYV=17.51 14.02 19.62 2.48 3 18.87 19.62 14.02
      PBYV=17.51 14.02 19.62 2.48 3 18.87 19.62 14.02 PIOV=35.9 21.68 49.15
      11.23 3 49.15 36.86 21.68
PATHS PID=P9F PTYPE=E PBUSYV=17.56 13.88 19.95 2.64 3 18.86 19.95 13.88
      PBYV=17.48 13.88 19.71 2.56 3 18.84 19.71 13.88 PIOV=35.77 21.64 48.89
      11.15 3 48.89 36.8 21.64
PATHS PID=PAE PTYPE=E PBUSYV=17.45 14.01 19.54 2.45 3 18.8 19.54 14.01
      PBYV=17.45 14.01 19.54 2.45 3 18.8 19.54 14.01 PIOV=35.89 21.63 49.15
      11.26 3 49.15 36.9 21.63
PATHS PID=PAF PTYPE=E PBUSYV=17.49 13.94 19.62 2.53 3 18.92 19.62 13.94
      PBYV=17.48 13.9 19.62 2.54 3 18.92 19.62 13.9 PIOV=35.88 21.7 49.04 11.19
      3 49.04 36.91 21.7
PATHS PID=PB0 PTYPE=E PBUSYV=17.47 14.02 19.52 2.45 3 18.88 19.52 14.02
      PBYV=17.47 14.02 19.52 2.45 3 18.87 19.52 14.02 PIOV=35.85 21.66 49 11.19
ACT S V=HBO001 A=0700 T=33903 R=1.04 SDS=6.57 Q=0 P=0.18 D=2.17 C=4.87 SDR=6.57
      SG=SGHBO DS=0.8 RWR=2077 RDHT=0.971 FWHT=1 SQSTG=0.666 PC=100 PFWD=0.02
      PCW=0 CWH=0 ST=N
ACT S V=HBO002 A=0701 T=33903 R=0.83 SDS=30.02 Q=0 P=0.19 D=20.9 C=45.66
      SDR=30.05 SG=SGHBO DS=0.8 RWR=6750 RDHT=0.951 FWHT=0 SQSTG=7.416 PC=100
      PFWD=0 PCW=0 CWH=0 ST=N
ACT S V=HBO003 A=0702 T=33903 R=0.83 SDS=32.71 Q=0 P=0.19 D=25.47 C=46.39
      SDR=32.75 SG=SGHBO DS=0.8 RWR=6748 RDHT=0.973 FWHT=0 SQSTG=7.432 PC=100
      PFWD=0 PCW=0 CWH=0 ST=N
ACT S V=HBO004 A=0703 T=33903 R=0.79 SDS=6.32 Q=0 P=0.18 D=11.29 C=10.26
      SDR=6.33 SG=SGHBO DS=7.6 RWR=12.609 RDHT=0.705 FWHT=0.996 SQSTG=1.841
      PC=100 PFWD=4.02 PCW=0 CWH=0 ST=N
ACT S V=HBO005 A=0704 T=33903 R=11.91 SDS=6.17 Q=1.57 P=0.19 D=5.92 C=6.32
      SDR=6.9 SG=SGHBO DS=2.1 RWR=5.362 RDHT=0.87 FWHT=0.999 SQSTG=0.874 PC=100
      PFWD=10.38 PCW=0 CWH=0 ST=N
......................
```

Sample BCU Map Produced with BCU=AUTO

```
         TIMESTMP SYSID=PRD0                                                00010001
VSPSSD0  BCUD  CTYPE=3990-2,CACH=0,NVS=0,                                   X00020002
   MFR=HTC,SER=03111,                                                       X00020003
   CHPID=(4C,4D,7C)                                                          00020004
         ADDRS AD=0200-0202,LCU=009C,RMF=Y,DEVT=33903                       00020005
VDEVTB7  BCUD  CTYPE=9393-2,CACH=2048,NVS=8,                                X00030006
   MFR=IBM,SER=24325,PLNT=91,                                               X00030007
   CHPID=(50,51,52,53,6E,6F,80,81)                                           00030008
         ADDRS AD=D600-D63F,LCU=01AC,RMF=Y,DEVT=33903,SSID=0061             00030009
         ADDRS AD=D640-D67F,LCU=01B0,RMF=Y,DEVT=33903,SSID=0062             00030010
         ADDRS AD=D680-D6BF,LCU=01B4,RMF=Y,DEVT=33903,SSID=0063             00030011
         ADDRS AD=D6C0-D6FF,LCU=01B8,RMF=Y,DEVT=33903,SSID=0064             00030012
VDEVT29  BCUD  CTYPE=9393-2,CACH=1024,NVS=8,                                X00040013
   MFR=IBM,SER=23774,PLNT=91,                                               X00040014
   CHPID=(50,51,52,53,6E,6F,80,81)                                           00040015
         ADDRS AD=D700-D73F,LCU=01BC,RMF=Y,DEVT=33903,SSID=0071             00040016
         ADDRS AD=D740-D77F,LCU=01C0,RMF=Y,DEVT=33903,SSID=0072             00040017
         ADDRS AD=D780-D7BF,LCU=01C4,RMF=Y,DEVT=33903,SSID=0073             00040018
         ADDRS AD=D7C0-D7FF,LCU=01C8,RMF=Y,DEVT=33903,SSID=0074             00040019
```

Sample PGN Map Produced with PGN=IPS

```
SYSID=SYSA    PGN=1    PRTY=203  DESC='BATCH INIT AND         '
SYSID=SYSA    PGN=2    PRTY=203  DESC='TERM.....              '
SYSID=SYSA    PGN=13   PRTY=0    DESC='TERM.....              '
SYSID=SYSA    PGN=20   PRTY=144  DESC='PROD BATCH PERIOD 1  '
SYSID=SYSA    PGN=22   PRTY=144  DESC='TEST BATCH PERIOD 1  '
SYSID=SYSA    PGN=24   PRTY=160  DESC='TOP PRIORITY BATCH  '
SYSID=SYSA    PGN=26   PRTY=112  DESC='LONG TEST BATCH P1 '
SYSID=SYSA    PGN=30   PRTY=223  DESC='NONE'
SYSID=SYSA    PGN=33   PRTY=223  DESC='NONE'
SYSID=SYSA    PGN=40   PRTY=176  DESC='STARTED TASKS        '
 .....
```

Sample PGN Map Input to CP2KEXTR

```
SYSID=*  PGN=0    PRTY=255  PER=* DESC='System'
SYSID=*  PGN=1    PRTY=10   PER=* DESC='Batch'
SYSID=*  PGN=2    PRTY=110  PER=1   DESC='TSO Trivial'
SYSID=*  PGN=2    PRTY=50   PER=2   DESC='TSO Medium'
SYSID=*  PGN=2    PRTY=5    PER=* DESC='TSO Complex'
SYSID=*  PGN=3    PRTY=35   PER=* DESC='WORK '
SYSID=*  PGN=5    PRTY=70   PER=* DESC='WORK '
 .....
```

Sample PGN Map Input to CP2KEXTR from PMMVS

Note _ input styles cannot be mixed.

```
   IN V(SYSID=GEN  PGN=0    DESC=TSO  PER=GEN TYPE='SYSTEM'     )
   IN V(SYSID=GEN  PGN=1    DESC=TSO  PER=GEN TYPE='Batch'      )
   IN V(SYSID=GEN  PGN=2    DESC=TSO  PER=1   TYPE='TSO trivial' )
   IN V(SYSID=GEN  PGN=2    DESC=TSO  PER=GEN TYPE='TSO complex' )
   IN V(SYSID=GEN  PGN=3    DESC=TSO  PER=GEN TYPE='WORK        )
   IN V(SYSID=GEN  PGN=5    DESC=TSO  PER=GEN TYPE='WORK        )
 .....
Statements which do not begin with "IN V(" will be ignored in this file.
```

Sample T30MAP Data Set

```
SYSID=ASYS    DESC='AP'   JOBN=PAP*
SYSID=ASYS    DESC='BE'   JOBN=PBE*
SYSID=ASYS    DESC='GL'   JOBN=PGL*
SYSID=ASYS    DESC='PC'   JOBN=PPC*
SYSID=ASYS    DESC='PO'   JOBN=PPO*
SYSID=ASYS    DESC='TR'   JOBN=PTR*
SYSID=ASYS    DESC='REST'   JOBN=*
```

Sample PGN Map (Goal Mode)

```
*  SYSID= MUST BEGIN IN COL 1 TO INDICATE  START OF RECORD
*  DESC, THE DESCRIPTION IS THE WORKLOAD NAME úú SERVICE CLASS
*  PGN IS UNUSED IN GOAL MODE (GOAL MODE IS WHEN WORKLOAD IS NOT BLANK
*  PRTY IN GOAL MODE IS 10-S723CIMP RELATIVE IMPORTANCE OF GOAL
*  CLS IS  SERVICE CLASS NAME S723MCNM
* REPORT CLASSES ARE COMMENTED OUT
*
SYSID=X001 PGN=0 PRTY=9    DESC='CICSTEST.FBTEST  '
 PER=*  WORKLOAD=CICSTEST SCLASS=FBTEST
*
SYSID=X001 PGN=0 PRTY=8    DESC='TSOTEST.TSOT1    '
 PER=1  WORKLOAD=TSOTEST  SCLASS=TSOT
SYSID=X001 PGN=0 PRTY=8    DESC='TSOTEST.TSOT2    '
 PER=2  WORKLOAD=TSOTEST  SCLASS=TSOT
SYSID=X001 PGN=0 PRTY=7    DESC='TSOTEST.TSOT3    '
 PER=3  WORKLOAD=TSOTEST  SCLASS=TSOT
SYSID=X001 PGN=0 PRTY=6    DESC='TSOTEST.TSOT4    '
 PER=4  WORKLOAD=TSOTEST  SCLASS=TSOT
SYSID=X001 PGN=0 PRTY=0    DESC='TSOTEST.TSOT*    '
 PER=*  WORKLOAD=TSOTEST  SCLASS=TSOT
*
SYSID=X001 PGN=0 PRTY=0    DESC='STUFF'
 PER=*  WORKLOAD=SYSTEM   SCLASS=SYSTEM
*
SYSID=X001 PGN=0 PRTY=0    DESC='STUFF'
 PER=*  WORKLOAD=SYSTEM   SCLASS=SYSSTC
*
SYSID=X001 PGN=0 PRTY=0    DESC='The Rest'
 PER=*  WORKLOAD=*        SCLASS=any
*
 .....
```

Sample Disk Magic Input file

```
/* DASD MAGIC Command (DMC) file */
/* Created by CP90 Extract Program (06/14/96) */
/*   on 14 June 96   13:02  */
new
model 'CP90EXTR' , description='Created 14 June 96   13:02'
/* CU configuration and systems: */
cu '0004',type='3990-3 ESCON', cachesize=64, nvssize=4,
   channel=(type=ESCON, count=1, datarate=17),
   dasd=(type='3390-2', count=2),
 dasd=(type='3390-2',
  volser=SHARE1,
  unit=(sysid=SYSA,devno=0120)),
 dasd=(type='3390-2',
  volser=SHARE2,
  unit=(sysid=SYSA,devno=0121));
cu '0005',type='3990-3 ESCON', cachesize=256, nvssize=8,
   channel=(type=ESCON, count=4, datarate=17),
   dasd=(type='3390-2', count=20),
 dasd=(type='3390-2',
  volser=HOSPL0,
  unit=(sysid=SYSA,devno=0220)),
 ..........................................................
  /*RMF/CRR data at device level*/
  Cu '0004'
   System 'SYSA'
    Load Unit=0120,
 rmf=(0.004, 0.000, 0.895, 0.412, 13.988),
 crr=(0.0, 0.0, 64.21, 0.80, 0.00, 0.512, 0.00, 0.0, 0.0, 0.0, 0.0, 0.0,
      0.000)
 ..............................
  /*RMF/CRR data at unit level*/
load cu='0004', system='SYSA',
 rmf=(0.005, 0.000, 0.891, 0.410, 13.899),
 crr=(0.0, 0.0, 64.21, 0.80, 0.00, 0.512, 0.00, 0.0, 0.0, 0.0, 0.0, 0.0,
      0.000)
load cu='0005', system='SYSA',
 rmf=(393.452, 1.641, 0.513, 5.499, 2.537),
 crr=(381.6, 381.6, 2.73, 0.63, 1.00, 0.012, 1.00, 0.0, 0.0, 0.0, 0.0, 0.0,
      0.077)
load cu='0006', system='SYSA',
 rmf=(132.471, 0.446, 0.223, 0.733, 3.120),
 crr=(109.9, 109.6, 1.26, 0.94, 0.97, 0.008, 1.00, 0.0, 0.0, 0.0, 0.0, 0.0,
      0.030)
load cu='0008', system='SYSA',
 rmf=(194.313, 0.257, 0.397, 3.069, 3.425),
 crr=(197.6, 191.2, 0.60, 0.70, 0.98, 0.008, 1.00, 0.0, 0.0, 0.0, 0.0, 0.0,
      0.056)
```

SORT JCL

This JCL is required to run RMF Post Processor. The MODS= statement is required only for the post processor.

```
//WICKSS JOB (????,????),MSGLEVEL=1,MSGCLASS=O,NOTIFY=WICKS,            00001000
//           REGION=4000K                                               00002000
//SORT    EXEC PGM=ICEMAN                                               00003000
//SYSOUT  DD   SYSOUT=*                                                 00004000
//SYSPRINT  DD  SYSOUT=*                                                00005000
//SORTMSGS  DD  SYSOUT=*                                                00006000
//SYSLMOD DD   UNIT=SYSDA,SPACE=(3600,(20,20,1))                        00007000
//SYSLIN  DD   UNIT=SYSDA,SPACE=(80,(10,10))                            00008000
//SORTLIB DD   DSNAME=SYS1.SORTLIB,DISP=SHR                             00009000
//SYSUT1  DD   UNIT=(SYSDA,SEP=(SORTLIB,SYSLMOD,SYSLIN)),              X00010000
//             SPACE=(1000,(60,20))                                     00020000
//SORTIN DD DISP=SHR,DSN=SYS1.MANDATA                                   00030001
//SORTOUT DD DISP=OLD,DSN=WICKS.RMF.CTS51                               00040003
//*ORTOUT DD DSN=&&SORTED,DISP=(NEW,PASS),                              00041003
//*       SPACE=(CYL,(100,50)),UNIT=SYSDA                               00050003
//SORTWK01 DD DSN=&&SORT1,SPACE=(CYL,(125,125)),UNIT=SYSDA              00060000
//SORTWK02 DD DSN=&&SORT1,SPACE=(CYL,(125,125)),UNIT=SYSDA              00070000
//* SORT  BY    DATE, TIME, RECTYPE                                     00071000
//SYSIN DD *                                                            00072000
 SORT FIELDS=(11,4,CH,A,                                                00074003
              7,4,BI,A,                                                 00075000
              6,1,BI,A),                                                00076000
      EQUALS,                                                           00076103
      FILSZ=E400000                                                     00076205
  MODS E15=(ERBPPSRT,500),E35=(ERBPPSRT,500)                            00076304
/*                                                                      00080000
```

Errors

There have a been several APARs describing times when RMF performance group reporting has invalid values in some of its fields. The Extract Program now has checking for some of these error conditions. If you receive a message stating that an SMF field is invalid, then you may want to consider installing the appropriate PTFs for APARs:

1. OY54654
2. OW11733
3. Others?? The message also states the number of the interval with the error. If it is only one or two intervals (and it is likely to be so as these errors appear to occur infrequently), it may be simpler to EXCLUDE the offending intervals.

The same problem seems to be occurring in goal mode. The field for the workload paging has been seen to have unreasonably high values. Checks have been inserted to discard these values. Read the messages to be sure that this has not happened. At the time of this writing no APAR has been opened although I did find a PMR which referenced this phenomenon. Hunt using the keyword R723CPIR.

EDFI Field Descriptions

Although the data in the EDFI is a subset of the data in the CP2KSAVE file it is organized the same way. The data is organized in a hierarchical manner. Each record is organized as follows.

1. Columns 1-4 indicate a data section
2. Column 5 will be "S" when a section is specified. (This value will change when CP2000 uses this format for output.)
3. Column 6 will be blank.
4. Columns 7-72 will be of the form KEYWORD=data appropriate to that data section.

For the EDFI, this looks as follows

```
  HEAD  --> CEC -> SYS
                     |->SAMP
                     |
                     |->WORK
                     |    |->WDSN
                     |
                     |->CF
                     |    |->CFS
                     |    |    |->CFE
                     |->DCM
                     |->BCU
                     |    |->PATH
                     |    |->ACT
                     |->BCUT
                          |->PTHT
                     |->BCUO
                          |->PTHO
The significance of each section is:
```

HEAD	Contains the description of the account or ENTerprise. A required section.
CEC	Contains the CECID and the number of System Images in the EDF. A required section.
SYS	The System Image. This section contains the variables describing the system image and the number of subsections. A required section.
SAMP	The sample section. A brief description of the file. One section for each sample.
WORK	Each of these sections describes a workload running on the system image. One section for each workload. Optional.
WDSN	For each workload, this section represents the Data Set data found in the Type 42 data. For DSNs with a low rate (<1), the data is accumulated by VOLSER. This will appear as *VOLSER.
CF	For each Coupling Facility there will be a section. Optional.
CFS	For each Coupling Facility Structure with a Coupling Facility there will be a section. Optional.
CFE	For each System using a Coupling Facility Structure there will be a section. Optional.
DCM	For each DCM channel in a SYS there will be a section. Optional.
BCU	For each DASD BCU (Controller) there will be a section. One section for each BCU. Optional.
PATH	The path section will contain data for those paths connected to this BCU. One section for each Path. Optional even if BCU is selected.

ACT The actuator section will contain the data for the actuators running on this controller. One section for each Actuator. Optional even if BCU is selected. Required if path is present.

BCUT For each TAPE BCU (Controller) there will be a section. One section for each BCU. Optional.

PTHT The path section will contain data for those paths connected to this BCU. One section for each Path. Optional even if BCU is selected.

BCUO For each other BCU (Controller) there will be a section. One section for each BCU. Optional.

PTHO The path section will contain data for those paths connected to this BCU. One section for each Path. Optional even if BCU is selected.

Table Enterprise Data File for Input

Key= or level ID	**Data Identification**	**Type (length)**	**MVS Source**	**SLR Bridge Field?**	Obsolete
HEAD	**Header Section**				
ENT=	Enterprise Name	Char (50)	User input	Y	
SOURCE=	Version which produced this file	Char	CP2KEXTRmm/yy	Y	
CEC	**CEC Section**				
CECID=	CEC Identifier	Char	User	Y	
SUPVR=	Supervisor	Char		Y	
CPUMOD=	CPU Model	Char	SMF70MOD SMF70VER	Y	Y
VC=	Version Code	Char	SMF70VER	N	Y
PR=	Number of Processors including ICFs	Number	SMF70BNP. Of last sample.	N	
SR=	Serial Number of the processor	Number	SMF70SER	N	
PRV=	Number of Processors including ICFs etc	Vector	SMF70BNP. Appears only if PR= changed during run.		
CPV=	Number of CP processors	Vector	Count of CP processors in PHYSICAL partition		
ICFV=	Number of ICF processors	Vector	Count of ICF processors in PHYSICAL partition		
IFLV=	Number of IFL	Vector	Count of IFL processors in PHYSICAL partition		
ZAAPV=	Number of ZAAPs	Vector	Count of IFA *or AAP* processors in PHYSICAL partition		
ZIIPV=	*Number of ZIIPs*	*Vector*	*Count of IIP processors in PHYSICAL partition*		

CPUMODV=	Names of the CPCs	Multiple text strings	SMF70MOD SMF70MDL		
CMIND=	Index of current CPC in CPUMODV	Vector			
HWCMODV=	Name of the hardware CPC	Multiple text strings	SMF70MOD SMF70HWM		
HWIND	Index of current hardware CPC in HWCMODV	Vector			
SYS	**System Image Section**				
SYSID=	SYSID	Char	SMF70SID	Y	
HPTSID=	Parallel Sysplex name	Char	SMF70XNM, only if SMF 74 subtype 4 (CF) records are present.		
NSAMPS=	Number of Samples	number			Y
SCP=	SCP	char	SMF70MVS	Y	
VERSION=	SCP level	char	SMF70RLS	Y	
SRM=	SRM Constant	number	SMF72ADJ	Y	
GL=	Goal or Compat Mode	1/0	1 if 72.3 records are present		
WC=	Wait Complete	1/0	1 if Wait Complete is set for this partition. (SMF70VPF bit SMF70WSA)		
BIT=	64 bit mode indicator	1	SMF70EME. Appears only if in 64 bit mode.		
PAR=	Peak:Avg. Ratio	number			
RMFINT=	same as input parameter	number			
DTSRC=	Source of Data	RMF or CMF	SMF70RV2 bit x'0080' on means CMF		
RMFINTL=	Average length of RMF interval in Minutes	number			
GMTOF=	offset GMT	number (-)hh:mm	SMF70LGO		
~~*SNAM=*~~	~~*System name*~~	~~*Char*~~	~~*SMF70SNM*~~		
CPUWAIT=	Seconds of CPU Wait Time	Vector	SMF70WAT	Y	
DASDIO=	DASD I/O start subchannel rate	number	SMF74SSC	Y	Y
PAGE=	System I/O Rate to Page DSNs	number	SMF75SIO	Y	Y
PAGEDS=	Number of Page Datasets	vector	SMF75PSN		

CS=	CS installed MB stats	number	SMF71TFC + SMF71FIN	Y	Y
ES=	ES installed MB stats	number	SMF71OLE	Y	Y
CSAVAIL=	Amount of available central storage in MB. stats	number	SMF71TFC+SMF71IN	Y	Y
ESAVAIL=	Amount of available expanded storage in MB. stats	number	SMF71ASA	Y	Y
SCPCS=	SCP pages (Nucleus, SQA, LPA, CSA)	number	SMF71FIN + ASR + ALP + AVP	Y	Y
SCPES=	SCP pages (Nucleus, SQA, LPA, CSA)	number	SMF71ASE + LAE + CAE	Y	Y
SWTOT=	vector of swaps per second by swap reason code. (1st Terminal Output Wait. 2nd Terminal Input Wait etc.)	Vector	SMF71TOT		
IOCPDT=	Date & time from IOCP file	Char			
IODFDT=	Date & time from IOQ data	Char	R783TDT & R783TTM		
CECUTILV=	Statistics from samples for CPU% for CEC.	Vector[1]	SMF70WAT SMF70PDT		
PAGEV=	System I/O Rate to Page DSNs	Vector	SMF75SIO		
CSV=	CS installed MB stats	Vector	SMF71TFC+SMF71FIN		
ESV=	ES installed MB stats	Vector	SMF71OLE		

[1] Statistics are, in order, the Average, Minimum, Maximum, Standard deviation and Count. Then presuming the maximum and minimum are different, these numbers are followed by the actual samples. The length then is either 5 or 5+count of samples.

Note that these values are of the intervals! That means that in the case of DASDRSPV (DASD Response Time) it is the average response time of all intervals. Even though some intervals may have done very few or no I/Os they will be weighted the same in the average as intervals with large numbers of I/Os.

Another issue occurs with values like DELAYV (the average wait for a tape mount) or ALLOCV (the average allocation time for a tape drive). Intervals which have no mounts will have zero for these intervals. These zero intervals will be included in the statistics and distort the Average, Minimum and Standard Deviation. The user may wish to calculate these averages using the detail data.

CSAVAILV=	Amount of available central storage in MB. stats	Vector	SMF71TFC+SMF71FIN		
ESAVAILV=	Amount of available expanded storage in MB. stats	Vector	SMF71ASA		
SCPCSV=	SCP pages (Nucleus, SQA, LPA, CSA)	Vector	SMF71FIN + ASR + ALP + AVP		
SCPESV=	SCP pages (Nucleus, SQA, LPA, CSA)	Vector	SMF71ASE + LAE + CAE		
PGTOES=	Total pages to ES	Vector	SMF71PES		
PGFROMES=	Total pages from ES	Vector	SMF71RES		
ESAUX=	Total pages migrated ES to AUX	Vector	SMF71PEA		
AUIC=	Average UIC stats	Vector	SMF71ACA		
MUIC=	Minimum UIC stats	Vector	SMF71LIC		
MXUIC=	Maximum UIC stats	Vector	SMF71HIC		
AMIGR=	Average migration age stats	Vector	SMF71AMA		
MMIGR=	Minimum migration age stats	Vector	SMF71LMA		
AINR=	Average in and ready	Vector	SMF70RTT		
AIN=	Average in	Vector	SMF70ITT		
AOUTR=	Average out and ready	Vector	SMF70OTT		
LOGSWAP=	Logical Swap %	Vector	SMF71OTT - AXD -ESD		
LOGEFF=	Logical Swap effective %	Vector	LOGSWAP - LES - LAX		
BATCHV=	Average Batch users	Vector	SMF70BTT		
STCV=	Average Started Task users	Vector	SMF70STT		
TSOV=	Average TSO users	Vector	SMF70TTT		
ASCHV=	Average ASCH users	Vector	SMF70PTT		

OMVSV=	Average OpenMVS users	Vector	SMF70XTT		
OCPU=	Percent of the time that the in/ready count is greater than # of processors	Vector	SMF70Rnn		
OCPU2=	Percent of the time that the in/ready count is greater than # of processors plus 1	Vector	SMF70Rnn		
LREV=	Logical Ready	Vector	SMF70LTT		
DASDIOV=	Total DASD I/O stats	Vector	SMF74SSC		
DASDRSPV=	Total DASD Response Time	Vector	computed from SMF74SSC + QUE + CNN + DSC + PEN + MEC		
DASDSERV=	Connect + Disconnect DASD Time	Vector	computed from SMF74SSC + CNN + DSC + MEC		
TAPEIOV=	Total TAPE I/O stats	Vector	SMF74SSC		
TAPEALC=	Max Allocated Tape Devices	Vector	SMF74MLC		
IFANF=	IFA Normalization Factior (mult this number time time on IFA divided by 256 for equivalent time on CP)	Vector	R723NFFI		
IIPNF	*IIP Normalization Factor (mult this number times time on IIP dived by 256 for equivalent time on CP)*	*Vector*	*R723NFFS*		
WLA=	Defined Capacity	Vector	SMF70WLA		
LAC=	Rolling 4 Hour Average	Vector	SMF70LAC		
LPARNO=	Index our LPAR in the LPAR array (next field)	Number	SMF70PTN		
LPAR=	Names of the LPARS	Vector	SMF70LPM		

LPWGT=	Vector of weights for the partitions. 3 per partition: min, def & max. A value of -1 for the defined weight means dedicated processors. (Physical partition omitted.)	Vector	SMF70MIS, SMF70BPS, SMF70MAS		Y
LPCAP=	Vector 1 and 0 corresonding to whether that partitions is capped. (Physical partition omitted.)	Vector	SMF70VPF bit SMF70CAP		
LPCS=	Vector of the central storage sizes for the LPARS.	Vector	SMF70CSF		
LPES=	Vector of the expanded storage sizes for the LPARS.	Vector	SMF70ESF		
LPCL=	Vector of the LPAR cluster names. 0 if given LPAR not participating.	Vector	SMF70SPN		
LMSUn=	Defined Capacity	Vector	SMF70MSU		
LPPCTMn=	Seconds of 'CP' CPU time for LPAR n.	Vector	SMF70PDT		
LPPITMn=	Seconds of 'ICF' CPU time for LPAR n.	Vector	SMF70PDT		
LPPZTMn=	Seconds of “zAAP CPU time for LPAR n.	Vector	SMF70PDT		
LPPLTMn=	Seconds of 'IFL' CPU time for LPAR n.	Vector	SMF70PDT		
LPPPTMn=	*Seconds of ‘zIIP” CPU time for LPAR n.*	*Vector*	*SMF70PDT*		
LPPOTMn=	Seconds of other (?) CPU time for LPAR n.	Vector	SMF70PDT		
LPPRCn=	Vector of the average number 'CP' processors for LPAR n.	Vector	sum of SMF70ONT (or whole interval if field is not present.)		

LPPRIn=	Vector of the average number 'ICF' processors for LPAR n.	Vector	sum of SMF70ONT (or whole interval if field is not present.)		
LPPRZn=	Vector of the average number of 'zAAP' processors for LPAR n.	Vector			
LPPRLn=	Vector of the average number 'IFL' processors for LPAR n.	Vector	sum of SMF70ONT (or whole interval if field is not present.)		
LPPRPn=	*Vector of the average number of "zIIP" processors in LPAR n*	*Vector*			
LPPROn=	Vector of the average number other(?) processors for LPAR n.	Vector	sum of SMF70ONT (or whole interval if field is not present.)		
LPWTn=	Vector of the average LPAR weights for LPAR n.	Vector	SMF70ACS		
LPWZn=	Vector of the average ZAAP LPAR weights for LPAR n	Vector	SMF70ACS		
LPWPn=	*Vector of the average ZIIP LPAR weights for LPAR n*	*Vector*	*SMF70ACS*		
LPWLMCn=	Vector of percentage of time that the LPAR was WLM capped.	Vector	SMF70NSW		
LPCAPn=	Vector of whether LPAR is capped. 1=capped. (never capped, field absent)	Vector	SMF70CAP		
LPCAPZn=	Vector of whether zAAPs in LPAR are capped.	Vector	SMF70CAP		
LPCAPPn=	*Vector of whether zIIPs in LPAR are capped.*	*Vector*	*SMF70CAP*		
LPWTNn=	Min weight value	Vector	SMF70MIS		
LPWTTn=	Actual weight value	Vector	SMF70BPS		

LPWTZn=	Actual weight of zAAP	Vector	SMF70BPS		
LPWTXn=	Max weight value	Vector	SMF70MAS		
SAMP	**Sample Data Profile**				
DATE=	Start date	mm/dd/yy	SMF70DAT	Y	
TIME=	Start Time	HH:MM	SMF70IST	Y	
DUR=	Duration	HH:MM	SMF70INT	Y	
SIO=	Sample I/O Rate	number	Sample DASDIO=	Y	Y
SPAGE=	Sample paging I/O	number	Sample PAGE=	Y	Y
SSTOR=	Sample processor storage used	number	(CS+ES) - (CSAVAIL+ESAVAIL)	Y	Y
WORK	**Workloads**				
WDESC=	Description	char	User parm	Y	
WTYPE=	Workload type	char	User Parm		
WIO=	"EXCP" Rates	num	SMF72ITS	Y	Y
WSTOR=	Storage Avg[2]	num MB	SMF72FT1,2		
WCS=	Central Stor"" on page	num MB	SMF72FT1,2	Y	Y
WES=	Expanded Stor"" on page	num MB	SMF72ER1,2	Y	Y
WMPL=	MPL	num	Calculate	Y	Y
WMPLV=	MPL	vector	Calculate		
WRESP=	Avg Response time (secs)	num	SMF72TTM	Y	Y
RESPV=	Avg Response time (secs)	vector	SMF72TTM	Y	
WTRANSRS=	Trans Rate/sec	num	RMF72TTX	Y	Y
TRANV=	Trans Rate/sec	vector(n)"" on page	SMF72TTX	Y	
WPAGE=	Demand Page Rate	num	RMF72PIN	Y	Y
WPRTY=	Workload priority	num	IEAIPSnn	Y	
WHPTSID=	Sysplex ID	char	SMF70XNM		
WIOV=	Start SubChannel count	vector	SMF72IRC & R723CIRC		
WIORESPV=	DASD I/O Response Time	vector	SMF72IRC, ICT, IWT, IDT, IOT & R723Cnnn		

[2] xSTOR will be used for versions prior to RMF421 and xCS and xES otherwise.

WCPUTM=	Workload CPU Time in Seconds(Includes CP & IFA time.)	vector	SMF72CTS + SMF72STS & R723Cnnn		
WPAGEV=	Workload Paging	vector	SMF72PIN + R723CPIN		
WCSV=	Workload Central Storage	vector	SMF72FT1,2		
WESV=	Workload Expanded Storage	vector	SMF72ER1,2		
WEXCPV=	Workload Excp Count	vector	SMF72ITS + R723CIOC		
WDCV=	Workload CPU Delay Vector	vector	R723CCDE / R723CTSA-R723CIDL		
WDIV=	Workload I/O Delay Vector	vector	R723CIOD / R723CTSA-R723CIDL		
WIFAV=	Workload zAAP time in seconds	Vector	R723IFAT		
WIFACPV=	Workload time on CP that could have executed on zAAP in seconds.	Vector	R723IFCT		
WIIPV=	*Workload zIIP time in seconds*	*Vector*	*R723CSUP*		
WIIPCPV=	*Workload time on CP that could have executed on zIIP in seconds.*	*Vector*	*R723CSUC*		
WK30	**Workloads when T30MAP is specified**				
WDESC=	Description	char	User parm	Y	
WTYPE=	Workload type	char	User Parm		
~~WHPTSID=~~	~~Parallel Sysplex name~~	~~Char~~	~~User~~	~~Y~~	
WMPLV=	MPL	vector	Calculate		
RESPV=	Avg Response time (secs)	vector	SMF30TRS	Y	
TRANV=	Trans Rate/sec	vector	SMF72TRS	Y	
WPRTY=	Workload priority	num	User parm	Y	
~~WHPTSID=~~	~~Sysplex ID~~	~~char~~	~~SMF70XNM~~		
WIOV=	Start SubChannel count	vector	SMF30AIS & SMF30EIS		
WIORESPV=	DASD I/O Response Time	vector	SMF30AIC, AID & AIW		
WCPUTM=	Workload CPU Utilization	vector	SMF30CPT, CPS HPT, RCT, IIP, ISB & ICU		

WPAGEV=	Workload Paging	vector	SMF30PGI, PGO, PSI, PSO, VPI, VPO, CPI, HPI, LPI & HPO		
WCSV=	Workload Central Storage	vector	SMF30PRV & SYS		
WESV=	Workload Expanded Storage	vector	SMF70ERS		
WEXCPV=	Workload Excp Count	vector	SMF30TEP		
WIFAV	Workload zAAP time in seconds	Vector	SMF30_TIME_ON_IFA & ENCLAVE & DEP_ENCLAVE		
WIFACPV	Workload time on CP that could have executed on zAAP in seconds	Vector	SMF30_TIME_IFA_ON_CP & ENCLAVE & DEP_ENCLAVE		
WDSN	**Workload Type 42**		Data Set by Workload		
WDSNV=	VOLSER	Char	S42DSVOL		
WDSNN=	Data set name	Char	S42DSN		
WDR=	I/O Rate	num	S42DSION		
WDHR=	Hot spot rate	num	S42DSION		
WDHRT=	Hot spot response time	num	S42SIOR		
WDC=	Connect time	num	S42DSIOC		
WDD=	Disconnect time	num	S42DSIOD		
WDP=	Pend time	num	S42DSIOP		
WDRT=	Reponse Time	num	S42DSIOR		
CF	**Coupling Facility**				
HPTSID=	Parallel Sysplex name	Char	SMF70XNM, only if SMF 74 subtype 4 (CF) records are present.		
CFNAME=	Coupling Facility name	Char	R744FNAM		
CFTYPE=	Coupling Facility Device	Char	User Parm or R744FMOD & R744FVER		
CFSTOR=	Total Storage	num MB 1 dec	R744GTSD		
CFDMPSTOR=	Dump Storage	num MB 1 dec	R744GDSA		

CFSTORAV=	Available Storage	num MB 1 dec	R744GTSF		
CFREQ=	Coupling Facility requests per second	num 2 dec	R744FTOR		
FSCCV=	Subchannel contention	vector	R744FSCC divided by R744FTOR times 100		
CFLINKS=	Number of paths available	num	R744FSCU		
CFUT=	CPU Utilization of the CF	vector	R744PBSY		
EFFCP=	Effective number of CPUs	vector	R744PBSY+R744PWAI		
CFREQV=	Coupling Facility requests per second	vector	R744FTOR		
CFNCPS=	Number of processors in the CF	num	SMF744PN		
FLVL=	Coupling Facility Level	num	R744FLVL		
CFSYSN=	*Names of systems connected to this CF*	*Vector*	*R744XSYS*		
~~*PEER=*~~	~~*Name of peer CF*~~	~~*Char*~~	~~*R744RNAM*~~		
~~*PRSR=*~~	~~*Serial of peer CF*~~	~~*hex*~~	~~*R744RNDE*~~		
~~*PRTYPE=*~~	~~*Type of peer CF*~~	~~*char*~~	~~*R744RNDE*~~		
~~*FPAM*~~	~~*Path available*~~	~~*num*~~	~~*R744FPAM*~~		
FSCL	*Limit to subchannels that could be used*	*num*	*R744FSCL*		
~~*FTAP*~~	~~*CHPID acronyms*~~	~~*Vector*~~	~~*R744FTAP*~~		
CFS	**Coupling Facility Structure**		For each Coupling Facility Structure there will be a complete description		
STRNAME=	CF Structure Name	Char	R744SNAM		
STRTYPE	CF Structure type	Char	R744STYP 1=LOCK, 2=LIST, 3=LIST, 4=CACH		
STRSIZE=	Storage Allocated to this Structure	num MB 1 dec	R744SSIZ		
FLG=	*Duplex primary or secondary*	*num*	*R744QFLG, 0=not duplex, 1= primary, 2= secondary.*		
CFE	**Coupling Facility Structure Entry**		For each Coupling Facility Structure there will be an entry for each connected system.		

SYS=	SYSID of connected system	Char	SMF74SID		
AREQV=	Asynchronous Requests per second	vector	R744SARC		
ASRVV=	Asynchronous Requests average service time in microseconds	vector	R744SATM divided by R744SARC		
SREQV=	Synchronous Requests per second	vector	R744SSRC		
SSRVV=	Synchronous Requests average service time in microseconds	vector	R744SSTM divided by R744SSRC		
SQV=	Number of requests queued per second	vector	R744SQRC		
SQTMV=	Average time queued in microseconds	vector	R744SQTM divided by R744SQRC		
SCNV=	Lock contention	vector	R744SCN divided by interval in seconds		
SFCNV=	False lock contention	vector	R744SFCN divided by interval in seconds		
SSTAV=	requests changed from SYNC to ASYNC	vector	R744SSTA divided by interval in seconds		
SHTOV=	requests waiting on high priority queue	vector	R744SHTO divided by interval in seconds		
SLTOV=	requests waiting on low priority queue	vector	R744SLTO divided by interval in seconds		
STRCV=	total requests to this structure from this system	vector	R744STRC divided by interval in seconds		
DCM	**DCM channel for DASD**		For each DCM channel there will be a complete section.		
PID=	Path ID	char chpid	SMF78CPID	Y	
PTYPE=	Channel Type	Char	SMF73ACR C'P' parallel, C'E' ESCON, C'V' or C'F' for FICON		
PBUSYV=	Path Busy	vector	SMF73BSY	Y	
PBYV=	Path Busy for this partition	vector	SMF73PBY / SMF73PTI		
BCU	**Basic Configurable Unit for DASD**		For each SYSID BCU there will be a complete description		
BCUID=	BCU identification	char	User Parm	Y	

CTYPE=	CU type	char	User Parm or SMF74CU	Y	
CACHE=	Cache Size	num	User Parm or CSCONF in CRR record	Y	
NVS=	Non Volatile Storage size	num	User Parm or CNCONF in CRR record		
NOAD=	number of addresses under this BCU	num	From BCU map (May include offline volumes for which there will be no ACT section.)		
BCUDASDi=	type of DASD	char	User input or SMF74DEV		
BCUDASDNi=	number of this type	num	computed		
BCUIO=	Total I/O Rate	num	SMF74SSC		Y
BCURESP=	Average Response Time	number	computed		Y
BCUSLO=	Service level objective	num	Maximum response time for devices with an I/O rate > 1		
BCUCONN=	Average connect time for BCU	num	SMF74CNN		Y
BCUDISC=	Average disconnect time for BCU	num	SMF74DIS		Y
BCUPEND=	Average disconnect time for PEND	num	SMF74PEN		Y
BCUQUE=	Average IOS queue time	num	SMF74QUE		Y
BCUSKEW=	Maximum device busy to average.	num	computed		
BCUIOV=	Total I/O Rate	Vector	SMF74SSC		
BCURESPV=	Average Response Time	Vector	computed		
BCUCONNV=	Average connect time for BCU	Vector	SMF74CNN		
BCUDISCV=	Average disconnect time for BCU	Vector	SMF74DIS		
BCUPENDV=	Average pend time for BCU	Vector	SMF74PEN		
BCUQUEV=	Average IOS queue time	Vector	SMF74QUE		
BCUR=	Read Write Ratio for entire BCU. (All devices cached or not.)	num	CRR: "Total (Cache) R/W Ratio"		Y

BCUH=	Read Hit Ratio for entire BCU. (All devices cached or not.)	num	CRR: "Total Read H/R"		Y
BCUW=	Fast Write Hit Ratio for entire BCU. (All devices cached or not.)	num	CRR: "Total F/W H/R"		Y
BCUG=	Sequential Stage Ratio for entire BCU. (All devices cached or not.)	num	CRR: "DASD to Cache Transfers - Sequential" divided by "Total I/O Requests"		Y
BCUC=	Percent of I/Os eligible to be cached for entire BCU.	num	CRR: "Total Cachable I/Os" divided by "Total I/O Requests" times 100		Y
BCUF=	Percent Fast Write Destages for entire BCU. (All devices cached or not.)	num	CRR: "Cache to DASD Transfers - Total" divided by "Total I/O" times 100		Y
BCUK=	Percent Count Key Data Writes for entire BCU. (All devices cached or not.)	num	CRR: "Count Key Data Writes" divided by "Total I/O" times 100 (Not supported with CRR 1.4)		Y
BCUD=	Count Key Data Write Hit Ratio for entire BCU. (All devices cached or not.)	num	CRR: "Count Key Data Write Hits" divided by "Count Key Data Writes" (Not supported with CRR 1.4)		Y
BCUB=	Cache Fast Write Bypass	num	CRR: “Cache Fast Write Bypass” divided by “Total I/O” times 100		Y
BCURV=	Read Write Ratio for entire BCU. (All devices cached or not.)	Vector	CRR: "Total (Cache) R/W Ratio"		
BCUHV=	Read Hit Ratio for entire BCU. (All devices cached or not.)	Vector	CRR: "Total Read H/R"		
BCUWV=	Fast Write Hit Ratio for entire BCU. (All devices cached or not.)	Vector	CRR: "Total F/W H/R"		
BCUGV=	Sequential Stage Ratio for entire BCU. (All devices cached or not.)	Vector	CRR: "DASD to Cache Transfers - Sequential" divided by "Total I/O Requests"		
BCUCV=	Percent of I/Os eligible to be cached for entire BCU.	Vector	CRR: "Total Cachable I/Os" divided by "Total I/O Requests" times 100		

BCUFV=	Percent Fast Write Destages for entire BCU. (All devices cached or not.)	Vector	CRR: "Cache to DASD Transfers - Total" divided by "Total I/O" times 100		
BCUKV=	Percent Count Key Data Writes for entire BCU. (All devices cached or not.)	Vector	CRR: "Count Key Data Writes" divided by "Total I/O" times 100.		
BCUDV=	Count Key Data Write Hit Ratio for entire BCU. (All devices cached or not.)	Vector	CRR: "Count Key Data Write Hits" divided by "Count Key Data Writes"		
BCUBV=	Cache Fast Write Bypass	Vector	CRR: “Cache Fast Write Bypass” divided by “Total I/O” times 100		
PATH	**BCU Path Data**				
PID=	Path ID (if P* then DCM and only the PIOV= will be present)	char chpid	SMF78CPID	Y	
PTYPE=	Channel Type	Char	SMF73ACR C'P' parallel, C'E' ESCON, C'V' or C'F' for FICON		
PSW=	Escon Director	Number	From the BCUMAP(IOCP input)		
PLK=	Link on Escon Director	Number	From the BCUMAP(IOCP input)		
OFFL=	Offline	num (0 or 1)	If no rmf data for path, assumed offline	Y	
CL=	FICON Chpid Level	1.0, 1.5 or 1.75	9672 is 1.0, 2064 is 1.5 or 1.75 (SMF73MBC > 40,000 is 1.75), 2066 & later 1.75		
PBUSYV=	Path Busy	vector	SMF73BSY	Y	
PBYV=	Path Busy for this partition	vector	SMF73PBY / SMF73PTI		
PIOV=	Chpid taken	vector	R783PT		
PBBY=	Ficon Bus Busy	vector	SMF73TBC and SMF73MBC		
PTRD=	Ficon Total read MB/Sec	vector	SMF73TRU and SMF73US		
PTWR=	Ficon Total write MB/Sec	vector	SMF73TWU and SMF73US		
PLRD=	Ficon LPAR read MB/Sec	vector	SMF73PRU and SMF73US		
PLWR=	Ficon LPAR write MB/Sec	vector	SMF73PWU and SMF73US		

ACT	**Actuators**		Detail needed for performance analysis	Y	
V=	Volser	char	SMF74SER	Y	
A=	Address	char	SMF74NUM	Y	
T=	DASD Type	char	User Parm or SMF74DEV	Y	
R=	I/O Rate	num	SMF74SSC	Y	
SDS=	Standard deviation for service	num	calc		
Q=	IOSQ	num	calc	Y	
P=	PEND	num	SMF74PEN	Y	
D=	DISC	num	SMF74DIS	Y	
C=	CONN	num	SMF74CNN		
SG=	Storage Group	char	SMF74SGN		
DS=	Total Number of Allocations in Effect for the Device	num	SMF74NDA		
SDR=	Standard deviation for response	num	calc		
RWR=	Read Write Ratio	num	CRR: "Total (Cache) R/W Ratio"		
RDHT=	Read Hit Ratio	num	CRR: "Total Read H/R"		
FWHT=	Fast Write Hit Ratio	num	CRR: "Total F/W H/R"		
SQSTG=	Sequential Stage Ratio	num	CRR: "DASD to Cache Transfers - Sequential" divided by "Total I/O Requests"		
PC=	Percent Cached	num	CRR: "Total Cachable I/Os" divided by "Total I/O Requests" times 100		
PFWD=	Percent Fast Write Destages	num	CRR: "Cache to DASD Transfers - Total" divided by "Total I/O" times 100		
PCW=	Percent Count Key Data Writes	num	CRR: "Count Key Data Writes" divided by "Total I/O" times 100		
CWH=	Count Key Data Write Hit Ratio	num	CRR: "Count Key Data Write Hits" divided by "Count Key Data Writes"		
CFWB	Cache Fast Write Bypass	num	CRR: “Cache Fast Write Bypass” divided by “Total I/O” times 100		

ST=	Actuator Status	char	CRR: "Device Status" N - Caching Activiated, DASD FW Allowed, D - Caching Deactiviated, DASD FW Deactivated, C - Caching Activiated, DASD FW Deactivated, F - Caching Deactivated, DASD FW Allowed.		
BCUT	**Basic Configurable Unit for Tape**				
BCUID=	BCU identification	char	User Parm	Y	
CTYPE=	CU type	char	User Parm or SMF74CU	Y	
CACHE=	Cache Size	num	User Parm or 0	Y	
NVS=	Non Volatile Storage size	num	User Parm or CNCONF in CRR record		
AAD=	Number of active addresses	num	Number of addresses on this bcu with RMF data		
NOAD=	number of addresses under this BCU	num	From BCU map (May include offline volumes for which there will be no ACT section.)		
BCUTAPEi=	type of Tape Device	char	User input or SMF74DEV		
BCUTAPENi=	number of this type	num	computed		
BCUIOV=	I/O Rate this BCU	Vector	SMF74SSC		
ALLOCV=	Average Allocation Time in seconds	Vector	computed from SMF74ALC + SAM		
MOUNTV=	Number of Mounts for this BCU	Vector	computed from SMF74MCT		
DELAYV=	Average Wait for mount in seconds	Vector	computed from SMF74MTP + SAM + MCT		
CONNV=	Total Connect time for this BCU in seconds	Vector	computed from SMF74CNN		
DISCV=	Total Disconnect time for this BCU in seconds	Vector	computed from SMF74DIS		
PTHT	**See PATH above**				
BCUO	**Basic Configurable Unit for Other**				
BCUID=	BCU identification	char	User Parm	Y	
CTYPE=	CU type	char	User Parm or SMF74CU	Y	

CACHE=	Cache Size	num	User Parm or 0	Y	
NVS=	Non Volatile Storage size	num	User Parm or CNCONF in CRR record		
NOAD=	number of addresses under this BCU	num	From BCU map (May include offline volumes for which there will be no ACT section.)		
PTHO	**See PATH above**				