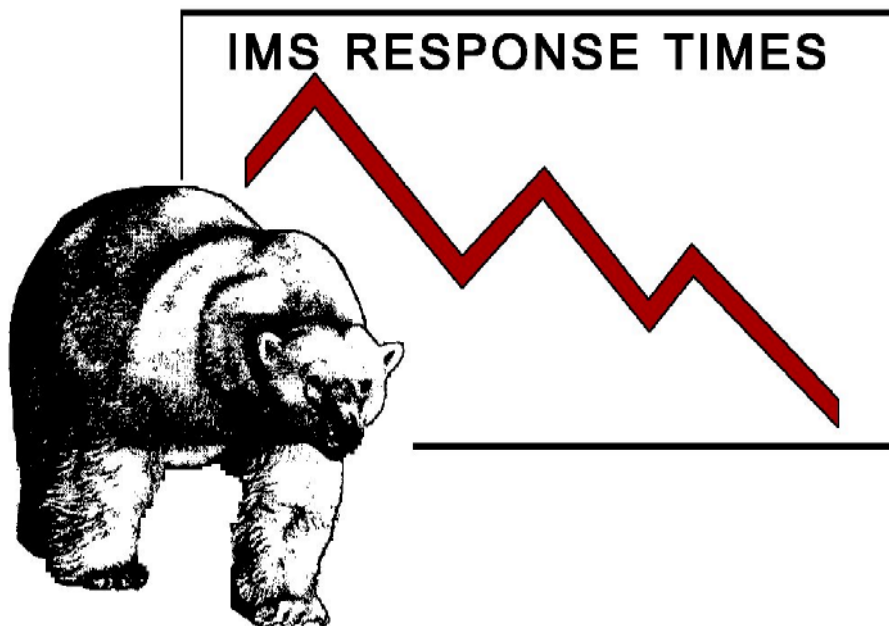


BEARS

IMS

General Usage Manual
VIO Systems Limited



Version: 03.6

Revision: 1st September 2004

Product Number: VIO-002

Manual Ref: BRS-02-013

NOTE: The following trademarks are referred to throughout this manual:

IBM, MVS, MVS/XA, MVS/ESA and IMS/ESA are trademarks of International Business Machines Corporation.

SAS is a trademark of SAS Institute Inc.

SPANEX, SPOUTPUT, SPZPARSE and SPIDER are trademarks of Span Software Consultants Limited.

BEARS / IMS is a trademark of VIO Systems Limited.

CONTENTS**Page**

1.	Introduction	5
1.1.	BEARS / IMS HIGHLIGHTS	5
1.2.	Changes with BEARS / IMS Version 3.6.....	6
1.3.	Changes with BEARS / IMS Version 3.5.....	7
1.4.	Changes with BEARS / IMS Version 3.4.....	7
1.5.	Changes with BEARS / IMS Version 3.3.....	7
1.6.	Changes with BEARS / IMS Version 3.2.....	8
1.7.	Changes with BEARS / IMS Version 3.1.....	8
1.8.	Changes with BEARS / IMS Version 3.0.....	9
2.	Using the BEARS / IMS Software	13
2.1.	Setting up the System	13
2.2.	How does BEARS / IMS work ?	18
2.3.	BEARS / IMS Control Cards.....	20
2.4.	BEARS / IMS optional DD Statements.....	26
2.5.	Table and DD Card Cross Reference.....	27
2.6.	FILE Contents and Structure	29
2.7.	BEARS / IMS SYSOUT Reports	30
2.8.	BEARS / IMS Return Codes	31
2.9.	Enquiring against the data	32
2.10.	Database Summarization Periods	33
3.	BEARS / IMS Common Problems	35
3.1.	Full Tables and Datasets - BEARS jobstep	35
3.2.	Full Datasets - BEARSUMM jobstep.....	35
3.3.	Full Databases - SAS jobsteps.....	36
3.4.	Allocation errors	36
3.5.	Missing fields in Response records	37
3.6.	Missing Mixed Mode records from MIXFILE and TRANFILE.....	37
3.7.	Message Switched Transactions	38
4.	Appendix A - BEARS / IMS SAS Database Contents.....	39
4.1.	SAS Database ABENDS	39
4.2.	SAS Database CHKPFIL	40
4.3.	SAS Database FASTFILE	43
4.4.	SAS Database MIXFILE	44
4.5.	SAS Database RESPFILE	46
4.6.	SAS Database RESAREA	48
4.7.	SAS Database RESSMB.....	49
4.8.	SAS Database RESTERM.....	50
4.9.	SAS Database TRANFILE	51
4.10.	SAS Database VIOFILE.....	53
5.	Appendix B - Detail file DSECT mapping.....	55
6.	Appendix C - Sample SAS Enquiries and Macros	57
6.1.	Configuring SAS macros.....	57
6.2.	Example macros.....	58
6.3.	SAS Allocations and Options	60
6.4.	SAS Performance Options.....	61
6.5.	SAS Sample Enquiries.....	63



List of Figures**Page**

Figure 1 - Example Archive JCL.....	13
Figure 2 - BEARSCNT example JCL	15
Figure 3 - BEARS/IMS Schematic.....	17
Figure 4 - Table and DD card Cross reference	27
Figure 5 - File Contents and Structure.....	29
Figure 6 - Reports Cross Reference.....	30
Figure 7 - Return Code Cross Reference	31
Figure 8 - Database Contents.....	32
Figure 9 - APPLIC member.....	58
Figure 10 - APPLIC2 member.....	59
Figure 11 - PROJDEFS member	59
Figure 12 - SAS allocations.....	60

BEARS / IMS Manuals**Order No**

BEARS / IMS Product Description	BRS-01
BEARS / IMS General Usage Manual	BRS-02
BEARS / IMS Installation and Maintenance Manual	BRS-03
BEARS / IMS Messages and Codes Manual	BRS-04

1. Introduction

BEARS / IMS (or **B**atch **E**valuation **A**nd **R**eporting **S**ystem / IMS), is a comprehensive performance monitoring system for IMS/VS and IMS/ESA.

The need exists for a single, sophisticated yet simple to use monitor which can combine Full Function, Fast Path and Mixed Mode systems. In addition it must keep historical data, allow trend analysis, produce Management reports and yet have NO OVERHEAD on the system it is monitoring. BEARS / IMS is the only product which satisfies all these requirements, and yet executes so fast that it can keep pace with the largest and busiest IMS systems in use anywhere.

BEARS / IMS runs under MVS/XA, MVS/ESA or z/OS and release 03.6 supports all IMS releases from 1.3 to 8.1.

1.1. BEARS / IMS HIGHLIGHTS

BEARS / IMS has the following highlights:

Full Function Response Monitor

BEARS / IMS produces Full response information for all Full Function transactions. Handles complex message switching, MSC and ISC. Statistics are given on response times, CPU utilization, DL/I calls made, Input and Output message lengths, Terminal and Region usage, etc. The user can specify whether a *response* is a reply to the input terminal or to any terminal.

Fast Path Response Monitor

BEARS / IMS produces full statistics on Fast Path response times, Buffer utilization, Input and Output message lengths, call statistics, CI contentions, Buffer waits, etc.

Mixed Mode Response Times

When an IMS transaction accesses Full Function resources, either the Message Queues or a DL/I database, then IMS records it as a *Full Function* transaction. When a transaction accesses Fast Path resources, either EMH, DEDBs or MSDBs, then IMS records it a *Fast Path* transaction. Should a transaction access both kinds of resource, as is often the case, then IMS records it as TWO transactions. BEARS / IMS will successfully tie the two transactions together to give you a true picture of the amount of work your IMS system is doing. Information is recorded in all the above areas. NOTE. For every Mixed Mode response record written there is a corresponding entry in both the Full Function and Fast Path response files.

Combined Response Times

BEARS / IMS will write each of the above response files on an individual basis. It is also possible to create a single file which contains all of the above types of transaction. The records will contain whichever fields are relevant to that transaction type, (i.e. Full Function transactions will have all Fast Path fields set to null values). This gives the user a single place to enquire on any transaction.

IMS Internal Statistics

BEARS / IMS will produce short concise reports each day advising of IMS Pool utilizations, ISAM/OSAM and VSAM I/O rates, Transaction scheduling statistics, Selective dispatching occurrences, Logging activity, etc. There is also a breakdown of the transaction workload into business applications. These easily understood reports enable you to free your experienced staff for more critical work.

Security Violations

When IMS reports a security violation it does so by Node name and not by User ID. BEARS / IMS corrects this and reports by both Node name and User ID.

Transaction Abends

BEARS / IMS produces information on all transaction abends in your system, enabling you to vet the quality of the code that you run, the fixes that you apply and allowing you to spot pseudo abends for deadlock reasons.

Resource Utilization

IMS systems tend to grow. This is often because there is no way to monitor the resources in your system that are no longer used. BEARS / IMS provides a way of monitoring the usage of each individual Terminal, Fast Path Area and SMB (Transaction/PSB pair). You will know how often, or not, any of the resources are used, enabling you to reduce "deadwood". Remember IMS allocates storage for these resources whether they are used or not.

Multiple Message Support

BEARS / IMS is primarily a response time monitor. As such, once a transaction has responded then BEARS / IMS normally has no further interest in any other messages that the transaction produced. When Multiple Message Support is enabled, BEARS / IMS will additionally record information on all messages within the system. These include multi-segment input and output messages, second and subsequent output messages and terminal to terminal messages which do not involve a transaction.

User Exit Support

BEARS / IMS supports a customer supplied user exit to which a copy of each IMS log record is presented. The user may perform any processing required on this copy record. Full diagnostics are produced in the case of any abends in the user exit. A sample exit showing linkage and parameter conventions is provided.

1.2. Changes with BEARS / IMS Version 3.6

The following changes have been made between Version 3.50 and Version 3.60 of BEARS / IMS:

- IMS/ESA Release 8.1 support. BEARS / IMS fully supports this release of IMS.

1.3. Changes with BEARS / IMS Version 3.5

The following changes have been made between Version 3.40 and Version 3.50 of BEARS / IMS:

- IMS/ESA Release 7.1 support. BEARS / IMS fully supports this release of IMS.
- OTMA support. Response time records for all transactions using OTMA as a transport mechanism are now fully supported.

1.4. Changes with BEARS / IMS Version 3.4

The following changes have been made between Version 3.30 and Version 3.40 of BEARS / IMS:

- IMS/ESA Release 6.1 support. BEARS / IMS fully supports this release of IMS.
- A new driver program, BEARSCAT, has been created. If the user writes the SLDS volumes as MVS Cataloged datasets, BEARSCAT may be appropriate. It is a complete replacement for the standard BEARSDRV program. It gathers the information required about the SLDS volumes created from the catalog, thereby saving tape/cassette unit allocations. This may help if the user has heavy contention for these devices.
- In previous version of BEARS / IMS, transactions in flight across OLDS switches were discarded. A new option has been created with this version which will allow users to carry forward any unmatched or partially matched records onto the next volume processed. BEARS / IMS will then attempt to resolve these records.
- IMS 6.1 introduces the concept of Co-ordinated Universal Time (UTC time). This allows IMS systems to remain active during local to GMT time changes. A new option has been created that will allow BEARS / IMS to report either in local or UTC time.
- SPA message lengths have been broken out of Full Function messages as separate fields. They will no longer be counted as part of the message length.
- The Checkpoint processor now reports on actual MFS IO's as well as MFS IO requests.

1.5. Changes with BEARS / IMS Version 3.3

The following changes have been made between Version 3.20 and Version 3.30 of BEARS / IMS:

- IMS/ESA Release 5.1 support. BEARS / IMS fully supports this release of IMS.

- A new DD statement DEBUGFIL is supported. When this card is present in the BEARS / IMS jobstep, BEARS will write a copy of the BRS1501 debugging information records to the file that is being pointed at. The file must have the following DCB information;

RECFM=FB, LRECL=80

- A dynamic allocation error during allocation of the SLDS with the TWOPASS option specified will no longer result in a U0407 abend. BEARS will end with a return code of 08, as it does when TWOPASS is not specified.
- The record matching for Full Function and Mixed Mode transactions has been improved.

1.6. Changes with BEARS / IMS Version 3.2

The following changes have been made between Version 3.10 and Version 3.20 of BEARS / IMS:

- A new parameter card ALLMESSAGES or ALLMSGs has been created. If this card is present in the PARMCARD deck then BEARS / IMS will write information regarding all messages within the system to the MSGFILE and MSGSUMM output files.
- BEARS / IMS now reports whether a transaction was processed via standard IMS schedule or via IMS Quick Reschedule.
- Several new fields are included in the detail and summary response files.

1.7. Changes with BEARS / IMS Version 3.1

The following changes have been made between Version 3.00 and Version 3.10 of BEARS / IMS:

- IMS/ESA Release 4.1 support. BEARS / IMS fully supports this release of IMS.
- A new parameter card TRANRESP or TRRESP has been created. If this card is present in the PARMCARD deck then BEARS / IMS will write transaction responses from all types of transaction, (Full Function, Fast Path and Mixed Mode), to the TRANFILE and TRANSUMM output files.
- A new parameter card RECEXIT or REC has been created. If this card is present in the PARMCARD deck then BEARS / IMS will use the sub-parameter EXITNAME as the name of the user supplied exit it is to pass a copy of all log records to. A sample exit, BRSUEXIT, is supplied to illustrate the interface conventions.

- A new parameter card DEVICE or DEV has been created. If this card is present in the PARMCARD deck then BEARS / IMS will use the sub-parameter UNITTYPE for SLDS allocation. Users may find this feature useful when using different unit types for automated cartridge mounting devices.
- A new parameter card CATALOG or CAT has been created. If this card is present in the PARMCARD deck then BEARS / IMS will allocate the SLDS volume via the catalog using dataset name only. No unit type will be specified.
- All SLDS allocations are now performed with a DISPosition of SHARE rather than OLD. This is to reduce the impact of running BEARS / IMS jobs against DASD SLDS volumes.
- A new field has been added to the Full Function, Mixed Mode and Combined Transaction detail files. The field shows whether the output message was multi-segment. (Note. For obvious reasons this field is dropped in the summary SAS databases).
- A new field has been added to the Full Function, Fast Path, Mixed Mode and Combined Transaction detail files. The field shows the RACF user-id of the user who initiated the transaction if the user had signed on. (Note. For obvious reasons this field is dropped in the summary SAS databases).
- The return code from BEARSSEC on discovering an invalid VSAM return code has changed from 08 to 12. This is due to return code 08 being used elsewhere in BEARS / IMS.
- The report generated for the TWOPASS option or when a table fills has been changed. This report used to show the table requirements for all table types regardless of the options specified. The report will now only show the table requirements for the options that the user specified in PARMCARD.
- The default value for the internal table BMP has been increased from 512k to 1024k. In very exceptional circumstances the value of 512k was insufficient for a SLDS volume.
- The DCB parameters for the POINTERS dataset have changed. The new DCB values are: RECFM = FB, LRECL=68, BLKSIZE=6800

1.8. Changes with BEARS / IMS Version 3.0

The following changes have been made between Version 2.10 and Version 3.00 of BEARS / IMS:

- Allocation of the TODOLIST will now be performed by BEARS, BEARSDRV and BEARSCNT in the following fashion;



The allocation will be attempted as normal.

Should it fail then the program will wait for ten seconds and then retry.

This will happen for a maximum of 6 times before the program will take whatever recovery action is normal.

- A new parameter card NODATE or NODT has been created. If this card is present in the PARMCARD deck then the notification of date change (msg. BRS1900) and the subsequent return code of 04 will not be produced.
- A new program BEARSCNT has been created. This program is run with a parameter of the TODOLIST dataset name (as BEARSDRV). It will issue a message BRS2802 informing the user of the number of entries in the TODOLIST. It will then end with a return code equal to the number of entries. Should the program be unable to allocate the TODOLIST then it will issue an informational message BRS2803 and end with a return code of 00.

If a step using the BEARSCNT program is added to the normal BEARS/IMS job then it should be possible to use JCL condition code checking to halve the number of BEARS/IMS jobs submitted. This could be done by not resubmitting a BEARS/IMS job following a 00 return code from the BEARSCNT step. This should eliminate a BEARS/IMS job finding no entries in the TODOLIST as normally happens with every other BEARS/IMS job.

- The BEARSDRV program will now produce a message BRS0102 showing how many entries are outstanding in the TODOLIST.
- A new parameter card TWOPASS or TWO has been created. If this card is present in the PARMCARD deck then the SLDS will be passed twice and the table sizes required to process it will be dynamically calculated. This allows the user not to specify any table sizes in the PARMCARD deck.

This removes the requirement for all table sizes to be at their individual high water mark and should significantly reduce the region size requirements of BEARS/IMS. It should also facilitate the use of high density cassette drives.

- A new parameter card called SPINOFF or SPIN has been created. The format of this card is as follows;

SPINOFF TYPE=xxyy or
 SPINOFF TYPE=(xxyy,xxyy,.....,xxyy) up to a maximum of ten occurrences.

where xx = recordtype (eg. 01, 03, 59) and yy = subtype.

Should the record involved have no subtype (e.g. 01, 03, etc) or you wish to select all subtypes (e.g. 5901, 5903, 5937, etc) then specify subtype as 00.

If this card is found in the PARMCARD deck then BEARS/IMS will check that a SPINFILE DD card exists in the BEARS/IMS job. If this is the case then all occurrences of the selected record types will be placed in the SPINFILE. It is the users responsibility that the size and DCB information for the SPINFILE are correct. It is recommended that the blocksize and lrecl of the SPINFILE are the same as the OLDS or SLDS datasets and that DCB=BUFNO=20 is specified for this file.

- The BEARS/IMS security monitor will now report the RACF security violation code. Refer to the IBM RACF documentation or the IMS type 10 log record DSECT for an explanation of these codes.
- The maximum number of volumes that a single SLDS can span and be processed by BEARS/IMS has been increased to 20.
- A new parameter card NONRESP or NONR has been created. If this card is present in the PARMCARD deck then any matched sets of input messages with no corresponding output messages will be accepted as valid. All timestamps in the output section of the record will be set to the schedule end time. This allows BEARS/IMS to track transactions which do not produce a response. The records can be identified by the BMFFCLB and BMFFCTB fields containing the literal string "NORESPNS".
- The log data required to produce the DMB resource utilisation information is no longer recorded by IMS. This feature and all associated reports have therefore been withdrawn from this release of BEARS/IMS.

This page intentionally left blank.



2. Using the BEARS / IMS Software

2.1. Setting up the System

There are a few tasks that must be performed before you can run the BEARS / IMS software. You must allocate the datasets required, assemble and link the release dependent BEARS / IMS modules and alter the supplied JCL to your naming standards. All these tasks are covered in the BEARS / IMS Installation and Maintenance Manual. You must also configure the control cards for the functions you wish to run. Select the functions that you want BEARS / IMS to run and set up the member BEARSCRD as required. You should also decide the Application groupings that you wish to use, and set up the member BEARSTRN as necessary. Remember to set the SAS members APPLIC, APPLIC2 and PROJDEFS to the same values. See the Sample SAS Enquiries and Macros section later for further details.

Now add the supplied BEARSDRV step to your archive procedure. Because the IMS Archive utility can sometimes archive more than one SLDS per run, we recommend that you make the stepname of the archive unique. Including the DBRC parameter %ARTIME in the stepname is one way of doing this.

```
//S%ARTIME EXEC PGM=DFSUARCO,.....
```

Figure 1 - Example Archive JCL

The Archive job will now place an entry in the TODOLIST and then optionally submit the normal BEARS / IMS job.

If your SLDS datasets are cataloged, you may wish to consider using BEARSCAT instead of BEARSDRV. This program performs exactly the same functionality with exactly the same parameters, however it gathers the required information regarding the SLDS from the catalog thereby not requiring a tape/cassette unit allocation. This may be of interest to sites with device contention issues.

The BEARS / IMS job will process the SLDS dataset created and place the output into the daily summary files, (i.e. the files whose dataset names end in SUMM). The BEARS / IMS job will then optionally resubmit itself until there are no more entries to be processed in the TODOLIST.

At the beginning of each day, or when the daily summary files are full, the job will end RC 04 from step BEARSUMM and inform you to run the BEARS / IMS daily job (Note. refer to the NODATE parameter in the BEARS / IMS control card section for the exception to this). This may be automated using JCL or a Job Scheduling package.

The BEARS / IMS daily job moves the data collected from the daily summary files to the weekly database. After you have done this you should restart the BEARS / IMS job from the BEARSUMM step if the job was run due to the daily summary files filling.

Once a week you should run the BEARS / IMS weekly job and once a month you should run the BEARS / IMS monthly job. These jobs move the data into the next level of database. The data is summarised to a greater degree at each stage.

There are several points to note about the BEARS / IMS SLDS processing job:

- Step BEARS requires a large region size to build the incore tables necessary. The storage is obtained from above the 16 megabyte line. However there is a system level enqueue held at this point so that you can run as many BEARS / IMS jobs as you like at the same time but the use of storage will be serialised. The exact region size you require for an individual tape can be determined by running the WHATIF process (See the BEARS / IMS control cards section later). If you use the TWOPASS parameter then there should be no table full conditions and the region size required is significantly reduced.
- BEARS / IMS as supplied will reschedule itself to see if there are any more SLDS waiting to be processed. It will continue until the TODOLIST dataset is empty when it will issue message BRS0604 saying that there are no SLDS waiting to be processed and then end with a return code of 04. This is to make automation of restart from the correct step easier for either JCL or a Job Scheduling and Restart package.

Another way of controlling the job execution is to code a BEARSCNT step before the resubmission. BEARSCNT will count the number of outstanding SLDS in the TODOLIST dataset and end with a return code equal to the number of entries. In this case, you would not resubmit the BEARS / IMS job following a return code of 00 from the BEARSCNT step. This will reduce the number of BEARS / IMS jobs scheduled as you will never have one running for an empty TODOLIST, but will mean that jobs that have ended successfully will not have all return codes of zero. The only parameter information that is required for BEARSCNT is the TODOLIST dataset name (as is supplied to BEARSDRV).

An example of using BEARSCNT and JCL condition code checking to schedule BEARS / IMS follows;

```

//BEARS EXEC PGM=BEARS
//      ..
//      ..
//BEARSUMMEC PGM=BEARSUMM,COND=(0,NE)
//      ..
//      ..
//BEARSCNTEC PGM=BEARSCNT,COND=(0,NE)
//      ..
//      ..
//RESUBMITC PGM=IEBGENER,
//      COND=(0,EQ,BEARSCNT),(0,NE,BEARSUMM)
//      ..
//      ..

```

Figure 2 - BEARSCNT example JCL

- BEARS / IMS has been designed not to hold up other jobs requiring the SLDS datasets, e.g. IMS Change Accumulation. Therefore, whenever there is contention for the SLDS dataset, BEARS / IMS will not wait but will terminate with a DARC error. This is not a problem as the next BEARS / IMS job to be run will retry the same SLDS dataset. All BEARS / IMS SLDS allocations are performed with a disposition of share.
- BEARS / IMS will read the SLDS dataset and then do all its processing in storage with no further I/O until it is ready to produce one of the output files. This means that at several points the job will use as much CPU as is freely available. The job profile during this stage is very high CPU utilisation and no IO. This is quite normal and should only last for a short period.
- For every Mixed Mode response record written there is a corresponding entry in both the Full Function and Fast Path response files. This means that should you wish to know the exact count of the number of transactions processed by your system, (without counting Mixed Mode transactions twice), then either use the TRANRESP feature or add the number of transactions processed from the Full Function and Fast Path response files and subtract from this total the number of Mixed Mode transaction responses.

We recommend the use of a Job Scheduling and Restart package, such as Span Software's SPANEX, to schedule the daily, weekly, and monthly jobs. This also makes step restart after an error much easier for operations staff.

An example of this would be as follows:

The normal BEARS / IMS job consists of two steps, BEARS which does the data collection and BEARSUMM which does the first stage summarization. Should either the date change or the Daily files fill then step BEARSUMM will end with

RC 04. The action that should then be taken is to run the Daily summarization job and then step restart the normal BEARS / IMS job from step BEARSUMM. With condition codes in the JCL or a Job Scheduling and Restart system, this can all be automated and require no operator intervention. An example of how to do this with SPANEX is supplied on the distribution tape. It is placed in the sample JCL dataset as member BRSRCMSA.

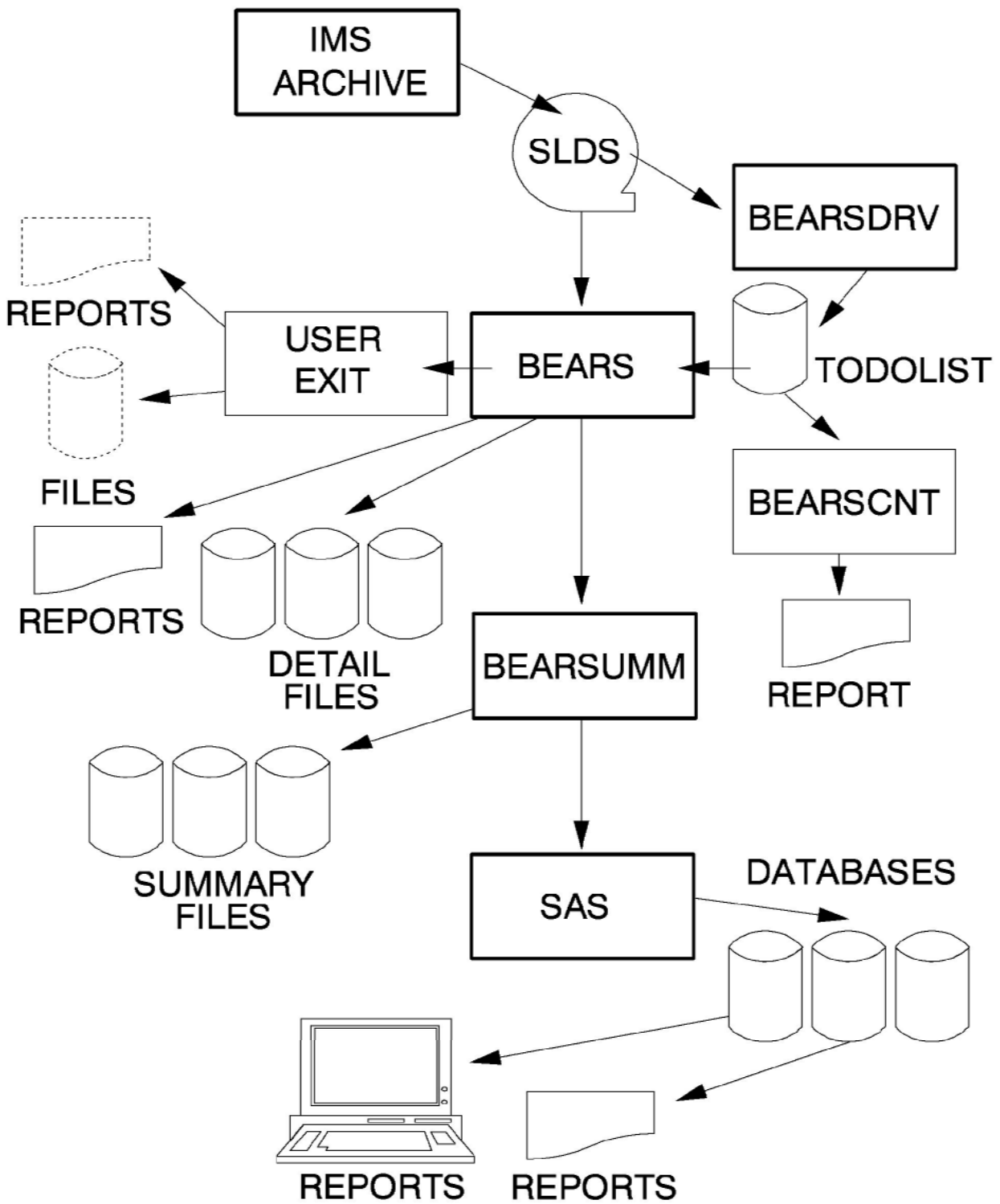


Figure 3 - BEARS/IMS Schematic

2.2. How does BEARS / IMS work?

The previous diagram shows how the BEARS / IMS system functions.

The user inserts a step in the IMS archive procedure to run the BEARSDRV program. This extremely fast step merely notes the Volume Serial number(s), Dataset name(s) and device type that are created by Archive. This information is placed into the TODOLIST dataset. It is important to note that this step does not interfere with IMS archive in any way. Even if it were to fail IMS would mark archive as having completed successfully. The BEARSDRV program will then copy whatever is specified on the DD statement SYSUT1 to the DD statement SYSUT2. If SYSUT1 points at the BEARS / IMS JCL and SYSUT2 points at the internal reader then this will automatically submit a BEARS / IMS job.

If the SLDS datasets are Cataloged, then the user may optionally replace the BEARSDRV program with BEARSCAT. This program performs the same functions as BEARSDRV and uses the same parameters but does not require a tape/cassette unit allocation.

The BEARS / IMS job as supplied contains two steps. The first step runs the actual BEARS / IMS program. This will allocate the first volume from the TODOLIST dataset and process it. The output files from step BEARS / IMS are very heavily detailed and are described in the supplied DSECTS. Refer to Appendix B for more information. Should there be a failure in this step the recovery action is to rerun the job. There is no need for any other action.

The second step in the BEARS / IMS job runs program BEARSUMM. This program summarizes the data from the files created by BEARS / IMS and places the output into the Daily Summary files. Should there be a date change or should one of the summary files fill, then this job will advise the operator to run the Daily SAS job and end RC04. The non-zero condition code is to ease the use of condition code checking in the JCL or with a Job Restart Package in automating the submission of the SAS summarization job. After the Daily SAS job has successfully completed, this job should be restarted from the BEARSUMM step. (Note. refer to the NODATE parameter in the BEARS / IMS control cards section later).

The job will then resubmit itself until there are no more volumes in the TODOLIST to be processed.

Another way of controlling the job execution is to code a BEARSCNT step before the resubmission. The BEARSCNT step will count the number of outstanding SLDS in the TODOLIST dataset and end with a return code equal to the number of entries. In this case, you would not resubmit the BEARS / IMS job following a return code of 00 from the BEARSCNT step. This will reduce the number of BEARS / IMS jobs scheduled as you will never have one running for an empty TODOLIST, but it will mean that jobs that have ended successfully will not have all return codes of zero. The only parameter information that is required for BEARSCNT is the TODOLIST dataset name (as is supplied to BEARSDRV).

The SAS jobs that are shown are the means of moving the data through the different databases. There are jobs supplied to move the data from the Daily summary files to the Weekly databases, from the Weekly databases to the Monthly databases, and finally from the Monthly to the Yearly databases. The data is summarized to a greater degree at each stage. Refer to

Section 2.10 for the default summarization periods of each type of data. Should any of these jobs fail then the recovery action is to rerun the job. The only exception to this is if the failure is caused by the target database being full. In this case the next level SAS job should be run first to create more space and then the failing SAS job be rerun.

2.3. BEARS / IMS Control Cards

The following are the control cards that step BEARS / IMS requires to run and their meanings. They should be placed in the PDS member pointed at by the PARMCARD DD statement.

ABEND or ABN

Run the IMS transaction Abend processor. This will cause any application transaction abends to be written to the ABENDS file and also a report to be generated.

ALLMESSAGES or ALLMSGS

As well as the standard response information, BEARS / IMS is to produce information on all messages in the system. This includes multi-segment input and output messages, second and subsequent output messages, terminal to terminal messages, printer output, unsolicited output and MSC/ISC routing. A record is written to the MSGFILE dataset for each individual input or output message. For conversational transactions, the reported input and output message lengths contain the length of the SPA as well as the message.

ALTDEST or ALT

Accept a message to any destination as the response for a Full Function transaction. If this option is specified then the first output message that a transaction issues to any destination will be accepted as the response to the input message. (Note. this is a system-wide option and should be used where transactions do not respond to the input LTERM.)

CARRYFWD or CARRY

Do not discard records relating to transaction in-flight across an OLDS switch. Write unmatched and partially matched records to the dataset allocated to the CARRYFWD DD statement. BEARS / IMS will attempt to resolve these records using the next volume processed ONLY. Any records which still fail to match will then be discarded.

CATALOG or CAT

Allocate the SLDS volumes via the catalog using the dataset name only. The unit type will not be included in the dynamic allocation.

CHECKPOINT or CHE

Run the IMS internal statistics monitor. This will cause IMS pool sizes, transaction rates, I/O rates, etc, to be written to CHKPFIL. (Note. for this feature to work effectively, the IMS SLDS datasets must be processed in the order that they were created.)

CYCLE n or CYC n

The number of logtapes to check in the history file (CTLFILE) for duplicates. This file contains a record of the last 255 volumes that have been processed. The volume serial number and the first log record sequence number on the volume being processed are checked against the last n entries in the file, where n is the specified number. Should a duplicate be found then the run will be abended with a U0606. This facility is to stop the same volume being processed more than once.

DEBUG or DEB

Produce debugging information. This option causes some additional informational messages to be generated detailing the numbers of records matched and processed. This is not normally of interest, but may be requested during diagnosis of suspected BEARS / IMS problems.

DEVICE unitype or DEV unittype

Allocate the SLDS volume using the specified unit type. This option is useful where sites are using a different unit type to denote the use of automatic cartridge mounting devices.

FASTRESP or FPR

Run the Fast Path Response monitor. This causes Fast Path response records to be written to FASTFILE. A single record is written for each transaction processed.

FULLRESP or FFR

Run the Full Function Response monitor. This causes Full Function response records to be written to RESPFILE. A single record is written for each transaction processed. If the ALLMESSAGES option is not also selected then the statistics reported (e.g. message lengths), refer to the first input and output message segments only. For conversation transactions, the message lengths reported will probably be the SPA.

MIXMODE or MIX

Run the Mixed Mode Response monitor. This causes Mixed Mode response records to be written to MIXFILE. A single record is written for each transaction processed. (Note. if this option is selected, FASTRESP and FULLRESP must also be specified.) If the ALLMESSAGES option is not also selected then the statistics reported (e.g. message lengths), refer to the first input and output message segments only. For conversation transactions, the message lengths reported will probably be the SPA.

NODATE or NODT

Suppress notification of date change (message BRS1900) and the associated return code of 04. With this option specified it becomes the users responsibility to schedule the SAS daily job in normal circumstances.

NONRESP or NONR

Track transactions that do not produce a response. Accept as valid any Full Function input messages that do not have corresponding output messages. All timestamps for the output message will be set to the schedule end time. These records can be identified by the BMFFCLB and BMFFCTB fields being set to the literal "NORESPNS".

ONEVOL or ONEV

It is possible for the IMS archive to create a multi volume dataset, should your OLDS be sufficiently large. By default BEARS / IMS will process all the volumes of this dataset in one pass. Should you wish BEARS / IMS to process the volumes individually then specify this parameter. (Note. do not remove this parameter from the input deck while BEARS / IMS is processing a multi volume dataset, or an Abend U0603 will occur.)

RECEXIT exitname or REC exitname

Run the user supplied exit of the specified name. BEARS / IMS will pass a copy of each log record on the volume being processed to this exit, whether BEARS / IMS has an interest in that record or not. The exit is also notified when the last record has been presented. The user may perform any processing required upon that copy of the record including writing it to an external file. BEARS / IMS processes the real record and not the copy, therefore any changes made to the record will be ignored by BEARS / IMS. Should the user exit fail, BEARS / IMS will produce diagnostic information via the SNPPRINT output file and disable the exit for the duration of this execution. Refer to the supplied sample exit BRSUEXIT for further details of this process.

RESOURCE or RES

Run the IMS Resource monitor. This causes IMS resource records for Fast Path Areas, Terminals and SMBs to be written to RESOURCE.

SECURITY or SEC

Run the IMS Security Violation monitor. This causes any IMS security violations to be written to VIOFILE and a report to be produced. (Note. for this feature to work effectively, the SLDS datasets must be processed in the order that they were created.)

SIZES or SIZ

Produce information on the actual storage used by BEARS / IMS internal tables. This report is useful should you wish to tune the amount of storage required to run BEARS / IMS.

SPINOFF TYPE=xxyy or SPIN TYPE=(xxyy,xxyy,..xxyy)

Where xyyy can be specified to a maximum of ten occurrences. xx is a logtape record type and yy is the subtype. When this option is specified then BEARS / IMS will check for the occurrence of a SPINFILE DD statement in the BEARS / IMS JCL. It will then copy every occurrence of the specified record type and subtype to this file. Should the record type have no subtype (e.g. 01, 03 etc.) or you wish to select all subtypes for a particular record type then specify a subtype of 00. **It is the users responsibility that the size and DCB information of the SPINFILE dataset is correct.** We recommend that the DCB of the SPINFILE is the same as that of the OLDS or SLDS datasets. We also recommend that DCB=BUFNO=20 is specified to reduce the performance degradation that using this option has on the BEARS / IMS job.

SYSOUT x or SOUT x

SYSOUT class for the all printed output reports. Refer to the section on BEARS / IMS SYSOUT Reports for a description of these. (Default is "**".)

TABLE x,y or TAB x,y

Allocate the incore table x to yK. The values of x depend upon the functions selected to be run. See the cross reference below. This storage is used by BEARS / IMS to hold the extracted records from the SLDS. There are several other tables used but the size of these is calculated from the input records.

TODOLIST dsn or **TODO dsn**

If the BEARS / IMS job does not have a LOGTAPE DD card, then allocate the TODOLIST file to the dsn supplied and process the SLDS from there. Should a LOGTAPE DD card be included in the JCL then this parameter will be ignored.

TRANRESP or **TRRESP**

Run the Combined Transaction Response monitor. This causes all transaction type records to be written to TRANFILE. A single record is written for each transaction processed. (Note. This option may be specified regardless of the FASTRESP, FULLRESP and MIXRESP options.) If the ALLMESSAGES option is not also selected then the statistics reported (e.g. message lengths), refer to the first input and output message segments only. For conversation transactions, the message lengths reported will probably be the SPA.

TWOPASS or **TWO**

Read the SLDS twice. The first time dynamically calculate the sizes required for all the tables and then use these sizes in the second pass. With this option specified then no table sizes are required in the PARMCARD deck. This will remove the requirement for all of the tables to be set at their individual high water marks and should significantly reduce the region size required to run BEARS / IMS. However the BEARS / IMS job will be slower as it must pass each SLDS twice.

UTCTIME or **UTC**

For IMS 6.1 systems and above, report all times as Co-ordinated Universal Times (UTC times) and not local times. Local time is the default. This parameter has no effect when processing SLDS created by an IMS system earlier than 6.1.

WHATIF or **WOT**

Produces a report showing the table sizes required to process the supplied SLDS. Also produced is the region size required for the job to process this tape with the recommended table sizes. This parameter should be specified by the user when setting up the system. It is automatically invoked by BEARS / IMS when an incore table size has been exceeded and the job has abended with a message showing which table it was. (Note. if this option is selected then all other options are ignored.)

The TRANCARD parameter deck is a user-defined cross reference of transaction codes to Applications. It defines which transactions are assigned to the various CHKPFIL Application buckets. The format is as follows:

SIZE FFNUM=,FPMNUM=n,FPBNUM=n

Where n is the number of groups you wish to define for Full Function, Fast Path Message driven and Fast Path Non-Message driven Applications respectively. The supplied SAS databases and control cards are set up for 50, 30 and 20 respectively.

TRANS FULLnn trancode

Where nn is one of the above groups and trancode is a list of Full Function transaction codes, separated by commas, of the transactions you wish to assign to this group. Each transaction code can either be an exact match or have a wildcard symbol of *. There is a maximum of 50 transaction entries within a group.

TRANS FASTMnn trancode

Where nn is one of the above groups and trancode is a list of Fast Path Message driven transaction codes, separated by commas, of the transactions you wish to assign to this group. The transaction code can either be an exact match or have a wildcard symbol of *. There is a maximum of 50 transaction entries within a group.

TRANS FASTBnn trancode

Where nn is one of the above groups and trancode is a list of Fast Path Batch transaction codes, separated by commas, of the transactions you wish to assign to this group. The transaction code can either be an exact match or have a wildcard symbol of *. There is a maximum of 50 transaction entries within a group.

Do not forget to place the same groupings of transactions in the SAS APPLIC member. Refer to Appendix C for details on how to configure these macros.

2.4. BEARS / IMS optional DD Statements

The following DD statements are optional for the BEARS job step.

DEBUGFIL

A copy of all the BRS1501 debugging messages will be written to the dataset pointed at by this DD statement. This facility can be used as an aid to testing IMS and BEARS service. The dataset must conform to the following DCB information;

RECFM=FB, LRECL=80

2.5. Table and DD Card Cross Reference

The following shows which tables and DD cards are required for each of the functions:

FUNCTION	TABLES	DD Cards
CARRYFWD	N/A	CARRYFWD
ABEND	07	ABENDS ABENSUMM
ALLMESSAGES	N/A	MSGFILE MSGSUMM
ALTDEST	N/A	N/A
CARRYFWD	N/A	CARRYFWD
CATALOG	N/A	N/A
CHECKPOINT	06, 07, 40, 45, 5937, 5938, 5950, 5955	CHKPFILE CHKPSUMM WORKFILE TRANCARD
CYCLE	N/A	N/A
DEBUG	N/A	N/A
DEBUGFIL	N/A	N/A
DEVICE	N/A	N/A
FASTRESP	5901, 5903, 5936, 5937, 5938, EXC	FASTFILE FASTSUMM
FULLRESP	01, 03, 07, 08, 31, 35, 36, 38, BMP, EXC	RESPFILE RESPSUMM
MIXMODE	N/A	MIXFILE MIXSUMM
NODATE	N/A	N/A
NONRESP	N/A	N/A
ONEVOL	N/A	N/A
RECEXIT	N/A	N/A
RESOURCE	40	RESOURCE RESSUMM
SECURITY	06, 10, 16	SECURITY VIOFILE VIOSUMM
SIZES	N/A	N/A
SPINOFF	N/A	SPINFILE
SYSOUT	N/A	N/A
TABLE	N/A	N/A
TODOLIST	N/A	N/A
TRANRESP	01, 03, 07, 08, 31, 35, 36, 38, BMP, EXC, 5901, 5903, 5936, 5937, 5938	TRANFILE TRANSUMM
TWOPASS	None for any functions	N/A
WHATIF	N/A	N/A

Figure 4 - Table and DD card Cross reference

The EXC table is used to store exception data on unmatched input log records during response time processing. Table BMP is used by BEARS / IMS for internal processing on BMP transactions. We suggest that the EXC table is allocated to 512k and the BMP table to 1024k. The TWOPASS option will set both the EXC and BMP tables to these values.

All functions require DD cards PARMCARD, to point at the input parameters. Except when running the WHATIF function, CTLFILE is also always required to check that you are not reprocessing a duplicate SLDS. If the TODOLIST parameter is not specified then a LOGTAPE DD card must be included. This must point to the valid SLDS that you wish to process. If the WHATIF parameter is coded then all other options are ignored.

2.6. FILE Contents and Structure

FILE	DSORG	USE
ABENDS	VB	Transaction Abend records from BEARS / IMS step
ABENSUMM	VB	Summarised Abend records for Daily file
CARRYFWD	VB	Unmatched and partially matched response records.
CHKPFILE	VB	IMS internal statistics records from BEARS / IMS step
CHKPSUMM	VB	Summarised statistics records for Daily file
CTLFILE	FB	History of SLDS processed
DEBUGFIL	FB	A copy of the BRS1501 debugging records
FASTFILE	VB	Fast Path response records from BEARS / IMS step
FASTSUMM	VB	Summarised Fast Path response records for Daily file
LOGTAPE	VB	IMS SLDS dataset(s). Should you chose not to run from the TODOLIST
MIXFILE	VB	Mixed Mode response records from the BEARS / IMS step
MIXSUMM	VB	Summarised Mixed Mode response records for the Daily file
MSGFILE	VB	Information on second and subsequent output messages and any messages not associated with transactions.
MSGSUMM	VB	Summarised Message records.
PARMCARD	FB	Input parameter deck for the BEARS / IMS step
POINTERS	FB	Internal control dataset
RESOURCE	VB	Resource utilization records from the BEARS / IMS step
RESSUMM	VB	Summarised resource records for the Daily file
RESPFILE	VB	Full Function response records from the BEARS / IMS step
RESPSUMM	VB	Summarised Full Function response records for the Daily file
SECURITY	KSDS	Cross reference of Signed on users and terminals
SPINFILE	VB	User defined
TRANFILE	VB	Combined transaction response records from the BEARS / IMS step
TRANSUMM	VB	Summarised Combined transaction response records for the Daily file
TODOLIST	FB	File of SLDS waiting to be processed
TRANCARD	FB	Transaction code to Application group cross reference
VIOFILE	VB	Security Violation records from BEARS / IMS step
VIOSUMM	VB	Summarised Security Violation records for the Daily file
WORKFILE	VB	Internal work file

Figure 5 - File Contents and Structure

2.7. BEARS / IMS SYSOUT Reports

Should the user request certain functions then BEARS / IMS will produce several SYSOUT reports (assuming that there are records to print). The SYSOUT file(s) will all be dynamically allocated if required with the exception of SYSPRINT which must be specified in the BEARS / IMS JCL. The default SYSOUT class used is "*" unless this has been overridden with the SYSOUT control card in the PARMCARD deck.

A cross reference of these reports follows below;

SYSOUT FILE	OPTION	CONTENTS
ABNPRINT	ABEND	A list of all program abends contained in the SLDS(s) processed. The report includes Date, Time, PSB, Transaction Code, Job name, Step name and Abend Code
EXCPRINT	DEBUG	A list of records that were not matched in the SLDS(s) processed. This file is normally only required by VIO Systems Limited personnel when investigating reported problems
SECPRINT	SECURITY	A list of all security violations that occurred in the period covered by the SLDS(s) processed. The report includes Date, Time, Resource violated, User, Node and RACF error code
SNPPRINT	RECEXIT	Diagnostic information resulting from any abend in the RECEXIT user exit.
SYSPRINT	ALL	Echo of PARMCARD and TRANCARD decks.

Figure 6 - Reports Cross Reference

2.8. BEARS / IMS Return Codes

BEARS / IMS may produce several return codes from either the BEARS or the BEARSUMM steps. Users may wish to set up their JCL to perform condition code checking and conditional execution based upon these return codes.

The optional jobstep BEARSCNT will end with a return code equal to the number of volumes waiting to be processed in the TODOLIST dataset.

A cross reference of the possible codes follows below ;

JOBSTEP	RETURN CODE	REASON
BEARS	04	No Volumes to process in the TODOLIST. Preceded by message BRS0604.
BEARS	08	There was an error encountered dynamically allocating the LOGTAPE dataset. Preceded by message BRS0407.
BEARS	08	There was an error encountered dynamically allocating the TODOLIST dataset. Preceded by message BRS0602.
BEARS	08	There was an error encountered dynamically allocating the LOGTAPE dataset. Preceded by message BRS0605.
BEARS	12	A VSAM error was encountered while processing the SECURITY dataset.
BEARSUMM	04	The date has changed since the last summarisation job was run. Preceded by message BRS1900.
BEARSUMM	04	One of the summary datasets has filled and an Abend X37 has been trapped. Preceded by message BRS1901.
BEARSCNT	nn	The number of logtape volumes waiting to be processed in the TODOLIST dataset.

Figure 7 - Return Code Cross Reference

2.9. Enquiring against the data

The BEARS / IMS data moves through various files at different levels of summarization. There are the very detailed files which come out of the BEARS / IMS step itself. Should you wish to enquire against these files then the DSECTs which map them are identified in Appendix B. Then there are the daily summary files which are produced by the BEARSUMM step. Again, these DSECTs are described in Appendix B. At the end of the day the daily job moves the data to the Weekly database. At the appropriate intervals it is then moved to the Monthly and Yearly databases.

In order not to cause contention with the gathering of data, we suggest that all enquiries are run against the SAS databases and not the detail or summary files.

All the databases contain the same internal files which are mapped in Appendix A, but briefly these are:

ABENDS	The Transaction Abend file
CHKPFILE	The IMS internal statistics file
FASTFILE	The Fast Path response file
MIXFILE	The Mixed Mode response file
MSGFILE	The Message record file (Note. This file is not input into the SAS system)
RESPFILE	The Full Function response file
RESAREA	The Fast Path Area resource utilization file
RESSMB	The Transaction/PSB resource utilization file
RESTERM	The Terminal utilization file
TRANFILE	The Combined transaction response file
VIOFILE	The Security Violation file

Figure 8 - Database Contents

You can use SAS to enquire in both Batch mode and Online mode (under the SAS Display Manager) against all of the data. These enquiries can be in either graphic or non-graphic mode. See the SAS manuals for a list of supported graphic devices.

We have supplied several example SAS enquiries both in graphic and non-graphic form. These are placed by the installation procedure into the SAS source library. There is a member ##INDEX which describes them. A list of them follows in Appendix C, however the BEARS / IMS distribution tape may have been updated to contain more enquiries. The ##INDEX member will contain the definitive list.

2.10. Database Summarization Periods

The following tables list the default summarization intervals used within the files and SAS databases used in the BEARS / IMS system.

Daily files

ABENDS	Not summarized.	One record for each Transaction abend.
CHKPFILE	Not summarized.	One record for each IMS System Checkpoint.
FASTFILE	Not summarized.	One record for each Fast Path sync point.
MIXFILE	Not summarized.	One record for each Mixed Mode sync point.
MSGFILE	Not summarized.	One record for each message not associated with a transaction.
RESPFILE	Not summarized.	One record for each Full Function sync point.
RESOURCE	Not summarized.	One record for each defined resource at every IMS System Checkpoint.
TRANFILE	Not summarized.	One record for each Transaction sync point.
VIOFILE	Not summarized.	One record for each Security Violation.

Weekly SAS Databases

ABENDS	Not summarized
CHKPFILE	Not summarized
FASTFILE	By minute intervals
MIXFILE	By minute intervals
RESPFILE	By minute intervals
RESAREA	By hourly intervals
RESSMB	By hourly intervals
RESTERM	By hourly intervals
TRANFILE	By minute intervals
VIOFILE	Not summarized

Monthly SAS Databases

ABENDS	Not summarized
CHKPFILE	Not summarized
FASTFILE	By 10 minute intervals
MIXFILE	By 10 minute intervals
RESPFILE	By 10 minute intervals
RESAREA	By daily intervals
RESSMB	By daily intervals
RESTERM	By daily intervals
TRANFILE	By 10 minute intervals
VIOFILE	Not summarized

Yearly SAS Databases

ABENDS	Not summarized
CHKPFILE	Not summarized
FASTFILE	By hourly intervals
MIXFILE	By hourly intervals
RESPFILE	By hourly intervals
RESAREA	By daily intervals
RESSMB	By daily intervals
RESTERM	By daily intervals
TRANFILE	By hourly intervals
VIOFILE	Not summarized

3. BEARS / IMS Common Problems

BEARS / IMS has been designed to be as user and operator friendly as possible. Recovery actions, are designed to be easy to understand. If you have an automatic job scheduling and step restart system, then we recommend that you include BEARS / IMS in it to ease your operator load.

The most common problems tend to be in the following areas.

3.1. Full Tables and Datasets - BEARS jobstep

Should the contents of the SLDS vary significantly from your norm, e.g. you had very heavy BMP activity on that day, then it is possible that one of the incore tables used by BEARS / IMS will fill. The program will issue a message and terminate. The recovery action as documented in the Messages and Code manual is as follows:

Read the report that was automatically produced by the WHATIF process.
Alter the offending TABLE entry(s) in the parameter deck to the value produced from the report.
Continue processing as normal.

An alternative to the above solution is to run with the TWOPASS parameter card. This option causes BEARS / IMS to read the SLDS dataset twice. The first time it will gather enough information to build the tablesizes dynamically and the second time it will perform normal processing. Given that the region size specified in the JCL is large enough then this should stop all occurrences of tables filling.

Should one of the detail files fill then the recovery action is as follows:

Delete and redefine the offending dataset increasing its space allocation.
Run the BEARSEOF program against the dataset using a DD statement name beginning with BRSDxxxx.
Continue processing as normal.
For an example of using the BEARSEOF program refer to job BRSINST1 that is placed in the BRSSRCE library by the install process.

3.2. Full Datasets - BEARSUMM jobstep

Should a Summary dataset fill during the BEARSUMM jobstep then the job will automatically back out all changes made in this run from all of the files. The job will then inform the operator to run the Daily summarization job. After this has been completed successfully, the BEARS / IMS job should be restarted from step BEARSUMM which will reapply all the changes. Again this process can be automated if a JCL condition code checking or a job scheduler is used.



3.3. Full Databases - SAS jobsteps

If the workload of your IMS system grows significantly, then obviously the sizes of your daily summary files and weekly, monthly and yearly databases will require alteration. However, should there be large fluctuations in the workload on either a regular or irregular basis, then perhaps you may not wish to allocate any more space to these datasets or databases as the total volume of data is remaining constant.

In this case it is likely that one of the databases will fill. This is not a major problem, BEARS / IMS has been designed to cope with this. If this happens then the job running will fail. The recovery action is to run the next level summarization job for the database which filled and then rerun the failed job.

Example. The Weekly job failed in step BRSW05 due to lack of space.

Solution. Run the Monthly job and then rerun the Weekly job. This entire recovery process can be automated if a JCL condition code checking or a job scheduler is used.

3.4. Allocation errors

BEARS / IMS dynamically allocates most of the datasets that it uses in order to make the running of it as user-friendly as possible. BEARS / IMS was also coded with the express intention of never impacting the IMS system it is monitoring. If there is any conflict for the SLDS dataset then BEARS / IMS will terminate with a documented dynamic allocation (DARC) error. Again, this is not a problem as the next BEARS / IMS run will retry the same SLDS dataset.

A special case is the dynamic allocation of the TODOLIST dataset. Should BEARS / IMS fail in an attempt to dynamically allocate this dataset then it will wait for 10 seconds and then retry the allocation. This will happen either until the allocation is successful or there have been six attempts. If there are six failures then the documented action for the program takes place.

Should you find that BEARS / IMS is consistently failing with dynamic allocation errors then things to check are, is BEARS / IMS being scheduled alongside Change Accumulation on the same machine, etc. If this is true then the solution is to schedule BEARS / IMS after Change Accumulation or on another machine.

BEARS / IMS will only allocate the SLDS for the period that is required and not until the end of the job. Therefore the length of time for which the dataset is held is very small.

Some common dynamic allocation errors are

- DARC 0210 - The dataset is allocated to another job or user.
- DARC 0484 - Request denied by operator.

If for some reason the SLDS that BEARS / IMS wishes to process is unavailable and will be so for a long time, then you may wish to stop BEARS / IMS trying to use it and to move onto the next volume to be processed. This is simply done by editing the TODOLIST dataset and deleting the first entry, which is the tape BEARS / IMS wishes to process next. When the SLDS does become available then it should be processed by adding a LOGTAPE DD card to the BEARS / IMS job pointing at the SLDS volume. The job can then be submitted and the DD card removed.

3.5. Missing fields in Response records

BEARS / IMS may produce response records with certain fields that have missing values. The most common cause of this is Abending transactions going through Suspend Queue processing or pseudo abending with a U0777 abend and being rescheduled.

The following is an example of a situation that can cause this:

- A user enters a transaction from a terminal, generating the appropriate input message log records.
- IMS loads the program into a region and primes it with the input message, producing various other log records.
- The transaction abends with a U3303.
- IMS will attempt the previous two steps 10 times before stopping the transaction.

What BEARS / IMS may see from the log is one set of message data (if the transaction does run to completion), and up to 10 program start and ends. It can only report this by leaving the message related fields blank in the aborted execution attempts. This is done so that you are aware that a program schedule took place and you can measure any CPU time used etc.

The user can determine if this is the case by checking the completion code field in the record for a U3303 or U0777 abend codes.

If your system runs transactions that do not issue any response at all, you may wish to specify the NONRESP parameter. This will cause BEARS / IMS to treat as valid any input message data that does not have corresponding output message data. All of the timestamp fields for the output message will be set to the timestamp of the program schedule end. The user can identify these records by checking the BMFFCLB and BMFFCTB fields for the value of "NORESPNS".

3.6. Missing Mixed Mode records from MIXFILE and TRANFILE

Due to the sequence of IMS log records generated, BEARS / IMS is unable to produce a single Mixed Mode transaction record for any Mixed Mode transaction driven BMP or WFI MPP. BEARS / IMS will report both the Full Function and Fast Path components separately.



3.7. Message Switched Transactions

It is important to understand that when a transaction is invoked via a message switch that the output queue time reported for the first transaction and the input queue time of the second transaction overlap. The response time shown for each individual transaction is accurate when taken in isolation. However, adding the total response times of the two transactions together will not give an accurate figure for the total elapsed time as the queue time for the output message from the first transaction and the input message to the second transaction are effectively counted twice. High output queue times can therefore be a result of other factors rather than just network time (e.g. scheduling problems).

4. Appendix A - BEARS / IMS SAS Database Contents

4.1. SAS Database ABENDS

This Database contains the transaction abend records

SAS Variable	Contents
BABNDPSB	PSB name of transaction.
BABNDTRN	Transaction name.
BABNDJOB	Job name.
BABNDSTP	Step name.
BABNDCDS	System Abend code.
BABNDCDU	User Abend code.
BABNDDAT	Date (Julian).
BABNDTIM	Time (SAS format).
BABNDDT2	Date (SAS format).

4.2. SAS Database CHKPFIL

This Database contains the IMS internal statistics records

SAS Variable	Contents
B0DBMP	# DL/I BMP trans.
B0DMPP	# DL/I MPP trans.
B0TRANT	Total # Full Func trans.
B0PSTHI	Highest PST # allocated.
B0IMSNAM	IMS System ID.
B0STDAT2	Start date (SAS format).
B0TIMEZ	Time interval (secs).
B0STDATE	Start date (Julian).
B0STTIME	Start time (SAS format).
B0ENDATE	End date (Julian).
B0ENTIME	End time (SAS format).
B0MQIO	Total # MSG Q I/Os.
B0MFPIO	Total # MSG Format I/O requests.
B0MFPIOA	Total # MSG Format I/O actions.
B0IOTOT	Total # ISAM/OSAM I/Os.
B0LDMBM	Max DMB utilization.
B0LDMBS	DMB Pool size.
B0LDMWM	Max DMB Work utilization.
B0LDMWS	DMB Work Pool size.
B0LIOPM	Max CIOP utilization.
B0LIOPS	CIOP Pool size.
B0LPSBM	Max PSB utilization.
B0LPSBS	PSB Pool size.
B0LPSSM	Max PSB/SAS utilization.
B0LPSSS	PSB/SAS Pool size.
B0LPSSW	Max PSB Work utilization.
B0LPSSWS	PSB Work Pool size.
B0FALTOT	# SMBs failed scheduling.
B0LICONF	# SMBs failed intent.
B0LOTHR	# SMBs failed other.
B0LPCONF	# SMBs failed program.
B0LPCUT	# SMBs failed priority.
B0SCHTOT	# SMBs attempted scheduling.
B0SCHPER	Percentage schedule failure.
B0LLOGAW	# AWE submitted on writes.
B0LLOGB1	# Logical logger wait for buffers during checkpoint.
B0LLOGB2	# Logical logger wait for buffers not during checkpoint.

SAS Variable	Contents
B0LLOGCW	# Check write requests.
B0LLOGOR	# OLDS reads initiated.
B0LLOGOW	# OLDS writes initiated.
B0LLOGSE	# Logical logger recs.
B0LLOGWE	# WADS EXCPVRs.
B0LLOGWW	# Wait writes.
B0CI01T	Total VSAM 1K I/Os.
B0CI02T	Total VSAM 2K I/Os.
B0CI04T	Total VSAM 4K I/Os.
B0CI08T	Total VSAM 8K I/Os.
B0CI12T	Total VSAM 12K I/Os.
B0CI16T	Total VSAM 16K I/Os.
B0LPIINC	PI increment size.
B0LPIMAX	Max PI utilization.
B0LPISIZ	PI Pool size.
B0LCWAM	Max CWAP utilization.
B0LCWAS	CWAP Pool size.
B0LNPSAP	# Selective Dispatching due to dynamic SAPs.
B0LCWAT	# Selective Dispatching due to CWAP.
B0LDSPT	Total # Dynamic SAPs.
B0LIOPT	# Selective Dispatching due to CIOP.
B0LSAPT	Total # SAPs.
B0LTSAP	Highest Dynamic SAP allocated.
B0FPRIOR	# Fast Path Read I/Os.
B0FBA	# Fast Path Batch trans.
B0FBMP	# Fast Path BMP trans.
B0FMPP	# Fast Path MPP trans.
B0FMD	# Fast Path MSG driven trans.
B0FNMD	# Fast Path non-MSG driven trans.
B0FTOT	Total # Fast Path trans.
B0FUT	# Fast Path Utilities.
B0OBA	# Fast Path MPP OBA users.
B0FPIOT	Total # Fast Path I/Os.
B0FPRIOW	Total # Fast Path write I/Os.
FULL00	Full Function group 1
FULL01	Full Function group 2
FULL02	etc
FULL49	
FASTM00	Fast Path Msg Driven group 1
FASTM01	Fast Path Msg Driven group 2
FASTM02	etc

SAS Variable	Contents
FASTM29	
FASTB00	Fast Path Batch group 1
FASTB01	Fast Path Batch group 2
FASTB02	etc
FASTB19	
B0TUNKDL	# unknown DL/I transactions.
B0TUNKFP	# unknown Fast Path transactions.

4.3. SAS Database FASTFILE

This database contains the Fast Path response records

SAS Variable	Contents
BMFPTRN	Transaction name.
BMFPSDAT	Date (SAS format).
BMFPTIM	Time (SAS format).
BMFPDAT	Date (Julian).
BMFPDCL	# DEDB calls.
BMFPRDS	# DEDB read I/Os.
BMFPMCL	# MSDB calls.
BMFPQTM1	BALG Q time Average. (1.024 ms)
BMFPPTM1	Processing time Average. (1.024 ms)
BMFPOTM1	Output time Average. (1.024 ms)
BMFPPTM1	Terminal time Average. (1.024 ms)
BMFPTOT1	Total Average Response time. (sec)
BMFPQTM2	BALG Q time Max. (1.024 ms)
BMFPPTM2	Processing time Max. (1.024 ms)
BMFPOTM2	Output time Max. (1.024 ms)
BMFPPTM2	Terminal time Max. (1.024 ms)
BMFPTOT2	Total Max Response time. (sec)
BMFPQTM3	BALG Q time Min. (1.024 ms)
BMFPPTM3	Processing time Min. (1.024 ms)
BMFPOTM3	Output time Min. (1.024 ms)
BMFPPTM3	Terminal time Min. (1.024 ms)
BMFPTOT3	Total Min Response time. (sec)
BMFPOBA1	Ave OBA buffers. (+ ve if used)
BMFPNBA1	Ave # NBA buffers overallocated
BMFPOBA2	Max OBA buffers. (+ ve if used)
BMFPNBA2	Max # NBA buffers overallocated
BMFPCIC1	Ave CI contentions.
BMFPCIC2	Max CI contentions.
BMFPWAT1	Ave waits for DEDB buffers.
BMFPWAT2	Max waits for DEDB buffers.
BMFPFREQ	# schedules covered by this rec.
BMFPNODE	Node name *
BMFPLTRM	Lterm name *
BMFPUSER	RACF userid *
BMFPILN	Input Message length *
BMFPOLN	Output Message length *
BMFPMID	MFS MID name *
BMFPMOD	MFS MOD name *

(Note. fields marked * are not passed to the summary databases.)

4.4. SAS Database MIXFILE

This database contains the Mixed Mode response records

SAS Variable	Contents
BMIXTXID	Transaction name.
BMIXSDAT	Start Date (SAS format).
BMIXTIM	Time (SAS format).
BMIXDAT	Date (Julian).
BMIX1MCT	# messages processed.
BMIX1GU1	# DL/I GU calls.
BMIX1GN	# DL/I GN calls.
BMIX1GNP	# DL/I GNP calls.
BMIX1GHU	# DL/I GHU calls.
BMIX1GHN	# DL/I GHN calls.
BMIX1HNP	# DL/I GHNP calls.
BMIX1SRT	# DL/I ISRT calls.
BMIX1LET	# DL/I DLET calls.
BMIX1EPL	# DL/I REPL calls.
BMIX1GUM	# DL/I GU MSG calls.
BMIX1GNM	# DL/I GN MSG calls.
BMIX1ISM	# DL/I ISRT MSG calls.
BMIX1IPL	Input Msg length.
BMIX1OPL	Output Msg length.
BMIX1PSB	PSB name.
BMIX1CLS	Class.
BMIX1IQ1	Input Q time Average (secs).
BMIX1EX1	Execution time Average (secs).
BMIX1OQ1	Output Q time Average (secs).
BMIX1CP1	CPU time Average (secs).
BMIX1IQ2	Input Q time Max (secs).
BMIX1EX2	Execution time Max (secs).
BMIX1OQ2	Output Q time Max (secs).
BMIX1CP2	CPU time Max (secs).
BMIX1IQ3	Input Q time Min (secs).
BMIX1EX3	Execution time Min (secs).
BMIX1OQ3	Output Q time Min (secs).
BMIX1CP3	CPU time Min (secs).
BMIX2DCL	# DEDB calls.
BMIX2RDS	# DEDB read I/Os.
BMIX2MCL	# MSDB calls.
BMIX2QT1	BALG Q time Average. (1.024 ms)
BMIX2PT1	Processing time Average. (1.024 ms)

SAS Variable	Contents
BMIX2OT1	Output time Average. (1.024 ms)
BMIX2TT1	Terminal time Average. (1.024 ms)
BMIX2TO1	Total Average Response time. (sec)
BMIX2QT2	BALG Q time Max. (1.024 ms)
BMIX2PT2	Processing time Max. (1.024 ms)
BMIX2OT2	Output time Max. (1.024 ms)
BMIX2TT2	Terminal time Max. (1.024 ms)
BMIX2TO2	Total Max Response time. (sec)
BMIX2QT3	BALG Q time Min. (1.024 ms)
BMIX2PT3	Processing time Min. (1.024 ms)
BMIX2OT3	Output time Min. (1.024 ms)
BMIX2TT3	Terminal time Min. (1.024 ms)
BMIX2TO3	Total Min Response time. (sec)
BMIX2OB1	Ave OBA buffers. (+ ve if used)
BMIX2NB1	Ave # NBA buffers overallocated
BMIX2OB2	Max OBA buffers. (+ ve if used)
BMIX2NB2	Max # NBA buffers overallocated
BMIX2CI1	Ave CI contentions.
BMIX2CI2	Max CI contentions.
BMIX2WA1	Ave waits for DEDB buffers.
BMIX2WA2	Max waits for DEDB buffers.
BMIXFREQ	# schedules covered by this rec.
BMIXNODE	Node name *
BMIXLTRM	Lterm name *
BMIXMSEGI	Was the input message multi-segment Y/N *
BMIXMSEGO	Was the output message multi-segment Y/N *
BMIXQSHD	Quick Reschedule flag Y/N *
BMIXNSGI	No of input message segments *
BMIXNSGO	No of output message segments *
BMIXUSER	RACF userid *
BMIXMFSI	Full Function MFS MID name *
BMIXMFSO	Full Function MFS MOD name *
BMIXMID	Fast Path MFS MID name *
BMIXMOD	Fast Path MFS MOD name *
BMIXTYP	Program Type *
BMIXJOB	Job name *
BMIXISPA	Input SPA message length *
BMIXOSPA	Output SPA message length *

(Note. fields marked * are not passed to the summary databases)

4.5. SAS Database RESPFILE

This database contains the Full Function response records

SAS Variable	Contents
BMFFTXID	Transaction name.
BMFFSDAT	Start Date (SAS format).
BMFFTIM	Time (SAS format).
BMFFDAT	Date (Julian).
BMFFMCT	# messages processed.
BMFFGU1	# DL/I GU calls.
BMFFGN	# DL/I GN calls.
BMFFGNP	# DL/I GNP calls.
BMFFGHU	# DL/I GHU calls.
BMFFGHN	# DL/I GHN calls.
BMFFGHNP	# DL/I GHNP calls.
BMFFISRT	# DL/I ISRT calls.
BMFFDLET	# DL/I DLET calls.
BMFFREPL	# DL/I REPL calls.
BMFFGUM	# DL/I GU MSG calls.
BMFFGNM	# DL/I GN MSG calls.
BMFFISM	# DL/I ISRT MSG calls.
BMFFIPL	Input Msg length.
BMFFOPL	Output Msg length.
BMFFPSB	PSB name.
BMFFCLAS	Class.
BMFFIQT1	Input Q time Average (secs).
BMFFEXT1	Execution time Average (secs).
BMFFOQT1	Output Q time Average (secs).
BMFFCPU1	CPU time Average (secs).
BMFFIQT2	Input Q time Max (secs).
BMFFEXT2	Execution time Max (secs).
BMFFOQT2	Output Q time Max (secs).
BMFFCPU2	CPU time Max (secs).
BMFFIQT3	Input Q time Min (secs).
BMFFEXT3	Execution time Min (secs).
BMFFOQT3	Output Q time Min (secs).
BMFFCPU3	CPU time Min (secs).
BMFFFREQ	# schedules covered by this rec.
BMFFNODE	Node name *
BMFFLTRM	Lterm name *
BMFFPTYP	Program type *

SAS Variable	Contents
BMFFMSW	# message switches *
BMFFMSEGI	Was the input message multi-segment Y/N *
BMFFMSEGO	Was the output message multi-segment Y/N *
BMFFQSHD	Quick Reschedule flag *
BMFFMSGI	# of input message segments *
BMFFNSGO	# of output message segments *
BMFFUSER	RACF User-id of the transaction submitter (if signed on). *
BMFFMFSI	Input MFS MID name *
BMFFMFSD	Output MFS MOD name *
BMFFJOB	Job name *
BMFFISPA	Input SPA message length *
BMFFOSPA	Output SPA message length *

(Note. fields marked * are not passed to the summary databases.)

4.6. SAS Database RESAREA

This database contains the Fast Path Area resource records

SAS Variable	Contents
BAREDMCB	DMCB name.
BAREAREA	Area name.
BAREDATE	Date (Julian).
BARETIME	Time (SAS format).
BAREDAT2	Date (SAS format).
BAREOPNO	Open Flag count.
BAREFREQ	# Observations covered by this record.

4.7. SAS Database RESSMB

This database contains the SMB (Transaction/PSB pair) records

SAS Variable	Contents
BSMBTRAN	Transaction code.
BSMBPSB	PSB name.
BSMBMENQ	# messages Enqueued since IMS started.
BSMBMDEQ	# messages Dequeued since IMS started.
BSMBDATE	Date (Julian).
BSMBTIME	Time (SAS format).
BSMBDAT2	Date (SAS format).
BSMBFREQ	# Observations covered by this record.

4.8. SAS Database RESTERM

This database contains the Terminal resource records

SAS Variable	Contents
BTRMNODE	Node name.
BTRMLINE	Line no.
BTRMIMSG	Input message count since IMS started.
BTRMOMSG	Output message count since IMS started.
BTRMDATE	Date (Julian).
BTRMTIME	Time (SAS format).
BTRMDAT2	Date (SAS format).
BTRMFREQ	# Observations covered by this record.

4.9. SAS Database TRANFILE

This database contains the Combined transaction response records

SAS Variable	Contents
BMTRTXID	Transaction name.
BMTRSDAT	Start Date (SAS format).
BMTRTIM	Time (SAS format).
BMTRDAT	Date (Julian).
BMTR1MCT	# messages processed.
BMTR1GU1	# DL/I GU calls.
BMTR1GN	# DL/I GN calls.
BMTR1GNP	# DL/I GNP calls.
BMTR1GHU	# DL/I GHU calls.
BMTR1GHN	# DL/I GHN calls.
BMTR1HNP	# DL/I GHNP calls.
BMTR1SRT	# DL/I ISRT calls.
BMTR1LET	# DL/I DLET calls.
BMTR1EPL	# DL/I REPL calls.
BMTR1GUM	# DL/I GU MSG calls.
BMTR1GNM	# DL/I GN MSG calls.
BMTR1ISM	# DL/I ISRT MSG calls.
BMTR1IPL	Input Msg length.
BMTR1OPL	Output Msg length.
BMTR1PSB	PSB name.
BMTR1CLS	Class.
BMTR1IQ1	Input Q time Average (secs).
BMTR1EX1	Execution time Average (secs).
BMTR1OQ1	Output Q time Average (secs).
BMTR1CP1	CPU time Average (secs).
BMTR1IQ2	Input Q time Max (secs).
BMTR1EX2	Execution time Max (secs).
BMTR1OQ2	Output Q time Max (secs).
BMTR1CP2	CPU time Max (secs).
BMTR1IQ3	Input Q time Min (secs).
BMTR1EX3	Execution time Min (secs).
BMTR1OQ3	Output Q time Min (secs).
BMTR1CP3	CPU time Min (secs).
BMTR2DCL	# DEDB calls.
BMTR2RDS	# DEDB read I/Os.
BMTR2MCL	# MSDB calls.
BMTR2QT1	BALG Q time Average. (1.024 ms)
BMTR2PT1	Processing time Average. (1.024 ms)

SAS Variable	Contents
BMTR2OT1	Output time Average. (1.024 ms)
BMTR2TT1	Terminal time Average. (1.024 ms)
BMTR2TO1	Total Average Response time. (sec)
BMTR2QT2	BALG Q time Max. (1.024 ms)
BMTR2PT2	Processing time Max. (1.024 ms)
BMTR2OT2	Output time Max. (1.024 ms)
BMTR2TT2	Terminal time Max. (1.024 ms)
BMTR2TO2	Total Max Response time. (sec)
BMTR2QT3	BALG Q time Min. (1.024 ms)
BMTR2PT3	Processing time Min. (1.024 ms)
BMTR2OT3	Output time Min. (1.024 ms)
BMTR2TT3	Terminal time Min. (1.024 ms)
BMTR2TO3	Total Min Response time. (sec)
BMTR2OB1	Ave OBA buffers. (+ ve if used)
BMTR2NB1	Ave # NBA buffers overallocated
BMTR2OB2	Max OBA buffers. (+ ve if used)
BMTR2NB2	Max # NBA buffers overallocated
BMTR2CI1	Ave CI contentions.
BMTR2CI2	Max CI contentions.
BMTR2WA1	Ave waits for DEDB buffers.
BMTR2WA2	Max waits for DEDB buffers.
BMTRFREQ	# schedules covered by this rec.
BMTRNODE	Node name *
BMTRLTRM	Lterm name *
BMTR1MSI	Multi segment input Y/N *
BMTR1MSO	Multi segment output Y/N *
BMTR1QSC	Quick Reschedule flag *
BMTR1NSI	# of input message segments *
BMTR1NSO	# of output message segments *
BMTR1USR	RACF User-id of the transaction submitter (if signed on). *
BMTR1MFI	Full Function MFS MID name *
BMTR1MFO	Full Function MFS MOD name *
BMTR2MID	Fast Path MFS MID name *
BMTR2MOD	Fast Path MFS MOD name *
BMTR1TYP	Program type *
BMTR1JOB	Job name *
BMTRISPA	Input SPA message length *
BMTROSPA	Output SPA message length *

(Note. fields marked * are not passed to the summary databases).

4.10. SAS Database VIOFILE

This database contains the Security Violation records

SAS Variable	Contents
BVIOSYS	System.
BVIONODE	Node name.
BVIODATE	Date (Julian).
BVIOTIME	Time (SAS format).
BVIODAT2	Date (SAS format).
BVIORES	Resource.
BVIOUSER	Userid.
BVIOERR	RACF error code.



This page intentionally left blank.



5. Appendix B - Detail file DSECT mapping

The following are the DSECTS that map the detail files that come from the BEARS / IMS jobsteps. They are supplied to enable you to perform more detailed enquiries if you wish. The BEARS / IMS installation process places these DSECTS in the BRSSRCE library.

The files from the BEARS jobstep contain the most detailed information. Some of this is dropped when moving to the BEARSUMM files. Certain fields are also dropped between the BEARSUMM files and the SAS databases to allow summarization by time.

BEARS JOBSTEP

FILE	DSECT
ABENDS	#BRSABND
CHKPFILE	#BRSOREC
FASTFILE	#BRSMFP
MIXFILE	#BRSMIX
MSGFILE	#BRSMMSG
RESOURCE	#BRSSMB #BRSAREA #BRSTERM
RESPFILE	#BRSMFF
TRANFILE	#BRSMTR
VIOFILE	#BRSVIO

BEARSUMM JOBSTEP

FILE	DSECT
ABENSUMM	#BRSABC
CHKPSUMM	#BRSCREC
FASTSUMM	#BRSCFP
MIXSUMM	#BRSCMX
MSGSUMM	#BRSCMSG
RESSUMM	#BRSSMB #BRSAREA #BRSTERM
RESPSUMM	#BRSCFF
TRANSUMM	#BRSCTR
VIOSUMM	#BRSVIC

This page intentionally left blank.



6. Appendix C - Sample SAS Enquiries and Macros

6.1. Configuring SAS macros

There are three macros that the user should configure, (if the CHKPFIL buckets are to be used), before you can begin to run BEARS / IMS. They are APPLIC, APPLIC2 and PROJDEFS. These macros contain information concerning the users transaction codes and Application groups. Several of the supplied example programs use these macros to gather and report on Application related information.

Examples of configuring each of these macros is included in the supplied examples.

The APPLIC macro cross references the BEARS / IMS internal groups (FULLnn, FASTMnn and FASTBnn) to your Business Application groups.

You simply decide which Application is to be your first Full Function group (FULL00) and set the PROJ variable to the required Application name. This process is repeated for all your Full Function groups (FULLnn), Fast Path Message Driven groups (FASTMnn) and Fast Path Batch groups (FASTBnn). The number of groups specified should correspond with the definitions in the TRANCARD parameter deck used in the BEARS jobstep.

The APPLIC2 macro allows you to define which IMS transaction codes belong to each Business Application and relate them to the SAS groups that you have chosen in the APPLIC macro.

You simply specify the transaction code to as many characters as you wish, (e.g. TRANSACT or TRANSA or TRA), and set the PROJECT variable to its Business Application name and the PROJCODE variable to the SAS group chosen in the APPLIC macro for this Application. These definitions should correspond to the TRANCARD parameter deck used in the BEARS jobstep.

The PROJDEFS macro is a cross reference of the SAS group names and the Business Application names.

You simply set each SAS group name to its Business Application name,
(e.g. FULL00 = PAYROLL)

These settings should correspond to the ones used in the APPLIC and APPLIC2 macros.

6.2. Example macros

An example of configuring the macros follows. The example shows two business applications.

The first is a Full Function Payroll System. One of the transaction names is STAFFPAY.

The second is a Fast Path cash withdrawal system. All of the transactions begin with the letters CWS.

```
Supplied preamble
..
..
IF FULL00 NE 0 THEN DO;
  TOT=FULL00;
  PCT=FULL00/B0TRANT*100;
  PROJ='PAYROLL';
  OUTPUT;
END;
IF FASTM01 NE 0 THEN DO;
  TOT=FASTM01;
  PCT=FASTM01/B0FTOT*100;
  PROJ='CASH WITHDRAWAL'
  OUTPUT;
END;
any other definitions
```

Figure 9 - APPLIC member

```
Supplied preamble
..
..
IF FFULTRN='STAFFPAY' THEN DO;
    PROJECT='PAYROLL';
    PROJCODE='FULL00';
    RETURN;
END;
IF FTRAN3='CWS' THEN DO;
    PROJECT='CASH WITHDRAWAL'
    PROJCODE='FASTM01'
    RETURN;
END;
any other definitions
```

Figure 10 - APPLIC2 member

```
Supplied preamble
..
..
%LET FULL00=PAYROLL;
%LET FASTM01=CASH WITHDRAWAL
any other definitions
```

Figure 11 - PROJDEFS member

6.3. SAS Allocations and Options

Several of the sample enquiries below include other SAS macros and routines. The samples are coded to include these macros and routines from the library allocated to the DD name BEARSSAS. The user must therefore allocate file BEARSSAS to the BEARS / IMS supplied SAS library. To run several of the graphic enquiries the user should also allocate file FT21F001 to a temporary dataset.

e.g.

```
ALLOC F(BEARSSAS) DA('your.BEARSIMS.SAS') SHR  
ALLOC F(FT21F001) SP(10,1) TR REUSE
```

Figure 12 - SAS allocations

BEARS / IMS also assumes that either SAS was installed with the System options of MACRO, MCOMPILE and NOTEXT82, or that these options are specified as parameters in the execution of SAS.

6.4. SAS Performance Options

The data collection and information production side of BEARS / IMS (i.e. the BEARS SLDS processing job), will perform with very much the same profile in any site. This should be extremely fast and efficient. However depending on the amount of data that the IMS system produces, then the SAS jobs for data summarization and reporting can run for much longer periods. It may therefore be advisable in large sites to try to tune this side of the system. Some tuning hints follow. Some are obvious and are included only for completeness. Others are not so obvious and may assist in reducing job execution times. The benefits to be gained from these hints depend upon many factors, IMS system size, DASD utilization, CPU availability, etc. However they have been proved to give up to a 50 percent reduction in elapsed times while running the SAS summarization jobs or certain enquiries at individual client sites.

SAS Load Modules

The following discussion relates to users of SAS Version 5 or earlier.

Because SAS is an interpreted language, certain of its component modules are loaded for each record processed. This means that if BEARS / IMS was performing a 'PROC SUMMARY' on a million records then there would be a million loads issued for the SAS summarisation load module SUMMARY1.

Some customers have attempted to solve this problem using third party Program Load software or by placing the SAS load library in the Linklist and using Linklist Look Aside (LLA). Another option would be to copy the SUMMARY1 load module into a very small VIO dataset and to place this dataset as the first entry in the STEPLIB of all SAS jobsteps.

We also recommend the use of the optional user built non-overlay version of the SAS supervisor module SASLPA instead of the overlay version SAS. Users should consult the SAS Installation Manual regarding this option.

SORT Options

A lot of time is spent both in enquiries and in the summarization jobs in performing sorts. Therefore it is important that the correct sort options are used. Adding the SAS options 'SORTMSG,SORTLIST' to the SAS input will produce information on which options are in effect. Some to check for are;

```
SIZE=MAX          (And that MAX has been set to a least 1 Megabyte).  
CHECK=NOCHECK  
EQUALS=NOEQUALS
```

It is also worth checking that the SORT program called is not SASSORT, but your system sort (e.g. DFSORT or SYNCSORT).

As a general point, it has been found that multiple small SORTWORK datasets give better performance than on large one. This is especially true if they are all on different spindles and the volume of data to be sorted is large. Therefore it is more efficient to have six SORTWORK datasets of 10 cylinders each, on six different spindles, than to have one SORTWORK dataset of 60 cylinders.

BEARS / IMS Datasets

As stated in the installation procedure, the BEARS / IMS datasets are all allocated on one spindle to simplify the install. However as is also stated, there are performance benefits to be gained from having them spread over as many spindles as possible. At the very minimum this should mean that the SAS databases, (Weekly, Monthly and Yearly), are not on the same spindle as each other or any other BEARS / IMS dataset.

6.5. SAS Sample Enquiries

The following is a list of the sample enquiries distributed on the BEARS / IMS tape.

<u>Enquiry</u>	<u>Description</u>
ABENDS	Graphic enquiry showing transaction abends for a given system and day
ABENDS2	Non-graphic version of ABENDS
APPLIC	Macro to cross-reference BEARS / IMS groups to Application projects. This macro MUST be edited by the user before beginning to run BEARS / IMS
APPLIC2	Macro to cross-reference transaction codes to Application projects. This macro MUST be edited by the user before beginning to run BEARS / IMS
AVGTRN	Graphic enquiry which builds a composite day over a month showing the peak against average transaction rates plotted by hour
FASTRESP	Graphic enquiry to show a Fast Path transactions response time across a day
FASTRSP2	Non-graphic version of FASTRESP
FASTTAB	Graphic enquiry to show a Fast Path transactions response time across a day in a tabular format
FASTTAB2	Non-graphic version of FASTTAB
GROWTH	Graphic enquiry to show the prime shift peak rate transaction growth since BEARS / IMS began monitoring
IO	Graphic enquiry to show the peak rate I/O breakdown
IO2	Non-graphic version of IO
MIXRESP	Graphic enquiry to show a Mixed Mode transactions response time across a day
MIXRESP2	Non-graphic version of MIXRESP
MIXTAB	Graphic enquiry to show a Mixed Mode transactions response time across a day in a tabular format
MIXTAB2	Non-graphic version of MIXTAB
POOLUTIL	Graphic enquiry to show the IMS Pool utilizations.
POOLUTL2	Non-graphic version of POOLUTIL
PRIMETR	Graphic enquiry to show the peak transaction rate during prime shift across a month
PRIMETR2	Non-graphic version of PRIMETR

<u>Enquiry</u>	<u>Description</u>
PROJDAY	Non-graphic enquiry to show the daily workload broken down into Application projects
PROJDAYP	Graphic version of PROJDAY in pie chart format
PROJDEFS	Macro to label the Application groups in the SAS databases. The user MUST edit this macro before beginning to run BEARS / IMS
PROJHOUR	Non-graphic enquiry to show the peak hour broken down into Application projects
PROJMON	Non-graphic enquiry to show the month broken down into Application projects
PROJMONP	Graphic version of PROJMON in pie chart format
RATE	Graphic enquiry to show the transaction rate across time for a day
RATE2	Non-graphic version of RATE
RATES	Graphic enquiry to show the transaction rate across time broken into its components, MPP, BMP, etc.
RATES2	Non-graphic version of RATES
RESAREA	Graphic enquiry to show Fast Path Area resource utilization
RESAREA2	Non-graphic version of RESAREA
RESPROJ	Non-graphic enquiry showing response times by Application project.
RESPTAB	Graphic enquiry to show the response time of a Full Function transaction in a tabular format
RESPTAB2	Non-graphic version of RESPTAB
RESPTIME	Graphic enquiry to show the response time of a Full Function transaction
RESPTIM2	Non-graphic version of RESPTIME
RESSMB	Graphic enquiry to show SMB resource utilization
RESSMB2	Non-graphic version of RESSMB
RESTERM	Graphic enquiry to show Terminal resource utilization
RESTERM2	Non-graphic version of RESTERM
SCHDFAIL	Graphic enquiry to show Transaction schedule failures
SCHDFAL2	Non-graphic version of SCHDFAIL
TENCALL	Graphic enquiry to show the top ten transactions which issue the most DL/I calls per schedule.
TENCALL2	Non-graphic version of TENCALL
TENCPU	Graphic enquiry to show the top ten transactions who used the most CPU per schedule
TENCPU2	Non-graphic version of TENCPU

<u>Enquiry</u>	<u>Description</u>
TENOBA	Graphic enquiry to show the top ten enquiries which used the most OBA per schedule
TENOBA2	Non-graphic version of TENOBA
TRACKING	Non-graphic enquiry to show the IMS system status for a day.
TRANRESP	Graphic enquiry to show a transactions response time across a day
TRANRSP2	Non-graphic version of TRANRESP
TRANSCHD	Graphic enquiry to show transactions per scheduling
TRANTAB	Graphic enquiry to show a transactions response time across a day in a tabular format
TRANTAB2	Non-graphic version of TRANTAB
TRNCLASS	Non-graphics example of using BEARS to assign transaction to classes by profile.
TOTALTRN	Graphic enquiry to show the total transaction throughput per day
TOTALTR2	Non-graphic version of TOTALTRN
VIOLATE	Graphic enquiry to show security violations
VIOLATE2	Non-graphic version of VIOLATE

This page intentionally left blank



This manual is published by

VIO Systems Limited

<http://www.viosystems.com>
email bears@viosystems.com

BEARS/IMS is marketed by

Span Software Consultants Limited

<http://www.spansoftware.com>
email bears@spansoftware.com

