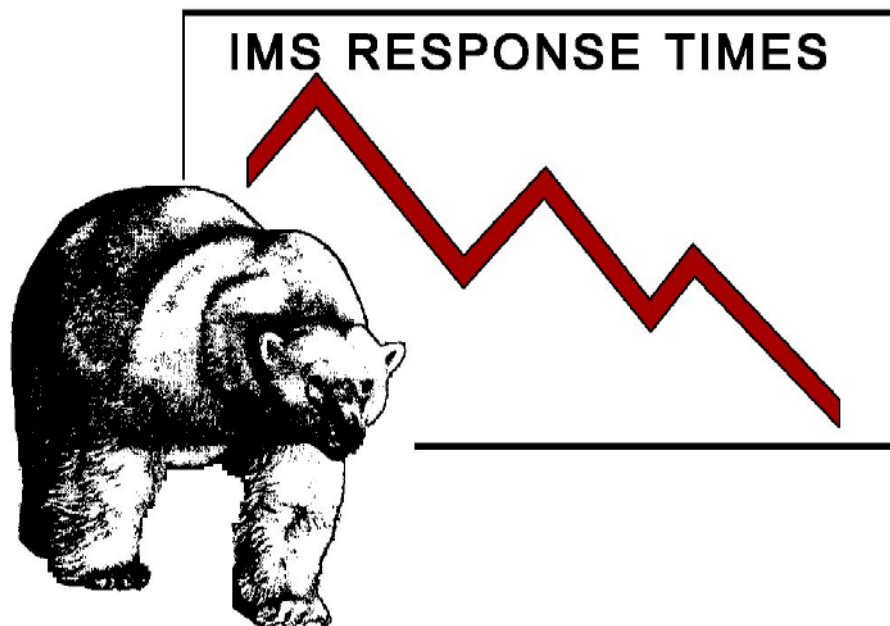


# BEARS

# IMS

## Installation and Maintenance Manual

VIO Systems Limited



Version: 03.6

Product Number: VIO-002

Revision: 1st September 2004

Manual Ref: BRS-03-013

This page intentionally left blank



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.



This page intentionally left blank



<b><u>CONTENTS</u></b>	<b><u>Page</u></b>
1 Introduction .....	7
1.1 BEARS / IMS DASD Space requirements.....	7
2 How does BEARS / IMS work?.....	9
2.1 BEARS / IMS jobs .....	10
3 Management of the BEARS / IMS Software .....	13
3.1 BEARS / IMS Release CDs .....	13
3.2 SPAN SPIDER System .....	13
3.3 BEARS / IMS Installation Procedure using SMP/E .....	14
3.3.1 BEARS / IMS Installation CD.....	14
3.3.2 BEARS / IMS Installation Procedure Overview.....	14
3.3.3 BEARS / IMS Basic Material Installation Steps .....	15
3.4 Application of BEARS / IMS Maintenance using SMP/E.....	23
4 BEARS / IMS Module Tables.....	25
4.1 BEARS / IMS CSECT Details .....	25
4.2 BEARS / IMS Module Functions.....	26
4.3 BEARS / IMS Macros .....	28
5 BEARS / IMS Generation Process.....	31
5.1 Use of #BRSGEN Macro .....	31
5.2 #BRSGEN Macro - Generate BEARS / IMS Installation JCL .....	32
6 Appendix A .....	37
6.1 BEARS / IMS generated JCL members and Control cards.....	37



**List of Figures****Page**

Figure 1 - Schematic of the BEARS/IMS System.....	9
Figure 2 - BEARS/IMS CD contents.....	14
Figure 3 - SMP/E installation procedure .....	16
Figure 4 - Installation JCL.....	17
Figure 6 - Sample job to add SMP/E procedure.....	19
Figure 7 - Sample SMP/E install .....	20
Figure 8 - Sample JCLIN.....	21
Figure 9 - Sample SAS unload .....	21
Figure 12 - Sample SMP/E Fix.....	24
Figure 14 - #BRSGEN Macro Syntax .....	32

**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

This manual explains the format in which maintenance to BEARS / IMS is distributed to users. The various forms of distribution are described, covering BEARS / IMS release CDs, and source and load module updates.

BEARS / IMS is released on Compact Disc (CD) in SMP/E format. Each new release of BEARS / IMS will be distributed on a new CD in the same format and will contain a complete replacement for all BEARS / IMS modules and macros. From time to time in between major releases, refresh release CDs of BEARS / IMS may be made available, and these will also be in the same format and will contain a complete replacement of all BEARS / IMS modules and macros.

BEARS / IMS is coded in such a way that only the very few modules which contain IMS or Operating system dependencies are released in source form. All other modules are supplied as object code only. Maintenance of these modules is via SUPERZAP updates. Section 4 of this manual contains a table of BEARS / IMS modules and indicates to which modules this applies.

At the end of this manual, some BEARS / IMS Check Lists are supplied. One of these covers the major tasks of installing BEARS / IMS either for the first time or from a refresh CD. The other covers the major tasks of applying BEARS / IMS maintenance. These Check Lists may be photo-copied and used by systems programming staff to check the progress of BEARS / IMS installation or maintenance activities, and should serve to reduce the time for these activities and to eliminate errors.

## 1.1 BEARS / IMS DASD Space requirements

This version of BEARS / IMS has the following DASD space requirements:

Distribution libraries.	Approx 35 cylinders of a 3380.
SAS Database system	Approx 350 cylinders of a 3380.

Note. The supplied SAS Database system is large enough to hold all the data from an average sized IMS system for one year.



This page intentionally left blank.



## 2 How does BEARS / IMS work?

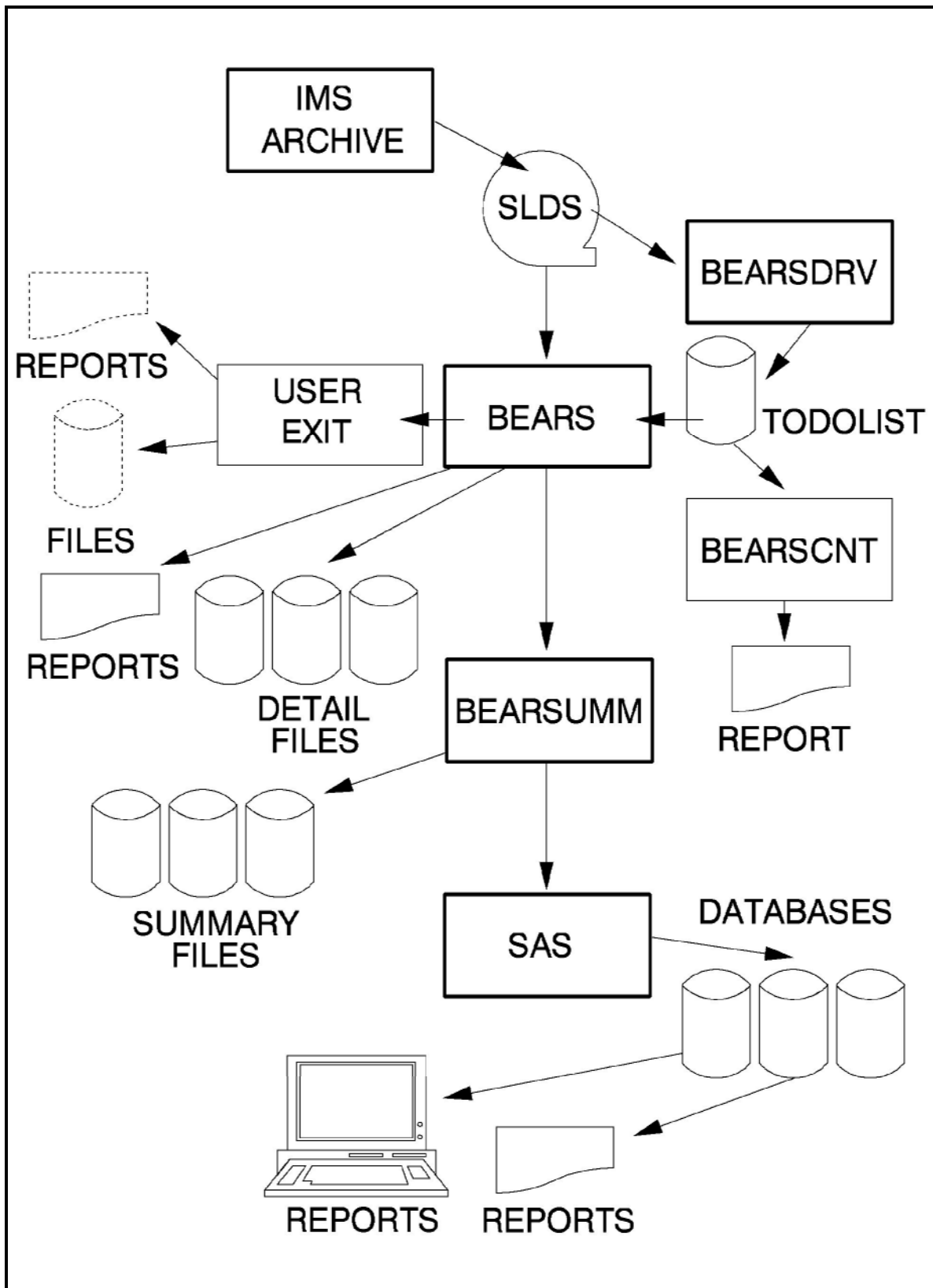


Figure 1 - Schematic of the BEARS/IMS System

## 2.1 BEARS / IMS jobs

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. It is important to note that this step does not interfere with IMS archive in any way. Even if this step were to fail IMS would mark archive as having completed successfully. This information is placed into the TODOLIST dataset. 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.

Should the user be using cataloged SLDS volumes, he may optionally replace the BEARSDRV program with BEARSCAT. This routine performs exactly the same functions as BEARSDRV, but because it accesses the catalog, will do so without the requirement of a tape allocation. Users who have heavy contention for the devices that record SLDS datasets may find this of use.

An example of this step (using BEARSDRV) is supplied in the BRSSRCE library as BRSINST2.

The BEARS / IMS job contains two steps. The first step runs the actual BEARS program. This will allocate the first volume from the TODOLIST dataset and process it. The output files from step BEARS are very heavily detailed and are described in the supplied DSECTS. Refer to Appendix B of the BEARS / IMS General Usage Manual 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.

An example of this job is supplied in the BRSSRCE library as BRSINST5. The required parameter cards are members BRSINST3 and BRSINST4.

The second step in the BEARS / IMS job runs program BEARSUMM. This program summarizes the data from the files created by BEARS 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. After the SAS job has successfully completed then this job should be restarted from the BEARSUMM step. (Note. Refer to the NODATE parameter card for the exception to this).

Users may wish to automatically submit the SAS summarization job in these cases by coding a step in the JCL which executes only when a condition code of 04 is issued from BEARSUMM. The two reasons given above are the only ones which result in a

condition code of 04 from BEARSUMM.

This step is also described in member BRSINST5. The parameter card used is member BRSINST3.

The BEARS / IMS job will then resubmit itself until there are no more volumes in the TODOLIST to be processed. It will then inform the operators that there are no more volumes to process and end with a return code of 04.

An optional third step can be coded using the BEARSCNT program. This program is executed with a parameter of the BEARS / IMS TODOLIST dataset name (as the BEARSDRV program). It will report on the number of entries in the TODOLIST and end with a return code equal to this number. This condition code can be used by a scheduling system or JCL condition code checking to avoid resubmitting a BEARS / IMS job when there are no entries in the TODOLIST (ie. the BEARSCNT step ends RC00). This should halve the number of BEARS / IMS jobs submitted in normal processing. However, it also means that jobs which complete normally will not end RC00.

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 2.7 of the BEARS / IMS General Usage manual for the 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 and then the failing SAS job be rerun. Examples of these jobs are given in BRSSRCE as members BRSINST6, BRSINST7 and BRSINST8.

This page intentionally left blank.



---

## 3 Management of the BEARS / IMS Software

### 3.1 BEARS / IMS Release CDs

All mainframe VIO Systems Products are distributed on CD. JCL for the installation of BEARS / IMS is generated by means of the BEARS / IMS #BRSGEN macro, operands of which specify all the global options available with BEARS. The #BRSGEN macro is described in Chapter 5 of this manual.

The BEARS / IMS Release CD is in SMP/E format, and the recommended procedures for installing the product using SMP/E are given in Section 3.3 below.

All BEARS / IMS Release CDs contain all distributed BEARS / IMS modules, so that BEARS / IMS distribution libraries can be completely replaced with the new data from a new CD.

### 3.2 SPAN SPIDER System

The SPIDER (Span Product Information Distribution and Error Reporting) system is essentially an Early Warning System designed to avoid the necessity of correcting the same problem multiple times when it occurs in more than one installation, and to ensure the fastest possible response to customer queries.

The SPIDER system is used for the distribution of corrections for problems found in VIO Systems Products, and also for the distribution of minor enhancements to the products.

SPIDERS are produced as emails and are sent to all affected customers as they occur. For corrections or enhancements that affect a small amount of coding, the source updates or SUPERZAP control statements will be supplied with the SPIDER form. The customer may also request that SPIDERS are delivered as paper.

Source is supplied for all BEARS / IMS modules that have IMS or Operating System dependencies. For these modules, SPIDERS involve source changes, which must be applied as supplied. For modules for which the source is not supplied with the BEARS / IMS release, SPIDERS involve the use of SUPERZAP (program AMASPZAP or HMASPZAP).

It is the users responsibility to ensure that all BEARS / IMS changes, whether in source-update or SUPERZAP form, are correctly applied, and that a record of the status of the BEARS / IMS system is maintained at all times. For SMP/E users, such a record is automatically held in the CSI.

#### SMP/E Maintenance

SPIDERS will be supplied in the SMP/E format of PTFs. They may be installed onto your system using the JCL procedure described in Section 3.3.3.1 in this manual.



The SMP/E processes RECEIVE and APPLY will update the BEARS / IMS target libraries. VIO Systems Limited recommends that all SPIDERS be ACCEPTed onto your distribution libraries, to ensure that no maintenance is inadvertently "lost" should re-assembly of the #BRSGEN macro be necessary. In this case furthermore, the SMP/E JCLIN function, as described in Section 3.3.3.3 of this manual (Step 8), should be re-run after any such change to the #BRSGEN parameters.

### 3.3 BEARS / IMS Installation Procedure using SMP/E

#### 3.3.1 BEARS / IMS Installation CD

The supplied CD contains a single TSO transmitted PDS file. When reconstructed this file will recreate the following files:

<u>File</u>	<u>Name</u>	<u>Content</u>
1	VBR36000	SMP modification control statements, including PTFs (if any)
2	VBR36001	BEARS / IMS source modules
3	VBR36002	BEARS / IMS load modules
4	VBR36003	BEARS / IMS macros
5	VBR36004	BEARS / IMS sample installation JCL
6	VBR36005	BEARS / IMS SAS source
7	VBR36006	BEARS / IMS SAS examples

Figure 2 - BEARS/IMS CD contents

BEARS / IMS is supplied on a standard CD in SMP/E Format.

#### 3.3.2 BEARS / IMS Installation Procedure Overview

The BEARS / IMS feature must be installed using the IBM System Modification Program service aid (SMP/E).

Installation JCL is included on the BEARS / IMS distribution CD. BEARS / IMS is distributed as a "feature", with FMID VBR3600. A BEARS-supplied macro, #BRSGEN, is placed in a macro library and an assembly of this macro must be performed during the installation procedure. Using parameters provided by the user, #BRSGEN will generate a jobstream that, when executed, will assemble all required modules and then perform all the necessary link-edits to place the feature's routines into a user-named library. BEARS / IMS may be supplied on the release CD as a Base Release, or as a Base Release plus PTFs. If PTFs are included on the CD, the appropriate ++PTF statements, etc, will be included in the

MCS file. RECEIVE, APPLY and ACCEPT should be performed for the FMID and all PTFs before attempting the BEARS / IMS generation process.

### **3.3.3 BEARS / IMS Basic Material Installation Steps**

#### **3.3.3.1 SMP/E JCL Procedure**

BEARS / IMS may be installed in an existing SMP/E CSI, or alternatively in a null CSI. Additional installation job steps are included to initialize a null CSI. A JCL procedure for SMP/E must be supplied or an existing SMP/E procedure may be modified. This procedure will be used in the RECEIVE, APPLY and ACCEPT steps of the SMP/E process.

A sample SMP/E procedure is shown below, and is also included in the VBR36004 file reconstructed from the BEARS / IMS distribution CD.

Following is an example of an SMP/E procedure for the installation of BEARS:

```

//BRSSMPE  PROC SOUT=A,NODE=BEARS,UNIT=3390,
//          NODE1=BEARS,VOL=BEARS
//S1       EXEC PGM=GIMSMP
//SYSPRINT DD SYSOUT=&SOUT
//SMPOUT  DD SYSOUT=&SOUT
//SMPCSI  DD DSN=&NODE..CSI,DISP=SHR
//SMPLOG  DD DSN=&NODE..SMPLOG,DISP=SHR
//SMPMTS  DD DSN=&NODE..SMPMTS,DISP=SHR
//SMPPTS  DD DSN=&NODE..SMPPTS,DISP=SHR
//SMPSTS  DD DSN=&NODE..SMPSTS,DISP=SHR
//SMPSCDS DD DSN=&NODE..SMPSCDS,DISP=SHR
//SMPTLIB DD UNIT=&UNIT,VOL=SER=&VOL,DISP=OLD
//SMPWRK1 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=6160,LRECL=80,RECFM=FB)
//SMPWRK2 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=6160,LRECL=80,RECFM=FB)
//SMPWRK3 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SMPWRK4 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)
//SMPWRK5 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=6233,RECFM=U)
//SMPWRK6 DD UNIT=SYSDA,SPACE=(CYL,(5,1,20)),DISP=(,DELETE),
//          DCB=(BLKSIZE=6160,LRECL=80,RECFM=FB)
//SYSLIB  DD DSN=&NODE1..BRSSRCE,DISP=SHR
//          DD DSN=SYS1.MACLIB,DISP=SHR
//          DD DSN=SYS1.AMODGEN,DISP=SHR
//SYSUT1  DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(5,1))
//SYSUT2  DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(5,1))
//SYSUT3  DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(5,1))
//SYSUT4  DD UNIT=SYSDA,DISP=(,DELETE),SPACE=(CYL,(5,1))
//*
//*      DISTRIBUTION LIBRARIES
//BRSLoad DD DSN=&NODE1..BRSLoad,DISP=SHR
//BRSSRCE DD DSN=&NODE1..BRSSRCE,DISP=SHR
//*
//*      INSTALLATION LIBRARIES
//VBR36001 DD DSN=&NODE1..VBR36001,DISP=SHR
//VBR36002 DD DSN=&NODE1..VBR36002,DISP=SHR
//VBR36003 DD DSN=&NODE1..VBR36003,DISP=SHR
//*
//*      TARGET LIBRARY
//BRSLKLIB DD DSN=&NODE1..BRSLKLIB,DISP=SHR

```

Figure 3 - SMP/E installation procedure

Note: You may change the DSN parameters in the sample SMP/E procedure to match your Installation standards. However you should ensure for the distribution and target libraries that:

- 1) The ddname is the same as the low level DSN qualifier;
- 2) You specify the same Dataset Name prefix when coding the #BRSGEN macro, i.e. the DSNPREF Parameter of the #BRSGEN macro and the NODE1 parameter of the above procedure should specify the same value.

### 3.3.3.2 Preparing for BEARS / IMS Installation

The steps necessary to install the BEARS / IMS feature onto your system are given below in Section 3.3.3.3, BEARS / IMS Installation Steps.

If you are using existing SMP/E datasets then bypass Steps 3 and 4.

In this case, ensure that the DSSPACE subparameter used to initialize your SMPPTS dataset has values of at least (300,300,500).

It may be necessary to REPLACE these values in your existing SMPPTS system entry using the SMP/E UCLIN process. Sample statements are provided in installation member INSTAL4 (see below).

The VBR36004 file reconstructed from the BEARS / IMS distribution CD contains unloaded PDS members to aid with BEARS / IMS installation. The following is a list of the members and their functions:

INSTAL1	Sample JCL to allocate SMP/E datasets
INSTAL2	Sample JCL for initializing SMP/E CSI and PTS datasets
INSTAL3	Sample JCL to allocate BEARS / IMS distribution libraries
INSTAL4	Sample JCL for BEARS / IMS Stage 1 assembly
BRSSMPE	Sample SMP/E procedure

Figure 4 - Installation JCL

### 3.3.3.3 BEARS / IMS Installation Steps

Step 1 Upload the contents of the CD

Using your preferred TSO emulator or FTP program upload the following files from the CD to the mainframe.

FileName	Contents	File Format
BEARS.XMITPDS	Compressed Distribution libraries	Binary
UNPKDLIB.TXT	JCL to recreate libraries	Text

Note. Ensure that you specify the correct file format for each file on the upload. (e.g. The process will fail if you specify text for XMITPDS)

Step 2 Recreate the Distribution Libraries and JCL

Once both files have been loaded, edit the Job UNPKDLIB to your local standards and execute it. This will recreate the DLIB libraries on your mainframe and unpack the installation JCL necessary to complete the installation (Figure 4) which is used in the following steps.

Step 3 Allocate SMP/E datasets. Use member INSTAL1. This step may be bypassed if an existing SMP/E environment is being reused.

Step 4 Initialize SMP/E CSI. Use member INSTAL2. Amend control statements as necessary before executing. This step may be bypassed if an existing SMP/E environment is being reused

Step 5 Allocate BEARS / IMS distribution libraries and target and object datasets. Use member INSTAL3.

Step 6 Place an SMP/E procedure, described in 3.3.3.1, in a procedure library. The following is an example of the statements required to place the BRSSMPE procedure in a procedure library:

```
//JOB2      JOB   [ Job Card Parameters ]
//UPDATE    EXEC  PGM=IEBUPDTE,PARM=NEW
//SYSPRINT  DD   SYSOUT=A
//SYSUT2    DD   DSN=SYS1.PROCLIB,DISP=SHR
//SYSIN     DD   DATA
./  ADD NAME=BRSSMPE,LIST=ALL
./  NUMBER NEW1=10,INCR=10
[   Sample source shown in Section 3.3.3.1   ]
./  ENDUP
/*
```

Figure 5 - Sample job to add SMP/E procedure

Note: You may modify your existing SMP/E procedure if you are installing BEARS / IMS into your existing SMP/E libraries.

Step 7 Perform the SMP/E operations of Receiving, Applying and Accepting the BEARS / IMS feature. This task will place all the modules, macros and source modules into the BEARS / IMS distribution datasets and update the SMP/E CSI dataset. The BEARS / IMS installation macro, #BRSGEN, will be unloaded by this task. Then, if any PTFs were included on the release CD (this may be determined from the SMP/E RECEIVE report), APPLY and ACCEPT all these PTFs, using the SELECT parameter of the SMP/E APPLY and ACCEPT statements to specify the SYSMOD IDs of the PTFs.

Following is an example of the statements required to perform this task for SMP/E:

```

//JOB3      JOB  [ Job Card Parameters ]
//          EXEC BRSSMPE
//SMPPTFIN DD DSN=chosenprefix.VBR36000,
//          DISP=OLD
//SMPCNTL  DD *
            SET BDY(GLOBAL) .
            RECEIVE SYSMODS.
            SET BDY(BRSTGT) .
            APPLY SELECT(VBR3600) .
            SET BDY(BRSDLIB) .
            ACCEPT SELECT(VBR3600) .

/*

```

Figure 6 - Sample SMP/E install

In these examples, BRSSMPE is the procedure names for executing the SMP/E program GIMSMP. A sample BRSSMPE procedure is shown in Section 3.3.3.1, and is included in the VBR36004 file reconstructed from the BEARS / IMS distribution CD.

Note: Change the UNIT parameter to accommodate your Installation standards.  
 IGNORE any warning messages issued during APPLY processing.  
 IGNORE any Condition Code 4 from the link-edit issued during  
 ACCEPT processing.

Step 6 Assemble the #BRSGEN macro. This macro has been placed in your BEARS / IMS BRSSRCE dataset by the procedure described in Step 5. The #BRSGEN macro is coded with the parameters described in Section 5 of this manual. An assembly of #BRSGEN will build, via SYSPUNCH output, the required SMP/E JCLIN statements you need to complete the installation of the feature. It is suggested that the assembly step for the #BRSGEN macro specifies the assembler parameter 'DECK' and that the SYSPUNCH DD statement is allocated to a sequential dataset or a PDS member (the same library used for the supplied sample JCL in Section 3.3.3.2 above may be used). Sample JCL for the assembly of the #BRSGEN macro is provided in member INSTAL4.

The "Stage 2" job stream generated by the assembly of the #BRSGEN macro consists of the JCL for a sequence of jobs, which **must be run, in the order generated**, to assemble and link-edit the BEARS / IMS modules into the executable target library (**BRSLKLIB**). This JCL must also be input as JCLIN to update the SMP/E CSI. In addition, the Stage 2 process will install the example JCL and parameter cards into the BEARS / IMS distribution dataset, BRSSRCE.

The syntax and options of the #BRSGEN macro are fully described in Section 5 of this manual.

Step 7 Run the job stream generated by Step 6, to update the BEARS / IMS target library with executable load modules. The jobs must be run in the order that they are generated.

Step 8 Perform "JCLIN" processing to update the SMP/E CSI dataset.

The following is an example of the statements required to perform this task for SMP/E:

```
//JOB4      JOB  [ Job Card Parameters ]
//S1        EXEC BRSSMPE
//SMPCNTL  DD *
            SET BDY(BRSTGT) .
            JCLIN ASM(PGM=IEV90) .
/*
//SMPJCLIN DD DSN=your.stage2.jcl,DISP=SHR
```

Figure 7 - Sample JCLIN

Step 9 Unload the SAS control card members and example enquiries.

The following is an example of the statements required to perform this task.

```
//JOB5      JOB  [ Job Card Parameters ]
//S1        EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//I1 DD DSN=chosenprefix.VBR36005,
//        DISP=SHR
//I2 DD DSN=chosenprefix.VBR36006,
//        DISP=SHR
//O1 DD DSN=dsnprefix.BEARS.SAS,DISP=OLD
//SYSIN DD *
            C O=O1,I=(I1,I2)
/*
```

Figure 8 - Sample SAS unload

Step 10 The BEARS / IMS source library contains samples of a number jobs that may be of use.

These include: a sample BEARSDRV archive step,

a sample BEARS / IMS job to process a SLDS dataset and place the data in the daily datasets,

a sample daily job to move the data from the daily datasets to the weekly database,

a sample weekly job to move the data from the weekly database to the monthly database,

and a sample job to move the data from the monthly to the yearly database.

All the SAS control card members required for these functions are provided.

A job is provided to allocate all the datasets and databases required. It will also write end of file markers to the necessary files. **This job must be run.**

Also provided is an example of the SAS code required to initialize and label the SAS databases. **This job must be run.**

The SAS macros which cross reference the users transaction codes to the business applications must now be configured. These are APPLIC, APPLIC2 and PROJDEFS. Refer to the BEARS / IMS General Usage Manual, Section 6 for more details.

Examples of all the control cards required to run BEARS / IMS are also included.

Refer to Appendix A for a list of the generated JCL members and control cards.

### Step 11

The installation of the BEARS / IMS system is now complete, however we suggest that you read Section 2 of the BEARS / IMS General Usage Manual for a discussion of some points of interest before continuing.

### 3.4 Application of BEARS / IMS Maintenance using SMP/E

SPIDERS are sent via email to all affected customers as they occur. For corrections or enhancements that affect a small amount of coding, the source updates or SUPERZAP control statements will be supplied, in SMP/E format, with the SPIDER form.

SPIDERS will be supplied in the SMP/E format of PTFs. They may be installed onto your system using the JCL procedure described in Section 3.3.3.1 in this manual. The SMP/E processes RECEIVE and APPLY will update the BEARS / IMS target libraries. VIO Systems Limited recommend that all SPIDERS be ACCEPTed onto your distribution libraries, to ensure that no maintenance is inadvertently "lost" should re-assembly of the #BRSGEN macro be necessary. In this case, furthermore, the SMP/E JCLIN function, as described in Section 3.3.3.3 of this manual (Step 8), should be re-run after any such change to the #BRSGEN parameters.

**Note** that assemblies *must* be performed for all modules to which source maintenance is applied, and that the 'ASSEM' parameter of the SMP/E APPLY and ACCEPT statements may be required to ensure this in your installation.

Source updates to BEARS / IMS are supplied in printed SMP/E format, without any Job Control Language. BEARS / IMS source modules are re-sequence-numbered only for major BEARS / IMS releases. The SPIDER number is always included in a header statement which is inserted at the beginning of each updated source module, and the low-order four characters of the SPIDER number are placed on each changed source statement.

SUPERZAP updates to BEARS / IMS are supplied in printed SMP/E format, without any Job Control Language. The SPIDER number is always included as IDRDATA in BEARS / IMS SUPERZAPS.

Note that BEARS / IMS PTFs are supplied with SMP/E '++VER' statements that may contain an incomplete pre-requisite list specified via the 'PRE' parameter. Other BEARS / IMS PTF numbers are specified as pre-requisites only if they are required to be applied for the correct functioning of the newer PTF. Other PTFs are not specified as pre-requisites merely because they apply to the same modules. This is to allow maintenance to be applied selectively when required. Thus it is possible validly to obtain "ID checks" when running SMP/E processes for BEARS / IMS PTFs. These can be avoided either by altering the pre-req list on each ++VER statement to include all PTFs that have been applied to your BEARS / IMS system for the same module, or by using the BYPASS(ID) SMP/E parameter.

Example of SUPERZAP Control Statements

Supplied in SMP/E format:

```
++ZAP (MOD1) .  
NAME CSECT1  
VER A0 4780  
REP A0 47F0  
IDRDATA XX00000
```

Figure 9 - Sample SMP/E Fix

## 4 BEARS / IMS Module Tables

The following tables list all BEARS / IMS and related modules, and identify the load module in which each object module resides. Each module is identified with an indicator of whether or not source code is supplied. If source is supplied, maintenance will be via source update and the module must be re-assembled. Updates to macros will always be in source form, and assemblies for related modules and/or SUPERZAPs may also be required for macro changes - these will be explained in the documentation accompanying the fix details. If source is not supplied, maintenance will be via SUPERZAP. Pre-assembled object modules are supplied for all BEARS / IMS modules.

### 4.1 BEARS / IMS CSECT Details

<u>Module</u>	<u>CSECT</u>	<u>Load Mod</u>	<u>Source</u>
BEARSABC	BEARSABC	BEARSUMM	NO
BEARSABN	BEARSABN	BEARS	NO
BEARSBLD	BEARSBLD	BEARS	YES
BEARSCAT	BEARSCAT	BEARSCAT	NO
BEARSCHC	BEARSCHC	BEARSUMM	NO
BEARSCHK	BEARSCHK	BEARS	NO
BEARSCNT	BEARSCNT	BEARSCNT	NO
BEARSCOL	BEARSCOL	BEARSUMM	NO
BEARSCTL	BEARSCTL	BEARS	NO
BEARSDRV	BEARSDRV	BEARSDRV	NO
BEARSEOF	BEARSEOF	BEARSEOF	NO
BEARSEST	BEARSEST	BEARS	NO
BEARSEXC	BEARSEXC	BEARS	NO
BEARSEXT	BEARSEXT	BEARS	NO
BEARFFC	BEARFFC	BEARSUMM	NO
BEARFFPC	BEARFFPC	BEARSUMM	NO
BEARFFST	BEARFFST	BEARS	NO
BEARSGET	BEARSGET	BEARS	NO
BEARSMG1	BEARSMG1	BEARS	NO
BEARSMG2	BEARSMG2	BEARS	NO
BEARSMG3	BEARSMG3	BEARS	NO
BEARSMG4	BEARSMG4	BEARS	NO
BEARSMG5	BEARSMG5	BEARS	NO
BEARSMIX	BEARSMIX	BEARS	NO
BEARSMSC	BEARSMSC	BEARSUMM	NO

<u>Module</u>	<u>CSECT</u>	<u>Load Mod</u>	<u>Source</u>
BEARSMXC	BEARSMXC	BEARSUMM	NO
BEARSPAR	BEARSPAR	BEARSUMM	NO
BEARSPAR	BEARSPAR	BEARS	NO
BEARSREL	BEARSREL	BEARS	YES
BEARSRUC	BEARSRUC	BEARSUMM	NO
BEARSRUT	BEARSRUT	BEARS	NO
BEARSSCC	BEARSSCC	BEARSUMM	NO
BEARSSEC	BEARSSEC	BEARS	NO
BEARSSIZ	BEARSSIZ	BEARS	YES
BEARSSRT	BEARSSRT	BEARS	NO
BEARSTRC	BEARSTRC	BEARSUMM	NO
BEARSTRN	BEARSTRN	BEARS	NO
BEARSTRX	BEARSTRX	BEARS	NO
BRSEEXIT	BRSEEXIT	N/A	YES
SPGLBL	SPGLBL	BEARS	NO
SPOUTPUT	SPOUTPUT	BEARS	NO
SPZPARSE	SPZPARSE	BEARS	NO
SPZQPAM	SPZQPAM	BEARS	NO

## 4.2 BEARS / IMS Module Functions

<u>Module</u>	<u>Function</u>
BEARSABC	BEARS / IMS Abend summarization routine
BEARSABN	BEARS / IMS Abend routine. Reports upon transaction abends
BEARSBLD	BEARS / IMS table build routine. Builds incore tables of the IMS log record information required
BEARSCAT	BEARS / IMS Driver program. May optionally run as the second step of IMS archive.
BEARSCHC	BEARS / IMS Checkpoint summarization routine
BEARSCHK	BEARS / IMS Checkpoint processor. Produces IMS internals information at checkpoint intervals
BEARSCNT	BEARS / IMS TODOLIST entry counter. Counts the number of SLDS entries in the TODOLIST.
BEARSCOL	BEARS / IMS Control routine. Controls all summarization tasks
BEARSCTL	BEARS / IMS Control routine. Control all data gathering tasks

<u>Module</u>	<u>Function</u>
BEARSDRV	BEARS / IMS Driver program. Runs as a second step to IMS archive
BEARSEOF	BEARS / IMS empty sequential file program
BEARSEST	BEARS / IMS Estae routine. Handles abends in user exits.
BEARSEXC	BEARS / IMS Full Function Response exception reporter
BEARSEXT	BEARS / IMS IMS SLDS extract program. Gathers the required log records
BEARSFFC	BEARS / IMS Full Function Response summarization routine
BEARSFPC	BEARS / IMS Fast Path Response summarization routine
BEARSFST	BEARS / IMS Fast Path Response routine
BEARSGET	BEARS / IMS GETMAIN routine. Allocates the required tables for the requested functions
BEARSMG1	BEARS / IMS Full Function Response routine stage 1
BEARSMG2	BEARS / IMS Full Function Response routine stage 2
BEARSMG3	BEARS / IMS Full Function Response routine stage 3
BEARSMG4	BEARS / IMS Full Function Response routine stage 4
BEARSMG5	BEARS / IMS Full Function Response routine stage 5
BEARSMIX	BEARS / IMS Mixed Mode Response routine
BEARSMSC	BEARS / IMS Message summarization routine
BEARSMXC	BEARS / IMS Mixed Mode Response summarization routine
BEARSPAR	BEARS / IMS control card parser. Uses SPZPARSE (see below)
BEARSREL	BEARS / IMS IMS release routine. Informs all routines of the current IMS release
BEARSRUC	BEARS / IMS Resource summarization routine
BEARSRUT	BEARS / IMS Resource routine. Monitors which IMS resources are defined and used
BEARSSCC	BEARS / IMS Security summarization routine
BEARSSEC	BEARS / IMS Security routine. Monitors which users are signed on and produces reports of all security violations
BEARSSIZ	BEARS / IMS table and region size estimate routine
BEARSSRT	BEARS / IMS incore sort routine
BEARSTRC	BEARS / IMS Combined Transaction Response summarization routine
BEARSTRN	BEARS / IMS Transaction to Application cross reference routine
BEARSTRX	BEARS / IMS Combined Transaction Response routine
BRUEXIT	BEARS / IMS Sample User Exit
SPGLBL	Span Software service routines global block
SPOUTPUT	Span Software output service routine
SPZPARSE	Span Software parse service routine
SPZQPAM	Span Software queued PDS access method

### 4.3 BEARS / IMS Macros

<u>Source Macro</u>	<u>Function</u>
#BEARSCB	Internal BEARS / IMS control block
#BEARSVR	Internal BEARS / IMS version literal
#BRS0REC	Internal BEARS / IMS dsect
#BRS01	Internal BEARS / IMS dsect
#BRS03	Internal BEARS / IMS dsect
#BRS06	Internal BEARS / IMS dsect
#BRS07	Internal BEARS / IMS dsect
#BRS08	Internal BEARS / IMS dsect
#BRS10	Internal BEARS / IMS dsect
#BRS16	Internal BEARS / IMS dsect
#BRS31	Internal BEARS / IMS dsect
#BRS35	Internal BEARS / IMS dsect
#BRS36	Internal BEARS / IMS dsect
#BRS37	Internal BEARS / IMS dsect
#BRS38	Internal BEARS / IMS dsect
#BRS40	Internal BEARS / IMS dsect
#BRS45	Internal BEARS / IMS dsect
#BRS5901	Internal BEARS / IMS dsect
#BRS5903	Internal BEARS / IMS dsect
#BRS5936	Internal BEARS / IMS dsect
#BRS5937	Internal BEARS / IMS dsect
#BRS5938	Internal BEARS / IMS dsect
#BRS5950	Internal BEARS / IMS dsect
#BRS5955	Internal BEARS / IMS dsect
#BRSABC	Internal BEARS / IMS dsect
#BRSABND	Internal BEARS / IMS dsect
#BRSAREA	Internal BEARS / IMS dsect
#BRSCFF	Internal BEARS / IMS dsect
#BRSCFP	Internal BEARS / IMS dsect
#BRSCMSG	Internal BEARS / IMS dsect
#BRSCMX	Internal BEARS / IMS dsect
#BRSCREC	Internal BEARS / IMS dsect
#BRSCTR	Internal BEARS / IMS dsect
#BRSEXIT	Generates Exit Point linkage
#BRSGEN	Generate BEARS / IMS installation JCL
#BRSGENA	Internal #BRSGEN macro
#BRSGENI	Internal #BRSGEN macro
#BRSGENJ	Internal #BRSGEN macro
#BRSBINR	Perform Binary Search of an ordered table
#BRSLTRG	Generates patch area
#BRSMFF	Internal BEARS / IMS dsect
#BRSMFP	Internal BEARS / IMS dsect

<u>Source Macro</u>	<u>Function</u>
#BRSM13	Internal BEARS / IMS dsect
#BRSMIX	Internal BEARS / IMS dsect
#BRSMMSG	Internal BEARS / IMS dsect
#BRSMODE	MVS/XA Addressing Mode macro
#BRSMTR	Internal BEARS / IMS dsect
#BRSNTRY	Generates Entry Point linkage
#BRSREGS	Register Equates
#BRSREL	BEARS / IMS and IMS Release macro
#BRSSMB	Internal BEARS / IMS dsect
#BRSTERM	Internal BEARS / IMS dsect
#BRSTODO	Internal BEARS / IMS dsect
#BRSVIC	Internal BEARS / IMS dsect
#BRSVIO	Internal BEARS / IMS dsect

This page intentionally left blank



## 5 BEARS / IMS Generation Process

### 5.1 Use of #BRSGEN Macro

With Release 03.6 of BEARS / IMS, the installation process is based upon a generation macro and a two-stage concept for installation of the product.

BEARS / IMS libraries must first be loaded from the BEARS / IMS Release CD, using the SMP/E process as shown in Section 3 of this manual.

Once this has been done, the next step is to prepare and run the assembly of the #BRSGEN macro. The #BRSGEN macro is coded with the parameters shown below, assembled by the normal system assembler, and produces customized JCL for the complete BEARS / IMS installation process. This JCL must then be executed to install BEARS.

Note. In some countries the '#' sign may be translated to another character. In this case please read all references to the '#' as whatever character the country in question uses.

Notation for the #BRSGEN Macro

Square brackets, [ ], denote (1) that a macro parameter is optional: if an entire parameter with its options is enclosed in square brackets, then that parameter is optional; (2) that a range of values is permissible for a given parameter: if a series of possible values for a parameter is shown in a vertical manner, all surrounded by additional square brackets, then choose one from the values shown.

Normal parentheses, ( ), signify that parentheses should appear when the macro is coded, denoting, for example, a list of sub-parameters.

Underlining,         , denotes default values for parameters.

Coding Conventions

Standard Assembler language coding conventions are used for all BEARS / IMS macros:

- Macro names and operands may be placed anywhere on the statement but are conventionally in columns 10 and 20 respectively for BEARS;
- Continuations are indicated by a non-blank character in column 72 of the continued statement;

Continuation statements must begin in column 16.



## 5.2 #BRSGEN Macro - Generate BEARS / IMS Installation JCL

The #BRSGEN macro should be used to specify all the BEARS / IMS installation options required, and all the environmental parameters (such as Job card formats) that apply to a user installation. Output from the #BRSGEN macro is executable JCL to perform Stage 2 of the BEARS / IMS generation process.

Format:

```
#BRSGEN BRSVOL=vvvvvvv

[ [A      ]
[,SOUT=[    ]
[ [sysoutclass]]

,DSNPREF=dsnameprefix

,IMSPREF=genlibprefix

,IMSSYS=imsid

[ [A      ]
,JOBNAME=jobprefix [,JOBCLAS=[    ]
[ [jobclass]]

[,JOBACCT=accountinfo] [,JOBOTHR=otherjobparms]

[ [IEV90]]
[,ASM=[    ]
[ [ASMA90]]

[ [SYSLIN]]
[,ASMDD=[    ]] [,MACBLK=nnnnn]
[ [ddname]]

[ [MIN      ]
[,ASSEM=[ALL      ]]
```

Figure 10 - #BRSGEN Macro Syntax

where:

- BRSVOL=** - valid volume serial number
- specifies the volume serial number of a DASD device that BEARS / IMS will allocate all its required datasets on. To ease the installation process all datasets will be allocated on the same device. For performance reasons we strongly recommend that the datasets are subsequently spread over as many spindles as are available.
- SOUT=** - valid SYSOUT Class
- specifies the SYSOUT Class to be used for the output from Stage 2 of the BEARS / IMS generation process. Default is SYSOUT=A.
- DSNPREF=** - library dataset name prefix for BEARS / IMS distribution and generation libraries
- specifies all levels of dataset name, except the low-order suffix, for all the BEARS / IMS distribution and generation libraries. The #BRSGEN macro will append some standard suffices to this prefix to form the fully-qualified dataset names that will be used throughout the Stage 2 generation process.

These suffices are:

**BRSSRCE** for the supplied BEARS / IMS distribution source library;

**BRSLOAD** for the supplied BEARS / IMS distribution load library;

**BRSOBJ** for the temporary object library used for assemblies of BEARS / IMS modules;

**BRSLKLIB** for the 'target' load library of the Stage 2 generation process, into which the completed BEARS / IMS load modules will be placed by the Linkage Editor during the execution of Stage 2.

Datasets using the appropriate fully-qualified names must be built and loaded before the JCL output from the #BRSGEN macro is executed.

- IMSPREF=** - library dataset name prefix for IMS GENLIB libraries
- specifies all levels of dataset name, except the low-order suffix, for all the IMS distribution and generation libraries.

- IMSSYS=** - IMS system id



- specifies the IMS system id of the IMS system that is to be monitored. The #BRSGEN macro will substitute this into the relevant JCL parameters for the BEARS / IMS jobs to run.
- JOBNAME=
- valid jobname prefix
  - specifies up to six characters that are to be the first characters of the jobnames generated for Stage 2 of the BEARS / IMS generation process. These characters will have one or two numeric digits placed after them to form each jobname generated by the #BRSGEN macro.
- JOBCLAS=
- valid JCL job class
  - specifies the job class to be used for jobs generated by the #BRSGEN macro. Default job class is 'A'.
- JOBACCT=
- valid job accounting information
  - specifies any accounting information that is to be included in the job statements generated by the #BRSGEN macro. If this parameter is omitted, no accounting information will be included in job cards. The operand for this parameter should be enclosed in parentheses or single quotes as required by the users JCL standards - these parentheses or quotes will be included in the accounting information parameter in the generated job cards.
- JOBOTHR=
- any other valid JOB card parameters
  - specifies any other parameters required for generated job statements. A maximum of 68 characters may be specified, and could include the MSGLEVEL, MSGCLASS, TYPRUN, NOTIFY, USER, PASSWORD parameters, etc.
- ASM=
- Assembler Program name
  - specifies the name of the Assembler you wish to use. The currently supported options are Assembler H IEV90, and the High Level Assembler ASMA90.
- ASMDD=
- assembler output DD name to be used
  - specifies the DDNAME used by the assembler for the object library - the default DDNAME is SYSLIN.
- MACBLK=
- MACLIB Blocksize
  - specifies the maximum blocksize of the system macro libraries to be used in the assembly steps generated for BEARS / IMS installation. The default value is 6160,

which is the supplied blocksize of the BEARS / IMS source library. If the installation's macro libraries SYS1.MACLIB or SYS1.AMODGEN have blocksize(s) greater than 6160, then the maximum blocksize used for a macro library must be specified via this parameter. The specified blocksize should be a multiple of 80.

- ASSEM=
- modules to assemble
  - specifies the BEARS / IMS modules that are to be assembled by the BEARS / IMS generation process. The default is 'ASSEM=MIN' which will assemble the minimum number of modules to ensure a correct BEARS / IMS installation. 'ASSEM=ALL' will assemble all the supplied BEARS / IMS source modules. Also a list of BEARS / IMS module names may be specified for which assemblies are required - these will be assembled in addition to the minimum required modules.



This page intentionally left blank



## 6 Appendix A

### 6.1 BEARS / IMS generated JCL members and Control cards

The following is a list of the JCL members and control cards that are generated by the BEARS / IMS stage 2 process. They are all placed in the BRSSRCE dataset.

- |         |   |
|---------|---|
| BRINST1 | This job will allocate and end-of-file all the datasets needed to run BEARS / IMS.  |
| BRINST2 | This is an example of the step to be added to IMS Archive to submit BEARS / IMS automatically.  |
| BRINST3 | This is an example of the parameter card (PARMCARD) required to run BEARS. Refer to the BEARS / IMS General Usage Manual for more details.                                  |
| BRINST4 | This is an example of the transaction to application cross reference card (TRANCARD) required to run BEARS. Refer to the BEARS / IMS General Usage Manual for more details. |
| BRINST5 | This is the standard BEARS / IMS job.   |
| BRINST6 | This is the SAS job to move the data from the daily BEARS / IMS files to the Weekly SAS databases. The data is summarised in the process.                                   |
| BRINST7 | This is the SAS job to move the data from the Weekly SAS databases to the Monthly SAS databases. The data is summarised in the process.                                     |
| BRINST8 | This is the SAS job to move the data from the Monthly SAS databases to the Yearly SAS databases. The data is summarised in the process.                                     |
| BRINST9 | This is the SAS job to label the variables in the SAS databases.  |

This page intentionally left blank.



This manual is published by

VIO Systems Limited

<http://www.viosystems.com>  
email [bears@viosystems.com](mailto:bears@viosystems.com)

BEARS/IMS is marketed by

Span Software Consultants Limited

<http://www.spansoftware.com>  
email [bears@spansoftware.com](mailto:bears@spansoftware.com)

