EPV for z/OS
Preparing Input for a Demo

IT Cost
Under Control

Supporting
EPV for z/OS version V11

November 2012

All the trademarks mentioned belong to their respective companies.

EPV Technologies contact details:

EPV Technologies
Viale Angelico, 54
00195 Roma
Tel. 06 86210880
Fax. 06 86387461
E-mail: epvtech@epvtech.com
WEB: http://www.epvtech.com

Contents

1 Introduction ..... - 5 -
2 Mandatory input data ..... - 6 -
2.1 SMF 30 subtype 2, 3 records ..... - 6 -
2.2 SMF 70 and 72 records ..... - 6 -
3 Other needed input data ..... - 7 -
4 Optional input data ..... - 8 -
5 Preparing data for a demo ..... - 10 -
5.1 Collect the data ..... - 10 -
5.2 Transfer the data to PC ..... - 14 -
5.3 Compress the data ..... - 21 -
5.4 Send the data ..... - 21 -
6 Customer support ..... - 22 -
Related documentation ..... - 23 -

About this manual

This manual is intended to help anyone who wants to provide the data needed to prepare an EPV for z/OS demo.

Changes

Technical changes or additions to the text are indicated by a vertical line to the left of the change.

1 Introduction

The best way to evaluate the benefits provided by EPV for z/OS for customers is to have a demo based on their data in their own environment.
Providing the data needed to prepare a demo is a quick and easy task to perform.
In this manual, after a short description of EPV for z/OS input data, a simple four step process to do that is presented.
Sample JCLs are also provided.

2 Mandatory input data

Some SMF records data are mandatory in order to run EPV for z/OS. If you don't provide them EPV will not produce any usable output.
They are:

- Record 30 subtype 2,3 (Address Space Interval activity);
- Record 70 and 72 (RMF CPU and Workload activity).

2.1 SMF 30 subtype 2, 3 records

SMF 30 subtype 2 and 3 records are not produced by default.

To activate SMF interval accounting using the global recording interval the following parameters have to be set in the SMFPRMxx member of the SYS1.PARMLIB library:

- INTVAL(mm) where mm is the interval duration; suggested values are 10 or 15 minutes;
- SYNCVAL(nn) where nn is the minute in the hour that starts the interval; suggested value is 00.

In addition the following parameter have to be set under SYS and SUBSYS sections:

- INTERVAL(SMF,SYNC).

Writing of these records has to be allowed in SMFPRMxx (under the TYPE sub parameter).

It's very important you synchronise SMF and RMF data; to do that you must set the following parameter in the ERBRMFxx member, used by RMF Monitor I, of your SYS1.PARMLIB library:

- SYNCH(SMF).

2.2 SMF 70 and 72 records

SMF 70 and 72 records are produced by default.
However the following parameters are normally explicitly specified in RMF monitor I (ERBRMFxx member of the SYS1.PARMLIB library):

- CPU, to produce CPU activity information;
- WKLD, to produce Workload activity information.

Writing of these records also has to be allowed in SMFPRMxx (under the TYPE sub parameter).

3 Other needed input data

Using only the mandatory data will result in only a small subset of the EPV for z/OS views and analysis. So you are strongly advised to also provide the following SMF records:

- Record 0 (IPL); produced by default;
- Record 30 subtype 5,6 (Address Space activity); produced by default;
- Record 71 (Paging activity); produced by default; PAGING is normally explicitly specified in RMF Monitor I options;
- Record 73, (Channel activity); produced by default; CHAN is normally explicitly specified in RMF Monitor I options;
- Record 74 subtype 1, (Device activity); produced by default; DEVICE(DASD) is normally explicitly specified in RMF Monitor I options;
- Record 74 subtype 5, (Cache activity); produced by default; CACHE is normally explicitly specified in RMF Monitor I options;
- Record 75, (Page dataset activity); produced by default; PAGESP is normally explicitly specified in RMF Monitor I options;
- Record 78 subtype 2, (Virtual Storage activity); produced by default; VSTOR is normally explicitly specified in RMF Monitor I options;
- Record 78 subtype 3, (I/O Queuing activity); produced by default; IOQ(DASD) is normally explicitly specified in RMF Monitor I options;
- Record 113 subtype 2, (hardware capacity, reporting, and statistics); produced only when activating Hardware Instrumentation Services (HIS) data collection.

Writing these records also has to be enabled in SMFPRMxx (under the TYPE sub parameter).

EPV supports disk volume space statistics, produced using the IBM utility IDCAMS with the DCOLLECT input option. This data needs to be gathered daily for all the systems.[1]
The production step is very quick, and the added value is great because EPV for z/OS will then provide views showing both available and used space, for all SMS groups.

[1] If a system can access all disks then its enough to gather DCOLLECT data from this system.

4 Optional input data

If you want EPV for z/OS to produce subsystems activity views the following SMF records have to be provided:

- Record 101, (DB2 Accounting);
- Record 110, (CICS activity);
- Record 116, (Websphere MQ activity);
- Record 120, (Websphere activity).

The default is that these records are not produced. You have to enable their production by the appropriate subsystem settings.
Writing of these records also has to be enabled in SMFPRMxx (under the TYPE sub parameter).

IMS activity information is not provided by any SMF records. The information is generally gathered through 3 different sources:

- 7 and 8 record types from the IMS LOG[2]; to use this option you have to create a separate dataset for each IMS and set the first and second qualifiers of the dataset name to system and IMS id (see examples in Chapter 6); you should also communicate the IMS level to EPV support;
- FA record type produced by Mainview for IMS by BMC Software;
- TAR record type produced by the utility IMFLEDIT by BMC Software, that correlates the 2 sources above.

This information are gathered by EPV for IMS DC only.

EPV supports all these sources, but we suggest the FA or TAR record type, because it is more accurate and complete.
Other optional SMF record types are:

- record 74 subtype 2 (XCF activity); produced by default by RMF Monitor III; no parameter has to be specified;
- record 74 subtype 4 (Coupling Facility activity); produced by default by RMF Monitor III; CFDETAIL is normally explicitly specified in RMF Monitor III options;
- record 74 subtype 8 (PPRC activity); not produced by default; NOESS is normally explicitly specified in RMF Monitor I options; ESS has to be set;
- record 74 subtype 7 (Ficon Director activity); not produced by default; NOFCD is normally explicitly specified in RMF Monitor I options; FCD has to be set.

[2] Reports based on Input Queue time are not provided using this option.

IBM VTS (also called Hydra) provides an internal facility called Bulk Volume Information Retrieval (BVIR) which also allows to get Historical Statistics by using a two step JCL running the standard IEBGENER utility. In alternative a SMF user record can be produced.

EPV supports both input types.

SUN STORAGETEK VTCS (also called VSM) produces a SMF user record with many subtypes. EPV supports 10, 11, 13, 14 and 20 subtype.

EPV also supports VSM tape volume statistics produced using the SWSADMIN utility with the QU MVCP NAME(ALL) option. These statistics must be gathered daily for all the systems.[3]

[3] If a system can access all ACS then its enough to gather statistics from this system.

5 Preparing data for a demo

To have a good demo, a few hours worth of data are enough. If you have more systems sharing resources the result will be better. If you had a bad day, with lot of problems, the EPV demo will probably help you understand what happened.
The following steps have to be performed in order to prepare input data for an EPV demo.

5.1 Collect the data

JCL1

The following JCL will select all the necessary SMF records. It will also select optional subsystems records for DB2, CICS, Websphere MQ and Websphere. The sss record type is the VTCS (VSM) user record. The iii record type is the VTS (Hydra) user record. If you are not interested in any of them simply remove the record type from the OUTDD line.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *smfinput* TO YOUR SMF INPUT FILE NAME
- CHANGE *smfpref* TO OUTPUT FILE PREFIX
- CHANGE *yyyyxxx* to the starting and ending Julian date you want to select
- CHANGE *hhmm* to the starting and ending hours you want to select

```
//SELSMF   EXEC PGM=IFASMFDP
//SYSPRINT DD SYSOUT=*
//INDD1    DD DSN=smfinput,DISP=SHR
//OUTDD1   DD DSN=smfpref.VBS,DISP=(,CATLG),
//         UNIT=SYSDA, SPACE=(CYL,(100,100),RLSE),
//         DCB=(LRECL=32760,BLKSIZE=27998,RECFM=VBS)
//SYSIN    DD *
 INDD(INDD1,OPTIONS(DUMP))
 OUTDD(OUTDD1,TYPE(0,30,70:78,101,110,113,116,120, sss, iii))
 DATE(yyyyxxx,yyyyxxx)
 START(hhmm)
 END(hhmm)
/*
```

JCL2

The following JCL will produce the necessary DCOLLECT information.
If a system can access all disks then it is sufficient to gather DCOLLECT data from this system.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *dcolpref* TO OUTPUT FILE PREFIX
- CHANGE *sysid* to the system id where you run the DCOLLECT job

```
//SELDCOL  EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//DCOUT    DD DSN=dcolpref.sysid,DISP=(,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(CYL,(10,1),RLSE),
//         DCB=(RECFM=VB,LRECL=4092,BLKSIZE=27998)
//SYSIN    DD    *
      DCOLLECT -
      OUTFILE(DCOUT) -
      VOLUMES( * ) -
      NODATAINFO
/*
```

JCL3 (option A)

The following JCL will select the 7 and 8 records from the IMS log. You have to run this job only if you are interested in IMS transaction information.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *reslib* TO YOUR IMS RESLIB FILE NAME
- CHANGE *sldsfile* TO SLDS FILE NAME
- CHANGE *imspref* TO OUTPUT FILE PREFIX
- CHANGE *imsid* to the IMS subsystem id
- CHANGE *sysid* to the system id where the IMS resides

```
//SELIMS   EXEC PGM=DFSERA10
//STEPLIB  DD DSN=reslib,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DSN=sldsfile,
//         DISP=(OLD,PASS),UNIT=TAPE
//SYSUT4   DD DSN=imspref.sysid.imsid,DISP=(,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(CYL,(100,100),RLSE),
//         DCB=(RECFM=VB,LRECL=30970,BLKSIZE=30974)
//SYSIN    DD *
CONTROL  CNTL STOPAFT=EOF
OPTION   COPY  OFFSET=5,FLDLEN=1,VALUE=07,COND=E
OPTION   COPY  OFFSET=5,FLDLEN=1,VALUE=08,COND=E
END
```

JCL3 (option B)

The following JCL will select the FA records produced by the Mainview for IMS BMC product. You have to run this job only if you are interested in IMS transaction information.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *reslib* TO YOUR IMS RESLIB FILE NAME
- CHANGE *sldsfile* TO SLDS FILE NAME
- CHANGE *imspref* TO OUTPUT FILE PREFIX

- CHANGE *imsid* to the IMS subsystem id
- CHANGE *sysid* to the system id where the IMS resides

```
//SELIMS   EXEC PGM=DFSERA10
//STEPLIB  DD DSN=reslib,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DSN=sldsfile,
//         DISP=(OLD,PASS),UNIT=TAPE
//SYSUT4   DD DSN=imspref.sysid.imsid,DISP=(,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(CYL,(100,100),RLSE),
//         DCB=(RECFM=VB,LRECL=30970,BLKSIZE=30974)
//SYSIN    DD *
CONTROL  CNTL STOPAFT=EOF
OPTION   COPY  OFFSET=5,FLDLEN=1,VALUE=FA,COND=E
END
```

JCL4

The following JCL will produce the necessary VSM pools information. You have to run this job only if you are interested in VSM information.

If a system can access all ACS tape libraries then it's enough to gather statistics from this system.
Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *libpref* TO YOUR VSM LIBRARY PREFIX
- CHANGE *vsmpref* TO OUTPUT FILE PREFIX
- CHANGE *sysid* to the system id where you run the VSM job

```
//SELVSM   EXEC PGM=SWSADMIN,PARM='MIXED'
//STEPLIB  DD DSN=libpref.SLSLINK,DISP=SHR
//MVCOUT   DD DUMMY
//SLSPRINT DD DSN=vsmpref.sysid,DISP=(,CATLG,DELETE),
//         UNIT=SYSDA,
//         SPACE=(TRK,(1,1),RLSE),
//         DCB=(LRECL=121,RECFM=FBA)
//SYSOUT   DD SYSOUT=*
//SLSIN DD *
  QU MVCP NAME(ALL)
/*
```

JCL5

The following JCL will produce VTS Historical Statistics. You have to run this job only if you are interested in VTS information.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *vtsunit* TO the esoteric name corresponding to the VTS
- CHANGE *vtspref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *vtsid* to the VTS id

- CHANGE *jjj* to the Julian dates you want to collect

```
//VTSCOMM  EXEC PGM=IEBGENER
//SYSPRINT DD  SYSOUT=*
//SYSIN    DD  DUMMY
//SYSUT2   DD  DSN=vtspref.COMMAND,
//         LABEL=(1,SL),DISP=(,CATLG),
//         UNIT=vtsunit,DCB=(RECFM=F,BLKSIZE=80,LRECL=80,TRTCH=NOCOMP)
//SYSUT1   DD  *
VTS BULK VOLUME DATA REQUEST
HISTORICAL STATISTICS FOR jjj-jjj
/*
//*
//SELVTS   EXEC PGM=IEBGENER
//SYSPRINT DD  SYSOUT=*
//SYSUT1   DD DSN= vtspref.COMMAND,DISP=SHR,
//         DCB=(RECFM=U,BLKSIZE=24000)
//SYSUT2   DD DSN= vtspref.vtsid.DATA,
//         LABEL=(1,SL),DISP=(,CATLG),
//         UNIT=vtsunit,DCB=(RECFM=U,BLKSIZE=24000)
//SYSIN    DD DUMMY
```

5.2 Transfer the data to PC

When transferring variable data (VB or VBS) from the mainframe to other platforms it is obviously important to do that without corrupting the logical structure of the records. The JCL to use differs slightly depending on the available File Transfer (FT) tools used.

JCL6 – (you need to use the standard IBM FTP)

The following JCLs will transfer the SMF files produced as described in chapter 5.1 using the standard IBM FTP.
Records are read as if in undefined format in order to avoid FTP eliminating the VB and VBS headers and so corrupting the records. As stated in the comments it is essential not to change the RECFM and BLKSIZE parameters. It's also required that the transfer is done in binary mode.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *smfpref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON //DDSMF
//FTPSMF   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//DDSMF    DD DSN=smfpref.VBS,RECFM=U,BLKSIZE=32760,DISP=SHR
//INPUT DD *
your.ftp.host.address
user password
quote PASV
bin
put //DD:DDSMF /smfpref.smf
close
quit
/*
```

JCL7 – (you need to use the standard IBM FTP)

The following JCLs will transfer the DCOLLECT files produced as described in chapter 5.1 using the standard IBM FTP.
Records are read as if in undefined format in order to avoid FTP eliminating the VB and VBS headers and so corrupting the records. As stated in the comments it is essential not to change the RECFM and BLKSIZE parameters. It's also required that the transfer is done in binary mode.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *dcolpref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *sysid* to the system id where you run the DCOLLECT job
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON //DDDCOL
//FTPDCOL  EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//DDDCOL   DD DSN=dcolpref.sysid,DCB=RECFM=U,BLKSIZE=32760,DISP=SHR
//INPUT DD *
your.ftp.host.address
user password
quote PASV
bin
put //DD:DDDCOL /dcolpref.sysid
close
quit
/*
```

JCL8 – (you need to use the standard IBM FTP)

The following JCLs will transfer the IMS files produced as described in chapter 5.1 using the standard IBM FTP.
Records are read as if in undefined format in order to avoid FTP eliminating the VB and VBS headers and so corrupting the records. As stated in the comments it is essential not to change the RECFM and BLKSIZE parameters. It's also required that the transfer is done in binary mode.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *imspref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *imsid* to the IMS subsystem id
- CHANGE *sysid* to the system id where the IMS resides
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON //DDIMF
//FTPIMS   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//DDIMS    DD DSN=imspref.sysid.imsid,DCB=RECFM=U,BLKSIZE=32760,DISP=SHR
//INPUT DD *
your.ftp.host.address
user password
quote PASV
bin
put //DD:DDIMS /sysid.imsid
close
quit
/*
```

JCL 9

The following JCLs will transfer the VSM sequential files produced as described in chapter 5.1 using the standard IBM FTP. *You only have to substitute JCL and commands in the second step to run any different FT tool.*

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *vsmpref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *sysid* to the system id where you run the VSM job
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//FTPVSM   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//INPUT    DD *
your.ftp.host.address
user password
quote PASV
ascii
put 'vsmpref.sysid'  vsmpref.sysid
close
quit
/*
```

JCL 10

The following JCLs will transfer the VTS files produced as described in chapter 5.1 using the standard IBM FTP. *You only have to substitute JCL and commands in the second step to run any different FT tool.*

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *vtspref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *vtsid* to the VTS id
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//FROMU2VB EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DSN=vtspref.vtsid.DATA,DISP=SHR
//SYSUT2   DD DSN=vtspref.vtsid.VB,DISP=(,CATLG),
//         DCB=(RECFM=VB,LRECL=32760,BLKSIZE=32760),
//         UNIT=SYSDA,SPACE=(CYL,(10,5),RLSE)
//SYSIN    DD DUMMY
/*
//FROMVB2U EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DSN=vtspref.vtsid.VB,DISP=SHR,
//         DCB=(RECFM=U,BLKSIZE=32760)
//SYSUT2   DD DSN=vtspref.vtsid.VBUND,DISP=(,CATLG),
//         DCB=(RECFM=U,BLKSIZE=32760),
//         UNIT=SYSDA,SPACE=(CYL,(10,5),RLSE)
//SYSIN    DD DUMMY
/*
//FTPVTS   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD  SYSOUT=*
//OUTPUT   DD  SYSOUT=*
//INPUT    DD *
your.ftp.host.address
```

```
user password
quote PASV
bin
put 'vtspref.vtsid.VBUND' /vtspref.vtsid
close
quit
/*
```

JCL6.1 – (alternative to JCL6 if you don't use the standard IBM FTP)

The following JCL will transfer the SMF files produced as described in chapter 5.1, using tools other than the standard IBM FTP.
A first preliminary IEBGENER step changes the DSCB from VB and VBS to undefined format. A second step transfers the data. It's required that the transfer is done in binary mode.
In the following examples we refer to IBM FTP, you have to substitute JCL and commands in the second step with the ones required to run your FT tool.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *smfpref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON BOTH DD
//UNDSMF EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DSN=smfpref.VBS,DISP=MOD,
//         DCB=RECFM=U
//SYSUT1   DD DSN=NULLFILE,DCB=*.SYSUT2
//SYSIN    DD DUMMY
/*
//FTPSMF   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//INPUT DD *
your.ftp.host.address
user password
quote PASV
bin
put 'smfpref.VBS' /smfpref.smf
close
quit
/*
```

JCL7.1 – (alternative to JCL7 if you don't use the standard IBM FTP)

The following JCL will transfer the DCOLLECT files produced as described in chapter 5.1, using tools other than the standard IBM FTP.
A first preliminary IEBGENER step changes the DSCB from VB and VBS to undefined format. A second step transfers the data. It's required that the transfer is done in binary mode.
In the following examples we refer to IBM FTP, you have to substitute JCL and commands in the second step with the ones required to run your FT tool.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *dcolpref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *sysid* to the system id where you run the DCOLLECT job
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON BOTH DD
//UNDDCOL  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DSN=dcolpref.sysid,DISP=MOD,
//         DCB=RECFM=U
//SYSUT1   DD DSN=NULLFILE,DCB=*.SYSUT2
//SYSIN    DD DUMMY
/*
//FTPDCOL  EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//INPUT DD *
your.ftp.host.address
user password
quote PASV
bin
put dcolpref.sysid /dcolpref.sysid
close
quit
/*
```

JCL8.1 – (alternative to JCL8 if you don't use the standard IBM FTP)

The following JCL will transfer the IMS files produced as described in chapter 5.1, using tools other than the standard IBM FTP.
A first preliminary IEBGENER step changes the DSCB from VB and VBS to undefined format. A second step transfers the data. It's required that the transfer is done in binary mode.
In the following examples we refer to IBM FTP, you have to substitute JCL and commands in the second step with the ones required to run your FT tool.

Cut and paste it in your JCL library, and do the following customizations:

- CHANGE *imspref* TO INPUT AND OUTPUT FILE PREFIX
- CHANGE *imsid* to the IMS subsystem id
- CHANGE *sysid* to the system id where the IMS resides
- CHANGE FTP parameters (*your.ftp.host.address, user and password*) to appropriate values

```
//* DO NOT CHANGE RECFM=U NOR BLKSIZE=32760 ON BOTH DD
//UNDIMS   EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT2   DD DSN=imspref.sysid.imsid,DISP=MOD,
//         DCB=RECFM=U
//SYSUT1   DD DSN=NULLFILE,DCB=*.SYSUT2
//SYSIN    DD DUMMY
/*
//FTPIMS   EXEC PGM=FTP,PARM='(EXIT'
//SYSPRINT DD SYSOUT=*
//OUTPUT   DD SYSOUT=*
//INPUT DD *
your.ftp.host.address
user password
```

```
quote PASV
bin
put ‘imspref.sysid.imsid’ /sysid.imsid
close
quit
/*
```

5.3 Compress the data

When the data is on PC you should compress it (the compression factor is usually very high).
Please be aware that compression tools may have limitations on the size of the file they can compress.

5.4 Send the data

You can send data to EPV Technologies in two main ways:

- Uploading the data to the EPV FTP server;
- Creating a CD/DVD and sending it to our local distributor or directly to EPV Technologies via a courier service.

It's always better before to send a small file with only one SMF record type by FTP or e-Mail, so we can confirm everything is correct before spending time sending large amounts of data.

6 Customer support

For any technical problems or questions about EPV for z/OS please email:

epv.support@epvtech.com

For any other issue about EPV for z/OS please email:

epv.info@epvtech.com

Related documentation

The following manuals complement the information provided in this manual:

- *EPV for z/OS V11 Installation and Customization Guide*
- *EPV for z/OS V11 Database Layout*
- *EPV for z/OS V11 Release Notes*
- *EPV for z/OS V11 List of Views*
- *EPV for z/OS V11 Getting Started*
- *EPV V11 User Interface*