

MAX Software, Inc.

Product Installation Guide V3.3.0



MAX SOFTWARE, INC.

Contact Information

Corporate Headquarters

MAX Software, Inc.
3609 S. Wadsworth Blvd.,
Suite 500
Denver, CO 80235

Telephone: (303) 985-1558
Fax: (303) 985-1556

www.maxsoftware.com

Technical Support

Toll-Free
USA Only: (888) 376-6629

International
+1 (303) 985-1558

Web Site
www.maxsoftware.com/support.html

Disclaimer

Disclaimer of Warranties and Limitation of Liabilities

The staff of MAX Software has taken due care in preparing this manual; however, nothing contained herein modifies or alters in any way the standard terms and conditions of the MAX Software, Inc. standard software evaluation agreement; purchase agreement; lease agreement; or rental agreement by which this software was acquired; nor increases in any way MAX Software's liability to the customer. In no event shall MAX Software be liable for incidental or consequential damages in connection with or arising from the use of this manual or any program contained herein.

Release 3.3.0 (March 2004)

Copyright (c) MAX Software, Inc. 1993 - 2004.

All Rights Reserved Licensed Material Unauthorized Duplication Prohibited. This manual contains confidential material protected by license agreements.

MVS, DB2, IMS, REXX, TSO/E, and ISPF are software products of International Business Machines Corporation.

MAX MVS/UTIL (a complete set of data file manipulation tools with the following 3 components: MAX DATA/UTIL, MAX/PDF and MAX/BATCH); MAX IMS/UTIL (a complete set of IMS database manipulation tools with the following 2 components: MAX/IMS ONLINE and MAX/IMS BATCH); MAX DB2/UTIL (a complete set of DB2 database manipulation tools); and MAX/REXX (an interface between REXX and VSAM, SAM, PDS and DB2 data) are trademarks of MAX SOFTWARE, INC.

TABLE OF CONTENTS

MAX Software, Inc.	i
Contact Information	i
Corporate Headquarters	i
Technical Support	i
Toll-Free	i
International	i
Web Site	i
Disclaimer	ii
Disclaimer of Warranties and Limitation of Liabilities	ii
Release 3.3.0 (March 2004)	ii
Table of Contents	iii
Revisions	1
Release 3.3.0: March 2004	1
Release 3.2.1: August 2003	1
Release 3.2.0: March 2003	1
Release 3.1.0: June 2002	1
Release 2.5.0, Level R4: November 2001	1
Release 2.5.0: July 2001	1
Release 2.4.0: September 2000	1
Release 1.5.4: June 1999	1
Release 1.5.3: January 1999	1
Summary	3
Requirements	3
Installation Requirements	3
Software Requirements	3
Storage Requirements	3
Version Numbering	3
Installation Summary	4
Detailed Installation Instructions	7
Step 1: Select installation media	7
Step 1 Tape: Tape install	7
Step 1 FTP: Obtain MAXRECV exec and zip install file	7
Step 2: Create the MAX libraries from the installation media	8
Step 2 Tape: Unload the MAX installation data sets from tape	8
Step 2 FTP: RECEIVE the MAX installation data sets	9
Step 3: Link edit the MAX load modules	11
Step 4: Link edit the optional MAX load modules	12
Step 4.1: Link edit the MAX IMS/UTIL load modules	12
Step 4.2: Link edit the MAX/COD load modules	12
Step 5: Ensure MAX's IRXFUSER module loaded by TSO	13

Step 6: Install MAX Data/Util and optionally MAX/PDF or MAX IMS/UTIL into TSO/ISPF	14
Step 6.1: Modify 'MAX' Startup REXX Procedure.	14
Step 6.2: Allocate the MAX data sets	15
Step 6.3: Make the MAX load modules accessible	15
Step 6.4: Allocate IMS RESLIB data set.	16
Step 6.5: Add MAX Data/Util, MAX/PDF, MAX IMS/UTIL and MAX DB2/UTIL into the ISPF Main Menu.	17
Step 6.6: Modify MAX/PDF "OPTIONS PROFILE" to set default user options	18
Step 6.7: Copy 'MAX' startup REXX procedure to SYSEXEC/SYSPROC data set.	18
Step 7: Execute MAX.	19
Step 7.1: Establish IMS Profile Parameters.	20
Step 8: Review/Update the installed .JCL library	20
Step 8.1: Check the README member for samples and special notes.	20
Step 8.2: Update MAXFT* members in the .JCL library	20
Step 8.3: Update CBMAP member in the .JCL library	20
Step 9: Complete MAX DB2/UTIL and MAX/REXX SQL feature installation	21
Step 9.1: Bind MAX DB2/UTIL and MAX/REXX SQL feature to DB2	21
Step 9.2: Grant access to DB2 catalog tables	22
Step 9.3: Verify WLM support for DB2 external functions	22
Step 9.4: Create TEMP database if none exists	22
Step 9.5: Create MAX DB2/UTIL sample tables	23
Step 10: Complete MAX IMS/UTIL installation for using Dynamic PSBs	24
Step 10.1: Dynamic PSBs	24
Step 10.2: Modify the MAXIOPTS installation options module for dynamic PSB parameters	25
Step 10.3: Modifications required when using Dynamic PSBs with BMP run mode	27
Step 11: Define Relationship Definition File if applicable (MAX DB2/UTIL)	28
Appendix A. Product Authorization	29
Production Password	29
Trial Password	30
Appendix B. Installation Verification	31
Reader Comment Form	33

REVISIONS

Release 3.3.0: March 2004

Release 3.2.1: August 2003

Release 3.2.0: March 2003

- Product Authorization completely revamped. AUTHxxx members replaced with ASMAUTHS and included member MAXAUTH.

Release 3.1.0: June 2002

- New products added to the release: MAX DB2/UTIL.
- Refer to the member README in the JCL library for more on new members.

Release 2.5.0, Level R4: November 2001

- Refer to the member README in the JCL library for more on new members.
- MXRXLINK member in the JCL library has been changed and now requires the MAXILEXT program object module.
- IMS RESLIB allocation can now be done using dataset names specified in the MAX IMS/UTIL PROFILE panel and does not require allocation to STEPLIB or an exit.

Release 2.5.0: July 2001

Release 2.4.0: September 2000

- New products added to the release: MAX IMS/UTIL.

Release 1.5.4: June 1999

Release 1.5.3: January 1999

SUMMARY

Requirements

Installation Requirements

The installation summary listed on the next page has been included for users familiar with software install procedures.

Installation for MAX/REXX, MAX MVS/UTIL, MAX IMS/UTIL, and MAX DB2/UTIL products is identical, except where noted.

Following the summary is a detailed description of each step that can be referenced for additional information when required.

Software Requirements

MAX/REXX and MAX MVS/UTIL operate under OS/390 and z/OS operating systems. Required software includes:

- TSO/E Version 2.3 or above.
- ISPF/PDF Version 4.1 or above.
- DB/2 Version 6.1 or above.

MAX IMS/UTIL operates under OS/390 and z/OS operating systems. Required software includes:

- TSO/E Version 2.3 or above.
- ISPF/PDF Version 4.1 or above.
- IMS Version 5.1 or above.

MAX DB2/UTIL operates under OS/390 and z/OS operating systems. Required software includes:

- TSO/E Version 2.3 or above.
- ISPF/PDF Version 4.1 or above.
- DB2 Version 6.1 or above.

Storage Requirements

Space equivalent to approximately 44 cylinders of storage on a 3380 device will be required for any MAX product configuration of MAX/REXX, MAX MVS/UTIL and MAX IMS/UTIL.

Version Numbering

Throughout this installation guide, you will encounter data set names containing the characters 'nnn'. 'nnn' is the number of the current MAX install level and should be set to the install level number supplied to you.

Installation Summary

- Step 1:** Select installation media.
Step 1 Tape: Tape install.
Step 1 FTP: Obtain MAXRECV exec and zip install file.
- Step 2:** Create the MAX libraries from the installation media.
Step 2 Tape: Unload the MAX installation data sets from tape.
Step 2 FTP: RECEIVE the MAX installation data sets.
- Step 3:** Link edit the MAX load modules.
- Step 4:** Link edit the optional MAX load modules
Step 4.1: Link edit the MAX IMS/UTIL load modules.
Step 4.2: Link edit the MAX/COD load modules.
- Step 5:** Ensure MAX's IRXFUSER module loaded by TSO.
- Step 6:** Install MAX Data/Util and optionally MAX/PDF or MAX IMS/UTIL into TSO/ISPF.
Step 6.1: Modify 'MAX' Startup REXX Procedure.
Step 6.2: Allocate the MAX data sets.
Step 6.3: Make the MAX load modules accessible.
Step 6.4: Allocate IMS RESLIB data set.
Step 6.5: Add MAX Data/Util, MAX/PDF, MAX IMS/UTIL and MAX DB2/UTIL into the ISPF Main Menu.
Step 6.6: Modify MAX/PDF "OPTIONS PROFILE" to set default user options.
Step 6.7: Copy 'MAX' startup REXX procedure to SYSEXEC/SYSPROC data set.
- Step 7:** Execute MAX.
Step 7.1: Establish IMS Profile Parameters.
- Step 8:** Review/Update the installed .JCL library.
Step 8.1: Check the README member for samples and special notes.
Step 8.2: Update MAXFT* members in the .JCL library.
Step 8.3: Update CBMAP member in the .JCL library.
- Step 9:** Complete MAX DB2/UTIL and MAX/REXX SQL feature installation.
Step 9.1: Bind MAX DB2/UTIL and MAX/REXX SQL feature to DB2.
Step 9.2: Grant access to DB2 catalog tables.
Step 9.3: Verify WLM support for DB2 external functions.
Step 9.4: Create TEMP database if none exists.
Step 9.5: Create MAX DB2/UTIL sample tables.

- Step 10:** Complete MAX IMS/UTIL installation for using Dynamic PSBs.
- Step 10.1:** Dynamic PSBs.
 - Step 10.2:** Modify the MAXIOPTS installation options module for dynamic PSB parameters.
 - Step 10.3:** Modifications required when using Dynamic PSBs with BMP run mode.
- Step 11:** Define Relationship Definition File if applicable (MAX DB2/UTIL).

DETAILED INSTALLATION INSTRUCTIONS

Step 1: Select installation media

MAX products may be installed from a 3480 tape cartridge or you may download from our FTP site. Perform “Step 1 Tape” for TAPE installs or “Step 1 FTP” for FTP installs.

Step 1 Tape: Tape install

File 1 on the installation tape is a job to copy files 2 through 7 to disk.

Using JCL similar to that shown below, copy the first file from the Installation tape. In the example below, the file has been copied to a member named MXLOAD.

```
//USERNDX      JOB(MAX), 'COPY FILE 1',
//             CLASS=x,MSGCLASS=x
//S1           EXEC PGM=IEBGENER
//SYSPRINT     DD DUMMY
//SYSIN        DD DUMMY
//SYSUT1       DD DISP=OLD,UNIT=TAPE,VOL=SER=MAXnnn,
//             DSN=MXS.UNLOAD
//SYSUT2       DD  DSN=xxxxxxx.xxxxxx.xxxxxx(MXLOAD),DISP=OLD
```

Note: In the volser of MAXnnn, the ‘nnn’ is equal to the release level of the product. This number will be supplied with the tape.

Upon completion of the above job, JCL required to copy the remaining files from the installation tape will reside in the member MXLOAD of the DSN you select for SYSUT2.

Step 1 FTP: Obtain MAXRECV exec and zip install file

You can obtain the MAXRECV exec, MXRXVnnn zip files, and product documentation via email or download from the MAX Software FTP site at <ftp://download.maxsoftware.com>.

Access to the ‘user’ directory on our FTP site requires a username and password. We will provide this logon information for existing customers, and trial customers upon receipt of the signed trial agreement.

The MAXRECV is a Rexx exec used to define and receive the installation datasets. It should be placed in a library in the SYSEXEC concatenation of your TSO logon proc.

Step 2: Create the MAX libraries from the installation media

Based on installation media, perform either “Step 2 Tape” or “Step 2 FTP”.

Step 2 Tape: Unload the MAX installation data sets from tape

Edit the member MXLOAD, which was copied from tape in [Step 1: Select installation media](#). Make the following changes to the PROC defaults as listed below.

MXQUAL	Set this equal to the high-level qualifier for all the data sets that will be allocated by this job.
MXDA	Set this equal to the DASD unit name, e.g. SYSALLDA.
MXVOL	Set this equal to the VOLSER where all the data sets allocated in this job will be placed.
MXTVOL	Set this equal to the VOLSER for the MAX installation tape. The VOLSER on the tape is MAXnnn, where ‘nnn’ is equal to the release level of the products.
MXTAPE	Set this equal to the unit name of the device reading the tape, e.g. 3490, or TAPE.
MXBLKSZ	Set this equal to the BLOCKSIZE to be used for all the created data sets. This value must be a multiple of 80. If these data sets are going to be concatenated with other ISPF libraries, the BLOCKSIZE chosen should be the same as those libraries.
MXCL	Set this equal to the SYSOUT class for all output.

After the appropriate changes have been made, submit this job.

Note: Expect a return code of zero (0) from each step of this job.

There will be six partitioned data sets created by this job and a load library created by MXRXLINK for a total of 7 PDS data sets .

The total space required for these seven data sets is approximately forty-four (44) 3380 cylinders.

Step 2 FTP: RECEIVE the MAX installation data sets

Edit the exec MAXRECV, and make the following changes to the PROC defaults as listed below.

- MXQUAL Set this equal to the high-level qualifier for all the data sets that will be allocated by this job.
- MXDA Set this equal to the DASD unit name, e.g. SYSALLDA.
- MXVOL Set this equal to the VOLSER where all the data sets allocated in this job will be placed.

This installation process requires one temporary PDS of approximately twenty-six (26) 3380 cylinders.

There will be seven (7) partitioned data sets created by the normal installation process. Six of these are downloaded, and the other is a load library created by the MXRXLINK job.

The total space required for the six data sets, plus the loadlib created by MXRXLINK is approximately forty-four (44) 3380 cylinders.

After the appropriate changes have been made, execute the MAXRECV program.

The first execution of the MAXRECV exec creates the .MXRXVnnn.RECV temporary work file. This interim file is temporary storage from which the production PDS files are built.

Step 2.1 FTP:Transfer files into the RECV data set.

Transfer each of the six (6) members unzipped from [Step 1: Select installation media](#) into the .MXRXVnnn.RECV PDS using Binary transfer. Each of these will create a PDS in the next step. You may perform this transfer using an available FTP or IND\$FILE solution, often provided with 3270 emulators.

Step 2.2 FTP:Execute the MAXRECV exec - second execution for load

The second execution of the MAXRECV exec will create a PDS file from each of the six members in the .MXRXVnnn.RECV file. Once this step is complete, you are done with the .RECV file and may delete it.

The following list gives the name of each data set and its description. All data sets will have a high-level qualifier equal to the value given to MXQUAL in [Step 2: Create the MAX libraries from the installation media](#).

Data Set	Description
.MXXVnnn.JCL	The JCL used to install MAX products and apply authorization codes.
.MXXVnnn.OBJECT	The object modules used to create the load modules.
.MXXVnnn.EXECS	Sample REXX programs to show how to invoke the MAX products.
.MXXVnnn.PANELS	ISPF panels which are used by the MAX product.
.MXXVnnn.MESSAGES	ISPF Messages which are used by the MAX products
.MXXVnnn.TABLES	ISPF tables which are used by the MAX products.
.MXXVnnn.LOADLIB	MAX load module library. Created by the job MXXLINK, as explained in Step 3: Link edit the MAX load modules .
.MXXVnnn.LOADAPF	MAX load module library for APF Authorized modules. Created by the optional job MXXLNKC, as explained in STEP 4.2. This load library is used ONLY if you are installing the MAX/COD (Rexx Compiler on Demand) product and is not a part of the standard installation process.

Note: This job will delete and reallocate these nine data sets if they already exist. If corresponding data sets exist for another version of the MAX products, they will not be deleted.

The first six (6) data sets are created by the MAXLOAD job. The load libraries are created by MXXLINK and MXXLNKC (MXXLNKC optional - MAX/COD only).

Step 3: Link edit the MAX load modules

Edit the member MXRXLINK in the MXRXVnnn.JCL data set created above. Make the appropriate changes to the PROC defaults listed below.

- MXQUAL** Set this equal to the high-level qualifier for the eight partitioned data sets created in [Step 2: Create the MAX libraries from the installation media](#). Use the same value that was specified for MXQUAL in [Step 2: Create the MAX libraries from the installation media](#).
- MXCL** Set this equal to the SYSOUT class desired for all output.
- ISPPRFX** Set this equal to the high level qualifier of the ISPF data set containing load module ISPLINK.

Submit the MXRXLINK job to create the MAX load modules. Expect a return code of four (4) from the LKED1 step and zero (0) from the other six steps of this job.

This job links the MAX modules from the OBJECT data set into load modules used by MAX. It also applies any Program Temporary Fixes (PTFs) and applies the encoded expiration zaps if needed. This job has the following steps:

Step	Description
SCRATCH	Delete MAX load module data set (if it already exists).
DEFINE	Defines the MAX load module data set.
LKED1	Links each of the MAX products CSECTs into a temporary data set.
ZAP1	Applies any PTFs to the temporary data set created by LKED1.
LKED2	Links the modules in the temporary data set into the MXLOAD data set.
LKED3	Links module IRXFUSER, the REXX function package table.

Note: Once the Link is done, you may apply either regular or trial zaps (product codes). See [“Appendix A: Product Authorization”](#) on page 29 for details.

Step 4: Link edit the optional MAX load modules

Step 4.1: Link edit the MAX IMS/UTIL load modules (optional)

Perform this step only if installing MAX IMS/UTIL.

Submit MXRXLNKI

Edit member MXRXLNKI in the .MXRXVnnn.JCL data set created previously, making the appropriate changes to the PROC defaults listed below.

- MXQUAL** Set this equal to the high-level qualifier for all the data sets except MXLOAD. Use the same value that was specified for MXQUAL in [Step 2: Create the MAX libraries from the installation media](#).
- RESLIB** Set this to the value of the data set containing your IMS system modules: ASMTDLI and AIBTDLI. The default set in the PROC is IMS710.RESLIB.

Submit the MXRXLNKI job to link the additional modules. Expect a return code of four (4) from the LKED1 step and a return code of zero (0) from steps ZAP1 and LKED2.

This job links these additional modules into .MXRXVnnn.LOADLIB

Step 4.2: Link edit the MAX/COD load modules (optional)

Perform this step only if installing MAX/COD.

Edit the member MXRXLNKC in the MXRXVnnn.JCL data set created above. Make the appropriate changes to the PROC defaults listed below.

- MXQUAL** Set this equal to the high-level qualifier for the eight partitioned data sets created in [Step 2: Create the MAX libraries from the installation media](#). Use the same value that was specified for MXQUAL in [Step 2: Create the MAX libraries from the installation media](#).
- MXCL** Set this equal to the SYSOUT class desired for all output.
- ISPPRFX** Set this equal to the high level qualifier of the ISPF data set containing load module ISPLINK.

Submit the MXRXLNKC job to create the MAX APF authorized load modules. Expect a return code of four (4) from the LKED1 step and zero (0) from the other six steps of this job.

This job links these additional modules into .MXRXVnnn.LOADAPF. Add this load library to your PROGxx parmlib member as an APF Authorized library, or copy these load modules into an existing APF authorized library.

Step 5: Ensure MAX's IRXFUSER module loaded by TSO

Caution: The MAX IRXFUSER module must be placed in the load search sequence before the IBM module with the same name. Severe performance degradation can result with the MAX products if this does not occur.

MAX IRXFUSER Module

A load module, IRXFUSER, is created during this install of the MAX products.

This module replaces an IBM default supplied module with the same name. This module is loaded by TSO when a TSO address space is created (at TSO logon). Because of this, any change to this module (such as rerunning MXRXLINK after maintenance) will require the user to LOG OFF and LOG ON to TSO to retrieve the updated version.

The IBM version of IRXFUSER is usually found in SYS1.LINKLIB. SYS1.LINKLIB is searched before libraries in the LINKLST. Adding the MAX loadlib to LINKLST by itself is NOT sufficient to ensure the MAX IRXFUSER is found before IBM.

The MAX IRXFUSER module must be placed in the load search sequence before the IBM module with the same name. Ensure that, whichever method was chosen to access the MAX load library, the MAX module IRXFUSER will be found before the IBM module with the same name.

When permanently installing the products for production use, we recommend that the new load module IRXFUSER be placed in LPA. To add to the dynamic LPA, you can use the command:

```
SETPROG LPA,ADD,MODNAME=IRXFUSER,DSNAME=hlq.MXRXVNNN.LOADLIB
```

If you currently have an older IRXFUSER in the dynamic LPA, you may delete it with the command:

```
SETPROG LPA,DELETE,MODNAME=IRXFUSER,FORCE=YES
```

In the event you have an older version in the LPA that was loaded at IPL, the SETPROG LPA, DELETE function cannot remove it, however you may still add the new version to the dynamic LPA with the SETPROG LPA,ADD command and the new dynamic version will be found first.

Step 6: Install MAX Data/Util and optionally MAX/PDF or MAX IMS/UTIL into TSO/ISPF

This is a seven part procedure (6.1 - 6.7).

Caution: Care should be taken to assure adequate recovery since a mistake in this process could prevent LOGON to TSO/ISPF. One approach would be to create an alternate TSO logon PROC by copying and renaming the existing one prior to modifying it to use MAX products.

Step 6.1: Modify 'MAX' Startup REXX Procedure

See member 'MAX' in the .MXRXVnnn.EXECS data set to invoke MAX components. All allocations can be handled by the REXX program except the STEPLIB. Edit member 'MAX' in the .MXRXVnnn.EXECS data set. Make changes to the variable assignments as listed below.

Variable name	Value	Description
MXQUAL	'max'	Change high level qualifier. Change this to activate LIBDEF usage. Do not change if data sets are to be allocated in TSO logon procedure (see Step 6.2: Allocate the MAX data sets).
MAXVER	'MXRXVnnn'	Change middle qualifier.
MAXCDSN	' '	Change initial MAX/PDF DSNL data set. The DSN assigned to 'MAXCDSN' is where MAX/PDF stores DSNLs. The default will be the user's ISPF profile data set. You may wish to override this and save the DSNLs in another PDS. This value is optional and may also be changed from within DSNL using the CREATE or PROFILE commands. To allow for sharing of DSNLs across an entire installation: MAXCDSN='your.dsnl.pds'.
MAXIDSN	' '	Change to installation EXEC data set. The DSN assigned to 'MAXIDSN' allows MAX/PDF to access members stored in the installation EXECS data set; e.g., MAXIDSN=MXQUAL'.MXRXVNNN'.EXECS' will set the value to the installation execs lib (refer to variable MXQUAL above).
DFLTMEM	' '	When an installation wants each user to have their own unique DSNL data set (MAXCDSN), there may be a common DSNL entry that you want placed in these user data sets at initialization time. The DFLTMEM variable is used to provide the name of a member to be copied from the installation table to the user's DSNL data set each time the product is initialized.

Step 6.2: Allocate the MAX data sets

You may skip this step if you choose to use the LIBDEFs to allocate the MAX data sets in 'MAX', as explained in [Step 6.1: Modify 'MAX' Startup REXX Procedure](#). Otherwise, you must allocate the MAX data sets by modifying your TSO/ISPF logon PROC. Concatenate the MAX data sets to appropriate DDnames in the logon PROC as shown below.

DDname	Data Set name	Status
SYSEXEC	.MXRXVnnn . EXECES	Required.
ISPPLIB	.MXRXVnnn . PANELS	Required.
ISPMLIB	.MXRXVnnn . MESSAGES	Required.
ISPSLIB	.MXRXVnnn . JCL	Required.
ISPTLIB	.MXRXVnnn . TABLES	Required.
ISPLLIB and STEPLIB	.MXRXVnnn . LOADLIB	Required.

Note: MXRXVnnn.PANELS must be the first data set in the ISPPLIB concatenation.

Step 6.3: Make the MAX load modules accessible

Concatenate the MAX Loadlib to the STEPLIB DDname in the TSO Logon PROC as shown below.

DDname	Data Set name	Status
STEPLIB	MXRXVnnn . LOADLIB	Required.

Access to MAX load modules .MXRXVnnn LOADLIB must be possible via an 'ISPF SELECT PGM'. Use either of the following methods to perform this:

- Allocate to ISPLLIB with a LIBDEF in an EXEC.
- Allocate to ISPLLIB by modifying the TSO logon PROC.

In addition, accessing the MAX load modules must be possible via an MVS LOAD. Use any of the following methods to accomplish this:

- Allocate to STEPLIB in the TSO logon JCL.
- Put the load modules in the LINKLIST. Remember to do an LLA refresh.
- Put the load modules in LPA. All MAX load modules are re-entrant. This may require an IPL with CLPA after the MAX installation.

Note: When adding libraries to Linklist or LPA, or performing an LLA refresh, the following command examples may be helpful.

- Adding the MAX Loadlib to your LNKLST:

```
SETPROG LNKLST,DEFINE,NAME=newlst,COPYFROM=CURRENT
SETPROG LNKLST,ADD,NAME=newlst,DSNAME=hlq.MXRXXVnnn.LOADLIB,
  VOLUME=volid
SETPROG LNKLST,ACTIVATE,NAME=newlst
```

- Refresh after updating a module in the current linklist:

```
F LLA,REFRESH
```

Please review the System Commands manual for full syntax and parameter description for these commands. There are many additional parameters available for some of these commands. These are basic syntax guidelines only.

Step 6.4: Allocate IMS RESLIB data set

Perform this step only if installing MAX IMS/UTIL.

The IMS RESLIB data set containing your IMS system modules must be allocated to run MAX IMS/UTIL.

DDname	Data Set name	Status
STEPLIB	RESLIB data set containing your IMS system modules (unauthorized).	Optional.
DFSRESLB	RESLIB data set containing your IMS system modules (authorized).	Required to run DLI mode.

Use any of the following methods to provide access to IMS system modules (i.e. DFSRRC00):

- Specify the IMS RESLIB data set(s) to be allocated as TASKLIB data sets using the Profile Options screen. *This is the preferred method and may be done following [Step 7: Execute MAX](#).* Select Option 0 and enter the RESLIB data sets according to the IMSID to be accessed. See Option 0 - PROFILE PARAMETERS in the MAX IMS/UTIL Online manual for information on this.
- Concatenate the IMS RESLIB data set to the STEPLIB DDname in the TSO Logon PROC.
- Put the IMS RESLIB in the LINKLIST. Remember to do an LLA refresh.
- Put the IMS RESLIB data set in LPA. This may require an IPL with CLPA after the MAX installation.
- Allocate the IMS RESLIB using LIBDEF in the DFSRRC00 processing exit. This permits assignment based upon some special criteria. See sample exit MAXIX002 in .MXRXXVnnn.EXECS. Appendix B in the MAX IMS/UTIL Online manual has more information on this exit.

If running DLI mode, the DFSRESLB DDname must be allocated to the IMS RESLIB. Use any of the following methods to accomplish this:

- a. If the IMS RESLIB data set(s) are specified using the Profile Options screen (see step a) above), they will also be automatically allocated to the DFSRESLB ddname.
- b. Allocate the IMS RESLIB data set to DFSRESLB in TSO logon JCL.
- c. Allocate the IMS RESLIB to DFSRESLB in the DFSRRC00 processing exit. This permits assignment based upon some special criteria. See sample exit MAXIX002 in .MXRXVnnn.EXECS. Appendix B in the MAX IMS/UTIL Online manual has more information on this exit.

Step 6.5: Add MAX Data/Util, MAX/PDF, MAX IMS/UTIL and MAX DB2/UTIL into the ISPF Main Menu

Modify your ISPF Primary Option Menu to include MAX product components. You may use the MCL@ALL member of the .MXRXVnnn.PANELS data set as a model. Shown below are sample lines that are added to the main menu panel.

MAX Data/Util

In the)BODY section: % U+DATA/UTIL (MAX Data File utilities)

In the)PROC section: U, 'CMD(%MAX)'

MAX/PDF

To the)BODY section: % M+MAX/PDF (Data set and DASD utilities)

To the)PROC section: V, 'CMD(%MAX PDF)'

MAX IMS/UTIL

To the)BODY section: % I+IMS/UTIL (MAX browse, edit and utilities for IMS)

To the)PROC section: I, 'CMD(%MAX IMS)'

MAX DB2/UTIL

To the)BODY section: % D+DB2/UTIL (MAX browse and editor for DB2)

To the)PROC section: D, 'CMD(%MAX DB2)'

Step 6.6: Modify MAX/PDF "OPTIONS PROFILE" to set default user options

Perform this step only if installing MAX/PDF or MAX MVS/UTIL.

Edit member 'VTLPROF' in the .MXRXVnnn.PANELS data set. Change the assignments between the /* BEGIN and /* END statements in the)INIT section in this panel, as indicated below.

Note: If these variables have been set in the 'MAX' member in the EXEC library, the values entered in this panel will not override the values set in the EXEC. ([Step 6.1: Modify 'MAX' Startup REXX Procedure](#)).

MAXCDSN= MAX/PDF saves the DSNs into the partitioned data set assigned to MAXCDSN. This data set may be assigned to the user profile data set, or to any other partitioned data set with an LRECL 80.

MAXIDSN= MAX/PDF accesses members stored in the mxqual.MXRXVnnn.EXECS data set. Change 'mxqual' and 'nnn' to the appropriate values. 'mxqual' was previously specified in [Step 2: Create the MAX libraries from the installation media](#), and 'nnn' is the version level of the MAX software.

Step 6.7: Copy 'MAX' startup REXX procedure to SYSEXEC/SYSPROC data set

Before you can execute MAX, the 'MAX' startup REXX procedure must be copied to a data set allocated to the SYSEXEC or SYSPROC ddnames.

Step 7: Execute MAX

There are several ways to execute MAX:

If the logon PROC and menu have been modified, Logon to ISPF.

Select 'M' from the menu.

At any COMMAND ==> prompt, enter 'MAX PDF' to invoke the MAX/PDF Primary Option Menu.

At any COMMAND ==> prompt, enter 'DSN' to invoke the MAX data set name panel.

From ISPF option 3.4 data set list, the 'MAX' command may be entered next to any data set name.

The supplied REXX program may be invoked from a command line by entering:

TSO %MAX PDF	to invoke MAX/PDF main menu
TSO %MAX DSNL	to invoke MAX/PDF project list
TSO %MAX DSN	to invoke the 'Specify a Data Set Name' panel
TSO %MAX	to invoke MAX Data/Util main menu
TSO %MAX IMS	to invoke MAX IMS/UTIL main menu upon initiation, the MAX IMS/UTIL profile parameters must be established. See Step 7.1: Establish IMS Profile Parameters for more information.
TSO %MAX IVP	to invoke the installation verification procedure
TSO %MAX DB2	to invoke MAX DB2/UTIL main menu.

Step 7.1: Establish IMS Profile Parameters

Perform this step only if installing MAX IMS/UTIL.

Upon initial start-up, the IMS/UTIL Profile Parameters screen will be displayed to establish required parameters for which there is either no default, or the default usually must be reviewed. These are:

- Application Group Name (AGN)
- DLI Batch DFSVSAMP data set name
- DBD Module Library (DBDLIB)
- PSB Module Library (PSBLIB)
- IMS RESLIB data set allocations

See Option 0 - PROFILE PARAMETERS in the MAX IMS/UTIL Online Manual for information on these parameters. Also, press the HELP key (PF1) while viewing the Profile Parameters screen to display detailed help information about each profile parameter.

After all profile parameters have been established, press Enter to proceed to the main menu and specify the PSB/PCB information required to access your IMS database. See Chapter 2 of the MAX IMS/UTIL Online Manual for information about using the product to access IMS databases.

In order to distribute these Profile Parameters to first time users, the installer may choose to edit the MILPROF member in the .PANELS library and provide these values as defaults.

Step 8: Review/Update the installed .JCL library

Step 8.1: Check the README member for samples and special notes

The README member in the .JCL library contains sample data, copybooks, exits and other information that is valuable in getting to know the product. There are also descriptions of the LINK and AUTHORIZATION jobs that are used to activate the various components of the MAX products.

Step 8.2: Update MAXFT* members in the .JCL library

The members beginning with prefix MAXFT* provide skeleton JCL for various functions. Each contains a STEPLIB statement that should be updated to reflect your naming of the MAX loadlib.

Step 8.3: Update CBMAP member in the .JCL library

The CBMAP member is a sample of Mapping Criteria. It contains two DSN= statements that should be updated to reflect your naming of the MAX JCL library. Note that this step is optional. CBMAP is used with the SAMDEF and VSAMDEF sample data sets.

Step 9: Complete MAX DB2/UTIL and MAX/REXX SQL feature installation

This is a 3-part procedure (9.1-9.3)

Perform only step 9.1 if you are installing the MAX/REXX SQL feature without MAX DB2/UTIL. Perform steps 9.1 through 9.3 if you are installing MAX DB2/UTIL. These steps must be performed once for each DB2 subsystem that needs to be accessed by the MAX products. Step 9.4 creates sample tables and is optional.

Step 9.1: Bind MAX DB2/UTIL and MAX/REXX SQL feature to DB2

Perform this step if installing either MAX DB2/UTIL or the SQL feature of the MAX/REXX SQL product. There are two similar members in the .MXXRVnnn.JCL data set created previously. Choose member BINDB2UT if installing MAX DB2/UTIL or choose member BINDRSQL if installing the MAX/REXX SQL feature without MAX DB2/UTIL. Edit the chosen member, making the appropriate changes to all occurrences as described in the comments section of that member and listed below.

Perform this step only if installing the SQL feature of the MAX/REXX product. Edit the member BINDRSQL in the .MXXRVnnn.JCL data set created previously, making the appropriate changes (to all occurrences) as described in the comments section of that member, and listed below.

MXQUAL	Set this equal to the high-level qualifier for all the data sets except MXLOAD. Use the same value that was specified for MXQUAL in Step 2: Create the MAX libraries from the installation media .
DSNT	This is the name of the DB2 subsystem that you want to bind the RXXSQL program to.

After the appropriate changes have been made, submit this job. Expect a return code of zero (0) from each step of this job.

This job binds MAX/REXX SQL to a DB2 subsystem and GRANT use authority to PUBLIC.

Note: If MXXRLINK is ever re-run for any reason (such as to apply a PTF) then this job must also be re-run.

Step 9.2: Grant access to DB2 catalog tables

Perform this step only if installing MAX DB2/UTIL. Edit member DB2GRANT in the .MXRXVnnn.JCL data set created previously, making the appropriate changes to all occurrences as described in the comments section of that member and listed below.

- | | |
|--------|--|
| MXQUAL | Set this equal to the high-level qualifier for all the data sets except MXLOAD. Use the same value that was specified for MXQUAL in Step 2: Create the MAX libraries from the installation media . |
| DSNT | This is the name of the DB2 subsystem that you want to grant catalog table access for. |

After the appropriate changes have been made, submit this job. Expect a return code zero (0) from each step of the job.

This job grants SELECT access to PUBLIC on the DB2 catalog tables required by MAX DB2/UTIL. If not all users are to be authorized to use MAX DB2/UTIL, replace PUBLIC with the appropriate authorization IDs.

Step 9.3: Verify WLM support for DB2 external functions

The Data Privacy functions provided in MAX DB2/UTIL are DB2 external functions, which are provided only through the WorkLoad Management (WLM). In order to use these features, please verify all the following conditions are met:

1. You are running WLM in Goal Mode.
2. The external function load modules are accessible to WLM (either by placing MAX loadlib in the STEPLIB of the WLM proc, or placing the external function modules in the linklist). These modules are: MAXDB2F2, MAXDB2F3 and MAXDB2F4

Step 9.4: Create TEMP database if none exists

MAX DB2/UTIL utilizes the Global Temporary Table feature of DB2. If you already have one defined, you may skip this step. Otherwise, you may use DB2TMPTB to create one. Edit member DB2TMPTB in the .MXRXVnnn.JCL data set created previously, making the appropriate changes to all occurrences as described in the comments section of that member and listed below.

- | | |
|--------|--|
| MXQUAL | Set this equal to the high-level qualifier for all the data sets except MXLOAD. Use the same value that was specified for MXQUAL in Step 8.2: Update MAXFT* members in the JCL library . |
| DSNT | This is the name of the DB2 subsystem that you want to grant catalog table access for. |

Step 9.5: Create MAX DB2/UTIL sample tables

This step is completely optional and is not required for the correct operation of MAX DB2/UTIL.

Edit member DB2SAMP in the .MXXVnnn.JCL data set created previously, making the appropriate changes to all occurrences as described in the comments section of that member and listed below.

- MXQUAL Set this equal to the high-level qualifier for all the data sets except MXLOAD. Use the same value that was specified for MXQUAL in [Step 2: Create the MAX libraries from the installation media](#).
- DSNT This is the name of the DB2 subsystem that you want to create the sample tables in.

After the appropriate changes have been made, submit this job. Expect a return code zero (0) from each step of the job.

This job creates some sample tables using the IBM-provided sample tables as source. These tables may be used to learn about MAX DB2/UTIL without harming real data. This job may be rerun to recreate the sample tables once the data has been modified. This job may also be run by users to provide individual copies of the sample tables.

Step 10: Complete MAX IMS/UTIL installation for using Dynamic PSBs

Perform these steps only if installing MAX IMS/UTIL and want to use the Dynamic PSB feature.

Step 10.1: Dynamic PSBs

MAX IMS/UTIL can dynamically generate a PSB from the contents of a specified Database Descriptor (DBD) module rather than use a predefined static PSB that must be generated and maintained by the DBA. The following are characteristics of Dynamic PSBs:

- When DYNAM or DYNAMSEG are specified in the PSB name field, a PSB is dynamically generated to support the requested IMS/UTIL online or batch function running in either DLI or BMP mode.
- The generated PSB exists only temporarily for duration of the requested function and then it is deleted.
- For batch functions, PCBs for all externally related databases are automatically included in the generated PSB
- When a database is to be accessed through a defined secondary index, a PSB containing the segment hierarchy inverted around the target segment is dynamically generated.
- The PSB PROCOPT is generated based upon the function requested. Read only functions generate PROCOPT=G, while update functions generate PROCOPT=A (or R if accessing with an alternate index).
- DYNAM specifies that all segment names in the DBD are to be included in the dynamically generated PSB for Browse and Edit functions. Mass Update, unload, load, and compare functions include only those segments in the PSB that are necessary to perform the operation as specified by the segment select/change criteria.
- DYNAMSEG displays all segment names in the DBD and prompts for which ones to include in the dynamically generated PSB. Requested segments and all higher level segments along the concatenated key will automatically be included in the PSB definition. All functions will operate on only the chosen segments.
- A PSBGEN into a temporary PSBLIB is performed automatically for the dynamically generated PSB. Programs ASMA90 and IEWL must be available for this to the MAX IMS/UTIL user when running in both TSO and Batch environments.
- When running in BMP mode, MAX IMS/UTIL generates an ACB from the dynamic PSB and places it in a user defined DOPT ACBLIB.

Using Dynamic PSBs can offer significant performance benefits, particularly in batch functions, where only the necessary segments and processing options (PROCOPT) that match the requested function are included in the generated PSB.

Dynamic PSBs require parameters to be setup in the MAXIOPTS installation options module as detailed in [Step 10.2: Modify the MAXIOPTS installation options module for dynamic PSB parameters](#) before they can be used. For BMP processing, changes to the MSGEN and control region JCL must also be performed as detailed in [Step 10.3: Modifications required when using Dynamic PSBs with BMP run mode](#).

Step 10.2: Modify the MAXIOPTS installation options module for dynamic PSB parameters

The MAXIOPTS installation options module specifies parameters that describe and control the dynamic PSB operating environment used by MAX IMS/UTIL. A default MAXIOPTS module has been distributed in the MXRXVnnn LOADLIB data set and may be reassembled and linked to change any and all of the default parameters. Member MAXIOPTJ in the MXRXVnnn JCL library contains JCL for the re-assembly and link of the MAXIOPTS load module.

Following is the MAXIOPTS source contained in the MXRXVnnn JCL library.

```

*****
*      MAXIOPTS - MAX IMS/UTIL INSTALLATION OPTIONS
*****
MAXIOPTA IMSIDVAL=N,          IMSID VALIDATION REQUIRED           X
DYNPRF=TEST,                DYNAMIC PSB PREFIX                 X
DYNMAX=0005,                DYNAMIC PSB MAXIMUM                X
DYNACB=MXS.IMS.ACBLIB,     DYNAMIC PSB ACBLIB                 X
DYNMAC=IMS710.SDFSMAC      DYNAMIC PSB IMS MACLIB
*
*****
*      SAMPLE IMSID TABLE - UP TO 32 IMSID ENTRIES FOR IMS/UTIL
*****
MAXIOPTI IMSID=IUP1,          X
IMSDESC='SAMPLE WITH DEFAULTS'
*
MAXIOPTI IMSID=MAX1,          X
IMSDESC='SAMPLE DYNAMIC PSBS ', X
DYNPRF=IUTL,                 X
DYNMAX=0100,                 X
DYNACB=IMS710.ACBLIB.DYNPSB, X
DYNMAC=IMS710.SDFSMAC
*
MAXIOPTI IMSID=MAX2,          X
IMSDESC='NO DYNAMIC PSB ',    X
DYNMAX=0
*
MAXIOPTI IMSID=MAX3,          X
IMSDESC='NO DYNAMIC PSB W/BMP', X
DYNACB=NONE
*
*****
*      GENERATE INSTALATION OPTIONS MODULE
*****
MAXIOGEN
END

```

MAXIOPTS Macro Definitions

The macros and parameters contained in the MAXIOPTS source module are described as follows:

Macro	Parameter	Length	Type	Description
MAXIOPTA	-	-	-	Default parameters that apply to all IMSID subsystems when the corresponding MAXIOPTI parameter is blank.
	IMSIDVAL	1	Y/N	IMSID validation required. Y=Only IMSID values specified in MAXIOPTS are valid with IMS/UTIL. N=No IMSID validation is performed.
	DYNPRF	4	A/N	Default Dynamic PSB assigned prefix name. Each IMSGEN defined dynamic PSB name begins with this prefix, e.g. TST.
	DYNMAX	4	NUM	Default Dynamic PSB maximum names and sessions allowed to be in use at the same time. Length determines generated length of each dynamic PSB name. e.g. 050 will generate names TST001 - TST050, while 0100 will generate names TST0001 - TST0100. Specify 0 to disable dynamic PSBs.
	DYNACB	44	DSN	Default Dynamic PSB DOPT ACBLIB data set name that will be concatenated to the IMS subsystem ACBLIB. Specify NONE to disable dynamic PSBs in BMP mode.
	DYNMAC	44	DSN	Dynamic PSB IMS MACLIB data set name that will be used for PSBGEN assembly.
MAXIOPTI	-	-	-	Parameters specified for a specific IMS subsystem. Up to 32 IMSIDs may be specified. These values override corresponding parameters specified in the MAXIOPTA macro.
	IMSID	17	A/N	IMS subsystem id for which parameters apply.
	IMSDESC	20	A/N	IMS subsystem description in quotes.
	DYNPRF	4	A/N	Dynamic PSB assigned prefix name.
	DYNMAX	4	NUM	Dynamic PSB maximum allowed. Specify 0 to disable dynamic PSB for this IMSID.
	DYNACB	44	DSN	Dynamic PSB DOPT ACBLIB data set name. Specify NONE to disable dynamic PSBs in BMP mode for this IMSID.
	DYNMAC	44	DSN	Dynamic PSB IMS MACLIB data set name.
MAXIOGEN	-	-	-	Generates source for MAXIOPTS.

Step 10.3: Modifications required when using Dynamic PSBs with BMP run mode

- A new DOPT ACBLIB must be allocated for each IMS subsystem that can be used by MAX IMS/UTIL to place the output of the dynamic PSB ACBGEN. Each ACBLIB must be allocated as a PDS with no secondary extents. DCB attributes should be the same as the standard subsystem ACBLIB. This name must have been specified in the MAXIOPTS installation options module. For example:

Data Set Name	MXS.IMS.ACBLIB	(any non-temporary set name)
Device type	3390	(any allowed by IMS)
Organization	PO	(must be a PDS)
Record format	U	
Record length	0	
Block size	32760	(same as subsystem ACBLIB)
1st extent cylinders	30	(make large enough)
Secondary cylinders	0	

- Each IMS subsystem must have their new DOPT ACBLIB concatenated after the standard subsystem ACBLIB. For example:

```
//IMSACBA DD DSN=standard.subsystem.acb.librarya,DISP=SHR
// DD DSN=MXS.IMS.ACBLIB,DISP=SHR
//IMSACBB DD DSN=standard.subsystem.acb.libraryb,DISP=SHR
// DD DSN=MXS.IMS.ACBLIB,DISP=SHR
```

- MAX IMS/UTIL users must have update authority to the new DOPT ACBLIB.
- Determine the prefix and maximum number of dynamic PSB names required for each IMS subsystem. Each TSO user or job simultaneously accessing a database with MAX IMS/UTIL and dynamic PSBs will use a uniquely generated PSB name. For example:

Dynamic PSB name prefix: MAXI

Dynamic PSB maximum: 0050

These will permit simultaneous access to 50 dynamic PSBs of the form: MAXI0001 to MAXI0050

- Each dynamic PSB name must be defined in the IMS subsystem IMSGEN with an APPLCTN macro using the DOPT operand. For example:

```
APPLCTN DOPT,PSB=MAX0001,PGMTYPE=(BATCH,,)
APPLCTN DOPT,PSB=MAX0002,PGMTYPE=(BATCH,,)
...
APPLCTN DOPT,PSB=MAX0050,PGMTYPE=(BATCH,,)
```

- If AGN security is being used, determine if new AGNs should be established for the PSB name ranges.

Step 11: Define Relationship Definition File if applicable (MAX DB2/UTIL)

The Relationship Definition File is a VSAM file that contains user defined relationships of DB2 tables. This file is used only by the Relational Unload feature found in the MAX DB2/UTIL product. If you do not use this feature, or have already defined this file for a previous release, you may skip this step.

The Relationship Definition File will be defined and initialized by this jobstream. Edit the job RELATDEF and update the job as defined in the comments before submitting.

APPENDIX A: PRODUCT AUTHORIZATION

When you receive your tape or download your software via FTP, you will be provided with authorization codes to enable the product(s). The codes are provided in member MAXAUTH and should be downloaded into the MAX products .JCL library. Once this is complete, the product codes are applied by submitting JCL for ASMAUTHS. Please follow the instructions in this JCL member before submitting. The same method is used for applying product codes for both TRIAL and PRODUCTION passwords, for all MAX products.

If you are upgrading from a prior release of the product, note that this procedure is completely new, and job(s) you used to apply the zaps to the prior release have been replaced with this new streamlined procedure.

Once authorization codes have been applied, the system is ready for use.

The following are examples of both production and trial zaps in the format that you will receive them. Instructions on using these as input to the ASMAUTHS job in the installation JCL library will be provided with the zaps that you receive.

Production Password

```

*****
* MAX Software:      Password Authorization Module          *
*                                                           *
*          (C) COPYRIGHT 2004 MAX Software, INC.          *
*                                                           *
* MODULE NAME:      MAXAUTH                                *
*                                                           *
* Customer Info   :  SAMPLE CUSTOMER                      *
* PURPOSE         :  Apply MAX product authorization codes.*
*   You should expect a return code 0 from MAXAUTH        *
*                                                           *
*****
MAXAUTH  CSECT
MAXAUTH  AMODE ANY
MAXAUTH  RMODE ANY
          DC  CL8'PASSWORD'
TRIAL    DC  2X'00000000'
PWD1     DC  X'67B1A47C4F18DA635A2D594549D3699F2EEF412C'  MAXX
PWD2     DC  X'67B1A47C4F18DA635A2D594549D3699F2F071F2C'  MREX
          DC  5X'FFFFFFFF'
          DC  C'(C) MAX/REXX, MAX DATA/UTIL, MAX/BATCH, MAX/PDF'
          DC  C'   MAX/IMS/UTIL, MAX/DB2/UTIL              '
          DC  C' Copyright MAX SOFTWARE INC.'
          DC  C' 1993, 2003. All rights reserved.'
          DC  C' Tel: 303-985-1558, Fax: 303-985-1556'
          DC  C' MAX Software, 3609 S. Wadsworth'
          DC  C' Denver, Colorado USA 80235'
          END  MAXAUTH

```

Trial Password

```
*****
* MAX Software:      Password Authorization Module      *
*                                                           *
*           (C) COPYRIGHT 2004 MAX Software, INC.      *
*                                                           *
* MODULE NAME:      MAXAUTH                            *
*                                                           *
* Customer Info   :  SAMPLE TRIAL CUSTOMER            *
*                                                           *
* PURPOSE         :  Apply MAX product authorization codes. *
*           You should expect a return code 0 from MAXAUTH *
*                                                           *
*****
MAXAUTH  CSECT
MAXAUTH  AMODE ANY
MAXAUTH  RMODE ANY
DC  CL8'PASSWORD'
TRIAL   DC  X'67B3734CADC4D236'      Trial
DC  5X'FFFFFFFF'
DC  C'(C) MAX/REXX, MAX DATA/UTIL, MAX/BATCH, MAX/PDF'
DC  C'   MAX/IMS/UTIL, MAX/DB2/UTIL      '
DC  C' Copyright MAX SOFTWARE INC.'
DC  C' 1993, 2004. All rights reserved.'
DC  C' Tel: 303-985-1558, Fax: 303-985-1556'
DC  C' MAX Software, 3609 S. Wadsworth'
DC  C' Denver, Colorado USA 80235'
END  MAXAUTH
```

APPENDIX B: INSTALLATION VERIFICATION

To verify that a release has been properly installed, a program has been distributed with the product. If questions arise concerning product release level, proper installation of maintenance or authorization zaps, this data will be important in problem resolution. Have this information available or access to the system to obtain this information when requesting product support.

To run this program, enter the following from a terminal attached to the system on which you intend to use the product.

Enter the command: TSO MAX IVP

The following output will be returned:

System Information

```
MAX SOFTWARE - INSTALLATION VERIFICATION PROCEDURE
DATE: 16 Mar 2003 TIME: 10:45:04

SYSTEM-ID: P390

CPU 04000001-7490
1 CPU(S) ON THIS SYSTEM

REXX370 3.48 May 01 1992
```

Assembly dates of components in the IRXFUSER user replaceable module

```
IRXFUSER COMPONENTS
-----
RXIFI    ASSEMBLED ON 11/24/02-11.26
RXMVS    ASSEMBLED ON 11/24/02-11.16
RXSQL    ASSEMBLED ON 09/16/02-06.37
RXVSAM   ASSEMBLED ON 11/27/02-13.41
```

A listing of the assembly dates of each load module comprising the product

```

PRODUCT LOAD MODULES
*****
Install tape at level Unnn Rn
*****
-----
MAXBAT  ASSEMBLED ON 02/19/03-17.08
MAXCOMP NOT FOUND
MAXDUTIL ASSEMBLED ON 02/19/03-17.06
MAXDB2UT ASSEMBLED ON 02/19/03-17.06
MAXIBAT  ASSEMBLED ON 02/19/03-17.28
MAXIBATO ASSEMBLED ON 02/19/03-17.28
MAXIDPSB ASSEMBLED ON 02/19/03-17.30
MAXIEDBO ASSEMBLED ON 02/19/03-17.26
....etc....etc.....

```

Authorization code information

```

PRODUCT AUTHORIZATION
-----
TRIAL ZAP:  REP 0008 00000000,00000000          **FAILED**
              NO TRIAL ZAP APPLIED
PRODN ZAP-1  REP 0010 67D68562,5CF4F597,4E6D5617,52EF5195,232F3DFE  **MAXX**
PRODN ZAP-2  REP 0024 67D68562,5CF4F597,4E6D5617,52EF5195,23471BFE  **MREX**
PRODN ZAP-3  REP 0038 67D68562,5CF4F597,4E6D5617,52EF5195,13603AFA  **D2UT**
PRODN ZAP-4  REP 004C 67D68562,5CF4F597,4E6D5617,52EF5195,184217FE  **IMAX**

END VERIFICATION

```

READER COMMENT FORM

Product Installation Guide V3.3.0

Please assist us in improving this document by returning your comments about its contents and current organization. Please refer to specific page numbers when applicable.

Please forward your comments to us on this Reader Comment Form with the understanding that all comments become property of MAX Software, Inc. We may use or distribute whatever information you supply in any way we believe appropriate without incurring any obligation to you.

Please fax this page to us at: (303) 985-1556.

Name		Company	
Title		Address	
Dept.		City	
Phone		State/Zip	
Fax		E-Mail	

Comments:
