



Installing the MSC/MVISION Databanks

This chapter details the installation of MSC/MVISION Databanks on Unix computer systems. It is recommended that you read this chapter thoroughly before proceeding with the installation.

Preparing for Installation

Before you begin the installation, please note the following items:

- You may need super user access to install the databanks or to install the license codes for access to the databanks. When a superuser account is used for the install, the `mscsetup` script changes only the directory in which the software is being installed.
- The software can be installed anywhere you choose on a machine's file system as long as the account you use has write permission to that area.
- The installation procedure should be performed when MSC/MVISION products are not in use, in order to avoid unpredictable results.
- The entire installation procedure could take as little as a half hour or as long as six hours, depending on the set of databanks you are downloading from the CD ROM.
- Review the system requirements for your platform outlined in the rest of this section.
- Review *Updating the form.definitions Files*, on page A-28. This installation may affect your current operating environment.
- MSC/MVISION Databanks have no specific hardware requirements. They are designed to work on the same hardware configuration as your MSC/MVISION software. Additional configuration modifications recommended in this section improve the performance of MSC/MVISION software on your system.

Note that the Databanks come in two formats: one format for DEC workstations and one for UNIX workstations. The UNIX workstations include HP, SUN, SGI, and IBM. These workstations can all read databanks created by the other workstations in the group. The DEC Ultrix and DEC Alpha machines use a byte swapping technique that makes them unique. Databanks created on these workstations can only be read by other DEC workstations.

Memory Requirements

The performance of the MSC/MVISION software products is affected by the availability and amount of main memory and system swap space on the operating system. The products are configured to run with a minimum amount of memory.

Performance is best when you have at least **64** megabytes of memory but you can run with as little as **32** megabytes.

Most MSC/MVISION products have a parameter to set the size of the databank buffer. The buffer default is usually set to 8 megabytes. Databanks smaller than 8 megabytes are loaded completely into memory for optimum performance. If you are working with databanks larger than 8 megabytes and your system has lots of memory then you want to increase the databank buffer size. For MSC/MVISION Builder and Evaluator, use the resource setting of `Mveval*DbBufferSize:` in the `Mveval` resource file. For MSC/MVISION Pro the default is 8 megabytes. To change it use the `db_buffer_size` parameter in the `mvpro.dat` file.

Warning: Use care in setting the size of the databank buffer. Setting it larger than available memory can result in swapping conflicts between the MSC/MVISION software and the operating system.

Disk Space Requirements

The amount of disk space required to install the databanks depends on the combination of databanks you have licensed or plan to license. *Step 5: Selecting Products to Install*, on page A-13 defines the process for determining the required disk space.

The `mscsetup` script determines the amount of free disk space, in kilobytes, on the file system where the databanks are to be installed. See *Step 4: Selecting the Installation Directory*, on page A-11.

Updating Your Databank Installation

The databanks are loaded in a separate directory to minimize changes to the software products installation and allow flexibility in locating the databanks.

The new release is installed in a directory named `<db_install_dir>/mvision_db/db_xqxx`, where `xqxx` is the current release number. The `mvinstall_db` script renames the current databank directory to `<db_install_dir>/mvision_db/db_old` and provides links from the databank directory, `<db_install_dir>/mvision_db/db_xqxx`, to any of the MSC/MVISION software products installed at your site.

If your system is limited in disk space, databanks from the previous release may be deleted prior to or after installing the new release. See *Removing Files or Databanks from Your System*, on page A-41.

Directories and Files Created During Installation

After installation, your system contains numerous files. The exact number and names of the files depend on your specific configuration and the version numbers of the products that are currently available for your system. The following directories and files may be created for a specific databank installation:

Directory or File in <db_install_dir>	Description
mvision_db/	Directory created in the <db_install_dir> which stores all the databank files, utilities required to install and link them to the different MSC/MVISION software products, and the directories where the links are defined.
maint/mvinstall_db*	Utility that places all files in their proper directory after loading (installation script).
dbl_k_eval/ dbl_k_mvpro/ dbl_k_patran/	Directory where the links to the databanks and associated files that MSC/MVISION software uses are defined. The contents of this directory are modified via the mvinstall_db script. A separate directory is created for each MSC/MVISION product as required.
db_xqxx/	Directory where all the MSC/MVISION databanks and associated files for a specific release are located. db_xqxx identifies the current release number .
form.definitions_dir/	Directory where the mvinstall_db script builds and stores the databank form.definitions file used by MSC/MVISION Evaluator and Builder.
<databank>.des	MSC/MVISION databank file (binary).
<databank>.index	MSC/MVISION indexing file (binary).
<databank>.unt	Databank units conversion file.
<databank>.map	Databank mapping file for MSC/MVISION Pro.
<databank>.dis	Databank disclaimer file.

Directories and Files Created During Installation



Directory or File in <db_install_dir>	Description
<databank>.des.mapping	Databank export mapping file for MSC/MVISION Evaluator and Builder and MSC/MVISION Pro. These files are differentiated by the additional suffix .eval or .pro while stored in the db_xqxx directory.
<databank>.mvtm	Databank mapping file for MSC/MVISION PATRAN MATERIALS.
<databank>.mvform	Forms created for each databank from which the form.definitions is compiled.
form.definitions	A file which defines the criteria, categories, and Materials List of the Browser in MSC/MVISION Evaluator and Builder. You may edit this form to change your viewing options. Refer to the <i>MSC/MVISION Evaluator User's Guide and Reference</i> .
schema/	Directory where the define files for all the MSC/MVISION databanks are located.
schema/<databank>.def	Define files for all MSC/MVISION Databanks. You may use these files as examples to assist in changing the way data is stored in your databanks or for creating new databanks.
<databank>.docs/	Directories containing text files used by specific databanks.
<databank>.docs/<filename>	Databank specific files which contain the information available for viewing in text format.

Directories and Files Created During Installation (Continued)

Installing MSC/MVISION Databanks

The following sections provide detailed instructions for completing the installation of MSC/MVISION Databanks.

Overview of the Installation Procedure

The installation of the MSC/MVISION Databanks is accomplished using the following basic procedure:

1. Create the `<db_install_dir>`. The install procedures link this directory to your MSC/MVISION software product directories.
2. Extract the installation utilities, the MSC/MVISION Databanks, and all databank auxiliary files from the delivery media.
3. Execute the installation utility to create the required links to the new databanks.
4. Update the `form.definitions` file, if necessary.

A

Extracting MSC/MVISION Databanks

The `mcssetup` utility performs many of the steps necessary to get MSC/MVISION Databanks installed. It creates a log file, `mcssetup.log`, as a record of all messages, prompts, and user input. The `mcssetup` utility also creates a playback file, `mcssetup.pbk`.

For a basic interactive installation, MSC recommends executing the `mcssetup` utility with no flags or options. For advanced installers, `mcssetup` offers the following additional capabilities:

- To generate several identical installations, run in install-only mode and specify an existing playback file:
`mcssetup mcssetup.pbk`
- To obtain usage information on the utility, execute:

`mscsetup help`

Command Line Arguments	Purpose
<code>-b <installation_dir></code>	Sets the default installation directory.
<code>-f <setup_file></code>	Runs <code>mscsetup</code> from a different setup file. For installation of MSC/MVISION it is recommended that the default setup files (in the CDROM tools directory) be used.
<code>-l <log_file></code>	Writes the installation log to <code><log_file></code> . Default is <code>mscsetup.log</code> .
<code>-m cd:<cd_path></code>	Specifies media type (cd for MSC/MVISION) and path to media.
<code>-n</code>	Turns off screen handling for text-only scrolling mode.
<code>-p <playback_file></code>	Writes the playback file to <code><playback_file></code> . Default is <code>mscsetup.pbk</code> .
<code>-r <user@node></code>	Sets the user and node for remote installation.
<code>-t <tmp_dir></code>	Sets the directory <code>mscsetup</code> will use for scratch. Default is <code>/tmp</code> .
<code>-x</code>	Expert mode. Shows menus and prompts with no explanatory text. Recommended only for users very familiar with <code>mscsetup</code> .

mscsetup Command Line Arguments

Step 1:
Mount CD-ROM

Mounting the CD-ROM may vary according to the setup of your operating system. The steps in this section represent default mounting procedures; however, simpler tools are often available as part of user environment packages such as the Common Desktop Environment (CDE) or SGI's 4DWM. Consult your operating system documentation for the best method of mounting CD-ROM devices.

1. Create a directory for the CD-ROM (if it does not already exist):

```
mkdir /CDROM
```

2. Mount the CD-ROM drive by issuing one of the following commands:

Digital UNIX: `mount -t cdfs /dev/rzUA /CDROM`
where U = CD-ROM unit, A = a or c.

HP 10.x: `mount -F cdfs /dev/dsk/c0tUd0 /CDROM`
where U = the SCSI controller # for the CD-ROM drive

IBM: `mount -rv cdrfs /dev/cd0 /CDROM`

SGI: `mount -rt iso9660 /dev/scsi/scCdU10 /CDROM`
where C = the SCSI controller # and U = CD-ROM unit #.
These values can be found by entering the command,
`/bin/hinv`.

NOTE: The “1” in `scCdU10` is the letter “L” in lowercase, not a “one”.

Sun Solaris: You do not need to enter a mount command, because the MSC/MVISION CD-ROM will automatically be mounted by the Solaris vold daemon. The actual mount point is specified in the file, `/etc/vold.conf`, which by default, is `/cdrom/cdrom0`. If this directory does not appear after inserting the CD-ROM, make sure the vold daemon is running.

3. Remove the MSC/MVISION CD-ROM, by entering:

```
cd <other_dir>
umount /CDROM
```

Where `<other_dir>` is any directory, other than `/CDROM`.

4. Press the Eject button on the CD-ROM drive. On Sun Solaris, enter:

```
eject cdrom
```

Step 2: Starting the mscsetup Installation Utility

You do not need to be superuser (root) to run `mscsetup`. Remember that the `mscsetup` utility keeps a log of important information so it is not necessary to make detailed notes. To execute `mscsetup` from the CD-ROM, enter:

```
<cdrom_dir>/mscsetup
```


The following information and instructions are then displayed on the screen¹:

Please wait...

MSC Installation and Setup - Introduction

System type: Silicon Graphics R4X00, R5K

NOTE: You might need superuser (root) privileges to use the default installation directory /msc.

The following products are available on this delivery:

* MVISION DataBanks db_4q98 for UNIX Systems

Do you want to continue (default = Y)? **Y**

MSC Installation and Setup - Introduction

Step 3:
Selecting an Installation Device

When using the `mscsetup` utility, you must identify the type of media from which you are installing, and the path to that media.

The `local CD` option assumes that the CD-ROM is available in the same file system as the installation directory. The CD-ROM must be either locally mounted or NFS mounted on the same machine as the installation directory.

The `remote CD` option allows accessing a CD-ROM on a remote machine using commands such as `rsh` and `rcp`.

The installation continues with the following screen display:

1. The example is for installation on an Silicon Graphics workstation. Other platforms are similar.

MSC Installation and Setup - Installation Device Location

1. Install from a local device.
Select this item if the installation CD is attached to this computer, or NFS mounted on this computer.
2. Install from a remote device.
Select this item if the installation device is attached to another computer.
- X. Completely exit this installation now.
Select this item to end this procedure now.

Enter 1, 2, or X (default = 1)? **1**

MSC Installation and Setup - Local Installation Device or File

1. Install from local tape.
 2. Install from local CD-ROM.
 3. Install from local disk file sets.
 4. Install from local baseline directory.
- R. Return to Installation Device Location menu.
X. Completely exit this installation now.

Enter 1-4, R, or X (default = 2)? **2**

Enter the pathname of the local CD-ROM mount point.

Pathname (default = `<cdrom_dir>`)

?<CR>

Install from "`<cdrom_dir>`"

Is this OK (default = Y)? **y**

Please wait.....

MSC Installation and Setup - Installation Device Location

Step 4:
Selecting the Installation Directory

The installation directory is the location where you will install all products. We refer to that directory as the `<db_install_dir>`.

The `mscsetup` utility creates a directory in the `<db_install_dir>` named `mvision_db`.



If you have not already created the directory in which to install the new databank release, <db_install_dir>, the mscsetup script will prompt you to create one.

It is suggested that you give the <db_install_dir> the name msc.

The utility then checks the amount of disk space available for that path.

mscsetup requires additional disk space for decompression of the files contained on the CD-ROM.

You will view the following dialog:

A

MSC Installation and Setup - Installation Base Directory

MSC strongly recommends installing this software in a "common" MSC product directory to allow MSC products to share common software and settings.

This can

- Reduce the amount of installation space needed by a product.
- Make future installations easier.
- Reduce the effort needed to administer MSC products.

MSC recommends the standard directory /msc, but alternate choices, such as /usr/local/msc, are also acceptable.

Enter the pathname of the directory you want to install into, X to completely exit this installation, or ? for a list of available filesystems.

Pathname (default = /msc)? **msc**

If the directory /msc does not exist, then:

The directory /msc does not exist.
Press ENTER or RETURN to continue...
Do you want to create it now (default = Y)?

MSC Installation and Setup - Installation Base Directory

If the directory `/msc` exists, then:

```
Directory /msc exists.
```

When given permission to proceed, the script checks the available space on that disk.

```
Checking available disk space
The size of the filesystem containing this directory is
973 MB, with 228 MB currently available.
```

```
Do you want to use this directory (default = Y)? Y
```

MSC Installation and Setup - Installation Base Directory (Continued)



Step 5: ***Selecting Products to Install***

After you have entered the appropriate file paths, you will be presented with a menu of all available product sets on the CD-ROM.

The available products are listed with an identification number in the left column. Initially, none of the products are selected, as indicated by the `No` in the `Mode` column. The following figure, *MSC Installation and Setup - Architectural Components*, on page A-14 is an example of the display on your screen. This varies with each release.

The `MB` column lists the approximate size in Megabytes that each databank and its auxiliary files occupy when copied to your disk. The total size of all selected databanks is shown at the bottom of the column.

Select each product by entering its identification number as specified by the script. After you select your products, the entire table is redisplayed and the `MB` column is recomputed to indicate the disk space required for the current selections. The `Mode` column changes to `Yes` for each selected product.

Deselect an item by entering its identification number a second time. This toggles the `Yes` in the `Mode` column to `No` and subtracts it from the total size.

If the total size of the selected products is too large for the space available in the current `<db_install_dir>`, you may elect to install some of the items now, and install the remainder of the delivery by repeating these steps at a later time.

MSC Installation and Setup - Architectural Components

No.	Architectural Component	Mode	MB
1.	Analysis Databank	No	6
2.	ASM Alloy Steel DataBank	No	4
3.	ASM Aluminum Databank	No	8
4.	ASM Composites Databank.	No	1
5.	ASM Copper Databank	No	4
6.	ASM Corrosion Databank	No	15
7.	ASM Magnesium Databank	No	1
8.	ASM Nylons Databank	No	4
9.	ASM Reference Library Databank	No	136
10.	ASM Stainless Steel Databank	No	6
11.	ASM Structural Steel Databank	No	1
12.	ASM Thermoplastics Databank	No	8
13.	ASM Thermosets Databank	No	5
14.	ASM Titanium Databank	No	1
15.	ASM Woldmans Engineering Alloys Databank	No	76
16.	ASM WW Guide Ferrous Databank	No	24
17.	ASM WW Guide Nonferrous Databank	No	26
18.	ASM Cross Reference Library Databank	No	140
19.	CAMPUS Support Files	No	1
20.	Demo Composites Databank	No	1
21.	Demo Metals Databank	No	3
22.	Demo Metals Databank (4q95)	No	2
23.	DYTRAN Databank	No	1
24.	EMASS Material Library Databank	No	1
25.	FATIGUE Databank	No	1
26.	Fiber Databank	No	1
27.	GE Plastics Databank	No	11
28.	Materials Selector Databank	No	63
29.	Mil 17A Databank	No	1
30.	Mil 17E Databank	No	2
31.	Mil 5 Databank	No	31
32.	New Mil 5 Databank	No	20
33.	PDLCOM Databank	No	113
34.	PDLCREEP Databank	No	3
35.	PDLTEMP Databank	No	3
36.	PMC90 Databank	No	1
37.	Producers Ceramics Databank	No	4
38.	Producers Composites Databank	No	7
39.	Producers Library Databank	No	181
40.	Producers Metals Databank	No	24
41.	Producers Plastics Databank	No	119
42.	Thermal Databank.	No	2

MSC Installation and Setup - Architectural Components

Space Required for Architectural Components 0

- C. Continue to the Optional Components Menu.
- X. Completely exit this installation now.

Enter a list of component numbers to select or deselect (you may use spaces or commas, a dash indicates a range, e.g., "1,3-5"), C. when you are done selecting architectural components, or X to exit this installation.

Enter 1-42, C, or X (no default)? **1-20**

775 MB available in installation directory /msc.

No.	Architectural Component	Mode	MB
1.	Analysis Databank	Yes	6
2.	ASM Alloy Steel DataBank	Yes	4
3.	ASM Aluminum Databank	Yes	8
4.	ASM Composites Databank	Yes	1
5.	ASM Copper Databank	Yes	4
6.	ASM Corrosion Databank	Yes	15
7.	ASM Magnesium Databank	Yes	1
8.	ASM Nylons Databank	Yes	4
9.	ASM Reference Library Databank	Yes	136
10.	ASM Stainless Steel Databank	Yes	6
11.	ASM Structural Steel Databank	Yes	1
12.	ASM Thermoplastics Databank	Yes	8
13.	ASM Thermosets Databank	Yes	5
14.	ASM Titanium Databank	Yes	1
15.	ASM Woldmans Engineering Alloys Databank	Yes	76
16.	ASM WW Guide Ferrous Databank	Yes	24
17.	ASM WW Guide Nonferrous Databank	Yes	26
18.	ASM Cross Reference Library Databank	Yes	140
19.	CAMPUS Support Files	Yes	1
20.	Demo Composites Databank	Yes	1
21.	Demo Metals Databank	No	3
22.	Demo Metals Databank (4q95)	No	2
23.	DYTRAN Databank	No	1
24.	EMASS Material Library Databank	No	1
25.	FATIGUE Databank	No	1
26.	Fiber Databank	No	1
27.	GE Plastics Databank	No	11
28.	Materials Selector Databank	No	63
29.	Mil 17A Databank	No	1



MSC Installation and Setup - Architectural Components (Continued)

30. Mil 17E Dat.	No	2
31. Mil 5 Databank	No	31
32. New Mil 5 Databank	No	20
33. PDLCOM Databank	No	113
34. PDLCREEP Databank.	No	3
35. PDLTEMP Databank	No	3
36. PMC90 Databank	No	1
37. Producers Ceramics Databank	No	4
38. Producers Composites Databank . . .	No	7
39. Producers Library Databank	No	181
40. Producers Metals Databank	No	24
41. Producers Plastics Databank	No	119
42. Thermal Databank	No	2
Space Required for Architectural Components		468

- C. Continue to the Optional Components Menu.
- X. Completely exit this installation now.

Enter a list of component numbers to select or deselect (you may use spaces or commas, a dash indicates a range, e.g., "1,3-5"),C. when you are done selecting architectural components, or X to exit this installation.

Enter 1-39, C, or X (no default)? **C**

MSC Installation and Setup - Architectural Components (Continued)

Step 6: **Product Licensing**

If you have not already received licenses from MSC, or would like to amend your existing contract to include additional databanks, the install procedure prompts you to generate a "Contract Amendment" as follows:

MSC Installation and Setup - Product Licensing Requirements

Product	Licensing Methods	Licensing Installed
MVISION DataBanks-SELECTED DataBanks	None	N/A

You can generate a contract amendment that can be completed and mailed or faxed to MSC.

Do you want to generate a contract amendment (default = N)? **N**

MSC Installation and Setup - Product Licensing Requirements



Step 7:
Installation Review

Prior to actually performing the installation, the `mscsetup` utility reviews your current selections. You may change the installation source, target install directory, or product selection.

MSC Installation and Setup - Installation Review

1. Installation Source: Local CD <cdrom_dir>
2. Installation Directory: /msc
 973 MB total
 778 MB available

3. Product	Type	Size	Test
MVISION DataBanks-SELECTED DataBanks	Custom	468MB	N/A

Enter Y to begin the installation now, or N to change the installation options.

Is this OK (no default)? **y**

MSC Installation and Setup - Installation Review

Step 8: **Installation Messages**

As the `mscsetup` utility performs the installation, it issues messages informing you of its progress.

This information is also recorded in the `mscsetup.log` file.

In this example we are installing the MSC/MVISION Databanks, including all auxiliary files.

MSC Installation and Setup - Installation

No additional user input will be required.

Writing new playback file, `"/msc/mscsetup.pbk"`

Changing working directory to `/msc`

Loading installation files. Fri Jan 15 15:28:44 PDT 1999

Using CD-ROM `<cdrom_dir>`

Begin loading files from `tools/gunzip.tar`

Verifying CD load operation...

Begin loading files from `unix/mvision_db.tar`

Verifying CD load operation...

Unpacking installation files. Fri Jan 15 15:32:53 PDT 1999

Unpack complete.

Decompressing installation files. Fri Jan 15 15:32:54 PDT 1999

Using `Gunzip/irix/gunzip`

Decompression complete.

Configuring Products. Fri Jan 15 15:54:56 PDT 1999

Configuring MVISION DataBanks - SELECTED DataBanks

Configuration complete. Fri Jan 15 15:54:57 PDT 1999

Installation complete. Fri Jan 15 15:54:57 PDT 1999

A log of this installation was written to:

`/msc/mscsetup.log`

MSC Installation and Setup - Installation

Step 9: Verifying the Installation

At this point in the process, all needed files have been read from the delivery media. Verify the directories and files extracted by entering the Unix list command in the `<db_install_dir>` directory:

```
ls -a /<db_install_dir>
./                Gunzip/           mscsetup.pbk
../              mscsetup.log     mvision_db/
```

Issue the `ls` command on the `<db_install_dir>/mvision_db` directory:

```
ls -a <db_install_dir>/mvision_db
```

Verify that the following files are in the `mvision_db` directory (This listing is for a UNIX platform.):

```
./          ../          db_4q98/   maint/
```

Verify that the following files are in the `maint` directory by issuing the following command:

```
ls -a <db_install_dir>/mvision_db/maint
./          ../          mvinstall_db* mvform*
```

The files labeled `mvform*` and `mvinstall_db*` are executables used to create the `form.definitions` directory and the links between the databanks and MSC/MVISION products installed at your site.

Installing Delivery Files and Providing Access

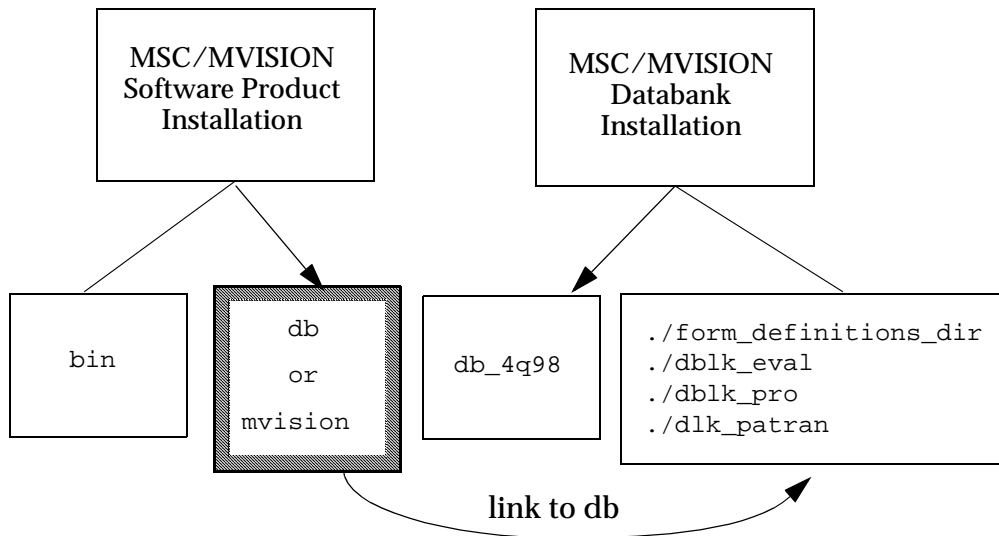
The `mvinstall_db` program creates the following directories and links required for MSC/MVISION software to access the databanks.

- The `<software_install_dir>/db` directory is created which links to the databank install directory (`<db_install_dir>/mvision_db/<dblk_software>`).



- Files in the `<db_install_dir>/<dblk_software>` directories are linked to specific databank files required by the software product, which are stored in the `db_xqxx` and user (`<db_user>`) databank directories.
- The `form.definitions` directory `<form_definitions_dir>` is created in which the `form.definitions` file, a compilation of all the `<databank_name>.mvform` files for installed databanks, is located.

Databank Link Between Databank Installation and Software Product Installation



The `mvinstall_db` script provides rapid conversion from one release to the next without deleting any data. It may be run at any time to update the current release or to change from one release to another. This method reduces the number of copies of databanks needed at an installation with more than one MSC/MVISION software product. It also isolates the databanks from the software products providing for independent management of databanks.

Before running the script, determine what links to the databank installation directory (`<db_install_dir>/mvision_db`) exist in your software installation directory (`<software_install_dir>`).

For each software product installed at your site, issue the following command:

```
ls -l <software_install_dir>
```

- For MSC/MVISION Builder and Evaluator:
Verify that the db directory exists in the <software_install_dir> and that it links to <db_install_dir>/mvision_db/dblk_eval.
- For MSC/MVISION Pro:
Verify that the db directory exists in the <software_install_dir> and that it links to <db_install_dir>/mvision_db/dblk_pro.
- For MSC/PATRAN MATERIALS:
Verify that the mvision directory exists in the <software_install_dir> and that it links to <db_install_dir>/mvision_db/dblk_patran.

If a previous installation of the MSC/MVISION Databanks exists, you can assume that the links exist without verification.

Running the Install Script

To run the install script, issue one of the following commands, using the case defined in *Step 1: Mount CD-ROM*, on page A-8:

```
cd <install_dir>/mvision_db
```

Issue one of the following commands:

```
/MAINT/MVINSTALL_DB or /maint/mvinstall_db
```

The script scans the <db_install_dir>/mvision_db directory for directories prefixed by db_*. These directories include those labeled:

db_[1-4]qxx	[1-4]qxx denotes the MSC/MVISION release version for the databanks and auxiliary files in that directory. The databanks read from the CD ROM are all placed in such a directory.
db_<user>	<user> is the name provided a directory storing your own databanks and auxiliary files.

When the script locates these directories it presents you with loading options. If only one db_* directory is found, type a **yes** or

no to install the databanks and files in that directory. Otherwise, type in the numbers corresponding to the directories to be installed and press <CR>.

The install script then prompts you for the MSC/MVISION software products installed at your site. Type in the numbers corresponding to the installed products to which you want to link the databanks being installed. Provide the script with the full path name to the product installation directory, when prompted.

When the path to an MSC/MVISION software product is requested, give the **exact** full path to set up the links between the software and the databanks correctly.

If your software installation directory already contains a link called db (or mvision) to the <install_db> directory, you may skip this step by pressing <CR>. The script confirms your response by printing a line:

```
Skipping the link for <Software_Product_and
version>.
```

Sample mvininstall_db Session, on page A-22 is a sample of an install session using the mvininstall_db program. Your session may vary, depending on the MSC/MVISION software products installed at your site.

```
% maint/mvininstall_db
```

```
      The MacNeal-Schwendler Corporation
Customer Delivery Program - mvininstall_db
      Version 4.0
      Platform xxxx
```

```
MVISION databank release 4q98 is available.
Do you wish to install or re-install this release?
[Yes or No] ==> y
```

```
Pick the MSC/MVISION products that are installed on this system.
Type a 'c' when you are done selecting products.
```

```
# Link Product Name
== =====
```

```
1 No   MSC/MVISION Builder 1.2A
2 No   MSC/MVISION Builder/Evaluator 3.x
3 No   MSC/MVISION Pro 2.0/15
4 No   MSC/PATRAN Materials (P3 MATERIALS SELECTOR)
```

Sample mvininstall_db Session

```
Enter a product number (1..4) or  
'c' - Continue with installation  
'q' - Quit without performing the installation  
CHOICE -> 2
```

```
# Link Product Name  
== =====  
1 No    MSC/MVISION Builder 1.2A  
2 Yes   MSC/MVISION Builder/Evaluator 3.x  
3 No    MSC/MVISION Pro 2.0/15  
4 No    MSC/PATRAN Materials (P3 MATERIALS SELECTOR)
```

```
Enter a product number (1..4) or  
'c' - Continue with installation  
'q' - Quit without performing the installation  
CHOICE -> c
```

```
-----  
Skipping setup of MSC/MVISION 1.2A  
-----
```

```
Building the dblk_eval directory for  
MSC/MVISION Builder/Evaluator 3.x from db_4q98
```

```
Building the form.definitions file and creating a  
link to it in dblk_eval.
```

```
Searching the directory dblk_eval for form files  
that match the databanks in the directory dblk_eval.
```

```
.....
```

```
Setting a link for MSC/MVISION Builder/Evaluator 3.x.
```

```
Enter the path to the MSC/MVISION Builder/Evaluator 3.x installation.  
(Press Return with no path to skip this operation.)
```

```
-> <software_install_dir>/mvision/mveval
```

```
-----Skipping  
setup of MSC/MVISION Pro 2.0/15  
-----
```

```
Skipping setup of MSC/PATRAN (PS MATERIALS SELECTOR)  
-----
```

```
mvininstall_db done.
```



Sample mvininstall_db Session (Continued)

After installation, issue the command to list the links created to the db directory for the software product(s) installed on your system:

```
ls -l dblk_<mvision_software_product>
```

To verify the linking, change directories to the software product directory and list the db directory. It should be the same as the previous list.

The list of databanks and auxiliary files and directories varies with your software installation and licensed databanks.

A

```
%ls -F dblk_eval
```

```
MVISION.UNT@           ge_plastics.mvform
fatigue.des@          ge_plastics.unt@
fatigue.dis@          materials_selector.General.mvform@
fatigue.index@        materials_selector.address.mvform@
fatigue.mvform@       materials_selector.coatings.mvform@
fatigue.unt@          materials_selector.composites.mvform@
fiber.des@             materials_selector.des@
fiber.dis@             materials_selector.des.mapping@
fiber.index@           materials_selector.dis@
fiber.mvform@          materials_selector.fabrication.mvform@
form.definitions@     materials_selector.index@
ge_plastics.des@      materials_selector.metals.mvform@
ge_plastics.des.mapping@ materials_selector.plastics.mvform@
ge_plastics.dis@      materials_selector.testing.mvform@
ge_plastics.index@    materials_selector.unt@
```

Sample mvinstall_db Session (Continued)

Installing Your Own Databanks

It is recommended that you store your databanks and their auxiliary files (i.e. units conversion files, mapping files, etc.) in a separate directory from the MSC/MVISION Databanks. This prevents alteration or overwriting your files when updating the MSC/MVISION Databanks. It also allows you to append your `form.definitions` file to the master `form.definitions` file. See *Append an Existing form.definitions file*, on page A-38.

To maintain your databanks in directories separate from the MSC/MVISION Databanks, store them in the `<db_install_dir>/mvision_db` in a sub-directory named `db_<user>`. Install them using the `mvinstall_db` program.

The `mvinstall_db` program is automatically copied to the `<db_install_dir>/mvision_db/maint` directory when the `mcssetup` script is run to unload the MSC/MVISION Databanks. Use it to update databanks and new releases without deleting previously installed files.

1. Change directories to `<db_install_dir>/mvision_db`:

```
cd <db_install_dir>/mvision_db
```
2. Create a new directory having a name prefixed by `db_`:

```
mkdir db_<user>
```
3. Change directories to `<db_install_dir>/mvision_db/<db_user>`:

```
cd db_<user>
```
4. Place all databanks and any auxiliary files you have created into this new directory. You can use the copy command:

```
cp ~user/pathname/<user_databank>.* .
```
5. Run the `mvinstall_db` program from the `<db_install_dir>/mvision_db` directory to "install" your databanks:

```
cd ..  
maint/mvinstall_db
```

The `install` program will include a listing of all the `db_*` directories in a numbered sequence. Choose the number of the `db_<user>` directory for installation at this time. An example of typical `mvinstall_db` session is presented in *Sample*

mvininstall_db When Installing Your Own Databanks, on page A-26.

For <databank>.des.mapping files to be linked to the correct software product, the file name must have the extension .eval or .pro as defined in *Directories and Files Created During Installation*, on page A-5

% **maint/mvininstall_db**

MacNeal-Schwendler Corporation

Customer Delivery Program - mvininstall_db

Version 4.0

Platform SGI5

The following databank releases are available:

```
# Databank Name
== =====
1 4q98
2 user
```

Choose a release (1-2)
or q to quit the databank install.

==>2

Pick the MSC/MVISION products that are installed on this system.
Type a 'c' when you are done selecting products.

```
# Link Product Name
== =====
1 No MSC/MVISION 1.2A
2 No MSC/MVISION Builder/Evaluator 3.x
3 No MSC/MVISION Pro 2.0/15
4 No MSC/PATRAN (P3 MATERIALS SELECTOR)
```

Enter a product number (1..4) or
'c' - Continue with installation
'q' - Quit without performing the installation

CHOICE -> 2

Sample mvininstall_db When Installing Your Own Databanks

```
# Link Product Name
== =====
1 No MSC/MVISION 1.2A
2 Yes MSC/MVISION Builder/Evaluator 3.x
3 No MSC/MVISION Pro 2.0/15
4 No MSC/PATRAN (P3 MATERIALS SELECTOR)

Enter a product number (1..4) or
'c' - Continue with installation
'q' - Quit without performing the installation
```

CHOICE -> **C**

```
-----
Skipping setup of MSC/MVISION 1.2A
-----
```

```
Building the dblk_eval directory for
MSC/MVISION Builder/Evaluator 3.x from db_user
.....
```

```
Building the form.definitions file and creating a
link to it in dblk_eval.
```

```
Searching the directory dblk_eval for form files
that match the databanks in the directory dblk_eval.
.....
```

```
Setting a link for MSC/MVISION Builder/Evaluator 3.x.
```

```
Enter the path to the MSC/MVISION Builder/Evaluator 3.x installation.
(Press Return with no path to skip this operation.)
```

->

```
Skipping the link for MSC/MVISION Builder/Evaluator 3.x.
-----
```

```
Skipping setup of MSC/MVISION Pro 2.0/15
-----
```

```
Skipping setup of MSC/PATRAN (P3 MATERIALS SELECTOR)
mvininstall_db done.
```



Sample mvininstall_db When Installing Your Own Databanks (continued)

Updating the form.definitions Files

MSC/MVISION Builder and Evaluator use the `form.definitions` file to customize the Materials Browser interface. This file is a compilation of individual form files created for each databank and named `<databank_name>.mvform`.

When this documentation refers to a databank form file, it is referring to `<databank_name>.mvform`.

MSC/MVISION Builder and Evaluator is provided with a script called `mvform` which automatically converts one or more `<databank_name>.mvform` files to a `form.definitions` file. This script is run each time `mvinstall_db` is used to install MSC/MVISION Databanks or to update an installation. See *Create a New form.definitions File*, on page A-36 for details of running the `mvform` script.

Since MSC/MVISION Builder and Evaluator read a databank form only from a `form.definitions` file, a `<databank_name>.mvform` file should be created for each databank developed at your site. This section presents the various ways you can create your own `<databank_name>.mvform` files and build the `form.definitions` file.

Creating Form Definitions for Your Databanks

When an MSC/MVISION databank has been created from your own data sources, there are several options for creating forms (`<databank_name>.mvform` files) for the databank:

- Write a `<databank_name>.mvform` file using the format described in the *MSC/MVISION Builder and Evaluator User's Guide and Reference*.
- Create the `<databank_name>.mvform` file from the default form. The default form is automatically built for a databank if a form does not exist in the `form.definitions` file or current forms listing at the time the databank is opened during an MSC/MVISION session. It is saved in the `$HOME/.mvision/<product>/settings/Forms` file after exiting the MSC/MVISION session or can be saved in a notebook. To convert the default form to a `<databank_name>.mvform` file, use one of the following procedures:

PROCEDURE 1:

1. Modify the default form using the **Edit** option in the MSC/MVISION Builder and Evaluator user interface.
2. At completion of the session, navigate to the `$HOME/.mvision/<product>/settings` directory. Copy the `Forms` file to a file named `$HOME/<databank_name.mvform>` to prevent it from being overwritten in subsequent sessions.
3. Open the file and remove the top line:

```
VERSION: "3.2";
```

Forms for other databanks will probably be saved in this file in addition to the default form created for your databank. Delete the other forms. The only portion that should remain in the `<databank_name>.mvform` file is the section beginning with:

```
FORM: "<Form_name> - <Your_Databank_Name>";
```

and ending with:

```
END_FORM.
```

You now have a `<databank_name.mvform>` file for your databank.

To prevent other databank forms from being included in the `$HOME/.mvision/<product>/settings/Forms` file, place a file named `form.definitions` with initial line reading "Version: "3.2"; in your start up directory.

PROCEDURE 2:

1. In a local directory create the following awk script, which will separate any `form.definitions` file or `Forms` file into its component `<databank_name>.mvform` files. Using your editor, create a file named `mvform_dump`, which contains the following lines:

```
# explode form.definitions and write mvform files
BEGIN { i=1 }
/^FORM/ {
    FILE = sprintf("%02d.mvform",i);
    print > FILE;
    do {
        getline;print >> FILE ;
    } while ( $1 != FORM)
    i++;
    close ( FILE );
}
```

"mvform_dump" awk Script

A

2. Make the file executable using the following command:


```
chmod +x mvform_dump
```
3. In the same directory, issue the following command at the UNIX prompt:


```
awk -f mvform_dump $HOME/.mvision/<product>/settings directory/Forms
```

 or


```
nawk -f mvform_dump $HOME/.mvision/<product>/settings directory/Forms
```

The individual forms will be placed in files named nn.mvform, where nn is the numerical order in which the form appeared in the Forms file.
4. Scan the nn.mvform files to determine which one corresponds to your databank and rename it <databank_name>.mvform.

It is recommended that you store the <databank_name>.mvform file in the <install_db>/db_<user> directory with your databank and all other databank auxiliary files. See *Installing Your Own Databanks*, on page A-25.

The <databank_name>.mvform is converted to a form.definitions file using the methods outlined in *Running the mvform Script*, on page A-34. It can also be created by copying the file to form.definitions and adding the following line to the top of the file:

```
VERSION: "3.2";
```

Search Path for form.definitions File

It is important to understand the search path for form.definitions files, because it affects where the forms for a databank are stored for a particular end use. MSC/MVISION Builder and Evaluator searches for the form.definitions file as follows:

1. In \$MV_PATH, set by default to the current directory, and then the user's home directory (\$HOME).
2. In the .mvision/<product>/settings directory. (If it does not find a Forms file in the settings directory, then the .mvision/<product>/settings directory has not yet been created.)
3. In software resources directory.
4. In <db_install_dir>/db. MSC/MVISION Builder and Evaluator creates the .mvision/<product>/settings directory and copies the form.definitions file in the databank <db_install_dir>/db to the .mvision/<product>/settings/Forms file if the Forms file does not exist.

This results in the following implications:

- If <databank_name>.mvform files are created for a databank used corporate-wide, they must be included in a master form.definitions file in the <db_install_dir>/db directory. See *Corporate-wide Forms*, on page A-33.
- If the <databank_name>.mvform files are created for a databank, but their usage is isolated to a group of end-users, it should be placed in a form.definitions file in the users' current or home directory (\$HOME). It is then accessed each time MSC/MVISION Builder and Evaluator is started from the current directory. However, if a user chooses to open a databank stored in the <db_install_dir>/mvision_db directory (i.e. an MSC/MVISION Databank), MSC/MVISION Builder and Evaluator will create a default form for use during that session for any databank that does not have a form in the local form.definitions file.

If the end-user wants to use the master form.definitions file and modify it to meet his needs, he has three options:

- Option 1:



1. Verify that there are no `form.definitions` files in his current directory or his home directory.
2. Rename the `$HOME/.mvision/<product>/settings` directory.
3. Start an MSC/MVISION session, open the databank, and modify the forms in the user interface using the **Edit** options.

At the conclusion of the MSC/MVISION Builder and Evaluator session, the modifications are saved in the `.mvision/<product>/settings/Forms` file and are available during subsequent sessions. To save this version permanently, rename the `Forms` file or copy it to another directory.

- Option 2:

1. Verify that there are no `form.definitions` files in the user's current directory or his home directory.
2. Remove or rename the `$HOME/.mvision/<product>/settings` directory.
3. Start an MSC/MVISION session, open the databank, and modify the forms in the user interface using the **Edit** options.
4. Save the session to a notebook.

The edited forms are saved in a `.mvision/<product>/notebook/Forms` file or you can save the notebook to a name of your choice. This option offers the most flexibility. The modified forms are then available only when an MSC/MVISION Builder and Evaluator session is initiated using the command:

```
mveval  
mvbuild -notebook <full_pathname>/<notebook_file>
```

- Option 3:

1. Copy the `form.definitions` file from the `<db_install_dir>/mvision_db/form_definitions_dir` directory to your current or home directory.
2. Open the file and modify it using the text editor. Refer to the *MSC/MVISION Builder and Evaluator User's Guide and Reference*.

Corporate-wide Forms

If you have created your own databank and are currently using a `form.definitions` file specific to that databank, you may want to extend its use corporate-wide. You can do this by appending it to the list of forms available when the user accesses MSC/MVISION Databanks.

Use the following steps:

1. Copy or rename the `form.definitions` file for your databank to `<databank_name.mvform>`:

```
cp form.definitions <databank_name.mvform>
```

or

```
mv form.definitions <databank_name.mvform>
```

2. Use the `mvform` script to append the form to the MSC or master `form.definitions` file. See *Append an Existing form.definitions file*, on page A-38.

This process requires write permission to the directory in which the `form.definitions` file is located. It is recommended that the databanks and auxiliary files be stored in the `<db_install_dir>/mvision_db/db_<user>` directory. See *Installing Your Own Databanks*, on page A-25.

Updating the Forms File in the Settings Directory

After the MSC/MVISION Databanks have been updated and a new `form.definitions` file is created, the users **must** be notified of the changes. When the databanks are updated, the forms are often modified, enhanced, or corrected. However, because MSC/MVISION Builder and Evaluator writes a copy of the `form.definitions` file to each user's `.mvision/<product>/settings/Forms` file, the new databanks are accessed using these forms (from the previous release). This could result in numerous error messages and possible internal system errors.

To access the new forms and save changes made to old forms, it is recommended that the user perform the following:

1. Navigate to the `.mvision/<product>/settings` directory and rename the `Forms` file.

```
cd ~$HOME/.mvision/<product>/settings
```

```
mv Forms Forms_old
```

The next session of MSC/MVISION Builder and Evaluator will then be started with the new `form.definitions` file, that will be copied to the `.mvision/<product>/settings/Forms` file at completion of the session.

2. Append the old `Forms` file to the new one by issuing the following command in the `.mvision/<product>/settings` directory:

```
cat Forms_old >> Forms
```

3. Open the `Forms` file. The databank forms in the `Forms_old` file are at the end of the `Forms` file. Review the changes made to the old forms and copy them into the new ones.
4. Go to the end of the file and delete the duplicate or unwanted forms. Verify that the file contains only one `VERSION` line and that it is the first line of the file:

```
VERSION: "3.2";
```

If the user is not familiar with the format of the `Forms` file, each databank form begins with the heading:

```
FORM: "<Form_name> - <Databank_Name>";
DATABASE: "<full_path_name>/<databank_name>.des";
```

It ends with the line:

```
END_FORM;
```

The text in between these two lines contains the commands that define the Criteria Panel and Material Sets for the edited version of the form. The forms the user wants to review for addition to the new forms will be identified by the `<Form_name>` and `DATABASE`. A `form.definitions` file must not contain two forms with identical form names.

Running the mvform Script

The `mvform` script is used to:

- Create a new `form.definitions` file from a set of `<databank_name>.mvform` files.

This should be done each time the MSC/MVISION Databanks are updated. Run the `mvform` script once for each databank installation (i.e. each `db_*` directory) at your site.

- Append a set of one or more `<databank_name>.mvform` files to an existing `form.definitions` file.

To do this, the `form.definitions` file must reside in (or be linked to) the same directory as the `<databank_name>.mvform` files.

The script requires:

- The complete path to the `<databank_directory>` and the databank `<forms_directory>`.
- That all the databanks reside in one `<databank_directory>` and all the forms (`<databank_name>.mvform` files) reside in one `<forms_directory>`. The `<databank_directory>` and `<forms_directory>` may be the same or different directories.

The `mvform` script may be used to create one `form.definitions` file from forms (or databanks) that reside in different directories. You must rerun the script for each directory using the append command. See *Append an Existing form.definitions file*, on page A-38.

Using the *mvform* Script

The `mvform` script has built-in command line options designed to streamline the compilation process by defining program parameters prior to running the script. They are entered on the command line when the script is invoked.

The following options and their usage can be viewed by issuing the command:

```
mvform -h
```

```
Usage: mvform [-db databank_dir] [-form mvforms_dir] [form_def_file]
          [-all] [-replace/-append]
          [-h] [-debug]
```

Where:

```
-db databank_dir - Name of the databank directory. The script
                  cannot make forms for databanks that are not in
                  the databank directory.
-form mvforms_dir - Name of the mvforms directory where all of the
                  individual databank forms to be compiled exist.
form_def_file    - Name of the form definitions file. Default is
                  form.definitions
-all            - Builds the form.definitions file from all the
                  <databank_name>.mvform files found in the
                  <db_install_dir>/db (databank) directory.
-replace        - Replaces an existing form_def_file.
-append         - Appends to an existing form_def_file.
-h              - Print this help message.
-debug          - Activates the debug mode (verbose).
```

mvForm Script Options

Create a New form.definitions File

To create a `form.definitions` file from a set of `<databank_name>.mvform` files, type the following command in a C shell window:

mvform

Use the subsequent dialog to input the full path names to the databank and form directories.

```
Enter the path to the databank directory.
(Default is the <current_directory>.)
```

```
--->
```

```
Enter the path to the databank form directory.
(Default is the <current_directory>/mvform.)
```

```
--->
```

```
Searching <forms_directory> for form files that match the databanks
in <databank_directory>.
```

The program compiles a list of the <databank_name>.mvform files in the <forms_directory>. Select those to be placed in the form.definitions file by entering the number next to the <databank_name>.mvform file name.

The following databank forms are available:

```
# Add Databank Form Name
== --- =====
1 No <databank_name>.mvform
2 No <databank_name>.mvform
```

Enter a form number (1..1) to toggle ADD flag or

A - to add all forms.

E - to erase all selections.

Q - to quit without making a form.definitions file.

C - to create the form.definitions file with current selections.

CHOICE -> **A**

The program confirms the selection by printing Yes in the Add column. To create the form.definitions file from this selection, type C at the prompt. To change the selection, type the number of the form to be removed at the prompt.

The following databank forms are available:

```
# Add Databank Form Name
== --- =====
1 Yes <databank_name>.mvform
2 Yes <databank_name>.mvform
```

Enter a form number (1..1) to toggle ADD flag or

A - to add all forms.

E - to erase all selections.

Q - to quit without making a form.definitions file.

C - to create the form.definitions file with current selections.

CHOICE -> **C**



If there are no databanks for the specified forms in the designated directory, the script returns the following error message:

```
Searching <form_directory> for form files
that match the databanks in <databank_directory>.
```

```
No action taken.
```

```
No form files were located in <form_directory> which matched
the databanks found in <databank_directory>.
```

Append an Existing form.definitions file

When a form.definitions file already exists and you want to append the forms for your databanks to that form, type the following command in a C shell window:

```
mvform
```

Remember to type in the full path name as requested.

```
Enter the path to the databank directory.
```

```
(Default is the <current_directory>.)
```

```
--->
```

```
Enter the path to the databank form directory.
```

```
(Default is the <current_directory>/mvform.)
```

```
--->
```

```
Searching <databanks_directory> for form files that match the
databanks in <forms_directory>.
```

A list of the forms in the <forms_directory> is compiled.
Select those to be included in the form.definitions file.

The following databank forms are available:

```
# Add Databank Form Name
== --- =====
1 No <databank_name>.mvform
2 No <databank_name>.mvform

Enter a form number (1..) to toggle ADD flag or
A - to add all forms.
E - to erase all selections.
Q - to quit without making a form.definitions file.
C - to create the form.definitions file with current selections.

CHOICE -> A
```

The software confirms the selection by printing *Yes* in the Add column. If this selection is correct, type **C** at the prompt to append the forms to form.definitions file. To change the selection, type the number of the form to be removed at the prompt.

The following databank forms are available:

```
# Add Databank Form Name
== --- =====
1 Yes <databank_name>.mvform
2 Yes <databank_name>.mvform

Enter a form number (1..) to toggle ADD flag or
A - to add all forms.
E - to erase all selections.
Q - to quit without making a form.definitions file.
C - to create the form.definitions file with current selections.

CHOICE -> C
```

Because the program locates a form.definitions file in the same directory, it offers the options to replace or append the



form.definitions file. Type **A** at the prompt to append the new forms to the existing form.definitions file.

The form.definitions already exists.

Do you want to replace the existing file or append to the old file?

Enter:

R - to replace form.definitions.

A - to append to form.definitions (Default).

RESPONSE ->**A**

A

If you choose to replace the existing form.definitions file, the current form.definitions file is renamed before the new form.definitions file is created.

Removing Files or Databanks from Your System

MSC/MVISION Databank releases contain the databanks and all files required to support each MSC/MVISION software product. Your software configuration may not require all of these files. Although most are small, deleting them from your system could save valuable disk space. If you upgrade to a new product in the future, they may be reinstalled from the original CD ROM.

Refer to your software operations manuals to determine which files are required for your configuration and compare it to the list of files in *Directories and Files Created During Installation*, on page A-5. Since the `<software_install_dir>/db` directory links to the actual location of these files, it is best to remove the files from both the `<db>` and `<db_*>` directories.

All software products require units conversion files and disclaimer files in addition to the databank files. Do not delete any files with the suffixes `<.unt>`, `<.UNT>`, `<.dis>` or `<.des>` if the related databank is installed on the system.

Position yourself in the `<db_install_dir>/mvision_db` and change directory to the `<db_xqxx>` directory.

```
cd <db_install_dir>/mvision_db/<db_xqxx>
```

For example, delete unwanted files using the Unix remove command (`rm`):

- | | |
|-----------------------------------|---|
| <code>rm *.map</code> | Removes files that specifically support MSC/MVISION Pro. |
| <code>rm *.mapping.eval</code> | Removes the export mapping files for MSC/MVISION Evaluator and Builder. |
| <code>rm *.mapping.pro</code> | Removes export mapping files for MSC/MVISION Pro. |
| <code>rm *.mvtm</code> | Removes the export mapping files for MSC/PATRAN Materials. |
| <code>rm forms.definitions</code> | Removes the compiled form file for MSC/MVISION Evaluator and Builder. |
| <code>rm -r .mvform</code> | Removes individual form files for MSC/MVISION Builder and Evaluator. |

rm *.dis	Removes disclaimer files for the MSC/MVISION databanks.
rm *.unt	Removes units conversion files for the MSC/MVISION databanks.

After the files have been removed from the <db_xqxx> directory, change directories to the <db> directory and reissue the same commands. If you don't, links to files that do not exist will remain. If deleted files are databanks, they remain in the list of available databanks when using the **File-Open Databank** command in MSC/MVISION Builder and Evaluator, but can not be opened.

Previous releases of the MSC/MVISION Databanks may be removed globally from the <db_install_dir>/mvision_db directory using the Unix command:

```
rm -rf <db_xqxx>
```

This command should be used with caution, having fully investigated the contents of that directory and the db directory, because the directory is removed inclusive of all its files. Any links in the db directory to the files in the removed directory become "stale" links since the actual file no longer exists.
