Marc® and Mentat® 2021.3

Installation and Operations Guide

HEXAGON | MSC Software

Corporate
MSC Software Corporation
5161 California Ave., Suite 200
University Research Park
Irvine, CA 92617
Telephone: (714) 540-8900
Email: americas.contact@mscsoftware.com

Europe, Middle East, Africa
MSC Software GmbH
Am Moosfeld 13
81829 Munich, Germany
Telephone: (49) 89 431 98 70
Email: europe@mscsoftware.com

Japan
MSC Software Japan Ltd.
KANDA SQUARE 16F
2-2-1 Kanda Nishikicho
Chiyoda-ku
Tokyo 101-0054, JAPAN
Telephone: (81) (3)-6275-0870
Email: MSCJ.Market@mscsoftware.com

Asia-Pacific
MSC Software (S) Pte. Ltd.
100 Beach Road
#16-05 Shaw Tower
Singapore 189702
Telephone: 65-6272-0082
Email: APAC.Contact@mscsoftware.com

Worldwide Web
www.mscsoftware.com

Support
http://www.mscsoftware.com/Contents/Services/Technical-Support/Contact-Technical-Support.aspx

Disclaimer

MA:V2021.3:Z:Z:Z:DC-IOG-PDF

Documentation Feedback

At MSC Software, we strive to produce the highest quality documentation and welcome your feedback. If you have comments or suggestions about our documentation, write to us at: documentation-feedback@mscsoftware.com.

Please include the following information with your feedback:

- Document name
- Release/Version number
- Chapter/Section name
- Topic title (for Online Help)
- Brief description of the content (for example, incomplete/incorrect information, grammatical errors, information that requires clarification or more details and so on).
- Your suggestions for correcting/improving documentation

You may also provide your feedback about MSC Software documentation by taking a short 5-minute survey at: http://msc-documentation.questionpro.com.

Note:	The above mentioned e-mail address is only for providing documentation specific feedback. If you have any technical problems, issues, or queries, please contact Technical Support.

Contents

Marc and Mentat: Installation and Operations Guide

SECTION 3: APPENDICES

Preface

About This Guide

This *Marc and Mentat Installation and Operations Guide* contains information about installing and running Marc, Mentat and its components. You will find references to following documents throughout this guide.

Marc Documentation

TITLE	VOLUME
Theory and User Information	Volume A
Element Library	Volume B
Program Input	Volume C
User Subroutines and Special Routines	Volume D
Demonstration Problems	Volume E
Release Guide	
Python: Tutorial and Reference Manual	

Purpose of This Guide

This guide explains the procedure for installing Marc and Mentat. It also describes how to install the associated documentation. This purpose of this guide is to:

- Help you install the licensing server on Windows and Linux platforms.
- Help you install Marc and Mentat on Windows and Linux platforms.
- Identify and ensure that the installation is successful.
- Provide you with basic troubleshooting.
- Provide you information about files, directories, and their location in the installed folders.
- Inform you about using Marc and Mentat on parallel networks.

Contents of This Guide

The principal categories of information are found under the following titles:

Chapter	Title	Description
SECTION 1: MICROSOFT WINDOWS		
Chapter 1	Microsoft Windows: Prerequisites for Marc and Mentat	Describes the prerequisites required for the installation and usage of the Marc and Mentat programs on Microsoft Windows platforms.
Chapter 2	Microsoft Windows: Marc and Mentat Installation	Provides a detailed installation procedure of Marc and Mentat on Microsoft Windows platforms The instructions provided in this chapter require a basic knowledge of the machine on which you are loading the Marc products. Also contains a section with details about the installation procedure and a section about the license management utility.
Chapter 3	Microsoft Windows: Running and Using Marc and Mentat	Explains running and usage of the Marc and Mentat for Microsoft Windows platforms. Sections related to Mentat external programs and making changes to MPI settings are also discussed in this chapter.
Chapter 4	Microsoft Windows: Marc Parallel Network	Provides the general description about the hardware and software requirements and definitions.Provides a step-by-step approach to installation of the network version. Provides miscellaneous information about executing a parallel job over a network and use of user subroutines.
Chapter 5	Windows Troubleshooting	This chapter contains information about troubleshooting general and problems.
SECTION 2: LINUX		
Chapter 6	Linux: Prerequisites for Marc and Mentat	Contains information regarding the necessary prerequisites (related to hardware and software) for installing and running Marc and Mentat. There is also a section which informs us about the supported platforms by this version.
Chapter 7	Microsoft Windows: Marc and Mentat Installation	Detailed procedure about installing Marc and Mentat on Linux platforms is explained here
Chapter 8	Linux: Running and Using Marc and Mentat	In this chapter running and using Marc and Mentat is described. Sections such as modifying MPI settings and Mentat interfaces are also discussed in this chapter.
Chapter 9	Linux: Marc Parallel Network	Provides the general description about the hardware and software requirements and definitions.Provides a step-by-step approach to installation of the network version. Provides miscellaneous information about executing a parallel job over a network and use of user subroutines.

Chapter	Title	Description
Chapter 10	Linux Troubleshooting	This chapter contains information about troubleshooting general and problems.
SECTION 3: APPENDICES		
Appendix A	Microsoft Windows: Marc Subdirectories and Installation	Contains information regarding Marc subdirectories and the old installation procedure for windows platform
Appendix B	Microsoft Windows: Mentat Files and Subdirectories	Mentat files and subdirectories are mentioned in this file.
Appendix C	Linux: Marc/Mentat Files, Subdirectories and Installation	Marc/Mentat files, subdirectories and the old installation procedure for Linux platform are briefly described.

Typographical Conventions

The section provides a brief overview of the typographical conventions used in the document to help the user better follow the *Marc and Mentat* documentation.

This section describes some syntax that will help you in understanding text in the various chapters and thus in facilitating your learning process. It contains stylistic conventions to denote user action, to emphasize particular aspects of Marc and Mentat to signal other differences within the text.

Adobe Garamond Pro	Body and general text
Courier New	■ Represents command-line options of Marc and Mentat. ■ Directory names and paths ■ File names and Paths ■ Linux terminal script Example: `lmreread -c <parent>/msc/MSC.Licensing/licenses/license.dat`
Bold Text	■ Highlights ■ Dialog box names ■ Buttons ■ Menus ■ User inputs ■ The commands/user inputs for all descriptions related to terminal commands. ■ Default values Example: `[root@vm-tmrhel73 MSC]#` **./msc_licensing_helium_linux64.bin**
HelveticaNeueLT Pro Cn 57	■ Hyperlinks ■ Weblinks Example: Appendix A: Microsoft Windows: Marc Subdirectories and Installation

Italic Text	Represents references to books. Example: *Volume A:Theory and User Information*
20XX	Represents the latest version number.

Accessing Marc Manuals

This section describes how to access the Marc documentation outside of MSC Software. Marc documentation is available through PDF files. The PDF files can be obtained from the following sources:

- Marc documentation installer
- SimCompanion
- Combined documentation

The PDF documentation files are appropriate for viewing and printing with Adobe Acrobat Reader (version 5.0 or higher), which is available for most Windows and Linux systems. These files are identified by a .pdf suffix in their file names.

Downloading the PDF Documentation Files

You can download the PDF documentation from SimCompanion (http://simcompanion.mscsoftware.com).

Navigating the PDF Files

For the purpose of easier online document navigation, the PDF files contain hyperlinks in the table of contents and index. In addition, links to other guides, hyperlinks to all cross-references to chapters, sections, figures, tables, bibliography, and index entries have been applied.

Printing the PDF Files

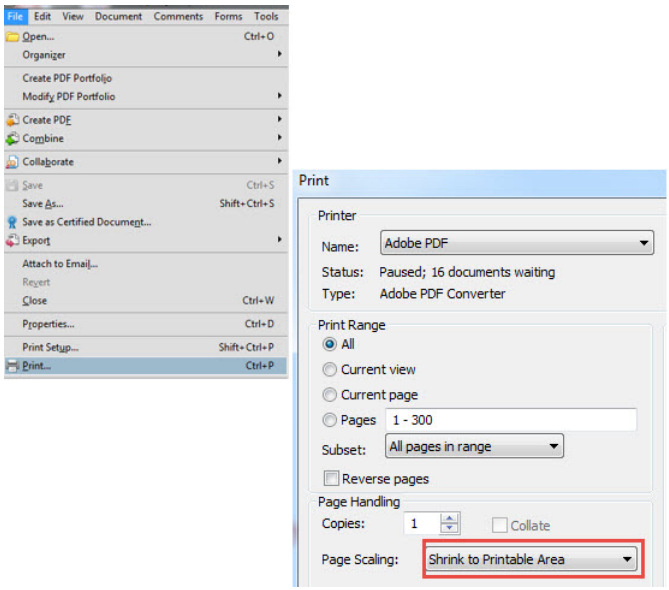
Adobe Acrobat PDF files are provided for printing all or part of the manuals. You can select the paper size to which you are printing in Adobe Acrobat Reader by doing the following:

1. Click **File**.
2. Select the **Print**.... option. The **Print** dialog box is displayed.
3. Select **Page Setup**....
4. Choose the required paper size in the **Page Setup** menu.

The PDF files are recommended when printing long sections since the printout will have a higher quality.

If the page is too large to fit on your paper size, you can reduce it by doing the following:

1. Select the **File -> Print**.
2. Under **Page Scaling,** choose the **Shrink to Printable Area** option.



Training and Internet Resources

Information about MSC's products, services and latest events is available on our website www.mscsoftware.com.

The information about MSC Seminars is available on the training link http://www.mscsoftware.com/msc-training. You can also use this link to schedule the seminars.

If you are a new Marc user, we recommend the following courses:

MAR101 - Basic Nonlinear Analysis using Marc and Mentat

The purpose of this course is to introduce the new Marc user to both Marc and Mentat by lectures and hands on modeling of nonlinear problems.

Pre-requisites:

A basic knowledge of statics and strength of materials is highly recommended. Previous finite element analysis experience is recommended.

Topics:

- Introduction to Mentat
- Nonlinear Finite Element Analysis
- Resolving Convergence Problems
- Numerical Analysis of Nonlinear Problems

MAR102 - Advanced Nonlinear Analysis using Marc and Mentat

The purpose of this course is to enhance the current Marc user's understanding of modeling nonlinear problems. Lectures are supported by hands-on modeling of nonlinear problems.

Pre-requisites:

A basic knowledge of nonlinear simulations - Familiarity with Mentat 2011 - Completion of MAR101 (Basic Nonlinear Analysis using Marc and Mentat) or equivalent experience.

Topics:

- Material Nonlinearity
- Contact
- Adaptive Meshing
- User Subroutines in Marc
- Heat Transfer and Thermal Stresses
- Global - Local (Structural Zooming) Analysis in Marc
- Restarts
- Performance
- Workshop Problems

MAR103 - Experimental Elastomer Analysis

The purpose of this course is to provide a fundamental understanding of how material testing and finite element analysis are combined to improve the design of rubber and elastomeric products.

Pre-requisites:

A basic knowledge of statics and strength of materials is highly recommended. Previous finite element analysis experience is recommended. And the knowledge of elastomeric materials.

Topics:

- Introduction
- Overview of Elastomer Testing and Analysis
- Uniaxial Tension/Compression Testing and Analysis
- Biaxial Tension/Compression Testing
- Pure Shear Testing
- Product Simulations with Specimen Data

MAR120 - Basic Nonlinear Analysis using Marc and Patran

MAR120 covers the use of Marc and Patran or AFEA (the interlocked combination of Patran and Marc) for the solution of complex engineering problems. Students who successfully complete this course will be able to: create finite element models representing nonlinear physical phenomena; select appropriate element types and mesh densities; understand the limitations of solving nonlinear FEA problems; select solution types for various nonlinear phenomena such as nonlinear

dynamics, metal forming, elastomers, and contact problems; select error tolerance parameters and properly use automatic time-stepping techniques; and understand the basis of large deformation, rotation, and strain finite element analysis. Patran provides a Marc Preference which directly supports most Marc features and indirectly supports all Marc features. MSC customers that have been using Advanced FEA (which is replaced by AFEA) for meeting their analysis needs will find this new Marc Preference to be the ideal environment to continue their work. They are especially encouraged to attend this course. All the class practice (16 exercises) is made using Patran and Marc rather than Marc and Mentat. Engineers who have attended the MAR101 and MAR102 will also benefit from attending this class if they intend to use the Patran Marc Preference.

Pre-requisites:

A basic knowledge of statics and strength of materials is highly recommended and previous finite element analysis experience is recommended.

MAR121 - Advanced Nonlinear Analysis using Marc and Patran

The purpose of this course is to enhance the current Marc user's understanding of modeling nonlinear problems. Lectures are supported by hands-on modeling of nonlinear problems.

Pre-requisites:

MAR120 - Basic Nonlinear Analysis using Marc and Patran

Topics:

- Expand knowledge from MAR120 (Basic Nonlinear Analysis using Marc and Patran) course
- Practical aspects of rubber simulation
- Creep
- Superplastic forming
- Composite failure techniques
- Advanced contact techniques
- Adaptive meshing
- User subroutines
- Global/Local modeling
- Heat transfer and thermal stress
- Coupled Thermal/Structural analysis
- Restarts
- Performance
- Workshop Problems

Technical Support

If you encounter difficulties while using Marc, first please refer to the section(s) of the manual containing information on the commands you are trying to use or the type of problem you are trying to solve.

Visit SimCompanion

The product documentation is available in SimCompanion (http://simcompanion.mscsoftware.com). The SimCompanion gives you access to a wealth of resources for MSC Software products. You will find various information such as:

- Product documentations
- Knowledge base articles
- Product error lists (fixed and known issues for each release)
- SimAcademy webinars
- Product and support contact information

SimCompanion is a searchable database which allows you to find articles relevant to your inquiry. Valid MSC customer entitlement and login is required to access the database and documents. It is a single sign-on that gives you access to product documentation for complete list of products from MSC Software, allows you to manage your support cases, and participate in our discussion forums.

Help Us Help You

Clients frequently call up the support engineers at MSC Software with enquiry regarding models that do not run correctly. Our technical support staff can help you much more efficiently and effectively if you are working with a small model, since debugging a small model is much easier, and the turnaround time to rerun a (hopefully) corrected test model is minutes rather than hours.

- For information on the latest events, products and services for all products, refer to the MSC Software corporate site (www.mscsoftware.com).
- For technical support phone numbers and contact information, please visit: http://www.mscsoftware.com/Contents/Services/Technical-Support/Contact-Technical-Support.aspx.

SECTION 1: MICROSOFT WINDOWS

1 Microsoft Windows: Prerequisites for Marc and Mentat

The requirements of Marc and Mentat are as in the following table:

Table 1-1 Requirements of Marc & Mentat

Operating System	Windows 10, or Windows 2016 Server.
CPU	X86-64
Graphics Card	SVGA or better running in at least 16 bit (64k) color mode
Hard Drive	Minimum: ■ 950 MB Marc ■ 800 MB Mentat ■ 350 MB documentation
DVD Drive	Required for media based installation. Not required for electronic download installation.
Ethernet Card	An ethernet card is required. Also, Microsoft TCP/IP Service must be installed.
Mouse	Three button mouse is recommended
Memory	Minimum 8 GB Recommended 16 GB
Fortran Compiler	A Fortran compiler is necessary if user subroutines are to be used. For other cases, no compiler is needed. The compiler needs to be compatible with the one used in the Marc build. For a list of supported compilers, refer to the *Marc and Mentat Release Guide*.

The instructions provided in this chapter require a basic knowledge of the machine on which you are loading the Marc products. No attempt is made to teach the use of Microsoft Windows commands.

This document contains a section containing details about the installation procedure, a section concerning the usage of the Marc and Mentat programs and a section about the license management utility.

Troubleshooting related hints are provided in Chapter 5: Windows Troubleshooting.

Note:	If you encounter a problem during installation, please contact the customer support staff at the nearest MSC Software office.

Installation Prerequisites

Before installing the software	The product is available for download at the MSC Solutions Download Center available at https://mscsoftware.subscribenet.com 1. Select the product **Marc** and select the 20XX version on the next level. 2. Download the installer for your platform. The FLEXlm security is installed separately. 3. Select the product **MSC Licensing** and the version Helium on the next level. 4. Download the installer for your platform. 5. Decide where you want the product to be installed and where to perform the installation. Marc requires approximately 950 Mbytes of permanent disk storage capacity. Mentat requires approximately 800 Mbytes of permanent disk storage capacity. Documentation, stored in the separate documentation directories `doc` and `examples`, contains approximately 350 Mbytes of data. It is preferred to first install security, then Marc and/or Mentat and finally the documentation.
Password protection	The Marc and Mentat version you have received is protected against illegal usage by means of Flexera Software's FLEXlm licensing software. You *cannot* run the program directly after you have installed the product from the installation media until you obtain a license file from MSC Software Corporation.
Should I be Administrator?	You need not to be logged in as *Administrator*. However, you will need administrator privileges since the system registry will be updated. Also, check that you have read and write permissions to the installation directory.
Fortran compiler	A Fortran compiler is necessary if you will use user subroutines. For other cases, no compiler is needed. **Note:** See Chapter 5: Windows Troubleshooting for important information regarding requirements for Intel Fortran compiler.
Computer Name	Your machine **must** have a computer name (host name). If no computer name is known, supply one by using the **Control Panel\Network** applet to set the **Computer Name**. You should also make sure the **Host Name** specified in the DNS tab of **Network\Protocols\TCP/IP Protocol** is the same as the "Computer Name".
Previous Versions	If you have previous versions of Marc and/or Mentat installed, you may want to adjust your PATH environment variable to remove the reference to the previous version.
NVIDIA driver	An NVIDIA driver is necessary if the GPGPU capability is to be used on Windows 64-bit machines. For other cases, no special driver is needed. The driver needs to be compatible with the one used in the Marc build; see *Marc Volume A: Theory and User Information*, Chapter 13 for the minimum driver version supported.

Supported Platforms

The supported Windows platforms are listed below:

Type	OS	Hardware	Fortran Version	Default MPI
Windows (64-bit)	Windows 10 Windows 2016 Server	Intel EM64T or AMD Opteron	Intel XE 19.04[1]	Intel MPI 2019 Update 4[2]

[1] For user subroutines, Intel Fortran XE 19.04 and Microsoft Visual Studio 2017 are required.

[2] Microsoft MPI program version 9.0 is also supported.

Note:	Installation of a Fortran compiler is only required if you need to run Marc with user subroutines.

2 Microsoft Windows: Marc and Mentat Installation

Installation Procedure

The installation of Marc (Marc and Mentat) can be done in four steps as follows:

1. Install MSC licensing Helium (security)
2. Install Marc and Mentat
3. Install Marc and Mentat documentation
4. Set environment variables and verify installation

Note:	For old installation procedure, refer to Appendix. A: Microsoft Windows: Marc Subdirectories and Installation **Ignore step 1 if** the license manager is already installed.

Install MSC Licensing Helium

1. Download msc_licensing_helium_windows64.exe from MSC download center.

- Default installation directory is `C:Program Files\MSC.Software\MSC Licensing\Helium`
- Obtain a license file from MSC Software. If a nodelocked license is to be used, then obtain FLEXlm hostid with
  - `C:Program Files\MSC.Software\MSC Licensing\Helium\lmtools`
  - Select the **System Settings** tab and click on **Save HOSTID Info to a File**.
  - Copy the license file to the proper location. The standard location is:

    `C:Program Files\MSC.Software\MSC Licensing\Helium\license.dat`

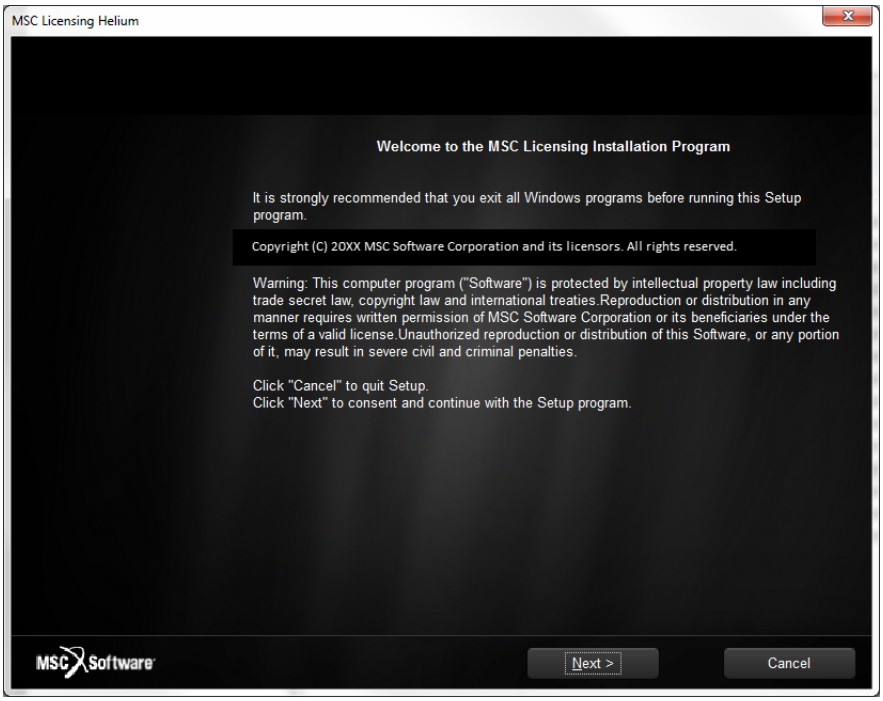
2. Run the executable file.

- If your system already has the older version of the MSC Licensing Server then it will show the following message:

  ```
  11.13 License Server process currently running. Do you want to un-install 11.13
  license server?
  ```

- Click **Yes** to continue.

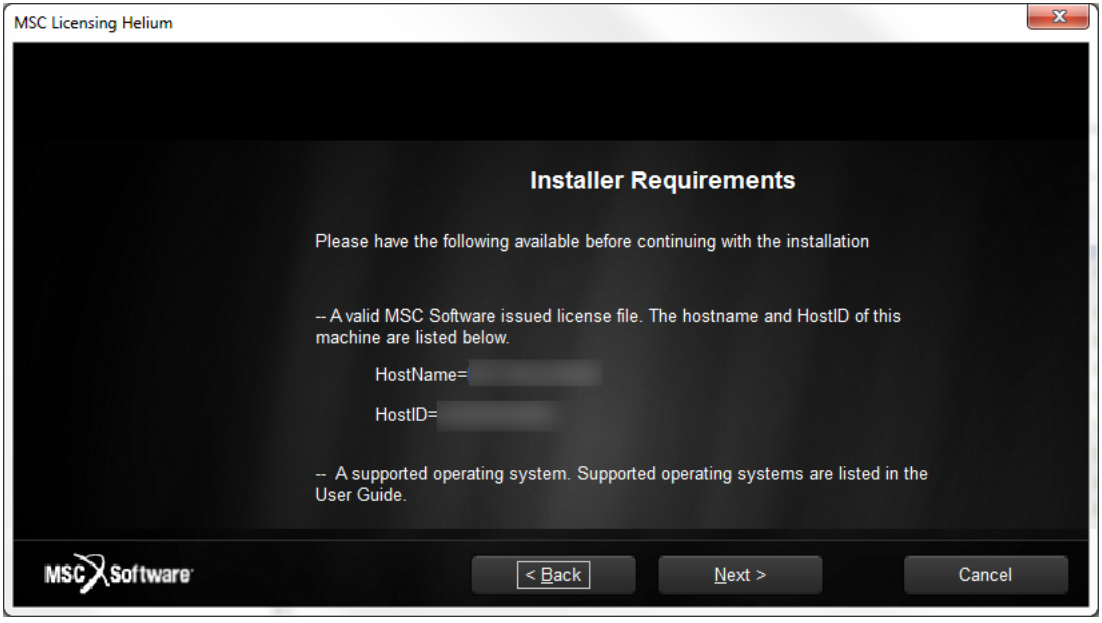
  The **MSC Licensing Helium** window appears.



Note:	■ The *MSC Licensing Helium User's Guide* opens up automatically the moment you run `msc_licensing_helium_windows64.exe`. For detailed information, refer to the *MSC Licensing Helium User's Guide*. ■ For detailed and elaborated information specific to Marc and Mentat installation refer to Appendix A: Microsoft Windows: Marc Subdirectories and Installation

3. Click **Next**.

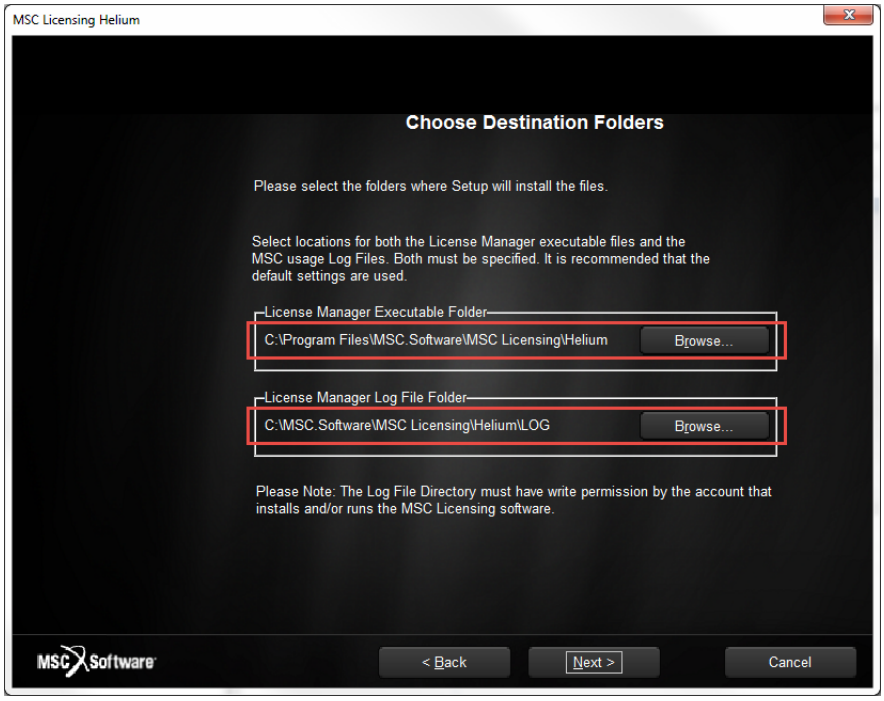
The **Installer Requirements** window appears.



You need a valid MSC Software license, HostName and HostID to fulfil the requirements.

4. Click **Next**.

The **Choose Destination Folders** window appears. Specify paths for **License Manager Executable Folder** and **License Manager Log File Folder**.

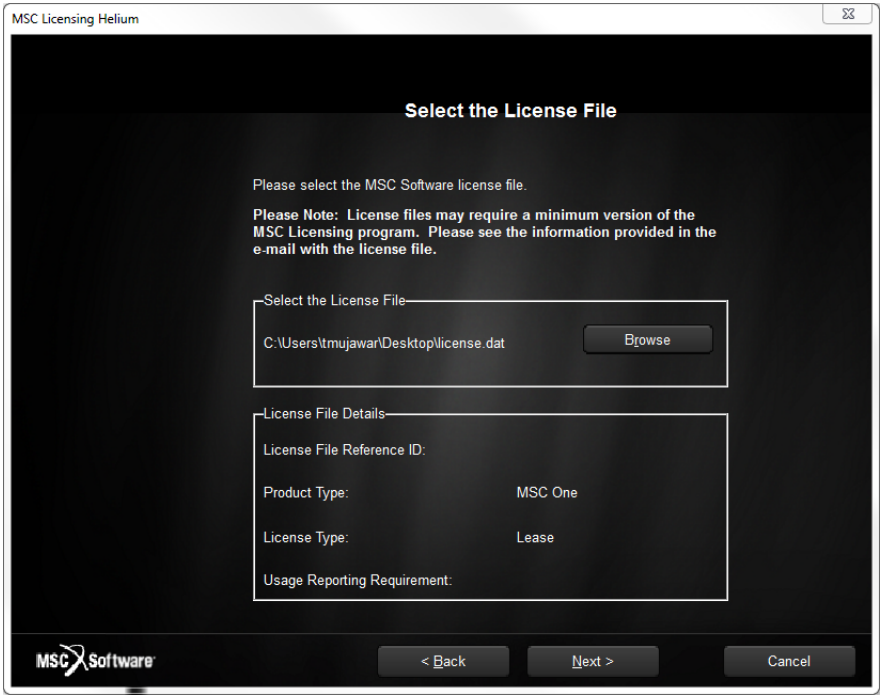


5. Click **Browse** and specify the path for the folders as follows:
   - **License Manager Executable Folder**

     `C:\Program Files\MSC.Software\MSC Licensing\Helium`
   - **License Manager Log File Folder**

     `C:\MSC.Software\MSC Licensing\Helium\LOG`
6. Click **Next**.

The **Select the License File** window appears.



7. Click **Browse** and select your `license.dat` file.

   Review the license file details.

   - If one or more feature entries in the license file has expired, the following error message appears:

     ```
     One or more FEATURE entries have maintenance end dates that have expired. These licenses can be used only by MSC products that were released prior to the maintenance expiration dare on the FEATURE entry.
     ```

     Click **OK** to continue

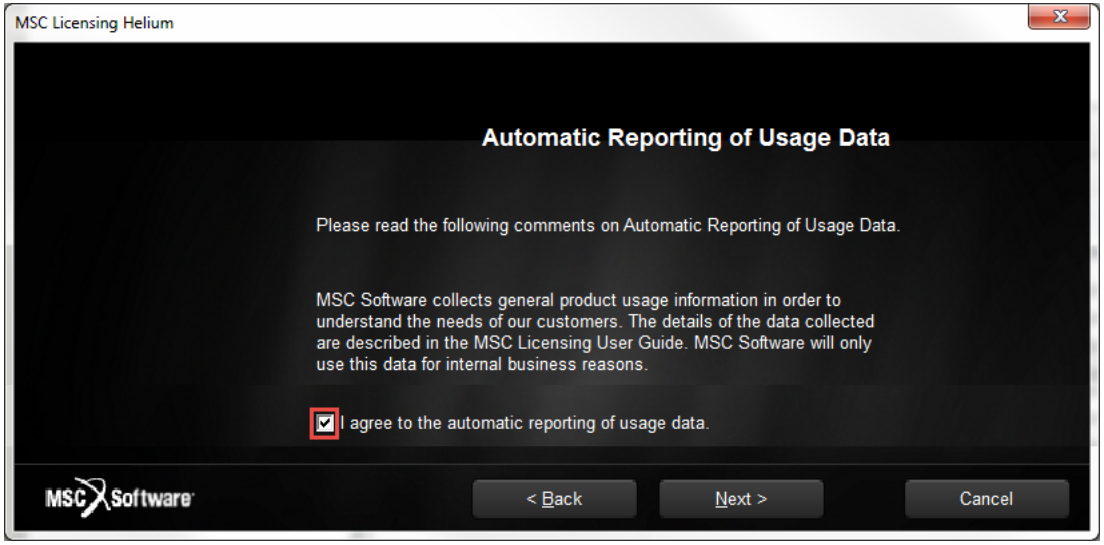
   - In case you face any problems related to the licensing, contact:

     msc_lic.support@mscsoftware.com or,

     MSC Software support center

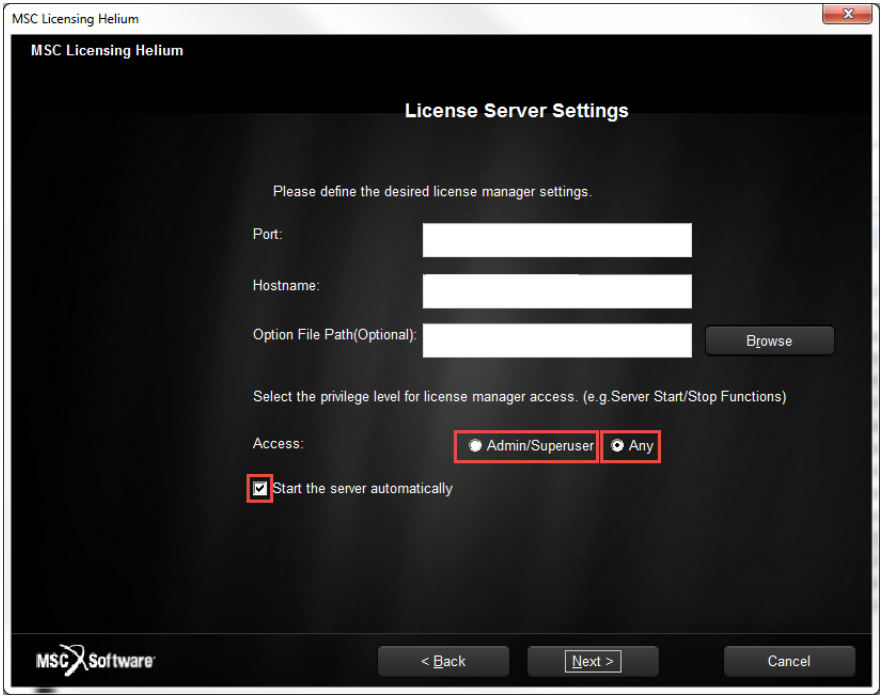
8. Click **Next**.

The **Automatic Reporting of Usage Data** window usage appears.



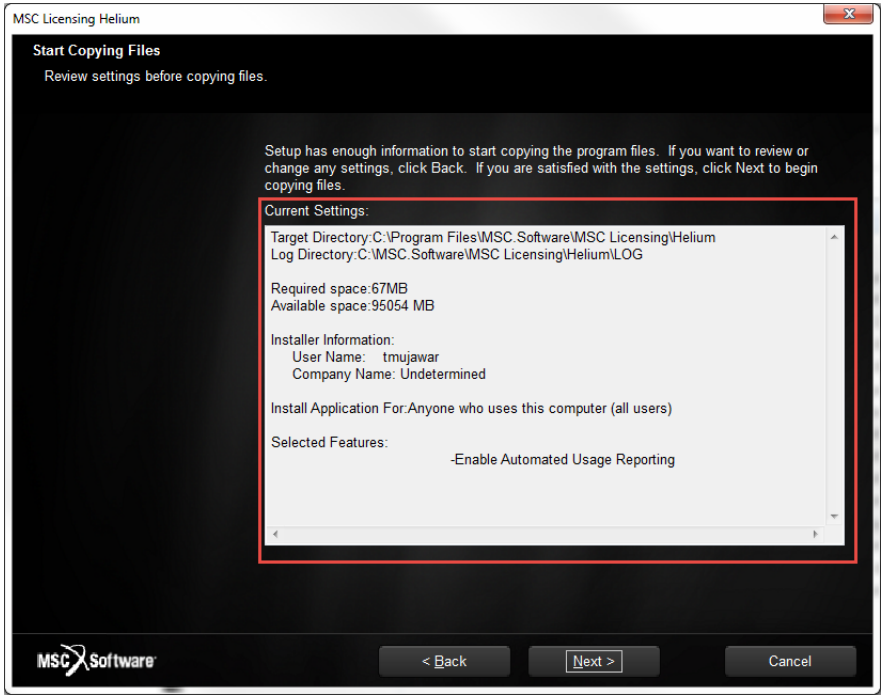
9. Click **Next**.

The **License Server Settings** window appears.



10. Enter port number and hostname.
11. Click **Next**.

A window about the files are being copied will appear.

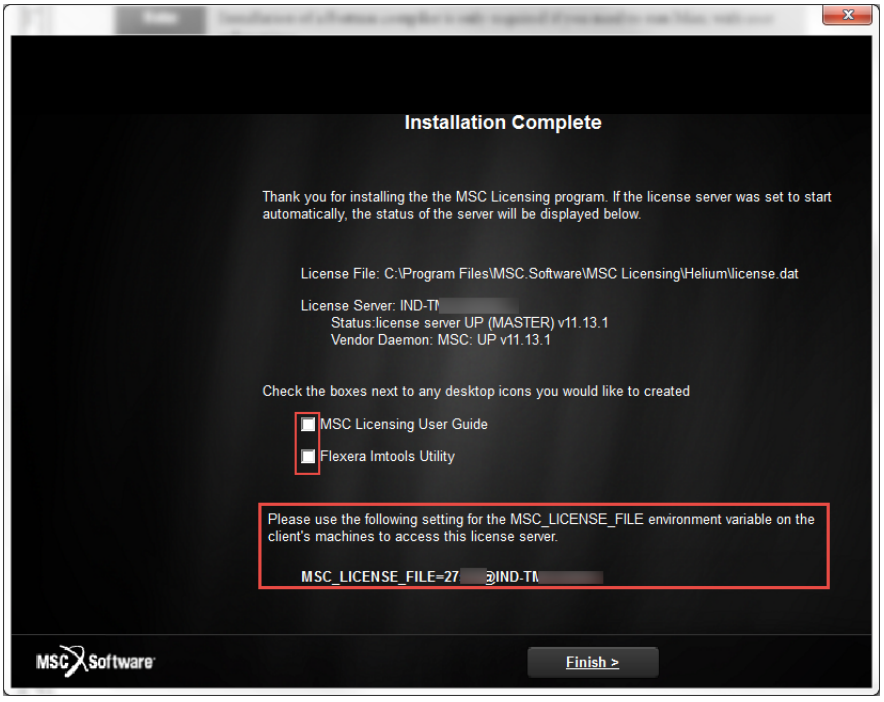


12. Click **Next**.The files are copied.

    The **MSC Licensing - InstallShield** Wizard appears with a prompt as follows:

    `MSC Licensing Helium is installed succesfully.`

A window, **Installation Complete** appears.



a. Check the boxes next to any of the options, if you want desktop icons.

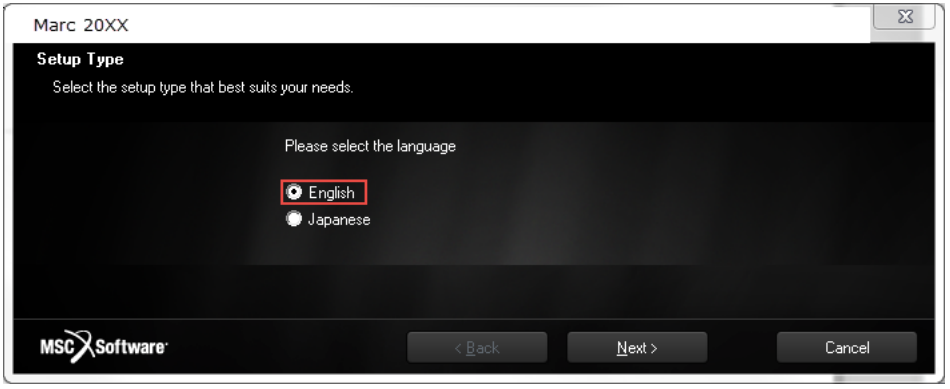
Note:	Important information related to **MSC_LICENSE_FILE** environment variable setting is displayed at the bottom of the window. Use this setting to create the environment variable.

13. Click **Finish**.

    The Licensing process is completed.

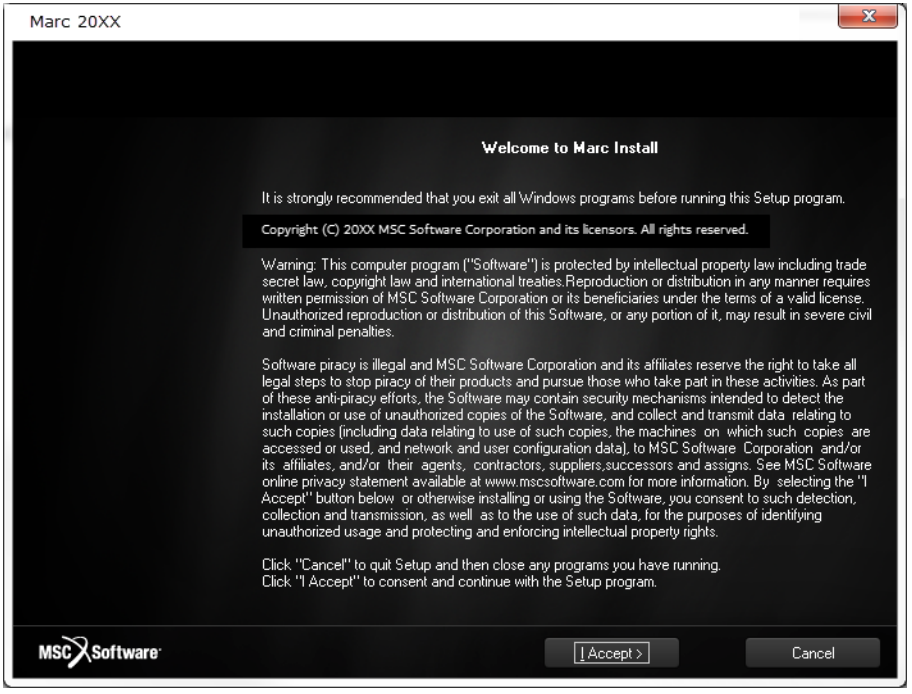
Install Marc and Mentat

1. Download the latest version from the MSC Software Download Centre.
2. Run the executable file.
3. The **Setup Type** window with language options appears.



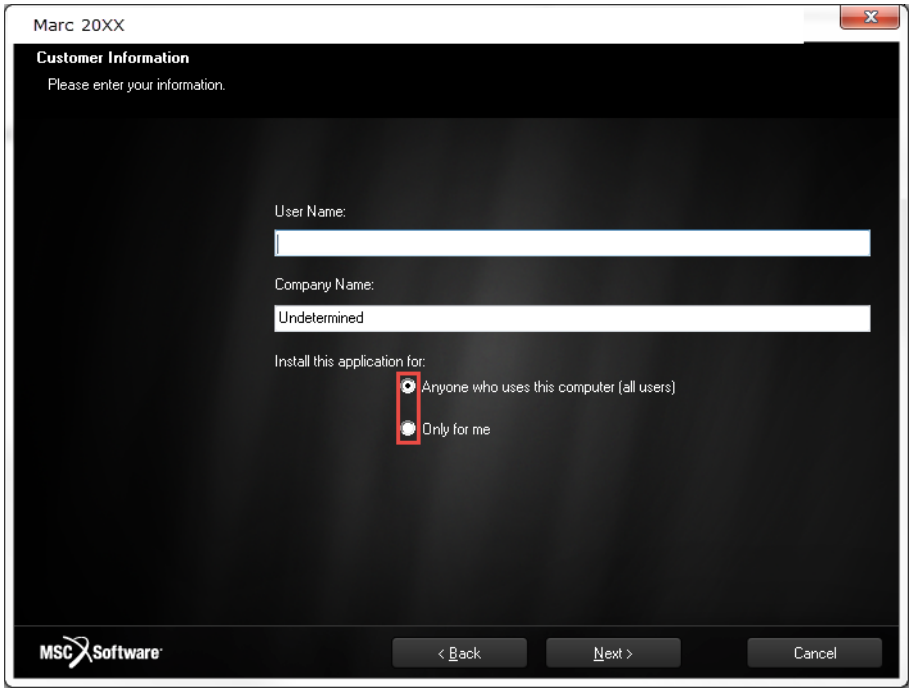
   a. Select **English** as your language.
4. Click **Next**.

   The **Welcome to Marc Install** window appears.



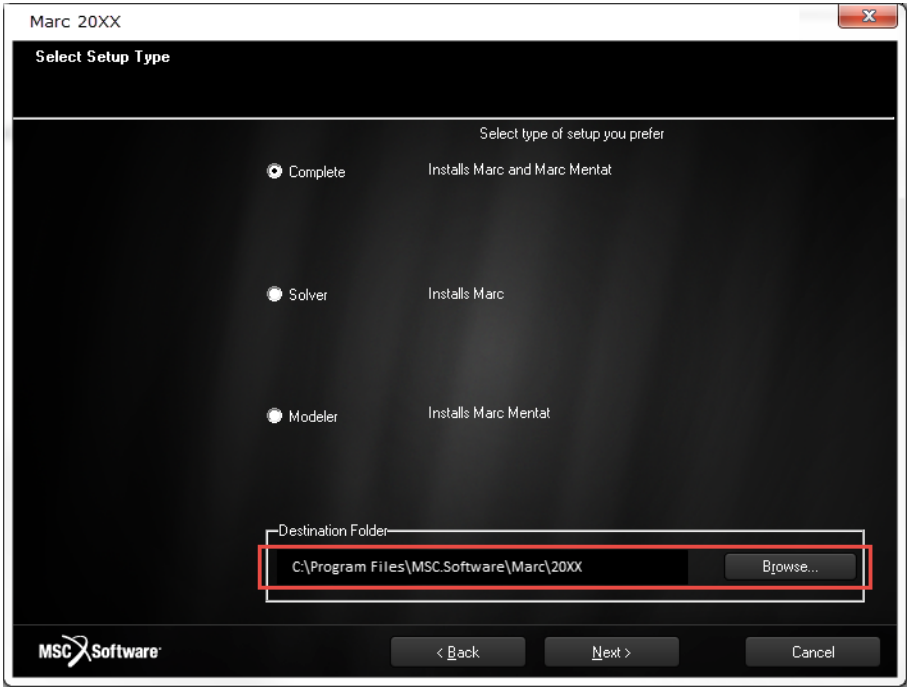
5. Click **I Accept** to consent.

   A window for customer information appears.



6. Enter **User Name** and **Company Name**.
   a. Select **Anyone who uses this computer (all users)** if you want to make it available for all users.
   b. Select **Only for me** if you want to make it available only for yourself.

The **Select Setup Type** window appears.

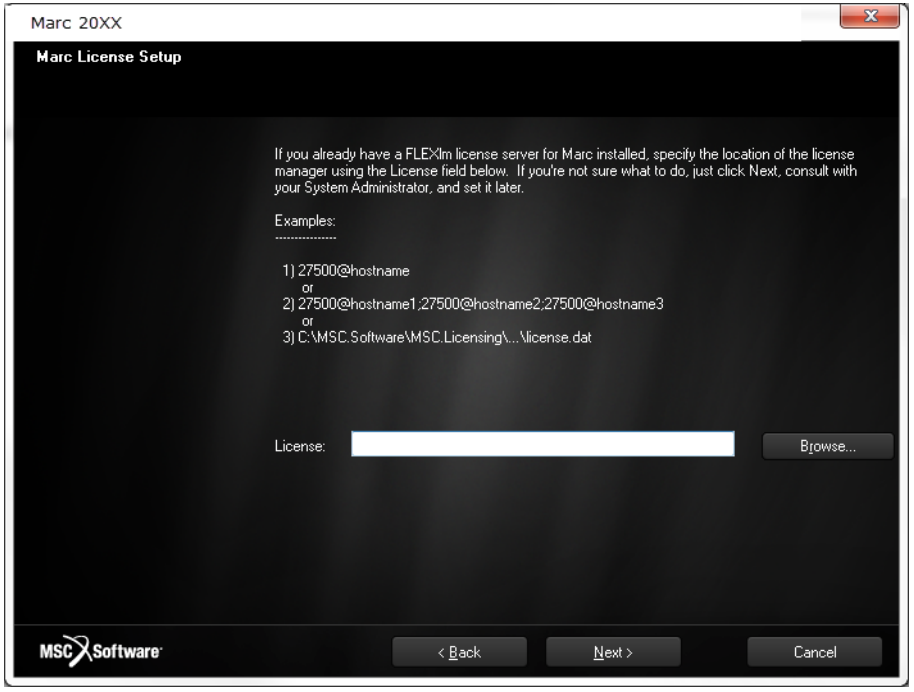


7. Select **Complete** to Install Marc and Mentat.
   a. Select Solver if you want to install only Marc Solver.
   b. Select Modeler if you want to install only Mentat.
   c. Notice the default installation path.
   d. Click **Browse** to change the Destination Folder.

      `C:\Program Files\MSC.Software\Marc\20XX.0.0`

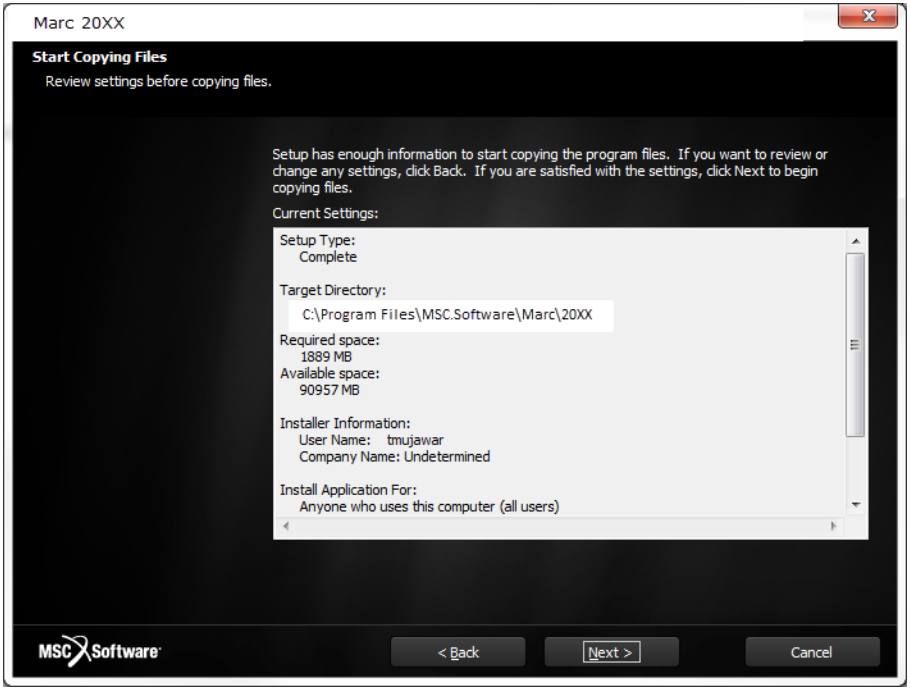
      It is recommended that you leave the **Destination Folder** path as is.
8. Click **Next**.

The **Marc License Setup** window appears.



9. Click **Browse** and specify the location of the license manager, if you already a FLEXlm license server installed. For e.g. **27500@hostname**.
10. Click **Next**.

A window informing that the files are being copied appears.



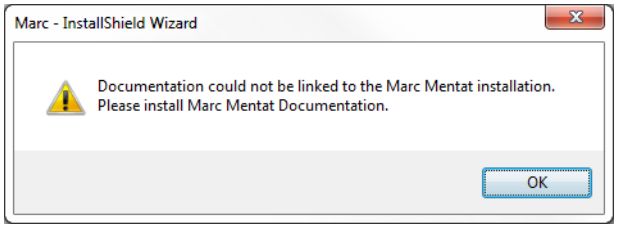
11. Review the current settings and click **Next**.

The Marc installation starts and the **Marc and Mentat 20XX** window appears.

The last window is **Setup Complete** window. It will present you with an option to view the `readme.txt` file.

12. Click the **Finish** button to complete setup.

The installation is complete with a prompt as follows:



Notes:

- Ensure that the environment variable **MSC_LICENSE_FILE** is set properly to the full pathname of your valid license file. If it is not, the product will fail due to licensing.
- To run from the command prompt, `cd` to the `<parent>\mentat20XX` directory, and enter the command `bin\mentat` to start Mentat.
- Check the Mentat program by running one of the standard Marc demonstration examples as proof of a successful installation.
- Check if the user subroutines are working by running one of the standard user subroutine demo problems:

```
cd C:\Program Files\MSC.Software\marc
cd 20XX.0.0\marc20XX\demo\tools\
run_marc -j e2x4 -u u2x4
```

- If you have a **Fortran compiler,** run a user subroutine example using:

```
run_marc -j e2x4 -user u2x4
```

  Marc should give a **Marc Exit number 3004.**

- If you plan on using the Marc Parallel Network feature, you need to setup MPI. This service is optionally installed on the system when you have installed Marc, however you need to install it on the remote machines. Refer Chapter 4: Microsoft Windows: Marc Parallel Network for important information on installing and running jobs with the network version.

Now, install the documentation. The following section demonstrates the procedure of installing the documentation.

Install Marc and Mentat Documentation

To make optimal use of the documentation, it should be installed on a computer where Marc and Mentat are available. This way, documents can be accessed interactively from within Mentat, and Marc example problems can be run.

You can install the documentation either before or after installing Marc and Mentat. The documentation installer will check if there is a corresponding Marc and Mentat version installed, where the Marc and Mentat installer will check if there is a corresponding documentation version installed, so that they can be linked together.

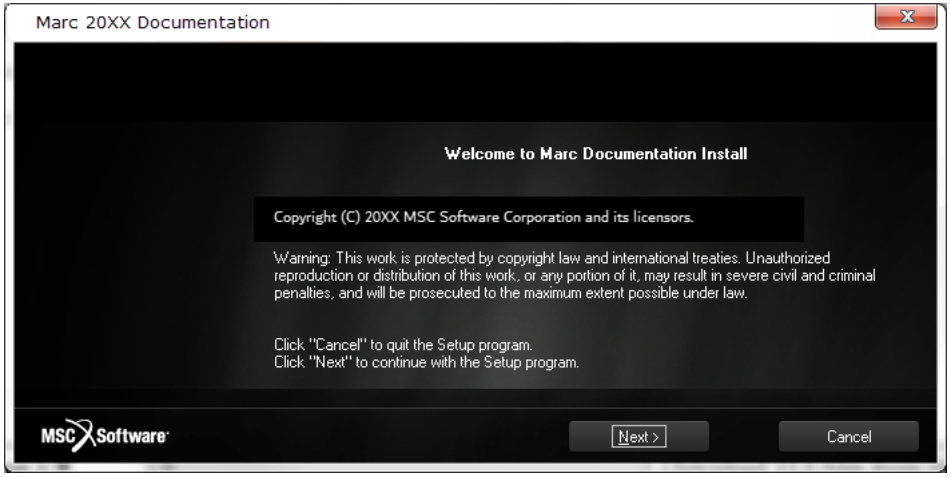
The installer will prompt you to set the location where the documentation needs to be installed. This default location is:

`C:\Program Files\MSC.Software\Marc_Documentation\20XX.0.0`

You can now install the documentation to any other location. The `doc` and `examples` directories will be created in the directory specified by you. Follow the steps to install documentation:

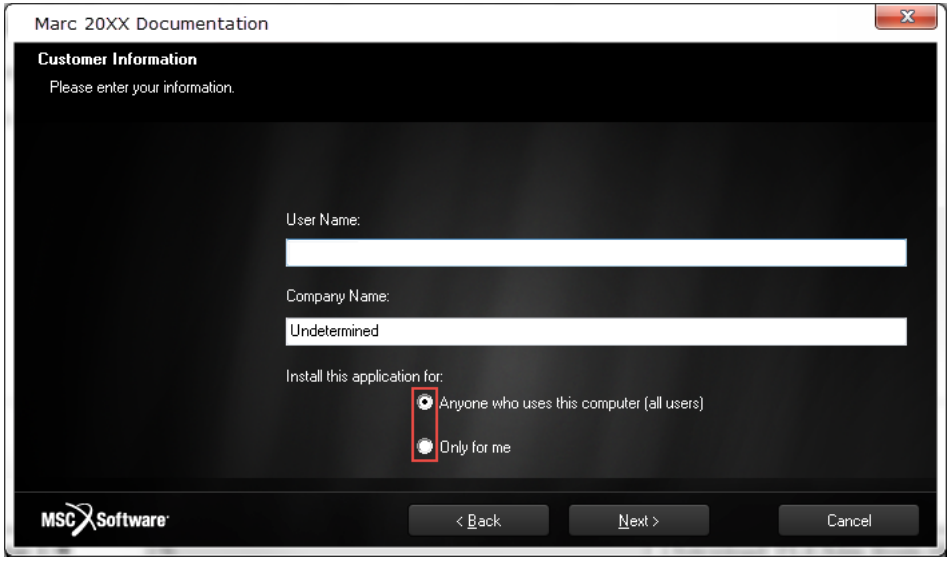
1. Download the latest executable file from the download center.
2. Run the executable file.

The **Marc 20XX Documentation** window appears.



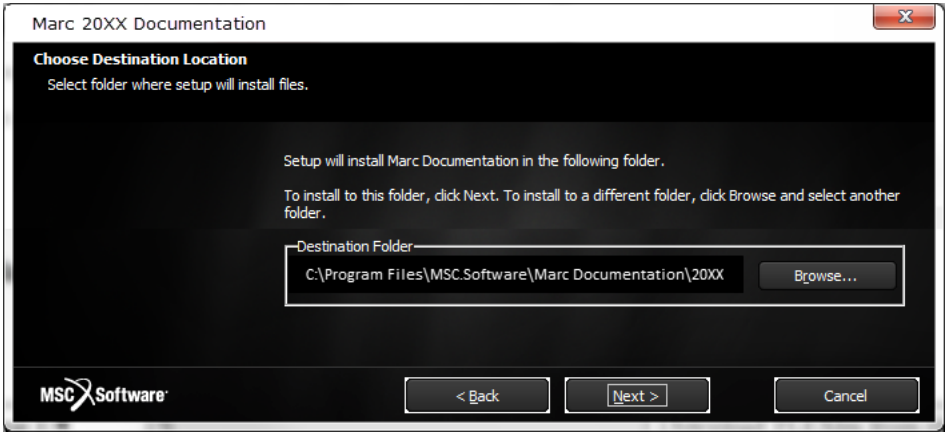
3. Click **Next**.

   The **Customer Information** window appears.



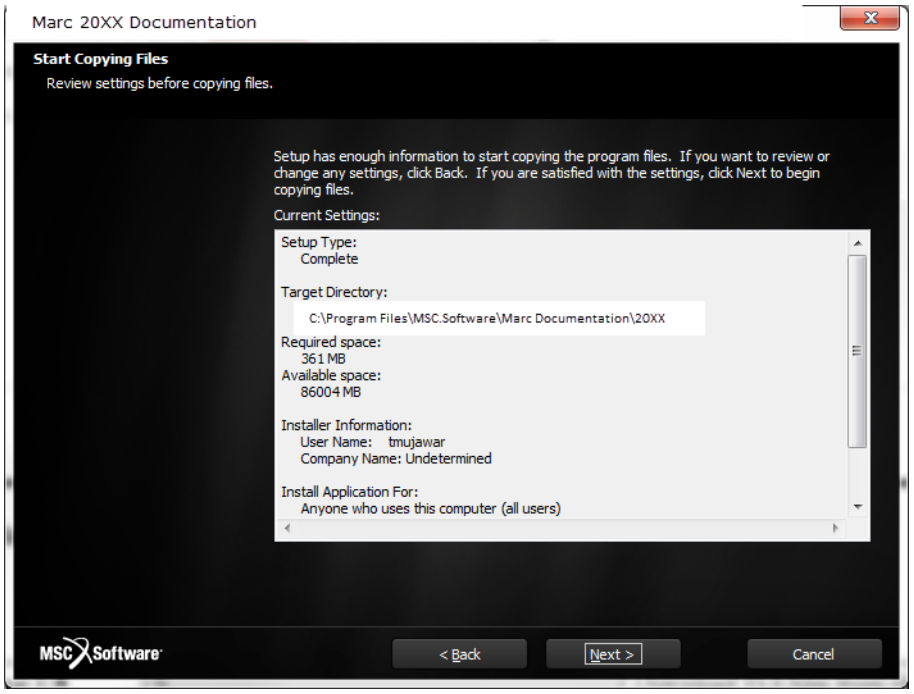
4. Enter **User Name** and **Company Name**.
   a. Select **Anyone who uses this computer** (**all users**) if you want to make it available for all users.
   b. Select **Only for me** if you want to make it available only for yourself.

The **Choose Destination Location** window appears.



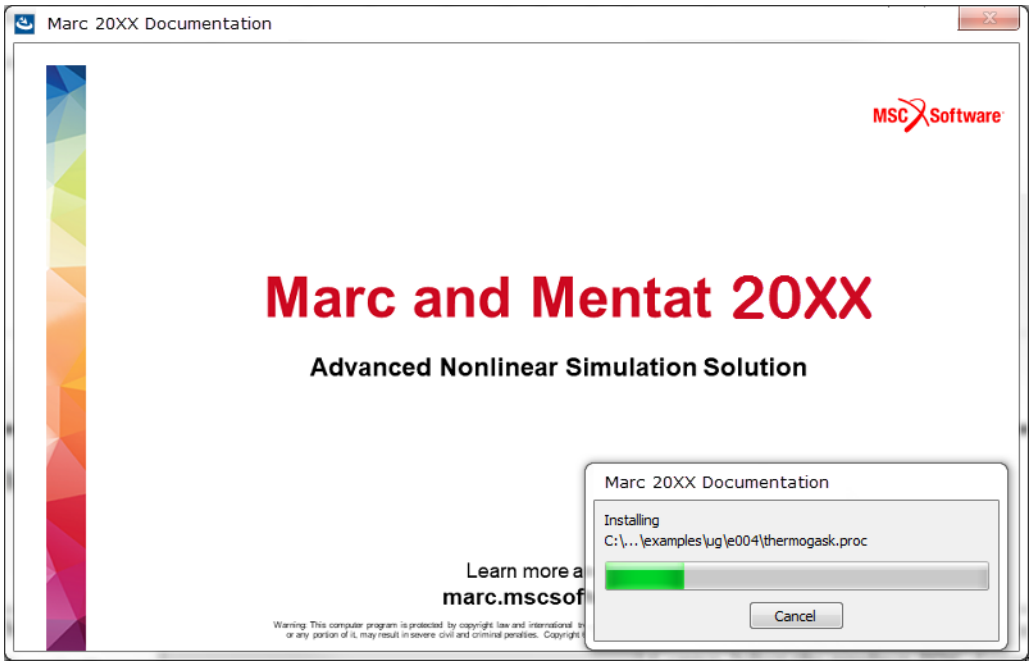
5. Retain the default path to `C:\Program Files\MSC.Software\Marc Documentation\20XX.0.0`
6. Click **Next**.

A window informing that the files are being copied appears.

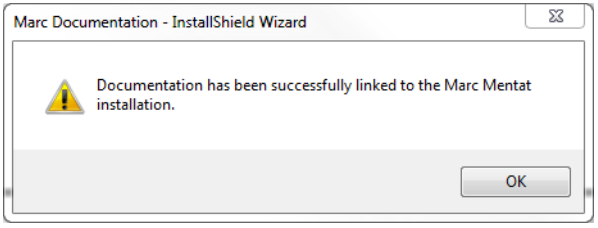


7. Review the current settings and click **Next**.

The installation process starts and can be seen in the window as follows:



An information message box is displayed:



8. Click **OK**.
9. Click **Finish**.

   The documentation is now installed and linked to Marc and Mentat.

Set Environment Variables and Verify Installation

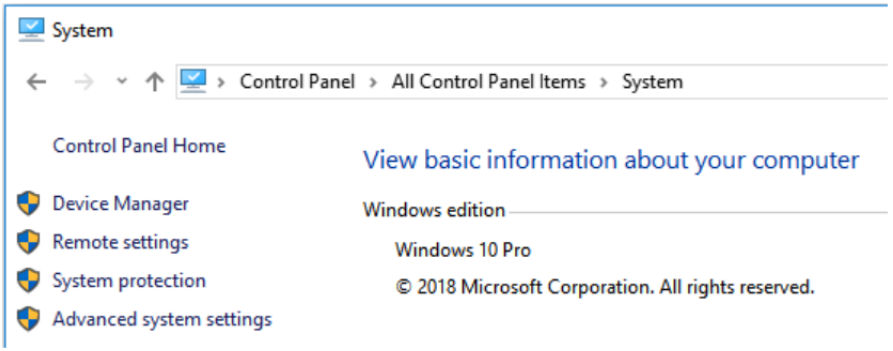
In this step, you will set the necessary environment variables and verify whether the installation is successful or not.

You will have to add the **MSC_LICENSE_FILE** environment variable on the license server and the client machines (if you have MSC Software installed on the client).

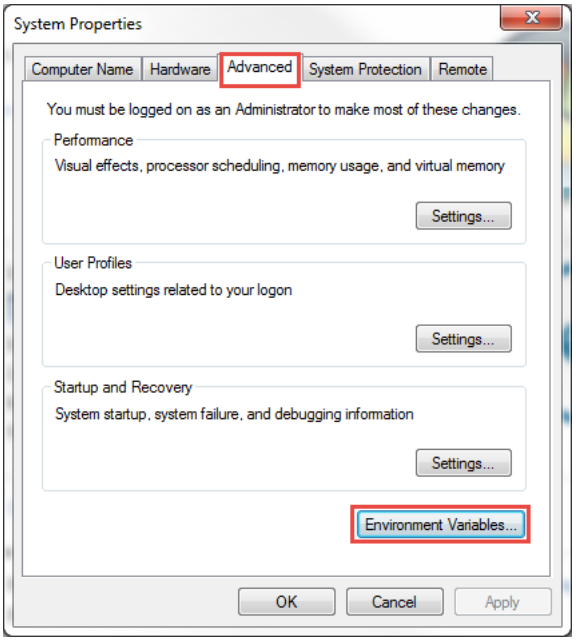
Follow the procedure to set the variable:

1. Right click on **My Computer >Properties**.

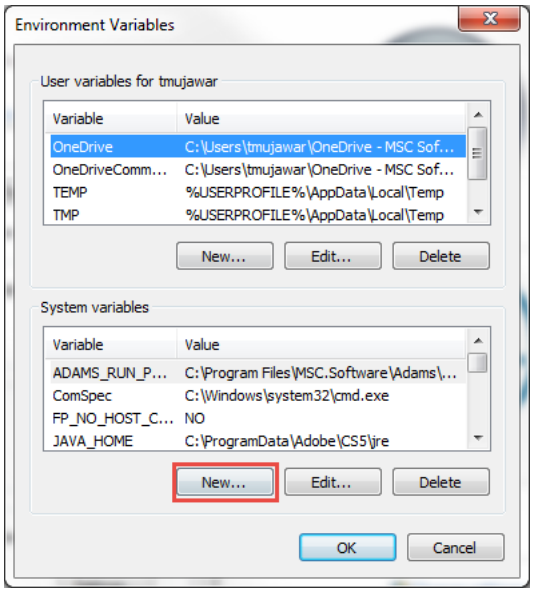
Control Panel Home window appears.



2. Choose the **Advanced system settings** option.
   The **System Properties** dialog appears.

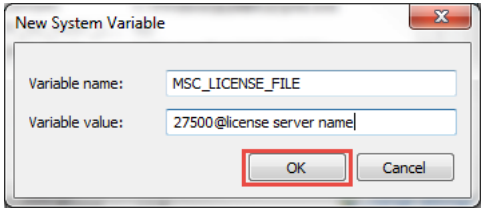


3. Select the **Advanced** tab.
   a. Click **Environment Variables** button at the bottom.
      The **Environment Variables** dialog appears.



b. Click **New** in the **System Variables** section.

The **New System Variable** dialog appears.



c. Enter **Variable Name** as `MSC_LICENSE_FILE`.

d. Enter **Variable Value** as `27500@license server name`.

4. Click **OK** thrice to close all the dialog boxes and exit control panel.

The required environment variables are set. Try launching **Marc Mentat**.

Notes:	■ If you face any problem in launching the application or any problems related to the licensing, contact MSC Software support center. ■ The license server must be running in order to run Marc and Mentat. ■ Previous versions of Marc and Mentat will run with the latest FLEXlm version, so it is safe to upgrade previous installations of FLEXlm.

FLEXlm 11.13 VS Helium(11.16.3.0) License Manager

This section contains information and directory paths according to Helium license manager. The changes in the default paths in comparison with FLEXlm 11.13 are as in the following table:

11.13	Helium (11.16.3.0)
C:\MSC.Software\MSC.Licensing\11.13\msc	C:\Program Files\MSC.Software\MSC Licensing\Helium\msc
C:\MSC.Software\MSC.Licensing\11.13\lmtools	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmtools
C:\MSC.Software\MSC.Licensing\11.13\installs	C:\Program Files\MSC.Software\MSC Licensing\Helium\installs
C:\MSC.Software\MSC.Licensing\11.13\lmgrd	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmgrd
C:\MSC.Software\MSC.Licensing\11.13\LOG	C:\MSC.Software\MSC Licensing\Helium\LOG
C:\MSC.Software\MSC.Licensing\11.13\lmutil	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmutil
C:\MSC.Software\MSC.Licensing\11.13\msclic.ini	C:\Program Files\MSC.Software\MSC Licensing\Helium\msclic.ini

Installation Information

License file	1. Two lines of the file `license.dat` need to be modified
	• **SERVER** line which specifies the system hostname
	• **DAEMON** line which specifies the daemon name and the path to the daemon program
	The content is typically something like:
	`SERVER this_host 0022192361f 1700`
	`DAEMON MSC /your_path/msc`
	2. Replace the string **this_host** with the hostname of the machine where the license server is running.
	3. Replace the string **your_path** by the full path to the program **MSC**. If the default location is used for the security installation it should be
	DAEMON MSC "`C:\Program Files\MSC.Software\MSC Licensing\Helium\MSC.exe`"
	4. Start the program
	`C:\Program Files\MSC.Software\MSC Licensing\Helium\lmtools`
	Select the **System Settings** tab and click on **Save HOSTID Info to a File.**
	The system identifier may also be found by running:
	`lmutil lmhostid`
	in the directory `C:\Program Files\MSC.Software\MSC Licensing\Helium`

hostid	■ For a nodelocked license it is necessary to obtain a FLEXlm hostid for the machine running Marc and Mentat. This is done as follows:
	• Start the program
	`C:\Program Files\MSC.Software\MSC Licensing\Helium\lmtools`
	• Select the **System Settings** tab and click on *S*ave **HOSTID Info to a File**.
	■ The system identifier may also be found by running:
	`lmutil lmhostid`
	in the directory `C:\Program Files\MSC.Software\MSC Licensing\Helium`.

Specify license file	You will be prompted to specify the path to your license file. It will default to the current setting of MSC_LICENSE_FILE. If it is not set, then it will attempt to use the license file specified for FLEXlm License Manager version Helium(11.16.3). If you do not already have your license file, you may leave it blank. However you must set the MSC_LICENSE_FILE variable before attempting to run any of the *Marc* products. **Note:** This is an important step. The installation will set the variable MSC_LICENSE_FILE to the setting that you enter. Failing to set it to a valid license file will result in a licensing failure and you will have to edit the environment variable setting by using the System applet in the Control Panel.

File types	The installation will associate the file types `.mfd`, `.mud`, `t16`, `.t19` and `.proc` to Mentat if they are not already associated. If they are, as it would be in the case that you have a previous Mentat installation, you will be asked if you wish to overwrite them. ■ If you answer **YES**, then the files are set to the current version and the PATH environment variable is updated with the current version specified first. ■ If you answer **NO**, then the files are not modified and the PATH environment variable is updated with the current version specified last.
Complete the installation	■ Select the **Finish** button in Setup. ■ You may want to check the contents against the list supplied in Appendix A of this document. Should any subdirectory be missing, contact MSC Software customer support for further details.
Starting the License Manager	You must start the **FLEXlm License Manager** before attempting to run Marc or Mentat on the system that is assigned the license. 1. To start the license manager, use the **Start** menu and select: **Programs → MSC Software → MSC.Licensing 11.16 → lmtools** 2. Select the **Config Services** tab and verify that the settings are correct; i.e. the **License File** is set to the proper license file. 3. Start the license manager from the **Start/Stop/Reread** tab by pressing the **Start Server** button. **Note:** If you already had the MSC installed, you should not need to perform these steps. You should specify the options Use Services and Start Server at Power Up under the Config Services tab. This will enable the license manager to start automatically at boot time.

Note:	Troubleshooting related hints are provided in Chapter 5: Windows Troubleshooting.

Managing FLEXlm

This section contains information about managing the license file and related tasks.

FLEXlm License File

When you receive your licenses from MSC Software, the license file, `license.dat`, should normally be placed in the `C:\Program Files\MSC.Software\MSC Licensing\Helium` directory. The file can be located elsewhere as long as the environment variable MSC_LICENSE_FILE points to it.

Two lines of the license file contains installation specific information:

- `SERVER this_host 0022192361f 27500`
- `DAEMON MSC /your_path/msc`

Here **this_host** should be set to the hostname of the machine where the license server is running and **your_path** should be set to the full path to the msc program, by default located in `C:\Program Files\MSC.Software\MSC Licensing\Helium`

FLEXlm License Manager

When you install Marc or Mentat, the FLEXlm License Manager is installed in the `MSC Licensing\Helium` directory. Once `lmgrd.exe` is running, it will read the license file `license.dat` which is located in the `MSC Licensing\Helium`. The license file contains the Marc and Mentat license (and other MSC Software product licenses, if necessary). In addition, `lmgrd.exe` will also start the MSC Software vendor daemon `MSC.exe`. The path to `MSC.exe` is specified in the license file on the DAEMON line. These processes must be running on the license server for the MSC Software security system to obtain a license. The only exception to this is for a **zero count** license. If the number of licenses for a feature (the number following the expiration date) is 0, then neither `lmgrd.exe` or `MSC.exe` are used. The license manager is only used to keep track of licenses that are checked in/out.

Marc contacts these daemons at regular intervals. If no contact is made after a specified time period, Marc terminates execution.

Note:	For the 20XX version, the FLEXlm License Manager must be at version 11.13 or higher.

You must start the FLEXlm License Manager before attempting to run Marc or Mentat on the system that is assigned the license. To start the license manager do the following:

1. Go to the **Start** menu and select **Programs → MSC Software → MSC.Licensing 11.16 → lmtools**
2. Select the **Config Services** tab and verify that the settings are correct; i.e. the **License File** is set to the proper license file.
3. Then start the license manager from the **Start/Stop/Reread** tab by pressing the **Start Server** button.

Environment Variables

The environment variable **MSC_LICENSE_FILE** is used to specify the `license.dat` file. This environment variable can be set using the System applet in the Control Panel, and is a semicolon separated list of file pathnames or hosts. The default setting will be `C:\Program Files\MSC.Software\MSC Licensing\Helium\license.dat`. When Marc executes, it checks the list of license files specified by this environment variable. You may have it point to a license server using the syntax port@host, as follows:

- Variable: **MSC_LICENSE_FILE**
- Value: **27500@myserver**

If you have other products that use FLEXlm and they are required to be available when Marc is running (such as a Fortran compiler license), then you should modify the `MSC_LICENSE_FILE` setting to point to the proper license file for that product.

You may instead want to combine the licenses into one file.

The environment variable **MSC_AUTHQUE** is used to specify the number of minutes a program will wait for a license after it has been requested. If the license request cannot immediately be honored, the request will be queued and the variable defines how long it should wait in this queue. The variable must be set in your environment before Marc and Mentat are started. The default value is 5 minutes. It may be set as follows:

```
set MSC_AUTHQUE=20
```

Client/Server Licensing

The default installation assumes that the system in which Marc is installed will function as the **license server**. The term **license server** only refers to the fact that **lmgrd** and **MSC** will be running on that system, and will maintain the state of available licenses. Even if you have purchased a nodelocked license, the nodelocked system will function as the license server for that license. A nodelocked license can be distinguished from a floating license by the string **HOSTID=xxx** specified in the feature line.

Configuring the Marc DCOM Server

This section contains a procedure about configuring a DCOM Server.

Specifying the Logon User

The Marc DCOM Server allows you to run jobs on a remote Microsoft Windows machine without actually being logged into it. Unlike Marc Parallel, it will only run a single CPU job.

When you install Marc, the installation will initially setup the Marc DCOM Server. The server must be setup on both the client and the server machine. You must specify a login user for the Marc DCOM Server using `dcomcnfg` before attempting to run a job remotely.

To run `dcomcnfg` do the following:

1. Open up a console window and type `dcomcnfg` at the prompt.
2. Perform one of the following

- Select **Start** menu and enter `Run` in the **Search Window,**
- Select **Windows logo key + r**

3. Enter `dcomcnfg` and press **OK. Component Services** window (shown in Figure 2-1)appears.

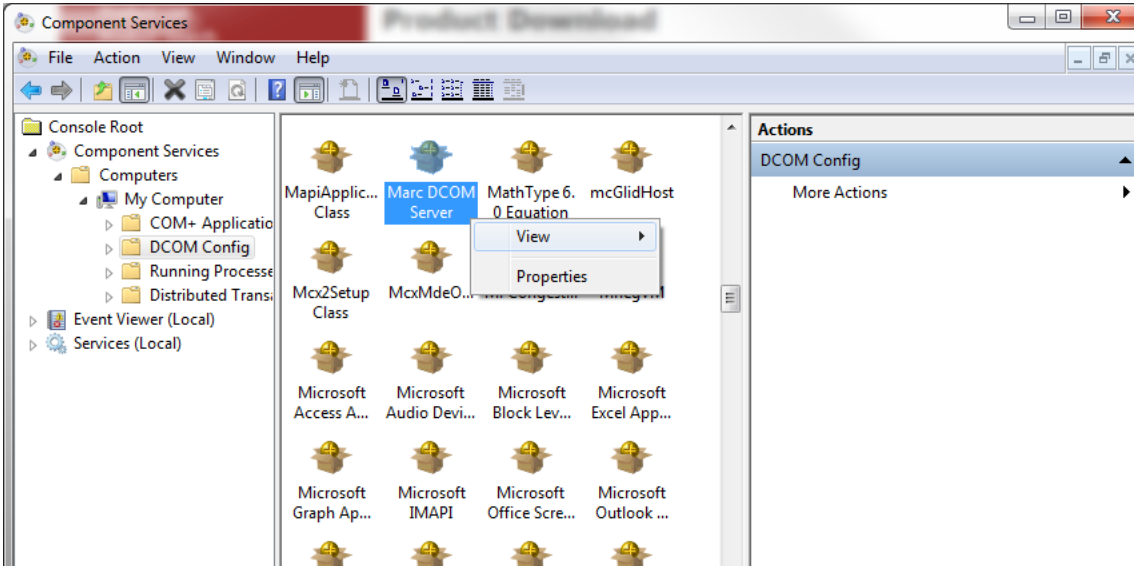
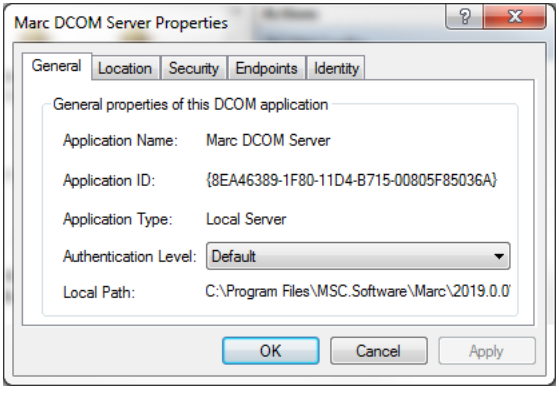


Figure 2-1 Component Services window

4. Select **Component Services** in the **Console Root** panel.
5. In column right to the **Console Root** panel, select **Computers → My Computer → DCOM Config**
6. Scroll down to the **Marc DCOM Server** item.
7. Right click **Marc DCOM Server** item and select the **Properties**.

   The **Marc DCOM Server Properties** dialog will appear.



The **Local Path** depends on the destination folder chosen during the installation procedure.

The only property that needs adjusting is the **Identity** property.

8. Select the **Identity** tab. `Dcomcnfg` will then display the identity view as shown in Figure 2-2.

Figure 2-2 Dcomcnfg Displaying the Identity Properties of the Marc DCOM Server

9. Select **This user** and specify a user that has access rights to this machine.
10. Select **Apply** and then **OK** to close the window.
11. Select **OK** in the main view.

The data files must be located in a shared directory. This is required so that the two machines can access the files. If the file is not located in a shared directory, Marc will issue an error message specifying that the file is not located in a shared directory.

To share a disk or a directory:

1. Select **My Computer** from the desktop and select the disk or browse to the directory.
2. Right click, and select **Sharing** to select the disk or the directory.
3. In the Sharing view, select the **Share this folder** button and enter a name for the share.

Testing the installation

You should first test the Marc DCOM Server on the server machine, and then test it on a client machine. First, copy a Marc data file to a shared directory. The file must be in a shared directory even if the job is to be run locally using the Marc DCOM Server. The data file `<parent>\marc20XX\demo\e2x1.dat` will suffice. Run the job from any command prompt window using the *-pc* option:

```
<parent>\marc20XX\tools\run_marc -pc <servername> -j e2x1
```

If it succeeds, then perform the same test on a client machine. If a failure occurs, refer to section Troubleshooting 56. Make sure the data file resides in a shared directory. When running the job, you may also use the UNC name in the path. If it is not specified, Marc will determine the UNC path and send it to the server.

Mentat Support

Running the job remotely may also be done within Mentat, however the menu is not displayed by default. You must first rebuild the menu file using:

```
cd <parent>\mentat20XX

bin\mentat -compile menus\win64\main.msb -df DCOM
```

In the **Jobs → Run → Advanced Job Submission** menu, there now is a **DCOM** option (as shown in Figure 2-3). Select the **DCOM** option and in the adjacent text area specify the name of the remote machine.

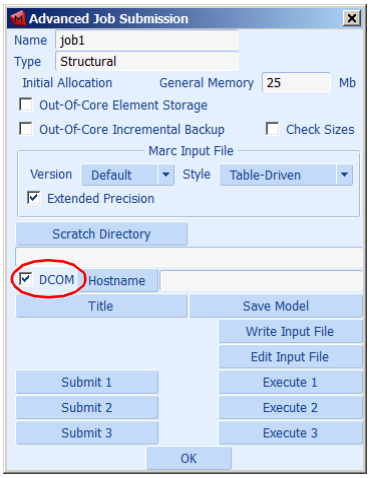


Figure 2-3 Advanced Job Submission Menu Displaying the DCOM Option

Troubleshooting

In the case of an error, try some of the following suggestions:

- Check your user ID is valid on both the server and the client machines.
- Check the user ID has read and write access to the directory where the job is to be run.
- Check the directory that the data file resides in is a shared directory. For the Windows Server, you must specify the user permissions using one of the following methods:
  - Using the /GRANT option for the net share command
  - Clicking the **Permissions** button under the **Sharing** tab in the Explorer
- Run the `run_marc` script using the *-it* option to print out debugging information. This will print out exactly what is being sent to the server:

```
<parent>\marc20XX\tools\run_marc -pc <servername> -j e2x1 -it 0
```

- The path name displayed in the debug output should be displayed as a proper UNC name. If not, verify that the directory is shared.

GPU Support on Windows

Hardware and Software Requirements

The list of supported solver types with the GPU option can be found in *Marc Volume A: Theory and User Information* Chapter 11: Solution Procedures for Nonlinear Systems in Table 11-3 and Table 11-4.

An NVIDIA driver is necessary if the GPGPU capability is to be used. The driver needs to be compatible with the one used in the Marc build; see Marc Volume A: Theory and User Information, Chapter 12: GPU Support in for the minimum driver version supported.

The GPU capability in Marc is developed using CUDA toolkit 11.0 Update 1 with supported compute capability ranging from 3.5 to 8.0. NVIDIA Tesla cards with higher compute capability values are recommended. Note that NVIDIA Quadro cards can also be used.

Deviceinfo Utility

A stand-alone utility `deviceinfo.bat` is provided in the tools directory to check available GPU cards on the machine. This utility provides information about the GPU cards and the installed NVIDIA drivers toolkit version. Typical output from the deviceinfo utility is shown below:

```
[ ** Marc GPGPU device diagnosis utility ** ]

 Installed CUDA driver version is 11.0

 Total 1 devices are detected on the machine

   Device id 0 : "Quadro M2200"

               Compute capability                : 5.2
               Number of Multiprocessors         : 8
               Number of Cores per Multiprocessor : 128
               Total Number of Cores             : 1024
               Total device memory (in MB)       : 4096
               TCC driver mode enabled           : No


   The supported compute capability for Marc GPGPU feature is from 3.5 to 8.0

   Below is the list of supported devices

       Device id 0               : "Quadro M2200"
       TCC driver mode enabled   : No*

       *Note:
           TCC drive mode is not enabled on device id 0. Hence device
           will not be detected for multi-host parallel (DDM) jobs.

           To use GPGPU device for multi-host DDM run (with Multifrontal solver),
           enable TCC mode. For more information on TCC mode please refer to
           NVIDIA help page.
           https://docs.nvidia.com/nsight-visual-studio-edition/reference/index.html#tesla-compute-cluster
```

Figure 2-4 Deviceinfo.bat Output

Mentat Support

Running the job with GPU may also be done from Mentat. The GPU menu is only displayed for supported solver types Multifrontal Sparse and Iterative Sparse. Note that solver multi-threading (which refers to CPU threads) is not supported when the GPU option is selected.

The GPU option in Mentat is shown in Figure 2-5. When the GPU option is chosen, there are two options available for GPU card selection:

a. **Automatic**: At run-time, Marc determines the GPU cards that are available and assesses the optimal card to be used.

b. **User:** The user can specify the GPU cards that need to be used for the job. The available device IDs can be determined by running the deviceinfo.bat utility.

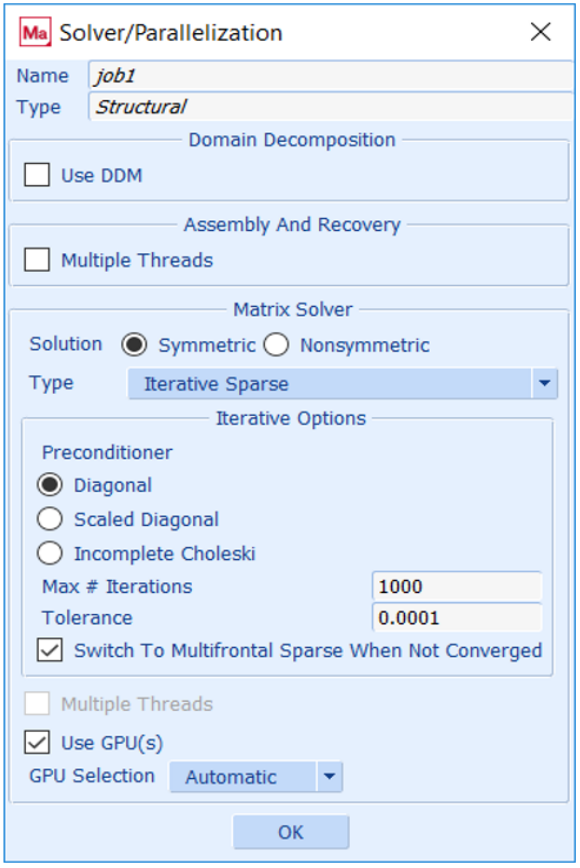


Figure 2-5 GPU Option in Mentat

Troubleshooting

Exit 63

Cause	The Marc version being used is not supported with the GPU version of the Multifrontal solver.
User Information	Non-GPU version of the Multi-frontal solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

Exit 64

Cause	Error occurs while attempting to use the GPU card specified by the user.
User Information	This error occurs while trying to use the GPU card specified by the user.
User Action	Use the deviceinfo utility to check for supported and available GPU cards on machine.

Exit 65

Cause	The Marc version being used is not supported with the GPU version of the Iterative solver.
User Information	Non-GPU version of the Iterative solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

missing library "nvcuda.dll"

Cause	This error occurs when required NVIDIA drivers are not available while running Marc with the GPU option.
User Information	To use GPU feature in Marc, necessary NVIDIA drivers need to be installed on machine.
User Action	Use deviceinfo utility to check if supported drivers are installed on machine. If not, download the required drivers from NVIDIA.

3 Microsoft Windows: Running and Using Marc and Mentat

Running Marc

This section describes the Marc usage on Microsoft Windows based machines. The Marc programs are mainly controlled by a batch script program called `run_marc.bat`, which is stored in the `<parent>` subdirectory `marc20XX\tools`.

The batch script will submit a job and must be executed in the directory where all relevant input and output files concerning the job are available. To use the batch script, each Marc job should have a unique name qualifier and all Marc output files connected to that job will use this same qualifier.

Marc input files should always be named `job_name.dat`, whereby the prefix `job_name` is the name qualifier which you are free to choose. The suffix `.dat` is obligatory.

To actually submit a Marc job from an MS-DOS Command Prompt window, the following command should be used. The single input line is split over multiple lines for clarity:

run_marc	-jid	job_name (required as minimum)
	`-rid`	restart file name
	`-pid`	post file name
	`-sid`	substructure file name
	`-prog`	name of a previously saved user executable
	`-user`	user subroutine name
	`-save`	save user executable
	`-back`	run in background
	`-vf`	view factor file name
	`-def`	defaults file name
	`-nprocd`	number of domains
	`-nprocds`	number of domains in Single Input File mode
	`-nts`	number of threads for parallel matrix solver (same as `-nthread_solver` or `-nthread` or `-nt`)
	`-nte`	number of threads for element assembly and stress recovery (same as `-nthread_elem`)
	`-nsolver`	number of matrix solver tasks
	`-mpi`	mpi version
	`-dir`	directory where job I/O takes place
	`-sdir`	directory where the scratch files are located
	`-host`	host file name; used for distributed execution in a network
	`-ci`	copy input files to remote machines in a network
	`-cr`	copy post files back from remote machines in a network
	`-ml`	memory limit in Megabytes
	`-gpuid`	GPU card ID or auto keyword

Table 3-1 describes the meaning of these input options and Table 3-2 gives examples.

Table 3-1 run_marc Input Options

Keyword	Options	Description
-jid (-j)	job_name	Input file (job) name identification.
-prog (-pr)	progname	Run saved executable progname.marc from a previous job (see -user and -save).
-user (-u)	user_name	User subroutine user_name.f is used to generate a new executable program called user_name.marc (see -save and -prog).
-save (-sa)	**no** yes	Do not save the new executable program user_name.marc. Save the executable program user_name.marc for a future time (see -prog and -user).
-rid -(r)	restart_name	Identification of previous job that created RESTART file.
-pid (-pi)	post_name	Identification of previous job that created the post file.
-sid (-si)	substructure	Identify the job that contains the solution to the external nodes of the superelement.
-back (-b)	**yes** no	Run Marc in the background. Run Marc in the foreground.
-def (-de)	default_file	File name containing user defined default data.
-nprocd (-np)	number	Number of domains for parallel processing.
-nprocds (-nps)	number	Number of domains for parallel processing using a Single Input File.
-nts (-nthread)	number	Number of threads per parallel matrix solver (8, 9, and 11).
-nte	number	Number of threads used for parallel matrix assembly and stress recovery.
-nsolver	number	Number of processes to use for the MUMPS parallel matrix solver (solver 12).
-dir	directory_name	Pathname to directory where the job I/O should take place. Defaults to current directory.
-sdir	directory_name	Directory where scratch files are placed. Defaults to -dir.
-host (-ho)	hostfile	Specify the name of the host file for running over a network (default is execution on one machine only in which case this option is not needed).
-ci	**yes** no	Copy input files automatically to remote hosts for a network run, if necessary.
-cr	**yes** no	Copy post files automatically from remote hosts used for a network run, if necessary.
-vf	viewfactor_name	Name of file containing viewfactors for radiation from previous analysis or from Mentat using either the Monte Carlo or Hemicube method.

Table 3-1 run_marc Input Options (continued)

Keyword	Options	Description		
-mpi	intelmpi msmpi	Select which version of mpi will be used for Domain Decomposition or the MUMPS solver. If not specified, the appropriate default shown below will be used:		
		Platform	Default MPI	Alternative MPI
		Windows 64	intelmpi	msmpi
-ml	memlimit	Memory limit for deciding if the solver should go out-of-core. Specified in Mbyte. Defaults to the physical amount of memory on the machine.		
-gpuid	<id:id:id> or auto	GPU card number or auto for automatic selection by program. For parallel (DDM) jobs, multiple card IDs can be specified if available.		

Table 3-2 Examples of Running Marc Jobs

Examples of running Marc jobs	Description:
run_marc -jid e2x1	Runs the job e2x1, the input file e2x1.dat resides in the current working directory.
run_marc -jid e2x14 -user u2x14 -save yes	Runs the job e2x14, using the user subroutine u2x14.f and the input file e2x14.dat. An executable program named u2x14.dat will be saved after completion of the job.
run_marc -jid e2x14a -prog u2x14	Runs the job e2x14a using the executable produced by job e2x14.
run_marc -jid e3x2a	Runs the job e3x2a.
run_marc -jid e3x2b -rid e3x2a	Performs a restart job using the results of the previous job e3x2a.
run_marc -jid e2x1 -nprocd 2	Runs a two processor job on a single parallel machine.
run_marc -jid e2x1 -nprocd 2 -host hostfile	Runs a two processor job over a network. The hosts are specified in the file hostfile (refer to the Microsoft Windows: Marc Parallel Network for runs on a network of machines.

Running Mentat

This section describes the Mentat usage on Microsoft Windows machines. The Mentat program is started by a batch script called **mentat.bat** which is stored in the <parent>\mentat20XX\bin directory. It may also be started by using the Marc Mentat 20XX menu item in the **Start** menu in the MSC.Software folder or by selecting the Marc Mentat 20XX icon that the **Setup** program created on your desktop.

You need not start the batch script from a specific directory.

The Mentat program creates the default files in your current working directory; i.e. where you are located at the time of starting the Mentat program (if run from the command line), or the Start in directory specified for the shortcut if started from the Marc Mentat 20XX icon.

Note:	Since Mentat is started by running a batch script, it is not possible to use a UNC path as the current working directory. It is recommended to map a UNC path to a drive letter, otherwise Mentat would use %**WINDIR**% (usually C:\Windows) as the current working directory.

The batch script `mentat.bat` contains a number of arguments which are passed on to the Mentat program. Table 3-3 gives the meaning of these input options. You are free to alter these commands to suit your preference.

Table 3-3 Mentat Input Options

Keyword	Option	Description
-mode	preset1/preset2	This will launch the application in the selected preset mode. For preset1, which is the default, the traditional theme, the traditional mouse button scheme and complete picking will be active. For preset2, the dark theme, the auto-dynamic mouse button scheme and partial picking will be active.
-ar	area ratio	This option sets the initial window size to the given fraction of the available space on the desktop.
-xr	horizontal ratio	This option sets the initial width of the window to the given fraction of the available width on the desktop [default: 0.92]
-yr	vertical ratio	This option sets the initial height of the window to the given fraction of the available height on the desktop [default: 0.92].
-aspr	aspect ratio	This option sets the aspect ratio (width over height) of the window [default: 1.6].
-maximize		Starts up Marc Mentat maximized.
-minimize		Starts up Marc Mentat minimized (iconified).
-bg		This option will run Mentat in the background.
-bp	`DIR\bin`	Directory path name where the external Mentat programs and shell scripts are located.
-compile	binary_menu_filename	This is used to compile ASCII menu files into a binary menu file. The -mf option would be used to specify the new binary file. To recompile the default binary menu file `main.msb`, compile as: bin\mentat -compile menus\win64\main.msb
-dr	True/False	Enable/disable direct rendering. Enabling this option will make OpenGL bypass the underlying window system and render directly from hardware to the screen, if this is supported by the system. The default is **True**.
-fn	font	Default font type.

Table 3-3 Mentat Input Options (continued)

Keyword	Option	Description
-fngr	font	Font used by the graphics windows (overrules the font selected via the -fn option).
-fnme	font	Font used by the menus (overrules the font selected via the -fn option).
-gradient		This option switches on a gradient background when displaying the various windows (Model, Table, History Plot, etc.).

-hd	*name filename*	Define a document viewer with the given *name* for viewing a document that consists of a set of HTML files. The *filename* must be the name of an XML file that defines the contents of the document. A menu item to open the viewer can be created by adding an `<html_doc>` element that references the viewer (via the "name" attribute) to the appropriate `<menu>` in menus\menubar.xml.
		The XML file must have the following contents:
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<html_doc>`
		`  <proc_root env="DEMO_DIR"/>`
		`  <title_page href="index.html"/>`
		`  <contents>`
		`  <chapter href="file1.html"/>`
		`  <chapter href="file2.html"/>`
		`  </contents>`
		`</html_doc>`
		The "href" attributes of the `<title_page>` and `<chapter>` elements specify, respectively, the path to the HTML file of the title page and the HTML files of the chapters of the document. The file paths must be given relative to the directory in which the XML file is located.
		For example, the Mentat startup script defines the document viewer for the User's Guide as follows:
		`-hd ug help\ug\ug.xml`
		and menubar.xml contains the following element to open the viewer:
		`<html_doc name="ug"`
		`title="User's Guide"/>`

-help		All of the options.
-hide_dialog		Hide the dialog window at startup.
-hide_dynamic_menu		Hide the dynamic window at startup.
-hide_main_menu		Hide the main window at startup.
-hp	`DIR\help`	Directory path name where the help files are located.

Table 3-3 Mentat Input Options (continued)

Keyword	Option	Description
-lf	*filename*	Specify the Mentat log file name.
-license_release	number	Enable Mentat to release its licenses if the program is inactive for the specified amount of time. The number is in minutes and must be greater than one.
-mf	`main.ms`	The name of the startup menu file.
-ml	`DIR\material`	Directory path name where the material files are located.
-mp	`DIR\menus`	Directory path name where the menu files are located.
-multiundo	off/on	This option enables the user to either set the one-level UNDO option (off), with data saved in memory, or the multi-level UNDO option (on), with data saved on disk [default: **on**].
-undo_levels	number	Number of UNDO levels when the multi-level UNDO option is active. The minimum is 1, the maximum is 50 [default: 10].
-nosolidmodeling		Utilize the Mentat_Parasolid_CAD which allows geometric models to be imported but does not allow solid geometry editing.
-num_session_files	number	Number of copies of Mentat session files (i.e. files with the extension .log and .proc) which will be left in the directory where Mentat is started. The minimum is 1, the maximum is 100000 [default: 3]. The session files are named mentat.*.log and mentat.*.proc. This option is omitted if the option **-lf** is used to point to a non-default Mentat log file.
-path	directory_name	Provides a directory in which Mentat searches when opening an existing input file. Multiple directories can be specified as follows: **-path directory_1 -path directory_2** etc.
-pr	*filename*	Any additional set-up commands you wish to add. Store these in a procedure file containing the Mentat commands.
-ra		This reads all of the ASCII Menu files.
-rf	*filename*	Record the Mentat commands in the procedure file `filename`.
-szgr	width height	Set the size (width and height in pixels) of the graphics area.
-ti	title	Append `title` to the name of the window.
-unicode		Allow the use of unicode characters in load case titles, job titles, annotations and user-defined names of results file variables.

Making Changes to the Marc Programs

Modifying the MPI Setting: selecting MPI version

The MPI version used by default is defined in the table below. It also lists an alternative MPI version to use for the respective platform. The default MPI version to use can be set by creating a `run_marc_defaults` file (refer Mentat Interfaces 69) and enter:

```
MARC_MPI alternative_mpi
```

for setting a new default. Use the exact word as given in then Alterative MPI colum. The MPI version can also be set with the command line option `-mpi` to `run_marc`. The command line option overrides the value set in `run_marc_defaults`.

Platform	Default MPI	Alternative MPI
Windows 64	intelmpi	msmpi

Setting defaults with the **`run_marc_defaults`** file

A file called `run_marc_defaults` located in the user's home directory (as given by the combined environment variables %HOMEDRIVE%%HOMEPATH%) or the tools directory of the Marc installation can be used for setting program defaults. If both are present, then the setting in the user's home directory will be used.

The following setting is currently available:

`MARC_MPI`	as given in the table in this section

This selects the MPI version to use.

Mentat Interfaces

This section contains information about the different interfaces available for different purposes in Mentat.

Mentat External Programs

Mentat supports a number of CAD interfaces: IGES, Patran, Ideas, VDAFS. These interfaces are programmed in external programs which are called from within Mentat. The interface programs are run from scripts stored in the `<parent>` subdirectory `mentat20XX\bin`. These programs read the data files in their native format and translate the contents into a Mentat model file. This file is subsequently read by Mentat. The external programs are called from within Mentat by means of the FILE submenu.

Jobs

The subdirectory `bin` contains batch script files to start a Marc job using the following batch scripts:

```
submit1.bat, submit2.bat, submit3.bat
```

These batch scripts are called by means of the buttons in the `job` menu. You may alter these files to suit your environment.

Plotter Interface

Because of the many variations in plotting environments, we have created plotting interfaces in the form of batch scripts that operate from within Mentat. Currently, Mentat recognizes the following plotting formats:

- PostScript
- WinDump (translated into Windows bitmap (`.BMP`) format)

This section describes a template batch script for each of the formats mentioned above. They are located in the `mentat20XX\bin` directory and are named as follows:

```
pscolor1.bat, pscolor2.bat, pscolor3.bat
psgray1.bat, psgray2.bat, psgray3.bat
xdump1.bat, xdump2.bat, xdump3.bat
```

PostScript

The PostScript function is activated by pressing the **Gray** or **Color Print** button from the **UTILS** menu on the **POSTSCRIPT** panel. The program captures the graphics portion of the screen into a file and sends this file to a PostScript printer using the `psgray` or `pscolor` batch scripts located in the `mentat20XX\bin` directory. In the example listed below, the file is sent to the printer LPT1. This can be a printer attached locally, or located somewhere on the network. After the file is sent, it is removed from disk automatically.

```
print /D:LPT1: %1
del %1
```

The argument `%1` is the filename handed to the batch script by Mentat. If there is more than one printer on-line, the `pscolor2.bat` and `pcolor3.bat` batch scripts may be used to address these other printers.

You can use the `setup_printer` program to configure a network printer. Run `setup_printer` for more details.

Edit

The `edit_window.bat` batch script is used to control the editor associated with the **EDIT** commands. It is possible to change the type of editor, for example, from notepad to emacs.

System Shell

The `system_window.bat` batch script is used to control the type of window opened with the `system_shell` command.

AVI Playback

The `avi_window.bat` batch script is used to control the program opened with the `play_avi` command.

MPEG Playback

The `mpeg_window.bat` batch script is used to control the program opened with the `play_mpeg` command.

4 Microsoft Windows: Marc Parallel Network

Hardware and Software Requirements

The list of supported capabilities in parallel can be found in "Supported and Unsupported Features" in Chapter 12 of *Marc Volume A, Theory and User Information*.

The list of supported platforms can be found in "Supported Platforms" in Chapter 1 of this guide.

To use Intel MPI, observe the following:

1. From the **Command Prompt** window, go into the directory

   ```
   C:\Program
   Files\MSC.Software\Marc\20XX.0.0\marc20XX\intelmpi\win64\bin\intelmpi\win64\bin
   ```

2. Type `.\mpiexec.exe -register`

Note:	When opening a **Command Prompt** window, always start the Command Prompt window by right clicking the icon and selecting **Run as Administrator**.

3. Enter your account (use a network account if you intend to run DMP jobs) and the password at the command prompt.

   account (domain\user) [<domain>\<username>]:

   password:

   confirm password:

   The following message appears in the **Command Prompt** window.

   `Password encrypted into the Registry` will show in the **Command Prompt** window.

Note:	If you change your account and password, you will need to repeat this step.

4. Type:

   ```
   .\hydra_service -install
   ```

5. To run DMP (network parallel jobs) using Intel MPI, create a host file using the following format:

   ```
   host1 n1 workdir1
   host2 n2 workdir2
   host3 n3 workdir3
   ```

   Use the directory name (UNC) as echoed by typing `net share` on your system for `workdir`.

   For example:

   ```
   sv-x861 2
   sv-x862 2 \\sv-x862\test2
   ```

   The DMP job will run using two processors on node `sv-x861` in the directory where the job is started and two processors on node `sv-x862` in the shared directory `test2`.

   The `test1` and `test2` directories in nodes `sv-x861` and `sv-x862` as well as the `marc20XX` directory in node `sv-x861` will need to be shared with a general permission to all users.

   The default MPI is Intel MPI 2017.1 for Windows. The other MPI option for the 64-bit Windows platform is MS-MPI.

6. To use MS-MPI, install Microsoft MPI; the version is listed in the `include_win64.bat` file in the `marc tools` directory.

7. Turn off the Windows firewall in your cluster and share the `marc` directory with a general permission to all users.

   Although no specific hardware requirements exist to run a job in parallel, it is preferable that for distributed parallel processing to have fast network connections between the machines. It is recommended that the network should have a speed of at least 100 MBit per second. If only two machines are to be used, a hub or a cross-over cable can be used to connect them. If more than two machines are to be used, a switch is preferable. TCP/IP is used for communications. Refer to the `include_win64.bat` file in the tools directory for requirements on O/S, compilers, etc. for more details regarding running on Microsoft Windows.

Definitions

The definition of some of the terms are as follows:

Table 4-1 Definitions

Sr. No	Term	Definition
1	Root machine	The machine on which Marc job is started.
2	Remote Machine	Any machine other than the root machine which is part of a distributed Marc run on the network.
3	Shared installation	Marc is installed in a UNC shared directory on one machine only. The other machines can access the Marc executable since the directory is shared.
4	Distributed installation	Marc is installed on all machines. Each machine accesses its own Marc executable.
5	Distributed execution	Marc is run on multiple machines which are connected with a network. Each machine loads the Marc executable either from a shared or a local directory and then executes the executable.
6	Shared I/O	Marc reads and writes data in a UNC shared directory. Each Marc executable running on the network reads/writes to the same directory.
7	Distributed I/O	Marc reads and writes data in a directory located on each machine. Transfer of data files and post files between the root machine and remote machines is done automatically by Marc.
8	UNC	Uniform Naming Convention.

Network Configuration

Marc needs to be installed on the root machine where the installation directory is UNC shared (shared installation). Marc can also be installed on the remote machines which then use their own executable (distributed installation). For this case the same path name must be used on all machines. The root machine is the one on which the Marc job is started, typically from within Mentat. The remote machines can be located anywhere as long as they are connected to the network. The working directory on each machine can be a shared directory on any machine on the network (shared I/O) or it can be a local directory on the hard disk of each machine in the analysis (distributed I/O). The User Notes 77 describes how to specify what working directory to use.

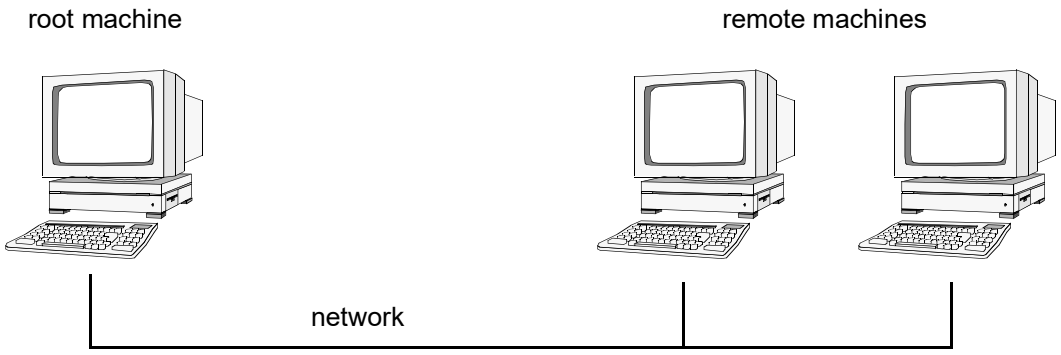


Figure 4-1 Network Configuration

Installation Notes

This part describes the specific steps needed to install and set up a network version of Marc. For general information on Marc installation, see Microsoft Windows: Marc and Mentat Installation.

Steps 1–6: Must be performed as Administrator.

Step 1: **Install Marc on the root machine.**

Step 2: **Make sure that the installation directory on the root machine is properly shared, so that the remote machines can access it. Assuming that Marc is installed under `C:\Program Files\MSC.Software`, share this directory by associating a UNC share name with it as follows. Use My Computer and locate the directory to be shared. Right click on the directory and choose Sharing, Choose Share As and give it a Share Name (this is the UNC name) and click OK.**

Marc restricts the UNC name to have a maximum of 10 characters and the name of the shared directory to have a maximum of 30 characters. If necessary, a directory higher up in the path can be shared (for instance, `C:\` instead of `C:\Program Files\MSC.Software`). It is sufficient that either `C:\` or `C:\Program Files\MSC.Software` is shared.

Step 3: **On the remote machines you can perform a full Marc installation or an installation of the Intel MPI Library Process Manager service only. In the latter case, the remote machine will be using the Marc installation of the root machine via the UNC share name.**

The Process Manager service must be installed and running on all hosts involved a distributed job across the network. To install the Process Manager on a machine without a full Marc installation, copy the `hydra_service.exe` program from the `intelmpi\win64\bin` directory (on the root machine) to a local directory on the remote machine. Open a **Command Prompt** window on the remote machine, go into that directory and execute:

```
.\hydra_service -install
```

Step 4: **Create a Marc file with the shared naming information. From the Command Prompt, change directory to the `tools` directory in the Marc installation directory (here assumed to be `C:\Program Files\MSC.Software`).**

```
c:
cd \Program Files\MSC.Software\Marc\20XX.0.0\marc20XX\tools
net share > marc.net
```

This file has to be recreated each time the shared name of the installation directory is changed. The file `marc.net` contains the connection between the path names on the root machine and the UNC names, and can be created only by the Administrator. If this file does not exist or contains outdated information, the remote machines will not be able to find the executable on the root machine.

Step 5: **Test the Marc installation for single processor execution by typing from the root machine:**

```
c:
cd \Program
Files\MSC.Software\Marc\20XX.0.0\marc20XX\test_ddm\exmpl2\exmpl2_1
run_marc -j cyl2
```

and Marc should exit in about three minutes if it is a successful run.

Step 6: **Test the Marc installation for multi-processor, distributed execution. Assume the host name of the root machine is host1 and one of the remote machine is host2. Type from host1:**

```
c:
cd \Program
Files\MSC.Software\Marc\20XX.0.0\marc20XX\test_ddm\exmpl2\exmpl2_2
```

and edit the file `hostfile` in this directory by replacing workdir with

```
c:\Program Files\MSC.Software\Marc\20XX.0.0\marc20XX\test_ddm\exmpl2\exmpl2_2
```

The host names and directory names should be replaced with the names on the current system. Finally, type:

```
run_marc -j cyl2 -nprocd 2 -host hostfile
```

and Marc should exit in about two minutes if it is a successful parallel run on host1 and host2 using one processor on each. If the job stalls or hangs at start-up time, exit it by typing `control-C` in the window in which is was started. See User Notes 77 in this chapter.

User Notes

This section assumes that the network version of Marc has been successfully installed on two machines that are to be used in a distributed analysis and that the appropriate Marc licenses are in order. Assume that host1 is the host name of the machine on which Mentat is running and from which the job is to be started (the root machine). The host name of the other machine (the remote machine) is host2.

How to Run a Network Job

First make sure that the two machines are properly connected. From host1, access host2 using Internet Explorer. If this is not possible, a network run will not be possible.
See User Notes 77 in this case.

Note:	The current version does not support the IPV6 protocol.

In order to perform an analysis over a network, you have to create a special file called the `host file`. This file defines which machines are to be used, how many processes are to run on each, what working directory should be used, and where the Marc executable can be found on each machine. The host file can be selected and edited in Mentat and the Marc job started as usual from within Mentat (see the following example). If Marc is run from the command line, it is done as for a normal run using an additional command line option.

For example:

```
run_marc -b no -jid test -nprocd 2 -host hostfile1
```

will run the two-processor job `test.dat` using the specification in the file `hostfile1`. No specific name or extension is used for the host file except that the name `jobid.host` (in this example `test.host`) must be avoided since it is used internally by Marc.

Specification of the Host File

The host file has the following general format:

```
host1 n1
host2 n2 workdir2
host3 n3 workdir3
```

Each line must start at column 1 (no initial blanks). Blank lines and lines beginning with a # (number symbol) are ignored.

- The first entry is the host name of a machine to be used in the analysis. The root machine must be listed first and each machine must only occur once.
- The second entry specifies the number of processes to run on the machine specified in the first entry. The sum of the number of processes given in the host file must equal the number of domains used. In a five-domain job, it is required that `n1+n2+n3=5`.
- The third entry specifies the working directory to use on this host. This is where the I/O for this host takes place. The Marc input files for this machine must be in this directory and the results files for this machine are created in this directory.

The different domains of the Marc job are associated with the different machines as follows. Suppose a five-domain job `test` is run using a host file defined as:

```
host1 2
host2 1 workdir2
host3 2 workdir3
```

with appropriate definitions of the third entry, see below. There will be six Marc input files associated with this job: `test.dat, 1test.dat, …, 5test.dat`. Domains 1 and 2 will be associated with host1, domain3 with host2 and domains 4 and 5 with host3.

Shared I/O

Suppose a job is to be run on host1 and host2. A shared directory on host1 is to be used for I/O. The UNC sharename for this directory is assumed to be `dir7`. The host file for a two-processor job would be:

```
host1 1
host2 1 \\host1\dir7
```

To verify that the work directory is accessible, use **Internet Explorer** on `host2` and browse to `host1->dir7`. The directory seen should be the same one as the working directory on host1.

It is also possible to use only two entries for each host in the host file (host name and number of processes). This requires that the working directory is shared and that the sharing information is up to date in the file `tools\marc.net` in the Marc installation directory.

Distributed I/O

If you want to have the I/O to be local on host2, specify the host file as:

```
host1 1
host2 1 D:\users\dir5
```

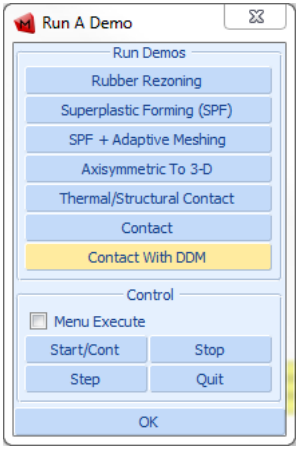
The I/O on host2 will now take place in the directory `D:\users\dir5` on the hard disk of host2. For this case, the Marc input files are transferred to `D:\users\dir5 on host2` before the job is started, and the results files are transferred back after the analysis for postprocessing. This transfer of files is done by Marc automatically.

Example

The definitions for a network run with Mentat is demonstrated with a simple example. We assume the simplest case where both the working directory and installation directories are shared.

1. Enter the menu **Help → Run a Demo...**

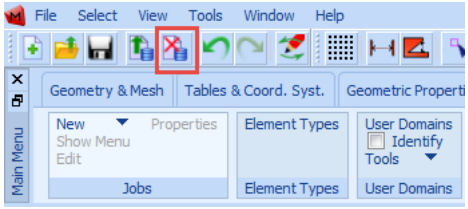
   A **Run A Demo** dialog appears.



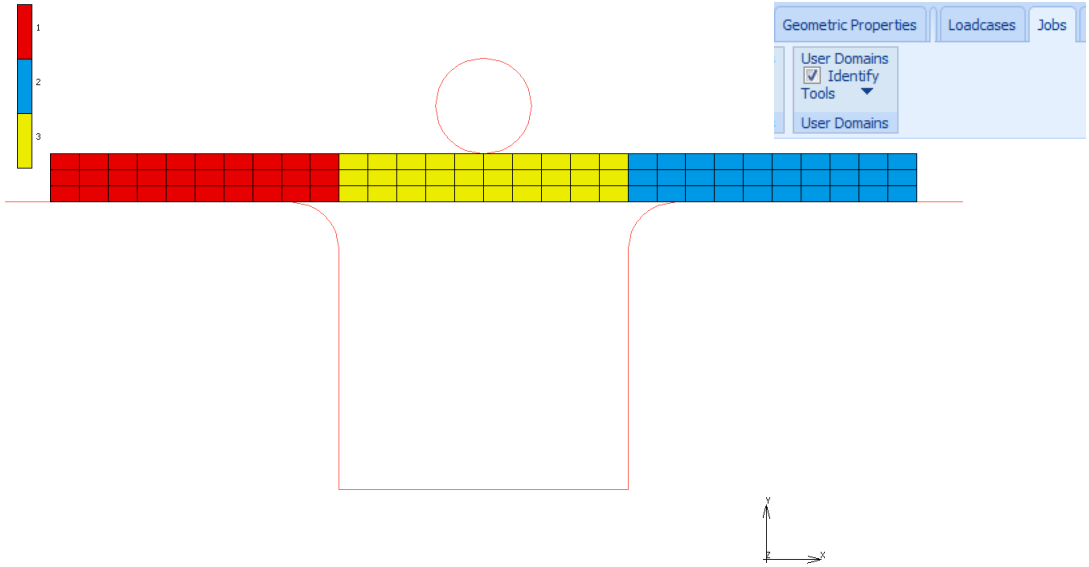
2. Select the example **Contact With DDM**.

Note:	The input creation for the demo problem automatically starts as you select any of the problems. It continues till the results are seen. To stop/pause the automatic inputs creation, you need to click on the **Stop** button. Once you stop the procedure use the **Step** button to see a step-by-step input creation.

3. After the process is completed. Click **Close result file** button.

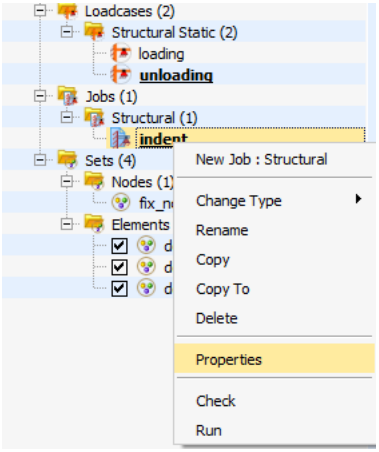


4. Select **Jobs** tab and check **Identify** to confirm your two domains as shown in the image as follows:

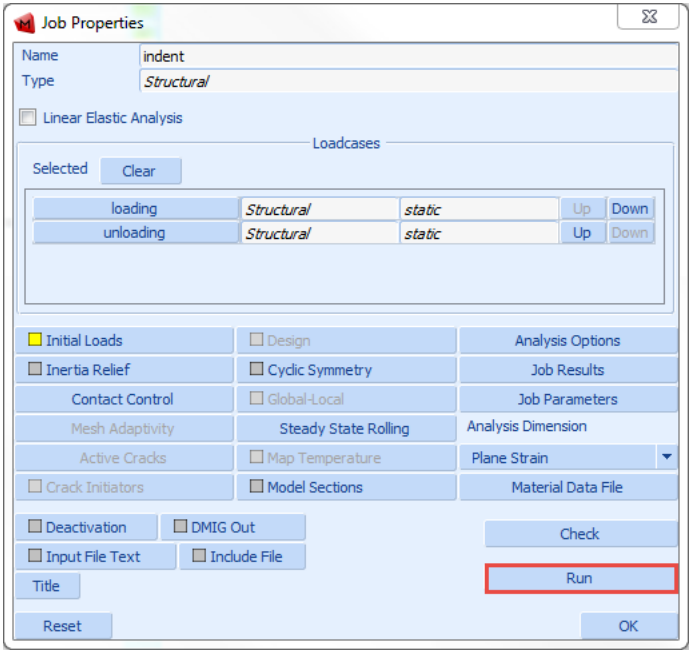


The complete model workflow can be seen in the **Model Navigator**.

5. Right click on **Indent** and select **Properties** from the drop down list.

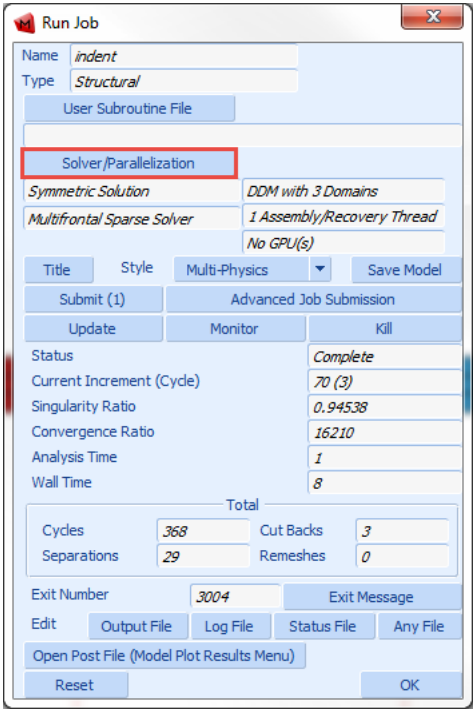


The **Job Properties** dialog appears.



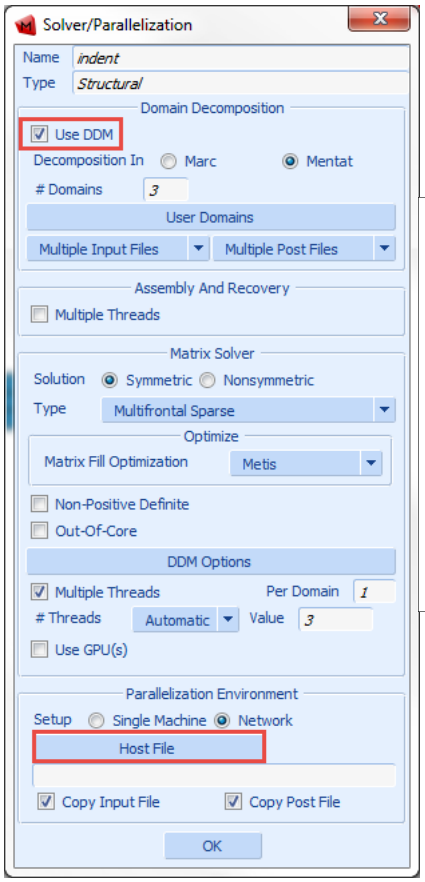
6. Click **Run**.

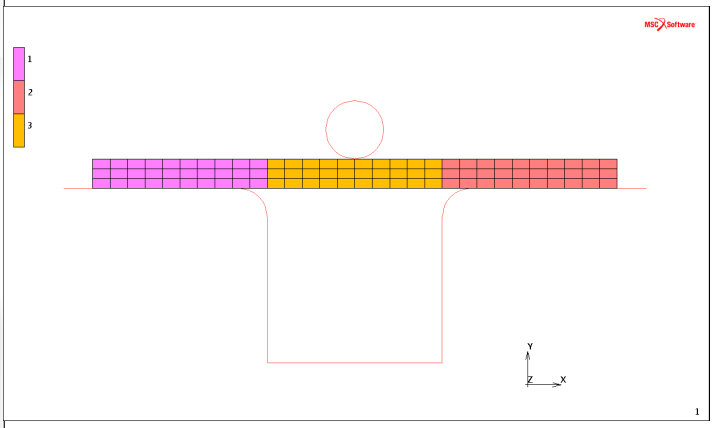
The **Run Job** dialog appears.



7. Click **Solver/Parallelization**.

The **Solver/Parallelization** dialog appears.





Ensure the **Use DDM** option is selected. In this case three domains are used.

8. Select **Network** option from the submenu.

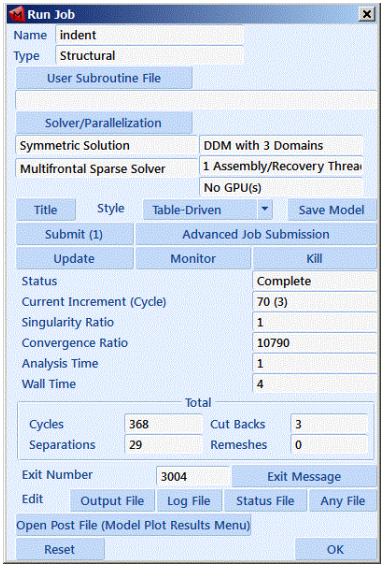
   An additional option to select the host file appears just below Network.

9. Click on the **Host File** and select the file called `hostfile` from the install directory and modify it to contain:

   ```
   host1 1
   host2 1  workdir installdir
   ```

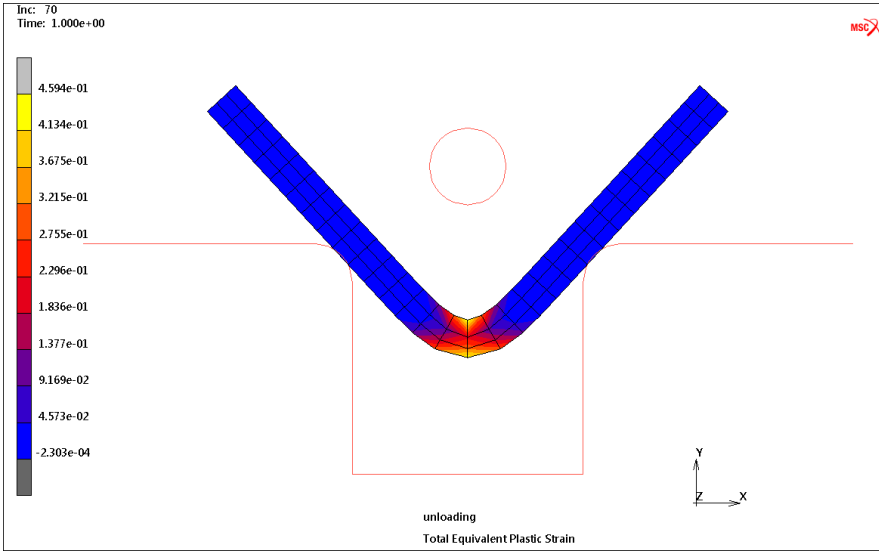
10. Set the working directory according to the share names on the current system. The host file places one domain on hostname1 and one domain on hostname2.

11. Run Marc from within Mentat using the **Submit 1** button. When the analysis is complete, the following dialog box appears:



12. Check your results by clicking the **Open Post File** (**Results Menu**) button.

    A plot of the Total Equivalent Plastic Strain in the final deformed configuration is shown as follows:



Marc creates a post file associated with each domain as well as a root post file associated with the job id. For the previous model, `1model1_job1.t16` and `2model1_job1.t16` are the processor files, while `model1_job1.t16` is the root file.

To post process the entire model, select `model1_job1.t16` as the post process file. If the model is very large, it may be convenient to view only a portion of the model by selecting any one of the processor post files, such as `2model1_job1.t16`. This file contains only data associated with domain 2as selected in the Domain Decomposition menu. As specified in the host file, this file was created by host2.

Shared vs. Distributed I/O

For jobs with very large post or restart files, it is usually more efficient to use distributed I/O. With distributed I/O, the input files, and the post files are located on the host's local disks. Marc, by default, automatically transfers the input files and the post files to and from the remote host if needed. It is possible to suppress this transferring with two buttons in the Network settings in the **JOBS** menu in Mentat.

To run a job using distributed I/O, specify a local directory in the host file:

```
host1 2
host2 1    d:\workdir
```

Jobs with User Subroutine

User subroutines can be used as usual. If local directories are used on remote hosts (distributed I/O), then the working directory on the first host must be shared so the other hosts can find the executable.

5 Windows Troubleshooting

General

This chapter contains information about troubleshooting general and problems according to FLEXlm version Helium.

The process for Helium(11.16.3.0) license manager is the same but the paths may vary. Use the paths in the following section to identify the difference between both the versions.

FLEXlm 11.13 VS Helium(11.16.3.0) License Manager

This section contains information and directory paths according to FLEXlm 11.13 license manager. The changes in the default paths for Helium license manager in comparison with FLEXlm 11.13 are as in the following table:

11.13	Helium (11.16.3.0)
C:\MSC.Software\MSC.Licensing\11.13\msc	C:\Program Files\MSC.Software\MSC Licensing\Helium\msc
C:\MSC.Software\MSC.Licensing\11.13\lmtools	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmtools
C:\MSC.Software\MSC.Licensing\11.13\installs	C:\Program Files\MSC.Software\MSC Licensing\Helium\installs
C:\MSC.Software\MSC.Licensing\11.13\lmgrd	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmgrd
C:\MSC.Software\MSC.Licensing\11.13\LOG	C:\MSC.Software\MSC Licensing\Helium\LOG
C:\MSC.Software\MSC.Licensing\11.13\lmutil	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmutil
C:\MSC.Software\MSC.Licensing\11.13\msclic.ini	C:\Program Files\MSC.Software\MSC Licensing\Helium\msclic.ini

Access is denied

Cause	No write permissions to the file or directories to which you are installing.
User Action	■ Do the following: Open up an MS-DOS Command Prompt window. 1. **cd** to the directory in which you are installing Marc 2. Run the **attrib** program to remove the read-only attributes (with the **-R** option). ■ Use the **File Manager's Security** menu to change ownership of files and directories and to change permissions on them.

Error during move process

Cause	An application or a file open during the installation process that the installation is trying to overwrite.
User Action	Exit all applications and documents during installation and restart the installation.

Security failed or Marc exit 67

Cause	1. The environment settings are not set properly. 2. You are attempting to run on a machine that according to the Marc password(s) you are not allowed to use 3. Your license period has expired
User Information	In this case you cannot access or read the `license.dat` file in the `C:\Program Files\MSC.Software\MSC Licensing\Helium` subdirectory. Every Marc user should have read and write rights for the `<parent>` subdirectory`11.13`
User Actions	■ Log out and log back in and try again. ■ Check that the environment variable `MSC_LICENSE_FILE` is set to a valid license file. ■ Install your passwords before security succeeds. ■ Check that the FLEXlm license manager has been started from the FLEXlm Configuration Utility applet in the **Start** menu under MSC.Software → MSC.Licensing 11.16. This must be done AFTER you have saved your `license.dat` file in the `MSC.Software\MSC Licensing\Helium` directory. Test that it is working by pressing the Status button in the Control menu ■ Check time and date on your machine. ■ Use the File Manager's Security menu to change Ownership of files and directories, and to change Permissions on them.

License Manager does not start

Cause	1. Incorrect hostname on the SERVER line. 2. Incorrect path to the license daemon **MSC.exe**
User Actions	■ Check that the hostname on the SERVER line is correct ■ Check that the DAEMON line contains the correct path to the license daemon **MSC.exe** ■ If you had an older version of the FLEXlm license manager installed, the new installation may replace it. If the Use NT Services button was previously checked, you should uncheck this button, start the license manager, and then select the Use NT Services button

Link failed

Cause	1. No Fortran compiler or Fortran libraries 2. Your PATH or LIB environment variables settings do not point to the proper location for your Fortran compiler.
User Information	User subroutine causes compiler errors
User Actions	■ Make Fortran compiler or Fortran libraries available ■ Point your PATH or LIB environment variables settings to the proper location for your Fortran compiler.

Fortran files are not being compiled

Cause	The Fortran compiler is not in your search path, or the INCLUDE and LIB environment variables are not set.
User Information	The Intel Fortran compiler requires that the Microsoft .NET 2017 framework is installed and the LIB environment variable setting includes the required dependency libraries. Running the compilervars script mentioned below to make sure that the environment is correctly set up

User Actions	Verify that your settings are correct with the **set** command. Run the following scripts to set it up:
	
	`C:\Program Files (x86)\IntelSWTools\compilers_and_libraries_2019\windows\bin\`
	`compilervars intel64 vs2017\`
	(or substitute your installation directory)

User subroutines are not being called

Cause	For Fortran, the argument list for subroutines does not match exactly.
User Information	If the argument does not match exactly, your subroutine will not replace the existing subroutine in the Marc Library. The linker will continue to use the subroutine that is defined in the Marc Library, and since your routine will not be linked in, it will never be called.
User Actions	Make Fortran argument list match exactly.

Marc Parallel Network

This section contains information about troubleshooting specific to Marc parallel network.

General

If you face any problems in launching Marc Menatat, make sure that:

1. The user ID that was registered using the mpiexec.exe utility exists on the root machine and all remote machines (see General). Also check that the password you entered is the same on all machines.

   Note that if you change your login password, you must register it again using mpiexec.exe
2. The remote machines have permission to read from the Marc installation on the root machine via the UNC sharename. For shared I/O, the remote machines also must have permission to read from and write to the shared (working) directory on the root host.
3. Your Marc and Marc Parallel licenses are valid.
4. The host names are valid.
5. The Intel Process Manager service is installed and running on all hosts involved in the distributed job across the network. Select **Start/Control Panel/Administrative Tools/Services** and look for Intel MPI® Library Process Manager, Intel. Make sure that it is as **Status:Started.**

   The typical error message that appears if the Process Manager service is not running on or more hosts is:

   ```
   abort: Unable to connect to 'hostname:8676'
   sock error: generic socket failure, error stack:
   ...
   ```

   Refer to Step 3 of the installation instructions (Installation Notes) on how to install the Process Manager on machines without a full Marc installation.

Running a Parallel Job when Not Connected to the Network

If you disconnect your system from the network and want to run a parallel job on that system, you will have to install the Microsoft Loopback Adapter. Do the following:

1. Go to Control Panel -->Add/Remove Hardware.
2. Select the hardware task you want to perform:

   ```
   Add/Troubleshoot a device
   ```
3. Choose a Hardware Device:

   ```
   Add a new device
   ```
4. Do you want Microsoft Windows to search for your new hardware?

   ```
   No, I want to select the hardware from a list
   ```
5. Select the type of hardware you want to install:

   ```
   Network adapters
   ```
6. Select Network Adapter:

```
Manufacturers:Microsoft
Network Adapter:Microsoft Loopback Adapter
```

It will now install the loopback adapter. You will have to enable/disable the loopback adapter as you remove/connect your machine to the network.

Running a Parallel job on a Windows System when not a member of a Domain

If you are running a parallel job on a Windows system that is not a member of a domain, you will have to modify a registry entry.

Using *regedt32*, look for the following key:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa
                        "forceguest" : REG_DWORD : 00000001
```

If you find this key, change the `REG_DWORD` value to 0. The name may also appear as ForceGuest.

If you do not have this registry entry, your system will function properly.

Running a Parallel Job on Windows

The RPC protocol does not permit anonymous requests to the RPC Endpoint Mapper but requires client requests be authenticated using Windows XP SP2 or later. This will cause an `Access is Denied` error when you attempt to run a Marc Parallel job.

Note: These settings are important to enable Microsoft Windows to run a parallel job. If they are not set properly, the error message `Host is unreachable (5)` or `Access is Denied (5)` will be issued.

As a work around for this problem, do the following:

1. From a command prompt, run **gpedit.msc**.
2. Select **Computer Configuration**
   a. Expand **Administrative Templates.**
   b. Expand **System.**
   c. Click **Remote Procedure Call.**
   d. Double click **RPC Endpoint Mapper Client Authentication.**
   e. Change the value to **Enabled**.

You will also need to check the Data Execution Prevention settings.

1. Right click **My Computer.**
2. Go to **Properties.**
   a. Select the **Advanced** tab.
   b. Select the Performance **Settings** button.

c. Select the **Data Execution Prevention** tab.

d. Select the button for **Turn on DEP for essential Windows programs and services only**

Required Privileges

Running a parallel job either locally or over a network requires certain user privileges. If the privileges are not sufficient, MPI will print an error code of 1314.

You will need to check the following settings from the **Control Panel**:

1. Select **Administrative Tools**
2. Select **Local Security Policy**
3. Check the following items under **Local Policies** → **User Rights Assignment**.

   Windows
   Adjust memory quotas for a process
   Replace a process level token

   You can add user names to these if needed by right clicking them and going to **Properties**.

Note:	The setting on the domain controller will also need to provide access to these settings for each user name, if the users login through a domain controller.

GPU Support on Windows

Hardware and Software Requirements

The list of supported solver types with the GPU option can be found in *Marc Volume A: Theory and User Information* Chapter 11: Solution Procedures for Nonlinear Systems in Table 11-3 and Table 11-4.

An NVIDIA driver is necessary if the GPGPU capability is to be used. The driver needs to be compatible with the one used in the Marc build; see Marc Volume A: Theory and User Information, Chapter 12: GPU Support in for the minimum driver version supported.

The GPU capability in Marc is developed using CUDA toolkit 11.0 Update 1 with supported compute capability ranging from 3.5 to 8.0. NVIDIA Tesla cards with higher compute capability values are recommended. Note that NVIDIA Quadro cards can also be used.

Deviceinfo Utility

A stand-alone utility `deviceinfo.bat` is provided in the tools directory to check available GPU cards on the machine. This utility provides information about the GPU cards and the installed NVIDIA drivers toolkit version. Typical output from the deviceinfo utility is shown below:

```
[ ** Marc GPGPU device diagnosis utility ** ]

Installed CUDA driver version is 11.0

Total 1 devices are detected on the machine

  Device id 0 : "Quadro M2200"

              Compute capability                : 5.2
              Number of Multiprocessors         : 8
              Number of Cores per Multiprocessor : 128
              Total Number of Cores             : 1024
              Total device memory (in MB)       : 4096
              TCC driver mode enabled           : No


  The supported compute capability for Marc GPGPU feature is from 3.5 to 8.0

  Below is the list of supported devices

      Device id 0               : "Quadro M2200"
      TCC driver mode enabled   : No*

      *Note:
          TCC drive mode is not enabled on device id 0. Hence device
          will not be detected for multi-host parallel (DDM) jobs.

          To use GPGPU device for multi-host DDM run (with Multifrontal solver),
          enable TCC mode. For more information on TCC mode please refer to
          NVIDIA help page.
          https://docs.nvidia.com/nsight-visual-studio-edition/reference/index.html#tesla-compute-cluster
```

Figure 5-1 Deviceinfo.bat Output

Mentat Support

Running the job with GPU may also be done from Mentat. The GPU menu is only displayed for supported solver types Multifrontal Sparse and Iterative Sparse. Note that solver multi-threading (which refers to CPU threads) is not supported when the GPU option is selected.

The GPU option in Mentat is shown in Figure 5-2. When the GPU option is chosen, there are two options available for GPU card selection:

a. **Automatic**: At run-time, Marc determines the GPU cards that are available and assesses the optimal card to be used.

b. **User:** The user can specify the GPU cards that need to be used for the job. The available device IDs can be determined by running the deviceinfo.bat utility.

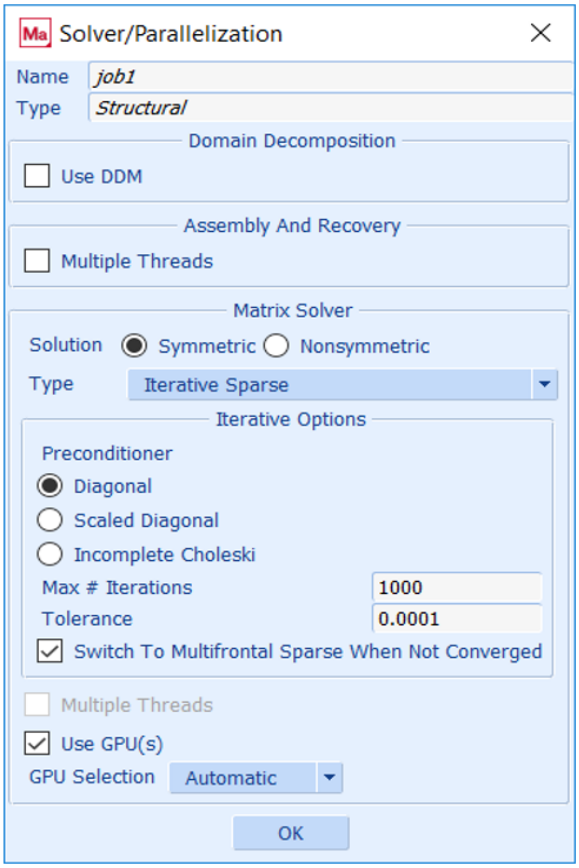


Figure 5-2 GPU Option in Mentat

Troubleshooting

Exit 63

Cause	The Marc version being used is not supported with the GPU version of the Multifrontal solver.
User Information	Non-GPU version of the Multi-frontal solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

Exit 64

Cause	Error occurs while attempting to use the GPU card specified by the user.
User Information	This error occurs while trying to use the GPU card specified by user.
User Action	Use the deviceinfo utility to check for support-ed and available GPU cards on machine.

Exit 65

Cause	The Marc version being used is not supported with the GPU version of the Iterative solver.
User Information	Non-GPU version of the Iterative solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

missing library "nvcuda.dll"?

Cause	This error occurs when required NVIDIA drivers are not available while running Marc with the GPU option.
User Information	To use GPU feature in Marc, necessary NVIDIA drivers need to be installed on machine.
User Action	Use deviceinfo utility to check if supported drivers are installed on machine. If not, download the required drivers from NVIDIA.

SECTION 2: LINUX

6 Linux: Prerequisites for Marc and Mentat

Installation Prerequisites

This section contains information regarding the hardware and software which are required in order to complete the installation successfully.

Before installing the software	The product is available for download at the MSC Solutions Download Center available at: https://mscsoftware.subscribenet.com 1. Select the product **Marc** and select the 20XX version on the next level. 2. Download the installer for your platform. 3. The FLEXlm security is installed separately. Select the product **MSC Licensing** and the version Helium(11.16.3.0) on the next level. 4. Download the installer for your platform. 5. Decide where you want the product to be installed and where to perform the installation. 6. First install security. If you are installing both Marc and Mentat, then first install Marc and then install Mentat. Marc requires approximately 1.1 GB of permanent disk storage capacity. Mentat requires approximately 1.3 GB of permanent disk storage capacity. Documentation, stored in the separate documentation directories `doc` and `examples`, contains approximately 350 MB of data.
Password protection	The Marc and Mentat version you have received is protected against illegal usage by means of Flexera Software's FLEXlm licensing software. You *cannot* run the program directly after you have installed the product from the installation media until you obtain a license file from MSC Software Corporation.
Multiple machines/NFS Fileserver	If you are installing Marc and/or Mentat on an NFS Fileserver, the install script needs to create directories in which to install Marc and/or Mentat; the default NFS export options do not allow this level of access by root. Two approaches are possible – do not install as root, or if *you must install as root, modify your NFS export options to include ~root=list (*where *list* can include hostnames and netgroups).

Should I be "root"?	■ You may need to be logged in as root if you want to create an optional `link` by which Marc and/or Mentat will be known system-wide under the name *marc20XX* and/or *mentat20XX*. This link will, by default, be placed in the directory `/usr/local/bin` to which you must have write permission. Logging in as `root` is one way of ensuring that you can create this link. Make sure that you have write permission to the installation directory before you start the installation script. ■ For NFS fileserver networks, read the previous paragraph.
Fortran compiler	A Fortran compiler is necessary if user subroutines are to be used. For other cases, no compiler is needed. The compiler needs to be compatible with the one used in the Marc build, see the *Marc and Mentat Release Guide* for a list of supported compilers.
NVIDIA driver	An NVIDIA driver is necessary if the GPGPU capability is to be used. For other cases, no special driver is needed. The NVIDIA driver needs to be compatible with the one used in the Marc build; see *Marc Volume A: Theory and User Information*, Chapter 13 for the minimum driver version supported.

Supported Platforms

The supported Linux platforms are listed below. The installation of a Fortran compiler is only required if you would like to run Marc with user subroutines.

Type	OS	Hardware	Fortran Version	Default MPI
Linux (64 bit)	RHEL 7.1/7.3/7.5/7.7 and SUSE 12 SP1/SP2/SP4	Intel EM64T or AMD Opteron	Intel XE 19.04[1]	Intel MPI 2019 Update 4
		Intel EM64T or AMD Opteron	Intel XE 19.04[1]	Intel MPI 2019 Update 4

[1] For user subroutines, Intel Fortran XE 19.04 is required.

7 Linux: Marc and Mentat Installation

Installation Procedure

The installation needs to be done in four steps:

1. Install MSC licensing Helium
2. Install Marc and Mentat
3. Install documentation
4. Verify installation

For old installation procedure, refer to Appendix. C: Linux: Marc/Mentat Files, Subdirectories and Installation

Install MSC Licensing Helium

1. Download the latest `.bin` file from the MSC Download Center.

Note:	To demonstrate the installation procedure, `/root/MSC/` this location is used as the installation directory for all three steps. You are advised to create a directory named MSC in your `home/root` directory. It is not mandatory to create this directory. You can create and install the application in any directory of your choice. The commands/user inputs are marked in **`bold`** for all descriptions related to terminal commands in this document. Rest of the script is auto generated. For more details refer to Typographical Conventions in Preface.

2. Start the installation procedure by using the command as follows:

```
[root@vm-tmrhel73 MSC]# ls
license.dat                          marc_20XX_linux_doc.tar.gz
marc_20XX_linux64_rh7.1_7.3.tar.gz   msc_licensing_helium_linux64.bin
[root@vm-tmrhel73 MSC]# ./msc_licensing_helium_linux64.bin
```

The following prompt appears on the screen.

3. Click **Yes**.

The *MSC Licensing Helium User's Guide* opens automatically.

- For detailed installation about licensing, refer to the *MSC Licensing Helium User's Guide*.
- For quick installation continue with the following procedure

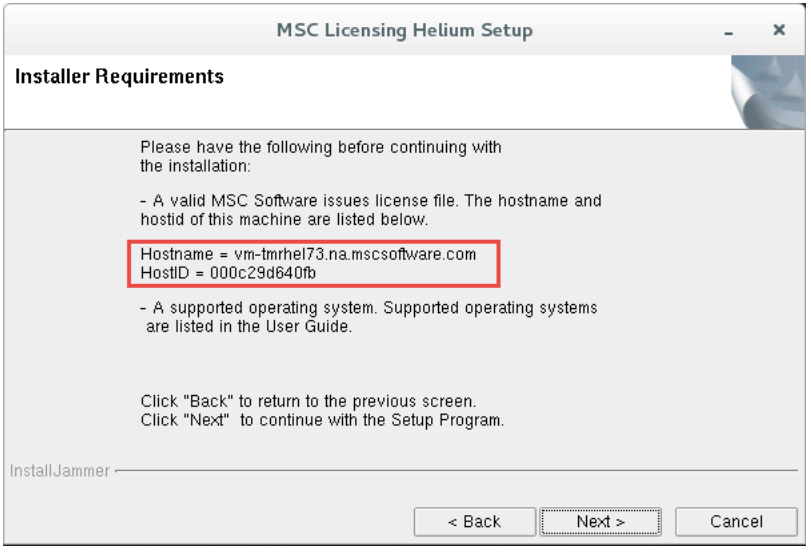
An **MSC Licensing Helium Setup** window appears.

4. Click **Next**
5. If your system already has the older version of the MSC Licensing Server, it will show the following message:

   ```
   11.13 License Server process currently running. Do you want to un-install 11.13 license server?
   ```

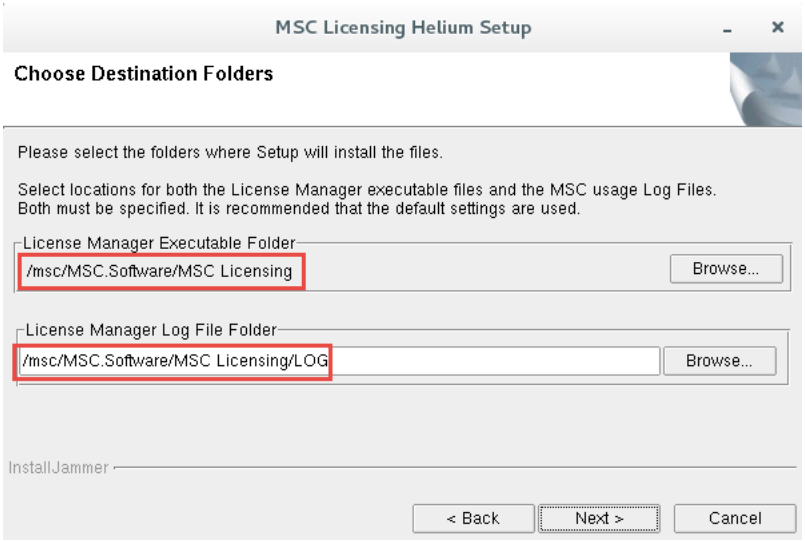
6. Click **Yes** to continue

   The **Installer Requirements** window appears.



7. Review the requirements and if all the requirements are fulfilled. Click **Next**.

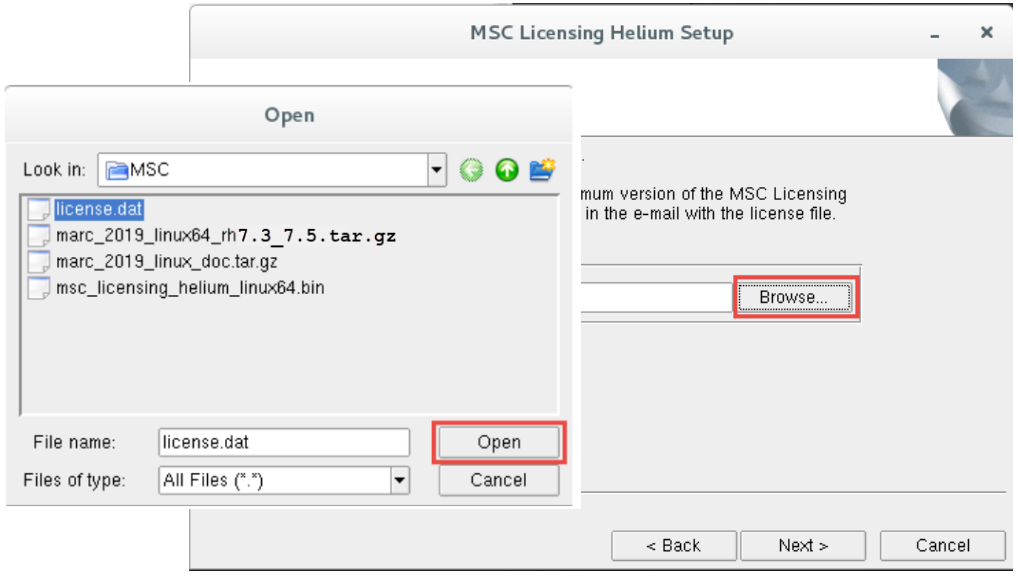
The **Choose Destination Folders** window appears.



- Review and note the default locations for both the folders.
- Keep the default paths as is (strongly recommended).

8. Click **Next**

   A window to select license file appears.



9. Click **Browse**. An **Open** dialog appears.
   a. Select the required `license.dat` file.

b. Click **Open**

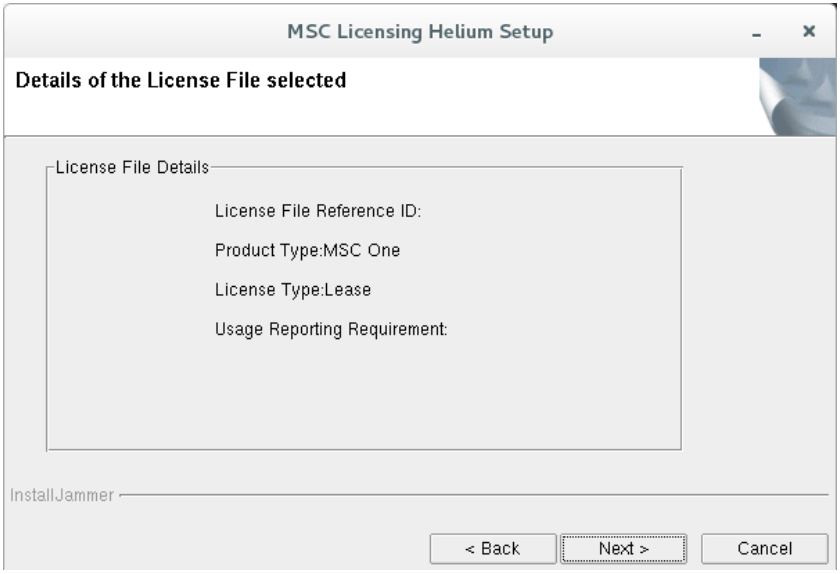
c. The following error message appears if one or more feature entries in the license file has expired:

```
One or more FEATURE entries have maintenance end dates that have expired. These
licenses can be used only by MSC products that were released prior to the
maintenance expiration dare on the FEATURE entry.
```

d. Click **OK** to continue.

10. Click **Next.**

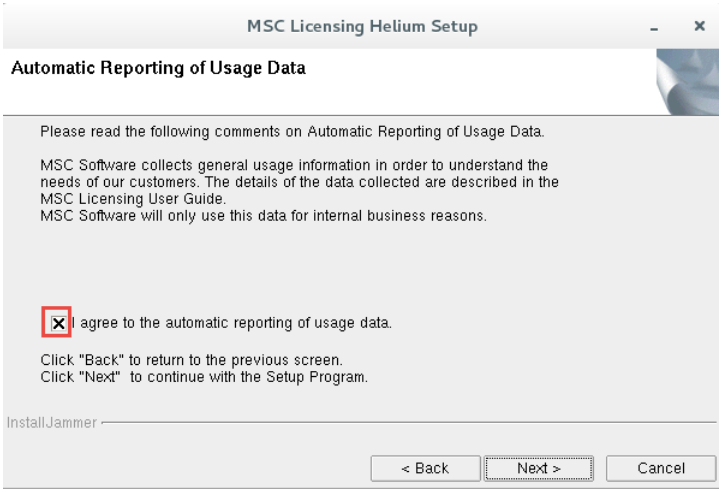
The **Details of the License File** window appears.



Note:	Here, MSC One license is used to demonstrate the installation procedure. For more information about the license types, refer to the *MSC Licensing Helium User's Guide*.

11. Click **Next**.

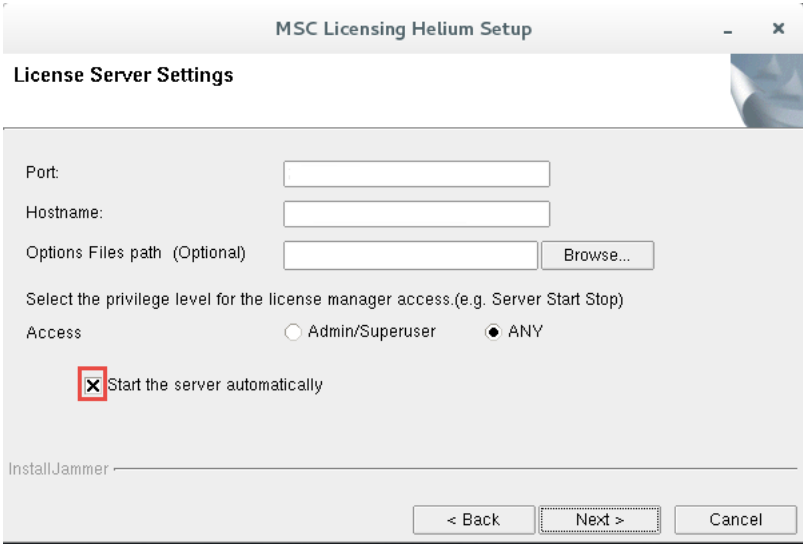
The **Automatic Reporting of Usage Data** appears.



a. Check the box against **I agree to the automatic reporting of usage data** (uncheck the box if you do not want automatic reporting).

12. Click **Next**.

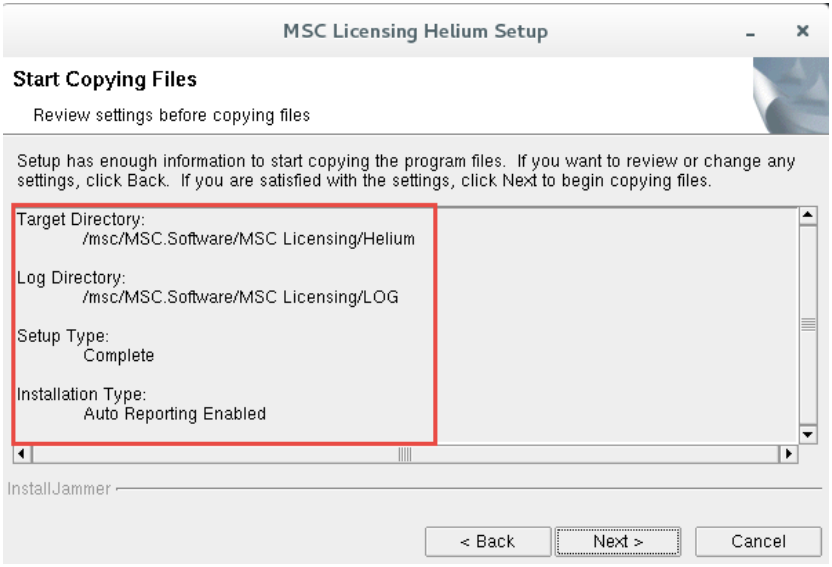
The License Server Setting options appears.



a. Open your `license.dat` file to verify your port number and hostname and enter values accordingly.
b. Click **Browse** and select the **Options File path**.
c. Against Access select **Admin/Superuser** or **ANY** according to your choice to control the access of license.
d. Check **Start the server automatically**.

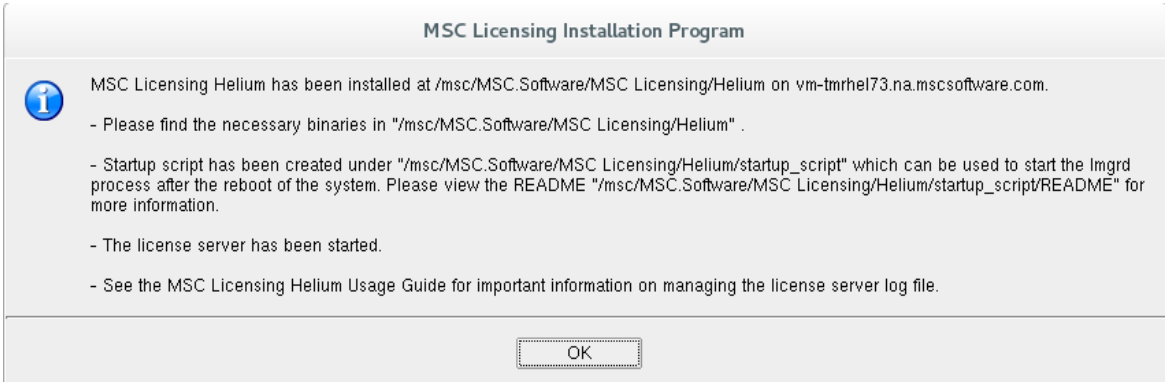
13. Click **Next**.

The Start Copying Files window appears.



Review the settings for one final time.

14. Click **Next**.
15. An **MSC Licensing Installation Program** prompt appears.



16. Click **OK**.

17. Click **Finish** to complete the installation.

Notes:	1. The installer creates a sample script that can be used to start the license server after a system reboot. This script and a README file with important details are written to a directory named `startup_script` under the installation directory you selected. 2. You can start your server (and create a log file) using the below command: `/msc/MSC.Software/MSC Licensing/Helium/lmgrd -c /msc/MSC.Software/MSC Licensing/Helium/license.dat -l /msc/MSC.Software/MSC Licensing/LOG/lmgrd.log`

Install Marc and Mentat

This part will be further divided into two steps as follows:

1. Install Marc
2. Install Mentat

Install Marc

1. Download the file `marc_20XX_linux64_rh7.1_7.3.tar.gz`. Extract it using the commands as follows:

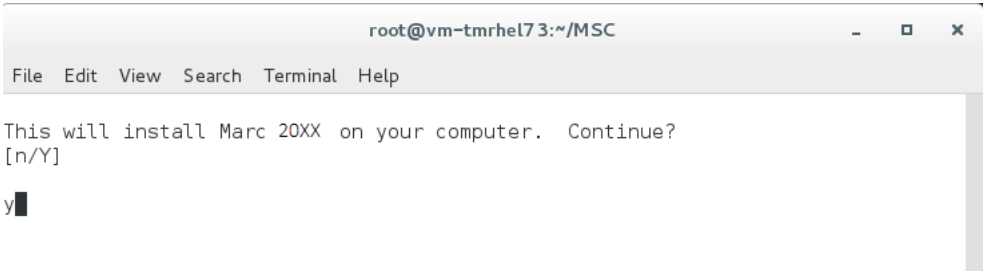
```
[root@vm-tmrhel73 MSC]# tar -xvf marc_20XX_linux64_rh7.1_7.3.tar.gz
install.exe
products/
products/gl27emt0.k10
products/common1.gui
products/al27emt0.k10
products/list
utils/
utils/read_pdf
utils/maintain_usage
```

   The file will be extracted.

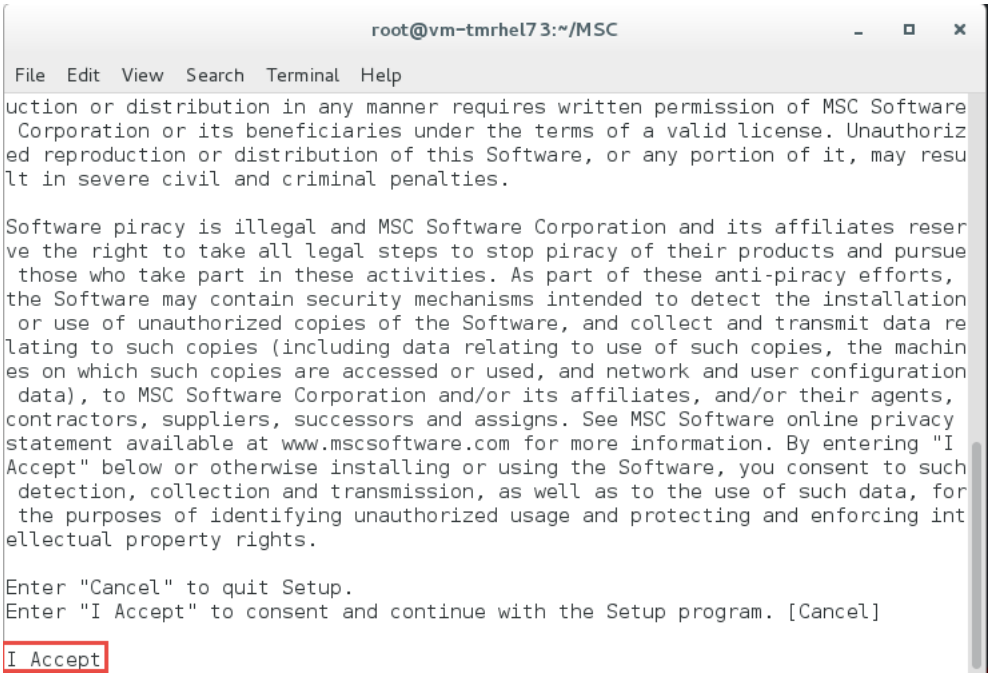
2. Give execution rights to the extracted `install.exe` file and run the installation using the following commands:

```
[root@vm-tmrhel73 MSC]# ls
install.exe                         marc_20XX_linux_doc.tar.gz        utils
license.dat                         msc_licensing_helium_linux64.bin
marc_20XX_linux64_rh7.1_7.3.tar.gz  products
[root@vm-tmrhel73 MSC]# chmod +x install.exe
[root@vm-tmrhel73 MSC]# ./install.exe
```

The terminal asks for permission to start the Marc installation as follows:

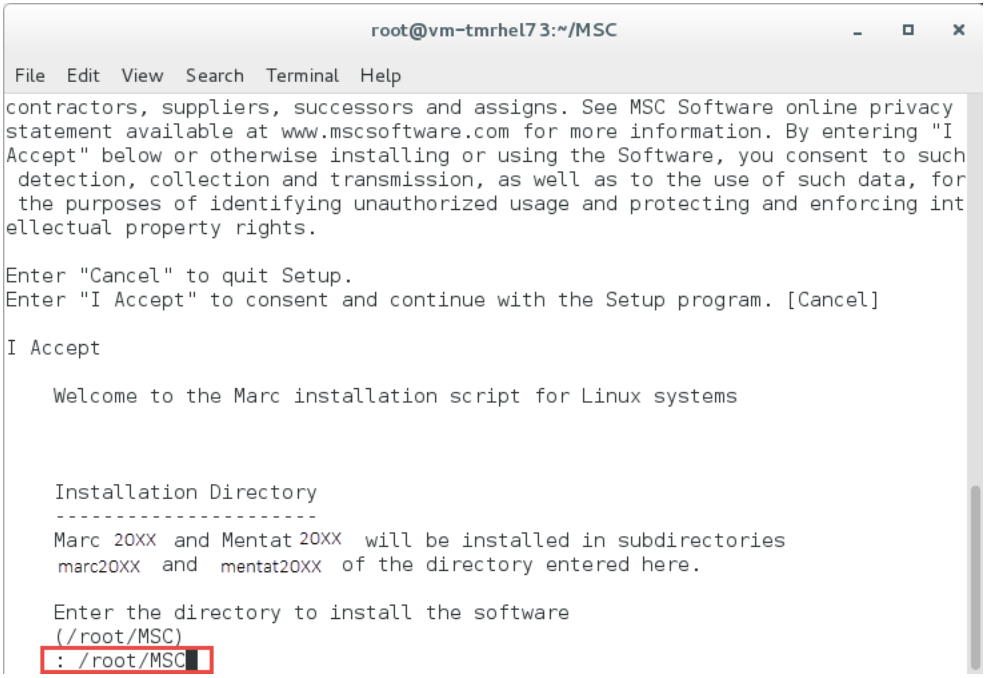


3. Type **Y**, press **Enter**. The copyright information appears on the terminal screen as follows:



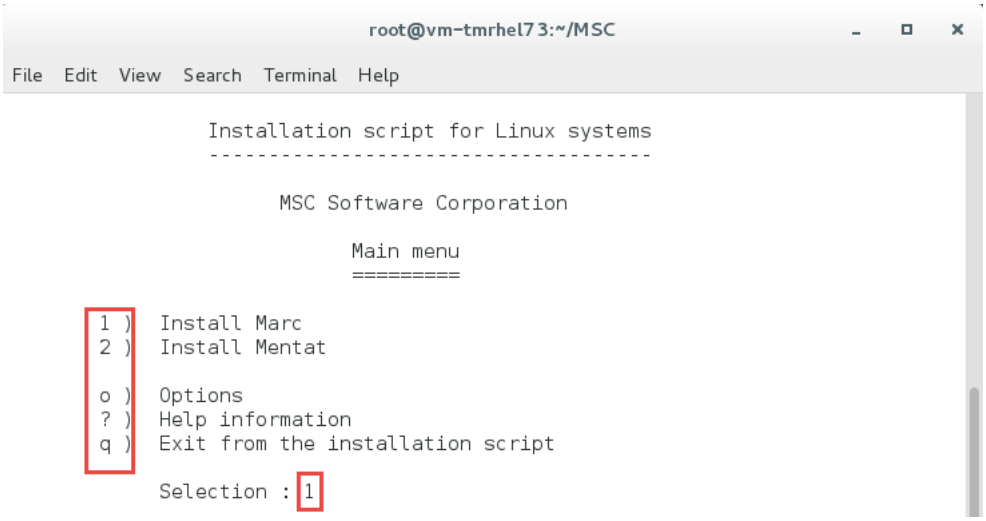
4. Read the information and type **I Accept**.

The following information can be seen on the terminal screen:



5. Type **/root/MSC**, press **Enter**.

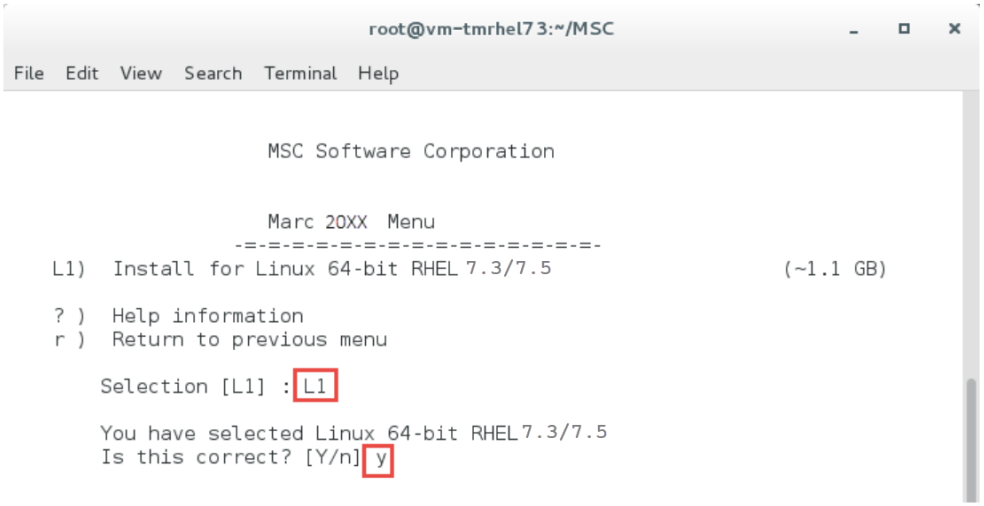
The Linux Installation script appears on the terminal screen as follows:



6. Type **1**, press **Enter**.

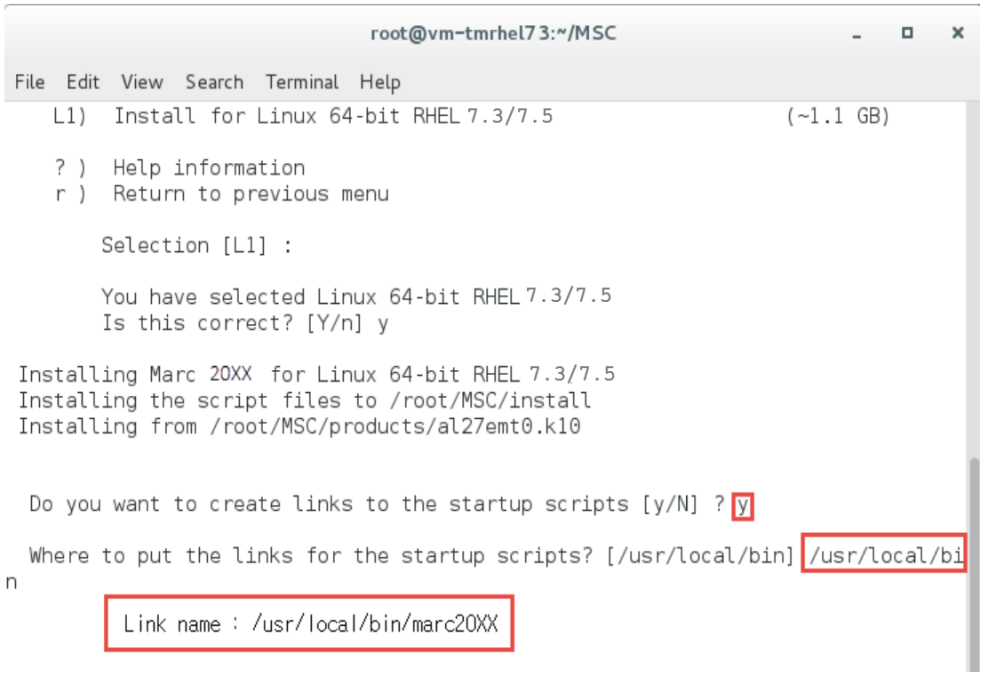
Note:	The installation of Marc should be done before installing Mentat. It will be done

The Marc 20XX Menu appears.



7. Type **L1**, press **Enter** to admit the operating system version.
8. Type **Y**, press **Enter** to give final confirmation.

A prompt to create links to the startup scripts appears.



9. Type **Y**, press **Enter** to create links.
10. Type **/usr/local/bin** to specify the location to put the links for the startup scripts.

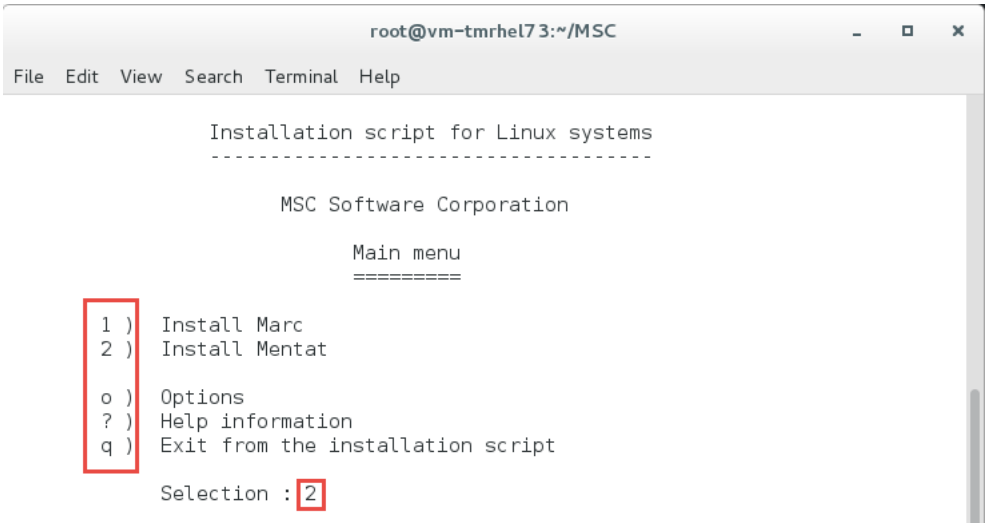
    The link is created and its name appears on the terminal screen.
11. Type **R**, press **Enter** (twice) to return to the main menu.

Install Mentat

In this part you will install Mentat using the same installer file.

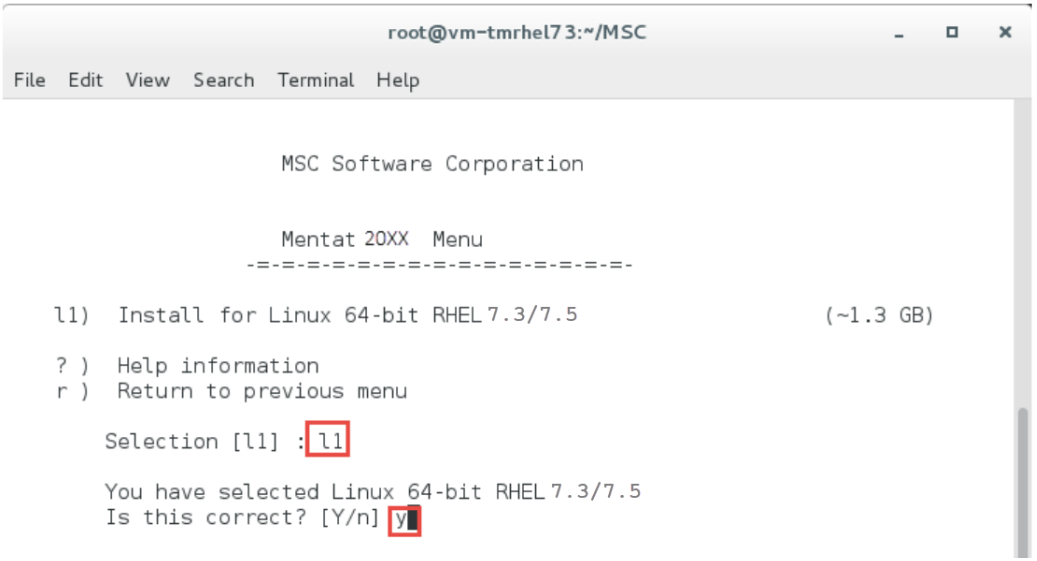
Note:	Mentat installation is in continuation with the last step of Marc installation.

The main menu can be seen on the terminal screen

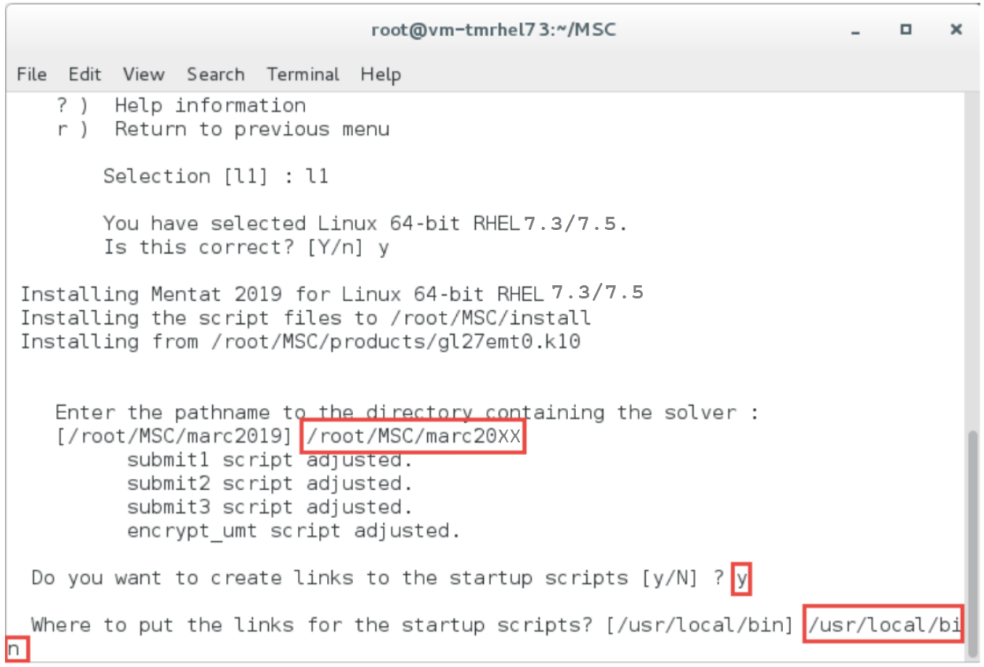


1. Type **2**, press **Enter.**

   The Mentat 20XX Menu appears.

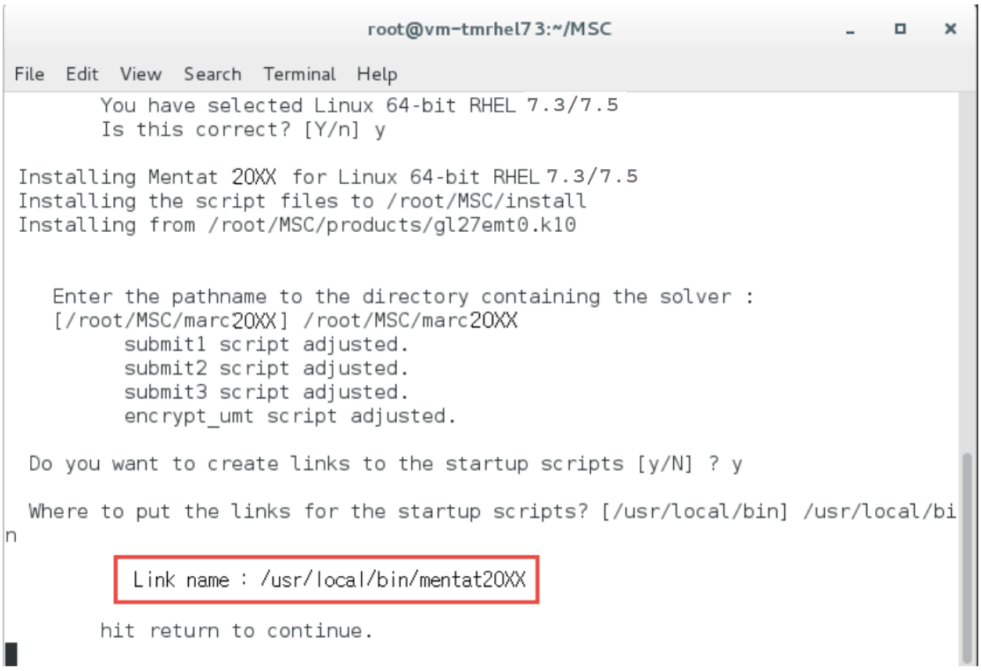


2. Type **l1**, press **Enter** to admit the operating system version.
3. Type **Y**, press **Enter** to give final confirmation.



4. Type **/root/MSC/marc20XX** as pathname to the directory containing the solver. Press **Enter**.
   A prompt to create links to the startup scripts appears.
5. Type **Y**, press **Enter**.
6. Type **/usr/local/bin** to specify the location to put the links for the startup scripts.

The link is created and its name appears on the terminal screen.



7. Type **R**, press **Enter** (twice) to return to the main menu.
8. Type **Q** to exit the Marc and Mentat installation.

Install Documentation

This is the third step of the total installation. In this step you will install the Marc and Mentat documentation which will be accessible through Mentat.

1. Download the file `marc_20XX_linux_doc.tar.gz`. Use the following commands to extract the files and start the documentation installation.

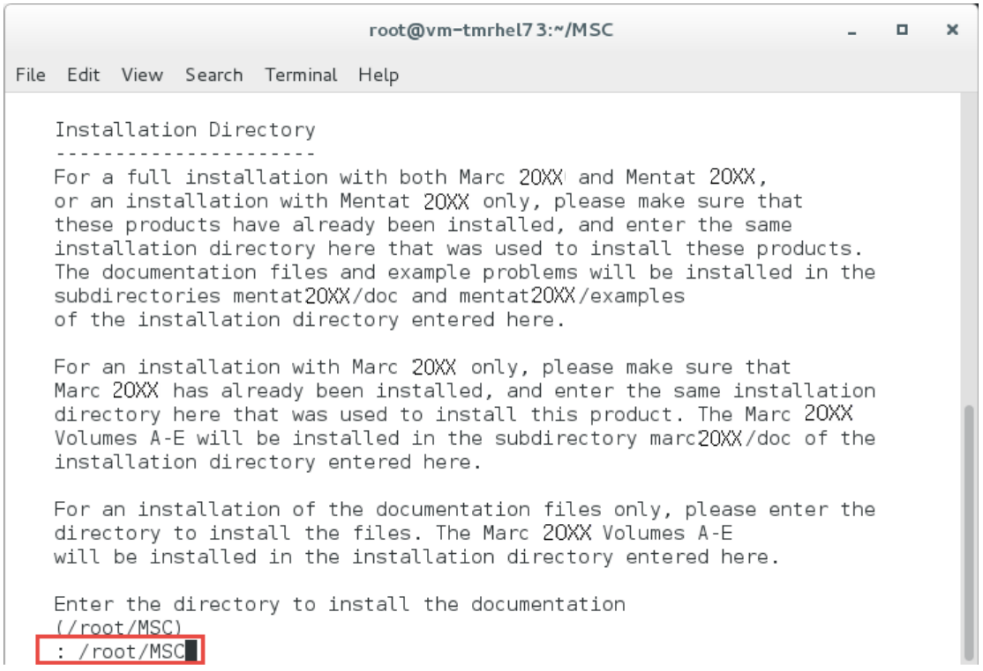
```
[root@vm-tmrhel73 MSC]# tar -xvf marc_20XX_linux_doc.tar.gz
install_doc.exe
products/
products/common3.gui
products/list
products/common2.gui
utils/
utils/maintain_usage
utils/read_pdf
[root@vm-tmrhel73 MSC]# ls
install                              marc_20XX_linux_doc.tar.gz
```

```
install_doc.exe                          mentat20XX
install.exe                              msc_licensing_helium_linux64.bin
license.dat                              products
marc20XX                                 utils
marc_20XX_linux64_rh7.3_7.5.tar.gz
```

2. Give execution rights to the documentation installer(`install_doc.exe`) using the following command:

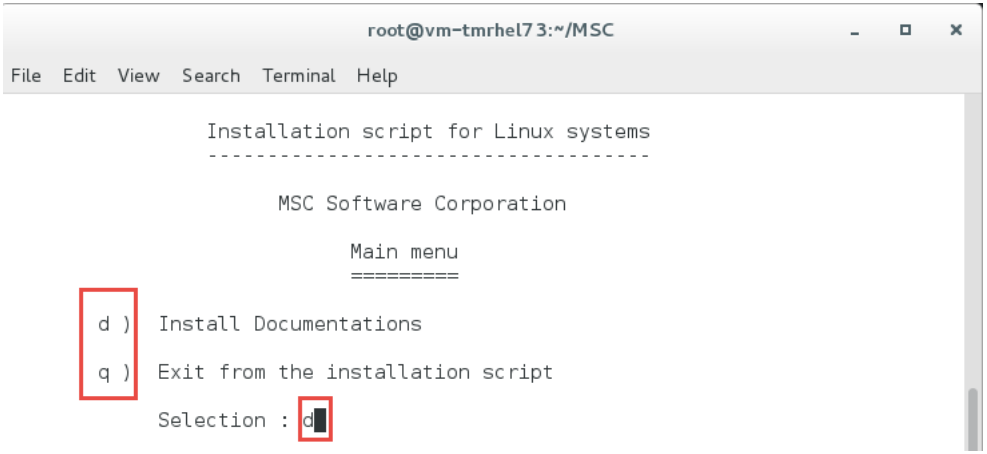
```
[root@vm-tmrhel73 MSC]# chmod +x install_doc.exe
[root@vm-tmrhel73 MSC]# ./install_doc.exe
```

Enter the directory path to install the documentation

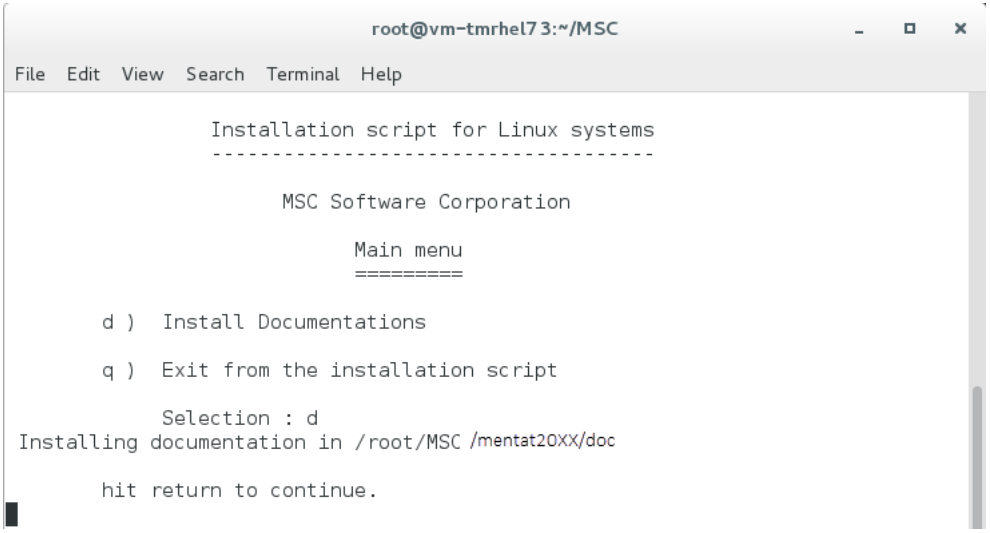


3. Type **/root/MSC**, press **Enter**.

Main menu for documentation appears.



4. Type **D**, press **Enter.**



The documentation is installed. The location is displayed on the terminal screen.

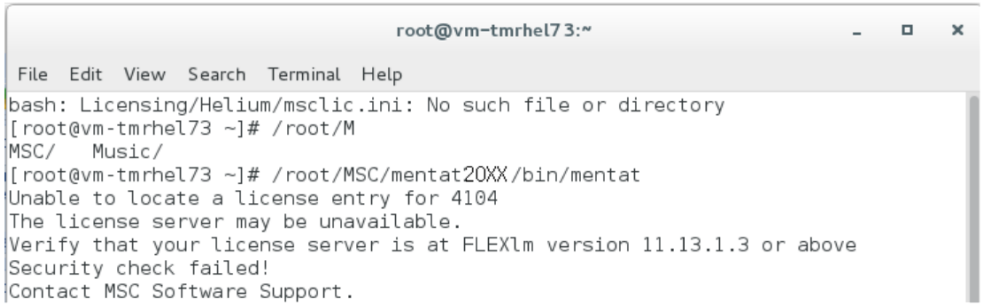
Verify Installation

In this step you will verify that Marc and Mentat are functioning correctly. Also the documentation will be checked whether or not it is linked with Marc and Mentat.

1. Open terminal and enter the path to run Marc and Mentat as follows:

   `/root/MSC/mentat20XX/bin/mentat`

   Ideally the application should start.It might fail for the security check as follows:



If Marc and Mentat is unable to fetch the license, you will have to set some environment variables.

You can use the following command to set required variable in different shells:

- **In Bash:**

  MSC_LICENSE_FILE= 27500@hostname

  export MSC_LICENSE_FILE

- **In CSH:**

  setenv MSC_LICENSE_FILE 27500@hostname

  export MSC_LICENSE_FILE

Notes:	
	■ This demonstrated procedure is done using VM ware installed on a windows machine. Here the license of a pre-installed Marc and Mentat package is fetched into Linux(VM ware). A vice versa process is also possible where a license from Linux can be fetched into windows. ■ Various types of licenses are available for the users to suit their requirements. For details refer to the *MSC Licensing Helium User's Guide*. ■ An elaborated and detailed installation procedure is in Appendix. C: Linux: Marc/Mentat Files, Subdirectories and Installation ■ In case you face any problems related to the licensing, contact: msc_lic.support@mscsoftware.com or, MSC Software support center

Installation Information

Options to the install.exe script	■ The `install.exe` script will accept the following options: • `-a` Turns on automatic installation – installs both Marc and Mentat from the installation media. The **-i** option (described below) is required. • `-i <path>` Specifies the installation path (`<parent>` directory). This option is required when specifying the automatic installation option **-a**. • `-l <file>`Specifies a file for product listing • `-v` Turns on verbose mode. ■ The **automatic** installation will install both Marc and Mentat. To perform an **automatic** installation, run the installation script as follows: `./install.exe -a -i <path>`

License file	■ Two lines of the file `license.dat` need to be modified
	• **SERVER** line which specifies the system hostname
	• **DAEMON** line which specifies the daemon name and the path to the daemon program
	• The content is typically something like:
	`SERVER this_host 0022192361f 1700`
	`DAEMON MSC /your_path/msc`
	■ The string `this_host` should be replaced by the hostname of the machine where the license server is running.
	■ The string `your_path` should be replaced by the full path to the program `msc`. If the same location as in mentioned in this document is used for the security installation it should be:
	`DAEMON MSC /msc/MSC.Software/MSC Licensing/Helium/MSC`

License server	■ The FLEXlm license manager needs to be started on the license server before running Marc and Mentat.
	■ The license server is started with (assuming default installation location)
	`/msc/MSC.Software/MSC Licensing/Helium/lmgrd -c`
	`/msc/MSC.Software/MSC Licensing/Helium/license.dat-l`
	`/msc/MSC.Software/MSC Licensing/Helium/lmgrd.log`

hostid	■ For a nodelocked license it is necessary to obtain a FLEXlm hostid for the machine running Marc and Mentat. This is done by running the command: `/msc/MSC.Software/MSC Licensing/Helium/lmhostid`

Installation Summary

In this section, installation summary is provided for a single license installation for Linux 64-bit. It is assumed that the licensing is already completed.

Steps	Command Information	Description
Step 1: Download the product and start the install script	`mkdir /tmp/marc` cd /tmp/marc gunzip <file>.gz tar xvf <file>.tar ./install.exe	Create a temporary directory and download the product file to that directory. Substitute the proper filename for `<file>`. Use `gunzip` to uncompress the file and extract the files using the `tar` command. Start the `install.exe` script which will be in the current directory. Confirm that Marc 20XX will be installed and enter "I Accept" if you accept the conditions echoed by the script.
Step 2: Install the files on your system		**Welcome to the Marc installation script for Linux systems** Enter a valid pathname to the directory to install the software (`<current directory>`).
	Enter the path	**/opt/marc**
		Marc Installation script for Linux systems MSC Software Corporation Main menu 1)Install Marc 2)Install Mentat o)Options ?)Help information q)Exit from the installation script
	Select option 1	Selection*: 1*
Marc Installation	The only options available will be those from the product file.	MSC Software Corporation Marc 20XX Menu L1)Install for Linux 64-bit RHEL 7.3/7.5 (~1.1 GB) ?)Help information r)Return to previous menu

Steps	Command Information	Description
	Select option L1	`Selection [L1]:` **L1**
		You have selected Linux 64-bit RHEL 7.3/7.5
		Is this correct? [Y/n] **y**
	Make your choice.	`Installing Marc 20XX for Linux 64-bit RHEL 7.3/7.5` Installing the script files to /opt/marc/install Installing from /tmp/marc/products/al27emt0.k10
	You may need to be root to create the links. You can use the default selection by just pressing the enter key.	`Do you want to create links to the startup scripts [y/N]?` Hit return to continue
Mentat Installation	The only options available will be those from the product file.	MSC Software Corporation Mentat 20XX Menu 11)Install for Linux 64-bit RHEL 7.3/7.5 (~1.3 GB) ?)Help information r)Return to previous menu
	Select option l1	`Selection [11]:` ***l1*** `You have selected Linux 64-bit RHEL 7.3/7.5` `Is this correct?` [Y/n]: **y** Installing Mentat 20XX for Linux 64-bit RHEL 7.3/7.5 Installing the script files to `/opt/marc/install` Installing from `/tmp/marc/products/gl27emt0.k10`

	Enter the path to the `marc20XX` directory.
	You can use the default selection by just pressing the enter key.
		`Enter the pathname to the directory containing the solver:`
		`[/opt/marc/marc20XX]`
		Press return to continue
		submit1 script adjusted
		submit2 script adjusted
		submit3 script adjusted

	Make your choice	`Do you want to create links to the startup scripts [y/N]` ? **n** Press return to continue

Steps	Command Information	Description
Step 3:	Return to main menu	r)Return to previous menu
Step 4: **Install Security**		Run the installation executable for security. Install and run `lmutil` to obtain a `hostid`. Send this information to MSC Software and put the obtained license file in `/msc/MSC.Software/MSC Licensing/Helium/license.dat`
Step 5: **Installing the documentation**	`mkdir /tmp/documentation` cd /tmp/documentation gunzip <file>.gz tar xvf <file>.tar ./install_doc.exe Enter the path	Enter the directory used to install Marc and Mentat: `/opt/marc`

Managing FLEXlm

FLEXlm is the network based licensing product from Flexera Software used in MSC products.

FLEXlm on-line documentation is available from Flexera, see the URL:

http://support.flexerasoftware.com/doc/

Note:	The above URL is not an MSC Software Corporation site, and MSC has no control over the site's content. MSC cannot guarantee the accuracy of the information on this site and will not be liable for any misleading or incorrect information obtained from this site.

FLEXlm License File

The license file, `license.dat`, should normally be placed in the `/msc/MSC.Software/MSC Licensing/Helium` directory once you receive your licenses from MSC Software. Everyone should have read permission to the file. The file can be located elsewhere as long as the environment variable MSC_LICENSE_FILE points to it.

Two lines of the license file contains installation specific information:

```
SERVER this_host 0022192361f 27500
DAEMON MSC /your_path/msc
```

Here `this_host` should be set to the hostname of the machine where the license server is running. `your_path` should be set to the full path to the msc program, by default located in `/msc/MSC.Software/MSC Licensing/Helium/flexlm/<platform>`

where <platform> is the name of the platform of the license server.

FLEXlm License Manager

For the 20XX version, the FLEXlm License Manager must be at version 11.13 or higher.

The license manager must be running on the license server machine.

The license manager is started with (assuming default installation location)

```
/msc/MSC.Software/MSC Licensing/Helium/lmgrd -c
/msc/MSC.Software/MSC Licensing/Helium/license.dat
-l /msc/MSC.Software/MSC Licensing/Helium/lmgrd.log
```

Environment Variables

The environment variable **MSC_LICENSE_FILE** is used to specify the full path to the `license.dat` file, and is a colon separated list of file pathnames. An example setting of `MSC_LICENSE_FILE` is:

```
MSC_LICENSE_FILE=/msc/MSC.Software/MSC Licensing/Helium/license.dat:27500@license
servername
```

The environment variable **MSC_AUTHQUE** is used to specify the number of minutes a program will wait for a license after it has been requested. If the license request cannot immediately be honored, the request will be queued and the variable defines how long it should wait in this queue. The variable must be set in your environment before Marc and Mentat are started. The default value is 5 minutes.

It may be set as follows in a Bourne shell:

```
MSC_AUTHQUE=20
export MSC_AUTHQUE
```

In a C shell, it may be set as:

```
setenv MSC_AUTHQUE 20
```

Machine hostid

For a nodelocked license it is necessary to obtain a FLEXlm hostid for the machine running Marc and Mentat. This is done by running the command

```
/msc/MSC.Software/MSC Licensing/Helium/lmutil lmhostid
```

GPU Support on Linux

Hardware and Software Requirements

The list of supported solver types with the GPU option can be found in *Marc Volume A: Theory and User Information* Chapter 11: Solution Procedures for Nonlinear Systems in Table 11-3 and Table 11-4.

An NVIDIA driver is necessary if the GPGPU capability is to be used. The driver needs to be compatible with the one used in the Marc build; see Marc Volume A: Theory and User Information, Chapter 12: GPU Support in for the minimum driver version supported.

The GPU capability in Marc is developed using CUDA toolkit 11.0 Update 1 with supported compute capability ranging from 3.5 to 8.0. NVIDIA Tesla cards with higher compute capability values are recommended. Note that NVIDIA Quadro cards can also be used.

Deviceinfo Utility

A stand-alone utility `deviceinfo` is provided in the tools directory to check available GPU cards on the machine. This utility provides information about the GPU cards and the installed NVIDIA drivers toolkit version. Typical output from the deviceinfo utility is shown below:

```
-sh-4.2$ deviceinfo

[ ** Marc GPGPU device diagnosis utility ** ]

 Installed CUDA driver version is 11.0

 Total 1 devices are detected on the machine

   Device id 0 : "Tesla V100-PCIE-32GB"

                 Compute capability                : 7.0
                 Number of Multiprocessors         : 80
                 Number of Cores per Multiprocessor : 64
                 Total Number of Cores             : 5120
                 Total device memory (in MB)       : 32510


   The supported compute capability for Marc GPGPU feature is from 3.5 to 8.0

   Below is the list of supported devices

       Device id 0               : "Tesla V100-PCIE-32GB"
-sh-4.2$
```

Figure 7-1 Deviceinfo Output

Mentat Support

Running the job with GPU may also be done from Mentat. The GPU menu is only displayed for supported solver types Multifrontal Sparse and Iterative Sparse. Note that solver multi-threading (which refers to CPU threads) is not supported when the GPU option is selected.

The GPU option in Mentat is shown in Figure 7-2. When the GPU option is chosen, there are two options available for GPU card selection:

a. **Automatic**: At run-time, Marc determines the GPU cards that are available and assesses the optimal card to be used.

b. **User:** The user can specify the GPU cards that need to be used for the job. The available device IDs can be determined by running the deviceinfo utility.

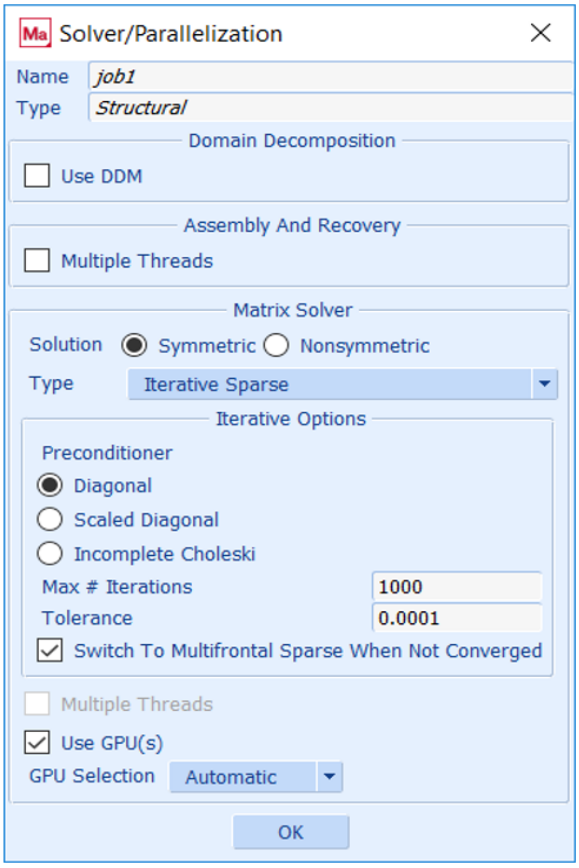


Figure 7-2 GPU Option in Mentat

Troubleshooting

Exit 63

Cause	The Marc version being used is not supported with the GPU version of the Multifrontal solver.
User Information	Non-GPU version of the Multi-frontal solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

Exit 64

Cause	Error occurs while attempting to use the GPU card specified by the user.
User Information	This error occurs while trying to use the GPU card specified by the user.
User Action	Use the deviceinfo utility to check for supported and available GPU cards on machine.

Exit 65

Cause	The Marc version being used is not supported with the GPU version of the Iterative solver.
User Information	Non-GPU version of the Iterative solver causes this error if used with the GPU option.
User Action	Contact MSC Software representative to obtain supported version.

missing shared object file "libcuda.so"

Cause	This error occurs when required NVIDIA drivers are not available while running Marc with the GPU option.
User Information	To use GPU feature in Marc, necessary NVIDIA drivers need to be installed on machine.
User Action	Use deviceinfo utility to check if supported drivers are installed on machine. If not, download the required drivers from NVIDIA.

8 Linux: Running and Using Marc and Mentat

Running Marc

This section describes the Marc usage on Linux based machines. The Marc programs are mainly controlled by a shell script called **run_marc** which is stored in the `marc20XX` subdirectory `tools`. If you have used the option to creating a link during the installation, this link is also known system wide as **marc20XX**. It is designed to handle practically all possible options.

The shell script will submit a job and must be executed in the directory where all relevant input and output files concerning the job are available. To use the shell script, each Marc job should have a unique name qualifier and all Marc output files connected to that job will use this same qualifier.

Marc input files should always be named `job_name.dat`, whereby the prefix `job_name` is the name qualifier which you are free to choose. The suffix `.dat` is obligatory.

To actually submit a Marc job, the following command should be used. The single input line is split over multiple lines for clarity:

run_marc	**-jid**	**job name (required as minimum)**
	`-rid`	restart file name
	`-pid`	post file name
	`-sid`	substructure file name
	`-prog`	name of a previously saved user executable
	`-user`	user subroutine name
	`-save`	save user executable
	`-back`	run in background
	`-ver`	verification flag
	`-vf`	view factor file name
	`-def`	defaults file name
	`-nprocd`	number of domains
	`-nprocds`	number of domains in Single Input File mode
	`-nts`	number of threads for parallel matrix solver (same as `-nthread_solver` *or* `-nthread` *or* `-nt`)
	`-nte`	number of threads for element assembly and stress recovery (same as `-nthread_elem`)
	`-nsolver`	number of matrix solver tasks
	`-mpi`	mpi version
	`-dir`	directory where job I/O takes place
	`-sdir`	directory where the scratch files are located
	`-host`	host file name; used for distributed execution in a network
	`-ci`	copy input files to remote machines in a network
	`-cr`	copy post files back from remote machines in a network
	`-ml`	memory limit in Megabytes
	`-gpuid`	GPU card ID or auto keyword

Table 8-1 describes the meaning of these input options and Table 8-2 gives examples. The default actions are shown in **bold** font.

Table 8-1 run_marc Input Options

Keyword	Options	Description
-jid (-j)	job_name	Input file (job) name identification.
-prog (-pr)	progname	Run saved executable progname.marc from a previous job (see -user and -save).
-user (-u)	user_name	User subroutine user_name.f is used to generate a new executable program called user_name.marc (see -save and -prog).
-save (-sa)	no yes	Do not save the new executable program user_name.marc. Save the executable program user_name.marc for a future time (see -prog and -user).
-rid -(r)	restart_name	Identification of previous job that created **RESTART** file.
-pid (-pi)	post_name	Identification of previous job that created the post file.
-sid (-si)	substructure	Identify the job that contains the solution to the external nodes of the superelement.
-back (-b)	**yes** no	Run Marc in the background. Run Marc in the foreground.
-ver (-v)	**yes** no	Ask for confirmation of these input options before starting the job. Start the job immediately.
-def (-de)	default_file	File name containing user defined default data.
-nprocd (-np)	number	Number of domains for parallel processing.
-nprocds(-nps)	number	Number of domains for parallel processing using a Single Input File.
-nts (-nthread)	number	Number of threads per parallel matrix solver (8, 9, and 11).
-nte	number	Number of threads used for parallel matrix assembly and stress recovery.
-nsolver	number	Number of processes to use for the MUMPS parallel matrix solver (solver 12).
-dir	directory_name	Pathname to directory where the job I/O should take place. Defaults to current directory.
-sdir	directory_name	Directory where scratch files are placed. Defaults to -dir.
-host (-ho)	hostfile	Specify the name of the host file for running over a network (default is execution on one machine only in which case this option is not needed).
-ci	**yes** no	Copy input files automatically to remote hosts for a network run, if necessary.

Table 8-1 run_marc Input Options (continued)

Keyword	Options	Description
-cr	**yes** no	Copy post files automatically from remote hosts used for a network run, if necessary.
-vf	viewfactor_name	Name of file containing viewfactors for radiation from previous analysis or from Mentat using either the Monte Carlo or Hemicube method.
-ml	memlimit	Memory limit for deciding if the solver should go out-of-core. Specified in Mbyte. Defaults to the physical amount of memory on the machine.
-gpuid	<id:id:id> or auto	GPU card number or auto for automatic selection by program. For parallel (DDM) jobs, multiple card IDs can be specified if available.

Table 8-2 Examples of Running Marc Jobs

Examples of running Marc jobs	Description:
run_marc -jid e2x1	Runs the job **e2x1** in the background, the input file `e2x1.dat` resides in the current working directory.
run_marc -jid e2x14 -user u2x14 -save yes	Runs the job **e2x14** in the background, using the user subroutine `u2x14.f` and the input file `e2x14.dat`. An executable program named `u2x14.marc` will be saved after completion of the job.
run_marc -jid e2x14a -prog u2x14	Runs the job **e2x14a** using the executable produced by job **e2x14.**
run_marc -jid e3x2a -ver no -back no	Runs the job *e3x2a* in the foreground. The job will run immediately without verifying interactively.
run_marc -jid e3x2b -rid e3x2a	Performs a restart job using the results of the previous job **e3x2a.**
run_marc -jid e2x1 -nprocd 2	Runs a two processor job on a single parallel machine.
run_marc -jid e2x1 -nprocd 2 -host hostfile	Runs a two-processor job over a network. The hosts are specified in the file `hostfile` (refer to the Linux: Marc Parallel Network for runs on a network of machines.

Running Mentat

This section describes the Mentat usage on Linux based machines. Mentat is started by a shell script called **mentat** which is stored in the `mentat20XX/bin` directory. If you used the option to create a link during the installation, this link is known system wide as **mentat20XX**.

You do not need to start the shell script from a specific directory.

Mentat creates the default files in your current working directory; that is, where you are located at the time of starting Mentat.

The shell-script *mentat* contains a number of arguments which are passed on to Mentat. Table 8-3 gives the meaning of these input options (more details can be obtained using **-help** option). You are free to alter these commands to suit your preference.

Table 8-3 Mentat Input Options

Keyword	Option	Description
-mode	preset1/preset2	This will launch the application in the selected preset mode. For preset1, which is the default, the traditional theme, the traditional mouse button scheme and complete picking will be active. For preset2, the dark theme, the auto-dynamic mouse button scheme and partial picking will be active.
-ar	area ratio	This option sets the initial window size to the given fraction of the available space on the desktop.
-xr	horizontal ratio	This option sets the initial width of the window to the given fraction of the available width on the desktop [default: 0.92]
-yr	vertical ratio	This option sets the initial height of the window to the given fraction of the available height on the desktop [default: 0.92].
-aspr	aspect ratio	This option sets the aspect ratio (width over height) of the window [default: 1.6].
-maximize		Starts up Marc Mentat maximized.
-minimize		Starts up Marc Mentat minimized (iconified).
-bg		Will run Mentat in the background. Note that the DISPLAY environment variable must point to a valid display.
-bp	`$(DIR)/bin/`	Directory path name where the external Mentat programs and shell scripts are located.
-compile	binary_menu_filename	This is used to compile ASCII menu files into a binary menu file. The -mf option would be used to specify the new binary file. To recompile the default binary menu file `main.msb`, compile as: bin\mentat -compile menus\win64\main.msb Substitute for <plat> the platform name `linux64`.
-dr	True/False	Enable/disable direct rendering. Enabling this option will make OpenGL bypass the underlying window system and render directly from hardware to the screen, if this is supported by the system. The default is **True**.
-fn	font	Default font type.
-fngr	font	Font used by the graphics windows (overrules the font selected via the `-fn` option).

Table 8-3 Mentat Input Options (continued)

Keyword	Option	Description
-fnme	font	Font used by the menus (overrules the font selected via the `-fn` option).
-gradient		Switches on a gradient background when displaying the various windows (Model, Table, History Plot, etc.).

-hd	*name filename*	Define a document viewer with the given *name* for viewing a document that consists of a set of HTML files. The *filename* must be the name of an XML file that defines the contents of the document. A menu item to open the viewer can be created by adding an `<html_doc>` element that references the viewer (via the "name" attribute) to the appropriate `<menu>` in menus/menubar.xml.
		The XML file must have the following contents:
		`<?xml version="1.0" encoding="UTF-8"?>`
		`<html_doc>`
		`  <proc_root env="DEMO_DIR"/>`
		`  <title_page href="index.html"/>`
		`  <contents>`
		`  <chapter href="file1.html"/>`
		`  <chapter href="file2.html"/>`
		`  </contents>`
		`</html_doc>`
		The "href" attributes of the `<title_page>` and `<chapter>` elements specify, respectively, the path to the HTML file of the title page and the HTML files of the chapters of the document. The file paths must be given relative to the directory in which the XML file is located.
		For example, the Mentat startup script defines the document viewer for the User's Guide as follows:
		`-hd ug help/ug/ug.xml`
		and menubar.xml contains the following element to open the viewer:
		`<html_doc name="ug"`
		`title="User's Guide"/>`

-help		Print a list of all of the options.
-hide_dialog		Hide the dialog window at startup.
-hide_dynamic_menu		Hide the dynamic window at startup.
-hide_main_menu		Hide the main window at startup.
-hp	`$(DIR)/help/`	Directory path name where the help files are located.
-lf	*filename*	Specify the Mentat log file name.

Table 8-3 Mentat Input Options (continued)

Keyword	Option	Description
-license_release	number	Enable Mentat to release its licenses if the program is inactive for the specified amount of time. The number is in minutes and must be greater than one.
-mf	`main.ms`	The name of the startup menu file.
-ml	`$(DIR)/material/`	Directory path name where the material files are located.
-mp	`$(DIR)/menus/`	Directory path name where the menu files are located.
-multiundo	off/on	Enables the user to either set the one-level UNDO option (off), with data saved in memory, or the multi-level UNDO option (on), with data saved on disk [default: off].
-undo_levels	number	Number of UNDO levels when the multi-level UNDO option is active. The minimum is 1, the maximum is 50 [default: 10].
-nosolidmodeling		Utilize the Mentat_Parasolid_CAD which allows geometric models to be imported but does not allow solid geometry editing.
-num_session_files	number	Number of copies of Mentat session files (i.e. files with the extension .log and .proc) which will be left in the directory where Mentat is started. The minimum is 1, the maximum is 100000 [default: 3]. The session files are named mentat.*.log and mentat.*.proc. This option is omitted if the option **-lf** is used to point to a non-default Mentat log file.
-path	directory_name	Provides a directory in which Mentat searches when opening an existing input file. Multiple directories can be specified as follows: *-path directory_1 -path directory_2* etc.
-pr	*filename*	Any additional set-up commands you wish to add. Store these in a procedure file containing the Mentat commands.
-ra		Reads all of the ASCII Menu files.
-rf	*filename*	Record the Mentat commands in the procedure file *filename*.
-szgr	width height	Set the size (width and height in pixels) of the graphics area.
-ti	title	Append *title* to the name of the window.
-unicode		Allows the use of unicode characters in load case titles, job titles, annotations and user-defined names of results file variables.

Making Changes to the Marc Programs

Modifying the MPI Setting

The MPI version used by default is defined in the table below. It also lists an alternative MPI version to use for the respective platform. The default MPI version to use can be set in the `run_marc_defaults` file (see next section). Enter a line

```
MARC_MPI alternative_mpi
```

for setting a new default. Use the exact word as given in the Alterative MPI column. The MPI version can also be set with the command line option `-mpi` to `run_marc`. The command line option overrides the value set in `run_marc_defaults`.

Mentat Interfaces

Mentat External Programs

Mentat supports a number of CAD interfaces: IGES, Patran, Ideas, VDAFS, etc. These interfaces are accessed using external programs called from within Mentat. The interface programs are stored in the *mentat20XX* subdirectory *bin*. These programs read the data files in their native format and translate the contents into a Mentat model file. This file is subsequently read by Mentat. The external programs are called from within Mentat by means of the FILE submenu.

Jobs

The subdirectory `bin` contains shell script files to start a Marc FEM job using the following shell scripts:

```
submit1, submit2, submit3,
```

These shell scripts are called by means of the buttons in the JOBS menu.

You can alter these files to suit your environment; for example, set up one of the *submit* scripts so that it starts a Marc job on a different machine on your network.

Plotter Interface

Because of the many variations in plotting environments, we have created plotting interfaces in the form of shell scripts that operate from within Mentat. Currently, Mentat recognizes the following plotting formats:

- PostScript
- Xdump (translated in either PostScript or HPGL format)

This section describes a template shell script for each of the formats mentioned above. They are located in the `bin` directory and are named as follows:

```
psgray1, psgray2, psgray3
pscolor1, pscolor2, pscolor3
xdump1, xdump2, xdump3
```

PostScript

The PostScript function is activated by pressing either the **Gray** or **Color Print** button from the **UTILS** menu on the POSTSCRIPT panel. The program captures the graphics portion of the screen into a file and sends this file to a PostScript printer using the `psgray` or `pscolor` shell scripts located in the `bin` directory. In the example listed below, the file is sent to a computer called 'mars' on the network. The `lpr` command with the `-Psupt` argument sends the file to a PostScript printer known to the spooler as **supt**. After the file is sent, it is removed from disk automatically.

```
#!/bin/csh
 rsh mars lpr -Psupt < $1
rm -f $1
```

The `argument $1` is the filename handed to the shell by Mentat. If there is more than one printer on-line, the `psgray` and `pscolor` shell scripts can be used to address these other printers.

Xdump

The `xwd` command, widely available on many platforms, dumps an image of an X window into a specially formatted dump file. This file can then be read by various other X utilities for redisplay, printing, editing, formatting, and archiving. Its complementary `xpr` command takes the window dump file as input and formats its output for a particular device, such as a PostScript or other color printer. Below is an example of a shell script that uses `xwd`, and, in conjunction with `xpr`, sends the information to printer `pjetxl`. See the `man` pages on your system for more details.

```
#!/bin/csh
xwd | xpr -device pjetxl -scale 2 | /etc/aprint -Abatphone2 -L25
```

Edit

The **edit_window** shell script is used to control the editor associated with the `EDIT` commands. It is possible to change the type of editor, for example, from **vi** to **emacs** or change the type of windowing environment.

System Shell

The **system_window** shell script is used to control the type of window opened with the `system_shell` command. It is possible to change the type of window.

Parallel Render

The **marc_render** shell script can be modified such that the photorealistic rendering is performed across multiple CPUs. The parameter **nbands** is used to specify the number of CPUs.

MPEG Playback

The **mpeg_window** shell script is used to control the program opened with the `play_mpeg` command.

9 Linux: Marc Parallel Network

Hardware and Software Requirements

This section provides the general description about the hardware and software requirements and definitions.

If your cluster requires a password to perform **ssh** or **rsh** between nodes, you may need to enter your password every time you run a parallel job. You can disable the password requirement in the use of **ssh** or **rsh**. Please consult your system administrator for password requirement in the use of **ssh** and **rsh**.

Although no specific hardware requirements exist for Marc to run in network mode, it is preferable to have fast network connections between the machines. It is recommended that the network should have a speed of at least 1 GBit per second. If only two machines are to be used, a hub or a cross-over cable can be used to connect them. If more than two machines are to be used, a switch is preferable. TCP/IP is used for communications.

The list of supported capabilities in parallel can be found in "Supported and Unsupported Features" in Chapter 12 of *Marc Volume A, Theory and User Information*.

The Supported platforms are listed in the section "Supported Platforms" in Chapter 6 of this guide.

Compatibility

All machines used in a network must use the same Marc version.

Definitions

The definitions of some of the terms are as follows:

Table 9-1 Definitions

Sr. No	Term	Definition
1	Root machine	The machine on which Marc job is started.
2	Remote Machine	Any machine other than the root machine which is part of a distributed Marc run on the network.
3	Shared installation	Marc is installed in an NFS shared directory on one machine only. The other machines can access the Marc executable since the directory is shared.
4	Distributed installation	Marc is installed on all machines. Each machine accesses its own Marc executable.

Table 9-1 Definitions

Sr. No	Term	Definition
5	Distributed execution	Marc is run on multiple machines which are connected with a network. Each machine loads the Marc executable either from a shared or a local directory and then executes the executable.
6	Shared I/O	Marc reads and writes data in a NFS shared directory. Each Marc executable running on the network reads/writes to the same directory.
7	Distributed I/O	Marc reads and writes data in a directory located on each machine. The user must make the input available in each directory and collect the results files after the analysis.
8	NFS	Network File System.

Network Configuration

Marc only needs to be installed on the root machine where the installation directory is shared via NFS (shared installation). Please note that if Intel MPI is used the installation directory must be available with the same path for every machine. Marc can also be installed on the remote machines which then use their own executable (distributed installation), but the name of the path must be the same.

- The root machine is the one on which the Marc job is started, typically from within Mentat.
- The remote machines can be located anywhere as long as they are connected to the network.

The working directory on each machine can be a shared directory on any machine on the network (shared I/O) or it can be a local directory on the hard disk of each machine in the analysis (distributed I/O). The User Notes describes how to specify what working directory to use.

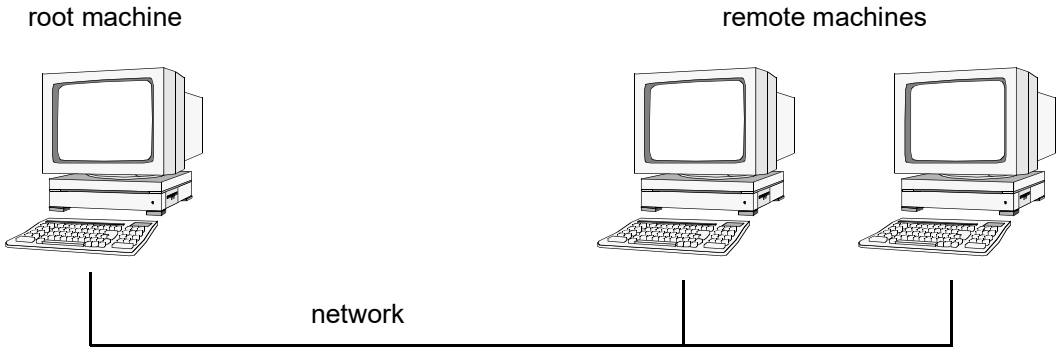


Figure 9-1 Network Configuration

Installation Notes

This section describes the procedure required to install and set up a network version of Marc. For general information on Marc installation, see Linux: Marc and Mentat Installation.

Install Marc on the root machine. Marc only needs to be installed on the root machine. However, if Marc is to be used on the remote machines as well, it can also be installed there. There is nothing special that needs to be done related to the installation itself for the network version. The installation directory must be available with the same path on every machine, regardless of if the installation is local or if it uses a shared directory.

In order to run parallel jobs on machines connected over the network, it has to be set up properly. If any of the remote hosts does not have Marc installed, the installation directory on the root machine needs to be shared using NFS or some other mechanism so that the Marc version is available from the remote machines. Users need to be able to connect between the machines using `rlogin or ssh` without having to provide a password. It is possible to define if rsh/rcp or ssh/rsh or similar command is to be used. This is done by modifying the variables RCP and RSH in the include file for the platform used.

Assume that, there are two machines with hostnames host1 and host2 to be used in a parallel job over the network. Marc has been installed on host1 and the job is to be started from this machine. A hypothetical naming convention is used for shared directories where a directory name on any machine starts with `/nfs/hostname`, where **hostname** is the name of the machine on which the directory is located.

1. Test the installation for single processor execution.
2. Change directory to the `test_ddm` subdirectory of the Marc installation directory on host1.
3. Then do

   ```
   cd exmpl2/exmpl2_1
   marc -j cyl2 -b no -v no
   ```

   Marc should exit in about three minutes if it is a successful run. Here we assume that the command **marc** points to the current Marc version.
4. Test the Marc installation for multi-processor execution. Do the following:

   ```
   cd ../exmpl2_2
   ```
5. Edit the file `hostfile` in this directory by replacing `workdir` with

   `/nfs/host1/marcinstall/test_ddm/examp2/exmpl2_2`

   and `installdir` with

   `/nfs/host1/marcinstall.`
6. For Intel MPI delete installdir (so only use three columns in the host file).
7. The host names and directory names should be replaced with the names on the current system.
8. Finally, type:

   ```
   marc -j cyl2 -b no -v no -nprocd 2 -host hostfile
   ```

   Marc should exit in about two minutes if it is a successful, parallel run on host1 and host2 using one processor on each.

User Notes

This section provides miscellaneous information about executing a parallel job over a network and use of user subroutines.

This section assumes that the network version of Marc has been successfully installed on at least one of two machines that are to be used in a distributed analysis and that the appropriate Marc licenses are in order. Assume that host1 is the host name of the machine on which Mentat is running and from which the job is to be started (the root machine). The host name of the other machine (the remote machine) is host2.

Running a network job

First, make sure that the two machines are properly connected. From host1, access host2 with

```
rlogin host2
```

or

```
ssh host2
```

If a password needs to be provided for the remote login, this has to be taken care of. If the `rlogin` or `ssh` is not possible without providing a password, a network run will not be possible. If you face any problems, refer to Chapter 10: Linux Troubleshooting.

Note: The current version does not support the IPV6 protocol.

In order to perform an analysis over a network, you have to create a specific file called `host file`. This file defines which machines are to be used, how many processes are to run on each, what working directory should be used, and where the Marc executable can be found on each machine. The host file can be selected and edited in Mentat and the Marc job started as usual from within Mentat (see the example below). If Marc is run from the command line, it is done as for a serial run using an additional command line option. For example:

```
marc -v no -b no -jid test -nprocd 2 -host hostfile1
```

will run the two-processor job `test.dat` using the specification in the file `hostfile1`. No specific name or extension is used for the host file except that the name `jobid.host` (in this example `test.host`) must be avoided since it is used internally by Marc.

Specification of the Host File

The host file has the following general format:

```
host1 n1
host2 n2 workdir2 installdir2
host3 n3 workdir3 installdir3
```

- Each line must start at column 1 (no initial blanks). Blank lines and lines beginning with a # (number symbol) are ignored. The fourth column with `installdir` should not be used with Intel MPI.
- The first entry is the host name of a machine to be used in the analysis. The root machine must be listed first and each machine must only occur once.

- The second entry specifies the number of processes to run on the machine specified in the first entry. The sum of the number of processes given in the host file must equal the number of domains used. In a five-domain job, it is required that `n1+n2+n3=5`.

The third entry specifies the working directory to use on this host. This is where the I/O for this host takes place. The Marc input files for this machine must be in this directory and the results files for this machine are created in this directory.

The fourth entry specifies where the Marc installation directory that this host should use is located. This entry can be omitted if the name of the Marc installation directory is the same on all machines (which could be a shared directory on host1 with the same name from host2 and host3). This is a requirement for Intel MPI so this entry should here be omitted.

The directories in the third and fourth entries will be used from the respective host. To check the correctness of the host file specification, log in to the respective machine and list the directories as specified in the host file. For the host file given above, do:

```
ssh host2
ls workdir2
ls installdir2
```

The second line should show the working directory to use on host2 and the third line the installation directory which will be used by host2.

The different domains of the Marc job are associated with the different machines as follows. Suppose a five-domain job `test` is run using a host file defined as

```
host1 2
host2 1 workdir2 installdir2
host3 2 workdir3 installdir3
```

with appropriate definitions of the third and fourth entries

There will be six Marc input files associated with this job: `test.dat, 1test.dat, ..., 5test.dat`. Domains 1 and 2 will be associated with host1, domain3 with host2 and domains 4 and 5 with host3.

Shared I/O

Suppose a job is to be run on host1 and host2. A shared directory on host1 is to be used for I/O and from host2 its name is `/nfs/host1/marc/workdir` (assuming a hypothetical naming convention for shared directories which starts with `/nfs/hostname`). The installation directory is assumed to have the same name on both machines. The host file for a two-processor job would simply be

```
host1 1
host2 1 /nfs/host1/marc/workdir
```

To verify the workdir given, do `rlogin host2 ; ls /nfs/host1/marc/workdir`. The directory seen should be the same one as the working directory on host1.

Distributed I/O

If you want to have the I/O to be local on host2, specify the host file as

```
host1 1
host2 1 /usr/people/marcuser
```

The I/O on host2 will now take place in the directory `/usr/people/marcuser` on the hard disk of host2. For this case, the Marc input files are transferred to `/usr/people/marcuser` on host2 before the job is started, and the results files are transferred back after the analysis for postprocessing. This transfer of files is done by Marc automatically.

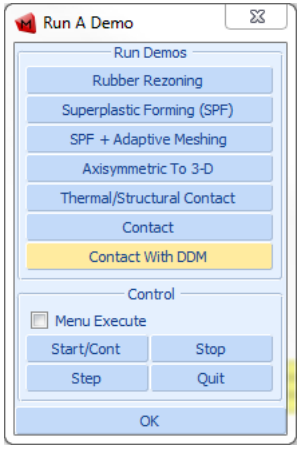
It is also possible to use only two entries in the host file. This requires that both the working directory and the installation directory have the same names on all machines.

Example

The definitions for a network run with Mentat is demonstrated with a simple example. We assume the simplest case where both the working directory and installation directories are shared.

1. Enter the menu **Help → Run a Demo...**

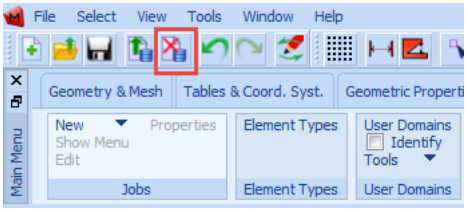
   A **Run A Demo** dialog appears.



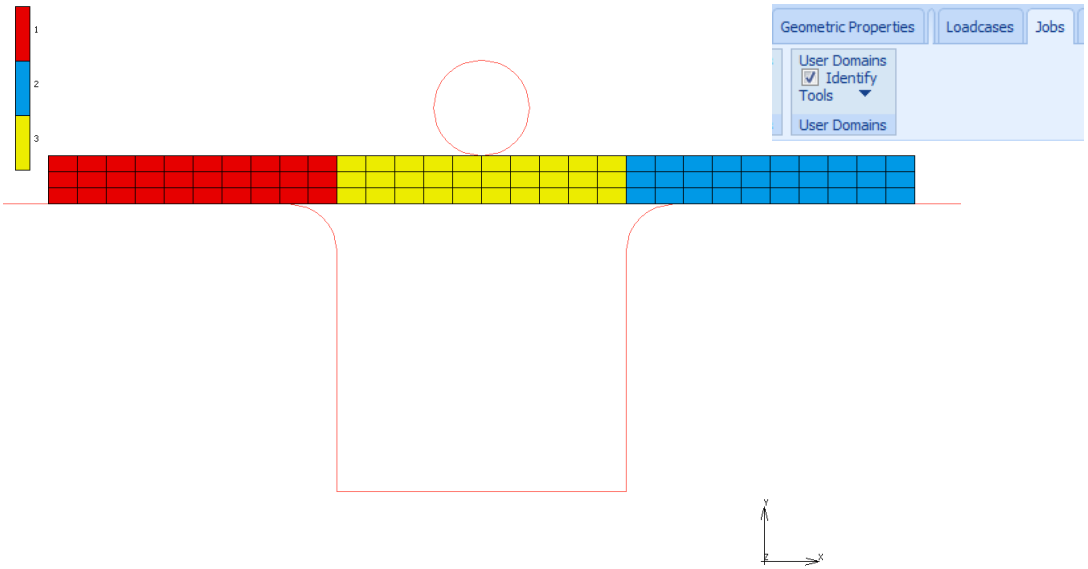
2. Select the example **Contact With DDM**.

Note:	The input creation for the demo problem automatically starts as you select any of the problems. It continues till the results are seen.To stop/pause the automatic inputs creation, you need to click on the stop button.Once you stop the procedure use the Step button to see a step-by-step input creation.

3. After the process is completed, click **Close result file** button.

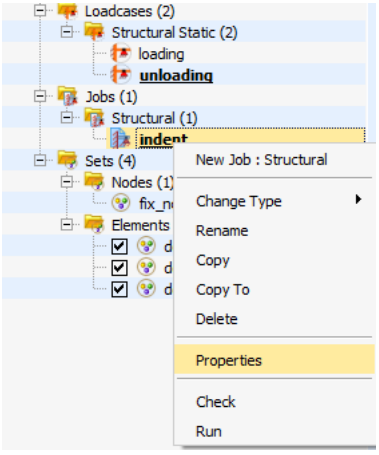


4. Select **Jobs** tab and check **Identify** to confirm your two domains as shown in the image as follows:

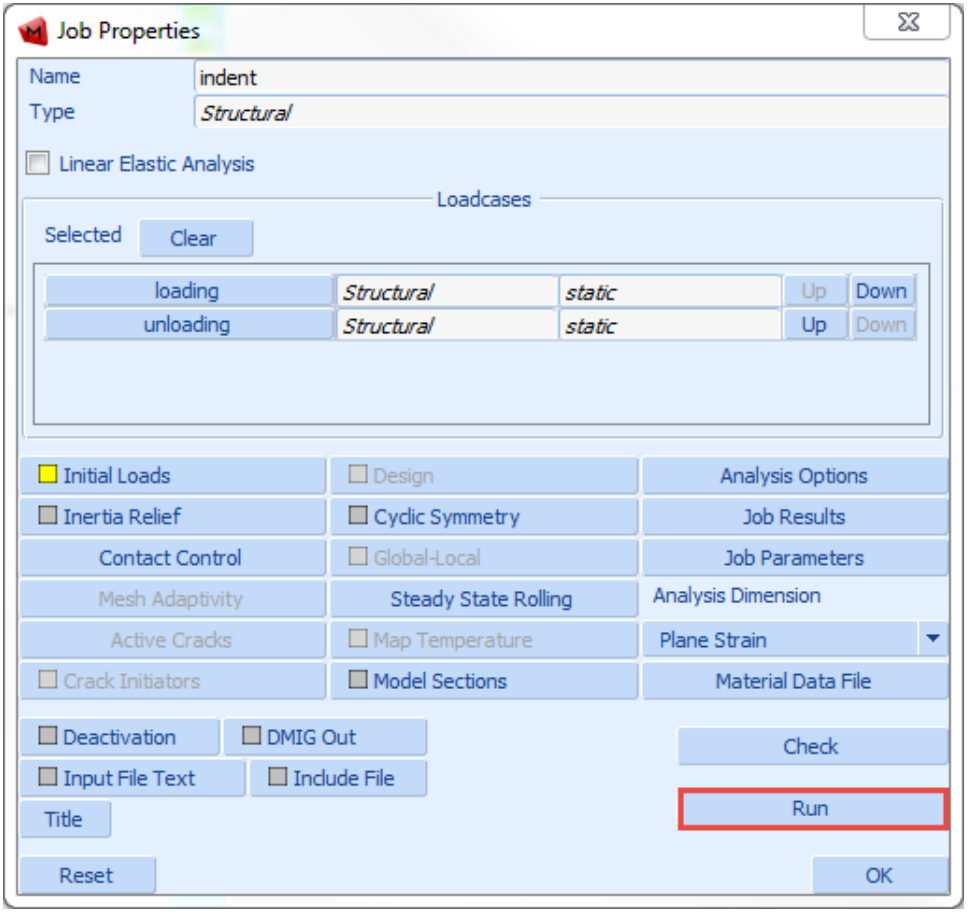


The complete model workflow can be seen in the **Model Navigator**.

5. Right click on **Indent** and select **Properties** from the drop down list.

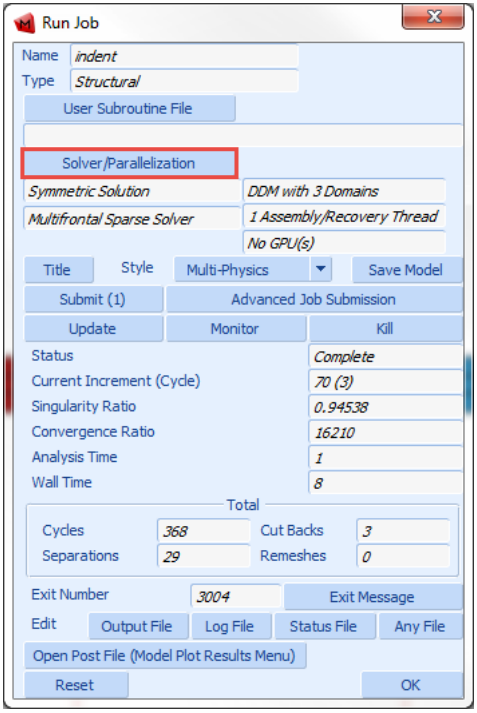


The **Job Properties** dialog appears.



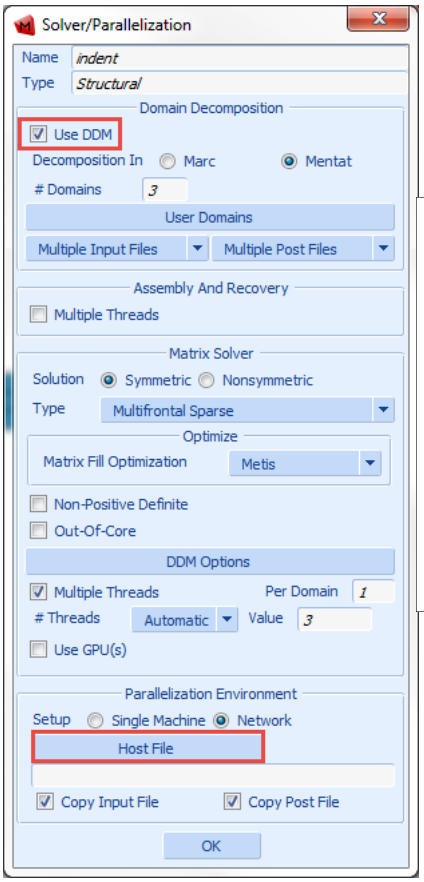
6. Click **Run**.

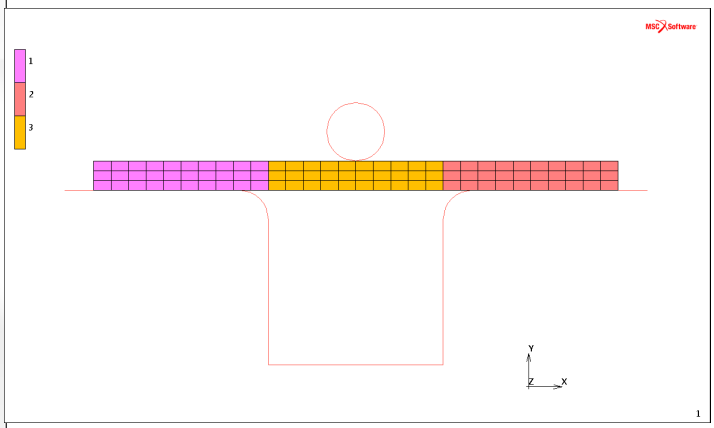
The **Run Job** dialog appears.



7. Click **Solver/Parallelization** button.

The **Solver/Parallelization** dialog appears.





a. Ensure the **Use DDM** option is selected. In this case three domains are used.

b. Select Network option from the submenu.

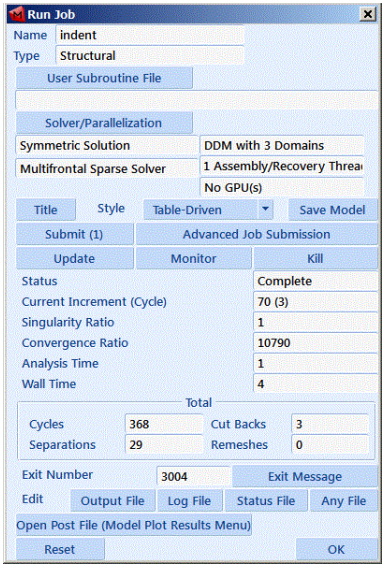
   An additional option to select the host file appears just below Network.

c. Click on the **Host File** button and select the file called `hostfile` from the install directory and modify it to contain:

```
host1 1
host2 1  workdir installdir
```

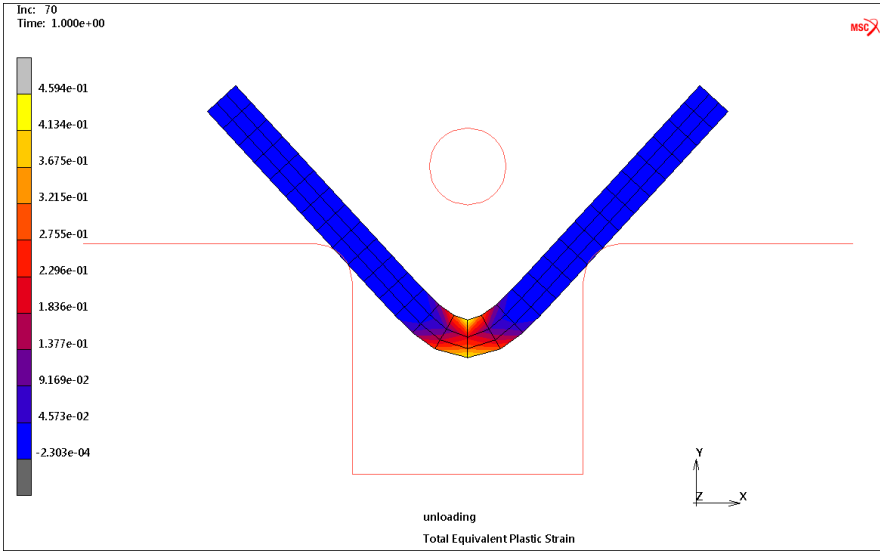
d. Set the working directory according to the share names on the current system. The host file places one domain on hostname1 and one domain on hostname2.

8. Run Marc from within Mentat using the **Submit 1** button. Upon completion of the analysis, the **Run Job** dialog box appears:



9. Check your results by clicking the **Open Post File** (**Results Menu**) button.

   A plot of the Total Equivalent Plastic Strain in the final deformed configuration is shown as follows:



Marc created a post file associated with each domain as well as a root post file associated with the job id. For the previous model, `1model1_job1.t16 and 2model1_job1.t16` are the processor files, while `model1_job1.t16` is the root file.

If the model is very large, it can be convenient to view only a portion of the model by selecting any one of the processor post files, such as `2model1_job1.t16`. This file contains only data associated with domain 2 as selected in the domain decomposition menu. As described in the earlier steps, this file was created by host2.

Shared vs. Distributed I/O

For jobs with very large post or restart files, it is usually more efficient to use distributed I/O. With distributed I/O, the input files and the post files are located on the host's local disks. Marc by default automatically transfers the input files and the post files to and from the remote host if needed. It is possible to suppress this transferring with two buttons in the Network settings in the **JOBS** menu in Mentat.

To run a job using distributed I/O, specify a local directory in the host file:

```
host1 2
host2 1 /usr/people/marcuser
```

Jobs with User Subroutine

User subroutines are fully supported in the network version.

The Fortran file with the subroutine is located in the working directory on the root machine. Marc automatically creates the executable and makes it available on all remote hosts. There is no need to modify the host file if it is correct for a job without a user subroutine.

If the working directory is shared for all remote hosts, the user subroutine is compiled on the root machine and the executable is available in the shared working directory.

If a remote host is using a local working directory, the executable will be automatically copied over to the remote machine using remote copy (`rcp or scp`). Marc automatically knows if a directory is shared or local.

Troubleshooting

If you face any problems in launching Marc Mentat, check the following:

1. The network connection between the hosts is working by using the command `ping host`.
2. A remote login using the command `rlogin` can be done between the hosts without providing a password. If not, contact your system administrator or check the man pages for `rlogin` and look for `.rhosts`.
3. The host names used in the hostfile are correct. It should be the same as the output from the command `hostname` on the respective host.
4. The working and installation directories on the host file are correct. Log onto the remote host, change directory to these directories to verify the host file content. The installation directory given should (among others) contain the executable in the bin directory.

5. The input files for each host are available in the respective working directory. An error message is printed out from Marc if they are not.

Error messages:

1. The error message `semget failed...` at job start-up means that the communication environment is not clean. This can be checked with the Linux command `ipcs`. If entries belonging to specific users except `root` show up, they may need to be removed. Run the script `tools/mpiclean` located in the Marc installation directory.

Note:	This will kill all parallel jobs currently running under the current user. Only entries belonging to the current user are deleted.

10 Linux Troubleshooting

Cannot create

Cause	You have no write permission in the `<parent>` directory.
User Actions	Change with **chmod**

Security failed

Cause

The possible causes for this are:

- Marc or Mentat was unable to obtain a license from the FLEXlm licensing software.
- The FLEXlm license manager is not running. See Appendix C for information on how to run the license manager.
- You are attempting to run on a machine that according to the license file you are not allowed to use.
- Your license period has expired. Check the date on your machine.

For counted licenses, currently running too many Marc jobs.

User Information

In this case, Marc or Mentat will exit.

User Actions

- If you have just modified the `license.dat` file, the `lmgrd` and `MSC` daemons may not have been restarted. Run the **lmreread** utility as follows:

```
lmreread -c <parent>/msc/MSC.Software/MSC
Licensing/Helium/license.dat
```

- If you get the FLEXlm error:

Invalid (inconsistent) license key (-8,130:2) No such file
or directory

it may be implying that the hostid value specified on the SERVER line are inconsistent with the hostids. Check the values and restart the license manager.

- If you get the following FLEXlm error when using a server license:

```
Cannot connect to license server (-15,12:146)
```

it means, the license manager (`lmgrd`) may not be running on the license server, or the **USE_SERVER** line in your client side `license.dat` file is incorrect. Also make sure that the TCP/IP port numbers used on the SERVER line are the same on both the client and the server.

- If you get the FLEXlm error:

```
No such feature exists (-5,147)
```

and your license is limited to certain systems, you may be trying to run on a system that is not licensed for use. Check that the **lmhostid** of the system you are trying to use and that on your `marc20XX` (**FEATURE MARC**) license is the same.

Link failed in Marc

Cause
- Your user subroutine causes compiler errors.
- You have no Fortran compiler.
- Fortran libraries not available.

User Actions
Check the variable **syslibs** in the file **include_linux64** in the `marc installation` subdirectory `tools`. It references special system libraries in `/usr/lib` which may not exist on your system.

Testing Marc installation failure

User Actions
If you are using `install.exe` to test Marc installation and the **Test and maintain installation** does not respond, remove `exec tcsh` from your `.cshrc` and restart installation testing.

Mentat cannot open display

Cause
- Mentat does not have X server access to your display device.
- You are using a terminal other than the default screen belonging to the machine, without setting the X-window output device.

User Actions
- Ensure Mentat has X server access to your display device. The following command allows Mentat to run on a remote screen:

  xhost +

  This command must be issued while logged onto the computer that owns the remote screen.
- If you are using a terminal other than the default screen belonging to the machine, you may have to set the X-window output device:

  C-shell: `setenv DISPLAY your_terminal_name:0.0`
- Bourne shell: `DISPLAY=your_terminal_name:0.0 export DISPLAY`

Mentat runs, then aborts.

Cause	The model you are working on becomes very large.
User Information	Mentat requires a considerable amount of memory to store the model. Ensure that a minimum of 8 GB core memory is available in your machine.
User Actions	■ Increase the core memory upto 8 GB ■ If the multi-level UNDO option is off, you can save memory by switching **UNDO** off in the **Tools → Program Settings...** menu.

You are prompted for a password when running parallel jobs

User Information

- In Marc 20XX, on Linux the default MPI version is IntelMPI. This MPI version requires that the ssh command has been set up such that it does not prompt for a password. Here is a description on how to set this up.
- Make sure there is a directory called .ssh in the home directory (note that in this home directory only you must have write access):
  - `cd $HOME`
  - `ls .ssh`
- If it does not exist, create it:
  - `mkdir .ssh`
  - `chmod 700 .ssh`
  - `cd .ssh`
- Execute the command
  - **ssh-keygen -t rsa -f id_rsa -P ''**
- It is -P followed by two single quotes. This will create two files: id_rsa and id_rsa.pub. Copy id_rsa into a file called identity:
  - cp id_rsa identity
  - Append `id_rsa.pub` to a file called `authorized_keys`
  - `cat id_rsa.pub >> authorized_keys`
  - chmod 600 authorized_keys
- The directory `$HOME/.ssh` should now contain the four files `id_rsa`, `id_rsa.pub`, `identity` and `authorized_keys` (and possibly more files).

- If all went well it should now be possible to do:

  ssh thishost

  - with thishost replaced by the hostname of the current machine, without getting a prompt for the password. After this it should be possible to run a parallel job without having to provide a password.
  - In order to be able to connect to other Linux machines without being prompted for a password (for example for running parallel network jobs), use the following steps:

- Copy the file `id_rsa.pub` that was created above to the other machine. Make sure it has a directory `$HOME/.ssh`. Append the file `id_rsa.pub` to the files `$HOME/.ssh/authorized_keys` and `$HOME/.ssh/authorized_keys2` and give them the appropriate permission:

```
cat id_rsa.pub >> $HOME/.ssh/authorized_keys
cat id_rsa.pub >> $HOME/.ssh/authorized_keys2
chmod 600 $HOME/.ssh/authorized_keys $HOME/.ssh/authorized_keys2
chmod 700 $HOME/.ssh
```

- The first time you login with ssh to the second system you will get a warning and asked if you want to continue. Type `yes` to accept and the remote host will be added to the file `$HOME/.ssh/known_hosts` and the next time you will not be prompted.

SECTION 3: APPENDICES

A Microsoft Windows: Marc Subdirectories and Installation

Marc Subdirectories

The Marc version you have received contains a full set of subdirectories listed below. You can save disk space by removing the example subsets that you do not need.

Table A-1 Contents of the Marc installation

Basic set	Contents: required as minimum
bin	executable Marc programs
tools	batch scripts to run and maintain the Marc programs
AF_flowmat	material data for database
xdr_lib	machine dependent libraries
lib_shared	machine dependent libraries
Extended set	**Contents: only for use with user subroutines**
lib	binary libraries with the compiled Marc routines
common	insert files containing Marc common blocks
user	templates for all available Marc user subroutines
Examples	**Contents: example files**
demo	input files and user subroutines for the *Marc Volume E: Demonstration Problems*
demo_ddm	input files and user subroutines for the single parallel machine as well as the network parallel version of Marc
demo_table	input files and user subroutines for *Marc Volume E: Demonstration Problems* based upon Table driven input format
test_ddm	one, two, and four processor test examples for installation testing of the single parallel machine as well as the network parallel version
Utilities	**Contents**
pldump	source routines for the post-file conversion programs pldump, pldump13 and pldump2000
intelmpi	MPI libraries for network parallel version

FLEXlm 11.13 VS Helium(11.16.3.0) License Manager

This section contains information and directory paths according to Helium license manager. The changes in the default paths in comparison with FLEXlm 11.13 are as in the following table:

11.13	Helium (11.16.3.0)
C:\MSC.Software\MSC.Licensing\11.13\msc	C:\Program Files\MSC.Software\MSC Licensing\Helium\msc
C:\MSC.Software\MSC.Licensing\11.13\lmtools	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmtools
C:\MSC.Software\MSC.Licensing\11.13\installs	C:\Program Files\MSC.Software\MSC Licensing\Helium\installs
C:\MSC.Software\MSC.Licensing\11.13\lmgrd	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmgrd
C:\MSC.Software\MSC.Licensing\11.13\LOG	C:\MSC.Software\MSC Licensing\Helium\LOG
C:\MSC.Software\MSC.Licensing\11.13\lmutil	C:\Program Files\MSC.Software\MSC Licensing\Helium\lmutil
C:\MSC.Software\MSC.Licensing\11.13\msclic.ini	C:\Program Files\MSC.Software\MSC Licensing\Helium\msclic.ini

Installation Procedure

Steps	Command Information	Description
Step 1: **Security**		Download FLEXlm from the Solutions Download Center. Select the product *MSC Licensing FLEXlm*, version Helium. Run the installation executable and follow the installation instructions. Default installation directory is `C:\MSC.Software\MSC.Licensing\Helium` Obtain a license file from MSC Software. If a nodelocked license is to be used, then obtain FLEXlm hostid with `C:\MSC.Software\MSC.licensing\Helium\lmtools` Select the **System Settings** tab and click on **Save HOSTID Info to a File**. Copy the license file to the proper location. The standard location is `C:\MSC.Software\MSC.Licensing\Helium\licenses\license.dat` Set the environment variable **MSC_LICENSE_FILE** to point to the license file. The license server must be running in order to run Marc and Mentat. Previous versions of Marc and Mentat will run with the latest FLEXlm version, so it is safe to upgrade previous installations of FLEXlm.
Step 2: **Install Marc and Mentat**	Start the installation from the product media obtained from the Solutions Download Center. Choose Destination Location: *Destination Folder*: **C:\Program Files\MSC.Software\Marc\20XX.0.0**	The first screen will prompt you to set the location where you want the products installed. This path is the `<parent>` directory. It defaults to C:\Program Files\MSC.Software\Marc\20XX.0.0 The directories `marc20XX` and `mentat20XX` will be created in the directory that you specify. The default destination folder for Mentat is `C:\Program Files\MSC.Software\Marc\20XX.0.0`

Steps	Command Information	Description
Setup type	You will then be presented with which product options to install. ☐ **Complete** ☐ **Solver** ☐ Modeler	Select the **Complete** option to install both Marc and Mentat. Select the **Solver** option to only install Marc, or select the **Modeler** option to only install Mentat.
Program folder	Select Program Folder: *Program Folder*: **MSC.Software**	Select the folder that you wish to place the shortcut to the Mentat startup script. The default program folder name is **MSC.Software**.
Specify license file	License: **C:\MSC.Software\MSC.Licensing\Helium\licenses\license.dat**	When you perform the **Marc** installation, it will ask for the location of a valid Marc license file. Specify the location of your license file.
	☐ **I would like to view the README file.**	The last screen is the **Setup Complete** screen. It will present you with an option to view the readme.txt file. Click on the **Finish** button to leave Setup. Then click on the X button in the lower left hand corner of the main installation menu window or press the escape button.
Step 3: **Start the License Manager**	Start the FLEXlm license manager. You may also want to enable the license manager to run as a service and to start the server at power-up by enabling these options in the Config Services section.	Select **Start Programs** → **MSC.Software** → **MSC License Helium** → **FLEXlm Configuration Utility** to configure FLEXlm. Select the **Config Services** tab and verify that the settings are correct; i.e., the **License File** is set correctly. Then start the license manager from the **Start/Stop/Reread** tab by pressing the **Start Server** button.

Steps	Command Information	Description

Step 4:
checking
	Run Mentat by either selecting the Mentat item in the program folder that you chose, or run it from the MS-DOS Command Prompt.
	First check that the variable MSC_LICENSE_FILE is set properly. Use an MS-DOS Command Pompt window and type:
	`set msc_license_file`
	If it is not correct, change it using the System applet in the Control Panel.
	Then run the program using:
	`cd C:\Program Files\MSC.Software\marc cd 20XX.0.0\mentat20XX bin\mentat`
	Run a demo problem by selecting the menu buttons:
	Help>Run A Demo>Thermal/Structural Contact
		Check the installation by running Mentat.
		There are three methods to run Mentat:
		1. Use the Mentat icon which is created on the desktop
		2. The link in the MSC.Software\Marc 20XX.0.0 program group
		3. Run it from an MS-DOS Command Prompt window.
		To run from the command prompt, do the following:
		1. Go to the `<parent>\mentat20XX` directory.
		2. Enter the command bin\mentat.
		You must first check that the environment variable **MSC_LICENSE_FILE** is set properly to the full pathname of your valid license file. If it is not, the product will fail due to licensing.
		Check the Mentat program by running one of the standard Marc demonstration examples as proof of a successful installation. From the **Help** menu, select **Run A Demo...**, and then select the **Thermal/Structural Contact** demo. It will run for 50 increments.

	To check that user subroutines are working by running one of the standard user subroutine demo problems: `cd C:\Program Files\MSC.Software\marc cd 20XX.0.0\marc20XX\demo\tools\ run_marc -j e2x4 -u u2x4`	If you have a **Fortran compiler,** run a user subroutine example using: `run_marc -j e2x4 -user u2x4` Marc should give a **Marc Exit number 3004.**
Step 5: **for Network Version only**		If you will be using the Marc Parallel Network feature, you need to setup MPI. This service is optionally installed on the system when you installed **Marc**, however you will need to install it on the remote machines. Follow the Microsoft Windows: Marc Parallel Network for Microsoft Windows (Parts 1 and 2) for important information on installing and running jobs with the network version.

Steps	Command Information	Description
Step 6: **Installing the documentation**	Start the installation from the product media obtained from the Solutions Download Center. Choose Destination Location: **C:\Program Files\MSC.Software\Marc_Documentation\20XX.0.0**	To make optimal use of the documentation, it should be installed on a computer where Marc and Mentat are available. This way, documents can be accessed interactively from within Mentat, and Marc example problems can be run. You can install the documentation either before or after installing Marc and Mentat. The documentation installer will check if there is a corresponding Marc and Mentat version installed, where the Marc and Mentat installer will check if there is a corresponding documentation version installed, so that they can be linked together. The installer will prompt you to set the location where the documentation needs to be installed. This defaults to: `C:\Program Files\MSC.Software\Marc_Documentation\20XX.0.0` You can now install the documentation to any other location. The directories `doc` and `examples` will be created in the directory specified by you.

B Microsoft Windows: Mentat Files and Subdirectories

Mentat Files and Subdirectories

The Mentat version you have received contains a full set of subdirectories listed below.

Table 2-1 Contents of the Mentat Directory

Basic set:	
3dx	3-D mouse driver
bin	Shell scripts and programs for Mentat
ctkernel	CAD import libraries
help	Mentat online help files
lang	Localization dictionaries
materials	Mentat material files
materials_pre2010	Old Mentat material files
menus	Mentat menu files
parasolid	Parasolid schema files
python	Python installation
qt	Graphics plug-ins
scasystem	Components interface
shlib	Shared libraries
utilities	SIFT Python scripts

C Linux: Marc/Mentat Files, Subdirectories and Installation

Marc Files and Subdirectories

The Marc version you have received contains a full set of subdirectories listed below. You can save disk space by removing the example subsets that you do not need.

Table C-1 Contents of the Marc installation

Basic set	Contents: required as minimum
bin	executable Marc programs
tools	shell scripts to run and maintain the Marc programs
AF_flowmat	material data for database
xdr_lib	machine dependent libraries
Extended set	**Contents: only for use with user subroutines**
common	insert files containing Marc common blocks
lib	binary libraries with the compiled Marc routines
lib_shared	system dependent shared libaries.
user	templates for all available Marc user subroutines
Examples	**Contents: example files**
demo	input files and user subroutines for the *Marc Volume E: Demonstration Problems*
demo_ddm	input files and user subroutines for the single parallel machine as well as the network parallel version of Marc
demo_table	input files and user subroutines for the Marc Volume E: Demonstration Problems based upon Table driven input format.
test_ddm	one, two, and four processor test examples for installation testing of the single parallel machine as well as the network parallel version
Utilities	**Contents**
pldump	source routines for the post-file conversion programs pldump, pldump13 and pldump2000
intelmpi	MPI libraries for network parallel version

Mentat Files and Subdirectories

The Mentat version you have received contains a full set of subdirectories listed below.

Table C-2 Contents of the Mentat directory unloaded from installation package

Basic set	
`3dx`	3D mouse driver
`bin`	Shell scripts and programs for Mentat
`ctkernel`	CAD import libraries
`help`	Mentat online help files
`lang`	Localization dictionaries
`materials`	Mentat material files
`materials_pre2010`	Old Mentat material files
`menus`	Mentat menu files
`parasolid`	Parasolid schema files
`python`	Python installation
`qt`	Graphics plug-ins
`scasystem`	Components interface
`shlib`	Shared libraries
`utilities`	SIFT Python scripts

Installation Procedure

Download the file marc_20XX_linux64_rh7.1_7.3.tar.gz.

Steps	Command Information	Description
Step 1: Security	chmod +x msc_licensing_Helium_<platform>.bin ./msc_licensing_Helium_<platform>.bin For nodelocked license, run /msc/MSC.Licensing/Helium/bin/lmhostid cp license.dat /msc/MSC.Licensing/Helium/licenses/license.dat For csh and similar: setenv MSC_LICENSE_FILE /msc/MSC.Licensing/Helium/licenses/license.dat For Bourne shell and similar MSC_LICENSE_FILE=/msc/MSC.Licensing/Helium/licenses/license.dat export MSC_LICENSE_FILE Edit license.dat to set SERVER and DAEMON (see chapter 3 below for details). Start the license server: /msc/MSC.Licensing/Helium/bin/lmgrd -c /msc/MSC.Licensing/Helium/licenses/license.dat -l /msc/MSC.Licensing/Helium/lmgrd.log	Download FLEXlm from the Solutions Download Center. Select the product *MSC Licensing*, version Helium. Run the installation executable. It may be necessary to give execution permission (chmod +x) to the downloaded file. Follow the installation instructions. Default installation directory is `/msc/MSC.Licensing/Helium` Obtain a license file from MSC Software. If a nodelocked license is to be used, then obtain FLEXlm hostid with /msc/MSC.Licensing/Helium/bin/lmhostid Copy the license file to the proper location. Standard location is `/msc/MSC.Licensing/Helium/licenses/license.dat` Set the environment variable MSC_LICENSE_FILE to point to the license file. The license server must be running in order to run marc and Mentat.

Steps	Command Information	Description
Step 2: Download the product and start the install script	mkdir /tmp/marc cd /tmp/marc Copy the file from Download Center to this directory gunzip <file>.tar.gz tar xvf <file>.tar ./install.exe	Download the product from the Solutions Download Center and save the file to a temporary location. Marc and Mentat are available for Red Hat 7.3/7.5 and SUSE 12SP1/SP2. Please download the version compatible with your system. Unzip the file and extract the contents. Run the Marc installation script `install.exe` that is in the top-level directory. In order to complete the installation, you need to confirm that you want to install Marc 20XX on your computer, and accept the installation conditions (by entering "I Accept").
	Welcome to the Marc installation script for Linux systems Enter the directory to install the software (<current directory>)	Enter the path for the directory in which you want to install the Marc product(s). The default selection will be your current directory. You must have write permission to this directory.
Step 3: **Extract the files from the installation media**	**Main Menu** 1)Install Marc 2)Install Mentat o)Options ?)Help information q)Exit from the installation script Selection: **1**	Select option **1** to install Marc. This will take you to the Marc submenu. **Note:** Remember to install Marc before you install Mentat.

Steps	Command Information	Description
Select the platform	**Marc 20XX Menu** L1) Install for Linux 64-bit RHEL 7.3/7.5 (~1.1 GB)	Select the platform that you will be running Marc on. The script will show the platform for which you have downloaded the version and if this is compatible with your system, it is shown as a default value in brackets after the Selection prompt. Just press Return/Enter to use the default value, otherwise enter L1 to install the Red Hat 7.3/7.5 version and L2 to install the SUSE 12SP1/SP2. You will also be prompted whether you want to create a system wide link to the *marc20XX* run script.
	?) Help information r) Return to previous menu	
	r)Return to the previous menu	Choose the Return to previous menu option to return to the main menu.
	Mentat Installation **Mentat 20XX Menu** l1) Install for Linux 64-bit RHEL 7.3/7.5 (~1.3 GB)	Select option **2** from the main menu to install Mentat. Select the platform that you will be running Mentat on. The script will show the platform for which you have downloaded the version and if this is compatible with your system, it is shown as a default value in brackets after the Selection prompt. Just press Return/Enter to use the default value, otherwise enter l1 to install the Red Hat 7.3/7.5 version and l2 to install the SUSE 12SP1/SP2.

Steps	Command Information	Description
		You will be prompted to supply the pathname to the directory where Marc is installed. If the script determines that Marc is already installed to the default location of `<parent>/marc20XX`, then a default value for the path is displayed. Press Return/Enter to use the default value. If you do not have Marc at this site, you can ignore the prompt. You will also be prompted whether you want to create a system wide link in `/usr/local/bin` to the *mentat20XX* run script. You must be `root` to perform this.
	r)Return to the previous menu	Choose the Return to previous menu option to return to the main menu. Select q to exit the installation script.
3D Mouse	Mentat supports a 3-D Mouse (or Spaceball) device. If you want to use such a device with Mentat, please make sure that the right spacenav driver is installed on your system (see http://spacenav.sourceforge.net).	
Step 4: Download the documentation and start the install script	mkdir /tmp/documentation cd /tmp/documentation Copy the file from Download Center to this directory gunzip <file>.tar.gz tar xvf <file>.tar ./install_doc.exe	Download the documentation file from the Solutions Download Center and save the file to a temporary location. Unzip the file and extract the contents. Run the documentation installation script `install_doc.exe` that is in the top-level directory. It is important to enter the same directory in which you want to install the documentation as the one in which Mentat has been installed. When doing so, it is possible to interactively access the various documentation files from within Mentat and to run a selected number of procedure files illustrating some analysis capabilities (see also Checking Mentat).

Steps	Command Information	Description
Step 5: Checking Marc	cd <parent>/marc20XX/tools ./maintain **Marc Tools Menu** 1)Test Marc installation **Test the Marc installation** 1.1)Run a Marc job without user subroutine	**If you installed Marc, Mentat and the documentation**, you can check the Marc installation by following the instructions under Checking Marc from Mentat. Change your directory to `marc20XX/tools` and start the `maintain` script. This brings up a new menu. Select option **1**, **Test Marc installation**. A new **Test** menu appears. You must choose either Serial or Parallel depending on the license you have. Run one of the standard Marc demonstration examples as proof of a successful installation by choosing option **1.1**. If all goes well, one of the final messages on the screen should read: **Marc 20XX** **Exit number 3004**
	1.2)Run a Marc job with user subroutine	**If you have a Fortran compiler on your system**, choose a second demonstration example, option **1.2**. Here again, Marc should give the same exit message as shown above.
		Troubleshooting: 1. If you get an error message of `f90 not found` or `ifort not found` when running a job with a user subroutine and there is a Fortran compiler, its path needs to be appended to your path in the .cshrc or .profile file.
	0)Return to previous menu 0)Exit from the maintenance script	Repeatedly choose the Return/Exit option to leave the installation script. Choose option **0** to leave the maintenance script.

Steps	Command Information	Description
Step 6: **Checking Mentat** **Checking Marc from Mentat**	cd <parent>/mentat20XX ./bin/mentat Run a Demo problem by selecting the menu buttons: Help Run A Demo... Thermal/Structural Contact	Next, change your current directory to be the `<parent>` directory in which you installed Mentat, and then cd to mentat20XX. Enter the command `./bin/mentat` to start Mentat. From the Help menu, select Run A Demo..., and then select the Thermal/Structural Contact demo. It will run for 50 increments.
Step 7: **For Marc Parallel Network version only**		Please follow the *Parallel Network Version for Linux Installation Instructions* (Chapter 9: Hardware and Software Requirements and Chapter 9: Installation Notes) on installing and running jobs with the network version.