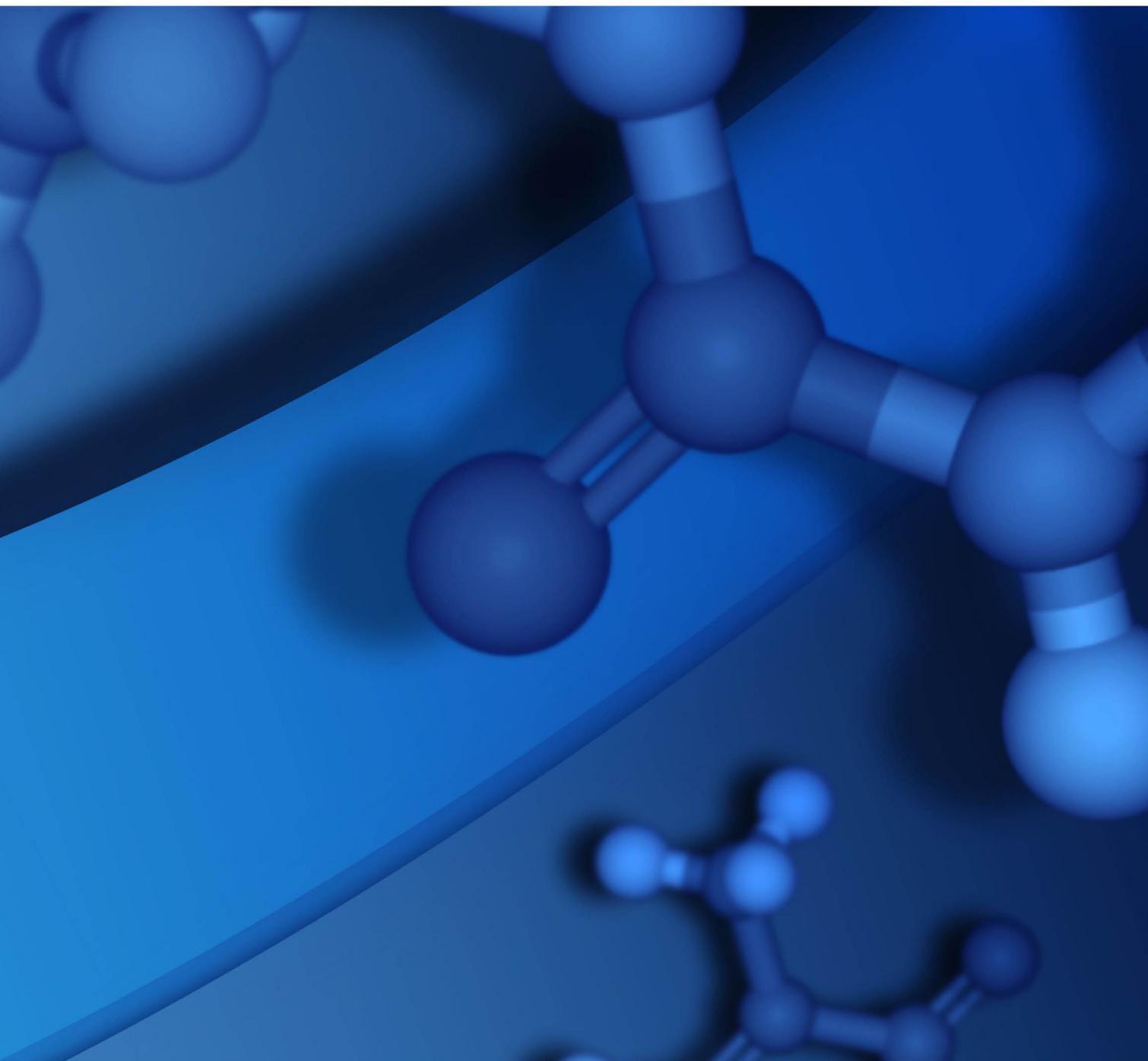


INSTALLATION AND ADMINISTRATION GUIDE

MATERIALS STUDIO
8.0



Copyright Notice

©2014 Dassault Systèmes. All rights reserved. 3DEXPERIENCE, the Compass icon and the 3DS logo, CATIA, SOLIDWORKS, ENOVIA, DELMIA, SIMULIA, GEOVIA, EXALEAD, 3D VIA, BIOVIA and NETVIBES are commercial trademarks or registered trademarks of Dassault Systèmes or its subsidiaries in the U.S. and/or other countries. All other trademarks are owned by their respective owners. Use of any Dassault Systèmes or its subsidiaries trademarks is subject to their express written approval.

Acknowledgments and References

To print photographs or files of computational results (figures and/or data) obtained using BIOVIA software, acknowledge the source in an appropriate format. For example:

"Computational results obtained using software programs from Dassault Systèmes Biovia Corp..
The *ab initio* calculations were performed with the DMol³ program, and graphical displays generated with Materials Studio."

BIOVIA may grant permission to republish or reprint its copyrighted materials. Requests should be submitted to BIOVIA Support, either through electronic mail to support@accelrys.com, or in writing to:

BIOVIA Support
5005 Wateridge Vista Drive, San Diego, CA 92121 USA

Contents

Introduction to Materials Studio	1	Gateway administration	28
Materials Studio Updates	1	Server gateway information	28
Further Information	1	Configuring a gateway	30
System requirements and recommendations ..	2	Server scripts	31
Client system requirements	2	Port	31
Hardware minimum	2	Managing a gateway	31
Operating systems	2	Editing gateway data	32
Server system requirements	3	Changing Gateway settings	32
Windows	3	Changing CPU core total	34
x86-64 Linux	3	Changing licensing parameters	34
System libraries	4	Server code administration	36
x86-64 Linux clusters	6	MPI command	36
System libraries	6	Archiving gateway job details	36
Finding Windows system information	8	Job information archiving	37
Available disk space	8	Job information file format	38
Operating system revision and service pack number	8	Gateway queuing	38
Memory	8	Gateway security	40
Internet Explorer version	8	Authentication and authorization	40
Graphics information	8	Access restrictions	40
Installing Materials Studio	9	Transport protocol	40
Installing Materials Studio on Windows systems	9	Gateway default security and enhancements Default security settings	40 41
Installing the Materials Visualizer client and the server programs	9	Security enhancements	41
Configuring gateways on Windows systems ..	10	Gateway access control	42
Installing Materials Studio on Linux systems ..	11	Access control definitions in allow.txt	42
Installation	11	IP addresses	43
Configuring gateways on Linux systems	14	Regular expression syntax	43
Configuring Intel MPI on parallel Linux systems	14	Gateway security via user authentication	43
Troubleshooting	17	Effects of user authentication	44
Installing Materials Studio with a Gaussian server	17	Administration level tasks	44
Installing Materials Studio on a server with a queuing system	18	User level tasks	47
Configuring Materials Studio for use with a queuing system	19	Gateway transport protocol	47
Configuring a queuing system from the command line	21	Effects of SSL transport	47
Using queues on clusters	21	Administration level tasks	48
Queue permissions	22	User level tasks	48
Troubleshooting	23	Licensing Materials Studio	49
Removing Materials Studio	25	New features in Accelrys License Pack for Materials Studio 8.0	49
Windows systems	26	Users with current maintenance	49
Linux systems	26	New users or users adding new products with this release	50
		Materials Studio license features	50
		Materials Visualizer auto trial licensing	52
		Installing a temporary license	52
		During Materials Studio installation	52

After Materials Studio installation	53
Licensing on Windows	53
Manually installing Accelrys License Pack	53
Removing Accelrys License Pack	54
Licensing environment	54
Connecting to a license server	54
Licensing on UNIX	54
Manually installing Accelrys License Pack	55
Removing Accelrys License Pack	55
Changing the License Pack location used by Materials Studio	55
Connecting to a license server	56
Accelrys license administration tools	56
Running the Accelrys Licensing Wizard	56
Running the Accelrys License Administrator ..	57
Accelrys License Pack help documentation	57

Introduction to Materials Studio

The following topics and their subtopics document installation, licensing, and gateway administration and security in Materials Studio:

- [System requirements and recommendations](#)
- [Installing Materials Studio](#)
- [Gateway administration](#)
- [Gateway security](#)
- [Licensing Materials Studio](#)

Materials Studio Updates

Occasionally, enhancements to Materials Studio are available post release.

To review the enhancements and access instructions for installation, login to the BIOVIA Download Center:

<https://accelrysdlc.subscribenet.com/>

Further Information

For more information about the Materials Studio and other Accelrys software products, visit BIOVIA Support on the Web: <https://community.accelrys.com/index.jspa>

System requirements and recommendations

This topic specifies the minimum hardware configurations required by Materials Studio and the operating systems on which Materials Studio is supported. In general, the better the hardware configuration of your computer, the better your experience with Materials Studio will be.

For information on the requirements of the Accelrys License pack please refer to the [License Pack documentation](#).

Client system requirements

Hardware minimum

- Intel® Pentium® 4 or compatible processor running at 1 GHz or greater
- Standard Microsoft® 2-button mouse (3-button mouse or 2-button wheel mouse strongly recommended)
- 1024 × 768 display resolution
- 2 GB RAM (4 GB recommended)
- 2.7 GB disk space for a complete installation
- 16-bit / 65536 colors

Note that the hard disk space required for installation can vary between 1 GB and 2.7 GB on Windows operating systems, depending on the components selected for installation. Additional space will be required if system updates are necessary.

To achieve a more satisfying experience with Materials Studio, particularly in terms of the client software, it is highly recommended that your system be substantially better than the minimum specification above. This is particularly important if running server software locally on the same PC as the client software.

Attention to the following areas can make a significant improvement to your overall user experience and enjoyment of the software:

- Memory - the more RAM your computer has, the better.
- CPU speed - the faster, the better.
- A good OpenGL® graphics card can make a significant difference to the graphics performance.
- The higher the display resolution, the more flexibility you will have in displaying and managing document and dialog windows.
- Mouse - although not absolutely necessary, a three-button mouse or a two-button mouse with a wheel is strongly recommended. It will make the more complex structure manipulations easier and increase your efficiency and enjoyment of Materials Studio.

For example, a PC with 2 GB RAM, a CPU speed of 2.0 GHz, a screen resolution of 1024 × 768, and a good OpenGL graphics card should perform well. PC hardware capabilities are currently advancing so rapidly that a new commodity desktop PC would have significantly higher specifications than these.

Operating systems

- Windows 7 (Professional & Enterprise) - SP1 (32-bit and 64-bit) uses 32-bit binary Materials Studio client
- Windows 8 (Basic, Professional & Enterprise) - (32-bit and 64-bit) uses 32-bit binary Materials Studio client
- Windows 8.1 (Basic, Professional & Enterprise) - (64-bit only) uses 32-bit binary Materials Studio client

Browsers

The following browsers are supported on client operating systems:

Browser	Client Operating System		
	Windows 7	Windows 8	Windows 8.1
Internet Explorer 9	•		
Internet Explorer 10	•	•	
Internet Explorer 11			•
Firefox ESR	•	•	•
Chrome	•	•	•

Server system requirements

Windows

Hardware minimum

- Intel Pentium 4 or compatible processor running at 1 GHz or greater
- 2 GB RAM (4 GB recommended)
- 2.2 GB disk space

This is a recommended minimum specification. See the [PC client system requirements](#) section above for further recommendations to help you get the best out of Materials Studio. There are no graphics-related requirements for running Materials Studio on Windows servers.

Operating systems

- 32-bit
 - Windows Server 2008, all editions - SP2
 - Windows 7 (Professional & Enterprise) - SP1
- 64-bit
 - Windows Server 2008, all editions - SP2 and R2 SP1
 - Windows 7 (Professional & Enterprise) - SP1
 - Windows 8 (Professional & Enterprise)
 - Windows 8.1 (Professional & Enterprise)
 - Windows Server 2012, all editions - base and R2

Note: If you intend to run server applications in parallel on a Windows system that does not have a network connection, you should install the Microsoft Loopback Adapter by following instructions appropriate to your operating system.

x86-64 Linux

Hardware minimum

- Intel Pentium 4 or compatible (x86-64) processor running at 1 GHz or greater
- 2 GB RAM (4 GB recommended)
- 2.7 GB disk space

This is a recommended minimum specification. Greater processing power, speed, and memory are recommended for any heavy duty use. There are no graphics-related requirements for running Materials Studio on Linux servers.

Operating systems

- x86-64 (64-bit)
 - Red Hat Enterprise Linux Server and Desktop 5
 - Red Hat Enterprise Linux Server 6
 - Red Hat Enterprise Linux Server 7
 - SuSE™ Linux Enterprise Server 11

System libraries

On Red Hat Enterprise Linux and SuSE Linux Enterprise Server systems, the standard C++ libraries for backward compatibility of the C++ compiler are required for proper operation of the Materials Studio server codes. On servers with x86-64 processors, 32-bit compatibility libraries are required for the proper operation of Materials Studio and Accelrys License Pack. These libraries should be installed prior to the installation of Materials Studio and Accelrys License Pack.

The packages that contain the required libraries are listed below for the two most recent OS minor versions at the time of Materials Studio release:

Operating system	Package
Red Hat Enterprise Linux 5.10	glibc-2.5-118 (32-bit and 64-bit)
	libgcc-4.1.2-54.e15 (32-bit and 64-bit)
	libstdc++-4.1.2-54.e15 (32-bit and 64-bit)
	redhat-lsb-4.0-2.1.4.e15 (32-bit and 64-bit)
	compat-libstdc++-33-3.2.3-61 (32-bit and 64-bit)
Red Hat Enterprise Linux 5.11	glibc-2.5-122 (32-bit and 64-bit)
	libgcc-4.1.2-55.e15 (32-bit and 64-bit)
	libstdc++-4.1.2-55.e15 (32-bit and 64-bit)
	redhat-lsb-4.0-2.1.4.e15 (32-bit and 64-bit)
	compat-libstdc++-33-3.2.3-61 (32-bit and 64-bit)
Red Hat Enterprise Linux 6.5	glibc-2.12-1.132.e16 (32-bit and 64-bit)
	libgcc-4.4.7-4.e16 (32-bit and 64-bit)
	libstdc++-4.4.7-4.e16 (32-bit and 64-bit)
	redhat-lsb-4.0-7.e16 (32-bit and 64-bit)
	compat-libstdc++-33-3.2.3-69.e16 (32-bit and 64-bit)

Operating system	Package
Red Hat Enterprise Linux 6.6	glibc-2.12-1.149.el6 (32-bit and 64-bit)
	libgcc-4.4.7-11.el6 (32-bit and 64-bit)
	libstdc++-4.4.7-11.el6 (32-bit and 64-bit)
	redhat-lsb-4.0-7.el6 (32-bit and 64-bit)
	compat-libstdc++-33-3.2.3-69.el6 (32-bit and 64-bit)
Red Hat Enterprise Linux 7.0	glibc-2.17-55.el7 (32-bit and 64-bit)
	libgcc-4.8.2-16.el7 (32-bit and 64-bit)
	libstdc++-4.8.2-16.el7 (32-bit and 64-bit)
	redhat-lsb-4.1-24.el7 (32-bit and 64-bit)
	compat-libstdc++-33-3.2.3-71.el7 (32-bit and 64-bit)
SuSE Linux Enterprise Server 11 (SP2)	glibc-2.11.3-17.31.1 (64-bit)
	libgcc46-4.6.1_20110701-0.13.9 (64-bit)
	libstdc++46-4.6.1_20110701-0.13.9 (64-bit)
	glibc-32bit-2.11.3-17.31.1 (32-bit)
	libgcc46-32bit-4.6.1_20110701-0.13.9 (32-bit)
	libstdc++46-32bit-4.6.1_20110701-0.13.9 (32-bit)
SuSE Linux Enterprise Server 11 (SP3)	glibc-2.11.3-17.54.1 (64-bit)
	libgcc_s1-4.7.2_20130108-0.15.45 (64-bit)
	libstdc++6-4.7.2_20130108-0.15.45 (64-bit)
	glibc-32bit-2.11.3-17.54.1 (32-bit)
	libgcc_s1-32bit-4.7.2_20130108-0.15.45 (32-bit)
	libstdc++6-32bit-4.7.2_20130108-0.15.45 (32-bit)

If you have a newer version of a package than those listed above, then no action should be required.

Red Hat libraries

To check if the appropriate package is installed on Red Hat, enter:

```
rpm -q <package name>
```

If a required package is missing on Red Hat, you can download and install it using the yum command:

```
yum install <package name>
```

For 32-bit missing packages on Red Hat you need to specify the .i686 version:

```
yum install <package name>.i686
```

If your Red Hat version is not listed in the system requirements please install the packages of the same major release which is closest to your Red Hat version.

SuSE libraries

To check if the appropriate package is installed on SuSE, enter:

```
rpm -q <package name>
```

If you are running SuSE Linux Enterprise Server and a required package is missing, ask your system administrator to download and install it using YaST (choose *Software* / *Software Management*).

x86-64 Linux clusters

IMPORTANT! Installation of Materials Studio on Windows clusters is not supported.

Hardware minimum

- Intel Pentium 4 or compatible (x86-64) processors running at 1 GHz or greater
- Homogeneous clusters (identical processors, identical OS, libraries, etc., on each node) only
- Beowulf architecture, with a head node that holds the file system, Materials Studio installation, and user data. These files must be mounted and visible to the compute nodes at the same location as on the head node
- 2.7 GB disk space on the head node
- 2 GB RAM (4 GB recommended) per node
- Interconnects (http://software.intel.com/sites/products/collateral/hpc/cluster/mpi_indepth_40.pdf) supported by Intel MPI.
- Compute nodes must be set up to communicate with each other and the head node through rsh or ssh without a password (for MPI)
- A license file or license server must be installed on the head node, unless the compute nodes are configured so that they can access a license server elsewhere

This is a recommended minimum specification. Greater processing power, speed, and memory are recommended for any heavy duty use. It is *strongly* recommended that you use a [supported queuing system](#) to ensure an even distribution of jobs between the different processors. There are no graphics-related requirements for running Materials Studio on Linux clusters.

For details of Intel MPI please refer to the Intel Cluster MPI Libraries documentation (<http://software.intel.com/en-us/articles/cluster-mpi-libraries/>).

Operating systems

- x86-64 (64-bit)
 - Red Hat Enterprise Linux Server and Desktop 5
 - Red Hat Enterprise Linux Server 6
 - Red Hat Enterprise Linux Server 7
 - SuSE™ Linux Enterprise Server 11

Currently, only the Forcite, Mesocite, CASTEP, DMol³, DFTB+, GULP, MesoDyn, and ONETEP servers are supported for parallel execution on Linux clusters. All server codes are supported for serial operation on Linux clusters. It is recommended to use one of the supported queuing systems to utilize computational resources of a Linux cluster efficiently.

System libraries

On Red Hat Enterprise Linux and SuSE Linux Enterprise Server systems, the standard C++ libraries for backward compatibility of the C++ compiler are required for proper operation of the Materials Studio server codes. On clusters with x86-64 processors, 32-bit compatibility libraries are required for the

proper operation of Materials Studio and Accelrys License Pack. These libraries should be installed prior to the installation of Materials Studio and Accelrys License Pack. The packages that contain the required libraries are listed in the table [above](#).

Finding Windows system information

Many system properties for computers running a Windows operating system are available from the System Properties dialog, which can be accessed by selecting *Settings | Control Panel | System* from the Windows *Start* menu.

More detailed information may be available from the following sources.

Available disk space

Double-click on the *My Computer* icon and then select the drive you want to check. On the *File* menu, select *Properties*. A pie chart shows how much free and used space is on the disk.

Operating system revision and service pack number

Use the System Information tool, accessed by selecting *Programs | Accessories | System Tools | System Information* from the Windows *Start* menu. The operating system revision, including the service pack number, will be displayed in the panel on the right.

Memory

Use the System Information tool, accessed by selecting *Programs | Accessories | System Tools | System Information* from the Windows *Start* menu. The amount of installed RAM will be displayed in the panel on the right.

Internet Explorer version

Start Internet Explorer and access the About dialog by selecting *Help | About Internet Explorer* from the menu bar.

Graphics information

Right-click in the desktop area and select *Properties* from the shortcut menu. In the Properties dialog, go to the *Settings* tab. The number of colors and desktop area are shown here. Click the *Advanced* button to show details of the graphics board, on the *Adaptor* tab. Click the *Properties* button next to the details of the graphics board. The driver version numbers are shown on the *Driver* tab.

Installing Materials Studio

The following topics describe the procedures involved in the installation and removal of Materials Studio.

Note: It is essential that the same version of Materials Studio is used in the client and server installations. Inconsistent versions of client and server may not interact correctly.

Note: <MS_INSTALL> is used to refer to the installation location of Materials Studio. On 32-bit Windows systems by default this is C:\Program Files\Accelrys\Materials Studio 8.0. On 64-bit Windows systems by default this is C:\Program Files (x86)\Accelrys\Materials Studio 8.0 x64 Server for the supported servers and C:\Program Files (x86)\Accelrys\Materials Studio 8.0 for the client.

Installing Materials Studio on Windows systems

See [Installing Materials Studio on Linux systems](#) for information about installing Materials Studio on those platforms. See [Installing Materials Studio on a server with a queuing system](#) for additional information about queuing system support and installation. See the "Working with gateways" topic in the Materials Studio Online help and the Server Console help system for details of gateway configuration and management.

Note: If you have a version of the Materials Studio client already installed, you must stop any license servers that are running and remove the existing client before you can install the new version. Refer to [Removing Materials Studio](#) for instructions on how to uninstall Materials Studio.

Installing the Materials Visualizer client and the server programs

IMPORTANT! Installation of Materials Studio on Windows clusters is not supported.

To install the Materials Visualizer client and the servers on a Windows system

1. Installation of Materials Studio may require that your machine be restarted once or twice. In order to ensure that you do not lose any valuable information, it is recommended that you close all other Windows applications and temporarily disable any antivirus software.
2. Insert the Materials Studio media into your computer. If AutoPlay is enabled, then the Materials Studio Setup program will start automatically. If it does not or if you have AutoPlay switched off, then browse to the media and double-click on `Autorun.exe` to start up the program.
3. Click the *Install Materials Studio* button on the Materials Studio Setup window to start the installation process.
4. Follow the directions given during the setup procedure.
5. During the installation process, you may be requested to restart your system either once or twice. If you choose not to restart your system, the installer will exit. You will not be able to continue the installation process and proceed further until the requested restart has occurred. If you restart, the installer will automatically continue when you log back into the machine.
6. When installation of Materials Studio is complete, you will be given the opportunity to install a license. Refer to the [licensing](#) topics for more details.
7. When the installation is complete, you can start Materials Studio from the Windows *Start* menu.
8. If you opted not to start the gateway service during installation, this must be initialized manually before any jobs are submitted.
This can be done by navigating to the Gateway subdirectory of the Materials Studio installation, then running `gwstartservice.bat`.

Note: If you intend to run server applications in parallel on a Windows system that does not have a network connection, you should install the Microsoft Loopback Adapter by following instructions appropriate to your operating system.

Configuring gateways on Windows systems

Gateways manage communication between client and server components of Materials Studio. If you have installed any server codes, then by default the gateway process is installed and started automatically. By default, this Windows service is installed to run under the *System* account. If you choose to run your gateway service under the *System* account, you will not be able to use the Windows Task Manager to kill processes spawned by the gateway service, for example, Materials Studio executables running when a job was launched.

If you have the [LSF or PBS queuing system](#) installed and wish to submit jobs from the Materials Studio gateway to the queuing system, you must modify the gateway service to run under an account other than the *System* account. This account must not be a member of the administrator or power users group.

To modify the gateway service to run under a different account

1. Log in using an account with system administrator privileges.
2. Select *Control Panel | Administrative Tools | Services* from the Windows Start menu to display the Services dialog.
3. Select the *Materials Studio Gateway* service from the list on the Services dialog and click the *Properties* button on the toolbar to display the Properties dialog.
4. Select the *This Account* option on the *Log On* tab, then enter the account name and the password for this account in the adjacent text boxes.

WARNING! The account name you enter must have "Full Control" permissions on the Gateway files in order for the Materials Studio gateway to start properly. These files are stored in %ALLUSERSPROFILE%\Accelerys\Materials Studio\8.0\Gateway-[i686 or x64]\root_default directory (where %ALLUSERSPROFILE% is usually C:\ProgramData on Windows).

5. Click the *OK* button on the Properties dialog.
6. The service that is currently running must be restarted to effect the change in accounts. Stop the service by pressing the *Stop Service* button on the Services dialog. Once the service has stopped, restart it by pressing the *Start Service* button.

Note: When Materials Studio is installed on Windows systems, the accompanying DHTML editor component is automatically also installed.

Installing Materials Studio on Linux systems

See [Installing Materials Studio on Windows systems](#) for information about installing Materials Studio on a PC. See [Installing Materials Studio on a server with a queuing system](#) for additional information about queuing system support and installation. See the Working with gateways topic in the Materials Studio Online help and the Server Console help system for details of gateway configuration and management.

Tip: If you are installing Materials Studio for use with a Gaussian server, some [further steps](#) are required before proceeding with installation of your Materials Studio server on Linux.

Installation

The Materials Studio gateway can be installed in either of two ways:

- [RPM-based](#): this installs system-wide and registers the Gateway service for system startup. [Automated](#) RPM-installation is also available.
- [non-root](#): this can be run as an unprivileged user account and unpacks into any directory which the user can access.

The second of these options is similar in nature to versions of Materials Studio before 6.0. In either case, a user account must exist to run the Gateway service; when installing as root this can be set up during installation.

If you do not have root access to the machine on which you wish to install Materials Studio, you may have to ask your system administrator to provide a suitable account and file space. Each host that you wish to use as a platform for Materials Studio server products must have its own private installation of the Materials Studio gateway and the relevant servers. Materials Studio gateway and server installations cannot be shared between multiple server host machines.

To install Materials Studio gateway and server products using RPM

1. Mount the media either locally or over the network.
2. Assuming the media drive is mounted on /CDROM, run the /CDROM/install command.
3. You will be asked whether you wish to perform an RPM installation. Press Y or ENTER to confirm.
4. When prompted, enter the directory in which you wish to install Materials Studio. For an RPM installation, this should be in an area writable only by the root user. The default is /opt/Accelrys.
5. You must specify the location of a License Pack installation. If you already have a sufficiently recent version installed, you can enter its location, otherwise enter a directory in which you wish to install it.
6. You will be prompted for the user, and possibly group, name under which you wish the Gateway to run. *Do not* enter root here; use an unprivileged account which the installer will create if it does not exist.
7. You will be asked if you wish to start the gateway. Press ENTER to start.
8. You will then be prompted for further information regarding [licensing](#).

Tip: The installer can also be run in non-interactive mode, with all options specified on the command line. Run /CDROM/install --help for more information on running the installer in this way.

To install Materials Studio gateway and server products as an unprivileged user

1. Mount the media either locally or over the network.
2. Assuming the media drive is mounted on /CDROM, run the /CDROM/install command.
3. When prompted, enter the directory in which you wish to install Materials Studio. This must be in an area to which the logged on user has write access.
4. You must specify the location of a License Pack installation. If you already have a sufficiently recent version installed, you can enter its location, otherwise enter a directory in which you wish to install it.
5. You will be asked if you wish to start the gateway. Press ENTER to start.
6. You will then be prompted for further information regarding [licensing](#).

Tip: You can also use this installation method as root, should you wish to maintain multiple installations of the same version of Materials Studio on one machine. Proceed as for an RPM installation, but answer N when asked whether to perform an RPM installation.

To install Materials Studio using automated system administration tools

The rpms directory of the Materials Studio installation media contains relocatable RPM files for Materials Studio. These can be installed using RPM directly or using system administration tools to deploy to multiple machines. In this case, the License Pack must be installed separately, and all configuration except for installation directory must be done using these environment variables:

- ACCELRYSS_LIC_PACK_DIR must point to an installation of the License Pack. This will be set by loading the profile settings installed by the License Pack.
- ACCELRYSS_USER_NAME can be used to set the user name that will run the Gateway service. This will be created if it does not exist. The default is AcceLrys.
- ACCELRYSS_GROUP_NAME can be used to set the default group of the user that the installer creates to run the Gateway service, if applicable. The default is AcceLrys.
- ACCELRYSS_DONT_START_GATEWAY will prevent the Gateway service from starting automatically, if set to any non-empty string. It will still be configured and set to run on system startup.

Cluster Node IP Address Considerations

When installing Materials Studio on clusters where the head node is connected to at least one external network in addition to the internal (inter-node) cluster network (and thus is addressable via more than one IP address and/or hostname) network addressing issues may possibly arise due to Materials Studio's use of Intel MPI for inter-nodal communication. As Intel MPI calls made from the head node to compute nodes are automatically stamped with a "callback" IP address to which the compute nodes' Intel MPI software will return the computed data, it is necessary to ensure that this callback IP address is accessible from the compute nodes in order for the software to function correctly.

To verify that your cluster meets this criteria, you should carry out the following test:

1. Login to the head node of your cluster as the superuser, and enter the command `ifconfig`.
2. For each active network interface the command will return details in the general format;

```
[name] Link encap:Ethernet HWaddr 00:17:A4:8F:65:18
inet addr:192.10.11.200 Bcast:192.10.11.255 Mask:255.255.255.0
.....
```

where [name] is the name of the network interface. For a cluster with one external and one internal network, the names *eth0* and *eth1* respectively are typically used.
3. Make a note of the value of the parameter `inet addr` for each named interface, with the exception of `lo` (the local loopback). These values are the various IP addresses of the head node on each network.
4. Login to one of the compute nodes in your cluster, and issue the `ifconfig` command again. This will allow you to determine which of the head node's network interfaces refers to the cluster's internal network, as this will be named in the output seen on both nodes.
5. From the compute node enter the command `ping -c 1 <IP address>`, using the IP address for the head node on the *internal* network.
6. You should obtain output in the general format;

```
PING 192.10.11.200 (192.10.11.200) 56(84) bytes of data.
64 bytes from 192.10.11.200: icmp_seq=0 ttl=64 time=0.133 ms

--- 192.10.11.200 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.133/0.133/0.133/0.000 ms, pipe 2
0% packet loss indicates that the head node IP is accessible from the compute node and confirms that the compute and head nodes are on the same (internal) network.
```
7. Repeat the previous step using the head node IP address for the *external* network. If you again see 0% packet loss, your cluster is suitably configured for use with Materials Studio and no further modification should be necessary.
8. If you instead obtain a "network unreachable" error message, you will need to modify your cluster by adding a gateway to the head node and modifying the routing tables of the compute nodes so that all network traffic is by default routed via this gateway. Contact your system administrator (or BIOVIA Support) if you need assistance.

NFS mounted disks on Linux clusters

The installation of parallel Materials Studio servers on Linux clusters requires that the NFS volumes of the head node were mounted in hard mode from each node. It is important to check the current mounting scheme as follows:

```
% mount -t nfs
c11:/home/uid on /home/uid type nfs (rw,hard,intr,addr=...)
```

If the `hard` option is specified, then the cluster is configured correctly and can be used by Materials Studio parallel servers. If this option is not on the list, follow the installation sequence below.

To mount an NFS disk on a Linux cluster

1. Run the `su` command and enter the superuser password to become the root user.
2. If the system uses an automounter daemon (see `man automount`), edit the `/etc/auto.master` file to add the `hard` option, for example:
`/home /etc/auto.home -o hard`
3. In the case of a manual file mount, add the same options to the mount command, for example:
`% mount -o hard c11:/home /mnt/home`
4. Repeat step 3 on each node of the cluster.

Tip: When you are exporting a shared folder from the head to child nodes of the cluster, you should specify the `sync` option.

More detailed instructions on the recommended usage of NFS on Linux clusters can be found in the Red Hat Documentation (<http://docs.redhat.com/docs/>).

Configuring gateways on Linux systems

Gateways manage communication between the client and server components of Materials Studio. If you have installed any server codes, then the gateway process is automatically installed, configured, and started, if required. The non-root installer does not update your system, this means that the server gateway will not be started automatically when the system is restarted. The Materials Studio Linux installer creates the files required to do this, but the installation of these files in their correct location requires system administrator privileges. As it is recommended that you do not install Materials Studio under an account with system administrator privileges, the final installation stages are not attempted by the installer.

To set up your system so that the gateway is started automatically after a system restart

1. Log in using an account with system administrator privileges.
2. Copy the `msgateway_control_<port number>` file (<port number> is the number of the port used by the gateway daemon) in the Gateway directory at the top of your installation tree to the `/etc/init.d` folder:
`cp msgateway_control_<port number> /etc/init.d`
3. Use the `chkconfig` tool to create all the required links to the `/etc/init.d/msgateway_control_<port number>` file:
`/sbin/chkconfig --add msgateway_control_<port number>`

Configuring Intel MPI on parallel Linux systems

The main parameters governing parallel job execution in Materials Studio are:

- `cpucoretotal`: The total number of CPUs available on a system
- `mpicommand`: Launches MPI and provides information about the runtime environment

The values of these variables can be setup manually by invoking the `config/configure` script (in the `<MS_INSTALL>/etc/Gateway/` directory of your installation) with an `edit` argument:

Number of CPUs

```
./config/configure edit cpucoretotal=???
```

MPI Command

```
./config/configure edit mpicommand=???
```

Example

```
./config/configure edit mpicommand="<MS_INSTALL>/bin/mpirun -f mpd.hosts -  
machinefile mpd.hosts -np"
```

Alternatively, this setting can be modified using the instructions provided in the [Editing gateway data](#) topic.

In order for Intel MPI to be able to send slave commands to start parallel processes, the user installing Materials Studio must be able to connect to the slave machine (usually localhost for SMP systems and one of the node hostnames in a cluster) without receiving a prompt for a password. By default Intel MPI uses the ssh program to connect to slaves (see below for instructions on how to configure [ssh](#)), but it also supports a variety of other methods which must be properly configured. For details of Intel MPI please refer to the Intel MPI Library - Documentation (<http://software.intel.com/en-us/articles/intel-mpi-library-documentation>).

Note: All the variables needed to modify Intel MPI runtime behavior (in particular those mentioned in Environment Variables at http://software.intel.com/sites/products/documentation/hpc/ics/impi/41/lin/Reference_Manual/Environment_Variables_hydra.htm) as well as other important Linux environment variables can be specified in the <MS_INSTALL>/etc/Gateway/ms_vars.sbd file either by manually editing the file or using the web interface. See the [Editing gateway data](#) topic for more information.

Enabling ssh

The ssh package is typically installed and enabled by default for SuSE Linux Enterprise Server, and for Red Hat Enterprise Linux (both operating systems use the *openssh* implementation of ssh, which is reflected in the package names). To check that ssh-server is enabled and, if necessary, to enable ssh-server, follow the instructions below.

To check whether ssh is enabled

1. Enter the following:
ssh localhost
2. You should get a new command prompt logged in to the home directory of the current user. If this doesn't happen, then you will not be able to run SMP parallel processes on this machine and you should check that ssh is enabled.
3. If you receive a password prompt, you should [enable passwordless login using ssh](#).
4. The first time that a connection is made to a machine using the ssh command, you will normally see a response similar to the example below:

```
The authenticity of host 'dewer2 (133.27.92.5)' can't be established.  
RSA key fingerprint is cf:d6:77:c6:db:42:99:6f:11:07:f3:38:46:7b:f9:05.  
Are you sure you want to continue connecting (yes/no)?
```

You should answer **yes** to continue connecting and then, after a short period, you will receive the new login prompt on the selected host as described in step 2.

Note: You can avoid further prompts the first time a connection is made by setting the StrictHostKeyChecking option on the ssh client set to "no". For instance, if you have OpenSSH make sure that the ~/.ssh/config file includes the following line:
StrictHostKeyChecking no

If ssh is not working (i.e., you are unable to log in and receive a message such as [connection refused](#) when attempting to connect to the machine using ssh) on your SuSE Linux system, you should consult your network administrator to request that it be enabled.

A standard installation of SuSE Linux Enterprise Server or Red Hat Enterprise Linux does not install and/or enable rsh, only ssh. If you need to use rsh rather than ssh to communicate between parallel processes, you should contact your network administrator to request that rsh (and rsh-server) are installed and enabled. If you are installing on a cluster, this will be necessary on all the machines in the cluster.

To enable passwordless connection using ssh

1. Create a public-private key pair for the user who is installing the Materials Studio gateway:

```
$> ssh-keygen -t rsa
```

Do not create a passphrase. When prompted for one, simply press the ENTER key twice.
2. Place the private and public keys (usually in files called `id_rsa` and `id_rsa.pub`) in the `~/.ssh` directory. Create this directory if it does not already exist.
3. Copy the `id_rsa.pub` file to a file called `authorized_keys2` in the `.ssh` directory. If this file already exists, you should stop and seek further advice before continuing as you may break existing authentication processes.
4. The directory `~/.ssh` and all of the files in it should not be accessible to any other users. To ensure that this is the case, follow the procedure below:
 - i. Change directory to the location where the keys are held:

```
$> cd ~/.ssh
```
 - ii. Set permissions on all files here to limit access to the owner only:

```
$> chmod 600 *
```
 - iii. Set permissions on the directory to limit access to the owner - group and world permissions prevent anyone else from seeing the content of the directory:

```
$> chmod 700 .
```
5. The home directory of the Materials Studio user must be read-only for other users. If it has write permissions for either group or world, then ssh will always request the user's password when a login attempt is made. To ensure passwordless connection, follow the procedure below:
 - i. Change directory to the user's home directory:

```
$> cd ~
```
 - ii. Ensure that there is no write permission for group or world. Use a stronger setting if appropriate (for example [700](#)):

```
$> chmod 755 .
```

If you still get a password request when attempting to connect to the machine using ssh, check the above conditions carefully. It is usually the case that either the `.ssh` directory or some files in it are accessible to group or world, or there are write permissions for either group or world on the user's home directory.

Note: This method of authentication carries a security risk due to the need to use a null passphrase on your key files. This would allow anyone who has access to your systems to potentially impersonate the user who installs and runs the Materials Studio software. If you choose to use ssh for SMP or cluster communications, Accelrys recommends that you consult your network administrator on the implications of this approach and agree methods to limit the impact of the security risks for your organization.

Linux clusters without queuing systems

To use the cluster efficiently you should access its resources using a queuing system which employs load balancing (if you do not use a queuing system you should not run more than one job at a time).

If there is no queuing system, the Intel MPI resource file `machines.LINUX` must be manually updated in order for `mpirun` to recognize all the processors in a Linux cluster. The `machines.LINUX` file is located in the Materials Studio installation under `share/data`. For each node in your cluster, you should add a line to the `machines.LINUX` file giving the name for each node as returned by the `hostname` command. If a node has more than one processor, this should be indicated by appending a colon and then the number of processors to the name. For example, the `machines.LINUX` file for a cluster of three nodes named 'venus', 'earth', and 'mars', each with two processors, would contain the following lines:

```
venus:2  
earth:2  
mars:2
```

Setting up a cluster for use with ssh

When setting up a cluster to use ssh, you need to ensure that the user installing Materials Studio can log in to all machines in the cluster without having to enter the password or passphrase. If you have difficulties in doing this, you should consult your network administrator and request that the appropriate configuration and system changes are made.

Troubleshooting

Problem	Possible corrective action
There is excessive network traffic between the Materials Studio client and the gateway server.	The rate at which the Materials Studio client polls the gateway server for updates in the status of running jobs can be configured using the web browser interface. By default, the polling interval is 1 s, but this setting can be configured for individual server programs by editing the value of <i>MinimumPollingInterval</i> (in ms) for the server program in question on the Installed Servers page of the web administration interface. Larger polling interval values will reduce network traffic, but will also reduce the responsiveness of the client to changes in job status.

Installing Materials Studio with a Gaussian server

Materials Studio will automatically recognize and connect to Gaussian installed on Windows servers. However, if you are installing Materials Studio for use with a Linux Gaussian server, you will need to perform some extra steps to ensure that Materials Studio can access and communicate with the Gaussian server correctly.

1. Ensure that the `$g03root` environment variable is set correctly (`$g09root` for Gaussian 09), in the command line enter:

```
echo $g03root
```

This should return the location of the Gaussian installation. For more information on this directory, please refer to the Gaussian documentation http://www.gaussian.com/g_tech/g_ur/m_running.htm

2. Ensure that Materials Studio will be able to access information from the Gaussian login scripts `g03.login` or `g03.profile` according to your shell (`g09.login` or `g09.profile` for Gaussian 09). For example, navigate to the directory containing the script and enter:

for bash:

```
./g03.profile
```

for csh:

```
source ./g03.login
```

3. Continue with your Materials Studio [Linux installation](#).

Tip: If Gaussian is installed after your Materials Studio server, you must perform these steps and then stop and restart your Materials Studio gateway in order to configure the gateway scripts correctly.

Installing Materials Studio on a server with a queuing system

This version of Materials Studio allows you to configure a Materials Studio gateway so that all jobs submitted to this gateway are fed into a queuing system instead of being launched instantaneously. This allows you to optimize the use of server resources on your network.

The following queuing systems are officially supported with the current version of Materials Studio:

- Altair PBS (Portable Batch System) Pro version 12.2 on Windows and Linux (www.pbsworks.com/Product.aspx?id=1)
- IBM Platform LSF (Load Sharing Facility) version 9.1 on Windows and Linux (www-03.ibm.com/systems/technicalcomputing/platformcomputing/products/lfs/index.html)
- Oracle Grid Engine (OGE) 6.2 Update 7 on Linux (www.oracle.com/us/products/tools/oracle-grid-engine-075549.html)
- Univa Grid Engine (UGE) 8.2 on Linux (<http://www.univa.com/products/grid-engine>)
- Adaptive Computing TORQUE 4.2 on Linux (<http://www.adaptivecomputing.com/products/open-source/torque/>)

If you have a Linux cluster, it is recommended that you use one of the supported queuing systems to ensure an even distribution of jobs between the different processors.

Note: Materials Studio supports PBS Pro, OGE, UGE, LSF, and TORQUE on clusters. For parallel execution, CASTEP, DMol³, Forcite, Mesocite, GULP, MesoDyn, ONETEP and DFTB+ are supported for submission on Linux clusters and SMP machines running PBS Pro, OGE, UGE, TORQUE, or LSF; DFTB+ jobs, however, only use cores in one SMP node and will not be distributed between cluster nodes. Different versions of these queuing systems may work with Materials Studio, but have not been tested. For the installation and configuration of the queuing system itself, refer to the appropriate documentation.

Tip: For OGE and UGE queuing systems a pre-defined parallel environment named make must be configured.

Notes:

Third party queuing systems do not check the availability of licenses before attempting to execute jobs from their queue. As a result, if a queue is configured to execute more jobs than there are licenses available, jobs may fail unexpectedly. There are two ways of overcoming this problem:

- Configure your queuing system to execute no more queued jobs than you have licenses. When submitting jobs of different types with different numbers of available licenses, create individual queues for each job type.
- Turn on the license waiting functionality in the Materials Studio gateway, which will cause the call from the server executable to the Accelrys License Pack to be blocked until the requisite licenses become available (provided the number of licenses requested does not exceed the maximum number available). See the [Gateway administration](#) topic or the "Materials Studio licensing" topic in the Accelrys License Administrator help for details of how to enable license waiting.

If you do not wish to implement either of the options above, you should take extra care when managing job submission in order to prevent unexpected job failures.

Configuring Materials Studio for use with a queuing system

It is recommended that the installation and configuration of the queuing system is carried out prior to the installation of the Materials Studio gateway.

To configure Materials Studio for use with a queuing system

1. Follow the installation instructions provided with your queuing system to ensure that it is started automatically after a system restart.
2. Log in under the account where you are going to perform the installation of the Materials Studio servers and verify that the path to the queuing commands is listed in the PATH environment variable. Under Linux, you may have to edit your shell resource file (for example `.cshrc` for the C shell) to add appropriate commands for setting the PATH variable.
 - For LSF, the PATH variable is set by sourcing either `<LSF_INSTALL>/conf/cshrc.lsf` (for the C shell) or `<LSF_INSTALL>/conf/profile.lsf` (for bash or sh). These files should be sourced at the end of the appropriate shell resource file.
 - For PBS on Windows, there is a file containing a cached copy of the PATH variable which is used by PBS instead of the environment variable itself. This file is called `pbs_environment`, and can be found in the PBS home directory - typically `<PBS_INSTALL>\home\pbs_environment` if PBS Pro is installed to the default location. This file must be edited so that the Accelrys License Pack shared library is "on" this PATH (adding it to the PATH environment variable will not achieve this). Note that there are two lines in the file, for Path and PATH, which should both be edited. The default location of the License Pack [depends on the operating system](#) (the library is called `ls_license_vs2008.dll`).
 - For OGE and UGE, the PATH and other variables are set by sourcing either `$SGE_ROOT/$SGE_CELL/common/settings.csh` (for the C shell) or `$SGE_ROOT/$SGE_CELL/common/settings.sh` (for bash or sh). These files should be sourced at the end of the appropriate shell resource file.
 - For TORQUE no changes to the PATH variable are required.

Note: <LSF_INSTALL> and <PBS_INSTALL> are used to refer to the installation locations of LSF and PBS respectively. These may differ depending on the operating system.

3. Confirm that the queuing system is properly configured and active; for example, verify that queuing system commands such as `qstat` are functioning correctly and that the appropriate daemons are running.
4. Ensure that the [queues are configured](#) to allow job submission from the user under whose account the Materials Studio servers will be installed, as well as from the accounts of all end users.
5. Install Materials Studio as described in [Installing Materials Studio on Windows systems](#) or [Installing Materials Studio on Linux systems](#), as appropriate.
6. To specify additional parameters for the `qsub` command for jobs submitted from the gateway, you can edit the respective configuration files (`pbs.cfg`, `sgc.cfg`, `torque.cfg`, and `lsf.cfg`) in the <MS_INSTALL>/etc/Gateway/root_default/dsd/commands/queues/[*queuing system*] directory, where [*queuing system*] is either PBS, SGE, TORQUE, or LSF. Please take special note of the restrictions described in the PBS configuration file (`pbs.cfg`) regarding PBS Pro, especially for Linux clusters.
7. You must configure the gateway for use with the queuing system. This is best done through the *Remote View* dialog of the Server Console, which can be accessed by selecting *Accelrys | Materials Studio 8.0 Server Console* from the list of programs on *Windows Start* menu on the Materials Studio client machine. Follow the instructions provided in the "Queuing system support" topic in the help system of the Server Console application.

Note: The Server Console has its own help system that is separate from the Materials Studio help.

8. **LSF under Windows only:** LSF cannot submit jobs from a service that is running under the *System* account. By default, the *Materials Studio Gateway* service runs under the *System* account. You should [change the gateway service](#) to the account under which Materials Studio was installed to enable LSF on Windows.
9. **PBS Pro under Windows only:** PBS cannot submit jobs from a service that is running under the *System* account. By default, the *Materials Studio Gateway* service runs under the *System* account. You should change the gateway service to logon using the account under which PBS Pro was installed to enable PBS Pro on Windows.

Note: Whenever you modify the configuration of the queuing system (for example adding or deleting queues), you should ensure that the Materials Studio gateway is stopped and restarted, otherwise there is no guarantee that the changes to the configuration will be picked up by the gateway. You can do this from the command line. Any users accessing the reconfigured gateway from the Windows client will have to perform a *Refresh Gateway Data* operation from the Server Console after any such configuration changes.

Configuring a queuing system from the command line

1. Ensure that the [supported](#) queuing system that you intend to use is installed and accessible to the gateway owner.
2. From the command line, navigate to the `etc/Gateway` directory and run the `configure` script:
Windows:
`config\configure queue <queuename>`
Where `<queuename>` is PBS or LSF.
Linux:
`config/configure queue <queuename>`
Where `<queuename>` is PBS, LSF, TORQUE, or SGE.
3. Start the gateway HTTP daemon. From the `etc/Gateway` directory enter:
Windows:
`gwstartservice`
This automatically configures the gateway and the servers, installs the gateway service, and starts the gateway.
Other useful commands are:
 - `gwstopservice` to stop the gateway, in order to add or remove queues, you must then restart the service.
 - `gwtestservice` to test the server functionality.
 - `gwuninstallservice` to uninstall the service.While the service is installed it will start automatically when the machine is restarted. To prevent the service starting it should be uninstalled.
Linux:
`./gwstart`
Other useful commands are:
 - `./gwtest` to test the server functionality.
 - `./gwstop` to stop the gateway, in order to add or remove queues, you must then restart the service.

Using queues on clusters

This version of Materials Studio supports queuing in conjunction with PBS Pro, OGE, UGE, LSF, and TORQUE on homogeneous Linux clusters only. Other configurations are only supported for local execution on the machine on which the gateway is installed.

To enable queuing on a Linux cluster with PBS Pro, OGE, UGE, TORQUE, or LSF installed

1. Follow the installation instructions provided with your queuing system to ensure that it is started automatically after a system restart.
2. Log in under the account where you are going to perform the installation of the Materials Studio servers and verify that the path to the queuing commands is listed in the PATH environment variable.
3. Confirm that the queuing system is properly configured and active; for example, verify that queuing system commands such as `qstat` are functioning correctly and that the appropriate daemons are running.
4. Ensure that the [queues are configured](#) to allow job submission from the user under whose account the Materials Studio servers will be installed, as well as from the accounts of all end users.
5. Install Materials Studio on your Linux cluster, carefully following the [instructions](#) provided.
6. To specify additional parameters for the `qsub` command for jobs submitted from the gateway, you can edit the respective configuration files (`pbs.cfg`, `sge.cfg`, `torque.cfg`, or `lsf.cfg`) in the `<MS_INSTALL>/etc/Gateway/root_default/dsd/commands/queues/[queuing system]` directory, where `[queuing system]` is either PBS, SGE, TORQUE, or LSF. Please take special note of the restrictions described in the PBS configuration file (`pbs.cfg`) regarding PBS Pro 10.4.
7. You must now configure the gateway for use with the queuing system. This is best done through the *Remote View* dialog of the Server Console, which can be accessed by selecting *Tools | Server Console* from the Windows *Start* menu on the Materials Studio client machine. Follow the instructions provided in the "Queuing system support" topic in the help system of the Server Console application. Note that the Server Console has its own help system that is separate from the Materials Studio help.
8. When a queuing system is installed on a Linux cluster, the Materials Studio gateway automatically sets the number of cores per CPU parameter, `corespercpu`, to AUTO. The license pack calculates this value for Intel processors; however, the number of cores for AMD processors must be [set manually](#). If the gateway is subsequently reconfigured from the shell via the `configure` script, you will need to reset this value.
9. The total number of cores available (across all processors and nodes) parameter, `cpucoresotal`, is calculated by the Materials Studio gateway based on the head node's resources *only*. You may need to [adjust this](#) to match the number of clustered CPU cores you wish to use. These nodes should also be explicitly named in the [machines.LINUX](#) file.

Queue permissions

All jobs submitted to a queuing system by the gateway are run as the user who installed the gateway, so it is necessary for this user to have permission to run jobs on any queue that is exposed through the gateway. However, Materials Studio allows fine-grained control over queue access by respecting the queuing system authorization for the user submitting the job. On password-protected gateways, the user name used to determine queue permissions is the name entered when logging on to the gateway. On non-protected gateways, the user name used is that of the Windows user submitting the job.

The methods used by the gateway to determine which queues are accessible to the current user are listed below. Note that where group membership is checked, it is the groups that the user of that name on the gateway server machine belongs to that are checked, not those of the Windows user on the client PC. For details on authorizing users for particular queues, refer to the documentation supplied with your queuing system.

Note: The list of authorized queues is calculated only when a new gateway is added or an existing gateway is refreshed using the Server Console.

Queuing system	Method
PBS Pro	The command <code>qstat -Qf</code> is run as the gateway owner to determine the full list of available queues. For each queue in that list, the queue configuration attributes <code>acl_group_enable</code> , <code>acl_groups</code> , <code>acl_user_enable</code> , and <code>acl_users</code> are read and used to determine whether the user has access to each queue.
LSF	The command <code>bqueues -w -u <user></code> is run as the gateway owner to determine the list of queues that <code><user></code> is permitted to access.
OGE UGE	The command <code>qconf -sql</code> is run as the gateway owner to determine the full list of available queues.
TORQUE	The command <code>qstat -Qf</code> is run as the gateway owner to determine the full list of available queues. For each queue in that list, the queue configuration attributes <code>acl_group_enable</code> , <code>acl_groups</code> , <code>acl_user_enable</code> , and <code>acl_users</code> are read and used to determine whether the user has access to each queue.

Troubleshooting

Due to the wide variety of different operating systems, queuing systems, and cluster configurations, it is difficult to provide instructions on how to get the Materials Studio gateway to communicate with a queuing system in every conceivable setup. In addition, some queuing systems may have their own technical weaknesses, which are discussed in pertinent news groups and mailing lists. Here is a short list of useful hints and tips that may help you resolve any difficulties you encounter.

Problem	Possible corrective action
The Gateway Data page, accessible through the <i>Remote View</i> dialog of the Server Console, shows incorrect/outdated information.	<ul style="list-style-type: none"> ■ Log in under the account where the servers are installed and verify that the path to the queuing commands is listed in the PATH environment variable. ■ Verify that the queuing system daemons (under Linux) or services (under Windows) are running. If not, you may wish to ensure that the queuing system is restarted automatically on reboot (see the installation instructions provided with your queuing system for further details). ■ Stop and restart the gateway.
I have installed a queuing system on a cluster or parallel machine with MPI enabled and the queues are available in the <i>Queue</i> dropdown list on the <i>Job Control</i> tab of a module's Calculation dialog. However, the Materials Studio client does not permit me to submit parallel jobs to more than one CPU.	Set the number of CPUs available on the machine manually by following the instructions provided in the "Editing gateway data" topic.

Problem	Possible corrective action
<p>The Materials Studio gateway appears to recognize the queuing system and I am able to specify a queue on the Materials Studio interface; however, any job that I submit immediately fails with either no output or a queuing error.</p>	<p>Log in under the account where the servers are installed. Try executing queuing commands from the command line, for example <code>qstat</code>. If these are not recognized:</p> <ul style="list-style-type: none"> ■ Follow step 2 above to ensure that the PATH environment variable is set correctly in the user's shell resource file. ■ Verify that the queuing system daemons (under Linux) or services (under Windows) are running. If not, you may wish to ensure that the queuing system is restarted automatically on reboot (see the installation instructions provided with your queuing system for further details).
<p>My LSF installation under Windows does not allow me to submit jobs.</p>	<p>Follow the instructions in step 9 above to modify the gateway service to run under a different account.</p>
<p>I have added/removed a queue/user from the queuing system, but my change is not reflected on the Gateway Data page or the Materials Studio client.</p>	<ul style="list-style-type: none"> ■ Stop and restart the Materials Studio gateway. ■ Ensure that all users accessing the gateway refresh the gateway information by selecting <i>Action Refresh Gateway Data</i> from the menu bar of the Server Console.
<p>The <i>Queue</i> dropdown list on the <i>Job Control</i> tab of a module's Calculation dialog does not contain any options. After submission of a job, I get the error There are no queues accessible to the user [user name].</p>	<ul style="list-style-type: none"> ■ The gateway may have been switched to use a queuing system, but the client machine still has outdated information. Whenever the gateway properties are modified, you must perform a <i>Refresh Gateway Data</i> operation from the Server Console on the client machine before accessing the reconfigured gateway. ■ Alternatively, check with your system administrator whether you have access permissions to any of the queues on the server machine. If not, ask to be granted access to the queues.

Problem	Possible corrective action
<p>A queued job terminates with failure, but the error message in the job log window provides little information, for example ERROR15.</p>	<ul style="list-style-type: none"> ■ Check the help topic "If a remote <i>[module]</i> job fails" in the Materials Studio Online help, where <i>[module]</i> is the name of the Materials Studio server code you tried to run. The problem may be unrelated to queuing and may be caused by a server failure instead. ■ Check the project log, the main job output file (for example <i>[seedname]</i>.castep for CASTEP), or the queuing system log for additional information. To access the queuing system log, you may have to log on to the server machine with administrator privileges. ■ A common failure condition with queuing systems is that a time or resource limit that was specified for the queue has been exceeded. This can result in the server job being terminated prematurely. In such cases, the gateway does not always provide an informative message.
<p>After I launch a job, the job status shown in the job log window and Job Explorer switches between running and queued several times during job execution.</p>	<p>This is expected behavior and occurs because these jobs launch several executables as part of a protocol.</p>
<p>I have an unsecured gateway accessed by many different users. I do not need to give each of these users different permissions on the gateway and wish to avoid having to create a user on the Linux gateway machine for each client user.</p>	<ul style="list-style-type: none"> ■ Consider removing security restrictions from your queues. ■ Consider making the gateway password protected with a single authorized user. It would then only be necessary to enable queue authorization for that one user.
<p>Queued jobs on the gateway return with an error if there are insufficient licenses available to run the specified server.</p>	<p>Consider enabling the license waiting functionality of the Accelrys License Pack as detailed in the Editing gateway data topic.</p>

Removing Materials Studio

Refer to the appropriate section of this topic for instructions on how to remove Materials Studio from your particular system.

Windows systems

To remove Materials Studio from a Windows system

1. Uninstalling Materials Studio may require that your machine be restarted. In order to ensure that you do not lose any valuable information, it is recommended that you close all other Windows applications, and temporarily disable antivirus software.
2. Stop any server jobs that are running, either locally or that have been started on servers remote from the Materials Visualizer that you may be uninstalling. You can use the Server Console to check whether any jobs are still running. Details of, and controls to stop, any jobs that are running on a particular server gateway can be found under each of the server nodes of the server gateway that you intend to remove. Available server gateways are listed under the *Console Root > Server Management > Server Gateways* node of the Server Console.
3. Close Materials Studio and any server consoles that are running.
4. Select *Settings | Control Panel | Add or Remove Programs* from the Windows Start menu. For Windows select *Control Panel | Programs | Programs and Features*.
5. Choose **Accelrys Materials Studio 8.0** from the list of programs and then click the *Remove* or the *Uninstall* button.
6. The Materials Studio uninstaller will start. Follow the instructions displayed on-screen to complete the removal process.
7. If a system restart is required, you will be prompted to do so at the end of the removal process. The uninstallation will not be fully completed until you have restarted your system.
8. Uninstallation of the Accelrys License Pack is permitted if no other Accelrys software is installed. Refer to the Accelrys License Pack documentation for detailed instructions.

Note: Although the DHTML editor is automatically installed with Materials Studio on Windows systems, it is not automatically uninstalled. If you wish to remove this component as well, you should do it manually using the *Uninstall a Program* option available through the *Control Panel*.

Linux systems

To remove Materials Studio servers from a Linux system

1. Log in using an account with administrative privileges.
2. Ensure that all server jobs running under this installation are stopped. You can use the Server Console from a Materials Visualizer installation under Microsoft Windows to check whether any jobs are still running. Details of, and controls to stop, any jobs that are running on a particular server gateway can be found under each of the server nodes of the server gateway that you intend to remove. Available server gateways are listed under the *Console Root > Server Management > Server Gateways* node of the Server Console.
3. Stop the gateway daemon.
Change directory to the Gateway folder and run:

```
./msgateway_control_<portnumber> stop
```
4. If your system is running Red Hat Linux and you have configured your gateway to start automatically, then:

- Use `chkconfig` to remove all the links to `/etc/rc.d/init.d/msgateway_control_<portnumber>` using the command:

```
/sbin/chkconfig --del msgateway_control_<portnumber>
```

- Remove the `/etc/rc.d/init.d/msgateway_control_<portnumber>` file using the command:

```
rm /etc/rc.d/init.d/msgateway_control_<portnumber>
```

5. If your system is running SuSE Linux and you have configured your gateway to start automatically, then:

- Use `chkconfig` to remove all the links to `/etc/init.d/msgateway_control_<portnumber>` using the command:

```
/sbin/chkconfig --del msgateway_control_<portnumber>
```

- Remove the `/etc/init.d/msgateway_control_<portnumber>` file using the command:

```
rm /etc/init.d/msgateway_control_<portnumber>
```

6. Remove all gateway configurations by running the `gwdeconfigureall` tool:

- `cd` into `etc/Gateway` at the top level of your Materials Studio installation and run:

```
./gwdeconfigureall
```

7. Check whether it is an RPM installation of Materials Studio that you intend to remove, as root enter:

```
rpm -qa | grep MaterialsStudio
```

If this is the case the name and version of the Materials Studio RPMs will be returned, use this command to remove the installation:

```
rpm -e MaterialsStudio MaterialsStudioGateway
```

8. Delete the top-level directory and all the subdirectories into which you installed Materials Studio using the command:

```
rm -rf <install directory>
```

Where `<install directory>` is the location of the Materials Studio installation.

9. Uninstallation of the Accelrys License Pack is permitted if no other Accelrys software is installed. Refer to the Accelrys License Pack documentation for detailed instructions.

Gateway administration

Gateway administration consists of a diverse set of tasks performed by the gateway administrator (or gatekeeper), who is normally the person under whose account the installation resides.

Note: On a 64-bit Windows system, the gateway directory is found in the `Materials Studio 8.0 x64 Server` directory. While the gateway scripts are also present in the 32-bit `Materials Studio 8.0` installation, these are for internal use by the Materials Studio client only and cannot be used to configure a Gateway service.

The gateway administrator can modify the default gateway configuration according to the requirements of a particular installation. Configuration changes can include:

- Changing [gateway security](#) settings
- Registering a gateway
- [Configuring](#) a gateway
- Editing gateway [data settings](#)
 - Specifying whether or not job info files are [archived](#) when a job is deleted
 - Making an installed [queuing system](#) accessible to gateway users
 - Changing the [total](#) number of CPU cores
 - Changing parameters used by the [Accelrys License Pack](#), including the number of cores per CPU (used by multicore licensing) and license waiting override parameters
 - Changing parameters for [individual server codes](#) (for example the polling interval)
- Archiving gateway [job details](#)

There are also certain routine administration duties that must be performed by the gatekeeper:

- Managing gateway resources
- Cleaning up abandoned jobs

These tasks require the use of two different interfaces:

- web browser interface
- [scripts interface](#)

Because these access the gateway via different mechanisms, they place different restrictions on who is permitted to administer the gateway.

Server gateway information

During the Materials Studio server installation a gateway is configured to provide access to the installed server programs. The details of gateway management differ somewhat between a [Linux](#) installation and a [Windows](#) installation.

The server gateway is a program that must be running to provide access to installed Materials Studio server programs. It runs on the same machine as the server programs and is installed alongside them.

It is possible to have multiple gateway installations on the same machine. They will use different port numbers so that you can distinguish between them. When installing a second server configuration, ensure that the original gateway is in a running state, so that the install process for the new gateway can detect that it needs to use a different port number.

If you have a gateway installed to a Materials Studio server, you can learn more about server management in the Server Console online help.

Windows

When server programs are installed on a Windows operating system, the server gateway is installed and started up as a *Service*. This means that it is started up again whenever the machine is rebooted, and so you do not need to manually restart the gateway.

To see a complete list of services running on your machine, select *Control Panel* from the Windows Start menu, open the Administration Tools and start the Services utility. This shows you the list of installed services and which ones are currently started. If installed, the gateway is listed as *Materials Studio Gateway* followed by the port number that it is using.

If you wish to manage the gateway in manually, there are a number of options available. If a gateway is not started, the server programs and jobs that it supports are inaccessible.

1. You can stop and start the service in the **Services** utility. Select the gateway service from the list and click the **Stop**, **Start**, or **Restart** button as appropriate. The status column of the Services utility indicates whether the gateway is currently started or not. Remember that if the gateway service is stopped when you switch off your computer, it will remain stopped after rebooting.
2. In the Gateway folder of the Materials Studio installation are a number of scripts configured for your gateway installation. Some of these scripts may be used to manage the gateway manually.
 - **gwstopservice** - Run this script to stop the gateway service. It will remain installed, but not started, so no server jobs can be run.
 - **gwstartservice** - Run this script to start a stopped gateway service and open access to the server programs installed on the gateway. To ensure that the gateway is running, open the Windows Task Manager utility, you should see two Apache .exe items on the *Processes* tab.
 - **gwrestartservice** - Run this script to stop and restart a gateway service. For a stopped service the effect is the same as **gwstartservice**.
 - **gwuninstallservice** - Run this script to completely remove the gateway as a service. You must reinstall it before being able to start the gateway as a service.
 - **gwinstallservice** - Run this script to reinstall an uninstalled gateway as a service. You must then start the service (see **gwstartservice**) to restore access to its installed server programs.
 - **gwtestservice** - Run this script to check the configuration syntax of the running gateway.
 - **gwinfo** - Run this script to obtain a report of the configuration parameters set when the gateway was installed. By default, a gateway is installed to use port number 18888. This can be a useful diagnostic tool to analyze any problems, and you may be asked for this information in the case of a customer support request.

Note: You should not move an installed gateway or its server programs in the file system, or change the path to the installation. During the install procedure, the gateway is configured to operate at its current location. If you do need to move the Materials Studio installation for some reason, then you should uninstall and reinstall it elsewhere.

To access a gateway installation from another computer, you should select *Accelrys | Materials Studio 8.0 Server Console* from the list of programs on the Windows Start menu. Select the *Server Gateways* and choose *New | Server Gateway* from the shortcut or *Action* menu. You need to know the network name of the server host machine and the port that the gateway is using. Use the **gwinfo** tool in the gateway installation to check these. You can run a test job from the Server Console to ensure that communications are working correctly. Select the new gateway and choose *Test Gateway...* from the shortcut or *Action* menu.

Linux

When server programs are installed the gateway is installed and started up as a *Service*. This means that it is started up again whenever the machine is rebooted, and so you do not need to manually restart the gateway. To establish which services are running, use the `ps` command.

If you wish to manage the gateway manually, there are a number of options available. If a gateway is not started, the server programs and jobs that it supports are inaccessible.

In the Gateway folder of the installation there are a number of scripts configured for your gateway installation. Some of these scripts may be used to manage the gateway manually.

- **./gwstop** - Run this script to stop the gateway service. It will remain installed, but not started, so no server jobs can be run.
- **./gwstart** - Run this script to start a stopped gateway service and open access to the server programs installed on the gateway. To ensure that the gateway is running, open the Windows Task Manager utility, you should see two `Apache.exe` items on the *Processes* tab.
- **./gwtest** - Run this script to check the configuration syntax of the running gateway.

Configuring a gateway

The topic is intended for gatekeepers and covers the configuration, running, stopping of the gateway service.

The scripts for managing the gateway are created in the Gateway directory on installation of Materials Studio. Normally, only the person under whose account the Materials Studio installation resides will have permission to run these scripts (on Linux platforms, these are the default permissions). The user name under which the gateway is installed should be used for script-based administration of that gateway. Individual scripts are discussed below.

Note: The server's Gateway directory is typically `<MS_INSTALL>/etc/Gateway`. See [Installing Materials Studio](#) for further details on the `<MS_INSTALL>` location. On 64-bit Windows you must ensure that you configure the gateway in the `Materials Studio 8.0 x64 Server` directory and not the 32-bit client version.

Notes:

On Windows the User Access Control (UAC) settings can affect this interface with the Gateway. The following scripts will result in errors if they are run from the command line with UAC on:

- `gwstopservice.bat`
- `gwstartservice.bat`
- `gwininstallservice.bat`

The first two of these scripts provide functionality which is equally accessible through the Services panel. The third does not.

A workaround exists, as follows:

1. Create a shortcut to `C:\WINDOWS\system32\cmd.exe`.
 2. In the Properties dialog for this shortcut on the Shortcut tab, click the *Advanced...* button.
 3. Select the option to permanently run this shell as administrator.
- Running the gateway scripts interface through command windows invoked in this manner works with UAC enabled.

Configuring the gateway

Server scripts

If the server directory has a `scripts` subdirectory, each file that it contains is copied to the gateway's `dsd/servers` directory, making it available to be invoked by the gateway daemon, according to the `server-executable` setting in the server's `.sbs` file located in `servers-info` folder.

It should normally not be necessary to configure the gateway as this will be done during installation, however manual configuration of the http gateway environment can be done by invoking the `config/configure` script using the argument "gateway":

Linux

```
./config/configure gateway
```

Windows

```
config\configure gateway
```

This modifies various files in the installation, and adds some command-line scripts to this directory.

Tip: Refer to the [Configuring a queuing system from the command line](#) section for information about setting up a gateway to use a supported queuing system.

Port

To configure the gateway to use a different port than the default (18888) run the `config/configure` script with the argument "gateway -port [port number]":

Linux

```
./config/configure gateway -port 18000
```

Windows

```
config\configure gateway -port 18000
```

Where 18000 is an example of a non-default port.

This modifies various files in the installation to use the specified port.

Managing a gateway

Note: It is important to run the commands from a command line. Do not click on the icon in the folder.

Windows

You must run the gateway http daemon as a Windows service.

Open a command line window and navigate to the gateway directory.

- If the gateway has not been configured, run:

```
gwstartservice
```

If necessary, this will automatically configure the gateway and the servers, install the gateway service, and start it.

- To run a simple test of the server functionality, run:

```
gwtestservice
```

- To stop the service, run:

```
gwstopservice
```

- To uninstall the service:

```
gwuninstallservice
```

If the service is not uninstalled it will start automatically when the machine is restarted.

Linux

You must run the gateway http daemon as a service.

Open a command line window and navigate to the gateway directory.

- To start the service, run:

```
./gwstart
```

- To run a simple test of the server functionality, run:

```
./gwtest
```

- To stop the service, run:

```
./gwstop
```

Editing gateway data

The gatekeeper can modify the default gateway configuration according to the requirements of a particular installation. Configuration changes can include:

- Specifying whether or not job info files are [archived](#) when a job is deleted
- Specifying an installed [queuing system](#) accessible to gateway users
- Changing the [total](#) number of CPU cores
- Changing parameters used by the [Accelrys License Pack](#), including the number of cores per CPU (used by multicore licensing) and license waiting override parameters
- Changing parameters for [individual server codes](#) (for example the polling interval)

Changing Gateway settings

Note: You can only edit gateway data if you have [gatekeeper](#) permissions. For further information please see the [gateway security](#) topic.

To view and edit gateway data

1. Open a URL to the gateway, of the form `http://<gateway>:<port>`, for example `http://localhost:18888`.
2. In the remote gateway page displayed in a web browser, click the **Gateway Data** option on the menu. This displays the current gateway data.

MATERIALS STUDIO Remote Gateway el3c10_18888		
Installed Servers Gateway Data Jobs Logs Web Server Info	Gateway Data	
	hostname	el3c10
	port	18888
	ipaddress	172.27.65.22
	gatewayname	el3c10_18888
	pollingfrequency	5000
	archivejobinfo	no
	osname	MSWin32
	osversion	Windows 6.1

3. Click the **Edit** button to open a dialog requesting a password.

Note: By default, the username of the gatekeeper of a gateway is set to `gatekeeper` and there is no password.

4. Enter the username and password of the gatekeeper and click the **OK** button.

MATERIALS STUDIO Remote Gateway el3c10_18888		
Installed Servers Gateway Data Jobs Logs Web Server Info	Gateway Data	
	hostname	el3c10
	port	18888
	ipaddress	172.27.65.22
	gatewayname	el3c10_18888
	pollingfrequency	<input type="text" value="5000"/>
	archivejobinfo	<input type="text" value="no"/>
	osname	MSWin32
	osversion	Windows 6.1

Some of the gateway settings can be edited.

5. Change the settings according to the parameter you wish to alter; this could be the [Queuing system administration](#), [CPU core total](#), [Licensing parameters](#) (such as the [number of cores per CPU](#) and the [license wait behavior](#)), [Server code administration](#), or [MPI command](#).

Note: On Linux you can change some runtime environment variables (such as temporary directory, PATH, library path, and so on). To make such changes click the "Edit runtime environment" link on the gateway settings page.

Changing CPU core total

The Materials Studio gateway communicates the total number of CPU cores available for calculation (for systems containing only single-core CPUs this is the total number of CPUs present) to the client using the `cpucorestotal` parameter, which is set automatically during installation. In certain cases, however, you may wish to override the value set by the installer, for example:

- For installation on [Linux clusters](#), where the installer bases its estimate on the cluster head node *only*.
- For installation on Intel systems featuring Hyper-Threading technology (<http://www.intel.com/technology/platform-technology/hyper-threading/index.htm>). If this is enabled at BIOS level, the installer will make the assumption that you intend to take advantage of it when running Materials Studio servers. For example, a system containing two dual-core CPUs with Hyper-Threading enabled will have an initial `cpucorestotal` value of 8.

To change the total number of CPU cores available to the gateway

- Follow the instructions [above](#) and then:
 5. Enter the appropriate value for the `cpucorestotal` property and click the **Save** button at the bottom of the Gateway Data page.

Changing licensing parameters

With Materials Studio, two features of the Accelrys License Pack are dependent on parameters under control of the gateway administrator:

- Multicore licensing, which depends on the number of cores per CPU as configured by the gateway
- License waiting functionality, which can depend on a gateway setting

Note: The information above may be updated when your Accelrys License Pack is upgraded. See the separate Accelrys License Pack help documentation for additional and up-to-date information.

Changing the number of cores per CPU

Note: Some applications, such as Forcite, ONETEP, and Mesocite, check out a license per job regardless of how many parallel processes are being used. For these applications, the following discussion can be ignored.

Parallel applications running on multicore machines can be configured by setting the environment variable `CORES_PER_CPU` in the runtime environment for the parallel processes. Beginning with License Pack 7.4, the number of cores on Intel chips is automatically detected and used to adjust the number of license copies used by parallel application processes running on machines containing multicore CPUs. If the autodetection fails for any reason, the environment variable is used to determine the number of cores per CPU on the machine. The maximum number of cores per CPU supported is 8. If more cores are detected, then the number of cores per CPU will be set to 8.

Note: For machines with AMD chips the `CORES_PER_CPU` environment variable must be set manually for the Accelrys licensing software to use this information. For Materials Studio users this can be done by setting the `corespercpu` parameter on the gateway.

For multi-core machines the following *formulas* will be applied to the number of licenses checked out for a job:

Floating licenses:

$\text{ROUNDUP}(\text{number of processes} / \text{number of cores per processor})$

Token licenses:

$\text{INT}[\sqrt{\text{number of processes}/\text{number of cores per processor}} * \text{token weight}]$

Note: The number of token licenses will never be scaled below the token value.

For example, if CASTEP, with token weight of 4, is executed in parallel on 8 processors (each with 2 cores), and the total number of processes used is 16, the number of tokens used is $\text{int}[\sqrt{(16/2)} * 4] = 11$. The number of floating licenses used is $\text{roundup}(16/2) = 8$.

Here is a sample table for a product whose token weight is 4, such as CASTEP.

Number of processes	Floating Copies Single Core	Floating Copies Dual Core	Tokens Single Core	Tokens Dual Core
1	1	1	4	4
2	2	1	5	4
3	3	2	6	4
4	4	2	8	5
5	5	3	8	6
6	6	3	9	6
7	7	4	10	7
8	8	4	11	8
16	16	8	16	11
32	32	16	22	16

To change the number of cores per CPU on the gateway

■ Follow the instructions [above](#) and then:

5. Enter the appropriate value for the **corespercpu** property and click the **Save** button at the bottom of the Gateway Data page.

Changing the license wait behavior

The Accelrys License Pack has been modified to support the addition of wait functionality for license acquisition. This will cause the licensing call in the relevant server executable to wait until a license becomes available, as long as there are sufficient licenses in the overall pool.

Note: Alternative schemes, such as causing a job to wait in a queue until the licenses become available, are not supported.

Wait functionality has been introduced through the concept of an interval (an integer value representing the waiting time in minutes) that can be associated with each checkout attempt. To allow additional features and ensure backward compatibility, the following semantics are associated with the integer value:

Waiting interval value	Wait behavior
< 0	Wait indefinitely until the requisite licenses become available.

Waiting interval value	Wait behavior
0	Do not wait. This is the default setting to ensure backward compatibility.
> 0	Wait for the specified number of minutes, checking periodically. If the requisite licenses cannot be acquired before the interval expires, the checkout call will return an error.

The wait interval is represented by the gateway `licensewaitoverride` property. The default value for this property is `undefined`, which means that any setting that is present in the license policy will be in operation (no override will take place).

To change the license wait property

- Follow the instructions [above](#) and then:
 5. Enter an appropriate value for the `licensewaitoverride` property, for example:
 - `-1` : the License Pack will wait indefinitely
 - `10` : the License Pack will wait for 10 minutes
 6. Click the **Save** button at the bottom of the Gateway Data page.

Server code administration

Individual server codes may be reconfigured to a small extent. The most common configuration change is to alter the `MinimumPollingInterval` property. This may help resolve issues with large numbers of servers all trying to poll the gateway at very short intervals, which is known to cause gateway communication problems.

To change the minimum polling interval property for a server code

- Follow the instructions [above](#) and then:
 5. Enter an appropriate value, in milliseconds, for the `pollingfrequency` property and click the **Save** button at the bottom of the Gateway Data page.

MPI command

To alter the way in which Intel MPI manages parallel jobs the `mpicommand` can be edited, care should be taken when doing so (this may be necessary when [Enabling ssh](#)).

To change the MPI options

- Follow the instructions [above](#) and then:
 5. Enter an appropriate string for the `mpicommand` property and click the **Save** button at the bottom of the Gateway Data page.

Note: For further information on MPI settings, please refer to the Intel MPI Library documentation (<http://www.intel.com/go/mpir>).

Archiving gateway job details

By default, when a job is deleted on a gateway, all of its details are deleted along with the job files. Optionally, the gateway administrator can configure the gateway so that the job information is archived when the job is deleted. The job details include the name of the gateway user that submitted the job, the job id, and either the process ids or the queue name and queued job id used to run the job (depending on whether or not the job was run through a queuing system).

Job information archiving

When job information archiving is enabled, then, on deletion of a job (whether automatically on job completion or manually from the Materials Visualizer or the web browser interface), the job information file (<job_id>.sbj) is moved to the %ALLUSERSPROFILE%\Accelrys\Materials Studio\8.0\Gateway-[i686 or x64]\root_default\dsd\jobs-info-archive directory (where %ALLUSERSPROFILE% is usually C:\ProgramData on Windows). The format of the job information file is described [below](#).

To enable job information archiving

1. From the command line, change directory to the Gateway folder of the Materials Studio installation.
2. Enable gateway job information archiving (there is no need to stop the gateway before changing this setting) by entering:
`gwarchivejobinfo -enable`

To disable job information archiving

1. From the command line, change directory to the Gateway folder of the Materials Studio installation.
2. Disable gateway job information archiving (there is no need to stop the gateway before changing this setting) by entering:
`gwarchivejobinfo -disable`

To view the current job information archiving setting

1. From the command line, change directory to the Gateway folder of the Materials Studio installation.
2. Enter the following (there is no need to stop the gateway to query the current value of this setting):
`gwarchivejobinfo -show`

Using the web browser interface

The job information archiving setting can also be viewed or altered using the *archivejobinfo* setting on the Gateway Data page of the web browser interface.

To view and edit job information archiving settings via the web browser interface

1. Enter the URL to the gateway, `http://<gateway>:<port>` (for example, `http://numbercruncher.Accelrys.com:18888`), directly into a web browser. Alternately, select *Accelrys | Materials Studio 8.0 Server Console* from the list of programs on Windows Start menu to display the Server Console. Select the gateway from the *Console Root > Server Management > Server Gateways* node in the tree view, then select *Remote View* from the *View* menu.
2. Click on *Gateway Data* in the navigation bar at the side of the remote view to display the Gateway Data page and click the *Edit* button at the bottom of the page.
3. Enter the gatekeeper login details at the prompt.

Note: By default, the user name of the gatekeeper of a gateway is set to *gatekeeper* and there is no password.

4. The current job information archiving setting is given by the *archivejobinfo* property. Adjust the setting by selecting the appropriate option from the dropdown list and click the *Save* button at the bottom of the Gateway Data page.

Job information file format

The job information file for a job is named `<job id>.sbj` and takes the form of a text file where each line has the format:

```
<property name>=<property value>
```

The same property name may appear multiple times in the same file. Depending upon the property in question, multiple values may be considered to represent successive values of the property over time or multiple values of that property. Some of the most significant properties are listed below:

Property name	Multiple	Description
job-uid	No	The user name of the job submitter. For password-protected gateways, this will be the login name used to connect to the gateway. Otherwise, it will be the current operating system user name on the client machine.
job-authenticated-uid	No	The authenticated gateway user name of the job submitter. For password-protected gateways, this will be the same as <code>job-uid</code> , otherwise this property will not be present.
job-primary-processid	No	The id of the process running the Perl script that launches and monitors the gateway job. For non-queued gateways, this should be the parent process of all processes associated with the job.
job-processid	Yes	There is an entry for each process associated with the job.
queued-job-queue	No	The name of the queue to which the job was submitted. This property is only present for jobs on gateways with a queuing system.
queued-job-id	No	The id of the queued job in the queuing system. This property is only present for jobs on gateways with a queuing system.
queued-job-name	No	The name of the queued job within the queuing system. This will be a string of the form <code>MS_<gateway_job_id></code> . This property is only present for jobs on gateways with a queuing system.
job-create	No	The creation time for the job as the number of seconds since 00:00:00 UTC, January 1, 1970.
job-start	No	The start time for the job as the number of seconds since 00:00:00 UTC, January 1, 1970.

Gateway queuing

The Materials Studio gateway can be configured so that all jobs submitted to this gateway are fed into a queuing system instead of being launched instantaneously. This allows you to optimize the use of server resources on your network. For further information on installing and configuring queuing systems, please refer to the [Installing Materials Studio on a server with a queuing system](#) help topic.

Jobs submitted to a gateway that is running a queuing system must be submitted to a queue. Some queuing systems allow the creation of multiple queues with different attributes. You can choose any queue from the list of available queues.

Notes:

Third party queuing systems do not check the availability of licenses before attempting to execute jobs from their queue. As a result, if a queue is configured to execute more jobs than there are licenses available, jobs may fail unexpectedly. There are two ways of overcoming this problem:

- Configure your queuing system to execute no more queued jobs than you have licenses. When submitting jobs of different types with different numbers of available licenses, create individual queues for each job type.
- Turn on the license waiting functionality in the Materials Studio gateway, which will cause the call from the server executable to the Accelrys License Pack to be blocked until the requisite licenses become available (provided the number of licenses requested does not exceed the maximum number available). See the [Gateway administration](#) and [Accelrys License Pack](#) documentation for details of how to enable license waiting.

If you do not wish to implement either of the options above, you should take extra care when managing job submission in order to prevent unexpected job failures.

Tip: A queue that allows unrestricted simultaneous jobs can be used to run a high priority job immediately.

To enable/disable the queuing mode of the gateway

You must have gatekeeper user permissions to carry out this operation.

1. Make sure that a supported queuing system is [installed](#) on the server machine and is running.
2. Log in as the Queue Administrator of the system where the gateway is installed. Create and configure queues and add necessary user and group permissions if required. Refer to your queuing system's documentation for details of queue administration tasks.
3. Open the Remote View of the gateway and navigate to the **Gateway Data** page. Click the **Edit** button.
4. The **queuingssystem** dropdown list shows entries for all the queuing systems detected by the gateway.
5. Select the appropriate queuing system, for example [IBM Platform LSF 8.3](#), for the gateway. Select [\[none\]](#) if you want to run jobs directly, bypassing the queuing mechanism.
6. Click the **Save** button.
7. Click the [queuingssystem](#) link to edit the queuing system configuration file.
8. Click the **Save** button.

Tip: The CheckJobStatusEvery setting can be used to control how frequently the status of the job is checked. This can be set or increased (the default and minimum is 120 s) when there is a high load of requests or if the queuing system shows signs of instability.

Note: To create appropriate user permissions for the queuing system, please note that on password-protected gateways, the username is the name entered when connecting to the gateway. On gateways that are not password-protected, the username is the username of the user currently logged on to the Windows operating system.

Materials Studio users must refresh gateway data in order to see changes effected by a queue modification. After refreshing, all of a user's available queues will appear on the Gateway Property Queue page.

To set the correct number of CPU cores for Linux clusters with PBS, OGE, UGE, or TORQUE

You must have gatekeeper user permissions to carry out this operation.

1. Open the Remote View of the gateway and navigate to the Gateway Data page.
2. Click the **Edit** button.
3. In the **cpucorestotal** property, enter the total number of CPU cores available on your cluster.
4. Click the **Save** button.
5. Close Materials Studio, all browser windows, and the Server Console.

See [Installing Materials Studio on a gateway with a queuing system](#) for further details of queuing system administration, configuration, installation, and troubleshooting.

Gateway security

The installer of the server gateway is responsible for determining the level of security appropriate to the installation. The user name under which the gateway is installed should also be used for administration of that gateway.

The security of the gateway has several different aspects, most of which can be controlled independently. This gives the administrator a wide range of choices about the overall level of security.

Authentication and authorization

Two levels of authentication are supported by the Materials Studio gateway:

- **Minimum** authentication supports user authentication for the gatekeeper user only. This implies a level of authorization which allows all users to view the results of any user.
- **Password** authentication supports the authentication and authorization of all users accessing the gateway, based on a list of registered user names and passwords. This level of authorization restricts users such that they can only view their own results.

Note: The authentication level of a gateway can be changed at any time after installation. You must stop the gateway before adjusting the authentication level and then restart it after making the changes.

Access restrictions

The gateway administrator can set access control parameters to restrict gateway access to a defined set of client PCs, if required. This is independent of any choice regarding authentication/authorization or transport.

Transport protocol

The gateway administrator can set the transport protocol of the gateway to HTTPS (HTTP over SSL), if required. This is independent of any choice regarding authentication/authorization or access.

Note: The transport setting of a gateway can be changed at any time after installation. You must stop the gateway before adjusting the transport setting and then restart it after making the changes.

Gateway default security and enhancements

The gatekeeper should have an understanding of the security of a gateway which is using the default settings and the various enhancements to the security that are available.

Default security settings

Gateway security is by no means mandatory. The gateway and its server programs will run whether or not security has been explicitly set and for many installations, a lack of explicit security may be appropriate.

If you choose to install a gateway and take no steps to configure extra security options, the following points should be noted:

- When the gateway is running, it may be accessed from any client machine that has access to the server machine where the gateway is installed. In most cases, a network firewall will limit such access, but where there is no firewall, this may be a concern.
- Anyone working on a client PC that has access to the server machine where the gateway is installed can run jobs on the gateway from an installation of the Materials Studio Visualizer.
- Anyone working on a client PC that has access to the server machine where the gateway is installed can browse the gateway web pages, view the content of jobs that currently exist on the gateway, and remove completed jobs.

An administrative user, the *gatekeeper*, is able to perform administrative tasks via the gateway's web pages. The following tasks require gatekeeper privileges and may be performed when browsing the gateway web pages:

- Editing the server gateway data settings
- Editing a server program's data settings
- Clearing the contents of one of the gateway log files
- Accessing all the jobs on a gateway, regardless of the gateway authentication level
- Removing completed jobs on a gateway, regardless of the gateway authentication level

As the gatekeeper user, by default, has no password, this gives a potentially large number of users access to a wide range of ways in which to modify gateway behavior and data.

Security enhancements

On a secure network in a trusted environment, the issues discussed above may not be considered significant. However, if you wish to upgrade the security level of the gateway, there is a graduated list of options you can implement, dependent upon the issues of concern to you (the administrator) and to the users of the gateway. These steps are all relatively simple and the administrative tasks should not be much of a burden at any level of security.

Add a password for the gatekeeper user

Adding a password for the gatekeeper is a simple way to prevent unauthorized or accidental changes being made to the gateway settings. As you add more levels of security, it also becomes important that gatekeeper privilege is protected from abuse by an unauthorized user who could run jobs or interfere with other users' jobs. You can change the gatekeeper user's password in the same way as for [any other user](#).

Increase access control restrictions

Access control is a means to limit which client machines can gain access to the gateway for any purpose. This does not restrict which users may connect to the gateway, but it does restrict where they can connect from.

This may be used to limit access only to specific client PCs or to one or more subnets. If the gateway is installed on a network with no firewall, it is *strongly* recommended that you set up some form of [access control](#), whether or not you also choose to introduce any of the further security measures described below.

Impose user authentication

By raising the gateway authentication to the *password* level, all job access and gateway web page browsing is controlled by checks for [user authentication](#). Since the gateway then knows who a user is, authorization is imposed by restricting access to specific jobs that they "own" (i.e., the jobs that they originally created).

This provides a mechanism to protect the data that exists in jobs from unauthorized access and prevents users from removing other users' jobs from the gateway.

Note: The gatekeeper has special privileges to browse all job contents and to remove all completed jobs.

Impose secure transport

By default, HTTP transport means that authentication information (for example, user's passwords) are transmitted in a very lightly encrypted form, which can be easily decrypted. It is possible to remove this security risk by configuring the gateway to use the [HTTPS protocol](#), which securely encrypts all communications.

Note: It is not currently possible to enable secure transport for authentication information only. If HTTPS is selected, then all gateway communication will be encrypted.

Gateway access control

A gateway administrator can restrict access to the gateway so that only certain client machines may run jobs and view results. This may be done on a per-machine basis or on a network basis.

Access control definitions in `allow.txt`

Access control is most simply specified by means of restricting access to machines whose [IP address](#) matches a given [regular expression](#). By default, the file `allow.txt` contains the single line:

```
SetEnvIf Remote_Addr ".*" AllowAccess
```

The IP address of any client will match the regular expression specified (`.*`) and, therefore, is allowed access. The fact that access is allowed is communicated to the Apache server by setting the environment variable named `AllowAccess`. To restrict access, you should modify the regular expression so that only certain client PCs' IP addresses will match it. So, for example, to restrict access to a single client PC, edit `allow.txt` so that it contains something like the following:

```
SetEnvIf Remote_Addr "^190\.101\.43\.21$" AllowAccess
```

Since only one client will have an IP address that matches this specification, only this machine will be allowed to access the gateway. However, such an explicit specification is not suitable for a network which employs a system that assigns dynamic IP addresses. In this case, it would be better to base restrictions at the subnet or site network level, as described below.

To allow access to two IP addresses, use two lines:

```
SetEnvIf Remote_Addr "^190\.101\.43\.21$" AllowAccess
SetEnvIf Remote_Addr "^190\.101\.43\.22$" AllowAccess
```

This could also be expressed as:

```
SetEnvIf Remote_Addr "^190\.101\.43\.(21|22)$" AllowAccess
```

To limit access to all machines on a subnet, the regular expression should be defined so as to only match the first three octets of the IP address:

```
SetEnvIf Remote_Addr "^190\.101\.43\." AllowAccess
```

Matching against only the first two octets could be used to restrict access to a single large site:

```
SetEnvIf Remote_Addr "^190\.101\." AllowAccess
```

Regular expression syntax provides many possibilities for defining a regular expression to match the IP addresses that are allowed access to the gateway. The `AllowAccess` environment variable can also be set based on a value other than `Remote_Addr`. See the Apache website (http://httpd.apache.org/docs/2.2/mod/mod_setenvif.html) for specifics of the `SetEnvIf` directive and the request details that may be used in this context.

To modify gateway access restrictions

1. Log in to the machine where the gateway is installed. Ensure that you are logged in as the owner of the account where the Materials Studio server environment is installed.
2. From the command line, navigate to the `Gateway/configurations/default/conf` folder of the Materials Studio installation.
3. Edit the file `allow.txt` to change the access settings as required.
4. Navigate back to the `Gateway` folder and run the command `./gwrestart` on Linux systems or `gwrestartservice` on Windows. A message will be displayed when the gateway has restarted successfully.

IP addresses

An IP address is a unique sequence of four 8 bit numbers (octets), identifying a computer connected to the Internet. IP addresses are used by Internet protocols such as FTP and HTTP. To find out the IP address of your PC, run the command `ipconfig` from the command line. Look for the output line marked [IP Address](#).

Regular expression syntax

A regular expression specifies a set of strings of characters. A member of this set of strings is said to be 'matched' by the regular expression.

The access control examples above make use of the following syntax:

- Use `.` to match any character.
- Use `*` to repeat the previous expression 0 to n times. So `.*` will match any character any number of times.
- Use `^` to indicate the start of the string to be matched.
- Use `$` to indicate the end of the string to be matched.
- Use `|` to indicate match alternatives and `()` to bound the list. So `(1|2|3)` will match with any one of the numbers 1, 2, or 3.
- Use `\` in front of a character to indicate a literal value that has some special meaning in the syntax. So `^190\.101$` only matches with the string 190.101. Without the backslash, it would match with a string containing any character in the middle, for example 190x101.

Gateway security via user authentication

When a gateway is set at the password level of authentication, access to the gateway and its jobs is restricted to a set of registered users. This includes the ability to create jobs, see the results of a job, kill a job, or remove a job from the gateway. Additionally, access to a particular job is restricted to the user who created that job.

Effects of user authentication

Setting Materials Studio gateway authentication level to [password](#) has a number of specific effects for users who run jobs on that gateway:

- The first time during a Materials Studio session that a user connects to a gateway to create a job or reconnect to an existing job, a standard login prompt will be displayed. To complete the operation, the user name and password of a user registered on the gateway must be entered.
- When browsing the gateway web pages, users will be prompted for a user name and password to gain access to any page other than the home page for the gateway.
- When viewing the list of jobs for a gateway, users will only be allowed to access job lists, job properties, and job files for those jobs created by the logged-in user.
- Only the creator of a completed job will be allowed to remove it from the gateway.

Note: If a [queuing system](#) is installed on a gateway, user authentication will also be required whenever a user registers or refreshes the gateway from the Server Console. See the Server Console help system for details.

The gateway administrator is responsible for setting the gateway's authentication level and for the registration and deregistration of users. When a gateway is installed, two commands, named `gwauthentication` and `gwusers` are created in the Gateway subdirectory of the Materials Studio installation. These commands are used by the administrator to manage the authentication aspects of the gateway.

Once a user name and password are registered, the user can change the password via a link in the gateway web pages. If a user forgets their password, the gateway administrator may assign a new password.

Administration level tasks

To set the gateway authentication level for user authentication

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.
2. Stop the gateway by running the command:
 - Linux

```
./gwstop
```
 - Windows

```
gwstopservice
```
3. Change the gateway authentication setting to restrict access to authenticated users only using the command:
 - Linux

```
./gwauthentication password
```
 - Windows

```
gwauthentication password
```

4. Restart the gateway by running the command:

- Linux

```
./gwstart
```

- Windows

```
gwstartservice
```

To remove gateway user authentication

1. Log in to the machine where the gateway is installed. Ensure that you are logged in as the owner of the account where the Materials Studio server environment is installed.

2. From the command line, navigate to the Gateway folder of the Materials Studio installation.

3. Stop the gateway by running the command:

- Linux

```
./gwstop
```

- Windows

```
gwstopservice
```

4. Change the gateway authentication setting to the minimum level using the command:

- Linux

```
./gwauthentication minimum
```

- Windows

```
gwauthentication minimum
```

5. Restart the gateway by running the command:

- Linux

```
./gwstart
```

- Windows

```
gwstartservice
```

To display a list of registered gateway users

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.

2. Use the gwusers command with the list option to view all registered users (there is no need to stop the gateway before performing user management tasks):

- Linux

```
./gwusers /list
```

- Windows

```
gwusers /list
```

To register a gateway user

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.
2. Use the `gwusers` command with the `add` option to register the new user (there is no need to stop the gateway before performing user management tasks):

- Linux

```
./gwusers /add <user name>
```

- Windows

```
gwusers /add <user name>
```

3. When prompted, enter and reconfirm a password for this user.

Note: User names must be unique. You will not be allowed to complete the registration process if the user name you are entering already exists in the user registry for the gateway.

To remove a registered user from gateway authorization

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.
2. Use the `gwusers` command with the `remove` option to delete the specified user from the user registry (there is no need to stop the gateway before performing user management tasks):

- Linux

```
./gwusers /remove <user name>
```

- Windows

```
gwusers /remove <user name>
```

To change the password of a registered user

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.
2. Use the `gwusers` command with the `update` option to change the user's password (there is no need to stop the gateway before performing user management tasks):

- Linux

```
./gwusers /update <user name>
```

- Windows

```
gwusers /update <user name>
```

3. When prompted, enter and then reconfirm the new password for this user.

Note: The gateway administrator does not need to know a user's current password to be able to change it.

User level tasks

To change your own password as a non-administrative user

1. Enter the URL to the gateway, `http://<gateway>:<port>` (for example, `http://numbercruncher.Accelrys.com:18888`), directly into a web browser. Otherwise, select *Accelrys | Materials Studio 8.0 Server Console* from the list of programs on the Windows *Start* menu to display the Server Console. Select the gateway from the *Console Root > Server Management > Server Gateways* node in the tree view, and select *Remote View* from the *View* menu.
2. Click on **Password** to display the *Password for user* page.
3. Enter your password in the **Current password** field and the new one in the **New password** field. Confirm the new password in the **Repeat password** field and click the **Update** button at the bottom of the *Password for user* page.

Gateway transport protocol

The Secure Sockets Layer (SSL) protocol is a cryptographic layer which may be placed between a reliable connection-oriented network layer protocol (for example TCP/IP) and the application protocol layer (for example HTTP). SSL provides for secure communication between clients and servers by allowing mutual authentication, the use of digital signatures for integrity, and encryption for privacy.

One common use of SSL is to secure HTTP communication between a browser and a web server over the Internet. The secure version is mainly plain HTTP over SSL (named HTTPS), but with one major difference: it uses the URL scheme 'https:' rather than 'http:'.

In Materials Studio, SSL is primarily used to secure communication between a gateway client and the gateway server, although it can also be used to verify the identity of the gateway server to prevent "man-in-the-middle" attacks. SSL is not used to identify and authenticate the gateway user, i.e., there is no need to generate private/public key pairs for each user.

However, a private/public key pair must be generated for each gateway server and, for real security, signed by a trusted authority. A self-signed certificate is provided with your Materials Studio installation to help you get started using SSL straight away.

Effects of SSL transport

Setting the gateway transport protocol to `ssl` has a number of specific effects for users of Materials Studio who run jobs on that gateway:

- When configuring Materials Studio to use the gateway, some changes need to be made using the New Server Gateway dialog of the Server Console.
- The first time that a user connects to the gateway, a standard prompt is displayed, asking whether the user accepts the gateway's certificate. To complete the operation, the user must accept the certificate, whereupon it will be installed on the client machine.
- When running jobs, some performance penalties will be encountered for an encrypted gateway compared to a similar unencrypted gateway.

The gateway administrator is responsible for setting the gateway's transport protocol. When a gateway is installed, a command named `gwtransport` is created in the Gateway directory of the Materials Studio installation. This command is used by the administrator to manage the transport protocol of the gateway.

Administration level tasks

To set the gateway transport protocol to HTTPS

1. From the command line, navigate to the Gateway folder of the Materials Studio installation.
2. Stop the gateway by running the command `./gwstop` on Linux systems or `gwstop` on Windows.
3. Change the transport protocol to use SSL with the command:

```
gwtransport sslon
```

4. Restart the gateway by running the command `./gwstart` on Linux systems or `gwstart` on Windows.

To set the gateway transport protocol to HTTP

1. Log in to the machine where the gateway is installed. Ensure that you are logged in as the owner of the account where the Materials Studio server environment is installed.
2. From the command line, navigate to the Gateway folder of the Materials Studio installation.
3. Stop the gateway by running the command `./gwstop` on Linux systems or `gwstop` on Windows.
4. Change the transport protocol to standard HTTP using the command `gwtransport sslloff`.
5. Restart the gateway by running the command `./gwstart` on Linux systems or `gwstart` on Windows.

User level tasks

When a new gateway is set up to use HTTPS, you will be required to change the default settings when adding this new gateway via the Server Console.

To add a new HTTPS gateway with the Server Console

1. Choose *Programs | Accelrys Materials Studio 8.0 | Server Console* from the Windows *Start* menu to display the Server Console.
2. Select the *Console Root > Server Management > Server Gateways* node in the tree view, then select *New | Server Gateway* from the *Action* menu to display the New Server Gateway dialog.
3. Click the *Connection >>* button to display the advanced controls on the dialog.
4. Check the *Use https* checkbox. Configure the other options in the *Https Settings* section of the New Server Gateway dialog according to the nature of the certificate installed on the new gateway.

Note: If you intend to use the certificate provided with your Materials Studio installation, you will need to check all the checkboxes in the *Https Settings* section of the New Server Gateway dialog.

Licensing Materials Studio

Materials Studio is licensed using the Accelrys License Pack. The Accelrys License Pack is a system for managing Accelrys software at your site. Licensing is provided as a service to help you access all the features of the software that your users require and to comply with your software purchase agreement. It is also designed to protect Accelrys software from unauthorized use, which, in turn, helps to minimize the cost of software and feature enhancements.

Since the Accelrys License Pack is used for many other Accelrys products, you can share licenses, including license tokens, with other products, such as Cerius², Insight II, and Discovery Studio.

For a list of the licensed products in this release of Materials Studio, refer to the [Materials Studio license features](#) topic. Each area of functionality requires at least one license feature to access it. To use Materials Studio, follow the instructions [below](#).

Note: <LP_INSTALL> is used to refer to the installation location of Accelrys License Pack. On 32-bit Windows systems this is by default C:\Program Files\Accelrys\LicensePack. On 64-bit Windows systems by default this is C:\Program Files (x86)\Accelrys\LicensePack.

New features in Accelrys License Pack for Materials Studio 8.0

See the What's New topic in the [Accelrys License Pack help documentation](#) for licensing related enhancements.

Offline access

Beginning with Materials Studio 4.3, you can use the Materials Studio Visualizer offline for a limited period of time. Refer to the Accelrys licensing documentation for details about offline licensing.

Automatic multicore detection

Support for the automatic detection of multicore hardware has been added to the licensing software. The gateway is automatically configured to use this new feature. See the "License usage for parallel processing" topic in the [Accelrys License Pack help documentation](#) for details of how license usage is calculated. Consult the [Gateway administration](#) topic in the Materials Studio help system for instructions on how to configure the gateway for multicore hardware when automatic detection is not supported.

Wait functionality

Beginning with Materials Studio 4.1, wait functionality has been added to the gateway console. You now have the option of enabling the wait functionality, which allows server codes to wait until a license becomes available instead of stopping and returning an error. See the [Gateway administration](#) topic in the Materials Studio help system or the "Materials Studio licensing" topic in the [Accelrys License Pack help documentation](#) for details of how to enable license waiting and how it works.

Users with current maintenance

No license installation necessary: If your Accelrys software maintenance is current and you have already installed the corresponding license file, no new license is necessary. You should be able to use the Materials Studio software once it is installed, as long as it is configured to use the license that is already installed. Accelrys recommends that you restart the FLEXlm license manager running on the license server machine after upgrading Accelrys License Pack on your license server to ensure that you

are using the most recent version of the license manager program. Instructions on how to restart the license manager program are provided in the [Accelrys License Pack help documentation](#).

Install a temporary license: If you wish to try new products included in this release, you will need to install a temporary license using the password included with your software or one obtained from BIOVIA Support. Details of how to install a temporary license are provided in the [Installing a temporary license](#) topic.

New users or users adding new products with this release

Install a temporary license: To enable the products included in this release, you will need to install a temporary license using the temporary password shipped with your software or one obtained from BIOVIA Support. If you do not have a password, an auto trial for the Materials Visualizer is automatically started on installation, allowing you to use this functionality for 30 days. See detailed information in the [Materials Visualizer auto trial licensing](#) topic. Note that you will be unable to start additional trials once the 30 day trial period has elapsed. Contact your Accelrys sales representative if you need to carry out an extended trial. To gain access to additional features in Materials Studio, you must have a license file.

Request a license file: To request a license file appropriate for your software purchase, you need to send specific machine information to BIOVIA Support so that a license can be generated for you. For more details, see the [Accelrys License Pack help documentation](#).

Install license file: Once you have received your Accelrys software license file, it needs to be correctly configured. See the [Accelrys License Pack help documentation](#) for further information.

Use a remote license server: PCs and Linux gateways without NFS support must be configured to use a remote license server through a TCP/IP connection. See the [Accelrys License Pack help documentation](#) for more details.

Materials Studio license features

The following are licensed features of Materials Studio:

Functionality	Feature name	Token weight
Materials Visualizer	MS_visualizer	1
Adsorption	MS_adsorption	2
Amorphous Cell	MS_amorphouscell	2
Blends	MS_blends	2
CASTEP	MS_castep	4
CASTEP Parallel	MS_castep_MP	6
NMR CASTEP	MS_castep_nmr	2
NMR CASTEP Parallel	MS_castep_nmr_MP	3
COMPASS	MS_compass	2
COMPASS Parallel	MS_compass_MP	3
Conformers	MS_conformers	2

Functionality	Feature name	Token weight
DFTB+	MS_dftbplus	2
DMol ³ Molecular	MS_dmol	2
DMol ³ Molecular Parallel	MS_dmol_MP	3
DMol ³ Solid State	MS_dsolid	2
	MS_dmol	2
DMol ³ Solid State Parallel	MS_dsolid_MP	3
	MS_dmol_MP	3
Forcite	MS_forcite	1
Forcite Plus	MS_forciteplus	4
Gaussian	MS_gaussian	1
GULP	MS_gulp	2
GULP Parallel	MS_gulp_MP	3
MesoDyn	MS_mesodyn	2
Mesocite	MS_mesocite	2
Morphology	MS_morphology	2
Motif	MS_motif	1
ONETEP	MS_onetep	7
Polymorph	MS_polymorph	7
QMERA	MS_qmera	2
QSAR	MS_qsar	2
	DESC1	1
	GeneticAlgorithm	1
	MS_vampdescriptors	1
QSAR Plus	MS_NNet	2
	MS_dmol3descriptors	2
	MS_qsar	2
	DESC1	1
	GeneticAlgorithm	1
	MS_vampdescriptors	1
Reflex	MS_reflex	1
		1
Reflex Plus	MS_powdersolve	4
	MS_reflex	1
Reflex QPA	MS_reflex_qpa	2
Sorption	MS_sorption	2

Functionality	Feature name	Token weight
Synthia	MS_synthia	1
VAMP	MS_vamp	1
X-Cell	MS_xcell	2

The following user interfaces no longer require license features; however, you must have a purchased product in order to be able use the functionality: CASTEP (MS_castep_ui), DMol³ (MS_dmol_ui), MesoDyn (MS_mesodyn_ui), Morphology (MS_morphology_ui), and VAMP (MS_vamp_ui).

Some Materials Studio products are also enabled by <feature>_MP and <feature>_site licenses, which enables per-job and per-site licensing models. For more details about these licensing models, see the [Licensing Materials Studio](#) help topic.

Materials Visualizer auto trial licensing

If you do not have a license file, auto trial licensing enables temporary access to a limited set of functionality in Materials Studio, notably the Materials Visualizer. Access is granted for a short duration of time, typically 30 days beginning from the day the software is installed.

To view the status of a Materials Visualizer auto trial

1. Select *Accelrys | Licensing | Configure Materials Studio 8.0 Licensing* from the Windows Start menu to start the [Accelrys Licensing Wizard](#).
2. Choose *For more options, start the License Administrator*.
3. Click the *Next >* button to display the [Accelrys License Administrator](#).
4. Click the *Auto Trial Licensing* option from the tree menu.
5. The current status and number of days remaining in your trial is shown on the *Auto Trial Licensing* page. This information is also displayed every time you start the trial installation of Materials Studio.
6. Select *File | Exit* from the menu bar to exit the Accelrys License Administrator.

Tip: Checking the *Show expired* checkbox on the *Auto Trial Licensing* page of the Accelrys License Administrator displays past evaluations of Materials Studio.

Installing a temporary license

To install a temporary license, you must first acquire a password from BIOVIA Support. A password is typically included with the media and can be used to enable the software while you are obtaining a regular license file.

If you have a current Accelrys software maintenance agreement, you do not need to install a new license. You should use the temporary license to evaluate new products and determine whether you would like to add them to your suite of licensed products.

A temporary license can be installed either during the installation of Materials Studio or afterwards.

During Materials Studio installation

When prompted to install a temporary license during the Materials Studio installation process, enter the password you were sent with your software. The temporary license will be installed automatically and you will be able to run the software immediately after the installation process is complete. If you are

given the option to do so, you should start the Accelrys Licensing Wizard at the end of the installation in order to enter the temporary license password.

After Materials Studio installation

The installation procedure for a temporary license varies according to the platform and operating system you are using.

Windows/Linux GUI environment

A [wizard interface](#) is available on Windows and Linux to present the license administration functionality. This easy-to-use wizard is integrated with the Materials Studio installer. It can also be accessed via the Materials Studio options on the Windows *Start* menu and can be started when an application licensing error occurs. The wizard offers a menu of licensing options, guiding you through the licensing configuration process.

To install a temporary license using the Accelrys Licensing Wizard

1. Launch the [Accelrys Licensing Wizard](#) using the appropriate method for your operating system.
2. Choose the *Enter a temporary license password* option from the first page of the Accelrys Licensing Wizard and click *Next >*.
3. Enter the password provided by BIOVIA Support or the password that was shipped with your software in the *Password* field and click *Next >*.
4. The next page of the wizard reports the results of the temporary license installation. Any previously installed temporary licenses are automatically merged with the newly installed temporary license. All expired and duplicate feature lines are removed from the merged file, leaving only the those with the latest expiration dates. The resulting file is named `msidemo.lic` and is saved with the `Licenses` folder of your Accelrys License pack installation.

Linux console mode

To install a temporary license from the command line

1. Execute the following command:
`<MS_INSTALL>/bin/install_temp_license`
2. Follow the on-screen prompts.

Refer to the [Installing Materials Studio](#) topic for information on the location of the `<MS_INSTALL>` directory.

Licensing on Windows

This topic provides details about the contents and configuration of Accelrys License Pack on the Windows platform. Accelrys License Pack is automatically installed when you install Materials Studio. New versions of Accelrys License Pack are also backward compatible with some previous versions of Materials Studio. For more details, see the [Accelrys License Pack help documentation](#).

Manually installing Accelrys License Pack

Accelrys License Pack is usually installed when you install Materials Studio. However, if you need to install it manually, run the `lp_setup.exe` program located in the `License Pack` folder on the Materials Studio media. A wizard will be started to guide you through the process of installing Accelrys License Pack.

Note: If the same version of Accelrys License Pack is already installed, you will not be able to install it again. To re-install Accelrys License Pack, you must first remove it from your system. Follow the instructions given below to remove Accelrys License Pack.

Removing Accelrys License Pack

Accelrys License Pack can be removed using the Windows Add or Remove Programs dialog.

To remove Accelrys License Pack

1. Select *Settings | Control Panel | Add or Remove Programs* from the Windows Start menu to display the Windows Add or Remove Programs dialog.
2. Choose [Accelrys License Pack](#) from the list of programs and then click the *Remove* button.
3. The Accelrys License Pack uninstaller will start. Follow the instructions displayed on screen to complete the removal process.
4. If a system restart is required, you will be prompted to do so at the end of the removal process. The uninstallation will not be fully completed until you have restarted your system.

For complete details of license pack removal, see the Accelrys License Pack help documentation.

Licensing environment

The licensing environment on your PC is recorded in the Windows registry. It includes details of configured license servers, the folder where your license file can be found and the location of remote license servers. All of these settings can be managed using the [Accelrys License Administrator](#) installed with Accelrys License Pack.

Note: The license settings on your PC are applied to everyone who uses your machine. For example, if you have specified which remote license server to use, a colleague who logs on and uses your PC will be able to use the same settings. However, they will be unable to change the settings unless they have administrator privileges on your machine.

Connecting to a license server

When using a floating/token license in a network that contains machines which need to communicate with the license server using a TCP/IP connection, the connection to the license server must be configured. This is most common for Windows clients communicating with a central license server. Detailed instructions on how to set up a license server connection are provided in the [Accelrys License Pack help documentation](#).

Note: Materials Studio does not support the use of multiple non-redundant license servers. Only one connection, local or remote, may be used at any one time. This means that system administration is simplified and keeps administration and support costs to a minimum.

Licensing on UNIX

This topic provides details of the contents and configuration of Accelrys License Pack on UNIX systems. Accelrys License Pack is installed with Materials Studio. New versions of Accelrys License Pack are backward compatible with some previous versions of Materials Studio. For more details, see the [Accelrys License Pack help documentation](#).

Note: The License Pack requires that certain compatibility files are installed; these may differ from the libraries required by Materials Studio. Please consult the License Pack documentation for further information and details of the required compatibility files.

Manually installing Accelrys License Pack

To install Accelrys License Pack independently from any other Accelrys software installation, follow the instructions below.

To manually install Accelrys License Pack

1. From the media, run the `lp_setup` script (assuming the media drive is mounted on `/CDROM`):
`/CDROM/UNIX/License_Pack/lp_setup -console`
2. Follow the prompts to indicate where Accelrys License Pack should be installed. If an existing version of Accelrys License Pack is detected during installation, it will be updated to the new version automatically. If an existing Accelrys License Pack is not found, you should enter the path to any currently installed license pack when prompted.

When all floating and token features are managed using the same license server, only a single Accelrys License Pack and license file are required to manage all of your Accelrys UNIX software installations, provided that they all support NFS access to the common Accelrys License Pack. After you have installed Accelrys License Pack, it may be updated, if necessary, during subsequent software installations. Only the tools are updated; your installed license file and log files are not modified.

Removing Accelrys License Pack

Accelrys License Pack can be removed by running the uninstall program located in the `Uninstaller` directory under `LicensePack`.

To remove Accelrys License Pack

1. Run the uninstall program:
`<LP_INSTALL>/Uninstaller/uninstaller`
2. The Accelrys License Pack uninstaller will start. Follow the instructions displayed on screen to complete the removal process.

For complete details of license pack removal, see the [Accelrys License Pack help documentation](#).

Changing the License Pack location used by Materials Studio

On UNIX, Accelrys recommends using a single common Accelrys License Pack that is installed on a shared file system and used for all Accelrys licensed products. As a result, you may need to reconfigure a UNIX installation to use a license pack different from the one that was configured during installation. Perform the following steps to modify the Accelrys License Pack location.

To change the location of Accelrys License Pack

1. Change directories to the `bin` folder of your Materials Studio installation:
`cd <MS_INSTALL>/bin`
2. Run the following command:
`config_lp_location <LP_INSTALL>`

This command modifies the file `share/license/data/lp_location` to point to the new location. When you next use a Materials Studio server, the Accelrys License Pack from the new location will be used.

Refer to the [Installing Materials Studio](#) topic for information on the location of the <MS_INSTALL> directory.

Note: If you need to reconfigure the Accelrys License Pack path, see the [Accelrys License Pack help documentation](#) for more details.

Connecting to a license server

On UNIX workstations that have only a TCP/IP connection to the license server (i.e., no NFS connection), you must configure the licensing to use the remote license server. Remote license servers have their own installation of Accelrys License Pack because they do not have NFS access to the common network installation. Detailed instructions on how to set up a license server connection are provided in the [Accelrys License Pack help documentation](#).

Note: Materials Studio does not support the use of multiple non-redundant license servers. Only one connection, local or remote, may be used at any one time. This means that system administration is simplified and keeps administration and support costs to a minimum.

Accelrys license administration tools

Running the Accelrys Licensing Wizard

Use the Accelrys Licensing Wizard to configure your Materials Studio licensing. There are a variety of ways in which you can access the Accelrys Licensing Wizard:

- During Materials Studio installation
 - As part of the Materials Studio installation process, you will have the opportunity to run the Accelrys Licensing Wizard to install a temporary license file, connect to a remote license server, or start the [Accelrys License Administrator](#).
- After Materials Studio installation
 - Once you have installed Materials Studio, you can use the Accelrys Licensing Wizard to manage your licensing.
 - **Windows**
Run the Accelrys Licensing Wizard by selecting *Accelrys | Licensing | Configure Materials Studio 8.0 Licensing* from the Windows *Start* menu. On machines with UAC activated right-click and select *Run as administrator* to launch the program.
 - **Linux GUI environment**
Launch the Accelrys Licensing Wizard from the command line by executing the following command:
`<MS_INSTALL>/bin/license_admin`
 - **Linux console environment**
At the command line, execute the following command:
`<MS_INSTALL>/bin/license_admin -cw`
or
`<MS_INSTALL>/bin/license_admin --console-wizard`
- If no license is available when running Materials Studio
 - If you start Materials Studio and neither a license file nor a license server is available, you will be offered the opportunity to run the Accelrys Licensing Wizard from the resulting error dialog.

Refer to the [Installing Materials Studio](#) topic for information on the location of the <MS_INSTALL> directory.

Running the Accelrys License Administrator

The Accelrys License Administrator allows you to perform detailed license administration tasks. There are two ways you can access the Accelrys License Administrator:

- Accessing the Accelrys License Administrator through the Accelrys Licensing Wizard
 - Launch the Accelrys Licensing Wizard as described [above](#). On the first page of the wizard, select *For more options, start the License Administrator*, and click *Next >*.
- Accessing the Accelrys License Administrator directly
 - **Windows**
 - Run the Accelrys License Administrator by selecting *Accelrys | Licensing | License Administrator 7.6.11* from the list of programs on the Windows *Start* menu.
 - **Linux GUI environment**
 - Launch the Accelrys License Administrator from the command line by sourcing <LP_INSTALL>/etc/lp_cshrc, then running lp_admin.

Note: If you do not have a GUI desktop environment on your Linux or UNIX workstation, you can obtain details of the basic command line administration tools and how to access additional help by executing the following command:

```
<MS_INSTALL>/bin/license_admin -cw
```

or

```
<MS_INSTALL>/bin/license_admin --console-wizard
```

Accelrys License Pack help documentation

Accelrys licensing documentation is available on two levels. The most basic procedures required for this product, including installation of Accelrys License Pack, passwords, and installing a temporary license, plus essential information unique to Materials Studio, is provided in the [Licensing Materials Studio](#) topic and subtopics. The second level of Accelrys licensing documentation is comprised of complete and detailed information on the operation of Accelrys License Pack and enables you to customize the licensing at your site. This separate packaging of licensing documentation allows you to update your licensing software and documentation independent of Materials Studio releases. You can access the latest licensing information on the BIOVIA License Pack home page (<http://accelrys.com/resource-center/downloads/updates/license-pack/index.html>).

There are two ways you can access the Accelrys License Pack help documentation:

- Launch the [Accelrys License Administrator](#) using the appropriate method for your operating system, then either click the *Help* button on the user interface, select *Help | Help Topics* from the menu bar, or press the F1 key.
- Locate the `MaterialsStudio_InstallGuide.htm` file on the Materials Studio media, open it in a web browser, and click "Licensing Materials Studio".