

SHELL

Version 12.0



Oasys Ltd
The Software house of Arup

For help and support from OASYS Ltd please contact:

UK

Arup Group Ltd
The Arup Campus
Blythe Gate
Blythe Valley Park
Solihull
West Midlands
B90 8AE
United Kingdom
Tel: +44 (0) 121 213 3399
Fax: +44 (0) 121 213 3302
Email: dyna.support@arup.com
Web: www.oasys-software.com/dyna

China

Arup
39/F-41/F
Huai Hai Plaza
Huai Hai Road (M)
Shanghai
China 200031
Tel: +86 21 3118 8875
Fax: +86 21 3118 8882
Email: china.support@arup.com
Web: www.oasys-software.com/dyna/cn

India

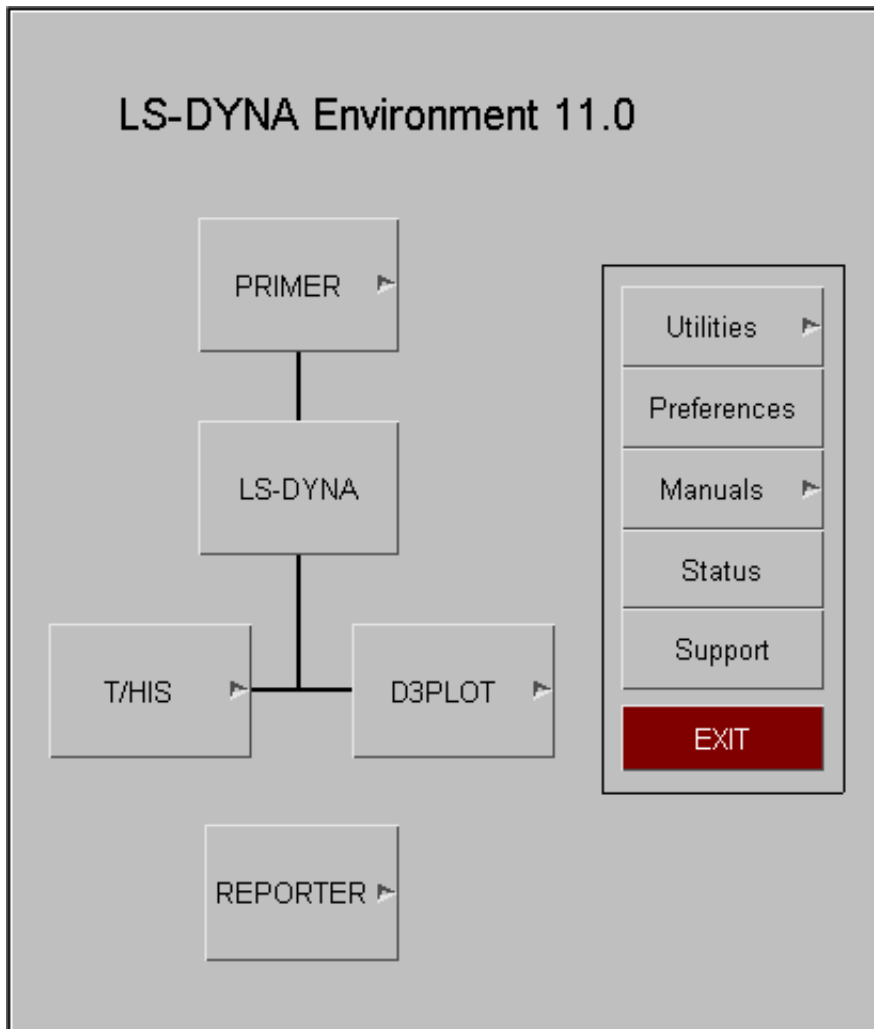
Arup
Plot 39, Ananth Info Park
Opp. Oracle Campus
HiTec City
Madhapur Phase II
Hyderabad 500081
India
Tel: +91 40 4436 9797/98
Email: india.support@arup.com
Web: www.oasys-software.com/dyna

or contact your local Oasys Ltd distributor

1 LS-DYNA Environment	1.1
1.1 Introduction	1.1
1.2 LS-DYNA	1.2
1.3 PRIMER	1.11
1.4 D3PLOT	1.13
1.5 T/HIS	1.15
1.6 Reporter	1.17
1.7 Utilities	1.22
1.8 Preferences	1.22
1.9 Manuals	1.22
1.10 Status	1.22
1.11 Support	1.22
1.12 Exit	1.22
1.13 Command Line Options	1.23
1.14 Command Line Submission Shell	1.23
2 Customising the Shell	2.1
2.1 LINUX Installation	2.1
2.2 Windows XP/VISTA/Windows 7 Installation	2.4
2.3 Customising the GUI Shell	2.7
2.4 Adding LS-DYNA versions to the Shell	2.17
2.5 Customising the LS-DYNA submission script	2.19
2.6 Creating an "oasys_queue" file	2.31
2.7 Adding items to the "Utilities" menu	2.34
3 Additional Windows Setup	3.1
3.1 Setting File Associations	3.1
4 Preferences	4.1
4.1 The oa_pref file	4.1
4.2 The preferences editor	4.2
Installation organisation	A.1
Version 12.0 Installation structure	A.1

1 LS-DYNA Environment

1.1 Introduction



1.1.1 LINUX

On LINUX the LS-DYNA Environment software is accessed by typing the command: **oasys_12**

1.1.2 Windows

On Windows the software is accessed via **Start -> Programs -> Oasys Ltd LS-DYNA Environment 12.0**



1.2 LS-DYNA

LS-DYNA

This button accesses the submission menu for LS-DYNA.

The first time this button is pressed the 'oa_pref' file will be read and any default values set (see [section 4](#) for more details on the 'oa_pref' file). After the file has been read following menu will be displayed.

LS-DYNA Submission

LS-DYNA

☒ Single Precision ☐ Double Precision

☒ SMP ☐ MPP ☐ Hybrid

Version : Win32 SP LS971vR3.1

Job Options

Input File :

File Format : Keyword ☐ Sequential Submission

CPU LIMIT : 0

MEMORY LIMIT : 9000000

MEMORY2 (MPP) :

Output Files : ARUP '.ptf ...'

☒ Increase Memory If Required

Parallel Options

Number of CPU's : 1 Consistency Flag : ☒ ON ☐ OFF

☒ Use Local Host

☐ Use Node File : .hostfile

☐ Use Node List :

Submission Options

☒ Online ☐ Delayed ☐ Batch ☐ Queue

Analysis Start Time : 00 : 00 Start Day : Now

Queue CPU LIMIT : 0

Queue MEMORY LIMIT : 15000000

Queue Name : CPU Limit : 0

Queue Options :

Restart Options

Restart Number : Restart Time : ☐ Input Deck

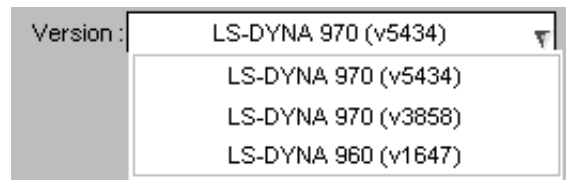
Post Processing Options

☐ Run T/HIS after analysis DISPLAY :

☐ Run REPORTER after analysis

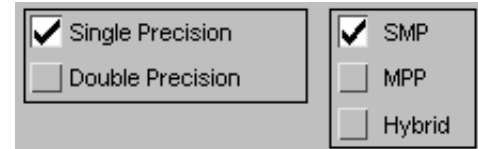
1.2.1 LS-DYNA version options

The submission shell can be used to access more than one version of LS-DYNA. Clicking on the tab labeled LS-DYNA version will bring up this drop down menu listing the versions of LS-DYNA present in the [dyna_versions](#) file.



Alternatively, the version of LS-DYNA can be specified using the tabs to the right of this drop down menu.

These tabs can be used to select the desired Precision and code-type for the analysis. Clicking on the tabs will cause the default LS-DYNA version corresponding to the options specified to be selected.



The options available are described below:

Precision:

Single Precision (32 bit floating point)

Double Precision (64 bit floating point)

Code Type:

SMP (Shared memory parallel)

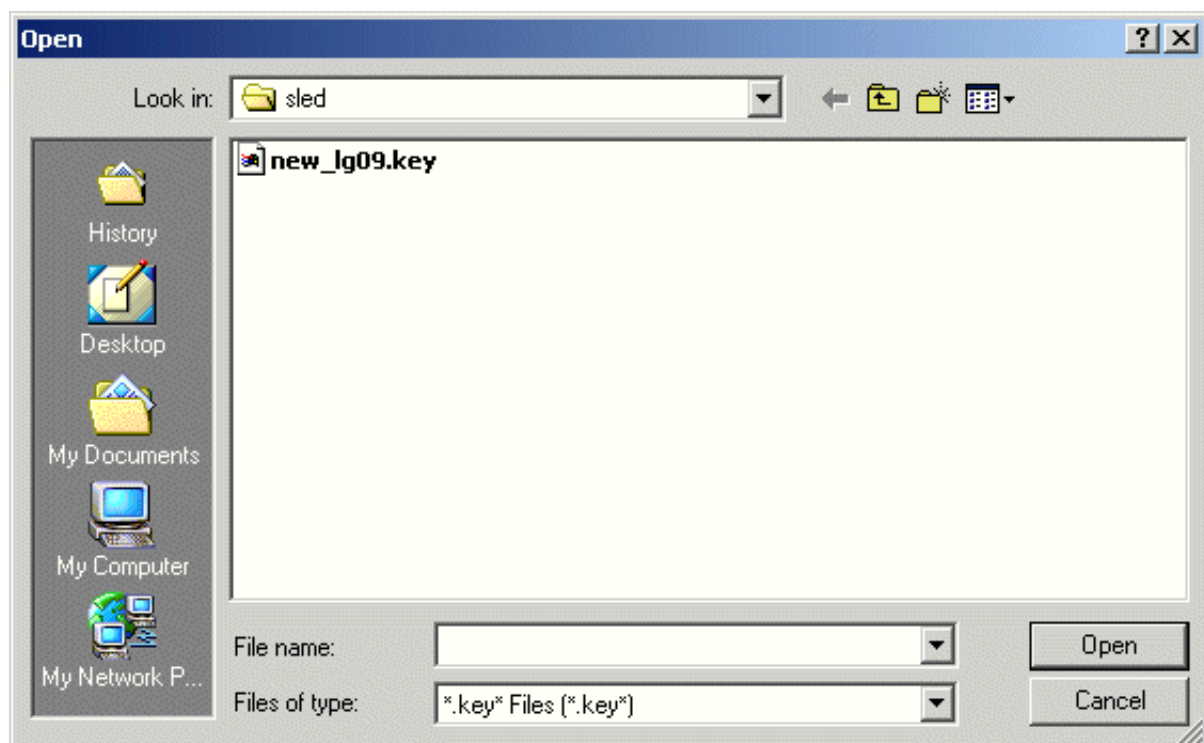
MPP (Distributed memory parallel)

Hybrid

1.2.2 Job options

1.2.2.1 Input File

Enter the name of the analysis file into the box provided. Jobs may be specified in any directory by entering either the full pathname of the input file or the pathname relative to the current directory. If the input file format is set to Keyword it will be assumed that the filename ends in '.key', otherwise it will be assumed that the filename ends in '.inf'. To search for the file, click on the icon to the left of the Jobname box. This will bring up a standard file selector box shown below.



The default search pattern will be set to '*.key' if the input file format is set to Keyword, '*.inf' if it is set to Fixed and '*.lst' if it is set to List.

1.2.2.2 Input file format

The input file format may be altered between Keyword and Fixed and List.

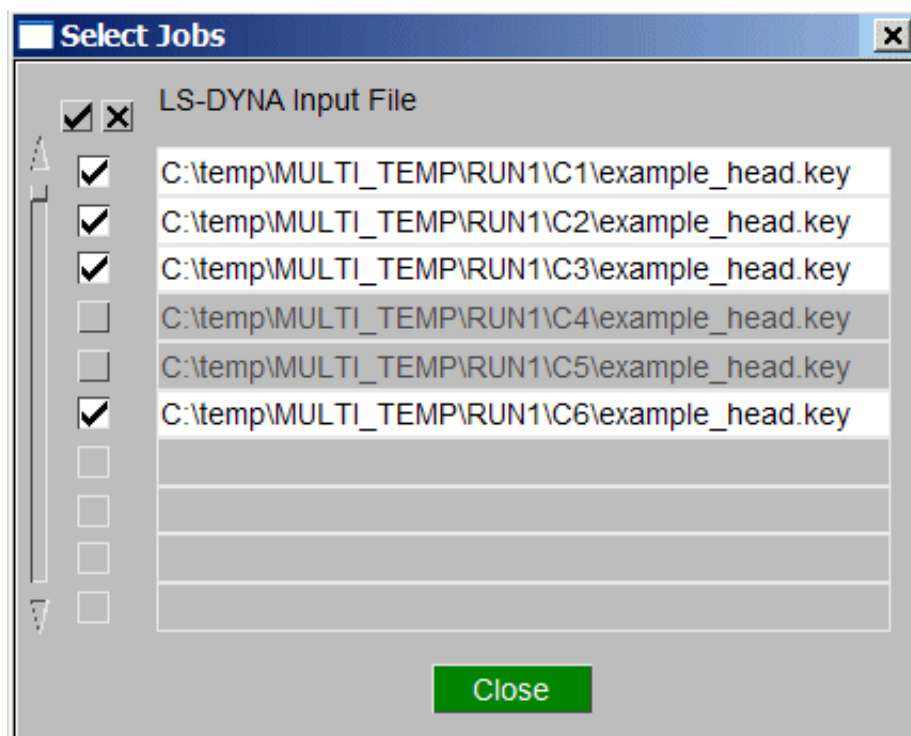
If the List option is chosen a file containing a list of LS-DYNA models can be selected for submitting to LS-DYNA. The List file can contain either just a list of LS-DYNA models or it can also contain additional information for running REPORTER automatically after the analysis terminates. (see [section 1.6.3](#) for details on the List file format).

All of the jobs can be submitted either [sequentially or simultaneously](#).

Keyword
Keyword
Fixed
List

1.2.2.3 Select Files To Run

If the input file format is set to [List](#) then a subset of the models listed in the file can be selected to be run when the List file is submitted.



1.2.2.4 Sequential Job Submission

<input checked="" type="checkbox"/> Sequential Submission

If a list of LS-DYNA jobs is submitted using the [List](#) option then they can either be submitted sequentially one after the other or they can all be submitted simultaneously.

1.2.2.5 CPU limit

The Analysis CPU Limit can be entered directly in the space provided. The units used to define the CPU limit can be switched between Seconds, Minutes and Hours by using the toggle available. The value entered is the actual amount of CPU time that LS-DYNA will use for the analysis. When this limit is reached LS-DYNA will terminate the job. A value of "0" means that no limit has been set and the job will run to termination.

1.2.2.6 MEMORY limit

The Analysis Memory Limit can be entered directly in the space provided. The units used to define the Memory Limit can be switched between Words and Megawords and Automatic using the available toggle. The value entered is the size of the main array declared internally within LS-DYNA that is used to store data in.

A value of "0" means that the analysis will use the default value built into LS-DYNA. If a value is specified on the *KEYWORD card in a input deck then that value will override all other memory definitions.

1.2.2.7 MEMORY2 (MPP only)

This option is only available for the MPP version of LS-DYNA. With the MPP version of LS-DYNA more memory is required for the first CPU that initialises the model than for the other processors. Generally if the first CPU requires X words of memory to initialise the job then the amount of memory required for each additional processor is X/n where n is the total number of CPU's.

1.2.2.8 Automatically Increase Memory If Required

If this option is selected then LS-DYNA will automatically increase the amount of memory it uses if the amount specified by [MEMORY LIMIT](#) is insufficient.

1.2.2.9 Optional Files

Clicking on the OPTIONS tab will display this menu of input and output options that can be used with LS-DYNA.

For more details on these options see the LS-DYNA User's manual.

ADDITIONAL FILES

Input Files

- ☐ Stress Initialization (.sif)
- ☐ Interface Segment (.isf2)
- ☐ VDA Geometry (.vda)
- ☐ CAL3D Input (.c3d)
- ☐ TOPAZ3D Temperature file (.htf)
- ☐ MADYMO Input File
- ☐ REMAP option
- ☐ MPP pfile
- ☐ GMINP (.gm)

Input Options

ENDTIM : 0.000000

ENDCYC : 0

PARA : 0

☐ case

☐ mcheck

Output Files

- ☐ Contact Force File (.ctf)
- ☐ Interface Segment (.isf1)
- ☒ Static Database File (.ztf)
- ☐ Winfrith Crack file (.crf)
- ☐ FSIFOR file (.fff)
- ☐ GMOUT (.gm)
- ☐ CPM Interface Force (.cpm)
- ☐ DEM Interface Force (.dem)
- ☐ FSILNK file (.fsl)

☐ .bern File :

Binary File Size 1024 (max 8192)

CLOSE

1.2.2.10 Output Files

This option controls the names of the output files generated by LS-DYNA, either ARUP or LSTC can be selected. With LS-DYNA 970 onwards the names of the output files can also be specified using the *KEYWORD_ID option where a filename prefix for all of the output files is specified within the input file. If a filename prefix is specified in the input file then this option will be replaced with the prefix specified in the file.

Output File	ARUP	LSTC	*KEYWORD_ID
Binary Complete State Database	'jobname'.ptf	d3plot	'prefix'.d3plot
Time History Database	'jobname'.thf	d3thdt	'prefix'.d3thdt
Extra Time History Database	'jobname'.xtf	xtfile	'prefix'.xtfile
Binary Output File	binout	binout	'prefix'.binout
Restart Dump File	'jobname'.dpf	d3dump	'prefix'.d3dump
Running Restart Dump File	'jobname'.adf	runrsf	'prefix'.runrsf

1.2.3 Parallel options

1.2.3.1 Number of cpus

This option will only be accessible if your system supports parallel versions of LS-DYNA. The required number of processors to run the job on can be selected using the arrow buttons. The maximum number of processors that can be selected is controlled through the 'oa_pref' file (see section 2.3)

The value selected will override the number of processors selected on the *CONTROL_PARALLEL card in the input deck.

1.2.3.2 Consistency Flag

This option controls whether parallel analyses are run with the accuracy option in LS-DYNA turned ON or OFF. The default is to turn the accuracy option ON which results in nearly identical results across different numbers of CPU's but a lower overall speedup in the job time.

The value selected will override that selected on the *CONTROL_PARALLEL card.

1.2.3.3 ☒ Use Local Host

Local
Host

This option will only be available when submitting MPP jobs using either the [Online](#), [Background](#) or [Batch](#) options.

When selected the MPP jobs will be submitted using only the machine that the shell is being run on.

1.2.3.4 ☐ Use Node File : .hostfile

Node File



This option will only be available when submitting MPP jobs using either the [Online](#), [Background](#) or [Batch](#) options.

This option can be used to select a file containing a list of Nodes and CPUS to use when submitting an MPP job. The format of the file will depend on the MPI library that the version of LS-DYNA has been built with.

MPICH2	cluster1:2 cluster2:2 cluster3 cluster4		
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"		
HP MPI / PLATFORM MPI	cluster1:2 cluster2:2 cluster3 cluster4		
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"		
INTEL MPI	cluster1:2 cluster2:2 cluster3 cluster4		
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"		
OPEN MPI	cluster1 cluster1 cluster2 cluster2 cluster3 cluster4	or	cluster1 slots=2 cluster2 slots=2 cluster3 cluster4
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"		

1.2.3.5 Node List

☐ Use Node List :

This option will only be available when submitting MPP jobs using either the [Online](#), [Background](#) or [Batch](#) options.

This option can be used to specify a string containing the list of Nodes and CPUS to use when submitting an MPP job. The format of the sting will depend on the MPI library that the version of LS-DYNA has been built with.

MPICH2	#hosts host1 cpus(1) host2 cpus(2) hostn cpus(n)
	4 cluster1 2 cluster2 2 cluster3 1 cluster4 1
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"
HP MPI / PLATFORM MPI	host1:cpus(1),host2:cpus(2),.....,hostn:cpus(n)
	cluster1:2,cluster2:2,cluster3,cluster4
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"
INTEL MPI	#hosts host1 cpus(1) host2 cpus(2) hostn cpus(n)
	4 cluster1 2 cluster2 2 cluster3 1 cluster4 1
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"
OPEN MPI	#hosts host1 cpus(1) host2 cpus(2) hostn cpus(n)
	cluster1,cluster1,cluster2,cluster2,cluster3,cluster4
	Submit a job using 4 hosts, use 2 CPU's on "cluster1" and "cluster2" and 1 CPU on "cluster3" and "cluster4"

1.2.4 Submission options

1.2.4.1 Submission method

The screenshot shows a 'Submission Options' dialog box. On the left, there are four radio buttons: 'Online', 'Delayed', 'Batch', and 'Queue'. The 'Queue' option is selected with a checkmark. To the right of these buttons are several input fields and dropdown menus. 'Analysis Start Time' is set to '00 : 00'. 'Start Day' is a dropdown menu set to 'Now'. 'Queue CPU LIMIT' is a text box with '0'. 'Queue MEMORY LIMIT' is a text box with '999.00'. 'Queue Name' is a dropdown menu set to 'dyna'. 'CPU Limit' is a dropdown menu set to 'NONE'. At the bottom, 'Queue Options' is a dropdown menu set to '2 CPU x 2 Nodes'.

Jobs may be submitted using 4 different submission methods. If some of the options are not available on your system then the option will be greyed out in the submission menu and will be unselectable. The parameters that can be specified under each option are different and those available under the Queue option will vary from system to system.

The four different submission options are :

- Online** Submit jobs interactively using an Xterm session. Delayed start times cannot be specified.
- Background** Submit jobs in background. If a delayed start times is specified then the job will be submitted using the LINUX 'at' command.
- Batch** Submit jobs into a batch queue using the LINUX 'batch' command. Delayed start times cannot be specified.
- Queue** Submit jobs to NQE batch queues using 'qsub' or equivalent commands. Delayed start times, Queue CPU/Memory Limits may be specified.

1.2.4.2 Analysis Start Time

If a job is submitted using either the Background or Queue options it is possible to specify a time at which the job will execute. The required time should be entered in the form HH:MM using a 24 hour clock. If no time is specified then the job will executed as soon as possible. If the time specified has already been passed then the job will start immediately.

(This option will not be available if the Queue option is submitting the job to a [SGE queuing](#) system).

1.2.4.3 Analysis Start Day

If a job is submitted using either the Background or Queue options it is possible to specify a day on which the job will execute. Use the toggle to specify the day on which the job will be executed. If a day is specified and no analysis start time has been set, the job will be submitted with a start time of 00:01 on the required day.

(This option will not be available if the Queue option is submitting the job to a [LSF or SGE queuing](#) system).

1.2.4.4 Queue CPU Limit

This option will only be accessible when jobs are being submitted using the Queue option. If your system does not require CPU Limits to be specified for NQS style queues then your system manager will have disabled this option. The Queue CPU Limit can be entered directly into the space provided. The units used to define the CPU Limit can be switched between Seconds, Minutes and Hours using the toggle available. The value entered is the total Queue CPU Limit that include the time taken to run the LS-DYNA analysis and any system time required. The value entered should be larger than the Analysis CPU Limit so that when the analysis finishes there is enough time to write LS-DYNA restart and result files. If the extra time specified is unlikely to be sufficient a warning message will be generated when the job is actually submitted to the Queue.

A value of "0" means that the analysis will be submitted with an unlimited Queue CPU Limit.

1.2.4.5 Queue Memory Limit

This option will only be accessible when jobs are being submitted using the Queue option. If your system does not require Memory Limits to be specified for NQS style queues then your system manager will have disabled this option. The Queue Memory Limit can be entered directly into the space provided. The units used to define the Memory Limit can be switched between Words and Megawords using the toggle available. The value entered defines the total amount of memory that the job will request from the system. In order to allow LS-DYNA to initialise this value should be set to a minimum of 5000000 words larger than the Analysis Memory Limit, (see section 2.3.1). If a smaller value is specified a warning message will be generated when the job is actually submitted to the Queue.

A value of "0" means that the analysis will be submitted with an unlimited Limit.

1.2.4.6 Queue Name

This option will only be accessible when jobs are being submitted using the Queue option. If the system has more than one NQS style queue then the required queue may be selected from a pull-down menu activated by the right hand mouse button. When a queue is selected the CPU limit for that queue will be displayed alongside the queue name. If the queue is a pipe queue the word PIPE will be displayed, if the queue has no CPU limit then the word NONE will be displayed. The default queue will be the first one listed in the `'oasys_queue'` file.

1.2.4.7 Queue Options

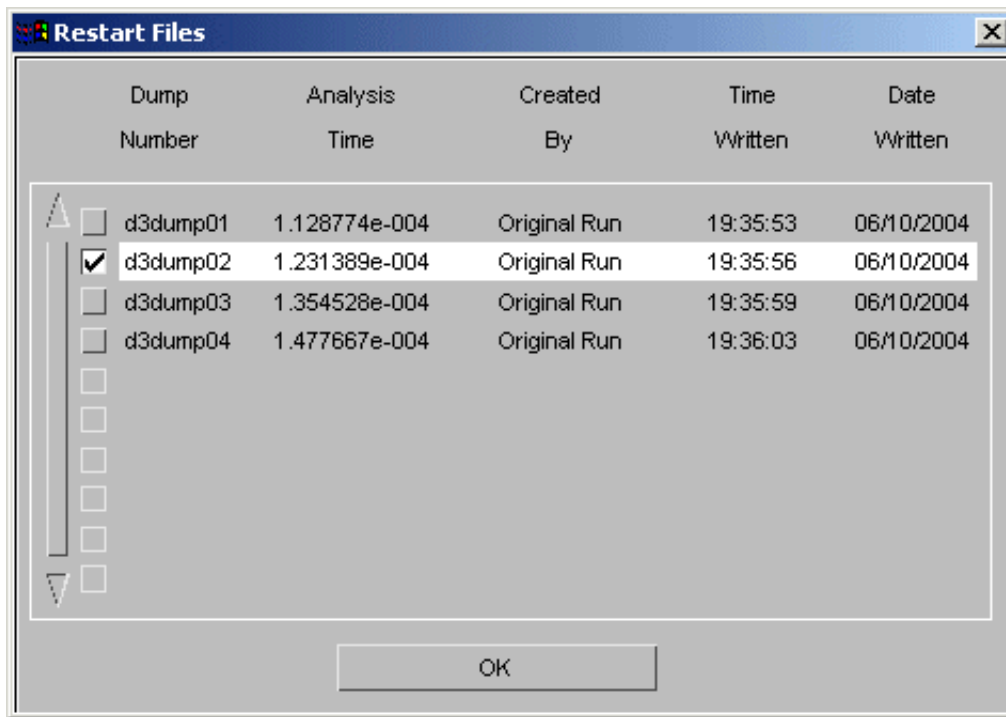
This option will only be accessible when jobs are being submitted using the Queue option. It is intended to be used so that user defined queue directives can be specified rather than using the ones produced by the shell (see [Section 2.3.34](#))

The options available will depend on the queue and number of cpus chosen and the contents of the `'oasys_queue'` file. Selecting an option will put the additional queue directives defined in this file into the submission script. If there are no options available for the selected number of cpus then the options for the next number of cpus up will be displayed, e.g. if the number of cpus chosen is 3, but no options are defined for 3 cpus, then if there are options defined for 4 cpus then the options for 4 cpus will be shown. If there are no options available then the default directives produced by the shell will be used.

1.2.5 Restarts

1.2.5.1 Select Restart Time

This option will only be available if the analysis is not a coupled LS-DYNA/ MADYMO3D analysis. When this button is selected all input to the Shell will be frozen while a search is carried out for any restart input files that exist for the specified job. When the search has been completed the window shown below with all of the available restart times will be displayed and the required time may be selected.



Select the desired restart file and press the OK button to return to the main submission window where the screen will be updated to reflect the details of the restart file selected. A check will also be carried out to see if a restart input deck exists and if one is found the user will be offered the opportunity to use it.

If, after a restart file has been selected, the user decides that he does not want to use it, the restart file selection menu should be re-displayed and the previously selected file should be deselected.

1.2.6 Post Processing Options

1.2.6.1 Run T/HIS after Analysis

Setting this option to ON will cause a fasttcf .inp script present in the job directory (the directory containing the analysis file) to be executed following termination of the job. Fast-tcf provides a fast, automated method of post-processing output data from LS-DYNA by producing a set of command files for T/HIS from an ASCII input file. This option will only be available if there is a jobname.inp file. Selecting OFF will disable this automatic post-processing

The Options button can be used to specify options which will affect how T/HIS operates when it is executed. This will open up the [T/HIS options window](#), which will allow T/HIS command line options to be explicitly defined.

1.2.6.2 Run Reporter after Analysis

Setting this option to ON will cause Reporter to automatically run following termination of the analysis. Selecting OFF will disable this automatic post-processing. The Options button can be used to set up input options for Reporter when it runs, see [section 1.6.1](#) for more details on these options.

1.2.6.3 Display

If you are running T/HIS or Reporter automatically at the end of an analysis then this option can be used to select the machine on which T/HIS and Reporter should be displayed when it runs. By default this will be set to the machine you submitted the job from and will not normally need to be changed.

1.2.7 Submit

Click on this button to submit the LS-DYNA job using all of the options previously set. Before the job is actually submitted the following checks will be carried out:

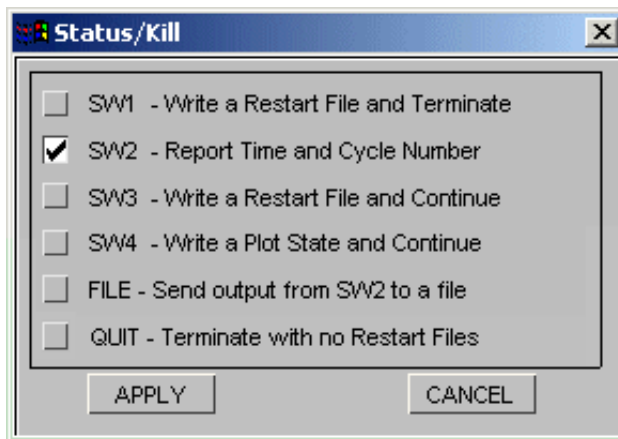
- The input file specified exists.
- The Queue CPU Limit, if specified, is larger than the Analysis CPU Limit.
- The Queue Memory Limit, if specified, is at least 6000000 words larger than the Analysis Memory Limit.
- An Account name, if required, has been specified.

1.2.8 Reset

Click on this button to reset all of the submission options to their default values.

1.2.9 Status Kill

This option can be used to either terminate an LS-DYNA analysis that is already running or to force the analysis to perform a range of tasks. Clicking on this button will bring up the menu shown below, displaying the possible options.



1.2.10 Cancel

Click on this button to close the submission menu and return to the main shell window without submitting an LS-DYNA job.

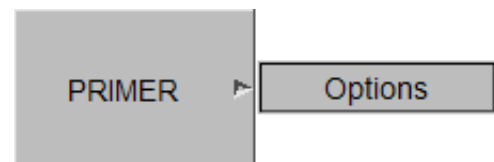
1.3 PRIMER

Select this option to access the PRIMER program. This program is a graphical model editor that will position occupant models, fold airbags and allow model parameters to be modified.

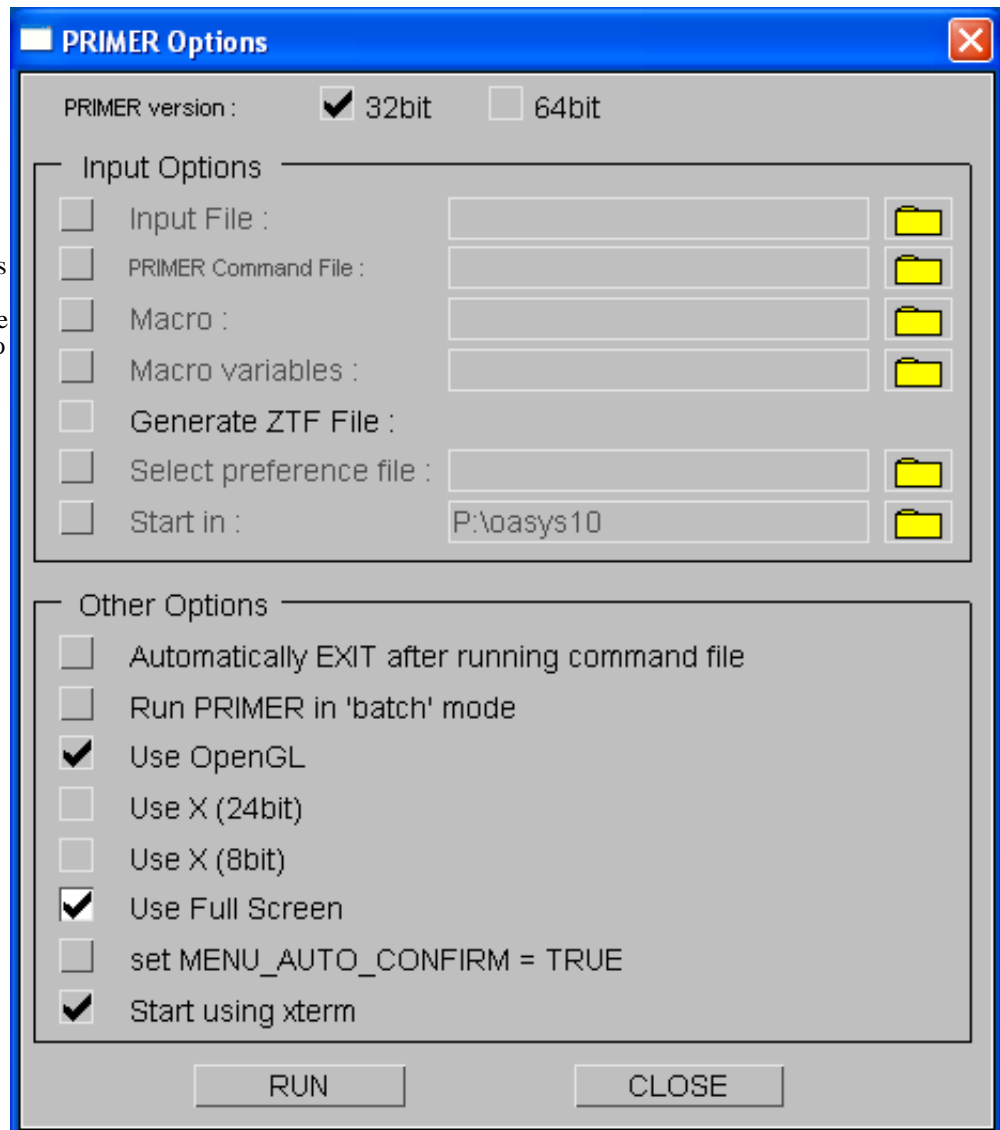


1.3.1 Primer Options

Right clicking on the Primer tab will cause an OPTIONS tab to appear. Selecting this option tab will cause the following window to appear



This menu allows the user to specify an input keyword file and a primer command file which will automatically run when primer opens. It also allows the user to pre-select other options such as graphics options. Select/ deselect the options by clicking on the respective tag. Click on the file divider icon in order to search for the respective files.



The image shows a Windows-style dialog box titled "PRIMER Options". It has a blue title bar with a close button (X) in the top right corner. The dialog is divided into two main sections: "Input Options" and "Other Options".

PRIMER version : ☒ 32bit ☐ 64bit

Input Options

- ☐ Input File : [text box] [file icon]
- ☐ PRIMER Command File : [text box] [file icon]
- ☐ Macro : [text box] [file icon]
- ☐ Macro variables : [text box] [file icon]
- ☐ Generate ZTF File :
- ☐ Select preference file : [text box] [file icon]
- ☐ Start in : P:\oasys10 [file icon]

Other Options

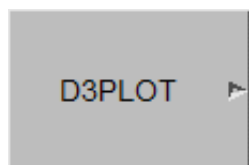
- ☐ Automatically EXIT after running command file
- ☐ Run PRIMER in 'batch' mode
- ☒ Use OpenGL
- ☐ Use X (24bit)
- ☐ Use X (8bit)
- ☒ Use Full Screen
- ☐ set MENU_AUTO_CONFIRM = TRUE
- ☒ Start using xterm

At the bottom of the dialog are two buttons: "RUN" and "CLOSE".

Input Options	
Input File	Specify the name of the keyword file you wish to be read into PRIMER
PRIMER Command File	Specify the name of a command file you wish to be executed on opening PRIMER
Macro	Specify the name of a macro file you wish to be executed on opening PRIMER
Macro variables	Specify the name of a macro variables file you wish to be used with the macro specified above
Generate ZTF File	Generate a ZTF file containing extra data for D3PLOT and T/HIS
Select preference file	Specify an oa_pref file to read from
Start in	Specify the directory where PRIMER will be fired up and where output files will be written to.
Other Options	
Automatically EXIT ...	Causes PRIMER to close once the specified command file has finished running
Run PRIMER in 'batch' mode	Starts PRIMER in batch mode; a non-graphical mode, which accepts commands from a file specified in the Input options
Use OpenGL	Starts PRIMER using OpenGL graphics mode
Use X (24 bit)	Starts PRIMER using 24bit X Windows graphics mode
Use X (8 bit)	Starts PRIMER using 24bit X Windows graphics mode
Use Full Screen	Opens the PRIMER window to fill the whole screen
Set MENU_AUTO_CONFIRM	This variable is often used when replaying command files which, when recorded, paused and asked the user to confirm things. (For example HELP and Warning messages.) If this option is selected PRIMER will continue automatically using the default option.
Start using xterm	If multiple instances of PRIMER are opened from the Shell on Linux systems, flexlm could count each of these as using a license. This is because if PRIMER is loaded from an xterm console, each instance is counted as using a license. This switch can be used to stop this behaviour.

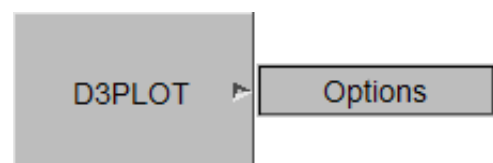
1.4 D3PLOT

Select this option to access D3PLOT. This program is a graphical post-processor designed to access the binary output files created by LS-DYNA. For more details see the D3PLOT Manual.

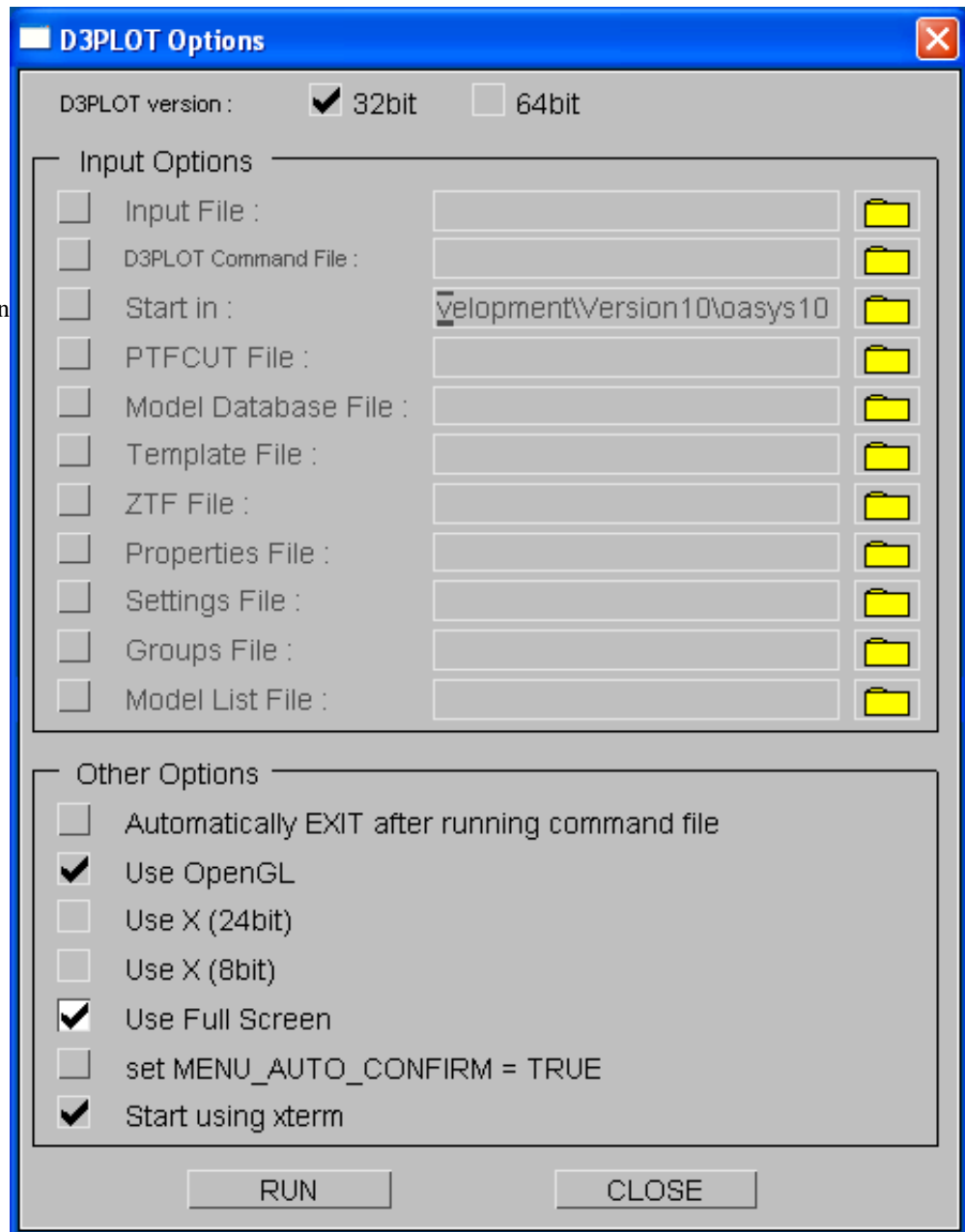


1.4.1 D3PLOT Options

Right clicking on the D3PLOT tab will cause an OPTIONS tab to appear. Selecting this option tab will cause the following window to appear














This menu offers the user the opportunity to specify a .ptf file that will be automatically read into D3PLOT on opening, a D3PLOT command file that will automatically be executed by D3PLOT on opening as well as other options. Select/ deselect the options by clicking on the respective tag. Click on the file divider icon in order to search for the respective files



D3PLOT Options

D3PLOT version : ☒ 32bit ☐ 64bit

Input Options

- ☐ Input File : 
- ☐ D3PLOT Command File : 
- ☐ Start in : 
- ☐ PTFCUT File : 
- ☐ Model Database File : 
- ☐ Template File : 
- ☐ ZTF File : 
- ☐ Properties File : 
- ☐ Settings File : 
- ☐ Groups File : 
- ☐ Model List File : 

Other Options

- ☐ Automatically EXIT after running command file
- ☒ Use OpenGL
- ☐ Use X (24bit)
- ☐ Use X (8bit)
- ☒ Use Full Screen
- ☐ set MENU_AUTO_CONFIRM = TRUE
- ☒ Start using xterm

Input Options	
Input File	Specify the name of the .ptf file you wish to be read into D3PLOT
D3PLOT Command File	Specify the name of a D3PLOT command file you wish to be executed on opening D3PLOT
Start in	Specify the directory where D3PLOT will be fired up and where output files will be written to.
PTFCUT File	Specify the name of a file to create a cutdown database (the file can be created in D3PLOT in the Utilities->Compress menu).
Model Database File	Specify the name of a model database file you wish to be read on opening D3PLOT
Template File	Specify the name of a template file you wish to be read on opening the model in D3PLOT
ZTF File	Specify the name of a ZTF file you wish to be read on opening the model in D3PLOT
Properties File	Specify the name of a properties file you wish to be read on opening the model in D3PLOT
Settings File	Specify the name of a settings file you wish to be read on opening the model in D3PLOT
Groups File	Specify the name of a groups file you wish to be read on opening the model in D3PLOT
Other Options	
Automatically EXIT ...	Causes D3PLOT to close once the specified command file has finished running
Use OpenGL	Open D3PLOT using OpenGL graphics mode
Use X (24 bit)	Open D3PLOT using 24bit X Windows graphics mode
Use X (8 bit)	Open D3PLOT using 24bit X Windows graphics mode
Use Full Screen	Opens the D3PLOT window to fill the whole screen
Set MENU_AUTO_CONFIRM	This variable is often used when replaying command files which, when recorded, paused and asked the user to confirm things. (For example HELP and Warning messages.) If this option is selected D3PLOT will continue automatically using the default option.
Start using xterm	If multiple instances of D3PLOT are opened from the Shell on Linux systems, flexlm could count each of these as using a license. This is because if D3PLOT is loaded from an xterm console, each instance is counted as using a license. This switch can be used to stop this behaviour.

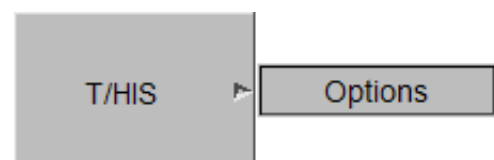
1.5 T/HIS

Left click on this option to access T/HIS. This program is a time history processing and graph plotting package designed to access the binary output files created by LS-DYNA. For more details see the T/HIS Manual.



1.5.1 T/HIS Options

Right clicking on the T/HIS tab will cause an OPTIONS tab to appear. Selecting this option tab will cause the following window to appear



This menu allows the user to specify a fasttcf .inp file or a T/HIS command file that will be automatically run when T/HIS is opened, along with other T/HIS options. Select/ deselect the options by clicking on the respective tag. Click on the file divider icon in order to search for the respective files.

T/HIS Options

T/HIS version : ☒ 32bit ☐ 64bit

Input Options

- ☐ Input File :
- ☐ Model List File :
- ☐ Curve File :
- ☐ Curve List File :
- ☐ FAST-TCF Input File :
- ☐ T/HIS Command File :
- ☐ Start in :
- ☐ Model Database File :

Other Options

- ☐ Automatically EXIT after running command file
- ☐ Run T/HIS in 'batch' mode
- ☒ Use Graphical Display ☒ OpenGL ☐ MESA
- ☐ Use Text Display
- ☒ Use Full Screen
- ☐ set MENU_AUTO_CONFIRM = TRUE
- ☒ Start using xterm

Input Options	
Input File	Specify the name of the .thf file for the analysis whose data you wish to process
Model List File	Specify the name of a file containing a list of models to be opened. Each model should be specified on a separate line in the file by giving the full pathname to one of the output files that T/HIS can read.
Curve File	Specify the name of the T/HIS curve file to read
Curve List File	Specify the name of a file containing a list of all the T/HIS curve files to be opened. Each curve file should be specified on a separate line in the file by giving the full pathname of the file.
FAST-TCF Input file	Specify the name of a fast-tcf .inp file for post-processing desired data output
T/HIS Command File	Specify the name of a T/HIS command file which collects and processes the desired data
Start in	Specify the directory where T/HIS will be fired up and where output files will be written to.
Model Database File	Specify the name of a model database file you wish to be read on opening T/HIS
Other Options	
Automatically EXIT ...	Causes T/HIS to close once the specified command file has finished
Run T/HIS in 'batch' mode	Opens up T/HIS in batch mode; a non-graphical mode, which accepts commands from a file specified in the Input options
Use Graphical Display	Opens up T/HIS in graphical mode. OpenGL or MESA graphics can be selected
Use Text (TTY) Display	Opens up T/HIS in text only mode, which allows the user to specify commands in the terminal window
Use Full Screen	Opens the T/HIS window to fill the whole screen
Set MENU_AUTO_CONFIRM	This variable is often used when replaying command files which, when recorded, paused and asked the user to confirm things. (For example HELP and Warning messages.) If this option is selected T/HIS will continue automatically using the default option.
Start using xterm	If multiple instances of T/HIS are opened from the Shell on Linux systems, flexlm could count each of these as using a license. This is because if T/HIS is loaded from an xterm console, each instance is counted as using a license. This switch can be used to stop this behaviour.

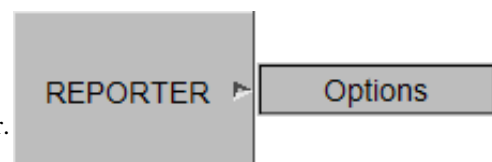
1.6 Reporter

Left click on this option to access REPORTER. This program is an automatic report generator for use with LS-DYNA, D3PLOT and T/HIS.



1.6.1 Reporter Options

Right clicking on the REPORTER tab will cause an OPTIONS tab appear. Selecting this option tab will cause the following window to appear.





This menu allows the user to define multiple sets of input options for REPORTER. These options can either be entered manually or they can be read from a file, [see section 1.6.3 for details on the LIST file format](#).

In addition to being accessed from the REPORTER Options menu this panel can also be selected from the LS-DYNA Submit menu.

1.6.1.1 Input List

This option can be used to select a predefined [LIST](#) file containing multiple sets of inputs for REPORTER. This option can not be modified if this panel was accessed from the LS-DYNA Submit menu.

Note: An attempt is made to determine a sensible name for the summary output directory based on the inputs in this file. If a summary input file is given then the pathname to this is used. If not, then if all the individual input files have the same pathname then this is used.

1.6.1.2 Input Options

Each LS-DYNA job can have the following options set.

Input File	Name of the LS-DYNA input file for post processing. This option can not be modified if this panel was accessed from the LS-DYNA Submit menu.
Template	Defines a REPORTER template file for the job
Output Dir	By default the output directory for any files generated by REPORTER will be set to the directory containing the input file.
Root Output Filename	By default the root of the input filename we will use as the default filename for any output files generated by REPORTER, i.e rootname.pdf, rootname.html ...
Variables	This option can be used to setup variables for use within REPORTER , see section 1.6.2

In addition to setting up options for individual LS-DYNA jobs this panel can also be used to generate a summary report. The options for the summary report are the same as for individual LS-DYNA jobs except that the Input file is optional.

If this panel was accessed from the REPORTER Options

1.6.1.3 Output Formats

REPORTER	Write a REPORTER report (orr) file after REPORTER has finished generating a report
HTML	Write an HTML file after REPORTER has finished generating a report
PDF	Write a PDF file after REPORTER has finished generating a report
VBA Script	Causes Reporter to write a visual basic macro (for PowerPoint) once it has finished generating a report
PowerPoint	Causes Reporter to write a Powerpoint file once it has finished generating a report

1.6.1.4 Combined Output Options

The normal Output formats allow REPORTER to write a PDF, HTML, or PowerPoint file for each analysis. However if you are doing lots of analyses it might be useful to combine all of the output together into a single PDF file or single PowerPoint file. The [Combined Output Options](#) section allows you to do this.



To do this REPORTER saves each analysis as a REPORTER report (orr) file after generating each file. Then it does an extra step, reading all of the report files and combining the output. In the above example a combined PDF file is going to be created.

If the report files already exist and so the reports do not need to be generated you can use [Skip generate](#) and the shell will only do the combining step.

1.6.1.5 Run Reporter

This option starts REPORTER once for each LS-DYNA job that has been defined and passes any input options that have been defined for that job to REPORTER. If a set of summary report options have been defined then an additional instance of REPORTER will be started to generate the summary report.

This option is not available if this panel was accessed from the LS-DYNA Submit menu as REPORTER will be started automatically after each LS-DYNA job has finished.

1.6.1.6 Clear Panel

This option can be used to quickly clear any options that have been set. This option is not available if this panel was accessed from the LS-DYNA Submit menu.

1.6.1.6 Save List As

This option can be used to save the currently defined set of REPORTER options to a new LIST file. By default this option will overwrite the [Input List](#).

1.6.2 Variables

Variable	Value
ZONE	C6
XCOORD	450
YCOORD	150
ZCOORD	177.477

Variable :

Value :

This windows allows the user to define variables for Reporter. Specify the variable name and its corresponding value in the relevant boxes and press **ADD** to add them to the list of defined variables. The variables to be fed into Reporter will appear in the greyed out boxes above. In order to delete one of the Variables in the list, enter the variable's name in the variable: box and press **DELETE**. In order to modify one of the variables, enter the variable's name into the Variable: box and press **Modify**. Once you have finished modifying all your variables, press **CLOSE**.

1.6.3 LIST File Format

The LIST file can be used to set up multiple inputs for REPORTER. The LIST file format consists of lines containing either Job information or Keywords.

Keyword	
\$VARIABLES	Defines a new list of comma separated Variable names for any following Job Lines. Any Variables defined before this Keyword are deleted. <code>\$VARIABLES, 1st Variable, 2nd Variable, 3rd Variable</code>
\$VARIABLES_ADD	Adds new Variables to the existing list of Variables <code>\$VARIABLES ADD, 1st Variable, 2nd Variable, 3rd Variable</code>
\$REPORTER_SUMMARY	Defines a set of inputs used to generate a summary report after a set of jobs have been run in LS-DYNA. <code>\$REPORTER_SUMMARY, Input file, Reporter Template, 1st Variable, 2nd Variable, 3rd Variable</code>
\$REPORTER_SUMMARY_DIR	Defines the directory used for summary output files <code>\$REPORTER SUMMARY DIR, Directory</code>
\$REPORTER_OUTPUT	Defines a list of output formats, possible options are REPORTER, HTML, PDF, POSTSCRIPT, VBA, POWERPOINT <code>\$REPORTER OUTPUT, format, format, format...</code>
\$REPORTER_ICONISE	Turns on/off the switch to iconise Reporter when running <code>\$REPORTER ICONISE,true or false</code>
\$REPORTER_OA_BATCH	Turns on/off the switch to run D3PLOT and T/HIS in batch <code>\$REPORTER OA BATCH,true or false</code>
\$REPORTER_STOP_ON_ERROR	Turns on/off the switch to stop Reporter if an error occurs <code>\$REPORTER ON ERROR,true or false</code>
\$REPORTER_EXIT	Turns on/off the switch to exit Reporter after running <code>\$REPORTER EXIT,true or false</code>
\$	Comment Line
	All other lines are assumed to contain Job information in the following format. <code>Input file, Reporter Template, 1st Variable, 2nd Variable, 3rd Variable</code>

e.g.

```

$ Define 4 variables for the following jobs
$
$VARIABLES,ID,X-COOD,Y-COORD,Z-COORD
$
$ Setup 4 jobs using the "head_impact.ort" template, with 4 variable values
$
C:\head_impact\r1\head.key,c:\templates\head_impact.ort, RUN 1, 0.03, 0.050, 0.06
C:\head_impact\r2\head.key,c:\templates\head_impact.ort, RUN 2, 0.03, 0.075, 0.06
C:\head_impact\r3\head.key,c:\templates\head_impact.ort, RUN 3, 0.03, 0.100, 0.06
C:\head_impact\r4\head.key,c:\templates\head_impact.ort, RUN 4, 0.03, 0.125, 0.06
$
$ Define a new set of variables
$
$VARIABLES,SUMMARY_ID
$
$ Add another variable
$
$VARIABLES_ADD,
$
$ Set up a summary report
$
$REPORTER_SUMMARY, , c:\templates\head_summary.ort
$REPORTER_SUMMARY_DIR,c:\head_impact\summary
$

```

\$ Specify output file formats

\$
\$REPORTER_OUTPUT,REPORTER,HTML,PDF

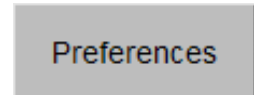
1.7 Utilities

This option will display an optional menu of additional utility programs. The exact list of programs given will depend on the options set up by the system administrator, see [Sect 2.7: Adding items to the "Utilities" menu](#) for more information.



1.8 Preferences

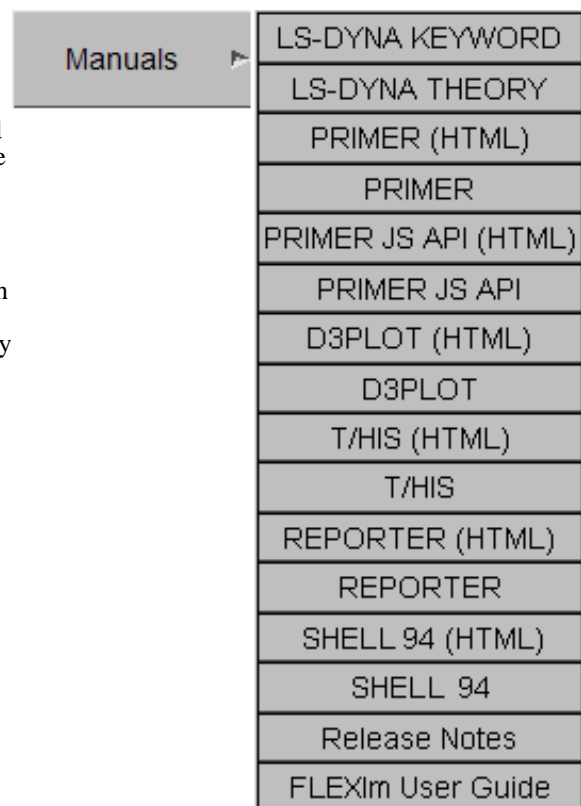
Selecting this tab will open up the **Preferences editor**. This is a graphical interface which allows for manual editing of the **oa_pref** file. For more information on the Preferences editor, see [Section 4](#)



1.9 Manuals

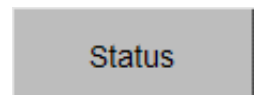
This option will only be available if the shell has been able to find a copy of the Adobe Acrobat software to use to display the on-line manuals. If the Acrobat software is available then this button will display a menu listing the manuals that are available for reading and printing.

The location of the Dyna keyword and theory manuals are set with the preferences `shell*dyna_keyword_manual` and `shell*dyna_theory_manual`, see [Section 2.3](#). If they are not set they will not be listed in the popup menu.



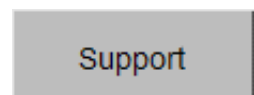
1.10 Status

This option will display information on the programs that have been licensed.



1.11 Support

This option will display support contact information.



1.12 Exit

This option will exit from the X-Windows shell.



1.13 Command Line Options

These options can be used to automatically start up programs and set default values.

Command	[Options]	Action
su	jobname	Open Submit window for job 'jobname'
pr	NONE	Start PRIMER
pl	NONE	Start D3PLOT
th	NONE	Start T/HIS

1.14 Command Line Submission Shell

When starting the shell on a LINUX system a command line version of the submission shell is available. It will start automatically if an X-connection cannot be established, or manually by putting 'cmd' as a command line option, i.e. 'oasys_12 cmd'.

The following screen will be displayed in the terminal: (Note that it may differ slightly depending on preference settings)

```

      Options                               Current selection
-----
(1) Code Version                          LS-DYNA          LS970 (v6763)
-----
      Job Options
-----
(2) Input File                            *** Nothing Selected ***
(3) Format                                Keyword
(4) CPU Limit                             0.00 Seconds
(5) Memory Limit                          Memory1 40.00 MWords
(6) Increase Memory If Required            On
(7) Filename Format                        Arup '.ptf...'
(8) Optional Files
-----
      Parallel Options
-----
(9) CPUs                                  2 CPUs, Consistency Flag On
-----
      Submission Options
-----
(11) Submission Type                       Queue, Now 00:00
(12) Queue CPU Limit                       0.00 Seconds
(13) Queue Memory Limit                    Memory 150.00 MWords
(14) Queue Name                           dyna, CPU Limit: 0
(15) Queue Options                         2 CPU x 1 Node
(99) Submit      (88) Reset      (77) Status/Kill      (-1)      Quit

```

The terminal window may have to be expanded for the contents to fit.

1.14.1 How To Use The Command Line Submission Shell

All the options available on the graphical version of the shell are available, except Restart, T/His and Reporter options. The defaults are the same as the graphical version.

The menu is set out with the options on the left and the current selection on the right.

The options are accessed by typing the number at the prompt and pressing return. This will bring up a menu with further options to make your selection. To return to the main menu, type in '-1' and press return. To quit from the program type in '-1' and press return in the main menu.

1.14.2 Code Version

At the main menu type in '1' and press return. A menu will be displayed with options for the code type:

```
Select the code type
(1) SMP (2) MPP (-1) Main menu
```

Select the required code type to bring up a menu with options for the precision:

```
Select the precision
(1) Single (2) Double (-1) Main Menu
```

Depending on the options chosen a menu will be displayed with the available LS-Dyna code versions (will differ depending on contents of dyna_versions file):

```
Select the code version
(1) LS970 (v6763) SP SMP RHE 3.0
(2) LS970 (v5434a) SP SMP RHE 3.0
(3) LS971 (R2 v7600_1224) SP SMP RHE 4.0
(4) LS971 R3_1 SP SMP
(-1) Main Menu
```

Select the required code version. The main menu will be displayed with the selected code version shown on the right hand side.

1.14.3 Input File

At the main menu type in '2' and press return. A menu will be displayed asking for the file name for the job:

```
Current directory: /u/mid/test
Enter the file name for the job (*.k*). (-1) Main menu
```

The directory the shell was started in is shown at the top of the menu. File names can be entered relative to this directory. For example, if the current directory is /u/mid/test and the job to run is in /u/mid/test/job/to_run.key then you can enter 'job/to_run.key' and press return. If the job to run is in /u/mid/to_run.key you can enter '../to_run.key'. Alternatively, the full path name can be entered.

If the file does not exist or is an invalid type, i.e. doesn't end in *.k* for keyword files, *.inf* for fixed format files or *.lst* for list files, you will be prompted to enter a file again.

Once a valid file has been chosen the main menu will be displayed with the full path name of the file selected shown on the right hand side.

1.14.4 File Format

At the main menu type in '3' and press return. A menu will be displayed asking for the file format for the job:

```
Select the job format
(1) Keyword (2) Fixed (3) List (-1) Main menu
```

Select the required format. If List was selected and the submission type is not Queue a further menu will be displayed asking whether the jobs should be sequential or not:

```
Turn sequential submission on or off
(1) Off (2) On (-1) Main menu
```

Select the required option. The main menu will be displayed with the selected format shown on the right hand side.

Note: Selecting a different job format to the current selection will clear the Input File selection.

1.14.5 CPU Limit

At the main menu type in '4' and press return. A menu will be displayed asking for the job CPU limit units:

```
Select the job cpu limit units
(1) Seconds (2) Minutes (3) Hours (-1) Main menu
```

Select the required units to bring up a menu asking for the job CPU limit:

```
Enter the job cpu limit. (-1) Main menu
```

Enter the required limit (can be an integer, decimal, or in the form 1.5E4). The main menu will be displayed with the selected job CPU limit on the right hand side.

1.14.6 Memory Limit

At the main menu type in '5' and press return. A menu will be displayed asking for the job memory limit units:

```
Select the job mem limit units
(1) Words (2) M Words (-1) Main menu
```

Select the required units to bring up a menu asking for the job memory limit:

```
Enter the job memory limit. (-1) Main menu
```

Enter the required limit (can be an integer, decimal, or in the form 1.5E4). If the Dyna version selected is MPP a second menu will ask for the other memory limit:

```
Enter the job memory2 limit. (-1) Main menu
```

The main menu will be displayed with the selected job memory limit on the right hand side.

1.14.7 Increase Memory If Required

At the main menu type in '6' and press return. A menu will be displayed asking if the job memory limit should be increase if required:

```
Increase the memory if required
(1) Off (2) On (-1) Main menu
```

Select the required option. The main menu will be displayed with the selected option on the right hand side.

1.14.8 Filename Format

At the main menu type in '7' and press return. A menu will be displayed asking for the filename format:

```
Select the filename format
(1) ARUP '.ptf ...' (2) LSTC '.d3plot ...' (-1) Main menu
```

Select the required option. The main menu will be displayed with the selected option on the right hand side.

1.14.9 Optional Files

Options	Current selection
----- Input Files -----	
(1) Stress Initialization (.sif)	Off
(2) Interface Segment (.isf2)	Off
(3) VDA Geometry (.vda)	Off
(4) CAL3D Input (.c3d)	Off
(5) TOPAZ3D Temperature File (.htf)	Off
(6) MADYMO Input File	Off
(7) REMAP Option	Off
(8) MPP pfile	Off
(9) GMINP (.gm)	Off
----- Output Files -----	
(10) Contact Force File (.ctf)	Off
(11) Interface Segment (.isf1)	Off
(12) Static Database File (.ztf)	On
(13) Winfrith Crack File (.crf)	Off
(14) FSIFOR File	Off
(15) GMOUT (.gm)	Off
(16) CPM Interface Force File	Off
(17) DEM Interface Force File	Off
----- Input Options -----	
(18) ENDTIM	0.000000
(19) ENDCYC	0
(20) PARA	0
(21) CASE	Off
(22) MCHECK	Off
----- Binary File Size -----	
(23) Binary File Size	1024
(-1) Main menu	

Options 1 - 17 can be toggled on and off simply by selecting them.

The analysis end time can be specified by selecting option 18. The following menu will be displayed:

Enter the end time. (-1) Optional files menu

Enter the required end time (can be an integer, decimal, or in the form 1.5E4). The optional files menu will be shown with the selected end time on the right hand side.

The analysis end cycle can be specified by selecting option 19. The following menu will be displayed:

Enter the end cycle. (-1) Optional files menu

Enter the required end cycle (an integer). The optional files menu will be shown with the selected end cycle on the right hand side.

The analysis para value can be specified by selecting option 20. The following menu will be displayed:

Para value
(1) Zero (2) One (-1) Optional files menu

Select the required option. The optional files menu will be shown with the selected para value on the right hand side.

To go back to the main menu type in '-1' and press return.

1.14.10 CPUs

At the main menu type in '9' and press return. A menu will be displayed asking for the number of CPUs:

Enter the number of cpus. (-1) Main menu

Enter the required number of CPUs (an integer). If the number of CPUs is too high for the code type (limit is defined in oa_pref) then you will be asked to enter again. Once a valid number has been entered a menu will be displayed asking if the consistency flag should be on or off:

**Turn the consistency flag on or off
(1) Off (2) On (-1) Main menu**

Select the required option. The main menu will be shown with the selected number of CPUs and the consistency flag status on the right hand side.

1.14.11 Node Options

This option will only be available if the submission type is not queue and the code type is MPP.

If it is available then at the main menu type in '10' and press return. A menu will be displayed asking for the node option:

**Select the node option
(1) Local Host (2) Node File (3) Node List (-1) Main menu**

Select the required option. If Local Host was selected the main menu will be displayed. If Node File was selected a further menu will be displayed asking for the location of the file:

**Current directory: /u/mid/test
Enter the node file name. (-1) Main menu**

The directory the shell was started in is shown at the top of the menu. In the same way as input files, node files can be specified relative to this directory or a full path name can be entered. No check is made to see if the file exists.

If Node List was selected a further menu will be displayed asking for a string:

Enter the node list string. (-1) Main menu

Enter the text and press return. The main menu will be shown with the selected option and file or node list string on the right hand side.

1.14.12 Submission Type

At the main menu type in '11' and press return. A menu will be displayed asking for the submission type:

```
Select the submission type
(1) Online
(2) Background
(3) Batch (unavailable)
(4) Queue (unavailable)
(-1) Main menu
```

Some submission types may be unavailable depending on the settings in the oa_pref file. If an unavailable option is chosen you will be asked to select again.

If Background or Queue are chosen a further menu will be displayed asking for the submission day:

```
Select the submission day
(1) Now (2) Monday (3) Tuesday (4) Wednesday
(5) Thursday (6) Friday (7) Saturday (8) Sunday (-1) Main menu
```

Select the required option to bring up a menu for the start time hour:

```
Enter the start time hour (0 - 23) (-1) Main menu
```

Enter the required hour to bring up a menu for the start time minute:

```
Enter the start time minute (0 - 59) (-1) Main menu
```

Enter the required minute. The main menu will be displayed with the selected submission type and start time, if applicable, on the right hand side.

Note: If you do not want a delay choose a submission time of Now 00:00.

1.14.13 Queue CPU Limit

If the submission type is Queue then an option to set the queue cpu limit will be available. At the main menu type in '12' and press return. A menu will be displayed asking for the queue CPU limit units:

```
Select the queue cpu limit units
(1) Seconds (2) Minutes (3) Hours (-1) Main menu
```

Select the required units to bring up a menu asking for the queue CPU limit:

```
Enter the job cpu limit. (-1) Main menu
```

Enter the required limit (can be an integer, decimal, or in the form 1.5E4). The main menu will be displayed with the selected queue CPU limit on the right hand side.

1.14.14 Queue Memory Limit

If the submission type is Queue then an option to set the queue memory limit will be available. At the main menu type in '13' and press return. A menu will be displayed asking for the queue memory limit units:

```
Select the queue mem limit units
(1) Words (2) M Words (-1) Main menu
```

Select the required units to bring up a menu asking for the queue memory limit:

```
Enter the queue memory limit. (-1) Main menu
```

Enter the required limit (can be an integer, decimal, or in the form 1.5E4). The main menu will be displayed with the selected queue memory limit on the right hand side.

1.14.15 Queue Name

If the submission type is Queue then an option to choose the queue name will be available. At the main menu type in '14' and press return. A menu will be displayed asking for the queue name:

```
Select the queue name
(1) Dyna
(2) Nastran
(3) Other
(-1) Main Menu
```

The options available will depend on the contents of the '[oasys_queue](#)' file.

Select the required option. The main menu will be displayed with the selected queue shown on the right hand side.

1.14.16 Queue Options

If the submission type is Queue then an option to specify some queue options will be available. At the main menu type in '15' and press return. A menu will be displayed with the available commands, depending on the queue and number of cpus chosen (will differ depending on the contents of the '[oasys_queue](#)' file):

```
Select the queue command
(1) 2 CPUs x 1 Node
(2) 1CPUs x 2 Nodes
(-1) Main Menu
```

Select the required option. The main menu will be shown with the selected queue command.

1.14.17 Submit

Once you are happy with the options chosen, at the main menu type in '99' and press enter to submit the job.

If output files already exist a menu asking whether to overwrite or delete them will be displayed:

```
Some output files already exist
(1) Overwrite (2) Delete (3) Cancel
```

Select the required option and if cancel was not selected the job will be submitted.

1.14.18 Reset

To reset the selected options to defaults at the main menu type in '88' and press enter.

1.14.19 Status / Kill

The status of a run can be queried or killed by typing in '77' and enter at the main menu. The following menu will be displayed:

Options Current selection

-
- (1) SW1 - Write a Restart File and Terminate Off
 - (2) SW2 - Report Time and Cycle Number On
 - (3) SW3 - Write a Restart File and Continue Off
 - (4) SW4 - Write a Plot State and Continue Off
 - (5) FILE - Send Ouput from SW2 to a File Off
- (99) Apply (-1) Main menu

Select the required option to toggle what to do. Select Apply to carry out the option selected.

Note: A job must be selected before this menu can be displayed.

2 Customising the Shell

This section is relevant to those responsible for installing the software suite.

2.1 LINUX Installation

On LINUX systems the Shell is accessed via the "oasys_12" command. If you have installed the software using the install script then the oasys_12 command file (in the /executables directory) will automatically be updated to contain the correct path for the directory the software was installed in and the name of the license server for the software installation. .

In addition to specifying the installation directory and the license server the "oasys_12" command file can also be set a number of other options on LINUX systems.

After installing the software the "oasys_12" command file should contain the following. (This assumes the software was loaded in a directory called /prg and that the license server in a machine called atghps50.

```
#!/bin/csh -f
onintr start_flush
#
# Set OA_INSTALL to point to the directory containing the software
#
setenv OA_INSTALL "/prg/oasys12/executables"
#
# Set OA_ADMIN directory
#
setenv OA_ADMIN
#
# Set OASYS_LICENSE_FILE to either the Oasys flexlm license file or
# the license server
#
# e.g setenv OASYS_LICENSE_FILE $OA_INSTALL/oasys_flexlm.dat
# or setenv OASYS_LICENSE_FILE @hostname
#
setenv OASYS_LICENSE_FILE @atghps50
#
# Set LSTC_FILE to either the lstc license file or the server
#
# e.g. setenv LSTC_LICENSE          local
#      setenv LSTC_FILE             $OA_INSTALL/LSTC_FILE
#
# e.g. setenv LSTC_LICENSE_SERVER  hostname
#      setenv LSTC_INTERNAL_CLIENT ON
#      setenv LSTC_LICENSE         network
#
setenv LSTC_FILE $OA_INSTALL/LSTC_FILE
#
# Other environment variables
#
setenv DISPLAY_FACTOR automatic
setenv USERID `whoami`
setenv LICENSE_TYPE "flexlm"
set noglob
#
#
# Set EDITOR (if not set) to the command to invoke an external editor. This
# EDITOR is currently used for editing comment lines in Oasys Ltd. PRIMER.
#
# if(! $?EDITOR) then
# setenv EDITOR /usr/bin/kedit # LINUX
# endif
#
# Environment variables for post processors
#
# If FILE_SKIP has not been set then set it to 5
#
if(! $?FILESKIP) then
    setenv FILE_SKIP 5
```

```

endif
#
# MACHINE SPECIFIC LINES
# =====
#
# extra line needed for some IBM machines
# setenv LANG En_US
#
# Extra line needed for SUN Solaris Machines
# setenv LD_LIBRARY_PATH /usr/openwin/lib
#
#
# Now start the main shell executable
#
set cwd = `pwd`
setenv PWD $cwd
$OA_INSTALL/xshell_12 $*
#
# exit
#
exit:
exit

```

The following parameters will require modifying in accordance with your system layout

```

setenv OA_INSTALL<option>
setenv OA_ADMIN / OA_ADMIN_12<option> (optional)
setenv OASYS_LICENSE_FILE / LM_LICENSE_FILE <option>
setenv LSTC_FILE / LSTC_LICENSE_SERVER <option>
setenv MENU_AUTO_CONFIRM <option> (optional)
setenv FILE_EXIST_ACTION <option> (optional)
setenv ECHO_PREFERENCE <option> (optional)

```

2.1.1 setenv OA_INSTALL <option>

The 'OA_INSTALL' Shell variable must be modified so that it references the directory in which the Shell is loaded. i.e if the software has been loaded in:

/prg/oasys12/executables

set this line to:

```
setenv OA_INSTALL"/prg/oasys12/executables"
```

This line should be set automatically by the installation script.

2.1.2 setenv OA_ADMIN / OA_ADMIN_12<option>

If a top level administration directory is to be used then OA_ADMIN_12 (for release 12.0) must be defined for all machines on which the software is to be run. This variable should be set to the full pathname of the administration directory.

A description of the installation organisation is given in the [Appendix](#).

2.1.3 setenv OASYS_LICENSE_FILE / LM_LICENSE_FILE <option>

Either OASYS_LICENSE_FILE or LM_LICENSE_FILE can be used to locate a valid license for the Oasys Ltd. LS-DYNA environment software. It is recommended that OASYS_LICENSE_FILE is used as this can speed up the checkout of licenses on systems where LM_LICENSE_FILE is used to find other license servers as well.

2.1.3.1 Floating Network License

If you are running the software using a license server then this variable should be set to point to the license server machine using the machine's hostname.

i.e **setenv OASYS_LICENSE_FILE hostname**
 or **setenv OASYS_LICENSE_FILE port@hostname** if you have specified a non default port for the server.

2.1.3.2 Fixed Stand-alone Licenses

If you are using a node locked license file then this variable should be set to point to the location of the license file.

i.e **setenv OASYS_LICENSE_FILE <INSTALL_DIR>/oasys_flexlm.dat**

2.1.3.3 Multiple FLEXlm license files

If you have other software that uses FlexLM then this Environment variable may already be set to point to a license file or a license server. If this variable is already set then you may add the license server or file for this software by specifying a colon separated list of values.

i.e **setenv OASYS_LICENSE_FILE port@hostname1: @hostname2**

2.1.4 setenv LSTC_FILE / LSTC_LICENSE_SERVER <option>

On LINUX machines LS-DYNA can use either a nodelocked license or a floating license system.

If you are using a nodelocked license then LSTC_FILE should be set to the full pathname of the license file. By default this file should be called 'LSTC_FILE' and it should be located in the 'executables' directory. The variable LSTC_LICENSE should also be set to "local"

i.e **setenv LSTC_FILE \$OA_INSTALL/LSTC_FILE**
setenv LSTC_LICENSE local

If you are using the floating license system then the variables LSTC_LICENSE_SERVER, LSTC_INTERNAL_CLIENT and LSTC_LICENSE should be set as follows.

i.e **setenv LSTC_LICENSE_SERVER hostname** where hostname is the LSTC license server
setenv LSTC_INTERNAL_CLIENT OFF
setenv LSTC_LICENSE network

2.1.5 setenv MENU_AUTO_CONFIRM <option>

This variable is often used when replaying command files which, when recorded, paused and asked the user to confirm things. (For example HELP and Warning messages.) Possible options for this variable are, **true** and **false**.

If the variable is set (**true**) then these will not pause and will behave as if the user had pressed "OK" - meaning that command files can play back without user intervention.

2.1.6 setenv FILE_EXIST_ACTION <option>

This variable controls the action to be taken when opening a file for output, and the file already exists. Possible options for this variable are **"none"**, **overwrite** and **append**.

Normally you will be prompted for the action to be taken when a file selected for output already exists. However if this variable is set to overwrite or append then the relevant action will be taken automatically.

This is generally used when playing automatic post-processing batch scripts

2.1.7 setenv ECHO_PREFERENCE <option>

If this variable is set to "1" then any command line arguments used to start T/HIS, PRIMER or D3PLOT will be echoed to the screen along with any settings read from preference files.

2.2 Windows XP/VISTA/Windows 7 Installation

When the software is installed a set of icons will automatically be added to:

Start -> Programs -> Oasys Ltd LS-DYNA Environment 12.0



After loading the software the following system variables should be set up.

OASYS_LICENSE_FILE / LM_LICENSE_FILE
OA_ADMIN / OA_ADMIN_12 (optional)
LSTC_FILE / LSTC_LICENSE_SERVER <option>
HOME
MENU_AUTO_CONFIRM (optional)
FILE_EXIST_ACTION (optional)
ECHO_PREFERENCE (optional)
DISPLAY_HEIGHT / DISPLAY_WIDTH (optional)

We recommend that these variables are set up by a user with Administrator privileges so that they then apply to all users on the system.

2.2.1 OASYS_LICENSE_FILE / LM_LICENSE_FILE

Either OASYS_LICENSE_FILE or LM_LICENSE_FILE can be used to locate a valid license for the Oasys Ltd. LS-DYNA environment software. It is recommended that OASYS_LICENSE_FILE is used as this can speed up the checkout of licenses on systems where LM_LICENSE_FILE is used to find other license servers as well.

2.2.1.1 Floating Network License

If you are running the software using a license server then this variable should be set to point to the license server machine using the machine's hostname.

i.e **OASYS_LICENSE_FILE = hostname**
 or **OASYS_LICENSE_FILE = port@hostname** if you have specified a non default port for the server.

2.2.1.2 Fixed Stand-alone Licenses

If you are using a node locked license file then this variable should be set to point to the location of the license file.

i.e **OASYS_LICENSE_FILE** = <INSTALL_DIR>/oasys_flexlm.dat

2.2.1.3 Multiple FLEXlm license files

If you have other software that uses FlexLM then this Environment variable may already be set to point to a license file or a license server. If this variable is already set then you may add the license server or file for this software by specifying a colon separated list of values.

i.e **OASYS_LICENSE_FILE** = port@hostname1;@hostname2

2.2.2 OA_ADMIN / OA_ADMIN_12<option>

If a top level administration directory is to be used then OA_ADMIN_12 (for release 12.0) must be defined for all machines on which the software is to be run. This variable should be set to the full pathname of the administration directory.

A description of the installation organisation is given in the [Appendix](#).

2.2.3 LSTC_FILE / LSTC_LICENSE_SERVER <option>

On Windows machines LS-DYNA can use either a nodelocked license or a floating license system.

If you are using a nodelocked license then LSTC_FILE should be set to the full pathname of the license file. By default this file should be called 'LSTC_FILE' and it should be located in the 'executables' directory. The variable LSTC_LICENSE should also be set to "local"

i.e **LSTC_FILE** = <INSTALL_DIR>\LSTC_FILE
LSTC_LICENSE = local

If you are using the floating license system then the variables LSTC_LICENSE_SERVER, LSTC_INTERNAL_CLIENT and LSTC_LICENSE should be set as follows.

i.e **LSTC_LICENSE_SERVER** = hostname where hostname is the LSTC license server
LSTC_INTERNAL_CLIENT = ON
LSTC_LICENSE = network

2.2.4 HOME

The software suite uses a preference file to control a number of settings. By default the software will look for the preference file in 4 locations, the OA_ADMIN_xx directory (if set), the installation directory, \$HOME and the current working directory. The global preference files located in the OA_ADMIN_xx directory and installation directory are read first and then any user specific options are read from the preference file located in \$HOME.

If this variable is not set the user will not be able to set up there own user preferences.

2.2.5 MENU_AUTO_CONFIRM

This variable is often used when replaying command files which, when recorded, paused and asked the user to confirm things. (For example HELP and Warning messages.) Possible options for this variable are "**none**", **overwrite** and **append**.

If the variable is set (**true**) then these will not pause and will behave as if the user had pressed "OK" - meaning that command files can play back without user intervention.

2.2.6 FILE_EXIST_ACTION

This variable controls the action to be taken when opening a file for output, and the file already exists. Possible options for this variable are "**none**", **overwrite** and **append**.

Normally you will be prompted for the action to be taken when a file selected for output already exists. However if this variable is set to overwrite or append then the relevant action will be taken automatically.

This is generally used when playing automatic post-processing batch scripts

2.2.7 ECHO_PREFERENCE

If this variable is set to "1" then any command line arguments used to start T/HIS, PRIMER or D3PLOT will be echoed to the screen along with any settings read from preference files.

2.2.8 DISPLAY_HEIGHT / DISPLAY_WIDTH

The software uses system functions to obtain screen dimensions which are used to calculate font sizes. Oasys Ltd have noticed that on some systems, the screen dimensions are not reported correctly (on identical systems Windows XP returned the correct dimensions while Windows Vista returned sizes approximately 20% larger).

If fonts used by the Oasys Ltd. LS-DYNA environment software appear to be the wrong size these 2 variables can be used to override the system calls to define the correct screen dimensions:

set DISPLAY_HEIGHT = (screen physical height in mm)
set DISPLAY_WIDTH = (screen physical width in mm)

2.2.9 Setting Environment Variables



To set these system variables, activate the control panel, using
Windows NT and 2000: Start -> Settings -> Control Panel
Windows xp: Start -> Control Panel

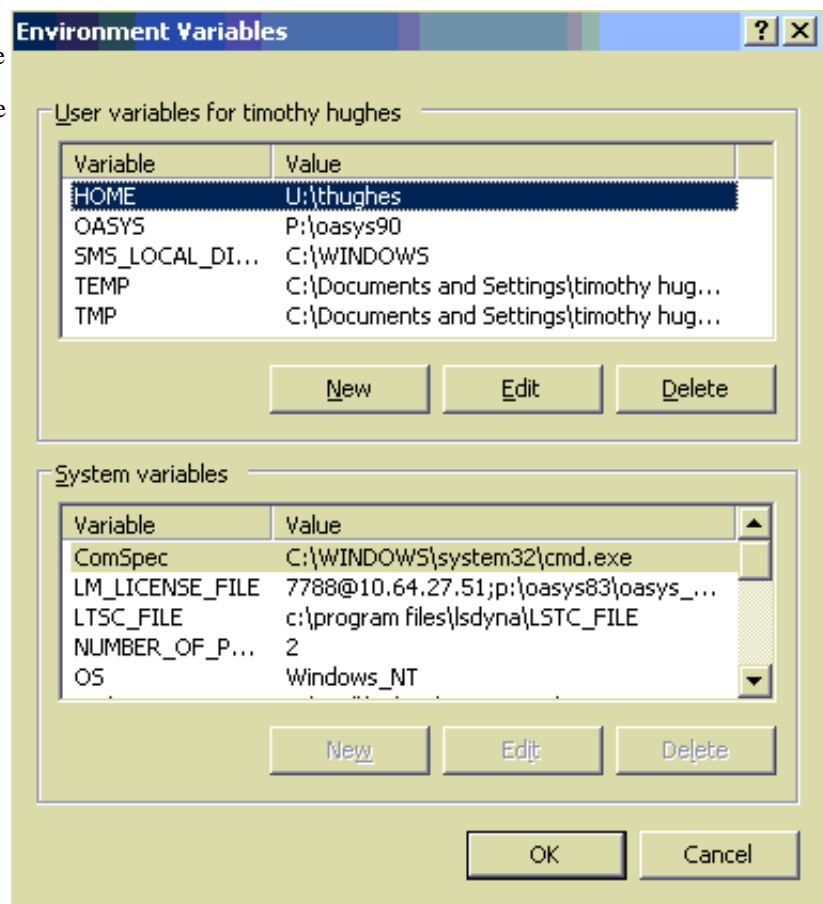
and then select the System Icon.

Windows NT - Followed by the Environment tab.

Windows 2000 and XP - Followed by the Advanced tab and then the Environment button.

This will bring up the window displayed left. To set up a new Environment Variable enter the variable's name in the Variable box followed by the variable's Value in the Value Box. After setting up each new Environment Variable press **Set** before entering the next new Environment Variable.

Finally, after setting up all the new variables, select **Apply** followed by **OK** to dismiss the System Properties menu.



2.3 Customising the GUI Shell

The '[oa_pref](#)' file controls the setup of a number of the programs in the software suite. Most of the programs will look for this file in the following 4 locations in the order given to allow individual users to customise the programs for their own needs :

- The [OA_ADMIN_xx](#) directory
- The directory containing the executables ([OA_INSTALL](#) directory)
- The user's login directory
- The current working directory

The '[xshell_12](#)' Shell will look for this file in all four directories. However to stop users trying to override the system settings, it will only read a selection of the shell preferences from the home and current working directory (those that do not affect system settings). This section only deals with the options for setting up the GUI Shell, the options for the other programs are covered in the relevant manuals.

From v94 preferences can be locked. If a preference is locked it cannot be changed in an [oa_pref](#) file in a more junior directory. To lock a preference use the syntax '**shell#**' rather than '**shell***'.

From v94 Environment variables can be used and they will be expanded by the shell. For example on Windows you could specify the temp folder with `shell*temp_folder: %USERPROFILE%/my_temp_folder`.

Any line in the '[oa_pref](#)' file that does not start with the string '**shell***' or '**shell#**' is ignored, normally comment lines will be prefixed with a # for clarity. The relevant sections of this file follows below.

Preferences file for software.

#

SHELL stuff:

#

Definition

Type Valid arguments

Default

online

logical TRUE or FALSE

TRUE

background

logical TRUE or FALSE

TRUE

batch

logical TRUE or FALSE

TRUE

queue

logical TRUE or FALSE

TRUE

# default	string	ONLINE or BACKGROUND or BATCH or QUEUE	ONLINE
# dyna_keyword_manual	string	Pathnames of up to 4 Dyna keyword manuals, separated by commas	
# dyna_theory_manual	string	Pathname of Dyna theory manual	
# batch_type	string	NQS,LSF,SGE,PBS,PBSPRO7 or PBSPRO71	NQS
# queue_output_file	logical	Add output file directive to job script	TRUE
# queue_error_file	logical	Add error file directive to job script	TRUE
# queue_error_path	logical	Add job path to output/error directive in job script	TRUE
# queue_memory	real	Batch Queue Memory Size	15000000
# queue_memory_units	string	WORDS or MWORDS	WORDS
# queue_cpu	real	Batch Queue CPU Limit	0
# queue_cpu_units	string	SECONDS, MINUTES or HOURS	SECONDS
# queue_space	real	Batch Queue File Space Limit	1024 1024
# queue_machine	string	Name of LSF queuing machine	
# request_cpu_limit	logical	TRUE or FALSE	TRUE
# request_memory_limit	logical	TRUE or FALSE	TRUE
# request_space_limit	logical	TRUE or FALSE	FALSE
# system_cpu	real	Minimum System Time (seconds)	90
# job_format	string	KEYWORD or FIXED	KEYWORD
# file_names	string	ARUP or LSTC	ARUP
# job_memory	real	Job Memory Size	9000000
# job_memory_units	string	WORDS or MWORDS	WORDS
# job_cpu	real	Job CPU Limit	0
# job_cpu_units	string	SECONDS, MINUTES or HOURS	SECONDS
# allow_mem_change	logical	TRUE or FALSE	TRUE
# file_size	integer	Binary File Size	1024
# write_ctf	logical	TRUE or FALSE	FALSE
# write_ztf	logical	TRUE or FALSE	TRUE
# batch_display	string	TRUE or FALSE	
# keep_files	logical	TRUE or FALSE	FALSE
# d3plot_version	string	name of D3PLOT executable	d3plot11.exe
# d3plot64_version	string	name of 64 bit D3PLOT executable	d3plot11_64.exe
# d3plot_default	integer	default D3PLOT version (32 or 64)	32
# this_version	string	name of T/HIS executable	this11.exe
# this64_version	string	name of 64 bit T/HIS executable	this11_64.exe
# this_default	integer	default T/HIS version (32 or 64)	32
# primer_version	string	name of PRIMER executable	primer11.exe
# primer64_version	string	name of 64 bit PRIMER executable	primer11_64.exe
# primer_default	integer	default PRIMER version (32 or 64)	32
# reporter_version	string	name of REPORTER executable	reporter11.exe
# reporter64_version	string	name of 64 bit REPORTER executable	reporter11_64.exe
# reporter_default	integer	default REPORTER version (32 or 64)	32
# shell_release	string	version # displayed for shell	12.0
# max_cpus	integer	Maximum number of CPUS	4

# max_mpp_cpus	integer	Maximum number of CPUs for MPP	4
# default_cpus	integer	Default number of CPUs for SMP	1
# default_mpp_cpus	integer	Default Number of CPUs for MPP	1
# max_node_cpus	integer	Maximum number of CPUs per node for MPP jobs	2
# pack_nodes	logical	Pack MPP jobs on the minimum number of nodes	TRUE
# submit_shell	string	CSHELL or BSHELL	CSHELL
# unix_type	string	SYSTEM5 or BSD	SYSTEM5
# manual_reader	string	location of ADOBE ACROBAT reader	acroread
# enable_job_monitoring	logical	TRUE or FALSE	TRUE
# temp_folder	string	Location to write .bat file to on a PC	C:\TEMP
# max_hybrid_mpp_threads	integer	Maximum number of MPP threads for Hybrid Dyna	4
# max_hybrid_smp_threads	integer	Maximum number of SMP threads for Hybrid Dyna	4
# default_hybrid_mpp_threads	integer	Default number of MPP threads for Hybrid Dyna	1
# default_hybrid_smp_threads	integer	Default number of SMP threads for Hybrid Dyna	1
# case_option	logical	case option	FALSE
# mcheck_option	logical	mcheck option	FALSE
# xterm_start	logical	Start programs from an xterm console	TRUE
# default_code_type	string	SMP, MPP or HYBRID	MPP
# default_precision	string	SINGLE or DOUBLE	SINGLE
# write_reporter	logical	Write a Reporter file as output from Reporter	FALSE
# write_html	logical	Write an HTML file as output from Reporter	FALSE
# write_pdf	logical	Write a PDF file as output from Reporter	TRUE
# write_vba	logical	Write a VBA file as output from Reporter	FALSE
# write_ppt	logical	Write a PPT file as output from Reporter	FALSE

Preferences marked with a '*' can be read from all four directories. If desired they can still be locked in the [OA_ADMIN_xx](#) or [OA_INSTALL](#) directories using the 'shell#' syntax:

```

shell*online: TRUE
shell*background: TRUE
shell*batch: TRUE
shell*queue: TRUE
shell*default: ONLINE
shell*dyna keyword manual: (*)
shell*dyna theory manual: (*)
shell*batch type: NQS
shell*queue output file: TRUE
shell*queue error file: TRUE
shell*queue error path: TRUE
shell*queue memory: 15000000
shell*queue memory units: WORDS
shell*queue cpu: 0
shell*queue cpu units: SECONDS
shell*batch type: NQS
shell*queue space: 1024
shell*queue machine:
shell*request cpu limit: TRUE
shell*request memory limit: TRUE
shell*request space limit: FALSE
shell*system cpu: 90
shell*job format: KEYWORD

```

```

shell*file names: ARUP (*)
shell*job memory: 9000000
shell*job memory units: WORDS
shell*job cpu: 0
shell*job cpu units: SECONDS
shell*allow mem change: TRUE
shell*file size: 1024 (*)
shell*write ctf: FALSE (*)
shell*write ztf: TRUE (*)
shell*batch display: (*)
shell*keep files: FALSE (*)
shell*d3plot version: d3plot11.exe (*)
shell*d3plot64 version: d3plot11 64.exe (*)
shell*d3plot default: 32 (*)
shell*this version: this11.exe (*)
shell*this64 version: this11 64.exe (*)
shell*this default: 32 (*)
shell*primer version: primer11.exe (*)
shell*primer64 version: primer11 64.exe (*)
shell*primer default: 32 (*)
shell*reporter version: reporter11.exe (*)
shell*reporter64 version: reporter11 64.exe (*)
shell*reporter default: 32 (*)>
shell*shell release: 12.0
shell*max cpus: 1>shell*max mpp cpus: 1
shell*default cpus: 1 (*)
shell*default mpp cpus: 1 (*)
shell*max node cpus: 2
shell*pack nodes: TRUE
shell*submit shell: CSHELL
shell*unix type: SYSTEM5
shell*manual reader: acroread (*)
shell*temp folder: C:\TEMP (*)
shell*max hybrid mpp threads: 1
shell*max hybrid smp threads: 1
shell*default hybrid mpp threads: 1
shell*default hybrid smp threads: 1
shell*case option: FALSE
shell*mcheck option: FALSE
shell*xterm start: TRUE
shell*default code type: MPP (*)
shell*default precision: SINGLE (*)
shell*write reporter: FALSE (*)
shell*write html: FALSE (*)
shell*write pdf: TRUE (*)
shell*write vba: FALSE (*)
shell*write ppt: FALSE (*)

```

2.3.1 shell*online / shell*background / shell*batch / shell*queue

The XSHELL can submit LS-DYNA jobs using four different options. Any of these methods can be made inaccessible by setting the relevant variable to **FALSE**. If an option is not available on a machine then the variable should also be set to **FALSE**.

- shell*online** - Allows jobs to be submitted interactively. Delayed start times cannot be specified.
- shell*background** - Allows jobs to be submitted to background. If a delayed start times is specified then the job will be submitted using the LINUX 'at' command.
- shell*batch** - Allows jobs to be submitted using the LINUX 'batch' command. Delayed start times cannot be specified.
- shell*queue** - Allows jobs to be submitted to NQE style batch queues using 'qsub' or equivalent commands. Delayed start times,CPU/Memory limits and accounts may be specified.

If **shell*queue** is set to **FALSE** then sections 2.3.3 to 2.3.10 can be ignored.

2.3.2 shell*default

This variable controls the default option that will be used to submit LS-DYNA jobs. It should be set to either **ONLINE**, **BACKGROUND**, **BATCH**, or **QUEUE** to select the required default.

2.3.3 shell*dyna_keyword_manual

This variable should be set to the location of the Dyna keyword PDF manuals. Set the full pathname. Up to four PDFs can be specified (e.g. if there are separate volumes) by separating the pathnames with a comma.

2.3.4 shell*dyna_theory_manual

This variable should be set to the location of the Dyna theory PDF manual. Set the full pathname.

2.3.5 shell*batch_type

This variable controls the type of queue the **QUEUE** command (see [shell*online](#) / [shell*background](#) / [shell*batch](#) / [shell*queue](#)) submits the job to. At present this variable may be set to **NQS**, **LSF**, **SGE** (SUN GRID ENGINE), **PBS**, **PBSPRO7** and **PBSPRO71**. Due to the limitations of some of the queuing systems not all options are available with all queue types, see table below.

Function	NQS	PBS / PBSPRO7 / PBSPRO71	LSF	SGE
Analysis Start Time	Available	Available	Available	Unavailable
Analysis Start Day	Available	Available	Unavailable	Unavailable

2.3.6 shell*queue_output_file

This variable controls if the Shell writes a queue directive to specify a file for output when a job is submitted to a queueing system. If this option is set to **TRUE** then the queue will add the following line to the top of the job submission script.

Queue Type	Command
NQS	#QSUB -o " <i>job_directory</i> "/" <i>jobname</i> ".log
PBS / PBSPRO7/ PBSPRO71	#PBS -o " <i>job_directory</i> "/" <i>jobname</i> ".log
SGE	#QSUB -o " <i>job_directory</i> "/" <i>jobname</i> ".log
LSF	#BSUB -o " <i>job_directory</i> "/" <i>jobname</i> ".log

2.3.7 shell*queue_error_file

These variable controls if the Shell writes a queue directive to specify a file for errors when a job is submitted to a queueing system. If this option is set to **TRUE** then the queue will add the following line to the top of the job submission script.

Queue Type	Command
NQS	#QSUB -eo #QSUB -o " <i>job_directory</i> "/" <i>jobname</i> ".err
PBS / PBSPRO7/ PBSPRO71	#PBS -e " <i>job_directory</i> "/" <i>jobname</i> ".err
SGE	#QSUB -e " <i>job_directory</i> "/" <i>jobname</i> ".err
LSF	#BSUB -e " <i>job_directory</i> "/" <i>jobname</i> ".err

2.3.8 shell*queue_error_path

This option can be used to specify if the queue directives for the [output](#) and [error](#) files should contain the job path or not. By default this option is set to **TRUE**.

2.3.9 shell*queue_memory

This variable controls the value used for the default Queue Memory limit. The required value should be specified using the units defined by the **shell*queue_memory_units** variable (see [Queue Memory Units](#)). A value of 0 means that the job will be submitted with a unlimited Queue Memory.

2.3.10 shell*queue_memory_units

This variable controls whether the default Queue Memory limit is specified in words or megawords. It should be set to either **WORDS** or **MWORDS** respectively.

2.3.11 shell*queue_cpu

This variable controls the value used for the default Queue CPU Limit. The required value should be specified using the units defined by the **shell*queue_cpu_units** variable (see [Queue Cpu Units](#)). A value of 0 means that the job will be submitted with a unlimited Queue CPU.

2.3.12 shell*queue_cpu_units

This variable controls whether the default Queue CPU Limit is specified in seconds, minutes or hours. It should be set to either **SECONDS**, **MINUTES** or **HOURS** respectively.

2.3.13 shell*queue_space

This variable controls the value used for the default File Space limit. The required value should be specified in Mbytes. A value of 0 means that the job will be submitted with an unlimited File Space Limit.

2.3.14 shell*queue_machine

This variable may be used to specify the name of the queuing machine used by **LSF** batch queues. This option is passed as the **-m** parameter to the LSF **bsub** command.

e.g. **bsub -m queue_machine**

2.3.15 shell*request_cpu/memory_limit

These variables control whether Queue CPU and Memory limits have to be specified when jobs are submitted to NQS style queues. If they do have to be specified then set the relevant variable to **TRUE**, otherwise set them to **FALSE**.

2.3.16 shell*request_space_limit

This variable controls whether a Per-request File Space limit has to be specified when jobs are submitted to NQS style queues. If they do have to be specified then set this variable to **TRUE**, otherwise set it to **FALSE**.

2.3.17 shell*system_cpu

This variable controls the minimum amount of 'system' time, in seconds, that a job submitted to a NQS style queue will require if the queue has been set up using a Queue CPU Limit, (see [Request Cpu Limit/ Request Memory Limit](#)). When a job is submitted to such a queue a 'system' CPU time will be calculated as either 2.5% of the Analysis CPU time or the value of this variable. If the difference between the Queue CPU limit and the Analysis CPU limit is smaller than this value the Queue CPU limit will automatically be increased to allow enough system time for the job to terminate normally when the Analysis CPU limit has been reached.

2.3.18 shell*job_format

This variable controls the default file format that will be expected by LS-DYNA. It can be set to either KEYWORD or FIXED.

2.3.19 shell*file_names

This variable controls the names of the output files generated by LS-DYNA. This variable can be set to either ARUP or LSTC.

With LS-DYNA 970 onwards the names of the output files can also be specified using the *KEYWORD_ID option where a filename prefix for all of the output files is specified within the input file.

Output File	ARUP	LSTC	*KEYWORD_ID
Binary Complete State Database	'jobname'.ptf	d3plot	'prefix'.d3plot
Time History Database	'jobname'.thf	d3thdt	'prefix'.d3thdt
Extra Time History Database	'jobname'.xtf	xtfile	'prefix'.xtfile
Binary Output File	binout	binout	'prefix'.binout
Restart Dump File	'jobname'.dpf	d3dump	'prefix'.d3dump
Running Restart Dump File	'jobname'.adf	runrsf	'prefix'.runrsf

2.3.20 shell*job_memory

This variable controls the value used for the default Analysis Memory limit, the amount of **CORE** memory that LS-DYNA will use to store data. The required value should be specified using the units defined by the shell*job_memory_units variable (see [Job Memory Units](#)). A value of 0 means that the job will be submitted with a the default amount of **CORE** memory built into that LS-DYNA. The default value of 9000000 should be enough for models of up to 60000 elements.

2.3.21 shell*job_memory_units

This variable controls whether the default Analysis Memory limit is specified in words or megawords. It should be set to either **WORDS** or **MWORDS** respectively.

2.3.22 shell*job_cpu

This variable controls the value used for the default Analysis CPU limit. The required value should be specified using the units defined by the shell*job_cpu_units variable (see [Job Cpu Units](#)). A value of 0 means that the job will be submitted with no Analysis CPU limit.

2.3.23 shell*job_cpu_units

This variable controls whether the default Analysis CPU limit is specified in seconds, minutes or hours. It should be set to either **SECONDS**, **MINUTES** or **HOURS** respectively.

2.3.24 shell*allow_mem_change

This variable controls whether users can change the amount of **CORE** memory used by LS-DYNA. If this variable is set to **FALSE** then users will not be able to modify the Analysis Memory Limit and will be forced to use the default value built into LS-DYNA.

2.3.25 shell*file_size

This variable controls the default binary file size produced by LS-DYNA. The value specified gives the required file size in Mbytes.

2.3.26 shell*write_ctf

By default the '**xshell_12**' is set up so that the default is for LS-DYNA to write out CTF binary files when a jobs is submitted (see the [Job Options menu](#)). If this variable is set to **FALSE** then this option will be turned off by default.

2.3.27 shell*write_ztf

By default the '**xshell_12**' is set up so that after an LS-DYNA run PRIMER is automatically run to generate a .ZTF file for D3PLOT to read (see the [Job Options menu](#)). If this variable is set to **FALSE** then this option will be turned off by default.

2.3.28 shell*batch_display

This variable can be used to specify a LINUX DISPLAY that can be used for batch processing commands.

2.3.29 shell*keep_files

This variable controls whether temporary files created during the submission process and any **CORE** files created if a job terminates abnormally are deleted automatically. If this variable is set to **TRUE** the files will not be deleted.

2.3.30 shell*<*****>_version

By default the '**xshell_12**' is set up to access and run software that uses the default program names. This option allows the name of any executable to be modified as required and for the '**xshell_12**' to be notified of this name change.

e.g. **shell*d3plot_version: d3plot90.exe**

would force the version 11 Shell to run the version 9.0 copy of d3plot, (**d3plot90.exe** would have to be in the directory containing the version 11 executables).

On machines where both 32 and 64 bit versions of D3PLOT, T/HIS, PRIMER and REPORTER are available the names for the 64 bit versions can be modified using

shell*this64_version:
shell*d3plot64_version:
shell*primer64_version:
shell*reporter64_version:

2.3.31 shell*<*****>_default

These options can be used to set the default versions of T/HIS, D3PLOT, PRIMER and REPORTER to either the 32 or 64 bit version.

2.3.32 shell*shell_release

By default the '**xshell_12**' is set up to display 12.0 as the version. This option allows the version number displayed to be changed if a customised installation is set up

2.3.33 shell*max_cpus/shell*max_mpp_cpus

These variable controls the maximum number of CPU's that can be selected for SMP and MPP parallel jobs.

2.3.34 shell*default_cpus/shell*default_mpp_cpus

These variable controls the default number of CPU's for SMP and MPP parallel jobs.

2.3.35 shell*max_node_cpus

This variable should be set to the number of CPUS each node has in a cluster. When a MPP job is submitted to a PBS queue this variable will be used to calculate the required number of nodes that will be requested from the PBS queuing system.

2.3.36 shell*pack_nodes

This variable is used along with [shell*max_node_cpus](#) to calculate the number of nodes required when submitting a MPP job to a PBS queuing system. If this is set to TRUE then the minimum number of nodes will be requested.

Note that if commands are specified in the "[oasys_queue](#)" file then the following defaults will be overridden.

# Job CPU's	max_node_cpus	pack_nodes	PBS node request	PBSPro 7.0 node request	PBSPro 7.1 node request
2	2	TRUE	nodes=1:ppn=2 (1x2)	nodes=1:ppn=1:ncpus=2	select=1:ncpus=2
2	2	FALSE	nodes=2:ppn=1 (2x1)	nodes=2:ppn=1:ncpus=1	select=2:ncpus=1
8	2	TRUE	nodes=4:ppn=2 (4x2)	nodes=4:ppn=1:ncpus=2	select=4:ncpus=2
8	2	FALSE	nodes=8:ppn=1 (8x1)	nodes=8:ppn=1:ncpus=1	select=8:ncpus=8
7	4	TRUE	nodes=1:ppn=4+1:ppn=3 (1x4 + 1x3)	nodes=1:ppn=1:ncpus=4+1:ppn=1:ncpus=3	select=1:ncpus=4+1:ncpus=3
7	4	FALSE	nodes=7:ppn=1 (7x1)	nodes=7:ppn=1:ncpus=1	select=7:ncpus=1

2.3.37 shell*submit_shell (LINUX only)

LS-DYNA jobs are submitted on LINUX systems using script files. This variable can be set to either **CSHELL** or **BSHELL** to create either CSHELL or BOURNE shell scripts.

2.3.38 shell*unix_type

This variable controls the default binary file size produced by LS-DYNA. The value specified gives the required file size in Mbytes.

2.3.39 shell*manual_reader (Linux only)

This option should be set to point to where a copy of the Adobe Acrobat reader software can be found on the system.

2.3.40 shell*temp_folder (Windows only)

This option can be used to specify a folder that the Shell can use to create temporary files while submitting LS-DYNA jobs. The directory should be on a local system disk not a networked disk as some Anti-Virus programs block running DOS batch files from Network Drives. The default folder is **C:\TEMP**

2.3.41 shell*max_hybrid_mpp_threads

This option controls the maximum number of MPP threads that can be selected for Hybrid jobs.

2.3.42 shell*max_hybrid_smp_threads

This option controls the maximum number of SMP threads that can be selected for Hybrid jobs.

2.3.43 shell*default_hybrid_mpp_threads

This option controls the default number of MPP threads that can be selected for Hybrid jobs.

2.3.44 shell*default_hybrid_smp_threads

This option controls the default number of SMP threads that can be selected for Hybrid jobs.

2.3.45 shell*case_option

This option can be used to specify whether the -case option is used in the command line when submitting a job.

2.3.46 shell*mcheck_option

This option can be used to specify whether the -mcheck option is used in the command line when submitting a job.

2.3.47 shell*xterm_start (Linux only)

This option can be used to specify whether the programs are started from an xterm console or not.

2.3.48 shell*default_code_type

This option can be used to specify the default program code type.

2.3.49 shell*default_precision

This option can be used to specify the default program precision

2.3.50 shell*write_reporter

This option can be used to specify whether to output a Reporter file from Reporter

2.3.51 shell*write_html

This option can be used to specify whether to output an HTML file from Reporter

2.3.52 shell*write_pdf

This option can be used to specify whether to output a PDF file from Reporter

2.3.53 shell*write_vba

This option can be used to specify whether to output a VBA file from Reporter

2.3.54 shell*write_ppt

This option can be used to specify whether to output a PPT file from Reporter

2.4 Adding LS-DYNA versions to the Shell

The list of LS-DYNA versions that can be accessed through the Shell is controlled through an ASCII file called "dyna_versions".

Note the format of this file was changed in version 9.3 and an additional column has been inserted before the label field for MPP codes which identifies which MPI library the version has been compiled with.

The format for SMP versions of LS-DYNA is :

<Code Type> <precision> <executable> <label>

The format for MPP or HYBRID versions of LS-DYNA is :

<Code Type> <precision> <executable> <MPP Type> <label>

Where:

- <Code Type> Is either **SMP** (shared memory parallel), **SERIAL** (single threaded non parallel), **MPP** (distributed memory parallel) or **HYBRID**
- <precision> Is either **Single** (32 bit floating point) or **Double** (64 bit floating point).
- <executable> Is the full path of the executable. If the full path name of the executables includes any spaces then the path must be enclosed in "s.
- <MPP Type> This is a string that can be used to identify the MPP library a MPP version of LS-DYNA has been compiled with. This column is not defined for **SMP** or **SERIAL** codes,
- <label> Is the label shown in the list of LS-DYNA versions displayed in the submission Shell

2.4.1 LINUX

Code Type	precision	executable	MPP Type	label
SMP	Single	/dyna/ls971_s_R3_1_xeon64_redhat40.exe		LS971 R3.1 SP SMP
SMP	Double	/dyna/ls971_d_R3_1_xeon64_redhat40.exe		LS971 R3.1 DP SMP
MPP	Single	/dyna/mp971_s_R3.1_..._hpmpi.exe	HP-MPI	LS971 R3.1 SP MPP HP-MPI

MPP	Double	/dyna/mpp971_d_R3.1_Intel_..._hpmpi.exe	HP-MPI	LS971 R3.1 DP MPP HP-MPI
MPP	Single	/dyna/mpp971_s_R3.1_Intel_..._lam659.exe	LAM659	LS971 R3.1 SP MPP LAM 6.5.9
MPP	Double	/dyna/mpp971_d_R3.1_Intel_..._lam659.exe	LAM659	LS971 R3.1 DP MPP LAM 6.5.9
HYBRID	Single	/dyna/mpp971_s_R5.1.1_65550_..._hybrid.exe	HP-MPI	LS971 R5.1.1 SP HYBRID HP-MPI
HYBRID	Double	/dyna/mpp971_d_R5.1.1_65550_..._hybrid.exe	HP-MPI	LS971 R5.1.1 DP HYBRID HP-MPI

Would display a menu containing 8 versions of LS-DYNA.

2.4.2 WINDOWS

Code Type	precision	executable	MPP Type	label
SMP	Single	"C:\LS-DYNA\ls971_s_R3_1_win32_p.exe"		Win32 SP LS971vR3.1
SMP	Double	"C:\LS-DYNA\ls971_d_R3_1_win32_p.exe"		Win32 DP LS971vR3.1
MPP	Single	"C:\LS-DYNA\mpp971_s_R3.1_win32_mpich125.exe"	MPICH	Win32 SP LS971vR3.1 (MPICH)
MPP	Double	"C:\LS-DYNA\mpp971_d_R3.1_win32_mpich125.exe"	MPICH	Win32 DP LS971vR3.1 (MPICH)
MPP	Single	"C:\LS-DYNA\mpp971_s_R3.1_win64_hpmpi.exe"	HPMPI	Win64 SP LS971vR3.1 (HPMPI)
MPP	Double	"C:\LS-DYNA\mpp971_d_R3.1_win64_hpmpi.exe"	HPMPI	Win64 SP LS971vR3.1 (HPMPI)

If MPP jobs are going to be run across multiple machines and the LS-DYNA executables are located on a network disk then the full UNC path should be specified instead of the local drive letter.

Code Type	precision	executable	MPP Type	label
MPP	Single	"\\SERVER\LS-DYNA\mpp971_s_R3.1_win64_hpmpi.exe"	HPMPI	Win64 SP LS971vR3.1 (HPMPI)
MPP	Double	"\\SERVER\LS-DYNA\mpp971_d_R3.1_win64_hpmpi.exe"	HPMPI	Win64 SP LS971vR3.1 (HPMPI)

2.5 Customising the LS-DYNA submission script

The Shell generates either a LINUX CShell/Bourne Shell script (`jobname.ct1`) or a DOS batch file (`jobname.bat`) that contains all of the selected LS-DYNA job options. On LINUX systems the choice of either CShell or Bourne Shell is controlled by the [shell*submit_shell](#) option in the `oa_pref` file.

In version 12.0 of the Shell the contents of these batch files can be modified by editing the "oasys.submit" file located in the installation directory.

The batch file created by the submission Shell is split into 4 sections

	Contents	
Queuing Commands	Queue commands for NQS, SGE, LSF and PBS queuing systems. If the job is not submitted to a queue then this section of the batch file will be empty.	Automatically generated by the Shell
Environment Variables	This section of the batch file sets up a number of Environment variables. The first group contain information relating to license systems while the second set contain information relating to the selected version of LS-DYNA and the job.	Automatically generated by the Shell
LS-DYNA submission	This section of the batch file contains the commands that are actually used to run LS-DYNA. The contents of this section are included from a file in the installations directory.	User configurable
Post Processing Options	Commands for running T/HIS and Reporter automatically after the LS-DYNA job terminates.	Automatically generated by the Shell

The following sections include examples taken from the batch file created for a job submitted to a PBS queuing system on a LINUX machine

2.5.1 Queuing Commands

The Shell will automatically add queuing commands to the start of the batch file. The options supported for the different queuing systems are as follows.

Option	NQS	PBS	SGE	LSF
log file	-o "logfile"	-o "logfile"	-o "logfile"	-o "logfile"
error file	-eo	-e "errorfile"	-e "errorfile"	-e "errorfile"
queue name	-q "queuenamename"	-q "queuenamename"	-l g="queuenamename"	-q "queuenamename"
start time	-a "start time"	-a "start time"		-b "start time"
Memory Limit	-lM "limit"			-M "limit"
CPU Limit	-lT "limit"		-l h_cpu="limit"	-c "limit"
File Limit	-lF "limit"			

The log and error file output options can be disabled by setting the preference options [shell*queue_output_file](#) and [shell*queue_error_file](#) to **FALSE**.

2.5.2 Environment Variables

The following environment variables are automatically set up in the batch file by the Shell.

Variable	Description
<u>OA_INSTALL</u>	Full pathname of the software installation directory
<u>LSTC_FILE</u>	Full pathname of the LSTC license file. If a LSTC license server is being used then this variable will be replaced with the LSTC_LICENSE_SERVER, LSTC_NOCLIENT and LSTC_LICENSE variables.
LSTC_MEMORY	Set to auto if the automatic memory option has been selected
<u>LM_LICENSE_FILE</u>	FLEXlm license information for software suite
LSDYNA_VERSION	Full pathname of the selected LS-DYNA version
LSDYNA_MPP	Set to TRUE if a MPP version of LS-DYNA had been selected
LSDYNA_HYBRID	Set to TRUE if a HYBRID version of LS-DYNA had been selected
LSDYNA_HYBRID_MPP_THREADS	Number of HYBRID MPP threads selected
LSDYNA_HYBRID_SMP_THREADS	Number of HYBRID SMP threads selected
LSDYNA_MPI_TYPE	MPI version from "dyna_versions" file
LSDYNA_DOUBLE	Set to TRUE if a double precision version of LS-DYNA had been selected
LSDYNA_ONLINE	Set to TRUE if the job has been submitted using the Online option.
LSDYNA_BACKGROUND	Set to TRUE if the job has been submitted using the Background option.
LSDYNA_BATCH	Set to TRUE if the job has been submitted using the Batch option.
LSDYNA_QUEUE	Set to TRUE if the job has been submitted using the Queue option.
LSDYNA_QUEUE_NAME	Name of the QUEUE selected
LSDYNA_QUEUE_COMMAND_<n>	Queue commands used (where <n> = 1-20)
LSDYNA_JOB_DIR	Full pathname of the directory containing the LS-DYNA job
LSDYNA_JOB_FILE	Name of the file containing the LS-DYNA job information
LSDYNA_JOB_NAME	Name of the LS-DYNA job
LSDYNA_JOB_CPUS	Number of CPU's selected for parallel jobs
LSDYNA_USER_ID	Username
LSDYNA_LOCAL_HOST	TRUE if submitting MPP jobs to the local machine
LSDYNA_NODE_FILE	Set to TRUE if submitting MPP jobs using a file containing the list of nodes to use
NODE_FILE	Filename containing the list of nodes to use.
LSDYNA_NODE_LIST	Set to TRUE if submitting MPP jobs using a string containing the list of nodes to use
NODE_LIST	String containing the list of nodes to use.
LSDYNA_RESTART_FILE	Filename of LS-DYNA restart dump file if the analysis is a restart.
LSDYNA_RESTART_INPUT_FILE	Filename of LS-DYNA restart input file if the analysis is a restart and an input file has been selected.
LSDYNA_CL_ARGS	List of command line arguments used. This will only be set if the CASE or MCHECK options have been selected.

```
#
# License file variables
#
setenv OASYS /data/dyna10/linux_executables
setenv LSTC_FILE /prg/LSTC_FILE
setenv LSTC_MEMORY auto
setenv LM_LICENSE_FILE @atuhp022:@atghps03
#
# Job variables
#
setenv LSDYNA_VERSION
/data/dyna10/linux_executables/ls970_s_5434_ia64_sgi_p.exe
setenv LSDYNA_MPP FALSE
setenv LSDYNA_DOUBLE FALSE
setenv LSDYNA_ONLINE TRUE
setenv LSDYNA_BACKGROUND FALSE
setenv LSDYNA_BATCH FALSE
setenv LSDYNA_QUEUE FALSE
setenv LSDYNA_JOB_DIR /local/test
setenv LSDYNA_JOB_FILE shell_test.temp
setenv LSDYNA_JOB_CPUS 2
setenv LSDYNA_USER_ID rogerh
setenv LSDYNA_LOCAL_HOST FALSE
setenv LSDYNA_NODE_FILE TRUE
setenv NODE_FILE /local/test/nodelist
setenv LSDYNA_NODE_LIST FALSE
#
```

2.5.3 LS-DYNA submission

The commands to run LS-DYNA are included from a user configurable file called "oasys.submit" located in the software installation directory.

As the file is included after the Environment Variable section of the batch file all of the variables that are set up can be used within the include file. This means that a single include file "oasys.submit" can be configured if required which contains separate sections for SMP / MPP jobs.

2.5.3.1 Example of a LINUX "oasys.submit" file

The following example shows the default "oasys.submit" file that is included with the version 12.0 software on LINUX platforms. The default script is setup to submit jobs on a LINUX machine using the PBS queueing system but it should be easy to reconfigure to other machines and queueing systems.

As MPP jobs can use a number of different MPI libraries on LINUX systems this script shows how to submit jobs using HP-MPI, INTEL MPI and MPICH. The default script is setup assuming that the different MPI libraries have been installed in the following directories:

MPI	Directory
HP-MPI	/opt/hpmpi
INTEL MPI	/opt/intel
OPEN MPI	/opt/openmpi

```
#
# The following Environment Variables are available for this script
#
# $LSDYNA_VERSION : full pathname of selected LS-DYNA executable
# $LSDYNA_MPP : TRUE if MPP version selected
# $LSDYNA_MPI_TYPE : MPI version
# $LSDYNA_DOUBLE : TRUE if double precision version selected
# $LSDYNA_ONLINE : TRUE if the job has been submitted ONLINE
# $LSDYNA_BACKGROUND : TRUE if the job has been submitted to BACKGROUND
# $LSDYNA_BATCH : TRUE if the job has been submitted using BATCH
# $LSDYNA_QUEUE : TRUE if the job has been submitted to a QUEUE
# $LSDYNA_JOB_DIR : full path of LS-DYNA job_directory
```

```

# $LSDYNA_JOB_FILE : filename containing LS-DYNA job options
# $LSDYNA_JOB_CPUS : number of CPU's selected
# $LSDYNA_USER_ID : username
# $LSDYNA_LOCAL_HOST : TRUE if submitting MPP jobs to the local machine
# $LSDYNA_NODE_FILE : TRUE if submitting MPP jobs using a Node File
# $NODE_FILE : filename containing node list for MPP jobs
# $LSDYNA_LOCAL_HOST : TRUE if submitting MPP jobs using a Node List
# $NODE_LIST : string containing node list for MPP jobs
# $LSDYNA_CL_ARGS : string containing command line arguments
# $LSDYNA_HYBRID : TRUE if HYBRID version selected
# $LSDYNA_MPP_THREADS : Number of MPP threads selected for the HYBRID version
# $LSDYNA_SMP_THREADS : Number of SMP threads selected for the HYBRID version
#
#
# =====
#
# THE REST OF THIS SCRIPT HAS BEEN CONFIGURED FOR SUBMITTING LS-DYNA JOBS TO A
# LINUX SYSTEM USING
# THE PBS QUEUEING SYSTEM. YOU WILL HAVE TO MODIFY THE FOLLOWING IF YOU ARE
# USING A DIFFERENT
# QUEUEING SYSTEM OR MACHINE TYPE.
#
# =====
#
# Uncomment the following commands if you want to create sub-directories for
# your analyses.
# This will only work if you are using the PBS queueing system, as the folders
# are named
# after the PBS job id.
#
# Make a new directory for the results - only uncomment one of these lines
# The first one will give the full job id, the second just the job id number
#
#set RESULTS_DIR = $PBS_JOBID
#set RESULTS_DIR = `echo $PBS_JOBID | awk '{split($0,a,"."); print a[1]}'`
#
#mkdir $LSDYNA_JOB_DIR/$RESULTS_DIR
#
# Get the root name of the job
#
#set ROOT = $LSDYNA_JOB_FILE:r
#
# Copy the .temp, names and key file to the results directory
#
#set KEY = ".key"
#set NAMES = "names"
#
#cp $LSDYNA_JOB_DIR/$ROOT$KEY $RESULTS_DIR/.
#cp $LSDYNA_JOB_DIR/$LSDYNA_JOB_FILE $RESULTS_DIR/.
#cp $LSDYNA_JOB_DIR/$NAMES $RESULTS_DIR/.
#
# Change job directory to the results directory
#
#set LSDYNA_JOB_DIR = $RESULTS_DIR
#cd $LSDYNA_JOB_DIR
#
# if ($LSDYNA_MPP != "TRUE" && $LSDYNA_HYBRID != "TRUE") then
#
# =====
# SMP LS-DYNA submission
# =====
#
#   $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
# else if ($LSDYNA_MPP == "TRUE") then
#
# =====
# MPP LS-DYNA submission
# =====

```

```

#
# HP-MPI
#
# if ($LSDYNA_MPI_TYPE == "HP-MPI") then
#
#     setenv HPMPI_DIR /opt/hpmpi/bin
#
# Submit a job to a PBS queueing system
#
#     if ($LSDYNA_QUEUE == "TRUE") then
#
#         if (-e $PBS_NODEFILE) then
#             rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
#             if (-f $PBS_NODEFILE) then
#                 foreach roger ( `cat $PBS_NODEFILE` )
#                     echo "-h $roger -np 1 $LSDYNA_VERSION $LSDYNA_CL_ARGS" >>
$LSDYNA_JOB_DIR/appfile
#                     set LSDYNA_CL_ARGS = ""
#                 end
#             endif
#         endif
#         $HPMPI_DIR/mpirun -prot -e MPI_WORKDIR=$LSDYNA_JOB_DIR -f appfile
#         rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
#
# Submit a job online
#
#     else if ($LSDYNA_ONLINE == "TRUE") then
#
#         if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
#             $HPMPI_DIR/mpirun -prot -np $LSDYNA_JOB_CPUS $LSDYNA_VERSION
$LSDYNA_CL_ARGS
#
#         else if ($LSDYNA_NODE_FILE == "TRUE") then
#
#             $HPMPI_DIR/mpirun -prot -np $LSDYNA_JOB_CPUS -hostfile $NODE_FILE -e
LSTC_LICENSE_SERVER=$LSTC_LICENSE_SERVER -e LSTC_LICENSE=$LSTC_LICENSE
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#         else if ($LSDYNA_NODE_LIST == "TRUE") then
#
#             $HPMPI_DIR/mpirun -prot -np $LSDYNA_JOB_CPUS -hostlist $NODE_LIST -e
LSTC_LICENSE_SERVER=$LSTC_LICENSE_SERVER -e LSTC_LICENSE=$LSTC_LICENSE
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#         endif
#
#     endif
#
# INTEL MPI
#
# else if ($LSDYNA_MPI_TYPE == "INTEL") then
#
#     setenv INTEL_DIR /opt/intel/impi/4.0.3/bin64
#     set mpi_dir=$INTEL_DIR
#     source $mpi_dir/mpivars.csh
#
#     if ($LSDYNA_QUEUE == "TRUE") then
#
#         $INTEL_DIR/mpirun -n $LSDYNA_JOB_CPUS -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#     else if ($LSDYNA_ONLINE == "TRUE") then
#
#         if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
#             $INTEL_DIR/mpirun -n $LSDYNA_JOB_CPUS -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#         else if ($LSDYNA_NODE_FILE == "TRUE") then
#

```

```

        $INTEL_DIR/mpirun -n $LSDYNA_JOB_CPUS -machinefile $NODE_FILE -wdir
$LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
        else if ($LSDYNA_NODE_LIST == "TRUE") then
#
        $INTEL_DIR/mpirun -n $LSDYNA_JOB_CPUS -hosts $NODE_LIST -wdir
$LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
        endif
#
    endif
#
# OPEN MPI
#
    else if ($LSDYNA_MPI_TYPE == "OPENMPI") then
#
        setenv MPI_ROOT /opt/openmpi
        setenv OPENMPI_DIR $MPI_ROOT/bin
#
        if !($?PATH) then
            setenv PATH $MPI_ROOT/bin
        else
            setenv PATH $MPI_ROOT/bin:$PATH
        endif
        if !($?LD_LIBRARY_PATH) then
            setenv LD_LIBRARY_PATH $MPI_ROOT/lib
        else
            setenv LD_LIBRARY_PATH $MPI_ROOT/lib:$LD_LIBRARY_PATH
        endif
#
        if ($LSDYNA_QUEUE == "TRUE") then
#
            if (-e $PBS_NODEFILE) then
                rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
                if (-f $PBS_NODEFILE) then
                    foreach host ( `cat $PBS_NODEFILE` )
                        echo "$host" >> $LSDYNA_JOB_DIR/appfile
                    end
                endif
            endif
            $OPENMPI_DIR/mpirun -machinefile appfile -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
            else if ($LSDYNA_ONLINE == "TRUE") then
#
                if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
                    $OPENMPI_DIR/mpirun -n $LSDYNA_JOB_CPUS -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
                    else if ($LSDYNA_NODE_FILE == "TRUE") then
#
                        $OPENMPI_DIR/mpirun -n $LSDYNA_JOB_CPUS -machinefile $NODE_FILE -x PATH
-x LD_LIBRARY_PATH -x LSTC_LICENSE_SERVER -x LSTC_LICENSE -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
                        else if ($LSDYNA_NODE_LIST == "TRUE") then
#
                            $OPENMPI_DIR/mpirun -n $LSDYNA_JOB_CPUS -host $NODE_LIST -x PATH -x
LD_LIBRARY_PATH -x LSTC_LICENSE_SERVER -x LSTC_LICENSE -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
                            endif
#
                        endif
#
                    endif
#
                endif
#
            else if ($LSDYNA_HYBRID == "TRUE") then
#
# =====

```

```

# HYBRID LS-DYNA submission
# =====
#
# HP-MPI
#
# if ($LSDYNA_MPI_TYPE == "HP-MPI") then
#
#   setenv HPMPI_DIR /opt/hpmmpi/bin
#
#   if ($LSDYNA_QUEUE == "TRUE") then
#
#     if (-e $PBS_NODEFILE) then
#       rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
#       if (-f $PBS_NODEFILE) then
#         set last_host = ""
#         foreach_host ( `sort $PBS_NODEFILE` )
#           if ( $host != $last_host ) then
#             set count = 0
#           endif
#           @ count = $count + 1
#           if ( $count == 1 ) then
#             echo "-h $host -np 1 $LSDYNA_VERSION" $LSDYNA_CL_ARGS >>
$LSDYNA_JOB_DIR/appfile
#             set LSDYNA_CL_ARGS = ""
#           endif
#           if ($count == $LSDYNA_HYBRID_SMP_THREADS) then
#             set count = 0
#           endif
#           set last_host = $host
#         end
#       endif
#     endif
#     $HPMPI_DIR/mpirun -cpu_bind -prot -e MPI_WORKDIR=$LSDYNA_JOB_DIR -f
appfile
#     rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
#
#     else if ($LSDYNA_ONLINE == "TRUE") then
#
#       if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
#         $HPMPI_DIR/mpirun -prot -np $LSDYNA_HYBRID_MPP_THREADS $LSDYNA_VERSION
$LSDYNA_CL_ARGS
#
#       else if ($LSDYNA_NODE_FILE == "TRUE") then
#
#         $HPMPI_DIR/mpirun -prot -np $LSDYNA_HYBRID_MPP_THREADS -hostfile
$NODE_FILE -e LSTC_LICENSE_SERVER=$LSTC_LICENSE_SERVER -e
LSTC_LICENSE=$LSTC_LICENSE $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#       else if ($LSDYNA_NODE_LIST == "TRUE") then
#
#         $HPMPI_DIR/mpirun -prot -np $LSDYNA_HYBRID_MPP_THREADS -hostlist
$NODE_LIST -e LSTC_LICENSE_SERVER=$LSTC_LICENSE_SERVER -e
LSTC_LICENSE=$LSTC_LICENSE $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
#       endif
#
#     endif
#
# INTEL MPI
#
# else if ($LSDYNA_MPI_TYPE == "INTEL") then
#
#   setenv INTEL_DIR /opt/intel/impi/4.0.3/bin64
#   set mpi_dir=$INTEL_DIR
#   source $mpi_dir/mpivars.csh
#
#   if ($LSDYNA_QUEUE == "TRUE") then
#
#     if (-e $PBS_NODEFILE) then
#       rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null

```

```

    if (-f $PBS_NODEFILE) then
        set last_host = ""
        foreach host ( `sort $PBS_NODEFILE` )
            if ( $host != $last_host) then
                set count = 0
            endif
            @ count = $count + 1
            if ( $count == 1) then
                echo "$host" >> $LSDYNA_JOB_DIR/appfile
                set LSDYNA_CL_ARGS = ""
            endif
            if ($count == $LSDYNA_HYBRID_SMP_THREADS) then
                set count = 0
            endif
            set last_host = $host
        end
    endif
    endif
    $INTEL_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -machinefile appfile -wdir
    $LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
    else if ($LSDYNA_ONLINE == "TRUE") then
#
        if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
            $INTEL_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -wdir $LSDYNA_JOB_DIR
            $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
        else if ($LSDYNA_NODE_FILE == "TRUE") then
#
            $INTEL_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -machinefile $NODE_FILE
            -wdir $LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
        else if ($LSDYNA_NODE_LIST == "TRUE") then
#
            $INTEL_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -hosts $NODE_LIST -wdir
            $LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
        endif
#
    endif
#
# OPEN MPI
#
    else if ($LSDYNA_MPI_TYPE == "OPENMPI") then
#
        setenv MPI_ROOT /opt/openmpi
        setenv OPENMPI_DIR $MPI_ROOT/bin
#
        if !($?PATH) then
            setenv PATH $MPI_ROOT/bin
        else
            setenv PATH $MPI_ROOT/bin:$PATH
        endif
        if !($?LD_LIBRARY_PATH) then
            setenv LD_LIBRARY_PATH $MPI_ROOT/lib
        else
            setenv LD_LIBRARY_PATH $MPI_ROOT/lib:$LD_LIBRARY_PATH
        endif
#
        if ($LSDYNA_QUEUE == "TRUE") then
#
            if (-e $PBS_NODEFILE) then
                rm -rf $LSDYNA_JOB_DIR/appfile >& /dev/null
            if (-f $PBS_NODEFILE) then
                set last_host = ""
                foreach host ( `sort $PBS_NODEFILE` )
                    if ( $host != $last_host) then
                        set count = 0
                    endif
                    @ count = $count + 1
                    if ( $count == 1) then

```

```

        echo "$host" >> $LSDYNA_JOB_DIR/appfile
        set LSDYNA_CL_ARGS = ""
    endif
    if ($count == $LSDYNA_HYBRID_SMP_THREADS) then
        set count = 0
    endif
    set last_host = $host
end
endif
endif
$OPENMPI_DIR/mpirun -machinefile appfile -wdir $LSDYNA_JOB_DIR
$LSDYNA_VERSION $LSDYNA_CL_ARGS
#
    else if ($LSDYNA_ONLINE == "TRUE") then
#
        if ($LSDYNA_LOCAL_HOST == "TRUE") then
#
            $OPENMPI_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -wdir $LSDYNA_JOB_DIR
            $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
            else if ($LSDYNA_NODE_FILE == "TRUE") then
#
                $OPENMPI_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -machinefile
                $NODE_FILE -x PATH -x LD_LIBRARY_PATH -x LSTC_LICENSE_SERVER -x LSTC_LICENSE
                -wdir $LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
                else if ($LSDYNA_NODE_LIST == "TRUE") then
#
                    $OPENMPI_DIR/mpirun -n $LSDYNA_HYBRID_MPP_THREADS -host $NODE_LIST -x
                    PATH -x LD_LIBRARY_PATH -x LSTC_LICENSE_SERVER -x LSTC_LICENSE -wdir
                    $LSDYNA_JOB_DIR $LSDYNA_VERSION $LSDYNA_CL_ARGS
#
                    endif
#
                endif
#
            endif
#
        endif
#
    endif
#
endif
#

```

2.5.3.2 Example of a Windows "oasys.submit" file

The following example shows the default "oasys.submit" file that is included with the version 12.0 software on PC platforms.

As MPP jobs can use either the HP-MPI or MPICH MPI libraries this script shows how to submit jobs using both of them. The default script is setup assuming that the different MPI libraries have been installed in the following directories.

MPI	Directory
HP-MPI	C:\Program Files (x86)\Hewlett-Packard\HP-MPI
PLATFORM MPI	C:\Program Files (x86)\Platform Computing\Platform-MPI
MPICH 2	C:\Program Files\MPICH2
INTEL MPI	C:\Program Files (x86)\Intel

This script also shows

1. How to submit MPP jobs to either the local machine or to multiple machines using either a file containing a list of hostnames or a string containing the hostnames.
2. How to pass environment variables for license options to the remote hosts via the mpirun command.

REM The following Environment Variables are available for this script
REM

```

REM LSDYNA_VERSION : full pathname of selected LS-DYNA executable
REM LSDYNA_MPP : TRUE if MPP version selected
REM LSDYNA_MPI_TYPE : MPI version
REM LSDYNA_DOUBLE : TRUE if double precision version selected
REM LSDYNA_ONLINE : TRUE if the job has been submitted ONLINE
REM LSDYNA_BACKGROUND : TRUE if the job has been submitted to BACKGROUND
REM LSDYNA_BATCH : TRUE if the job has been submitted using BATCH
REM LSDYNA_QUEUE : TRUE if the job has been submitted to a QUEUE
REM LSDYNA_JOB_DIR : full path of LS-DYNA job directory
REM LSDYNA_JOB_FILE : filename containing LS-DYNA job options
REM LSDYNA_JOB_CPUS : number of CPU's selected
REM LSDYNA_USER_ID : username
REM LSDYNA_LOCAL_HOST : TRUE if submitting MPP jobs to the local machine
REM LSDYNA_NODE_FILE : TRUE if submitting MPP jobs using a Node File
REM NODE_FILE : filename containing node list for MPP jobs
REM LSDYNA_LOCAL_HOST : TRUE if submitting MPP jobs using a Node List
REM NODE_LIST : string containing node list for MPP jobs
REM LSDYNA_CL_ARGS : string containing command line arguments
REM
REM Set windows operating system to determine the path to the MPI executable
REM Uncomment if you want to run the 64bit version
REM
REM set WINOS=WIN32
REM set WINOS=WIN64
REM
REM Set LSTC variables if not already set via system (uncomment if needed)
REM
REM Set LSTC variables if not already set via system
REM - Local licenses
REM set LSTC_LICENSE=local
REM LSTC_FILE=C:\Licenses\lstc_file
REM Network licenses
REM set LSTC_LICENSE=network
REM set LSTC_LICENSE_SERVER=vdgcls01
REM set LSTC_INTERNAL_CLIENT off
REM
REM SMP LS-DYNA submission
REM =====
REM
REM IF %LSDYNA_MPP% == TRUE GOTO :MPP
REM %LSDYNA_VERSION% %LSDYNA_CL_ARGS%
REM GOTO :DONE
REM
REM MPP LS-DYNA submission
REM =====
REM
REM :MPP
REM
REM SETUP MPI RUN COMMANDS
REM
REM - HP MPI
REM
REM IF NOT %LSDYNA_MPI_TYPE% == HPMPI GOTO :PLATFORM
REM IF %WINOS% == WIN32 set MPI_ROOT=C:\Program Files\Hewlett-Packard\HP-MPI
REM IF %WINOS% == WIN32 set MPIRUN="C:\Program
Files\Hewlett-Packard\HP-MPI\bin\mpirun"
REM IF %WINOS% == WIN64 set MPI_ROOT=C:\Program Files (x86)\Hewlett-Packard\HP-MPI
REM IF %WINOS% == WIN64 set MPIRUN="C:\Program Files
(x86)\Hewlett-Packard\HP-MPI\bin\mpirun"
REM GOTO :RUN_MPP
REM
REM - PLATFORM MPI
REM
REM :PLATFORM
REM IF NOT %LSDYNA_MPI_TYPE% == PMPI GOTO :INTEL
REM IF %WINOS% == WIN32 set MPI_ROOT=C:\Program Files\Platform
Computing\Platform-MPI
REM IF %WINOS% == WIN32 set MPIRUN="C:\Program Files\Platform
Computing\Platform-MPI\bin\mpirun"
REM IF %WINOS% == WIN64 set MPI_ROOT=C:\Program Files (x86)\Platform
Computing\Platform-MPI
REM IF %WINOS% == WIN64 set MPIRUN="C:\Program Files (x86)\Platform

```

```

Computing\Platform-MPI\bin\mpirun"
  IF %LSDYNA_MPI_TYPE% == PMPI set LSDYNA_MPI_TYPE=HPMPI
  GOTO :RUN_MP
REM
REM - INTEL MPI
REM
:INTEL
  IF NOT %LSDYNA_MPI_TYPE% == IMPI GOTO :MPICH
  IF %WINOS% == WIN32 set MPIRUN="C:\Program
Files\Intel\MPI-RT\4.1.0.028\ia32\bin\mpiexec.exe"
  IF %WINOS% == WIN64 set MPIRUN="C:\Program Files
(x86)\Intel\MPI-RT\4.1.0.028\em64t\bin\mpiexec.exe"
  GOTO :RUN_MPP
REM
REM - MPICH2
REM
:MPICH
  IF NOT %LSDYNA_MPI_TYPE% == MPICH2 GOTO :NO_MATCH
  IF %WINOS% == WIN32 set MPIRUN="C:\Program Files\MPICH2\bin\mpiexec.exe"
  IF %WINOS% == WIN64 set MPIRUN="C:\Program Files\MPICH2\bin\mpiexec.exe"
  GOTO :RUN_MPP
REM
REM
:NO_MATCH
  ECHO MPI TYPE NOT RECOGNISED
  exit
REM
:RUN_MPP
  cd %LSDYNA_JOB_DIR%
REM
REM RUN ANALYSIS
REM
REM HP-MPI or PLATFORM MPI using local machine
REM
  IF %LSDYNA_MPI_TYPE% == HPMPI (
    IF %LSDYNA_LOCAL_HOST% == TRUE (
      %MPIRUN% -prot -np %LSDYNA_JOB_CPUS% %LSDYNA_VERSION% %LSDYNA_CL_ARGS%
      GOTO :DONE
    )
    IF %LSDYNA_NODE_FILE% == TRUE (
      %MPIRUN% -prot -cache -np %LSDYNA_JOB_CPUS% -hostfile %NODE_FILE% -e
LSTC_LICENSE_SERVER=%LSTC_LICENSE_SERVER% -e LSTC_LICENSE=%LSTC_LICENSE%
LSDYNA_VERSION% %LSDYNA_CL_ARGS%
      GOTO :DONE
    )
    IF %LSDYNA_NODE_LIST% == TRUE (
      %MPIRUN% -prot -cache -np %LSDYNA_JOB_CPUS% -hostlist %NODE_LIST% -e
LSTC_LICENSE_SERVER=%LSTC_LICENSE_SERVER% -e LSTC_LICENSE=%LSTC_LICENSE%
LSDYNA_VERSION% %LSDYNA_CL_ARGS%
      GOTO :DONE
    )
  )
  REM
  REM INTEL-MPI using local machine
  REM
  IF %LSDYNA_MPI_TYPE% == IMPI (
    IF %LSDYNA_LOCAL_HOST% == TRUE (
      %MPIRUN% -n %LSDYNA_JOB_CPUS% -wdir %LSDYNA_JOB_DIR% -localonly
LSDYNA_VERSION% %LSDYNA_CL_ARGS%
      GOTO :DONE
    )
    IF %LSDYNA_NODE_FILE% == TRUE (
      %MPIRUN% -n %LSDYNA_JOB_CPUS% -machinefile %NODE_FILE% -wdir
LSDYNA_JOB_DIR% -mapall -genv LSTC_LICENSE_SERVER %LSTC_LICENSE_SERVER% -genv
LSTC_LICENSE %LSTC_LICENSE% LSDYNA_VERSION% %LSDYNA_CL_ARGS%
      GOTO :DONE
    )
    IF %LSDYNA_NODE_LIST% == TRUE (
      %MPIRUN% -hosts %NODE_LIST% -wdir %LSDYNA_JOB_DIR% -mapall -genv
LSTC_LICENSE_SERVER %LSTC_LICENSE_SERVER% -genv LSTC_LICENSE %LSTC_LICENSE%
LSDYNA_VERSION% %LSDYNA_CL_ARGS%

```

```

        GOTO :DONE
    )
)
REM
REM MPICH2
REM
    IF %LSDYNA_MPI_TYPE% == MPICH2 (
        IF %LSDYNA_LOCAL_HOST% == TRUE (
            %MPIRUN% -n %LSDYNA_JOB_CPUS% -wdir %LSDYNA_JOB_DIR% -localonly
            %LSDYNA_VERSION% %LSDYNA_CL_ARGS%
            GOTO :DONE
        )
        IF %LSDYNA_NODE_FILE% == TRUE (
            %MPIRUN% -n %LSDYNA_JOB_CPUS% -machinefile %NODE_FILE% -wdir
            %LSDYNA_JOB_DIR% -mapall -genv LSTC_LICENSE_SERVER %LSTC_LICENSE_SERVER% -genv
            LSTC_LICENSE %LSTC_LICENSE% %LSDYNA_VERSION% %LSDYNA_CL_ARGS%
            GOTO :DONE
        )
        IF %LSDYNA_NODE_LIST% == TRUE (
            %MPIRUN% -hosts %NODE_LIST% -wdir %LSDYNA_JOB_DIR% -mapall -genv
            LSTC_LICENSE_SERVER %LSTC_LICENSE_SERVER% -genv LSTC_LICENSE %LSTC_LICENSE%
            %LSDYNA_VERSION% %LSDYNA_CL_ARGS%
            GOTO :DONE
        )
    )
)
REM
REM
:DONE

```

2.5.4 Post Processing options

When a LS-DYNA job is submitted using the Shell automatic post processing options for Reporter and T/HIS can be selected. This section of the batch file contains any commands required to carry out the selected post processing options.

```

#
# PRIMER 'ztf' file creation
#
/data/dyna11/linux_executables/primer11.exe -d=batch
-ztf=/local/test/shell_test.key > \
shell_test.ztf_log
#
# T/HIS batch processing
#
setenv MENU_AUTO_CONFIRM true
/data/dyna10/linux_executables/this11.exe -d=x -tcf=shell_test.tcf -maximise
shell_test.thf

```


2.6 Creating an "oasys_queue" file

This file lists alternative batch queues and queue directives. It only needs to be created for systems on which jobs can be submitted to a NQS style queue.

The file is in xml format, with tags to define data blocks. It contains three distinct blocks, defining queue directives which are written for: specific queues; specific queue and cpu combinations and; all queues. There can be no blank lines in data blocks.

The first block defines the queue names, their cpu limit and any queue directives specific to the queue. This block is required.

```
<queue_names>
  <name1>
    cpu_limit="300"
    command="#PBS -m abe"
  </name1>
  <name2>
    cpu_limit="pipe"
  <name2>
  <name3>
    cpu_limit="none"
  </name3>
</queue_names>
```

Where:

<code><queue_names></code> and <code></queue_names></code>	are tags to indicate the start and end of the block of data
<code><name1></code> and <code></name1></code>	are the names of the queues and tags to indicate the start and end of the data for that queue (up to 20 can be defined)
<code>cpu_limit</code>	is the cpu limit for the queue (in seconds) or "pipe" or "none"
<code>command</code>	is an optional queue directive specific for the queue (up to 20 for each queue can be defined)

The second block defines what queue options will be available to the user (see [Section 1.2.4.7](#)), depending on the queue and number of cpus chosen. It is intended to be used to define queue directives which define the number of nodes and cpus to use. This block is optional.

```
<queue_commands>
  <name1>
    <1>
      display_string="1 CPU x 1 Node"
      command="#PBS -l nodes=1:ppn=1:name1
    </1>
    <2>
      display_string="2 CPU x 1 Node"
      command="#PBS -l nodes=1:ppn=2:name1
    </2>
    <2>
      display_string="1 CPU x 2 Node"
      command="#PBS -l nodes=2:ppn=1:name1
    </2>
  </name1>
  <name2>
    <4>
      mpp_only
      display_string="2 CPU x 2 Node"
      command="#PBS -l nodes=2:ppn=2:name2
    </4>
  </name2>
  <name3>
    <1>
      display_string="1 CPU x 1 Node"
      command="#PBS -l nodes=1:ppn=1:name3
    </1>
  </name3>
</queue_commands>
```

Where:

<code><queue_commands></code> and <code></queue_commands></code>	are tags to indicate the start and end of the block of data
<code><name1></code> and <code></name1></code>	are the names of the queues and tags to indicate the start and end of the data for that queue (note that they must be the same as the names defined in the first block)
<code><1></code> and <code></1></code>	are the number of cpus and tags to indicate the start and end of the data for that number of cpus
<code>mpp_only</code>	if this is included then the option will only be available if an MPP version of Dyna is selected
<code>display_string</code>	is the text that is displayed to the user in the popup menu on the Shell (see Section 1.2.4.7)
<code>command</code>	is a queue directive specific to the queue and number of cpus (up to 20 can be defined for the queue and cpu combination)

Up to 100 queue and cpu combinations can be defined. If this block is not included in the file then the queue options popup will be greyed out and the default directives produced by the Shell will be used.

The third block defines queue directives that apply to all queues. This block is optional.

```
<all_queue_commands>
  command="#PBS -m abe"
</all_queue_commands>
```

Where:

<code><all_queue_commands></code> and <code></all_queue_commands></code>	are tags to indicate the start and end of the block of data
<code>command</code>	is a queue directive (up to 20 can be defined)

Example :

```

$
<queue_names>

    <dyna>

        cpu_limit="none"

    </dyna>
    <fast>

        cpu_limit="3600"

    </fast>

</queue_names>
$
<queue_commands>

    <dyna>

        <1>

            display_string="1 CPU x 1 Node"
            command="#PBS -l
            nodes=1:ppn=1:dyna"

        </1>
        <2>

            display_string="2 CPU x 1 Node"
            command="#PBS -l
            nodes=1:ppn=2:dyna"

        </2>
        <2>

            mpp_only

            display_string="1 CPU x 2
            Nodes"

            command="#PBS -l
            nodes=1:ppn=2:dyna"

        </2>
        <4>

            mpp_only

            display_string="2 CPU x 2
            Nodes"

            command="#PBS -l
            nodes=2:ppn=2:dyna"

        </4>
        <4>

            display_string="4 CPU x 1 Node"
            command="#PBS -l
            nodes=1:ppn=4:dyna"

        </4>
    </dyna>

```

```

                                mpp_only

                                display_string="1 CPU x 4
                                Nodes"

                                command="#PBS -l
                                nodes=4:ppn=1:dyna"

                                </4>

                                </dyna>

                                <fast>

                                <2>

                                display_string="2 CPU x 1 Node"

                                command="#PBS -l
                                nodes=1:ppn=2:dyna"

                                </2>

                                <2>

                                mpp_only

                                display_string="1 CPU x 2
                                Nodes"

                                command="#PBS -l
                                nodes=1:ppn=2:dyna"

                                </2>

                                </fast>

</queue_commands>

$

<all_queue_commands>

                                command="#PBS -m abe"

</all_queue_commands>

$

```

2.7 Adding items to the "Utilities" menu

The "Utilities" menu within the Xshell may be customised to contain upto 20 items. The contents of this menu is controlled by a file called "oasys_utilities" that is located in the directory containing the version 11 executables. The format of this file is:

```
<Menu Name>    <Program>    <Xterm>
```

Where:

Menu Name: Name to be given to the program in the menu.

Action: The name of the program to execute.

Xterm: Does the program require an xterm window define as **YES** or **NO**

3 Additional Windows Setup

3.1 Setting File Associations

Under windows on PC it is possible to set up file associations so that double clicking on files automatically loads the file into the associated program. As part of the PC installation the following file associations will be set up automatically

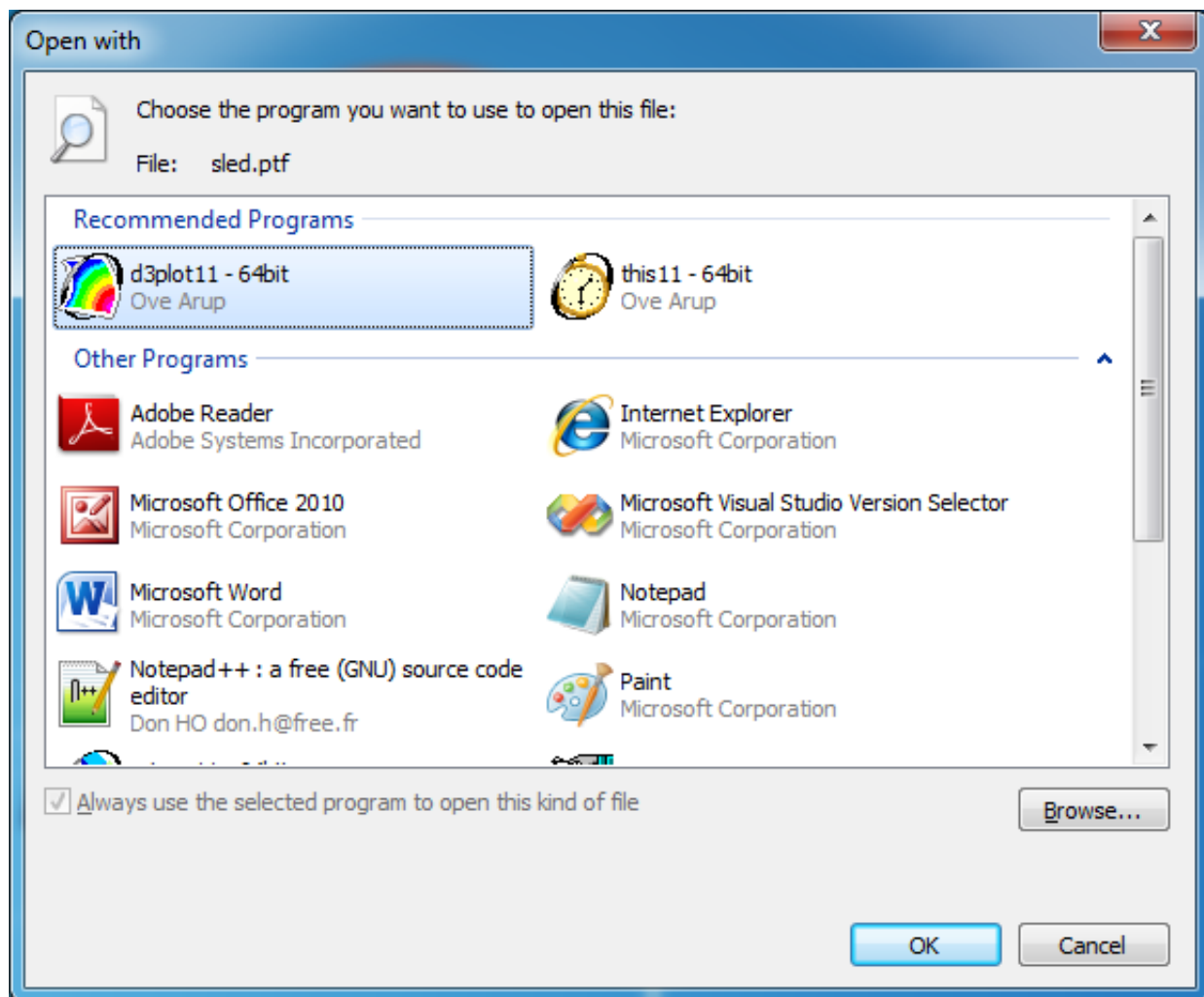
File Extension	Program
.ptf	D3PLOT 11
.thf	T/HIS 11
.xtf	T/HIS 11
.key	PRIMER 11
.orr	REPORTER 11
.ort	REPORTER 11

Sections 3.1.1 to 3.1.3 describe in detail how these file associations can be manually set up if required.

3.1.1 To make .ptf files open in D3PLOT by double-clicking on them

If no application is currently associated with .ptf files, a "double-click" won't work, and some non-specific, usually "windows", icon will be displayed with the file.

Right click on any **.ptf** file, and select **properties** then press the Change... tab next to Opens with: from the popup menu.



1. This will bring up the "Open with" panel.
2. Ensure the **Always use...** box is ticked
3. Use the directory browsing window to find the correct D3PLOT executable. You are looking for file **d3plot11.exe** or **d3plot11_x64.exe**.
4. Select the executable and click on **OK** to close the "Open With" window.

D3PLOT should now open and read in the selected file and you should now find that:

1. All **.ptf** files on your system show the D3PLOT icon.
2. Double-clicking on any such file starts D3PLOT and opens that file.

It is not possible to set up the filename "d3plot" for double-clicking in this way since Windows requires filename extensions when assigning applications to files.)

3.1.2 To make .thf, .xtf, .cur and .bdf files open in T/HIS by double-clicking on them

The procedure is exactly the same as for D3PLOT, and must be carried out for each of the file types that you wish to process by double-clicking:

- .thf**: LS-DYNA Time History file
- .xtf**: LS-DYNA Extra Time History file
- .cur**: T/HIS Curve file
- .bdf**: T/HIS Bulk Data file

The only difference is:

1. The application to use should be **this11.exe** or **this11_x64.exe**.

Note that:

1. File types **.thf** and **.xtf** are opened in this way, but no contents are read in.
2. File types **.cur** and **.bdf** are opened and their complete contents read in.

LS-DYNA default filenames "**d3thdt**" and "**xtfile**" cannot be set up for double-click access in this way because Windows requires filenames to have extensions when assigning applications to them.

3.1.3 To make .key files open in PRIMER by double-clicking on them

The procedure is exactly the same as for D3PLOT, and must be carried out for each of the file types that you wish to process by double-clicking:

- .key**: LS-DYNA Keyword Input File

The only difference is:

1. The application to use should be **primer11.exe**.

4 Preferences

The Preferences editor provides a graphical interface for viewing, checking and editing your user preferences stored in your [oa_pref](#) file

4.1 The oa_pref file

This file contains code-specific preferences that can be used to modify the behaviour of the software suite. It is optional and, where entries (or the whole file) are omitted, programs will revert to their default settings.

4.1.1 "oa_pref" naming convention and locations

The file is called "oa_pref".

It is looked for in the following places in the order given:

- The optional administration directory defined by the environmental variable (`$OA_ADMIN` or `$OA_ADMIN_xx` where xx is the release number).
- The site-wide installation directory defined by the environment variable (`$OA_INSTALL`)
- The user's home directory: `$HOME` (Linux) or `%USERPROFILE%` (Windows)
- The current working directory

See [Installation organisation](#) for an explanation of the directory structure.

All four files are read (if they exist) and the last preference read will be the one used, so the file can be customised for a particular job or user at will.

Files do not have to exist in any of these locations, and if none exists the programme defaults will be used.

On Linux:

`$HOME` on Linux is usually the home directory specified for each user in the system password file. The shell command "**printenv**" (or on some systems "**setenv**") will show the value of this variable if set. If not set then it is defined as the "~" directory for the user. The command "**cd; pwd**" will show this.

On Windows:

`%USERPROFILE%` on Windows is usually `C:\Documents and Settings\<user id>\`. Issuing the "**set**" command from an MS-DOS prompt will show the value of this and other variables.

Generally speaking you should put

- Organisation-wide options in the version in `$OA_ADMIN_xx` and/or `$OA_INSTALL`,
- User-specific options in `$HOME / %USERPROFILE%`
- Project-specific options in the current working directory.

The file contains preferences for the SHELL (lines commencing shell*), THIS (lines commencing this*), D3PLOT (lines commencing d3plot*), PRIMER (lines commencing primer*) and REPORTER (lines commencing reporter*). All lines take the format <preference name> <preference value>.

The general copy of the preference file should be present in the [\\$OA_ADMIN_xx](#) and/or [\\$OA_INSTALL](#) directory. This should contain the preferences most suitable for all software users on the system.

An individual's specific preferences file can be stored in the individual's home area. This can be used to personally customise the software to the individual's needs.

Whenever one of the programs whose preferences can be stored in the oa_pref file is fired up, the program will take preferences first from the general preference file in the [\\$OA_ADMIN_xx](#) directory (if it exists) then the [\\$OA_INSTALL](#) directory, then from the file in the user's home area, then from the current working directory.

Preferences defined in the general oa_pref file can be modified in the user's personal file but they can't be removed by it.

From version 9.4 onwards preferences can be locked. If a preference is locked it cannot be changed in an oa_pref file in a more junior directory. To lock a preference use the syntax '**shell#**' rather than '**shell***'.

An example of the file is shown below to illustrate the content of the file

```
# Preferences file for software.
#
# Preferences for SHELL
```

```

shell*queue_cpu: 0
#
# Preferences for THIS
this*laser_paper_size: A4
#
# Preferences for D3PLOT
d3plot*overlay_colour: grey
#
# Preferences for PRIMER
primer*overlay_mode off

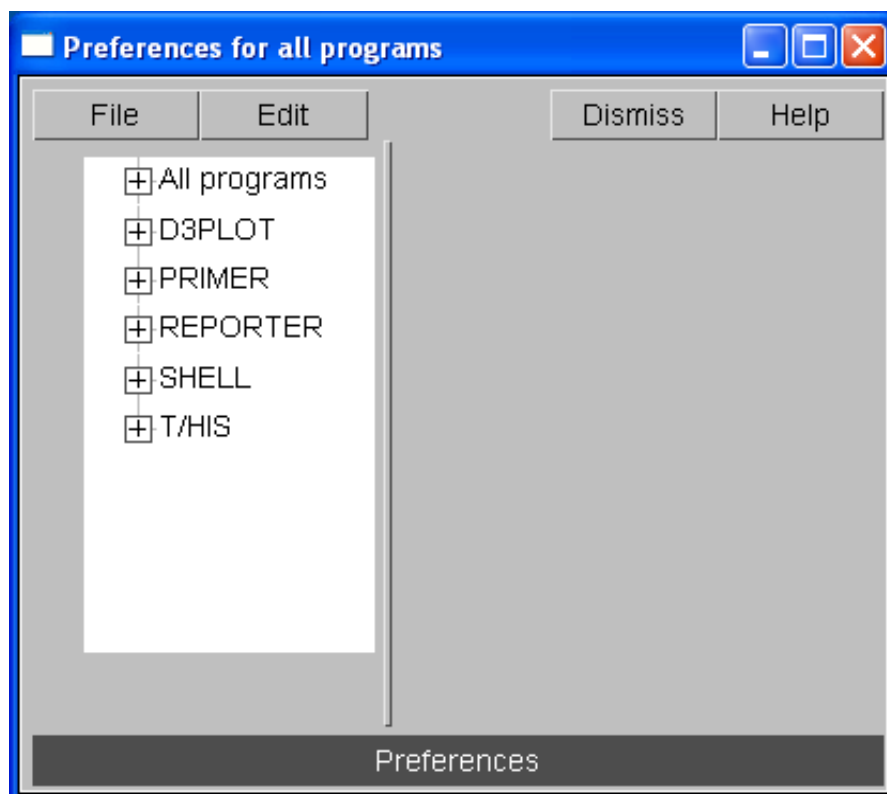
```

4.2 The preferences editor

The editor can be accessed from within the Shell or from within D3PLOT, T/HIS, PRIMER and REPORTER.

4.2.1 The preferences editor layout

The preferences editor window is divided into two frames with a menu bar across the top.



4.2.1.1 Menu Bar



File options:

Save preferences: Save current preference settings. This will save the personal oa_pref file in the user's home directory. Only those preferences which differ from the preferences saved in the general oa_pref file will be saved.

Exit: Exit the preferences editor without saving.

Edit options:

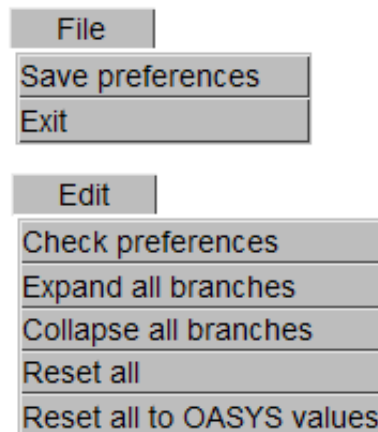
Check Preferences: Checks the current preferences for any errors. These errors will be listed in a separate window detailing the preferences with the errors and the nature of those errors

Expand all branches: Expands the categories in the [Left hand frame](#).

Collapse all branches: Collapses the categories in the [Left hand frame](#)

Reset all: Resets all values.

Reset all to OA_INSTALL values: Resets all values to the defaults stored in the main \$OA_INSTALL preference file



4.2.1.2 The preferences editor Left hand frame

The left hand frame will contain the names of all preferences available to set. Preferences will be listed under the headings: PRIMER, D3PLOT, T/HIS, REPORTER and SHELL according to which program they are applicable to.

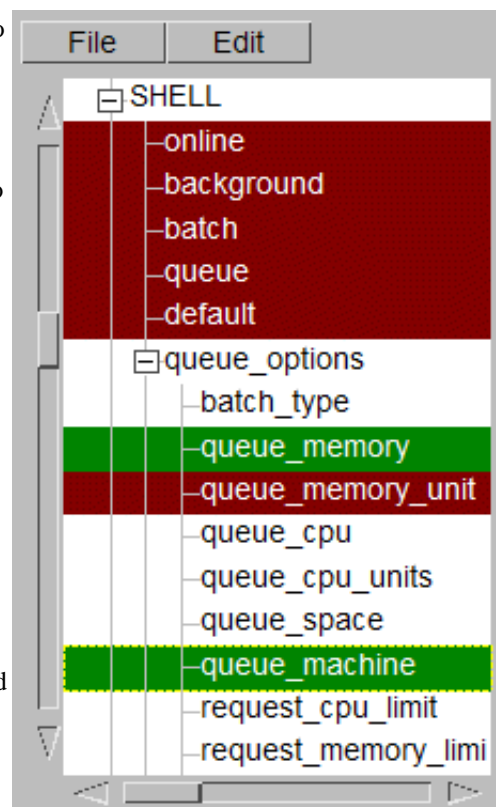
These categories can be expanded to reveal their respective preferences/contracted to hide their preferences by clicking on the box to the left of the respective category, alternatively, use the edit drop down menu and select Expand all branches or collapse all branches.

Green	Means that the option has been read from your \$HOME/\$USERPROFILE file.
Red	Means that the option has been read from the \$OA_INSTALL file.
Magenta	Means that the option had been read from the \$OA_ADMIN file.

Preferences which aren't highlighted indicate preferences that haven't been set.

Preferences in **bold** type indicate preferences which haven't been assigned the default value.

A list of all the preferences available and their default value can be found in any oa_pref file written by the preferences editor.



4.2.1.3 The preferences editor Right hand frame

The right hand frame will contain information about the currently selected preference and provides the opportunity to edit this preference.

Name:	d3plot*maximise
Type:	<logical>
Default:	FALSE
Description:	Maximise window when D3PLOT starts
Active:	<input checked="" type="checkbox"/>
Value:	FALSE

Name: States the name of the currently selected preference.

Type: Specifies the type of variable applicable to this preference.

Default: States the default value of the preference. **Description:** Provides a brief description of the function performed by this preference.

Active tab: Highlighted in Green when the preference has been assigned a value. Press this tab to activate/deactivate the currently selected preference. If the currently selected preference was defined in the general oa_pref file, deselecting this will bring up an error message as it is not possible to deselect preferences stored in the general oa_pref file.

Value: States the currently selected value for the preference. Clicking on the arrow to the left of this box brings up a drop-down menu which lists the possible values this preference can take and allows the user to select one of these values.

4.2.1.4 Locking Preferences

From version 9.4 onwards preferences can be locked. Beside each option in the preference editor is a padlock symbol. If the symbol is green then the option is unlocked, if it is red then it is locked. If a preference option has been locked in a file that the user can not modify then an error message will be generated if the user tries to edit that option.



If a user manually edits the "oa_pref" file to try and set an option that has been locked in another preference file then the option will be ignored in the users preference file.

Installation organisation

The version 12 installation can be customised to try and avoid a number of issues that often occur in large organisations with many users.

- Large organisations generally imply large networks, and it is often the case that the performance of these networks can be intermittent or poor, therefore it is common practice to perform an installation of the software on the local disk of each machine, rather than having a single installation on a remote disk.

This avoids the pauses and glitches that can occur when running executable files over a network, but it also means that all the configuration files in, or depending upon, the top level "Admin" directory have to be copied to all machines and, more to the point, any changes or additions to such files also have to be copied to all machines.

- In larger organisations the "one person per computer" philosophy may not apply, with the consequence that users will tend to have a floating home area on a network drive and may not use the same machine every day.

This is not usually a problem on Linux where the "home" directory is tied to the login name not the machine. However on Windows platforms it means that %USERPROFILE%, which is typically on the local C drive of a machine, is not a good place to consider as "home" since it will be tied to a given computer, therefore a user who saves a file in his home directory on machine A may not be able to access it from machine B.

- In a similar vein placing large temporary files on the /tmp partition (Linux) or the C: drive (Windows) may result in local disks becoming too full, or quotas exceeded.

This section gives only a brief summary of the installation organisation, and you should refer to the separate Installation Guide if you want to find out more about the details of installation, licensing, and other related issues.

Version 12.0 Installation structure

In version 12.0 the option is provided to separate a top-level 'administration' directory from the 'installation' one where the executables are located.

For large installations on many machines this allows central configuration and administration files to exist in one place only, but executables to be installed locally on users' machines to give better performance. Version 12.0 also allows the following items to be configured

- The location for user manuals and other documentation.
- The definition of a user's home directory.
- The definition of the temporary directory for scratch files.

In addition parsing of the 'oa_pref' (preferences) file will now handle environment variables, so that a generic preference can be configured to give a user-specific result, and preferences may be 'locked' so that those set at the administration level cannot be changed by users.

These changes are entirely optional, and users performing a simple installation on a single machine do not need to make any changes to their existing installation practice.

Directory	Status	Directory Content and purpose	oa_pref file option
OA_ADMIN_xx	Optional	Top level configuration files. (xx =12 for release 12.0, thus OA_ADMIN_12) Admin level oa_pref file Other configuration files Timeout configuration file	

OA_ADMIN	<i>Optional</i>	Same as OA_ADMIN_12 , provided for backwards compatibility with earlier releases. It is recommended that plain OA_ADMIN , without the _xx version suffix, is not used since otherwise there is no easy way of distinguishing between parallel installations of different releases of the Oasys Ltd software in an installation. <i>If OA_ADMIN_12 is not defined then this non-release specific version is checked.</i>	
OA_INSTALL_xx	<i>Optional</i>	(xx =12 for release 12.0, thus OA_ADMIN_12 All executables Installation level oa_pref file	oasys*install_dir: <pathname>
OA_INSTALL	<i>Optional</i>	Same as OA_INSTALL_12 . If no " OA_ADMIN_xx " directory is used and all software is simply placed in this "install" directory, which would be typical of a single-user installation, then it is recommended that the _xx version suffix is used in order to keep parallel installations of different releases of the Oasys Ltd software separate on the machine. <i>If OA_INSTALL_12 is not defined then this non-release specific version is checked</i>	oasys*install_dir: <pathname>
OA_MANUALS	<i>Optional</i>	Specific directory for user manuals. If not defined then will search in: OA_ADMIN_xx/manuals (xx = major version number) OA_INSTALL/manuals	oasys*manuals_dir: <pathname>
OA_HOME	<i>Optional</i>	Specific "home" directory for user when using Oasys Ltd software. If not defined will use: \$HOME (Linux) %USERPROFILE% (Windows)	oasys*home_dir: <pathname>
OA_TEMP	<i>Optional</i>	Specific "temporary" directory for user when using Oasys Ltd software. If not defined will use: P_tmpdir (Linux, typically /tmp) %TEMP% (Windows, typically C:\temp)	oasys*temp_dir: <pathname>

It will be clear from the table above that no Environment variables have to be set, and that all defaults will revert to pre-9.4 behaviour. In other words users wishing to keep the status quo will find behaviour and layout unchanged if they do nothing.

OA_INSTALL_XX

Previously the software used the **OA_INSTALL** (renamed from **OASYS**) environment variable to locate the directory the software was installed in.

- On Windows this is no longer required as the software can work out its own installation directory. As this environment variable is no longer required it is recommended that it is removed from machines it is currently set on as in some cases where more than one version has been installed in different directories it can cause problems.
- On LINUX systems the "oasys_12" script that starts the SHELL automatically sets this Environment Variable and passes it to any application started from the SHELL. If you run applications directly from the command line and bypass the SHELL then you should set **OA_INSTALL_XX** so that the software can locate manuals and other required files.

OA_ADMIN_XX

Users wishing to separate configuration and installation directories will be able to do so by making use of the new top level **OA_ADMIN_xx** directory.

Installation Examples

The following diagrams illustrate how the installation might be organised in various different scenarios..

a) Single user installation on one machine

There is no need to worry about separating administration and installation directories, and the default installation of all files in and below the single installation directory will suffice.

It is suggested that the **xx** version suffix of **OA_INSTALL_xx** is used in order to keep parallel installations of different releases of the Oassys Ltd software separate on the machine.

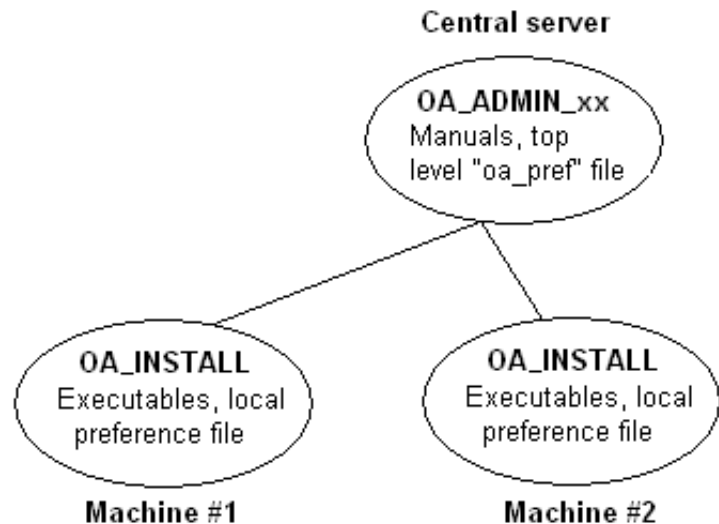


b) A few machines on a small network, each user has his own machine

The top level administration directory can be installed on a network server, possibly also locating the manuals centrally.

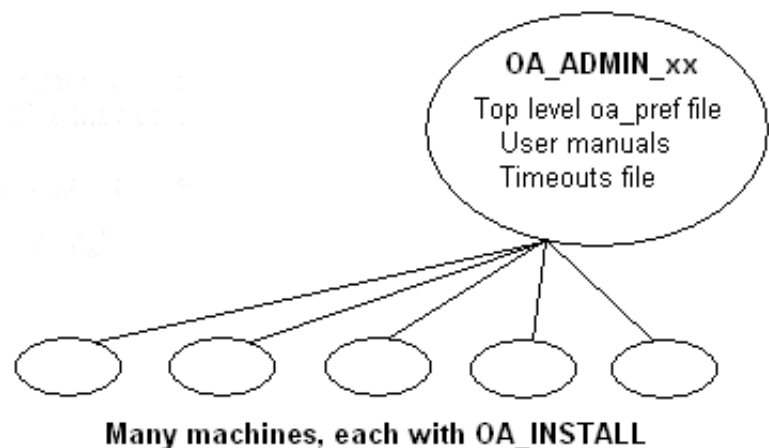
Each user's machine has its own 'installation' directory to give good performance, but there is no need to manage home or temporary directories centrally since each user 'owns' his machine.

If network performance is good an alternative would be to install executables on the central server, meaning that local OA_INSTALL directories are not required.



c) Large corporate network

There is no need to worry about separating administration and installation directories, and the default installation of all files in and below the single installation directory will suffice.



Dynamic configuration using the top level oa_pref file.

A further improvement is that all environment variables below **OA_ADMIN_xx** may either be set explicitly, or dynamically using the options in the oa_pref file at the top **OA_ADMIN_xx** level. This permits parallel installations of different versions of the software to co-exist, with only the top level administration directory names being distinct. For example:

Release 12.0	Release 12.1
Top level directory OA_ADMIN_12	Top level directory OA_ADMIN_121
oa_pref file in OA_ADMIN_12 contains: oasys*install_dir: <pathname for 12.0 installation> oasys*manuals_dir: <pathname for 12.0 manuals> oasys*home_dir: <pathname for home directory> oasys*temp_dir: <pathname for temporary files>	oa_pref file in OA_ADMIN_121 contains: oasys*install_dir: <pathname for 12.1 installation> oasys*manuals_dir: <pathname for 12.1 manuals> } would almost certainly be unchanged between major } versions, although they could be different if desired
Pathnames in the oa_pref file may contain environment variables which will be resolved before being applied.	

The hierarchy of oa_pref file reading

It will be clear from the above that in a large installation the "oa_pref" files have a significant role. Each piece of software reads them in the following order:

OA_ADMIN_xx	Top level configuration
OA_INSTALL_xx	Installation level
OA_HOME	User's personal "home" file
Current working directory	File specific to the current directory (rarely used)

The rules for reading these files are:

- If a given directory does not exist, or no file is found in that directory, then no action is taken. This is not an error.
- A more recently read definition supersedes one read earlier, therefore "local" definitions can supersede "global" ones (unless it was locked).
- If two of more of the directories in the table above are the same then that file is only read once from the first instance.

Locking Preference Options

From version 9.4 onwards preference options can be locked. If a preference option is locked in a file then that preference option will be ignored in any of the subsequent preference files that are read.

Therefore by locking a preference in a top-level file in the hierarchy above, eg in **OA_ADMIN_xx**, and then protecting that file to be read-only, an administrator can set preferences that cannot be altered by users since any definitions of that preference in their private oa_pref files will be ignored.

Preferences are locked by using a hash (#) rather than an asterisk (*) between the code name and the preference string. For example:

primer*maximise: true	Normal case using "*", means an unlocked preference
------------------------------	---

<code>primer#maximise: true</code>	Locked case using "#"
------------------------------------	-----------------------

These changes may be made either by editing the file manually, or by using the preferences editor.