



---

---

**Smooth Velocity  
v5.3.14  
Documentation**

---

---

Tsunami Development  
tsunamidevelopment.com  
713-783-1435

## **Tsunami Development**

713-783-1435

[tsunamidevelopment.com](http://tsunamidevelopment.com)

The Information contained in this document is subject to change without notice. Tsunami Development assumes no responsibility for any error that may appear in this manual. Tsunami Development does not warrant that this document is error free. Please report any errors in this document in writing to Tsunami Development. Some states or jurisdictions do not allow disclaimer of expressed or implied warranties in certain transactions; therefore this statement may not apply to you.

---

# CONTENTS

---

Introduction.....	4
Installation.....	5
System Requirements.....	6
Installation of Tsunami.....	6
GUI Installation.....	7
Floating License.....	9
Cluster Configuration.....	10
Node Utility.....	12
Getting Started.....	15
Generating a Smooth Velocity .....	16
Files Tab.....	16
Velocity File Tab.....	19
Options Tab.....	20
Stopping and Restarting a Job.....	23
Appendix A : Log file Information.....	25
Example Log file.....	25
Example Check License Log.....	28
Example Clear Nodes Log.....	29
Appendix B : Parameter File Information .....	30
Required Parameters.....	30
Alphabetical List of Parameters.....	31
Example Parameter File.....	33

---

# Smooth Velocity Introduction

---

The Raytracer program uses raytracing to calculate the travel times used for PSDM. If the velocity model input to raytracing is not adequately smoothed, the rays may become scattered, resulting in erratic or incorrect travel times. Smoothing is allowed within the Raytracer but this requires a bit of guesswork since you cannot look at the smoothed velocities that are actually being used. The Raytracer will provide feedback on how smooth the velocity model is but if this is not satisfactory, the program must be stopped, the smoothing parameter changed, and the program restarted. Also, since the user does not have the smoothed model that is used to calculate travel times, the model cannot be input directly into Tomography or other velocity modeling programs for further updates. For these reasons we have build a separate utility that allows the user to intelligently smooth and QC the velocity model.

Smoothing does not produce a cumulative error in the depth of events. The smoothing process is a bit of averaging of velocities, so that they alternate between getting sped up a bit, and slowed down a bit, but on average stay the same. So in some places where there are large gradients, the smoothing can slightly shift events, but these shifts are local. A table listing the average depth shift vs. depth for twenty evenly spaced depths is provided in the Smoother log file. Even for velocity models with large velocity gradients, smoothing to acceptable levels can easily be smaller than one or two depth sample intervals. Another advantage is that smoother velocity models should give better raytracing performance.

To measure how smooth a velocity model is, we calculate the maximum derivative of the velocity changes in the x, y and z directions (dx, dy, dz). The smoother the velocity model, the smaller these derivatives will be. For the velocity smoother, the user specifies target derivatives for the output smoothed velocity model. The program will increase the size of the smoothing operator until the derivatives for the model are less than the target value. This allows the user to get the desired amount of smoothing without having to guess at a smoothing factor size.

A good rule of thumb is that the target derivatives for dx and xy should be no larger than 15. For dz, the target should be no larger than 50. These are the default values.

The smoothed velocity can be output as either a SEGY file for QC, a block model file, or both. The block model velocity file is required by the Raytracer. The block file can be created from the SEGY file inside the Raytracer. If using the Smoother it is more efficient to build the block file in the Smoother. Make sure that the Raytracer option to Build New Block Model is turned off. For more details on the block model file see the Chapter: Performance Considerations, Velocity Scratch Block File in the Raytracer User Manual.

The Smoother also allows the user to resample the velocity model in the depth direction. This allows greater smoothing with less shift in depth. The resampling factor divides the depth sample interval by the value specified. For example, a resample factor of two will divide the depth sample in half, creating twice as many velocity samples. The trade off is that this makes the velocity files larger so very large resample values should be avoided. Large resample values can also slow down the performance of the Raytracer. The sampling interval of the velocity file does not however have any effect on the PSDM run times. PSDM uses the calculated travel times and not the velocity model or block model files.

---

---

# Installation

---

## System Requirements

Tsunami will run on any combination of Linux, SGI, Opteron, and Itanium systems.

Linux must be: version 2.4 or greater, with the gcc compiler

SGI must be: 6.2 of IRIX or greater

Itanium: must have the Intel v8.0 compiler

Tsunami is very flexible such that all systems within a cluster need not be running the same version of the operating system. Even different versions of Linux can be accommodated by using the features of the node database file.

Please contact Tsunami Development for if you need to mix multiple versions of Linux operating systems.

The amount of memory suggested is 256 MBytes per processor for PSTM, and 512 MBytes per processor for PSDM and Raytracer. Smaller amounts may work with some performance penalty, or if the jobs to be run are small.

No disk space is required on any of the compute nodes, only the master node needs to be able to see the data files and the file system for the output files. The /tmp file system needs to exist on the compute nodes and permissions need to be open to write to /tmp. Tsunami will put the executable and local logfiles for the compute nodes in /tmp.

It's suggested that NFS not be used within the cluster. NFS is not used by Tsunami, and can cause problems as the clusters get larger. This is especially true as the cluster exceeds 100 processors, as NFS can cause the systems to hang.

The rsh and rcp commands must be enabled. Tsunami uses the rsh and rcp commands to copy and start the executable on the compute nodes. Therefore permissions must be set, so that rsh and rcp can work in the users accounts. You can test the rsh command by typing the following at the prompt to get the current date:

Example: <prompt> : rsh < node name > date

You can test the rcp by copying a file to the node's /tmp directory:

Example: <prompt>: rcp <file> node\_name: /tmp

For most installations a 100 Mbit network is sufficient to support Tsunami. Each node should be on a 100 Mbit switch. This will be sufficient for most jobs.

## Installation of Tsunami

1. Obtain tar file of application from the ftp site provided by Tsunami Development.
2. Change directory to the apps directory.

Example: <prompt>: cd /apps

3. Create a directory named tsunami.

Example: <prompt>: mkdir tsunami

4. Copy the tar file into the tsunami directory created in step 3.  
Untar the Tsunami tar file.

Example: tar -xvpf tsunami.tar

You should now have the following structure:

```
apps/tsunami/tsunami_5.3.14/pstm_5.3.14
                             psdm_5.3.14
                             rays_5.3.14
                             tomo_5.3.14
                             smth_5.3.14
```

Your license file will control whether or not you can run all executables. Your final directory structure should look like the following:

```
/apps/tsunami/tsunami_5.3.14/rays_5.3.14/itanium
                                     itanium_3.5
                                     linux
                                     linux_3.4
                                     opteron
                                     opteron32
                                     opteron_3.4
                                     sgi
```

5. Obtain license file from Tsunami Development via email. Save license file in the TSUNAMI directory created in step 3. Please see the *Floating License* section for more information on license files.

## GUI Installation

1. No installation of the GUI is required. In order to run the GUI (/apps/tsunami/tsunami\_5.3.14/tsunami.jar), you must have Version 1.4.2 or higher of Java 2 Platform, Standard Edition (J2SE) installed. To determine your java information use the following commands:

- a) `java -version`

This will give you the version number of your java installation.

- b) `which java`

This will tell you where your java installation is located.

Should these commands return no information or you have a lower version of java, please have your system administrator ensure that the correct version of java is installed on your machine and the JAVA\_HOME environment variable is set in the users .cshrc or .bashrc file. Should you need to install java, please see *Appendix C: Java Installation* for more information.

2. Edit the users “.cshrc” or “.bashrc” file to include the following variables:

- a) If the environment variable JAVA\_HOME does not already exist, add to .cshrc or .bashrc file.

For .bashrc: `export JAVA_HOME="java directory"`

Where java directory is the java install directory determined in step 1.

For .cshrc: `setenv JAVA_HOME java directory`

Where java directory is the java install directory determined in step 1.

- b) Add JAVA\_HOME/bin to PATH variable.

For .bashrc: PATH is located in the .bash\_profile file. Add JAVA\_HOME/bin to the end of the existing PATH variable.

Ex: `export PATH=$PATH:$HOME/bin:$JAVA_HOME/bin`

For .cshrc: PATH is located in the .login file.

Add the JAVA\_HOME directory/bin to the end of the existing set path= variable.

Ex: `set path=(/bin /usr/bin /sbin /`

usr/etc /usr/local/bin /usr/java2/bin )  
Where /usr/java2 is JAVA\_HOME

- c) Add an environment variable TSUNAMI that points to the tsunami installation directory.

For .bashrc: export TSUNAMI="/apps/tsunami"

For .cshrc: setenv TSUNAMI /apps/tsunami

- d) Add an alias that points to the tsunami GUI executable.

For .bashrc: alias tsunami="/apps/tsunami/tsunami\_5.3.14/tsunami.sh"

For .cshrc: alias tsunami /apps/tsunami/tsunami\_5.3.14/tsunami.sh

## Floating License

The license file registers the nodes that are licensed for the products. Up until 3.1.7 the user could run the application on any and all nodes licensed in the file. With the floating license, the user may have all their nodes in the license file but only check out how many nodes that they have purchased. With version 4.15.9, cpu's will also be taken into consideration to accommodate those users who have more than 2 cpu's per system.

The floating license allows the user to have any number of nodes in the license file, and then license the number that they purchased. The software will check out the licenses from the license file when a job runs, and check in the license when the job completes, or is aborted. Any number of jobs can run on the nodes that are checked out, but only the number allowed by the license can be checked out. An example would be to have 50 nodes, license 20, and be able to use any 20 out of those 50 at a given time.

For the floating cpu license, if only one job is running on a node the license manager will only check out the number of cpus specified by the node.db file. If multiple jobs are running on a node then all cpus licensed for that node will be checked out and a warning will be printed to the logfile.

If not all the cpus on a node are licensed then the user will not be allowed to specify more cpus in the node.db file than are licensed for that node. If not all cpus are licensed for a node, then the user will not be able to run multiple jobs on the node. The software checks the number of physical cpus on the node, and compares it to the number licensed, and the number in the node db file.

You can see the status of the license file by running the utility: `check_license_file`. It will give you a list of the node name, the mac address, the expiration date, the number of cpus licensed for each node and how many jobs have the node checked out.

Example: `<prompt>: check_license_file <license file name>`

If for some reason a job fails to check in the license you can reset the license file for a group of nodes by running `clear_nodes`. Clear nodes will kill jobs running on the nodes listed in the node.db file you submitted to `clear_nodes`, as well as reset the licenses for only those nodes in the node.db submitted to the `clear_nodes` command. Clear nodes has been changed for 3.1.7 and no longer uses a user id for an argument, it will now clear all Tsunami processes on the nodes requested. The `clear_nodes` will now create a `clear_nodes.log` file in `$TSUNAMI`.

Example: `<prompt>: clear_nodes < node db file >`

The location of the license file has changed from 3.1.6. It is now in the `/apps/tsunami` directory instead of the application directory.

## Cluster Configuration

A node description file must be created for tsunami. This node db will be the master node list. It is an ASCII list of the nodes for the job in the following format:

node name	number of processors	memory in Mbytes	speed factor	operating system
-----------	----------------------	------------------	--------------	------------------

The node name is the name of the system as listed in the /etc/hosts file.

The speed factor is the relative speed of the processor compared with the other nodes. The speed factor and the number of processors are used to balance the workload between nodes. This allows one to mix nodes with different number of processors, and different speeds in the same cluster.

Typically the fastest processors are given a factor of 1.0, and slower ones numbers less than one. For instance, a 3000 MHz processor might have a factor of 1.0, and a 1500 MHz processor a factor of .5.

At the end of the log file for a job, statistics are provided for each node. You should use the relative values for “Millions of shift and sums per second kernel time” as the factor. This gives the speed of migrating the data when the code is executing the kernel.

The operating system tells the software which one of the executables to use. The supported operating systems are linux, sgi, solaris, opteron and itanium (linux).

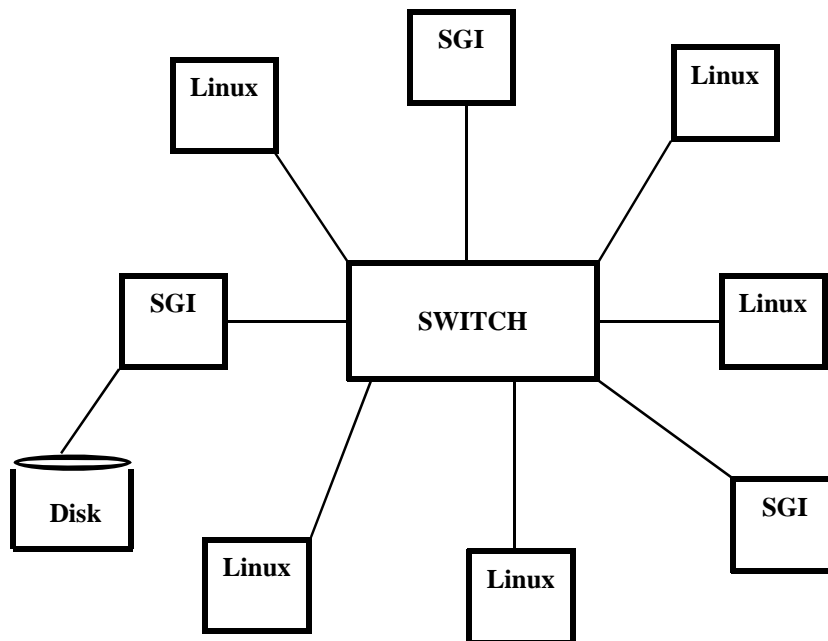
The server node must be the first entry in the db. This is the node from which the job is launched, and is the only node that must have access to the input data, and the file system where the output data will be written. The server may also be a compute node, in that case the server should be listed twice in the first two lines of the db file.

The following is an example of a node db, there is one record per line with spaces as the delimiter. You can use the # sign to comment out a line if necessary for a particular job.

server1	2	512	1.0	linux
linux1	2	512	1.0	linux
linux2	2	512	1.0	linux
sgi1	4	512	.75	sgi
itanium1	8	1000	.65	Itanium

In order for the program to execute, it must be possible to issue a remote shell from the node where the program is initiated to all the nodes in the node database. If you have questions about this see your system administrator or contact ESS. There also must be a /tmp directory on all the nodes. This is where the executable is copied to when the program begins. The executable is removed at the end of the job.

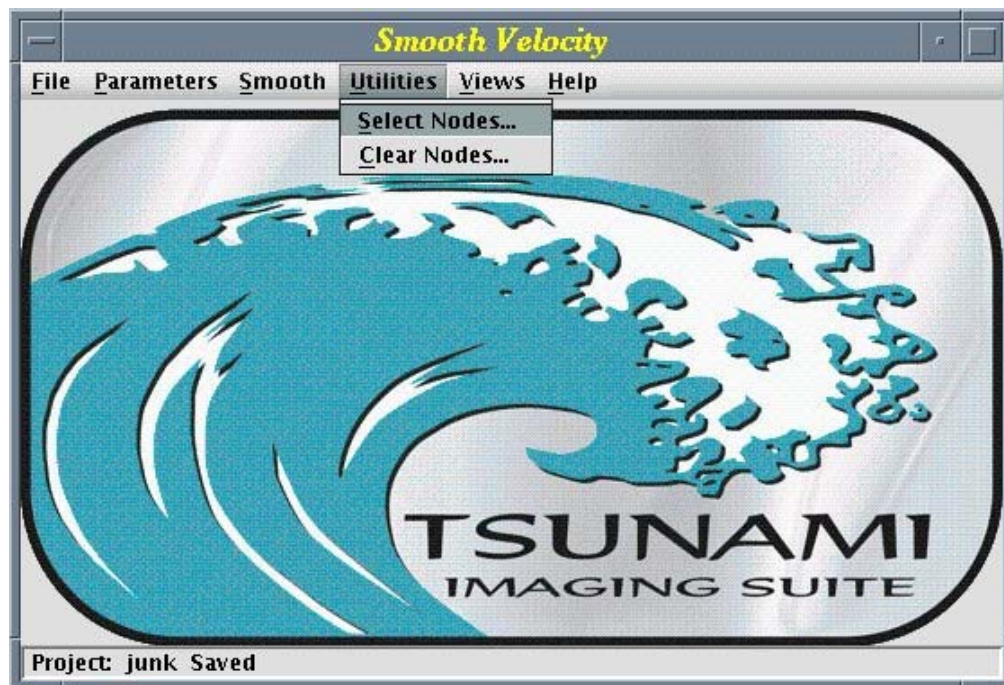
## Example Cluster Configuration



## Node Utility

Once the master node database has been created, you can use the Select Node Utility from the GUI. \*See Getting Started for GUI start up steps.

1. Select **Utility** —> **Select Nodes**.



The select nodes dialog box opens and displays node files, as well as creates new node files.

To open a node list file or to create a new one, the program must first search for a master node list file. The master node list should be located in \$TSUNAMI. Where \$TSUNAMI = the tsunami directory - such as "/apps/tsunami". If no master node list is found, the user will then be prompted to enter the correct path (where the master node list resides) into the Application Directory (hmdr) on the files panel of the parameters edit box.

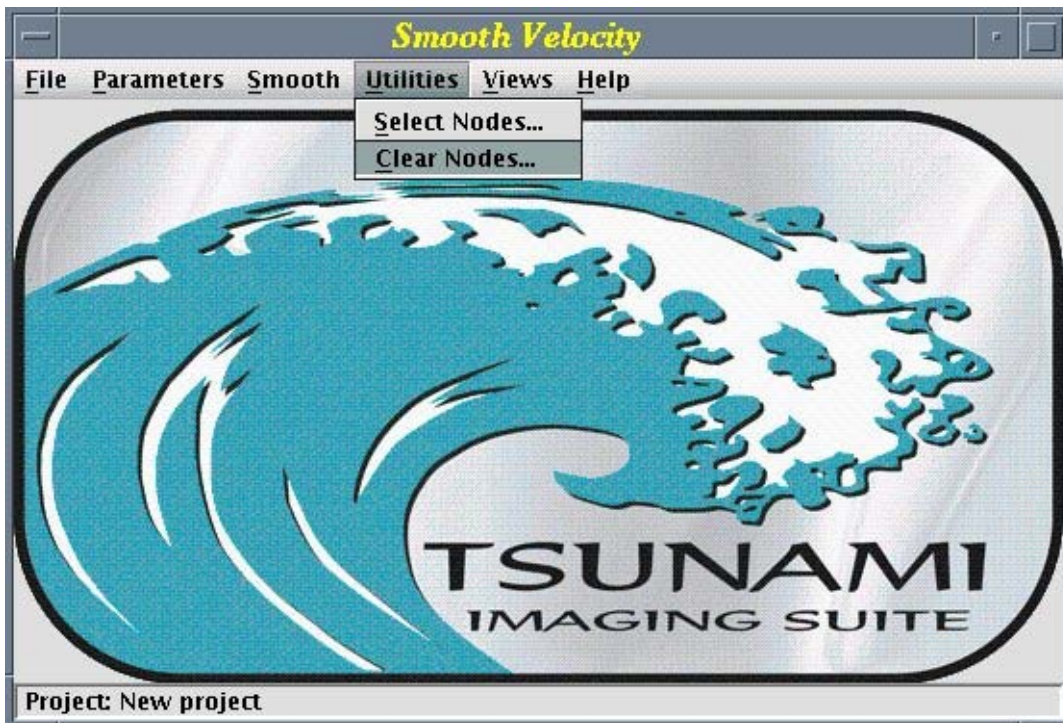
If a master node list is found and a project node list file is supplied in the node list (nddb) parameter of the parameter file (located on the processing panel of the parameters edit box), the dialog box will open and display the selected nodes. If no project node list is supplied, the dialog box will open and the user can select from the available nodes in order to create a new node list file.

When the user selects to save the node list file, the current file will automatically be entered into the node list (nddb) parameter of the parameter file (located on the processing panel of the parameters edit box).

For more information and help on the node list file and scaling a job on multiple nodes see the *Performance Considerations and Advanced Features* section.



- When a job is aborted or is stopped due to a hardware problem, Tsunami makes every effort to kill all the processes on the compute nodes. Sometimes it is not successful in killing all orphans so you must use the Clear Nodes Utility. Clear nodes will kill jobs running on the nodes listed in the node.db file you submitted to the clear\_nodes command, as well as reset the licenses for only those nodes in the node.db submitted to the clear\_nodes command. Clear nodes will not kill any processes on the master node. It is very likely that processes on the master node will abort as a result of killing the processes on the compute nodes, however you may need to verify this. Please see the *Floating License* section for more information. The clear nodes utility will create a clear\_nodes.log file in \$TSUNAMI.



### Clear Nodes from Command Line:

At the prompt type:

```
clear_nodes < node database file >
```

Clear nodes will kill jobs running on the nodes listed in the node.db file you submitted to clear\_nodes, as well as reset the licenses for only those nodes in the node.db submitted to the clear\_nodes command. Clear nodes will not kill any processes on the master node. It is very likely that processes on the master node will abort as a result of killing the processes on the compute nodes, however you may need to verify this. Please see the *Floating License* section for more information.

---

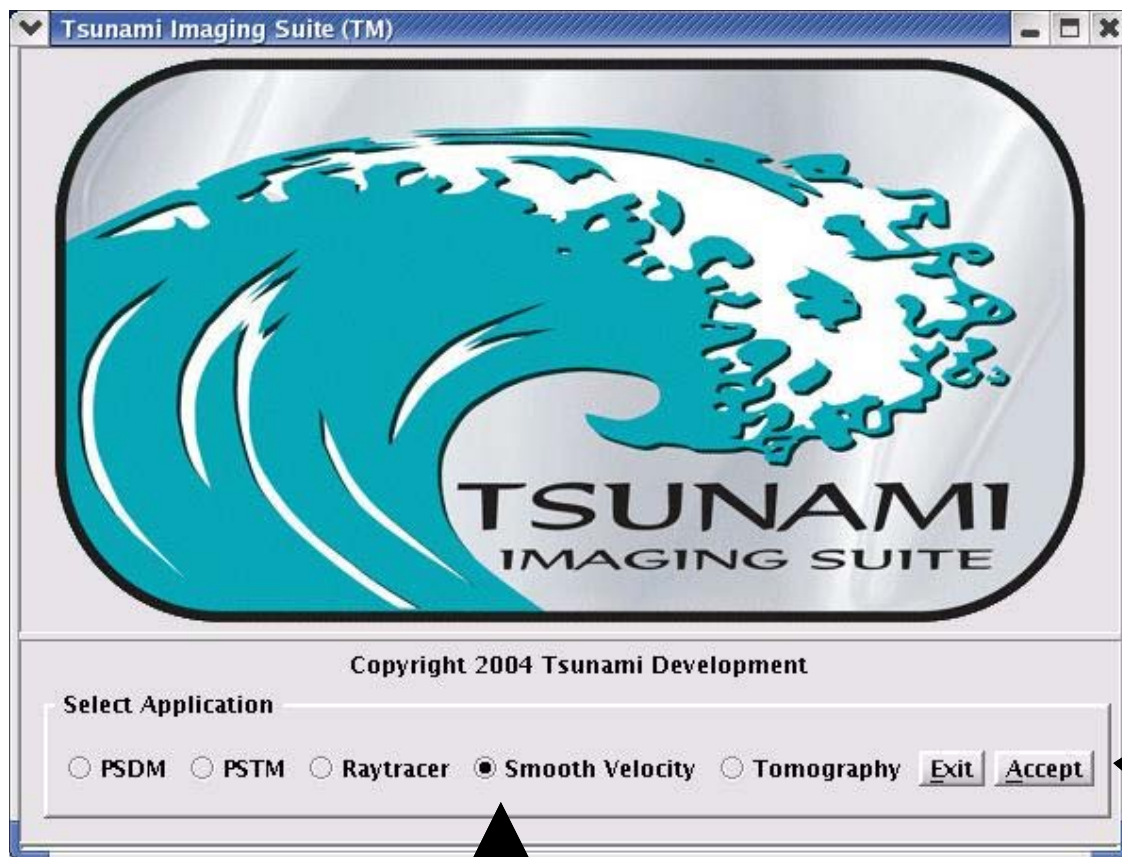
## Getting Started

---

1. Use the alias set up in step 2 of the GUI Installation section to start the Tsunami GUI.

Example: <prompt>: tsunami

2. Toggle the Smooth Velocity button on the Startup Window and click Accept.



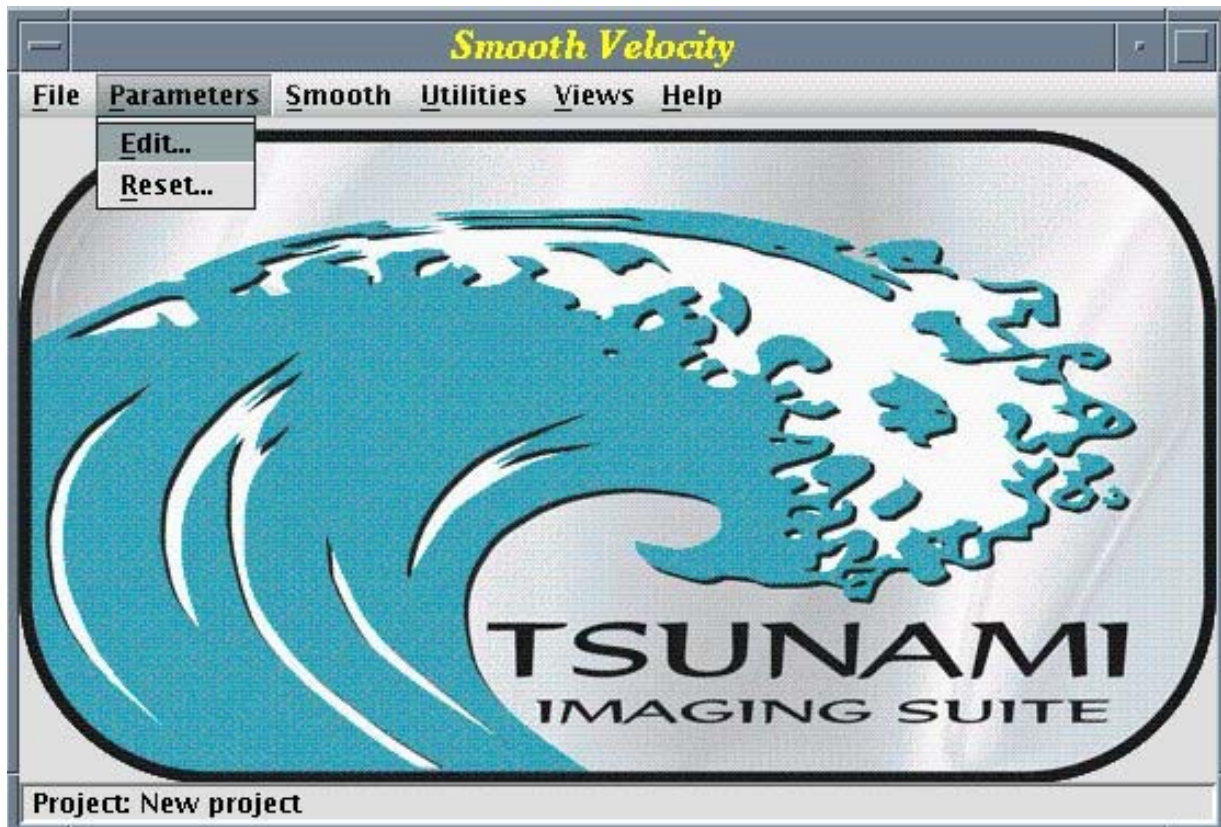
3. Select File —> New to begin new job.

---

# Generating a Smooth Velocity File

---

1. Select **Parameter** —> **Edit** from the Smooth Velocity startup window.



2. Select the **Files** tab in the **Parameter Selection Box**. Required information is highlighted in yellow.

## File Settings:

**Application Directory (hmdr):** Points to the version of Smooth Velocity that you plan to use. Ex. For version 4.17.12 it should point to ~/tsunami\_4.17.12/smith\_4.17.12

**Project Directory (prjdr):** The directory where the logfile will be located as well as other temporary files created by the job. It is also the directory where the scratch velocity block file is stored.

**Log file name (logfile):** If no log file name is entered, log file name defaults to project\_dir/jobname.log. If designating a specific name for your logfile, it is recommended that you use the full path to that logfile name.

**Input Interval Velocity File Name ( velf):** SEGY interval velocity file.

The velocity mode file is a set of interval velocities in a SEGY format. They must be stored in regular increments in the depth, xline, and inline directions. They also must be sorted by depth, xline and inline, with depth being the most rapidly varying dimension. It is important that the model be sorted by inline. Most processing systems will not sort the model by default. It is also important that the model provided be rectangular. All inlines must have the same number of xline traces

The format can be either IEEE or IBM format. If the model is in IEEE format it can be in either big or little endian format.

**Output SEGY Velocity File ( velfo):** Name and location of output SEGY velocity file. The SEGY velocity file can be used to QC the smoothing and to see the velocity model being used to calculate in the RayTracer. This model may also be used in the Tomography program.

**Output Velocity Block File (blkf):** The name and location of the velocity block file created by the program. This block file is for use in the RayTracer program. To use the block file in the RayTracer, you must point to the block file on the RayTracer Files tab (blkf=) and ensure that the Build New Block Model option is not selected on the RayTracer Velocity File.

**Please note:** You must have one output format selected or you can create **both** the SEGY and block file. If you plan to use Anisotropy for your depth migration, do not create a block file. You must create the block file in the Raytracer with your smoothed SEGY velocity file and the anisotropy parameters turned on.

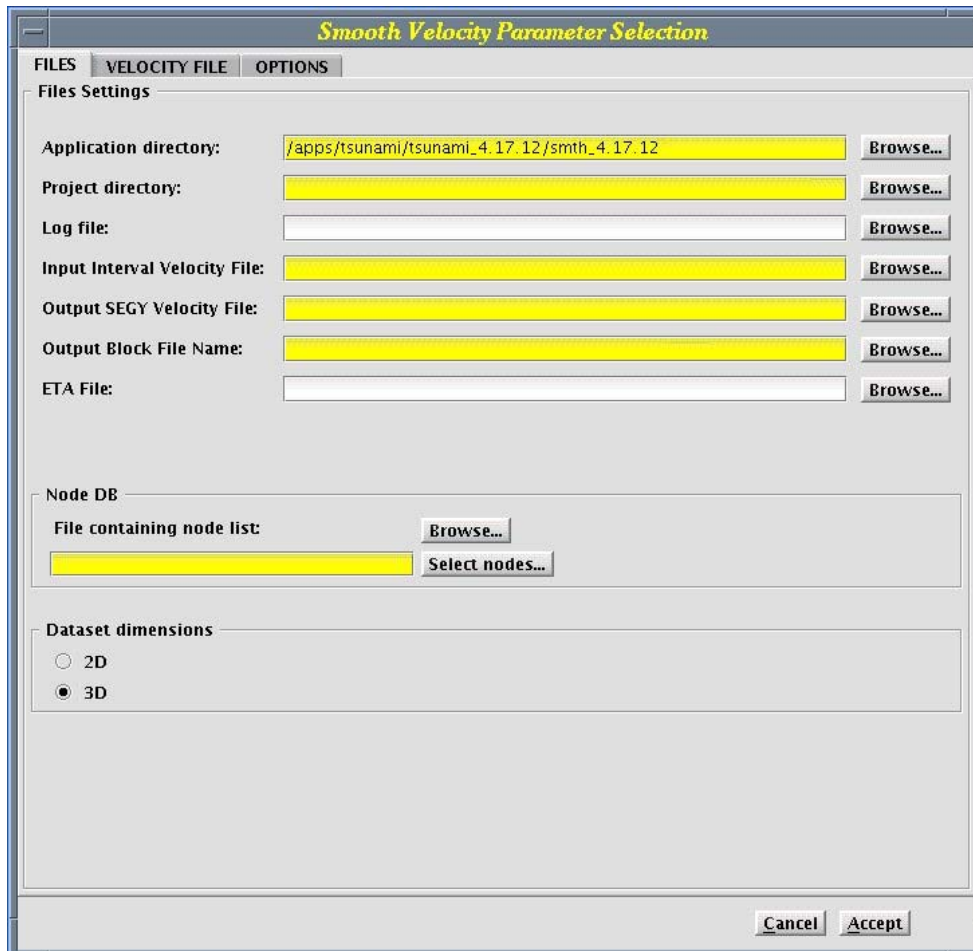
**ETA File (etaf):** The name and location of the SEGY file of ETA values. ETA values must be sampled at the same inline/xline increment and depth increment as the velocity model. They must be in SEGY format and the inline/xline header values need to be in the same location and same format as the velocity model.

## **Node List:**

**File containing node list (nddb):** List of node names to use in job. For more information on the node list file please refer to the *Cluster Configuration* section.

**Browse:** Opens dialog box to select node file.

**Select Nodes:** Allows you to edit the selected node file. From this window you may add new nodes to your list, delete selected nodes, and turn nodes on/off for your job.



### Parameter File For Command Line Use:

The **Files Setting** page would be represented in the command line parameter file by the following list of parameters:

**hmdr**= home directory of software  
**prjdr**= project directory  
**logfile**= logfile name  
**velf**= SEGY interval velocity file  
**velfo**= SEGY smoothed velocity output  
**blkf**= Velocity Block File Created by the program  
**nddb**= node database file  
**twod**= dataset dimensions

For more information on creating the command line parameter file and complete list of required parameters please see *Appendix B: Parameter File Information*.

3. Select the **Velocity File** tab on the **Parameter Selection** box. Enter in necessary information. Required information is highlighted in yellow.

**Header Locations:**

**Header loc of Xline (vxlb):** The byte location of Xline labels in velocity model headers. (No Default)

**Header loc of inline (vilb):** The byte location of the inline labels in the model headers. ( No Default)

**Depth Increment (vdz):** Depth increment of the velocity model.

**Velocity File Label (vxlilty):** Velocity file label integer. Default is 0 or 4 byte integer. 0 = 4 byte integer and 1= 2 byte integer.

**Distance:**

**Distance between xlines (xldist):** distance between xlines in the survey. (No Default)

**Distance between inlines (ildist):** distance between inlines in the survey.

**Smooth Velocity Parameter Selection**

FILES VELOCITY FILE OPTIONS

Velocity File Settings

Inline/Xline Byte Location

Inline Byte Location: [Yellow Input Field]

Xline Byte Location: [Yellow Input Field]

Velocity file label

2 bytes

4 bytes

Velocity Depth Increment

Velocity Depth Increment: [Yellow Input Field]

Distance

Distance between inlines: [Yellow Input Field]

Distance between xlines: [Yellow Input Field]

Velocity Model Format

IEEE Little Endian

IEEE Big Endian

IBM

Velocity Model Header Format

IEEE Little Endian

IEEE Big Endian

IBM

Velocity Model Format Output

IEEE Little Endian

IEEE Big Endian

IBM

Cancel Accept

**Velocity Model Format (dataf):** Indicates the format of the velocity model headers.  
Default to the input format or IEEE\_LE, IEEE\_BE or IBM. (Default is IBM).

**Velocity Model Header Format (hdrfmt):** Indicates the format of the velocity model header.  
Default to the input format or IEEE\_LE, IEEE\_BE or IBM. (Default is IEEE\_BE).

**Trace data format output (datafo):** Format of the output trace amplitudes. Choice of IEEE Little Endian, IEEE Big Endian or IBM. Default is same as the input data format.

#### **Parameter File For Command Line Use:**

The **Velocity File** page would be represented in the command line parameter file by the following list of parameters:

- vxlb=** The byte location of Xline labels in velocity model headers. (No Default)
- vilb=** The byte location of the inline labels in the model headers. ( No Default)
- vdz=** Depth increment of the velocity model.
- xldist=** Distance between xlines in the survey. (No Default)
- ildist=** Distance between inlines in the survey. ( No Default)
- dataf=** Indicates the format of the velocity model headers. Default to the input format or IEEE\_LE, IEEE\_BE or IBM. (Default is IBM).
- hdrfmt=** Indicates the format of the velocity model header. Default to the input format or IEEE\_LE, IEEE\_BE or IBM. (Default is IBM).
- vxliltyp=** Velocity file label integer. (Default is 4 byte integer)
- datafo=** Indicates the format fo the output velocity model headers. No Default.

For more information on creating the command line parameter file and complete list of required parameters please see *Appendix B: Parameter File Information*.

4. Select the **Options** tab from the **Parameter Selection Box**. Required items are highlighted in yellow.

**Smoothing Operator Controls:**

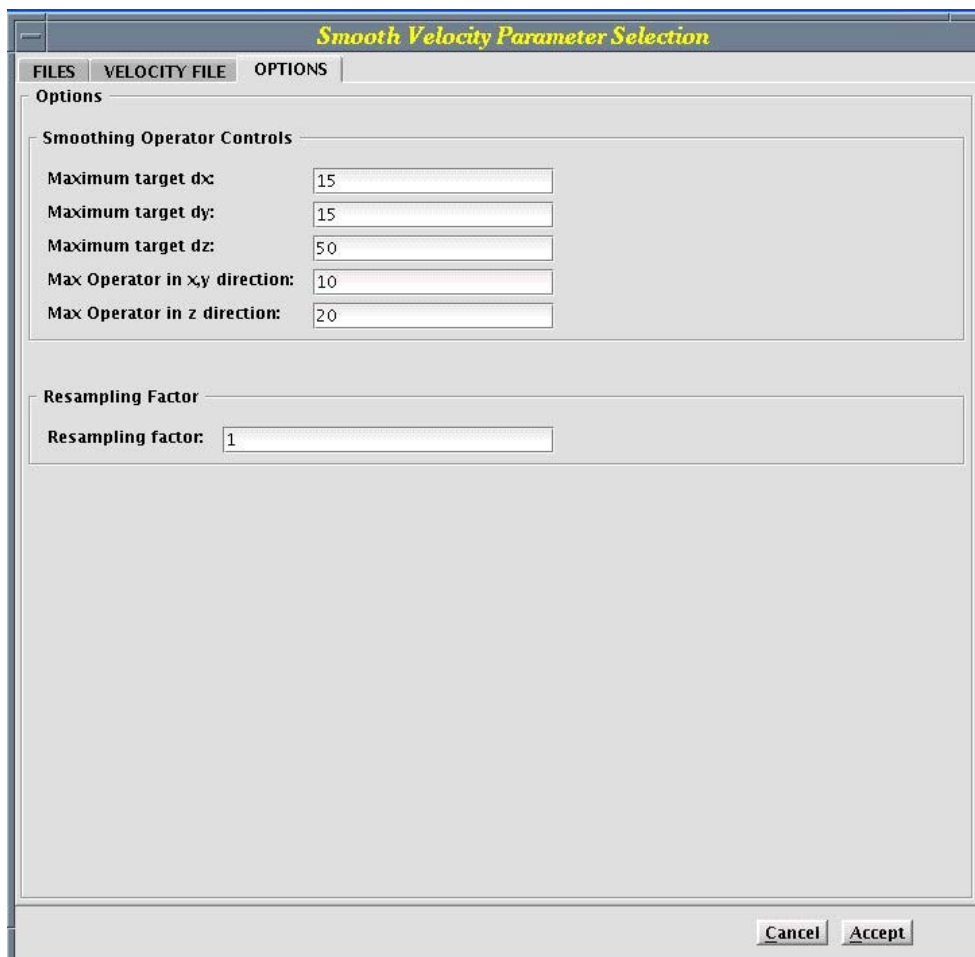
**Maximum Target dx (dxmx):** Max derivative that will exist after smoothing.  
Default = 15

**Maximum Target dy (dymx):** Max derivative that will exist after smoothing.  
Default = 15

**Maximum Target dz (dzmx):** Max derivative that will exist after smoothing.  
Default = 50

**Max Operator in dx, dy direction (mxopxy):** Default is 10

**Max Operator in dz direction (mxopz):** Default is 20. This is operator length after resampling if applied



The smoother will increase the size of the operator until the max resulting dx, dy, dz derivative is less than what the user has selected. This allows the user to get the desired velocity derivatives without having to guess at the smoothing factor size.

Once the operators in the three dimensions are selected the same operators are applied over the full model. X, Y and Z operators can be different from each other.

To avoid smoothing in a particular dimension, set that max dx, dy or dz to a very large number.

Your resulting log file will list your starting max dx, dy and dz as well as your final max dx, dy and dz. Please check your log to ensure correct values. An example log file can be found in *Appendix A: Log File Information*.

**Resampling Factor (resamp):** Default = 1. The resample simply divides the sample increment by the resample factor, so a three for the resample factor will produce traces with three times as many samples and one third the sample increment. This will make the velocity file larger. The performance of the raytracer may be slowed if a large resampling factor is used in conjunction with a large file. The sampling increment of the velocity file has no bearing on the runtime or output sampling of the depth migration itself. The sampling increment can be a fractional amount so a 9 meter sample can be divided into four samples of 2.25 meters each. You must remember to put the new sample size into the Raytracer when you run it with the smoothed model. Your new sample size can be found in the logfile. Please see *Appendix A: Logfile Information* for an example.

---

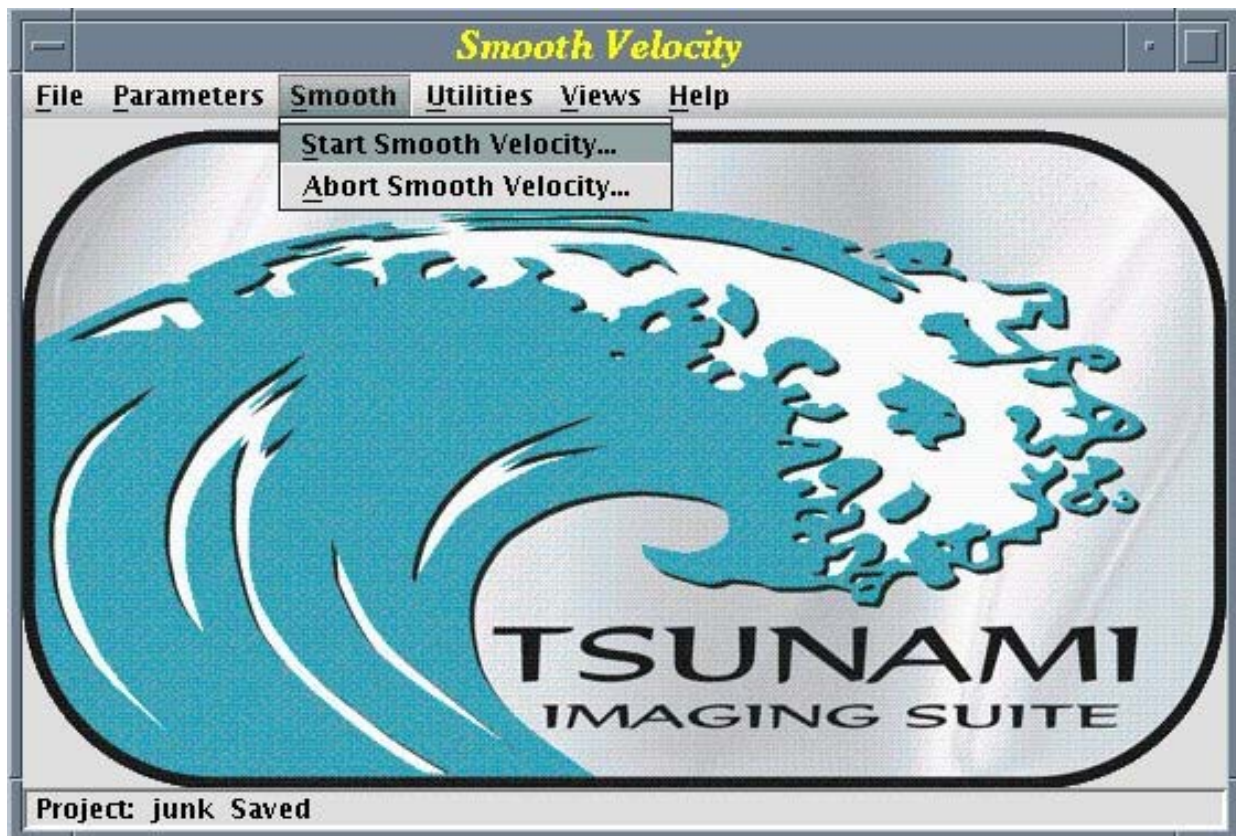
# Stopping and Restarting a Job

---

## Stop/Restart Job from GUI

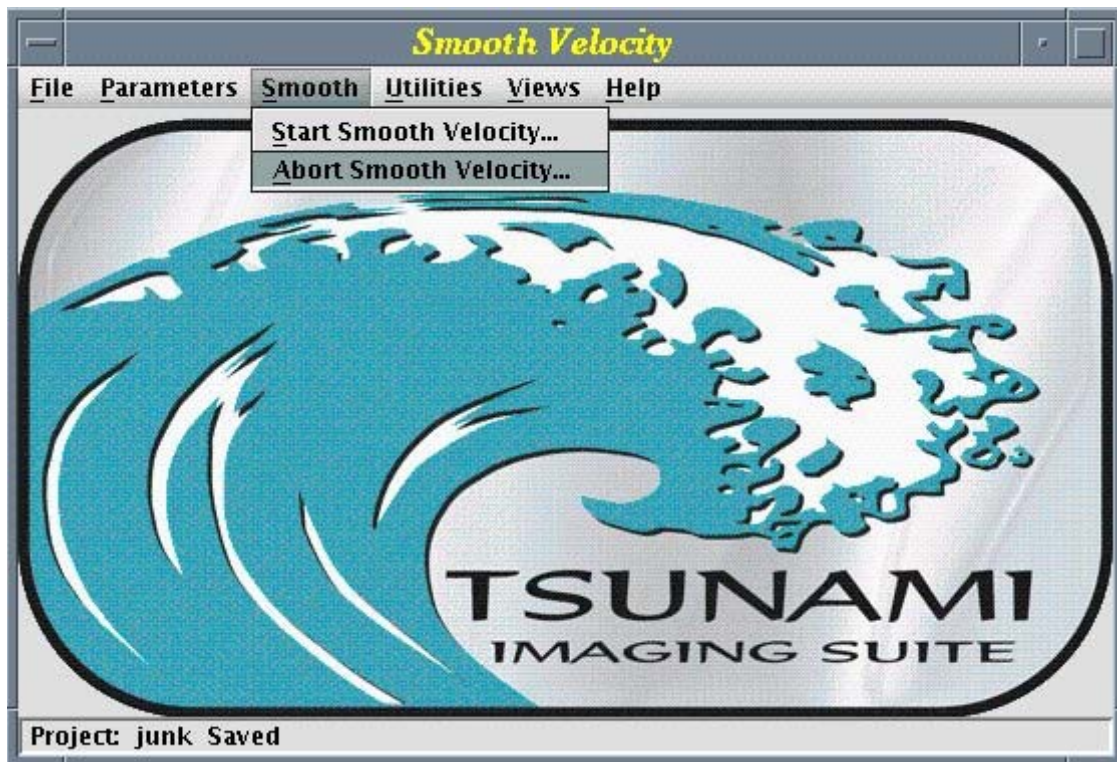
### Smooth Velocity - Start

Starts a job with the current parameter file. The current parameter file is displayed on the top bar of the user interface. Once the file is started, the status bar will display "Starting Smooth Velocity for file: (file name)."



### Smooth Velocity - Abort

Stops the job immediately. This will abort the job, perform necessary cleanup and prevent the job from restarting.



### **Abort/Restart from Command line:**

To start a job type:

```
smth_start < parameter file name > &
```

To abort a job type:

```
rays_abort < parameter file name >
```

This will abort the job and perform all the cleanup.

---

## Appendix A: Log file Information

---

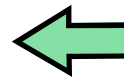
The log file is located in `~proj_dir/parameter_file.log`

Smooth Velocity provides a great deal of information in the log file. If you are running a new dataset through Smooth Velocity it is critical that you verify that the program is reading the velocity data correctly.

To assist in this effort, at the start of the job the program will print out the range of inlines and xlines covered in the velocity file. It will print out the minimum and maximum velocity values used in the job, and it will print out header data from the first trace it finds in the input data files. These should be checked to verify that the data formats are being read correctly.

```
Version 4.17 Tsunami Velocity Smoother
Parameter file: smth_29.param

hmdr= '/data3/software/tsunami_4.17.10/smith_4.17.10'
velf= '/data1/summit/grid32_Vint_ibm.segy'
velfo= '/data3/smith_test/summit/grid32_Vint_ibm_smth-1.segy'
blkf= '/data3/smith_test/summit/smooth.blk'
nddb= '/data3/smith_test/summit/node.db'
prjdr= '/data3/smith_test/summit'
ildist= 82.5
xldist= 82.5
resamp 3
vilb= 9
vxlb= 17
vdz= 32
dxmx= 15
dymx= 15
dzmx= 50
mxopxy= 20
mxopz= 20
Sample rate in header 32000
Reading the velocity model
Checking the sort order, and the limits of model
Read 10000 velocity traces
Read 20000 velocity traces
Read 30000 velocity traces
Read 40000 velocity traces
Read 50000 velocity traces
Read 60000 velocity traces
Read 70000 velocity traces
Read 80000 velocity traces
Read 90000 velocity traces
Read 100000 velocity traces
Read 110000 velocity traces
Read 120000 velocity traces
Read 130000 velocity traces
Read 140000 velocity traces
```



### CHECK 1

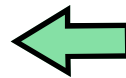
Shows parameters you input either through GUI or in your parameter file. Check the `vdz=`, `vilb=`, `vxlb=`, `xldist=` and `ildist=` parameters for correct values. Please see *Appendix B: Parameter File Information* for parameter definitions.



Length of smoothing operator x 11 y 12 z 8  
Minimum vel in model 5413.09  
Maximum vel in model 14500.00  
Initial dx 108.3 dy 108.3 dz 281.0  
Final dx 14.9 dy 14.2 dz 46.9

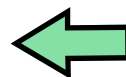
Depth	Ave Shift
961.60	0.80
1923.20	1.65
2884.80	2.63
3846.40	3.75
4808.00	4.85
5769.60	5.79
6731.20	6.30
7692.80	6.85
8654.40	7.57
9616.00	8.19
10577.60	8.65
11539.20	9.14
12500.80	9.71
13462.40	10.94
14424.00	13.04
15385.60	15.32
16347.20	17.47
17308.80	19.45
18270.40	21.09
19232.00	22.71

Successful Completion: Smooth Velocities



**CHECK 3**



Verify final dx, dy, dz are less than values input for dxmx, dymx, dzmx.



**CHECK 4**

Average shift is less than one depth sample interval. (vdz= 32) Resample allows operator smoothing with less shifts in depth . This creates a layer velocity file.

**Example Check License Log File:**

Product name rays			 <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p><b>NOTE</b> Beginning and ending date of license.</p> </div>
Creation date 010104			
Start date 010104			
End date 123104			
Length of hw record 96			
Number of licensed nodes 100			
Mac Address	Node Name	Checked Out	
00:A0:C9:FB:25:B4	dual450		
00:42:52:00:13:E1	rlx-0-0-1		
00:42:52:00:0F:37	rlx-0-0-2		
00:42:52:00:13:36	rlx-0-0-3	1	
00:42:52:00:11:38	rlx-0-0-4		
00:42:52:00:23:0B	rlx-0-0-5		
00:42:52:00:0B:EF	rlx-0-0-6		
00:42:52:00:18:FD	rlx-0-0-7	1	
00:42:52:00:1C:3C	rlx-0-0-8	1	 <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p><b>NOTE</b> A 1 indicates that the node is checked out.</p> </div>
00:42:52:00:19:27	rlx-0-0-9		
00:42:52:00:25:B4	rlx-0-0-11		
00:42:52:00:13:24	rlx-0-0-13		
00:42:52:00:17:26	rlx-0-0-15		
00:42:52:00:1A:35	rlx-0-0-17		
00:42:52:00:1A:0B	rlx-0-0-19		
00:42:52:00:18:7C	rlx-0-0-21		
00:E0:81:03:38:C6	linux1		
00:03:47:71:D2:54	linux2	1	
00:03:47:71:E9:F5	linux3		
00:03:47:6B:45:47	linux4	1	
00:E0:81:02:B4:3E	linux5		
00:03:47:71:62:FC	linux6		
00:03:47:71:5B:73	linux7		
00:03:47:71:D2:41	linux8		
00:03:47:71:65:23	linux9		
00:03:47:71:62:2D	linux10		
00:03:47:71:5D:B9	linux11		
00:03:47:71:5D:BB	linux12	1	
00:03:47:71:5B:7C	linux13	1	
00:03:47:71:D2:49	linux14	1	
}			

## Example Clear Nodes Log File:

```
clearing node rlx-0-0-1
clearing node rlx-0-0-2
clearing node rlx-0-0-3
clearing node rlx-0-0-4
clearing node rlx-0-0-5
clearing node rlx-0-0-6
clearing node rlx-0-0-8
clearing node rlx-0-0-9
clearing node rlx-0-0-11
clearing node rlx-0-0-13
clearing node rlx-0-0-15
clearing node rlx-0-0-17
clearing node rlx-0-0-19
clearing node rlx-0-0-21
Resetting license file /apps/tsunami/pstm_license.dat
Reading license file /apps/tsunami/pstm_license.dat
Resetting license file /apps/tsunami/psdm_license.dat
Reading license file /apps/tsunami/psdm_license.dat
```

---

# Appendix B: Parameter File Information

---

## Required Parameters

- hmdr=** Application home directory  
Required
- prjdr=** Project directory  
Required
- velf=** This is a segy interval velocity file  
Same rules apply, must be regularly sampled in inline, xline and depth  
Required  
No Default
- blkf=** This is the name of the velocity block file created by the program  
Required if Output SEGY Velocity File not being created. The block file and SEGY file may be created at the same time.  
No Default
- velfo=** The name of the output SEGY Velocity File  
Required if Block file not being created. The block file and SEGY file may be created at the same time.  
No Default
- xldist=** Distance between xlines  
Required  
No Default
- ildist=** Distance between inlines  
Required  
No Default
- vilb=** Header location of inline in velocity model  
Required  
No Default
- vxlb=** Header location of xline in velocity model  
Required  
No Default
- vdz=** Depth increment of the velocity model  
Required  
No Default
- nddb=** Node database - See Cluster Configuration for a description  
Required  
No Default

## Alphabetical List of Parameters

- blkf=** This is the name of the velocity block file created by the program  
Required  
No Default
- dataf=** Format of velocity model  
0= IEEE\_LE  
1= IEEE\_BE  
2 = IBM  
Default = 2 or IBM
- datafo=** The output format of the trace amplitudes.  
No Default  
0=IEEE little endian  
1=IEEE big endian  
2= IBM
- dxmx=** Maximum targeted dx.  
Default is 15.0
- dymx=** Maximum targeted dy.  
Default is 15.0
- dzmx=** Maximum targeted dz.  
Default is 50.0
- etaf=** ETA file name  
No Default
- hdrfmt=** Format of velocity model headers  
0= IEEE\_LE  
1= IEEE\_BE  
2 = IBM  
Default = 1
- hmdr=** Application home directory
- ildist=** Distance between inlines  
Required  
No Default
- logfile=** User specified logfile name

**mxopxy=** Max Operator in the x, y direction  
Default = 10

**mxopz=** Max Operator in the z direction  
Default = 20

**nddb=** Node database - See Cluster Configuration for a description  
Required  
No Default

**prjdr=** Project directory

**resamp=** Resampling Factor  
Default = 1

**twod=** 2D/3D switch  
0 = 3D (threed)  
1 = 2D (twod)  
Default= 0 or 3D

**vdz=** Depth increment of the velocity mode  
Required  
No Default

**velf=** This is a segy interval velocity file. Same rules apply, must be regularly sampled in  
Inline, xline and depth  
Required  
No Default

**velfo=** Output SEG Y Velocity File  
No Default

**vilb=** Header loc of inline in velocity model  
Required  
No Default

**vxlb=** Header location of xline in velocity model  
Required  
No Default

**vxlityp=** Velocity file label integer  
Default = 0  
0 = 4 byte integer  
1 = 2 byte integer

**xldist=** Distance between xlines  
Required  
No Default

## Example Parameter File

The parameters can be in the parameter file in any order. Anything not a parameter is taken as a comment. All filenames, and pairs of numbers must be enclosed with single quotes. All parameters must include the equal sign, that is attached to the parameter string with no spaces. Parameter values should be separated from the equal sign by a space.

```
hmdr= '/data3/software/tsunami_4.17.10/smith_4.17.10'  
velf= '/data1/test/grid32_Vint_ibm.segy'  
velfo= '/data3/smith_test/test/grid32_Vint_ibm_smith-1.segy'  
blkf= '/data3/smith_test/test/smooth.blk'  
nddb= '/data3/smith_test/test/node.db'  
prjdr= '/data3/smith_test/test'  
ildist= 82.5  
xldist= 82.5  
resamp 3  
vilb= 9  
vxlb= 17  
vdz= 32  
dxmx= 15  
dymx= 15  
dzmx= 50  
mxopxy= 20  
mxopz= 20
```

---

## Appendix C: Java Installation Information

---

For the Tsunami GUI and Viewer you must have version 1.4.2 or higher of Java 2 Platform, Standard Edition (J2SE) installed along with Java 3D. Should java not be installed on your system, your system administrator can download and install using the following steps.

### To Check For Correct Java Version on Your Machine:

- 1) `java -version`  
This will give you the version number of your java installation.
- 2) `which java`  
This will tell you where your java installation is located.

Should these commands return no information, please ensure that the `JAVA_HOME` environment variable is set in the users `.cshrc` or `.bashrc` file. Your system administrator may need to be consulted in order to correctly set the path for `JAVA_HOME`.

### To Download and Install Java and Java 3D on Linux and Itanium Platforms:

- 1) Go to Java website via following link.  
<http://java.sun.com/j2se/index.jsp>
- 2) Select the latest release (non Beta version). Must be J2SE SDK version 1.4.2 or higher. Please note: you must download J2SE version 5.0 if using AMD Opteron 64 bit linux.
- 3) Please read all installation documentation on the Java website regarding J2SE SDK for your particular platform. The self extracting binary file method is recommended.
- 4) Download self extracting binary file to desired location on your system. Please be sure to download the appropriate executable file for your platform.  
Example: Linux 32 bit vs. Linux 64 bit Itanium 2
- 5) Follow java installation directions to correctly install J2SE SDK version 1.4.2 or higher on your machine.
- 6) Edit users `.cshrc` or `.bashrc` file to include `JAVA_HOME`. Set path to directory created in java install in step 5.

For `.cshrc`: `setenv JAVA_HOME path`

For `.bashrc`: `export JAVA_HOME="path"`

- 7) Add JAVA\_HOME/bin to PATH variable.

**For cshell:** PATH is located in the **.login** file.

Add the JAVA\_HOME directory/bin to the end of the existing set path= variable.

Ex: set path=(/bin /usr/bin /sbin /usr/etc /usr/local/bin /usr/j2se\_1.4.2/bin )

Where /usr/j2se\_1.4.2 is JAVA\_HOME

**For bourne shell:** PATH is located in the **.bash\_profile** file. Add JAVA\_HOME/bin to the end of the existing PATH variable.

Ex: export PATH=\$PATH:\$HOME/bin:\$JAVA\_HOME/bin

- 8) To ensure correct set up, please perform the following commands:

- a) java -version

This will give you the version number of your java installation.

- b) which java

This will tell you where your java installation is located.

- 9) Once J2SE SDK version 1.4.2 or higher has been installed correctly, download Java 3D SDK Version 1.3.1 or higher (fcs or non-Beta version only) into the directory created in step 5 from the following website:

<http://www.blackdown.org/java-linux/java-linux-d1.html>

Please ensure that you have downloaded the correct version for your platform.

Example: Linux 32 bit vs. Linux 64 bit Itanium 2

Please be sure to download the Java 3D license and Readme file as well.

- 10) Please read all installation documentation on the Java/Blackdown website regarding Java 3D SDK Version 1.3.1 or higher for your particular platform before install to ensure all requirements met.
- 11) Run tests detailed in installation documentation to verify correct installation of Java 3D SDK Version 1.3.1 or higher on your system.

## To Download and Install Java and Java 3D on SGI :

- 1) Ensure that Java SDK version 1.4.1\_06 or higher for IRIX is not installed. This install will **overwrite** any previous installation of Java SDK version 1.4.1 or lower.
- 2) Please read and verify all requirements and installation instructions from the SGI website before downloading the executable.

[http://www.sgi.com/products/evaluation/6.5\\_java2\\_1.4.1\\_06/](http://www.sgi.com/products/evaluation/6.5_java2_1.4.1_06/)

Select the **Check Requirements** button at bottom of page.

After viewing requirements, please read all installation documentation. Select the **Install** button at bottom of Check Requirements page. Agree to license agreement by selecting the **Accept License** button at bottom of page. Fully read the Installation Instructions page and then select the **Troubleshooter** button at bottom of page to see more install tips. Once all install documentation has been read, go back to the Installation Instructions page and select the **Install** button for java2\_eoe.

- 3) After installation, edit users .cshrc or .bashrc file to include JAVA\_HOME.  
Set path to directory created in java install in step 2.

For .cshrc: setenv JAVA\_HOME path

For .bashrc: export JAVA\_HOME="path"

- 4) Add JAVA\_HOME/bin to PATH variable.

**For cshell:** PATH is located in the **.login** file.

Add the JAVA\_HOME directory/bin to the end of the existing set path= variable.

Ex: set path=(/bin /usr/bin /sbin /usr/etc /usr/local/bin /usr/java2/bin )

Where /usr/java2 is JAVA\_HOME

**For bourne shell:** PATH is located in the **.bash\_profile** file. Add JAVA\_HOME/bin to the end of the existing PATH variable.

Ex: export PATH=\$PATH:\$HOME/bin:\$JAVA\_HOME/bin

- 5) To ensure correct set up, please perform the following commands:

- a) java -version

This will give you the version number of your java installation.

- b) which java

This will tell you where your java installation is located.

6) Prepare to install Java 3D version 1.3.1 or higher for IRIX by checking requirements on the following page:

[http://www.sgi.com/products/evaluation/6.5\\_java3d\\_1.3.1/](http://www.sgi.com/products/evaluation/6.5_java3d_1.3.1/)

Select the **Check Requirements** button at bottom of page.

After viewing requirements, select the **Continue** button at bottom of page. Agree to license agreement by selecting the **Accept License** button at bottom of page. Fully read the Installation Instructions page and the **More Instructions** page located in step 3 of Installation Instructions page. Then select the **Troubleshooter** button at bottom of page to see more install tips. Once all install documentation has been read, go back to the Installation Instructions page and select the **Install** button located in step 2.