



RayTracer

Программа лучевого трассирования

версия 5.3.14

Описание

Tsunami Development

tsunamidevelopment.com

+1-713-783-1435

Tsunami Development

713-783-1435 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.

СОДЕРЖАНИЕ

Введение.....	4
Часто Задаваемые Вопросы.....	5
Инсталляция программы	6
Системные требования	6
Установка Tsunami	6
Установка графического интерфейса GUI.....	8
Плавающая лицензия	9
Конфигурация кластера	10
Утилита нодов	11
Начало работы	15
Создание файла времен пробега	16
Страница File Settings:	16
Страница Input	18
Страница Velocity File	20
Страница Ray Shooting.....	23
Страница Output	27
Остановка и перезапуск задания.....	31
Остановить/Перезапустить задание из GUI.....	31
Остановка/Перезапуск задания из командной строки:	32
Анализ производительности и улучшенные возможности	33
Временный блок файл скоростной модели.....	33
Миграция от плавающего уровня или топографии	33
Приложение А: Информация лог файла	37
Пример лог-файла RayTracer	37
Проверка лог-файла лицензии	41
Пример лог файла очистки нодов:.....	42
Приложение В: Информация файла параметров	43
Необходимые параметры.....	43
Список параметров в алфавитном порядке	43
Пример файла параметров.....	47
Приложение С: Инсталляция Java	48

Введение

Raytracer - программа для создания времен пробега для глубинной миграции до суммирования (PSDM). Она трассирует лучи, используя решение уравнения Рунга-Кутта IV порядка, и поддерживает 3 метода выбора времен: минимальное время, минимальное расстояние, максимальная энергия. Для трассирования лучей Raytracer использует метод восстановления волнового фронта для поддержания точности во время прохождения лучей через скоростную модель. Пользователь может контролировать интерполяцию и сглаживание времен пробега, а также плотность точек записи. Может использоваться скоростная модель любого размера. Однако если объем скоростной модели превышает суммарную память на всех узлах, то производительность может снизиться.

Лучи трассируются группами, пучками или другими словами группой треугольников: по три соседних луча. Время пробега каждого луча экстраполируется в общую точку в пределах пучка и затем сравнивается между собой. Если времена лучей отличаются больше дозированной величины, то создается новый луч и соответствующий пучок. Поэтому, при рассеивании лучей, точность лучевого фронта сохраняется.

Размер шага для трассирования данного луча определяется автоматически и зависит от градиента скорости. При увеличении градиента скорости шаг уменьшается. Когда изменение угла, зависящее от изменения скорости, увеличивается, ошибка при решении уравнения Рунга-Кутта увеличивается.

Лучи распространяются радиально от источника, со всеми лучами в пределах пучка шаг за шагом. В точке производится поиск для гридов, расположенных внутри каждого пучка. Времена пробега рассчитываются для любых гридов внутри данного пучка. Если грид в пределах нескольких пучков, то чтобы определить, какой пучок использовать для расчета времен пробега, выбирается критерий: минимальное время, минимальное расстояние, максимальная энергия.

Можно использовать несколько выходных файлов времен пробега Raytracer для одной миграции. Вы можете создавать несколько файлов из-за их размеров или из-за неравномерной расстановки источников в пределах проекта.

Все интерполяции времен пробега, как в пределах «Raytracer», так и в «PSDM» совершаются в единицах медленности, а не в секундах. Это сильно повышает точность интерполяции и позволяет не потерять точность вычислений при увеличении шага точек запуска лучей и шага между точками их отражения.

Raytracer запускается из основного окна графического интерфейса, а также из командной строки. Если запуск задания происходит из окна графического интерфейса, то автоматически создается файл параметров, и задание начинает считаться. Можно создать этот файл вручную, а затем запустить задание из командной строки. Оба пути будут рассмотрены в этой документации. Для каждого из них ниже приведено описание с подробным перечислением используемых параметров.

Часто Задаваемые Вопросы

У меня версия java 1.4.2 или выше описана в файле `.cshrc` (`.bashrc`), но GUI её не распознаёт

- 1) Проверьте, не прописана ли в вашем `.cshrc` (`.bashrc`) файле строка: `source ~Epos3_env`. Эта строка переписывает заданную вами директорию `$JAVA_HOME` и использует java для программы Paradigm / Geodepth.
- 2) Некоторые версии Linux имеют свою директорию для java. Чтобы её переписать, необходимо поставить ваш `$JAVA_HOME/bin` первым в ваш `$PATH` в файле `.cshrc/.bashrc_profile`.

- **Я не могу найти `node.db` файл в окне выбора нодов Node Selection в GUI.**

С помощью текстового редактора, проверьте свой файл `node.db` на предмет опечаток, дополнительных пробелов, и т.д. Если вы его делали на PC, то вам необходимо использовать команду `dos2unix`.

- **У меня постоянно появляется сообщение об ошибке: OS Error: could not map rld from file /lib32/rld. There are not enough file descriptors available.**

Наберите в командной строке для `ssh`

```
>limit descriptors 500 и проверьте, набрав
```

```
>limit
```

для `bash`

```
>ulimit -n 500 и проверьте, набрав
```

```
>ulimit -a
```

- **Как мне создать `rsh` для моей системы?**

<http://evuraan.blogspot.com/2005/02/how-to-turn-on-rsh-and-rlogin-on.html>

- **Я запустил задание на счет из GUI, но оно не считается.**

Убедитесь, что команды `rsh` и `rcp` исполняются:

```
>rsh node_name date
```

```
>rcp some_file node_name: /tmp
```

Проверьте, есть ли команда Raytracer Start в терминальном окне, из которого вы запустили Tsunami. Убедитесь, что вы выбрали правильный `node.db` файл.

Инсталляция программы

Системные требования

Пакет Tsunami устанавливается на любых системах или их комбинациях: Linux, SGI, Opteron, Solaris и Itanium. Linux должен быть версии 2.4 или выше, с компилятором *gcc* 3.2, SGI IRIX версии 6.2 или выше, а также Itanium должен иметь Intel v8.0 компилятор.

У Tsunami удобная система инсталляции, которая позволяет соединять в один кластер различные версии операционных систем. Более того, ноды могут работать под различными версиями Linux. Пожалуйста, обратитесь в Tsunami Development, если вам необходимо соединить несколько версий Linux ОС.

Для расчета PSTM необходимо всего 256 MBytes памяти на каждый процессор, для расчета PSDM и Raytracer - 512 Mbytes. Меньший объем памяти может повлиять на производительность и время счета.

На вычислительных нодах не требуется дисковое пространство. Жесткий диск с данными подключается только к главному ноду. На вычислительных нодах должна существовать и быть доступной папка */tmp*. Во время работы, Tsunami мастер нод рассылает executable файлы на каждый из нод, и там же создаются локальные лог файлы.

Для работы Tsunami рекомендуется не использовать NFS систему на кластере. С увеличением количества процессоров до 100, NFS вызывает зависание процессов.

Команды *rsh* и *rcc* должны быть включены. Tsunami их использует для запуска процессов на вычислительных нодах. Поэтому пользователю необходимо открыть допуск и разрешение. Протестировать эти команды можно, набрав в командной строке следующее:

```
>rsh <node name> date  
  
>rcc <file> node_name:/tmp
```

Для большинства инсталляций достаточно 100 Mbit сети. Каждый нод должен иметь 100 Mbit переключатель. Этого вполне достаточно для работы Tsunami.

Установка Tsunami

1. Загрузите последнюю версию Tsunami, в формате tar с веб-сайта Tsunami Development.
2. Перейдите в директорию *apps* `>cd /apps`
3. Создайте директорию *tsunami* `>mkdir tsunami`
4. Скопируйте tar файл в директорию *tsunami*, созданную в п.3.
5. Раскройте tar файл: `>tar -xvpf tsunami.tar`

У вас должны появиться следующие модули:

```
apps/tsunami/tsunami_5.2.13/pstm_5.2.13
    psdm_5.2.13
    rays_5.2.13
    tomo_5.2.13
```

В директории каждого модуля содержится *readme* файл, а также директории Linux, SGI, Opteron, Solaris and Itanium с программными файлами. Лицензия контролирует, какие именно выполняемые файлы вы будете использовать. Конечная директория должна выглядеть следующим образом:

```
/apps/tsunami/tsunami_5.2.13/rays_5.2.13/itanium
    itanium_3.5
    linux
    linux_3.4
    opteron
    opteron32
    opteron_3.4
    sgi
    solaris
```

6. По электронной почте получите файл с лицензией от Tsunami Development. Сохраните его в директории *tsunami*, созданной в п. 3.

Для инсталляции на SGI: Нет необходимости в дополнительных действиях.

Для инсталляции на Linux: Для Linux инсталляции, возможно, потребуется подключить программные файлы с компилятором *gcc*, чтобы они соответствовали текущим библиотекам Linux. Tsunami подсоединена к компилятору *gcc v3.2*. Вы можете проверить версию, набрав в командной строке `gcc -v`. Если у вас отличная от 3.2 версия, вам необходимо сделать следующее:

- a) Стереть *rays_linux* файл из директории `/apps/tsunami/tsunami_5.2.13/rays_5.2.13/linux`

```
>rm /apps/tsunami/tsunami_5.2.13/rays_5.2.13/linux/rays_linux
```

- b) `>cd /apps/tsunami/tsunami_5.2.13/rays_5.2.13/linux/rays_objs`

- c) `>make -f makefile_rays_linux`

Эта команда соединит программные файлы с библиотеками и поместит их в папку `/apps/tsunami/tsunami_5.2.13/rays_5.2.13/linux`.

- d) Повторите для PSDM and PSTM при необходимости.

Для инсталляции на Itanium: Intel v8.0 компилятор необходим для того, чтобы системные библиотеки были доступны программе. Мы рекомендуем поместить

компилятор в директорию, доступную для всех вычислительных узлов, так чтобы все они могли иметь доступ к библиотекам. Также компилятор Intel v8.0 может быть установлен на каждый узел. После того, как вы проверили наличие компилятора, добавьте в ваш “.cshrc” или “.bashrc” файл следующие строки:

Для *.bashrc*: `export PATH="/opt/intel_cc_80/bin:$PATH" source /opt/intel_cc_80/iccvars.sh`

Для *.cshrc*: допишите `/opt/intel_cc_80/bin` к списку переменных *PATH*.

Установка графического интерфейса GUI

1. Отдельной инсталляции для GUI не требуется. Для того чтобы запустить программу GUI (`/apps/tsunami/tsunami_5.2.13/tsunami.jar`), необходима *Java 2 Platform, Standard Edition (J2SE) Version 1.4.2* или выше. Чтобы определить версию java на вашей системе наберите:

a) `>java -version`

Покажет версию java в вашей системе.

b) `>which java`

Покажет путь и директорию, где инсталлирована java.

Если у вас не инсталлирована java, или у вас имеется более ранняя версия, пожалуйста, обратитесь к вашему системному администратору. Он должен установить необходимую версию и прописать домашнюю директорию *JAVA_HOME* в *.cshrc* или *.bashrc* файл. Ниже, в дополнении С, приведена инструкция по установке Java .

2. Отредактируйте “.cshrc” или “.bashrc” файлы, включив следующие переменные:

a) Добавьте переменную *JAVA_HOME*, если таковой не существует.

Для *.bashrc*: `export JAVA_HOME="java_directory"` Где *java_directory* – директория, куда инсталлирована java.

Для *.cshrc*: `setenv JAVA_HOME java_directory` где *java_directory* – директория, куда инсталлирована java.

b) Добавьте *JAVA_HOME/bin* в переменные *PATH*.

Для *.bashrc*: *PATH* находится в файле *.bash_profile*. Поместите *JAVA_HOME/bin* в конце существующего списка переменных *PATH*.

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

Для *.cshrc*: *PATH* находится в файле *.login* Поместите

JAVA_HOME/bin в конец существующего списка при *set path=*.
Например: *set path=(/bin /usr/bin /sbin /usr/etc /usr/local/bin /usr/java2/bin)* где */usr/java2* является домашней директорией *JAVA_HOME*

- c) Добавьте переменную *\$TSUNAMI* которая указывает на домашнюю директорию *tsunami*.

Для *.bashrc*: *export TSUNAMI="/apps/tsunami"*

Для *.cshrc*: *setenv TSUNAMI /apps/tsunami*

- d) Создайте *alias*, указывающий на выполняемый файл:

Для *.bashrc*: *alias tsunami="/apps/tsunami/tsunami_5.2.13/tsunami.sh"*

Для *.cshrc*: *alias tsunami /apps/tsunami/tsunami_5.2.13/tsunami.sh*

Плавающая лицензия

Лицензия контролирует количество работающих узлов. До версии 3.1.7 пользователь мог запускать программу на всех или любых, выбранных им и лицензированных узлах. При использовании плавающей лицензии, пользователь может перечислить в файле все имеющиеся у него узлы, но работать будет только то количество, на которое приобретена лицензия. Начиная с версии 4.15.9, количество процессоров также учитывается, для случаев, где узел имеет больше чем 2 процессора.

Программа проверит лицензию при запуске задания на счет, а также по его окончании или остановке с последующим перезапуском. Любое количество заданий может считаться на узлах, но только на том количестве узлов, которое лицензировано. Например, из 50 узлов, лицензированы только 20. При этом пользователь может использовать любые 20 из 50 одновременно.

При использовании плавающей лицензии для процессоров, во время счета только одного задания на узле, лицензия проверит количество процессоров, указанных в *node.db* файле. Если на узле считается несколько заданий, то все лицензированные процессоры будут проверены и в лог файле появится предупреждающее сообщение.

В случае если не все процессоры на узле лицензированы, то пользователь не сможет запустить несколько параллельных заданий. Программа проверяет количество процессоров на узле, сверяет их с количеством, указанным в лицензии и с *node db* файлом.

Проверить статус лицензии можно при помощи утилиты: *check_license_file*, вызываемой из меню *Utilities*. В окошке появится сообщение об именах и количестве узлов, их *mac* адресах, дате истечения срока действия лицензии и количестве лицензированных процессоров

```
>check_license_file <license_file_name>
```

В случае если проверка лицензии по какой-то причин не сработала, то перезапустить лицензию можно при помощи утилиты *clear_nodes*. Эта программа убивает все процессы на нодах, перечисленных в *node.db* файле, а также переустанавливает лицензию на эти ноды. Утилита *clear_nodes* была изменена, начиная с версии 3.1.7, и больше не использует user id для определения и выбора процессов, идущих на нодах. Теперь, при запуске *clear_nodes*, все процессы идущие на нодах убиваются, при этом создается файл *clear_nodes.log* в домашней директории *\$TSUNAMI*.

```
>clear_nodes <node db file>
```

Файлы лицензий помещаются в директорию */apps/tsunami* .

Конфигурация кластера

Для работы Tsunami необходимо создать файл с именами нодов, *node.db*. Это ASCII файл следующего формата:

```
Node_ name number_of_processors memory_in_Mbytes speed_factor operating_system
master04          1           1000           1.0           linux
node01            2           1000           1.0           linux
node02            2           1000           1.0           linux
```

Имена нодов указаны в */etc/hosts* файле.

Показатель скорости, *speed_factor*, отражает относительную скорость процессора по отношению к другим нодам. Показатель скорости и количество процессоров используются для распределения задания по нодам, позволяя одновременно подключать на один кластер ноды с различным количеством процессоров и их разной скоростью.

Обычно, быстрым процессорам задается значение 1.0, а более медленным – значение меньше 1.0. К примеру, 3000 MHz процессор имеет значение 1.0, при этом 1500 MHz процессор - значение 0.5.

По завершении задания, в конце лог файла приводится статистика для каждого нода. Используйте значения “*Millions of shift and sums per second kernel time*” как показатель скорости счета миграции на нодах.

Название операционной системы, указывает программе, какой из выполняемых файлов использовать для расчета задания: *linux, sgi, solaris, opteron* или *itanium (linux)*.

Главный нод (master node) обязательно прописывается в первой строке *node.db* файла. Это тот нод, с которого запускается программа, и только тот, который имеет доступ к входным данным и файлам для записи выходных данных. Он также может быть использован и как вычислительный нод. В этом случае, его необходимо указать еще раз, второй строкой.

Например, *node db* такой конфигурации:

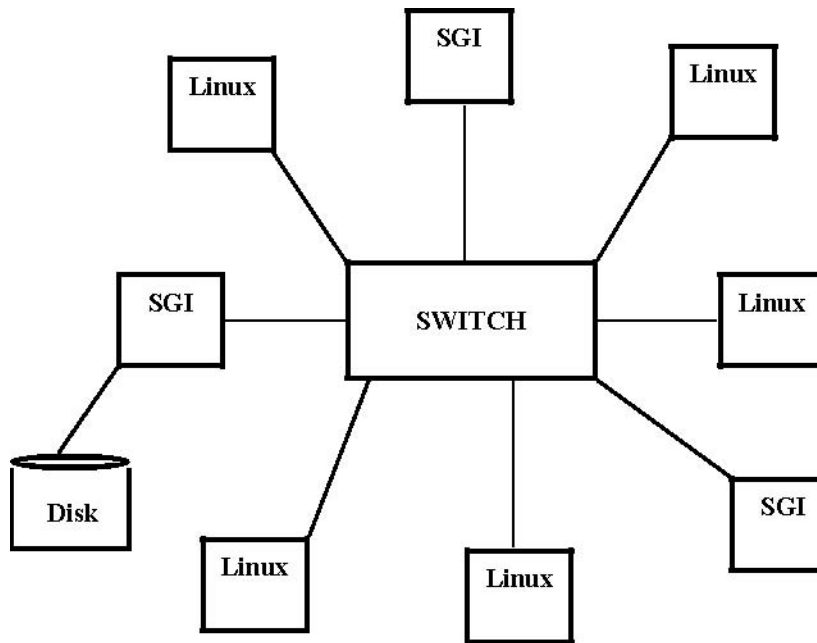
```

server1 2      512    1.0    linux
linux1  2      512    1.0    linux
linux2  2      512    1.0    linux
sgil    4      512    .75    sgi
itanium18 1000  .65    Itanium

```

Для того чтобы программа работала, должна быть доступна опция *rsh* с главного нода на все остальные вычислительные ноды. Если у вас возникнут вопросы, обратитесь к вашему системному администратору или к ESS. Директория */tmp* должна быть доступна на всех нодах. В неё записываются выполняемые файлы, при запуске задания, которые затем удаляются после его завершения.

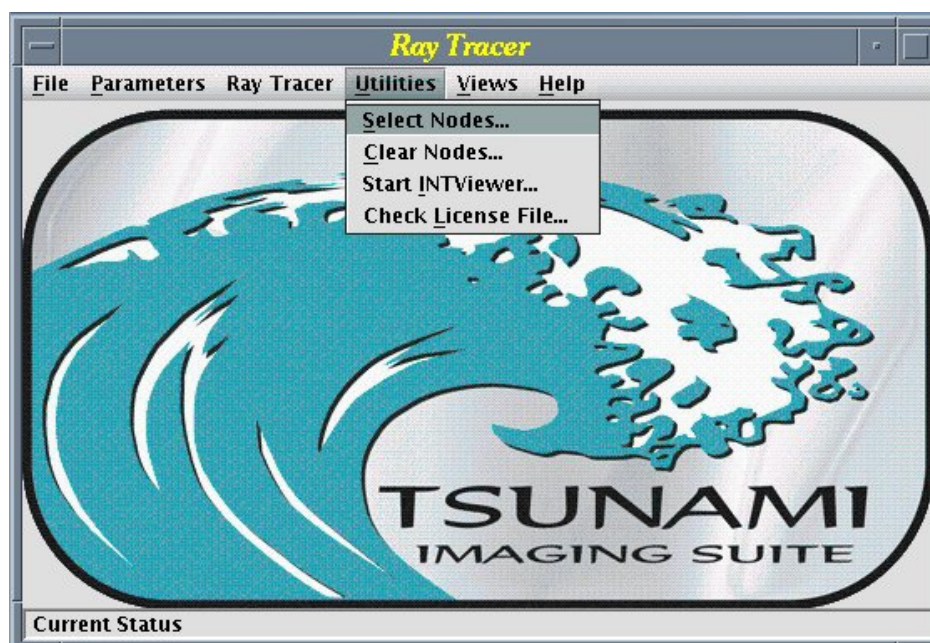
Пример конфигурации кластера



Утилита нодов

После того, как вы создали файл *node.db*, вы можете использовать опцию **Select Node** из главного меню GUI. *Смотри: **Начало работы с графическим интерфейсом (GUI)**.

1. В заголовке **Utility** —> выберите **Select Nodes**.



Нажав на пункт меню, пользователь открывает окошко с дополнительным меню с возможностью выбора нодов, а также создания новых.

Для того чтобы открыть файл со списком нодов, или создать новый файл, программа должна найти master node файл. Он находится в домашней директории $\$TSUNAMI$, например `"/apps/tsunami"`. Если он не найден, то пользователю предлагается указать директорию, в которой он находится.

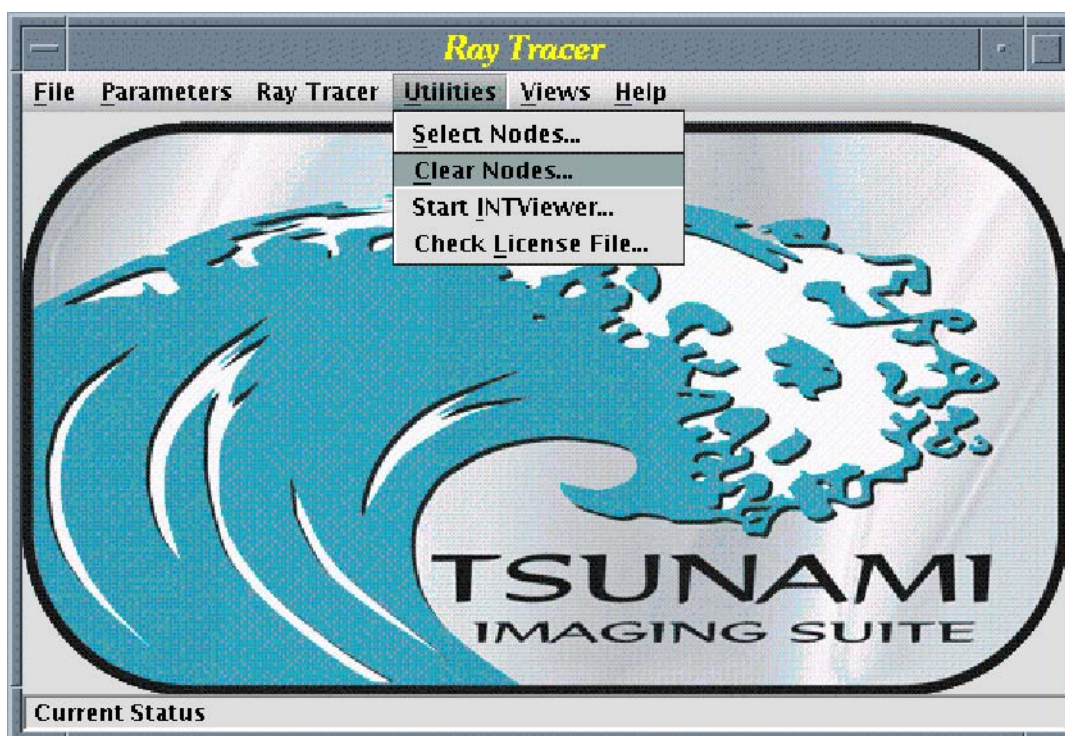
Если master node файл найден и указана директория текущего проекта (на закладке **Processing**), откроется диалоговое окошко с заданными нодами. Пользователь также может выбрать нужные ему ноды и сохранить их, как новый файл для данного задания.

Если пользователь сохраняет данный файл со списком нодов, то этот файл автоматически сохраняется в файл параметров, как параметр (**nddb=**). (Находится на закладке **Processing**).

Для более детальной информации по созданию файла со списком нодов, а также распределению задания по нодам смотрите раздел **“Анализ производительности и улучшенные возможности”**.



2. Если задание было остановлено или прекратилось считаться по причине проблем с “железом“, программа Tsunami сразу же удаляет все процессы на вычислительных нодах. В некоторых случаях необходимо запустить утилиту `clear_nodes`. Эта программа убивает все процессы на нодах, перечисленных в файле `node.db`, кроме мастер нода, а также переустанавливает лицензию на эти ноды. Лог файл будет помещен в домашнюю директорию `$TSUNAMI`.



Clear Nodes с командной строки:

В командной строке наберите:

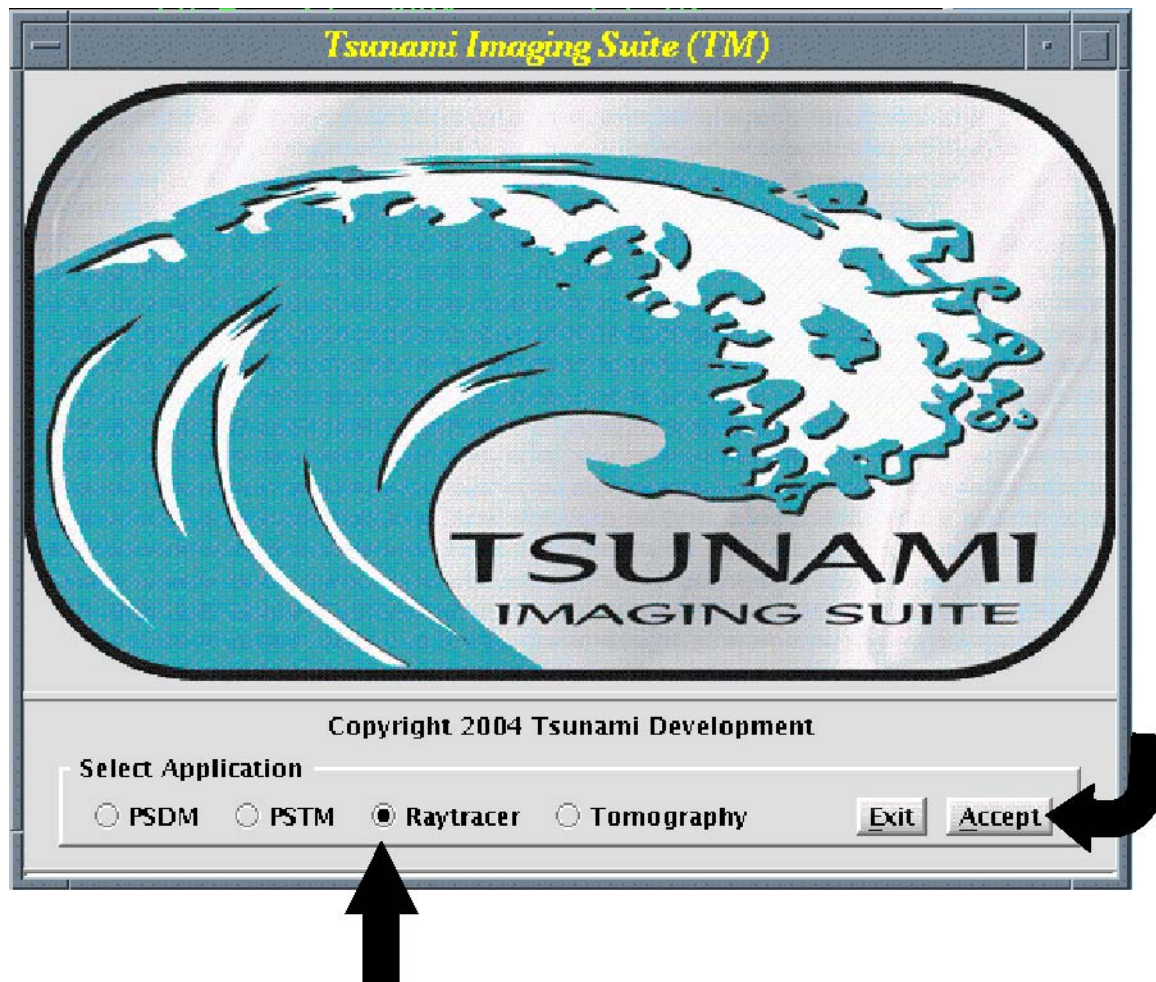
```
> clear_nodes < node.db file >
```

Эта программа убивает все процессы на нодах, перечисленных в файле *node.db*, кроме мастер нода, а также переустанавливает лицензию на эти ноды. Процессы, идущие на мастер ноде удалятся, как результат удаления процессов на вычислительных нодах. Для дополнительной информации см. раздел **Плавающая лицензия**.

Начало работы

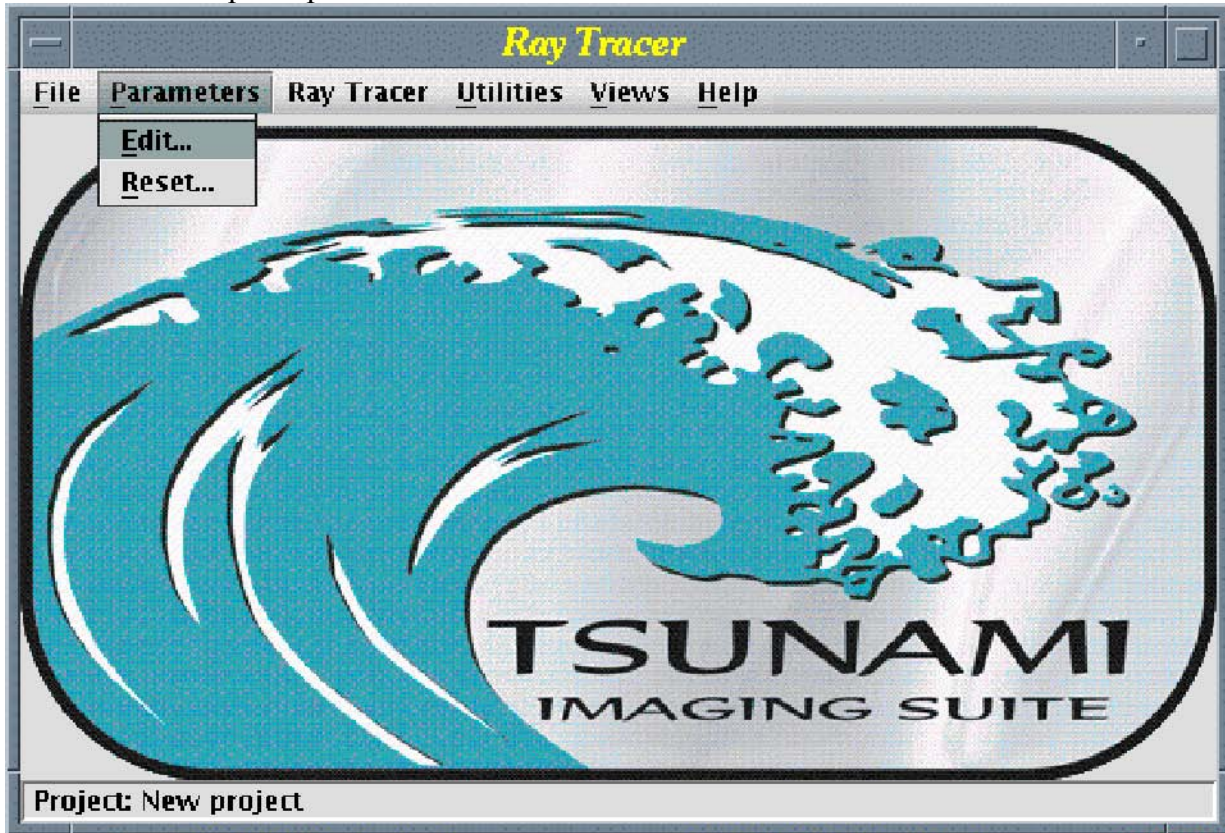
Используйте *alias* , созданный в п.2 раздела инсталляция GUI для запуска программы: Пример
>tsunami

- 1 Выберите кнопку **Raytracer** из окошка **Startup** и нажмите **Accept**.
- 2 Откройте новый файл, выберите **File** → **New** для создания параметров нового задания.



Создание файла времен пробега

- 1 В окошке Ray Tracer выберите **Parameter** → **Edit**.
- 2 Откройте закладку **Files** в окошке выбора параметров. Все требуемые для заполнения поля параметров выделены жёлтым цветом.



Страница *File Settings*:

Application Directory (hmdr): Указывает путь к выбранной вами версии Tsunami RAYTRACER . Например: Для версии 5.2.13 она должна показывать на `~/tsunami_5.2.13/rays_5.2.13`

Project Directory (prjdr): Директория задания, в которую будут помещены лог файлы, а также временные файлы, создаваемые во время расчета задания. В этой же директории создается временный блок файл скоростей.

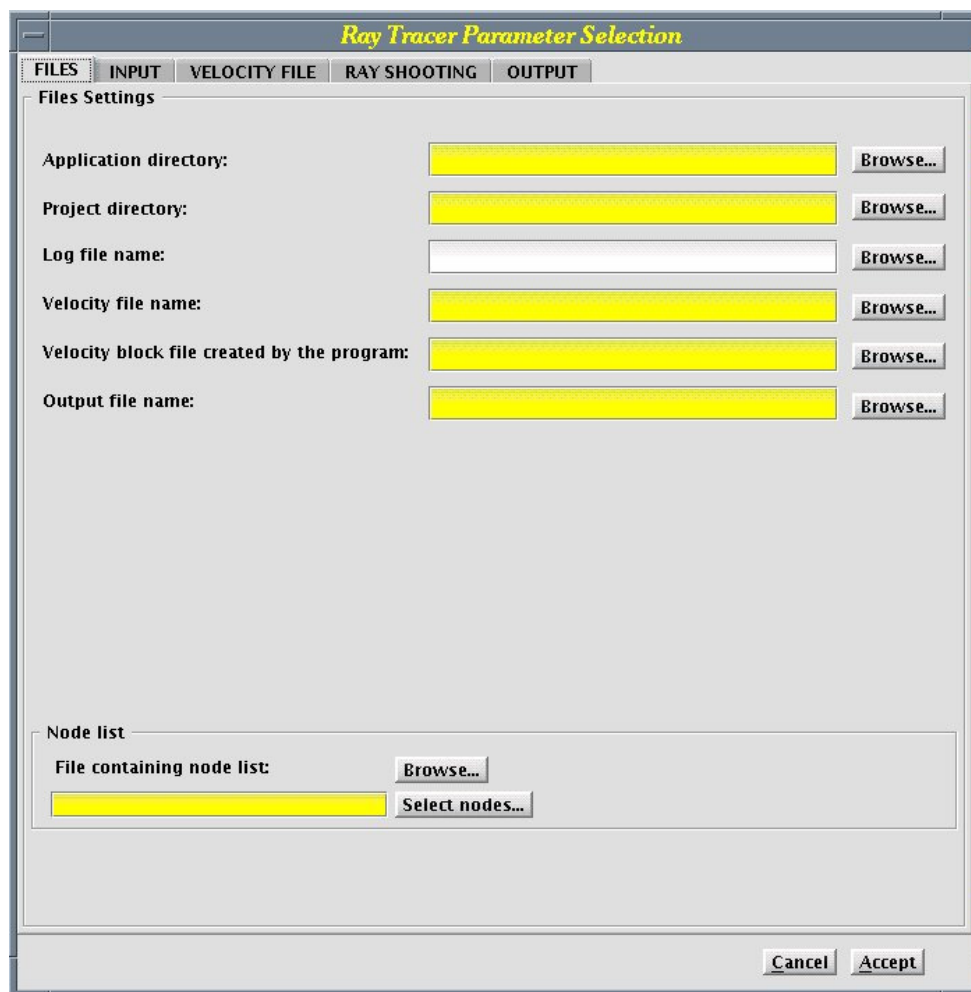
Log file name (logfile): Если вы не указываете имя лог файла вручную, то по умолчанию он создается с именем `project_dir/jobname.log`. Если вы указываете имя лог файла вручную, то рекомендуется указывать весь путь, включая директорию задания.

Velocity File Name (velf): Указываете путь к кубу интервальных скоростей в формате SEG-Y. Файл должен иметь регулярную дискретизацию по инлайнам,

кросслайнам и глубине.

Velocity Block File Created by the program (blkf): Задаете имя и указываете путь к временному блоку файла скоростей, создаваемому программой. Если поле оставлено пустым, то по умолчанию файл создаётся в директории проекта и называется *vels.blk*. Дополнительную информацию по данному параметру смотрите в разделе **Анализ производительности и улучшенные возможности: Временный блок-файл скоростей**.

Output File Name (outf): Имя файла записи времен пробега.



Список нодов:

File containing node list (nddb): Список нодов, используемых для расчета задания. Дополнительную информацию по созданию списка см. в разделе **Конфигурация кластера**.

Browse: Открывает диалоговое окошко для выбора файла со списком нодов.

Select Nodes: Позволяет редактировать выбранный вами список нодов. В этом окошке можно добавить новые ноды, удалить имеющиеся и сделать доступными или недоступными ноды для конкретного задания.

Файл параметров для запуска программы с командной строки:

Страничка **Files** в командной строке представляет собой следующий список параметров:

hmdr= домашняя директория программы
prjdr= директория задания (проекта)
logfile= имя лог файла
velf= куб скоростей в формате SEG Y
blkf= блок файл скоростей, созданный программой
outf= имя файла записи времен пробега
nddb= файл со списком нодов

Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Информация файла параметров**.

Страница Input

3. Выберите страничку **Input**. Введите необходимую информацию. Все требуемые для заполнения поля параметров выделены жёлтым цветом.

Dataset Dimensions (twod): Контролирует формат данных - 2D или 3D задание. По умолчанию 3D. При выборе 2D, все источники будут располагаться вдоль одного профиля, и, на запись, времена пробега лучей будут расположены по одному инлайну для каждого источника.

RayTracer работает, запуская лучи на всей поверхности сейсмической площади, покрываемой источниками и приемниками. Для данного PSDM задания, площадь источников лучей должна покрывать площадь источников /приемников исходных данных. У вас может быть несколько выходных файлов времен пробега для одного задания. Шаг между источниками запуска лучей не обязательно должен быть одинаковым во всех выходных файлах.

Shot Locations:

First Inline to Shoot (ilfrst): Первый инлайн для запуска лучей

Last Inline to Shoot (illst): Последний инлайн для запуска лучей

Inline Increment Between Shots (ilinc): Шаг по инлайну между источниками лучей (по умолчанию равен 10)

First Xline to Shoot (xlfrst): Первый кросслайн для запуска лучей

Last Xline to Shoot (xllst): Последний кросслайн для запуска лучей

Xline Increment Between Shots (xlinec) : Шаг по кросслайну между источниками лучей (по умолчанию равен 10)

The image shows a software dialog box titled "Ray Tracer Parameter Selection". It has several tabs: "FILES", "INPUT", "VELOCITY FILE", "RAY SHOOTING", and "OUTPUT". The "INPUT" tab is selected. Under "Input settings", there are three sections:

- Dataset dimensions:** Two radio buttons, "2D" and "3D". The "3D" button is selected.
- Shot locations - covers the area of input source and receivers:** A grid of six input fields. The first two columns are "First inline to shoot" and "Last inline to shoot", both with yellow background. The last two columns are "First xline to shoot" and "Last xline to shoot", also with yellow background. The bottom two fields are "Inline increment between shots:" and "Xline increment between shots:", both containing the value "10".
- Anisotropy:** Two input fields: "ETA File:" and "Epsilon/Delta Ratio:". To the right of the "ETA File:" field is a button labeled "Browse for ETA File".

At the bottom right of the dialog box are two buttons: "Cancel" and "Accept".

Anisotropy:

ETA File (etaf): Имя segy файла со значениями эта (η). Значения эта должны быть записаны с точно таким же шагом по инлайнам, кросслайнам и глубине, как и скоростная модель. Файл должен быть в формате SEG-Y и с теми же параметрами в заголовках, что и скоростная модель. Вся эта информация проверяется программой RayTracer при запуске.

Epsilon/Delta Ratio (epdelr): коэффициент эpsilon(ϵ)/дельта(δ).

Оба файла, и файл значений эта и файл коэффициента эpsilon/дельта требуются для расчета анизотропии.

NB: Функция анизотропии не работает для расчета времен пробега методом Эйконала, программа выдаст сообщение об ошибке. Аналогично для запуска psdm.

NB: Если вы используете модуль Smooth Velocity, то обратите внимание, что использовать для анизотропии нужно блок-файл, созданный модулем RayTracer, а не Smooth Velocity

Файл параметров для запуска с командной строки:

Страничка **Input** в командной строке представляет собой следующий список параметров:

twod= 2D(1) or 3D(0). По умолчанию 0 или 3D.
xlfrst= Первый кросслайн для запуска лучей
xllst= Последний кросслайн для запуска лучей
xline= Шаг по кросслайнам между источниками лучей (по умолчанию равен 10)
ilfrst= Первый инлайн для запуска лучей
illst= Последний инлайн для запуска лучей
ilinc= Шаг по инлайнам между источниками лучей (по умолчанию равен 10)
etaf= файл со значениями «эта» в формате segy
epdelr= коэффициент эпсилон/дельта

Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Информация файла параметров**.

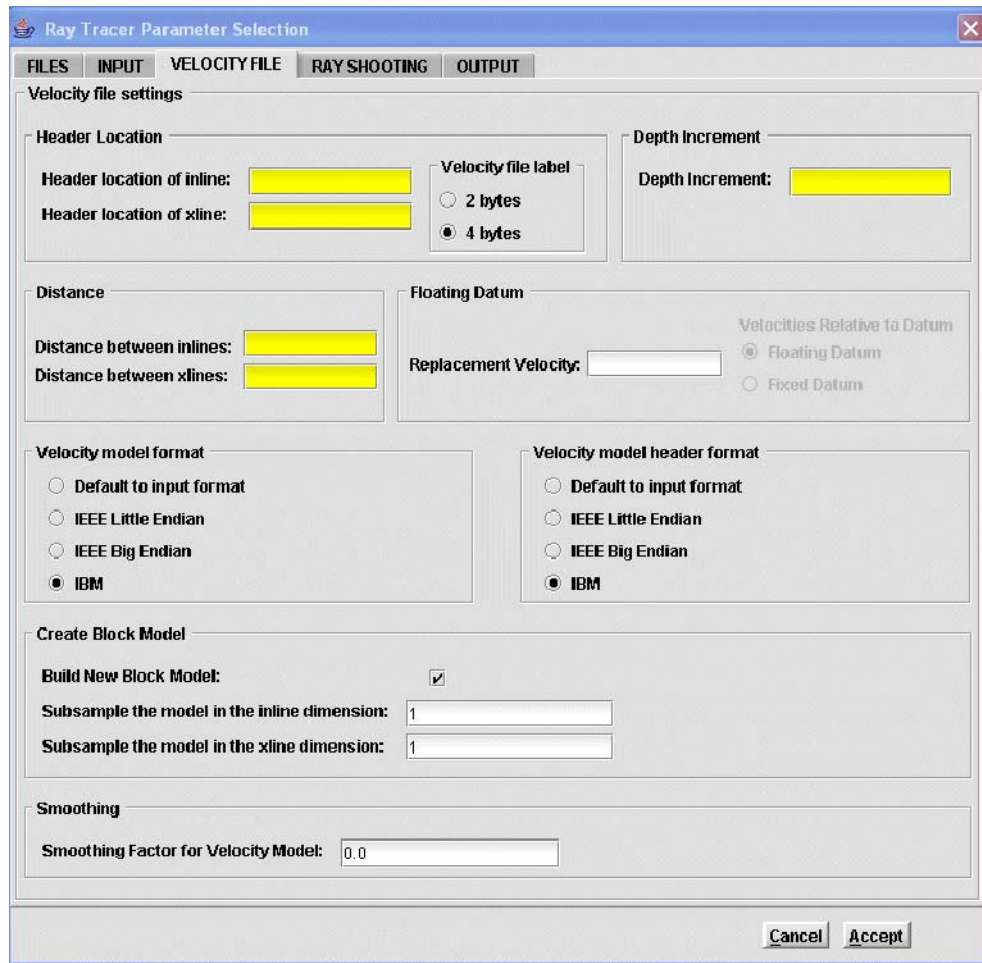
Страница Velocity File

4. Выберите страничку **Velocity File**. Все требуемые для заполнения поля параметров выделены жёлтым цветом.

Куб интервальных скоростей должен быть в формате SEG Y, и скорости должны быть отсортированы по инлайн, с регулярным шагом по глубине, кросслайн и инлайн. Многие обрабатывающие системы не делают сортировку по инлайн, поэтому убедитесь, что ваш куб имеет правильный формат. Также важно, чтобы скоростная модель имела форму прямоугольника, т.е. все инлайны имеют одинаковое количество трасс.

Формат может быть как IEEE, так и IBM. IEEE формат принимается как big endian, так и little endian.

Модель не обязательно должна содержать все количество инлайнов и кросслайнов. Если это будет необходимо, модель автоматически экстраполируется, беря последнее существующее значение.



Параметры Header Locations:

Header loc of Xline (vxlb): Номер байта с номером кросслайна в заголовке скоростной модели. (По умолчанию значения нет)

Header loc of inline (vilb): Номер байта с номером инлайна в заголовке скоростной модели. (По умолчанию значения нет)

Depth Increment (vdz): Шаг скоростной модели по глубине.

Velocity File Label (vxlityp): Формат значений в заголовках (0) 4 byte integer или (1) 2 byte integer, по умолчанию 0

Поле Distance:

Distance between xlines (xldist): Расстояние между кросслайнами (По умолчанию значения нет)

Distance between inlines (ildist): Расстояние между инлайнами (По умолчанию значения нет)

Поле Floating Datum:

RayTracer позволяет пользователю изображать данные от топографии или плавающего уровня. Дополнительная информация приводится в разделе **Анализ производительности...: Миграция данных от плавающего уровня или топографии.**

Replacement Velocity (vdtm): Скорость замещения, используемая между фиксированным и плавающим уровнями.

Floating or Fixed Datum (frdtm): Скорости берутся от фиксированного или плавающего уровня. (0) фиксированный уровень, (1) плавающий уровень. По умолчанию 1.

Velocity Model Format (dataf): Показывает формат скоростной модели. Может быть выбран как исходный формат, так и IEEE_LE, IEEE_BE или IBM. (По умолчанию - IBM).

Velocity Model Header Format (hdrfmt): Показывает формат заголовка скоростной модели.. Может быть выбран как исходный формат, так и IEEE_LE, IEEE_BE или IBM. (По умолчанию - IBM).

Поле Create Block Model:

Дополнительная информация приводится в разделе **Анализ производительности...: Создание временного блок-файла скоростей.**

Build New Block Model (newb): Создание нового блок-файла скоростей. По умолчанию - создать.

Subsample the model in the inline dimensions (subil): шаг скоростной модели по инлайнам.

Subsample the model in the xline dimensions (subxl): шаг скоростной модели по кросслайнам.

Поле Smoothing:

Smoothing Factor for Velocity Model (smfctr): По умолчанию программа RayTracer не сглаживает модель. Дополнительная информация приводится в разделе **Анализ производительности...: Создание временного блок-файла скоростей.**

Файл параметров для запуска с командной строки:

Страничка **Velocity File** в командной строке представляет собой следующий список параметров:

vxlb= номер байта кросслайна в заголовке (По умолчанию значения нет)
vilb= номер байта инлайнф в заголовке (По умолчанию значения нет)
vdz= шаг скоростной модели по глубине
xldist= Расстояние между кросслайнами (По умолчанию значения нет)
ildist= Расстояние между инлайнами (По умолчанию значения нет)
dataf= Формат скоростной модели. Может быть выбран как исходный формат, так и IEEE_LE, IEEE_BE или IBM. (По умолчанию - IBM).
hdrfmt= Показывает формат заголовка скоростной модели.. Может быть выбран как исходный формат, так и IEEE_LE, IEEE_BE или IBM. (По умолчанию - IBM).
vxlityp= Формат значений в заголовках, по умолчанию 0 или 4 byte integer.
newb= создание нового блок-файла.
subil= шаг скоростной модели по инлайнам. .
subxl= шаг скоростной модели по кросслайнам.
smfctr= фактор сглаживания скоростной модели.

Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Информация файла параметров**.

Страница Ray Shooting

5. Выберите страничку **Ray Shooting**. Все требуемые для заполнения поля параметров выделены жёлтым цветом.

Поле **Travel Time Algorithm**. Выберите алгоритм расчета времен пробега лучей

Параметры для алгоритма Ray Tracing:

Точки запуска лучей находятся на равных промежутках в двух направлениях: в радиальной плоскости и азимутальном угле, и в вертикальной плоскости и под углом ϕ . Исходное количество лучей контролируется параметром:

Angle Increments to Shoot: Инкремент угла для запуска лучей

Angle increment in the vertical plane (phinc): Инкремент угла в вертикальной плоскости (по умолчанию 2.5 градуса)

Angle increment in the horizontal plane (azinc): Инкремент угла в азимутальной (горизонтальной) плоскости (по умолчанию 2.5 градуса)

Лучи трассируются группами по три луча или так называемыми пучками. Когда пучок проходит через модель, времена пробега 3-х лучей, составляющих пучок, сравниваются друг с другом. Если времена пробега 3-х лучей отличаются друг от друга более чем на заданную величину, пучок разделяется; создаются новый луч и новый пучок. Пользователь может контролировать, как часто пучок делится, используя коэффициент точности, и может контролировать максимальное число живых лучей в пучке.

Accuracy factor to control creation of new rays (accfac): коэффициент точности, меньшее число повышает точность и число делений пучка (по умолчанию 4.0)

Maximum rays per beam (maxray): максимальное количество живых лучей в пучке.

(по умолчанию 75)

DT of seismic data (dtacc): шаг сейсмических данных по глубине в миллисекундах (по умолчанию 4)

Лог файл показывает статистику по точности расчета времен пробега при выборе параметров. Когда каждое время пробега луча сохраняется, разница между каждым из 3-х лучей заносится в таблицу. Кроме того, в таблицу заносится расстояние между тремя лучами. В лог-файле показаны: среднее время разницы, 80 % разницы времен, среднее расстояние между лучами, 80 % расстояния между лучами. Вы можете увеличить точность, изменяя эти факторы. Конечно, это влияет на время расчета задания.

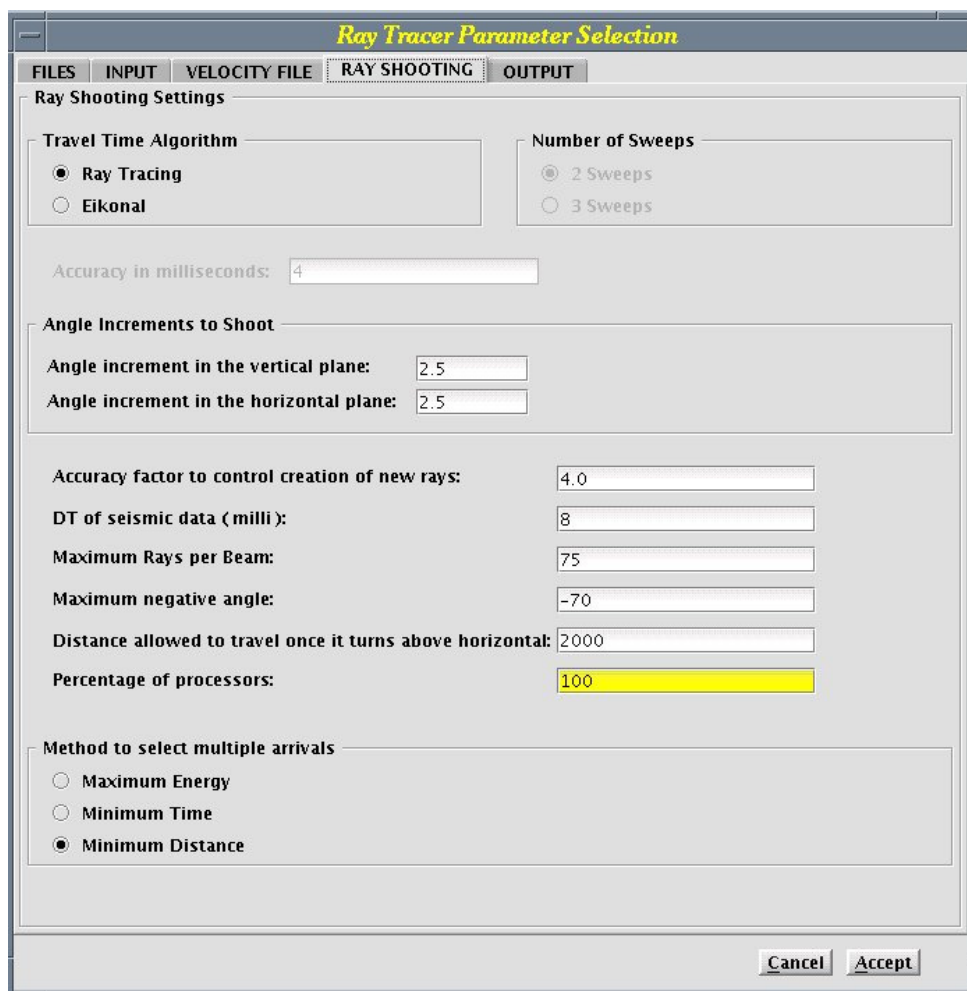
Следующие два поля относятся к поворотным лучам

Raytracer предполагает существование поворотных лучей. Это лучи, преломляющиеся таким образом, что, в конце концов, трассируются вверх от горизонтали. Эти лучи необходимы при изображении выступов (карнизов) солей. Однако время расчета задания может заметно увеличиться, если параметры этих лучей не контролируются. Поэтому, пользователь может ограничить максимальный угол, на который лучи могут отклониться вверх от горизонтали, и расстояние, на которое лучи могут распространяться от места перехода от нисходящего к восходящему. Считается, что если параметр угла равен 0, то восходящие лучи не трассируются, а если -90 – то трассируются все восходящие лучи.

Maximum Negative Angle (mnegang): максимальный отрицательный угол, от 0 до -90 (по умолчанию -60)

Distance allowed to travel once it turns above horizontal (mxnegdst): расстояние по горизонтали, на которое могут распространяться поворотные лучи: от 0 to 10000 (по умолчанию 2000)

Method to select multiple arrivals (mode): метод выбора лучей: по максимальной энергии, минимальному расстоянию или минимальному времени.



Файл параметров для запуска с командной строки:

Страничка **Ray Shooting** в командной строке представляет собой следующий список параметров:

- eiknl**= для метода Ray Tracing должен быть = 0
- phinc**= инкремент угла в вертикальной плоскости
- azinc**= Инкремент угла в азимутальной (горизонтальной) плоскости
- accfac**= коэффициент точности, меньшее значение повышает точность и число делений пучка (по умолчанию 4.0)
- maxray**= максимальное количество живых лучей в пучке (по умолчанию 75)
- dtacc**= шаг сейсмических данных по глубине миллисекундах (по умолчанию 4)
- mnegang**= максимальный отрицательный угол, от 0 до -90 (по умолчанию -60)
- mxnegdst**= расстояние по горизонтали, на которое могут распространяться поворотные лучи: от 0 до 10000 (по умолчанию 2000)
- mode**= метод выбора лучей: по максимальной энергии, минимальному расстоянию или минимальному времени
- vdtm**= скорость замещения, используемая между фиксированным и плавающим

уровнями.
frdtm= фиксированный или плавающий уровень
rqpcnt= требуемый процент процессоров

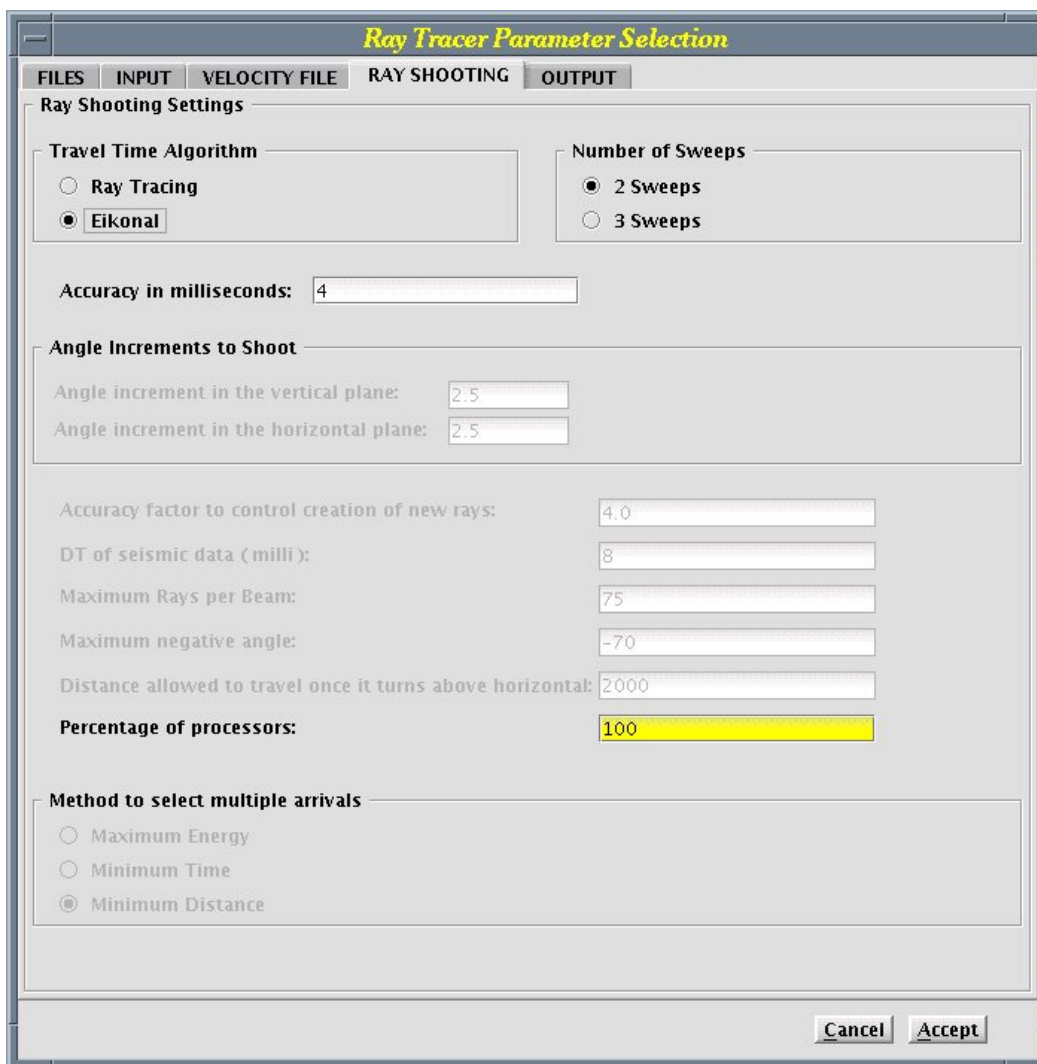
Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Данные файла параметров**.

Параметры для алгоритма Eikonal:

Метод Эйконала использует сетку (грид) переменной плотности, который увеличивает точность расчета в более сложных областях.

Accuracy in milliseconds (tres): Точность в миллисекундах: этот параметр устанавливает порог для деления грида на части. Этот параметр задается в миллисекундах (от 4-8) . Уменьшение этого параметра увеличивает время счета.

Number of Sweeps (nsweep): Количество проходов трассировки: для повышения точности и проверки необходимости деления грида используется несколько проходов через грид. Различия между первым и вторым проходом практически всегда существенны, в то время как улучшения после четвертого прохода незначительны, поэтому величина данного параметра может быть 2 или 3. При 3-х проходах время счета увеличивается, рекомендовано только при наличии очень сложной скоростной модели.



Файл параметров для запуска с командной строки:

Страничка **Ray Shooting** для метода **Eikonal** в командной строке представляет собой следующий список параметров:

eiknl= равен 1 для метода Eikonal
rqpcnt= требуемый процент процессоров
tres= точность в миллисекундах
nsweep= количество проходов трассировки

Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Информация файла параметров**.

Страница **Output**

6. Выберите страничку **Output**. Все требуемые для заполнения поля параметров выделены жёлтым цветом.

Поле Subsurface Locations

Точки фокусировки лучей должны охватить всю площадь миграции. Если шаг по источникам запуска лучей – 10 инлайнов, то шаг по точкам фокусировки (subsurface locations) должен быть 10, 5, 2 или 1. Это те точки, на которых рассчитанные времена пробега записываются. Шаг выходного файла (т.е. положений точек фокусировки) должен быть кратным шагу по источникам лучей.

Если вы используете несколько файлов с временами пробега, то все они должны иметь одинаковый шаг точек фокусировки.

Шаги файла записи времен пробега не обязательно задавать такими же, как у миграции. Если миграция считается на каждый инлайн и кросслайн, то выходной шаг для времен пробега рекомендуется задавать 4-5. Таким образом, вы улучшите производительность, сократив время расчета, а также объем выходного файла.

Если вы используете миграцию для анализа скоростей и ваши выходные ОГТ расположены далеко друг о друга, то точки расчета времен пробега должны совпадать с выходными ОГТ на миграцию. В этом случае рекомендуется, чтобы первая, последняя локация и инкремент совпадали для эффективности и минимизации интерполяции.

First inline to output (ilfirstout): Первый инлайн на выход

Last inline to output (illstout): Последний инлайн на выход

Inline increment to output (ilincout): Шаг по инлайнам на выход (по умолчанию 5)

First xline to output (xlfirstout): Первый кросслайн на выход

Last xline to output (xllstout): Последний кросслайн на выход

Xline increment to output (xlincout): Шаг по кросслайнам на выход (по умолчанию 5)

Апертура миграции и максимальное удаление (офсет)

Программа RayTracer определяет размер куба времен пробега, основываясь на апертуре миграции и максимальном офсете. Размер файла времен пробега никогда не должен превышать по диаметру две апертуры плюс максимальный офсет. Апертура определяется как максимальное расстояние от исходной трассы до ее мигрированного изображения. Основываясь на этих данных, объем файла времен пробега вычисляется автоматически наряду с количеством точек фокусировки, в которых записываются времена пробега. Эти данные затем заносятся в лог-файл и в заголовок файла времен пробега.

Migration aperture (aper): Апертура миграции

Maximum offset (mxoff): Максимальный офсет данных

Шаг по глубине - Depth Sampling Increment

Вы можете задать постоянный шаг по глубине, или позволить программе RayTracer рассчитать переменный шаг, основываясь на ряде пар «частота-глубина». Пары частота-глубина определяют ожидаемую выходную частоту данных, а затем RayTracer рассчитывает необходимый шаг по глубине для получения данной частоты. **Как правило, этот метод позволяет миграции идти быстрее.** В любом случае, у вас есть возможность задать любой постоянный шаг по глубине для вашего задания.

Применяя постоянный шаг по глубине, вы переписываете заданные до этого пары частота-глубина. Если пользователь не задает пары или постоянный шаг по глубине, то шаг

рассчитывается, основываясь на частоте Найквиста.

Fixed depth sampling increment (dinc): шаг по глубине. По умолчанию принимается значение 30 м (в метрической системе единиц) или 90 футов (в английской системе).

Depth frequency pairs (frqpr): пары «глубина/частота».

Maximum depth (dmax): значение максимальной глубины для записи

Smoothing Factor (smth): сглаживание выходных времен пробега

Тип выходного формата - Output Format Type (fmto):

Начиная с версии 3.1.9, пользователь может дополнительно вывести файл времен пробега в формате SEG-Y. Эта опция позволяет контролировать и проверять качество времен пробега, как тест, перед тем как запускать расчет на весь куб. Этот SEG-Y файл не будет считываться миграцией, он создается только для вывода на дисплей и проверки. Значения инлайн, кросслайн и порядковый номер источника лучей записываются в формате 4-byte в байтах 181,185, и 17 соответственно.

Пользователь может установить запись на segy, задав параметр **fmto**= 2. По умолчанию параметр **fmto**= 1, что означает необходимый формат для последующей миграции.

The screenshot shows the 'Ray Tracer Parameter Selection' dialog box with the 'OUTPUT' tab selected. The 'Output Settings' section includes 'Subsurface Locations: Output area of the migration' with fields for 'First inline to output', 'Last inline to output', 'Inline increment to output' (set to 5), 'First xline to output', 'Last xline to output', and 'Xline increment to output' (set to 5). Below this are 'Migration aperture', 'Maximum depth' (set to -1), 'Maximum offset', and 'Smoothing factor' (set to 3). The 'Output Format Type' section has two radio buttons: 'Format for Depth Migration' (selected) and 'Output a QC SEG-Y file'. The 'Depth Sampling' section has two radio buttons: 'Fixed depth sampling increment' (selected) and 'Depth frequency pairs'. A text box next to the selected radio button shows 'Fixed depth sampling increment: 50' and a disabled 'Depth frequency pairs' button. At the bottom are 'Cancel' and 'Accept' buttons.

Файл параметров для запуска с командной строки:

Страничка **Output** в командной строке представляет собой следующий список параметров:

xlfirstout= Первый кросслайн на выход
xllstout= Последний кросслайн на выход
xlincout= Шаг по кросслайнам на выход (по умолчанию 5)
ilfirstout= Первый инлайн на выход
illstout= Последний инлайн на выход
ilincout= Шаг по инлайнам на выход (по умолчанию 5)
aper= Апертура миграции
mxoff= Максимальный офсет (удаление) сейсмических данных
dinc= шаг по глубине
frqpr= Пары глубина-частота, заключаются в одинарные кавычки (например: '1000 90 3000 70 5000 40')
smth= сглаживание выходных времен пробега
dmax= максимальная глубина выходных данных
fmo= формат записи

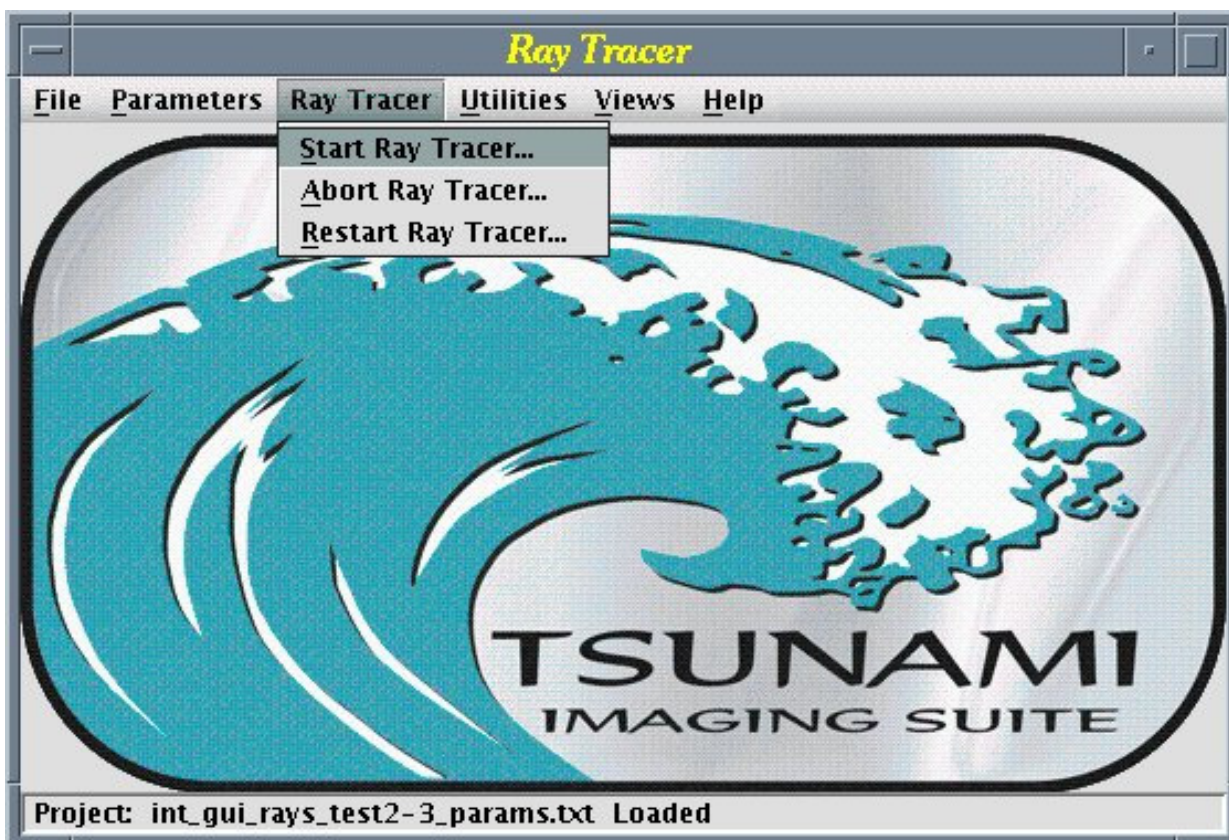
Дополнительная информация по созданию файла параметров с командной строки и полный список требуемых параметров приводится в **Приложении В: Информация файла параметров**.

Остановка и перезапуск задания

Остановить/Перезапустить задание из GUI

Ray Tracer – Запуск лучевого трассирования

Параметр **Start Ray Tracer** запускает задание с выбранным или созданным файлом параметров. Текущий файл параметров показан в нижней строке графического интерфейса. Как только задание начнет считаться, то появится сообщение : "*Starting Ray Tracer for file: (file name).*"



Параметр Rays Tracer - Abort

Как только пользователь задает этот параметр, задание немедленно прерывается, на нодах удаляются все процессы, и задание полностью останавливается.

Параметр Ray Tracer - Restart

Выберите 'Yes' для перезапуска задания.

Как только задание перезапустилось, появится сообщение "*Restarting Ray Tracer for file: (file name).*"

Обратите внимание: Если задание не перезапускается, то, возможно, оно не находит файл *.pids (с параметрами трассирования). Проверьте директорию проекта на наличие этого файла и, если его там нет, пересоздайте его, после чего перезапустите задание.

Остановка/Перезапуск задания из командной строки:

Для запуска задания наберите в командной строке следующее:

```
>rays_start <parameter file name> &
```

Задания, запущенные на счет с командной строки, перезапустятся автоматически, если они остановились по какой-либо причине.

Для полной остановки задания, наберите:

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

Задание немедленно прервется, на нодах удалятся все процессы и задание полностью остановится.

Для перезапуска остановленного ранее задания, наберите:

```
>rays_restart <parameter file name>
```

Анализ производительности и улучшенные возможности.

Временный блок файл скоростной модели

Программа создана с таким расчетом, что она может работать с файлом скоростной модели любого объема, даже больше чем память системы. Если ваша система является системой с распределённой памятью, то RayTracer будет распределять часть скоростной модели в памяти процессоров, помещая таким образом как можно больше частей модели в память. Затем, автоматически осуществляется контроль за остальным объемом. Для того, чтобы добиться наивысшей производительности, при запуске программы, скоростная модель считывается и переформатируется в блок файл скоростной модели. Поскольку все блоки данного файла являются блоками одного размера и объема, то программа округлит модель до границы последнего блока. Затем она поместит скорости в блоки, как целые значения по источникам. Как результат, созданный блок файл будет больше чем исходная модель в полтора раза, но из-за переформатирования приобретёт свойства скоростного доступа. Пользователь должен убедиться, что на диске достаточно свободного пространства для создания блок файла. Местонахождение и имя этого файла задаются параметром **blkf=**.

Точный объем блок файла скоростной модели зависит от геометрии исходного файла. Модель будет округлена к ближайшему инкременту 25 во всех измерениях. Несмотря на то, что файл занимает дополнительное пространство на диске, он позволяет программе считать скоростные модели, превышающие объем памяти системы, и при этом поддерживать эффективность и производительность.

По умолчанию, блок файл создается при каждом запуске задания. Иногда эта процедура занимает относительно много времени, поэтому пользователь может использовать параметр **newb=** для того, чтобы задать программе команду использовать уже созданный блок файл .

newb= 0 использовать существующий блок файл, 1 – создать новый (по умолчанию 0). Если используется блок-файл, отличный от такового по умолчанию (*vels.blk* в директории проекта), то не забудьте указать его в параметре **blkf=**.

Для минимизации рассеивания лучей возможно будет необходимо использование сглаженной скоростной модели. Модель можно сгладить в программном продукте, использовавшемся для построения скоростной модели, либо использовать модуль **Velocity Smoothing**, который сглаживает модель и записывает новый скоростной или блок файл. По умолчанию Ray Tracer не сглаживает модель. Для более подробной информации, смотрите описание модуля Velocity Smoothing.

Миграция от плавающего уровня или топографии

Когда съемка получена от плавающего уровня или топографии, то было бы правильно мигрировать данные также от плавающего уровня. В Tsunami PSTM имеется два метода для

миграции данных от плавающего уровня. Первый и исходный метод состоит в том, что статическая поправка по времени или глубине считывается из заголовков входных трасс. Уровень приведения будет равен нулю. Метод 2, новый, внесенный в программу, начиная с версии 4.11.6. Для этого метода статическая поправка должна быть задана по глубине, и пользователь указывает значение уровня приведения. Оба эти метода описаны ниже.

Метод 1

Прежде чем запустить задание RayTracer от плавающего уровня на счет, вы должны создать базу трасс, используя модуль PSDM. Для миграции от плавающего уровня сейсмические трассы в своих заголовках должны содержать значения статических поправок во времени или глубине. Статическая поправка по времени подразумевает собой время пробега в одну сторону в миллисекундах, записанную в формате четырехразрядного целого числа (4 byte integer). Глубинная статическая поправка может быть в метрах или футах. При положительном значении поправки, подразумевается, что плавающий уровень находится ниже уровня приведения. Это наиболее распространенный случай, хотя иногда встречаются отрицательные значения поправок и рассчитываются они соответственно (см. Диаграмму метода 1). Поправка должна быть сглаженной статикой, и поправка для всех трасс на одну ОГТ одной и той же. При создании базы трасс в PSDM, все значения поправок занесутся в эту базу. Raytracer, в свою очередь, во время счета запрашивает базу данных трасс для трассирования лучей от плавающего уровня. Параметр, описывающий порядковый номер байта, в котором записана поправка называется **fldtm=**. Во время работы PSDM лог файл выводит информацию о минимальном и максимальном значениях статических поправок, считанных из заголовков трасс. Пользователю рекомендуется проверить эти значения и сопоставить с ожидаемыми.

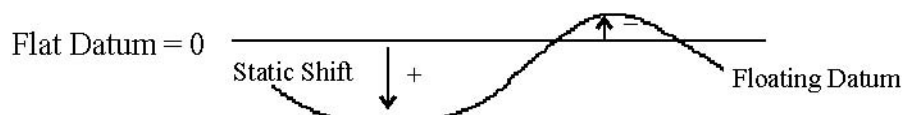
fldtm= требуется для плавающего уровня

staticd= статическая поправка во времени (мсек) или глубине (футы, метры). По умолчанию – во времени.

vdtm= скорость замещения. Используется только в тех случаях, когда поправка по глубине (**staticd =1**)

Исходные сейсмические данные для миграции от плавающего уровня подразумеваются приведенными к плавающему уровню, где плавающий уровень время = 0.

Миграция от плавающего уровня метод 1



Трасса скорости для задания RayTracer может приниматься (отсчитываться) как от плавающего уровня, так и от времени =0. Различные обрабатывающие системы записывают скоростную модель по разному, поэтому пользователю необходимо проверить каким способом был записан файл. Параметром RayTracer, контролирующим от какого уровня

будут рассчитываться скорости, является **frdtm=** где 1 (по умолчанию) означает, что скорости берутся от плавающего уровня; **frdtm=0** означает, что скорости берутся от времени =0 (time zero).

frdtm=0 фиксированный уровень

frdtm=1 плавающий уровень (по умолчанию)

Параметр RayTracer **vdtm=** используется для задания скорости замещения. Эта скорость необходима для конвертации временной статической поправки в глубинную.

vdtm= скорость замещения, используемая между фиксированным и плавающим уровнями.

После того, как вы создали базу трасс и задали дополнительные параметры, вы запускаете задание RayTracer на счет.

Во время последующего расчета PSDM вы укажете параметры номера байта для поправок (**fldtm=**), указатель статической поправки (**staticd=**) и скорость замещения (**vdtm=**) в вашем файле параметров.

Метод 2

Прежде чем запустить задание Raytracer от плавающего уровня на счет, вы должны создать базу трасс, используя модуль PSDM. Для миграции от плавающего уровня, сейсмические трассы в заголовках должны содержать значения высоты плавающего уровня. Для Метода 2, в байте записаны значения расстояния в метрах или футах между плавающим уровнем и уровнем =0. Положительные числа означают, что уровень над 0 (см. Диаграмму метод 2). Если у вас такой случай, вам необходимо задать значение уровня приведения (**dtmelev=**) и значение заменяющей скорости (**vdtm=**). Также необходимо отметить, что поправка по глубине **staticd=** 1.

dtmelev= уровень приведения в метрах или футах (по умолчанию 0.0) Выходные трассы будут записаны от этого уровня . Высота над уровнем моря задается отрицательным значением.

vdtm= скорость замещения, используемая между фиксированным и плавающим уровнями. Используется только в тех случаях, когда поправка задана по глубине (**staticd=1**)

staticd= статическая поправка во времени (мсек) или глубине (футы, метры). По умолчанию – во времени.

Трасса скорости для задания Raytracer может приниматься (отсчитываться) как от плавающего уровня, так и от времени =0. Различные обрабатывающие системы записывают скоростную модель по разному, поэтому пользователю необходимо проверить каким способом был записан файл. Параметром Raytracer, контролирующим от какого уровня будут рассчитываться скорости, является **frdtm=** где 1 (по умолчанию) означает, что скорости берутся от плавающего

уровня; **frdtm**=0 означает, что скорости берутся от времени =0 (time zero). Задание значения заменяющей скорости включает кнопку плавающего уровня :

frdtm= 0 фиксированный уровень

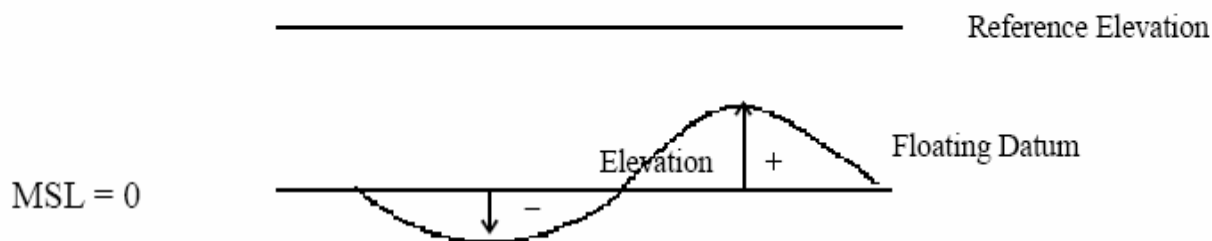
frdtm=1 плавающий уровень (по умолчанию)

Параметр RayTracer **vdtm**= используется для задания заменяющей скорости. Эта скорость необходима для конвертации временной статической поправки в глубинную.

vdtm= заменяющая скорость, используемая между фиксированным и плавающим уровнями. После того, как вы создали базу трасс и задали дополнительные параметры, вы запускаете задание RayTracer на счет.

Во время последующего расчета PSDM вы укажете параметры номера байта для поправок (**fldtm**=), указатель статической поправки (**staticd**=) и заменяющую скорость (**vdtm**=) в вашем файле параметров.

Миграция от плавающего уровня метод 2



Донная коса

В версию 5.2.13 включена опция, позволяющая обрабатывать данные, записанные на донную косу. При включении данной опции программа подразумевает, что ПВ находятся на плоском уровне (поверхность моря), а ПП – на плавающем уровне, соответствующем профилю дна; опция **Shots/Receivers Corrected To** автоматически переключится на значение «плавающий уровень». Данная опция позволяет вычислить правильные углы вхождения, подобно опции миграции от рельефа; статические поправки вносят в данные углы искажения

Просматриваем времена пробега.

Начиная с версии 3.1.9, пользователь может по усмотрению записать файл времен пробега в формате SEG Y. Этот файл можно использовать для контроля качества времен пробега перед запуском самой миграции. Данный SEG Y не будет считываться миграцией, он создается только для просмотра. Значения Инлайн, Кросслайн и ПВ записаны как четырехразрядные целые числа в байтах 181,185 и 17 соответственно.

Вы можете записать segy файл, указав параметр **fmto**= 2. По умолчанию **fmto**= 1 , что означает необходимый для глубинной миграции формат.

Приложение А: Информация лог файла

Пример лог-файла RayTracer

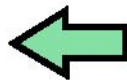
Лог-файл находится в директории проекта : `~proj_dir/parameter_file.log`

Программа RayTracer предоставляет много полезной информации в своем лог файле. При запуске нового задания на трассирование, проверьте и убедитесь, что вы правильно считываете сейсмические заголовки и данные скоростной модели.

Для удобства, при старте программы лог распечатает интервал значений инлайнов и кросслайнов, которые покрывает скоростная модель. Далее, будут распечатаны значения минимальной и максимальной скоростей в данной модели, а также данные заголовка первой трассы исходных данных. Эти значения должны быть проверены пользователем и сопоставлены с ожидаемыми.

```
Version 4.11 Tsunami Ray Tracer
Parameter file: new_ttime.example
```

```
nddb= '/home/example/new_tests/node02.db'
hmdr= '/home/software/tsunami_4.11.6/rays_4.11.6'
prjdr= '/home/example/new_tests'
logfile= '/home/example/new_tests/ttime.log'
velf= '/home/example/velocities.segy'
blkf= '/home/example/new_tests/tmp.blk'
outf= '/home/example/ttimes.out'
xlfrst= 3920
xllst= 5541
ilinc= 20
ilfrst= 1982
illst= 3325
xlinc= 20
vxlb= 5
vilb= 1
vdz= 50
newb= 1
dataf= 2
hdrfmt= 2
eiknl= 0
nswep= 2
mode= 2
xlfrstout= 3920
xllstout= 5541
xlincout= 10
ilfrstout= 1982
illstout= 3325
ilincout= 10
aper= 23000
mxoff= 21000
```



Проверка 1

Входные параметры, заданные либо интерактивно, либо из файла параметров. Проверьте параметры `vdz=`, `vilb=`, `vxlb=`, `fnt=`, `xldist=`, `ildist=`, `aper=` and `mxoff=` на правильность значений. Полный список параметров находится в Приложении В.

xldist= 55
ildist= 55
rqpcnt= 60

Jun 23 15:54
Reading license file /home/software/psdm_license.dat
Current license expires month 1 day 31 year 2006
Reading the velocity model
Checking the sort order, and the limits of model
Velocity model coverage: First xline 3920 Last xline 5540 xline inc 20
First inline 1982 Last inline 3322 inline inc 20
Velocity model z increment: 50



Проверка 2
Проверьте правильность
лицензионного файла

Model dimensions

Model header size 3600

First x 0.0 Number of x values 82 X distance 1100.0
First y 0.0 Number of y values 68 Y distance 1100.0
First z 0.0 Number of z values 680 Z distance 50.0

Minimum velocity in model 4971
Maximum velocity in model 14486
Building velocity block model
Smoothing operator x dimen 1
Smoothing operator y dimen 1
Smoothing operator z dimen 1
Allocated 5.0 percent of the block model
Allocated 10.0 percent of the block model
Allocated 15.0 percent of the block model
Allocated 20.0 percent of the block model
Allocated 25.0 percent of the block model
Allocated 30.0 percent of the block model
Allocated 35.0 percent of the block model
Allocated 40.0 percent of the block model
Allocated 45.0 percent of the block model
Allocated 50.0 percent of the block model
Allocated 55.0 percent of the block model
Allocated 60.0 percent of the block model
Allocated 65.0 percent of the block model
Allocated 70.0 percent of the block model
Allocated 75.0 percent of the block model
Allocated 80.0 percent of the block model
Allocated 85.0 percent of the block model
Allocated 90.0 percent of the block model
Allocated 95.0 percent of the block model
Allocated 100.0 percent of the block model
Max Derivatives in Smoothed Model
dx 0.6 dy 0.4 dz 10.4
Some calculated parameters:
Diameter of ttime cube x direction: 67000
Diameter of ttime cube y direction: 67000
Number of subsurface locations in x direction 122



Проверка 3
Проверьте правильность
пределов скоростного файла

Number of subsurface locations in y direction 122
Number of output depth levels 307
Outputting file format 1
Header successfully written

Server has connected to node node01
Server has connected to node node02
Server has connected to node node03
Server has connected to node node04
Server has connected to node node05
Server has connected to node node06
Server has connected to node node07
Server has connected to node node08
Server has connected to node node09
Server has connected to node node10
Server has connected to node node11
Server has connected to node node12
Server has connected to node node13
Server has connected to 13 nodes with 26 processors
Server failed to connect to 0 nodes
Server has connected to 100.0 percent of processing power
Min vel in model 4971.097656
Max vel in model 14485.589844
Node node01 successfully allocated 250 mbytes of memory
Node node02 successfully allocated 250 mbytes of memory
Node node03 successfully allocated 250 mbytes of memory
Node node04 successfully allocated 250 mbytes of memory
Node node05 successfully allocated 250 mbytes of memory
Node node06 successfully allocated 250 mbytes of memory
Node node07 successfully allocated 250 mbytes of memory
Node node08 successfully allocated 250 mbytes of memory
Node node09 successfully allocated 250 mbytes of memory
Node node10 successfully allocated 250 mbytes of memory
Node node11 successfully allocated 250 mbytes of memory
Node node12 successfully allocated 250 mbytes of memory
Node node13 successfully allocated 250 mbytes of memory
Total number of shots 5727

***** BEGIN SHOOTING RAYS *****

Processing shot 1 at inline 1982, xline 3920 at 0.0 min, node node01
Processing shot 2 at inline 1982, xline 3940 at 0.0 min, node node01
Processing shot 13 at inline 1982, xline 4160 at 0.0 min, node node07
Processing shot 14 at inline 1982, xline 4180 at 0.0 min, node node07
Processing shot 11 at inline 1982, xline 4120 at 0.0 min, node node06
Processing shot 12 at inline 1982, xline 4140 at 0.0 min, node node06

Однообразные строчки стёрты для экономии места

Processing shot 2323 at inline 2522, xline 5540 at 0.0 min, node node12
Processing shot 2324 at inline 2522, xline 5560 at 0.0 min, node node13
Processing shot 2325 at inline 2542, xline 3920 at 0.0 min, node node13
Shot 2325 number of fill in rays 16524

Однообразные строчки стёрты для экономии места

Shot 5717 median ray diff 0.0070 80 percentile diff 0.0137
Shot 5717 percent empty grids 2.45
Shot 5725 number of fill in rays 14111
Shot 5725 median dist to ray 453 80 percentile dist 710
Shot 5725 median ray diff 0.0068 80 percentile diff 0.0130
Shot 5725 percent empty grids 1.46
Shot 5718 number of fill in rays 13664
Shot 5718 median dist to ray 449 80 percentile dist 719
Shot 5718 median ray diff 0.0070 80 percentile diff 0.0135
Shot 5718 percent empty grids 2.20
Shot 5724 number of fill in rays 13758
Shot 5724 median dist to ray 453 80 percentile dist 711
Shot 5724 median ray diff 0.0067 80 percentile diff 0.0130
Shot 5724 percent empty grids 1.58
Shot 5723 number of fill in rays 13863
Shot 5723 median dist to ray 455 80 percentile dist 715
Shot 5723 median ray diff 0.0066 80 percentile diff 0.0131
Shot 5723 percent empty grids 1.79
Shot 5722 number of fill in rays 13802
Shot 5722 median dist to ray 454 80 percentile dist 718
Shot 5722 median ray diff 0.0066 80 percentile diff 0.0130
Shot 5722 percent empty grids 2.66
Shot 5721 number of fill in rays 13637
Shot 5721 median dist to ray 453 80 percentile dist 714
Shot 5721 median ray diff 0.0067 80 percentile diff 0.0131
Shot 5721 percent empty grids 2.92
Shot 5720 number of fill in rays 13665
Shot 5720 median dist to ray 443 80 percentile dist 707
Shot 5720 median ray diff 0.0071 80 percentile diff 0.0136
Shot 5720 percent empty grids 3.65
Shot 5719 number of fill in rays 13555
Shot 5719 median dist to ray 447 80 percentile dist 715
Shot 5719 median ray diff 0.0073 80 percentile diff 0.0139
Shot 5719 percent empty grids 3.42

*****RAY SHOOTING COMPLETED*****

***** Median Stats on Shots *****

Median distance between rays 422 80 percentile distance 690
Median difference between rays 0.0063 80 percentile difference 0.0124

RAYS Main: Successful Completion
Jun 25 15:40

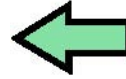


Проверка 4

Проверьте успешность
завершения трассирования
лучей

Проверка лог-файла лицензии

Product name rays
Creation date 010104
Start date 010104
End date 123104
Length of hw record 96
Number of licensed nodes 100



Проверка 1
Даты начала и окончания
действия лицензии

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	
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	1
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

}



Проверка 2
Единичка означает, что
нод отключён от лицензии

Пример лог файла очистки нодов:

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

Приложение В: Информация файла параметров

Необходимые параметры

- hmdr=** домашняя директория программы Необходима
- prjdr=** Директория проекта Необходима
- velf=** segy файл интервальных скоростей Файл должен иметь регулярную дискретизацию по инлайн, кросслайн и глубине. Необходим Нет значения по умолчанию
- blkf=** имя блок файла скоростной модели, созданного программой Необходим Нет значения по умолчанию
- outf=** имя файла записи времен пробега Необходим Нет значения по умолчанию
- xldist=** расстояние между кросслайнами Необходим Нет значения по умолчанию
- ildist=** расстояние между инлайнами Необходим Нет значения по умолчанию
- xlfrst=** первый кросслайн для запуска лучей Необходим Нет значения по умолчанию
- xllst=** последний кросслайн для запуска лучей Необходим Нет значения по умолчанию
- ilfrst=** первый инлайн для запуска лучей Необходим Нет значения по умолчанию
- illst=** последний инлайн для запуска лучей Необходим Нет значения по умолчанию
- ilfrstout=** первый инлайн для записи Необходим Нет значения по умолчанию
- illstout=** последний инлайн для записи Необходим Нет значения по умолчанию
- xlfrstout=** первый кросслайн для записи Необходим Нет значения по умолчанию
- xllstout=** последний кросслайн для записи Необходим Нет значения по умолчанию
- aper=** используемая апертура миграции (радиус) Необходим Нет значения по умолчанию
- mxoff=** максимальный используемый офсет Необходим Нет значения по умолчанию
- vilb=** номер байта для инлайна в скоростной модели Необходим Нет значения по умолчанию
- vxlb=** номер байта для кросслайна в скоростной модели Необходим Нет значения по умолчанию
- vdz=** шаг по глубине скоростной модели Необходим Нет значения по умолчанию
- nddb=** список нодов - Необходим Нет значения по умолчанию
- См: Конфигурация кластера, описание
- ilincout=** инкремент по инлайнам для записи по умолчанию = 5
- xlincout=** инкремент по кросслайнам для записи по умолчанию = 5

Список параметров в алфавитном порядке

Параметр	Описание	Необх одим	Возможные значения	Значение по умолчанию
accfac=	Фактор точности создания новых лучей, функция от угла к ПВ. Уменьшение параметра ведёт к увеличению			4.0

	точности.			
angnum=	Количество инкрементов углов по сфере			10
ani=	Латеральная анизотропия			
aper=	Апертура	Да		Нет
azinc=	Инкремент угла в горизонтальной плоскости			2.5
blkf=	This is the name of the velocity block file created by the program	Да		Нет
dataf=	Формат скоростной модели		0= IEEE_LE 1= IEEE_BE 2 = IBM	2 = IBM
dinc=	Фиксированный инкремент глубинного дискрета. Этот параметр отменяет действие параметра frqpr=		.	30 м (межд.) 90 футов (англ.)
dmax=	Максимальная глубина выходного файла			Максимальная глубина модели
dtacc=	Дискрет входного сейсмического файла, в миллисекундах			8
eiknl=	Метка Эйконал/RayTracer		0 = Эйконал 1 = ray tracer	0 = Эйконал
epdelr=	Отношение ϵ/δ			Нет
etaf=	Имя файла с ETA-параметром			
fmto=	Создание QC SEG Y файла или файла пробега времён		1 = пробег 2 = SEG Y	
frdtm=	Скорости от плоского или плавающего уровня приведения		0 = плоский 1 = плавающий	1 = плавающий
frqpr=	Пары «глубина-частота», в одинарных кавычках.			нет
hdrfmt=	Формат заголовков файла скоростей		0= IEEE_LE 1= IEEE_BE 2 = IBM	1= IEEE_BE
hmdr=	Директория программы			
ildist=	Расстояние между инлайнами	Да		Нет
ilfrst=	Номер первого входного	Да		Нет

	инлайна			
ilfrstout=	Номер первого выходного инлайна	Да		Нет
ilinc=	Инкремент инлайнов на входе			Нет
ilincout=	Инкремент инлайнов на выходе			Нет
illst=	Номер последнего инлайна на вход	Да		Нет
illstout=	Номер последнего инлайна на выходе	Да		Нет
logfile=	Имя лог-файла			
maxray=	Максимально количество лучей			75
mnegang=	Максимальный негативный (т.е. вверх по отношению к горизонтальной плоскости) угол трассировки..		Максимум = 0 Минимум = -89.9	-70
mode=	Метод учёта неоднократных отражённых лучей		0 = максимум энергии 1 = минимум времени 2 = минимум расстояния	1 = минимум времени
mxnegdst=	Максимально возможное расстояние для прохождения луча, идущего вверх от горизонтальной плоскости		Максимум = 10000 Минимум = 0	2000
mxoff=	Максимальный используемый оффсет	Да		Нет
nddb=	Список нодов – см. раздел <i>Конфигурация кластера</i>	Да		Нет
newb=	Создание нового блок-файла		1 = создать новый 0 = использовать имеющийся файл	1 = создать новый
nsweep=	Количество sweeps			2
outf=	Имя файла пробега времён	Да		нет
phinc=	Инкремент угла в вертикальной плоскости			2.5
prjdr=	Директория проекта			
rqpcent=	Процент использования			100

	нодов			
rystff=	Фактор «жёсткости» трассирования лучей			0
smfctr=	Сглаживание скоростной модели		0 = выключено 1 = включено	0 = выключено
smth=	Параметр сглаживания времён пробега		0 = нет сглаживания	3
stpsz=	Шаг при трассировке			вычисляется
subil=	Интервал разрежения скоростной модели по инлайнам		0 = нет разрежения 1 = есть разрежение	1 = есть разрежение
subxl=	Интервал разрежения скоростной модели по кросслайнам		0 = нет разрежения 1 = есть разрежение	1 = есть разрежение
tres=	Фактор точности в уравнении Эйконала			4
twod=	2D/3D		0 = 3D 1 = 2D	0 = 3D
vdtm=	Скорость приведения от плавающего к плоскому уровню.			0 = не используется
vdz=	Шаг дискретизации в скоростной модели	Да		Нет
velf=	Файл скоростной модели (SEG-Y). Интервалы по инлайнам, кросслайнам и глубине должны быть равномерными	Да		Нет
vilb=	Байт заголовка скоростной модели, в котором записан номер инлайна	Да		Нет
vxlb=	Байт заголовка скоростной модели, в котором записан номер кросслайна	Да		Нет
vxlilty=	Формат скоростного файла		0 = 4 byte integer 1 = 2 byte integer	0 = 4 byte integer
xldist=	Расстояние между кросслайнами	Да		Нет
xlfrst=	Первых входной кросслайн	Да		Нет
xlfrstout=	Первый выходной кросслайн	Да		Нет
xlinc=	Инкремент по			Нет

	кросслайнам на входе			
xlincout=	Инкремент по кросслайнам на входе			Нет
xllst=	Последний входной кросслайн	Да		Нет
xllstout=	Последний выходной кросслайн	Да		Нет

Пример файла параметров

Параметры могут располагаться в файле параметров в любом порядке. Любая строчка не являющаяся параметром автоматически становится комментарием. Все имена файлов или пары чисел заключаются в одинарные скобки. После названия параметра следует знак равенства без пробела, а значение параметра пишется через пробел. Ниже приводятся примеры задания параметров:

velf= '/data/velocity.segy'	Скоростной файл
blkf= '/data/velocity.blk'	Скоростной блок файл
logfile= '/data/logfile.out'	Лог-файл
outf= '/data/times.out'	Файл пробега времен
nddb= 'node_database.db'	Файл со списком узлов
vdz= 30	Дискрет по глубине в скоростном файле
vilb= 180	Байт заголовка скоростной модели, в котором записан номер инлайна
vxlb= 184	Байт заголовка скоростной модели, в котором записан номер кросслайна
xldist= 25.0	Расстояние между кросслайнами
ildist= 50.0	Расстояние между инлайнами
xlfrst= 10	Первый входной кросслайн
xllst= 100	Последний входной кросслайн
xlinc= 5	Шаг (инкремент) кросслайнов
ilfrst= 20	Первый входной инлайн
illst= 100	Последний входной инлайн
ilinc= 5	Шаг инлайнов
ilfrstout= 20	Первый выходной инлайн
illstout= 100	Последний выходной инлайн
ilincout= 5	Шаг инлайнов на выходе
xlfrstout= 10	Первый выходной кросслайн
xllstout= 100	Последний выходной кросслайн
xlincout= 5	Шаг кросслайнов на выходе
aper= 6000	Апертура
mxoff= 6000	Максимальный оффсет
dinc=100	Шаг (дискрет) по глубине
newb= 1	Создание нового скоростного блок-файла
mode= 1	Расчет лучей по способу минимальных расстояний

Приложение С: Инсталляция Java

Для графического интерфейса Tsunami и программы просмотра Viewer необходимо иметь инсталляцию Java 2 (J2SE) версии 1.4.2 или выше, а также Java 3D. Если у вас не имеется необходимой версии, то ваш системный администратор может скачать и установить ее, проделав следующее:

Проверить текущую версию Java на вашем кластере:

- 1) `java -version` Эта команда выведет на дисплей текущую версию.
- 2) `which java` Эта команда покажет где установлена Java.

Если при наборе этих команд никакой информации не показано, то проверьте, создана ли переменная `JAVA_HOME` в файле пользователя `.cshrc` or `.bashrc`.

Скачать и установить Java и Java 3D на Linux и Itanium платформы:

- 1) Зайдите на веб-сайт Java :
<http://java.sun.com/j2se/index.jsp>
- 2) Выберите последний релиз (не Beta версия). Должен быть J2SE SDK версии 1.4.2 или выше. Заметьте: вам необходимо скачать J2SE версии 5.0 если вы используете AMD Opteron 64 bit linux.
- 3) Пожалуйста, внимательно прочитайте инструкции по установке Java для вашей системы. Метод самораспаковывающегося файла рекомендуется.
- 4) Скачайте файл в вашу директорию. Еще раз проверьте для какой системы выбран файл: Linux 32 bit или Linux 64 bit Itanium 2
- 5) Следуйте инструкциям по инсталляции J2SE SDK версии 1.4.2 или выше для вашей платформы.
- 6) Отредактируйте пользовательские файлы `.cshrc` или `.bashrc` включив переменную `JAVA_HOME`. Пропишите путь к директории `java`.

Для `.cshrc`: `setenv JAVA_HOME path`

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

- 7) Добавьте `JAVA_HOME/bin` в переменную `PATH`.

Для **cshell**: PATH находится в **.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

Для **bourne shell**: PATH находится в **.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) Для проверки правильности заданных параметров и переменных, наберите следующие команды:

a) java -version Покажет номер версии вашей java.

b) which java Покажет, где проинсталлирована java.

9) После того, как вы проинсталлировали J2SE SDK версии 1.4.2 или выше, скачайте Java 3D SDK версии 1.3.1 или выше (не Beta версия) в директорию, созданную в п.5 :

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

Пожалуйста, удостоверьтесь, что вы выбрали нужную версию: Linux 32 bit или Linux 64 bit Itanium 2 Не забудьте скачать лицензию и файл Readme.

10) Пожалуйста, внимательно прочитайте инструкции по установке Java 3D SDK версии 1.3.1 для вашей системы.

11) Прогоните тесты, рекомендуемые в инструкции по установке для подтверждения правильности.

Скачать и установить Java и Java 3D на SGI :

1) Убедитесь, что у вас нет инсталляции Java SDK версии 1.4.1_06 и выше в вашей IRIX системе. Эта инсталляция **перепишет** все предыдущие версии Java SDK 1.4.1 или ниже.

2) Пожалуйста, внимательно прочитайте инструкции по установке Java для SGI Метод самораспаковывающегося файла рекомендуется.

http://www.sgi.com/products/evaluation/6.5_java2_1.4.1_06/

Внизу страницы выберите **Check Requirements** кнопку.

После просмотра, выберите и нажмите кнопку **Install**. Подтвердите согласие на лицензию, нажав кнопку **Accept License**. Полностью прочитайте инструкцию по установке и нажмите кнопку **Troubleshooter** внизу страницы для дополнительных опций. После того, как вы прочитали все инструкции по

установке, вернитесь на страницу Installation Instructions и выберите **Install** опцию для java2_еое.

- 3) Отредактируйте пользовательские файлы .cshrc или .bashrc включив переменную JAVA_HOME. Пропишите путь к директории java.

Для .cshrc: setenv JAVA_HOME path

Для .bashrc: export JAVA_HOME="path"

- 4) Добавьте JAVA_HOME/bin в переменную PATH.

Для cshell: PATH находится в .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

Для bourne shell: PATH находится в .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) Для проверки правильности заданных параметров и переменных, наберите следующие команды:

- a) java -version Покажет номер версии вашей java.

- b) which java Покажет, где проинсталлирована java.

- 6) Приготовьтесь инсталлировать Java 3D версии 1.3.1 или выше для IRIX, проверив требования на страничке:

http://www.sgi.com/products/evaluation/6.5_java3d_1.3.1/

Выберите **Check Requirements** кнопку внизу страницы.

После просмотра, выберите кнопку **Continue** внизу страницы. После просмотра, выберите и нажмите кнопку **Install**. Подтвердите согласие на лицензию, нажав кнопку **Accept License**. Полностью прочитайте инструкцию по установке и нажмите кнопку **Troubleshooter** внизу страницы для дополнительных опций. После того, как вы прочитали все инструкции по установке, вернитесь на страницу Installation Instructions и выберите опцию **Install**.