

**УФИМСКИЙ
УНИВЕРСИТЕТ
НАУКИ И ТЕХНОЛОГИЙ**

Курс «Технологии
параллельного
программирования»

**Порядок использования
учебного сегмента
суперкомпьютера УУНиТ
при выполнении
лабораторных работ**

Юлдашев Артур Владимирович
art@ugatu.su

Спеле Владимир Владимирович
spele.vv@ugatu.su

Добровольцев Александр Сергеевич
dobrovolcev.as@ugatu.su

Сохатский Михаил Александрович
sohatskii.ma@ugatu.su

**Кафедра высокопроизводительных
вычислений и дифференциальных
уравнений (ВВиДУ)**

Выход на кластер

ip-адрес (головного узла) кластера – 194.190.227.28

имя пользователя – student

1. Терминальный доступ (ssh)
Linux: консольный ssh-клиент (команда ssh), например:
ssh [student@194.190.227.28](ssh://student@194.190.227.28)

Windows: средствами терминального клиента PuTTY

2. Передача данных (ftp/sftp/ssh)

Linux: средствами файлового менеджера mc
(F9->Left/Right->FTP Link; [student@194.190.227.28](ftp://student@194.190.227.28))

Windows: средствами файлового менеджера
Far Manager (Alt+F1/F2->FTP)

Домашняя папка пользователя student – /gss/home/student

В ней необходимо создать папку группы, а в ней индивидуальную папку,
в которую складывать свои файлы.

Компиляция и запуск MPI-программ

1. Компиляция (используется реализация MPI – Intel MPI 4.x):

#исполняемый файл будет называться a.out
> **mpicc mpiprog.c**

или

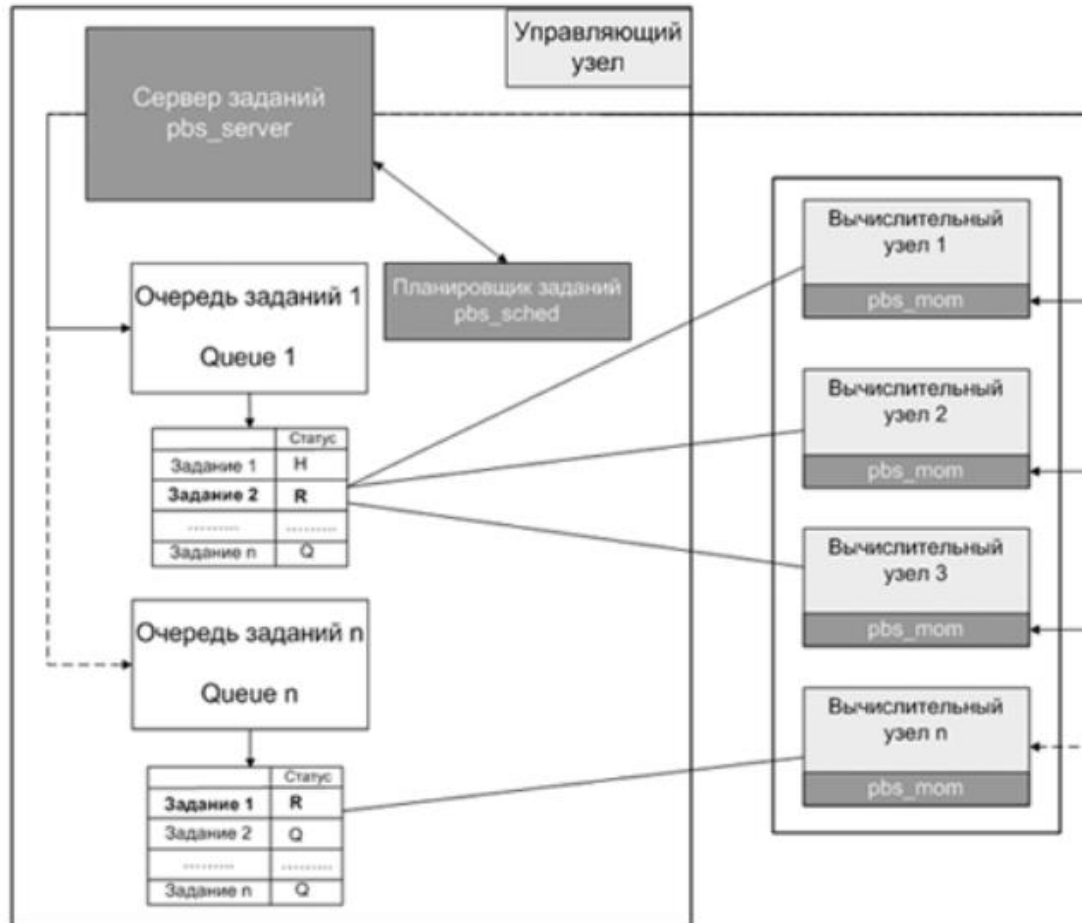
#исполняемый файл будет называться mpiprog
> **mpicc -o mpiprog mpiprog.c**

или

#для максимальной производительности
> **mpiicc -o mpiprog_icc mpiprog.c**

2. **Запускать MPI-программы на головном узле запрещено!** Для проведения расчетов используется система очередей/менеджер ресурсов/система пакетной обработки TORQUE.

Система очередей TORQUE



http://hpc.ssau.ru/files/doc/torque_manual.pdf

Пример сабмит-скрипта

Приведенный скрипт расположен по адресу `/gss/home/student/nbody_labs/init_file/`
Там же лежит тестовая MPI-программа `test.cpp`

<code>#!/bin/bash</code>	
<code>#PBS -l nodes=4:student:ppn=8</code>	<code>#ресурсный запрос, в данном примере</code>
	<code>#запрашивается 4 узла по 8 ядер на каждом</code>
<code>#PBS -l walltime=30:00</code>	<code>#ограничение времени выполнения – 30 мин.</code>
<code>#PBS -A email</code>	<code>#имя (e-mail) владельца задачи</code>
<code>#PBS -N group_lastname</code>	<code>#имя задачи</code>
<code>cd \${PBS_O_WORKDIR}</code>	<code>#переход в рабочую папку</code>
<code>date</code>	<code>#вывод текущего времени</code>
<code>HOSTFILE=\${PBS_JOBID}.hosts</code>	<code>#здесь формируется файл</code>
<code>cat \${PBS_NODEFILE} sort uniq > \${HOSTFILE}</code>	<code># со списком узлов для mpirun</code>
<code>NUM_NODES=`cat \${HOSTFILE} wc -l`; cat \${HOSTFILE}</code>	
<code>mpirun -r ssh -f \${HOSTFILE} -ppn 8 -n 32 ./a.out</code>	<code>#запуск MPI-программы</code>
	<code>#32 процесса по 8 на каждый узел</code>

Постановка задач в TORQUE

Запуск осуществляется посредством постановки задачи, описанной в специальном файле - “субмит-скрипте”, в систему очередей TORQUE командой **qsub**:

> qsub job.pbs

где **job.pbs** – имя “субмит-скрипта” (пример приведен на следующем слайде).

При успешной постановке в очередь задача получает уникальный идентификатор (**JOBID**), который выводится после выполнения **qsub**, к примеру:

393827.login.nodes

Далее можно управлять задачей по JOBID – 393827.login.nodes
или короче – 393827

После запуска задачи в папке, откуда она была запущена, создаются файлы с расширением .oJOBID и .eJOBID. В них содержится информация, выводимая задачей в потоки stdout и stderr.

Работа с системой очередей

Дополнительные команды работы с системой очередей TORQUE:

qstat - просмотр состояния поставленных задач

примеры:

> **qstat -f JOBID** #просмотр подробной информации о задаче JOBID

> **qstat -n1** #форматированный вывод списка задач

qdel JOBID - отмена задачи

qalter JOBID – изменение параметров задачи

qhold JOBID - блокировка задачи

qrls JOBID - вывод из состояния блокировки

pbsnodes - просмотр состояния вычислительных узлов

пример:

> **pbsnodes -l free :student** #просмотр списка свободных узлов

По всем командам доступна справочная информация через **man**,
например,

> **man qstat**

Состояния задач в очереди

При просмотре состояния поставленных задач командой `qstat` можно увидеть следующие статусы:

C - Job is completed after having run.

E - Job is exiting after having run.

H - Job is held.

Q - Job is queued, eligible to run or routed.

R - Job is running.

T - Job is being moved to new location.

W - Job is waiting for its execution time (-a option) to be reached.

```
watch -n 1 qstat
```


Задание

1. Создать свою папку в /gss/home/student/nbody_labs/

```
mkdir Фамилия_Имя
```

```
cd /gss/home/student/nbody_labs/Фамилия_Имя/
```

2. Скопировать код, реализующий гравитационное взаимодействие тел, сабмит-скрипт и соурс-файл в свою папку

```
cp /gss/home/student/nbody_labs/init_file/test.cpp .
```

```
cp /gss/home/student/nbody_labs/init_file/job.pbs .
```

```
cp /gss/home/student/nbody_labs/init_file/source.sh .
```

```
source source.sh
```

3. Скомпилировать исходный код

```
mpicc -O2 -o test test.cpp
```

Задание

4. Открыть сабмит-скрипт в своей папке и изменить данные, отмеченные красным цветом

```
#!/bin/bash
#PBS -N UserName

#PBS -l nodes=1:student:ppn=8

#PBS -l walltime=36000:00

cd /gss/home/student/nbody_labs/Фамилия_Имя/

source source.sh

date                                     #вывод текущего времени

HOSTFILE=${PBS_JOBID}.hosts            #здесь формируется файл

cat ${PBS_NODEFILE} | sort | uniq > ${HOSTFILE} # со списком узлов для mpirun

NUM_NODES=`cat ${HOSTFILE} | wc -l`; cat ${HOSTFILE}

mpirun -r ssh -f ${HOSTFILE} -ppn 8 -np 1 ./test 122880
```

Задание

5. Провести тестирование производительности MPI-версии с 122880 телами. Запуски проводить с числом процессов 1, 2, 4, 8, 16, 32, 64, 128. Для запуска сабмит-скрипт использовать команду

`qsub job.pbs`

6. Построить графики ускорения и эффективности параллельной реализации.

Гравитационное взаимодействие N тел.

Моделирование гравитационного взаимодействия N тел численно аппроксимирует эволюцию системы тел, в которой каждое тело непрерывно взаимодействует с любым другим телом. Известным примером является астрофизическое моделирование, в котором каждое тело представляет собой галактику или отдельную звезду, и тела притягиваются друг к другу с помощью силы притяжения.

Движение каждого тела описывается уравнением Ньютона:

$$\frac{m_i d^2 r_i}{dt^2} = \sum_{j \neq i} \frac{G m_i m_j}{|r_j - r_i|^3} (r_j - r_i), \text{ где}$$

m_i, m_j — массы тел

r_i, r_j — координаты тел

G — гравитационная постоянная