Высокопроизводительный вычислительный центр НИЯУ МИФИ

Руководство пользователя

Версия 1.3.0

Москва, 2014

Все замечания и предложения по этой документации с благодарностью принимаются по адресу hpc@lists.ut.mephi.ru.

Создано при помощи XeLaTeX и BibTeX.

Последнее изменение: 25 марта 2014 г.

Содержание

1. Общие положения

Высокопроизводительный вычислительный центр НИЯУ МИФИ (далее — HPC-Центр) предназначен для выполнения ресурсоёмких и/или распределённых вычислений сотрудниками и учащимися Университета при выполнении научных, исследовательских и образовательных задач, в частности, для обучения использованию современных HPC-технологий.

В рамках имеющихся аппаратных ресурсов пользователям предоставляется возможность выполнять задачи с использованием следующих технологий на основе ОС Linux:

- PBS (менеджер ресурсов Torque [1], планировщик Maui [2]);
- MPI (реализация OpenMPI [3]);
- OrangeFS [4] (распределённая параллельная виртуальная файловая система, поддерживающая ROMIO [5]).

2. Как стать пользователем

Перед тем как подавать заявку на использование ресурсов Центра, пожалуйста, ознакомьтесь с правилами использования. Фактом подачи заявки Вы подтверждаете своё согласие с данными правилами и обязательность их исполнения Вами.

2.1. Правила использования

1. Ресурсы HPC-Центра предназначены для поддержки фундаментальных и прикладных научных исследований,исследовательских и образовательных задач, требующих привлечения HPC.
2. Пользователями могут быть: сотрудники, учащиеся НИЯУ МИФИ и организаций, имеющих с НИЯУ МИФИ совместные научные проекты.
3. Пользователям категорически запрещается передавать свою учётную запись, пароль к ней или секретный ssh-ключ иным лицам.
4. Пользователь обязуется не использовать Центр для задач не указанных в п.1, в т.ч. для какой-либо деятельности, противоречащей законодательству РФ.

5. Установка и использование нелегального программного обеспечения категорически запрещена, как и любое нарушение лицензий на используемое ПО (например, попытка использования в HPC-Центре бесплатного только для частного использования ПО).

6. Пользователям запрещается пытаться обойти систему защиты, квот или административных ограничений HPC-Центра, в частности, эксплуатировать уязвимости.

7. В случае обнаружения уязвимости системы, пользователь обязан незамедлительно сообщить об этом администраторам лично по e-mail hpc-private@ut.mephi.ru.

8. Административные объявления, а также проблемы и предложения пользователей обсуждаются через список рассылки hpc@lists.ut.mephi.ru. Почта администраторов (hpc-private@ut.mephi.ru) предназначена только для решения проблем безопасности.

9. Администрация HPC-Центра выполняет тщательный аудит действий пользователей.

10. В случае обнаружения нелегитимной активности или нарушения данных правил со стороны пользователя, администрация оставляет за собой право блокирования соответствующей учётной записи с возможным удалением нелегальных данных.

11. Пользователи в публикациях работ, выполненных при помощи HPC-Центра, обязуются ссылаться на использование его ресурсов. Для русскоязычных работ следует использовать формулировки вида «при проведении работ были использованы ресурсы высокопроизводительного вычислительного центра НИЯУ МИФИ» для англоязычных — “our work was performed using resources of NRNU MEPhI high-performance computing center”.

12. Администрация прилагает все возможные усилия для безотказной работы HPC-Центра и сохранности пользовательских данных. Однако, в силу объективных обстоятельств, невозможно гарантировать абсолютную стабильность и сохранность информации, поэтому пользователи должны регулярно сохранять полученные результаты и хранить копии особо важных данных вне ресурсов HPC-Центра.

2.2. Получение учётной записи

Для получения учётной записи Вам необходимо заполнить заявку http://redmine.ut.mephi.ru/projects/hpc-requestи подписаться на список рассылки hpc@lists.ut.mephi.ru. Затем с Вами будет согласовано время визита в В-123 для идентификации пользователя (возьмите с собой пропуск в МИФИ или удостоверение) и Вам будет выдан логин/пароль от сервера аутентификации.

При желании Вы сможете изменить пароль с помощью команды passwd, но при этом он должен будет соответствовать строгим требованиям безопасности как по длине, так и по сложности. Система не позволит Вам установить слабый пароль.

2.2.1. Получение учётных данных по e-mail

Так же можно получить учётные данные по электронной почте в зашифрованном виде. Для этого необходимо создать PGP-ключ для указанного e-mail и разместить его открытый подключ на любом из публичных GPG-серверов, затем рекомендуется отправить отпечаток ключа с регистрируемого e-mail на hpc-private@ut.mephi.ru. Период синхронизации всех серверов составляет около суток, поэтому создавайте ключ заблаговременно. Инструкцию по созданию ключей и работе с GnuPG можно найти в работе [6].

Обратите внимание, что данный механизм получения пароля не отменяет необходимость персональной идентификации пользователя.

2.2.2. Регистрация группы пользователей

При необходимости получить учётные записи для большой группы пользователей, можно подать заявку сразу на всю группу в виде служебной записки от руководителя группы или подразделения на имя начальника управления информатизации Романова Николая Николаевича. В служебной записке должны быть перечислены ФИО пользователей, их e-mail, цель и срок завершения работ (например, срок окончания обучения). Ответственность за достоверность предоставленных данных находится на руководителе, подавшем служебку.

Данный механизм авторизации требует использования цифровых ключей для почты пользователей, описанных в разделе 2.2.1. Пароли для всей группы на руки не выдаются, участникам группы приходить в В-123 не нужно.

Для групповой заявки также рекомендуется заполнить он-лайн заявку http://redmine.ut.mephi.ru/projects/hpc-request.

2.2.3. Использование ssh-ключей

При желании, пользователь после получения пароля может использовать ssh-ключи для доступа к ресурсам HPC-Центра. Для этого необходимо *на клиентской машине* создать ключ с помощью:

```
ssh-keygen -b 521 -t ecdsa
```

Не забудьте указать сложный пароль для защиты ключа! Затем поместите ключ в файл `~/.ssh/authorized_keys` на сервере аутентификации. Обратите внимание, что передача ключа иным лицам категорически запрещена.

3. Ресурсы HPC-Центра

3.1. Ферма «Университетский кластер»

Вычислительные ресурсы Унивеситетского кластера составляют:

- 128 ядер;
- 512 GB RAM;
- 1.4 TB полезного дискового пространства;
- сеть 1 Gbit/s;
- пиковая производительность $\sim$ 1.5 TFlops.

Кластер состоит из 16 узлов. Каждый узел состоит из:

- 2 x E5450 Intel Xeon CPU;

- 4 физических ядра на CPU.
- 32 GB RAM (DDR2 667 MHz);
- 120 GB HDD;
- BCM5715S Gigabit Ethernet.

Управляющий узел Университетского кластера выполняет функции сервера аутентификации

3.2. Ферма «Basov»

Основные параметры фермы:

- 340 физических ядер;
- 2.5 TiB RAM;
- 7.3 TiB полезного дискового пространства;
- сеть 80 Gbit/s;
- пиковая производительность $\sim$ 7.2 TFlops.

Ферма включает в себя 1 управляющий, 3 файловых и 20 вычислительных узлов. Каждый узел состоит из:

- 2 x E5-2680 Intel Xeon CPU;
- 8 физических ядер на CPU;
- 128 GiB RAM (DDR3 1600 MHz);
- 300 GiB HDD;

3.3. Программная инфраструктура

Фермы работают на базе операционной системы Linux, дистрибутив Gentoo [7].

Пользователи непосредственно работают по ssh только с управляющими узлами ферм, они же предназначены для компиляции приложений. Работа с вычислительными узлами осуществляется посредством инструментов PBS без прямого доступа пользователя.

На фермах предоставляются следующие инструменты:

PBS На фермах используется система управления распределёнными вычислениями (PBS, Portable Batch System) на основе менеджера ресурсов Torque [1] версии 3.0.6 и диспетчер задач Maui [2] версии 3.3.1. В рамках PBS реализована поддержка MPI задач.

MPI Поддержка MPI (Message Passing Interface, инструмент для обмена данными между параллельно работающими задачами на разных узлах) реализована с помощью пакетов OpenMPI [3] и mpich2 [8]. Текущие версии пакетов можно узнать при помощи утилиты eix (подробнее в 4.4).

С точки зрения пользователя, MPI-приложение работает внутри PBS-задачи. Поддерживается ROMIO [5] I/O API.

Работа MPI-приложений ускорена с использованием технологии KNEM [9], особо эффективной для передачи больших объёмов данных, асинхронного и векторного обмена данными.

PVFS2 На Университетском кластере применяется распределённая параллельная виртуальная файловая система OrangeFS [4] версии 2.9_beta-r1,являющаяся ветвью PVFS2. Данное решение позволяет максимально полно использовать имеющиеся ресурсы дискового пространства, а также предоставить пользователям все достоинства параллельного ввода-вывода данных, что позволяет на грамотно спроектированных приложении получать скорости доступа к файлам, ограниченные лишь пропускной способностью сети.

Distcc Предусмотрена возможность использования ферм для помощи в распределённой компиляции во внутренних сетях НИЯУ МИФИ с использованием технологи distcc [10] версии 3.2. На данный момент находится в стадии тестирования.

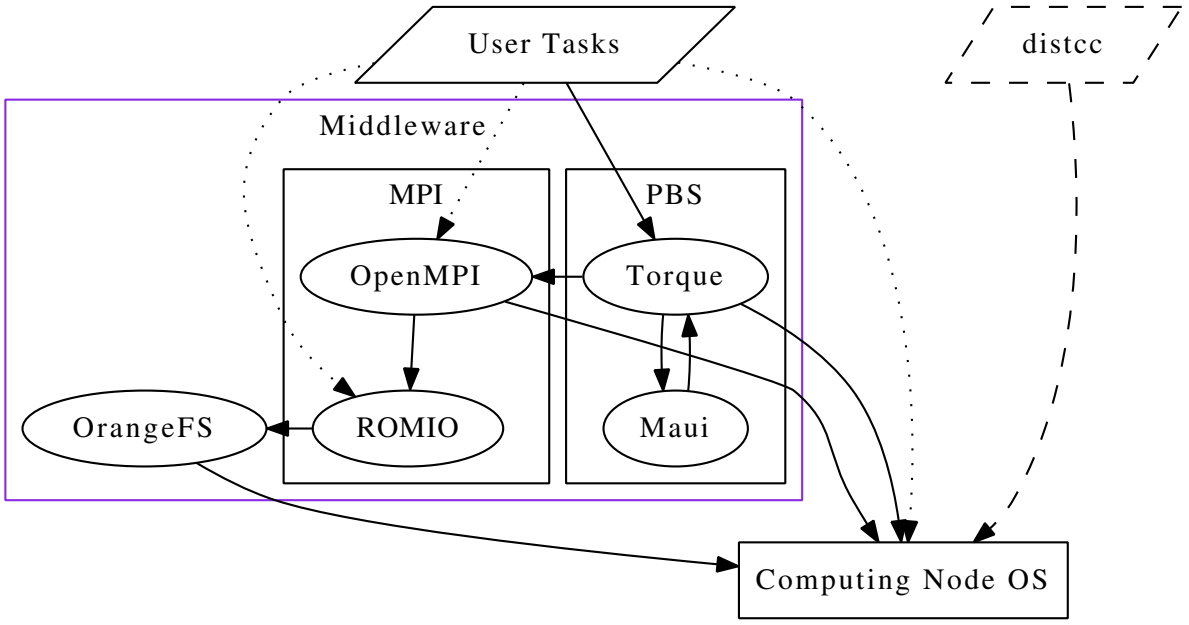

Рис. 1. Основная инфраструктура ПО кластера

GCC Для компиляции пользовательских приложений предоставляется стандартная для Linux коллекция компиляторов GCC [11] версий 4.7.3 (основная системная), 4.6.4 и 4.5.4, поддерживающая следующие языки: C, C++, Assembler, Fortran, Objective C, Objective C++. Поддерживается технология OpenMP [12]. Для компиляции MPI приложений следует использовать соответствующие команды с префиксом «mpi»: mpicc, mpic++, mpif77, mpi90.

Ccache Для ускорения повторной компиляции приложений и упрощения разработки предоставляется кеш компилятора ccache [13] версии 3.1.9. Пакет уже настроен и для использования пользователю ничего не нужно делать: ccache будет задействован автоматически при использовании gcc [11] для языков: C, C++, Objective C, Objective C++.

Текстовые редакторы Представлены текстовые редакторы Vim, Emacs и, для новичков, mcedit и nano.

Взаимодействие основной инфраструктуры ПО кластера с пользовательскими задачами отображено на рис. 1.

Обратите внимание, что X-сервер на наших фермах не поддерживается. Аппаратные акселераторы визуализации физически отсутствуют. Фермы предназначен для вычислительных задач, а не для визуализа-

ции полученных результатов, которую необходимо выполнять на клиентских системах.

3.4. Предустановленное ПО

На фермах имеется предустановленное ПО для задач области физики частиц: ROOT, Geant, Pythia. Также, имеется ПО из областей механики сплошных сред (OpenFOAM) и молекулярной динамики (LAMMPS). При поступлении заявок, иное научное программное обеспечение может быть установлено общесистемно и поддерживаться системными администраторами, при условии что оно доступно в стандартных репозиториях Gentoo (portage, science overlay).

3.5. Политика обновлений

С целью поддержания актуальности и безопасности установленного общесистемного ПО, будут проводится регулярные обновления системы. Плановое обновление будет проходить раз в полгода (или раз в год, в зависимости от обстановки), между учебными семестрами. Администраторы оставляют за собой право проводить экстренное обновление отдельных компонент при выявлении серьёзных проблем безопасности.

4. Использование ферм

4.1. Предварительные требования

Для работы с фермами пользователю необходимо обладать базовыми навыками работы в ОС Linux; в частности, необходимо владеть `bash`, `ssh`, `gcc`, одним из установленных текстовых редакторов (`vim`, `emacs`, `nano`, `mcedit`), уметь устанавливать приложения и обладать навыками программирования, достаточными для компиляции пользовательских приложений.

Большинство этих аспектов выходит за рамки данного руководства, которое посвящено описанию предоставляемой рабочей среды (см. раздел 3.3) и особенностям, специфичным для ферм HPC-Центра. Для каждого приложения на фермах установлена документация, доступная посредством `man`, `info` и в `/usr/share/doc`. Однако, начинающим в Linux можно порекомендовать следующие материалы для ознакомления: Linux in a Nutshell [14], Advanced Bash-Scripting Guide [15]. Для отладчика `gdb` имеется доступное руководство: RMS's gdb Debugger Tutorial [16].

4.2. Работа с ssh

Доступ пользователей на фермы осуществляется по протоколу ssh на адрес hpc.mephi.ru. Функциии сервера аутентификации выполняет управляющий узел Университетского кластера. Управляющий узел фермы Basov доступен с сервера аутентификауии по адресу basov

```
ssh basov
```

При работе с ssh следует учитывать ограничения на интенсивность соединений, вызванные соображениями политики безопасности. Необходимо избегать повторных подключений, удерживая уже установленные соединения. При необходимости реализации нескольких соединений можно использовать механизм сокетов ssh. Для использования этого механизма следует включить его на стороне пользователя. Например, для пакета OpenSSH достаточно дописать в файл /etc/ssh/ssh_config:

```
ControlMaster auto
ControlPath ~/.ssh/socket-%r@%h:%p
```

При отсутствии доступа к фермам по протоколу ssh следует обращаться к системным администраторам. Обращаем Ваше внимание на то, что утилита ping не предназначена для проверки доступности ssh соединений.

4.3. Работа с файловой системой

Пользователям предоставляется три вида дисковых хранилищ:

- /home на NFS [17];
- /tmp;
- ~/pool на OrangeFS [4].

4.3.1. /home

Домашняя директория пользователей расположена на NFS-4 разделе и обладает полной POSIX-совместимостью. В силу ограниченности ресурсов, дисковое пространство для каждого пользователя в $HOME ограничено размером 2 GB и количеством файлов 100 000. Ограничения можно кратковременно превышать в определённых пределах, однако, при длительном превышении (7 дней и более) пользователь будет

автоматически заблокирован. Узнать о Вашей текущей дисковой квоте можно с помощью команды `quota`.

Данный раздел предназначен для компиляции пользователями своих приложений, а также для использования специальных файлов (сокеты, fifo и т.п.) приложениями, для которых нужно, чтоб данные файлы были доступны для всех запущенных процессов на разных узлах одновременно.

4.3.2. ~/basov и ~/unicluster

Домашняя директория пользователя на ферме Basov доятупна с узлов Университетского кластера как `~/basov`. Аналогично, на ферме Basov осуществляется доступ к `~/unicluster`.

Внимание! Пропускная способность интерконнекта между фермами на порядок ниже, чем между узлами одной фермы. Не следует использовать эти точки монтирования в запускаемых процессах. Они предназначены только для подготовки запуска. Также следует помнить о существенных различиях в аппаратном обеспечении ферм. Запускаемые приложения нужно компилировать для различных ферм отдельно.

4.3.3. /tmp

Для хранения временных данных рабочего процесса на конкретном вычислительном узле предназначен раздел /tmp, так же обладающий полной POSIX-совместимостью. На дисковое пространство действуют те же ограничения, что и на раздел /home (см. 4.3.1), но срок временного превышения квоты сокращён до трёх дней.

Обратите внимание, что размер квоты относится ко всем пользовательским процессам, работающим на данном узле, суммарно. Файлы на /tmp, не используемые в течении 7 дней, будут автоматически удалены.

4.3.4. ~/pool

Основное хранилище данных каждой фермы доступно как ~/pool. На ферме Basov для хранилища данных используется файловая система NFS-4. На Университетском кластере используется кластерная распределённая параллельная виртуальная файловая система OrangeFS [4].

Хранилище предназначено для исходных данных, результатов обработки и прочих пользовательских данных, а также для установки пользовательских приложений. Система квот не применяется, но пользователям не рекомендуется занимать более необходимого и более 1/3 от

общего объёма хранилища. На данный момент на всей файловой системе Университетского кластера доступно 1.3 TB для всех пользователей суммарно (это предел технических возможностей оборудования). На ферме Basov доступно 7.3 TB.

Как и при работе с любой распределённой параллельной файловой системой, при работе с OrangeFS следует учитывать её характерные особенности. Данная файловая система является ветвью PVFS, предназначенной для HPC вычислений и оптимизированной для работы MPI приложений. Она не является полностью POSIX-совместимой, в частности, отсутствует механизм блокировок (POSIX file locks) — целостность данных гарантируется за счёт атомарности операций. Так же на ней нельзя создавать специальные файлы и жёсткие ссылки (для этих задач используйте `$HOME` или /tmp), но можно использовать символические ссылки.

Файловая система хорошо оптимизирована для параллельного доступа большого числа процессов с разных узлов, т.о. Вы можете использовать эффективный ввод-вывод данных при работе распределённых MPI-приложений. Подробнее о возможностях файловой системы можно узнать в вики [18] проекта.

При работе с OrangeFS настоятельно рекомендуется максимизировать размер данных в операциях чтения/записи (вплоть до 1MB). Если Ваше приложение будет производить работу с файлами блоками данных по несколько десятков байт, Вы получите резкое падение скорости чтения или записи. При невозможности исправить приложение для корректной работы с распределёнными параллельными файловыми системами, рекомендуется использовать /tmp для локального кеширования данных. Обратите внимание, что данное требование распространяется так же на любой сетевой обмен данными, будь то NFS или MPI.

В процессе штатной работы доступ к обычным файлам на OrangeFS, с точки зрения пользовательского процесса, ничем не отличается от работы с локальной файловой системой. На случай возникновения аварийных ситуаций есть специальные утилиты доступа к данным, начинающиеся с префикса `pvfs-`, подробно описанные в соответствующих `man` руководствах. При возникновении проблем нужно сообщить администраторам по форме `http://redmine.ut.mephi.ru/projects/hpc-support`.

4.4. Подготовка пользовательских приложений

В общем случае, пользовательское приложение должно быть откомпилировано на управляющем узле соответствующей фермы. Для этого используется стандартный набор компиляторов `gcc`, описанный в разделе 3.3. Также доступны отладчики `gdb` и `valgrind`.

На фермах предоставляется широкий набор системных и научных библиотек. Для того, чтоб узнать, есть ли библиотека (или любой пакет) в подключенных репозиториях и установлена ли она в системе, необходимо использовать команду eix, например (для библиотеки быстрых Фурье-преобразований fftw):

```
$ eix fftw
[U] sci-libs/fftw
     Available versions:
        (2.1)   2.1.5-r8
        (3.0)   3.2.2 (~)3.2.2-r2 (~)3.3.2
      {{altivec avx doc float fortran mpi neon openmp paired-single
quad sse sse2 static-libs threads zbus}}
    Installed versions:  3.3.1(3.0){tbz2}(03:00:31 PM 04/10/2012)
(fortran mpi openmp sse sse2 threads -altivec -avx -doc -neon -quad
-static-libs -zbus)
     Homepage:            http://www.fftw.org/
    Description:        Fast C library for the Discrete Fourier
Transform
```

Подробно использование данной команды описано в `man eix`.

Обращаем внимание пользователей на наличие различных реализаций библиотек линейной алгебры, поддерживающих следующие интерфейсы программирования приложений (API): blacs, blas, cblas, lapack, lapacke.

Если Вам необходима библиотека, имеющаяся в репозиториях, но не установленная в системе, сообщите об этом администраторам, используя форму http://redmine.ut.mephi.ru/projects/hpc-support. Если Вам нужна версия библиотеки, отличная от установленной, то в некоторых случаях возможна установка доплнительной версии, если не возникает конфликта с основной системной.

Если Вам необходимо приложение, имеющееся в официальных репозиториях, но не установленное на фермах, также обратитесь к администраторам. Но обратите внимание, что X сервер и сопутствующие приложения не поддерживаются. Вычислительные задачи можно откомпилировать без поддержки X.

4.5. Запуск задач

Запуск и удаление задач выполняются с помощью инструментов менеджера ресурсов Torque [1], исчерпывающе описанных в работе [19] и в соответствующих страницах `man`. На каждой ферме используется свой менеджер ресурсов.

В простейшем случае для запуска задачи достаточно выполнить команду:

```
qsub myjob.sh
```

В результате скрипт myjob.sh будет поставлен в очередь long и впоследствии запущен на одном из вычислительных узлов, с возможностью использовать одно ядро и 4GB оперативной памяти, с ограничением на время исполнения в 168 часов (1 неделя). Ограничение по времени исполнения астрономическое (walltime) и не зависит от степени загрузки CPU пользовательским процессом.

Независимо от директории, из которой была выполнена команда, приложение будет запущено в `$HOME` пользователя. После завершения работы программы, stdin и stdout будут размещены в файлах вида `~/${jobname}.o${job_id}` и `~/${jobname}.e${job_id}` соответственно.

Если Вашему приложению нужно передать аргументы или запустить его в директории, отличной от `$HOME`, необходимо использовать скрипт для выполнения соответствующих действий и с помощью qsub запускать этот скрипт, а не само приложение.

4.5.1. Запрос ресурсов

Вы можете явным образом указать необходимые для работы ресурсы, в частности, если необходимо запросить несколько ядер или изменить время исполнения. Например, следующая команда:

```
qsub -q medium -l nodes=2:ppn=8,walltime=10:00:00 job.sh
```

поставит задачу `job.sh` в очередь `medium`, запросив 2 узла с 8 ядрами на каждом и ограничив время исполнения до 10 часов.

Аналогичный результат можно получить, используя специально сформированный заголовок задачи, содержащий директивы `#PBS`:

```
#!/bin/bash
#
#PBS -q medium
#PBS -l nodes=2:ppn=8,walltime=10:00:00
```

В этом случае для запуска задачи достаточно выполнить:

```
qsub job2.sh
```

При наличии параметров как в заголовке задачи, так и в опциях командной строки, учитываются и те и другие, с приоритетом за опциями командной строки.

Пожалуйста, объективно оценивайте необходимые ресурсы и максимально точно их указывайте — это позволяет повысить эффективность работы диспетчера задач и уменьшить задержки в очередях. Полное описание доступных ресурсов можно найти в [20].

Внимание! Задачи, превысившие отведённое им время walltime, уничтожаются.

Внимание! На ферме Basov используется технология Hyper-threading, позволяющая использовать два вычислительных потока на одном физическом ядре процессора. Таким образом, менеджер ресурсов Torque предоставляет пользователю до 32 вычслительных потоков на каждом 16 ядерном узле. Если Вы не хотите пользоваться данной технологией, Вам необходимо конфигурировать свое приложение для числа потоков, вдвое меньшего, чем `ppn`.

4.5.2. Очереди задач

Задачи распределяются по очередям в зависимости от запрошенного времени исполнения. Чем меньшее время исполнения разрешено в очереди, тем выше её приоритет и тем быстрее начнут исполняться задачи. Это сделано для того, чтоб можно было быстро просчитать небольшие задачи без помех со стороны долгих задач. Кроме того, каждая очередь имеет свои ограничения по числу процессов, которые могут быть в ней одновременно запущены.

Можно явно запросить очередь с помощью:

```
qsub -q $queue_name
```

Естественно, если при этом указан walltime, он не должен противоречить параметрам очереди, если же он не указан, то применяются параметры по-умолчанию, установленные для данной очереди.

Просмотреть список доступных очередей можно с помощью:

```
qsub -Q
```

и детальную информацию по каждой очереди:

```
qsub -Q -f
```

Доступные очереди задач приведены в табл. 1. Для каждой очереди приведено минимальное время исполнения задачи t_{min}, максимально время t_{max} и присваиваемое время по-умолчанию t_{def}.

Если не задана ни очередь задачи, ни время исполнения, задача помещается в очередь long (168 часов), а для интерактивных задач — short (6 часов).

Очередь	t_{min}	t_{max}	t_{def}
short	0:00:00	6:00:00	6:00:00
medium	6:00:01	24:00:00	24:00:00
long	24:00:01	168:00:00	168:00:00
xxl	168:00:01	4320:00:00	2160:00:00
auto	Автоопределение по walltime		

Таблица 1. Очереди задач

4.5.3. Работа с MPI-приложениями

На фермах предусмотрена возможность работы с разными реализациями MPI: openmpi и mpich2.

Выбор MPI окружения По умолчанию на фермах используется openmpi, для его применения никаких дополнительных действий выполнять не нужно.

Для выбора иных реализций mpi следует использовать команду `eselect mpi`.

Для просмотра доступных альтернатив следует выполнить

```
eselect mpi list
Available MPI classs:
  mpi-mpich2-1_5-x86_64     --
```

Для выбора версии MPI следует выполнить

```
eselect mpi set version_name
```

Вернуться к стандартной версии openmpi можно при помощи

```
eselect mpi unset
```

Внимание! После каждого изменения конфигурации MPI следует выполнять команду

```
source /etc/profile
```

в активных шеллах. При открытии новых оболочек (либо при новых логинах пользователя) изменения будут задействованы автоматически.

Запуск MPI-приложений Для запуска MPI-приложений необходимо запросить нужное число ядер и/или узлов (см. раздел 4.5.1) и использовать mpirun внутри скрипта запуска задачи для выполнения нужного приложения. Указывать число процессов для mpirun не нужно: оно будет определено автоматически исходя из суммарного запрошенного числа ядер. В остальном работа с MPI-задачами не отличается от обычных задач.

Пример файла описания задачи:

```
#!/bin/bash
#
#PBS -l nodes=2:ppn=8,walltime=05:00:00

cd ~/workdir/
mpirun ./my_mpi_program
```

4.5.4. Интерактивные задачи

При возникновении проблем, в отладочных целях удобно использовать интерактивные задачи, которые предоставляют возможность отладки приложения непосредственно на вычислительном узле.

Для запуска интерактивной задачи следует использовать:

```
qsub -I
```

Интерактивные задачи не могут быть поставлены в очереди long или xxl.

4.6. Мониторинг и управление задачами

Пользователь может просмотреть статус *собственных* задач с помощью команды `qstat`. С параметрами загрузки очередей диспетчера задач Maui [2] можно ознакомиться используя `showq`.

Для удаления задач следует использовать: `qdel $job_id`.

5. Контакты

Сообщить о проблеме можно используя форму http://redmine.ut.mephi.ru/projects/hpc-support.

При обнаружении проблем, связанных с безопаснотью ферм, следует воспользоваться e-mail hpc-private@ut.mephi.ru. *Пожалуйста*, не пишите непосредственно администраторам, по всем вопросам обращайтесь только на указанный e-mail.

Подать заявку на использование ресурсов НРС-Центра можно по адресу http://redmine.ut.mephi.ru/projects/hpc-request, но перед этим Вы *должны* тщательно ознакомиться с разделом 2.

В остальных случаях можно связаться с нами и более опытными пользователями через список рассылки e-mail hpc@lists.ut.mephi.ru.

Список литературы

[1] Adaptive Computing Inc. — TORQUE™ Resource Manager, 2013. — URL: http://www.adaptivecomputing.com/products/open-source/torque/. 3, 8, 14

[2] Cluster Resources Inc. — Maui Cluster Scheduler, 2013. — URL: http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php. 3, 8, 18

[3] The Open MPI Project. — Open MPI: A High Performance Message Passing Library, 2013. — URL: http://www.open-mpi.org/. 3, 8

[4] Orange File System, 2013. — URL: http://www.orangefs.org/. 3, 8, 11, 12

[5] ROMIO: A High-Performance, Portable MPI-IO Implementation, 2013. — URL: http://www.mcs.anl.gov/romio/. 3, 8

[6] Миллер В. В., Наги Д. А. — Создание и использование ключа OpenPGP с подключами, 2007. — URL: https://www.pgpru.com/chernowiki/rukovodstva/bezopasnostj/upravleniekljuchami/podkljuchiopenpgp. 5

[7] Gentoo Foundation Inc. — Gentoo Linux, 2013. — URL: http://www.gentoo.org/. 8

[8] MPICH: High-Performance Portable MPI, 2013. — URL: http://www.mpich.org/. 8

[9] KNEM: High-Performance Intra-Node MPI Communication, 2013. — URL: http://runtime.bordeaux.inria.fr/knem/. 8

[10] distcc: distributed compilation for faster C/C++ builds, 2013. — URL: http://distcc.org/. 8

[11] Free Software Foundation Inc. — GCC, the GNU Compiler Collection, 2013. — URL: http://gcc.gnu.org/. 9

[12] The OpenMP® API specification for parallel programming, 2013. — URL: http://openmp.org/wp/. 9

[13] ccache — a fast C/C++ compiler cache, 2013. — URL: http://ccache.samba.org/. 9

[14] Linux in a Nutshell / Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins. — 6th edition. — O'Reily Media, 2009. — P. 944. — Url for 3rd edition. Google Books : wXIvheS3r_gC. 10

[15] Cooper Mendel. — Advanced Bash-Scripting Guide, 2013. — URL: http://tldp.org/LDP/abs/abs-guide.pdf. 10

[16] Schmidt Ryan Michael. — RMS's gdb Debugger Tutorial, 2013. — URL: http://www.unknownroad.com/rtfm/gdbtut/gdbtoc.html. 10

[17] Network file system, 2013. — URL: http://linux-nfs.org/wiki/index.php/Main_Page. 11

[18] OrangeFS Wiki, 2013. — URL: http://www.orangefs.org/trac/orangefs/wiki/WikiStart. 13

[19] Adaptive Computing Inc. — TORQUE™ Administrator Guide, 2013. — URL: http://www.adaptivecomputing.com/resources/docs/torque/3-0-3/. 14

[20] Adaptive Computing Inc. — TORQUE™ Job Submission, 2013. — URL: http://www.adaptivecomputing.com/resources/docs/torque/3-0-3/2.1jobsubmission.php. 16