Експериментална платформа - Суперкомпютър IBM BlueGene/P

Суперкомпютърът IBM BlueGene/P се използва и поддържа от Българския суперкомпютърен център. Архитектурата на суперкомпютъра е показана в таблиза 1.

Таблица 1. Архитектура на български суперкомпютър BlueGene/P

Елемент	Описание
Процесорно ядро (CPU core)	PowerPC 450 Тактова честота 850 MHz
Чип (Chip)	4 процесора PowerPC 450 Производителност 13,6 Gflops
Изчислителен възел (Compute card)	1 чип с производителност 13,6 Gflops Памет 2 GB DDR2 SDRAM
Възлова карта (Node Card)	32 изчислителни + 1 вх./изх. възела Памет 64 GB DDR2 SDRAM
Междинна платка (MidPlane)	16 възлови карти Памет 1 TB DDR2 SDRAM
Системен блок (Rack)	2 междинни платки Памет 2 TB DDR2 SDRAM
Суперкомпютър IBM Blue Gene/P	2 системни блока Памет 4 TB DDR2 SDRAM

Суперкомпютърът има още следните параметри:

- Теоретична производителност – 27,85 Tflops
- Измерена производителност (LINPACK) – 23,42 Tflops
- Енергийна ефективност – 371,67 Mflops/W
- Шестнадесет входно-изходни възела, свързани посредством оптични влакна към 10 Gbps мрежов комутатор.

Освен двата системни блока с изчислителни възли, суперкомпютърната система включва и следните по-важни компоненти:

- Front-End Node: сървър, общодостъпен за регистрираните потребители, чрез който пускат своите задачи. Архитектурата на процесорните ядра е PowerPC 64, а операционната система – SuSE Linux Enterprise Server 10 (SLES 10).
- Service Node: служебен сървър, който управлява цялостната работа на системата.
- Два файлови сървъра, посредством които Front-End Node сървъра и изчислителните възли осъществяват достъп до споделен дисков масив с размер 12 TB.

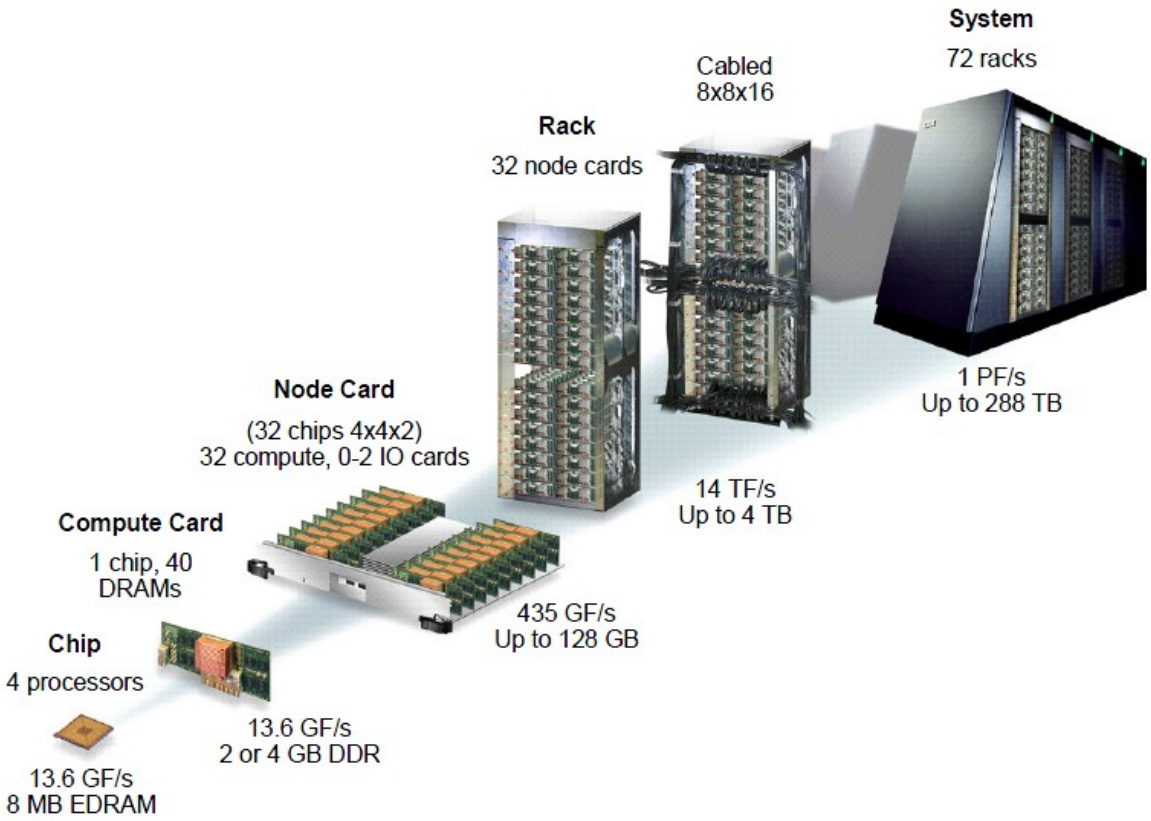
Допълнителна информация за IBM BlueGene/P

Достъп

Достъпът до суперкомпютъра е отдалечен и се осъществява посредством SSH протокол, който работи на порт 22.

Името на машината е bg-fen.scc.acad.bg.

Данни от и към машината могат да се копират посредством всяка програма, която поддържа SSH протокол.



Фиг. 1. Архитектура на суперкомпютър IBM BlueGene/P [IBM]

Файлова организация

Всички действия със системата се извършват от Front-End Node сървъра, който работи с операционната система Linux. Във файловата му система, освен стандартните Linux каталози (/etc, /root, /usr, /dev и т.н.), присъстват и специфични такива:

- /storage – основната потребителска файлова система с обем 4.4 TB. Всеки потребител разполага с каталог в /storage, който може да използва свободно.
- /bgsys – системен каталог на суперкомпютъра. Там е разположен системният софтуер, компилаторите, библиотеките и готовия софтуер, компилиран от екипа на центъра.

Компилиране на софтуер

Компилирането на софтуер (приложения и библиотеки), който се изпълнява на изчислителните възли, става посредством крос-компилация (cross-compiling): компилаторът работи на сървъра Front-End Node, но генерира код за изчислителните възли.

Системата разполага с два комплекта компилатори за C, C++ и FORTRAN: GNU Toolchain и IBM XL.

Изходният код на приложение, което използва библиотеките MPI или OpenMP, може да се компилира на суперкомпютъра, като във файла Makefile или в скрипта configure се укаже компилатора, с който ще се използва (или по друг начин, в зависимост от самото приложение).

Изпълнение на задачи

Разпределението на задачите се осъществява чрез Tivolli Workload Scheduler LoadLeveler. Това е паралелна система за планиране, разработена специално от IBM, която в зависимост от конкретните изчислителни нужди и приоритета на всяка задача преценява коя от тях да обработи най-напред, като се съобразява със свободните в момента ресурси и специални инструкции за максималното им използване.

Подготвените задачи се стартират за изпълнение с командата *llsubmit*. Изпълнимият файл, неговото обкръжение и аргументите се описват в т.н. Job Control File. Изпълнението на командата *llsubmit* довежда до поставянето на задачата в опашката от чакащи задачи, където получава уникален номер. Когато се появи възможност за изпълнение, задачата се изпраща на суперкомпютъра.

llsubmit gadget_128.jcf

Списъкът с чакащи задачи може да се види с командата *llq*, която извежда уникалния им номер, потребителите, които са ги стартирали, часът на тяхното постъпване в опашката от чакащи задачи, техният приоритет и статус. Ако статусът на дадена задача е *R*, то тя се изпълнява, ако е *I* – изчаква, а ако е *H*, означава, че е възникнал някакъв проблем и трябва да се премахне. Премахването става с командата *llcancel* и уникалният номер на задачата. Всеки потребител може да премахва единствено своите задачи.

llcancel bgpfen.23100.0

Съдържание на Job Control File (JCF)

```
# @ job_name = gadget_128
# @ comment = "This is a Gadget-2 Program"
# @ error = $(jobid).err
# @ output = $(jobid).out
```

```
# @ environment = COPY_ALL;
# @ wall_clock_limit = 20:00:00
# @ notification = never
# @ notify_user = never
# @ job_type = bluegene
# @ bg_size = 128
# @ class = n0128
# @ queue
time scalasca -analyze /bgsys/drivers/ppcfloor/bin/mpirun -
mode VN -np 512 ./Gadget2.galaxy galaxy.param
```

# @ job_name	Произволно име на задачата.
# @ comment	Произволен коментар (за собствено използване).
# @ error	Име на файл, в който се записват съобщенията за грешки.
# @ output	Име на файл, в който се записват изходните данни.
# @ environment	Указва, че всички валидни по време на изпълнението на llsubmit променливи от обкръжението трябва да се установят при пускането на задачата на изпълнителните възли.
# @ wall_clock_limit	Времева граница, след изтичането, на която LoadLeveler ще прекрати изпълнението на задачата. Тази граница не може да надвишава установената граница за класа на задачата.
# @ notification	Тъй като не е изградена инфраструктура за изпращане и получаване на електронна поща, тук се записва never.
# @ job_type	Този параметър съдържа стойността bluegene.
# @ bg_size	Цяло число, кратно на 128 и не по-голямо от 2048. Определя броя на възлите, които ще бъдат използвани за изпълнение на задачата. Трябва да отговаря на класа на задачата.
# @ class	Клас на задачата. Това е най-важният параметър и определя приоритетът, с който ще бъде пусната задачата, максималният брой изчислителни възли и времето за тяхното изпълнение.
# @ queue	Инструктира LoadLeveler да сложи задачата в опашката.

Следната команда води до изпращането на задачата до изчислителните възли на суперкомпютъра.

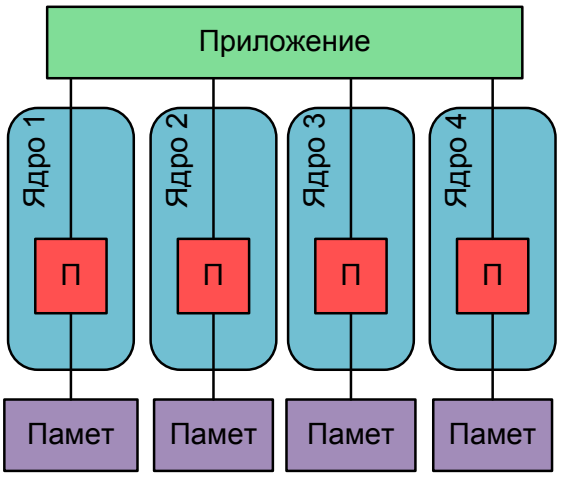
```
time scalasca -analyze /bgsys/drivers/ppcfloor/bin/mpirun -
mode VN -np 512 ./Gadget2.galaxy galaxy.param
```

Някои от параметрите на командата са:

`-exe <executable_file>`	Указва програмата, която ще се изпълнява.
`-args "<arguments>"`	Аргументи, които се подават на изпълнимата програма.
`-verbose 1`	Инструктира mpirun да изписва подробна информация за процеса на пускане на задачата.
`-mode VN/SMP/DUAL`	Указва режима на изпълнение на задачата.
`-np N`	Указва броят на процесорите, върху които ще се стартира програмата.

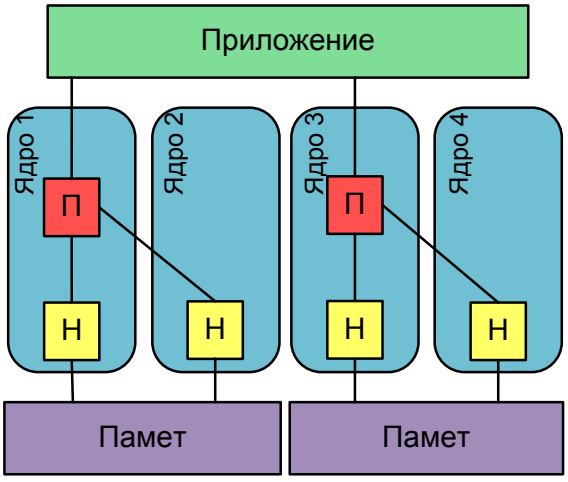
Режими на изпълнение

Режим VN: изчислителният възел се разделя на четири отделни процесора (фиг. 2). Всеки процесор изпълнява едно копие на програмата, като тя не може да използва нишки. Паметта се разделя на четири блока, по един за всеки процесор. В този режим 128 възела изпълняват 512 копия на програмата, а цялата система – 8192 копия. Всяко от копията има достъп до 512 MB памет.



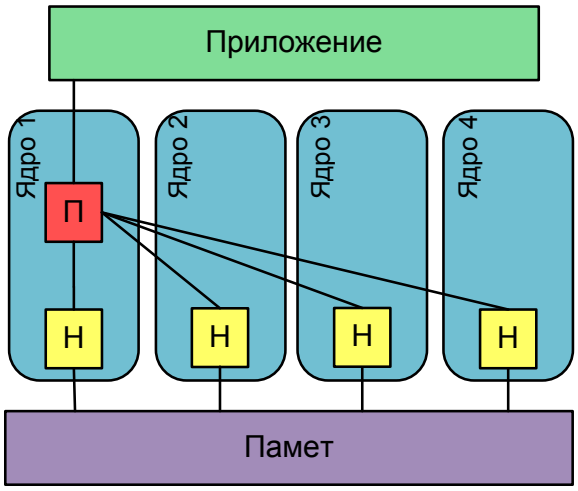
Фиг. 2. Режим на изпълнение VN. П – процес.

Режим DUAL: изчислителният възел се разделя на две двойки процесори (фиг. 3). Всяка двойка изпълнява едно копие на програмата, като тя може да стартира две нишки, всяка, от които се изпълнява на съответния процесор в рамките на двойката. Паметта се разделя на два блока, по един за всяка двойка. В този режим 128 възела изпълняват 256 копия на програмата, а цялата система – 4096 копия. Всяко от копията има достъп до 1 GB памет и може да пусне по две нишки.



Фиг. 3. Режим на изпълнение DUAL. П – процес, Н – нишка.

Режим SMP: изчислителният възел не се разделя (фиг. 4). Всеки възел изпълнява едно копие на програмата, като тя може да активира четири нишки, всяка, от които се изпълнява на съответния процесор в рамките на възела. Паметта не се разделя. В този режим 128 възела изпълняват 128 копия на програмата, а цялата система – 2048 копия. Всяко от копията има достъп до 2 GB памет и може да активира по четири нишки.



Фиг. 4. Режим на изпълнение SMP. П – процес, Н - нишка

Класове задачи

Класът на задачата определя:

- Приоритизацията – по-големите задачи се пускат с предимство пред по-малките;
- Максималният брой възли, които могат да се използват;
- Максималното време, за което задачата може да се изпълнява.

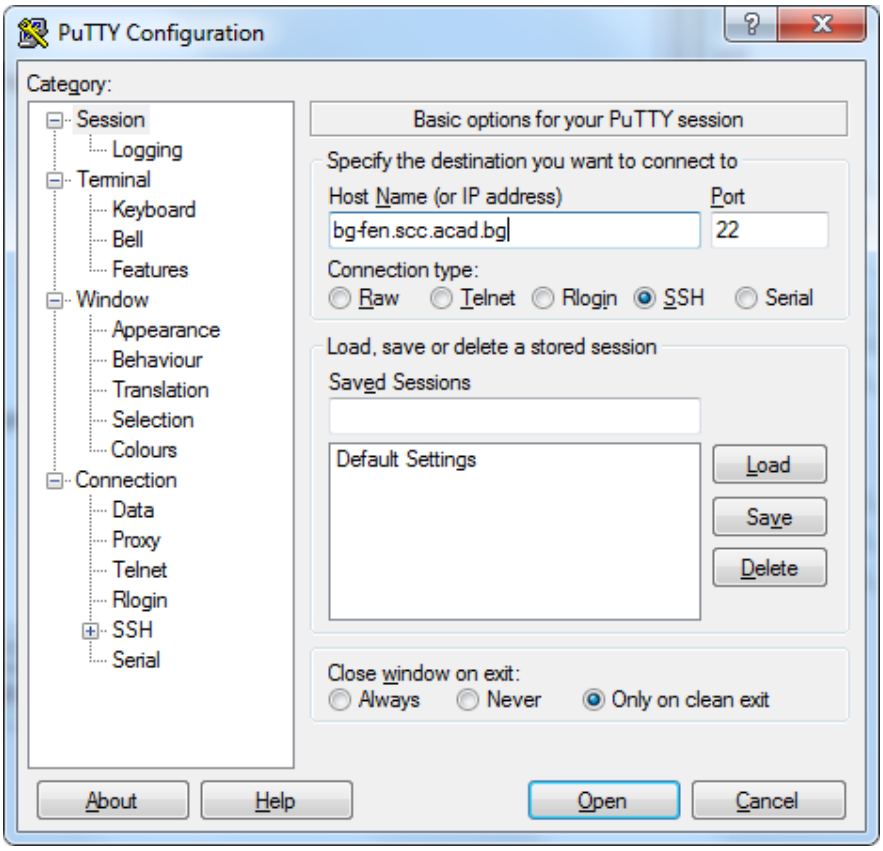
Едновременно могат да бъдат изпълнявани определен брой задачи от даден клас. Дефинирани са следните класове:

Таблица 2. Класове задачи

Клас	Брой задачи	Максимален брой възли	Максимално време за изпълнение
n0128	16	128	**24 часа**
n0128long	16	128	**7 дни**
n0256	6	256	**24 часа**
n0512	3	512	**24 часа**
n1024	1	1024	**24 часа**
n2048	**1**	**2048**	**24 часа**

Достъп до BlueGene/P

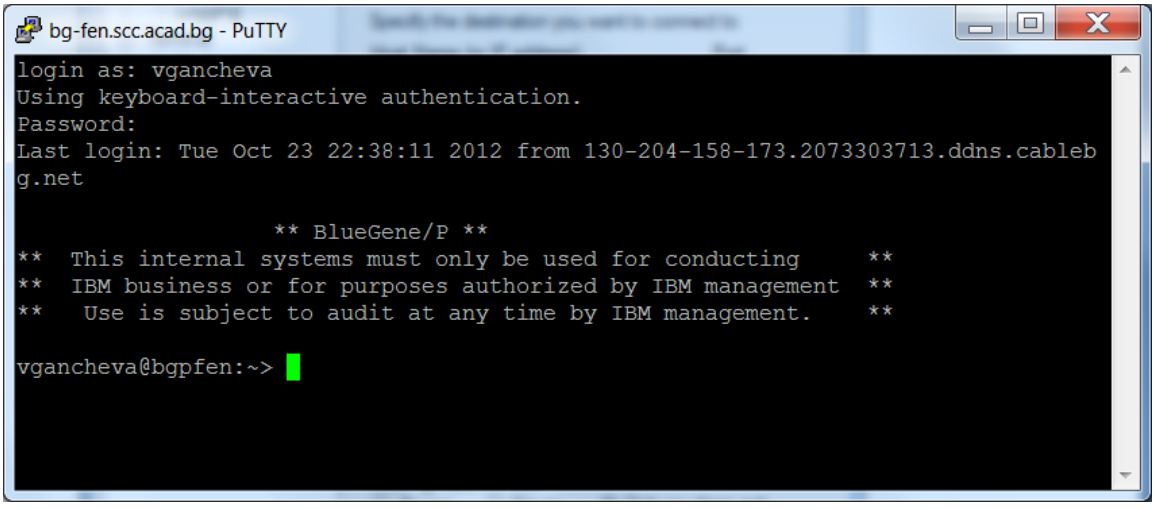
1. Сваляне на необходимите приложения. Необходими са SSH и SFTP клиент.
   a. SSH клиент – putty
   b. SFTP клиент – putty sftp



2. Стартирайте SSH клиента и се свържете към адреса на клъстера:

bg-fen.scc.acad.bg

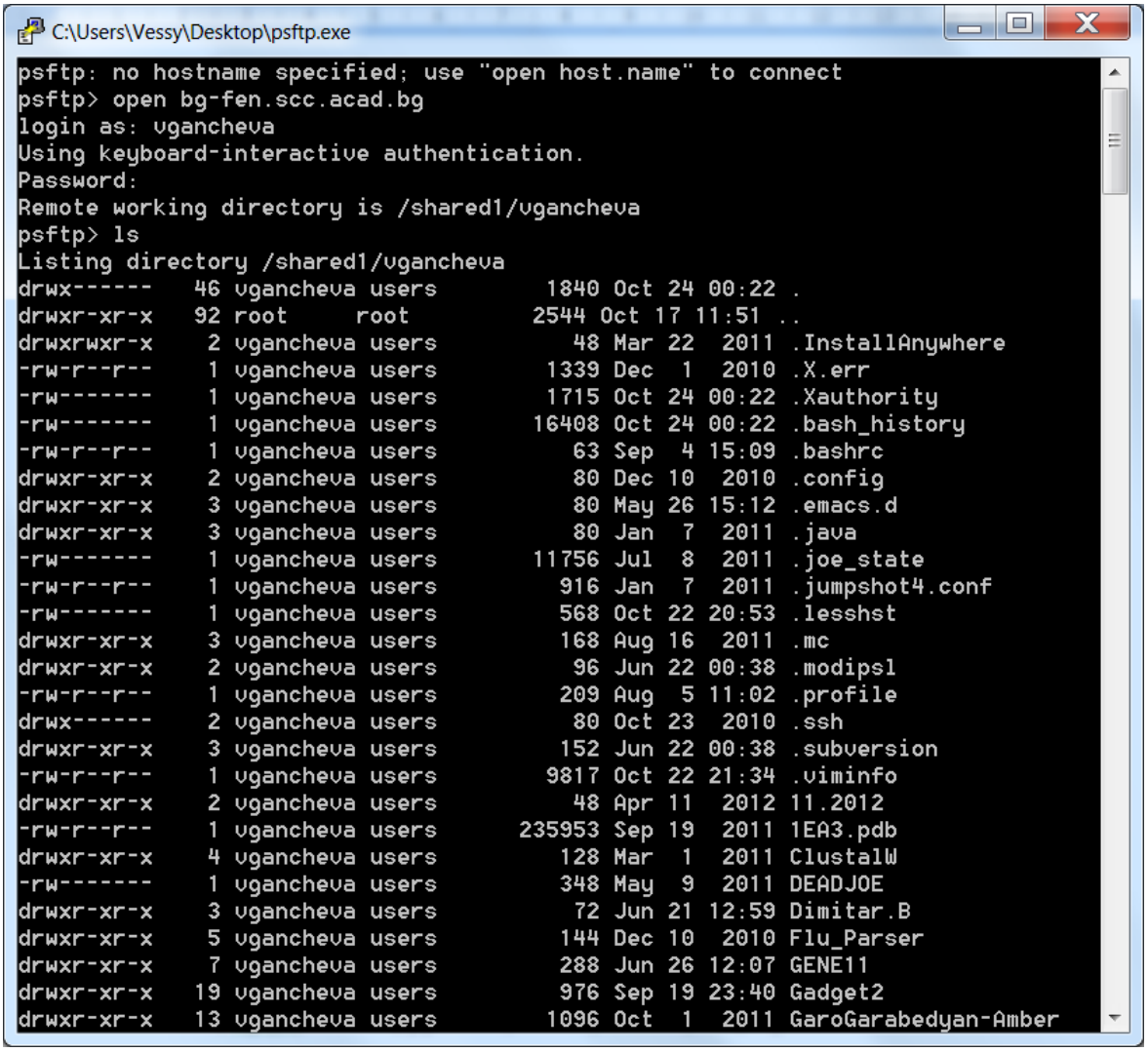
3. Въведете потребителя и паролата за достъп.



4. Когато вече сте се логнали може да използвате почти всички команди на linux. Някои от по-важните са:
    a. ls – показва съдържанието на директорията в която сте.
    b. cd – сменя директорията в която сте.
    c. cat – показва съдържанието на файл.
    d. vi – текстови редактори.

e. mv – преместване на файл или директория.
f. cp – копиране на файл или директория.
g. mkdir – създаване на директория.
h. pwd – показва къде се намирате.
i. tar – архивиране и разархивиране на файлове и директории.
j. unzip – разархивиране на ZIP архиви.

5. Можете да копирате файлове от локалната си машина на клъстера с помощта на SFTP клиента.
   a. Стартирайте psftp.exe.
   b. За да се свържете с клъстера въведете следната команда: *psftp> open bg-fen.scc.acad.bg*
   c. Въведете потребителското име и паролата от точка 2.



   d. Използвайте командата *mput,* за да качите файлове и директории. Файловете трябва да зададете с относителен път от директорията от която сте стартирали SFTP клиента или с абсолютен път.·

psftp> mput code.zip

e. За да свалите файлове от клъстера, използвайте командата *mget*. Файловете ще се свалят в директорията, от която сте стартирали SFTP клиента.

psftp> mget code.zip

6. Компилиране на кода. На клъстера има инсталирани два вида компилатори: GNU и XL. XL компилаторите са оптимизирани за BlueGene/P.
   a. /bgsys/drivers/ppcfloor/comm/default/bin/mpicxx - GNU C++
      *vgancheva@bgpfen:~>/bgsys/drivers/ppcfloor/comm/default/bin/mpicxx –o container *.cpp *.h*
   b. /bgsys/drivers/ppcfloor/comm/default/bin/mpixlc - IBM XL
   c. /bgsys/drivers/ppcfloor/comm/default/bin/mpixlc_r - IBM XL thread safe version
      *vgancheva@bgpfen:~>/bgsys/drivers/ppcfloor/comm/default/bin/mpixlcx -O3 -qarch=450d -qtune=450 -o container *.cpp*
   d. Опцията -qsmp=omp се използва ако искаме да компилираме приложението с поддръжка на OpenMP.
      *vgancheva@bgpfen:~>/bgsys/drivers/ppcfloor/comm/default/bin/mpixlcx -O3 -qarch=450d -qtune=450 -qsmp=omp -o container *.cpp*
7. Събмитване на задача: извършва се с предварително създаден JCF файл.

   # @ job_name = gadget_128
   # @ comment = "This is a Gadget-2 Program"
   # @ error = $(jobid).err
   # @ output = $(jobid).out
   # @ environment = COPY_ALL;
   # @ wall_clock_limit = 20:00:00
   # @ notification = never
   # @ notify_user = never
   # @ job_type = bluegene
   # @ bg_size = 128
   # @ class = n0128
   # @ queue
   time scalasca -analyze /bgsys/drivers/ppcfloor/bin/mpirun -mode VN -np 512 ./Gadget2.galaxy galaxy.param

   Събмитване и контролиране на задачата се извършва със следните три команди
   a. llsubmit <JCF fail> - събмитване на задачата
   b. llcancel <Job ID> - спиране на задачата
   c. llq – показва опашката от задачи
8. Резултатите или възникналата грешка при изпълнението ще са записани в <Job ID>.err и <Job ID>.out файлове в home дирктория Ви.
9. За повече информация прочетете **bluegene-quick-guide.pdf**