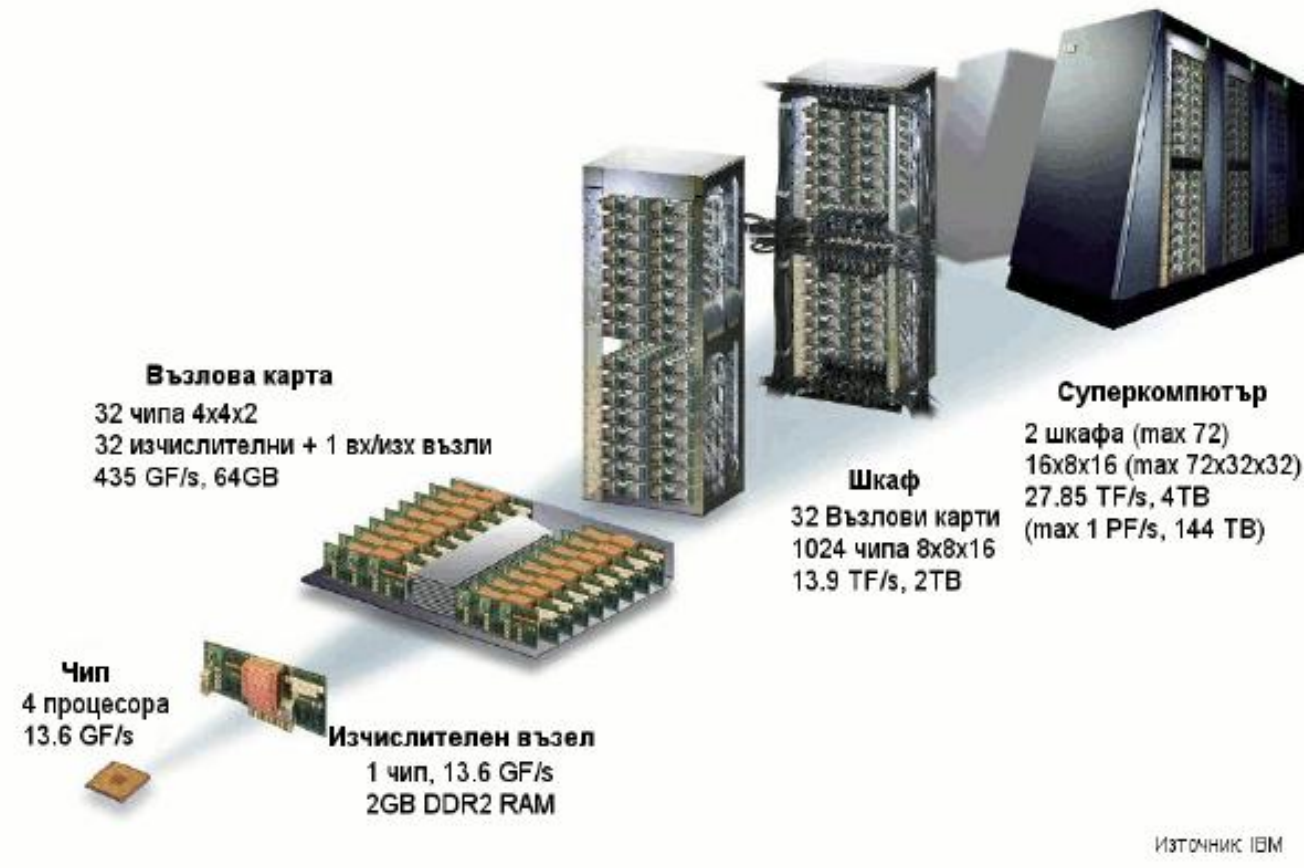
Запознаване и работа със суперкомпютър IBM BlueGene/P.

Пейчо Петков

Валентин Павлов

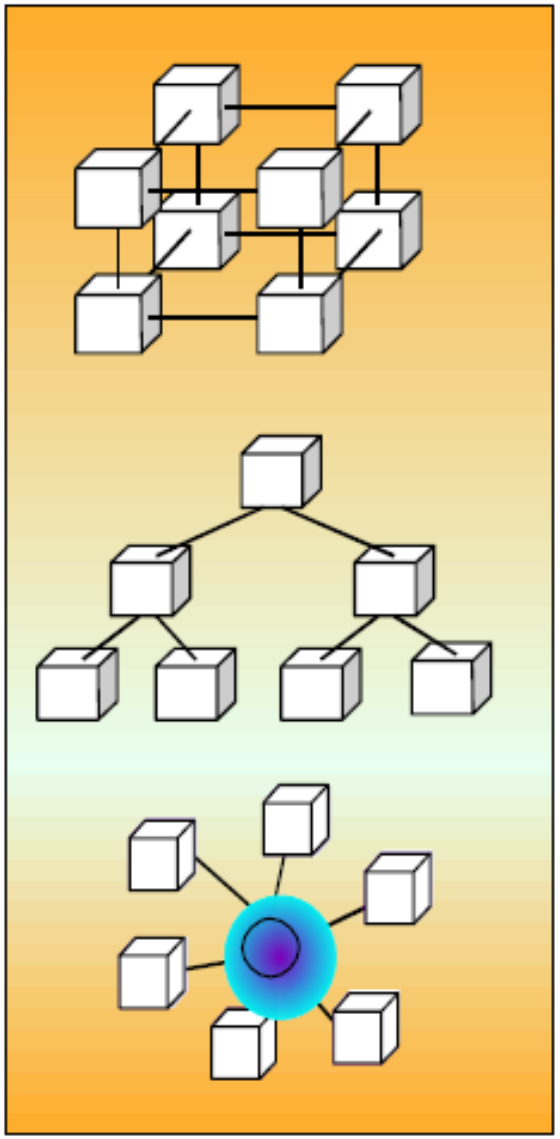
Организация на BlueGene/P



Технически данни

Property		BG/P
Node Properties	Node Processors	4* 450 PowerPC
	Processor Frequency	0.85GHz (target)
	Coherency	SMP
	L1 Cache (private)	32KB/processor
	L2 Cache (private)	14 stream prefetching
	L3 Cache size (shared)	8MB
	Main Store/node	2GB
	Main Store Bandwidth	13.6 GB/s (2*16B wide)
	Peak Performance	13.6 GF/node
Torus Network	Bandwidth	6*2*425MB/s=5.1GB/s
	Hardware Latency (Nearest Neighbor)	160ns (32B packet) 500ns(256B packet)
	Hardware Latency (Worst Case)	5us(64 hops)
Collective Network	Bandwidth	2*0.85GB/s=1.7GB/s
	Hardware Latency (round trip worst case)	4us
System Properties	Peak Performance (72k nodes)	1PF
	Total Power	2.7 MW

Blue Gene/P Interconnection Networks



3 Dimensional Torus

- Interconnects all compute nodes
  - Communications backbone for computations
- Adaptive cut-through hardware routing
- 3.4 Gb/s on all 12 node links (5.1 GB/s per node)
- 0.5 µs latency between nearest neighbors, 5 µs to the farthest
  - MPI: 3 µs latency for one hop, 10 µs to the farthest
- 1.7/2.6 TB/s bisection bandwidth, 188TB/s total bandwidth (72k machine)

Collective Network

- Interconnects all compute and I/O nodes (1152)
- One-to-all broadcast functionality
- Reduction operations functionality
- 6.8 Gb/s of bandwidth per link
- Latency of one way tree traversal 2 µs, MPI 5 µs
- ~62TB/s total binary tree bandwidth (72k machine)

Low Latency Global Barrier and Interrupt

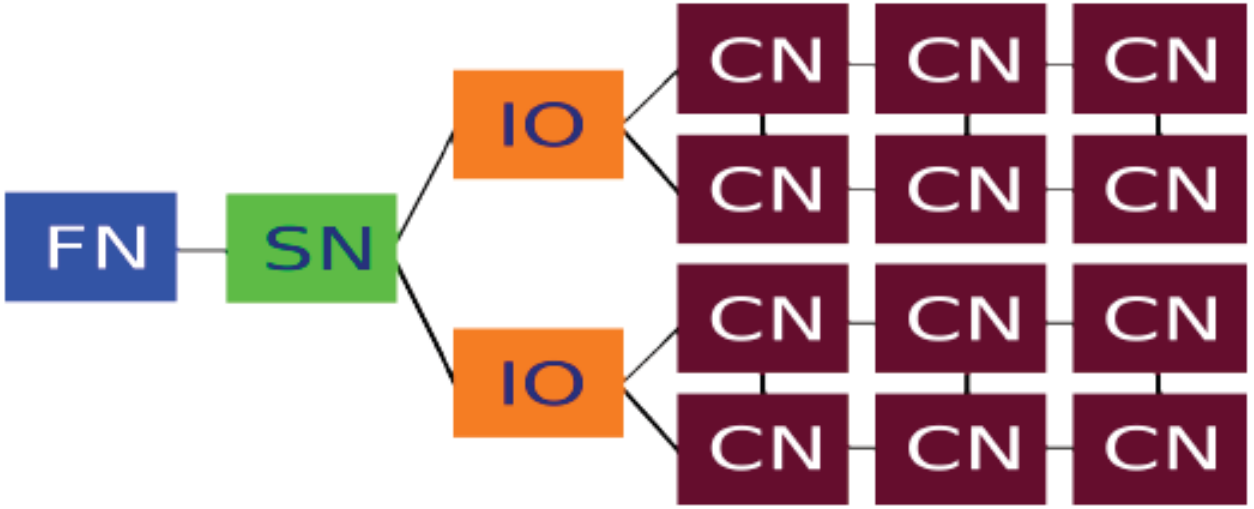
- Latency of one way to reach all 72K nodes 0.65 µs, MPI 1.6 µs

Other networks

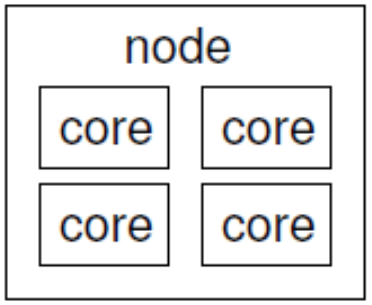
- 10Gb Functional Ethernet
  - I/O nodes only
- 1Gb Private Control Ethernet
  - Provides JTAG access to hardware. Accessible only from Service Node system

Blue Gene Hierarchical Organization

- Front-end nodes - dedicated for user's to login, compile programs, submit jobs, query job status, debug applications
- Compute nodes – run user applications, use simple compute node kernel (CNK) operating system, ship I/O-related system calls to I/O nodes
- I/O nodes – provide a number of Linux/Unix typical services, such as files, sockets, process launching, signals, debugging; run Linux
- Service nodes - perform partitioning, monitoring, synchronization and other system management services. Users do not run on service nodes directly.

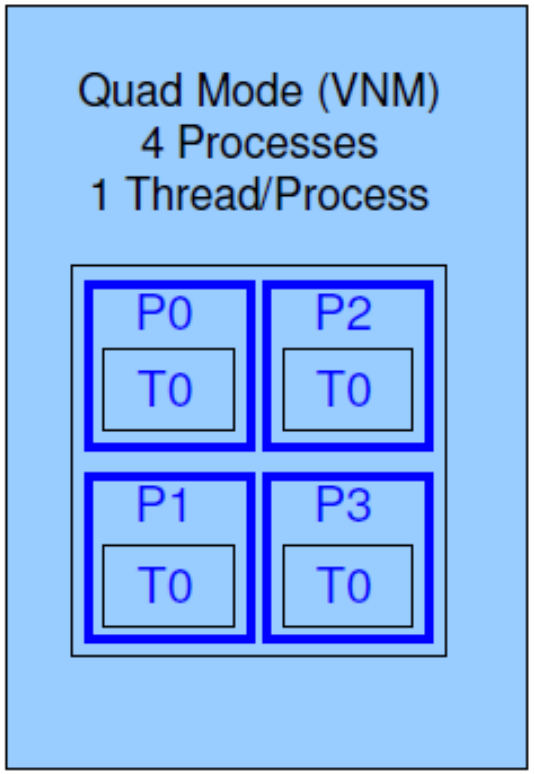


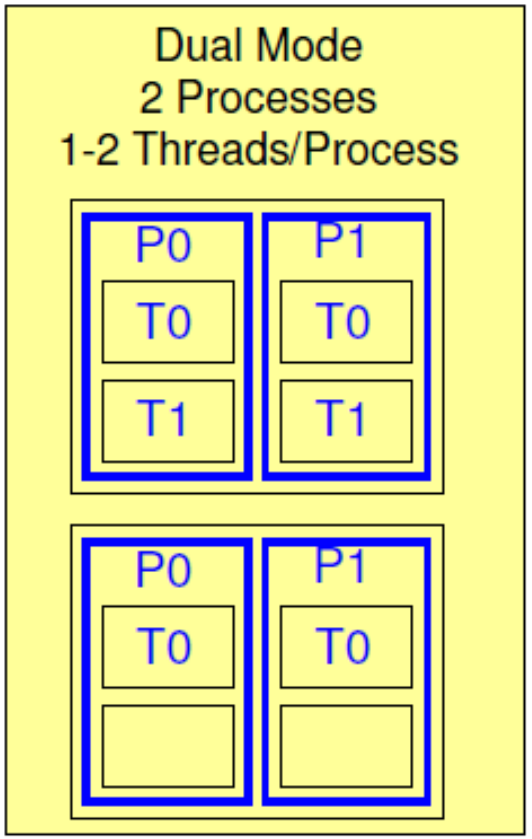
Execution Modes in BG/P per Node

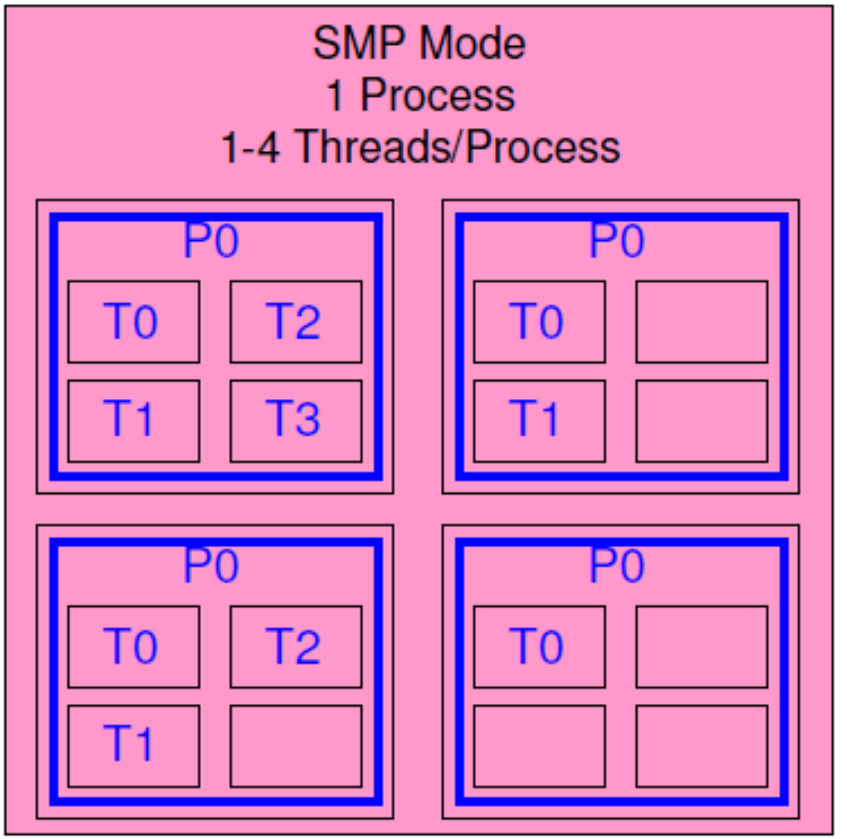


Software Abstractions Blue

- Next Generation HPC
  - **Many Core**
  - **Expensive Memory**
  - **Two-Tiered Programming Model**







Предимства

- Енергийно ефективен
- Заема малко място (висока плътност)
- Прозрачна, високонадеждна и високоскоростна мрежа между възлите
- Програмиране посредством стандартни интерфейси MPI и OpenMP
- Висока скалируемост (хиляди ядра)
- Висока надеждност

BlueGene/P в БСЦ

- Българският суперкомпютърен център (БСЦ) експлоатира и осигурява достъп до суперкомпютър IBM Blue Gene/P, състоящ се от 2048 изчислителни възела (8192 PowerPC ядра, 4TB RAM)
- Връзката на изчислителните възли към останалата част на системата е посредством 16 канала 10 Gb/s
- Производителност: 27.85 Tflops
- Енергийна ефективност: 371.67 Mflops/W

Организация на суперкомпютърния комплекс

Освен шкафовете с изчислителните възли, суперкомпютърната система включва и следните поважни компоненти:

- Front-End Node (bgfen): сървър, до който потребителите имат достъп и от който пускат задачите си. Архитектурата на машината е PowerPC 64, а операционната система – SuSE Linux Enterprise Server 10 (SLES 10);
- Service Node (SN): служебен сървър, който управлява цялостната работа на системата 2 файлови сървъра, посредством които FEN и изчислителните възли достъпват споделен дисков масив с размер 12 TB

Файлови системи

Файловата система на bgfen, освен стандартните Linux места (/etc, /root, /usr, /dev и т.н.) присъстват и специфични места, които представляват интерес за потребителите:

- /shared1 – основната потребителска файлова система с обем 4.4TB. Всеки потребител разполага с поддиректория в /shared1, в която може да се разполага.
- /bgusr – старата основна потребителска файлова система. Home директориите на някои от потребителите са разположени в /bgusr, но те имат и съответна директория в /shared1
- /bgsys – системната директория на BlueGene/P. Там е разположен системния софтуер, компилатори, библиотеки и готов софтуер, компилиран от екипа на центъра.

Предоставяне на достъп

За предоставяне на достъп е необходимо:

1) Наличие на подробно и научно обосновано описание на изследователските дейности
2) Способност да бъде демонстрирана скалируемост на приложението най-малко до 512 ядра.
3) Спазване на Политиката за допустимо използване http://www.scc.acad.bg/index.php?option=com_content&view=article&id=102&Itemid=125&lang=bg
4) Изпратете запитване на bgteam@scc.acad.bg и ще Ви предоставим подробна информация и формулярите, които трябва да попълните
5) Исканията се разглеждат и одобряват от ръководството на БСЦ

Осъществяване на достъпа

При одобрение на Вашата заявка, ще Ви бъде създаден акаунт

- Достъпът до суперкомпютъра е отдалечен и се осъществява посредством SSH протокол
- Името на машината е
  - bg-fen.scc.acad.bg
- Примерна Linux команда за достъп:
  - ssh bg-fen.scc.acad.bg -l username

  (тук, и по-надолу, username е името на акаунта Ви)

- SSH работи на TCP порт 22. Ако имате firewall, вх. и изх. трафик към този адрес/порт трябва да се разреши

Осъществяване на достъпа от Windows

- Примерни клиенти
  - ssh -> PuTTY
  - scp -> WinSCP

Смяна на паролата

При първоначално влизане, системата ще поиска от Вас да си смените паролата:

vpavlov@linux-hoe5:~> ssh bg-fen.scc.acad.bg -l vpavlov

Password:

Password change requested. Choose a new password.

Old Password:

New Password:

Reenter New Password:

Password changed.

Last login: Wed Oct 14 21:31:46 2009 from XXX.XXX.XXX.XXX

** BlueGene/P **

** This internal systems must only be used for conducting **

** IBM business or for purposes authorized by IBM management **

** Use is subject to audit at any time by IBM management. **

vpavlov@bgpfen:~>

- Забележете, че се налага да въведете старата парола два пъти, както и да въведете новата парола два пъти

Копиране на данни

- Данни от и към машината могат да се копират посредством secure shell copy клиент, напр. scp:
  - scp somefile username@bg-fen.scc.acad.bg:
  - scp -r somedir username@bg-fen.scc.acad.bg:
- Във втория пример ключа -r служи за указване на рекурсивно действие, т.е. копиране на всички под-директории и файлове, съдържащи се в somedir

Компилиране

- Компилирането на софтуер (приложения и библиотеки), който да се изпълнява на изчислителните възли, става посредством cross-compiling: компилаторът върви на bgfen, но генерира код за изчислителните възли.
- Системата разполага с 2 комплекта C, C++ и FORTRAN компилатори: GNU Toolchain и IBM XL компилаторите.
- Компилаторите се намират в

```
/bgsys/drivers/ppcfloor/comm/default/bin/
```

- За улеснение, можете да добавите този път в PATH променливата във Вашия `~/.profile` файл, така че компилаторите да са достъпни без изписването му:

```
export PATH=/bgsys/drivers/ppcfloor/comm/default/bin/:$PATH
```

GNU Toolchain

- C компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpicc
```

- C++ компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpicxx
```

- FORTRAN-77 компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpif77
```

- FORTRAN-90 компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpif90
```

IBM XL компилатори

- C/C++ компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlc
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlc_r
```

- FORTRAN-77 компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlf77
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlf77_r
```

- FORTRAN-90 компилатор:

```
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlf90
/bgsys/drivers/ppcfloor/comm/default/bin/mpixlf90_r
```

- Версиите, които завършват на `_r`, генерират thread-safe код

Компилиране на софтуер

- Изходния код на приложение, което използва MPI или OpenMP може да се компилира на BlueGene/P, като се укаже компилатора, с който ще се използва в Makefile-а или в configure скрипта (или по друг начин, в зависимост от самото приложение)
- Най-често за целта се установяват променливите от обкръжението `CC`, `FC`, `F77` и `CXX`
- Например:

```
CC=mpixlc
```

Компилиране на софтуер (прод.)

- Когато се използват IBM XL компилаторите, могат да се укажат опции, които да доведат до по-оптимален код специално за BlueGene/P. Това става чрез указването на следната последователност от опции на компилатора:

```
CFLAGS="-O3 -qarch=450d -qtune=450"
```

Тези опции важат и за C++ и FORTRAN, като обикновено се установяват `CXXFLAGS` и `FCFLAGS`

Пример

- В /bgsys/local/samples/helloworld се намира примерна програма, Makefile и LoadLeveler JCF файл, които могат да се използват като пример за компилация и пускане на задача:

```
cd /bgsys/local/samples/helloworld
make
```

- Тези команди създават файл `hello`, който може да се изпълни върху BlueGene/P
- Можете да използвате примерните файлове като шаблони за създаването на Ваши скриптове за компилация и изпълнение

Изпълнение на задачите

- За разпределението на задачите се грижи Tivolli Workload Scheduler LoadLeveler
- Подготвените задачи се пускат за изпълнение с командата `llsubmit`, която получава т.н. Job Control File, който описва изпълнимия файл, обкръжението и аргументите му.
- Изпълнението на `llsubmit` води до поставянето на задачата в опашка от чакащи задачи. Когато се появи възможност за изпълнение, задачата се изпраща на BlueGene/P.
- Списъка с чакащи задачи може да се види с командата `llq`. Ако статуса на задачата Ви е **R**, тя е пусната, ако е **I** – изчаква, а ако е **H**, значи е възникнал някакъв проблем и трябва да се премахне. Премахването става с командата `llcancel`.

Съдържание на JCF

- /bgusr/local/samples/helloworld.jcf е примерен Job Control File:

```
# @ job_name = hello
# @ comment = "This is a HelloWorld program"
# @ error = $(jobid).err
# @ output = $(jobid).out
# @ environment = COPY_ALL;
# @ wall_clock_limit = 01:00:00
# @ notification = never
# @ job_type = bluegene
# @ bg_size = 128
# @ class = n0128
# @ queue
/bgsys/drivers/ppcfloor/bin/mpirun -exe hello -verbose 1 -mode VN -np 512
```

- Следната последователност от команди води до изпращане на задачата за изпълнение:

```
cd /bgsys/local/samples/helloworld

llsubmit hello.jcf
```

Параметри в JCF

- `# @ job_name = hello` Произволно име на задачата
- `# @ comment = "This is a HelloWorld program"` Произволен коментар, за собствена употреба
- `# @ error = $(jobid).err` Име на файл, в който се пренасочва стандартния файлов дескриптор stderr. Демонстрирана е употребата на $(jobid), променлива, която динамично се попълва с идентификатора на задачата. Напр., ако LoadLeveler даде на задачата идентификация 4242, то стандартния дескриптор stderr ще бъде пренасочен във файла 4242.out
- `# @ output = $(jobid).out` Име на файл, в който се пренасочва стандартния файлов дескриптор stdout
- `# @ environment = COPY_ALL;` Указва, че всички валидни по време на изпълнението на llsubmit променливи от обкръжението трябва да се установят и при пускането на задачата на изпълнителните възли

Параметри в JCF (прод.)

- `# @ wall_clock_limit = 01:00:00` Времева граница, след изтичането на която LoadLeveler ще прекрати изпълнението на задачата. Тази граница не може да надвишава установената за класа задачи граница (виж по-долу)
- `# @ notification = never` Тъй като не е изградена инфраструктура за изпращане и получаване на поща, тук се записва never.
- `# @ job_type = bluegene` Този параметър трябва да съдържа стойността bluegene
- `# @ bg_size = 128` 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 (виж по-долу)

Параметри в JCF (прод.)

- `# @ class = n0128` Клас на задачата. Най-важният от параметрите, определя приоритета, с който ще бъде пусната задачата, максималния брой изчислителни възли, максималното време за изпълнение (виж по-долу информация за класовете и съответните ограничения)
- `# @ queue` Инструктира LoadLeveler да сложи задачата в опашката
- `/bgsys/drivers/ppcfloor/bin/mpirun -exe hello -verbose 1 -mode VN -np 512` Същинската команда, която води до изпращането на задачата до изчислителните възли на BlueGene/P.
- Някои от параметрите й са:
- `-exe <executable_file>` – указва файла, който ще се изпълнява
- `-args "<arguments>"` – аргументи, които ще бъдат подадени на изпълнимия файл
- `-verbose 1` – инструктира mpirun да изписва подробна информация за процеса на пускане на задачата

Параметри в JCF (прод.)

- `-mode VN|SMP|DUAL` – указва режима на изпълнение на задачата (виж по-долу)
- `-np N` – указва броя процесори, върху които ще се стартира програмата
- `- env BG_MAXALIGNEXP=-1` – **много важен** (недокументиран!) параметър, който указва на BlueGene/P да НЕ изисква data alignment. Голяма част от готовия софтуер не е предвиден да работи на системи, които изискват data alignment и съответно това води до невъзможност за изпълнението им на BG/P. Този аргумент кара системата да не обръща внимание на не-подравнените данни и съответно позволява изпълнението на такъв софтуер, въпреки, че това води до намалена производителност.

Note: *Alignment is a property of a memory address, expressed as the numeric address modulo a power of 2.*

Режими на изпълнение

- Всеки от изчислителните възли разполага с 4 отделни PowerPC 450 процесора и обща памет от 2GB
- Изчислителните възли могат да работят в 3 различни режима на изпълнение: VN, DUAL и SMP
- **Режим VN:** изчислителният възел се разделя на 4 отделни процесора. Всеки процесор изпълнява едно копие на програмата, като тя не може да ползва нишки. Паметта се разделя на 4 блока, по 1 за всеки процесор. В този режим 128 възела изпълняват 512 копия на програмата, а цялата система – 8192 копия. Всяко от копията има достъп до 512MB памет.

Режими на изпълнение (прод.)

- **Режим DUAL:** изчислителният възел се разделя на 2 двойки процесори. Всяка двойка изпълнява едно копие на програмата, като тя може да пусне 2 нишки, всяка от които се изпълнява на съответния процесор в рамките на двойката. Паметта се разделя на 2 блока, по 1 за всяка двойка. В този режим 128 възела изпълняват 256 копия на програмата, а цялата система – 4096 копия. Всяко от копията има достъп до 1GB памет и може да пусне по 2 нишки.

Режими на изпълнение (прод.)

- **Режим SMP:** изчислителният възел не се разделя. Всеки възел изпълнява едно копие на програмата, като тя може да пусне 4 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 2GB памет и може да пусне по 4 нишки.

Режими, процесори и bg_size

- За да бъде коректно зададена задачата, трябва да бъде изпълнено следното неравенство:

bg_size >= np / km

- **bg_size** е стойността, указана в LoadLeveler директивата `# @ bg_size`
- **np** е стойността, указана в `-np` аргумента на `mpirun` (броя копия на програмата)
- **km** е коефициент на режима: 1 за SMP, 2 за DUAL, 4 за VN
- Освен това, bg_size трябва да е кратно на 128 и да отговаря на указания клас на задачата

Примери за bg_size

- `-mode VN -np 400` : bg_size = 128
- `-mode VN -np 600` : bg_size = 256
- `-mode DUAL -np 400` : bg_size = 256
- `-mode DUAL -np 600` : bg_size = 512
- `-mode SMP -np 400` : bg_size = 512
- `-mode SMP -np 600` : bg_size = 1024

Геометрия на задачата

- Един шкаф се дели на две полуравнини (Midplane) – горна и долна. Всяка полуравнина съдържа 512 изчислителни възела
- **Когато bg_size >= 512** може да се укаже геометрията на възлите, които изпълняват задачата. Това може да повиши бързодействието (напр. при GROMACS). Това става чрез следните LoadLeveler директиви:

Геометрия на задачата (прод.)

- # @ bg_connection = MESH | TORUS | PREFER_TORUS
  - MESH е нормалният режим. Възлите са свързани в куб (всеки със съседните 6). Съобщение от първия до последния възел преминава транзитно през всички останали по пътя между тях, което води до известна латентност
  - TORUS – В допълнение на MESH, по всяко измерение първите възли са свързани с последните. Така съобщение от първия до последния възел преминава директно (т.е. средния път на съобщенията се съкръщава двойно.
  - PREFER_TORUS – Ако е възможно, геометрията е TORUS, ако не – MESH

Геометрия на задачата (прод.)

- # @ bg_shape = XxYxZ – съответно X, Y и Z бр. Полуравнини в съответните посоки, или пермутация между тях (виж и bg_rotate)
- # @ bg_rotate = True | False – във връзка с bg_shape, разрешава или забранява пермутациите.
- За нашата система тези директиви имат смисъл само за случая bg_size = 1024 (при bg_size = 512 полуравнината е само 1, а при bg_size = 2048 се използват и 4-те полуравнини на 2-та шкафа, с които разполагаме), при това без пермутация (bg_rotate = False). Тогава bg_shape = 1x2x1 означава, че двете полуравнини ще бъдат в 1 шкаф, а bg_shape = 2x1x1 – че ще бъдат в различни шкафове.

Класове задачи

- Класът на задачата определя:
  - Приоритизацията – по-големите задачи се пускат с предимство пред по-малките
  - Максималния брой възли, които могат да се използват
  - Максималното време, което задачата може да се изпълнява
- Едновременно могат да бъдат изпълнявани определен брой задачи от даден клас
- Дефинирани са следните класове:

Класове задачи (прод.)

Клас	Брой задачи	Макс. Брой възли	Макс. 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

n2048 – специално отношение

- **Внимание:** Задачи от клас n2048, които заемат цялата машина, се пускат само със знанието на Администратора. Ако имате нужда да пуснете такава задача, моля пишете на bgteam@scc.acad.bg.
Злоупотреба и неспазване на това правило може да доведе до блокиране на акаута Ви!

Advanced simulations and Software

Prof. Stoyan Markov
Dr. Peicho Petkov

Areas covered by the software and libraries installed on the IBM BlueGene/P at Bulgarian National Supercomputing Center

- Computional chemistry, biochemistry, pharmacy and material science.
- Molecular dynamics , ab initio molecular dynamics;
- Structural bioinformatics, protein folding and 3D structures predictions, mutant 3D protein structure deformations, high resolution refinement, protein-protein docking;
- Virtual screening and computer aided drug designs;

Areas covered by the software and libraries installed on the IBM BlueGene/P at Bulgarian National Supercomputing Center (II)

- Computational fluid dynamics, large eddy simulations, nuclear reactor cooling simulations, gas combustions, coal combustions, pulverized coal furnaces (optimization, slagging, pollutants), semi-transparent radiation heat transfer, lagrangian modeling for multi-phase flows;
- Seismic wave propagation simulations, object impact and hazard risk calculations;
- Preconditioning Techniques for Large Linear Systems, Solving large sparse linear systems;
- Multi-scale linear solvers for very large linear systems etc.;

GROMACS

(GROningen Machine for Chemical Simulations)

- Primarily designed for biochemical molecules (proteins, lipids and nucleic acids) - a lot of complicated bonded interactions;
- Extremely fast at calculating the nonbonded interactions (that usually dominate simulations) – research on non-biological systems, e.g. polymers.
- Advantages:
  - Domain decomposition – particles;
  - PME 2D decomposition – long range electrostatics;
  - Leapfrog integration algorithm;
  - Variety of simulation box shapes;

http://www.gromacs.org

NAMD

(Not (just) Another Molecular Dynamics program)

- A parallel, object-oriented molecular dynamics code designed for high-performance simulation of large biomolecular systems.
- Developed using Charm++ - adaptive communication-computation overlap and dynamic load balancing.
- NAMD pioneered the use of hybrid spatial and force decomposition.
- Scales to thousands of processors. NAMD is tested up to 64,000 processors.
- Simulation preparation and analysis is integrated into the visualization package VMD.

LAMMPS
(Large scale Atomic / Molecular Massively Parallel Simulator)

- A classical molecular dynamics simulation code designed to run efficiently on parallel computers developed at Sandia National Laboratories.
- Integrates Newton's equations of motion for collections of atoms, molecules, or macroscopic particles that interact via short- or long- range forces with a variety of initial and/or boundary conditions.
- Uses spatial decomposition techniques to partition the simulation domain into small 3d sub domains.
- Most efficient (in a parallel sense) for systems whose particles fill a 3d rectangular box with roughly uniform density.

DL POLY 4

- DL_POLY is a general purpose classical molecular dynamics (MD) simulation software developed at Daresbury Laboratory by I.T. Todorov and W. Smith.
- DL_POLY_4 is based on the Domain Decomposition (DD) strategy and is best suited for large molecular simulations from 10^3 to 10^9 atoms on large processor counts.
- DL POLY 4 offers a selection of MD integration algorithms couched in both Velocity Verlet (VV) and Leapfrog Verlet (LFV) manner
- It is relatively easy to adapt DL POLY 4 to user specific force fields.

CP2K

(Car-Parrinello molecular dynamics 2000)

- A suite of modules:
  - variety of molecular simulation methods at different levels of accuracy, from ab-initio DFT to classical Hamiltonians, passing through semi-empirical NDDO approximation.
- Used for:
  - predicting energies, molecular structures, vibrational frequencies of molecular systems, reaction mechanisms;
- Ideally suited for performing molecular dynamics studies.
- Performs atomistic and molecular simulations of solid state, liquid, molecular, biological systems.

CPMD

(Car-Parrinello Molecular Dynamics)

- Ab Initio Electronic Structure and Molecular Dynamics Program.
- Includes ultrasoft pseudopotentials, free energy density functional, wavefunction optimization, geometry optimization,molecular dynamics:
  - constant energy, constant temperature and constant pressure, path integral MD, many electronic properties, time-dependent DFT,coarse-grained non-Markovian metadynamics, response functions and many electronic structure properties,hybrid quantum mechanical / molecular dynamics

NWChem 5.1
(Northwest Chemistry)

- Designed for parallel computer systems including parallel supercomputers and large distributed clusters.
- Scalable:
  - ability to treat large problems efficiently
  - utilization of available parallel computing resources.
- Molecular calculations including:
  - density functional, Hartree-Fock, Müller-Plesset, coupled-cluster, configuration interaction;
  - molecular dynamics, mixed quantum mechanics, geometry optimizations, vibrational frequencies, relativistic corrections;
  - ab-initio molecular dynamics;
  - extended DFT
- Periodic system modeling;

Quantum Espresso

Quantum Espresso can currently perform the following kinds of calculations:

- Ground-state energy and one-electron (Kohn-Sham) orbital's;
- Atomic forces, stresses, and structural optimization;
- Molecular dynamics on the ground-state Born-Oppenheimer surface, also with variable cell;
- Nudged Elastic Band (NEB) and Fourier String Method Dynamics (SMD) for energy barriers and reaction paths;
- Macroscopic polarization and finite electric fields via the modern theory of polarization (Berry Phases).

GAMESS

(General Atomic and Molecular Electronic Structure System)

- General purpose electronic structure code
- Primary focus is on ab initio quantum chemistry calculations
- Density functional theory calculations
- QM/MM calculations
- Energy-related properties
- Numerical Hessians from finite differences of analytic gradients
- Molecular dynamics Effective fragment potential (EFP) method

DALTON

- Powerful molecular electronic structure program, with an extensive functional for the calculation of molecular properties at the HF, DFT, MCSCF, and CC levels of theory.

Qbox

- First-principles molecular dynamics (FPMD) – atomistic simulation method that combines an accurate description of electronic structure with the capability to describe dynamical properties by means of molecular dynamics (MD) simulations.
- Specific attention paid to the requirement to distribute nearly all data structures on a platform as large as the Blue Gene/P platform.

ROSETTA 3

(High - Resolution Protein Structure Prediction Codes)

- Library based object-oriented software suite which provides a robust system for predicting and designing protein structures, protein folding mechanisms, and protein-protein interactions.
- Rosetta Functionality Summary: RosettaAbinitio, RosettaDesign, RosettaEnzymeDesign, RosettaDock, RosettaAntibody, RosettaFragments, RosettaNMR, RosettaDNA, RosettaRNA, RosettaLigand

mpiBLAST and ScalaBLAST

- BLAST (Basic Local Alignment Search Tool) – the tool most frequently used for calculating sequence similarity.
  - Search large databases.
  - Comparison of nucleotide or protein sequences from the same or different organisms is a very powerful tool in molecular biology.
  - Finding similarities between sequences, scientists can infer the function of newly sequenced genes, predict new members of gene families, and explore evolutionary relationships.
- mpiBLAST is a parallel implementation of the Blast algorithm.
- ScalaBLAST: A Scalable Implementation of BLAST for High-Performance Data-Intensive Bioinformatics Analysis. A typical query list might contain thousands or millions of individual sequences, each of which is meant to be scored against a large database of publicly available sequence information, such as the non-redundant protein sequence database (nr).

SPECFEM3D
(seismic wave propagation)

- Unstructured hexahedral mesh generation is a critical part of the modeling process in the Spectral-Element Method (SEM).
- An advanced 3D unstructured hexahedral mesh generator that offers new opportunities for seismologist to design, assess, and improve the quality of a mesh in terms of both geometrical and numerical accuracy.
- The main goal is to provide useful tools for understanding seismic phenomena due to surface topography and subsurface structures such as low wave-speed sedimentary basins.

Code Saturne / Syrthes1.3.2

- **Code Saturne**:
  - Solve the Navier-Stokes equations in the cases of 2D, 2D axis symmetric or 3D flows.
  - The main module is designed for the simulation of flows which may be steady or unsteady, laminar or turbulent, incompressible or potentially dilatable, isothermal or not. Scalars and turbulent fluctuations of scalars can be taken into account.
  - Includes specific modules for the treatment of: lagrangian particle tracking, semi-transparent radioactive transfer, gas, pulverized coal and heavy fuel oil combustion, electricity effects (Joule effect and electric arcs) and compressible flows.
  - Engineering module (Matisse) - simulation of nuclear waste surface storage.

- **Syrthes –** conjugate heat transfer and transparent radiative heat transfer, Independent FE solver with tetrahedral mesh and arbitrary fluid-solid interface, Thermal shock, striping, fatigue Fuel combustion, ionic mobility under development.

Aster
(Structural mechanics code)

- A general code directed at the study of the mechanical behaviour of structures.
- For the expertise and the maintenance of power plants and electrical networks
- The main range of application is deformable solids: explains the great number of functionalities related to mechanical phenomena.
  - behaviour of industrial components – influence of physical phenomena (internal or external fluids, temperature, metallurgic phase changes, electro-magnetic stresses ...).
- Can « link » mechanical phenomena and thermal and acoustic phenomena together.
- Provides a link to external software, and includes a coupled thermo-hydro-mechanics kit.

Code Salome

- A generic platform for pre and post processing and code coupling for numerical simulation with the following aims:
  - Supports interoperability between CAD modeling and computation software;
  - Facilitate implementation of coupling between computing codes in a distributed environment;
  - Makes easier the integration of new components into heterogeneous systems for numerical computation;
  - Sets the priority to multi-physics coupling between computation software;
  - Pool production of developments (pre and post processors, calculation distribution and supervision) in the field of numerical simulation

Available libraries

- ✓ PETCs
- ✓ ParMETIS
- ✓ LAPACK
- ✓ Linpack
- ✓ ScalaPAck
- ✓ SuperLU
- ✓ MUMS
- ✓ HYPRE
- ✓ ParFE
- ✓ GotoBLAS
- ✓ FFTW
- ✓ GlobalArray
- ✓ PVFS2
- ✓ LUSTRE
- ✓ Trilinos

Local Infrastructure and work flow

BlueGene/P
FEN

Pre and Post
Data Processing

Linux Clusters

User Workstations

Conclusions

- ***More than 16 software packages*** and ***more than 10 libraries*** are installed on the IBM BlueGene/P at the National Supercomputing Center
- The software installed covers scientific areas like life science, computational chemistry, material science, environmental science, computational fluid dynamics, transfer simulations, etc.