

*Архитектура и программное  
обеспечение вычислительной системы  
IBM BlueGene/P*

*Н.Н.Попова,*  
доцент факультета ВМК МГУ

Международная летняя суперкомпьютерная академия, Москва,  
МГУ, 25 июня - 7 июля 2012 г

# План

- Обзор архитектуры вычислительной системы BlueGene/P

<http://hpc.cs.msu.su>

- Программное обеспечение Blue Gene/P
- Запуск и мониторинг задач
- Основной шаблон работы пользователя на Blue Gene/P

# Проект Blue Gene

## ▪ **Blue Gene/L**

- Начинаясь как массивно-параллельный компьютер для изучения фолдинга белков
- Первый прототип - 2004 г. Первая строка в Top 500 с производительностью в 70.72 Тфлоп/с
- 2-х ядерный чип

## ▪ **Blue Gene/P**

- Продолжение линейки Blue Gene
- Увеличена частота процессора и объем памяти
- 4-х ядерный чип (технология system-on-a-chip)
- Самая большая система на основе Blue Gene/P установлена в Германии (JUGENE)
  - 1 Пфлоп/с пиковая, 825 Тфлоп/с реальная

## ▪ **Blue Gene/Q**

- 2012 год, производительность ~20 Пфлоп/с
- 8-ядерный чип

# Топ500, июнь 2012

- Blue Gene/P

7 entries.

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
25	Forschungszentrum Juelich (FZJ) Germany	JUGENE - Blue Gene/P Solution IBM	294912	825.5	1002.7	2288
38	DOE/SC/Argonne National Laboratory United States	Intrepid - Blue Gene/P Solution IBM	183840	458.6	557.1	1280
42	DOE/NNSA/LLNL United States	Dawn - Blue Gene/P Solution IBM	147456	415.7	501.4	1134
85	King Abdullah University of Science and Technology Saudi Arabia	Shaheen - Blue Gene/P Solution IBM	65536	190.9	222.8	504
149	IDRIS France	Blue Gene/P Solution IBM	40960	119.3	139.3	315
201	EDF R&D France	Frontier2 BG/L - Blue Gene/P Solution IBM	32768	95.5	111.4	252
377	Rice University United States	BlueGene/P, PowerPC 450 4C 850 MHz, Proprietary IBM	24576	71.4	83.6	189

Международная летняя суперкомпьютерная академия, Москва,  
МГУ, 25 июня - 7 июля 2012 г

# Топ500, июнь 2012

- Blue Gene/Q

20 entries.

Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	DOE/NNSA/LLNL United States	Sequoia - BlueGene/Q, Power BQC 16C 1.60 GHz, Custom IBM	1572864	16324.8	20132.7	7890
3	DOE/SC/Argonne National Laboratory United States	Mira - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	788432	8162.4	10066.3	3945
7	CINECA Italy	Fermi - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	163840	1725.5	2097.2	821.9
8	Forschungszentrum Juelich (FZJ) Germany	JuQUEEN - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	131072	1380.4	1677.7	657.5
13	Science and Technology Facilities Council - Daresbury Laboratory United Kingdom	Blue Joule - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	114688	1207.8	1468.0	575.3
20	University of Edinburgh United Kingdom	DiRAC - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	98304	1035.3	1258.3	493.1
29	EDF R&D France	Zumbrota - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	65536	690.2	838.9	328.8
30	IDRIS/GENCI France	BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	65536	690.2	838.9	328.8
31	Victorian Life Sciences Computation Initiative Australia	Avoca - BlueGene/Q, Power BQC 16C 1.60GHz, Custom IBM	65536	690.2	838.9	328.8

Международная летняя суперкомпьютерная академия, Москва, МГУ, 25 июня - 7 июля 2012 г

# Общая характеристика систем Blue Gene

- Массивно-параллельные системы с распределенной памятью
- Технология System-on-chip (4 ядра, 8 FPU, контроллер памяти и др. на одном ASIC)
- Высокая плотность упаковки
  - процессоры с низким энергопотреблением
- Высокопроизводительный интерконект
  - несколько коммуникационных подсистем для различных целей
- Ультра легкая ОС
  - выполнение вычислений и ничего лишнего
- Стандартное ПО
  - Fortran/C/C++ и MPI

# Blue Gene/P Hardware



**Rack**  
Cabled 8x8x16

**Rack**  
32 Node Cards

**System**  
Up to 72 Racks

**1 PF/s**  
**144 TB**

**Node Card**  
32 Chips, 4x4x2  
(32 compute, 4 I/O cards)

**Compute Card**  
1 Chip,  
1x1x1

**Chip**  
4 processors

**13.6 GF/s**  
2 or 4 GB DDR  
**8 MB EDRAM**

**435 GF/s**  
**64 or 128 GB**

**14 TF/s**  
**2 or 4 TB**



# Blue Gene P

## 1 стойка

- 1024 четырехъядерных вычислительных узлов
- производительность одного вычислительного узла – 13.6 GF/s
- производительность 1 стойки– 13.9 Tflops
- оперативная память одного узла – 2 GB
- суммарная оперативная память в стойке– 2 TB
- узлов ввода/вывода 8 – 64
- Размеры - 1.22 x 0.96 x 1.96
- занимаемая площадь 1.17 кв.м.
- энергопотребление (1 стойка) - 40 kW (max)



# Конфигурация BlueGene/P факультета ВМиК

<http://hpc.cs.msu.ru>

- пиковая производительность  
27.8 Tflop/s
- 2 стойки
- 2048 4-ех ядерных узлов
- общий объем ОЗУ 4 ТВ



# Компоненты Blue Gene P

- Основная единица – четырехядерный вычислительный узел (процессор) , ядро – PowerPC 450 850Mhz + память (2GB)
- Node card = 32 вычислительных узла + до 2х узлов ввода-вывода
- Стойка – 32 node cards
- Число процессоров в стойке
- Итоговое число ядер на стойку - 4096

# Power PC 450 процессор

- Развитие процессора PPC440
- 32-bit архитектура, 850 MHz
- integer unit
- 
- load/store unit
- Специальное устройство double floating-point unit (dfpu)

# Power PC 450

## процессор

- L1 Data cache:
  - 32 KB общий размер, 32-Byte строки,
  - 64-way associative, round-robin замещение
  - 4 ядра – L1 cache coherent
- L2 Data cache : prefetchbuffer, 16 128-byte строк
- L3 Data cache : 8 MB
- ОЗУ : 2 GB DDR, ~13.6GB/s пропускная способность
  - Double FPU: 32 первичных floating-point регистров, 32 secondary floating-point registers, SIMD инструкции для 64-битовой вещественной арифметики (fpadd, fpmadd, fpre, ...)
- Конвейер для вещественной арифметики : 5 циклов

# BGP Node Card

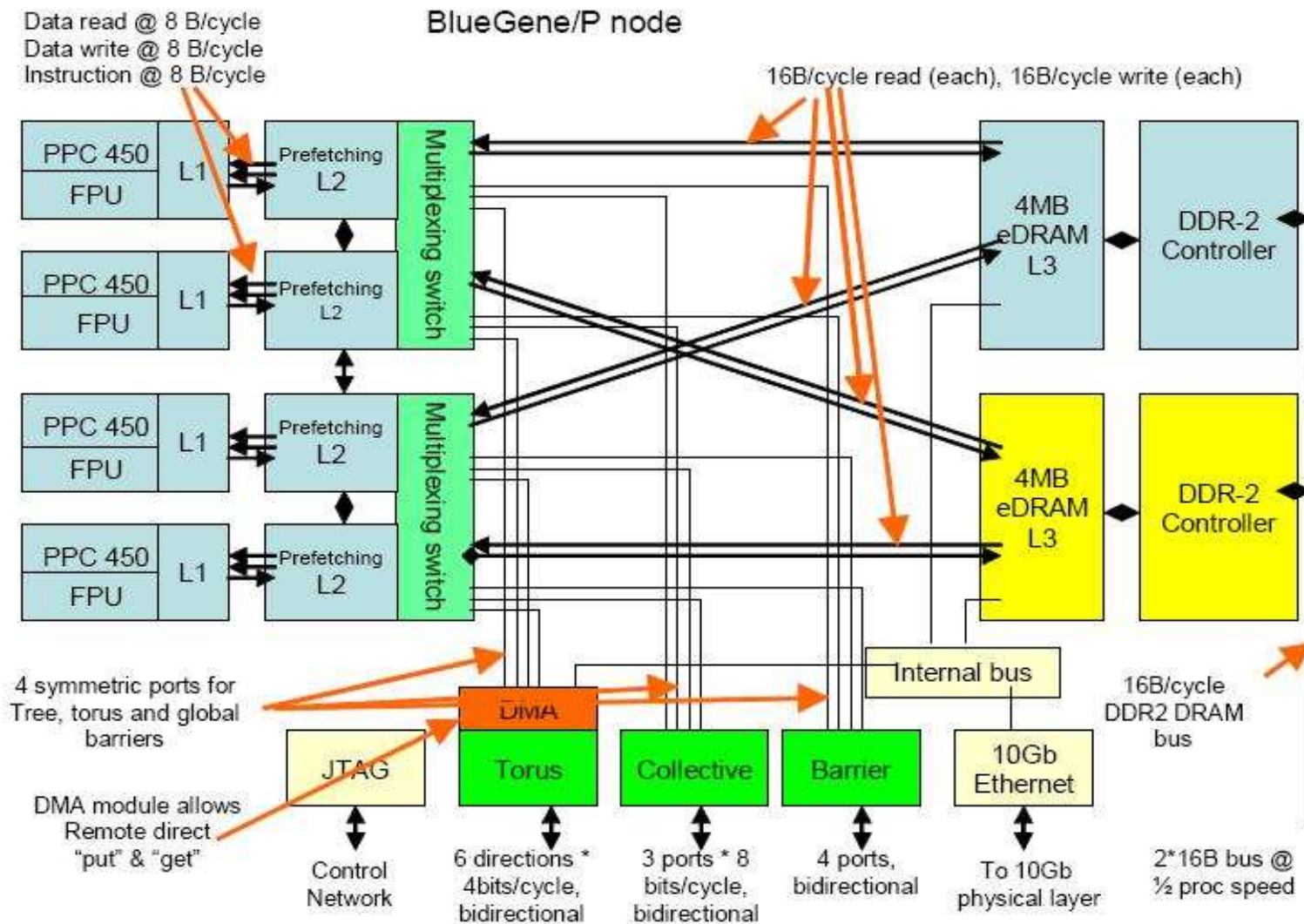
32 Compute nodes



Optional IO card  
(one of 2 possible)  
with 10Gb optical link

Local DC-DC  
regulators  
(6 required, 8 with  
redundancy)

# Вычислительный узел



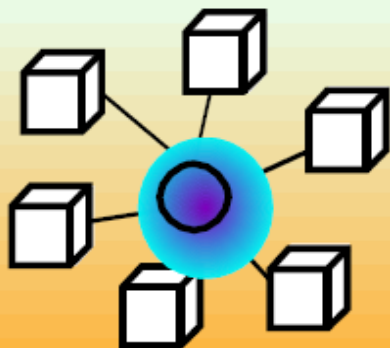
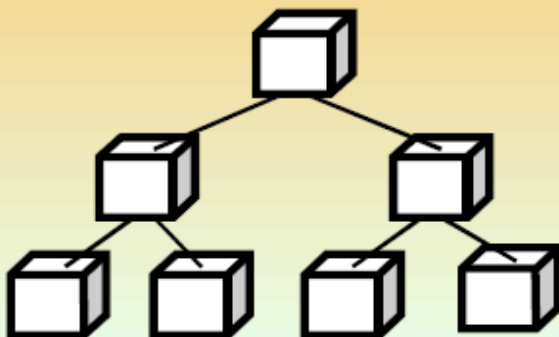
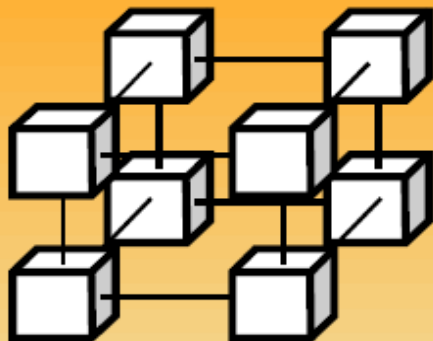
# Компоненты Blue Gene P

- Помимо вычислительных стоек, в состав системы также входят:
  - модуль управления системой (сервисный узел)
  - не менее одного узла front end (через них осуществляется доступ пользователей к системе)
  - сеть, связывающая компоненты системы
  - специализированная сеть для сообщения между сервисным узлом и узлами ввода-вывода

# Коммуникационные сети BGP

- Каждый вычислительный узел подключается к нескольким сетям:
  - 6 выходов в сеть, объединяющую вычислительные узлы в трёхмерный тор
  - 3 выхода в сеть для обмена коллективными сообщениями и связи с узлами ввода-вывода
  - 4 выхода в сеть для обмена прерываниями
  - 1 выход в управляющую сеть





- 3-мерный тор
  - Виртуальная аппаратная маршрутизация без буферизации
  - 3.4 Гбит/с на всех 12 портах (5.1 ГБ/с на узле)
  - Аппаратные задержки: 0.5 мс между соседними узлами, 5 мс между самыми далекими
  - Задержки MPI: 3мс между соседними узлами, 10 мс между самыми далекими
  - Основная коммуникационная сеть
    - Используется в том числе для многих коллективных операций
- Коллективная сеть – дерево
  - Для глобальной коммуникации один-ко-всем (broadcast, reduction)
  - 6.8 ГБ/с на порт
  - Задержка на один проход дерева: аппаратная 1.3 мс, MPI 5 мс
  - Соединяет все вычислительные узлы и узлы ввода-вывода
  - Используется для коллективных операций и коммутатора MPI\_COMM\_WORLD
- Высокоскоростная сеть для глобальных прерываний
  - Задержка на оповещение всех узлов: аппаратная 0.65 мс, MPI 5 мс
  - Для MPI\_Barrier

# Процессоры ввода-вывода

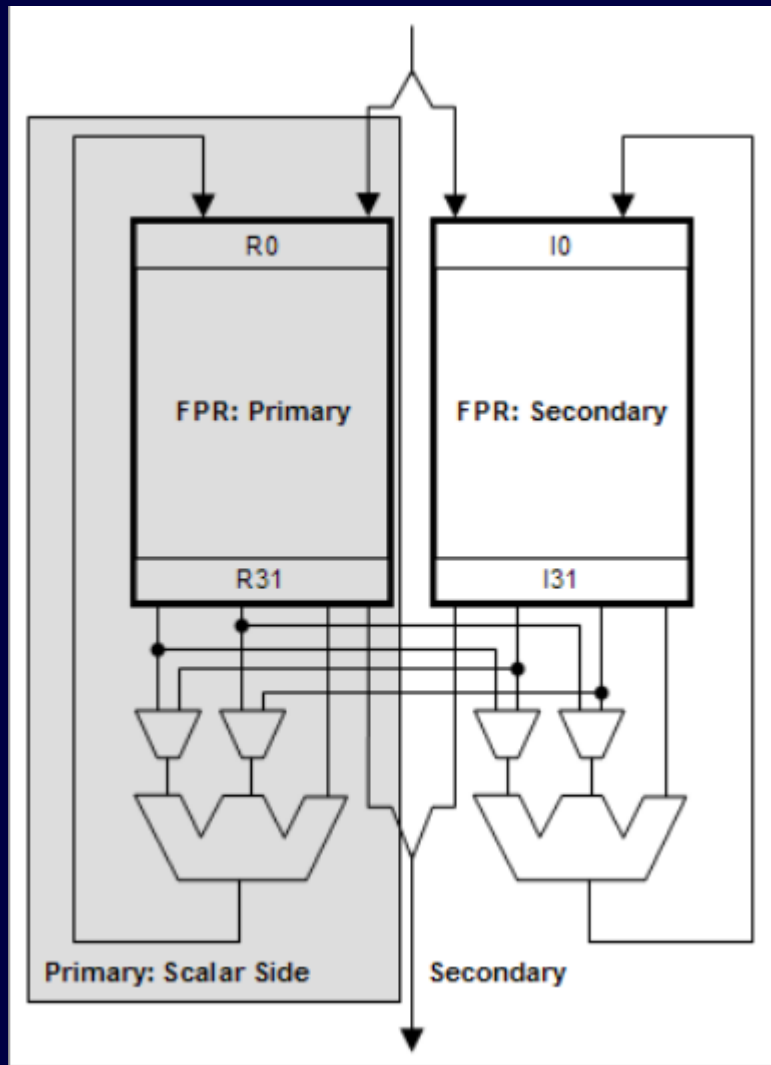
Отличия по сравнению с вычислительным узлом:

- Установлена полноценная ОС
- Отсутствует подключение к сети тору
- Имеется выход в 10-гигабитную сеть Ethernet

# Память

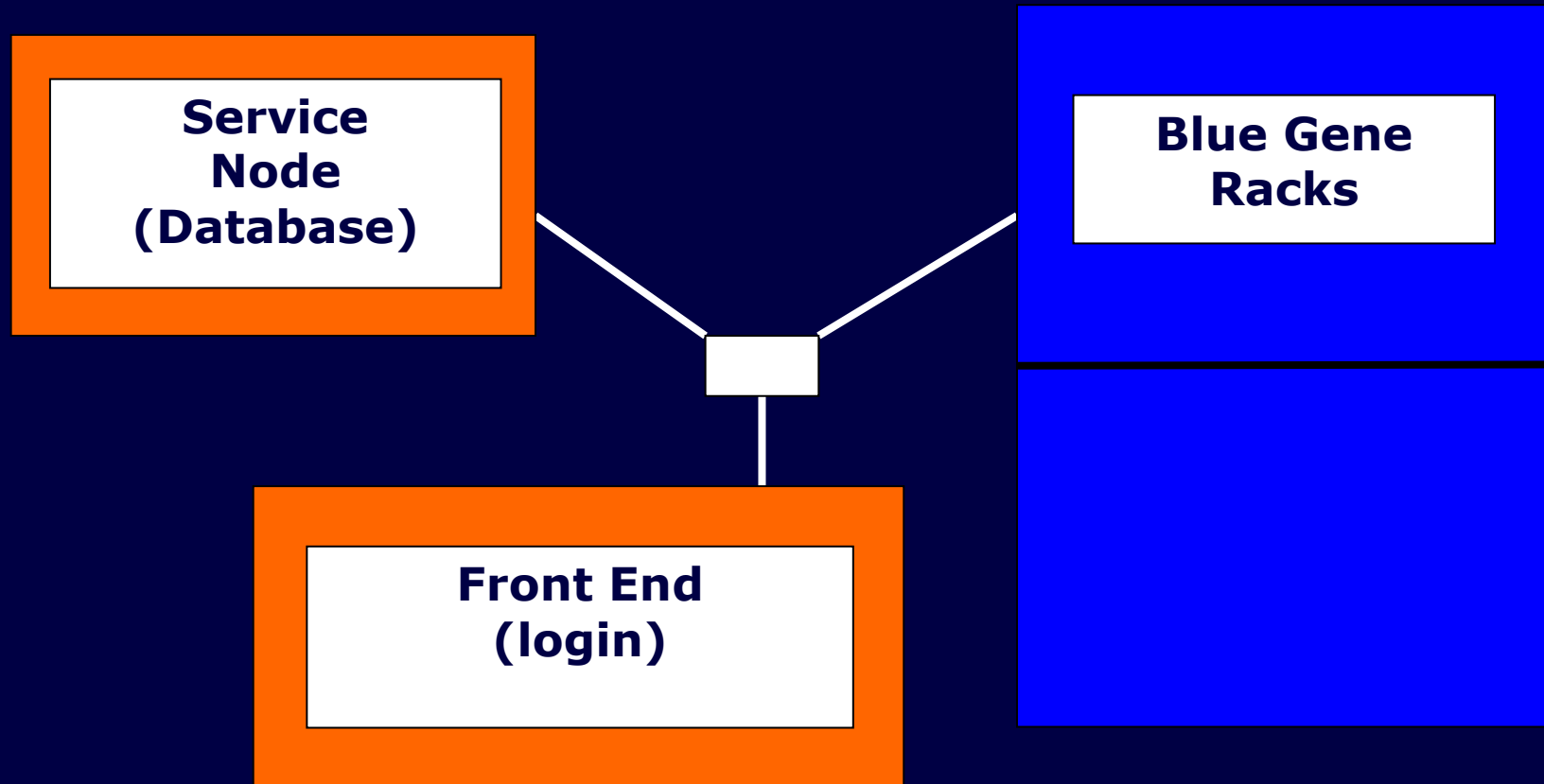
- Оперативная память – до 2GB на вычислительный узел, пропускная способность 13.6GBps
- Трёхуровневый кэш:
  - L1 – отдельный для каждого ядра, размер 32Кб
  - L2 – отдельный для каждого ядра, используется для предварительной выборки информации в кэш L1.
  - L3 – разделен на две части по 4МВ, доступ к ним имеют все четыре ядра, для каждого есть канал чтения и канал записи. Связан с 10-гигабитной сетью (в том случае, если на карте имеется узел ввода-вывода)

# Double Hammer FPU



- SIMD инструкции могут выполняться одновременно на двух FPU
- Параллельные операции load/store
- Данные должны быть выровнены по 16-байтовой границе
  - Иначе производительность будет значительно снижена
  - Даже хуже, чем при использовании только одного FPU
- Компилятор сможет сгенерировать SIMD инструкции, только если данные в памяти расположены подряд (stride-one access)
  - Хотя при более высоких (-O4, -O5) уровнях оптимизации компилятор попытается сгенерировать SIMD инструкции и для данных, расположенных не подряд
  - -O3 -qarch=450d -qtune=450

# BlueGene P



# Состав ПО

- Linux® на узлах ввода\вывода
- MPI (MPICH2) и OpenMP (2.5)
- Стандартное семейство компиляторов IBM XL: XLC/C++, XLF
- Компиляторы GNU
- Система управления заданиями LoadLeveler
- Файловая система GPFS
- Инженерная и научная библиотека подпрограмм (ESSL), математическая библиотека (MASS)

# ОС вычислительного узла BlueGene P

- Compute Node Kernel (CNK)
  - "linux-подобная" ОС
  - Нет некоторых системных вызово (fork() в основном). Ограниченная поддержка mmap(), execve()
  - Минимальное ядро – обработка сигналов, передача системных вызовов к узлам ввода-вывода, старт-завершение задач, поддержка нитей
  - Большинство приложений, которые работают под Linux, портируются на BG/P

# Компиляторы Blue Gene

- IBM XL компиляторы (xlc, xlf77, xlf90)
- Компиляция программ производится на front end узлах
  - Fortran: mpixlf, mpixlf90, mpixlf95
  - C: mpixlc
  - C++: mpixlcxx
- GNU компиляторы mpicc
- MASS математическая библиотека



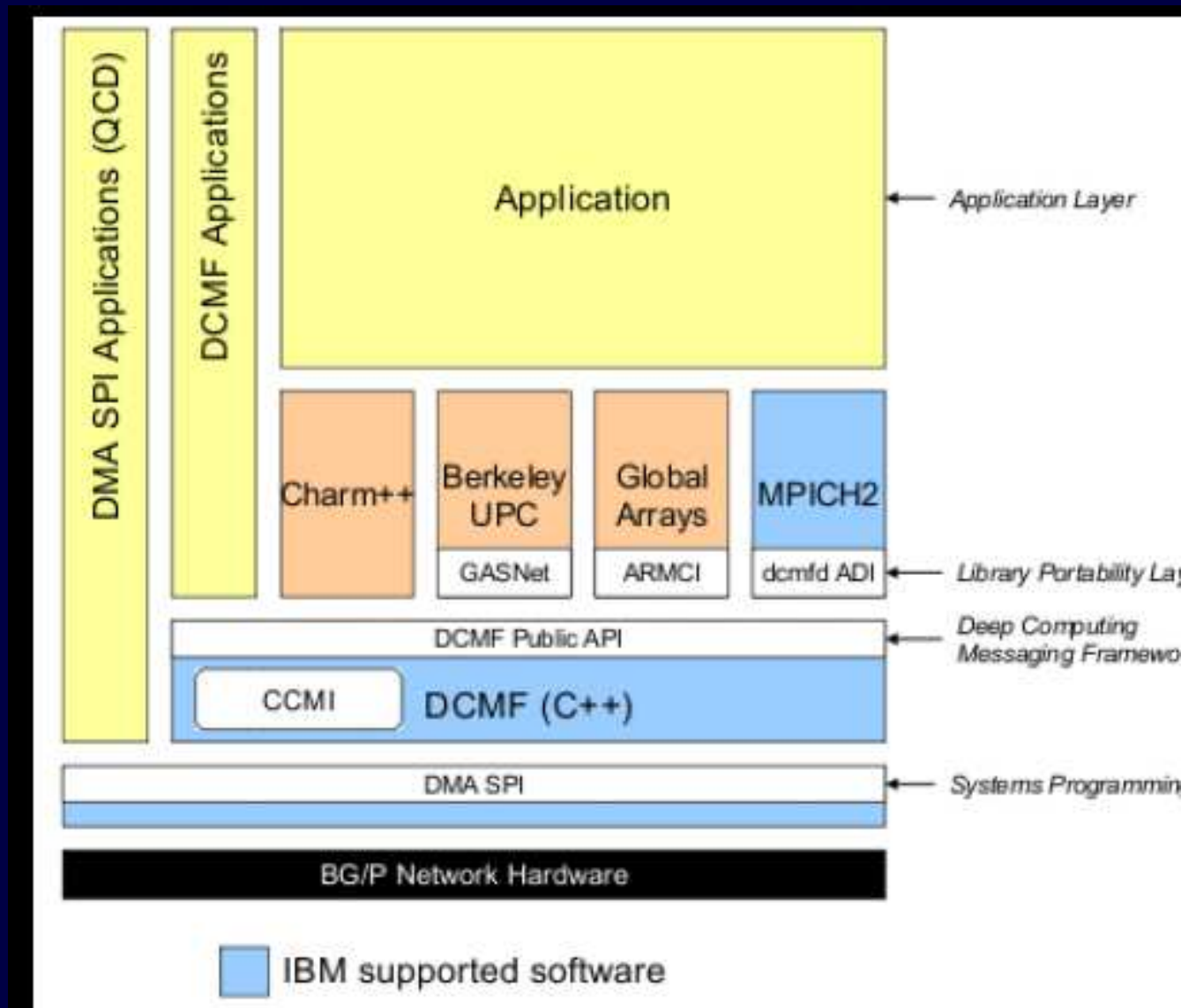
# Реализация MPI

- MPICH2 1.0.x ( стандарт MPI 2.0)
- Не поддерживает управление динамическими процессами
- Для поддержки аппаратного обеспечения Blue Gene/P сделаны добавления и модификации в программной архитектуре MPICH2:
  - коллективные операции могут использовать различные сети при разных обстоятельствах (не только коллективную сеть, но и сеть с топологией тора или сеть глобальных прерываний)
  - Существуют оптимизированные версии функций MPI\_Dims\_create, MPI\_Cart\_create, MPI\_Cart\_map
  - Добавлены функции MPIX - расширение MPI, учитывающее специфику аппаратного обеспечения

# OpenMP

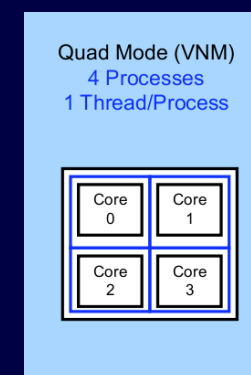
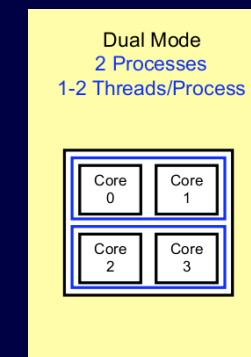
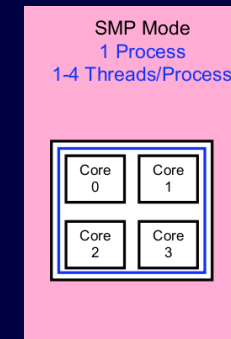
- `_r` суффикс для имени компиляторов например, `mpixlc_r`
- `-qsmpr=omp`  
указание компилятору интерпретировать OpenMP директивы
- Автоматическое распараллеливание  
`-qsmpr`

# Альтернативы MPI



# Режимы использования ядер

- 3 режима
  - SMP: 1 MPI процесс из 4 SMP нитей,  
2 Гб памяти  
- *mode smp*
  - DUAL: 2 MPI процесса по 2 SMP нити,  
1 Гб памяти на MPI процесс  
- *mode dual*
  - VNM: 4 MPI процесса  
- *mode VN*



# Процессорные партии

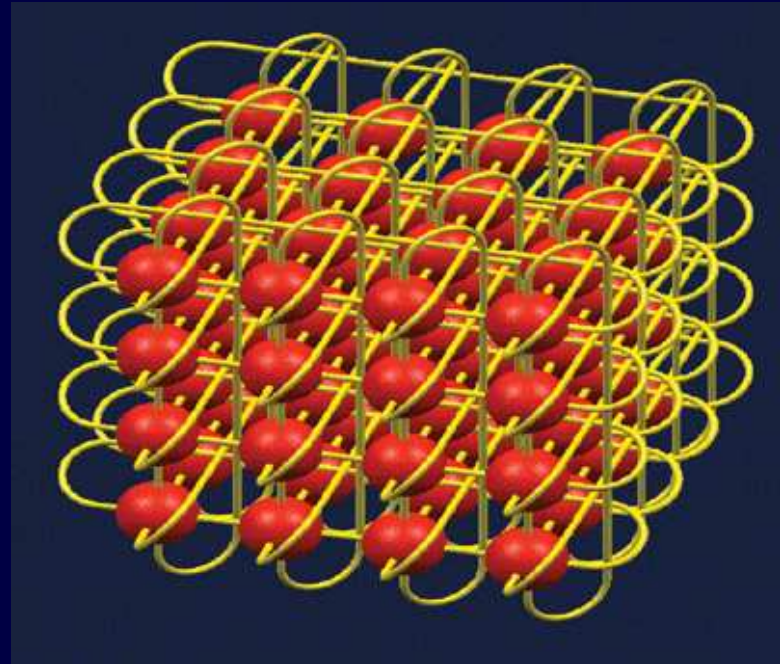
- Подмножества вычислительных узлов, выделяемых задаче
- Каждой задаче выделяется своя партия
- Загрузка задачи на исполнение производится независимо от других задач
- Размер партии определяется кратным 32
- (на текущий момент на системе ВМК - кратным 128 )
- Для партий размером кратным 512 поддерживается топология тора

# Система управления заданиями LoadLeveler

- Система управления заданиями на многопользовательских системах, состоящих из нескольких вычислительных узлов
- Оптимизирует использование имеющихся вычислительных ресурсов
  - Учет приоритета задач и пользователей
  - Динамическое распределение ресурсов
  - Допускается использование разнородных вычислительных узлов
- Используется для запуска как последовательных, так и параллельных задач
- Пользователь формулирует задания в виде командных файлов
- Очередь заданий

# Назначение процессов на процессоры (mapping)

Распределение процессов по процессорам по умолчанию: XYZT, где  
<XYZ> - координаты процесса в торе,  
T - номер ядра внутри процесса  
Сначала увеличивается X - координата, затем Y и Z-координаты, после этого T- номер ядра



# Mapping

2 способа назначения процессов на процессоры:

- с помощью аргумента командной строки  
***-mapfile TXYZ*** (задаем порядок TXYZ или другие перестановки X,Y,Z,T: TYXZ, TZXY и т.д.)
- указание map- файла в командной строке  
***-mapfile my.map***, где my.map – имя файла.
- Синтаксис файла распределения – четыре целых числа в каждой строке задают координаты для каждого MPI-процесса (первая строка задает координаты для процесса с номером 0, вторая строка – для процесса с номером 1 и т.д.).

*0 0 0 1*

*0 0 1 1*

*Очень важно, чтобы этот файл задавал корректное распределение, с однозначным соответствием между номером процесса и координатами <X, Y, Z, T>.*



# Скрипт для запуска задач mpisubmit.bg

```
ivanov@fen1: /home/ivanov/examples>mpisubmit.bg -n 128 -w 00:15:00  
-e OMP_NUM_THREADS=4 -m smp example - arg1 arg2
```

<code>-n, --nproc</code>	128	Запрашиваемое число узлов
<code>-w, --wtime</code>	00:15:00	Максимальное время выполнения
<code>-m, --mode</code>	smp	Режим использования ядер процессора
<code>-e, --env</code>		Переменные окружения
<code>-t, --top</code>	PREFER_TORUS	Топология
<code>-d, --debug</code>		Вывести содержимое командного файла на экран без постановки задачи в очередь
<code>-h, --help</code>		

# Основной шаблон протокола работы пользователя (1)

1. Выход на BGP:

```
%ssh <опции> <логин>@bluegene1
```

Например:

```
%ssh -X ivanov@bluegene1
```

2. Копирование файлов с локального компьютера на Blue Gene/P:  
(локальная машина)

```
%scp example.cpp ivanov@bluegene1:~ivanov/examples
```

## Основной шаблон протокола работы пользователя (2)

3. Компиляция MPI-программы (на языке C, C++ и Fortran90 соответственно): (BGP, front-end)

```
%mpixlc example.c -o c_ex
```

```
%mpixlcxx example.cpp -o cpp_ex
```

```
%mpixlf90 example.f90 -o f_ex
```

4. Компиляция гибридной MPI-OpenMP программы:

```
%mpixlc_r -qsmp=omp hw.c -o hw
```

```
% mpixlcxx_r -qsmp=omp hw.cpp -o hw
```

## Основной шаблон протокола работы пользователя (3)

б. Постановка MPI-программы в очередь задач с лимитом выполнения 15 минут на 128 узлов в режиме VN с параметром командной строки:

```
%mpisubmit.bg -w 00:15:00 -m VN -n 128 hw - 0.1 200
```

б. Постановка MPI+OpenMP программы **prog** в очередь задач с лимитом выполнения 15 минут на 128 узлов в режиме SMP с тремя нитями на каждом узле и с параметром командной строки **parameter**:

```
%mpisubmit.bg -w 00:15:00 -m SMP -n 128  
-e «OMP_NUM_THREADS=3» example -- 100
```

## Основной шаблон протокола работы пользователя (4)

5. Постановка MPI-программы в очередь задач с лимитом выполнения 15 минут на 128 узлов в режиме **VN** с параметром командной строки :

```
%mpisubmit.bg -w 00:15:00 -m VN -n 128 prog - 0.1 200
```

6. Постановка MPI+OpenMP программы **prog** в очередь задач с лимитом выполнения 15 минут на 128 узлов в режиме **SMP** с 4 нитями на каждом узле и с параметром командной строки **parameter**:

```
%mpisubmit.bg -w 00:15:00 -m SMP -n 128  
-e «OMP_NUM_THREADS=4» prog -- parameter
```

# Основной шаблон протокола работы пользователя (5)

7. Проверка состояния очереди задач:

***%llq***

8. Удаление задачи из очереди:

***%llcancel <task\_id>***

# LoadLeveler: llq

- Пример вывода команды llq

```
angel.cs.msu.su - PuTTY
-rw-rw-r-- 1 popova cmc      384 2012-06-25 08:44 ysz.bin
-rw-rw-r-- 1 popova cmc      384 2012-06-25 08:44 zs.bin
-rw-rw-r-- 1 popova cmc      384 2012-06-25 08:44 zsdx.bin
-rw-rw-r-- 1 popova cmc      384 2012-06-25 08:44 zsdz.bin
popova@fen1:~/amp12/academy> llq
Id                               Owner      Submitted   ST PRI Class      Running O
-----
fen1.158483.0                     popova     6/25 08:37 R  50  n128_m15    fen1
fen1.158478.0                     irina      6/25 08:19 I  50  n128_m15
fen1.158479.0                     popov      6/25 08:20 I  50  n128_m15

3 job step(s) in queue, 2 waiting, 0 pending, 1 running, 0 held, 0 preempted
popova@fen1:~/amp12/academy> llmap

      Jobs Distribution
Midplane:      M0          M1
I/O node:    0 1 2 3    0 1 2 3
Rack R00 |           A |           |
+-----+-----+
Rack R01 | * * * * |           |
+-----+-----+

Priority ID      Job      User      Reqst Alloc  Class      Requested Remaining
===== == =====
Running:
          A 158483.0    popova    128   128   n128_m15   00:15:00   00:17:04

Idle:
-13777    158478.0    irina     128           n128_m15   00:15:00
-13815    158479.0    popov     128           n128_m15   00:15:00
popova@fen1:~/amp12/academy>
```

# LoadLeveler

- Файлы .out .err

```
popova@fen1:~/amp12/academy> ll
пБяБпБпЕпБ 19136
-rwxr-xr-x 1 popova cms 12656146 2012-06-25 08:36 par_n
-rw-r--r-- 1 popova cms 878099 2012-06-25 08:49 par_n.158483.out
-rw-r--r-- 1 popova cms 0 2012-06-25 08:51 par_n.158484.err
-rw-r--r-- 1 popova cms 0 2012-06-25 08:51 par_n.158484.out
-rw-rw-r-- 1 popova cms 61396 2012-06-25 08:56 pcsit.bin
-rw-r--r-- 1 popova cms 2148 2012-06-25 08:36 probl2.21.10.in
-rw-r--r-- 1 popova cms 2160 2012-06-25 08:36 probl2.in
```

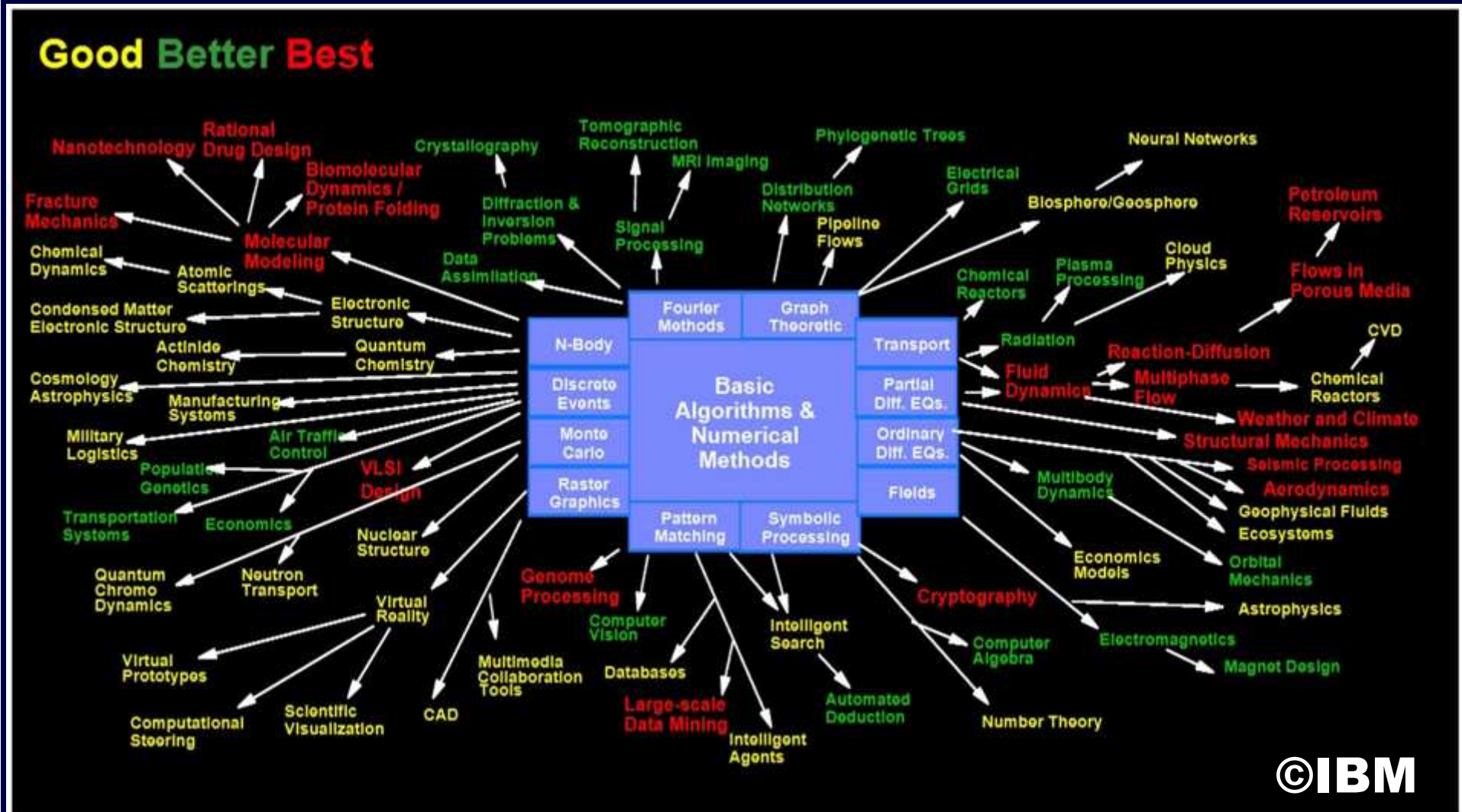


# llstatus

```
angel.cs.msu.su - PuTTY
llstatus: 2512-848 No base partition list was specified. A base partition
s required with the -B option.
llstatus: Usage: llstatus [-?] [-H] [-v] [-W] [-R] [-F] [-M] [-l] [-a] [-C
[-B {base_partition_list | all}] [-P {partition_list | all}] [-X {cluster
| all}] [-f category_list] [-r category_list] [-h hostlist | hostlist]
popova@fen1:~/amp12/academy> llstatus -all
=====
Name = fen1
Machine = fen1
Arch = PPC64
OpSys = Linux2
SYSPRIO = (0 - QDate)
MACHPRIO = (0 - LoadAvg)
VirtualMemory = 4193936 kb
Disk = 15023472 kb
KeyboardIdle = 24
Tmp = 911972 kb
LoadAvg = 0,230000
ConfiguredClasses = n128_m15(16) n128_m30(16) n128_h01(16) n128_h02(16)
10(8) n256_m30(8) n256_h01(8) n256_h02(8) n256_h04(8) n256_h08(8) n256_h12
12_m05(4) n512_m30(4) n512_h01(4) n512_h02(4) n512_h06(4) n512_h12(4) n512
) n1024_m03(2) n1024_m15(2) n1024_m30(2) n1024_h01(2) n1024_h02(2) n1024_h
n1024_h12(2) n1024_h24(2) n2048_m03(1) n2048_m15(1) n2048_m30(1) n2048_h01
048_h02(1) n2048_h06(1) n2048_h12(1) n2048_h24(1)
AvailableClasses = n128_m15(15) n128_m30(16) n128_h01(16) n128_h02(16)
10(8) n256_m30(8) n256_h01(8) n256_h02(8) n256_h04(8) n256_h08(8) n256_h12
12_m05(4) n512_m30(4) n512_h01(4) n512_h02(4) n512_h06(4) n512_h12(4) n512
) n1024_m03(2) n1024_m15(2) n1024_m30(2) n1024_h01(2) n1024_h02(2) n1024_h
n1024_h12(2) n1024_h24(2) n2048_m03(1) n2048_m15(1) n2048_m30(1) n2048_h01
048_h02(1) n2048_h06(1) n2048_h12(1) n2048_h24(1)
DrainingClasses =
DrainedClasses =
Pool =
FabricConnectivity =
```




# Обзор алгоритмов и методов, реализованных на Blue Gene

Материаловедение и нанонауки, ядерная энергетика, биоинформатика, вычислительная физика, гидродинамика



Международная летняя суперкомпьютерная академия, Москва, МГУ, 25 июня - 7 июля 2012 г

# Производительность конкретных приложений

Application	Domain	Cores	Scaling (%)	Sustained TF
Qbox	Materials Science	131,072	57%	207 
CPMD	Molecular Dynamics	131,072	30%	110
Columbia	Physics (QCD)	131,072	30%	110 
ddcMD	Molecular Dynamics	131,072	28%	101 
sPPM	CFD (simplified Piecewise Parabolic Method)	131,072	18%	66
WRF	Geosciences (Weather)	32,768	14%	51
HOMME	Geosciences (Climate)	32,768	12%	44

©IBM  Gordon Bell Award

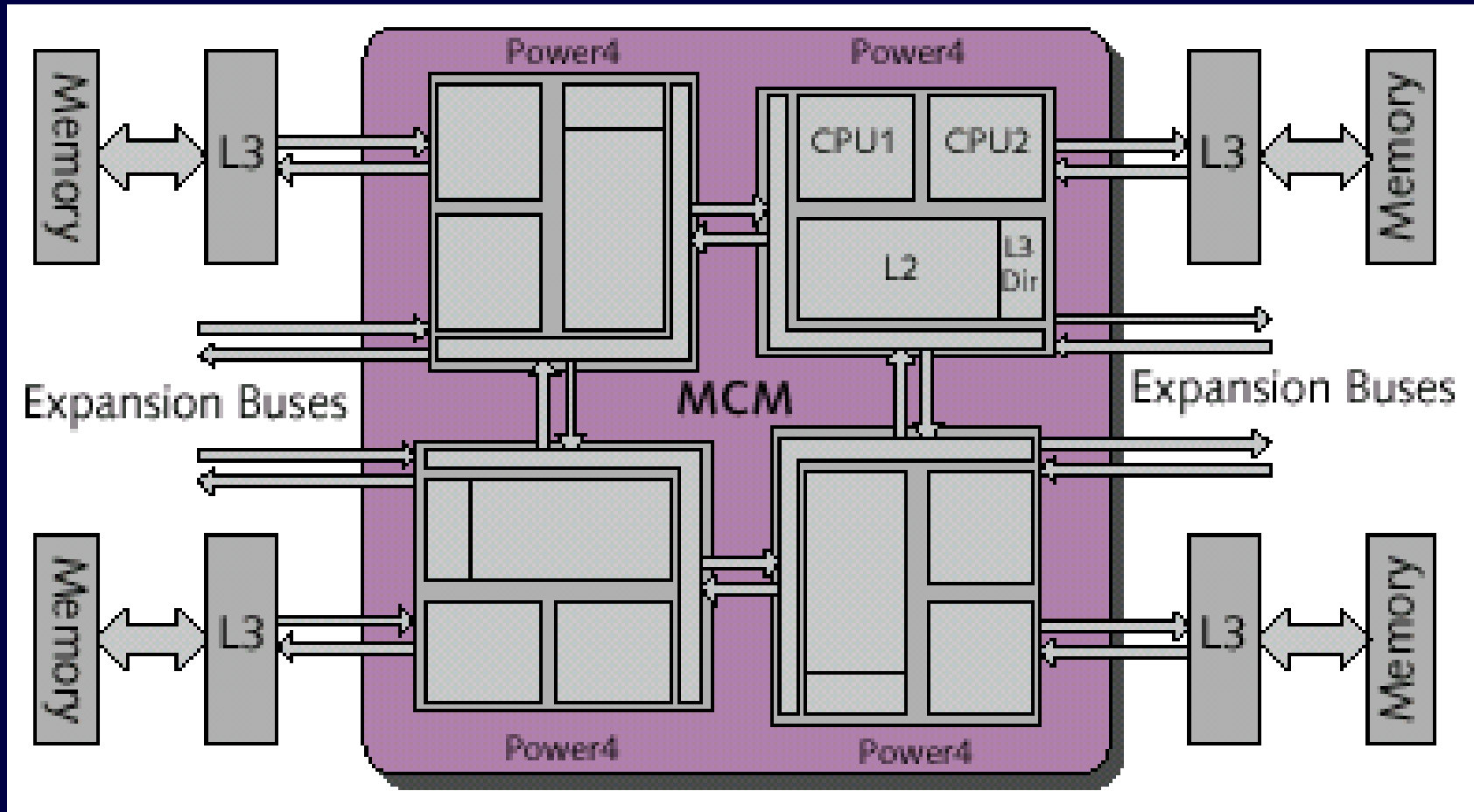
# Дополнительные слайды

- Обзор вычислительной системы  
IBM pSeries 690 Regatta

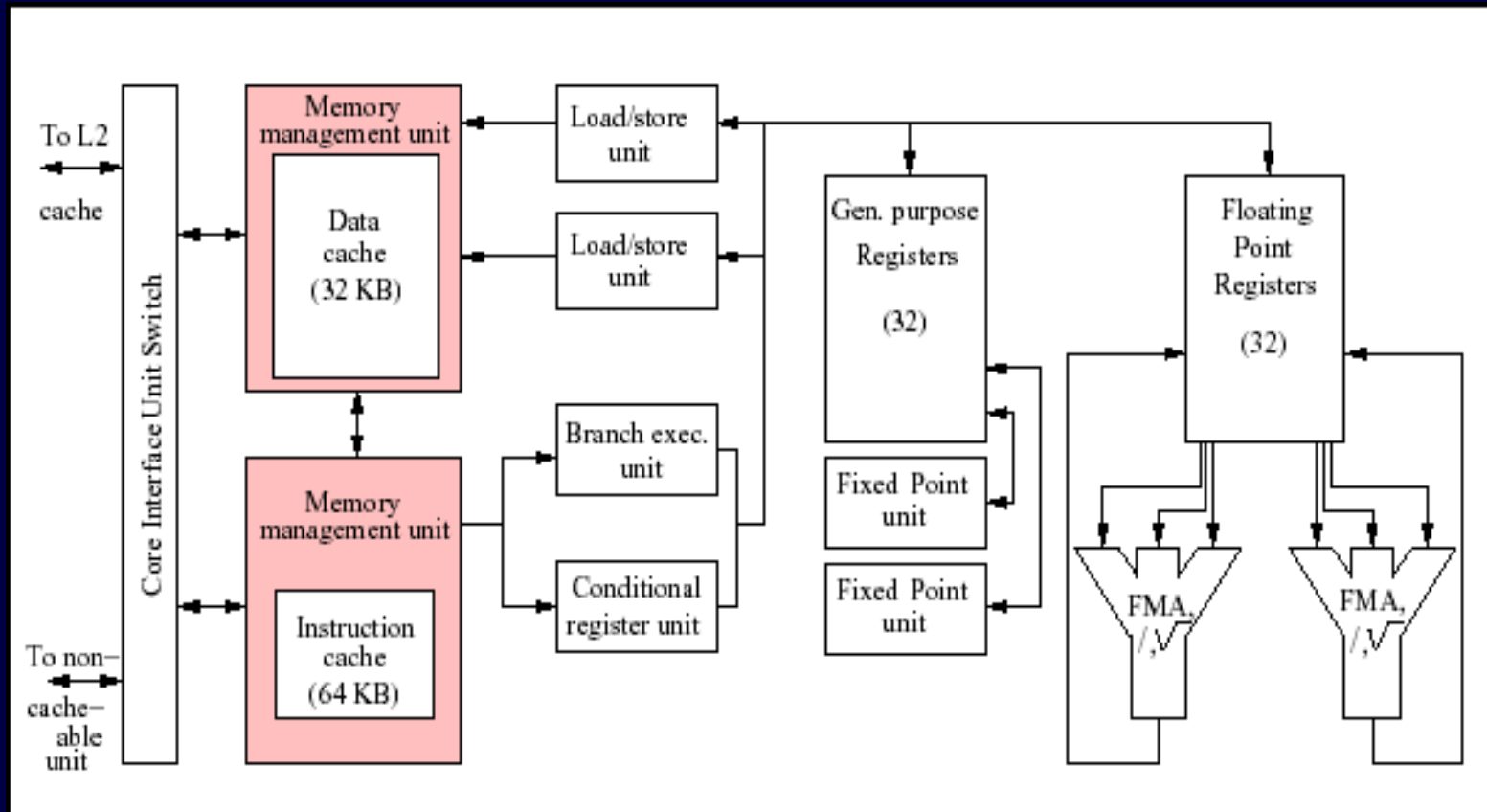
## p-series 690 Регатта ([regatta.cs.msu.su](http://regatta.cs.msu.su))

- 16-процессорная SMP система
- IBM Power4 процессоры
- 1.3 GHz - тактовая частота
- 83 GFlops - максимальная производительность
- 64 Gbytes ОЗУ
- 32 KB L1 cache на процессор
- 1.41 MB L2 cache (общий для 2-ух процессор.)
- 128 MB L3 cache (общий для 8 процессоров)

# Архитектура IBM pSeries690 Regatta



# Архитектура IBM pSeries690 Regatta. Процессор POWER4.



# Опции компиляторов

- Уровни оптимизации
  - O базовая оптимизация
  - O2 то же, что -O
  - O3 более агрессивная оптимизация
  - O4 еще более агрессивная оптимизация: межпроцедурный анализ (IPA) , оптимизация с учетом особенностей архитектуры
  - O5 агрессивный IPA



# IBM компиляторы (1)

- -O5 -qarch=pwr4 -qtune=pwr4
- -O3 (-qhot) -qarch=pwr4 -qtune=pwr4

## IBM компиляторы (2)

- При использовании O3 или ниже рекомендуется оптимизация под архитектуру (выполняется автоматически для -O4 и -O5):
  - qarch=auto      оптимизация под архитектуру
  - qtune=auto      оптимизация под процессор
  - qcache=auto    оптимизация под кэш

# OpenMP

- Автоматическое распараллеливание

- qreport=smp`list`

- Исходный текст

- `mycode.lst`

- Информация о распараллеленных циклах

- Размер стека

- По умолчанию 4 MB

- Может быть увеличена

- `setenv XLSMPOPTS $XLSMPOPTS\stack=size`

- где *size* - размер в байтах

# Математическая библиотека (1)

- MASS library
  - Mathematical Acceleration SubSystem
- sqrt, rsqrt, exp, log, sin, cos, tan, atan, atan2, sinh, cosh, tanh, dnint,  $x^{**}y$
- Fortran и C

# Математическая библиотека (2)

- Опции:

Fortran:            -lmass

C:                    -lmass -lm

- Ускорение:

<hr/> exp	2.4
<hr/> log	1.6
<hr/> sin	2.2
<hr/> complex atan	4.7

## Ссылки

- <http://www.ibm.com/servers/deepcomputing/bluegene.html>
- <http://www3.ibm.com/systems/deepcomputing/bluegene/>
- IBM System Blue Gene Solution: Blue Gene/P Application Development, SG24-7287-00  
<http://www.research.ibm.com/journal/rd/521/toc.pdf.html>
- <http://hpc.cs.msu.su>