

Системы и средства параллельного программирования

3 курс кафедры СКИ
сентябрь – декабрь 2016 г.

Лектор доцент Н.Н.Попова

Лекция 6
24 октября 2016 г.

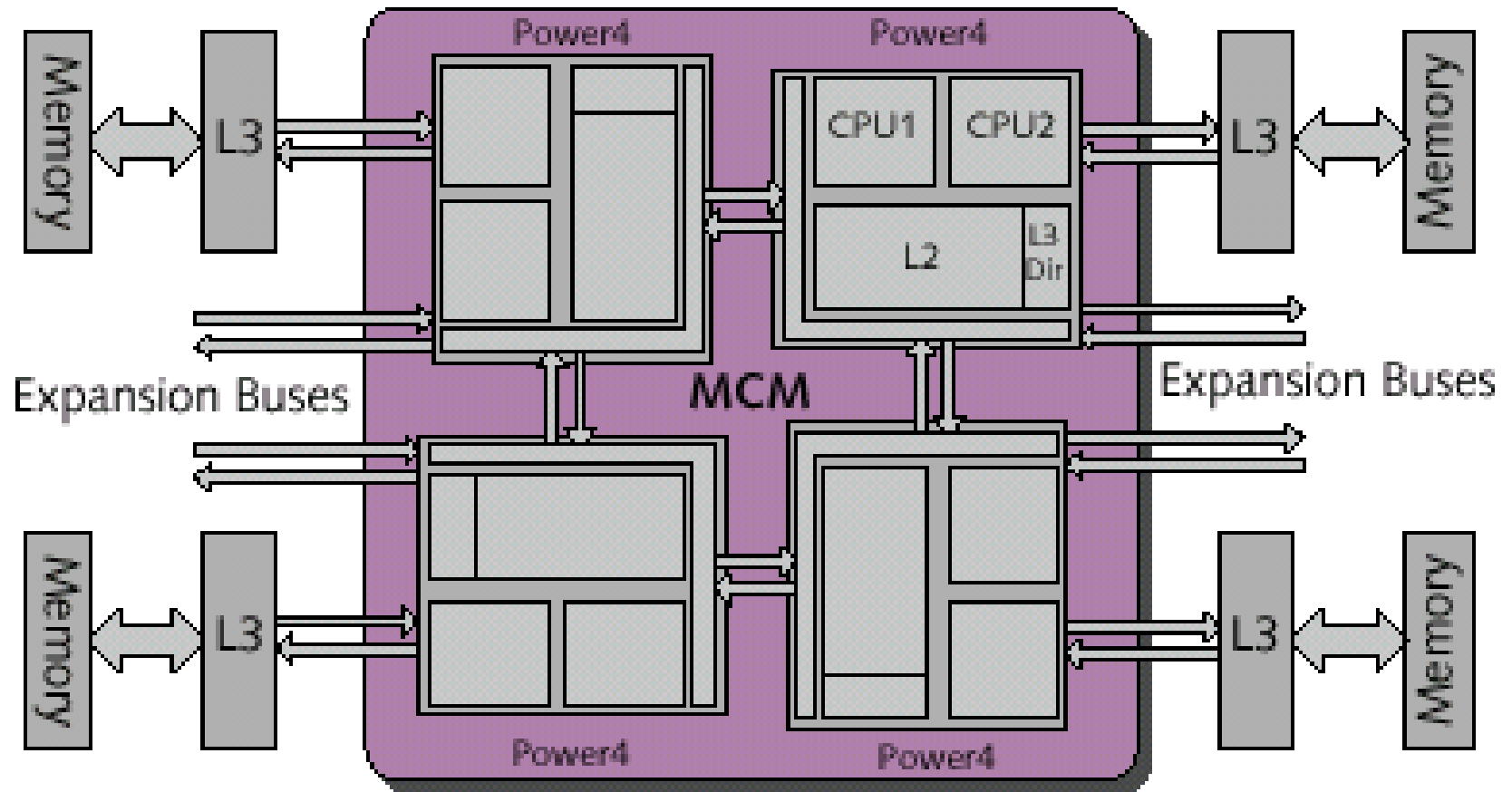
Тема

- Программно-аппаратная архитектура BC pSeries690 и Blue Gene/P
- Производные типы данных MPI

IBM p-series 690 Regatta (Регатта)

- 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



Компиляторы

	Посл.	MPI	OpenMP	Mixed
Fortran 77	xl f	mpxl f	xl f_r	mpxl f_r
Fortran 90	xl f90	mpxl f90	xl f90_r	mpxl f90_r
Fortran 95	xl f95	mpxl f95	xl f95_r	mpxl f95_r
C	cc	mpcc	cc_r	mpcc_r
	xl c	mpxl c	xl c_r	mpxl c_r
C++	xl C	mpCC	xl C_r	mpCC_r

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
- Поддерживает очередь заданий

Схема выполнения заданий на системе Regatta

- Выход на удаленную систему.
`ssh ivanov@regatta.cs.msu.su`
- Компиляция MPI-программы
`mpicc -o prog prog.c`
- Компиляция OpenMP- программы
`gcc -fopenmp -o prog prog.c`
- Постановка MPI-программы в очередь на выполнение
`mpisubmit -w 10:00 -n 8 prog`
- Постановка OpenMP-программы в очередь на выполнение
`ompsubmit -n <число_процессоров> -w <лимит_счетного_времени> <имя_программы> <параметры_программы>`
- Просмотр состояния очереди
`llq`
- Удаление задания из очереди в случае необходимости
`llcancel <id>`
- Копирование результатов на локальную машину.

Архитектура и программное обеспечение массивно-параллельной вычислительной системы

IBM Blue Gene / P

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

Дополнительная информация на сайте

hpc@ctmc | Высокопроизводительные вычисления на ВМК МГУ - Mozilla Firefox

File Edit View History Bookmarks Tools Help

hpc@ctmc | Высокопроизводительные ... x +

hpc.cs.msu.su

hpc@ctmc Высокопроизводительные вычисления на ВМК МГУ

Blue Gene/P Общие вопросы Форум Поддержка Ссылки Регистрация

▼ Blue Gene/P

- Новым пользователям
- Быстрый старт
- Подключение
- Файловая система
- ▶ Разработка программ
- ▶ Запуск заданий
- Программное обеспечение
- Официальная документация
- FAQ
- Серия Blue Gene в мире
- Фотографии
- Публикации

▼ Общие вопросы

- Доступ по SSH
- Выбор пароля
- VPN-подключение

○ Форум


○ Поддержка

Суперкомпьютер IBM Blue Gene/P на факультете ВМК МГУ

С 2008 года на факультете ВМК МГУ имени М. В. Ломоносова работает суперкомпьютер IBM Blue Gene/P, который является одной из первых систем данной серии среди установленных в мире. Архитектура Blue Gene была предложена компанией IBM в рамках проекта по исследованию возможностей достижения новых рубежей в супервычислениях. Более крупные машины данной серии в настоящее время занимают лидирующие позиции в списке пятисот самых мощных компьютеров мира [Top500](#), а машина Blue Gene/P, установленная на ВМК МГУ, в редакции рейтинга от 18 ноября 2008 года оказалась на [128-м месте](#) (в редакции от 16 ноября 2009 года — 348-е место). В списке самых высокопроизводительных компьютеров стран СНГ, опубликованном 22 сентября 2009 года, она находится на [4-й строчке](#).

Система IBM Blue Gene/P принадлежит к новому семейству суперкомпьютеров, обладающих высокой производительностью, масштабируемостью, возможностью обрабатывать данные большего объема, потребляя при этом значительно меньше энергии и занимая меньшую площадь по сравнению с предыдущими системами.

На факультете ВМК МГУ представлена конфигурация, состоящая из двух стоек, содержащих в общей сложности 2048



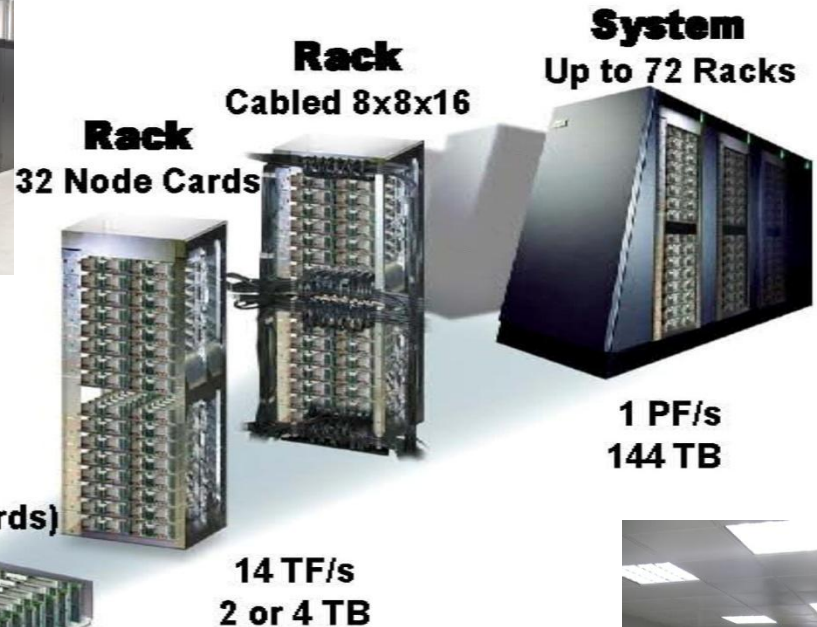
Общая характеристика систем Blue Gene

- 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 интерконнект
 - несколько коммуникационных подсистем для различных целей
- Ультра легкая ОС
 - выполнение вычислений и ничего лишнего
- Стандартное ПО
 - Fortran/C/C++ и MPI

Общая характеристика систем Blue Gene

- Массивно-параллельные системы с распределенной памятью
- Высокая плотность упаковки
 - процессоры с низким энергопотреблением (40 W ~ лампочка)
- Высокопроизводительный интерконект
 - несколько коммутационных подсистем для различных целей
- Ультра легкая ОС
 - выполнение вычислений и ничего лишнего
- Стандартное ПО Standard software
 - Fortran/C/C++ и MPI

Blue Gene/P Hardware



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

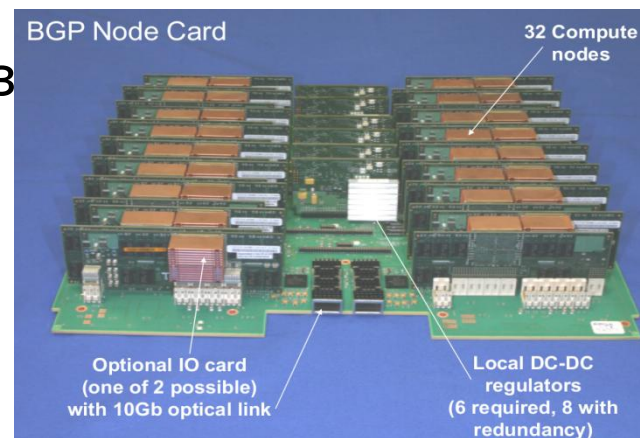
<http://hpc.cs.msu.ru>

- 2048 4-ех ядерных узлов
- пиковая производительность 27.2 Tflop/s
- Реальная производительность по тесту Linpack:
23.2 Тфлоп/с
 - 85% от пиковой
- общий объем ОЗУ 4 TB



Компоненты Blue Gene P

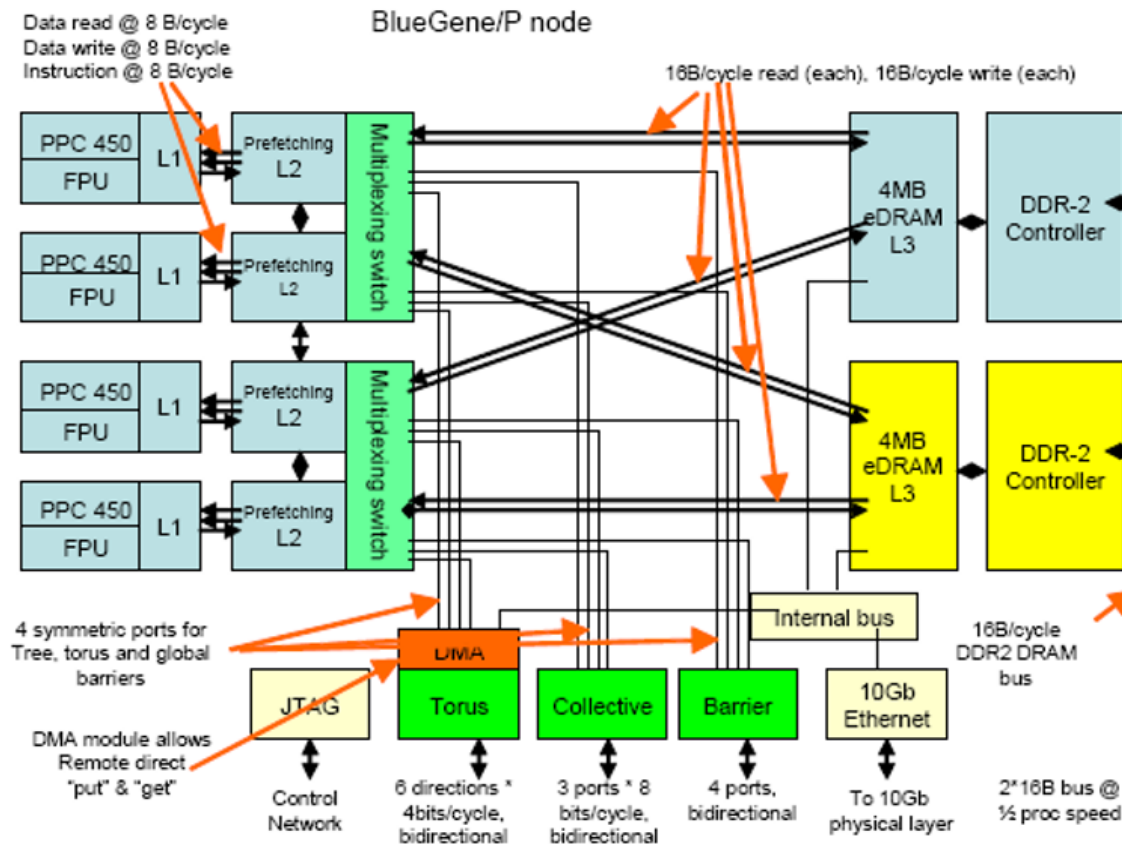
- Основная единица – четырехядерный вычислительный узел (процессор) , ядро – PowerPC 450 850Mhz + память (2GB)
- Node card = 32 вычислительных узла ввода-вывода
- Стойка – 32 node cards
- Число процессоров в стойке 1024
- Итоговое число ядер на стойку - 4096



Характеристики вычислительного узла

- 4 ядерный 32-битный процессор PowerPC 850 МГц
 - Двойное устройство для работы с вещественными числами с плавающей точкой (double precision)
 - 2 Гб памяти
 - Работает под управлением облегченной ОС
 - Создание процессов и управление ими
 - Управление памятью
 - Отладка процессов
 - Ввод-вывод
 - Объем виртуальной памяти равен объему физической

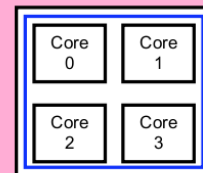
Архитектура процессора



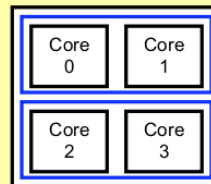
Характеристики вычислительного узла

- 3 режима использования ядер
 - **SMP:**
1 MPI процесс из 4 SMP нитей,
2 Гб памяти
 - **DUAL:**
2 MPI процесса по 2 SMP
нити, 1 Гб памяти на MPI
процесс
 - **VNM:**
4 MPI процесса

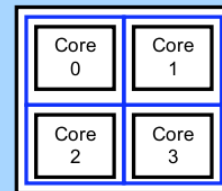
SMP Mode
1 Process
1-4 Threads/Process



Dual Mode
2 Processes
1-2 Threads/Process



Quad Mode (VNM)
4 Processes
1 Thread/Process



Компоненты Blue Gene/P

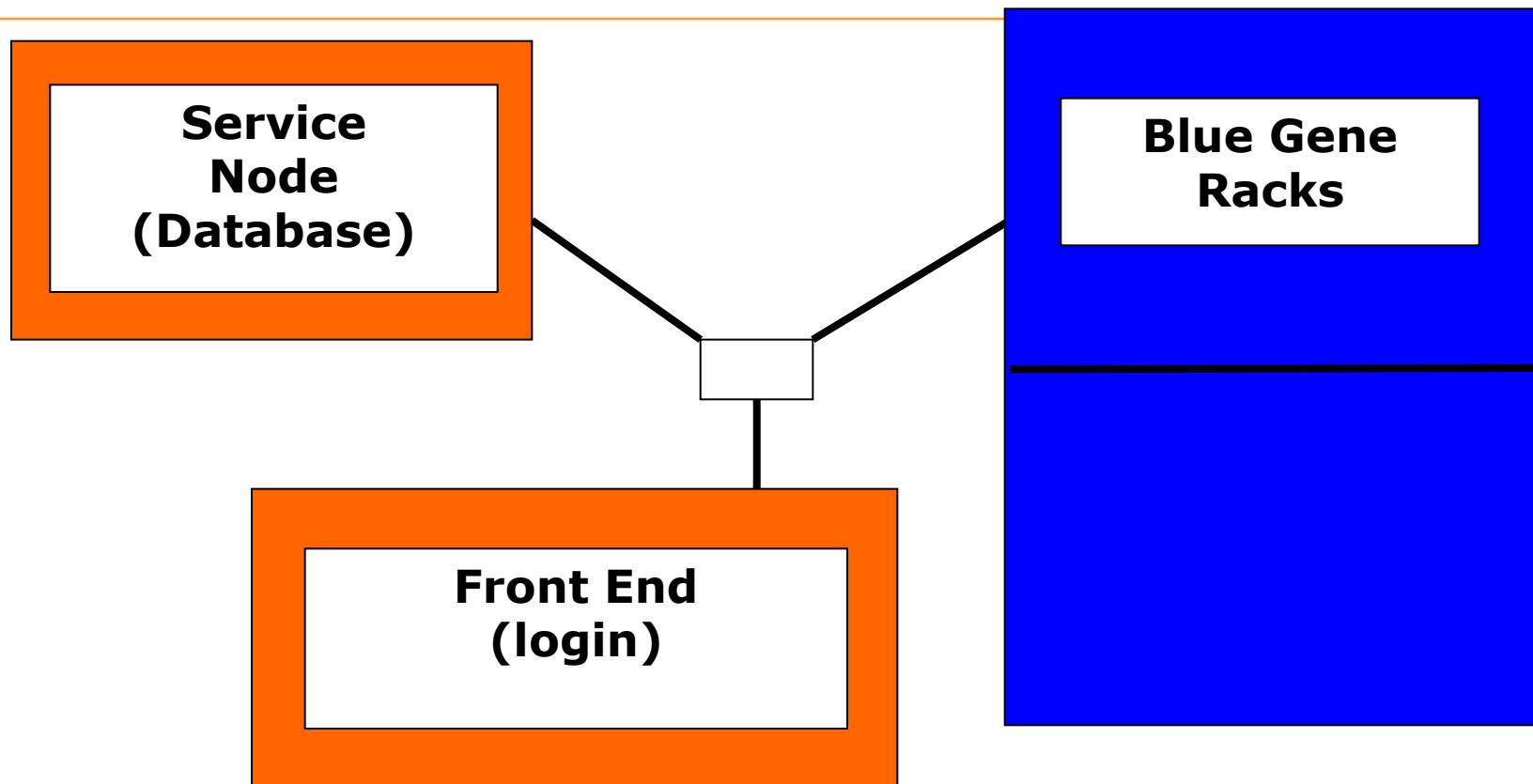
- Помимо вычислительных узлов, в состав системы также входят:
 - узлы ввода-вывода
 - узел управления системой
 - не менее одного узла front end (через них осуществляется доступ пользователей к системе)
 - сеть, связывающая компоненты системы
 - специализированная сеть для сообщения между сервисным узлом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

Процессоры ввода-вывода

Отличия по сравнению с вычислительным узлом:

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

BlueGene/P



3-мерный тор

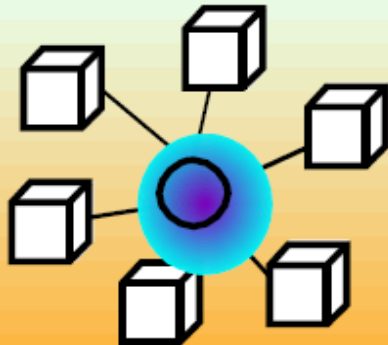
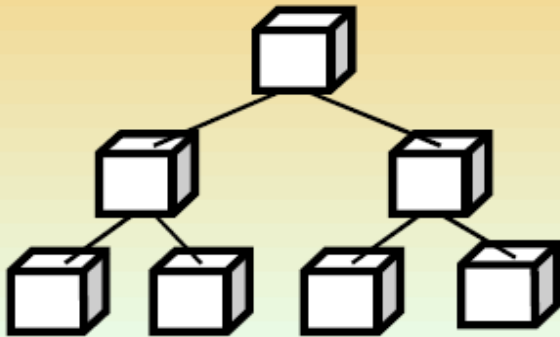
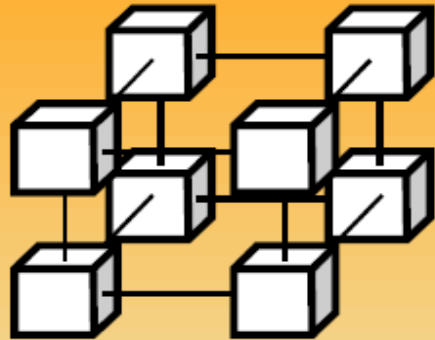
- Используется для обмена сообщениями между соседними узлами, а также для многих коллективных операций

Коллективная сеть – дерево

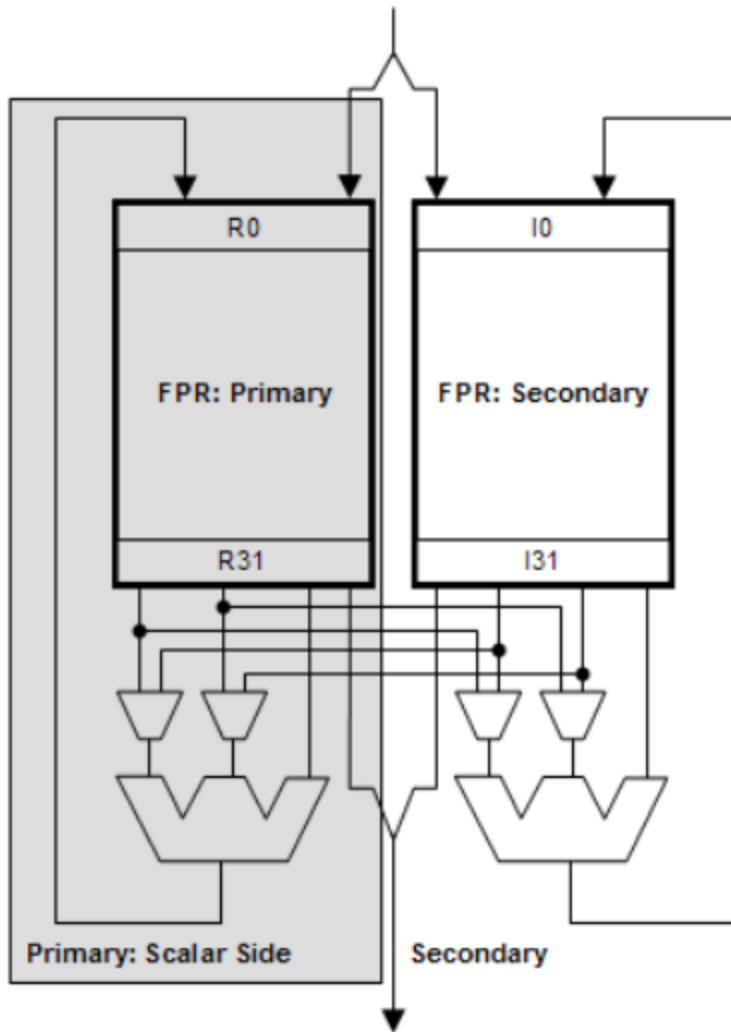
- Для глобальной коммуникации одно-ко-всем (broadcast, reduction)
- 6.8 ГБ/с на порт
- Соединяет все вычислительные узлы и узлы ввода-вывода
- Используется для коллективных операций и коммутатора MPI_COMM_WORLD

Высокоскоростная сеть для глобальных прерываний

- Для MPI_Barrier



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

Память

- Оперативная память – до 2GB на вычислительный узел, пропускная способность 13.6GBps
- Трёхуровневый кэш:
 - L1 – отдельный для каждого ядра, размер 32Kb
 - L2 – отдельный для каждого ядра, используется для предварительной выборки информации из кэша L1. Считывает\записывает по 16b за одно обращение.
 - L3 – разделен на две части по 4MB, доступ к ним имеют все четыре ядра, для каждого есть канал чтения и канал записи.

Организация cash

Cache	Total per node	Size	Replacement policy	Associativity
L1 instruction	4	32 KB	Round-Robin	<ul style="list-style-type: none">▶ 64-way set-associative▶ 16 sets▶ 32-byte line size
L1 data	4	32 KB	Round-Robin	<ul style="list-style-type: none">▶ 64-way set-associative▶ 16 sets▶ 32-byte line size
L2 prefetch	4	14 x 256 bytes	Round-Robin	<ul style="list-style-type: none">▶ Fully associative (15-way)▶ 128-byte line size
L3	2	2 x 4 MB	Least recently used	<ul style="list-style-type: none">▶ 8-way associative▶ 2 bank interleaved▶ 128-byte line size

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 вызовов (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

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`
- `-qsmp=omp`
указание компилятору интерпретировать OpenMP директивы
- Автоматическое распараллеливание
`-qsmp`

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

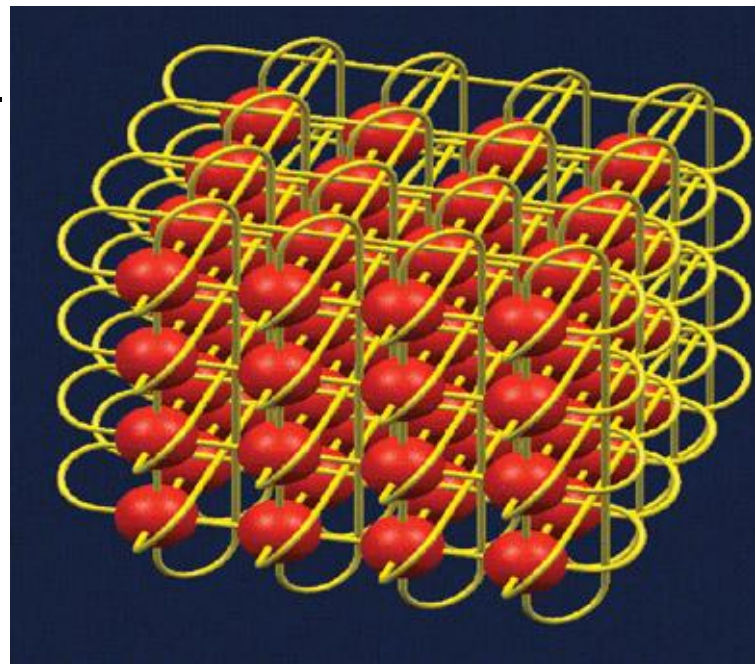
Рекомендации по использованию MPI на Blue Gene/P

- Объединять взаимодействия и вычисления, используя MPI_Irecv и MPI_Isend.
- Уделять особое внимание балансировке нагрузки.
- Избегать типа данных vector и непоследовательных типов данных. Хотя производные типы данных в MPI могут элегантно описывать сложные структуры данных, но их использование, как правило, уменьшает производительность.
- Реализация MPI на Blue Gene/P чувствительна к выравниванию буферов. Выравнивание по 32 байта или хотя бы по 16 байт может существенно улучшить производительность.

Назначение процессов на процессоры (mapping)

Распределение процессов по процессорам по умолчанию для -

- режима SMP:
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
- Для режимов DUAL и VN:
TXYZ



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>.

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:

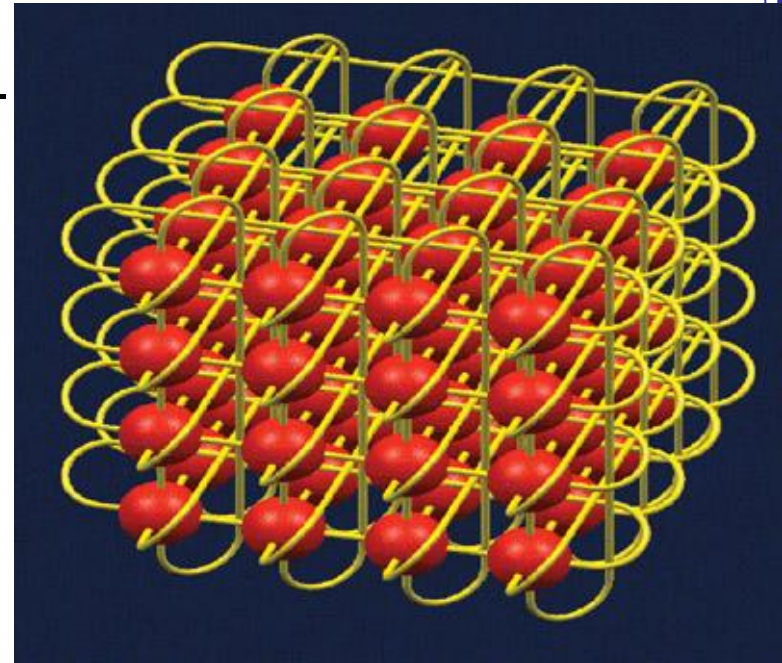
(локальная машина)

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

Назначение процессов на процессоры (mapping)

Распределение процессов по процессорам по умолчанию для -

- режима SMP:
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
- Для режимов DUAL и VN:
TXYZ



Основной шаблон протокола работы пользователя (2)

3. Компиляция MPI-программы на языке C или C++ : (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)

6. 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

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» 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
```