NUMA-системы (Non-Uniform Memory Access)

Подготовил
Студент группы А-13-06
Александр Свистунов

ОСНОВНЫЕ КЛАССЫ СОВРЕМЕННЫХ ПАРАЛЛЕЛЬНЫХ КОМПЬЮТЕРОВ.

Массивно-параллельные системы (MPP)

MPP система состоит из однородных вычислительных узлов, включающих:

- один или несколько центральных процессоров (обычно RISC),
- локальную память (прямой доступ к памяти других узлов невозможен),
- коммуникационный процессор или сетевой адаптер
- иногда - жесткие диски (как в SP) и/или другие устройства В/В

Примеры: IBM RS/6000 SP2, Intel PARAGON/ASCI Red, CRAY T3E, Hitachi SR8000, транспьютерные системы Parsytec.

Симметричные мультипроцессорные системы (SMP)

Система состоит из нескольких однородных процессоров и массива общей памяти (обычно из нескольких независимых блоков). Все процессоры имеют доступ к любой точке памяти с одинаковой скоростью. Процессоры подключены к памяти либо с помощью общей шины (базовые 2-4 процессорные SMP-сервера), либо с помощью crossbar-коммутатора (HP 9000). Аппаратно поддерживается когерентность кэшей.
Примеры : HP 9000 V-class, N-class; SMP-сервера и рабочие станции на базе процессоров Intel (IBM, HP, Compaq, Dell, ALR, Unisys, DG, Fujitsu и др.).

Системы с неоднородным доступом к памяти (NUMA)

NUMA (nonuniform memory access) – гибридная архитектура, так как по сути она представляет собой MPP (массивно-параллельная архитектура), элементы которой более мелкие SMP. Главная особенность такой архитектуры - неоднородный доступ к памяти.
Примеры : HP 9000 V-class в SCA-конфигурациях, SGI Origin3000, Sun HPC 10000, IBM/Sequent NUMA-Q 2000, SNI RM600.

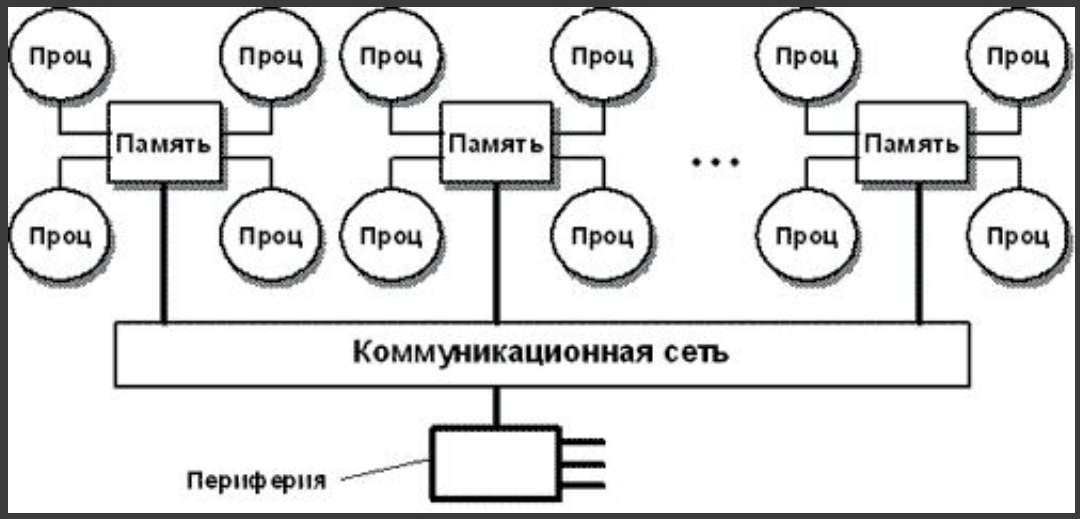
Параллельные векторные системы (PVP)

Основным признаком PVP-систем является наличие специальных векторно-конвейерных процессоров, в которых предусмотрены команды однотипной обработки векторов независимых данных, эффективно выполняющиеся на конвейерных функциональных устройствах.
Как правило, несколько таких процессоров (1-16) работают одновременно над общей памятью (аналогично SMP) в рамках многопроцессорных конфигураций. Несколько таких узлов могут быть объединены с помощью коммутатора (аналогично MPP).
Примеры: NEC SX-4/SX-5, линия векторно-конвейерных компьютеров CRAY: от CRAY-1, CRAY J90/T90, CRAY SV1, CRAY X1, серия Fujitsu VPP.

Кластерные системы

Набор рабочих станций (или даже ПК) общего назначения, используется в качестве дешевого варианта массивно-параллельного компьютера. Для связи узлов используется одна из стандартных сетевых технологий (Fast/Gigabit Ethernet, Myrinet) на базе шинной архитектуры или коммутатора.
При объединении в кластер компьютеров разной мощности или разной архитектуры, говорят о гетерогенных (неоднородных) кластерах.
Узлы кластера могут одновременно использоваться в качестве пользовательских рабочих станций. В случае, когда это не нужно, узлы могут быть существенно облегчены и/или установлены в стойку.
Примеры: NT-кластер в NCSA, Beowulf-кластеры.

ОПИСАНИЕ NUMA-АРХИТЕКТУРЫ.



Гибридная архитектура воплощает в себе удобства систем с общей памятью и относительную дешевизну систем с раздельной памятью.

Суть этой архитектуры - в особой организации памяти, а именно:

- Память является физически распределенной по различным частям системы, но логически разделяемой, так что пользователь видит единое адресное пространство.
- Система состоит из однородных базовых модулей (плат), состоящих из небольшого числа процессоров и блока памяти. Модули объединены с помощью высокоскоростного коммутатора.
- Поддерживается единое адресное пространство, аппаратно поддерживается доступ к удаленной памяти, т.е. к памяти других модулей. При этом доступ к локальной памяти осуществляется в несколько раз быстрее, чем к удаленной.
- По существу архитектура NUMA является MPP (массивно-параллельная архитектура) архитектурой, где в качестве отдельных вычислительных элементов берутся SMP (симметричная многопроцессорная архитектура) узлы.

ВОЗНИКНОВЕНИЕ И РАЗВИТИЕ NUMA-СИСТЕМ

Первый этап. Прототипы Numa-систем.

- Понятие NUMA-архитектуры возникло в конце 80-х годов с появлением интереса к компьютерам с распределенной памятью. Среди них выделялись так называемые DSM-машины, т. е. машины с распределенной общей памятью. В этих машинах общая память была физически разнесена по компьютеру.

- В силу этого сочетались легкость программирования для машин с общей памятью и серьезные трудности для поддержания режима общей памяти. При этом кусок общей памяти, максимально приближенный к узлу вычислительной системы, имеет время доступа в десятки раз меньше, чем к удаленным кускам. Заметим, что максимально приближенный к узлу кусок общей памяти не есть локальная память этого узла, хотя и имеет много общего с ней.

- Другая проблема, решение которой требует наличия распределенной памяти, есть проблема масштабирования. Разбитая на куски, разнесенная память позволяет организовать параллельные машины в виде сети пар процессор–память. К масштабируемым машинам этого типа относятся машины фирмы BBN (Batterfly, TC 2000, GP 2000), KSR-1 и KSR-2 фирмы Kendall Square Research, NCUBE 2, а также более поздние — IBM SP-2, Convex SPP 1200/XA, Intel Paragon, Thinking Machine CM5, IBM RP3, проект DASH (Стенфорд), проект ALEWIFE (МТИ), Horizon/Tera.

Второй этап.

Первую машину основанную на гибридной архитектуре предложил Стив Воллох и воплотил в системах серии Exemplar. Вариант Воллоха - система, состоящая из 8-ми SMP узлов. Фирма HP купила идею и реализовала на суперкомпьютерах серии SPP. Идею подхватил Сеймур Крей (Seymour R.Cray) и добавил новый элемент - когерентный кэш, создав так называемую архитектуру cc-NUMA (Cache Coherent Non-Uniform Memory Access), являющуюся 2 поколением Numa-архитектуры, которая расшифровывается как "неоднородный доступ к памяти с обеспечением когерентности кэшей". Он ее реализовал на системах типа Origin.

Организация когерентности многоуровневой иерархической памяти.

Понятие когерентности кэшей описывает тот факт, что все центральные процессоры получают одинаковые значения одних и тех же переменных в любой момент времени. Действительно, поскольку кэш-память принадлежит отдельному компьютеру, а не всей многопроцессорной системе в целом, данные, попадающие в кэш одного компьютера, могут быть недоступны другому. Чтобы избежать этого, следует провести синхронизацию информации, хранящейся в кэш-памяти процессоров.

Для обеспечения подобной когерентности кэшей существуют несколько возможностей:

- Использовать механизм отслеживания шинных запросов (snoopy bus protocol), в котором кэши отслеживают переменные, передаваемые к любому из центральных процессоров и, при необходимости, модифицируют собственные копии таких переменных.
- Выделять специальную часть памяти, отвечающую за отслеживание достоверности всех используемых копий переменных.

Третий этап.

Следующим этапом развития архитектуры NUMA является появление архитектуры NUMAFlex. 25 июля 2000 г. компания SGI анонсировала новое семейство систем SGI 3000. Эти системы являются первыми системами, построенными на основе технологии NUMAFlex. Технология NUMAFlex, которая является третьим поколением технологии NUMA, дает возможность наращивания и изменения системы вплоть до использования различных процессоров одновременно в рамках единой системы.

Основным блоком конструкции новых серверов SGI 3000 стал “кирпич” (brick); при этом кирпичи бывают разных типов, в зависимости от их содержимого. Однако основные элементы архитектуры S2MP сохранены, т.е. сохраняется то, как связываются между собой процессоры, оперативная память, концентраторы, маршрутизаторы и подсистема ввода-вывода. То, что ранее было реализовано в виде плат, “превратилось” в кирпичи, а “провода” на системной плате типа midplane заменены кабелями (таких плат в NUMAFlex больше нет).

МАШТАБИРУЕМОСТЬ.

Масштабируемость NUMA-систем ограничивается объемом адресного пространства, возможностями аппаратуры поддежки когерентности кэшей и возможностями операционной системы по управлению большим числом процессоров. На настоящий момент, максимальное число процессоров в NUMA-системах составляет 1000 (Origin3000).

СИММЕТРИЧНОСТЬ.

Симметричность имеет два важных аспекта: симметричность памяти и ввода-вывода. Память симметрична, если все процессоры совместно используют общее пространство памяти и имеют в этом пространстве доступ с одними и теми же адресами. Симметричность памяти предполагает, что все процессоры могут исполнять единственную копию ОС. В таком случае любые существующие системы и прикладные программы будут работать одинаково, независимо от числа установленных в системе процессоров. Требование симметричности ввода-вывода выполняется, если все процессоры имеют возможность доступа к одним и тем же подсистемам ввода-вывода (включая порты и контроллеры прерывания), причем любой процессор может получить прерывание от любого источника.

ОПЕРАЦИОННАЯ СИСТЕМА.

- Обычно вся система работает под управлением единой ОС, как в SMP. Но возможны также варианты динамического "подразделения" системы, когда отдельные "разделы" системы работают под управлением разных ОС (например, Windows NT и UNIX в NUMA-Q 2000).

ОСОБЕННОСТИ ПО NUMA-СИСТЕМ

- Для разработчика ПО следствием указанных выше особенностей NUMA-машин является то, что программы должны не только эксплуатировать параллелизм, но и управлять данными там, где это возможно, для исключения нелокальных ссылок; там, где нелокальные ссылки необходимы, они должны быть сгруппированы для блочной пересылки.

- Существующая компиляторная технология ориентирована большей частью на машины с *однородным* доступом к памяти, в которых одна забота — эксплуатация параллелизма. Параллельный код генерируется путем распределения среди процессоров итераций самого внешнего цикла в гнезде циклов вместе с вставкой команд синхронизации, чтобы позаботиться о зависимостях, порожденных этим циклом. Для уменьшения синхронизации применяются преобразования типа перестановки циклов для перемещения параллельных циклов в сторону самого внешнего, где это возможно. Этот подход не является универсальным; он не пригоден для генерации хорошего кода для NUMA-архитектур.

- Альтернативный подход, реализованный в языке Фортран-D, состоит в том, чтобы дать программисту управление тем, как структуры данных распределяются по процессорам. Компилятор использует информацию о *декомпозиции данных* для определения того, как распределять работу по процессорам. Один простой способ сделать это — использовать так называемое правило *собственности* — процессор исполняет оператор присваивания, если левосторонняя переменная оператора отображается в локальную память этого процессора. Процессор исполняет итерацию цикла, если он способен выполнить любую работу в теле цикла на этой итерации.

- Хотя эта стратегия принимает в расчет отображения данных, компилятор может генерировать неэффективный код, в котором все процессоры исполняют все итерации “разыскивая работу для выполнения”, если структура гнезда циклов не соответствует распределению данных. Во многих таких случаях реструктуризация цикла может улучшить качество кода, но никакой общий подход к преобразованиям цикла недопустим в этом контексте.

ДОСТОИНСТВА

- Одним из достоинств является то, что можно использовать модель программирования, аналогичную SMP, в силу этого легкость сравнима с программирования для машин с общей памятью.
- Высокая масштабируемость. В настоящее время существуют систым более чем с 1000 процессоров (SGI Origin3000)
- Система представляет собой гибрид SMP и MPP.
- Быстрое время доступа к локальной памяти.

НЕДОСТАТКИ

- Серьезные трудности связанные с управлением данными и когерентности кэшей, если она не реализованна аппаратно.
- Относительная дороговизна и сложность по сравнению с SMP системами.
- Значительное время доступа к удаленной памяти.

ПРИМЕРЫ

Наиболее известными системами архитектуры cc-NUMA являются: HP 9000 V-class в SCA-конфигурациях, SGI Origin3000, Sun HPC 15000, IBM/Sequent NUMA-Q 2000. На настоящий момент максимальное число процессоров в cc-NUMA-системах может превышать 1000 (серия Origin3000). Обычно вся система работает под управлением единой ОС, как в SMP. Возможны также варианты динамического "подразделения" системы, когда отдельные "разделы" системы работают под управлением разных ОС. При работе NUMA-системами, также как с SMP, используют так называемую парадигму программирования с общей памятью (shared memory paradigm).

Наиболее наглядный и доступный вариант представлен подсистемой памяти многопроцессорных платформ AMD Opteron и Intel Xeon, и существует он, можно сказать, с момента анонса самих процессоров AMD Opteron 200-х и 800-х серий, поддерживающих многопроцессорные конфигурации.

ИТОГИ:

Архитектура	Система состоит из однородных базовых модулей (плат), состоящих из небольшого числа процессоров и блока памяти. Модули объединены с помощью высокоскоростного коммутатора. Поддерживается единое адресное пространство, аппаратно поддерживается доступ к удаленной памяти, т.е. к памяти других модулей. При этом доступ к локальной памяти в несколько раз быстрее, чем к удаленной. В случае, если аппаратно поддерживается когерентность кэшей во всей системе (обычно это так), говорят об архитектуре cc-NUMA (cache-coherent NUMA)
Примеры	HP HP 9000 V-class в SCA-конфигурациях, SGI Origin3000, Sun HPC 10000, IBM/Sequent NUMA-Q 2000, SNI RM600.
Масштабируемость	Масштабируемость NUMA-систем ограничивается объемом адресного пространства, возможностями аппаратуры поддержки когерентности кэшей и возможностями операционной системы по управлению большим числом процессоров. На настоящий момент, максимальное число процессоров в NUMA-системах более 1000(Origin3000).ду собой, однако накладывает сильные ограничения на их число - не более 32 в реальных системах. Для построения масштабируемых систем на базе SMP используются кластерные или NUMA-архитектуры.
Операционная система	Обычно вся система работает под управлением единой ОС, как в SMP. Но возможны также варианты динамического "подразделения" системы, когда отдельные "разделы" системы работают под управлением разных ОС (например, Windows NT и UNIX в NUMA-Q 2000).
Модель программирования	Аналогично SMP