

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.451.42

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ ИНТЕГРИРОВАННОЙ СРЕДОЙ ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ВЫЧИСЛЕНИЙ «МЕТАКЛАСТЕР»

© 2010 г.

В.П. Гергель, А.В. Сенин

Нижегородский госуниверситет им. Н.И. Лобачевского

SeninAndrew@gmail.ru

Поступила в редакцию 12.04.2010

Рассмотрено текущее состояние системы управления интегрированной средой высокопроизводительных вычислений «Метакластер». Описаны основные особенности системы, компонентная архитектура и особенности реализации ключевых функций.

Ключевые слова: «Метакластер», система управления кластером, планирование параллельных задач, мониторинг вычислительных ресурсов, Microsoft High Performance Computing Server 2008.

Введение

Кластерные технологии играют лидирующую роль в мире высокопроизводительных вычислений [1]. Согласно текущему (на ноябрь 2009 г.) списку TOP 500 наиболее высокопроизводительных вычислительных систем, кластеры занимают 417 позиций, что составляет 83% от общего числа установок [2]. В большинстве случаев кластер предлагает оптимальное соотношение цена/производительность для решения широкого круга вычислительно трудоемких задач: финансовая математика, моделирование физических процессов, поиск новых лекарств и др. Кластерные системы отличаются легкостью модернизации, доступностью аппаратных составляющих, возможностью выполнения всех операций по техническому обслуживанию силами специалистов, не имеющих глубоких знаний в области высокопроизводительных вычислений. Указанные факторы привели к массовому распространению кластеров в научном сообществе, на промышленных предприятиях, и даже индивидуальные исследователи зачастую имеют возможность приобрести или собрать самостоятельно небольшой кластер.

Рост числа кластерных систем делает заманчивой идею совместного использования кластеров с целью увеличения суммарной доступной вычислительной мощности, а также уменьшения среднего времени ожидания задачей запуска. Последнее достигается за счет применения

расширенных стратегий планирования, учитывающих временные колебания загруженности различных кластеров (если один из кластеров простаивает, то на него могут быть отправлены задачи с самого загруженного в настоящий момент кластера) и одновременно гибкость при распределении ресурсов (так, например, задачи владельцев кластера могут иметь более высокий приоритет и т.п.).

Идея объединения кластеров находит свое отражение в концепции grid [3]. В настоящее время разработано несколько популярных технологий, позволяющих объединять вычислительные ресурсы в grid: Globus Toolkit, gLite, Unicore и др. Некоторые grid-системы объединяют кластеры, расположенные на разных континентах (например, World Community Grid, [4]), тысячи организаций и миллионы вычислительных узлов. Однако зачастую возникает необходимость объединить вычислительные ресурсы, принадлежащие одной организации: несколько кластеров или рабочих станций лабораторий. Если указанные ресурсы находятся физически близко друг к другу, то между ними можно установить выделенное соединение, характеристики которого будут близки к характеристикам сети внутри кластера. Это позволяет владельцу вычислительных ресурсов ожидать от полученного *мультикластера* большей эффективности, чем при объединении аналогичных ресурсов через Интернет. Для решения указанной задачи также может быть использована та или иная реализация

технологии grid. Только платой за это может стать излишняя громоздкость полученного решения, сложность администрирования, поддержки и использования конечными пользователями. Концепция grid не предполагает наличия централизованного управления, что естественно в случае объединения ресурсов разных владельцев. Но в случае одной организации такой подход вносит дополнительные издержки и затрудняет планирование.

Альтернативный подход при создании мультикластера состоит в централизованном управлении всеми подключенными вычислительными ресурсами. Такой подход может быть неприменим при управлении тысячами ресурсов, разбросанных по миру. Но в случае принадлежности всех кластеров, составляющих мультикластер, одной организации централизованное управление может быть более эффективно и естественно, так как отражает факт принадлежности ресурсов одному владельцу. Система управления интегрированной средой высокопроизводительных вычислений «Метакластер», разрабатываемая в ННГУ, основана на указанном принципе. Основная задача системы – повышение эффективности использования вычислительных ресурсов за счет балансировки нагрузки между кластерами и эффективных стратегий планирования внутри каждого кластера.

1. Общее описание системы «Метакластер»

В ходе разработки для системы «Метакластер» были подготовлены три последовательно развиваемые версии системы. Первый вариант «Метакластера» разработан в 2002 г. и был ориентирован на организацию эффективного управления отдельными кластерами при одновременном обеспечении высокой надежности и отказоустойчивости при предоставлении удаленного доступа к кластеру через Интернет. Важной особенностью разработанной системы явилась ее установка на кластеры, функционирующие под управлением операционных систем семейства Microsoft Windows. Выбор операционной системы был обусловлен желанием упростить для пользователей практическое использование высокопроизводительных кластерных систем – многие прикладные программисты имеют большой опыт работы именно в среде ОС Windows и для них разработка программ для Linux-кластеров может оказаться достаточно сложной. Кроме того, большое количество вычислительных ресурсов, которые могут быть использованы при построении кластеров, как правило, работают под управлением ОС Win-

dows. И, наконец, на момент начала разработки «Метакластера» существовало достаточное ограниченное множество доступных, надежных и простых в использовании систем управления, работающих с операционной системой Windows, а практика организации высокопроизводительных вычислений на Windows не имела широкого распространения. Как показало последующее развитие событий, выбор для «Метакластера» операционной системы Windows был обоснованным, так как это позволило привлечь к теме высокопроизводительных вычислений широкий круг пользователей. В настоящее время большинство ведущих производителей ПО для управления кластерами также поддерживают ОС Windows (Platform LSF, Condor, Microsoft HPC 2008 и др.).

Следующий вариант «Метакластера» подготовлен в 2005 г. и был ориентирован на поддержку многокластерности – возможности управлять одновременно несколькими кластерами, предоставляя к ним единую точку доступа. Это открыло еще большие возможности для пользователей, так как у них появилась возможность запускать задания на широком спектре вычислительных ресурсов, объединенных в мультикластер (или интегрированную среду высокопроизводительных вычислений), а также повысило общую эффективность использования кластеров за счет динамической балансировки нагрузки между ними.

Новейшая версия «Метакластера» разработана в 2008 г. – к возможностям системы управления добавилась кроссплатформенность (возможность управления кластерами, работающими под управлением ОС Linux) и способность включать в интегрированную среду «Метакластера» кластеры под контролем систем управления сторонних разработчиков, в частности Microsoft High Performance Computing Server 2008.

Ключевыми особенностями системы «Метакластер» являются: многокластерность, многоплатформенность, возможность интеграции с системами управления сторонних разработчиков. Рассмотрим каждую из приведенных характеристик подробнее.

1.1. Многокластерность. «Метакластер» позволяет подключать к интегрированной среде управления высокопроизводительными вычислениями произвольное число кластеров, возможно, территориально удаленных и не состоящих в одной подсети. Система берет на себя работу по доставке необходимых входных файлов на требуемый кластер и по копирова-

нию результатов вычислений на виртуальный рабочий стол пользователя. Это позволяет пользователю не задумываться о том, где физически выполняются его задачи. При временной недоступности определенного кластера, а также в случае подключения к «Метакластеру» новых ресурсов для пользователя ничего не меняется: система сама определяет, на какой ресурс эффективнее отправить задание. Пользователю нужно лишь указать требования к задаче: архитектура и число процессоров, объем оперативной памяти, свободное место на дисках и другие характеристики. При необходимости возможно и явное задание кластера и далее узлов. Но в этом случае пользователь должен будет самостоятельно перезапускать задание в случае отключения указанных ресурсов.

1.2. Многоплатформенность. «Метакластер» позволяет интегрировать кластеры, работающие под управлением различных операционных систем. Это достигается за счет специфичной реализации платформенно-зависимого кода и использования открытых протоколов коммуникации компонентов. В настоящее время реализована поддержка основных операционных систем семейств Windows и Linux. При постановке задачи пользователь выбирает операционную систему и реализацию коммуникационной среды (например, определенная реализация MPI), требующиеся его приложению. Задача «Метакластера» – подобрать соответствующие ресурсы и осуществить запуск способом, характерным для выбранной операционной системы и выбранной реализации MPI.

Гибкость в выборе операционной системы позволяет существенно расширить круг пользователей «Метакластера» и спектр решаемых задач. Однако «Метакластер» частично сохраняет свою ориентированность на работу в ОС Windows. Так, компоненты «Интегратор» (распределение задач по кластерам) и «Удаленный доступ» работают только в ОС Windows.

1.3. Интеграция со сторонними системами управления кластерами. «Метакластер» может взаимодействовать с системами управления сторонних разработчиков. При подключении такого кластера «Метакластер» возьмет на себя работу по взаимодействию с его системой управления: добавление задач, отслеживание их состояния, мониторинг вычислительных ресурсов и пр. Это позволяет подключать кластеры к интегрированной среде управления высокопроизводительными вычислениями без нарушения работы существующих пользователей, которые могут продолжать работать привычным способом. В настоящее время реализована интеграция с Microsoft High Performance Computing Server 2008.

2. Архитектура системы

Система управления «Метакластер» имеет четыре основных компонента: менеджер удаленного доступа, интегратор кластеров, менеджер кластера и инспектор узла. На рис. 1 представлена схема архитектуры разрабатываемой системы.

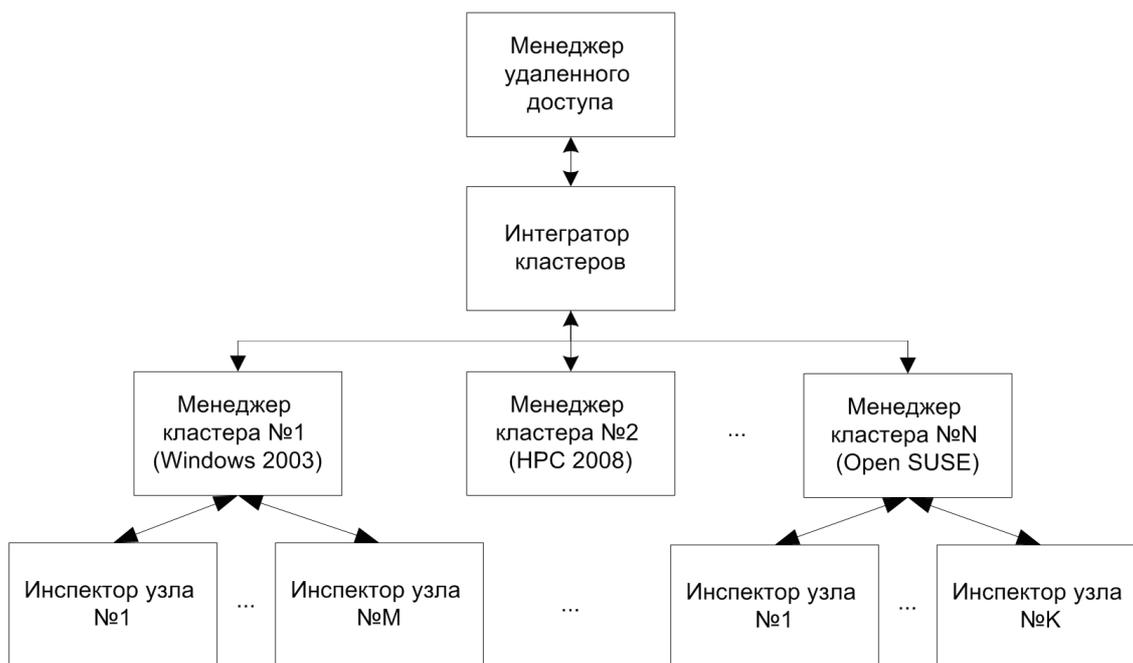


Рис. 1. Архитектура системы управления «Метакластер»

2.1. *Менеджер удаленного доступа.* Компонент «Менеджер удаленного доступа» – единая точка доступа к ресурсам под управлением «Метакластера». Доступ возможен либо через веб-службу [5], либо через веб-интерфейс. Доступ через веб-службу позволяет интегрировать «Метакластер» с пользовательскими приложениями, которым необходимо использовать ресурсы мультикластера для расчетов. Открытые стандарты, на которых основана веб-служба: XML, WSDL, SOAP, – позволяют взаимодействовать с «Метакластером», используя практически любой язык программирования и любую операционную систему.

Веб-интерфейс «Метакластера» предоставляет пользователям удобный и простой доступ к вычислительным ресурсам через Интернет и не требует установки специального программного обеспечения. После прохождения процедуры аутентификации пользователь «Метакластера» получает доступ к своему виртуальному рабочему столу, где отображаются загруженные им программы, файлы данных, а также результаты предыдущих запусков программ. Поддерживаются базовые операции управления файлами:

создание каталогов, перемещение/ копирование, переименование, архивирование/ разархивирование, загрузка/ скачивание файлов и пр. Для добавления новой задачи пользователю необходимо выбрать исполняемый файл и задать свойства создаваемой задачи: требуемые вычислительные ресурсы, входные параметры, результат перенаправления потока вывода, приоритет задачи и др. После этого за состоянием задачи можно будет наблюдать в очереди задач. Веб-интерфейс предоставляет возможность получить статистическую информацию об использовании кластера в виде различных отчетов и графиков загруженности за выбранные периоды в прошлом. Почти все указанные операции можно выполнить из пользовательских программ через использование веб-службы – фактически, веб-интерфейс является всего лишь одним из клиентов веб-службы (рис. 2).

2.2. *Интегратор кластеров.* Интегратор кластеров является центральной точкой управления мультикластером. Интегратор принимает задания от менеджера удаленного доступа, распределяет их по кластерам и отслеживает ход

Имя файла:	crane.exe
Имя задачи:	tracking3-21-45-63-19-smooth
Файл результатов:	22112010.txt
Выбрать формат запуска:	<input checked="" type="radio"/> Задать количество процессов <input type="radio"/> Задать маску запуска <input type="radio"/> Задать узлы для запуска
Количество процессов:	150
Маска запуска:	
Узлы для запуска:	
Параметры запуска:	..verbose=1
Приоритет задачи:	Высокий
Список переменных окружения:	
Кластер:	HPC2008
Режим работы:	MSMPI (HPC 2008) Задачи, которые скомпилированы с библиотекой MSMPI
<input type="button" value="Запустить"/> <input type="button" value="Отмена"/>	

Рис. 2. Диалоговое окно веб-интерфейса удаленного доступа для постановки задачи в очередь

их выполнения. Интегратор поддерживает связь с каждым из кластеров, включенных в «Мета-кластер». При этом не требуется, чтобы узлы кластера и интегратор были в одной подсети. Достаточно наличия канала связи между интегратором и менеджером кластера. Интегратор периодически проверяет доступность каждого кластера. В случае потери соединения интегратор может принять решение о перемещении заданий на другой кластер. Интегратор хранит общую очередь заданий, запущенных на всем мультикластере, отслеживает их состояние и транслирует эту информацию менеджеру удаленного доступа.

Интегратор выбирает только кластер, где будет запущено задание, решение же о выборе узлов принимает соответствующий менеджер кластера. Однако в случае когда пользователь явно указал кластер и узлы, где он хочет выполнить свое приложение, интегратор просто транслирует эту информацию менеджеру кластера.

Для коммуникации с менеджером удаленного доступа и менеджерами кластеров используется механизм .NET Remoting, обеспечивающий высокую надежность и скорость передачи данных. Наличие открытой кроссплатформенной реализации механизма .NET Remoting (проект Mono – см. [6]) позволяет просто соединять интегратор с менеджерами кластеров, работающих на Linux и других операционных системах. Сам интегратор кластеров работает только на ОС Windows.

2.3. Менеджер кластера. Компонент «Менеджер кластера» – это система управления отдельным кластером. В задачу компонента входит распределение заданий по узлам кластера в соответствии с локальной стратегией планирования, мониторинг вычислительных ресурсов кластера, обеспечение устойчивости к сбоям отдельных узлов. Менеджер кластера хранит локальную очередь заданий, состоящую из заданий, переданных интегратором, и задач, поставленных локальными пользователями. Менеджер отслеживает состояние запущенных заданий и транслирует информацию интегратору кластеров.

Специфические реализации менеджера кластера позволяют подключать к «Метакластеру» кластеры под управлением различных операционных систем, а также кластеры, где уже установлена система управления сторонних разработчиков. В последнем случае менеджер кластера выступает в роли посредника между интегратором кластеров и установленной системой

управления, преобразуя поступающие команды в форму, понятную локальной системе управления. В настоящее время реализованы версии менеджеров кластеров для Windows, Linux, а также кластеров под управлением Microsoft High Performance Computing Server 2008.

2.4. Инспектор узла. Инспектор узла – компонент, устанавливаемый на каждый узел кластера. В его задачу входит выполнение команд на удаленном узле по запросу менеджера кластера: старт и остановка процессов, перезапуск «зависших» сервисов, предоставление менеджеру кластера информации о состоянии и загрузке узла и отдельных процессов. Для коммуникации с менеджером кластера используется механизм .NET Remoting.

3. Особенности реализации системы

Одной из наиболее важных задач системы управления кластерами является обеспечение эффективного использования имеющихся вычислительных ресурсов. Только в этом случае может быть достигнута максимально возможная нагрузка вычислительной системы, что, в свою очередь, сведет к минимуму время ожидания результатов выполнения пользовательских программ. Таким образом, подсистемы планирования и мониторинга – важнейшие функции системы управления кластером. В случае мультикластера они еще более важны, так как от них требуется умение работать в гетерогенной территориально распределенной среде. Рассмотрим реализацию указанных функций в «Метакластере» более подробно.

3.1. Планирование распределения ресурсов. Планирование в системе управления «Метакластер» осуществляется на двух уровнях: сначала интегратор кластеров выбирает наиболее подходящий для задачи кластер, затем менеджер соответствующего кластера осуществляет распределение задачи по вычислительным узлам. При выборе кластера интегратор учитывает требования, указанные пользователем: необходимые ресурсы (число процессоров, память, размер дискового пространства и пр.), операционная система, необходимые программы и пакеты. Если указанным требованиям соответствуют несколько кластеров, то выбирается наименее загруженный, то есть тот, на котором ожидаемое время окончания счета задачи меньше.

В качестве алгоритма распределения на кластере применяется алгоритм обратного заполнения [7] – один из наиболее используемых и

эффективных алгоритмов планирования на кластерах. Алгоритм является модификацией метода FCFS (First Come First Served), составляющего расписание в порядке, определяющемся взвешенной суммой нескольких параметров: время ожидания задачей запуска, приоритет задания, приоритет пользователя, добавившего задание, и др. Для работы алгоритма обратного заполнения необходимо при постановке задачи указать ожидаемое время ее работы. Используя данную информацию, алгоритм планирования может ставить на выполнение менее приоритетные, но небольшие задачи раньше приоритетных крупных задач, пользуясь свободными «окнами» в расписании. Тем самым уменьшается среднее время ожидания задач в очереди.

Для пояснения работы алгоритма обратного заполнения можно рассмотреть пример составления расписания для 8 параллельных задач на 6 процессорах.

Приоритет задачи	Необходимое количество процессоров	Ожидаемое время выполнения
1	2	1
2	2	2
3	5	2
4	1	3
5	4	1
6	2	2
7	1	3
8	1	1

Следующее расписание получено методом FCFS при планировании первых 4 наиболее приоритетных задач (рис. 3).

В результате мы получаем расписание с «окнами», в которые процессоры простаивают без работы. Алгоритм обратного заполнения позволяет заполнить образовавшиеся «окна» менее приоритетными задачами, до которых не дошла очередь при работе FCFS (на практике число

планируемых задач ограничивается по количеству – в нашем примере 4 – или по времени планирования, чтобы не получилось расписание на слишком долгий период вперед, что неизбежно привело бы к его быстрому устареванию из-за несоответствия прогнозов времени выполнения и фактических времен выполнения задач). Алгоритм просматривает список задач (в том же порядке, что и FCFS) и проверяет, может ли рассматриваемая задача поместиться в какое-либо из имеющихся «окон». Если да, то для нее резервируется необходимый интервал в указанном «окне», а алгоритм переходит к рассмотрению следующей задачи. В рассматриваемом примере нашлись 2 задачи: задача 6 и задача 8, которые поместились в «окна» и, таким образом, будут выполнены раньше задачи 3, хотя последняя и имеет более высокий приоритет (рис. 4).

Менеджер кластера может также интегрироваться с Maui – популярным планировщиком заданий на кластерных системах [8]. Планировщик Maui предоставляет администратору кластера дополнительные возможности по настройке используемых политик планирования. Интеграция с Maui осуществлена посредством обмена сообщениями через сетевые сокеты по протоколу WIKI [9]. Более подробно механизм интеграции менеджера кластера с Maui представлен, например, в [10].

3.2. Мониторинг вычислительных ресурсов.

Основные задачи системы мониторинга «Метакластера» состоят в следующем:

- предоставление планировщику актуальной информации о динамических характеристиках вычислительных узлов: загрузка центрального процессора, объем свободной оперативной памяти, свободного места на жестких дисках и др.;
- графическое отображение статистической информации о загруженности кластера в веб-интерфейсе «Метакластера».

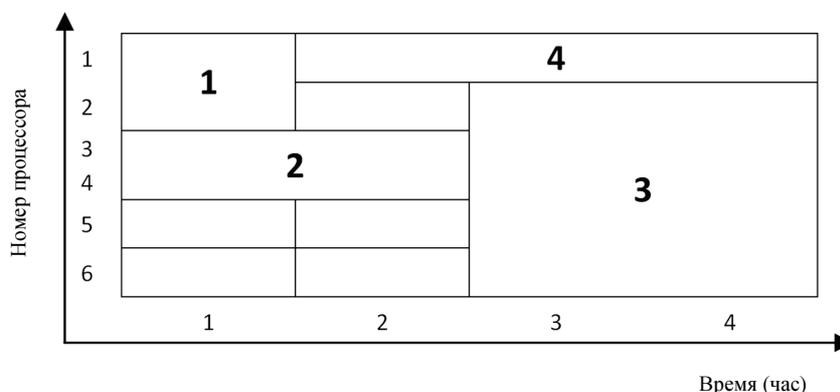


Рис. 3. Результат планирования алгоритмом FCFS

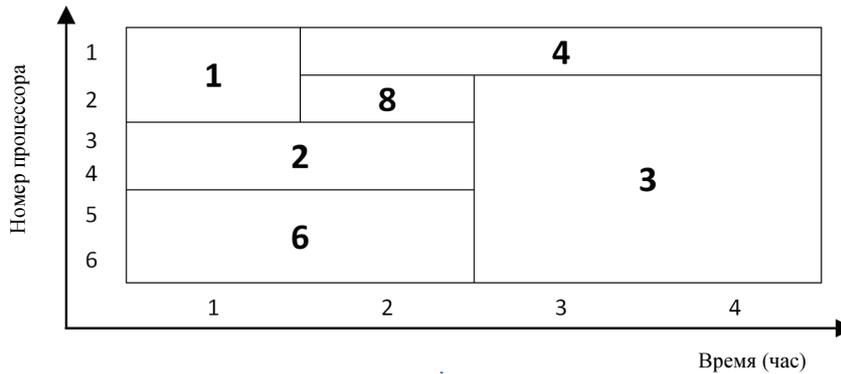


Рис. 4. Результат работы алгоритма обратного заполнения

Система мониторинга должна быть кросс-платформенной и стабильно работать на гетерогенной территориально распределенной многокластерной системе. В результате проведенных исследований было установлено, что указанным требованиям наилучшим образом удовлетворяет распределенная система мониторинга высокопроизводительных вычислительных систем Ganglia [11]. Она состоит из нескольких компонентов:

- Gmond – сервис мониторинга, устанавливаемый на каждом узле для мониторинга текущего его состояния;
- Gmetad – сервис сбора статистики, устанавливаемый на головном узле кластера для сбора и сохранения данных в циклическую базу данных. При наличии нескольких кластеров сервис может аккумулировать статистику с других сервисов gmetad;
- веб-интерфейс – визуальная часть для графического отображения статистики.

Ganglia – система с открытым исходным кодом, написанная на алгоритмическом языке C. Сервис мониторинга был портирован на широкий круг различных ОС, однако сервис сбора статистики предназначен для использования только на Unix-подобных системах. В частности, по заявлению разработчиков, сервис сбора статистики не компилируется на Windows. Тем не менее в рамках разработки системы «Метакластер» было выполнено портирование сервиса на Windows [12].

Для получения планировщиком «Метакластера» информации мониторинга была разработана библиотека, общающаяся с сервисом сбора статистики через сетевые сокет (в качестве языка сообщений используется XML) и транслирующая информацию монитора в удобный планировщику вид. Веб-интерфейс Ganglia предоставляет удобные и наглядные средства отображения информации о загрузке кластера. Веб-интерфейс Ganglia был встроены в стра-

ницу статистики «Метакластера» для отображения пользователям графиков загрузки кластера за выбранный интервал.

3.3. Управление процедурами запуска заданий. «Метакластер» не налагает на пользователей ограничений по типу запускаемых заданий. Это могут быть последовательные программы, параллельные программы, написанные с использованием различных реализаций MPI, задания для установленных на кластере пакетов. Подобная гибкость достигается за счет выноса части логики по запуску заданий в Perl-скрипты, доступные для редактирования администратором кластера. Администратор сам определяет, какие типы заданий разрешено запускать на кластере, например последовательные исполняемые файлы, скрипты на языке Python, параллельные MPICH2-программы, OpenMPI-программы, задания для Fluent. Каждому типу заданий администратор сопоставляет скрипт-обработчик.

Скрипты-обработчики принимают на вход следующие параметры:

- Имя модуля. Файл, выбранный пользователем для запуска в веб-интерфейсе «Метакластера». В случае запуска собственной программы это будет исполняемый файл программы или скрипт. В случае запуска пакета, установленного на кластере, это имя входного файла с необходимыми параметрами запуска;
- Аргументы командной строки;
- Файл для перенаправления стандартных потоков вывода и ошибок;
- Рабочая директория. Физический путь до директории, соответствующей той виртуальной папке, из которой пользователь произвел запуск через веб-интерфейс;
- Переменные окружения, установленные пользователем;

- Узлы, выделенные планировщиком для запуска.

По переданным параметрам скрипт должен сформировать полную строку команды, которая будет запущена на выполнение. Например, в случае MPICH2-программы это будет команда вида:

```
<директория установки MPI>\mpiches -hosts
<список узлов> -env <переменные окружения> <имя модуля> <аргументы командной строки> > <имя файла вывода> (часть параметров была опущена для наглядности).
```

Кроме формирования командной строки скрипт может выполнить служебные операции по подготовке к запуску задания: проверка корректности переданных параметров и проверка возможности запустить задачу, регистрация переменных окружения, запись сообщений в лог и пр. Исполняемый файл, запускаемый сгенерированной командой, должен работать до окончания работы задания. В противном случае «Метакластер» расценит это как признак завершения задания и запустит процедуру освобождения ресурсов, которая закрывает все дерево запущенных задач процессоров.

Заключение

Система управления высокопроизводительными вычислениями «Метакластер» объединяет 3 кластера Центра суперкомпьютерных технологий ННГУ: кластер из 64 двухпроцессорных двухъядерных серверов на базе Intel Xeon 3.2 ГГц, 4 Гб RAM и 2 мини-кластера по 4 и 5 узлов соответственно. Внутри кластеров используется Gigabyte Ethernet в качестве сети передачи данных.

Ресурсы под управлением «Метакластера» активно используются широким кругом сотрудников и студентов университета и другими пользователями. Была произведена интеграция «Метакластера» с проектами, выполняемыми в университете: ParaLab, Global Expert. Выполнена адаптация системы для подключения кластера в экспериментальный грид-сегмент программы «СКИФ-ГРИД». «Метакластер» используется в проекте «Разработка высокопроизводительного программного комплекса для квантово-механических расчетов и моделирования наноразмерных атомно-молекулярных систем и комплексов» [13, 14].

В «Метакластере» зарегистрировано свыше 100 пользователей, в периоды максимальной загрузки через систему запускалось свыше 500 заданий в день. «Метакластер» показал себя как простая в использовании, надежная, расширяемая система.

В настоящее время ведутся исследования по использованию распределенных кластеров в рамках одного MPI-задания, а также работы по расширению возможностей удаленного доступа.

Список литературы

1. Гергель В.П. Теория и практика параллельных вычислений. М.: Интернет-университет информационных технологий – ИНТУИТ.ру, 2007.
2. TOP500 Highlights, November 2009 [Электронный ресурс]. URL: <http://www.top500.org/lists/2009/11/highlights> (дата обращения: 15.01.2010).
3. Foster I., Kesselman C., Nick J., Tuecke S. The physiology of the grid: An open grid services architecture for distributed systems integration. 2002.
4. World Community Grid official website [Электронный ресурс]. URL: <http://www.worldcommunitygrid.org/> (дата обращения: 15.01.2010).
5. Web Services Glossary [Электронный ресурс]. URL: <http://www.w3.org/TR/ws-gloss/> (дата обращения: 15.01.2010).
6. Mono project official site [Электронный ресурс]. URL: http://www.mono-project.com/Main_Page (дата обращения: 15.01.2010).
7. Lee C. Parallel Job Scheduling Algorithms and Interfaces. San Diego, Department of Computer Science and Engineering, University of California, 2004.
8. Maui Cluster Scheduler [Электронный ресурс]. URL: <http://www.clusterresources.com/products/maui-cluster-scheduler.php>.
9. Wiki Interface Specification, version 1.2 [Электронный ресурс]. URL: <http://www.clusterresources.com/products/mwm/docs/wiki/wikiinterface.shtml>.
10. Кустикова В.Д., Сенин А.В. Интеграция системы управления кластерами «Метакластер» с планировщиком Maui // Материалы конференции «Технологии Microsoft в теории и практике программирования». Н. Новгород: Изд-во ННГУ им. Н.И. Лобачевского, 2008.
11. Massie M., Chun B., Culler D. The Ganglia Distributed Monitoring System: Design, Implementation, and Experience // Parallel Computing. 2003. Vol. 30.
12. Лозгачев И.Н., Сенин А.В. Мониторинг высокопроизводительных вычислений в рамках системы управления кластерами «Метакластер» // Материалы конференции «Технологии Microsoft в теории и практике программирования». Н. Новгород: Изд-во ННГУ, 2008.
13. Васильев В.Н., Бухановский А.В., Козлов С.А. и др. Высокопроизводительный программный комплекс моделирования наноразмерных атомно-молекулярных систем // Научно-технический вестник СПбГУ ИТМО «Технологии высокопроизводительных вычислений и компьютерного моделирования». № 54. СПб.: Университетские телекоммуникации, 2008.
14. Кустикова В.Д., Линев А.В., Лозгачев И.Н. и др. Организация взаимодействия с различными системами управления кластерами в рамках программного комплекса для квантово-механических расчетов и моделирования наноразмерных атомно-

молекулярных систем и комплексов // Материалы Девятой Международной конференции-семинара «Высокопроизводительные параллельные вычисле-

ния на кластерных системах». Владимир: Издательство Владимирского государственного университета, 2009.

**THE DEVELOPMENT OF «METACLUSTER» SYSTEM
FOR HIGH PERFORMANCE COMPUTING INTEGRATED ENVIRONMENT MANAGEMENT**

V.P. Gergel, A.V. Senin

The current state of the high performance computing integrated environment management system Metacluster is reviewed. Key features, component-based architecture and implementation details of main functions are described.

Keywords: Metacluster, cluster management system, high-performance computing, parallel task scheduling, computing resource monitoring, Microsoft High Performance Computing Server 2008.