ИССЛЕДОВАНИЕ ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ПЛАНИРОВЩИКА MAUI В СОСТАВЕ СУПЗ МВС-1000

А.В. Баранов, Д.М. Голинка

Начиная с 1994 года силами НИИ "Квант", ряда институтов РАН и МСЦ РАН развивается архитектура отечественных супер-ЭВМ типа МВС 1000, к которой принадлежат известные высокопроизводительные вычислительные установки МВС-1000М, МВС-15000, МВС-100К.

Архитектура МВС-1000 подразумевает, что вычислительная система используется в многопользовательском режиме работы. На вход системы поступает поток параллельных задач от разных пользователей. Каждая из поступающих параллельных задач требует для своего исполнения определенный параллельный ресурс. Функция управления задачами и ресурсами возложена на систему управления прохождением задач (СУПЗ)[1], которая обеспечивает:

- прием входного потока параллельных задач от разных пользователей;
- выделение для параллельной задачи требуемого параллельного ресурса;
- при невозможности немедленного выделения требуемого ресурса - ведение очереди задач;
- освобождение любого занятого любой задачей ресурса в любой момент времени;
- удовлетворение требованиям защиты от НСД.

Важнейшей компонентой СУПЗ является сервер очередей, который в настоящий момент реализует алгоритм приоритетного планирования задач. Алгоритм обеспечивает эффективное распределение задач по ресурсам вычислительной системы, при этом администратор МВС-1000 имеет возможность гибкой настройки параметров планирования.

Структура сервера очередей складывалась исторически, алгоритм планирования постоянно совершенствовался и усложнялся. В настоящее время возможности модификации существующего алгоритма практически исчерпаны. Алгоритм планирования жестко встроен в сервер очередей СУПЗ, и невозможно стандартным образом подключить внешний планировщик.

В то же время существуют сторонние программные разработки, решающие задачи планирования очередей. Наиболее известной из них является планировщик Maui, разработанный в Maui High Performance Computing Center [2, 3]. Maui может подключаться к различным системам пакетной обработки (СПО), к числу которых относятся OpenPBS/Torque, LoadLeveler, LSF.

Перечислим некоторые преимущества Maui, делающие этот планировщик весьма привлекательной разработкой [2].

1. Алгоритм обратного заполнения (backfill) и справедливого распределения ресурсов для повышения эффективности системы и уменьшения времени ожидания задачи в очереди.
2. Система автоматического определения приоритетов заданий.
3. Расширенная статистика. С помощью Maui администратор может получать полную историческую и текущую информацию о заданиях, очередях, планировщике, состоянии системы.
4. Возможность моделирования работы СПО, с помощью которой можно проверить настройки планировщика, позволяющие подобрать конфигурацию системы, наиболее полно отвечающую требованиям конкретной производственной обстановки.
5. Maui обладает собственным командным интерфейсом, позволяющим осуществлять внешнее административное управление планировщиком.

Указанные преимущества позволяют сделать вывод о целесообразности исследования возможности подключения Maui к СУПЗ МВС 1000.

Поскольку задача носит исследовательский характер, то заранее была неизвестна ее трудоемкость. В процессе решения пришлось столкнуться со следующими трудностями:

- информация по интеграции планировщика Maui в СПО разрозненна и неточна;
- интерфейсы Maui плохо документированы, либо вообще не документированы;
- удобочитаемость исходных текстов Maui находится на невысоком уровне;
- Maui применяется в сравнительно узком кругу специалистов, поэтому информация в сети Интернет по преодолению возникших трудностей отсутствует.

Для взаимодействия успешного взаимодействия Maui и СУПЗ необходим специальный механизм, который позволял бы обмениваться данными и осуществлять управляющие воздействия. В частности:

1. Планировщик должен получать информацию о наличии вычислительных узлов и их ресурсах.

2. Планировщик должен информироваться об изменении состояния вычислительных узлов.

3. Планировщик должен получать информацию о наличии вновь поступивших задачах и их требованиях.

4. Планировщик должен информироваться об изменении состояния задач.

5. Планировщик на основе полученной информации должен выдавать команды для СУПЗ на постановку, снятие с выполнения задач.

6. СУПЗ должна иметь возможность подтвердить или сообщить об ошибке выполнения полученной команды.

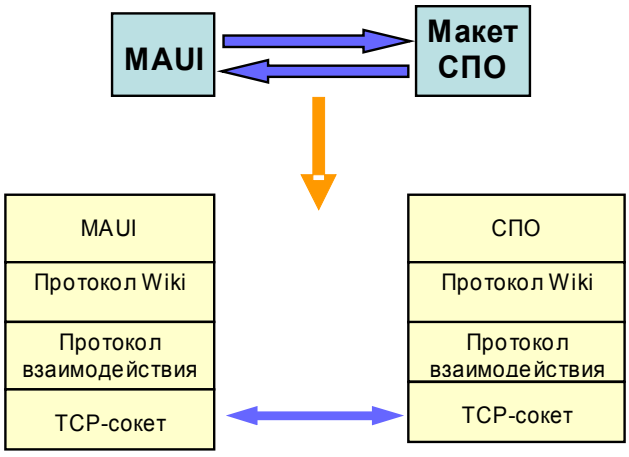
Напомним, что для ряда СПО (OpenPBS/Torque, LoadLeveler, SGE, LSF) уже реализованы интерфейсы с Maui. Недостатком этих интерфейсов является их ориентация на конкретную СПО и отсутствие документации. В то же время разработчиками Maui предусмотрен универсальный документированный интерфейс взаимодействия с СПО - Wiki, но авторами не было найдено примеров успешного использования этого интерфейса третьей стороной.

Первоначально исследование пошло по следующему пути. В качестве основы было решено рассмотреть связку Maui-OpenPBS/Torque, т.к. данный вариант на практике доказал свою стабильную работоспособность. Было выдвинуто предположение, что существует четкий интерфейсный разрез между Maui и Torque, через который взаимодействуют эти системы. В этом случае задача сводилась к выявлению интерфейсного разреза и его поддержке со стороны СУПЗ МВС-1000. Другими словами, система OpenPBS/Torque заменялась на СУПЗ с сохранением интерфейса с Maui. При этом Maui, работая в уже в составе СУПЗ, "считал" бы, что находится в составе OpenPBS/Torque.

К сожалению, данный путь привел в тупик. Отсутствие документации на интерфейсный модуль Maui-OpenPBS заставило обратиться к исходным текстам, исследование которых показало, что четкого интерфейсного разреза между Maui и OpenPBS/Torque не существует. Более того, системы являются настолько сильно интегрированными друг в друга, что, например, Maui в составе OpenPBS/Torque имеет возможность непосредственного управления вычислительными модулями многопроцессорного массива. При этом во время трансляции исходных текстов Maui для включения поддержки интерфейса OpenPBS/Torque необходимо наличие исходных кодов OpenPBS/Torque.

После этого внимание авторов было сосредоточено на универсальном интерфейсе Wiki [4]. Хотя существует документация, содержащая описание этого интерфейса, в процессе работы были выявлены ошибки в документации и недокументированные особенности работы интерфейса Wiki.

В результате удалось "разобрать" логику работы интерфейса Wiki и построить макет СПО, взаимодействующий с Maui с помощью данного интерфейса. Схема взаимодействия приведена на рисунке.



Сообщения интерфейса Wiki инкапсулируются в сообщения протокола взаимодействия [5], которые и передаются между СПО и Maui через установленное TCP-соединение. Для защиты протокола используется вычисление контрольной суммы сообщения по алгоритму DES.

Порядок взаимодействия следующий:

1. СПО-сервер ожидает соединения;
2. Maui устанавливает соединение с сервером;
3. Maui передает команду серверу;
4. Сервер выполняет команду и посылает ответ Maui;
5. Maui закрывает соединение.

Соединение всегда инициирует Maui, взаимодействие происходит итерационно. Maui с определенным интервалом времени опрашивает сервер о наличии вычислительных узлов (ресурсов) и наличии задач, посылая команды GETJOBS, GETNODES и ожидая ответа, содержащего запрошенную информацию.

Команды протокола Wiki можно разделить на две категории:

- опрос состояния;
- управление заданиями.

К команда опроса состояния относятся команды:

- GETJOBS;
- GETNODES.

В ответ этим командам должен быть отправлен список задач или вычислительных узлов соответственно.

Примером команд управления заданиями могут служить команды STARTJOB (поставить задачу на выполнение) и CANCELJOB (снять задачу с исполнения).

С помощью созданного макета СПО удалось практически подтвердить работоспособность интерфейса Wiki. Ближайшей перспективой работы является окончательная интеграция (с помощью интерфейса Wiki) планировщика Maui в состав СУПЗ МВС-1000.

ЛИТЕРАТУРА:

1. Руководство пользователя системы СУПЗ МВС-1000М // http://www.jscc.ru/informat/1000MUsrGuide.zip
2. Коваленко В.Н., Орлов А.В. Управление заданиями в распределенной среде и протокол резервирования ресурсов // http://www.keldysh.ru/papers/2002/prep1/prep2002_1.html#Toc535316030
3. Maui cluster scheduler // http://www.clusterresources.com/pages/products/maui-cluster-scheduler.php
4. Wiki Interface Specification, version 1.1 // http://www.clusterresources.com/products/maui/docs/wiki/wikiinterface.shtml
5. Maui Scheduler Socket Protocol Description // http://www.clusterresources.com/products/maui/docs/wiki/socket.shtml