ВСЕСТОРОННИЙ АНАЛИЗ КАЧЕСТВА РАБОТЫ БОЛЬШИХ СУПЕРКОМПЬЮТЕРНЫХ КОМПЛЕКСОВ

В. В. Воеводин¹

В настоящее время проблема недостаточной эффективности работы суперкомпьютерных комплексов во многом связана с тем, что администраторы таких систем не всегда могут своевременно обнаруживать и устранять причины снижения эффективности. Это в большей степени касается не выхода из строя оборудования (такие случаи можно отслеживать с помощью систем мониторинга), а неявного снижения эффективности работы определенных компонентов суперкомпьютера при условии, что внешне они продолжают работать корректно. Возникновение подобной ситуации связано с тем, что на данный момент нет достаточно гибких и удобных средств для оперативного и комплексного анализа всех характеристик качества работы вычислительных систем. Существующие решения либо позволяют анализировать только небольшую часть таких характеристик, либо представляют собой не универсальные решения, удовлетворяющие только небольшой набор конкретных нужд администраторов определенной системы. В настоящей статье описывается системный подход к решению этого вопроса, который позволит проводить комплексный анализ различных аспектов работы суперкомпьютеров, связанных в первую очередь с выполнением суперкомпьютерных приложений. Разрабатываемый на основе этого подхода программный инструмент предназначен для сбора в рамках единой модели всех наиболее важных данных о свойствах и качестве выполняющихся на суперкомпьютере задач — данные об эффективности их выполнения, размере и длительности, наличии характерных или аномальных сценариев поведения, использовании прикладных пакетов и библиотек и др. С помощью гибких возможностей по агрегации будет задаваться нужная степень детализации, с какой необходимо предоставлять полученную информацию — по отдельным пользователям, проектам, прикладным пакетам, предметным областям, разделам суперкомпьютера, временным диапазонам и др. Это позволит создавать сотни и тысячи различных представлений для анализа состояния суперкомпьютера, что поможет администраторам выбирать наиболее подходящий для них вариант.

Ключевые слова: суперкомпьютер, параллельные вычисления, суперкомпьютерные приложения, производительность, анализ эффективности, данные мониторинга.

1. Введение. Потенциал современных суперкомпьютерных систем огромен, однако эффективность их функционирования зачастую очень низка [1]. Причины могут быть самые разные — низкая производительность выполняемых приложений, неоптимальная работа менеджера ресурсов, некорректная настройка прикладных пакетов и многие другие, причем чаще всего имеет место совокупность таких причин. В целом же число факторов, которые могут влиять на подобную эффективность, очень велико. Это усугубляется чрезвычайно высокой сложностью строения суперкомпьютерных систем, что, с одной стороны, позволяет увеличивать производительность систем, но одновременно ведет и к дальнейшему снижению эффективности их использования.

В результате существенная часть суперкомпьютерных ресурсов либо простаивает, либо используется неэффективно. При этом пользователи, администраторы и руководство вычислительной системы зачастую не знают, да и не могут определить, в чем причина такого поведения. Нередко они даже не знают, есть ли какие-то проблемы с эффективностью в целом и насколько они критичны. И подобная ситуация характерна практически для всех больших суперкомпьютеров мира.

Для решения этой проблемы необходимо иметь полную информацию и о работе суперкомпьютера в целом, и о его программных и аппаратных компонентах в отдельности. Такая информация должна давать возможность изучать различные аспекты работы суперкомпьютера за любые интервалы времени — общую утилизацию вычислительных ресурсов; интенсивность использования коммуникационной сети или подсистемы памяти; статистику по использованию различных технологий параллельного программирования;

¹ Московский государственный университет им. М. В. Ломоносова, Научно-исследовательский вычислительный центр, Ленинские горы, 119991, Москва; ст. науч. сотр., e-mail: vadim@parallel.ru

динамику изменения вычислительной нагрузки на узлах; востребованность установленных прикладных пакетов; свойства задач, работающих с ускорителями; активность пользователей из различных предметных областей; типовой характер нагрузки на файловую систему и многое другое. При этом должна быть возможность исследовать эти аспекты с разной степенью детализации — на уровне отдельных пользователей, проектов, организаций, прикладных программных пакетов, предметных областей, разделов суперкомпьютеров. Подобный всесторонний анализ позволит администраторам вычислительных систем оценивать как общие свойства и параметры функционирования суперкомпьютера в целом, так и определять локальные причины возникновения тех или иных особенностей его работы. Как следствие, это позволит, например, оптимизировать квоты и политики предоставления ресурсов, оценивать востребованность прикладного программного обеспечения, планировать изменения в конфигурации суперкомпьютера, а также осознанно выполнять множество других действий для постоянного обеспечения его эффективной работы.

В настоящее время существуют различные работы, направленные на исследование качества функционирования вычислительных систем. Однако одна часть таких работ направлена на изучение только определенных аспектов поведения — например, работы [2–4] направлены только на обнаружение аномального поведения вычислительных задач в облачных средах, а работы [5, 6] помогают определять аномальное состояние отдельных вычислительных узлов. Другие работы могут подходить к этому вопросу более глобально, изучая совокупность некоторых аспектов поведения вычислительных систем (например, [7]), но при этом они являются непереносимыми, поскольку нацелены на изучение только одной определенной вычислительной системы. Таким образом, существующих исследований недостаточно для осуществления всестороннего анализа, поскольку отсутствует возможность получить единое и полное представление о работе суперкомпьютерного комплекса (СК).

В настоящей статье представлен проект, разрабатываемый в НИВЦ МГУ, который нацелен на создание и реализацию подходов, методов и инструментальных средств, позволяющих собирать воедино всю основную информацию об эффективности работы суперкомпьютера и сформировать наиболее важные сценарии использования этой информации для осуществления всестороннего анализа качества работы суперкомпьютеров. С помощью термина “всесторонний” мы делаем акцент на том, что влияние всех значимых причин снижения качества работы суперкомпьютеров можно будет увидеть и проанализировать путем применения создаваемых средств. Таким образом, глобальная цель данного проекта — это всесторонний анализ качества и повышение эффективности работы суперкомпьютеров, для чего будет разработан программный инструментальный, доступный администраторам и руководству суперкомпьютерных центров (СКЦ).

Разработанное решение позволит отвечать на множество конкретных возникающих вопросов. С какой интенсивностью приложения, построенные на основе широко распространенного пакета GROMACS (Groningen Machine for Chemical Simulations) [8], используют коммуникационную сеть Infiniband и какова зависимость интенсивности от числа используемых узлов суперкомпьютера? Участники какой научной группы наиболее активно используют пакет VASP (Vienna Ab initio Simulation Package)? Насколько часто пользователи задействуют технологию OpenMP совместно с MPI и каково распределение используемого числа потоков на узлах? Какой из компиляторов (и какая версия данного компилятора) наиболее распространен? Часто ли пользователи запускают самостоятельно собранные прикладные пакеты или библиотеки? Существующие решения зачастую не позволяют автоматически получать подобную информацию и требуют немалой ручной работы от администраторов, что заметно затрудняет процесс эффективного сопровождения суперкомпьютерных систем.

Разработка и апробация предложенного решения будет производиться в рамках Суперкомпьютерного центра МГУ. Однако одним из главных требований к этому решению является его переносимость, что позволит применять созданный программный инструмент и на других вычислительных системах. Отметим, что указанная проблема не так актуальна для небольших и гомогенных систем, поскольку число различных компонентов и их взаимосвязь в таких системах невелико, и это позволяет выполнять анализ качества их работы с помощью достаточно простых инструментов. Однако для больших систем, особенно для суперкомпьютеров петафлопсного уровня производительности и выше, решение подобной задачи чрезвычайно актуально. Апробацию разработанного решения предполагается провести на суперкомпьютере Ломоносов-2.

Создание программного инструмента для проведения всестороннего анализа качества работы суперкомпьютеров требует реализации нескольких важных задач:

- интеграция разнородных данных о производительности компонентов СК,
- разработка модели для определения качества работы суперкомпьютерных систем,

- практическая реализация предложенных методов и подходов.

Основные разделы настоящей статьи посвящены описанию указанных задач. В разделе 2 приведено описание источников информации, которые планируется использовать, а также особенности их интеграции и представления. Раздел 3 посвящен описанию методов анализа собранной информации, в то время как раздел 4 кратко описывает особенности реализации предложенных идей в будущем. В разделе 5 приведено заключение, в котором обсуждаются планы на ближайшее будущее по реализации описанных идей и методов.

2. Сбор и интеграция доступной информации о качестве работы суперкомпьютера. Вся работа суперкомпьютерного комплекса направлена на эффективное выполнение задач, запущенных пользователями для проведения численных экспериментов. Таким образом, анализ качества работы суперкомпьютера должен быть построен вокруг изучения производительности, свойств и поведения выполняющихся на нем задач. Вследствие этого, основными входными данными для проведения всестороннего анализа качества работы являются различные данные о производительности выполнения задач на вычислительных узлах, а также информация об эффективности использования остальной вычислительной инфраструктуры СК. Можно выделить следующий набор входных данных, которые планируется изучать в рамках данной работы:

- параметры выполняющихся задач: имя исполняемого файла, логин пользователя, время начала и конца выполнения задачи, название использованного раздела суперкомпьютера, список используемых библиотек и др.;
- данные о производительности выполняющихся задач: загрузка процессоров, интенсивность передачи данных по сети, производительность работы с памятью, загрузка ускорителей;
- информация о качестве работы компонентов вычислительной инфраструктуры: производительность файловой системы, характер использования коммуникационной сети, качество работы других служебных серверов;
- аналитические данные по задачам, полученные с помощью внешних инструментов анализа производительности; эти инструменты направлены, в частности, на обнаружение задач с аномально неэффективным поведением, выявление различных вариантов типового поведения задач (например, задач с низкой локальностью обращений в память и задач с малым числом активных процессов на узел).

Важный вопрос заключается в грамотной интеграции и организации всей собранной входной информации. Существенная сложность здесь состоит в разнородности источников входных данных. Данные о производительности задач предоставляет система мониторинга. В то же время для получения данных о параметрах задач необходимо взаимодействие как с менеджером ресурсов, так и с другим системным ПО. Например, для сбора информации по каждой задаче о списке используемых библиотек, параметрах компиляции или значениях переменных окружения предполагается задействовать установленную на суперкомпьютере Ломоносов-2 систему XALT [9], которая за счет подмены скриптов запуска и компиляции позволяет отслеживать нужные параметры для отдельных задач.

Далее, часть данных о работе компонентов вычислительной инфраструктуры может собираться также системой мониторинга, однако для получения подробной информации необходимо задействовать интерфейсы работы напрямую с каждым конкретным компонентом (например, с файловой системой), которые в большинстве своем устроены по-разному и являются независимыми. А получение аналитических данных (последний пункт из списка выше) целиком завязано на работу сторонних источников. Эти источники могут предоставлять более “интеллектуальную” информацию о свойствах и производительности выполнения задач. Отметим, что такие внешние инструменты разрабатываются в рамках других проектов. В частности, в НИВЦ МГУ разработан программный инструмент для обнаружения аномально неэффективных запусков программ [10], а также ведется разработка комплексной системы для массового анализа эффективности приложений [11].

Поскольку полученное решение должно масштабироваться на системы петафлопсного уровня производительности, необходимо учитывать, что источников данных может быть очень много. В частности, при апробации на суперкомпьютере Ломоносов-2 нужно будет получать и обрабатывать данные системного мониторинга с ~ 1500 вычислительных узлов и анализировать качество выполнения 500–1000 задач в сутки. При этом также необходимо учитывать, что в целом в Суперкомпьютерном центре МГУ более

2500 пользователей из 40 областей науки, работающих в рамках ~ 400 проектов и представляющих ~ 300 организаций.

Помимо большого набора входных данных, которые необходимо рассматривать, существуют различные уровни детализации собранной информации, что позволяет исследовать поведение разных сущностей. В данном проекте предполагается проводить анализ следующих сущностей:

- отдельный запуск задачи на суперкомпьютере;
- пользователь суперкомпьютера;
- прикладной пакет, используемый различными пользователями для решения задач из определенной предметной области;
- проект (включает нескольких пользователей), направленный на решение определенной задачи;
- организация, к которой принадлежат различные пользователи СК (например, МГУ им. М. В. Ломоносова или Институт прикладной физики РАН);
- предметная область, к которой принадлежат различные проекты (например, физика конденсированных сред или фундаментальная медицина);
- весь суперкомпьютер целиком.

Разрабатываемый инструмент будет предоставлять возможность выполнять анализ качества работы для любого из указанных типов сущностей.

Далее, форматы представления выбранных данных могут быть разные. В некоторых случаях представляет интерес анализ исторических данных за длительный период (например, распределение затраченных ядро-часов по различным предметным областям за последние 12 месяцев). В других ситуациях может быть полезным сравнение показателей для различных сущностей одного типа, в частности построение списка прикладных пакетов, наиболее интенсивно использующих коммуникационную сеть в течение последнего месяца. А иногда достаточно анализа всего одного интегрального значения — например, для грубой оценки производительности в рамках всего суперкомпьютера в определенный момент времени можно отслеживать среднее число выполненных инструкций за такт (instructions per cycle, IPC).

Все эти возможные варианты предполагается реализовать в рамках предлагаемого решения. Таким образом, разрабатываемый инструмент позволит получать огромное количество самых различных “снимков” состояния (т.е. отдельных сценариев использования указанных входных данных для выбранных сущностей) всей вычислительной системы в целом или отдельных его компонентов, что позволит выполнять тотальный анализ разных аспектов работы СК.

3. Разработка модели для определения качества работы суперкомпьютерных систем. Для того чтобы собранная в рамках первой задачи информация была действительно полезной, необходимо разработать адекватную модель, определяющую качество работы суперкомпьютерных систем — т.е. нужно понять, какая именно структура и способ представления данных об определенном аспекте работы СК помогут быстро и качественно провести его анализ. Для этого необходимо разработать различные методы анализа отдельных аспектов, на что в целом и направлена вторая задача, решаемая в рассматриваемом исследовании. В каждом случае необходимо определить, какое сочетание выбранных входных данных, уровня детализации и форматов представления может быть оптимальным. Для выполнения этой задачи потребуются провести подробный анализ целей и задач, решаемых администраторами суперкомпьютеров на практике, для того чтобы определить наиболее удобные для них “снимки” состояния вычислительной системы.

Разработка полноценной модели еще не завершена, однако уже сейчас можно выделить некоторые наиболее важные “снимки” состояния СК, которые будут формировать основу для данной модели. Следует отметить, что на текущий момент реализацию подобных “снимков” можно встретить и в других СКЦ (например, [12] или [7]), однако при этом они создаются вручную и/или только для данного СКЦ, в то время как в рамках данной работы такое создание планируется выполнять с помощью универсального подхода и если не в автоматическом, то в автоматизированном режиме.

3.1. Описание сценариев использования данных о качестве работы суперкомпьютера. Далее приведено описание некоторых наиболее важных и интересных “снимков” — сценариев использования собранных данных о состоянии суперкомпьютерных центров. С учетом нашего накопленного опыта сопровождения СКЦ можно утверждать, что реализация подобного функционала будет очень полезна

для администраторов и руководства СКЦ, поскольку позволяет оперативно анализировать качество его работы.

Сводная таблица по пользователям. Этот “снимок” направлен на отображение наиболее важной информации об активности пользователей и эффективности утилизации ресурсов в их задачах. Анализ такого аспекта работы СКЦ должен позволять быстро определять, соответствует ли ожиданиям эффективность работы наиболее активных пользователей, насколько оптимально работают политики и квоты выделения ресурсов и т.д.

Пример возможного визуального представления подобной информации приведен на рис. 1 (здесь и далее примеры были получены на суперкомпьютере Ломоносов-2). Каждая строка — это один пользователь; столбцы соответствуют различным характеристикам, которые оцениваются для пользователей. Число в ячейке обозначает место, которое занимает данный пользователь в рейтинге по данной характеристике (вместо относительного ранжирования пользователей можно показывать абсолютные значения характеристик); цвет — насколько низко место пользователя в рейтинге. Как видно из рисунка, здесь отображается, например, информация о числе задач и объеме затраченных каждым пользователем ядро-часов, средние значения по основным динамическим характеристикам производительности (интенсивности использования сети Infiniband и работе с памятью), а также данные о доле задач различных классов. Рассматривается, в частности, класс “подходящих для суперкомпьютера” задач — задач, поведение которых практически оптимально для запуска на суперкомпьютере с нашей точки зрения (высокая утилизация вычислительных узлов и высокая интенсивность использования коммуникационной сети).

Логин	Число задач	Число ядро-часов	% от всех ядро-часов	Среднее число ядер	% последовательных задач	% ядро-часов последовательных задач	% задач на <=14 ядрах	% ядро-часов задач на <=14 ядрах	% числа "подходящих для суперкомпьютера" задач	% ядро-часов "подходящих для суперкомпьютера" задач	Среднее Loadavg	Интенсивность получения данных по IB	Интенсивность передачи данных по IB	Число операций записи в память	Частота кэш-промахов в L1	Частота кэш-промахов в L2	Частота кэш-промахов в L3	
	1 (131)	12	12	40	8	8	39	33	28	28	52	40	5	5	24	30	8	
	2 (67)	5	5	32	8	8	42	38	26	21	47	12	1	24	32	75	51	
	10 (23)	10	10	30	8	8	37	34	11	12	54	7	4	54	4	9	44	
	10 (23)	20	20	31	8	8	43	43	28	28	82	51	39	84	67	41	7	
	12 (22)	6	6	25	8	8	40	41	28	28	15	39	17	6	37	19	35	

Рис. 1. Сводная таблица по пользователям

С помощью данной таблицы можно, например, быстро оценить, равномерно ли используются ресурсы различными активными пользователями СК, насколько активно они задействуют Infiniband сеть и др. Подобное представление позволяет администраторам в рамках единой таблицы быстро оценить наличие серьезных проблем с эффективностью.

Разбор структуры потока задач. Другой сценарий использования данных о состоянии суперкомпьютера — детальное рассмотрение структуры потока задач. Для этого необходимо уметь классифицировать задачи в потоке, т.е. определять их принадлежность к различным типам в зависимости от их поведения. Подходы к классификации могут быть самые разные, в нашем случае мы предлагаем выделять различные особенности поведения задач на основе данных мониторинга. Примеры характерных типов задач:

- “подходящие для суперкомпьютера” задачи (описаны ранее);
- “необщительные” задачи, которые были запущены на большом числе узлов, но при этом практически не задействуют коммуникационную сеть, что является нежелательным поведением на СК;
- задачи с подозрительно слабой утилизацией всех ресурсов; надо стараться по возможности уменьшать долю подобных задач (например, путем повышения их эффективности), поскольку при их выполнении большая часть вычислительных ресурсов простаивает;
- последовательные задачи; наличие существенной доли таких задач тоже нежелательно, поскольку для их выполнения зачастую больше подходят другие типы вычислений, например с использованием облачных технологий;
- высокопроизводительные гибридные задачи, которые показывают высокую утилизацию как обычных процессоров, так и графических ускорителей;

- задачи с низкой локальностью обращений в память — обнаружение таких задач позволяет отслеживать проблемы при взаимодействии с подсистемой памяти.

Подобная информация важна для администраторов, поскольку позволяет получить статистику о том, задачи какого типа чаще всего запускаются на их системе, что помогает, например, соответствующим образом подстраивать политики использования ресурсов. Наш опыт сопровождения суперкомпьютеров говорит о том, что удобнее всего отслеживать динамику изменения числа задач каждого интересующего типа за последние несколько дней/недель, поскольку это позволяет оперативно определить нежелательные всплески. Появление таких всплесков — существенного роста числа задач “плохих” типов — говорит о вероятной проблеме с эффективностью, которая требует оперативного вмешательства администраторов.

Другой интересный способ классификации задач заключается в выделении типовых шаблонов вычислений и коммуникаций. Самый известный способ такой классификации представлен в работе [13], в которой выделяется 13 “гномов” (dwarfs) — алгоритмических методов с различными шаблонами поведения. Среди таких “гномов” выделяют, например, методы решения задач линейной алгебры с плотными матрицами, спектральные методы или с использованием структурированных сеток, методы Монте-Карло, обходы графов и др. Определение принадлежности выполняемых задач к подобным характерным шаблонам может очень многое рассказать о структуре потока задач. В частности, это позволяет понять, насколько востребованы коммуникационные сети с высокой пропускной способностью, какая топология сети может быть наиболее оптимальной, насколько важна быстрая память и др. Однако основная сложность здесь заключается в том, что определить принадлежность задачи к определенному шаблону может лишь пользователь — создатель данной программы, поскольку для этого необходима информация о внутреннем строении задачи, которую невозможно получить только лишь на основе данных мониторинга.

Рис. 2. Изменение числа запусков прикладных пакетов с течением времени

Качество использования прикладных пакетов. В настоящее время очень большой процент выполняющихся задач составляют запуски готовых прикладных пакетов. В суперкомпьютерном центре МГУ установлено и используется множество пакетов для решений задач из различных предметных областей — молекулярной динамики, астрофизики, линейной алгебры, вычислительной химии и др. Поэтому отдельный интерес представляет изучение качества использования таких программных пакетов.

В первую очередь нас интересует эффективность их использования. Проблемы с эффективностью при работе с пакетом могут возникнуть, например, после неаккуратного обновления его самого или каких-то библиотек, которые он использует. Для обнаружения подобных случаев надо отслеживать динамику изменения эффективности — если в определенный момент произошло достаточно заметное падение средней эффективности среди всех пользователей данного пакета, скорее всего, возникла подобная проблема. Кроме того, изучение статистики по использованию пакетов позволяет обнаруживать пользователей, которые существенно менее эффективно работают с данным пакетом. Во многих случаях это говорит о том, что пользователь не обладает достаточными знаниями об особенностях работы с ним, поэтому задачей администратора является обнаружение таких пользователей и помощь им в правильной настройке пакетов.

Помимо анализа эффективности важно также изучать и общие тенденции по активности работы с пакетами. Это позволяет адекватно оценивать востребованность пакетов: например, если на какой-то прикладной пакет куплена лицензия на 100 человек, однако было обнаружено, что за последние полгода его

использовали только 30 пользователей, стоит задуматься о приобретении менее дорогостоящей лицензии в будущем. Кроме того, нужно отслеживать и динамику изменения числа запусков задач с использованием определенных пакетов — если за последнее время число запусков резко и существенно снизилось, может быть произошел какой-то сбой в работе пакета (или в программно-аппаратном окружении, которое данный пакет использует), который мешает его нормальной работе. Наш опыт сопровождения СКЦ показал, что динамику использования пакетов удобно отслеживать, оценивая относительное изменение числа запусков, например в формате, представленном на рис. 2.

На этом рисунке показано изменение числа запусков по 4 пакетам за 8 дней работы суперкомпьютера Ломоносов-2. “Unknown” обозначает отдельную категорию задач, куда попадают все задачи, принадлежность которых к пакетам не удалось определить (в абсолютном большинстве случаев это запуски самописных программ пользователей). Выделяется 5 категорий изменения числа запусков по сравнению с предыдущим днем (5 верхних горизонтальных линий на рисунке), а также отдельно выделяется случай, когда запусков мало (в данном случае — менее 10 запусков за день). Подобные графики можно строить и для оценки затраченных процессорно-часов, и для числа уникальных пользователей пакетов. Всю эту функциональность планируется заложить при разработке модели, определяющей качество работы суперкомпьютерных систем.

Рис. 3. Распределение числа запусков задач в зависимости от эффективности использования графических ускорителей (данные за одну неделю работы суперкомпьютера Ломоносов-2)

Активность использования отдельных компонентов СК. Достаточно часто администратору или руководству суперкомпьютерного центра необходимо изучить качество функционирования отдельных программных или аппаратных компонентов системы. Это может касаться многих компонентов вычислительной инфраструктуры, в частности вычислительных узлов, коммуникационной сети, файловой системы и т.д. Например, для суперкомпьютера Ломоносов-2 отдельный важный вопрос заключается в качестве использования графических ускорителей. Необходимо иметь возможность быстро оценить, насколько активно пользователи задействуют ускорители и какова в среднем эффективность их использования.

На рис. 3 приведена подобная статистика, собранная за одну неделю в начале 2019 г. на суперкомпьютере Ломоносов-2 (здесь не учитывались короткие задачи, а также задачи, которые не использовали GPU вообще). Можно увидеть, что многие задачи слабо используют GPU — максимальное значение у самого левого столбца, который соответствует средней эффективности загрузки GPU в 0–10%. Причины этому могут быть разные: пользователь или прикладной пакет не умеет эффективно работать с GPU; гра-

фические устройства используются для вычисления небольшого фрагмента программы, а все остальное время простаивают; произошел сбой в работе системы мониторинга, которая показала активность работы с ГПУ, хотя ее на самом деле не было. В любом случае администратору стоит более детально изучить подобные случаи, чтобы разобраться в причинах подобного поведения. Следует отметить, что при этом большинство остальных задач показывает высокую эффективность использования ГПУ (более 50%), что, безусловно, является хорошим показателем.

Здесь приведены лишь некоторые примеры “снимков”, которые лягут в основу разрабатываемой модели, определяющей качество работы суперкомпьютерных систем. Реализация подобной функциональности, на наш взгляд, позволит существенно облегчить сопровождение суперкомпьютерных систем в части всестороннего анализа качества их использования.

4. Подходы к реализации программного инструмента на практике. На данный момент разработка предлагаемого инструмента только начата, поэтому в этом разделе технические особенности его реализации описываются лишь в общих чертах.

Программная реализация инструмента для всестороннего анализа будет представлять собой клиент-серверное приложение. Основная сложность в реализации будет, вероятно, заключаться в интеграции и преобразовании собранной информации — в силу гетерогенности и больших масштабов собираемых данных. Потребуется подключение множества источников данных — менеджера ресурсов, системы мониторинга для получения информации о вычислительных узлах, системного ПО (например, инструмента XALT для определения используемых прикладных пакетов), средств анализа производительности, специализированных систем мониторинга для получения данных о состоянии компонентов инфраструктуры СК. В связи с этим одним из главных принципов при разработке инструмента является модульность, что обеспечит возможность простого добавления или изменения источников входных данных, форматов представления этих данных или методов проведения анализа.

Хранение и доступ к данной информации планируется выполнять посредством СУБД PostgreSQL. Клиентская часть (для получения результатов работы инструмента) будет реализована в виде отдельного web-сайта, доступного в сети Интернет. В рамках этого сайта будет реализован конфигуратор, который позволит выбирать типы входных данных, форматы представления, фильтры по объему выдаваемой информации. Для визуализации полученных результатов в браузере планируется использовать существующие средства визуализации, такие как, например, Grafana [14] или D3.js [15].

Использование подобных технологий позволит сделать разрабатываемое решение переносимым. Однако основная сложность для достижения переносимости будет заключаться в правильной реализации методов интеграции данных. Для решения этого вопроса планируется реализовать универсальный API (Application Programming Interface), который позволит в едином удобном формате получать все необходимые данные. При переносе данного решения на другую систему можно будет легко модифицировать параметры конфигурации для получения информации из новых источников согласно предложенному API.

Однако главным принципом, учитываемым при разработке инструмента для всестороннего анализа качества работы СКЦ, следует выделить полноту предоставляемой информации и удобство ее представления. Именно для этих целей ведется разработка модели, которая позволит объединить все доступные данные и реализовать их комплексный анализ в рамках единого подхода. Чтобы иметь возможность быстро проанализировать определенный аспект работы СКЦ, предполагается реализовать набор готовых отчетов (а также средство для генерации новых отчетов) о состоянии выбранных программных и аппаратных компонентов суперкомпьютера или вычислительной системы в целом, которые будут описывать самые важные стороны функционирования суперкомпьютеров.

Выполнение указанных требований позволит легко подстраивать работу инструмента под текущие нужды администраторов суперкомпьютера, а также переносить решение на другие вычислительные системы.

5. Заключение. В настоящей статье приведено описание методов и подходов для проведения всестороннего анализа качества работы больших суперкомпьютерных комплексов. Принципиальным отличием разрабатываемого решения от существующих аналогов является осуществление всестороннего анализа, что позволит получать единое и полное представление о работе суперкомпьютерного комплекса. В рамках создаваемого инструмента будут объединены данные мониторинга о поведении выполняемых задач, информация о наличии проблем с эффективностью в задачах, полученная с помощью внешних средства анализа, сведения о качестве работы различных компонентов инфраструктуры суперкомпьютера и т.д.

Одной из главных задач при решении этого вопроса является определение структуры и способов представления данных по каждому рассматриваемому компоненту СК. Для этих целей ведется разработка общей модели, которая позволит определять качество работы СКЦ. Определен стартовый набор

“снимков” — важных сценариев использования полученных данных о состоянии СК, которые будут формировать основу для данной модели.

В дальнейшем планируется расширить набор “снимков”, требуемых на практике для анализа качества работы суперкомпьютеров. На основе полученных результатов будет разработан набор методов, формирующих общую модель, которые позволят наиболее удобным образом оценивать различные аспекты работы СКЦ. Данная функциональность будет реализована на практике и апробирована на больших суперкомпьютерных системах петафлопсного уровня производительности, в частности на суперкомпьютере Ломоносов-2.

Результаты, описанные в разделах 3 и 4, получены при финансовой поддержке гранта Президента РФ (МК–2330.2019.9). Остальные результаты получены при финансовой поддержке РФФИ, грант № 17–07–00664. Работа выполнена с использованием оборудования Центра коллективного пользования сверхвысокопроизводительными вычислительными ресурсами МГУ им. М. В. Ломоносова.

СПИСОК ЛИТЕРАТУРЫ

1. Voevodin V., Voevodin V. Efficiency of exascale supercomputer centers and supercomputing education // High Performance Computer Applications. Vol. 595. Cham: Springer, 2016. 14–23.
2. Guan Q., Fu S. Adaptive anomaly identification by exploring metric subspace in cloud computing infrastructures // 2013 IEEE 32nd International Symposium on Reliable Distributed Systems. Washington, DC: IEEE Press, 2013. 205–214.
3. Fu S. Performance metric selection for autonomic anomaly detection on cloud computing systems // 2011 IEEE Global Telecommunications Conference. New York: IEEE Press, 2011. doi 10.1109/GLOCOM.2011.6134532.
4. Ibidunmoye O., Hernández-Rodríguez F., Elmroth E. Performance anomaly detection and bottleneck identification // ACM Computing Surveys. 2015. **48**, N 1. doi 10.1145/2791120.
5. Tuncer O., Ates E., Zhang Y., et al. Diagnosing performance variations in HPC applications using machine learning // Lecture Notes in Computer Science. Vol. 10266. Cham: Springer, 2017. 355–373.
6. Lan Z., Zheng Z., Li Y. Toward automated anomaly identification in large-scale systems // IEEE Transactions on Parallel and Distributed Systems. 2010. **21**, N 2. 174–187.
7. Jones M.D., White J.P., Innus M., et al. Workload Analysis of Blue Waters. <https://arxiv.org/abs/1703.00924>.
8. Abraham M.J., Murtola T., Schulz R., et al. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers // SoftwareX. 2015. **1–2**. 19–25.
9. Agrawal K., Fahey M.R., McLay R., James D. User environment tracking and problem detection with XALT // Proceedings of the First International Workshop on HPC User Support Tools. Piscataway: IEEE Press, 2014. 32–40.
10. Shaykhislamov D., Voevodin V. An approach for dynamic detection of inefficient supercomputer applications // Procedia Computer Science. 2018. **136**. 35–43.
11. Shvets P., Voevodin V., Zhumatiy S. Primary automatic analysis of the entire flow of supercomputer applications // Proceedings of the 4th Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists. CEUR Workshop Proceedings. Vol. 2281. 2018. 20–32.
12. Simakov N.A., White J.P., DeLeon R.L., et al. A workload analysis of NSF’s innovative HPC resources using XDMoD. <https://arxiv.org/abs/1801.04306>.
13. Asanović K., Bodik R., Catanzaro B.C., et al. The landscape of parallel computing research: A view from Berkeley. Technical Report UCB/EECS-2006-183. Berkeley: University of California, 2006.
14. Grafana: The Open Platform for Beautiful Analytics and Monitoring. <https://grafana.com/>.
15. D3.js: Data-Driven Documents. <https://d3js.org>.

Поступила в редакцию
3.06.2019

A Comprehensive Analysis of Performance Quality of Large Supercomputer Complexes

V. V. Voevodin¹

¹ Lomonosov Moscow State University, Research Computing Center; Leninskie Gory, Moscow, 119991, Russia; Ph.D., Senior Scientist, e-mail: vadim@parallel.ru

Received June 3, 2019

Abstract: Currently, the problem of low performance of supercomputer complexes is largely due to the fact that administrators of such complexes cannot always timely detect and eliminate the root causes of reduced

efficiency. This largely concerns not the equipment failure (such cases can usually be detected using monitoring systems), but an implicit performance decrease of certain supercomputer components, provided that they seems to continue working correctly. Such a situation arises because there are no sufficiently flexible and convenient software tools for prompt and comprehensive analysis of all the performance quality characteristics of computer systems at the moment. The existing solutions either allow analyzing only a small part of such characteristics or are made as non-universal solutions that satisfy only a small set of specific needs provided by administrators of a particular system. This paper describes a systematic approach to solving this issue, which will allow one to perform a comprehensive analysis of various aspects of supercomputer functioning, primarily related to the execution of supercomputer applications. A software tool developed on the basis of this approach will collect, within a single model, all the most important data on the properties and quality of jobs running on the supercomputer — data on their execution performance, size and duration, presence of specific or abnormal behavior scenarios, the usage of application packages and libraries, etc. Using flexible aggregation capabilities, the required level of detail will be specified - individual users, projects, application packages, subject areas, supercomputer partitions, time ranges, etc. This will allow one to create hundreds and thousands of different views for analyzing the state of the supercomputer, which will help administrators to choose the most suitable option for them.

Keywords: supercomputer, parallel computing, supercomputer applications, performance, efficiency analysis, monitoring data.

References

1. V. Voevodin and V. Voevodin, "Efficiency of Exascale Supercomputer Centers and Supercomputing Education," in *High Performance Computer Applications* (Springer, Cham, 2016), Vol. 595, pp. 14–23.
2. Q. Guan and S. Fu, "Adaptive Anomaly Identification by Exploring Metric Subspace in Cloud Computing Infrastructures," in *Proc. IEEE 32nd Int. Symp. on Reliable Distributed Systems, Braga, Portugal, September 30–October 3, 2013* (IEEE Press, Washington, DC, 2013), pp. 205–214.
3. S. Fu, "Performance Metric Selection for Autonomic Anomaly Detection on Cloud Computing Systems," in *Proc. IEEE Global Telecommunications Conf., Kathmandu, Nepal, December 5–9, 2011* (IEEE Press, New York, 2011), doi 10.1109/GLOCOM.2011.6134532
4. O. Ibidunmoye, F. Hernández-Rodríguez, and E. Elmroth, "Performance Anomaly Detection and Bottleneck Identification," *ACM Comput. Surv.* **48** (2015). doi 10.1145/2791120
5. O. Tuncer, E. Ates, Y. Zhang, et al., "Diagnosing Performance Variations in HPC Applications Using Machine Learning," in *Lecture Notes in Computer Science* (Springer, Cham, 2017), Vol. 10266, pp. 355–373.
6. Z. Lan, Z. Zheng, and Y. Li, "Toward Automated Anomaly Identification in Large-Scale Systems," *IEEE Trans. Parallel Distrib. Syst.* **21** (2), 174–187 (2010).
7. M. D. Jones, J. P. White, M. Innus, et al., "Workload Analysis of Blue Waters," arXiv preprint: 1703.00924v1 [cs.DC] (Cornell Univ. Library, Ithaca, 2017), available at <https://arxiv.org/abs/1703.00924>
8. M. J. Abraham, T. Murtola, R. Schulz, et al., "GROMACS: High Performance Molecular Simulations through Multi-Level Parallelism from Laptops to Supercomputers," *SoftwareX* **1–2**, 19–25 (2015).
9. K. Agrawal, M. R. Fahey, R. McLay, and D. James, "User Environment Tracking and Problem Detection with XALT," in *Proc. First Int. Workshop on HPC User Support Tools, New Orleans, USA, November 21–21, 2014* (IEEE Press, Piscataway, 2014), pp. 32–40.
10. D. Shaykhislamov and V. Voevodin, "An Approach for Dynamic Detection of Inefficient Supercomputer Applications," *Procedia Comput. Sci.* **136**, 35–43 (2018).
11. P. Shvets, V. Voevodin, and S. Zhumatiy, "Primary Automatic Analysis of the Entire Flow of Supercomputer Applications," in *Proc. 4th Ural Workshop on Parallel, Distributed, and Cloud Computing for Young Scientists, Yekaterinburg, Russia, November 15, 2018*. CEUR Workshop Proc. Vol. 2281, 20–32 (2018).
12. N. A. Simakov, J. P. White, R. L. DeLeon, et al., "A Workload Analysis of NSF's Innovative HPC Resources Using XDMoD," arXiv preprint: 1801.04306v1 [cs.DC] (Cornell Univ. Library, Ithaca, 2018), available at <https://arxiv.org/abs/1801.04306>
13. K. Asanović, R. Bodik, B. C. Catanzaro, et al., *The Landscape of Parallel Computing Research: A View from Berkeley*, Report UCB/EECS–2006–183 (Univ. of California, Berkeley, 2006).
14. Grafana: The Open Platform for Beautiful Analytics and Monitoring. <https://grafana.com>. Cited May 28, 2019.
15. D3.js: Data-Driven Documents. <https://d3js.org>. Cited May 28, 2019.