Облачный сервис SciMQ по обработке потоков экспериментальных данных на суперкомпьютере*

В.А. Щапов[1,2], Г.Ф. Масич[1,2]

Институт механики сплошных сред УрО РАН[1], Пермский национальный исследовательский политехнический университет[2]

В работе представлен прототип облачного сервиса для обработки интенсивных потоков данных в распределенной вычислительной среде ИМСС УрО РАН. Работа направлена на создание сервиса, позволяющего пользователям экспериментальных установок УрО РАН эффективно использовать существующие суперкомпьютерные и телекоммуникационные ресурсы для обработки интенсивных потоков данных. Отличительные особенности – использование модели прямой передачи данных (память-память), лямбда-каналов оптической WDM сети между источниками данных и суперкомпьютерами, веб-интерфейса для централизованного запроса ресурсов пользователями и применение технологий виртуализации для запуска унифицированных расчетных задач.

Ключевые слова: Суперкомпьютер, Облачные вычисления, Потоковая обработка данных, DWDM.

1. Введение

В настоящее время активно развивается новая парадигма предоставления услуг информационных технологий – облачные вычисления. Ее суть заключается в обеспечении повсеместного сетевого доступа к пулу ресурсов (например, сетям передачи данных, серверам, хранилищам данных, суперкомпьютерам, приложениям и сервисам, и т.д.), которые могут быть предоставлены или освобождены по требованию конечного пользователя [1].

В работе представлен прототип облачного сервиса для обработки интенсивных потоков данных в распределенной вычислительной среде ИМСС УрО РАН, построенный на базе разработанных нами технологий и программной платформы по обработке интенсивных потоков данных [2-3]. Облачная модель предоставления доступа позволит автоматизировать исследователям доступ к коммуникационной и вычислительной среде, а также исследовать способы обслуживания интенсивных потоков данных в вычислительных сетях.

2. Вычислительные ресурсы

2.1 Применимость публичных провайдеров облачных инфраструктур для обработки потоков данных

Современные научные экспериментальные установки генерируют большие объемы данных, нуждающихся в обработке в реальном времени. Одна из таких задач, решающаяся в ИМСС УрО РАН, – это обработка экспериментальных данных, получаемых по методу PIV (Particle Image Velocimetry) [4] – оптическому методу измерения полей скорости жидкости или газа в выбранном сечении потока. Интенсивность порождаемого потока данных зависит от числа, разрешения и частоты работы камер и может достигать нескольких гигабит в секунду.

Одними из наиболее известных провайдеров облачных инфраструктур являются сервисы Amazon Web Services, Microsoft Azure, Google Compute Engine и другие. Данные сервисы предоставляют вычислительные ресурсы в виде виртуальных машин, доступ к различному промежуточному программному обеспечению в виде сервисов (например, система хранения объектов Amazon S3, облачные базы данных Amazon RDS или системы очередей Amazon SQS и т.д.).

* Работа выполнена при поддержке гранта РФФИ № 16-37-60043.

Все это позволяет использовать данные сервисы для расширения собственных вычислительных возможностей.

Однако в публикациях отмечается проблема низкой скорости обмена данными с инфраструктурой основных облачных провайдеров через сеть интернет[5]. Еще одним фактором является отсутствие доступной магистральной сетевой инфраструктуры, соединяющей экспериментальные установки и центры обработки данных поставщиков облачных услуг, которые могут находиться на значительном удалении или в других странах. В совокупности это делает затрудненным или даже невозможным использование внешнего вычислительного ресурса публичных облачных провайдеров для задач, связанных с обработкой потоков экспериментальных данных или задач, связанных с обработкой очень большого количества данных.

В то же время у научных организаций существуют собственные вычислительные и коммуникационные ресурсы, которые могут предоставлять вычислительное и коммуникационное обеспечение для таких задач. В связи с этим мы считаем перспективным создание облачного сервиса для обработки потоков экспериментальных данных, работающего по модели PaaS (Platform as a Service), поверх существующих вычислительных и коммуникационных ресурсов УрО РАН. В рамках платформы планируется реализовать доступ к разработанному промежуточному программному обеспечению и доступ к вычислительным ресурсам путем запуска предустановленного программного обеспечения или пользовательского программного обеспечения, предоставленного в виде готовых к запуску контейнеров.

2.2 Вычислительная платформа УрО РАН

Распределенная вычислительная среда УрО РАН [6] базируется в двух центрах обработки данных (ЦОД): в ИМСС УрО РАН (Пермь) и ИММ УрО РАН (Екатеринбург) (рис. 1).

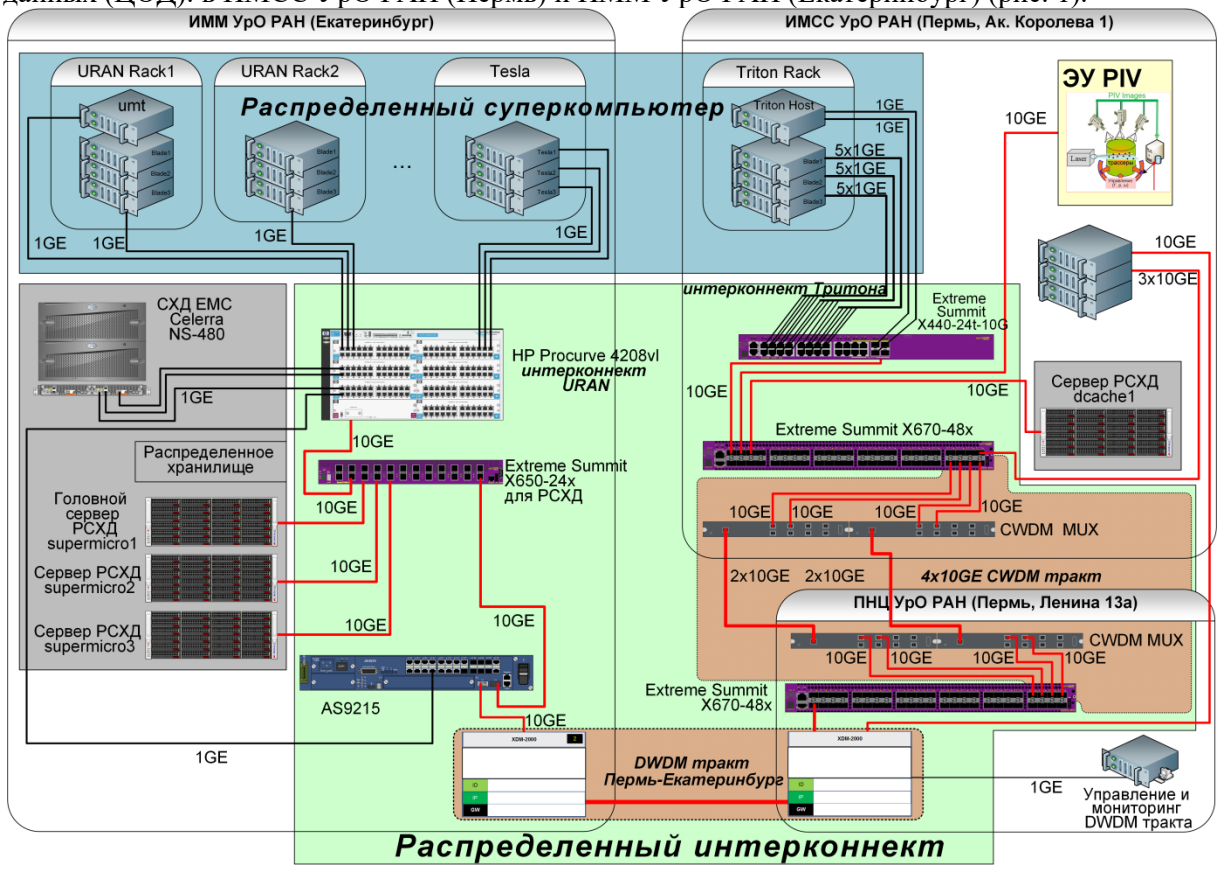

Рис. 1. Распределенная вычислительная среда УрО РАН

В ЦОД ИММ располагаются суперкомпьютер «Уран», пиковой производительностью 225.85 Тфлопс, система хранения данных EMC Celerra NS-480 и три сервера распределенной системы хранения данных (РСХД) производства компании Supermicro [7].

В ЦОД ИМСС располагается вычислительный кластер «Тритон», пиковой производительностью 4,5 Тфлопс, один сервер распределенной системы хранения данных и три сервера приложений HP ProLiant DL360p Gen8 (2x Intel Xeon CPU E5-2660, 2.20 ГГц; RAM 128 Гб).

Коммуникационная среда распределенной системы, показанной на рис. 1, сформирована посредством Ethernet-коммутаторов, соединенных каналами связи DWDM тракта Пермь-Екатеринбург (30 Гбит/с) научно-образовательной сети GIGA URAL [2]. CWDM тракт ИМСС-ПНЦ соединяет на скорости 40 Гбит/с вычислительные ресурсы ИМСС с местом окончания DWDM тракта в Перми. Для большей гибкости проводимых исследований на площадках установлены L2 коммутаторы ECI AS9215 и L3-коммутаторы Extreme Summit X670-48x, образующие в совокупности с Ethernet портами DWDM мультиплексоров гарантированные и негарантированные каналы связи 1-10 Гбит/с.

На площадках присутствия в городах Пермь и Екатеринбург существует возможность подключения выделенных каналов связи до экспериментальных установок других потенциальных пользователей облачного сервиса.

3. Архитектура сервиса SciMQ

На рис. 2 приведена схема архитектуры облачного сервиса SciMQ обработки потоков экспериментальных данных на суперкомпьютере.

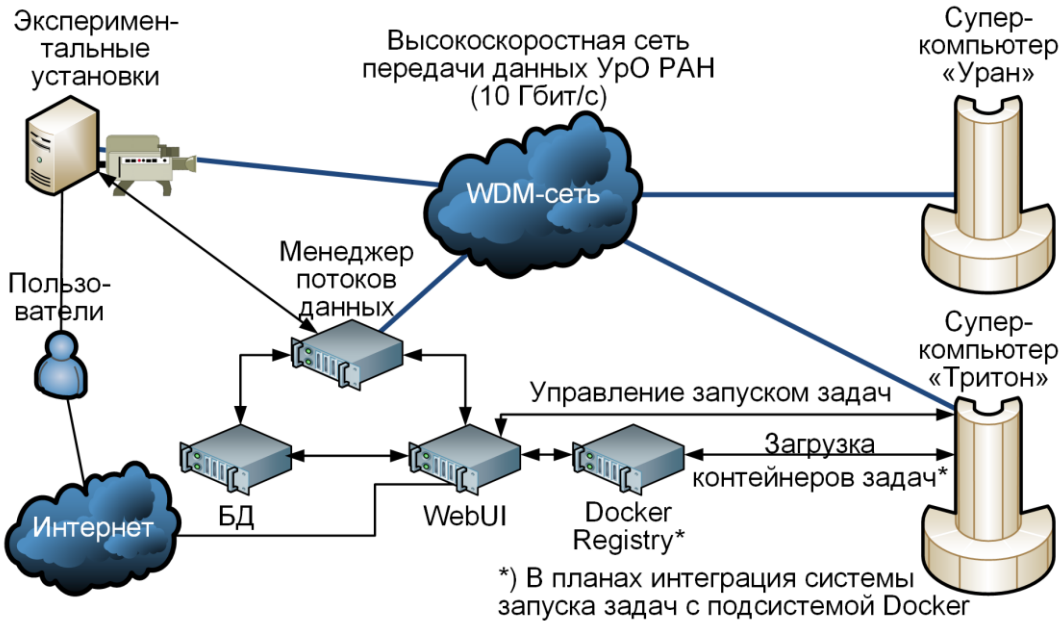

Рис. 2. Архитектура облачного сервиса SciMQ

Облачный сервис SciMQ состоит из следующих основных компонентов:

1. Менеджер потоков данных;
2. База метаданных (БД);
3. WebUI – веб-интерфейс к системе;
4. Docker Registry.

Менеджер потоков данных – это высокопроизводительный сервер очередей SciMQ, разработанный авторами и рассчитанный на эффективную работу с большими сообщениями, размер которых может достигать десятков мегабайт [2]. Менеджер отвечает за управление передаваемыми пользовательскими сообщениями (измерениями отдельных экспериментов) в очередях, включая их получение от экспериментальных установок, промежуточное хранение, распределение по запросам вычислительных узлов суперкомпьютеров и/или по запросам от других внешних систем.

Отличительные особенности предлагаемого решения:

- отказ от промежуточного хранения передаваемых данных в хранилищах;
- прямой ввод данных в вычислительные узлы;

- параллелизм L4-соединений между оконечными системами.

Параллелизм L4-соединений позволяет решить проблему эффективного использования надежных транспортных протоколов, работающих по протяженным скоростным линиям связи. Этим обеспечивается возможность использования в качестве транспортных протоколов (L4 OSI RM) стандартного протокола TCP или протокола UDT, который реализует потоковый протокол передачи данных с гарантией доставки поверх UDP пакетов с собственным алгоритмом управления потоком.

На текущий момент менеджер потоков данных может работать только на одном физическом сервере, однако, планируется реализация механизмов кластеризации для повышения его отказоустойчивости.

В сервер очередей встроен механизм обеспечения гарантии обработки данных, основанный на подтверждениях обработки от получателей данных, и таймерах неактивности, позволяющих поставить ранее отправленные, но необработанные сообщения обратно в очередь. Таким образом, система реализует паттерн доставки данных, при котором каждое пользовательское сообщение будет обработано хотя бы один раз. Помимо этого реализуется как хранение пользовательских данных только в оперативной памяти, так и с записью всех действий на диск. Реализован механизм переноса сообщений на диск при нехватке оперативной памяти. В случае переполнения диска, вновь поступающие данные будут контролируемо удаляться с записью информации об этих событиях в журнал работы сервера. Наличие механизма принудительной потери данных необходимо на случай форс-мажорных ситуаций, когда обработка данных полностью прекращается (например, из-за аварии на сетевой магистрали), а суммарный объем потока данных превышает возможности буферизации хранения на сервере очередей.

База метаданных (БД) – это база данных PostgreSQL, в которой хранится служебная информация, необходимая для работы сервиса, и статистика загрузки сервера очередей менеджера потоков данных. Часть схемы базы данных, отвечающая за хранение статистики загрузки сервиса, приведена на рис. 3.

Основные данные статистики записываются в таблицу stat.history, которая партицирована по метке времени добавляемой записи. Партицирование реализовано с использованием механизма наследования таблиц PostgreSQL и триггеров, которые определяют таблицу, в которую необходимо проводить запись и, при необходимости, создают новые таблицы. Сбор и агрегирование (усреднение за интервалы времени) статистики выполняется отдельным демоном в составе менеджера потоков данных. По умолчанию сбор метрик происходит раз в 10 секунд. Метрики старше месяца агрегируются путем усреднения пятиминутных интервалов.

Служебная метаинформация состоит из данных учетных записей пользователей системы, их прав доступа и лимитов, а также из токенов авторизации, которые используют клиентские приложения для авторизации в менеджере потоков данных.

WebUI – веб-интерфейс к системе. Текущая реализация веб-интерфейса менеджера очередей SciMQ позволяет полностью управлять им и просматривать архив статистики загрузки системы. В рамках сервиса планируется реализовать интеграцию с суперкомпьютером «Тритон» для возможности запуска задач обработки экспериментальных данных из единого веб-интерфейса.

Создание единого интерфейса запуска расчетных задач требует унифицированного механизма доставки вычислительных приложений на вычислительные узлы суперкомпьютера. На суперкомпьютере «Тритон» была успешно апробирована технология контейнерной виртуализации Docker для запуска одноузловых расчетных задач [8], а сейчас реализуется проект по применению Docker для запуска многоузловых MPI-задач с использованием InfiniBand и Ethernet интерконнектов для межузловых обменов данных.

Доставка приложений обработки данных на вычислительные узлы в виде Docker-контейнеров позволит отделить расчетные приложения от имеющегося на суперкомпьютере программного окружения: системы запуска задач, версий библиотек, компиляторов и MPI с сохранением всех преимуществ высокоскоростного интерконнекта и минимальными накладными расходами на работу приложений в контейнерах[9]. Планируемая поддержка собственного Docker Registry позволит централизовано управлять хранилищем образов задач, как подготавливаемых пользователями, так и заранее созданных администраторами суперкомпьютеров.

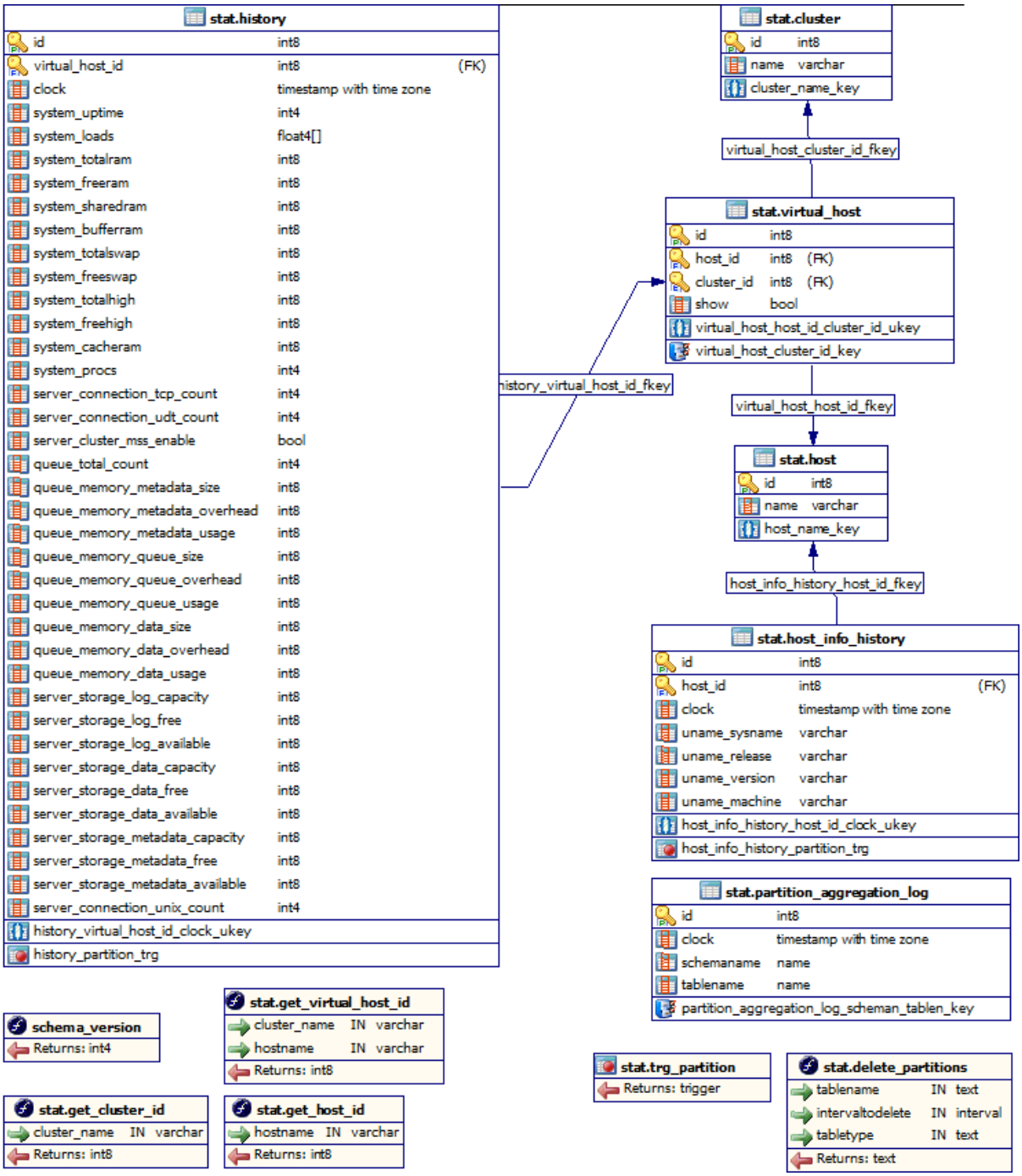

Рис. 3. Схема БД хранения статистики сервера очередей SciMQ

4. Подход к определению требуемых характеристик оборудования для запуска компонент сервиса

Наиболее требовательным к аппаратным ресурсам компонентом системы является менеджер потоков данных, который пропускает через себя весь поток данных обрабатываемых экспериментов. Наиболее важными параметрами для него являются объем доступной оперативной памяти для промежуточной буферизации данных и пропускная способность подключений к WDM-сети. Опыт эксплуатации показывает, что производительность CPU является менее важным параметром, чем вышеперечисленные.

На текущем этапе нами было принято решение применить экспериментально-статистический метод по определению системных требований для различных условий путем

обобщения данных о потреблении ресурсов, собранных по результатам проведения экспериментов с модельными потоками данных, эмулирующие некоторые из реальных физических экспериментов. Сейчас проводится этап сбора данных, в котором участвуют суперкомпьютер «Тритон», генератор модельного потока сообщений, позволяющий использовать различные стратегии для определения интенсивности потока и экспериментальная инсталляция менеджера потоков данных и БД статистики.

5. Заключение

Переход к облачной модели предоставления доступа к инфраструктуре обработки интенсивных потоков данных на суперкомпьютере позволит автоматизировать исследователям доступ к существующей коммуникационной и вычислительной среде и разработанным технологиям обработки интенсивных потоков экспериментальных данных, что приведет к повышению эффективности различных экспериментальных исследований.

Литература

1. The NIST Definition of Cloud Computing URL: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf (дата обращения: 22.05.2016).
2. V. Shchapov, G. Masich, A. Masich, Platform for Parallel Processing of Intense Experimental Data Flow on Remote Supercomputers // Procedia Computer Science. Vol. 66. 2015. P. 515-524. ISSN 1877-0509. doi:10.1016/j.procs.2015.11.058.
3. G. Masich, V. Shchapov The software platform of transmission of intense data streams on remote supercomputers // CEUR Workshop Proceedings (Proceedings of the 1st Russian Conference on Supercomputing (RuSCDays 2015), Moscow, Russia, September 28-29, 2015). Vol. 1482. 2015. P. 720-731.
4. Степанов Р.А., Масич А.Г., Масич Г.Ф. Инициативный проект «Распределенный PIV» // Научный сервис в сети Интернет: масштабируемость, параллельность, эффективность: труды Всероссийской суперкомпьютерной конференции / М. Изд-во МГУ, 2009. – С. 360-363.
5. M. G. McGrath, P. Raycroft, P. R. Brenner Intercloud Networks Performance Analysis // IEEE International Conference on Cloud Engineering (IC2E), Tempe, AZ. 2015. P. 487-492. doi:10.1109/IC2E.2015.85.
6. Г.Ф. Масич, А.Г. Масич, В.А. Щапов, С.Р. Латыпов, А.В. Созыкин, Е.Ю.Куклин Архитектура распределенной вычислительной среды УрО РАН // Материалы XIV Международной конференции «Высокопроизводительные параллельные вычисления на кластерных системах (HPC 2014)». Пермь: Издательство ПНИПУ, 2014. С. 281-287.
7. Kuklin E.Yu., Sozykin A.V., Bersenev A.Yu., Masich G.F. Distributed dCache-based storage system of UB RAS // Computer Research and Modeling, 2015, vol. 7, no. 3, pp. 559-563 ISSN: 2076-7633 (Print), 2077-6853 (Online), URL: http://crm-en.ics.org.ru/journal/author/2719/.
8. V. Shchapov, D. Chugunov Using of container virtualization to run tasks on a distributed supercomputer // CEUR Workshop Proceedings (Proceedings of the 1st Russian Conference on Supercomputing (RuSCDays 2015), Moscow, Russia, September 28-29, 2015). Vol. 1482. 2015. P. 601.
9. W. Felter, A. Ferreira, R. Rajamony and J. Rubio An updated performance comparison of virtual machines and Linux containers // IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Philadelphia, PA, 2015, pp. 171-172.

Cloud service SciMQ for stream processing of experimental data on a supercomputer

V.A. Shchapov[1,2], G.F. Masich[1,2]

Institute of Continuous Media Mechanics of the Ural Branch of Russian Academy of Science[1], Perm National Research Polytechnic University[2]

We present a cloud service prototype designed to process intensive data streams in the distributed computing environment of ICMM UB RAS. Our goal is to create a server allowing the users of experimental setups located at UB RAS facilities to work more efficiently with existing supercomputing and telecommunication resources for intensive data stream processing. The server has the advantage of using a direct data transfer model (memory-memory), WDM lambda-channels between data sources and supercomputers, a web-interface enabling users to access their resources and visualization techniques for invoking unified computational problems.

Keywords: Supercomputer, Cloud, Data stream processing, DWDM.

References

1. The NIST Definition of Cloud Computing URL: http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf (accessed: 22.05.2016).
2. V. Shchapov, G. Masich, A. Masich, Platform for Parallel Processing of Intense Experimental Data Flow on Remote Supercomputers // Procedia Computer Science. Vol. 66. 2015. P. 515-524. ISSN 1877-0509. doi:10.1016/j.procs.2015.11.058.
3. G. Masich, V. Shchapov The software platform of transmission of intense data streams on remote supercomputers // CEUR Workshop Proceedings (Proceedings of the 1st Russian Conference on Supercomputing (RuSCDays 2015), Moscow, Russia, September 28-29, 2015). Vol. 1482. 2015. P. 720-731.
4. Stepanov R.A., Masich A.G., Masich G.F. Initsiativnyy proekt “Raspredelennyy PIV” [Initiative project “Distributed PIV”] // Nauchnyy servis v seti Internet: masshtabiruemost', parallel'nost', effektivnost': trudy Vserossiyskoy superkomp'yuternoy konferentsii [Scientific service on the Internet: scalability, parallelism, efficiency: Proceedings of the Russian Supercomputer Conference], Moscow. Publishing of the MSU, 2009. – P. 360-363.
5. M. G. McGrath, P. Raycroft, P. R. Brenner Intercloud Networks Performance Analysis // IEEE International Conference on Cloud Engineering (IC2E), Tempe, AZ. 2015. P. 487-492. doi:10.1109/IC2E.2015.85.
6. G.F. Masich, A.G. Masich, V.A. Shchapov, S.R. Latypov, A.V. Sozykin, E.Yu. Kuklin Arkhitektura raspredelennoy vychislitel'noy sredy UrO RAN [The architecture of a distributed computing environment UB RAS] // Materialy XIV Mezhdunarodnoy konferentsii “Vysokoproizvoditel'nye parallel'nye vychisleniya na klasternykh sistemakh (HPC 2014)” [Proceedings of the XIV International Conference "High-performance parallel computing on cluster systems (HPC 2014)"]. Perm: Publishing of the PNRPU, 2014. P. 281-287.
7. Kuklin E.Yu., Sozykin A.V., Bersenev A.Yu., Masich G.F. Distributed dCache-based storage system of UB RAS // Computer Research and Modeling, 2015, vol. 7, no. 3, pp. 559-563 ISSN: 2076-7633 (Print), 2077-6853 (Online), URL: http://crm-en.ics.org.ru/journal/author/2719/.
8. V. Shchapov, D. Chugunov Using of container virtualization to run tasks on a distributed supercomputer // CEUR Workshop Proceedings (Proceedings of the 1st Russian Conference on Supercomputing (RuSCDays 2015), Moscow, Russia, September 28-29, 2015). Vol. 1482. 2015. P. 601.

9. W. Felter, A. Ferreira, R. Rajamony and J. Rubio An updated performance comparison of virtual machines and Linux containers // IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS), Philadelphia, PA, 2015, pp. 171-172.