

Copyright © 2014 by Academic Publishing House *Researcher*



Published in the Russian Federation
European Researcher
Has been issued since 2010.
ISSN 2219-8229
E-ISSN 2224-0136
Vol. 76, No. 6-1, pp. 1051-1058, 2014

DOI: 10.13187/issn.2219-8229
www.erjournal.ru



Engineering Sciences

Технические науки

UDC 004.75

A Distributed Computational Infrastructure for Science and Education

¹ Rustam K. Bazarov

² Dilshot D. Akhmedov

³ Raida K. Djaparova

⁴ Dmitriy K. Bazarov

¹ Institute of mathematics, Uzbekistan
Durmon Yuli street 29, Tashkent city, 100125
Junior research scientist
E-mail: rustam.bazarov@gmail.com

² Centre for the development of software and hardware program complexes, Uzbekistan
Durmon Yuli street 25, Tashkent city, 100125
Junior research scientist
E-mail: a.dilshot@mail.ru

³ Tashkent Pediatric Medical Institutes, Uzbekistan
Bogishamol street 229, Tashkent city, 100140
Associate Professor

⁴ Institute of mathematics, Uzbekistan
Durmon Yuli street 29, Tashkent city, 100125
Junior research scientist

Abstract. Researchers have lately been paying increasingly more attention to parallel and distributed algorithms for solving high-dimensionality problems. In this regard, the issue of acquiring or renting computational resources becomes a topical one for employees of scientific and educational institutions. This article examines technology and methods for organizing a distributed computational infrastructure. The author addresses the experience of creating a high-performance system powered by existing clusterization and grid computing technology. The approach examined in the article helps minimize financial costs, aggregate territorially distributed computational resources and ensures a more rational use of available computer equipment, eliminating its downtimes.

Keywords: high-performance system; distributed computational infrastructure; Beowulf cluster; grid computing.

Введение. В настоящее время наблюдается устойчивая тенденция к расширению класса задач, для решения которых необходимо привлечение значительных

вычислительных ресурсов. Данное обстоятельство во многом связано с тем, что в последнее годы в научном сообществе возросла популярность направлений численного моделирования и численного эксперимента.

Численное моделирование представляет собой компромиссный вариант между физическим экспериментом и аналитическим подходом. Оно позволяет изучать явления и процессы, которые слишком сложны для исследования аналитическими методами либо, по различным причинам, не представляются возможными для экспериментального изучения. В то же время, численный эксперимент заметно удешевляет и делает более удобным процесс научного изыскания. Численный эксперимент даёт возможность моделировать в режиме реального времени сложные физико-химические процессы, глобальные и региональные атмосферные процессы, динамику многокомпонентных сред и т.д.

Эффективное применение различных численных методов, например: интегрирования, механики сплошных сред, решения систем линейных и дифференциальных уравнений, методов Монте-Карло и т.д., до недавнего времени, было проблематичным. В виду того, что решение задач большой размерности требует значительных вычислительных ресурсов. Следовательно, для учёных и специалистов, сталкивающихся с необходимостью осуществления большого объема вычислений, становится насущным вопрос увеличения скорости их проведения.

Приобретение и совокупная стоимость владения высокопроизводительной вычислительной инфраструктурой, а также специализированным программным обеспечением, как и прежде, остаётся довольно высокой. Для большинства научных организаций и университетов покупка и содержание такой инфраструктуры по объективным причинам недоступно. К тому же, как показывает практика, в руках конечного пользователя высока вероятность ограниченного нерегулярного использования подобной инфраструктуры, что делает дорогостоящее приобретение и содержание крайне нецелесообразным.

Выход из противоречивого положения может быть осуществлён несколькими путями. Сегодня пользователь имеет возможность приобретать вычислительные ресурсы по необходимости в виде услуг, известных как «облачные сервисы». В частности, IaaS (англ. Infrastructure-as-a-Service) или инфраструктура как услуга – один из таких сервисов, обеспечивающий возможность использования вычислительной инфраструктуры для самостоятельного управления ресурсами обработки, хранения, сетями и т.д. Данный вариант, характеризуется рядом преимуществ, например, сравнительно небольшая плата за услугу, использование на постоянной основе или только при необходимости, отсутствие сторонних затрат, обеспечение надёжности со стороны провайдера услуги. Единственным существенным недостатком может быть лишь обстоятельство того, что пользователь в своих действиях ограничивается техническими рамками и другими возможными условиями, установленными провайдером.

Другим способом получить в своё распоряжение высокопроизводительную систему является самостоятельное её конструирование на базе доступных технических и программных средств. Мощности процессоров, объёма оперативной и дисковой памяти современных персональных компьютеров (ПК) вполне достаточно для проведения большинства типов расчётов, а объединение нескольких десятков или сотен обычных компьютеров в единую систему позволяет быстро и эффективно решать задачи большой размерности с помощью параллельных и распределённых алгоритмов. Такой способ стал возможен благодаря удешевлению и доступности ПК с одной стороны, и развитию свободного открытого программного обеспечения – с другой. Это даёт возможность при минимальных затратах на базе имеющегося парка ПК и доступных коммуникационных сетей создавать вычислительные комплексы, сопоставимые по производительности с суперкомпьютерами.

Материалы и методы. Объединение отдельных компьютеров в единую систему с целью организации вычислительного процесса называют вычислительным кластером. Кластер с точки зрения пользователя представляет собой единый унифицированный компьютерный ресурс и является базовой единицей в больших распределённых системах.

Подобную систему, построенную из готовых промышленных компонент, например, ПК и стандартного оборудования Ethernet сетей, не содержащую специфических аппаратных средств и обладающую свойством быстрой и лёгкой подготовки к эксплуатации, принято называть Beowulf кластером [1].

Beowulf – это технология кластеризации, подразумевающая создание мультимьюльтикомпьютерной архитектуры, обычно состоящей из одного серверного узла и одного или более клиентских узлов, объединённых при помощи Ethernet или некоторой другой сети. Серверный узел управляет всем кластером и является файл-сервером для клиентских узлов. Он также является консолью управления и шлюзом доступа к внешней сети. Как правило, клиентские узлы в Beowulf пассивны. Они конфигурируются и управляются серверным узлом и выполняют только его предписания (рис. 1).

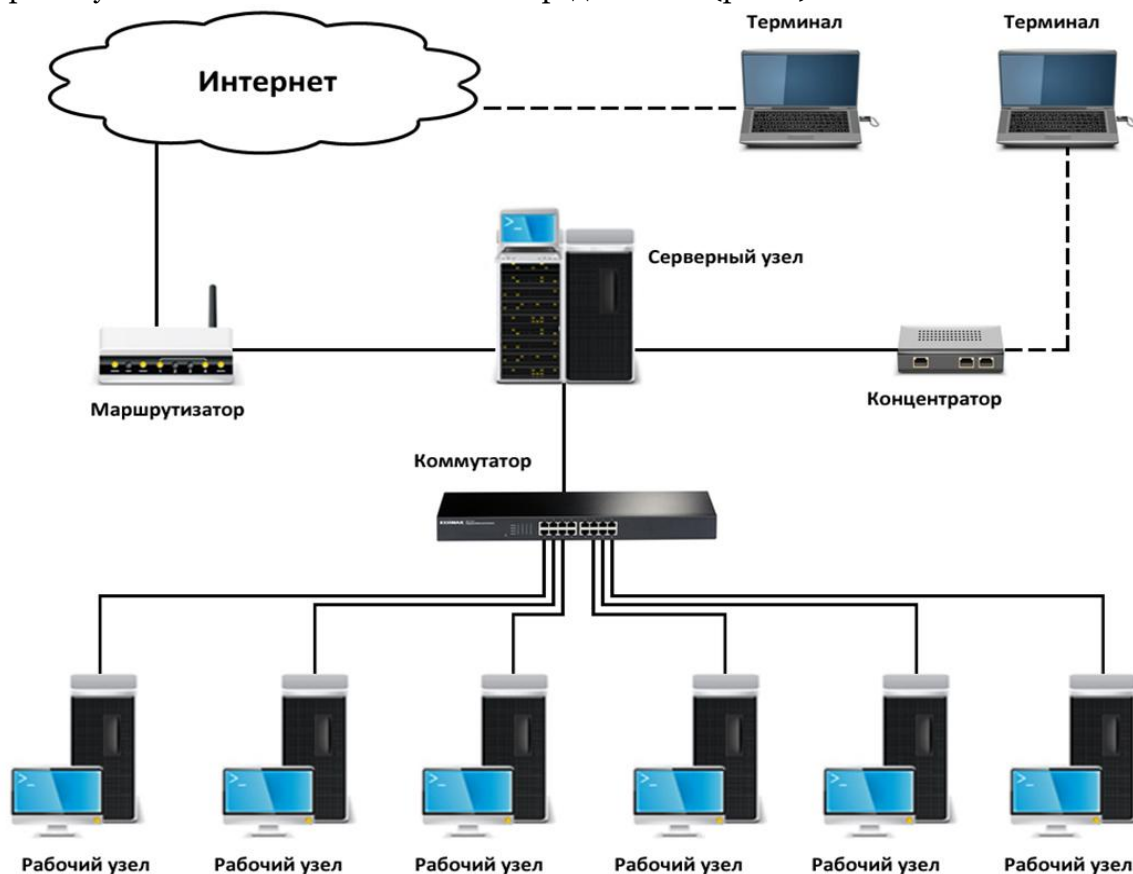


Рис. 1. Типичная схема организации Beowulf кластера

Одно из основных свойств Beowulf состоит в том, что вся система работает как одна машина. В большинстве случаев рабочие узлы не оснащаются клавиатурой и монитором, и доступны только посредством удаленного подключения. Рабочие узлы можно рассматривать как элементы процессор + память, которые вставляются в кластер так же как процессор и модуль памяти вставляются в материнскую плату ПК или сервер-лезвие в блейд-систему.

Кластеры Beowulf успели положительно себя зарекомендовать среди научных организаций и университетов в качестве хорошей платформы для изучения параллельного и распределённого программирования, а также в качестве недорогой производительной вычислительной системы для научных задач.

При всей привлекательности технологии Beowulf, масштаб вычислительной инфраструктуры на её основе территориально ограничен. То есть добавление рабочих узлов, как правило, осуществляется лишь в рамках локальной сети организации, подразделения или даже отдельного здания. Дело не в том, что дальнейшее масштабирование кластера принципиально невозможно. При построении по настоящему распределённой гетерогенной инфраструктуры возникает ряд вопросов, включая технические аспекты, проблемы

безопасности и главные организационные сложности. Для примера можно отметить следующие основные «узкие места»:

- сложность управления территориально распределёнными гетерогенными вычислительными ресурсами, сложность управления заданиями (запуск, перезапуск, планирование, время работы, миграция и т.д.). В этих целях используются программы - диспетчеры ресурсов и заданий, но учитывая, как правило, изолированный характер отдельных Beowulf кластеров, возможностей только типичного диспетчера недостаточно для работы в условиях Интернет соединений;

- проблемы безопасности проистекают из открытой природы технологии Beowulf. Работа с кластером предполагает авторизованный вход в операционную систему центрального узла с ролью и полномочиями системного пользователя, поэтому перед владельцем кластера встаёт вопрос обеспечения мер безопасности при подключении сторонних удалённых пользователей. С другой стороны пользователь удалённого кластера, например, может быть озабочен тем, что его рабочие данные и файлы могут быть доступны другим пользователям, особенно в условиях разделяемой файловой системы;

- организационные вопросы управления всей распределённой вычислительной инфраструктурой, централизованной регистрации и авторизацией пользователей, разработки политик и регламента пользования системой и т.д.

В условиях распределённой вычислительной среды отдельный кластер становится конечной структурной единицей, и возможностей инструментария типичной Beowulf системы становится недостаточно. Соответственно появляется необходимость в дополнительной технологии, которая содержит в себе средства промежуточного уровня для решения отмеченных выше проблем.

Здесь мы подходим к рассмотрению такого явления как Грид-вычисления (Grid Computing). Суть технологии в идеале – нацеленность на объединение всех компьютеров мира в единую систему – «виртуальный суперкомпьютер», который позволит распределять и перераспределять ресурсы между пользователями в соответствии с их запросами. В максимально упрощённом толковании Грид-технологии – это способ создания кластера кластеров в глобальном масштабе. Иначе говоря, то самое средство, необходимое для организации территориально распределённой гетерогенной вычислительной инфраструктуры.

Теоретическое обобщение опыта и развития идей Грид, было сделано американскими учеными Я. Фостером и К. Кессельманом [2]. В своей работе авторы приводят аналогию использования Грид, как использование людьми электричества в единых энергетических сетях. Имея такой суперкомпьютер, пользователь может в любое время и в любом месте запросить столько вычислительных ресурсов, сколько ему требуется для решения своей задачи. По представлению авторов, Грид является некой «надстройкой» над сетью Интернет, предназначенной для распределённых вычислений при решении задач высокой сложности в области науки и технологий.

В реальности ситуация с Грид на сегодняшний день обстоит несколько иначе. Грид-инфраструктура концептуально проектируется таким образом, что некоторое число пользователей может агрегировать и использовать большую часть или вообще все вычислительные ресурсы на время запуска своих задач. Этот процесс происходит в соответствии с очередностью на основе заданных приоритетов. Так как продолжительность и частота занятости имеющихся ресурсов может сильно варьироваться в зависимости от трудоёмкости задач, то задержки в очереди могут быть довольно значительными. Кроме того, само по себе вхождение в реально действующую Грид-инфраструктуру, в любой роли, по сложности, мягко говоря, превосходит процесс подключения к электросети. Трудности по большей части носят организационный характер и призваны отсеивать случайных пользователей или ненадёжных поставщиков услуг. Тем не менее, Грид остаётся открытой системой, которой могут пользоваться все, но для этого они должны иметь вполне конкретные цели, понимание и мотивацию.

Базируясь на такой концепции, Грид отчасти проигрывает в популярности облачным системам. В отличие от Грид, облачные системы поддерживают большое количество пользователей, создавая тем самым иллюзию «бесконечного масштаба». В облачной модели использование ресурсов отдельным пользователем не может вызвать их недостаток, таким

образом, обеспечивая запросы всех остальных без задержек, хотя конечно каждый из пользователей имеет доступ лишь к малой доле от общего пула ресурсов облака.

Облачные системы промышленного масштаба требуют существенных затрат на техническое обеспечение и обычно ориентированы на экономическую отдачу, поэтому чаще всего предстают в виде коммерческих проектов. Поэтому, если говорить о самостоятельном построении доступной вычислительной системы для научных и образовательных целей без явной ориентации на коммерческий эффект, то Грид архитектура и промежуточное программное обеспечение на его основе, являются наиболее выгодным и целесообразным выбором. Средства Грид, позволяют эффективно решать отмеченные выше проблемы, связанные с организацией распределённой гетерогенной инфраструктуры. Существует большой выбор открытых и коммерческих продуктов для создания Грид, таких как BOINC, gLite, UNICORE, Hadoop, Globus Toolkit, Sun Grid Engine, Xgrid и др.

Программный комплекс Globus Toolkit [3, 4], признан ведущими производителями программного обеспечения и специалистами де-факто стандартом в качестве платформы grid-вычислений. В процессе разработки проекта Globus, изначальный акцент сместился с поддержки высокопроизводительных вычислений в сторону сервисов поддержки виртуальных организаций и предоставление возможности приложениям работать с распределёнными разнородными вычислительными ресурсами как с единой виртуальной машиной. Виртуальная организация – это ряд людей и/или организаций, объединённых общими правилами коллективного доступа к определенным вычислительным ресурсам [5, 6]. Таким образом, последние версии системы Globus Toolkit представляют собой набор модулей для построения виртуальной организации распределённых вычислений. Каждый модуль определяет интерфейс, используемый приложениями верхнего уровня, и имеет реализацию для различных сред выполнения. Модули могут быть разбиты на следующие группы (рис. 2):

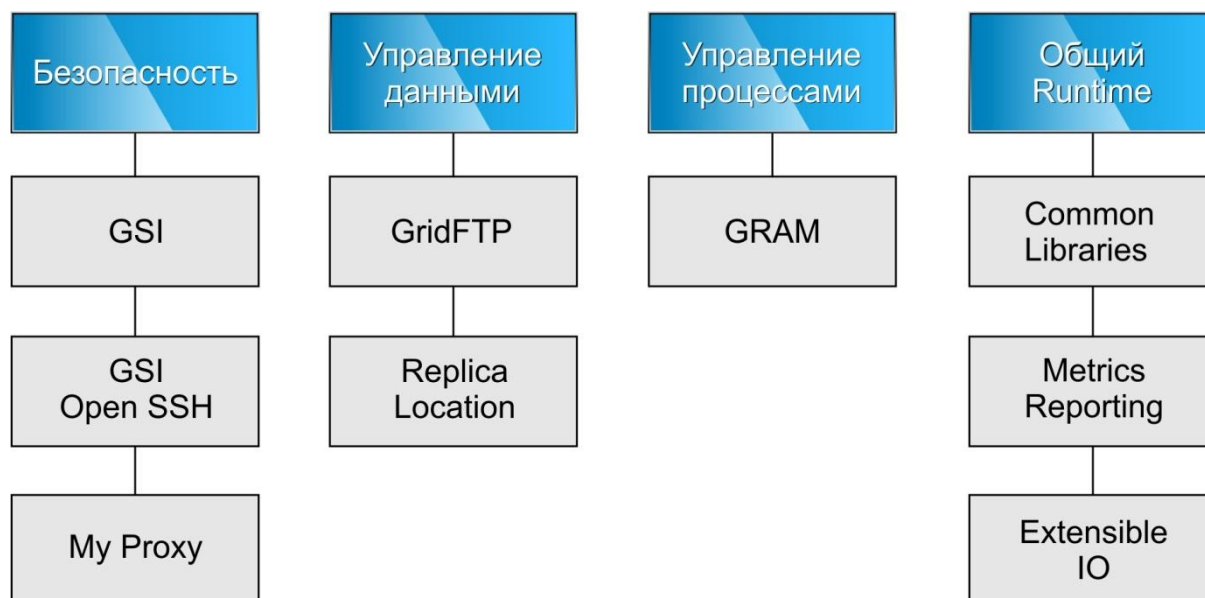


Рис. 2. Архитектура Globus Toolkit

Globus Toolkit обладает набором базовых сервисов, позволяющих ликвидировать, отмеченные выше «узкие места» при организации распределённой вычислительной системы. Так, в частности, в основе безопасности Globus Toolkit лежат принципы авторизации и сертификации (X.509), что позволяет контролировать действия всех пользователей в системе и защищать передаваемые данные. Также компоненты безопасности поддерживают редактирование полномочий пользователей, сохранение информации о членстве в группах доступа, позволяют осуществлять подключение и использование ресурсов различных организаций.

Globus Toolkit предоставляет доступ к вычислительным процессам при помощи компонента GRAM (Grid Resource Allocation and Management). С его помощью пользователь может непосредственно добавлять, контролировать, отслеживать и отменять задания в системе. При этом GRAM, фактически, не является планировщиком заданий – это набор сервисов и клиентов для взаимодействия, посредством стандартного текстового протокола, с рядом локальных планировщиков заданий уровня Beowulf, таких как Condor [7], Torque [8] и прочие.

При построении распределённой вычислительной инфраструктуры Грид и Beowulf не просто дополняют друг друга – они неразрывно связаны. Интерфейсы и протоколы Грид обеспечивают предоставление вычислительных ресурсов, ресурсов хранения и управление ими, а также предоставляют более высокий уровень абстракции для создания пользовательских приложений полностью ориентированных на работу в подобных вычислительных средах.

Обсуждение результатов и выводы. На базе рассмотренных технологий построена экспериментальная распределённая вычислительная система, включающая в себя локальные кластеры трёх учреждений: Центра разработки программных продуктов и аппаратно-программных комплексов при Ташкентском университете информационных технологий, Института математики при Национальном университете Узбекистана, Института ядерной физики АН РУз. Локальные вычислительные кластеры представлены разнородным оборудованием с различными техническими характеристиками, включая персональные компьютеры, rack сервера и виртуальные машины. Конфигурирование кластеров было осуществлено с использованием системы KestrelHPC [9]. На уровне Globus Toolkit каждое из представленных учреждений представляет собой отдельную виртуальную организацию с собственным центром сертификации, выступая, таким образом, одновременно пользователем и провайдером вычислительных ресурсов.

Реализация сетевой инфраструктуры основана на технологии VPN, которая обеспечивает следующие преимущества: обход проблемы трансляции частных IP адресов; дополнительную безопасность и исключение сложной настройки фильтрации трафика; значительное упрощение конфигурации серверов и рабочих узлов. Кроме того VPN позволяет настраивать и использовать механизм внутреннего доменного пространства для прямого и обратного разрешения имён. Доменные имена необходимы как для корректной работы программного обеспечения, так и для удобства пользователей.

Проведённые тесты и вычислительные эксперименты показали эффективность созданной вычислительной среды [10]. Это дало основание проводить, в настоящее время, дальнейшую работу по расширению инфраструктуры путём привлечения новых участников и пользователей из числа академических институтов и университетов Узбекистана.

Резюмируя изложенное выше отметим, что созданная распределённая вычислительная инфраструктура введена в опытную эксплуатацию и успешно используется в качестве полигона при проведении экспериментов, тестировании разрабатываемых параллельных и распределённых алгоритмов. В частности такой подход к организации распределённых вычислений показал свою эффективность при решении задач:

- многокритериальной динамической оптимизации процессов в распределённых нестационарных системах, на примере нахождения оптимальных трафиков в телекоммуникационных сетях [10].

- поиска третичной структуры белка по его первичной аминокислотной последовательности (protein folding problem), которая решалась на основе модели HP-PFP-2 эвристическим методом глобальной оптимизации (параллельным муравьиным алгоритмом).

Использование технологий кластеризации Beowulf и Грид-вычислений в купе с имеющимся парком компьютерной техники и доступными сетевыми коммуникациями обеспечивают существенную экономию материальных ресурсов, экономию времени пользователей на проведение вычислений в процессе проведения научно-исследовательских работ.

Примечания:

1. Beowulf Cluster Computing with Linux / edit. T.L. Sterling ; Massachusetts Institute of Technology. Cambridge : The MIT Press, 2002. 496 p.
2. Foster I., Kesselman C. The Grid 2: Blueprint for a New Computing Infrastructure. – 2d ed. San Francisco : Morgan Kaufmann, 2003. 748 p.
3. Foster I., Kesselman C. Globus: a Metacomputing Infrastructure Toolkit // International Journal of High Performance Computing Applications. 1997. № 2. Vol. 11. pp. 115-128.
4. Foster I. Globus Toolkit Version 4: Software for Service-Oriented Systems // Proceedings of the 2005 IFIP international conference on Network and Parallel Computing. 2005. pp. 2-13.
5. Foster I., Kesselman C., Tuecke The S. Anatomy of the Grid: Enabling Scalable Virtual Organizations // International Journal of High Performance Computing Applications. 2001. Issue 3. Vol. 15. pp. 200-222.
6. Радченко Г.И. Распределенные вычислительные системы. Челябинск: Фотохудожник, 2012. 184 с.
7. Thain D., Tannenbaum T., Livny M. Distributed Computing in Practice: The Condor Experience // Concurrency and Computation : Practice and Experience. 2005. № 2-4. Vol. 17. pp. 323-356.
8. Сбитнев Ю. Менеджер ресурсов Torque [Электронный ресурс] // Linux. Кластер. Практическое руководство по параллельным вычислениям. – URL: <http://kestrelhpc.sourceforge.net> (дата обращения: 13.11.2013).
9. KestrelHPC : Simple Diskless Clustering [Электронный ресурс]. – Режим доступа: <http://kestrelhpc.sourceforge.net> (дата обращения: 09.12.2013).
10. Базаров Р., Ахмедов Д.Д. Параллельный алгоритм оптимизации муравьиной колонии // Перспективы эффективного развития информационных технологий и телекоммуникационных систем : сборник докладов республиканской научно-технической конференции. Ч. 2. Т., 2014. С. 211-213.

References:

1. Beowulf Cluster Computing with Linux / edit. T.L. Sterling ; Massachusetts Institute of Technology. Cambridge : The MIT Press, 2002. 496 p.
2. Foster I., Kesselman C. The Grid 2: Blueprint for a New Computing Infrastructure. 2d ed. San Francisco : Morgan Kaufmann, 2003. 748 p.
3. Foster I., Kesselman C. Globus: a Metacomputing Infrastructure Toolkit // International Journal of High Performance Computing Applications. 1997. № 2. Vol. 11. pp. 115-128.
4. Foster I. Globus Toolkit Version 4: Software for Service-Oriented Systems // Proceedings of the 2005 IFIP international conference on Network and Parallel Computing. 2005. pp. 2-13.
5. Foster I., Kesselman C., Tuecke The S. Anatomy of the Grid: Enabling Scalable Virtual Organizations // International Journal of High Performance Computing Applications. 2001. Issue 3. Vol. 15. pp. 200-222.
6. Radchenko G.I. Raspredelennye vychislitel'nye sistemy. Chelyabinsk: Fotokhudozhnik, 2012. 184 s.
7. Thain D., Tannenbaum T., Livny M. Distributed Computing in Practice: The Condor Experience // Concurrency and Computation : Practice and Experience. 2005. № 2-4. Vol. 17. pp. 323-356.
8. Sbitnev Yu. Menedzher resursov Torque [Elektronnyi resurs] // Linux. Klaster. Prakticheskoe rukovodstvo po parallel'nyim vychisleniyam. – URL: <http://kestrelhpc.sourceforge.net> (data obrashcheniya: 13.11.2013).
9. KestrelHPC : Simple Diskless Clustering [Elektronnyi resurs]. – Rezhim dostupa: <http://kestrelhpc.sourceforge.net> (data obrashcheniya: 09.12.2013).
10. Bazarov R., Akhmedov D.D. Parallel'nyi algoritm optimizatsii murav'inoi kolonii // Perspektivy effektivnogo razvitiya informatsionnykh tekhnologii i telekommunikatsionnykh sistem : sbornik dokladov respublikanskoi nauchno-tekhnicheskoi konferentsii. Ch. 2. T., 2014. S. 211-213.

УДК 004.75

Распределённая вычислительная инфраструктура для науки и образования

¹ Рустам Камильевич Базаров² Дильшот Дильмурадович Ахмедов³ Раида Каримджановна Джапарова⁴ Дмитрий Камильевич Базаров¹ Институт математики, Узбекистан

100125, г. Ташкент, ул. Дурмон йули 29

Младший научный сотрудник

E-mail: rustam.bazarov@gmail.com

² Центр разработки программных продуктов и аппаратно-программных комплексов, Узбекистан

100125, г. Ташкент, ул. Дурмон йули 29

Младший научный сотрудник

E-mail: a.dilshot@mail.ru

³ Ташкентский педиатрический медицинский институт, Узбекистан

100140, г. Ташкент, ул. Богишамол 229

Доцент

⁴ Институт математики, Узбекистан

100125, г. Ташкент, ул. Дурмон йули 29

Младший научный сотрудник

Аннотация. В последнее время исследователями всё большее внимание уделяется параллельным и распределённым алгоритмам решения задач большой размерности. В этой связи, для сотрудников научных и образовательных учреждений становится актуальным вопрос приобретения или аренды вычислительных ресурсов. В статье рассматриваются технологии и методы организации распределённой вычислительной инфраструктуры. Освещается опыт создания высокопроизводительной системы на базе существующих технологий кластеризации и Грид-вычислений. Рассмотренный подход позволяет минимизировать финансовые затраты, агрегировать территориально распределённые вычислительные ресурсы и обеспечивает более рациональное использование имеющейся компьютерной техники, исключая её простои.

Ключевые слова: высокопроизводительная система; распределённая вычислительная инфраструктура; кластер Beowulf; Грид-вычисления.